AIML

Artificial Intelligence And Machine Learning

Page No. Date

20/4/22

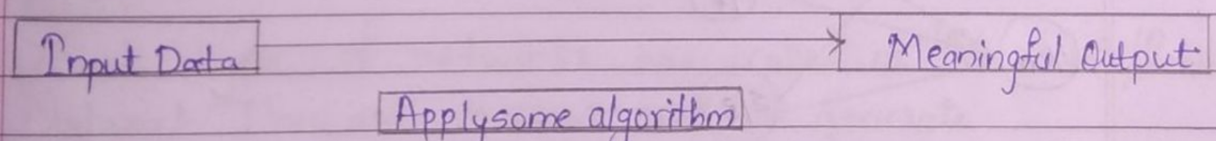
* Unit 3 Artificial Neural Network (ANN)
- Neural → comes from Neurons
- Network → Neurons are connected with each other
- Artificial → Manmade.

* Artificial Neural Network

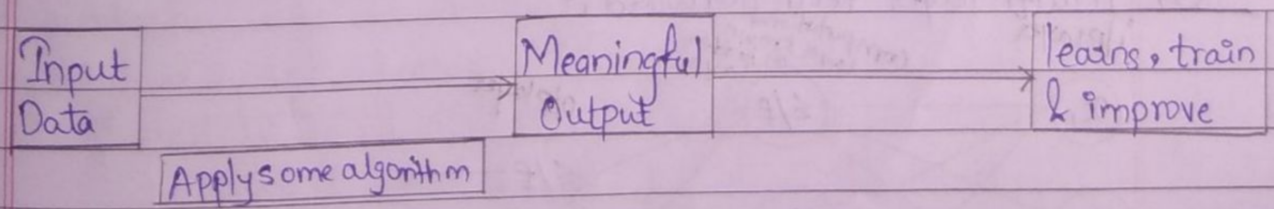
1. The visual information of an object is to be carried out to brain to recognized an object with the help of neurons.
2. We try to create a machine in such a way that a machine should have neurons (Artificial neurons) and they will be connected to each other in same manner as we do have neurons in our brain. Those artificial neurons will train & learn data to generate better output.

* Why Neuron Network

- Normal Computer



- Computer with ANN



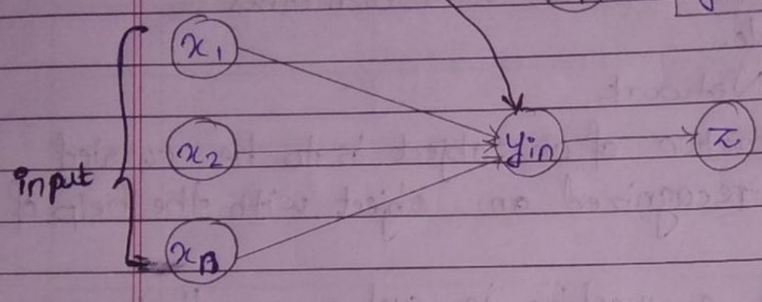
Page No. | Date

* Basic structure of ~~Neural~~ Neural Network

Net input also written as Σ/f

$y_{in} = x_1 \times w_1 + x_2 \times w_2 + x_3 \times w_3$

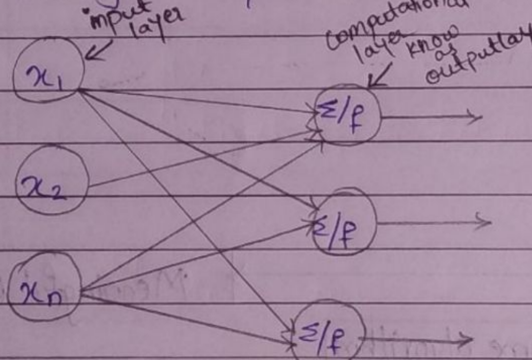
$\boxed{y_{in} = \sum_{i=1}^{n} x_i w_i}$



$\boxed{z = f(y_{in})}$ ← activation function.

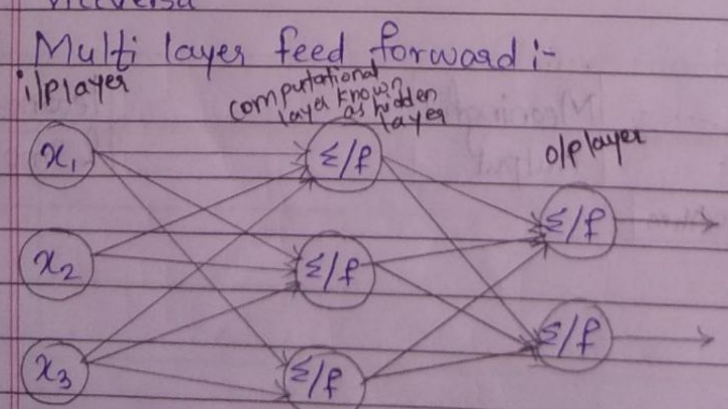
* What are the models of neural networks?

Ans. (1) Single layer feed forward Network :-



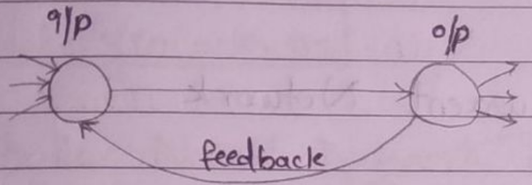
- There is only one computational layer that is output layer so it is a single layer network
- Input layer forward data to output layer only not viceversa

(2) Multi layer feed forward :-

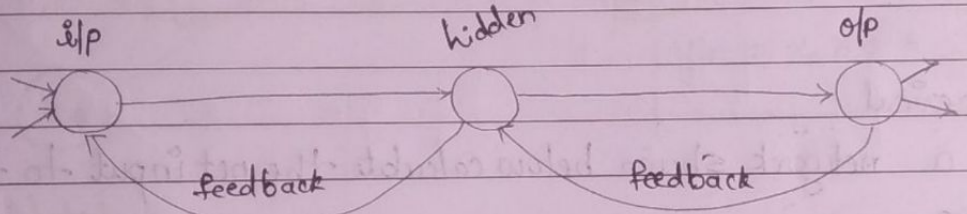


- Multi layered means multiple numbers of layers or more than one computational layer and there will be hidden layers which work as computational layer.

(3) Single node with its own feedback:-

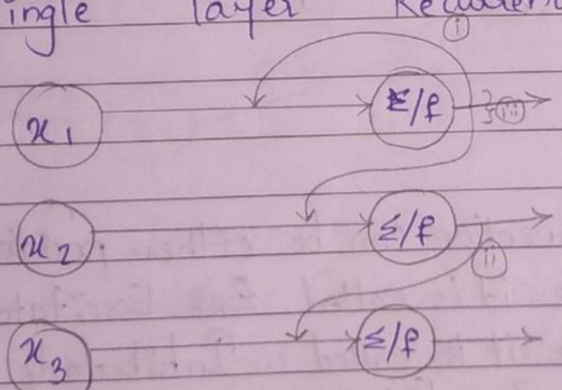


- Signals travels from Input node to Output node here signals can comeback to Inputnode that is direction of signals are allowed in both ways.



Note:-
Such type of networks are called as Recurrent/Recursive Network. They are complex used to generate

(4) Single layer Recurrent Network:-



- Single layer Recurrent Network is a network with feedback connection in which processing elements output can be directed back

Conditions:-

(i) To processing element itself

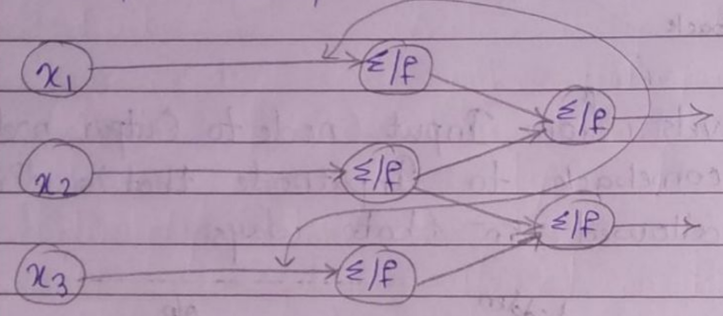
OR

(ii) To other processing element

OR

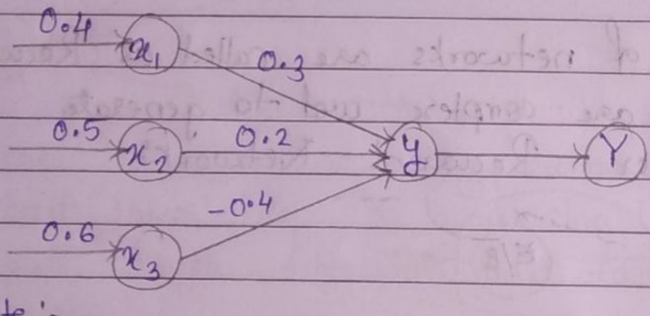
(iii) To both

(5) ~~Multipa~~ Multilayer Recurrent Network



* Numerical

5mrks Q For a network shown below calculate the net input to the output neuron



Note:-

Weights of the connection can be either positive or negative. if the weight is positive it is called ~~Exi~~ Excitatory

if the weight is negative it is called Inhibatory

Sol^n Inputs => $[x_1 \ x_2 \ x_3] = [0.4 \ \ 0.5 \ \ 0.6]$

Weights => $[w_1 \ w_2 \ w_3] = [0.3 \ \ 0.2 \ \ -0.4]$

∴ The net input can be calculated as

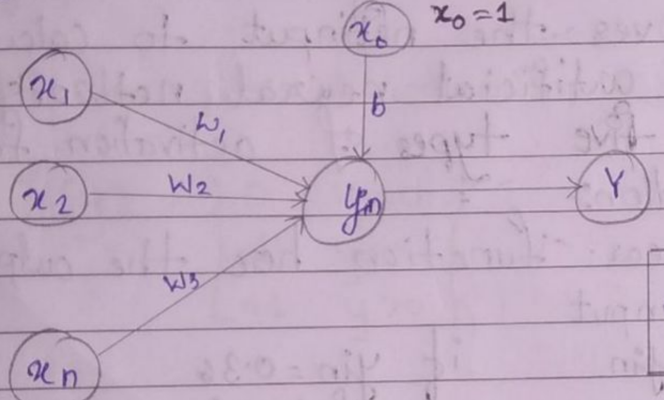
$$y_{in} = \sum_{i=1}^{n} x_i w_i$$

$$= x_1 w_1 + x_2 w_2 + x_3 w_3$$
$$= 0.4(0.3) + 0.5(0.2) + 0.6(-0.4)$$
$$= 0.12 + 0.10 - 0.24$$
$$= 0.22 - 0.24$$
$$= -0.02$$

$$\boxed{\therefore y_{in} = -0.02}$$

* Basic Neural Network with Biased connection.

