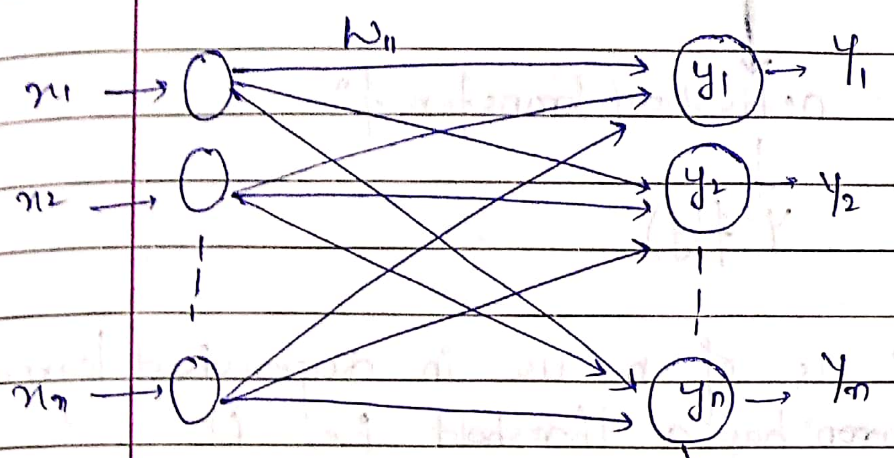
• ARTIFICIAL NEURAL NETWORK (ANN)



(i/p layer) (o/p layer)

net i/p and further processing is done to get o/p y

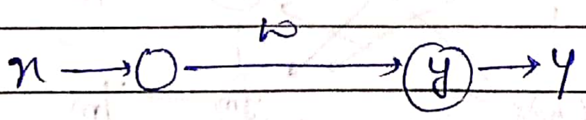
Single layer feed forward artificial neural network

one stage of transfer is there only

∵ feeding data in forward dirn only

✓ All connections are weighted.

w_{11} represent connection from x_1 to y_1 with some wt.



$\boxed{y = wx}$

net i/p

wt

+ve → Excitatory wt. (∵ inc o/p)

−ve → inhibitory wt. (∵ dec o/p)

$\boxed{y = xw}$

↓

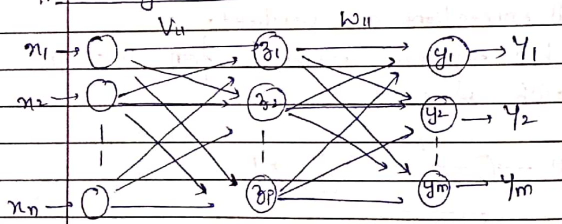
Activation/transfer f^n

↓

$Y = f(y)$

- net i/p is of no use in supervised learning
- Every neuron has a threshold i.e. θ
- When neuron exceed its threshold then only it comes in firing state and generate o/p.
- Rest, it remains in non-firing state
- After acheiving firing state once, each neurons needs come rest before again acheiving firing state.
- While retraining, we change wt. of connections.

Multilayer Architecture



$\boxed{z_1 = xv}$

↓

Activation f^n

$\boxed{f(z_1) = Z_1}$

↓

$\boxed{y_1 = Z_1 w}$

↓

activation f^n

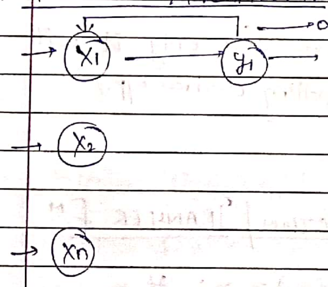
$Y_1 = f(y_1)$

- Any layer added b/w first and last layer is called hidden layer.
- There is no restriction on no. of hidden layers as well as on no. of neurons in each layer.

(individual multiplication as well as matrix multiplication both depend on coding style)

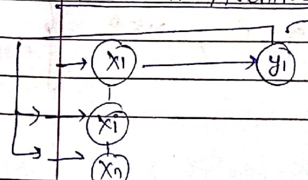
Feedback Architecture

→ o/p is fed to the same i/p



Recurrent Architecture (Eg. Hop field Network)

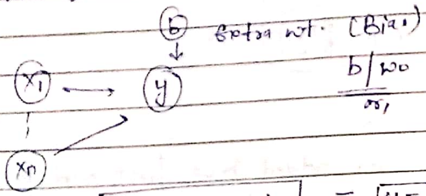
→ o/p is fed to other neurons except the same one.



Page No.
Date : / /

Bias Architecture

extra wt. applied to neurons and has a unit activation i.e. const. (+1) *



$$\boxed{y = xw + b} \equiv \boxed{y = mx + c}$$

st. line eqn

Here, 'b' decides shape of curve.

Importance of bias wt.

→ It helps in better conversion
→ Conversion closer to the desired o/p

Why b is +ve

→ b if +ve then only it will help in getting desired o/p.

Activation Function / Transfer f^n

(1) Identity f^n : $f(x) = x;\ \forall x$

(2). Binary f^n : $f(x) = \begin{cases} 1 ; & x \geq 0 \\ 0 ; & x < 0 \end{cases}$

Page No.
Date : / /

(3). Bipolar step f^n : $f(x) = \begin{cases} 1 ; & x \geq 0 \\ -1 ; & x < 0 \end{cases}$

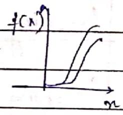
discrete function

w equality sign can be shifted for desired o/p's.

(4). Binary sigmoidal f^n :

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

λ → stepness f^n



derivative of f(x)

$$\boxed{f'(x) = \lambda f(x) [1 - f(x)]}$$

* λ → scalar
generally (0.5 to 1)
if not given, assume ($\lambda = 1$)

(5). Bipolar sigmoidal :

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

continuous activation f^n
(mainly used in back propagation f^n.)

derivative

$$\boxed{f'(x) = \frac{\lambda}{2} \left(1 + f(x) \right) \left(1 - f(x) \right)}$$

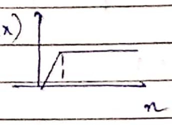
Scanned by CamScanner

6. Ramp f^n :

$$f(x) = \begin{cases} 1 & ; x>0 \\ x & ; x=0 \\ 0 & ; x<0 \end{cases}$$

$$\begin{cases} 1 & ; x>0 \\ 0 & ; x=0 \\ -1 & ; x<0 \end{cases}$$

↳ for perceptual neuron



Learning rate α'

Learning machine to get desired o/p which we already know.

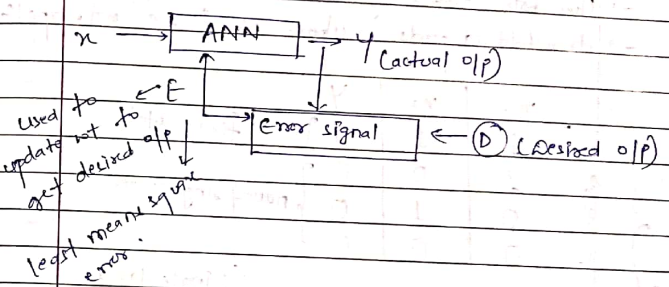
if, actual o/p ≠ desired o/p

- retraining is done by changing wt.
- Δw is calculated using α' to get desired o/p
- α need to smaller to provide fast learning

α → not given assume (α=1) → for ease of computation.

Types Of Learning :-

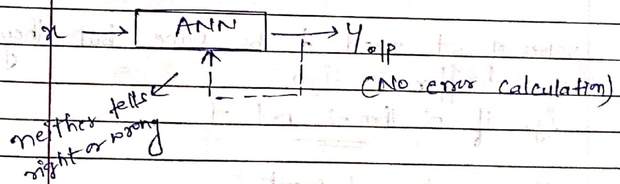
1. Supervised Learning   Eg. Classification



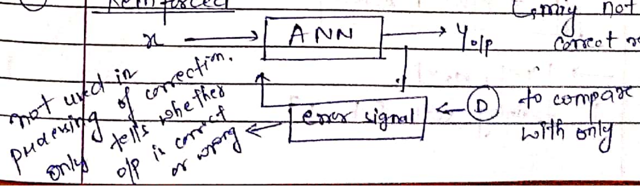
$$(MSE(E) = \frac{1}{2}(D-Y)^2$$

↳ random factor to cancel out 2 at time of derivative.

2. Unsupervised   Eg. Clustering   (Have learing rule)



3. Reinforced ← (b/w supervised & unsupervised)
↳ may not give correct result



Scanned by CamScanner

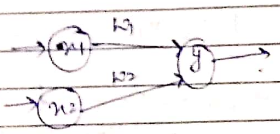
Page No.
Date : / /

Mc Culloch Pitts Model (MCP)

- the i/p & o/p value will be 0 or 1.
- discrete binary step f^n. (activation)
- uses both exhibitory & inhibitory model
- threshold depends upon -ve & +ve wt. applied

	x_1	x_2	y	
1.	1	1	1	→ firing state
2.	1	0	0	non-firing state.
3.	0	1	0	
4.	0	0	0	

training pairs.



(Single layer feed forward model)

x, y or $s : t$
string target

values of wt need to be same but they can differ in sign.
eg. if 2 then -2 not -1

y_{in} → denotes net input

$$\boxed{y_{in} = x_1 w_1 + x_2 w_2}$$

→ $w_1 = 1$, $w_2 = 1$

1st training sets [1 1]

Page No.
Date : / /

$y_{in_1} = 2$

→ [1 0]

$y_{in_2} = 1$

→ [0 1]

$y_{in_3} = 1$

→ [0 0]

$y_{in_4} = 0$

Find threshold value.

$$y = \begin{cases} 1 \ ; & y_{in} \geq \theta \\ 0 \ ; & y_{in} < \theta \end{cases}$$

from computation, [1 1] results in 2.

