WARSAW UNIVERSITY OF TECHNOLOGY
FACULTY OF MATHEMATICS
AND INFORMATION SCIENCE

Neural Networks

Lecture 13

Logic operations with neural networks

Logic networks

Most publications on neural networks focus on pattern recognition and associative memories. Here will be presented new area – logic operations. A multilayer system composed of simple identical elements **can perform any Boolean function of two, three or more variables.**

Logic networks

Long ago, M. Minsky and S. Pappert describing perceptron, or rather describing its faults used the **XOR** function as the example of operation cannot be performed by the one-layer perceptron.
This simple logical function can be realized on many ways

Logic networks

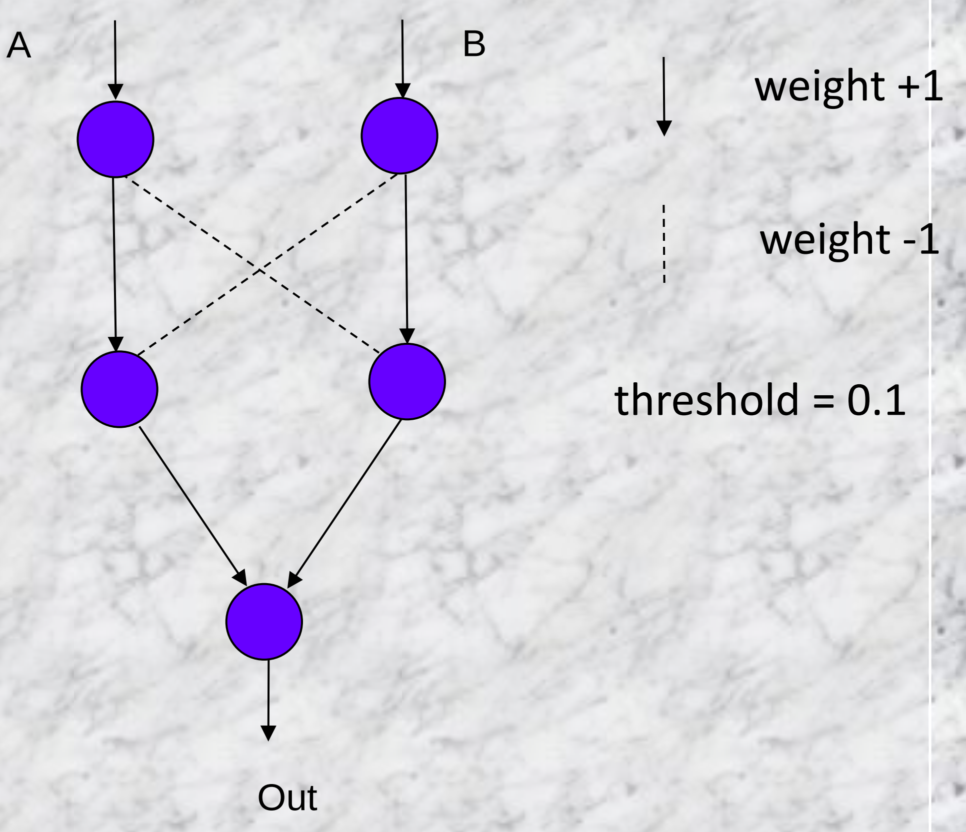
Example

The connections with an arrow have the positive weight equal +1, connections without arrows have the weights equal to -1. All elements are identical, with the nonlinear characteristics and threshold equal to 0.1. Input signals components are equal to one or zero.

Logic networks

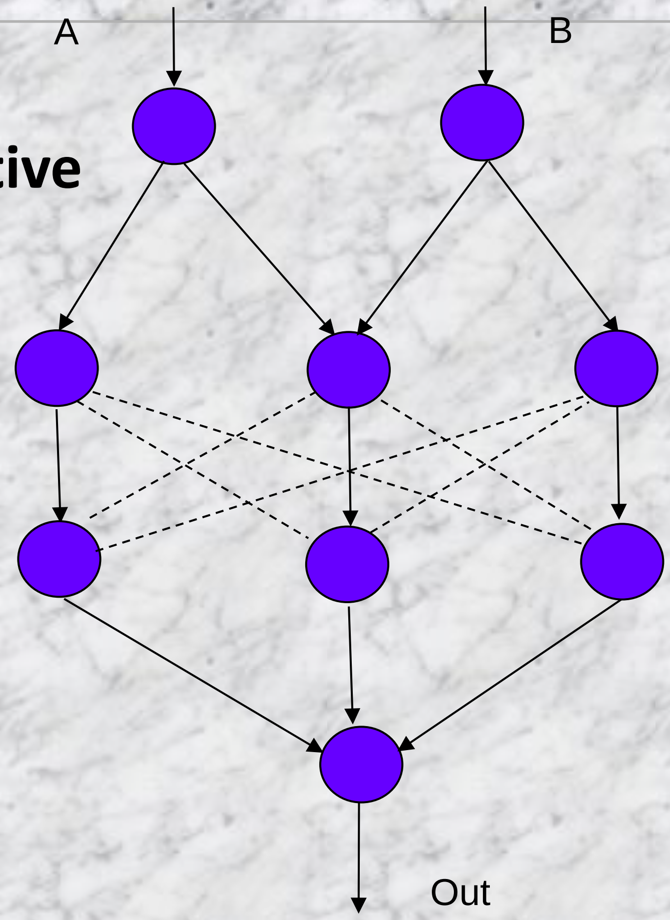
Example of the network able to perform XOR operation