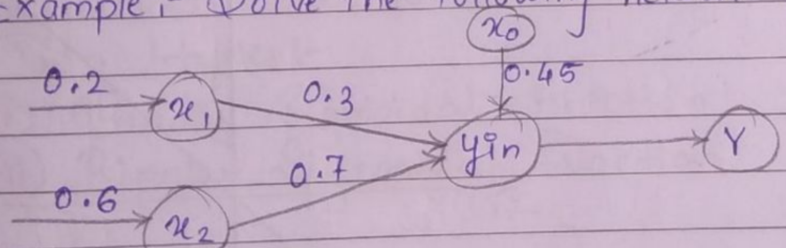
$x_0 = 1$



$$y_{in} = x_0 \times b + \sum_{i=1}^{n} x_i w_i$$

Note:-
The Input value of biased connection will always be 1

* Example:- Solve the following network to calculate net input



Sol^n Given = inputs $= [x_1 \; x_2 \; x_0] = [0.2 \;\; 0.6 \;\; 1]$, $b = 0.45$

weights $= [w_1 w_2] = [0.3 \; 0.7]$

$\therefore$ The net input can be calculated as

$$y_{in} = x_0 \times b + \sum_{i=1}^{n} x_i w_i$$
$$= 1 \times 0.45 + 0.2 \times 0.3 + 0.6 \times 0.7$$
$$= 0.45 + 0.6 + 0.42$$
$$= 0.93$$

$$\boxed{\therefore y_{in} = 0.93}$$

21/4/22

8 to 10 marks:- * What is activation function and what are different types of activation functions

Ans. • Activation function:-

Let us assume a person is performing some work to make the work more efficient and to get exact output some force or activation may be given this activation helps in achieving the exact output. In a similar way the activation function is applied over the netinput to calculate the output of an artificial neural network.

• There are five types of activation function

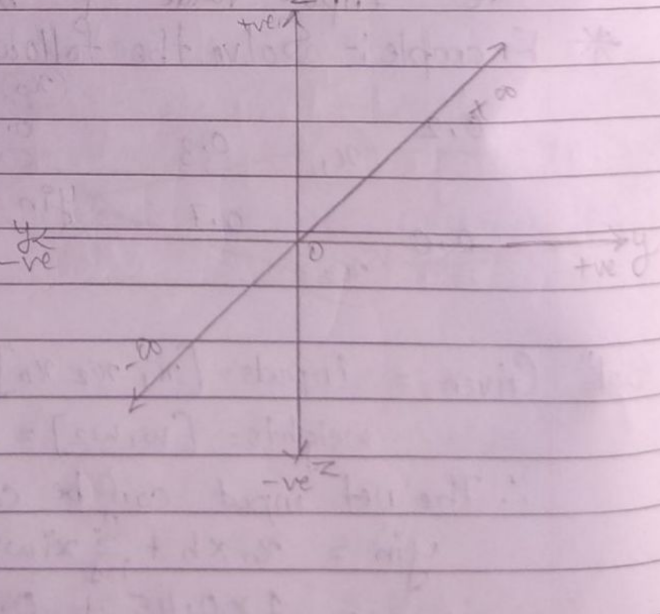
(1) Identity function:-

It is a Linear function here the output remains same as the input

$f(y_{in}) = y_{in}$ if $y_{in} = 0.36$

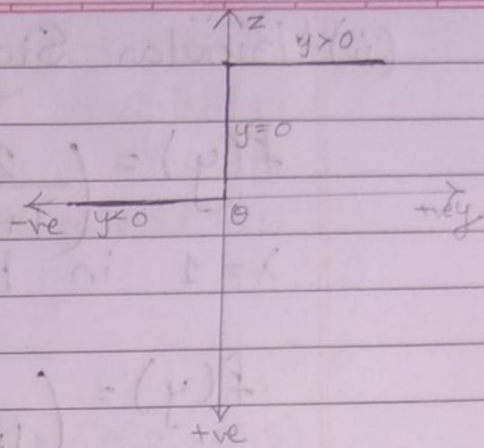
$f(y) = y$ $z = f(y_{in}) = 0.36.$

range = $-\infty$ to +ve



(2) Binary Step function:-

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



θ = threshold value

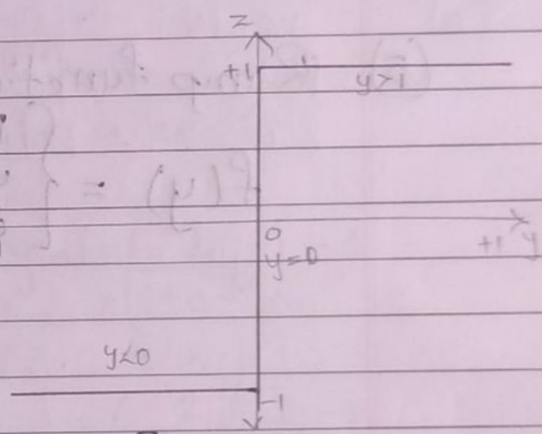
if y=0 then z=1

if y<0 then y=-1

if y>0 then z=1 & y=1

(3) Bipolar Step function:-

$$f(y) = \begin{cases} 0 & y = 0 \\ -1 & y < 0 \\ +1 & y > 0 \end{cases}$$



(4) Sigmoidal function:- [Mostly used]

Two types:-

(i) Binary Sigmoidal function

(ii) Bipolar Sigmoidal function.

(i) Binary Sigmoidal function

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

$\lambda = 1$ in basic neural network

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

(ii) Bipolar Sigmoidal function:-

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

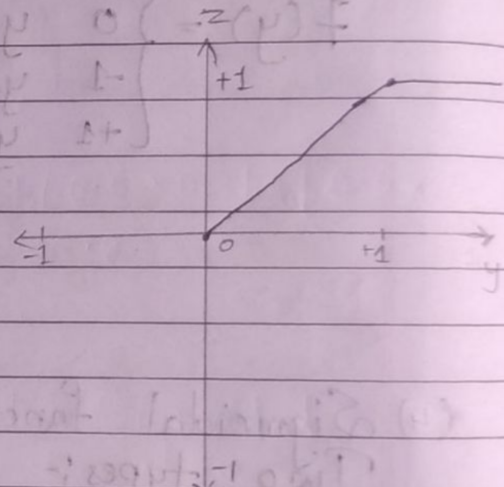
$\lambda = 1$ in basic neural network

$$f(y) = \left(\frac{2}{1+e^{-y}}\right) - 1$$

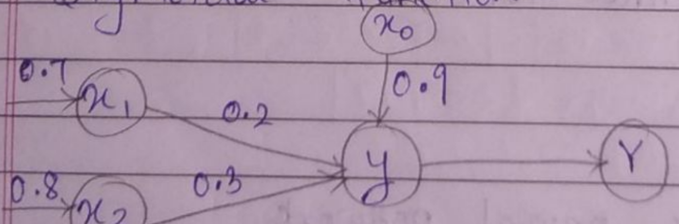
where λ is steepness parameter
$\lambda = 1$ in basic neural network always.

(5) Ramp function:-

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



* Calculate the output of the neuron y for the net given below use binary and bipolar sigmoidal function



$x_0 = 1$ always.

Page No.
Date

Sol^n Inputs $\Rightarrow [x_1 \;\; x_2] = [0.7 \;\; 0.8]$
Weights $\Rightarrow [w_1 \;\; w_2] = [0.2 \;\; 0.3]$
bias $\Rightarrow [b] \; [0.9]$
$\therefore$ The net input can be calculated as.

$$y_{in} = x_0 \times b + \sum_{i=1}^{n} x_i w_i$$

$$= 1 \times 0.9 + (0.7 \times 0.2) + (0.8 \times 0.3)$$

$$= 0.9 + 0.14 + 0.24$$

$$\therefore y_{in} = 1.28$$

(i) Binary Sigmoidal

$\because y_{in} = 1.28$

$$f(y_{in}) = \frac{1}{1+e^{-y_{in}}}$$

$$= \frac{1}{1+e^{-1.28}}$$

$$= \frac{1}{1+0.2780}$$

$$= \frac{1}{1.2780}$$

$$f(y_{in}) = 0.7824$$

(ii) Bipolar Sigmoidal

$\because y_{in} = 1.28$

$$f(y_{in}) = \left(\frac{2}{1+e^{-y_{in}}}\right) - 1$$

$$= \left(\frac{2}{1+e^{-1.28}}\right) - 1$$

$$= \left(\frac{2}{1+0.2780}\right) - 1$$

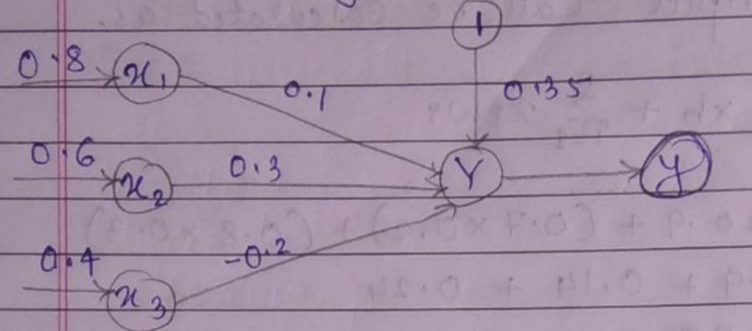
$$= \left(\frac{2}{1.2780}\right) - 1$$

$$= 1.5649 - 1$$

$$f(y_{in}) = 0.5649$$

Page No.
Date

Q. Obtain the output of neuron of the following net given below using Binary Sigmoidal & Bipolar Sigmoidal activation function.



Sol^n Inputs $\Rightarrow [0.8 \quad 0.6 \quad 0.4]$
Weights $\Rightarrow [0.1 \quad 0.3 \quad -0.2]$
bais $\Rightarrow [0.35]$

$$y_{in} = 0.35 \times 1 + [0.8 \times 0.1] + [0.6 \times 0.3] + [0.4 \times (-0.2)]$$
$$= 0.35 + 0.08 + 0.18 - 0.8$$
$$\therefore y_{in} = 0.53$$

(i) Binary Sigmoidal	(ii) Bipolar Sigmoidal
$= \dfrac{1}{1+e^{-y_{in}}}$	$\left(\dfrac{2}{1+e^{-y_{in}}}\right) - 1$
$= \dfrac{1}{1+e^{-0.53}}$	$\left(\dfrac{2}{1+e^{-0.53}}\right) - 1$
$= \dfrac{1}{1+0.5886}$	$\left(\dfrac{2}{1+0.5886}\right) - 1$
$= \dfrac{1}{1.5886}$	$= \left(\dfrac{2}{1.5886}\right) - 1$
$f(y_{in}) = 0.6294$	$= 1.2589 - 1$
	$f(y_{in}) = 0.2589$

* Models:-

(1) Mc Culloc Pitt

7 to 8 marks Q. Implement **AND** function using Mc Culloc Pitt neuron network.

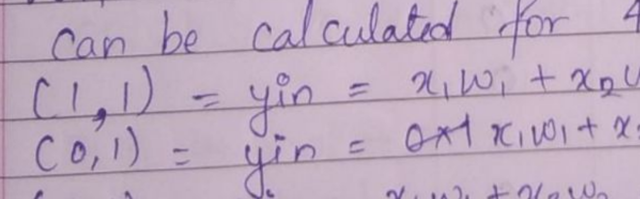
Sol^n: The truth table for AND function is :-

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

Let us assume the weights be $w_1 = 1$, $w_2 = 1$

Note :-
In Mc Culloc Pitt Neuron only analysis is being performed.