⇒ $(\theta = 2)$ ∵ [1 1] is the only firing state

$$y = \begin{cases} 1 \ ; & y_{in} \geq 2 \\ 0 \ ; & y_{in} < 2 \end{cases}$$

Assumption model for realization

Page No.
Date: / /

Decision Boundary

if the pts. can be demarked on two sides of a line then it is linearly separable.

OR
Q. Realise ~~and~~ gate using MCP model.

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

] firing states
] non-firing "

$w_1 = w_2 = 1$

$x_1 w_1 + x_2 w_2 = y_{in}$

[1 1]

$y_{in} = 1+1 = 2$

[1 0]

$y_{in} = 1+0 = 1$

$\theta = 1$ based on computation.

[0 1]

$y_{in} = 1$

[0 0]

$y_{in} = 0$

Page No.
Date: / /

$$y = \begin{cases} 1 ; & y_{in} \geq 1 \\ 0 ; & y_{in} < 1 \end{cases}$$

AND-NOT gate. $(x_1 \wedge x_2')$

and first variable with not of second variable.

x_1	x_2	x_2'	y
1	1	0	0
1	0	1	1 → firing state.
0	1	0	0
0	0	1	0

$\boxed{w_1 = 1 \quad w_2 = -1}$

[1 1] ($w_1 = 1$ & $w_2 = 1$) ✗

$y_{in} = x_1 w_1 + x_2 w_2$
$= 1 + 1 = 2.$

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

[1 0]
$y_{in} = 1+0 = 1$

$1(1) + 0(-1)$
$1 + 0 = 1$ ✓

[0 1]
$y_{in} = 1$ ✗

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

[0 0]
$y_{in} = 0$

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

$\therefore \theta = 1$

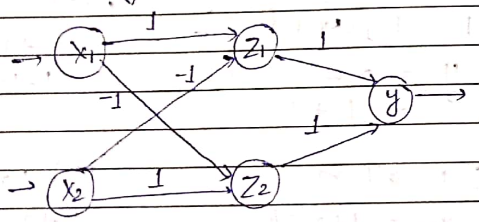
$$y = \begin{cases} 1 ; & y_{in} \geq 1 \\ 0 ; & y_{in} < 1 \end{cases}$$

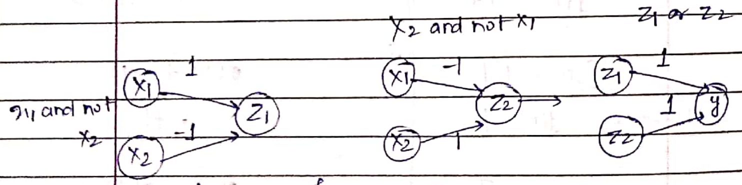
XOR Gates

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

On computing, we unable to get a linear separable architecture.

Now, we'll go for multilayer architecture by introducing a hidden layer.





find θ for each three gates.

x_1	x_2	z_1	
0	0	0	1
0	1	0	0
1	0	1	1
1	1	0	0

$0\times1 + 0(-1) = 0$

$0\times1 + 1(-1) = -1$

$1\times1 + 0\times(-1) = 1$

$1\times1 + 1\times(-1) = 0$

$\theta = 1$

22nd – Test

x_2 and not x_1

x_1	x_2	z_2	
0	0	0	1
0	1	1	1
1	0	0	0
1	1	0	0

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

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

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

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

$\theta = 1$

Z_1 or Z_2

x_1	x_2	z_1	z_2	w
0	0	0	0	0
0	1	0	1	1
1	0	1	0	1
1	1	0	0	0

$0(1) + 0(1) = 0$

$0(1) + 1(1) = 1$

$1(1) + 0(1) = 1$

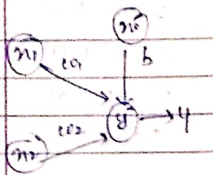
$0(1) + 0(1) = 0$

$\theta = 1$

$$\begin{cases} 1 ; & y_{in} \geq 1 \\ 0 ; & y_{in} < 1 \end{cases}$$

LINEAR SEPRABILITY

Date : / /



$y_{in} = x_1 w_1 + x_2 w_2 + b$

if pt. lies on line.

$\Rightarrow y_{in} = 0$

$\Rightarrow x_1 w_1 + x_2 w_2 + b = 0$

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

eq^n for decision line.

compare for, $\boxed{y = mx + c}$

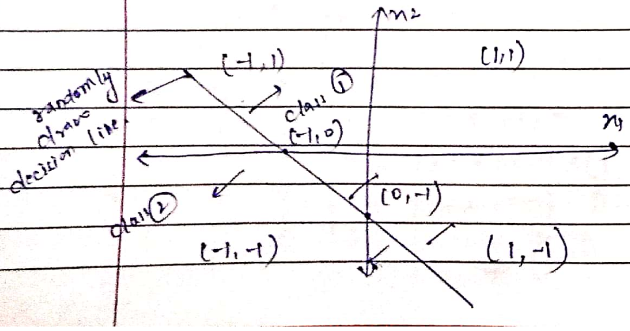
Eg. 1.

OR Gate

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

same o/p ⇒ belongs to same class. (first three rows)

−1 − Class 2 (last row)



Page No.
Date : / /

Outliers → can never be classified under any class.

→ They do not listen to problem.

→ Every problem consist of such pts.

∴ Ignore them as they will always dec. the performance.

Now,

$(x_1, y_1) = (-1, 0)$

$(x_2, y_2) = (0, -1)$

$m = \frac{y_2 - y_1}{x_2 - x_1} = -1$

$y_1 = m x_1 + c$

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

$\Rightarrow c = -1$

$\Rightarrow \boxed{y = -x_1 - 1}$

Now compare it with standard eq^n.

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

$\Rightarrow \frac{w_1}{w_2} = 1$, $\frac{b}{w_2} = 1$

$\Rightarrow \boxed{w_1 = w_2 = b}$

Now, for ease of computation take them 1.

$\Leftarrow \boxed{w_1 = w_2 = b = 1}$

Page No.
Date : / /

x_1	x_2	x_0	y_{in}	y
1	1	1	3	1
1	-1	1	1	1
-1	1	1	1	1
-1	-1	1	-1	-1

const. activation of (+1).

firing state in original one

$\Rightarrow \theta = 1$

table based on this eqn

$\boxed{y_{in} = x_1 w_1 + x_2 w_2 + x_0 b}$

Compare with y to see 'θ'

$$y = \begin{cases} 1 & y_{in} \geq +1 \\ -1 & y_{in} < 1 \end{cases}$$

$\because \theta$ can be figured out

$\Rightarrow$ OR gate is linearly separable.

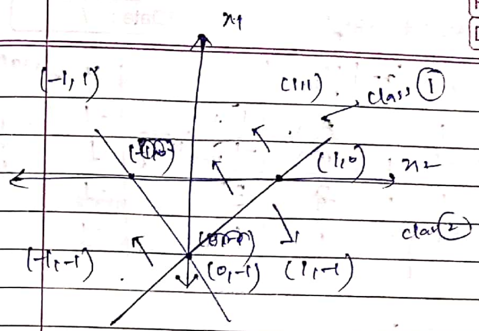
(if coming y is same as expected or desired y.) linearly separable.

Q. 2. Check whether AND NOT gate is linearly separable or not.

x_1	x_2	x_2'	y
1	1	-1	-1 ✓
1	-1	1	1 — 2nd class
-1	1	-1	-1 ✓
-1	-1	1	-1 ✓

1 class

Page No.
Date : / /



$(x_1, y_1) = (+1, 0)$

$(x_2, y_2) = (0, -1)$

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$= \frac{+1 - 0}{0 - (-1)} = +1.$$

$\boxed{y_1 = +x + c}$

compare.

(1,0)

$0 = x + c$

$0 = 1 + c$

$c = -1$

$\boxed{y_1 = x - 1}$

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

$\Rightarrow \frac{w_1}{w_2} = -1$ & $\frac{b}{w_2} = +1$

$\Rightarrow w_1 = -w_2 = b = 1$

$\Rightarrow \boxed{w_1 = -1,\ w_2 = +1\ \&\ b = +1}$

Page No.
Date : / /

$w_1=1,\ w_2=+1,\ b=+1$

x_1	x_2	x_0	y_{in}		y
1	1	1	~~1~~	1	-1
1	-1	1	-1	-1 firing	1
-1	1	1	-3	3	-1
-1	-1	1	-1	1	-1

→ same ∴ linearly seperable.

$y_{in} = x_1 w_1 + x_2 w_2 + x_0 b$

$= \cancel{x_1 0 - x_2 0 + 1}$ $\boxed{-x_1 + x_2 + b}$

$y_{in} = 1 - 1 + 1 = 1$	$-(1) + (1) + (1) = 1$
$1 - (-1) + 1 = -1$	$-(1) + (-1) + (1) = -1$
$-1 - 1 + 1 = -3$	$-(-1) + (1) + (1) = 3$
$-1 - (-1) + 1 = -1$	$-(-1) + (-1) + (1) = 1$

X

$\theta = 1$

$$y = \begin{cases} -1 ; & y_{in} \geq 1 \\ 1 ; & y_{in} < 1 \end{cases}$$

* IMP // ∵ coming wrong ⇒ we should take $\boxed{w_1 = 1,\ w_2 = -1 \text{ and } b = -1}$

x_1	x_2	x_0	y_{in}	y
1	1	1	-1	-1
1	-1	1	1	1
-1	1	1	-3	-1
-1	-1	1	-1	-1

1). $1(1) + 1(-1) + 1(-1) = 1 - 1 - 1 = -1$
2). $1(1) + (-1)(-1) + 1(-1) = 1$
3). $1(-1) + (-1)(1) + 1(-1) = -3$
4). $1(-1) + (-1)(-1) + (-1)(1) = -1$

$\boxed{\theta = 1}$

$$y = \begin{cases} 1 ; & y_{in} \geq 1 \\ -1 ; & y_{in} < 1 \end{cases}$$

LEARNING RULES

Page No.
Date : / /

$W_{new} = W_{old} + \Delta w$

$\Delta w = \alpha (LS)$ → product of learning rate or signal.