A	B	Wy
0	0	0
1	0	1
0	1	1
1	1	0



Logic networks

Model of associative memory type



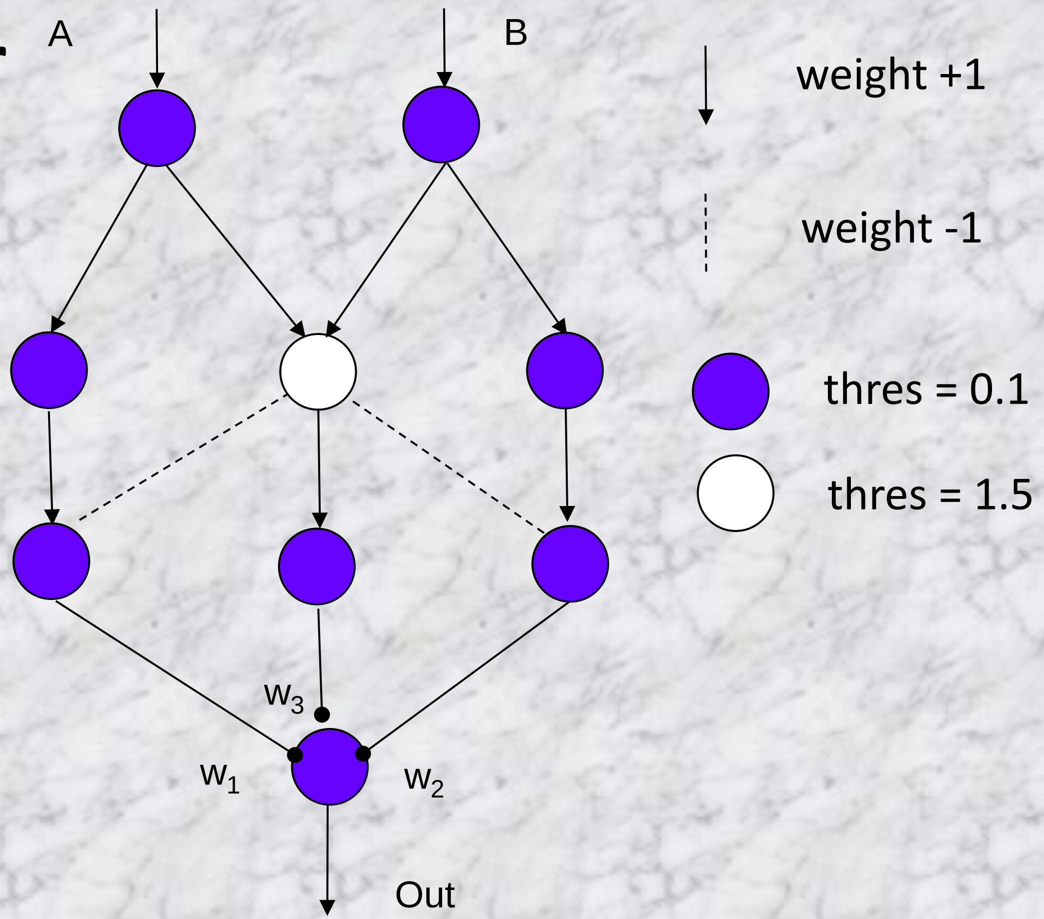
weight = +1

weight = -0.5

threshold = 0.1

Logic networks

Logic module for many functions



Logic networks

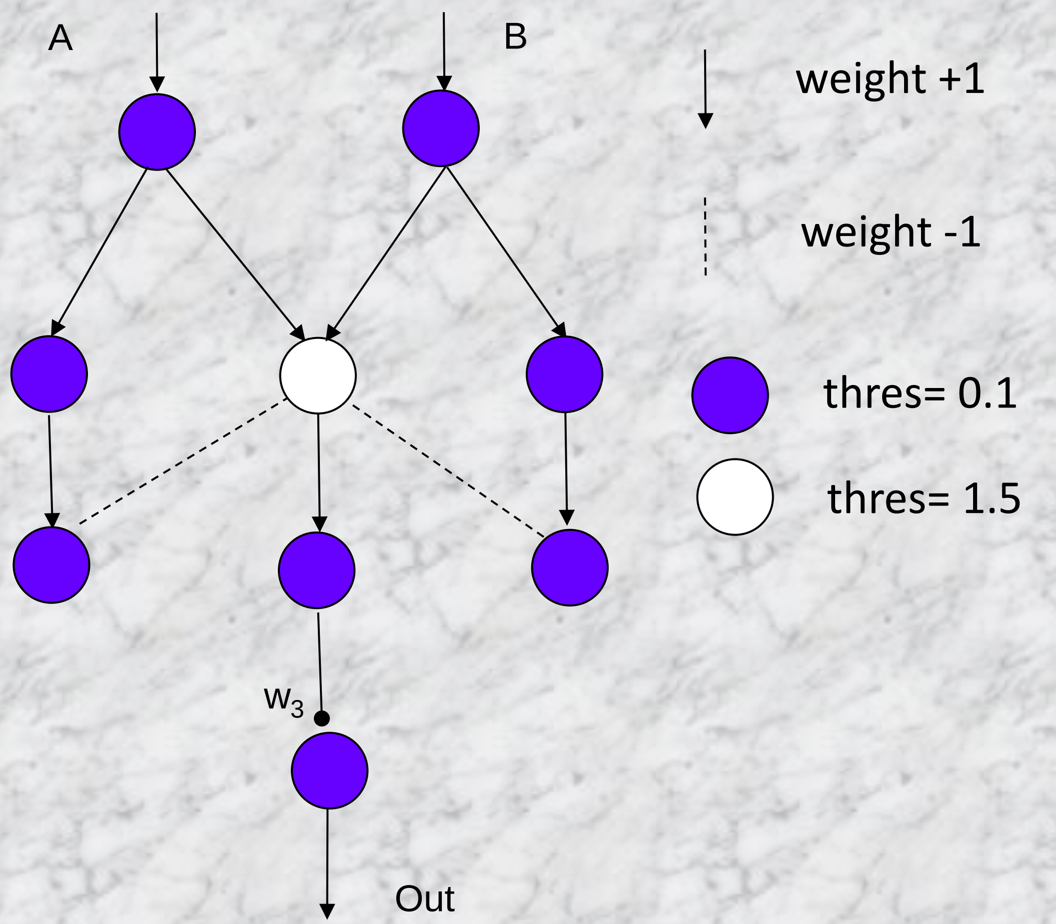
Examples of functions

function	w_1	w_2	w_3
A OR **B**	1	1	1
A AND **B**	0	0	1
A XOR **B**	1	1	0
A AND (NOT **B**)	1	0	0
A AND (**B** OR (NOT **B**))	1	0	1

Logic networks

Example

A and **B**



Logic networks

Logical operations

For *n* logical variables one can creates 2^{2^n} different functions.

number of variables n	number of functions of n variables
1	4
2	16
3	256
4	65 536

Logic networks

Any logical function can be written in ***a canonical form***.

The canonical form: An expression is said to be in a canonical *sum-of-product* form when variables are logically ANDed into groups (called minterms), that are logically ORed to form a function.

Every variable appears in every minterm once in the canonical sum-of-product form. All 2^n minterms of n variables can be generated in a network of n+1 levels, and the minterm can be combined into arbitrary function in an additional level..

