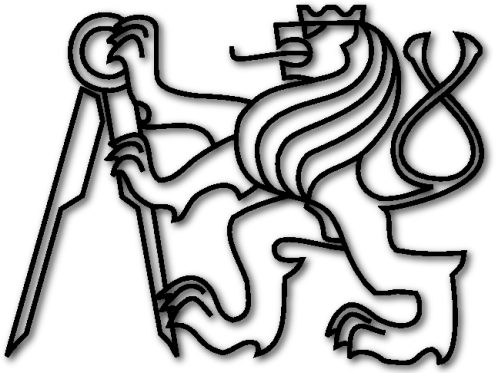
Artificial Neural Networks
The Introduction

Jan Drchal
drchajan@fel.cvut.cz



Computational Intelligence Group
Department of Computer Science and Engineering
Faculty of Electrical Engineering
Czech Technical University in Prague

Outline

- What are Artificial Neural Networks (ANNs)?
- Inspiration in Biology.
- Neuron models and learning algorithms:
  - MCP neuron,
  - Rosenblatt's perceptron,
  - MP-Perceptron,
  - ADALINE/MADALINE
- Linear separability.

Information Resources

- https://cw.felk.cvut.cz/doku.php/courses/a4m33bia/start
- M. Šnorek: **Neuronové sítě a neuropočítače.** Vydavatelství ČVUT, Praha, 2002.
- **Courseware**: http://neuron.felk.cvut.cz/courseware/
- R. Rojas: **Neural Networks - A Systematic Introduction**, Springer-Verlag, Berlin, New-York, 1996, http://www.inf.fu-berlin.de/inst/ag-ki/rojas_home/pmwiki/pmwiki.php?n=Books.NeuralNetworksBook
- J. Šíma, R. Neruda: **Theoretical Issues of Neural Networks**, MATFYZPRESS, Prague, 1996, http://www2.cs.cas.cz/~sima/kniha.html
- This presentation is based on materials for 36NAN and 336VD.

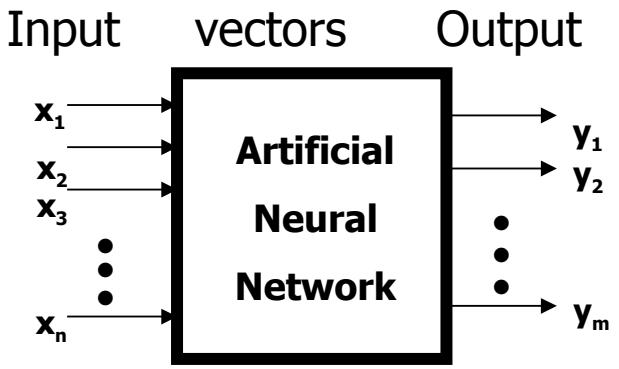
What Are Artificial Neural Networks (ANNs)?

- Artificial information systems which imitate functions of neural systems of living organisms.
- Non-algorithmic approach to computation → learning, generalization.
- Massively-parallel processing of data using large number of simple computational units (neurons).
- Note: we are still very far from the complexity found in the Nature! But there have been some advances lately – we will see ;)

ANN: the Black Box

- Function of an ANN can be understood as the transformation T of an input vector **X** to the output vector **Y**

$$\boldsymbol{Y} = T(\boldsymbol{X})$$



- What transformations T can be realized by neural networks ? It is the scientific question since the beginning of the discipline.

What Can Be Solved by ANNs?

- Function approximation - regression.
- Classification (pattern/sequence recognition).
- Time series prediction.
- Fitness prediction (for optimization).
- Control (i.e. robotics).
- Association.
- Filtering, clustering, compression.

ANN Learning & Recall

- ANNs work most frequently in two phases:
- **Learning phase** – adaptation of ANN's internal parameters.
- **Evaluation phase (recall)** – use what was learned.