Hence we can draw the basic neural network according to the basic neural network



With the above assumed weights the net input can be calculated for 4 inputs as follows:-

$(1,1) = y_{in} = x_1 w_1 + x_2 w_2 = 1\times1 + 1\times1 = 2$

$(0,1) = y_{in} = 0\times1\ x_1 w_1 + x_2 w_2 = 0\times1 + 1\times1 = 1$

$(1,0) = y_{in} = x_1 w_1 + x_2 w_2 = 1\times1 + 0\times1 = 1$

$(0,0) = y_{in} = x_1 w_1 + x_2 w_2 = 0\times1 + 0\times1 = 0$

For AND function output is high if both input is high

Thus, for this condition i.e (1,1) the net input is ~~high~~ calculated as ~~to~~ 2 and On the based of this ~~threshold~~ input value the threshold value is set.
If threshold value is greater than or equal to 2 then only neuron fires else it doesnot fire.
Threshold value can be calculated by using the formula.

$$\theta \geq nw - p$$

where n = net input w = excitatory weight
p = inhabitory weight

$$\theta \geq nw - p$$
$$\theta \geq 2 \times 1 - 0$$
$$\theta \geq 2$$

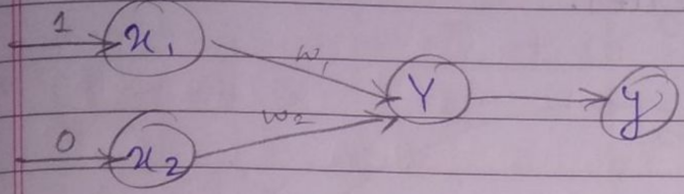
$$f(y_{in}) = \begin{cases} 1 & y_{in} \geq 2 \\ 0 & y_{in} < 2 \end{cases}$$

Q. Implement AND NOT function using McCulloch Pitt neuron

Sol^n Truth Table of AND NOT function

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

The Given function gives an output only when $x_1 = 1$ and $x_2 = 0$ the network can be represented as.



Case 1 :-

Assume both the weights w_1 and w_2 as 1

(1,1) = yin = 2

(1,0) = yin = 1

(0,1) = yin = 1

(0,0) = yin = 0

Comparing calculated output with truthtable all 3 are going to fire means their is error. To overcome this problem we have to update weight but there is no training algorithm so they will assume weights by ourself.

for case 1 $\theta \geq 2$

Case 2 :-

Assume $w_1 = 1$ and $w_2 = -1$

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

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

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

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

$\theta \geq nw - p$

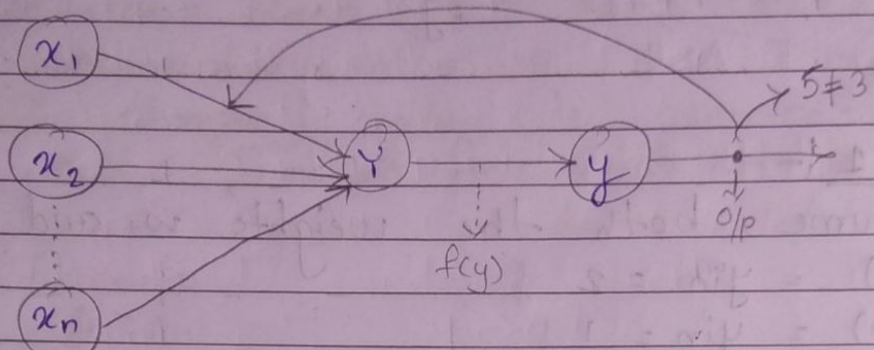
$\theta \geq 1 \times 1 - 1$

$\theta \geq 0$

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

22/4/22

(2) Perceptron Model.



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

If there will be any error at the output ~~leve~~ layer then we will update weight and bias values.

* Weights.

$W_1(new) = W_1(old) + \alpha \times t \times x_1$

$W_2(new) = w_2(old) + \alpha \times t \times x_2$

where α = learning rate

t = desired output

bias (new) = bias(old) + $\alpha \times t$

* Q. Implement AND function using Perceptron Network for Bipolar inputs and target

Sol^n The Perceptron Network Architecture for 2 inputs are as follows:-

INPUT			Target	Net input	Calculated	Weight changes			Weight		
									w1	w_2	b
x_1	x_2	1	(t)	y in	Y	Δw_1	Δw_2	Δw_3	0	0	0
Epoch 1											
1	1	1	1	0	0	$w_1(new) - w_1(old) = 1-0 = 1$	1	1	1	1	1
1	-1	1	-1	1	1	-1	1	-1	0	2	0
-1	1	1	-1	2	1	1	-1	-1	1	1	-1
-1	-1	1	-1	-3	-1	0	0	0	1	1	-1

- The ~~inputs~~ input patterns are presented to the network one by one, when all four input patterns are presented then 1 Epoch is said to be completed
- Initializing weights and bias :-
  $w_1 = 0 \quad w_2 = 0 \quad b = 0$

Page No.
Date

Epoch 1

(1) for the first input pattern ~~i.e $w_1=0$ $w_2=0$ and $b=0$~~ i.e. $x_1=1$ $x_2=1$ & $t=1$

Net input will be as follows:-

$$y_{in} = b + x_1 w_1 + x_2 w_2$$
$$= 0 + 1\times 0 + 1\times 0$$
$$y_{in} = 0$$

Threshold condition given:-

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

Since net input i.e. $y_{in}=0$ ~~i.e y~~
therefore, $\boxed{y=0}$

but, $t \neq y$ $\therefore$ weights bias updation required

$$w_1(new) = w_1(old) + \alpha \times t \times x_1$$
$$= 0 + 1\times 1\times 1$$
$$\therefore w_1(new) = 1$$

$$w_2(new) = w_2(old) + \alpha \times t \times x_2$$
$$= 0 + 1\times 1\times 1$$
$$\therefore w_2(new = 1$$

$$b(new) = b(old) \times \alpha \times t$$
$$= 0 + 1\times 1$$
$$\therefore b(new) = 1$$

Date

(2) for the second input pattern i.e. $x_1 = 1$ $x_2 = -1$ $t = -1$ we will use updated weight and bias. ∴ $w_1 = 1$ $w_2 = 1$ $b = 1$

Net input will be as follows :-

$$y_{in} = b + x_1 w_1 + x_2 w_2$$
$$= 1 + (1)(1) + (-1)(1)$$
$$= 1 + 1 - 1$$
$$\therefore y_{in} = 1$$

Since net input is greater than 0 i.e

$y_{in} = \cancel{0}\ 1$

therefore $y = 1$

~~but, t=1 t ≠ y~~ but $t = -1$ ∴ $t \neq y$

∴ weight and bias changes are required

$$w_1(new) = w_1(old) + \alpha \times t \times x_1$$
$$= 1 + 1 \times -1 \times 1$$
$$= \cancel{2}\ 0$$

$$w_2(new) = w_2(old) + \alpha \times t \times x_2$$
$$= 1 + 1 \times (-1) \times (-1)$$
$$= 1 + 1$$
$$= 2$$

$$b(new) = b(old) + \alpha \times t \times 1$$
$$= 1 + 1 \times (-1)$$
$$= 0$$

Page No.
Date

(3) for the third input pattern i.e. $x_1 = -1$ $x_2 = 1$ $t = 1$ we will use updated weight and bias value i.e. $w_1 = 0$ $w_2 = 2$ $b = 0$
Net input will be as follows

$$y_{in} = b + x_1 w_1 + x_2 w_2$$
$$= 0 + (-1)(0) + (1)(2)$$
$$\therefore y_{in} = 2$$

Since net input i.e $y_{in} = 2$ therefore $y = 1$
but $t = -1$ $\therefore$ $t \neq y$

$$w_1(new) = w_1(old) + \alpha \times t \times x_1$$
$$= 0 + 1 \times (-1) \times (-1)$$
$$\therefore w_1(new) = 1$$

$$\therefore w_2(new) = w_2(old) + \alpha \times t \times x_2$$
$$= 2 + 1 \times (-1) \times (1)$$
$$= 2 - 1$$
$$\therefore w_2(new) = 1$$

$$b(new) = b(old) + \alpha \times (t)$$
$$= 0 + (1)(-1)$$
$$\therefore b(new) = -1$$

(4) for the fourth input pattern i.e. $x_1 = -1$ $x_2 = -1$ $t = -1$ we will use update weight & bias i.e. $w_1 = 1$ $w_2 = 1$ $b = -1$

Net input will be as follows:-

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

Since net input i.e $y_{in} = -3$ therefore $y = -1$ $\therefore t = y$ So no updation required.

Epoch 2

Input			target (t)	Net Input y_{in}	Calculated Output (y)	Weight changes			Weights		
x_1	x_2	1				Δw_1	Δw_2	Δb	w_1	w_2	b
									+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	−3	−1				+1	1	−1

* Initial weights and bias for Epoch 2 is $w_1 = 1$, $w_2 = 1$ and $b = -1$

(1) for the first input pattern in Epoch 2 i.e $x_1 = 1$ $x_2 = 1$ $t_1 = 1$ we will use the updated weights & bias i.e $w_1 = 1$ $w_2 = 1$ $b = 1$.

Net input will be as follows:-

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

Page No.

Date

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

$$\therefore = -1 + 1 + 1$$

$$= +1$$

$\therefore y_{in} = +1$

Since net input i.e $y_{in} = +1$ $\therefore$ $y = +1$
$t = y$ so ~~there is~~ no updation required

(2) $x_1 = 1$ $x_2 = 1$ $t = -1$

weight & bias are $w_1 = 1$ $w_2 = 1$ $b = -1$

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

$y_{in} = -1$

$y = -1$ $\because$ $t = -1$

$y = t$ So no updation in weight and bias are required.

(3) for third input pattern i.e. $x_1 = -1$ $x_2 = 1$ $t = -1$

weight & bias are $w_1 = 1$ $w_2 = 1$ $b = -1$

Net input will be as follows:-

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

$\because y_{in} = -1$ $\boxed{y = -1}$

$\because t = -1$ $\therefore t = y$. So ~~ther~~ no updation in weight & bias required.

(4) for fourth input pattern i.e. $x_1 = -1$ $x_2 = -1$ $t = -1$

weight and bias are $w_1 = 1$ $w_2 = 1$ $b = -1$

Net input will be as follows:-

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

Page No. | Date

$= -3$

$\because y_{in} = -3 \quad \therefore y = -1$

$\because t = -1 \quad \therefore t = y$

So no updation in weight & bias required.

