Group 35

Name: Shaokat Hossain

Id: 1144107

Name: Md Abu Noman Majumdar

Id: 1144101

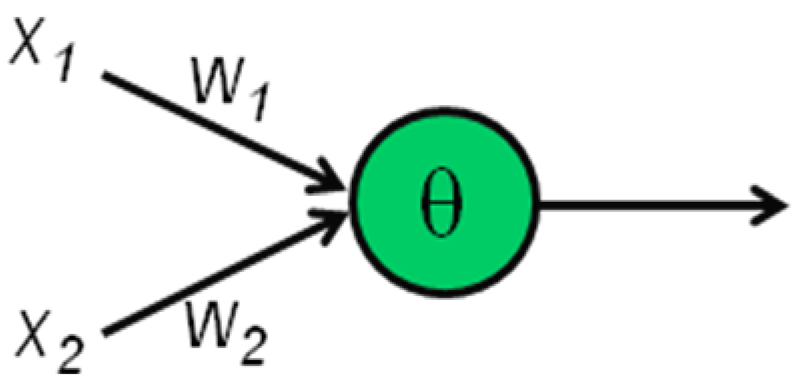
Name: Abdullah Al Amin

Id: 1152610

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

The output of an XOR gate is 1 only when exactly one of its inputs is 1.

x1 = 1 x2 = 0

1.w1 + 0.w2 >= θ ---- (1)

x1 = 0 x2 = 1

0.w1 + 1.w2 >= θ ---- (2)

x1 = 0 x2 = 0

$0.w1 + 0.w2 < \theta$ ---- (3)

x1 = 1 x2 = 1

$1.w1 + 1.w2 < \theta$ ---- (4)

we find a contradiction in equation (3) and (4) because the summation of both weights cannot be less than threshold You cannot draw a straight line to separate the points, therfore, a single-layer perceptron can't implement XOR

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

OR(Linearly seperable)

x1	x2	ouput
0	0	0
0	1	1
1	0	1
1	1	1

AND(Linearly seperable)

x1	x2	ouput
0	0	0
0	1	0
1	0	0
1	1	1

XOR(Linearly not seperable)

x1	x2	ouput
0	0	0
0	1	1
1	0	1
1	1	0

NOR(Linearly seperable)

x1	x2	ouput
0	0	1
0	1	0
1	0	0

x1	x2	ouput
1	1	0

NAND(Linearly seperable)

x1	x2	ouput
0	0	1
0	1	1
1	0	1
1	1	0

XNOR(Linearly not seperable)

x1	x2	ouput
0	0	1
0	1	0
1	0	0
1	1	1

from the above truth tables we can see that xor and xnor are not linearly seperable, thus they cannot be realised through single layer perceprton and all the others can be since they are linearly seperable.

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.

