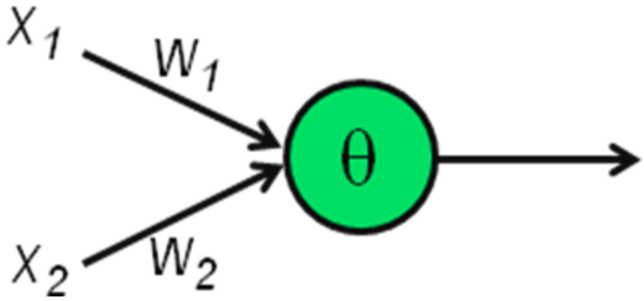
Faiz Ahmed	Mithun Das	Mutasim Fuad Ansari	Mohammad Abir Reza
1152231	1151651	1152109	1151705
stu225473@mail.uni-kiel.de	stu225039@mail.uni-kiel.de	stu225365@mail.uni-kiel.de	stu225093@mail.uni-kiel.de

Exercise 1 (Single-layer perceptron and Boolean functions with 2 inputs)

a) Show that the Boolean function XOR cannot be realized by a (single-layer) perceptron (with 2 inputs).

Note: The output y of a single-layer perceptron with 2 inputs x_1 and x_2, threshold θ and weights w_1 and w_2 is given by $y = \Theta[x_1 w_1 + x_2 w_2 - \theta]$
(Θ is the Heaviside function)



Answer

Boolean function can not be realized by a (single-layer) perceptron .
The Truth table of XOR is given below -

X1	X2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Given equation is y= Θ[x1w1+x2w2−θ]

Where,

- x1,x2 are inputs
- w1,w2 are weights.
- threshold θ
- Θ is Heaviside function.

From truth table, For x1 = 0 and x2 = 0 we have, 0 * w1 + 0 * w2 < θ implies, 0 < θ ------(1)

For x1 = 0 and x2 = 1 we have, 0 * w1 + 1 * w2 >= θ implies, w2 >= θ----(2)

For x1 = 1 and x2 = 0 we have, 1 * w1 + 0 * w2 >= θ implies, w1 >= θ ----(3)

For x1 = 1 and x2 = 1 we have, 1 * w1 + 1 * w2 < θ implies, w1 + w2 < θ----(4)

Here, threshold θ is positive. Applying the value of x1 and x2 on the equation 4 we get a contradiction. So, the Boolean function XOR cannot be realized by a (single-layer) perceptron.

b) Give all Boolean functions with 2 inputs (i.e. for each Boolean function: the output for each input combination) and indicate whether they can be realized by a (single-layer) perceptron.

Answer:

For 2 inputs there are 4 different input combinations.For 4 different input combinations we have 2ˆ4 = 16 different Boolean functions.

- i. F1 = AND:

X1	X2	AND
0	0	0
0	1	0
1	0	0
1	1	1

Indication : Can be realized by Single layer perceptron.

- ii. F2 = FALSE:

X1	X2	FALSE
0	0	0
0	1	0
1	0	0
1	1	0

Indication : Realized by Single layer perceptron.

- iii. f3 = x1 ^ ¬x2

x1	x2	x1 ^ ¬x2
0	0	0
0	1	0
1	0	1
1	1	0

Indication : Realized by Single layer perceptron.

- iv. F4 = x1

X1	X2	X1
0	0	0
0	1	0
1	0	1
1	1	1

Indication : Realized by Single layer perceptron.

- v. F5 = ¬ x1 ∧ x2

X1	X2	¬ x1 ∧ x2
0	0	0
0	1	1
1	0	0
1	1	0

Indication : Realized by Single layer perceptron.

- vi. F6 = x2

X1	X2	X2
0	0	0
0	1	1
1	0	0
1	1	1

Indication : Realized by Single layer perceptron.

- vii. F7 = XOR

X1	X2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Indication : Can not be Realized by Single layer perceptron.

- viii. F8 = OR

X1	X2	OR
0	0	0
0	1	1
1	0	1
1	1	1

Indication : Realized by Single layer perceptron.

