Prof. Dr. Carsten Meyer
Faculty of Engineering
Christian-Albrechts-Universität Kiel

Neural Networks and Deep Learning – Summer Term 2018

Exercise sheet 2

Submission due: Tuesday, May 08, 11:30 sharp

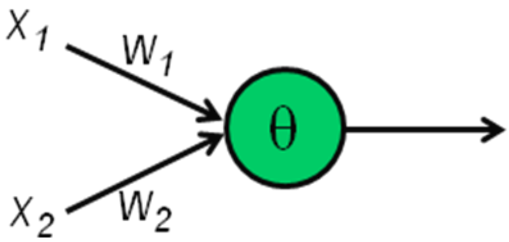
Exercise 1 (Single-layer perceptron and Boolean functions with 2 inputs):

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)

Solution:



Boolean function XOR:

x_1	x_2	XOR
0	0	0
0	1	1
1	0	1
1	1	0

Insert all possible input pairs into the equation of a single-layer perceptron and formulate for each input pair a threshold condition so that the target output is realized; then rearrange the equations and formulate a contradiction:

$$
\begin{array}{llll}
0\cdot w_1 + 0\cdot w_2 < \theta & \theta > 0 & (1) & \\
0\cdot w_1 + 1\cdot w_2 \ge \theta & w_2 \ge \theta & (2) & (2)+(3):\ w_1 + w_2 \ge 2\cdot\theta \\
1\cdot w_1 + 0\cdot w_2 \ge \theta & w_1 \ge \theta & (3) & (4):\ \ w_1 + w_2 < \theta \\
1\cdot w_1 + 1\cdot w_2 < \theta & w_1 + w_2 < \theta & (4) &
\end{array}
\Leftrightarrow \quad \Leftrightarrow
$$

The last two equations cannot hold simultaneously (contradiction) if $\theta > 0$ (first equation)
$\rightarrow$ 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.

Solution:

- Inputs x_1 and x_2, 16 Boolean functions $f_0, \ldots, f_{15}$
- 14 linearly separable Boolean functions indicated by "+", 2 not linearly separable Boolean functions indicated by "–"

x_1	x_2	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	1	0	0	0	1	1	1	0	0	0	0	1	1	1	1
0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	1	1	1
1	0	0	0	0	1	0	0	1	0	1	0	1	1	1	0	1	1
1	1	0	0	0	0	1	0	0	1	0	1	1	1	1	1	0	1
		FALSE	NOR			AND	NOT x_1	NOT x_2	NXOR	XOR	x_2	x_1	OR			NAND	TRUE
Linearly separable		+	+	+	+	+	+	+	–	–	+	+	+	+	+	+	+

f_2 : not x_1 and x_2

f_3 : x_1 and not x_2

f_{12} : x_1 or not x_2

f_{13} : not x_1 or x_2

a) 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.

Solution:

i) Boolean function OR (f_{11}):

x_1	x_2	OR
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

$w_1 = w_2 = 1.0,\ \theta = 0.5$

$$y = \Theta[x_1 w_1 + x_2 w_2 - \theta] = \Theta[x_1 + x_2 - 0.5]$$
$$x_1 = 0, x_2 = 0 \Rightarrow y = \Theta[0 - 0.5] = \Theta[-0.5] = 0$$
$$x_1 = 0, x_2 = 1 \Rightarrow y = \Theta[1 - 0.5] = \Theta[0.5] = 1$$
$$x_1 = 1, x_2 = 0 \Rightarrow y = \Theta[1 - 0.5] = \Theta[0.5] = 1$$
$$x_1 = 1, x_2 = 1 \Rightarrow y = \Theta[2 - 0.5] = \Theta[1.5] = 1$$

o.k.

ii) Boolean function f_3 :

x_1	x_2	f_3
0	**0**	**0**
0	**1**	**0**
1	**0**	**1**
1	**1**	**0**

$w_1 = 1.0,\ w_2 = -1.0,\ \theta = 0.5$

$$y = \Theta[x_1 w_1 + x_2 w_2 - \theta] = \Theta[x_1 - x_2 - 0.5]$$
$$x_1 = 0, x_2 = 0 \Rightarrow y = \Theta[0 - 0.5] = \Theta[-0.5] = 0$$
$$x_1 = 0, x_2 = 1 \Rightarrow y = \Theta[-1 - 0.5] = \Theta[-1.5] = 0$$
$$x_1 = 1, x_2 = 0 \Rightarrow y = \Theta[1 - 0.5] = \Theta[0.5] = 1$$
$$x_1 = 1, x_2 = 1 \Rightarrow y = \Theta[0 - 0.5] = \Theta[-0.5] = 0$$

o.k.

iii) Boolean function f_5 :

x_1	x_2	f_3
0	**0**	**1**

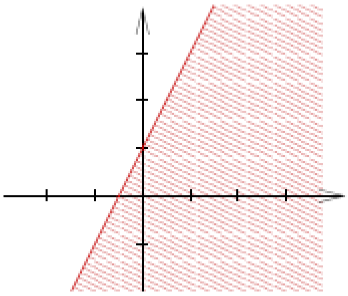
0	**1**	**1**
1	**0**	**0**
1	**1**	**0**

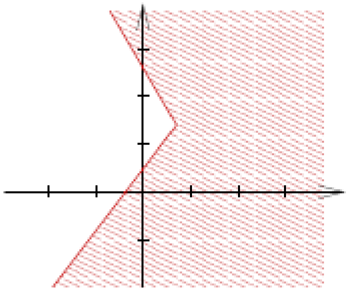
$w_1 = -1.0,\ w_2 = 0.0,\ \theta = -0.5$

