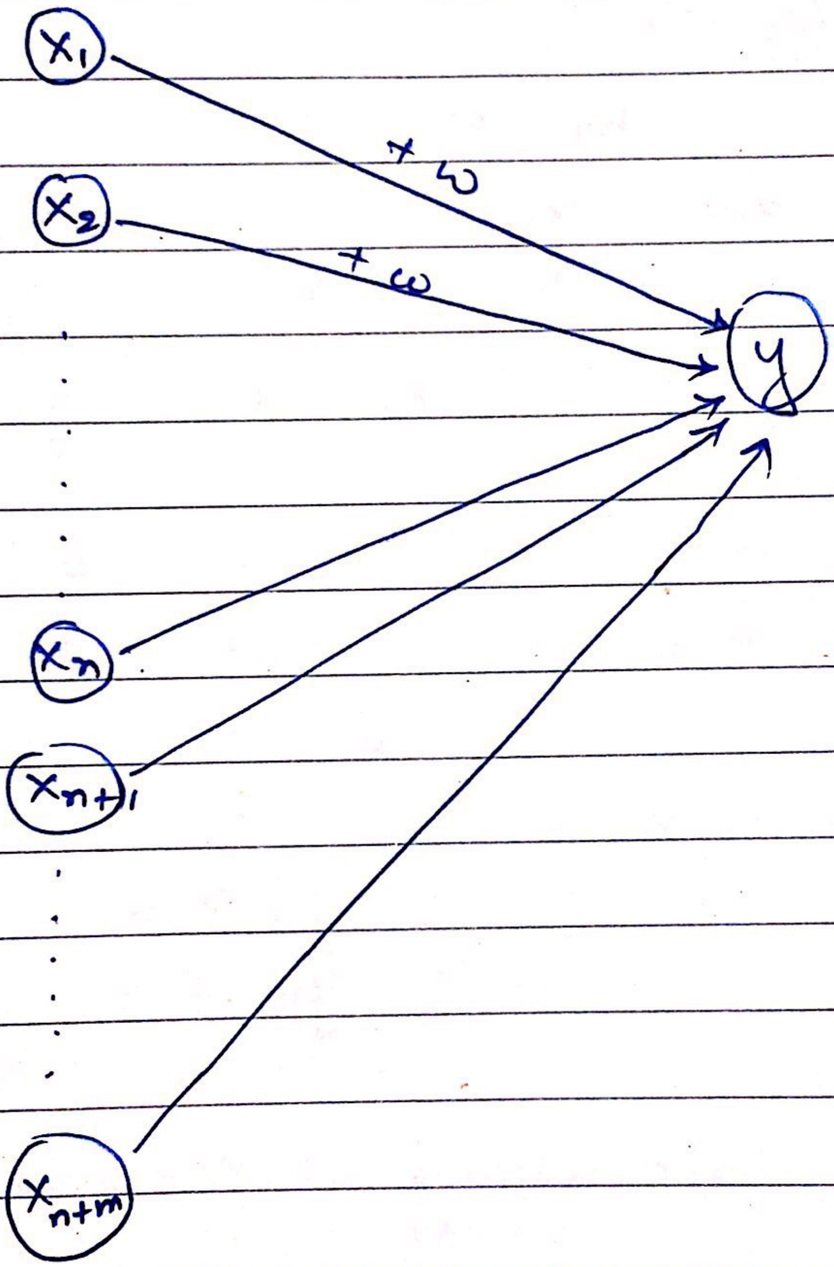
principles of soft computing with
↑ MATLAB

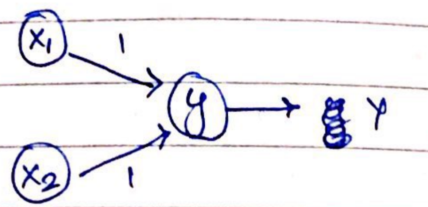
16 / 08 / 18

Book - principles of soft computing (Sivanandam & Deepa)

Mc. Cullock Pitts Model (MCP)