- ix. F9 = NOR

X1	X2	NOR
0	0	1
0	1	0
1	0	0
1	1	0

Indication : Realized by Single layer perceptron.

- x. F10 = XNOR

X1	X2	XNOR
0	0	1
0	1	0
1	0	0
1	1	1

Indication : Can not be Realized by Single layer perceptron.

- xi. F11 = ¬ x2

X1	X2	¬ x2
0	0	1
0	1	0
1	0	1
1	1	0

Indication : Realized by Single layer perceptron.

- xii. F12 = x1 ∨ ¬x2

X1	X2	x1 ∨ ¬x2
0	0	1
0	1	0
1	0	1
1	1	1

Indication : Realized by Single layer perceptron.

- xiii. F13 = ¬x1

X1	X2	¬x1
0	0	1
0	1	1
1	0	0
1	1	0

Indication : Realized by Single layer perceptron.

- xiv. F14 = ¬x1 ∨ x2

X1	X2	¬x1 ∨ x2
0	0	1
0	1	1
1	0	0
1	1	1

Indication : Realized by Single layer perceptron.

- xv. F15 = NAND

X1	X2	NAND
0	0	1
0	1	1
1	0	1
1	1	0

Indication : Realized by Single layer perceptron.

- xvi. F16 = TRUE

X1	X2	TRUE
0	0	1
0	1	1
1	0	1
1	1	1

Indication : Realized by Single layer perceptron.

c) Select three Boolean functions with two inputs and give values for the synaptic weights w_1,w_2 and threshold θ so that the Boolean function is realized by a single-layer perceptron. Show for each of the three Boolean functions and each input pair that the Boolean function is indeed realized by the chosen combination of weights and threshold.

Answer

We have to give values for synaptic weights w1, w2 and threshold θ for three different Boolean functions, where the equation is: y = Θ [x1 * w1 + x2 * w2 – θ]

AND :

Let's assume w1 = 1.0, w2 = 1.0, θ = 1.5

- For x1 = 0 and x2 = 0 output is 0.

We have: 0 * 1.0 + 0 * 1.0 - 1.5 = -1.5. Therefore, Θ[-1.5] = 0

- For x1 = 0 and x2 = 1 output is 0.

We have: 0 * 1.0 + 1 * 1.0 - 1.5 = -0.5. Therefore, Θ[-0.5] = 0

- For x1 = 1 and x2 = 0 output is 0.

We have: 1 * 1.0 + 0 * 1.0 - 1.5 = -0.5. Therefore, Θ[-0.5] = 0

- For x1 = 1 and x2 = 1 output is 1.

We have: 1 * 1.0 + 1 * 1.0 - 1.5 = 0.5. Therefore, Θ[0.5] = 1

So, AND function is realized by a single-layer perceptron.

NOR :

Let's assume w1 = -1.0, w2 = -1.0, θ = -0.5

- For x1 = 0 and x2 = 0 output is 0. We have: 0 * -1.0 + 0 * -1.0 + 0.5 = 0.5. Therefore, Θ[0.5] = 1
- For x1 = 0 and x2 = 1 output is 0. We have: 0 * -1.0 + 1 * -1.0 + 0.5 = -0.5. Therefore, Θ[-0.5] = 0
- For x1 = 1 and x2 = 0 output is 0. We have: 1 * -1.0 + 0 * -1.0 + 0.5 = -0.5. Therefore, Θ[-0.5] = 0
- For x1 = 1 and x2 = 1 output is 1. We have: 1 * -1.0 + 1 * -1.0 + 0.5 = -1.5. Therefore, Θ[-1.5] = 0

So, NOR function is realized by a single-layer perceptron.