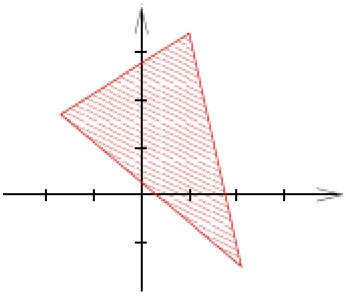
$$y = \Theta[x_1 w_1 + x_2 w_2 - \theta] = \Theta[-x_1 + 0.5]$$
$$x_1 = 0, x_2 = 0 \Rightarrow y = \Theta[0 + 0.5] = \Theta[0.5] = 1$$
$$x_1 = 0, x_2 = 1 \Rightarrow y = \Theta[0 + 0.5] = \Theta[0.5] = 1$$
$$x_1 = 1, x_2 = 0 \Rightarrow y = \Theta[-1 + 0.5] = \Theta[-0.5] = 0$$
$$x_1 = 1, x_2 = 1 \Rightarrow y = \Theta[-1 + 0.5] = \Theta[-0.5] = 0$$

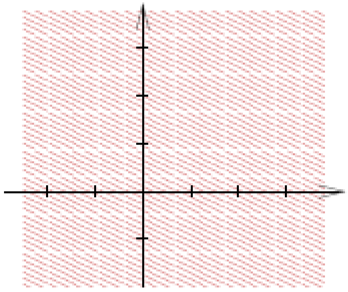
o.k.

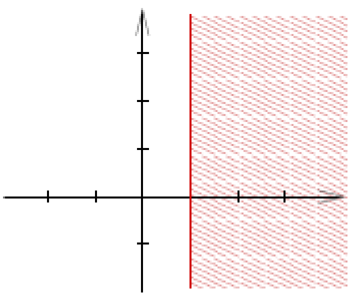
d) Which of the following partitioning of $\Re^2$ can be realized by a single-layer perceptron with two inputs? For those that can be realized, give weights and threshold of the perceptron. (Consider abscissa as x_1 and ordinate as x_2).

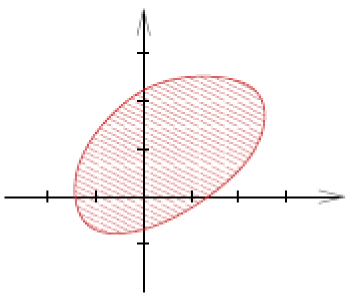

i)


ii)


iii)


iv)


v)


vi)

(From: Riedmiller)

Solution:

i) Boundary straight line $\rightarrow$ partitioning realizable by single-layer perceptron
Weights: Boundary defined by equation $x_2 = 2x_1 + 1.0 \Leftrightarrow 2x_1 - x_2 + 1.0 = 0$
Comparison with perceptron equation

$$y = \Theta[x_1 w_1 + x_2 w_2 - \theta]$$

$\Rightarrow w_1 = 2.0,\ w_2 = -1.0,\ \theta = -1.0$

ii) Boundary not straight line $\rightarrow$ partitioning NOT realizable by single-layer perceptron

iii) Boundary not straight line $\rightarrow$ partitioning NOT realizable by single-layer perceptron

iv) trivial boundary $\rightarrow$ partitioning realizable by single-layer perceptron
all inputs mapped to 1.0 which can be realized e.g. by choosing
$w_1 = 0.0,\ w_2 = 0.0,\ \theta = 1.0$

v) Boundary straight line $\rightarrow$ partitioning realizable by single-layer perceptron
Weights: Boundary defined by equation $x_1 = 1.0 \Leftrightarrow x_1 - 1.0 = 0$
Comparison with perceptron equation

$$y = \Theta[x_1 w_1 + x_2 w_2 - \theta]$$

$\Rightarrow w_1 = 1.0,\ w_2 = 0.0,\ \theta = 1.0$

vi) Boundary not straight line $\rightarrow$ partitioning NOT realizable by single-layer perceptron

(From: Riedmiller)

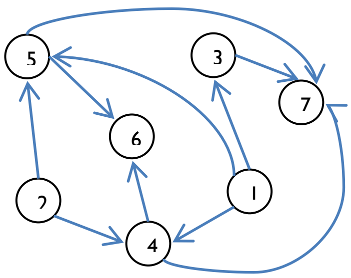
Exercise 2 (Types of neural networks, synaptic weight matrix):

a) Explain the following terms related to neural networks:

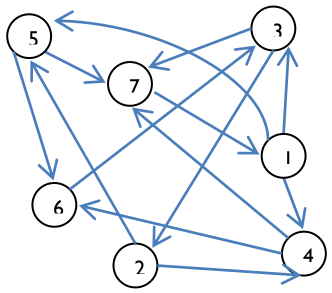
- Boolean function
  - Boolean function with n inputs: function of the form $f: \{0,1\}^n \rightarrow \{0,1\}$
- Feedforward neural network
  - Artificial neural network where
    - the neurons are organized in various layers
    - connections are only from lower to higher layers („feedforward“), i.e. there are no loops / no cycles, no feedback (even no lateral feedback!)
    - mathematically: directed acyclic graph
- Recurrent neural network
  - Artificial neural network with bidirectional data flow, i.e. including feedback loops
- Multi-layer perceptron
  - Special type of feedforward neural network where the neurons are perceptrons

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

i)

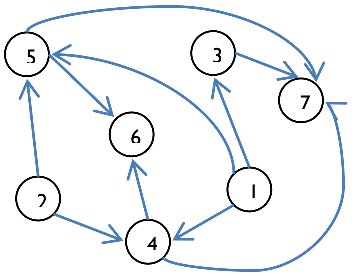


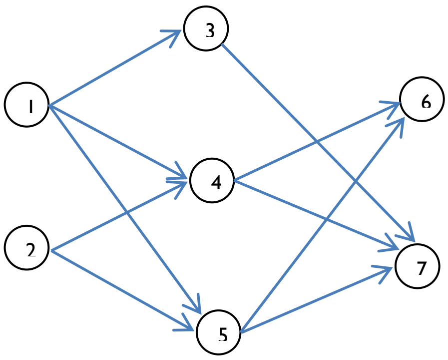
ii)