27/4/22

* Practical

- '.' ~~completes the fact~~ mean period.
- What we write in a parenthesis we call it as arguments and those arguments are called as object
- Outside the parenthesis are relations
- Save the file in D drive with extension .pl in All files.
- in GNU console select file → change dir → select d drive.
- command [prog]. (it compiles the program)
- command cat(X). (output X = tom)

Note :- Donot use capital letter in code

- Facts can be defined as expliait relation between object and properties of object
- Rules they are conditionally true. It define inplicit relationship between object
- ':-' is 'if' condition.
- ',' for 'and' and ';' for ~~at~~ 'or'
- In Rules we use more than one fact

- Rules have two parts
  (i) Conclusion part
  (ii) Conditional part
  Example:-

Code:- rules.pl [file]

```
likes (pooja, geeta).
likes (geeta, pooja).
friendship( X,Y) :- likes(X ,Y) , likes (Y, X).
```

- GNU console

```
[rules].
friendship(X,Y).
```

To run second fact ';' then enter.

* Relationship in Prolog.

- Relationship can also be a Rule
- We can define Relationship in form of Rules.

Code:- Relationship using rules.

```
parent (smith, reema).
parent (smith, seema).
female(reema)
female (seema)
sister (X,Y) :- parent (Z, X) , parent(Z,Y), female(X), female(Y)
```

GNU console:-

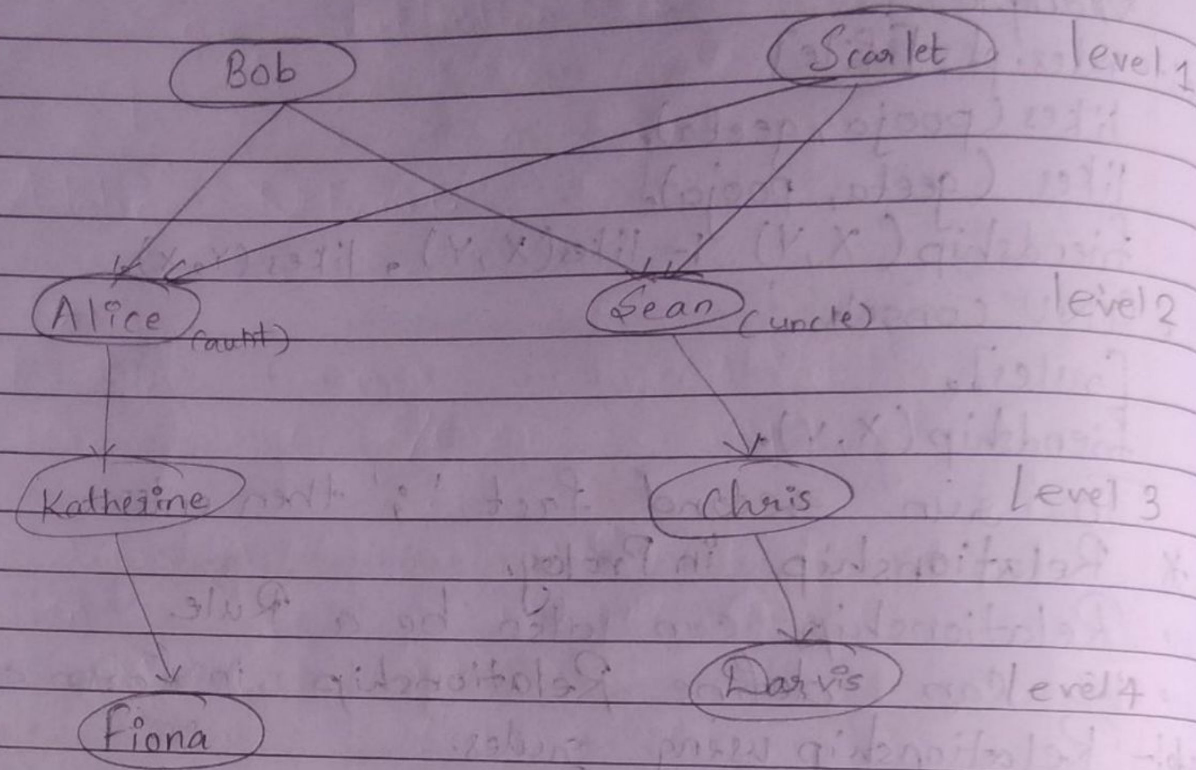
```
[sister].
?- sister (X, Y).
```

Output:-

```
X= reema
Y= reema ? ;
X = seema
Y= seema ? ;
X= Seema
Y= reema ? ;
X = seema
Y= seema.
Yes.
```

Page No.
Date

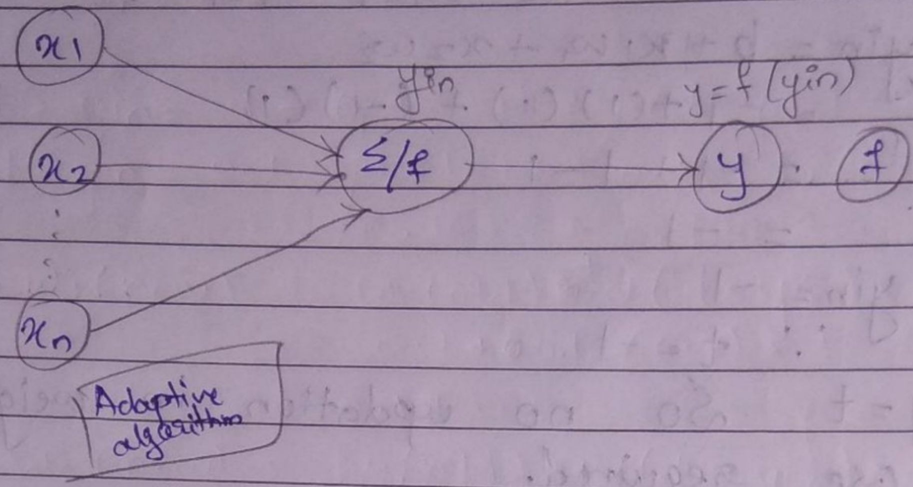
* Creating knowledge based on the basis of facts and rules.



Code:-
```
female (alice).
female (scarlet).
female (katherine).
female (fiona).
~~male (bob)~~ male (sean).
male (bob).
male (chris).
male (darvis).
parent (bob, alice).
parent (bob, sean).
parent (scarlet, alice).
parent (scarlet, sean).
parent (alice, Katherine).
parent (sean, chris).
parent (Katherine, fiona).
```

parent(chris, davis).
grandparent(X, Y) :- parent(X, Z), parent(Z, Y).
sister(X, Y) :- parent(Z, X), parent(Z, Y), female(X),
male(Y).

(3) Adaline Network



- $(t - y_{in})^2 \rightarrow$ least mean square error
- According to Adaline Network we perform Weight & bias changes as follows:-

$$W_1(new) = w_1(old)\ \alpha(t - y_{in}) \times x_1$$
$$W_2(new) = w_2(old)\ \alpha[t - y_{in}) \times x_2$$
$$b(new) = b(old)\ \alpha(t - y_{in})$$

Page No. Date

* Implement OR function with Bipolar inputs & targets using Adaline Network

Sol^n Step 1

Truth table for OR function with Bipolar inputs & targets can be drawn as follows:-

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

For $w_1 = 0.1$ $w_2 = 0.1$ $b = 0.1$ $\longrightarrow$ initial value of weights and bias.

$\alpha = 0.1$

Epoch 1

Input target (t) net Input (yin) Calculated o/p (y) Error t-yin Weight Changes Δw_1 Δw_2 Δb weights w_1 w_2 b

x_1, x_2 1 0.1 0.1 0.1

Epoch 1

Inputs		target (t)	net Input (yin)	Calculated o/p (y)	Error t-yin	Weight changes			weights			$(t-yin)^2$
									w_1	w_2	b	
x_1	x_2		(yin)	(y)		Δw_1	Δw_2	Δb	0.1	0.1	0.1	
1	1	1	0.3		0.7	0.07	0.07	0.07	0.17	0.17	0.17	0.49
1	-1	1	0.17		0.83	0.083	-0.083	0.083	0.253	0.087	0.253	0.6889
-1	1	1	0.087		0.913				0.1617	0.1753	0.3443	0.83
-1	-1	-1										

(1) for 1^{st} input pattern i.e $x_1 = 1$ $x_2 = 1$ $t = 1$
we use initial weights ~~w~~ $w_1 = 0.1$ $w_2 = 0.1$
$b = 0.1$

$$y_{in} = b + (x_1)(w_1) + (x_2)(w_2)$$
$$= 0.1 + (1)(0.1) + (1)(0.1)$$
$$= 0.1 + 0.1 + 0.1$$
$$= 0.3$$

$y_{in} = 0.3$

$t - y_{in} = 1 - 0.3$
$t - y_{in} = 0.7$

$$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$$
$$= 0.1 + 0.1\,(0.7) \times 1$$
$$= 0.17$$

$$w_2(new) = w_1(old) + \alpha(t - y_{in}) \times x_2$$
$$= 0.1 + 0.1\,(0.7) \times 1$$
$$= 0.17$$

$$b(new) = b(old) + \alpha(t - y_{in})$$
$$= 0.1 + 0.1(0.7)$$
$$= 0.17$$

(2) for second input pattern $x_1 = 1$ $x_2 = -1$
$t = 1$ we use updated weight & bias
i.e $w_1 = 0.17$ $w_2 = 0.17$ $b = 0.17$

$$y_{in} = b + x_1 w_1 + x_2 w_2$$
$$= 0.17 + 1(0.17) + -1(0.17)$$
$$= 0.17 + 0.17 - 0.17$$
$$= 0.17$$

$t - y_{in} = 1 - 0.17$
$= 0.83$

$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$

$= 0.17 + 0.1(0.83) \times 1$

$= 0.17 + 0.083$

$= 0.253$

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times x_2$

$= 0.17 + 0.1(0.83) \times -1$

$= 0.17 - 0.083$

$= 0.087$

$b(new) = b(old) + \alpha(t - y_{in})$

$= 0.17 + 0.1(0.83)$

$= 0.17 + 0.083$

$= 0.253.$

(3) For third input pattern i.e. $x_1 = -1$ $x_2 = 1$ $t = 1$
we use updated weights $w_1 = 0.253$ $w_2 = 0.087$
$b = 0.253$.

Net input will be as follows:-

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

$= 0.253 + 0.253(-1) + (1)(0.087)$

$y_{in} = 0.087$

$t - y_{in} = 0.913$

$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$

$= 0.253 + 0.1(0.913) \times (-1)$

$= 0.1617$

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times x_2$

$= \cancel{0.253}\ 0.087 + 0.1(0.913) \times (1)$

$= 0.1753$

$b(new) = b(old) + \alpha[t - y_{in})$

$= 0.253 + 0.1(0.913)$

$= 0.3443.$

Date

(4) for fourth input pattern i.e $x_1 = -1$ $x_2 =$ [illegible]
$t = -1$ we use updated weights & bias.
$w_1 = 0.1617$ $w_2 = 0.1753$ $b = 0.3443$.
Net input will be as follows

$y_{in} = b + x_1 w_1 + x_2 w_2$
$= 0.3443 + (-1)(0.1617) + (-1)(0.17$[illegible]
$= 0.3443 - 0.1617 - 0.1753.$
$= 0.0073.$

$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$
$= 0.1617 + 0.1(-1.0073) \times -1$
$= 0.1617 + 0.10073$
$= 0.26243.$

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times x_2$
$= 0.1753 + 0.10073$
$= 0.27603$

$b(new) = b(old) + \alpha(t - y_{in})$
$= 0.3443 - 0.10073$
$= 0.24357$

Input		Target (t)	net input	Error (t-yin)	Weight changes			Weight			$(t - yin)^2$
x_1	x_2				Δw_1	Δw_2	Δb	w_1	w_2	b	
1	1	1	0.7847					0.2836	0.3003	0.2653	
1	-1	1	0.2487					0.3587	0.2251	0.3405	
-1	1	1	0.2069					0.2791	0.3044	0.3898	
-1	-1	-1	0.1945					0.3599	0.3849	0.3092	

Page No.
Date

Epoch 2.