AND

let us assume w1 = 1.0, w2 = 1.0, θ = 1.5

For x1 = 0 , x2 = 0, output: 0 * 1.0 + 0 * 1.0 - 1.5 = -1.5. so, Θ[-1.5] = 0

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

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

For x1 = 1 and x2 = 1 output: 1 * 1.0 + 1 * 1.0 - 1.5 = 0.5. so, Θ[0.5] = 1

OR

let us assume w1 = 1.0, w2 = 1.0, θ = 0.5

For x1 = 0 , x2 = 0, output: 0 * 1.0 + 0 * 1.0 - 0.5 = -0.5. so, Θ[-0.5] = 0

For x1 = 0 and x2 = 1, output: 0 * 1.0 + 1 * 1.0 - 0.5 = 0.5. so, Θ[0.5] = 1

For x1 = 1 and x2 = 0, output: 1 * 1.0 + 0 * 1.0 - 0.5 = 0.5. so, Θ[0.5] = 1

For x1 = 1 and x2 = 1 output: 1 * 1.0 + 1 * 1.0 - 0.5 = 0.5. so, Θ[1.5] = 1

NAND

let us assume w1 = -1.0, w2 = -1.0, θ = -1.5

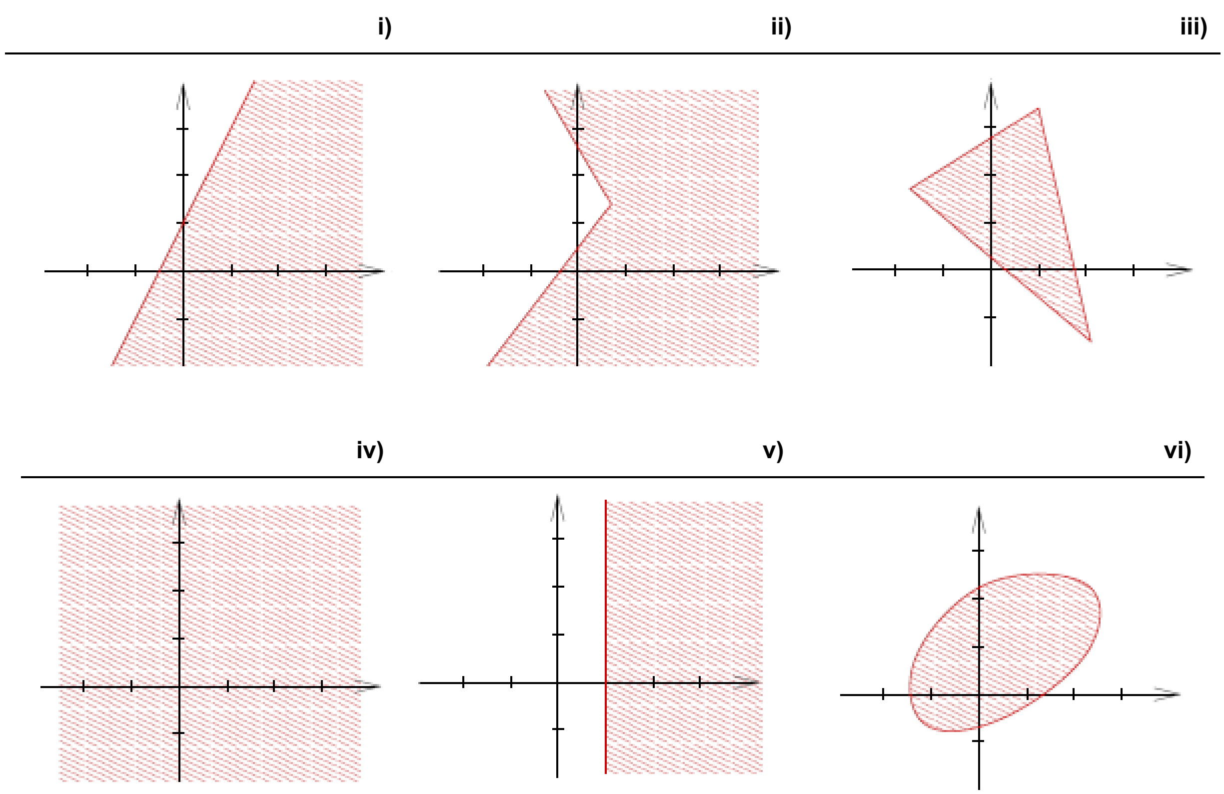
For x1 = 0 , x2 = 0, output: 0 * (-1.0) + 0 * (-1.0) - (-1.5) = 1.5. so, Θ[1.5] = 1

For x1 = 0 and x2 = 1, output: 0 * (-1.0) + 1 * (-1.0) - (-1.5) = 0.5. so, Θ[0.5] = 1

For x1 = 1 and x2 = 0, output: 1 * (-1.0) + 0 * (-1.0) - (1.5) = 0.5. so, Θ[0.5] = 1