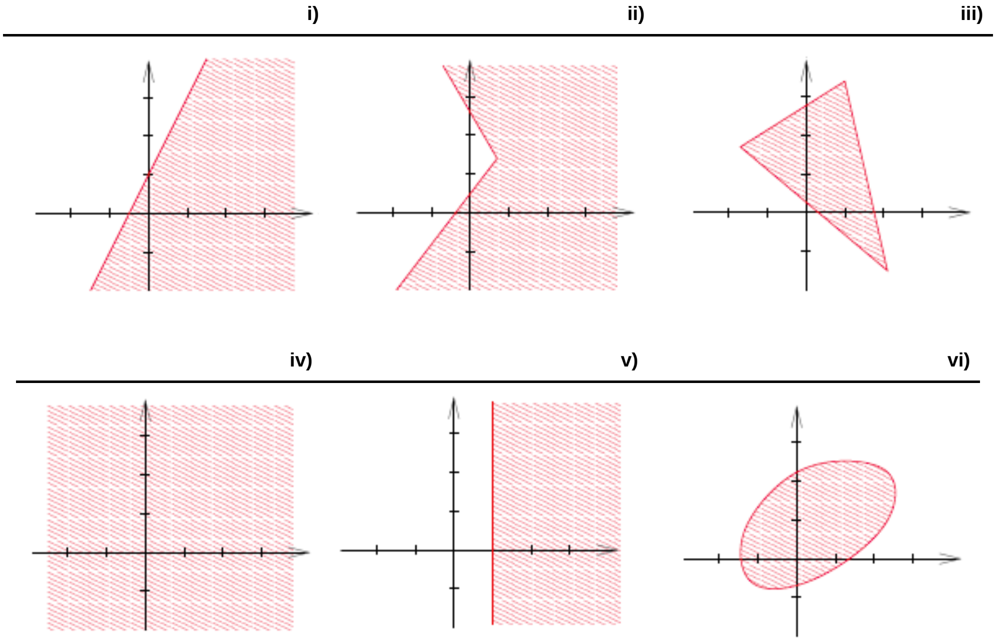
OR :

Let's assume w1 = 1.0, w2 = 1.0, θ = 0.5

- For x1 = 0 and x2 = 0 output is 0. We have: 0 * 1.0 + 0 * 1.0 - 0.5 = -0.5. Therefore, Θ[-0.5] = 0
- For x1 = 0 and x2 = 1 output is 0. We have: 0 * 1.0 + 1 * 1.0 - 0.5 = 0.5. Therefore, Θ[0.5] = 1
- For x1 = 1 and x2 = 0 output is 0. We have: 1 * 1.0 + 0 * 1.0 - 0.5 = 0.5. Therefore, Θ[0.5] = 1
- For x1 = 1 and x2 = 1 output is 1. We have: 1 * 1.0 + 1 * 1.0 - 0.5 = 1.5. Therefore, Θ[1.5] = 1

So, OR function is realized by a single-layer perceptron.

d) Which of the following partitioning of $\mathfrak{R}^2$ can be realized by a single-layer perceptron with two inputs? For those that can be realized, give weights and threshold of the perceptron. (Consider abscissa as x_1 and ordinate as x_2).



(From: Riedmiller)

Answer

i. 1st Graph: This Graph is linearly Separable into two different regions. So, this Geometrical Representation can be realized by a single-layer perceptron.

ii. 2nd Graph : As it is not straight line, this Geometrical Representation cannot be realized by a single-layer perceptron.

iii. 3rd Graph: As it is not straight line, this Geometrical Representation cannot be realized by a single-layer perceptron

iv. 4th Graph : This graph is as similar as Boolean TRUE function.For any input value of x1 and x2 we will get output from neuron. So, this Geometrical Representation can be realized by a single-layer perceptron.

v. 5th Graph : This Graph is linearly Separable into two different regions. So, this Geometrical Representation can be realized by a single-layer perceptron.

vi. 6th Graph : As it is not straight line, this Geometrical Representation cannot be realized by a single-layer perceptron.