(1) first input pattern

$x_1 = 1 \quad x_2 = 1 \quad t = 1$

$w_1 = 0.2621 \quad w_2 = 0.2787 \quad b = 0.2438$

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

$= 0.7847$

$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$

$= 0.2621 + 0.1(1 - 0.7847) \times 1$

$= 0.2621 + 0.02153$

$= 0.28363$

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times x_2$

$= 0.2787 + 0.1(1 - 0.7847) \times 1$

$= 0.2787 + 0.02153$

$= 0.30023.$

$b(new) = b(old) + \alpha(t - y_{in})$

$= 0.2438 + 0.1(1 - 0.7847)$

$= 0.2438 + 0.02153$

$= 0.26533$

(2) Second input pattern

$x_1 = 1 \quad x_2 = -1 \quad t = 1$

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

$= 0.2654 + (1)(0.28363) + (-1)(0.30023)$

$= 0.24874$

$w_1(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$

$= 0.28363 + 0.1(0.75126) \times 1$

$= 0.28363 + 0.075126$

$= 0.3587$

Page No.
Date

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times (-1)$
$= 0.3003 + 0.1(0.75126) \times (-1)$
$= 0.3003 + 0.075126$
$= + 0.2251$

$b(new) = b(old) + \alpha(t - y_{in})$
$= 0.2654 + 0.075126$
$= 0.3405.$

(3) for third input pattern

$x_1 = -1 \quad x_2 = 1 \quad t = 1$

$y_{in} = 0.3405 + (-1)(0.3587) + (1)(0.2251)$
$= 0.3405 - 0.3587 + 0.2251$
$= 0.2069$

$w(new) = w_1(old) + \alpha(t - y_{in}) \times x_1$
$= 0.3587 + \cancel{0.5}\ 0.1(0.7931) \times -1$
$= \cancel{0.43801}\ 0.3587 - 0.07931 = 0.2794$

$w_2(new) = w_2(old) + \alpha(t - y_{in}) \times x_2$
$= 0.2251 + 0.07931$
$= 0.3044$

$b(new) = b(old) + \alpha(t - y_{in})$
$= 0.3405 + 0.07931$
$= 0.38981$

(4) for fourth input pattern

$x_1 = -1 \quad x_2 = -1 \quad t = -1$

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

$= 0.3898 + (-1)(0.2794) + (-1)(0.3044)$

$=$

$= 0.3898 - 0.2794 - 0.3044$

$= -0.1945$

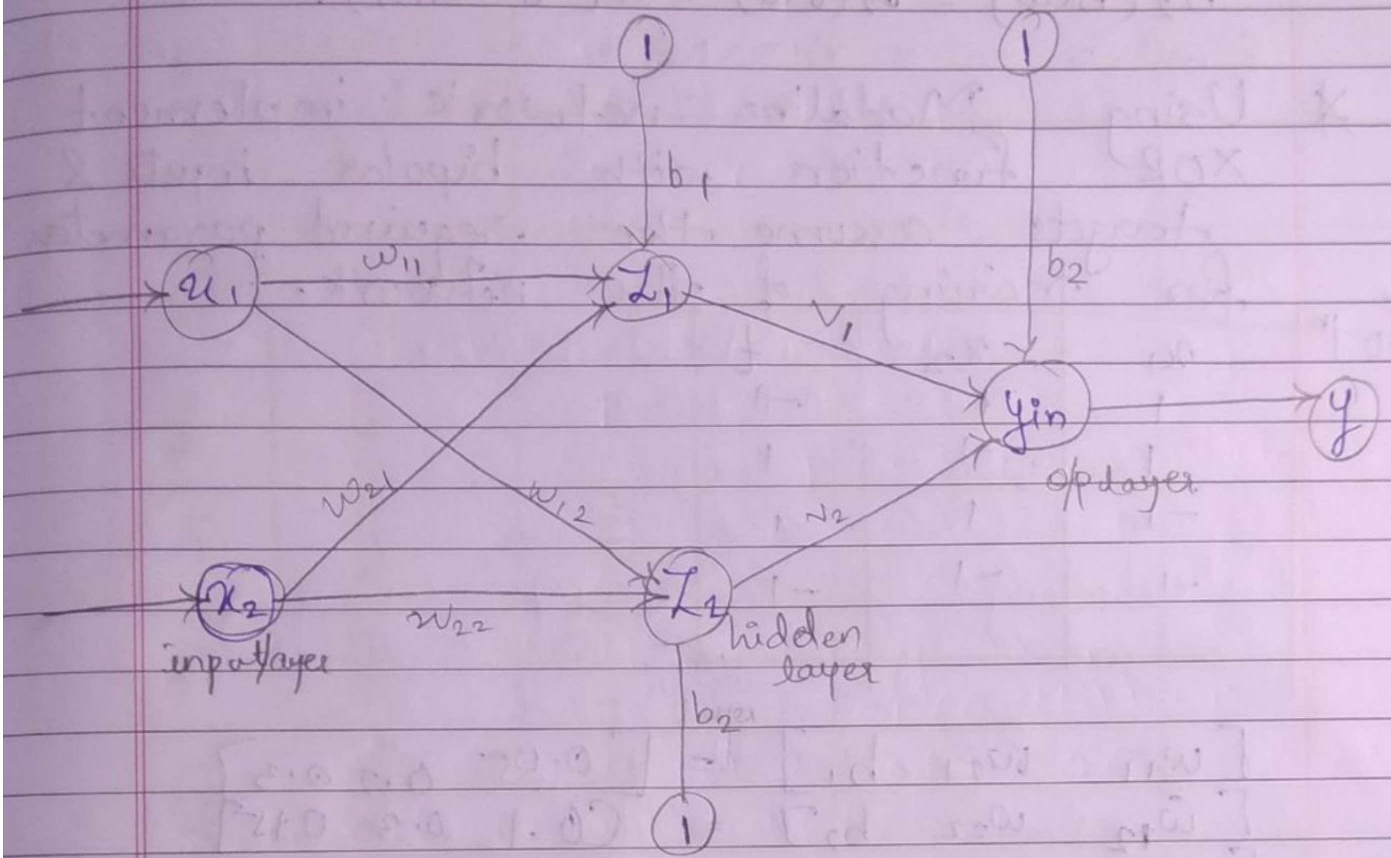
$$w_1(\text{new}) = w_1(\text{old}) + \alpha(t - y_{in}) \times x_1$$
$$= 0.2794 + 0.1(-1 - (-0.1945)) \times -1$$
$$\cancel{= 0.2794 +}$$
$$= 0.3599$$

$$w_2(\text{new}) = 0.3044 + 0.1(-1 - (-0.1945)) \times -1$$
$$= 0.3849$$

$$b(\text{new}) = b(\text{old}) + \alpha(t - y_{in})$$
$$= 0.3092$$

Page No.
Date

* Madaline (Multiple Adaline Network).
(More than one computation layer)



(1) Calculate net input at hidden layer

$$Zin1 = b_1 + x_1 \times w_{11} + x_2 \times w_{21}$$
$$Zin2 = b_2 + x_1 \times w_{12} + x_2 \times w_{22}$$
$$Z_1 = f(Zin1)$$
$$Z_2 = f(Zin2).$$

(2) Calculating net input at output layer.

$$yin = b_3 + Z_1 v_1 + Z_2 v_2$$
$$y = f(yin).$$

(3) Updating or changing bias value.

$$\left.\begin{aligned} w_{11}(new) &= w_{11}(old) + \alpha(t - Zin1) \times x_1 \\ w_{21}(new) &= w_{21}(old) + \alpha(t - Zin1) \times x_2 \\ b_1(new) &= b_1(old) + \alpha(t - Zin1) \end{aligned}\right\} Zin1$$

Page No.
Date

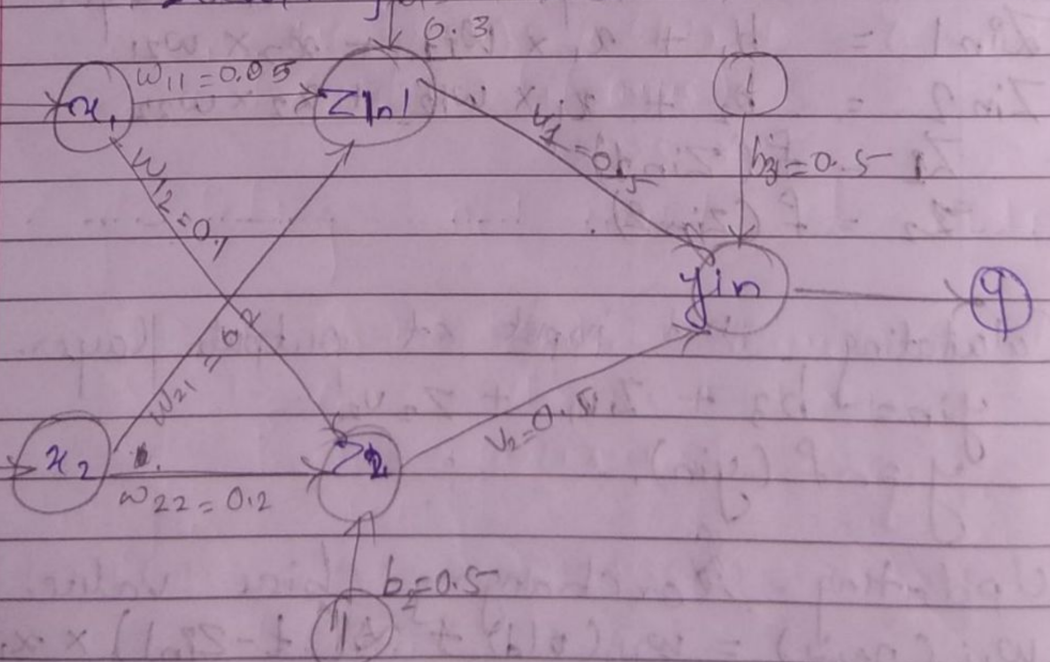
$$w_{12}(new) = w_{12}(old) + \alpha(t - Z_{in2}) \times x_1$$
$$w_{22}(new) = w_{22}(old) + \alpha(t - Z_{in2}) \times x_2$$
$$b_2(new) = b_2(old) + \alpha(t - Z_{in2}).$$
Z_{in2}

* Using Madaline network implement XOR function with bipolar inputs & targets assume the required parameters for training of the network.