$b_{new} = b_{old} + \Delta b$

$\Delta b = \alpha (LS)$

* It is diffn for each learning rule.

(1). Hebb Learning Rule → eg of unsupervired learning.

- If neuron A is in firing mode & B is adjacent to it then, it will push neuron B towards acheiving firing state.

- Bipolar mode → o/p is 1 (ON) or -1 (OFF).

- initial value of weights is always 0.
$w_1 = w_2 = \dots = w_n = 0$

* { We are finding new wts & training our system bcz. it assume's that old wt. might not be correct. }

$\Delta w = \alpha (x \cdot y)$

$\Delta b = \alpha (1)(y)$
$= \alpha y$

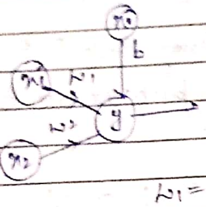
* taking bias is a compulsion.

Page No.
Date: / /

Q. Implement a Hebb Net for AND gate.

	x_1	x_2	x_0	y
1.	1	1	1	1
2.	1	-1	1	-1
3.	-1	1	1	-1
4.	-1	-1	1	-1

IMP. target o/p y is based only on x_1 and x_2. bias x_0 is only for implementation purpose.



$w_1 = w_2 = b = 0$

s : t
storage target

Training pair

1). [1 1 1] : 1 ← o/p

∵ unsupervised learning ⇒ no calculation for net i/p

⇒ θ i.e. threshold is not calculated.

⇒ m/c is learned with the pair we provide them.

But for ease of computation, use, $\alpha = 1$

0.2 generally give good result

$\alpha = 1$

Page No.
Date: / /

$\Delta w_1 = \alpha . x_1 . y$
$= (1)(1)(1) = 1$

$\Delta w_2 = \alpha . x_2 . y$
$= (1)(1)(1) = 1$

$\Delta b = 1$ ($\alpha . y$)

IMP. o/p depends on sequence of i/p taken.

$w_1 new = 1 + 0 = 1$
$w_2 new = 1 + 0 = 1$
$b new = 1 + 0 = 1$

IMP. one epoch is equal to one iteration i.e. one learning/training cycle.

2). [1 -1 1] : -1

$\Delta w_1 = \alpha . x_1 . y$
$= 1 (1) (-1) = -1$

IMP. when to stop.
when, $\Delta w = 0$
⇒ wt. will not change
⇒ ans. has been obtained.

$\Delta w_2 = \alpha . x_2 . y$
$= 1 (-1)(-1) = 1$

$\Delta b = \alpha . y = 1(-1) = -1$

$w_1 new = 1 - 1 = 0$
$w_2 new = 1 + 1 = 2$
$b_{new} = 1 - 1 = 0$

3). [-1 1 1] : -1

$\Delta w_1 = \alpha . x_1 . y$
$= 1(-1)(-1) = 1$

$\Delta w_2 = \alpha . x_2 . y = 1(1)(-1) = -1$

Page No.
Date : / /

$\Delta b = 1(-1) = -1$

$w_1 new = 0 + 1 = 1$
$w_2 new = 2 + (-1) = 1$
$b_{new} = 0 + (-1) = -1$

diff^n implies, $\boxed{w_{old} = w_{new}}$ consecutive.

~~this old no needs to be immediate.~~
~~It could have came in history~~

4). [-1 -1 1] : -1

$\Delta w_1 = \alpha . x_1 . y = 1(-1)(-1) = 1$
$\Delta w_2 = \alpha . x_2 . y = 1(-1)(-1) = 1$
$\Delta b = \alpha y = 1(-1) = -1$

$w_1 new = 1 + 1 = 2$
$w_2 new = 1 + 1 = 2$
$b_{new} = -1 - 1 = -2$

Q. Implement a Hebb Net for OR gate upto 2 epochs.

x_1	x_2	x_0	y
1	1	1	1
1	-1	1	1
-1	1	1	1
-1	-1	1	-1

** Neural network which uses Hebb Rule is called Hebb Net.

Page No.
Date : / /

$\alpha = 1$.

1. [1 1 1] : 1

$\Delta w_1 = \alpha . x_i y$
$= 1(1)(1) = 1$

$\Delta w_2 = \alpha x_2 . y$
$= 1(1)(1) = 1$

$\Delta b = \alpha . y = 1(1) = 1$

$w_{1 new} = 1 + 0 = 1$
$w_{2 new} = 1 + 0 = 1$
$b_{new} = 1$

2. [1 -1 1] : 1

~~$\Delta w_1 = 1(1)(1) = 1$~~
~~$\Delta w_2 = 1(-1)(1) = -1$~~
$\Delta b = 1(1) = 1$

$w_{1 new} = 1 + 1 = 2$
$w_{2 new} = 1 + (-1) = 0$
$b_{new} = 1 + 1 = 2$

3. [-1 1 1] : 1

$\Delta w_1 = 1(-1)(1) = -1$
$\Delta w_2 = 1(1)(1) = 1$
$\Delta b = 1(1) = 1$

$w_{1 new} = 2 - 1 = 1$
$w_2 new = 0 + 1 = 1$
$b_{new} = 2 + 1 = 3$

4. [-1 -1 +1] : -1

$\Delta w_1 = (-1)(-1)(1) = 1$
$\Delta w_2 = 1(-1)(-1) = 1$
$\Delta b = 1(-1) = -1$

$w_{1 new} = 1 + 1 = 2$
$w_2 new = 1 + 1 = 2$
$b_{new} = 3 - 1 = 2$

2nd epoch

(2, 2, 2)

Page No.
Date: / /

1. $\begin{bmatrix} 1 & (1) & 1 \end{bmatrix} : 1$

~~$\Delta w_1 = 1(1)(1) = 1$~~
~~$\Delta w_2 = 1(1)(1) = 1$~~
~~$\Delta b = 1(1) = 1$~~

$w_{1N} = 3$
$w_{2N} = 3$
$b_N = 3$

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

$w_{1N} = 3 + 1 = 4$
$w_{2N} = 3 + (-1) = 2$
$b_N = 3 + 1 = 4$

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

$w_{1N} = 4 - 1 = 3$
$w_{2N} = 2 + 1 = 3$
$b_N = 4 + 1 = 5$

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

$w_{1N} = 3 + 1 = 4$
$w_{2N} = 3 + 1 = 4$
$b_N = 5 - 1 = 4$

Tut – (12–1) – Tut-2
Thurs.

Page No.
Date: / /

Q. Create a ~~simpl~~ suitable Heb Net to identify the patterns L and U where, L ∈ Class 1 and U ∈ class 2. find the wt.'s after learning for one epoch.

Class 1: L (+1)   Class 2: U (−1)

3x3 matrix

```
*  .  .        *  .  *
*  .  .        *  .  *
*  *  *        *  *  *
```

Common platform for both.

Min. possible was 3x3 matrix.

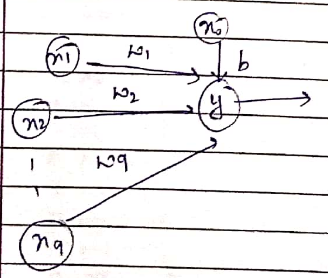
* pst U in firing state i.e. (+1)

• pst is in non-firing state i.e. (−1)

Pattern	I/P's										O/P
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_0	
L	1	-1	-1	1	-1	-1	1	1	1	1	+1
U	1	-1	1	1	-1	1	1	1	1	1	-1

$w_1 = w_2 = w_3 = w_4 = w_5 = w_6 = w_7 = w_8 = w_9 = b = 0$

(initially, wt. = 0)



Page No.
Date : / /

($\alpha = 1$)

$[1\ -1\ -1\ 1\ -1\ -1\ 1\ 1\ 1\ 1] : +1$

$\Delta w_1 = \alpha x_1 \cdot y$
$= 1 \times 1 \times (1) = 1$
$\Delta w_2 = 1(-1)(1) = -1$
$\Delta w_3 = 1(-1)(1) = -1$
$\Delta w_4 = 1(1)(1) = 1$
$\Delta w_5 = 1(-1)(1) = -1$
$\Delta w_6 = -1$
$\Delta w_7 = 1$
$\Delta w_8 = 1$
$\Delta w_9 = 1$
$\Delta b = \alpha(y) = 1(1) = 1$

w_1 new
w_2 new

Page No.
Date : / /

Q. Use the heb net to learn the patterns, I and O. Where I ∈ Class 1 and O ∈ Class 2.
Find the final weights after 2 epochs.