Logic networks

Functions of two variables

The Canonical form

$$f = \overline{\mathrm{A}}\,\overline{\mathrm{B}}\,f_0 + \overline{\mathrm{A}}\mathrm{B}f_2 + \mathrm{A}\overline{\mathrm{B}}\,f_1 + \mathrm{AB}f_3$$

Logic networks

Table of 16 possible two-element logical operations

	Function	Descr	coefficients			
			f_0	f_1	f_2	f_3
1	$\overline{A}\,\overline{B}$	NOR	1	0	0	0
2	$A\overline{B}$		0	1	0	0
3	$\overline{A}B$		0	0	1	0
4	AB	AND	0	0	0	1

Logic networks

Table of 16 possible two-element logical operations

	Function	Descr	coefficients			
			f_0	f_1	f_2	f_3
5	$\overline{A}\,\overline{B} + A\overline{B}$	~B	1	1	0	0
6	$\overline{A}\,\overline{B} + \overline{A}B$	~A	1	0	1	0
7	$A\overline{B} + AB$	A	0	1	0	1
8	$\overline{A}B + AB$	B	0	0	1	1

Logic networks

Table of 16 possible two-element logical operations

	Function	Descr	coefficients			
			f_0	f_1	f_2	f_3
9	$\bar{A}B + A\bar{B}$	XOR	0	1	1	0
10	$A\bar{B} + \bar{A}B + AB$	OR	0	1	1	1
11	$\bar{A}\bar{B} + AB$	~XOR	1	0	0	1
12	$\bar{A}\bar{B} + A\bar{B} + AB$	A+~B	1	1	0	1