Exercise 2 (Types of neural networks, synaptic weight matrix)

a) Explain the following terms related to neural networks:

- Boolean function
- Feedforward neural network
- Recurrent neural network
- Multi-layer perceptron

Answer

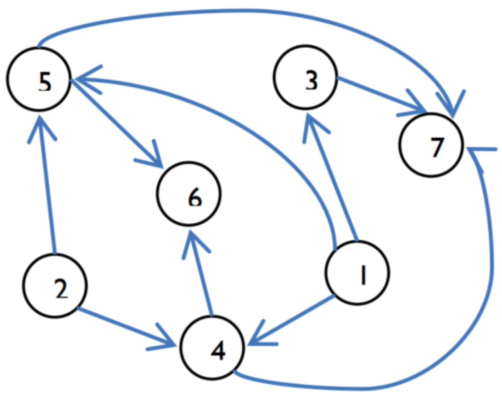
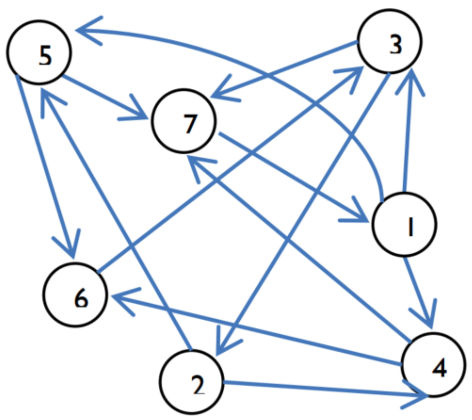
Boolean Functon: A function whose domain and co-domain belongs only to a two-element set (0,1). The function performs basic logic operations ***And***, ***Or***, ***Not***. For n inputs 2 exponent 2 exponent n numbers of boolean functions get generated.

Feedforward neural network: A network structure where connections between neurones are only allowed from lower to higher layers. Any kind of loop, feedback/lateral feedback, direct or indirect path from a higher layer neurone to a lower layer neurone is not allowed. There are tow types of feedforward neural network such as first order and secon order feedforward neural network.

Recurrent neural network: A network structure where connections between neurones are allowed from any layer. Here cycles and anu kind of feedbacks are allowed.

Multilayer perceptron: Combination of some perceptrons in a feedforward network which are organized in three layers such as, ***input***, ***hidden*** and ***output layer***. Input layer recieves input pattern, output layer represents the response of the network. Remaining layers are considered as hidden layers. Each layer can recieve inputs only from higher layers.

b) Specify whether the following artificial neural networks are feedforward or recurrent neural networks and explain your selection.

i)	ii)
	

Answer

i) Feedforward neural network.

No cycle, feedback, lateral feedback and connections from higher layer to lower layer is found.

ii)Recurrent neural network.

Cycle exist between neurones (1->3->7) and (1->4->7)connection from higher to lower level neurone (6,3).

c) Using the neuron numbers from 1 to 7 given in the circles, fill out the following general weight matrix by marking the corresponding field entries. Example: Mark the field in row i and column j (weight w_{ij}) if there is a connection from neuron j to neuron i.

	1	2	3	4	5	6	7
1							
2							
3							
4							
5							
6							
7							

Answer

Weight matrix corresponding to exercise (b) (i).

	1	2	3	4	5	6	7
1							
2							
3	x						
4	x	x					
5	x	x					
6				x	x		
7			x	x	x		

Weight matrix corresponding to exercise (b) (ii).

	1	2	3	4	5	6	7
1							x
2			x				
3	x					x	
4	x	x					
5		x					
6				x	x		
7			x	x	x		

Exercise 3 (Computing the output of a feedforward neural network)