* * *   * * *
. * .   * . *
* * *   * * *

I (+1)   O (−1)

Pattern	I/P										O/P
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_0	
I	1	1	1	−1	1	−1	1	1	1	1	+1
O	1	1	1	1	−1	1	1	1	1	1	−1

($\alpha = 1$)

Set 1

$\Delta w_1 = 1$ $\quad w_1$ new $= 1$
$\Delta w_2 = 1$ $\quad w_2 = 1$
$\Delta w_3 = 1$ $\quad w_3 = 1$
$\Delta w_4 = -1$ $\quad w_4 = -1$
$\Delta w_5 = 1$ $\quad w_5 = 1$
$\Delta w_6 = -1$ $\quad w_6 = -1$
$\Delta w_7 = 1$ $\quad w_7 = 1$
$\Delta w_8 = 1$ $\quad w_8 = 1$
$\Delta w_9 = 1$ $\quad w_9 = 1$
$\Delta b = 1$ $\quad b = 1$

Set 2

$1(-1) = -1$

$\Delta w_1 = -1$ $\quad w_1 = 0$ $\quad b = 0$
$\Delta w_2 = -1$ $\quad w_2 = 0$
$\Delta w_3 = -1$ $\quad w_3 = 0$
$\Delta w_4 = -1$ $\quad w_4 = -2$
$\Delta w_5 = 1$ $\quad w_5 = 2$
$\Delta w_6 = -1$ $\quad w_6 = -2$
$\Delta w_7 = -1$ $\quad w_7 = 0$
$\Delta w_8 = -1$ $\quad w_8 = 0$
$\Delta w_9 = -1$ $\quad w_9 = 0$
$\Delta b = -1$

Page No.

Date : / /

epoch ②

		set ②
set ①	$w_1 = 1$	$w_1 = 0$
	$w_2 = 1$	$w_2 = 0$
	$w_3 = 1$	$w_3 = 0$
	$w_4 = -3$	$w_4 = -4$
	$w_5 = 3$	$w_5 = 4$
	$w_6 = -3$	$w_6 = -4$
	$w_7 = 1$	$w_7 = 0$
	$w_8 = 1$	$w_8 = 0$
	$w_9 = 1$	$w_9 = 0$
	$b = 1$	$b = 0$

SUPERVISED LEARNING

Page No.

Date : / /

1. Perceptron Algorithm

$$w_{new} = w_{old} + \Delta w$$

$$\Delta w = \alpha t x$$

calculate

$y \neq t \rightarrow$ target o/p

actual o/p we are getting

✓ This w_{new} can be calculated for more than one epoch

✓ Irrespective of I/P, we assume target o/p t in bi-polar form

✓ activation f^n taken $\rightarrow$ step activation f^n.

$$y = \begin{cases} 1 & \text{if } y_{in} > \theta \\ 0 & \text{if } y_{in} = \theta \\ -1 & \text{if } y_{in} < \theta \end{cases}$$

✓ if θ is not given assume it to be 0

Eg. AND Gate

x_1	x_2	x_0	t
1	1	1	1
1	-1	1	-1
-1	1	1	-1
-1	-1	1	-1

$\Rightarrow$ There is no condition on i/p weight

Page No.
Date : / /

if not given, we take them to be '0' initially

$$\begin{cases} w_1 = w_2 = b = 0 \\ \alpha = 1 \\ \theta = 0 \end{cases}$$

Now

$$y = \begin{cases} 1 & ; y_{in} > 0 \\ 0 & ; y_{in} = 0 \\ -1 & ; y_{in} < 0 \end{cases}$$

1. $[1 \quad 1 \quad 1] : 1$

$y_{in} = 1(0) + 1(0) + 1(0) = 0$

$y = 0$

$\Rightarrow y \neq t$

$\Rightarrow$ training needs to be done.

$\Delta w_1 = \alpha . t . x_1 = 1(1)(1) = 1$	$w_{1N} = 0+1 = 1$	correct set of for 1st.
$\Delta w_2 = 1(1)(1) = 1$	$w_{2N} = 0+1 = 1$	
$\Delta b = 1(1) = 1$	$b_N = 0+1 = 1$	

2. $[1 \quad -1 \quad 1] : -1$

$y_{in} = 1(1) + (-1)(1) + 1(1)$
$= 1$

$y = 1$

$y \neq t$

Page No.
Date : / /

↓ train

$\Delta w_1 = 1(-1)(1) = -1$	$w_{1N} = 1-1 = 0$
$\Delta w_2 = 1(-1)(-1) = 1$	$w_{2N} = 1+1 = 2$
$\Delta b = 1(-1)(1) = -1$	$b_N = 1-1 = 0$

3. $[-1 \quad 1 \quad 1] : -1$

$y_{in} = -1(0) + 1(2) + 1(0) = 2$

$y = 1$

$y \neq t$

↓ train

$\Delta w_1 = 1(-1)(-1) = 1$	$w_{1N} = 0+1 = 1$
$\Delta w_2 = 1(-1)(1) = -1$	$w_{2N} = 2-1 = 1$
$\Delta b = 1(-1)(1) = -1$	$b_N = 0-1 = -1$

4. $[-1 \quad -1 \quad 1] : -1$

$y_{in} = -1(1) + (-1)(1) + 1(-1) = -3$

$y = -1$

$\Delta w_1 = 1(-1)(-1) = 1$
$\Delta w_2 = 1(-1)(-1) = 1$
$\Delta b = 1(-1)(1) = -1$

$y = t.$

$\therefore$ No training is reqd.

* After 1st epoch :-

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

{ if not given, assume $\alpha = 0$
{ make step activation

Page No.
Date: / /

Q. Use a perceptron network to find the final weights of the given data with $\alpha=1$, initial wt. are 0 after 2 epochs.

x_1	x_2	x_3	x_4	~~b~~	t
1	1	1	1		1
-1	1	-1	-1		1
1	1	1	-1		-1
1	-1	-1	1		-1

(don't assume bias if not mentioned. IMP)
IMP = only assume it in Gates by default

$$y = \begin{cases} 1 & ; y_{in} > 0.2 \\ 0 & ; -0.2 \le y_{in} \le 0.2 \\ -1 & ; y_{in} < -0.2 \end{cases}$$

1. $[1 \quad 1 \quad 1 \quad 1] : 1$

$y_{in} = 1(0) + 1(0) + 1(0) + 1(0) = 0$

$y = 0$

$y \ne t$

$\downarrow$ train

$\Delta w_1 = 1(1)(1) = 1$	$w_1' = 1$
$\Delta w_2 = 1(1)(1) = 1$	$w_2' = 1$
$\Delta w_3 = 1(1)(1) = 1$	$w_3' = 1$
$\Delta w_4 = (1)(1)(1) = 1$	$w_4' = 1$

Page No.
Date: / /

2. $[-1 \quad 1 \quad -1 \quad -1] : 1$

$y_{in} = -1(1) + 1(1) + -1(1) + (-1)(1)$
$= \cancel{-1} \cancel{+1} \cancel{-1} \cancel{-1} = 0$

$y = 0$

$y \ne t$

$\downarrow$ train

$\Delta w_1 = 1(1)(-1) = -1$	$w_1' = 0$
$\Delta w_2 = 1(1)(1) = 1$	$w_2' = 2$
$\Delta w_3 = 1(1)(-1) = -1$	$w_3' = 0$
$\Delta w_4 = 1(1)(-1) = -1$	$w_4' = 0$

3. $[1 \quad 1 \quad 1 \quad -1] : -1$

$y_{in} = 1(0) + 1(2) + 1(0) + (-1)(0) = 2$

$y = 1$

$y \ne t$

$\downarrow$ train.

$\Delta w_1 = -1(1) = -1$	$w_1' = -1$
$\Delta w_2 = -1(1) = -1$	$w_2' = 1$
$\Delta w_3 = -1(1) = -1$	$w_3' = -1$
$\Delta w_4 = -1(-1) = 1$	$w_4' = 1$

Page No.
Date : / /

4 [1 -1 -1 1] : -1

$y_{in} = -1(1) + 1(-1) + (-1)(-1) + 1(1)$
$= -1 -1 +1 +1 = 0$

$y = 0$

$y \neq t$

↓ train

$\Delta w_1 = (-1)(1) = -1$	$w_1' = -2$
$\Delta w_2 = (-1)(-1) = 1$	$w_2' = 2$
$\Delta w_3 = (-1)(-1) = 1$	$w_3' = 0$
$\Delta w_4 = (-1)(1) = -1$	$w_4' = 0$

2nd epoch

1 [1 1 1 1] : 1

$y_{in} = -2(1) + 2(1) + 0 + 0 = 0$

$y = 0$

$y \neq t$

↓ train

$\Delta w_1 = 1(1) = 1$	$w_1' = -1$
$\Delta w_2 = 1(1) = 1$	$w_2' = 3$
$\Delta w_3 = 1(1) = 1$	$w_3' = 1$
$\Delta w_4 = 1(1) = 1$	$w_4' = 1$

Page No.
Date : / /

2 [-1 1 -1 -1] : 1

$y_{in} = -1(-1) + 3(1) + 1(-1) + 1(-1)$
$= 1 + 3 - 2 = 2$

$y = 1$

$y = t$

⇒ wt remains same.

$w_1' = -1$
$w_2' = 3$
$w_3' = 1$
$w_4' = 1$

3 [1 1 1 -1] : -1

$y_{in} = -1(1) + 3(1) + 1(1) + (-1)(1)$
$= -1 + 3 + 1 - 1 = 2$

$y = 1$

$y \neq t$

↓ train

$\Delta w_1 = -1$	$w_1' = -2$
$\Delta w_2 = -1$	$w_2' = 2$
$\Delta w_3 = -1$	$w_3' = 0$
$\Delta w_4 = 1$	$w_4' = 2$

Page No.
Date : / /

Y [1 -1 -1 1] : -1

$y_{in} = 1(-2) + (-1)(2) + (-1)(0) + 1(2)$
$= -2 - 2 + 0 + 2 = -2$

$\boxed{Y = -1}$

$Y = t$

∴ final wt. are,

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

✓ Perceptron is easily applicable to Linear Classification.