Logic networks

Table of 16 possible two-element logical operations

	Function	Descr	coefficients			
			f_0	f_1	f_2	f_3
13	$\overline{A}\,\overline{B}+A\overline{B}+\overline{A}B$	NAND	1	1	1	0
14	$\overline{A}\,\overline{B}+\overline{A}B+AB$	~A+B	1	0	1	1
15	$\text{out always}=0$	FALSE	0	0	0	0
16	$\text{out always}=1$	TRUE	1	1	1	1

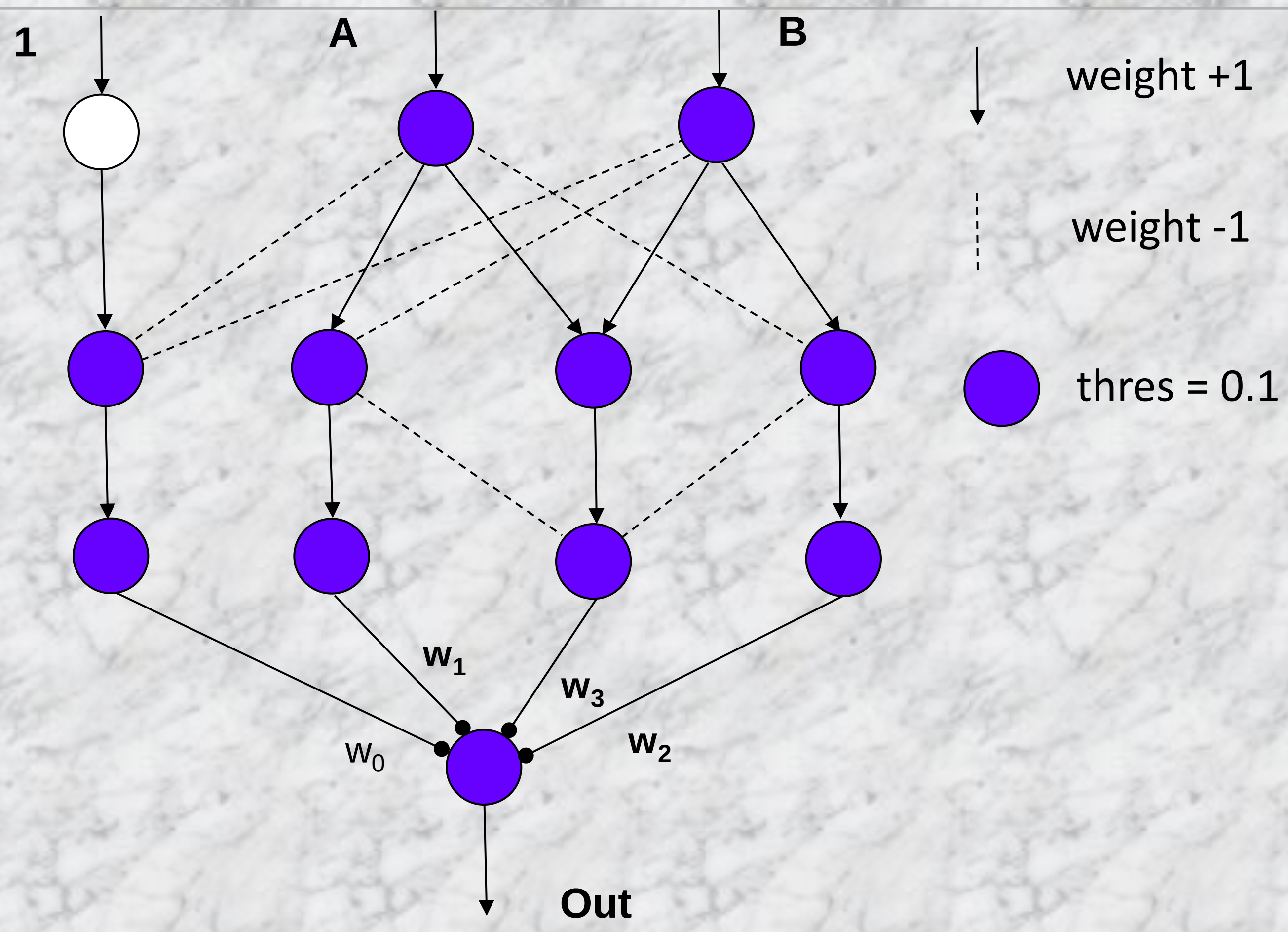
Logic networks

Any out of 16 two-element logic operations can be programmable by a universal logic module.

Model assumptions:

- Input signals ar equal to 1 or 0.
- Connections with arrow are equal to +1.
- Connections without arrows are equal to -1.
- The element shown white is always activated by the input signal equal to +1.