AND Gate



$w_1 = w_2 = 1$

$y_{in1} = x_1 w_1 + x_2 w_2$
$= 1 + 1 = 2$

1) [1 1] → 2
2) [1 0] → 1
3) [0 1] → 1
4) [0 0] → 0

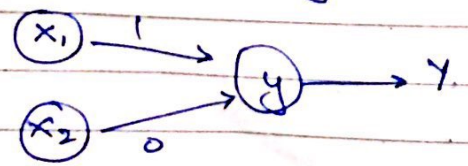
x_1	x_2	y
1	1	1
1	0	0
0	1	0
0	0	0

threshold → 2
θ

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

$Y = f(y)$

Realise OR Gate using MCP model



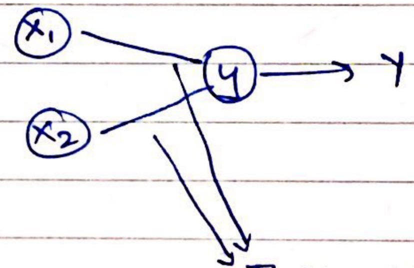
$w_1 = 1$

$w_2 = 1$

1) [1 1] → 2
2) [1 0] → 1
3) [0 1] → 1
4) [0 0] → 0

$\theta \to 1$
(threshold)

AND NOT Gate (x_1 and $\bar{x}_2$)



These weights can't have different value (sign can be different)

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

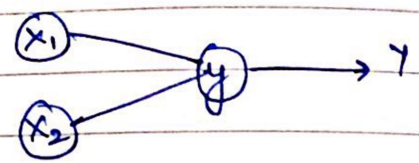
With weights as 1 and 1 we see that koi threshold ni mila

$w_1 = 1$ $w_2 = -1$

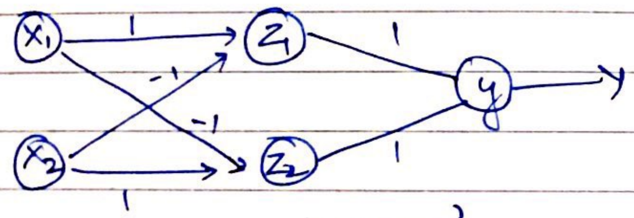
[1 1] → 0
[0 1] → -1
[1 0] → 1
[0 0] → 0

$\theta = 1$

XOR Gate



For XOR gate, we need to change the architecture



AND NOT Gates (2) — OR Gate

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

x_1 and not x_2

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

~~x_1~~	~~x_2~~	~~z_1 OR z_2~~
~~0~~	~~0~~	~~0~~
~~0~~	~~1~~	
~~1~~	~~0~~	
~~1~~	~~1~~	

x_2 and not x_1

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

x_1	x_2	z_1 OR z_2 (y)
0	0	0
0	1	1
1	0	1
1	1	0

1) [0 0] $z_1 \to 0(0)$ $y_{in} \to 0$
$z_2 \to 0(0)$

2) [0 1] $z_1 \to -1(0)$ $y_{in} \to 1$
$z_2 \to 1(1)$

3) [1 0] $z_1 \to 1(1)$ $y_{in} \to 1$
$z_2 \to -1(0)$

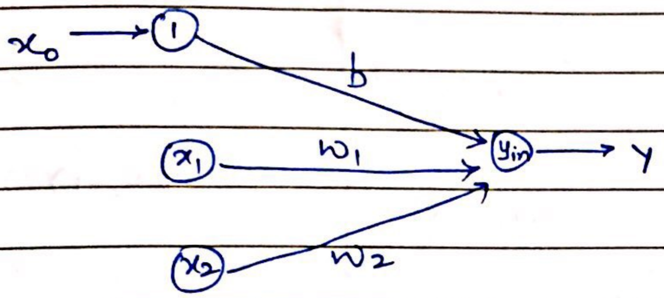
4) [1 1] $z_1 \to 0(0)$ $y_{in} \to 0$
$z_2 \to 0(0)$