UCI Repository

Page No.
Date : / /

* Delta Learning Rule

$w_{new} = w_{old} + \Delta w$
$\Delta w = \alpha \cdot (t - y_{in}) \cdot x$
Error, $E = (t - y_{in})^2$ → least mean square error.

* $Y = \begin{cases} 1 & ; y_{in} \geq 0 \\ -1 & ; y_{in} < 0 \end{cases}$

This activation fn is used when we replace y_{in} in above formulas by Y.

[ideally, bipolar in nature]

→ initially, weights should not be zero. They as low as possible but > 0.
→ This helps in better conversion as well as avoid over training.
→ multiple epoch as possible.
→ Error is calculated for each training pair.
→ epoch error is → sum error of each training pair & divide it by no. of training cases.

Que.

OR Gate :-

x_1	x_2	x_0	t
1	1	1	1
1	-1	1	1
-1	1	1	1
-1	-1	1	-1

$\alpha = 1$

$w_1 = w_2 = b = 0.1$

1) [1 1 1] : 1]

$y_{in} = 1(0.1) + 1(0.1) + 1(0.1) = 3 \times 0.1 = 0.3$

$\Delta w_1 = \alpha(t - y_{in}).x_1$

$= 1(1 - 0.3).1$

$= 1(0.7).1 = 0.7$

$\Delta w_2 = \alpha(t - y_{in}).x_2$

$= 0.7 \times 1 = 0.7$

$\Delta b = 0.7$

$Error = (t - y_{in})^2 = (1 - 0.3)^2$

$= (0.7)^2$

$= 0.49$

$w_1\ new = 0.1 + 0.7 = 0.8$

$w_2\ new = 0.1 + 0.7 = 0.8$

$b_{new} = 0.1 + 0.7 = 0.8$

2) [1 −1 1] : 1

$y_{in} = 1(0.8) - 1(0.8) + 1(0.8)$

$\cancel{= 1.7 - 1.7 + 1.7 = 1.7} = 0.8 - 0.8 + 0.8 = 0.8$

$\Delta w_1 = 1(1 - 0.8)(1) = 1(0.2)(1) = 0.2$

$\Delta w_2 = (+0.2)(-1) = -0.2$

$\Delta b = (+0.2)(1) = 0.2$

$E = (t - y_{in})^2 = \cancel{(-0.7)^2 = 0.49}\ (0.2)^2 = 0.04$

$w_{1N} = \cancel{1.7 - 0.7 = 1}$ | $0.8 + 0.2 = 1$

$w_{2N} = \cancel{1.7 + 0.7 = 2.4}$ | $0.8 - 0.2 = 0.6$

$b_N = \cancel{1.7 + 0.7 = 1}$ | $0.8 + 0.2 = 1$

3) [−1 1 1] : 1

$y_{in} = \cancel{(-1)(1) + 1(2.4) + 1(1)} = -1 + 0.6 + 1 = 0.6$

$= \cancel{2.4}\ 1 - 0.6$

$\Delta w_1 = 1(\cancel{-2.4})(-1) = -0.4$

$\Delta w_2 = 0.4(1)\ \cancel{(-1.4)\ (1)} = 0.4$

$\Delta b = 0.4(1)\ \cancel{(-1.4)\ (1)} = 0.4$

$E = (0.4)^2 = 0.16$

$w_{1N} = 1 + 1.4 = 1 + (-0.4) = 0.6$

$w_{2N} = 2.4 - 1.4 = 1 \quad 0.6 + 0.4 = 1$

$b_N = 1 - 1.4 = \quad 1 + 0.4 = 1.4$

$-1 + 3.8$

2.8

4) [−1 −1 1] : −1

~~$y_{in} = -2.4 - 1 - 0.4 = -3.4 - 0.4 = -3.8$~~

~~$\Delta w_1 = 1(-1-(-3.8))(-1) = -2.8$~~

~~$\Delta w_2 = (2.8)(-1) =$~~

~~$\Delta b = (2.8)(1)$~~

$y_{in} = -0.6 \cancel{-1 + 1.4} = -0.6 + 0.4 = -0.2$

$\Delta w_1 = 1(-1 + 0.2)(-1) = 0.8$

$\Delta w_2 = 1(-0.8)(-1) = 0.8$

$\Delta b = (-0.8)(1) = -0.8$

Page No.
Date : / /

$E = (-0.8)^2 = 0.64$

$w_{1N} = 0.6 + 0.8 = 1.4$

$w_{2N} = 1 + 0.8 = 1.8$

$b_N = 1.4 - 0.8 = 0.6$

epoch error $= \frac{0.49 + 0.04 + 0.16 + 0.64}{4}$

$= 0.33$

Q2.

x_1	x_2	x_3	x_4	x_0	t
1	1	-1	1	1	1
1	1	1	1	1	1
1	-1	-1	-1	1	-1
-1	1	-1	-1	1	-1

$\alpha = 0.2$

$w_1 = w_2 = w_3 = w_4 = b = 0.1$

1). [1 1 -1 1 1] : 1

$y_{in} = 0.1 + \cancel{0.1} - \cancel{0.1} + 0.1 + 0.1$

$= 0.3$

$0.2 \times 0.7 = 0.14$

$\Delta w_1 = 0.2(1-0.3)(1) = 0.14$

$\Delta w_2 = 0.14\ (1) = 0.14$

$\Delta w_3 = 0.14\ (-1) = -0.14$

$\Delta w_4 = 0.14\ (1) = 0.14$

$\Delta b = 0.14\ (1) = 0.14$

Page No.
Date : / /

$E = (0.7)^2 = 0.49$

$w_1 = 0.1 + 0.14 = 0.24$

$w_2 = 0.1 + 0.14 = 0.24$

$w_3 = 0.1 - 0.14 = -0.04$

$w_4 = 0.1 + 0.14 = 0.24$

$b = 0.1 + 0.14 = 0.24$

2). [1 1 1 1 1] : 1

$y_{in} = 0.24 + 0.24 - 0.04 + 0.24 + 0.24$

$= 0.96 - 0.04 = 0.92$

$0.2 \times 0.08 = 0.016$

$\Delta w_1 = 0.2(1-0.92) = 0.016$

$\Delta w_2 = 0.2\ (\text{"}) = 0.016$

$\Delta w_3 = 0.016$

$\Delta w_4 = 0.016$

$\Delta b = 0.016$

$E = (0.08)^2 = 0.0064$

$w_1 = 0.24 + 0.016 = 0.256$

$w_2 = 0.24 + 0.016 = 0.256$

$w_3 = -0.04 + 0.016 = -0.024$

$w_4 = 0.24 + 0.016 = 0.256$

$b = 0.24 + 0.016 = 0.256$

3). [1 -1 -1 -1 1] : -1

$y_{in} = 0.256 - \cancel{0.256} + 0.024 - \cancel{0.256} + \cancel{0.256}$

$= 0.024$

Scanned by CamScanner

$0.2 \times (-1.024)$
$= -2.048$

Page No.
Date : / /

$\Delta w_1 = 0.2(-1-0.024)\ (1) = -2.048 \quad -0.2048$
$\Delta w_2 = -2.048 \times (-1) = 2.048 \quad 0.2048$
$\Delta w_3 = (-1) = 2.048 \quad 0.2048$
$\Delta w_4 = (-1) = 2.048 \quad 0.2048$
$\Delta b = (1) = -2.048 \quad -0.2048$

$E = (-1.024)^2 = 1.048576$

$w_1 = 0.256 - 2.048 = -1.792$
$w_2 = 0.256 + 2.048 = 2.304$
$w_3 = -0.024 + 2.048 = 2.024$
$w_4 = 0.256 + 2.048 = 2.304$
$b = 0.256 - 2.048 = -1.792$

4). $[-1 \quad 1 \quad -1 \quad -1 \quad 1] : -1$

$y_{in} = 1.792 + \cancel{2.304} - 2.024 - \cancel{2.304} + 1.792$
$= 2.584 - 2.024$
$= \underline{0.560}$

$\Delta w_1 = 0.2(-1-0.560) = (-1.560)\,0.2$
$\Delta w_2 = -0.312$
$\Delta w_3 =$
$\Delta w_4 =$
$\Delta b =$

Page No.
Date : / /

$E = (-1.560)^2 = \underline{2.433}$

$$\text{net error} = \frac{2.4336 + 1.048576 + 0.0064 + 0.49}{4}$$

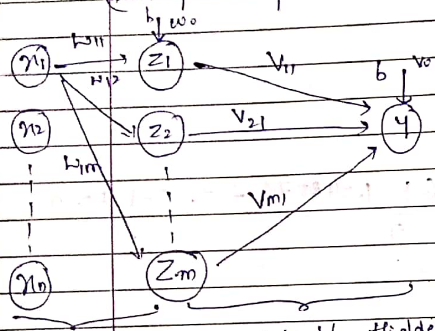
$=$

Scanned by CamScanner

ADELINE
(ADaptive Linear Neuron)

→ wt. are adjustable in nature.

Madelines
(Multiple adaptive Linear Neuron)



wt. b/w i/p layer & Hidden layer are adjustable.

wt. b/w Hidden layer and o/p layer are fixed.