Logic networks – Universal logic module



Logic networks

Description of network operation

The network input signal IN = [1,A,B,]

Input signal to the elements of the 1st intermediate layer $X = IN * W^1$

W^1 matrix of connections between input elements and elements of the 1st intermediate layers

$$W^1 = \begin{bmatrix} +1 & 0 & 0 & 0 \\ -1 & +1 & +1 & -1 \\ -1 & -1 & +1 & +1 \end{bmatrix}$$

Nonlinear threshold function Φ $\hat{X} = \Phi(X) = \begin{cases} 1 & \text{for} \quad x_i > 0 \\ 0 & \text{for} \quad x_i \le 0 \end{cases}$

Logic networks

Description of network operation

Input signal to the element of the 2nd intermediate layer $Y = \hat{X} * W^2$

W^2 matrix of connections between elements of the 1st and 2nd intermediate layers

$$W^2 = \begin{bmatrix} +1 & 0 & 0 & 0 \\ 0 & +1 & -1 & 0 \\ 0 & 0 & +1 & 0 \\ 0 & 0 & -1 & +1 \end{bmatrix}$$

Nonlinear threshold function Φ

$$\hat{Y} = \Phi(Y)$$

Logic networks

Description of network operation

Network output signal OUT = $\Phi(\hat{Y} * W^3)$

W^3 matrix of connections between the elements of the 2^{nd} intermediate layer and the output element

$$W^3 = \begin{bmatrix} w_0 & w_1 & w_3 & w_2 \end{bmatrix}$$

Finally, for the network

$$OUT = \Phi\left\{\Phi\left[\Phi(IN * W^1) * W^2\right] * W^3\right\} =$$

$$= \Phi\{\Phi(1 - A - B)w_0 + \Phi(A - B)w_1 + \Phi(B - A)w_2 +$$

$$\Phi[\Phi(A + B) - \Phi(A - B) - \Phi(B - A)]w_3\}$$

Logic networks

Example

The well-known operation **OR (A+B)** using logical theorems (expansion, distributive, commutative, De Morgan's etc)**,** can be rewritten into a canonical form

$$\mathbf{A} + \mathbf{B} = \mathbf{A}(\mathbf{B} + \overline{\mathbf{B}}) + \mathbf{B}(\mathbf{A} + \overline{\mathbf{A}}) =$$

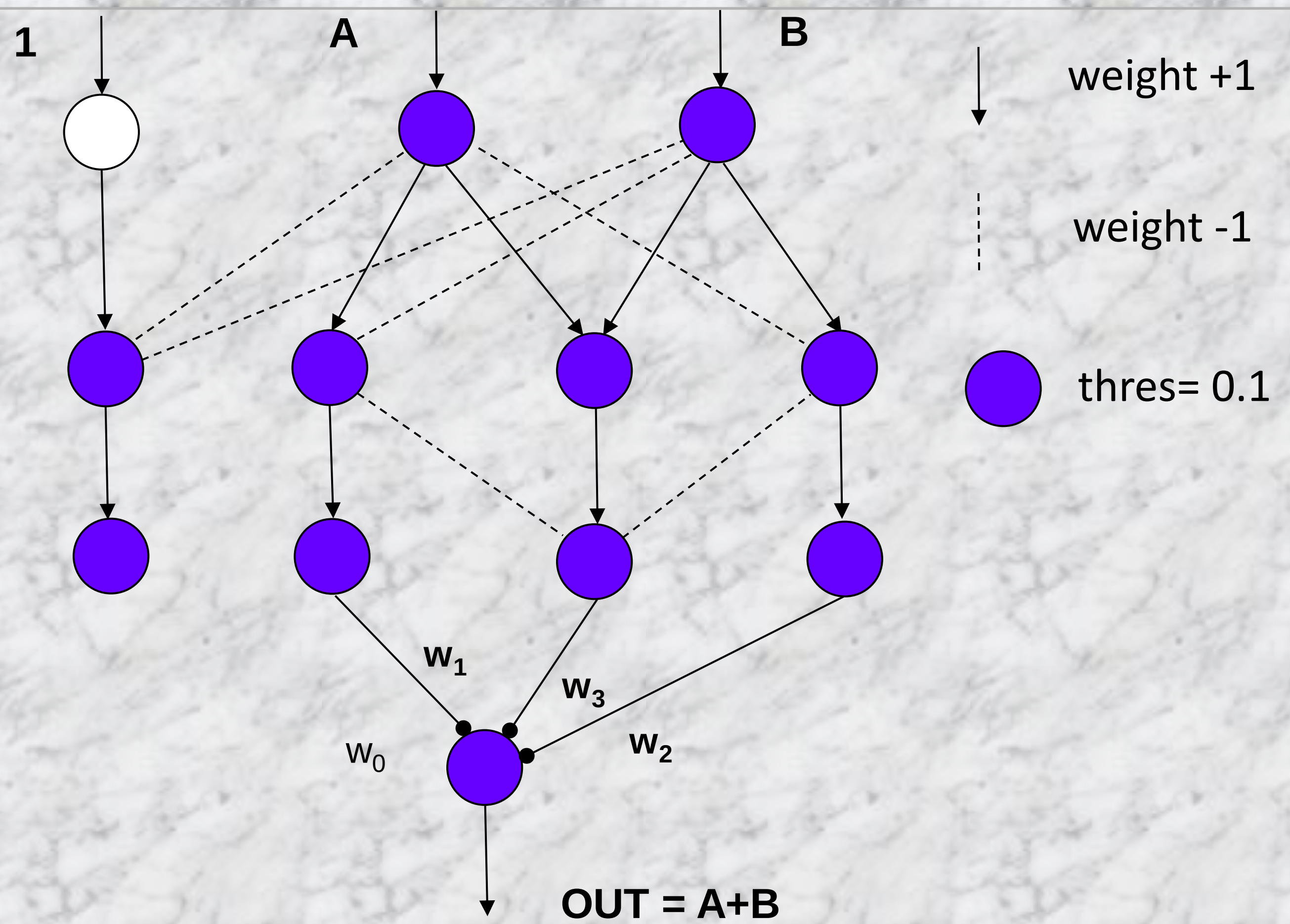
$$= \mathbf{AB} + \mathbf{A}\overline{\mathbf{B}} + \mathbf{BA} + \mathbf{B}\overline{\mathbf{A}} =$$