Soln

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

$[w_{11} \quad w_{21} \quad b_1] = [0.05 \quad 0.2 \quad 0.3]$
$[w_{12} \quad w_{22} \quad b_2] = [0.1 \quad 0.2 \quad 0.15]$
$[v_1 \quad v_2 \quad b_3] = [0.5 \quad 0.5 \quad 0.5]$
learning rate = 0.5



x_1	x_2	t	Z_{in1}	Z_{in2}	Z_1	Z_2	y_{in}	y	W_{11}	W_{21}	b_1	W_{21}	W_{22}	b_2	V_1	V_2	b_3
1	1	-1	0.55	0.45	1	1	1.5	1	-0.725	-0.575	-0.475	-0.625	-0.525	-0.575			
1	-1	1	-0.625	-0.675	-1	-1	-0.5	-1	0.0875	-1.3875	0.3375						
-1	1	-1															
-1	-1	-1															

Date

~~Zin1 =~~ Epoch 1

(1) 1^{st} input pattern

$x_1 = 1 \quad x_2 = 1 \quad t = -1$

$$Zin1 = b_1 + x_1 \times w_{11} + x_2 \times w_{21}$$
$$= 0.3 + 1 \times 0.05 + 1 \times 0.2$$
$$= 0.55$$

$$Zin2 = b_2 + x_1 \times w_{12} + x_2 \times w_{22}$$
$$= 0.15 \times 1 \times 0.1 + 1 \times 0.2$$
$$= 0.15 + 0.1 + 0.2$$
$$Zin2 = 0.45$$

Calculating Z_1 & Z_2 by applying activation function given as below.

$$Z = f(zin) = \begin{cases} 1 & \text{if } zin \geq 0 \\ -1 & \text{if } zin < 0 \end{cases}$$

$$Z1 = f(Zin1) = f(0.55) = 1$$
$$\therefore Z_1 = 1$$
$$Z_2 = f(Zin_2) = f(0.45) = 1$$

Calculating net input at output layer

$$yin = b_3 + Z_1 v_1 + Z_2 v_2$$
$$= 0.5 + 1 \times 0.5 + 1 \times 0.5$$
$$= 0.5 + 0.5 + 0.5$$
$$yin = 1.5$$

$$y = f(yin)$$
$$\therefore y = 1$$

Page No.
Date

$\because t = -1 \quad \therefore t \neq y$ So weight & bias changes required.

$w_{11}(new) = w_{11}(old) + \alpha(t - z_{in1}) \times x_1$
$= 0.05 + 0.5(-1.55) \times 1$
$= -0.725$

$w_{21}(new) = w_{21}(old) + \alpha(t - z_{in1}) \times x_2$
$= 0.2 + 0.5(-1.55) \times (1)$
$= 0.2 + (-0.775)$
$= -0.575$

$b_1(new) = b_1(old) + \alpha(t - z_{in})$
$= 0.3 - 0.775$
$= -0.475$

$w_{12}(new) = w_{12}(old) + \alpha(t - z_{in2}) \times x_1$
$= 0.1 \cancel{+ 0.5(0.7} + 0.5(-1 - 0.45) \times 1$
$= 0.1 + 0.5(-1.45) \times 1$
$= 0.1 - 0.725$
$= -0.625$

$w_{22}(new) = \cancel{0.2}\ w_{22}(old) + \alpha(t - z_{in2}) \times x_2$
$= 0.2 - 0.725$
$= -0.525$

$b_2(new) = b_2(old) + \alpha(t - z_{in2})$
$= 0.15 - 0.725$
$= -0.575$

(2) 2^{nd} input pattern.

$x_1 = 1 \qquad x_2 = -1 \qquad t = 1.$

$Zin1 = -0.475 + 1(-0.725) + (-1)(-0.575)$
$= -0.475 + -0.725 + 0.575$
$= -0.625$

$Zin_2 = \cancel{-0.475+} -0.575 + 1(0.625) + (-1)(0.525)$
$= -0.575 - 0.625 + 0.525$
$= -0.675$

$Z_1 = f(Zin1)$
$= f(-0.625)$
$= -1$

$Z_2 = f(-0.675)$
$= -1$

$yin = 0.5 + (-1)(0.5) + (-1) \times (0.5)$
$= 0.5 - 0.5 + -0.5$
$= -0.5$

$y = -1$

$\because t = 1 \quad \therefore t \neq y$ So weights & bias changes required.

Note if the value of zin1 and zin2 are in -ve then we will apply weight changes only to the nearest to zero value of zin.

~~$W_{11}(new) = W_{11}(old) + \alpha(t - Zin1) \times x_1$~~
~~$= (-0.725) + 0.5(0.375) \times 1$~~
~~$= -0.725 + 0.1875$~~
~~$= -0.9125$~~

$$w_{21}(\text{new}) = w_{21}(\text{old}) + \alpha(t - Zin_1) \times x_2$$
$$= -0.575 + 0.1875 \times (-1)$$
$$= -0.575 - 0.1875$$
$$= -0.7625$$
$$b_1(\text{new}) = b_1(\text{old}) + \alpha(t - Zin_1)$$
$$= -0.475 + 0.1875$$
$$= -0.2875$$

$$w_{11}(\text{new}) = w_{11}(\text{old}) + \alpha(t - Zin_1) \times x_1$$
$$= -0.725 + 0.5\,(1.625) \times 1$$
$$= -0.725 + 0.8125$$
$$= 0.0875$$
$$w_{21}(\text{new}) = w_{21}(\text{old}) + \alpha(t - Zin_1) \times x_2$$
$$= -0.575 + 0.8125 \times (-1)$$
$$= -0.575 - 0.8125$$
$$= -1.3875$$
$$b_1(\text{new}) = b_1(\text{old}) + \alpha(t - Zin_1)$$
$$= -0.475 + 0.8125$$
$$= 0.3375$$

(3) for 3^{rd} input pattern

$x_1 = -1$ $\quad x_2 = +1$ $\quad t = 1$

$$Zin_1 = 0.3375 + (0.0875)(-1) + (-1.3875)(1)$$
$$= 0.3375 - 0.0875 - 1.3875$$
$$= -1.1375$$

$$Zin_2 = -0.575 + (-1)(0.625) + (+1)(0.525)$$
$$= -0.575 + 0.625 - 0.525$$
$$= -0.475$$