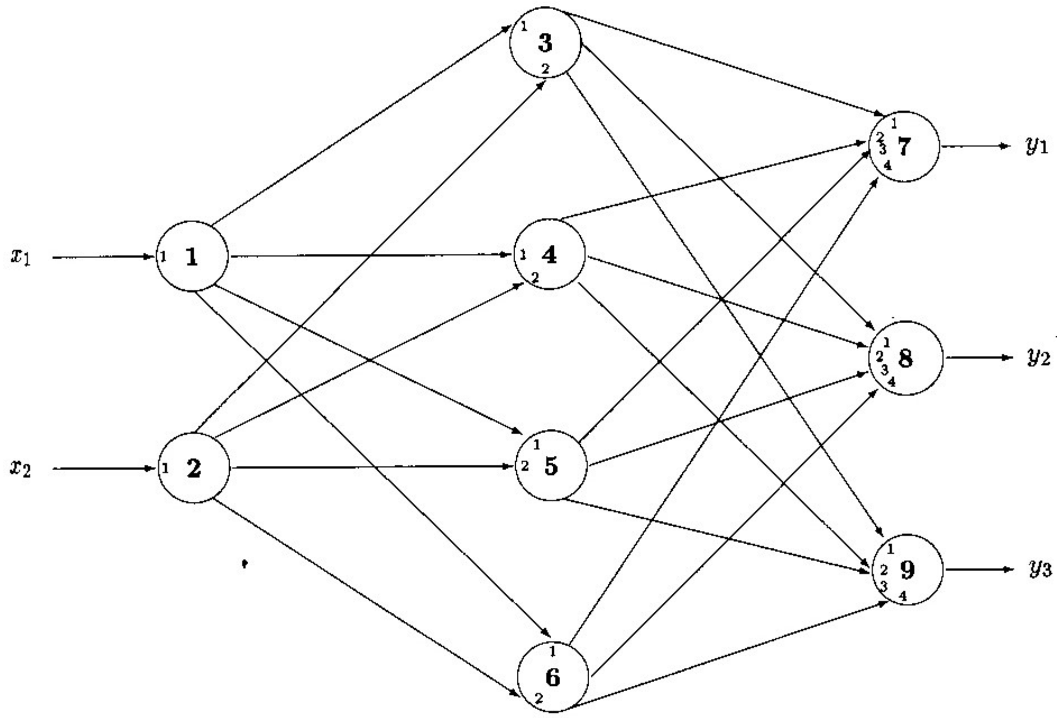
a) Compute the output of the following feedforward neural network for the input $x_1 = 3; x_2 = 1$. Which neurons can be computed in parallel, which have to wait?

Note:

- The small numbers in each circle correspond to the components of the weight vector; see example below. In this part of the exercise, the threshold is set to θ =0 for all neurons.
- c is the slope of the linear activation function: $f(h) = c \cdot h$
- “Threshold element” means that the activation function is the Heaviside function

Neuron	Activation function of neuron	Weight vector
1	Linear; c=1	(1)
2		(1)
3	Threshold element; θ =0	(1,-2)
4		(-1,0)
5		(3,2)
6		(0,2)
7	Linear; c=1	(0,2,-3,1)
8		(1,-2,3,8)
9		(0,2,3,-4)

The network:



Example for weight vector of neuron 8:
1st component of weight vector (1) refers to connection neuron 3 → neuron 8
2nd component of weight vector (-2) refers to connection neuron 4 → neuron 8
3rd component of weight vector (3) refers to connection neuron 5 → neuron 8
4th component of weight vector (8) refers to connection neuron 6 → neuron 8

(Source: Stefan Hartmann, Cesar Research)

a) The weight matrix corresponding of the network:

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1									
2									
3	x	x							
4	x	x							
5	x	x							
6	x	x							
7			x	x	x				
8			x	x	x				
9			x	x	x				

Assume,

inputs = x1,x2 , outputs = y1,y2,y3 , position of neuron = p[i] , weight = w[i][j]

1st Step Calculation : Form chart we get, x1 = 3 , x2 =1, w[1][0] = 1, w[2][0] = 1.

c is the slope of the linear activation function: f(h) = c.h where c = 1

p[1] = (3 * 1) = 3 P[2] = (1 * 1) = 1

2nd Step Calculation : Form chart we get, Threshold θ = 0, w[3][1] = 1, w[3][2] = -2, w[4][1] = -1, w[4][2] = 0, w[5][1] = 3, w[5][2] = 2, w[6][1] = 0, w[6][2] = 2

p[3] = Θ (3 * 1 + 1 * (-2) - 0) = Θ(1) = 1

p[4] = Θ (3 * (-1) + 1 * 0 - 0) = Θ(-3) = 0

p[5] = Θ (3 * (3) + 1 * 2 - 0) = Θ(11) = 1

p[6] = Θ (3 * 0 + 1 * 2 - 0) = Θ(2) = 1

3rd Step Calculation : Form chart we get, Linear c = 1, w[7][3] = 0, w[7][4] = 2, w[7][5] = -3, w[7][6] = 1, w[8][3] = 1, w[8][4] = -2, w[8][5] = 3, w[8][6] = 8, w[9][3] = 0, w[9][4] = 2, w[9][5] = 3, w[9][6] = -4