Learning Paradigms

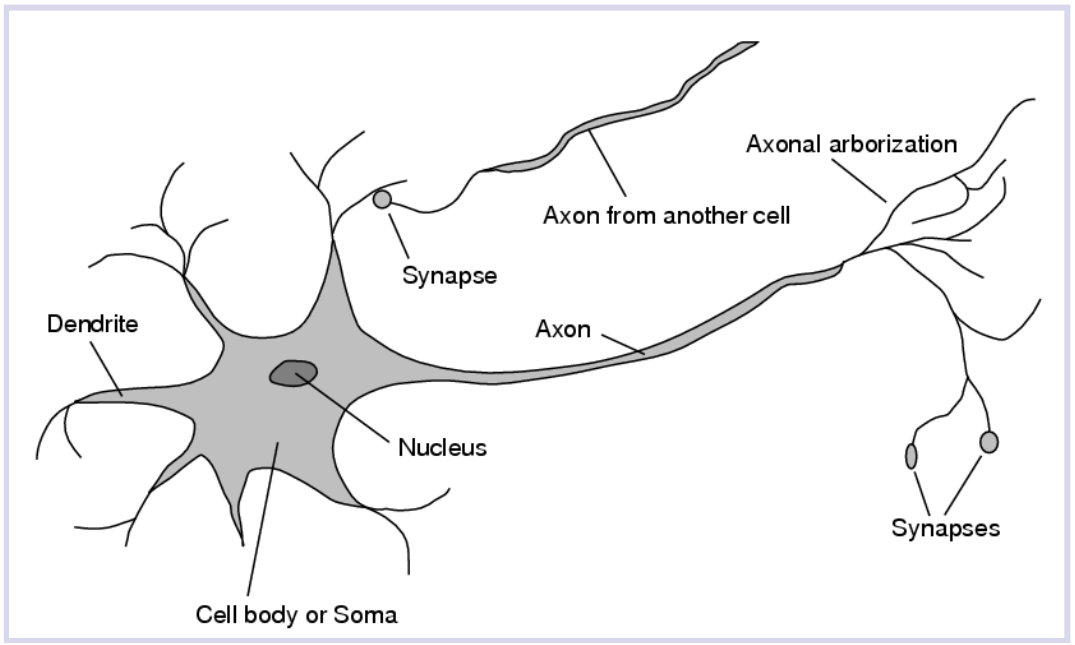
- **Supervised** – teaching by examples, given a set of example pairs $P_i=(x_i,y_i)$ find transformation **T** which approximates $y_i=\boldsymbol{T}(x_i)$ for all i.
- **Unsupervised** – self-organization, no teacher.
- **Reinforcement** – Teaching examples not availiable → they are generated by interactions with the environment (mostly control tasks).

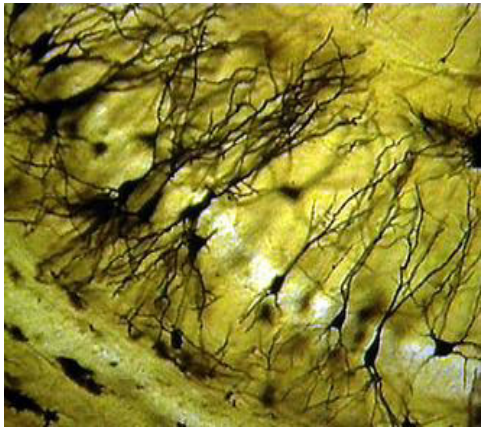
When to Stop Learning Process?

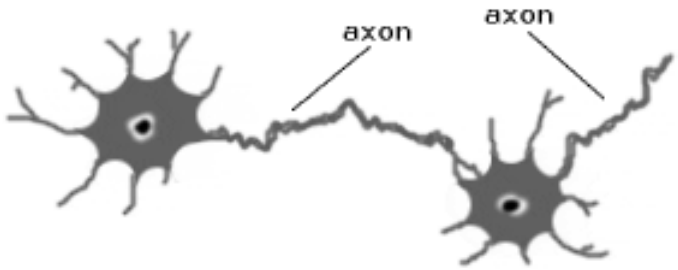
- When for all patterns of a **training set** a given criterion is met:
  - the ANN's error is less than...,
  - stagnation for given number of **epochs**.

- **Training set** – a subset of example pattern set dedicated to learning phase.
- **Epoch** – application of all training set patterns.

Inspiration: Biological Neurons







Biological Neurons II

- Neural system of a 1 year old child contains approx. $10^{11\text{-}12}$ neurons (loosing 200 000 a day).
- The diameter of soma nucleus ranges from 3 to 18 μm.
- Dendrite length is 2-3 mm, there is 10 to 100 000 of dendrites per neuron.
- Axon length – can be longer than 1 m.