→ target o/p is 1.

$$\boxed{\Delta W = \alpha(1 - z_{in})\, x}$$

$z_{in} = x \cdot w$

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

$$\boxed{y_{in} = z v}$$

∵ o/p of hidden layer is used to calculate y_{in} ∴ we need to apply activation fn. to get z

To converge the system to desired o/p, we take value of wt's as small as possible to avoid overshooting.

$$\boxed{v_1 = v_2 = \cdots = v_n = v = \frac{1}{2}}$$

we try to keep them same & as small as possible.

→ Also, apply activation fn on y_{in} to get y.

Now, compare, t and y.

if, $t = y$; no training

if, $t \neq y$; update wt.

$\boxed{t \neq y}$

Check, $\boxed{t = 1}$ → True → update neurons with +ve z_{in} as o/p.

for randomised t

false → update neurons whose z_{in} is as close to 0. (both +ve & −ve)

* parameter for selecting close to 0 i.e. range which we will consider depends on our coding style.

Q. Classify ~~the~~ i/p pattern I and F using perceptron network.

I ∈ Class 1

F ∈ Class 2

Assume, the initial wt & threshold as 0 & ($\alpha = 1$).

```
*  *  *        *  *  *
   *           *  *  *
*  *  *        *
```

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	t
1	1	1	-1	1	-1	1	1	1	1
1	1	1	1	1	1	1	-1	-1	-1

activation.

$$y = \begin{cases} 1 & ; y_{in} > 0 \\ 0 & ; y_{in} = 0 \\ -1 & ; y_{in} < 0 \end{cases}$$

(1). $y_{in} = 0$

$y = 0$

$y \neq t$

$\Delta w_1 = 1(1)(1) = 1$ $\Delta w_7 = 1$

$\Delta w_2 = 1(1)(1) = 1$ $\Delta w_8 = 1$

$\Delta w_3 = 1$ $\Delta w_9 = 1$

$\Delta w_4 = -1$ $\Delta b = 1$

$\Delta w_5 = 1$

$\Delta w_6 = -1$

$w_1 = 1$ $w_6 = -1$

$w_2 = 1$ $w_7 = 1$

$w_3 = 1$ $w_8 = 1$

$w_4 = -1$ $w_9 = 1$

$w_5 = 1$ $b = 1$

(2). $y_{in} = 1 + 1 + \cancel{1} - \cancel{1} + \cancel{1} - \cancel{1} + \cancel{1} - \cancel{1} - \cancel{1} + \cancel{1}$

$= 2$

$\boxed{y = 1}$

$y \neq t$

training

$\Delta w_1 = 1(-1)(1) = -1$ $\Delta w_5 = -1$ $\Delta w_9 = 1$

$\Delta w_2 = -1$ $\Delta w_6 = -1$ $\Delta b = -1$

$\Delta w_3 = -1$ $\Delta w_7 = -1$

$\Delta w_4 = -1$ $\Delta w_8 = 1$

$w_1 = 0$ $w_7 = 0$

$w_2 = 0$ $w_8 = 0$

$w_3 = 0$ $w_9 = 2$

$w_4 = -2$ $b = 0$

$w_5 = 0$

$w_6 = -2$

Q. Use adaline network to train AND NOT f^n with bipolar i/p's & targets. Assume the wt & learning rate to be (0.2).

x_1	x_2	x_0	y
1	1	1	-1
1	-1	1	1
-1	1	1	-1
-1	-1	1	-1

(1). $y_{in} = 1(0.2) + 0.2 + 0.2 = 0.6$

$\Delta w_1 = 0.2(-1-0.6)(1) = 0.2(-1.6) = -0.32$

$\Delta w_2 = 0.2(-1-0.6)(1) = 0.2(-1.6) = -0.32$

$\Delta b = 0.2(-1.6)(1) = -0.32$

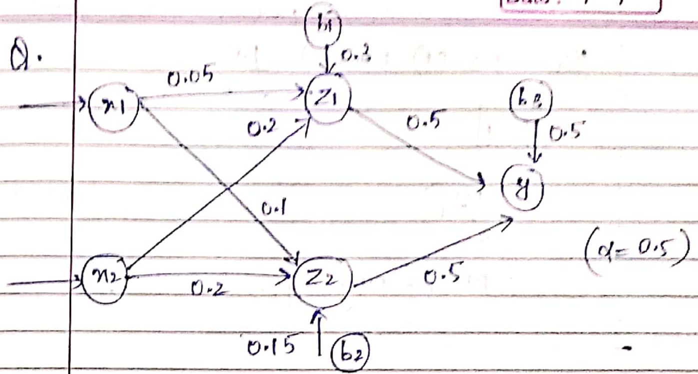
$w_1 = 0.2 - 0.32 = -0.12$

$w_2 = 0.2 - 0.32 = -0.12$

$b = 0.2 - 0.32 = -0.12$

$E = (t - y_{in})^2 = (-1.6)^2 = 2.56$

Q.



($\alpha = 0.5$)

Using MADLINE Network, implement XOR f^n with bipolar i/p's & targets.

(XOR-Table)

x_1	x_2	x_0	t
1	1	1	-1
1	-1	1	+1
-1	1	1	+1
-1	-1	1	-1

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

1). [1 1 1]:-1

$\boxed{\Delta w = \alpha(1 - z_{in})x}$ → perform, when $y \neq t$ not equal.

for Z_1

$z_{in} = 1(0.05) + 1(0.2) + 0.3 = 0.05 + 0.2 + 0.3 = 0.55$

$\boxed{Z_1 = 1}$

for Z_2

$z_{in} = 1(0.1) + 1(0.2) + 0.15 = 0.1 + 0.2 + 0.15 = 0.45 \Rightarrow Z_2 = 1$

Scanned by CamScanner

Page No.
Date : / /

$y_{in} = 0.5(1) + 0.5(1) = 1.5$

$\Rightarrow \boxed{y=1}$

$y \neq t$

$\therefore$ No training req.

Q). [1 -1 1] : -1

z_1. $z_{in} = 1(0.05) + (-1)(0.2) + 0.3$
$= 0.05 + 0.3 - 0.2$
$= 0.05 + 0.1 = 0.15 > 0$

0.10
0.15
0.05

$z_1 = 1$

z_2. $z_{in} = 1(0.1) + (-1)(0.2) + 0.15$
$= 0.1 - 0.2 + 0.15$
$= -0.1 + 0.15$
$= 0.05 > 0$

$z_2 = 1$

$y_{in} = 1(0.5) + 1(0.5) = 1$

$\Rightarrow y = 1$

$y \neq t$

Page No.
Date : / /

Perform training

$w_{11} = 0.05$, $w_{21} = 0.2$, $w_{22} = 0.2$, $w_{12} = 0.2$

Now, (Training)

$\because \boxed{t \neq 1}$ $(t = -1)$

$\Rightarrow$ Training whi's close to 0.

$w_{11} = 0.05$, $w_{12} = 0.1$, $w_{21} = 0.2$, $w_{22} = 0.2$

Now, train, (LOO). z_{in} of z_2 is close.
$\Rightarrow$ train w_{12} & w_{22}.

$\Delta w_{12} = \alpha(t - z_{in}).x_i$

$= 0.5(-1 - 0.45)(1) = 0.5(-1.45)$
$= -0.725$

$\Delta w_{22} = 0.5(-1-0.45)(1) = 0.5(-1.45)$
$= -0.725$

$w_{12} = 0.1 - 0.725 = -0.625$

$w_{22} = 0.2 - 0.725 = -0.525$

$\Delta b_2 = \alpha(-1.45) = 0.5(-1.45) = -0.725$
$b_2 = 0.15 - 0.725$
$= -0.575$

Scanned by CamScanner

Page No.
Date : / /

$w_{12} = -0.625$

$w_{22} = -0.625$

$\Delta b_2 = -0.575$

2). [1 -1 1] : +1

(z1) $z_{in} = 1(0.05) + (-1)(0.2) + 0.3$
$= 0.05 - 0.2 + 0.3$
$= 0.05 + 0.1$
$= 0.15$

$z_1 = 1$

(z2). $z_{in} = 1(-0.625) + (-1)(-0.625) + 1(-0.575)$
$= -0.625 + 0.625 - 0.575$
$= -0.575$

$z_2 = -1$

$y_{in} = 1(0.5) + (-1)(0.5) + 0.5$
$= 0.5$

$y = 1$

$y = t$

Page No.
Date : / /

3). [-1 1 1] : 1

(z1). $z_{in} = (-1)(0.05) + 1(0.2) + 0.3$
$= -0.05 + 0.5$
$= 0.45$

$z_1 = 1$

$$\begin{array}{r} 0.50 \\ -0.05 \\ \hline 0.45 \end{array}$$

(z2) $z_{in} = -1(-0.625) + (1)(-0.625) + 1(-0.575)$
$= 0.625 - 0.625 - 0.575$
$= -0.575$

$z_2 = -1$

$y_{in} = 1(0.5) + (-1)(0.5) + 0.5$
$= 0.5$

$y = 1$

$y = t$.