$$= \mathbf{AB} + \mathbf{A}\overline{\mathbf{B}} + \overline{\mathbf{A}}\mathbf{B}$$

The universal logic module can perform this operation by setting of weights

$w_0 = 0 \quad w_1 = 1 \quad w_2 = 1 \quad w_3 = 1$

Logic networks – Universal logic module



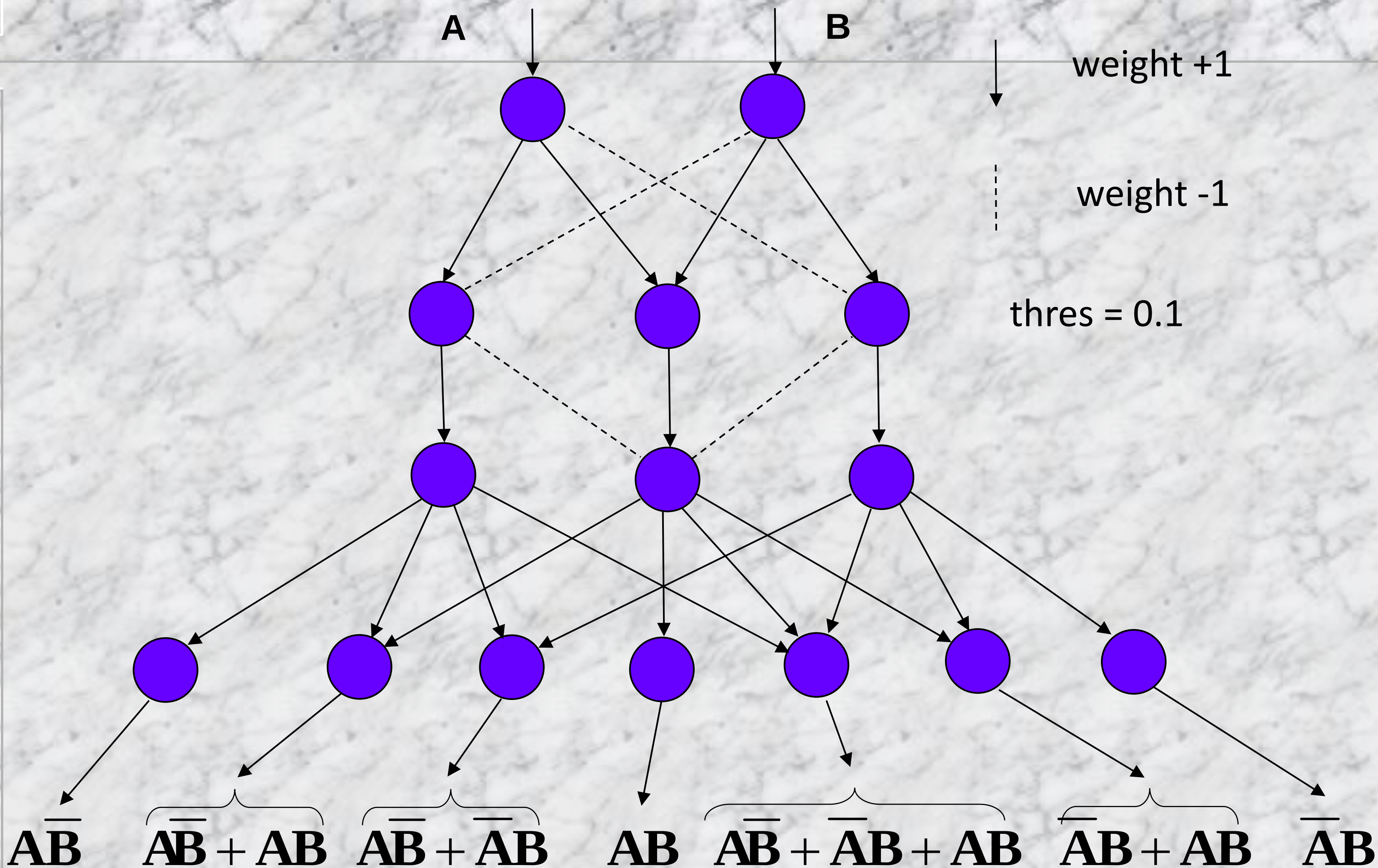
Logic networks

The other solutions

By replacing a single output element by a layer of 2^{2^n} elements (for n=2 by 16 elements), and by fixing the