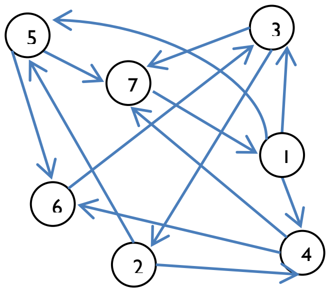
Solution:

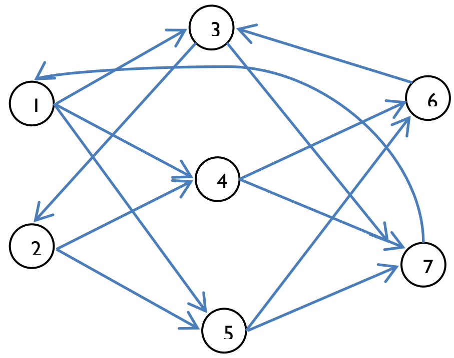
i) Feedforward neural network, because there are no feedback loops and it can be organized into separate layers



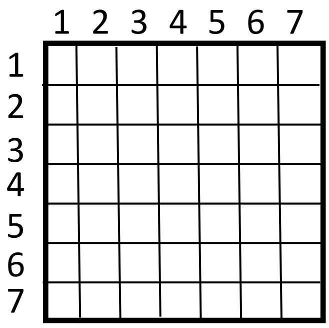


ii) Recurrent neural network, because there are feedback loops, e.g. $1 \rightarrow 3 \rightarrow 7 \rightarrow 1$ and $2 \rightarrow 4 \rightarrow 6 \rightarrow 3 \rightarrow 2$





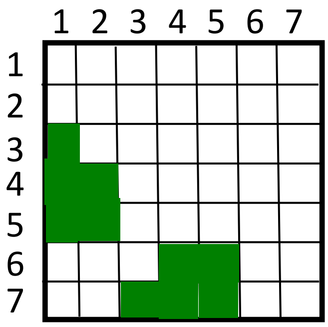
c) Using the neuron numbers from 1 to 7 given in the circles, fill out the following general weight matrix by marking the corresponding field entries. Example: Mark the field in row *i* and column *j* (weight w_{ij}) if there is a connection from neuron *j* to neuron *i*.



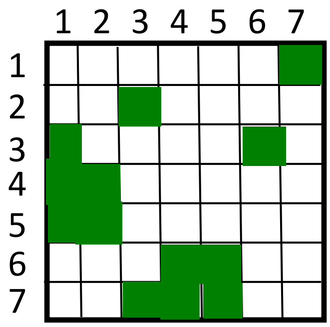
Solution:

(for the numbering of neurons see solution of part b)

i)

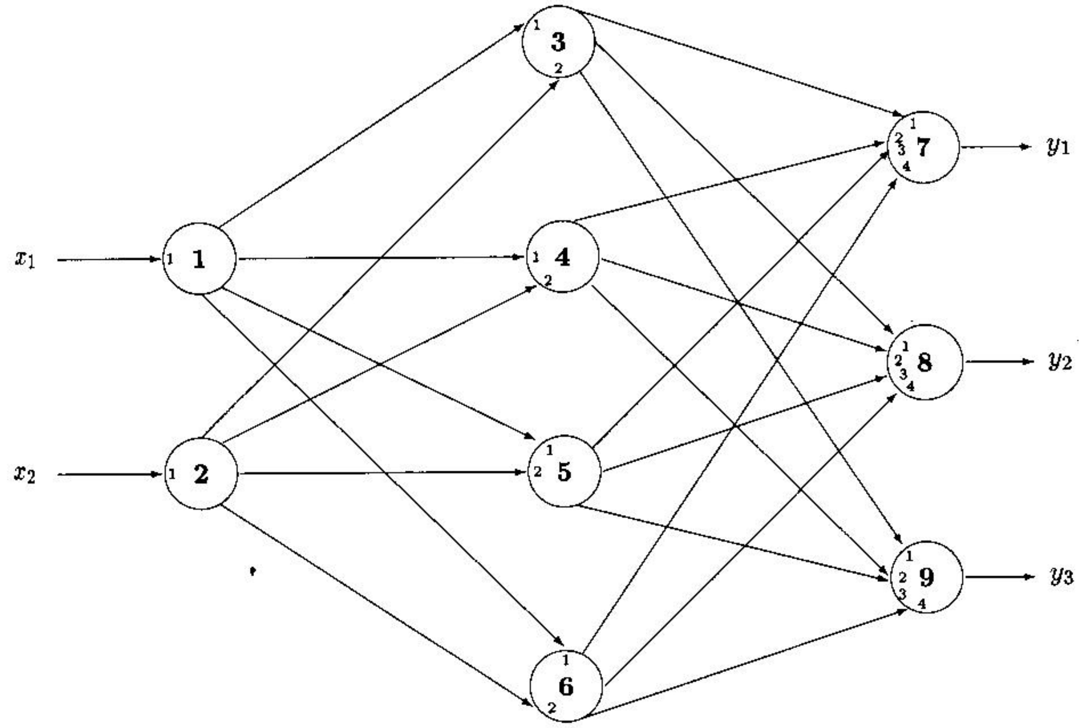


ii)



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?



Input values: x_1=3; x_2=1

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.