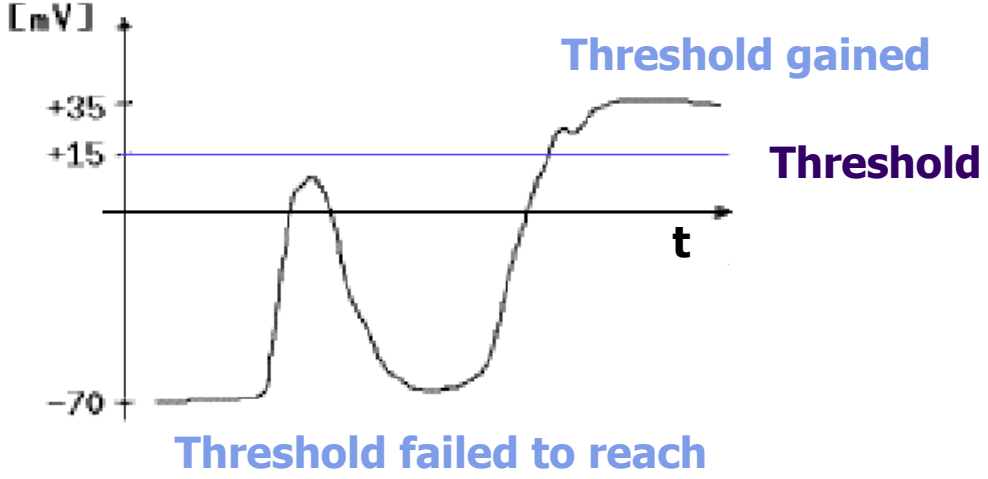
Synapses

- The synapses perform a transport of signals between neurons. The synapses are of two types:
  - excitatory,
  - inhibitory.
- Another point of view
  - chemical,
  - electrical,
  - others.

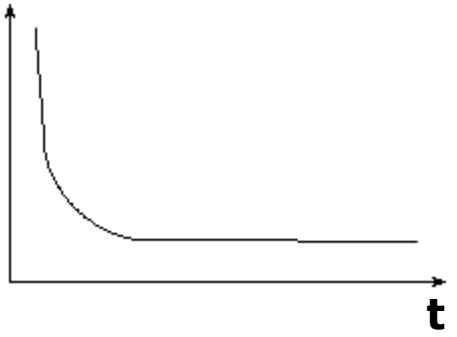
Neural Activity - a Reaction to Stimuli

- After neuron fires through axon, it is unable to fire again for about 10 ms.
- The speed of signal propagation ranges from 5 to 125 m/s.

Neuron's potential



Activity in time



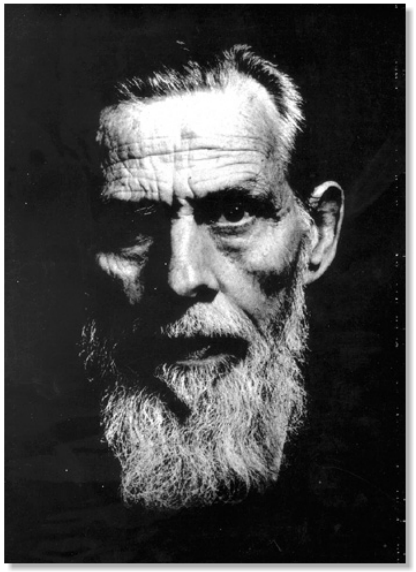
Hebbian Learning Postulate

- Donald Hebb 1949.
- *"When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased."*

  → ***"cells that fire together, wire together"***

Neuron Model: History

- Warren McCulloch, Walter Pitts

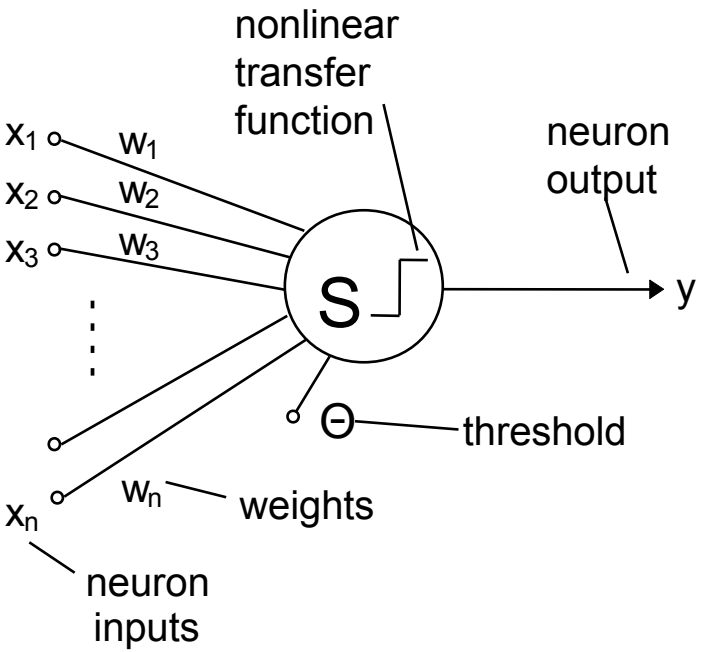


1899 - 1969

1923 - 1969

McCulloch, W. S. and Pitts, W. H. (1943).
A logical calculus of the ideas immanent in nervous activity.
Bulletin of Mathematical Biophysics, 5:115-133.