$y_{in} = z_1 \times v_1 + z_2 v_2 + b_3$ $\quad y = -1$

$= 0.5 + (-1)(0.5) + (-1)(0.5)$

$= 0.5 - 0.5 - 0.5$

$= -0.5$

Page No. Date

$z_1 = -1 \quad z_2 = -1$

$\because y = -1 \quad \because t = 1 \quad t \neq y$

$w_{12}(new) = w_{12}(old) + \alpha(t - z_{in2}) \times x_1$

$= -0.625 - 0.7375$

$= -1.3625$

$w_{22}(new) = w_{22}(old) + \alpha(t - z_{in2}) \times x_2$

$= -0.525 + 0.7375 \times 1$

$= 0.2125$

$b_2(new) = b_2(old) + \alpha(t - z_{in2})$

$= -0.575 + 0.7375$

$= 0.1625$

(4) for 4^{th} input pattern

$x_1 = -1 \quad x_2 = -1 \quad t = -1$

$z_{in1} = 0.3375 + (0.0875)(-1) + (-1)(-1.3875)$

$= 0.3375 - 0.0875 + 1.3875$

$= 1.6375$

$z_{in2} = 0.1625 + (-1)(-1.3625) + (-1)(0.2125)$

$= 0.1625 + 1.3625 - 0.2125$

$= 1.3125$

$z_1 = f(z_{in1}) = 1$

$z_2 = f(z_{in2}) = 1$

$y_{in} = b_3 + z_1(v_1) + z_2 + v_2$

$= (0.5) + (1)(0.5) + (1)(0.5)$

$= 1.5$

$y = f(y_{in}) = 1$

$\because t = -1 \quad \because y = 1 \quad \therefore t \neq y.$

$w_{11}(new) = w_{11}(old) + \alpha(t - \cancel{y_{in}}\, z_{in1}) \times x_1$

$= 0.0875 + 0.5(-1 - 1.6375) \times -1$

$= 0.0875 \; -2.6375(-1)$

$= 0.0875 + \cancel{2.6375}\; 1.31875$

$= 1.40625$

$w_{21}(new) = w_{21}(old) + \alpha(t - z_{in1}) \times x_2$

$= -1.3875 + 1.31875$

$= -0.06875$

$b_1(new) = b_1(old) + \alpha(t - z_{in1})$

$= 0.3375 - 1.31875$

$= -0.98125$

$w_{12}(new) = w_{12}(old) + \alpha(t - z_{in2}) \times x_1$

$= -1.3625 + 0.5(-1 - 1.3125) \times (-1)$

$= -1.3625 + \cancel{0.5(-3.29375) \times (-1)}$

$= -1.3625 - \cancel{1.6468(-1)}\; 1.15625(-1)$

$= -1.3625 + \cancel{1.6468}\; 1.15625$

$= \cancel{3.0093} \;\; \cancel{0.2843} \;\; 2. \;\; -0.20625$

$w_{22}(new) = w_{22}(old) + \alpha(t - z_{in2}) \times (x_2)$

$= 0.2125 + \cancel{1.6468}\; 1.15625$

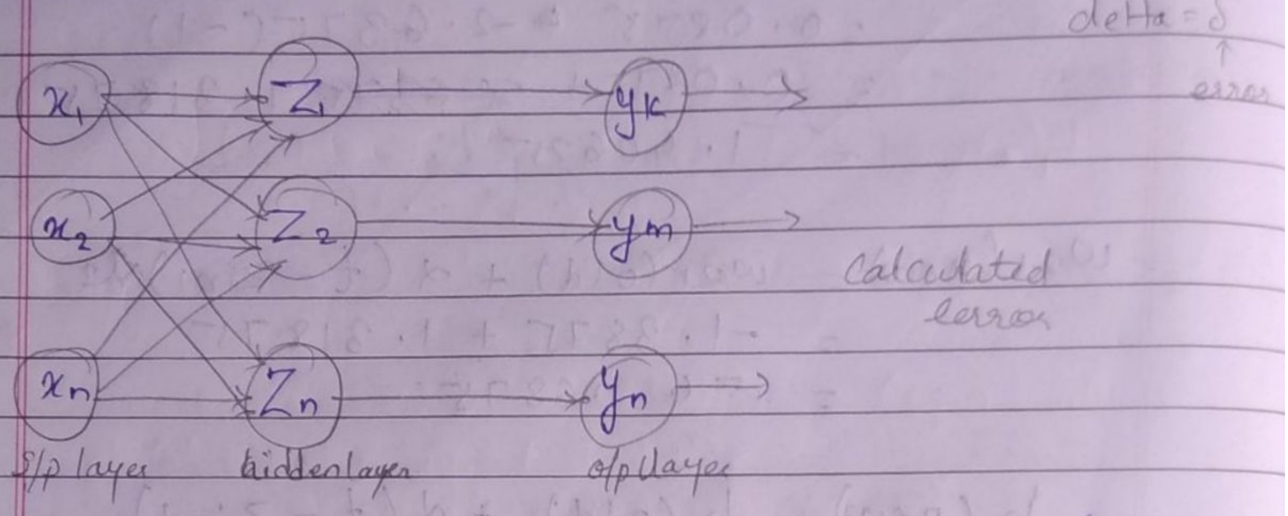
$= \cancel{1.8593}\; 1.36875$

$b_2(new) = b_2(old) + \alpha(t - z_{in2})$

$= 0.1625 \; \cancel{-1.6468} \; -1.15625$

$= \cancel{-1.4843} \; -0.99375$

* Back propogation Algorithm

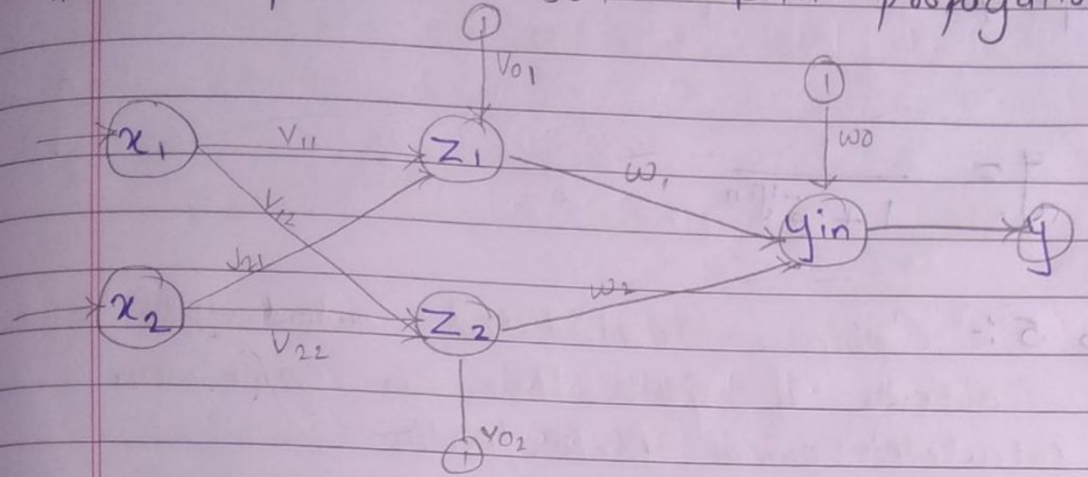


- Back propogation algorithm is one of the most important development in Neural Network.
- This learning algorithm is applied to multilayer feed forward network. In this method error is propogated back to the hidden unit.
- To update weight we require (δ) delta which is calculated error.
- When targeted output is not equal to calculated output then there is some error so in back propogation algorithm we calculate error at the output layer. which is the difference between targeted output (t) & calculated output (y).
- The training of a back propogation network is done in 3 stages
  1) The feed forward of input training pattern
  2) The calculation & back propagation of the error
  3) Updation of weights.

Note: The newtork associated with BPA is called BPN.

Page No.
Date

* Steps to solve Back propagation Network



* Note :- In BPN we use binary sigmoidal activation function to calculate output

Step 1 :- Calculate net input at Z layer (hidden layer)

$Zin1 = V_{01} + x_1(V_{11}) + x_2(V_{21})$

$Zin2 = V_{02} + x_1(V_{12}) + x_2(V_{22})$

Step 2 :- ~~Zif~~ $\cancel{Z_1 = f(Zin1)}$, $\cancel{Z_2 = f(Zin2)}$ ~~Apply~~ Appling binary sigmoidal on Zin1 & Zin2 to calculate Z_1 & Z_2

$$Z_1 = \frac{1}{1+e^{-Zin1}}$$

$$Z_2 = \frac{1}{1+e^{-Zin2}}$$

Step 3 :- Calculate net input yin at output layer

$yin = w_0 + Z_1 w_1 + Z_2 w_2$

Date

Step 4 :- Applying binary Sigmoidal on yin to calculate y.

$$y = \frac{1}{1+e^{-y_{in}}}$$

Step 5 :- Compare Calculated output with target output if $y \neq t$ there is some error
∴ calculate error factor/portion.

$\delta(\text{error}) = t - y \times f'(y_{in})$

Where $f'(y_{in}) = f(y_{in}) \times (1 - f(y_{in}))$

Step 6 :- find the changes in weight between output layer & hidden layers

$\Delta w_1 = \alpha\, \delta_1 \times z_1$

$\Delta w_2 = \alpha\, \delta_1 \times z_2$

$\Delta b_2 = \alpha \delta$

Step 7 :- Compute the error portion between hidden layer & input layer

$\delta_1 = \delta_{in1} \times f'(z_{in1})$ where $\delta_{in1} = \delta_1 \times w_1$ (where δ_1 is error from previous step)

$\delta_2 = \delta_{in2} \times f'(z_{in2})$ where $\delta_{in2} = \delta_1 \times w_2$

$f'(z_{in1}) = f(z_{in1}) \times (1 - f(z_{in1}))$

$f'(z_{in2}) = f(z_{in2}) \times (1 - f(z_{in2}))$

Step 8 :- Find the changes in weights between hidden layer & input layer

$\Delta V_{11} = \alpha\, \delta_1\, x_1$

$\Delta V_{21} = \alpha\, \delta_1\, x_2$

$\Delta V_{01} = \alpha\, \delta_1$

$\Delta V_{12} = \alpha \delta_2 x_1$

$\Delta V_{22} = \alpha \delta_2 x_2$

$\Delta V_{02} = \alpha \delta_2$

Step 9: Compute the final weights of the network

$V_{11}(new) = V_{11}(old) + \Delta V_{11}$

$V_{12}(new) = V_{12}(old) + \Delta V_{12}$

$V_{21}(new) = V_{21}(old) + \Delta V_{21}$

$V_{22}(new) = V_{22}(old) + \Delta V_{22}$

$V_{01}(new) = V_{01}(old) + \Delta V_{01}$

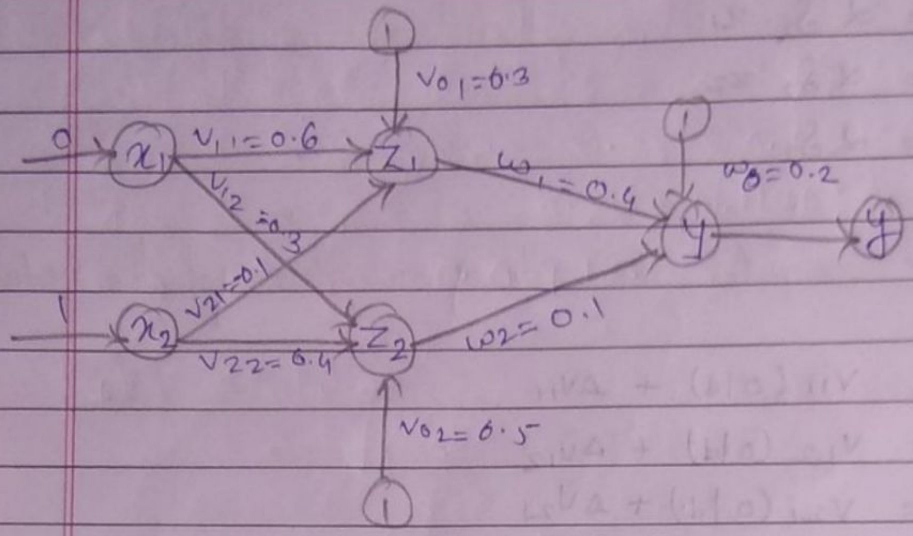
$V_{02}(new) = V_{02}(old) + \Delta V_{02}$

$w_1(new) = w_1(old) + \Delta w_1$

$w_2(new) = w_2(old) + \Delta w_2$

$w_0(new) = w_0(old) + \Delta w_0.$

Page No.			
Date			



Solⁿ $[v_{11}\ v_{21}\ v_{01}] = [0.6\ -0.1\ 0.3]$

$[v_{12}\ v_{22}\ v_{02}] = [-0.3\ 0.4\ 0.5]$

$[w_1\ w_2\ w_0] = [0.4\ 0.1\ 0.2]$

① Calculate net input at Z layer (computational)

$Zin1 = v_{01} + x_1 \times v_{11} + x_2 \times v_{21}$
$= 0.3 + 0(0.6) + (1)(-0.1)$
$= 0.3 + 0 + (-0.1)$
$= 0.3 - 0.1$
$= 0.2$

$Zin2 = v_{02} + x_1 \times v_{12} + x_2 \times v_{22}$
$= 0.5 + (0)(-0.3) + (1)(0.4)$
$= 0.5 - 0 + 0.4$
$= 0.9$

② Applying Binary Sigmoidal function to calculate Z_1 & Z_2 as:

$Z_1 = \dfrac{1}{1+e^{-zin1}} = \dfrac{1}{1+e^{-0.2}} = \dfrac{1}{1.8187} = 0.5498$

Page No. Date

$$Z_2 = \frac{1}{1+e^{-z_{in2}}} = \frac{1}{1+e^{-0.9}} = \frac{1}{1.4065} = 0.7109$$

(3) Calculate net input i.e y_{in} -

$$y_{in} = w_0 + z_1 \times w_1 + z_2 \times w_2$$
$$= -0.2 + 0.5498 \times (0.4) + (0.7109)(0.1)$$
$$= -0.2 + 0.2199 + 0.07109$$
$$= 0.09099 \cong 0.09101$$

(4) Applying Binary Sigmoidal function to calculate y

$$y = \frac{1}{1+e^{-y_{in}}} = \frac{1}{1+e^{-0.09101}} = \frac{1}{1.9130} = 0.5227$$

But the target output is given as 1 and we are obtaining calculated output ~~as 0.5~~ from the network as 0.5227 which means there is some error ($t \neq y$).

(5) Compute error portion (δ_1)

$$\delta_1 = (t-y) \times f'(y_{in}) = 0.4773 \times 0.2495 = 0.11908$$

$$f'(y_{in}) = f(y_{in}) \times (1 - f(y_{in}))$$
$$= f(y_{in}) \times 0.4773$$
$$= 0.5227 \times 0.4773$$
$$= 0.2494$$

Date

Using error portion δ_1

(6) Calculate the weight changes ~~between~~

$\Delta w_1 = \alpha \delta_1 \times z_1 = 0.25 \times 0.1191 \times 0.5498 = 0.0164$

$\Delta w_2 = \alpha \delta_1 \times z_2 = 0.25 \times 0.1191 \times 0.7109 = 0.021178$

$\Delta w_3 = \alpha \delta_1 = 0.02978.$

(7) Calculating error portion between hidden & output layer.