4). [-1 -1 1] : -1

$$\begin{array}{r} 0.1 \\ 0.05 \\ \hline 0.5 \end{array}$$

(z1) $z_{in} = -1(0.05) + (-1)(0.2) + 0.3$
$= -0.05 - 0.2 + 0.3$
$= -0.05 + 0.1$
$= 0.05$

$z_1 = 1$

(z2) $z_{in} = 0.625 + 0.625 - 0.575$
$= 1.250 - 0.575 = 0.675$ $z_2 = 1$

$y_{in} = 1(-1)5 + 1(0.5) + 0.5$

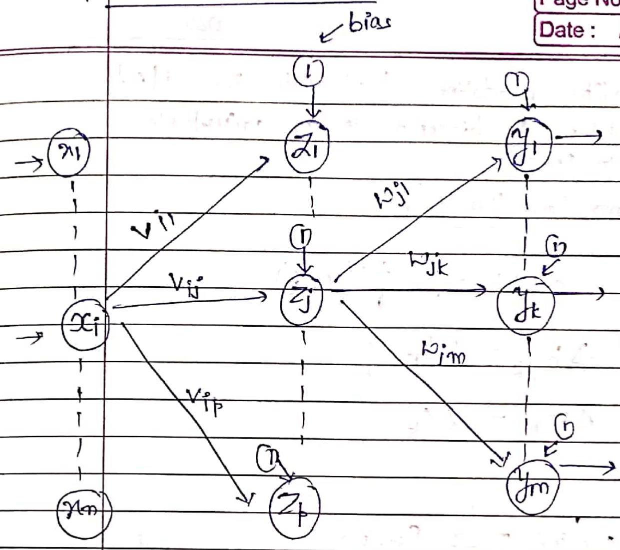
$= 1.5$

$y = 1$

$y \neq t$

training

BACKPROPOGATION



feedforward phase:

$$Z_{in\,j} = V_{oj} + \sum_{i=1}^{n} x_i v_{ij}$$

$$Z_j = f(Z_{inj})$$

$$y_{ink} = W_{ok} + \sum_{j=1}^{p} Z_j w_{jk}$$

$$y_k = f(y_{ink})$$

- applied to multilayer feedforward architecture.
- continuous activation fn → sigmoidal
- works in two phase → feed forward / back propogation

Page No.
Date : / /

* unlike madaline, both wts i.e. i/p to hidden & hidden to o/p are adjustable.

Back Propagation Phase:

Error Correction Term

$$\delta_k = (t_k - y_k) f'(y_{ink})$$

$$\Delta w_{jk} = \alpha . \delta_k . Z_j$$

$$\Delta w_{0k} = \alpha \delta_k$$

Error from o/p & hidden layer is backpropagated to hidden & i/p layer.

$$\delta_{inj} = \sum_{k=1}^{m} \delta_k . w_{jk}$$

$$\delta_j = (\delta_{inj}) f'(z_{inj})$$ derivative of activation fn.

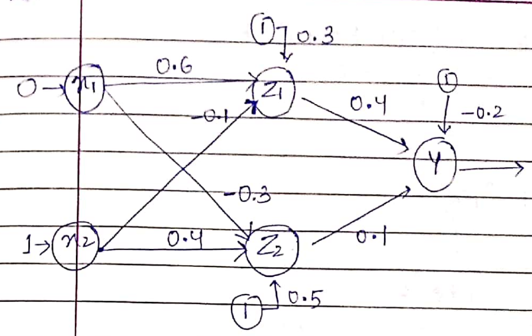
$$\Delta V_{ij} = \alpha . \delta_j . x_i$$

$$\Delta V_{0j} = \alpha . \delta_j$$

all new wts w & v are calculated i.e. updated at the end at the same time.

Page No.
Date : / /

Que.



for the given network, apply backpropagation algo to calculate the change in wt from i/p to hidden layer & from hidden layer to o/p layer.

$$\alpha = 0.25, \quad t = 1$$

$$f(x) = \frac{1}{1+e^{-\lambda x}} ; \quad \lambda = 1$$

$$f'(x) = f(x)[1 - f(x)]$$

x_1	x_2	x_0	t
0	1	1	1

[0, 1, 1] : 1

$$z_{1in} = 0(0.6) + 1(-0.1) + 0.3$$
$$= -0.1 + 0.3$$
$$= 0.2$$

Scanned by CamScanner

Page No.
Date : / /

$z_{2in} = 0(-0.3) + 0.4(1) + 0.5$
$= 0.4 + 0.5$
$= 0.9$

$Z_1 = f(z_{1in})$
$= \frac{1}{1+e^{-z_{1in}}}$
$= \frac{1}{1+e^{-0.2}} = 0.55$

$Z_2 = f(z_{2in})$
$= \frac{1}{1+e^{-0.9}} = 0.71$

$y_{in} = 0.55(0.4) + 0.71(0.1) - 0.2$
$= 0.22 + 0.071 - 0.2$
$= 0.02 + 0.071$
$= 0.091$

$Y = \frac{1}{1+e^{-0.091}} = 0.52$

Page No.
Date : / /

Error - Correction

$\delta_k = (t_k - y_k) f'(y_{ink})$

$\delta = (1 - 0.52)(0.25) = 0.48(0.25)$
$= 0.12$

$f'(y_{ink}) = f(y_{ink})[1 - f(y_{ink})]$
$= 0.52[1 - 0.52]$
$= 0.52(0.48)$
$= 0.25$

$\boxed{\delta_k = 0.12}$

$\Delta w_{11} = \alpha \cdot (0.12)(z_1)$
$= 0.25(0.12)(0.55)$
$= 0.0165$

$\Delta b = 0.25(0.12)$
$= 0.03$

$\Delta w_{21} = 0.25(0.12)(0.71)$
$= 0.0213$

$\delta_{inj} = \cancel{0.0165(0.12) + 0.0213(0.12)}$
$= \cancel{0.12(0.0165 + 0.0213)}$
$= \cancel{0.0045}$

Page No.
Date : / /

$\delta_{in1} = \cancel{0.0165}(\cancel{0.1})$

$\delta_k w_1 = 0.12(0.4)$
$= 0.048$

$\delta_{in2} = \delta_k w_2 = 0.12(0.1)$
$= 0.012$

$\delta_1 = \delta_{in1} f'(0.2)$
$= 0.048 [f(0.2)(1-f(0.2))]$
$= 0.048 [0.55(1-0.55)]$
$= 0.0118$

$\delta_2 = 0.012 [0.71(1-0.71)]$
$= 0.00247$

$\Delta V_{11} = \alpha \delta_1 (x_1) = 0.25(0.0118)(0) = 0$

$\Delta V_{12} = \alpha \delta_2 (x_1) = 0$

$\Delta V_{21} = 0.25(0.0118)(1) = 0.003$

$\Delta V_{22} = 0.25(\cancel{0.0118}\ 0.00247)(1) = 0.000617$

$\Delta V_{01} = 0.25(0.0118) = 0.003$

$\Delta V_{02} = 0.25(0.00247) = 0.000617$

Updated weights :-

Page No.
Date : / /

$V_{11} = 0.6 + 0 = 0.6$

$V_{12} = -0.3 + 0 = -0.3$

$V_{21} = -0.1 + 0.003 = -0.097$

$V_{22} = 0.4 + 0.0006 = 0.4006$

$V_{01} = 0.3 + 0.003 = 0.303$

$V_{02} = 0.5 + 0.0006 = 0.5006$

$W_{11} = 0.4 + 0.0165 = 0.4165$

$W_{21} = 0.1 + 0.0213 = 0.1213$

$W_0 = -0.2 + 0.03 = -0.17$

Scanned by CamScanner

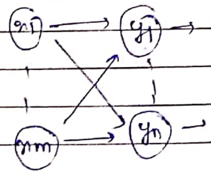
Associative Memories

two phases
1. store → [w] we need to find wt. ↓ ← x.
2. Recall → y
   ↳ we need to find out o/p

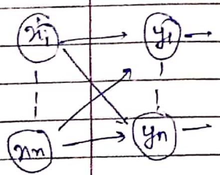
Two types of memories :- (based on structure)

Autoassociative Memories
↓
(same i/p & o/p) exactly same.
⇒ no. of i/p's = no. of o/p's.
→ No self loop.
∴ Diagonal elements in a matrix are zero.

Heteroassociative Memories.
↓
– no. of i/p's ≠ no. of o/p's.
→ nature of i/p is diff^n from nature of o/p.



Uses :- RAM And ROM in computers.



On addressing mode :-

Memories

Content addressable Memory (CAM)
Eg. → (naming in Phonebook)

Address addressable Memory (AAM)
Eg. → Connecting via address.
Example → Ptr.

• STORE PHASE

Calculating Values Of WT

1. Hebb Rule

$W_{new} = W_{old} + \Delta w$

$\Delta w = \alpha \cdot x \cdot y$
↓ Here, we always assume. * $\alpha = 1$

⇒ $\boxed{\Delta w = xy}$ [updates wt. individually]

2. Outer Product Rule

s t → o/p pattern.
↓ store pattern

$\boxed{w = s^T t}$ → o/p pattern matrix
↓ transpose of store matrix

[updates wt. in matrix form]

Page No.
Date: / /

• RECALL PHASE

activation f^n used strictly in auto associative memory,

$$Y = \begin{cases} 1 & ; y_{in} \geq 0 \\ -1 & ; y_{in} < 0 \end{cases}$$

In case of Heteroassociative memory

if, binary i/p's are there i.e. 0 and 1 then,

$$Y = \begin{cases} 1 & ; y_{in} > 0 \\ -1 & ; y_{in} \leq 0 \end{cases}$$

else,

$$Y = \begin{cases} 1 & ; y_{in} > 0 \\ 0 & ; y_{in} = 0 \\ -1 & ; y_{in} < 0 \end{cases}$$

Que Train autoassociative network for the i/p pattern,

[-1 1 1 1]

and test network,

a) for same i/p vector
b) with one missing entry
c) with one mistaken entry
d) two missing entries
e) two mistaken entries.