For x1 = 1 and x2 = 1 output: 1 * (-1.0) + 1 * (-1.0) - (-1.5) = -0.5. so, Θ[-0.5] = 0

d) Which of the following partitioning of $\Re^2$ 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) Region is linearly seperated. Can be realised through single layer perceprtron. w1 = 2.0, w2 = -1.0, θ = -1.0

ii) Region is not linearly seperated. Cannot be realised.

iii) Region is not linearly seperated. Cannot be realized.

iv) Region is not seperated. So for any value it would be true. Similar to the TRUE function. so it can be realised through single layer perceptron. w1 = 0.0, w2 = 0.0, θ = 0.0

v) Region is linearly seperated. Can be realised through single layer perceprtron. w1 = 1.0, w2 = 0.0, θ = 1.0

vi) Region is not linearly seperated. Cannot be realized.

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 function: A function with output from a two-element set (usually {0,1}).

Feedforward neural network: A feedforward neural network is an artificial neural network where connections between the nodes do not form a cycle. The name feedforward comes from the nature of the connections between the perceptrons of the neural network as it goes only forward.

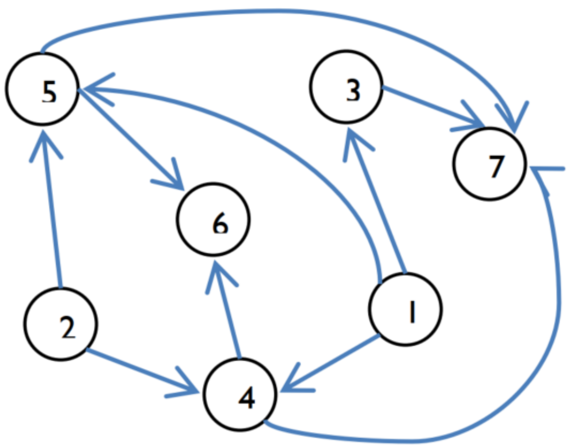
Recurrent neural network: A neural network is a ANN where the data flow is bi-directional and any kind of connection is possible without even layers.

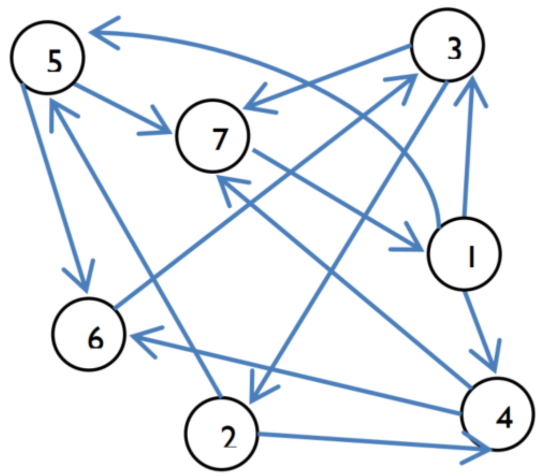
Multi-layer perceptron: A kind of ANN where atleast 3 layer is present. input, output and hidden layer. There can be more than one hidden layers in between.

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

i)	ii)

i)	ii)