θ for and not gates $\to 1$
θ for OR gate $\to 1$

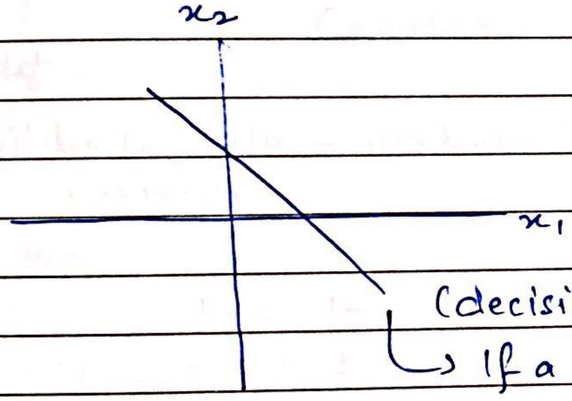
20/08/18

Linear Separability

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



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



(decision boundary)
→ If a line like this separates one set of inputs from the other, they are called linearly separable

If (x_1, x_2) lies on decision boundary $y_{in} = 0$

$$b + x_1 w_1 + x_2 w_2 = 0 \quad (1)$$

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

(w_2 should be a non-zero value for this)

Compare with

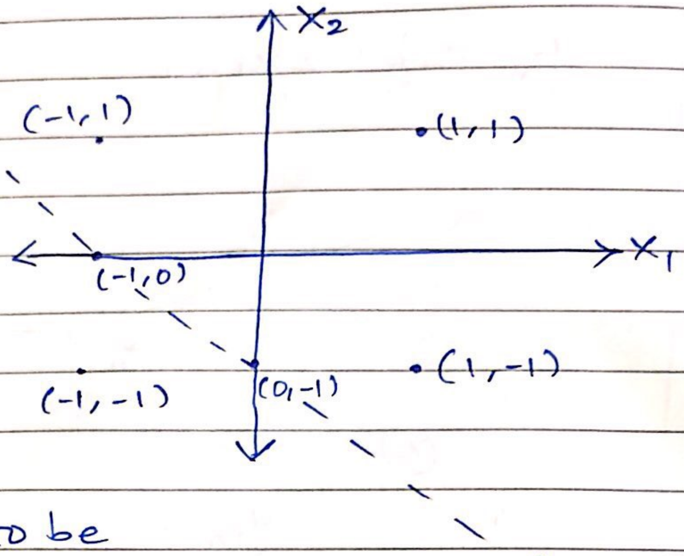
$$y = mx + c$$

Eq^n for decision boundary

20/08/18

Q Use the concept of linear separability to check whether an OR Gate is linearly separable or not

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



For the concept of LS, input & outputs have to be bipolar (not binary in nature)

(we can take any line)

Decision boundary → also called linear classifier

$$m = \frac{-1-0}{0-(-1)} = \frac{-1}{1} = -1$$

$$y_1 = mx_1 + c$$

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

$$y = -x - 1 \qquad y = -x - c \rightarrow +c?$$

$$x + y + 1 = 0$$

↓

Compare with 1

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

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

$$w_1 = w_2 = b = 1$$

We can take any value, but we prefer to take the simplest

20/08/18

x_1	x_2	b	$y_{in} = b + x_1 w_1 + x_2 w_2$	
1	1	1	3	1
1	-1	1	1	1
-1	1	1	1	1
-1	-1	1	-1	-1

(θ) threshold $= 1$

Activation function : $\underbrace{f(y_{in})}_{y} = \begin{cases} 1 & y_{in} \geq 1 \\ -1 & y_{in} < 1 \end{cases}$