p[7] = (1 * 0 + 0 * 2 + 1 * (-3) + 1 * 1 - 0) = -2

p[8] = (1 * 1 + 0 * (-2) + 1 * 3 + 1 * 8 - 0) = 12

p[9] = (1 * 0 + 0 * 2 + 1 * 3 + 1 * (-4) - 0) = -1

The values are y1 = -2 , y2 = 12 , y3 = -1

After getting the solution we can write that Neuron (1 and 2) , Neuron (3,4,5) and Neuron (7,8,9) can compute in parallel in the group and one step had to wait as long as previous step get executed.

b) Assume the following weight matrix, where an entry w_{ij} (ith row, jth column) corresponds to the synaptic weight from neuron j to neuron i. (No entry means the synaptic weight is 0).

Further assume that the activation function of the neurons of hidden layer 2 (neurons 8, 9 and 10) is linear (with slope $c = 1$), whereas the activation function of all other neurons is a Heaviside step function.

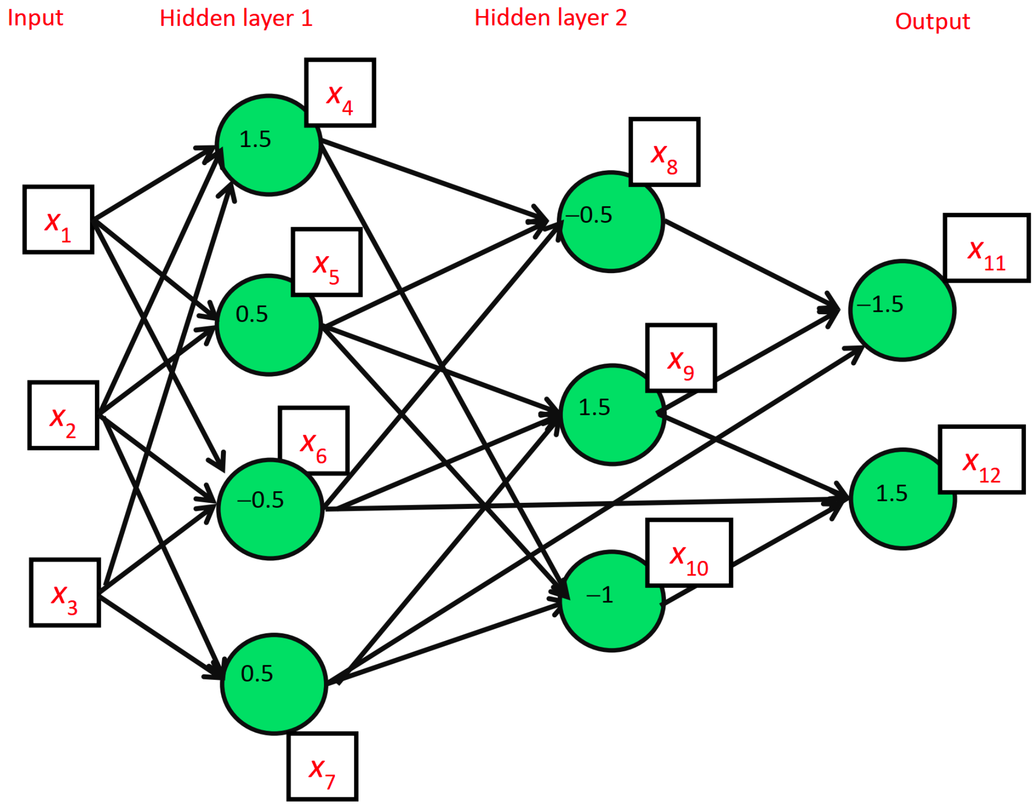
In this part of the exercise, the threshold θ of each node is indicated in the network graph as number in the corresponding neuron.