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)

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

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)

Solution:

Input values: x_1=3; x_2=1

Since the inputs are denoted by x_i and the outputs by y_i, we denote the states of the other neurons 1 – 9 by s_i.
In general, the output y_i of a neuron i from input x_j and synaptic weight w_{ij} connecting neuron state x_j to the postsynaptic neuron i is given by

$$y_i = f(h_i) = f\left(\sum_{j=1}^{m} w_{ij} \cdot x_j\right)$$

First step: Inputs to neurons 1 and 2
Neurons 1 and 2 can be computed in parallel, since their inputs are completely specified. Furthermore, these two neurons have a linear activation function with slope c = 1 so that $f(h)=h$, and each neuron receives only one input. Therefore, the states of neurons 1 and 2 are given by

$$w_{1in} = 1 \quad w_{2in} = 1$$
$$x_1 = 3 \qquad x_2 = 1$$
$$s_1 = f(h_1) = h_1 = w_{1in} \cdot x_1 = 1 \cdot 3 = 3$$
$$s_2 = f(h_2) = h_2 = w_{2in} \cdot x_2 = 1 \cdot 1 = 1$$

The states of all other neurons can only be computed afterwards.

Second step: Neurons 3, 4, 5 and 6

Neurons 3,4,5 and 6 and be computed in parallel, since – after calculation of neurons 1 and 2 – their inputs are completely specified. Furthermore, neurons 3, 4, 5 and 6 are threshold elements with threshold θ = 0; and each neuron gets input from the two neurons 1 and 2. Thus, their output s_i for i = 3, 4, 5, 6 can be written as

$$i = 3,4,5,6 \Rightarrow s_i = f(h_i) = \Theta[h_i - \theta] = \Theta[h_i] = \Theta\left[\sum_{j=1}^{2} w_{ij} \cdot s_j\right]$$

The synaptic weights and inputs have the following values:

$$w_{31} = 1 \quad w_{32} = -2$$
$$w_{41} = -1 \quad w_{42} = 0$$
$$w_{51} = 3 \quad w_{52} = 2$$
$$w_{61} = 0 \quad w_{62} = 2$$
$$s_1 = 3 \quad s_2 = 1$$

Thus, the states of neurons 3, 4, 5 and 6 can be computed as

$$s_3 = f(h_3) = \Theta\left[\sum_{j=1}^{2} w_{3j} \cdot s_j\right] = \Theta\left[w_{31} \cdot s_1 + w_{32} \cdot s_2\right] = \Theta[3-2] = \Theta[1] = 1$$

$$s_4 = f(h_4) = \Theta\left[\sum_{j=1}^{2} w_{4j} \cdot s_j\right] = \Theta\left[w_{41} \cdot s_1 + w_{42} \cdot s_2\right] = \Theta[-3+0] = \Theta[-3] = 0$$

$$s_5 = f(h_5) = \Theta\left[\sum_{j=1}^{2} w_{5j} \cdot s_j\right] = \Theta\left[w_{51} \cdot s_1 + w_{52} \cdot s_2\right] = \Theta[9+2] = \Theta[11] = 1$$

$$s_6 = f(h_6) = \Theta\left[\sum_{j=1}^{2} w_{6j} \cdot s_j\right] = \Theta\left[w_{61} \cdot s_1 + w_{62} \cdot s_2\right] = \Theta[0+2] = \Theta[2] = 1$$

The states of neurons 7, 8 and 9 can only be computed afterwards.

Third step: Neurons 7, 8 and 9

Neurons 7, 8 and 9 and be computed in parallel, since – after calculation of neurons 3, 4, 5 and 6 – their inputs are completely specified. Furthermore, neurons 7, 8 and 9 have a linear activation function with slope c = 1 so that again $f(h)=h$; each neuron 7, 8 and 9 gets input from the four neurons 3, 4, 5 and 6. Thus, their output can be written as

$$i = 7{,}8{,}9 \Rightarrow s_i = f(h_i) = h_i = \sum_{j=3}^{6} w_{ij} \cdot s_j$$

The synaptic weights and inputs have the following values:

$$\begin{array}{llll} w_{73} = 0 & w_{74} = 2 & w_{75} = -3 & w_{76} = 1 \\ w_{83} = 1 & w_{84} = -2 & w_{85} = 3 & w_{86} = 8 \\ w_{93} = 0 & w_{94} = 2 & w_{95} = 3 & w_{96} = -4 \\ s_3 = 1 & s_4 = 0 & s_5 = 1 & s_6 = 1 \end{array}$$

Thus, the states of neurons 7, 8 and 9 can be computed as

$$s_7 = f(h_7) = \sum_{j=3}^{6} w_{7j} \cdot s_j = w_{73} \cdot s_3 + w_{74} \cdot s_4 + w_{75} \cdot s_5 + w_{76} \cdot s_6 = 0+0-3+1 = -2$$

$$s_8 = f(h_8) = \sum_{j=3}^{6} w_{8j} \cdot s_j = w_{83} \cdot s_3 + w_{84} \cdot s_4 + w_{85} \cdot s_5 + w_{86} \cdot s_6 = 1+0+3+8 = 12$$

$$s_9 = f(h_9) = \sum_{j=3}^{6} w_{9j} \cdot s_j = w_{93} \cdot s_3 + w_{94} \cdot s_4 + w_{95} \cdot s_5 + w_{96} \cdot s_6 = 0+0+3-4 = -1$$