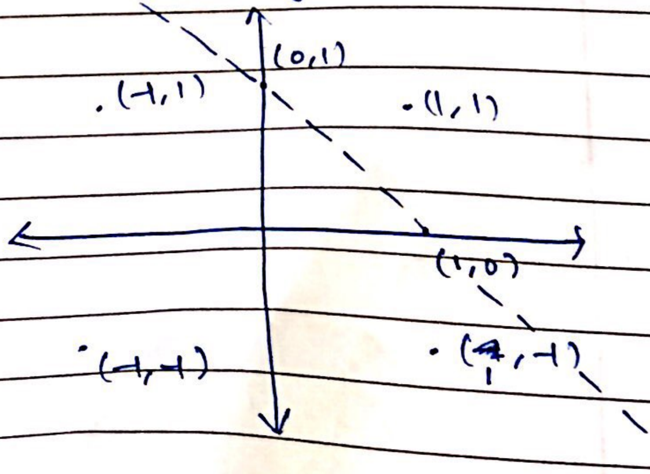
$Y_1 = 1$, $Y_2 = 1$,
$Y_3 = 1$, $Y_4 = -1$

<u>In exam</u>

- draw graph
- write eq. for decision boundary
- find out w_1, w_2, b
- write θ
- write Activation function
- Make table

Q. Use the concept of Linear Separability to check whether an AND Gate is linearly separable or not.

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



20/08/18

$$m = \frac{0-1}{1-0} = -1$$

$y_1 = mx_1 + c$

$0 = -1(1) + c$

$c = 1$

$y = -1 \cdot x + 1$

$x + y = 1$

$y = -x + c$

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

$\frac{-b}{w_2} = 1 \Rightarrow b = -w_2$

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

x_1	x_2	b	y_{in}	
1	1	1	-1	} Y = 1
1	-1	1	1	} Y = -1
-1	1	1	1	
-1	-1	1	3	

$\theta = 1$

$$f(y_{in}) = \begin{cases} 1 & y_{in} < 1 \\ -1 & y_{in} > 1 \end{cases}$$

23/08/18

Learning Rules

1) Hebb Rule (Unsupervised learning rule)

$$W_{new} = W_{old} + \Delta W$$
$$\Delta W = xy$$

→ The initial values of weights is 0
→ This rule uses unsupervised learning

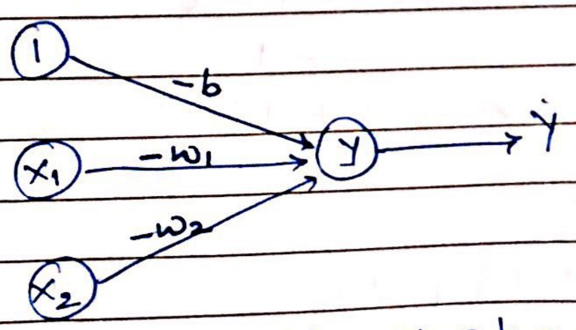
Q. Create a hebb net for AND Gate

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

s			t
x_1	x_2	b	y
[1	1	1]	1
1	-1	1	-1
-1	1	1	-1
-1	-1	1	-1

Imp: When doing question on learning rules, always take bias

When we train our system once with all the inputs, it is called 1 epoch.



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

23/08/18

$s : t$

(a) $[1 \quad 1 \quad 1] : [1]$

$$[\underset{x_1}{1} \quad \overset{x_2}{1} \quad \underset{b}{1}] \quad [\underset{y}{1}]$$

$$w_{1new} = w_{1old} + \Delta w_1$$
$$= 0 + x_1 y$$
$$= 0 + 1 \times 1 = 1$$
$$w_{2new} = w_{2old} + \Delta w$$
$$= 0 + x_2 y$$
$$= 0 + 1 \times 1 = 1$$
$$b_{new} = b_{old} + \Delta b$$
$$= 0 + 1 \times 1$$
$$= 1$$

(b) $[1 \quad -1 \quad 1] \quad [-1]$

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

(c) $[-1 \quad 1 \quad 1] \quad [-1]$

$$w_1 new = 0 + (-1)(-1)$$
$$= 1$$
$$w_2 new = 2 + 1(-1)$$
$$= 1$$
$$bnew = 0 + 1(-1)$$
$$= -1$$