Compute the output of the following feedforward neural network for the inputs $x_1 = 1, x_2 = 0, x_3 = 1$ and $x_1 = 0, x_2 = 1, x_3 = 1$.

Weight matrix:

	1	2	3	4	5	6	7	8	9	10	11	12
1												
2												
3												
4	−2	5	−4									
5	1	−2										
6	3	−1	6									
7		7	1									
8				−1	4	−2						
9					−3	5	1					
10				8	2		−3					
11							6	1	−2			
12						1			−4	3		

Network:



Note: this is a feedforward neural network of second order

Answer

b) For x1 = 1, x2 = 0, x3 = 1

From Weight matrix we get the value :

1st Step Calculation :

w[4][1] = -2 , w[4][2] = 5, w[4][3] = -4, w[5][1] = 1 , w[5][2] = -2, w[6][1] = 3, w[6][2] = -1, w[6][3] = 6 w[7][2] = 7, w[7][3] = 1

θ4 = 1.5, θ5 = 0.5, θ6 = -0.5, θ7 = 0.5

x4 = Θ (1 * (-2) + 0 * 5 + 1 * (-4) - 1.5) = Θ(-7.5) = 0

x5= Θ (1 * 1 + 0 * (-2) - 0.5) = Θ(0.5) = 1

x6 = Θ (1 * 3 + 0 * (-1) + 1 * 6 + 0.5) = Θ(9.5) = 1

x7 = Θ (0 * 7 + 1 * 1 – 0.5) = Θ(0.5) = 1

2nd Step Calculation :

w[8][4] = -1 , w[8][5] = 4, w[8][6] = -2

w[9][5] = -3 , w[9][6] = 5, w[9][7] = 1

w[10][4] = 8, w[6][5] = 2, w[6][7] = -3

θ [8] = -0.5, θ[9] = 1.5, θ[10] = -1

activation function c = 1.

x8 = 0 * (-1) + 1 * 4 + 1 * (-2) + 0.5 = 2.5

x9 = 1 * (-3) + 1 * 5 + 1 * 1 – 1.5= 1.5

x10 = 0 * 8 + 1 * 2 + 1 * (-3) + 1= 0

3rd Step Calculation :

w[11][7] = 6, w[11][8] = 1, w[11][9] = -2

w[12][6] = 1, w[11][9] = -4, w[11][10] = 3

θ [11] = -1.5, θ[12] = 1.5

X11 = Θ (1 * 6 + 2.5 * 1 + 1.5 * (-2) + 1.5) = Θ(7) = 1

X12 = Θ (1 * 1 + 1.5 * (-4) + 0 * 3 - 1.5) = Θ(-6.5) = 0

Outputs are x11 = 1 and x12 = 0

For x1 = 0, x2 = 1, x3 = 1

1st Step Calculation :

w[4][1] = -2 , w[4][2] = 5, w[4][3] = -4

w[5][1] = 1 , w[5][2] = -2

w[6][1] = 3, w[6][2] = -1, w[6][3] = 6

w[7][2] = 7, w[7][3] = 1

θ [4] = 1.5, θ[5] = 0.5, θ[6] = -0.5, θ[7] = 0.5

x4 = Θ (0 * (-2) + 1 * 5 + 1 * (-4) - 1.5) = Θ(-0.5) = 0

x5= Θ (0 * 1 + 1 * (-2) - 0.5) = Θ(-2.5) = 0

x6 = Θ (0 * 3 + 1 * (-1) + 1 * 6 + 0.5) = Θ(5.5) = 1

x7 = Θ (1 * 7 + 1 * 1 – 0.5) = Θ(7.5) = 1

2nd Step Calculation :

w[8][4] = -1 , w[8][5] = 4, w[8][6] = -2

w[9][5] = -3 , w[9][6] = 5, w[9][7] = 1

w[10][4] = 8, w[6][5] = 2, w[6][7] = -3

θ [8] = -0.5, θ[9] = 1.5, θ[10] = -1

x8 = 0 * (-1) + 0 * 4 + 1 * (-2) + 0.5 = -1.5

x9 = 0 * (-3) + 1 * 5 + 1 * 1 – 1.5= 4.5

x10 = 0 * 8 + 0 * 2 + 1 * (-3) + 1= -2

3rd Step Calculation :