Therefore, the output to the inputs x_1=3 and x_2=1 is given by

$y_1 = s_7 = -2$
$y_2 = s_8 = 12$
$y_3 = s_9 = -1$

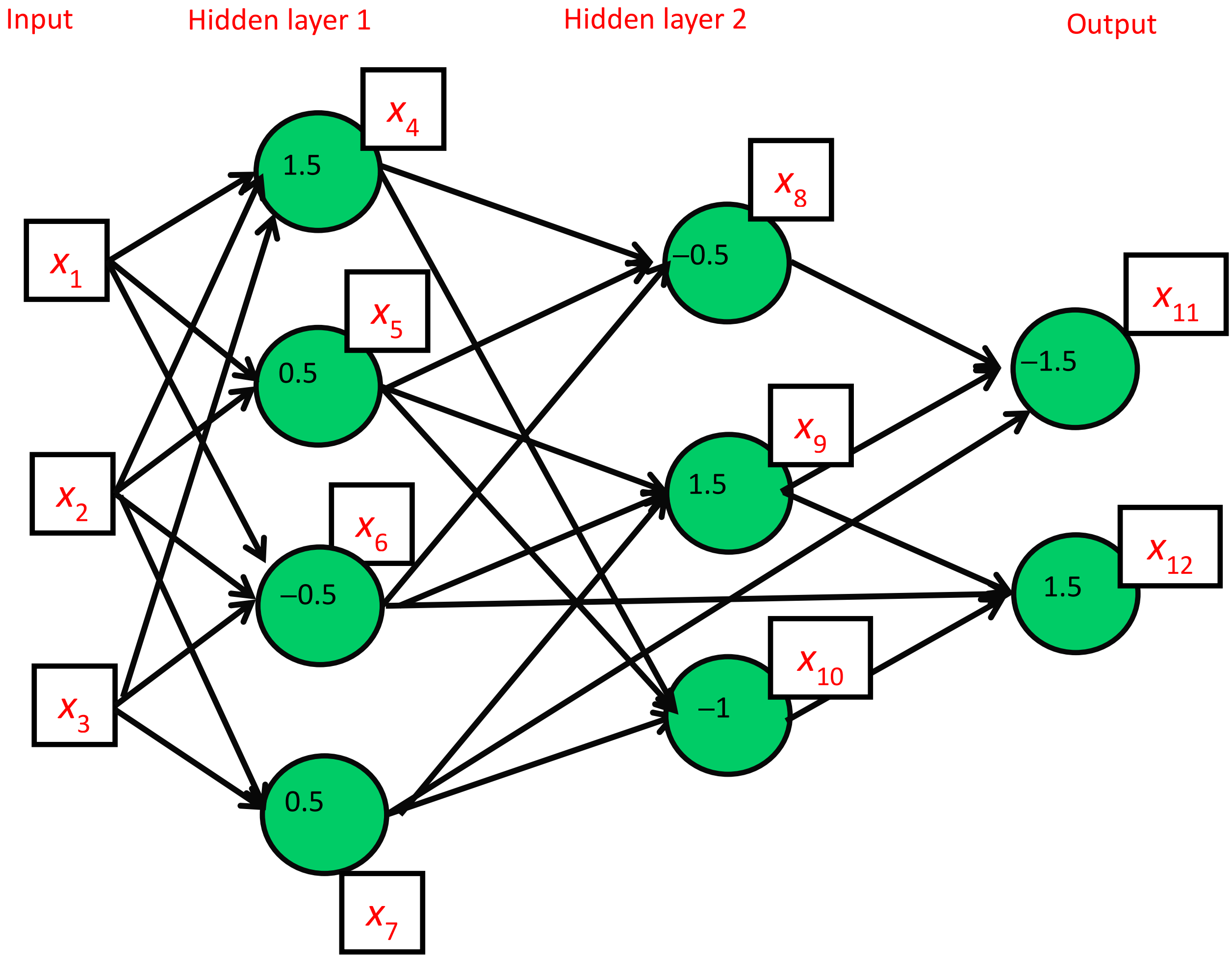
b) Assume the following weight matrix, where an entry w_{ij} (ith row, jth column) corresponds to the synaptic weight from neuron j to neuron i. (No entry means the synaptic weight is 0). Further assume that the activation function of the neurons of hidden layer 2 (neurons 8, 9 and 10) is linear (with 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:



Solution:

Consider that neurons 8, 9 and 10 have a linear activation, whereas the other neurons have a Heaviside activation function. From the weight matrix and the threshold values, we have the following equations:

$$x_4 = \Theta[-2x_1 + 5x_2 - 4x_3 - 1.5]$$
$$x_5 = \Theta[x_1 - 2x_2 - 0.5]$$
$$x_6 = \Theta[3x_1 - x_2 + 6x_3 + 0.5]$$
$$x_7 = \Theta[7x_2 + x_3 - 0.5]$$
$$x_8 = -x_4 + 4x_5 - 2x_6 + 0.5$$
$$x_9 = -3x_5 + 5x_6 + x_7 - 1.5$$
$$x_{10} = 8x_4 + 2x_5 - 3x_7 + 1$$
$$x_{11} = \Theta[6x_7 + x_8 - 2x_9 + 1.5]$$
$$x_{12} = \Theta[x_6 . - 4x_9 + 3x_{10} - 1.5]$$

- Consider the input x_1=1, x_2=0, x_3=1:

$$x_4 = \Theta[-2x_1 + 5x_2 - 4x_3 - 1.5] = \Theta[-7.5] = 0$$
$$x_5 = \Theta[x_1 - 2x_2 - 0.5] = \Theta[0.5] = 1$$
$$x_6 = \Theta[3x_1 - x_2 + 6x_3 + 0.5] = \Theta[9.5] = 1$$
$$x_7 = \Theta[7x_2 + x_3 - 0.5] = \Theta[0.5] = 1$$
$$x_8 = -x_4 + 4x_5 - 2x_6 + 0.5 = 2.5$$
$$x_9 = -3x_5 + 5x_6 + x_7 - 1.5 = 1.5$$
$$x_{10} = 8x_4 + 2x_5 - 3x_7 + 1 = 0$$
$$x_{11} = \Theta[6x_7 + x_8 - 2x_9 + 1.5] = \Theta[7] = 1$$
$$x_{12} = \Theta[x_6 - 4x_9 + 3x_{10} - 1.5] = \Theta[-6.5] = 0$$

$\rightarrow$ the output for input (x_1=1, x_2=0, x_3=1) is (x_{11}=1, x_{12}=0).