(d) $[-1 \quad -1 \quad 1] \quad [-1]$

$$w_1 new = 1 + (-1)(-1)$$
$$2$$

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

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

Q Design a hebb net for patterns L and U where L ∈ class1 and U ∈ class2

×	•	•		×	•	×		$\times \rightarrow 1$
×	•	•		×	•	×		$\bullet \rightarrow -1$
×	×	×		×	×	×		
	(L)				(U)			

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	b	y
L	1	-1	-1	1	-1	-1	1	1	1	1	1
U	1	-1	1	1	-1	1	1	1	1	1	-1

$w_{1_{old}} \cdots w_{9_{old}} = 0 \quad b_{old} = 0$

	b_{new}	x_1 new (w_{1new})	x_{2new} (w_{2new})	x_{3new} (w_{3new})	x_{4new} (w_{4new})	x_{5new} (w_{5new})	x_{6new} (w_{6new})	x_{7new} (w_{7new})	x_{8new} (w_{8new})	x_{9new} (w_{9new})
	1	1	-1	-1	1	-1	-1	1	1	1
Final →	0	0	0	-2	0	0	-2	0	0	0

29/08/18

Perceptron Learning

→ Perceptron Rule is a supervised learning rule.

→ Hebb Rule was unsupervised learning

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

$$\Delta W = \alpha t x$$ (In hebb rule it was xy)

$\alpha \rightarrow$ Learning signal

Imp. If value of α is not given assume it as 1.

$t \rightarrow$ target output (the output we want)

$x \rightarrow$ designated input

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

$$\Delta b = \alpha t$$

In hebb rule we were using bipolar inputs.

Here we can use bipolar & binary inputs

In case of bipolar inputs we can always take activation function as

$$f(y_{in}) = \begin{cases} 1 & y_{in} > \theta \\ 0 & 0 \le y_{in} \le \theta \\ -1 & y_{in} < \theta \end{cases}$$ Ramp function

Q. Implement AND Gate using perceptron learning rule given initial weights as 0, threshold = 0 & learning rate = 1

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

$\theta = 0$

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

1. [1 1] [1]

$y_{in} = x_1 w_1 + x_2 w_2$
$= 0$

$Y = f(0) = 0$

↳ not equal to the target value (1) ⇒ this set of weights is incorrect

$w_{1new} = 0 + 1 \cdot 1 \cdot 1$
$= 1$

$w_{2new} = 0 + 1 \cdot 1 \cdot 1$
$= 1$

2. [1 -1] [-1]

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

$Y = 0$

$w_{1new} = 1 + 1 \cdot (-1) \cdot (1)$
$= 0$

$w_{2new} = 1 + 1 \cdot (-1)(-1)$
$= 2$

3. [-1 1] [-1]

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

$= 2$

$Y = 1$

29/08/18.

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

4. $[-1 \;\; -1] \; [-1]$

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

$Y = -1$ ✓

1 epoch

This set of weights is correct
If it were incorrect and we had to solve for more than 1 epoch, we would do the process all over again

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

Q. Find the weights required for the given training target pair where learning rate = 1, initial weights = 0, bias = 0. Do this process till 1 epoch

$(1,1,1,1) \; ; \; (-1,1,-1,-1) \rightarrow +1$
$(1,1,1,-1) \; ; \; (1,-1,-1,1) \rightarrow -1$

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

	x_1	x_2	x_3	x_4	b	t
1.	1	1	1	1	0	1
2.	-1	1	-1	-1	0	1
3.	1	1	1	-1	0	-1
4.	1	-1	-1	1	0	-1

29/08/18

1. [1 1 1 1 0] [1]

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

$Y = 0$

$w_1 new = 0 + 1 \cdot 1 \cdot 1$
$= 1$

$w_2 new = w_3 new = w_4 new = 1$

$b new = 0 + 1 \cdot 1 = 1$

2. [-1 1 -1 -1 1] [1]

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

$Y = -1$

$w_1 new = 1 + 1 \cdot (1)(-1)$
$= 0$

$w_3 new = 1 + 1(1)(-1)$
$= 0$

$w_2 new = 1 + 1 \cdot (1)(1)$
$= 2$

$w_4 new = 0$

$b new = 1 + 1 \cdot 1$
$= 2$

3. [1 1 1 -1 2] [-1]