interconnections to the output layer we get the network where each output element corresponds to one logical function. Each element of the second intermediate slab reacts to only one term of the canonical form of a logical function.

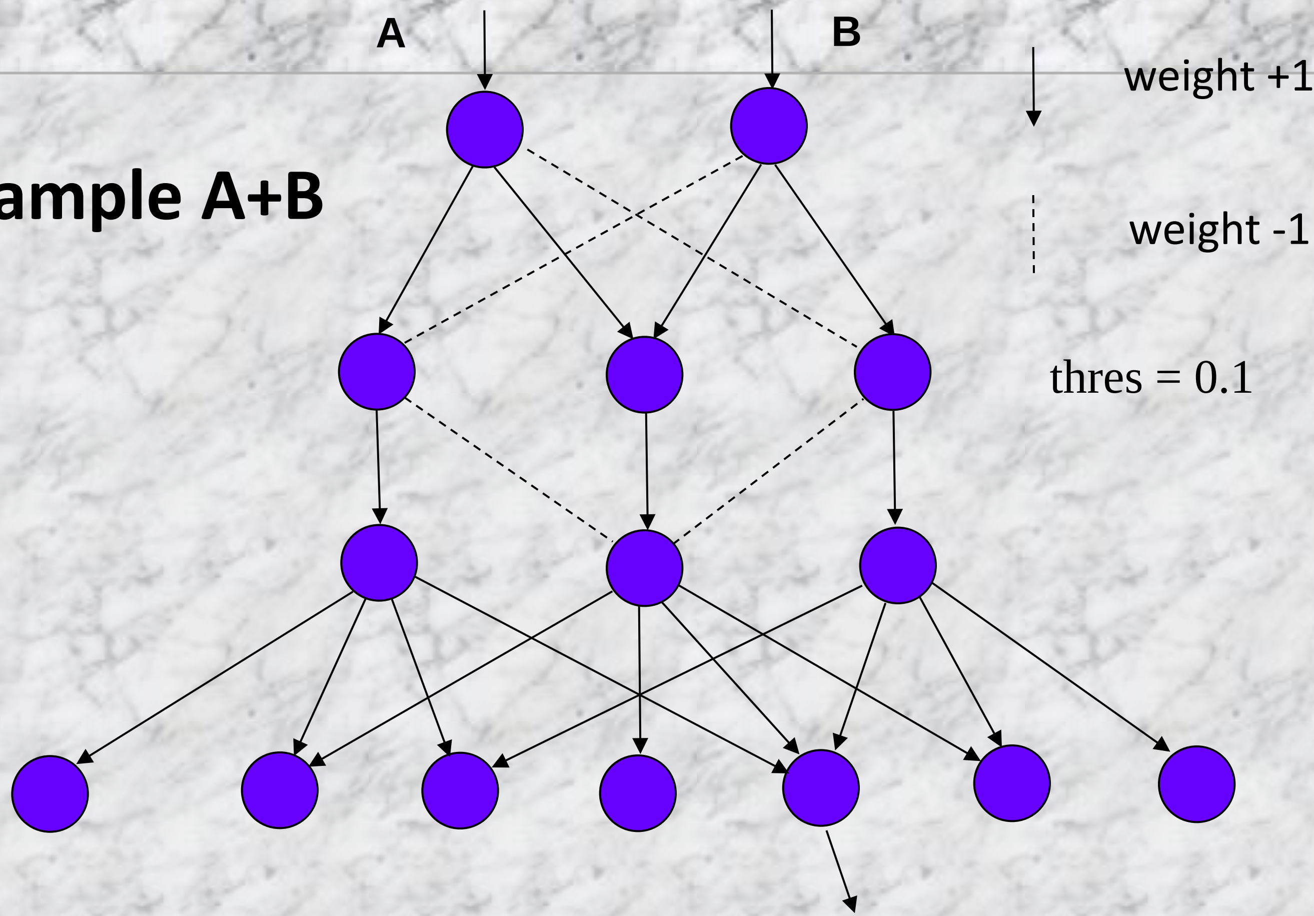
A simplified version of the network which can perform seven of all 16 two element logical operations, neglecting only those for which a total zero input lead to a non-zero output is shown.