- Now consider the input x_1=0, x_2=1, x_3=1:

$$x_4 = \Theta[-2x_1 + 5x_2 - 4x_3 - 1.5] = \Theta[-0.5] = 0$$
$$x_5 = \Theta[x_1 - 2x_2 - 0.5] = \Theta[-2.5] = 0$$
$$x_6 = \Theta[3x_1 - x_2 + 6x_3 + 0.5] = \Theta[5.5] = 1$$
$$x_7 = \Theta[7x_2 + x_3 - 0.5] = \Theta[7.5] = 1$$
$$x_8 = -x_4 + 4x_5 - 2x_6 + 0.5 = -1.5$$
$$x_9 = -3x_5 + 5x_6 + x_7 - 1.5 = 4.5$$
$$x_{10} = 8x_4 + 2x_5 - 3x_7 + 1 = -2$$
$$x_{11} = \Theta[6x_7 + x_8 - 2x_9 + 1.5] = \Theta[-3] = 0$$
$$x_{12} = \Theta[x_6 - 4x_9 + 3x_{10} - 1.5] = \Theta[-24.5] = 0$$

$\rightarrow$ the output for input (x_1=0, x_2=1, x_3=1) is (x_{11}=0, 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.

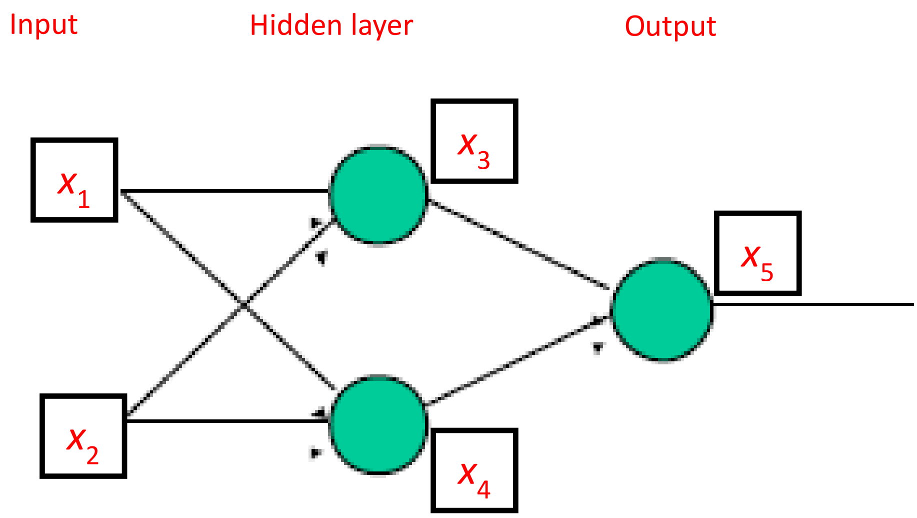
Solution:

The function XOR(x_1, x_2) can be separated into
OR(x_1, x_2) AND NOT_AND(x_1, x_2)

x_1	0	0	1	1
x_2	0	1	0	1
XOR	0	1	1	0
OR	0	1	1	1
AND	0	0	0	1
NOT_AND	1	1	1	0
OR AND NOT_AND	0	1	1	0

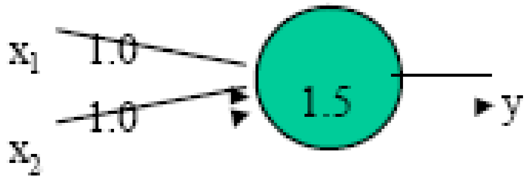
Therefore, we construct a perceptron realizing OR and another perceptron which realizes NOT_AND on the same inputs; the outputs of both perceptrons are then fed into another perceptron realizing AND.

Thus, the general architecture of the multi-layer perceptron can be described as

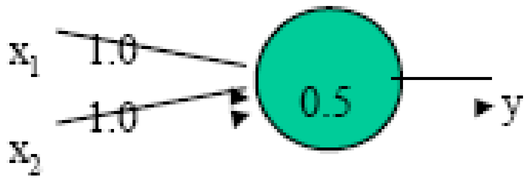


x_3 is supposed to realize NOT_AND on x_1 and x_2,
x_4 is supposed to realize OR on x_1 and x_2 and
x_5 is supposed to realize AND on x_3 and x_4
The Boolean functions AND and OR can be realized by (see lecture):