$y_{in} = 0 + 2 + 0 + 0 + 2$
$= 4$

$Y = 1$

$w_1 new = 0 + 1(-1)(1)$
$= -1$

$w_3 new = 0 + 1(-1)(1)$
$= -1$

$w_2 new = 2 + 1(-1)(1)$
$= 1$

$w_4 new = 0 + 1(-1)(-1)$
$= 1$

$b new = 2 + 1(-1) = 1$

29/08/18

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

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

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

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

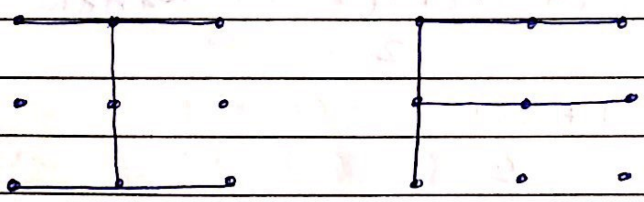
$w_{3new} = -1 + 1(-1)(-1)$
$= 0$

$w_{4new} = 1 + 1(-1)(1)$
$= 0$

$b_{new} = 1 + 1 \cdot (-1)$
$= 0$

30/08/18

Q. Classify 2-dimensional input pattern ~~E~~ I & F using perceptron network, initial weights are 0, $\theta = 0$, $\alpha = 1$. Pattern I $\in$ class 1 and pattern F $\in$ class 2



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

I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	b	O
1	1	1	-1	1	-1	1	1	1	0	1
1	1	1	1	1	1	1	-1	-1	0	-1

1. $[1\ 1\ 1\ -1\ 1\ -1\ 1\ 1\ 1\ 0]\ [1]$

$$y_{in} = 0$$
$$Y = 0$$

$w_{1_{new}} = 0 + (1)(1)(1) = 1$

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

$b_{new} = 0 + (1)(1) = 1$

30/08/18

2. [1 1 1 1 1 1 1 -1 -1 0] [-1]

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

+1

$= 2$

$Y = 1$

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

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

$w_{3new} = 1 + 1(-1)(1) = 0$

$w_{4new} = -1 + 1(-1)(1) = -2$

$w_{5new} = 1 + 1(-1)(1) = 0$

$w_{6new} = -1 + 1(-1)(1) = -2$

$w_{7new} = 1 + 1(-1)(1) = 0$

$w_{8new} = 1 + 1(-1)(-1) = 2$

$w_{9new} = 1 + 1(-1)(-1) = 2$

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

Q. Implement OR Gate with binary inputs & bipolar targets using perceptron algorithm for 2 epochs where the initial weights are 0, $\alpha = 1$ and $\theta = 0.2$.

$$Y = \begin{cases} 1 & y_{in} \geq 0.2 \\ 0 & y_{in} < 0.2 \end{cases}$$

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

1. [0 0 0] [-1]

$y_{in} = 0$

$Y = 0$

30/08/18

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

2. [0 1 0] [1]

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

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

3. [1 0 0] [1]

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

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

4. [1 1 0] [1]

$y_{in} = 1 + 1 + 0 = 2$
$y = 1$

2nd epoch.

5. [0 0 0] [-1]

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

30/08/18.

$w_1 new = 1 + 1(-1)(0)$
$= 1$
$w_2 new = 1 + 1(-1)(0)$
$= 1$
$b new = 1 + 1(-1)$
$= 0$

6. [0 1 0] [1]
$y_{in} = 0 + 1 + 0 = 1$
$Y = 1$

7. [1 0 0] [1]
$y_{in} = 1 + 0 + 0$
$= 1$
$Y = 1$

8. [1 1 0] [1]
$y_{in} = 1 + 1 + 0$
$= 2$
$Y = 1$

$w_1 = 1 , \; w_2 = 1 , \; b = 0$

Imp: Follow this in exam :

		bias	t
1	1	1	
1	-1	1	
-1	1	1	
-1	-1	1	

6/9/18

Missed a class (4 Pics) – Delta Learning Rule

Bunk Pages - The Social Notebook

Q. Use $\boxed{\text{Adaline N/w}}$ to train AND NOT Function with bipolar inputs & targets. The initial weights & bias have value 0.2 & the learning rate is 0.2. Train the network for 2 epochs.