Logic networks – Universal logic module



Logic networks – Universal logic module

Example A+B

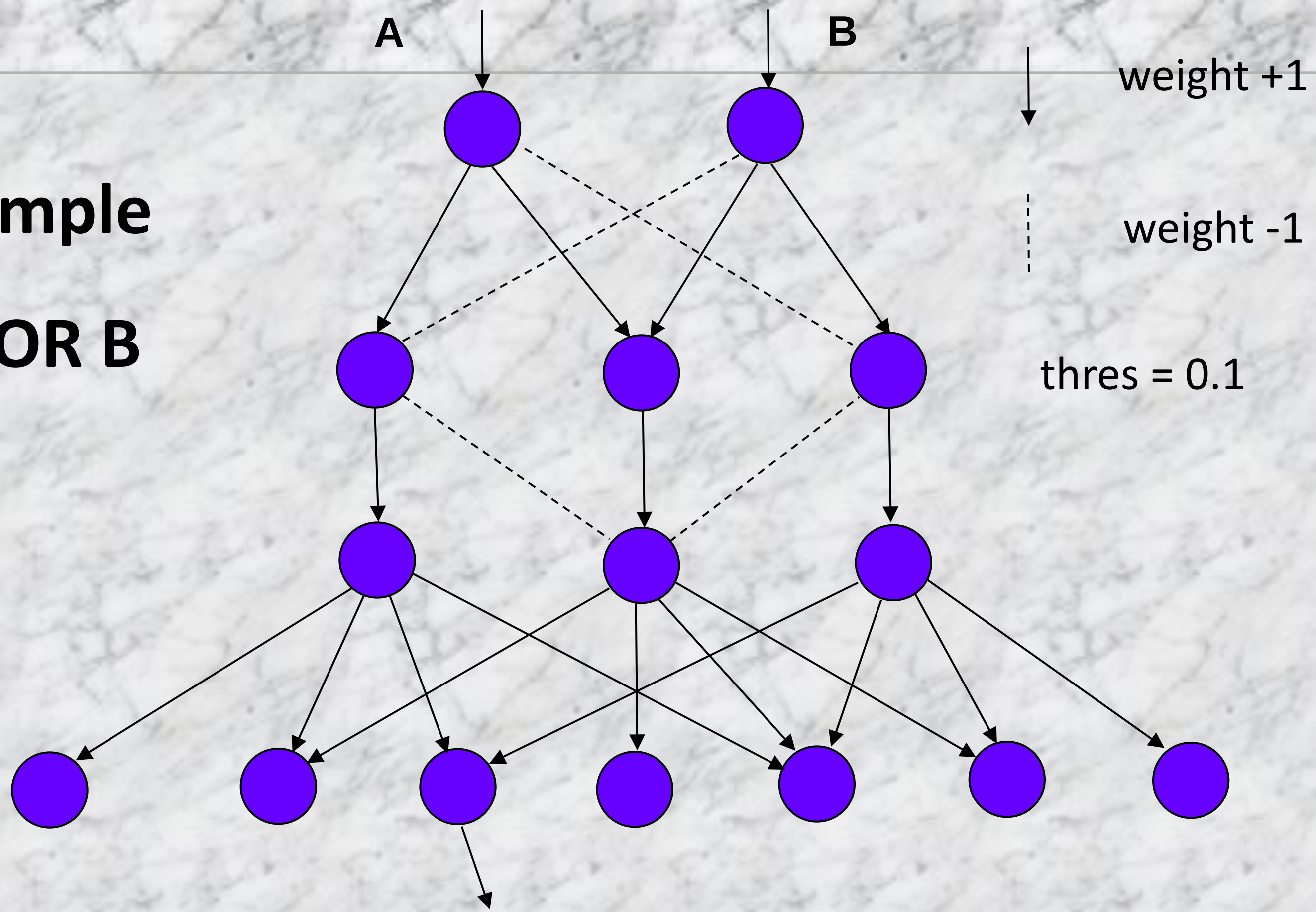


$$A\overline{B} + \overline{A}B + AB$$

Logic networks – Universal logic module

Example

A XOR B



$$\mathbf{A\overline{B} + \overline{A}B}$$