McCulloch-Pitts (MCP) Neuron

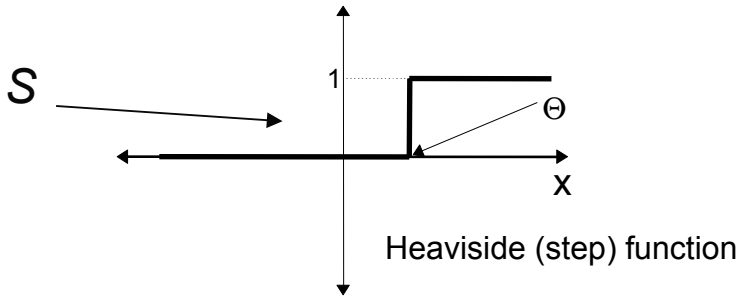


- y neuron's **output (activity)**
- x_i neuron's *i*-th **input** out of total *N*,
- w_i *i*-th synaptic **weight**,
- S (nonlinear) **transfer (activation) function**,
- Θ **threshold, bias** (performs shift).
- The expression in brackets is **the inner potential**.
- They worked with binary inputs only.
- No learning algorithm.

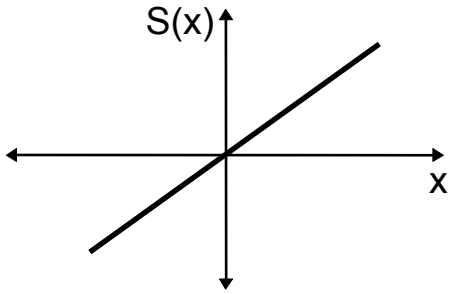
$$y = S\left(\sum_{i=1}^{N} w_i x_i + \Theta\right)$$

OR

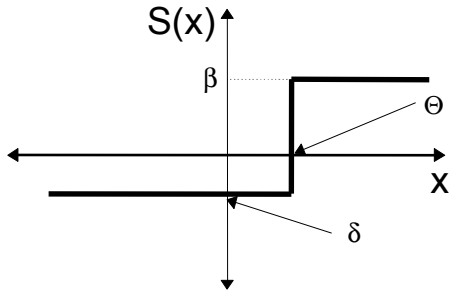
$$y = S\left(\sum_{i=0}^{N} w_i x_i\right)$$, where $w_0 = \Theta$ and $x_0 = 1$