$\rightarrow x_1 \bar{x}_2$

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

1. $[1 \; 1 \; 1] : [-1]$

$y_{in} = 1 \times 0.2 + 1 \times 0.2 + 1 \times 0.2$
$= 0.6$

$E = (-1 - 0.6)^2 = (-1.6)^2 = 2.56$

$w_{1new} = \cancel{0.2 + 0.2 \times 2.56 \times (-1)}$
$\cancel{= -0.312}$

$= w_{1old} + \alpha (t - y_{in}) x_1$
$= 0.2 + 0.2(-1 - 0.6)(1)$
$= 0.2 - 0.32 = -0.12$

$w_{2new} = 0.2 + 0.2(-1 - 0.6)(1)$
$= -0.12$

$b_{new} = 0.2 + 0.2(-1 - 0.6)$
$= -0.12$

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

$y_{in} = 1 \times (-0.12) - 1 \times (-0.12) + 1 \times (-0.12)$
$= -0.12$

$E = (1 + 0.12)^2 = 1.2544$

$w_{1new} = -0.12 + 0.2(1 + 0.12)(1) = 0.104$

Ans till 2 decimal places

$w_{2new} = -0.12 + 0.2(1+0.12)(-1)$
$= -0.344$

$b_{new} = -0.12 + 0.2(1+0.12)$
$= 0.104$

③ $[-1 \quad 1 \quad 1] : [-1]$

$y_{in} = (-1) \times (0.104) + 1 \times (-0.344) + 1 \times (0.104)$
$= -0.344$

$E = (-1+0.344)^2 = 0.43$

$w_{1new} = 0.104 + 0.2(-1+0.344)(-1)$
$= 0.2352$

$w_{2new} = -0.344 + 0.2(-1+0.344)(1)$
$= -0.4752$

$b_{new} = 0.104 + 0.2(-1+0.344)$
$= \cancel{0.3728} \quad -0.0272$

④ $[-1 \quad -1 \quad 1] : [-1]$

$y_{in} = -(1 \times 0.2352) - 1 \times (-0.4752) + 1 \times \cancel{0.3728}$ -0.0272
$= \cancel{0.6128} \quad 0.2128$

$E = (-1-0.2128)^2 = \cancel{2.6} \quad 1.47$

$w_{1new} = 0.2352 + 0.2(-1-0.2128)(-1)$
$= \cancel{0.21} \quad 0.47$

$w_{2new} = -0.4752 + 0.2(-1-0.2128)(-1)$
$= -0.23$

$b_{new} = -0.0272 + 0.2(-1-0.2128)$
$= -0.269$

1 epoch completed

6/9/18

(5) $[1 \; 1 \; 1] : [-1]$

$$y_{in} = 0.47 + (-0.23) + (-0.26)$$
$$= -0.02$$
$$E = (-1+0.02)^2 = 0.96$$

$$w_{1new} = 0.47 + 0.2(-1+0.02)$$
$$= 0.27$$
$$w_{2new} = -0.23 + 0.2(-1+0.02)$$
$$= -0.42$$
$$b_{new} = -0.26 + 0.2(-1+0.02)$$
$$= -0.45$$

(6) $[1 \; -1 \; 1] : [1]$

$$y_{in} = 0.27 + 0.42 - 0.45$$
$$= 0.24$$
$$E = (1-0.24)^2 = 0.577$$
$$w_{1new} = 0.27 + 0.2(1-0.24)(1)$$
$$= 0.422$$
$$w_{2new} = -0.42 + 0.2(1-0.24)(-1)$$
$$= -0.572$$

w