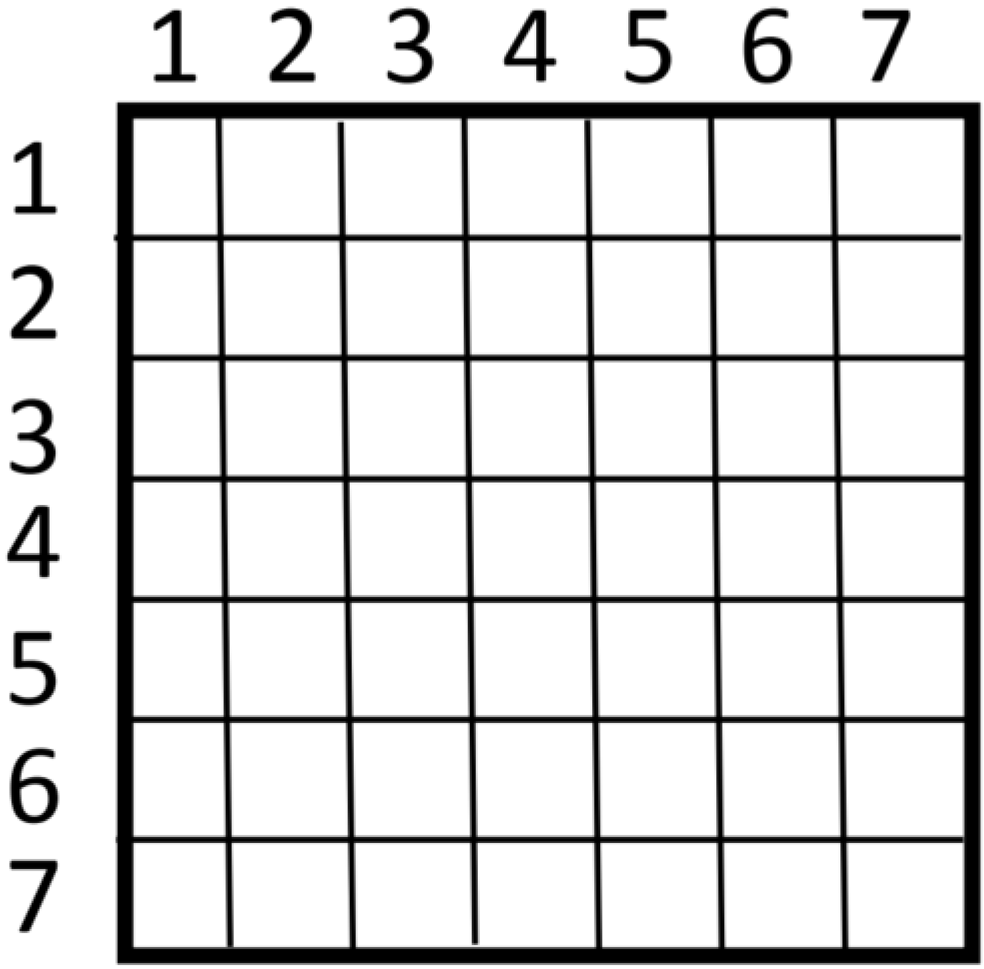


Answer

i) It is a feedforward neural network because there is no connection cycle(feedback loop) and the connections are layered(going from lower to higher)

ii) It is a recurrent neural network because there exists feedback loop like 7->1->4->7 , 7->1->3->7 etc

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.



Answer

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		

ii)