$\delta_1 = \delta_{in1} \times f'(z_{in1})$

$\delta_2 = \delta_{in2} \quad f'(z_{in2})$

$f'(z_{in1}) = f(z_{in1}) \times (1 - f(z_{in1}))$

$= 0.5498 \quad \cancel{\times 0.8} \; 0.4502$

$= \cancel{0.16} \; 0.2475$

$f'(z_{in2}) = f(z_{in2}) \times (1 - f(z_{in2}))$

$= \cancel{0.9 \times 0.1} \; 0.7109 \times 0.2891$

$= \cancel{0.09} \; 0.2055$

0.0118

$\delta_1 = \cancel{0.2495} \; 0.1191 \times 0.2475 = 0.0295 \times 0.4 = 0$

$\delta_2 = 0.1191 \times 0.2055 = 0.0245 \times 0.1 = 0.002$

(8) Calculate the weight changes

$\Delta v_{11} = \alpha \times \delta_1 \times x_1 = 0$

$\Delta v_{21} = \alpha \times \delta_1 \times x_2 = 0.00295$

$\Delta v_{01} = \alpha \times \delta_1 = 0.00295$

$\Delta v_{12} = \alpha \times \delta_2 \times x_1 = 0$

$\Delta v_{22} = \alpha \times \delta_2 \times x_2 = \cancel{0.00245} \; 0.0006125$

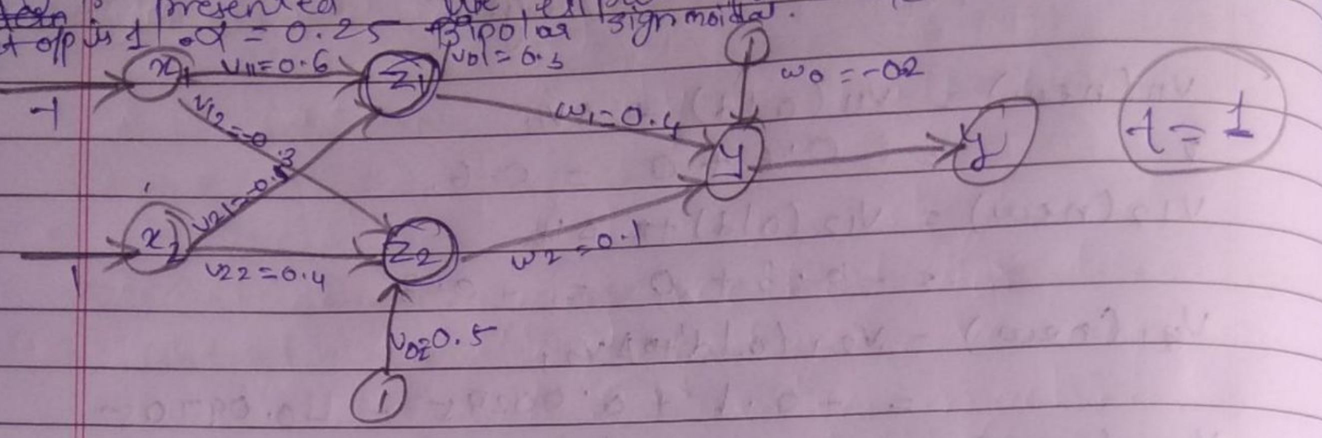
$\Delta v_{02} = \alpha \times \delta_2 = 0.0006125$

Page No.
Date

(a) Compute final weight

$v_{11}(new) = v_{11}(old) + \Delta v_{11}$
$= 0.6 + 0 = 0.6.$
$v_{12}(new) = v_{12}(old) + \Delta v_{12}$
$= -0.3 + 0 = -0.3$
$v_{21}(new) = v_{21}(old) + \Delta v_{21}$
$= -0.1 + 0.00295 = -0.09705$
$v_{22}(new) = v_{22}(old) + \Delta v_{22}$
$= 0.4 + 0.0066125 = 0.4006125$
$v_{01}(new) = v_{01}(old) + \Delta v_{01}$
$= 0.3 + 0.00295 = 0.30295$
$v_{02}(new) = v_{02}(old) + \Delta v_{02}$
$= 0.5 + 0.006125 = 0.5006125$
$w_1(new) = w_1(old) + \Delta w_1$
$= 0.4 + 0.0164 = \cancel{0.00656}\ 0.4164$
$w_2(new) = w_2(old) + \Delta w_2$
$= 0.1 + 0.02117 = \cancel{0.002117}\ 0.12117$
$w_0(new) = w_0(old) + \Delta w_0$
$= -0.2 + 0.02978$
$= \cancel{-0.05956}\ -0.17022$

Find a new weights using back propagation Network Assignment for the network shown below the network ~~pattern~~ is presented the input pattern ~~[-1 & 1~~ [-1, 1] & target o/p is 1. $\alpha = 0.25$ Bipolar sigmoidal.



* Revision

1) Perceptron – (a) we calculate output on the basis of net input value. (b) We calculate new weight & bias values as follows [$w_1(new) = \alpha \times t \times x_1 + w_1(old)$]

x_1	x_2	t	yin	y	Δw_1	Δw_2	Δb	w_1	w_2	b

2) Adaline – (a) We calculate error factor & using error factor we calculate weight & bias value.
[$w_1(new) = w_1(old) + \alpha \times (t - yin) x_1$]

Note :- For each an every iteration we calculate mean square error

x_1	x_2	t	$t - yin$	Δw_1	Δw_2	Δb	w_1	w_2	b	$(t-yin)^2$

Page No.
Date

1/5/22 UNIT 4 - ML.

Unsupervised learning :-

Association Analysis with Apriori Algorithm

- Association Rule is a rule based on machine learning method for discovering interest relations between variables in large data sets. It involves machine learning models to analyze data for pattern & for co-occurences in a database.
If there are 2 products A & B if we purchase product A then product B so we write A ⇒ B (A then B) where A is called as Antecedent & B is called as consequent.
2 measurement units are used to measure association between item sets
- Support - It denotes percentage of transactions that contains both A & B.
- Confidence - % of transaction containing A which also contains B.

Example :-
The rule is Milk ⇒ Bread.
So, formula for support (Milk ⇒ Bread) =

$$= \frac{\text{no of tuples containing both milk \& bread}}{\text{total no. of tuple}}$$

$$\text{Confidence (Milk} \Rightarrow \text{Bread)} = \frac{\text{no. of tuples contain both Milk \& bread}}{\text{no. of tuples containing milk}}$$

Page No.
Date

Suppose,
No. of tuples containing M & B = 4
total No of Tuple = 5
No of tuples containing M is 4

$S = \frac{4}{5} = 80\%$ there is 80% of support (Milk ⇒ Bread)

$C = \frac{4}{4} = 100\%$ there is 100% of confidence.

Q.

Trans-id	Item Sets
01	Bag, Uniform, Water Bottle
02	Book, Bag, Uniform
03	Bag, Uniform, Pen
04	Bag, Pen, Book
05	Uniform, WaterBottle, Bag
06	Bag, Pen, Book
07	WaterBottle, Uniform, Bag
08	Book, Water Bottle, Bag
09	Uniform, WaterBottle Pen
10	Pen, Uniform, Book

Find Association Rule with 30% support & 75% confidence.

12/5/22

Soln part 1 :- (Iteration 1)

Elements	Frequency	Support
Bag	8	$(8/10)^* 100 = 80\%$
Uniform	7	$(7/10)^* 100 = 70\%$
WaterBottle	5	$(5/10)^* 100 = 50\%$
Book	5	$(5/10)^* 100 = 50\%$
Pen	5	$(5/10)^* 100 = 50\%$

Page No.
Date

Since all elements are having support percentage more than 30 %.
Therefore all the elements will pass to second Iteration.

~~Part 2~~ (2nd Iteration).

Item Set	Frequency / Support	Support %
Bag, Uniform	5	(5/10) * 100 = 50 %.
Bag, Water Bottle	4	(4/10) * 100 = 40 %.
Bag, Book	4	(4/10) * 100 = 40 %.
Bag, Pen	3	(3/10) * 100 = 30 %.
Uniform, Water Bottle	4	(4/10) * 100 = 40 %.
~~Uniform, Book~~	~~2~~	~~(2/10) * 100 = 20 %~~
Uniform, Pen	3	(3/10) * 100 = 30 %.
~~Water Bottle, Book~~	~~1~~	~~(1/10) * 100 = 10 %.~~
~~Water Bottle, Pen~~	~~1~~	~~(1/10) * 100 = 10 %~~
Book, Pen	3	(3/10) * 100 = 30 %.

So the combination of item sets having support percentage more than 30 % will pass on to 3rd iteration.

~~Part 3~~

Page No.
Date

~~Phase~~ (3rd Iteration).

ItemSet	Frequency Support	Support %
Bag, Uniform, Water Bottle	3	30%
Bag, Uniform, Book	1	10%
Bag, Water Bottle, ~~Pen~~ Book	1	10%
Bag, Water Bottle, Pen	0	0%
Bag, Uniform, Pen	1	10%
Bag, Book, Pen	2	20%
Uniform, Water Bottle, Pen	1	10%
Uniform, Pen, Book	1	10%

Since only {Bag, Uniform, Water Bottle} data set is ~~having~~ satisfying the minimum support 30% therefore this dataset will pass to next iteration.

Part 2 - 4th Iteration

x = { Bag, Uniform, Water Bottle }

Subsets :- ~~1. { }~~ ×
2. {Bag}
3. {Uniform}
4. {Water Bottle}
5. {Bag, Uniform}
6. {Bag, Water Bottle}
7. {Uniform, Water Bottle}
~~8. {Bag, Uniform, Water Bottle}~~ ×

We never consider null set & the whole set.

Page No.
Date

{Bag} ⇒ {Uniform, Water Bottle}

$$= \frac{P(\text{Bag} \cap \text{Uniform} \cap \text{Watter Bottle})}{P(\text{Bag})}$$

$$= \frac{3}{\cancel{8}\cancel{4}\,2} \times \frac{\overset{25}{\cancel{50}}}{\cancel{100}}$$

$$= 37.5$$

{Uniform} ⇒ {Bag, Watter Bottle}

$$= \frac{P(\text{Bag} \cap \text{Uniform} \cap \text{Watter Bottle})}{P(\text{Uniform})}$$

$$= \frac{3}{7} \times 100$$

$$= 42.85\,\%$$

{Water Bottle} ⇒ {Bag, Uniform}

$$= \frac{P(\text{Bag} \cap \text{Uniform} \cap \text{WaterBottle})}{P(\text{WaterBottle})}$$

$$= \frac{3}{5} \times 100$$

$$= 60\,\%$$

{Uniform, Water Bottle} ⇒ {Bag}

$$= \frac{P(B \cap U \cap W)}{P(\text{Uniform, WaterBottle})}$$

$$= \frac{3}{4} \times 100$$

$$= 75\,\%$$

Page No.
Date

$\{Bag, WaterBottle\} \Rightarrow \{Uniform\}$

$$= \frac{P(B \cap U \cap W)}{P(Bag \cap Waterbottle)}$$

$$= \frac{3}{4} \times 100$$

$$= 75\%$$

$\{Bag, Uniform\} \Rightarrow \{WaterBottle\}$

$$= \frac{P(B \cap U \cap W)}{P(Bag \cap Uniform)}$$

$$= \frac{3}{5} \times 100$$

$$= 60\% \quad \times$$

Q. Find support & confidence % for the rule $\{Bag\} \rightarrow \{Uniform\}$

$$support = \frac{P(B \cap U)}{P0} = \frac{5}{10} \times 100 = 50\%$$

$$Confidence = \frac{P(B \cap U)}{P(B)} = \frac{5}{8} \times 100 = 62.5\%$$