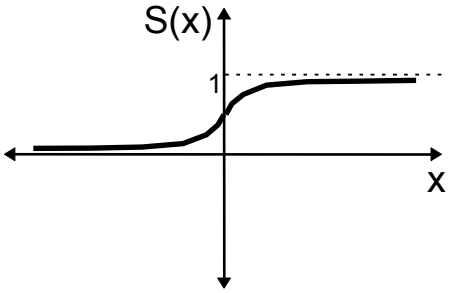
Other Transfer Functions



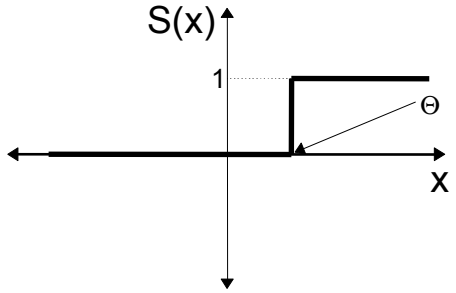
a) linear



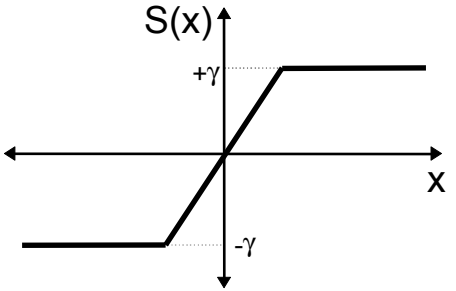
b) two-step function



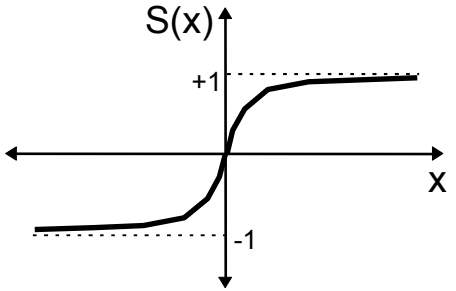
c) sigmoid



d) Heaviside (step) function



e) semilinear



f) hyperbolic tangent

MCP Neuron Significance

- The basic type of neuron.
- Most ANNs are based on perceptron type neurons.
- Note: there are other approaches (we will see later...)

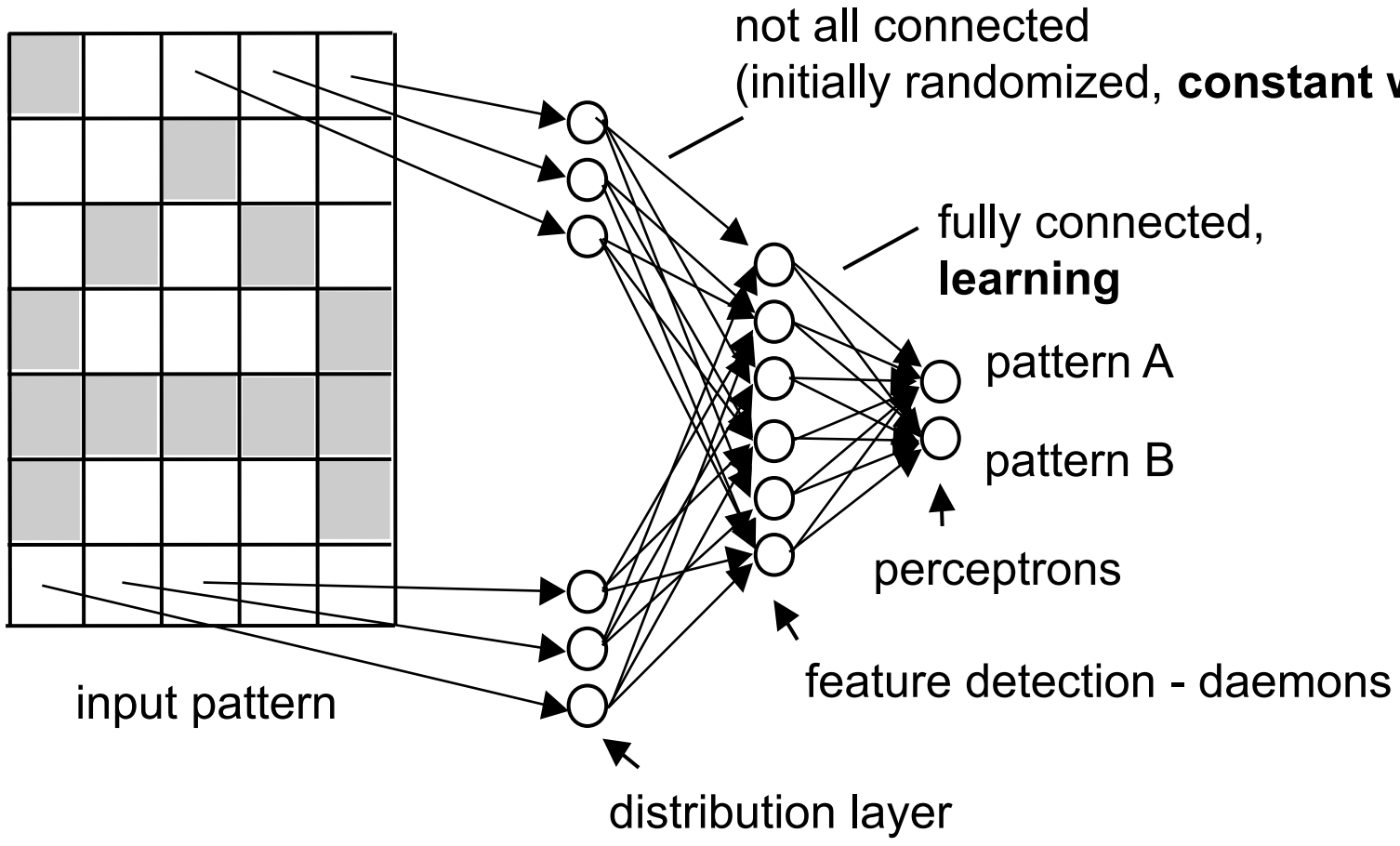
Rosenblatt's Perceptron (1957)

Frank Rosenblatt -

The creator of the first neurocomputer: MARK I (1960).