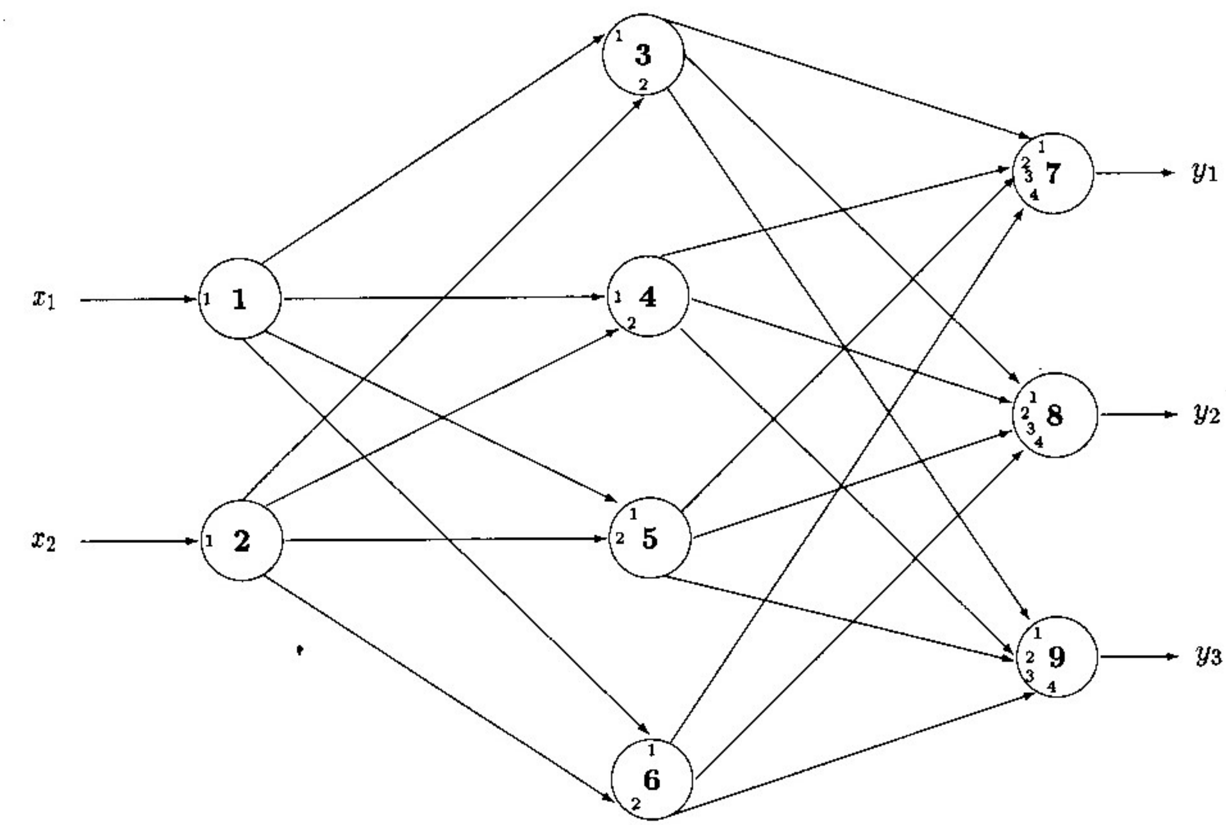
	1	2	3	4	5	6	7
1							x
2			x				

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

The network:



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; $\theta = 0$	(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)

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)

Answer

State of neurons: s[ ]

First layer:

x[1] = 3 , x[2] =1, w[1][0] = 1, w[2][0] = 1.
linear activation function with slope c = 1.
s[1] = (3 * 1) = 3
s[2] = (1 * 1) = 1

2nd layer:

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 and
threshold element θ = 0.

s[3] = Θ(3 * 1 + 1 * (-2) - θ) = Θ(1) = 1
s[4] = Θ(3 * (-1) + 1 * 0 - θ) = Θ(-3) = 0
s[5] = Θ(3 * (3) + 1 * 2 - θ) = Θ(11) = 1
s[6] = Θ(3 * 0 + 1 * 2 - θ) = Θ(2) = 1

3rd layer

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

linear activation function with slope c = 1.

s[7] = (1 * 0 + 0 * 2 + 1 * (-3) + 1 * 1 - 0) = -2
s[8] = (1 * 1 + 0 * (-2) + 1 * 3 + 1 * 8 - 0) = 12
s[9] = (1 * 0 + 0 * 2 + 1 * 3 + 1 * (-4) - 0) = -1

output

y[1] = -2
y[2] = 12
y[3] = -1

output

y[1] = -2
y[2] = 12
y[3] = -1

i. Neuron 1 and 2 can be computed in parallel.
ii. Neuron 3, 4, 5 and 6 can be computed in parallel.
iii. Neuron 7, 8 and 9 can be computed in parallel.
Computation of neurons in a layer dependent on the computation of the previous layer.

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.