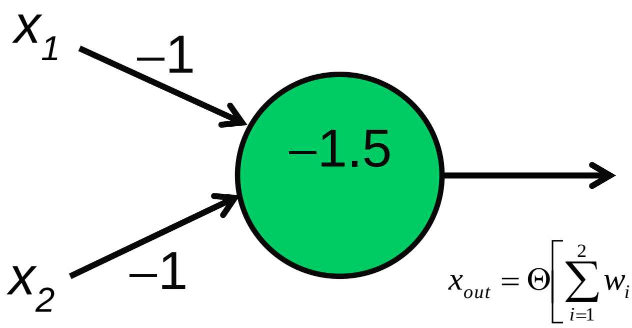
- **AND-Funktion:**



- **OR-Funktion:**



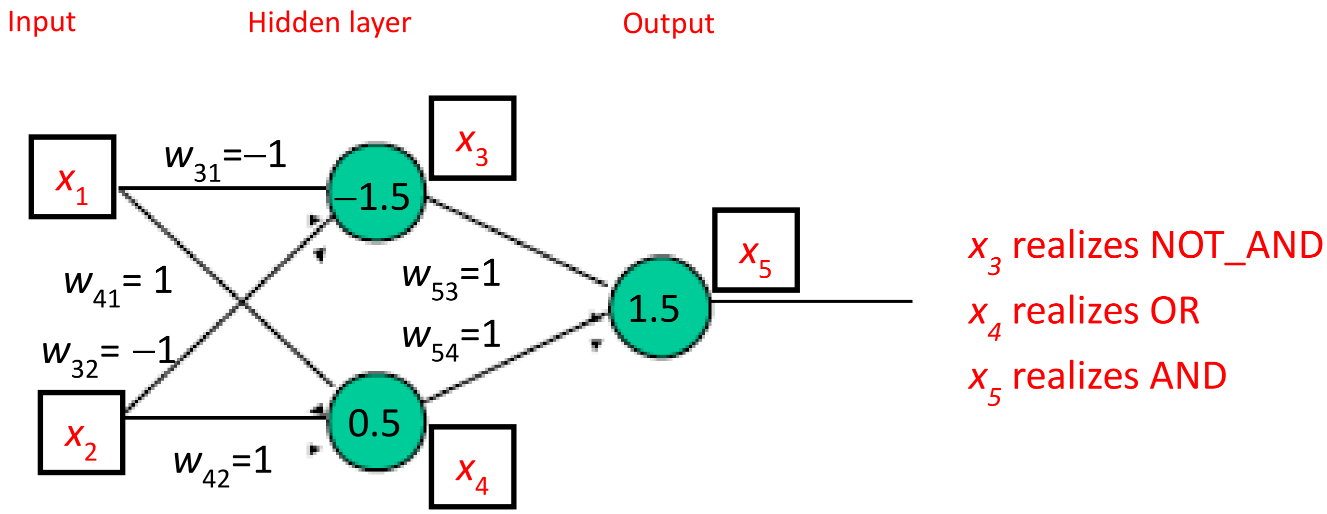
NOT_AND is given by multiplying the synaptic weights and the threshold by –1



$$x_{out} = \Theta\left[\sum_{i=1}^{2} w_i \cdot x_i - \theta\right] = \Theta[w_1 \cdot x_1 + w_2 \cdot x_2 - \theta] = \Theta[-x_1 - x_2 + 1.5]$$

x_1	0	0	1	1
x_2	0	1	0	1
NOT_AND	1	1	1	0

Thus, the weights and thresholds in the multi-layer perceptron can be immediately assigned:



Proof:

$$h_3 = w_{31}x_1 + w_{32}x_2 = -x_1 - x_2\ ;\quad x_3 = \Theta[h_3 + 1.5]$$

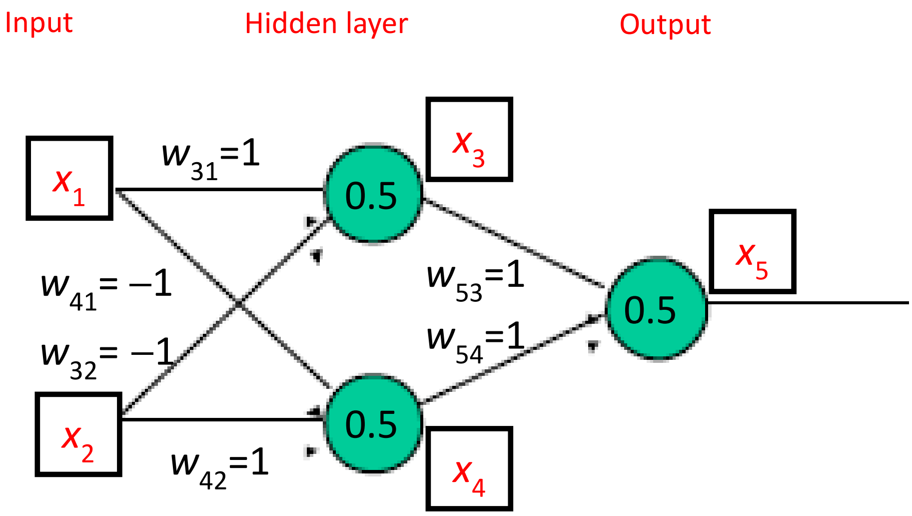
$$h_4 = w_{41}x_1 + w_{42}x_2 = x_1 + x_2\quad ;\quad x_4 = \Theta[h_4 - 0.5]$$

$$h_5 = w_{53}x_3 + w_{54}x_4 = x_3 + x_4\ ;\quad x_5 = \Theta[h_5 - 1.5]$$

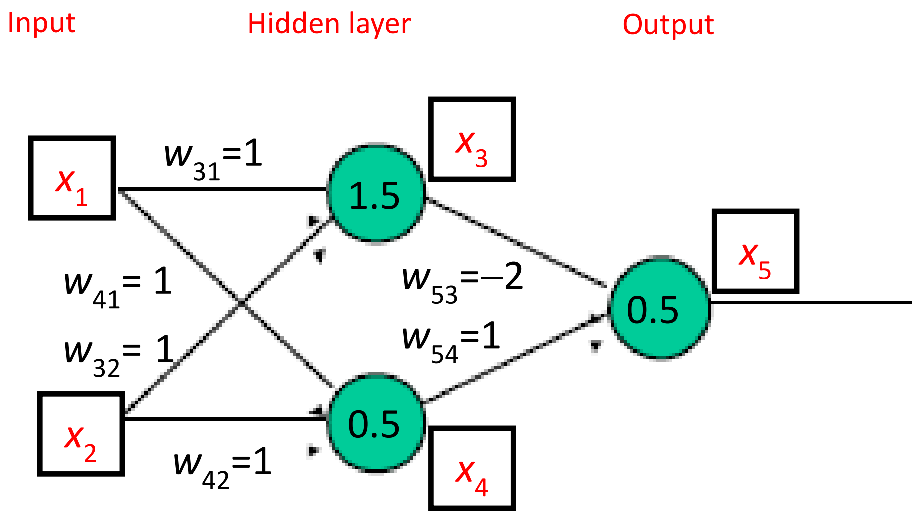
x_1	x_2	h_3	x_3	h_4	x_4	h_5	x_5	XOR
0	0	0	1	0	0	1	0	0
1	0	–1	1	1	1	2	1	1
0	1	–1	1	1	1	2	1	1
1	1	–2	0	2	1	1	0	0