w[11][7] = 6, w[11][8] = 1, w[11][9] = -2

w[12][6] = 1, w[11][9] = -4, w[11][10] = 3

θ [11] = -1.5, θ[12] = 1.5

X11 = Θ (1 * 6 + (-1.5) * 1 + 4.5 * (-2) + 1.5) = Θ(-3) = 0

X12 = Θ (1 * 1 + 4.5 * (-4) + (-2) * 3 - 1.5) = Θ(-24.5) = 0

outputs are x[11] = 0 and x[12] = 0

Exercise 4 (Multi-layer perceptron and XOR):

a) Find a multi-layer perceptron which realizes the Boolean function XOR. Demonstrate that the found perceptron indeed performs XOR on all possible input pairs.

Answer

Table 1 : Truth Table of XOR

X1	X2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Table 2 : Another way of represting XOR

X1	X2	(X1 OR X2)	NOT(X1 AND X2)	(X1 OR X2) AND (NOT(X1 AND X2))
0	0	0	1	0
0	1	1	1	1
1	0	1	1	1
1	1	1	0	0

We can not demonstrate XOR with signle layer Perceptron. From "Table 2" we can say that, XOR could be demonstrated with three Perceptron. In the following picture h1 and h2 are two hidden layar perceptron and y is the output(XOR).

Figure 1: Multi-layer perceptron (I collected the following picture from internet)

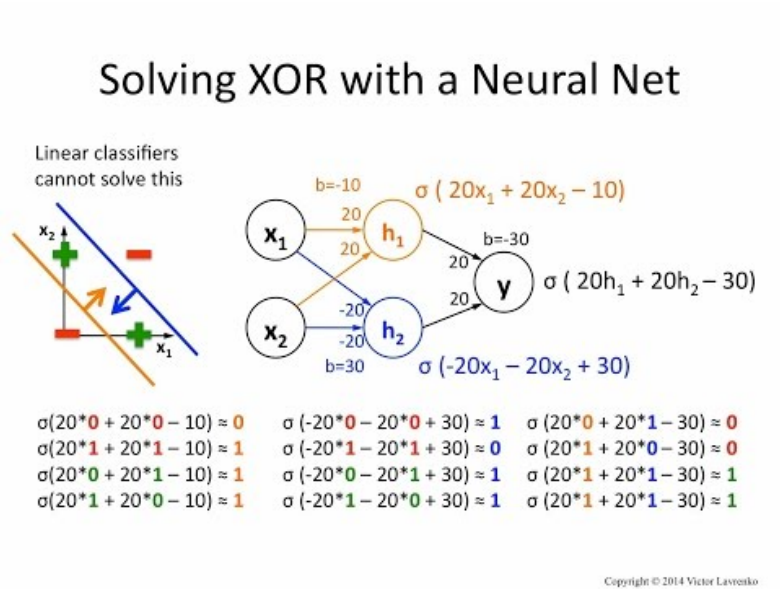


Table 3 : Multi-layer Perceptron

X1	X2	h1	h2	y
0	0	0	1	0
0	1	1	1	1
1	0	1	1	1
1	1	1	0	0

Here, perceptron h1 works as logical OR, h2 works as logical NAND the final output y is the result of the multi-layer perceptron, which is similar with XOR.

b) Find a perceptron with two (binary) inputs which realizes the function

$$F(x_1, x_2) = \begin{cases} 1 & \text{if } x_1 + x_2 = 1 \\ 0 & \text{else} \end{cases}$$

Note: “+” denotes mathematical addition.

Answer

The truth table of the function F is following.

x1	x2	F(x1,x2)
0	0	0
0	1	1
1	0	1
1	1	0

If we have a close look at the table above, we can say that the function F(x1,x2) is another way of representing XOR. So the demonstrated perceptron at 'Figure 1' (i.e. Answer 4(a)) with two inputs can realize the funciton F(x1,x2).