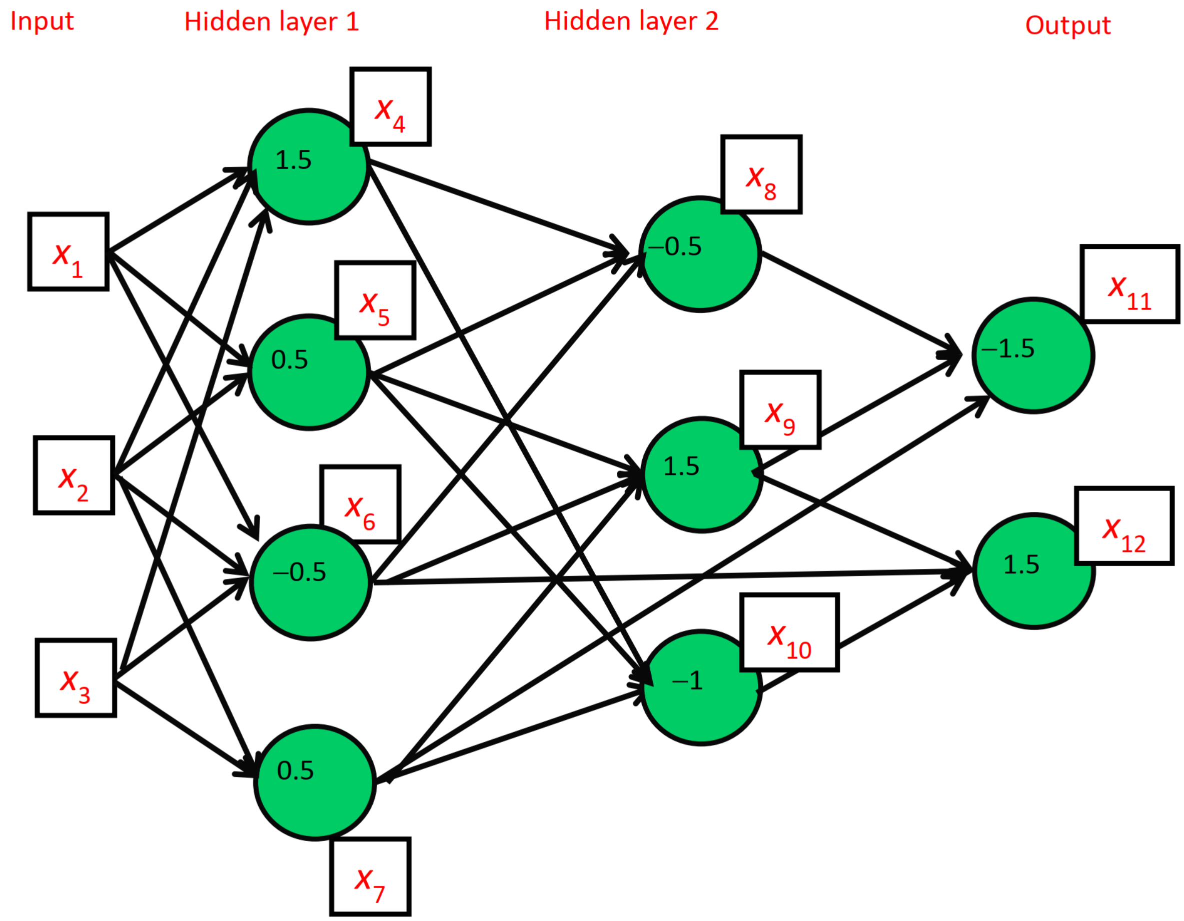
In this part of the exercise, the threshold θ of each node is indicated in the network graph as number in the corresponding neuron.

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

First layer:

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 (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

Answer

First layer:

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 (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

3rd layer:

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

X[11] =Θ(1 * 6 + 2.5 * 1 + 1.5 * (-2) + 1.5) = Θ(7) = 1
X[12] = Θ(1 * 1 + 1.5 * (-4) + 0 * 3 - 1.5) = Θ(-6.5) = 0

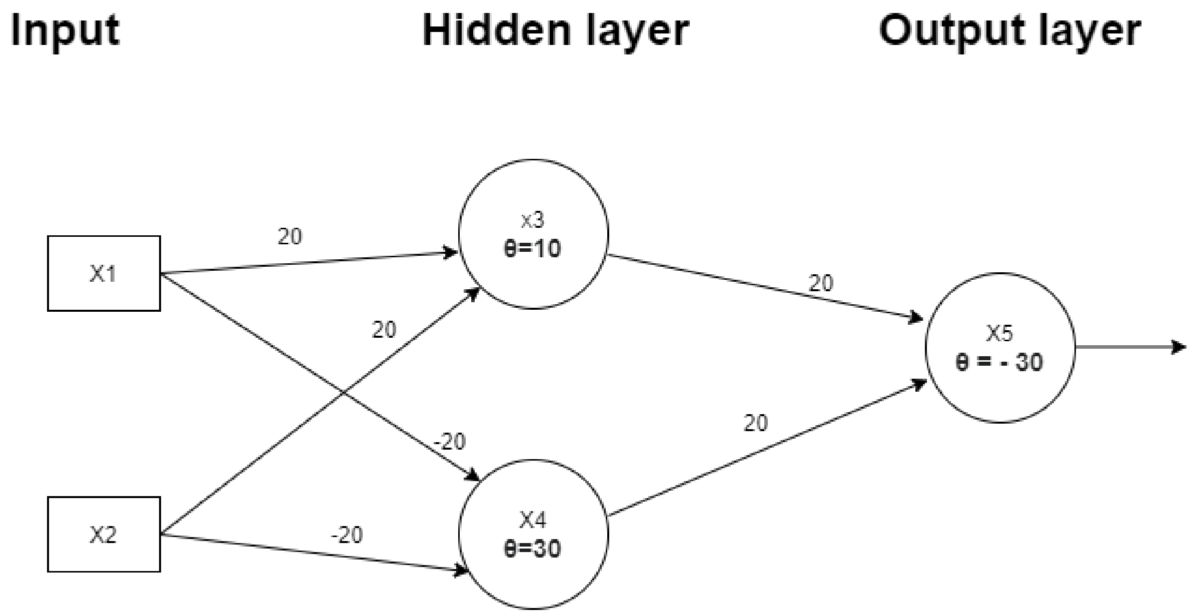
output

x[11] = 1 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



Assinging weights

w[3][1] = 20, w[3][2] = 20
w[4][1] = -20, w[4][2] = -20
w[5][3] = 20, w[5][4] = 20
θ [3] = 10, θ[4] = 30, θ [5] = -30

when x1 = 0, x2 = 0

second layer

x[3] = Θ (0 $*$ 20 + 0 * 20 - 10) = Θ(-10) = 0
x[4] = Θ (0 * -20 + 0 * -20 + 30) = Θ(30) = 1

Third layer

x[5] = Θ(0 * 20 + 1 * 20 - 30) = Θ(-10) = 0

when x1 = 0, x2 = 1

second layer

x[3] = Θ (0 $*$ 20 + 1 * 20 - 10) = Θ(10) = 1
x[4] = Θ (0 * -20 + 1 * -20 + 30) = Θ(10) = 1

Third layer

x[5] = Θ(1 * 20 + 1 * 20 - 30) = Θ(10) = 1

when x1 = 1, x2 = 0

second layer

x[3] = Θ (1 $*$ 20 + 0 * 20 - 10) = Θ(10) = 1
x[4] = Θ (1 * -20 + 0 * -20 + 30) = Θ(10) = 1

Third layer

x[5] = Θ(1 * 20 + 1 * 20 - 30) = Θ(10) = 1

when x1 = 1, x2 = 1

second layer

x[3] = Θ (1 $*$ 20 + 1 * 20 - 10) = Θ(30) = 1
x[4] = Θ (1 * -20 + 1 * -20 + 30) = Θ(-10) = 0

Third layer

x[5] = Θ(1 * 20 + 0 * 20 - 30) = Θ(-10) = 0

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

if we observe the the function closely we can see that it will yield 1 for if and only if x1=1 & x2=0 and x1=0 & x2=1. This is the exact same behaviour of the XOR function. and its perceptron is described in previous answer