- Learning algorithm for MCP-neurons.
- Classification of letters.

Rosenblatt's Perceptron Network



Rosenblatt = McCulloch-Pitts + learning algorithm

Rosenblatt - Learning Algorithm

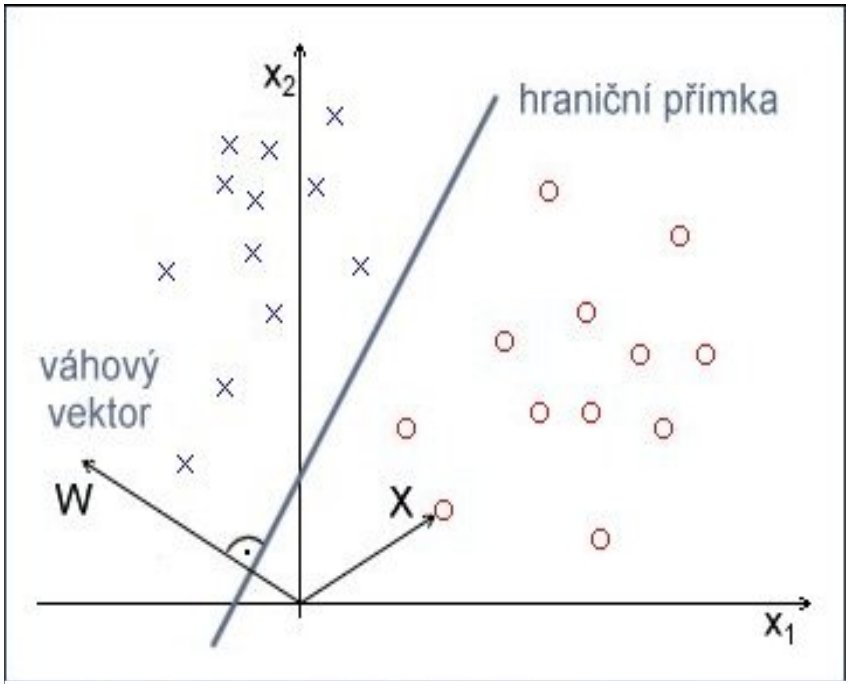
- Randomize weights.
- If the output is correct, no weight is changed.
- Desired output was 1, but it is 0 → increment the weights of the *active* inputs.
- Desired output was 0, but it is 1 → decrement the weights of the *active* inputs.
- Three possibilities for the amplitude of the weight change:
  - Increments/decrements: only fixed values.
  - I/D based on error size. It is advantageous to have them higher for higher error → may lead to instability.
  - Fixed or variable I/D combination based on error size

Geometric Interpretation of Perceptron

- Let's have a look at the inner potential (in 2D)

$$w_1 x_1 + w_2 x_2 + \Theta = 0$$

$$x_2 = -\frac{w_1}{w_2} x_1 - \Theta / w_2$$

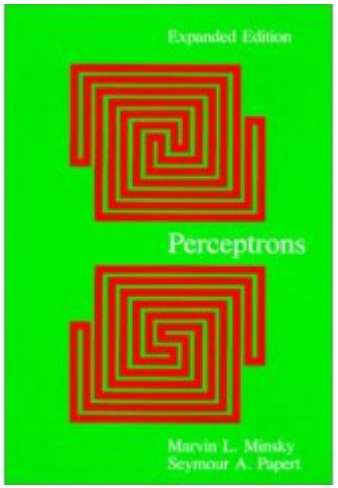


Line with direction $-w_1/w_2$ a and shift $-\Theta/w_2$.

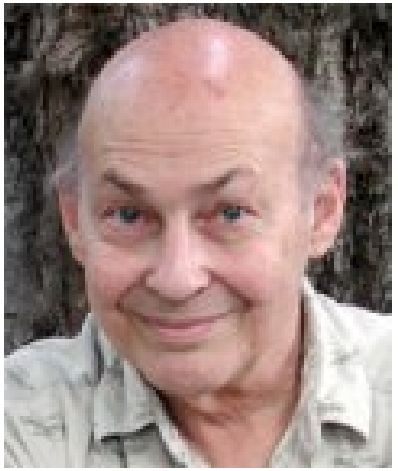
- Linear dividing (hyper)plane: decision boundary.
- What will the transfer function change?