Note: There are many other multi-layer perceptrons realizing XOR, e.g.

i)

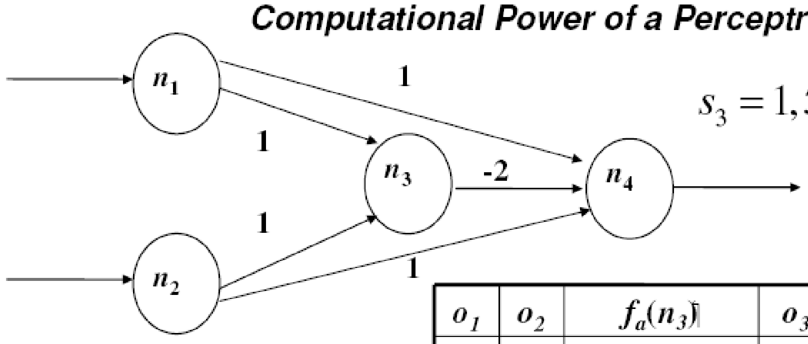


ii)



iii) (from: Lippe)

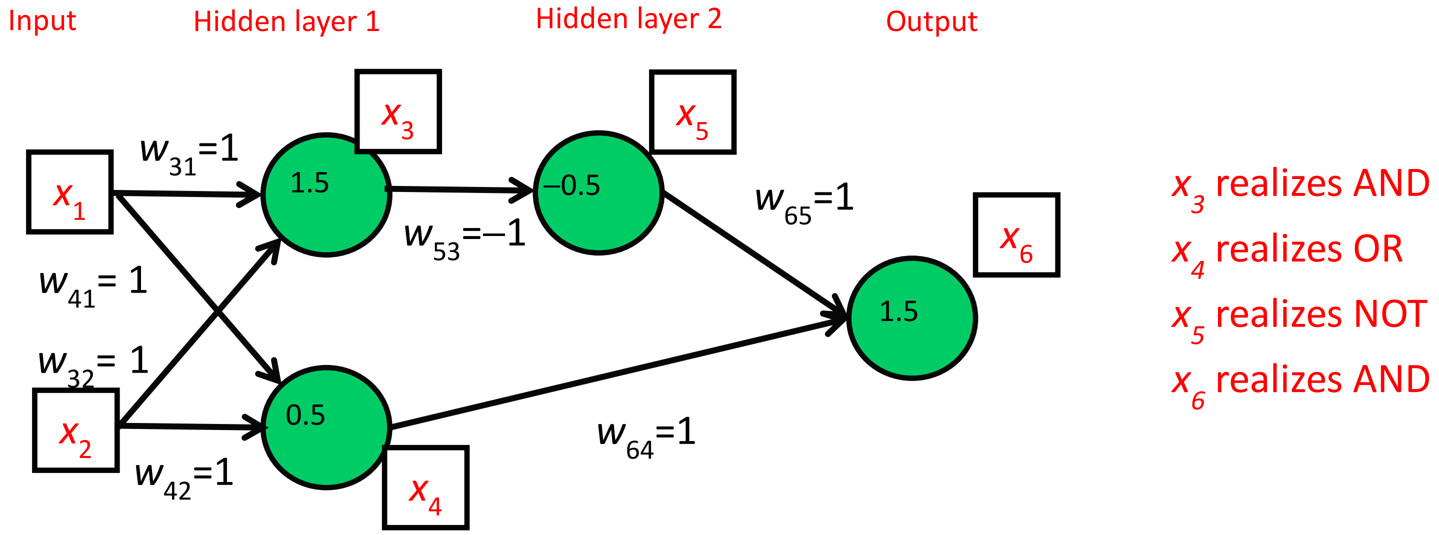
Computational Power of a Perceptron



$s_3 = 1,5$ and $s_4 = 0,5$

o_1	o_2	$f_a(n_3)$	o_3	$f_a(n_4)$	o_4
0	0	$0 \cdot 1 + 0 \cdot 1 = 0$	0	$0 \cdot 1 + 0 \cdot 1 + 0 \cdot (-2) = 0$	0
0	1	$0 \cdot 1 + 1 \cdot 1 = 1$	0	$0 \cdot 1 + 1 \cdot 1 + 0 \cdot (-2) = 1$	1
1	0	$1 \cdot 1 + 0 \cdot 1 = 1$	0	$1 \cdot 1 + 0 \cdot 1 + 0 \cdot (-2) = 1$	1
1	1	$1 \cdot 1 + 1 \cdot 1 = 2$	1	$1 \cdot 1 + 1 \cdot 1 + 1 \cdot (-2) = 0$	0

iv) (feedforward neural network of order 2)



x_3 realizes AND, x_4 realizes OR, x_5 realizes NOT, x_6 realizes AND

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

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

Note: “+” denotes mathematical addition.

Solution:

For the individual inputs x_1 and x_2 the values of the function F are given by

x_1	0	0	1	1
x_2	0	1	0	1
F	0	1	1	0

This is, however, the function XOR for which a solution has been given in part a.