* IMP. If, it is not mentioned about self loop, then do not make diagonal matrix $=0$

Page No.
Date: / /

Ans. i/p & o/p are same.

Using outer product Rule.

$$W = \begin{bmatrix} -1 \\ 1 \\ 1 \\ 1 \end{bmatrix}_{4\times1} \begin{bmatrix} -1 & 1 & 1 & 1 \end{bmatrix}_{1\times4}$$

$$W = \begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}_{4\times4}$$

(a) i/p $= [-1 \quad 1 \quad 1 \quad 1]$

$$= [-1 \quad 1 \quad 1 \quad 1]_{1\times4} \begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}_{4\times4}$$

$$= \begin{bmatrix} -1-1-1-1 \\ 1+1+1+1 \\ 1+1+1+1 \\ 1+1+1+1 \end{bmatrix} = \begin{bmatrix} -4 \\ 4 \\ 4 \\ 4 \end{bmatrix} = [-4 \quad 4 \quad 4 \quad 4]$$

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

(same o/p)

⇒ training is correct

Page No.
Date : / /

Missing

one missing entry.
⇒ ~~if bipolar i/p's~~
⇒ we can't take bipolar
⇒ we'll take binary i.e. 0

Mistaken

any one bit with its wrong entry
i.e. if bipolar i/p's
then replace 1 with -1 or
-1 with 1

** Important

original → [-1 1 1 1]

(b) 1st bit = 0

[0 1 1 1]

$$y_{in} = [0\ 1\ 1\ 1]\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}$$

$$= [-3\ \ 3\ \ 3\ \ 3]$$

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

Page No.
Date : / /

(c) 1st bit = 1

[1 1 1 1]

$$y_{in} = [1\ 1\ 1\ 1]\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}$$

$$= [-2\ \ 2\ \ 2\ \ 2]$$

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

(d) 1st = 0 | last = 0

[0 1 1 0]

$$y_{in} = [0\ 1\ 1\ 0]\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}$$

$$= [-2\ \ 2\ \ 2\ \ 2]$$

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

(e) 1st = 1 , last = -1

[1 1 1 -1]

~~$y_{in} = [1\ 1\ 1\ -1]$~~
$$\begin{bmatrix} 1 & -1 & -1 & -1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix}$$

$$y_{in} = [0\ 0\ 0\ 0] \quad y = [-1\ -1\ -1\ -1]$$

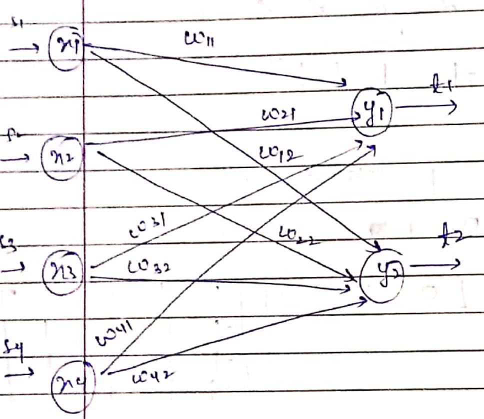
Scanned by CamScanner

* Mid Sem— Hetero associative Memory till,

Page No.
Date : / /

Q. Train and Test the following network using hetero associative memory.

s_1	s_2	s_3	s_4	t_1	t_2
1	0	0	0	0	1
1	1	0	0	0	1
0	0	0	1	1	0
0	0	1	1	1	0



(i) (store phase)

$$w = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix}^{T}_{4\times4} \begin{bmatrix} 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{bmatrix}_{4\times2}$$

Page No.
Date : / /

$$w = \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}_{4\times2} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \\ w_{31} & w_{32} \\ w_{41} & w_{42} \end{bmatrix}$$

$$\cancel{w_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}_{4\times1} \begin{bmatrix} 0 & 1 \end{bmatrix}_{1\times2} = \begin{bmatrix} \ \end{bmatrix}_{4\times2}}$$

Method → corresponding transpose i/p matrix, multiplied with o/p matrix & then corresponding sum all of them to get final matrix.

* Alternate

Acc. to book * for Binary

$$y = \begin{cases} 1 & y_{in} > 0 \\ 0 & y_{in} \le 0 \end{cases}$$

Now,

(I). $$y_{in} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}_{1\times4} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}_{4\times2}$$

$$y_{in} = \begin{bmatrix} 0 & 2 \end{bmatrix}_{1\times2}$$

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

Page No.
Date : / /

(2). $y_{in} = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$

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

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

(3). $y_{in} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$

$y_{in} = \begin{bmatrix} 2 & 0 \end{bmatrix}$

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

(4). $y_{in} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$

$y_{in} = \begin{bmatrix} 3 & 0 \end{bmatrix}$

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

Page No.
Date : / /

(b) For the above mentioned problem, test the heteroassociative memory with similar & unsimilar test vectors.

** IMP

(more than one bit say (2 bits))

changing one bit only, like mistaken. (in each i/p set)

A. Similar (1st bit)

s_1	s_2	s_3	s_4	t_1	t_2
0	0	0	0	0	1
0	1	0	0	0	1
1	0	0	1	1	0
1	0	1	1	1	0

$$y_{in} = \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$$

$$y_{in} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 2 & 2 \\ 3 & 2 \end{bmatrix} \Rightarrow y = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix} \begin{array}{l} \text{not} \\ \text{correct} \\ \text{not} \\ \text{not} \end{array}$$

unsimilar (2nd & 3rd bit)

Page No.
Date : / /

$$y_{in} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 2 \\ 0 & 1 \\ 1 & 0 \\ 2 & 0 \end{bmatrix}$$

$$y_{in} = \begin{bmatrix} 1 & 3 \\ 1 & 2 \\ 3 & 1 \\ 2 & 1 \end{bmatrix}$$

$$y = \begin{array}{c|cc} & t_1 & t_2 \\ 1. & 1 & 1 \\ 2. & 1 & 1 \\ 3. & 1 & 1 \\ 4. & 1 & 1 \end{array}$$

not matching in all training pair

Page No.
Date : / /

s_1	s_2	s_3	s_4	t_1	t_2
1	-1	-1	-1	-1	1
1	1	-1	-1	-1	1
-1	-1	-1	1	1	-1
-1	-1	1	1	1	-1

Test
(i) for same data
(ii) Test with missing data.
(iii) Test with mistaken data.

$$W = \begin{bmatrix} 1 & +1 & -1 & -1 \\ -1 & +1 & -1 & -1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix}_{4\times4} \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \\ 1 & -1 \end{bmatrix}_{4\times2}$$

$$W = \begin{bmatrix} -4 & 4 \\ -2 & 2 \\ 2 & -2 \\ 4 & -4 \end{bmatrix}_{4\times2}$$

$1+\cancel{1}-\cancel{1}+1$
$-1-1+\cancel{1}-\cancel{1}$
$1+1+1+1$
$-1-1-1-1$

(i)

$$y_{in} = \begin{bmatrix} 1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & -1 & -1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -4 & 4 \\ -2 & 2 \\ 2 & -2 \\ 4 & -4 \end{bmatrix}$$

$$= \begin{bmatrix} -8 & 8 \\ -12 & 12 \\ 8 & -8 \\ 12 & -12 \end{bmatrix}$$

$-4+\cancel{2}-\cancel{2}+4$
$4-\cancel{2}+\cancel{2}+4$
$-4-2-2-4$
$4+2+2+4$

Using activation fn

Page No.
Date : / /

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

(ii) 1st bit → 0

$$y_{in} = \begin{bmatrix} 0 & -1 & -1 & -1 \\ 0 & 1 & -1 & -1 \\ 0 & -1 & -1 & 1 \\ 0 & -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -4 & 4 \\ -2 & 2 \\ 2 & -2 \\ 4 & -4 \end{bmatrix}$$

$$y_{in} = \begin{bmatrix} -4 & 4 \\ -8 & 8 \\ 4 & -4 \\ 8 & -8 \end{bmatrix}$$

$\cancel{2} \cancel{-2} -4$
$-\cancel{2} \cancel{+2} +4$
$-2 -2 -4$
$-4-4=-8$
$2 +2 +4$
$\cancel{2} \cancel{-2} +4 \quad -2 +2 -4$
$2 +2 +4 \quad -2 -2 -4$

$$y = \begin{bmatrix} -1 & 1 \\ -1 & 1 \\ 1 & -1 \\ 1 & -1 \end{bmatrix}$$ Correct.

(iii) 2nd bit mistaken.

if not mentioned then by default, only one bit

Page No.
Date : / /

$$y_{in} = \begin{bmatrix} 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & -1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -4 & 4 \\ -2 & 2 \\ 2 & -2 \\ 4 & -4 \end{bmatrix}$$

$$y_{in} = \begin{bmatrix} -12 & 12 \\ -8 & 8 \\ 4 & -4 \\ 8 & 0 \end{bmatrix}$$

$-4 -2 -2 -4$
$4 +2 +2 +4$
$-4 \cancel{+2} \cancel{-2} -4$
$4 \cancel{-2} \cancel{+2} +4$
$4 -2 -2 +4$
$2 +2 = 4$
$-4 +2 +2 -4$
$-2 -2$
$4 -2 +2 +4$
$-4 \cancel{+2} \cancel{-2} -4$

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

→ not matching.

Syllabus Midterm

Ch 1 — 1.1, 1.2, 1.6 (flowchart also, Algo, Short-notes.)
Ch 2 —
Ch 3 — (3.1, 3.2, 3.5, 3.3, 3.4)
Ch 4 — (4.1 to 4.4)

(draw architecture.
write activation fn.)