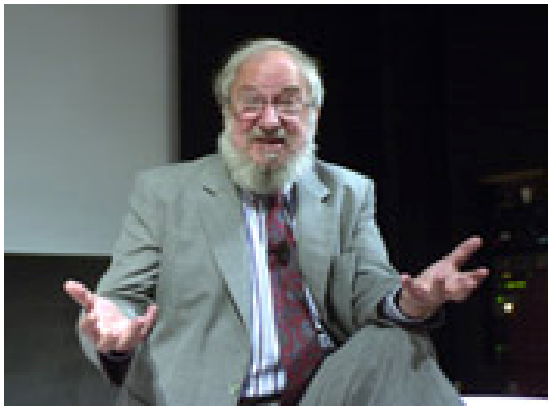
MP-Perceptron

- MP = Marvin Minsky a Seymour Papert,
  - MIT Research Laboratory of Electronics,
  - In 1969 they published:

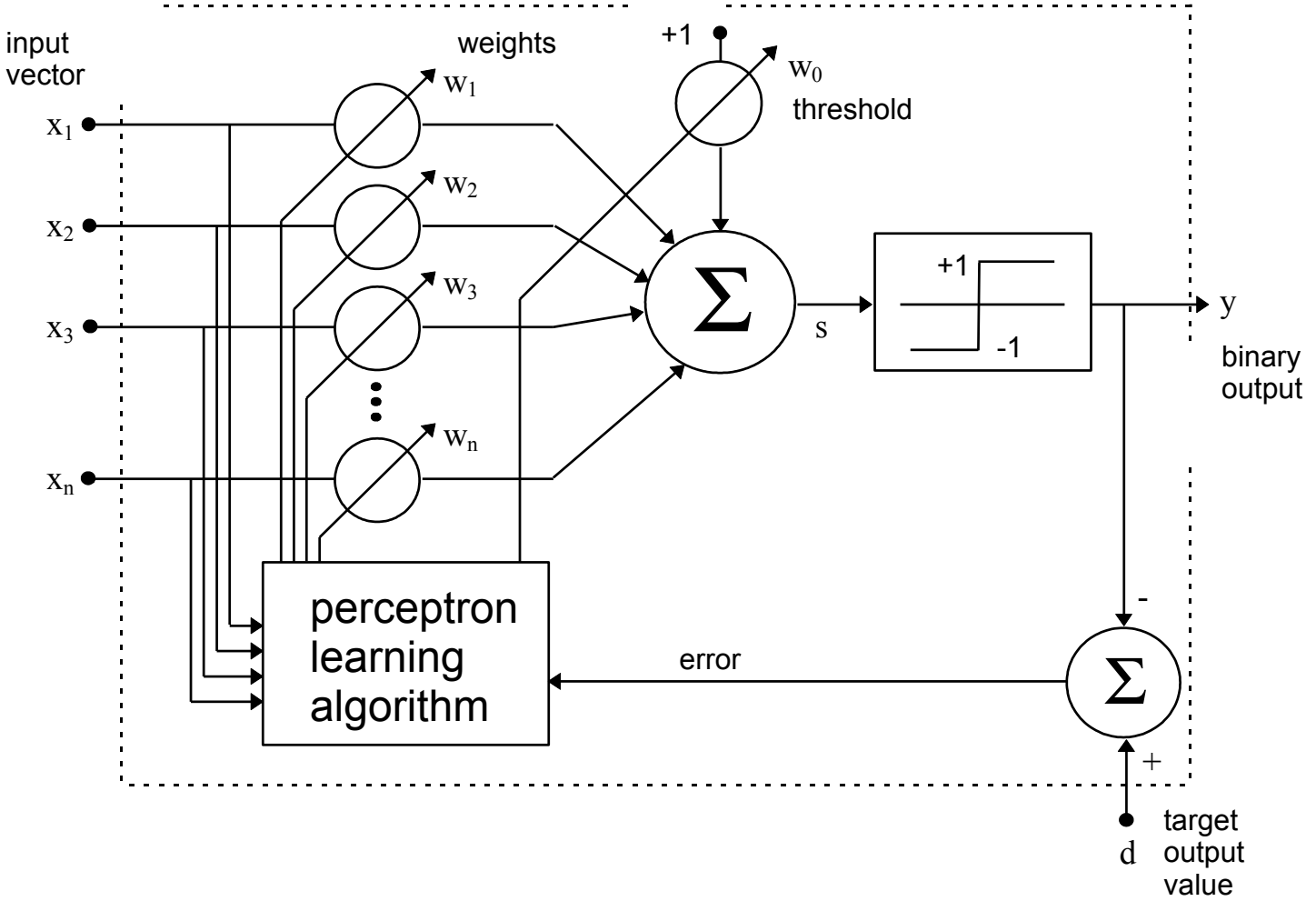


Perceptrons





MP-Perceptron



MP-Perceptron: Learning

For each example pattern, modify perceptron weights:

$$w_i(t+1) = w_i(t) + \eta\, e(t)\, x_i(t)$$

$$e(t) = d(t) - y(t)$$

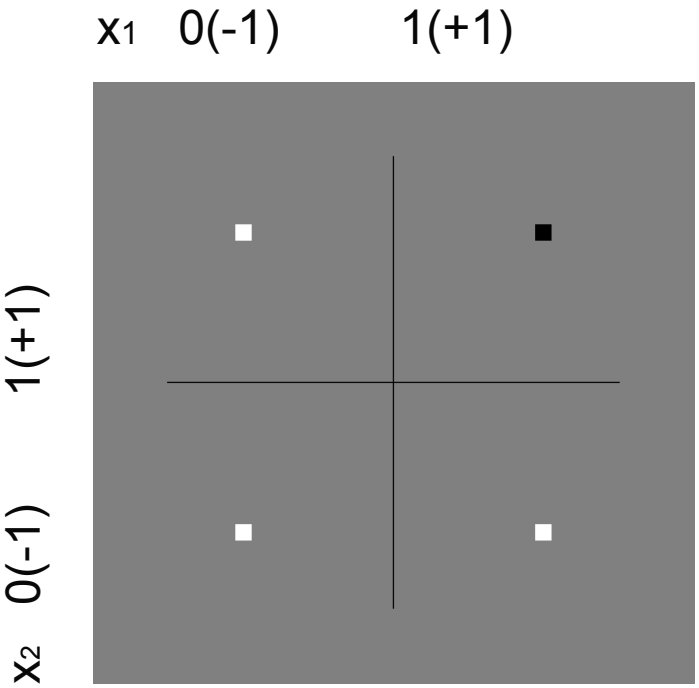
error

where

w_i	weight of i-th input	x_i	value of i-th input
η	learning rate <0;1)	y	neuron output
		d	desired output

Demonstration

- Learning AND...



white square 0, black 1

0 encoded as -1, 1 as +1

-1 AND -1 = false

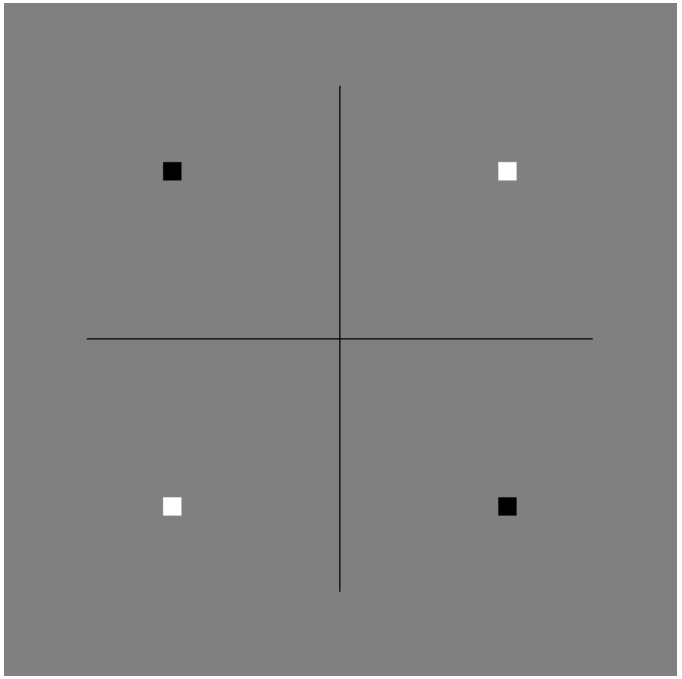
-1 AND +1 = false

+1 AND -1 = false

+1 AND +1 = true

Minsky-Papert's Blunder 1/2

- Both have exactly proved, that this ANN (understand: perceptron) is not convenient even for implementation of so simple logic function as XOR.



-1 XOR -1 = *false*

-1 XOR +1 = *true*

+1 XOR -1 = *true*

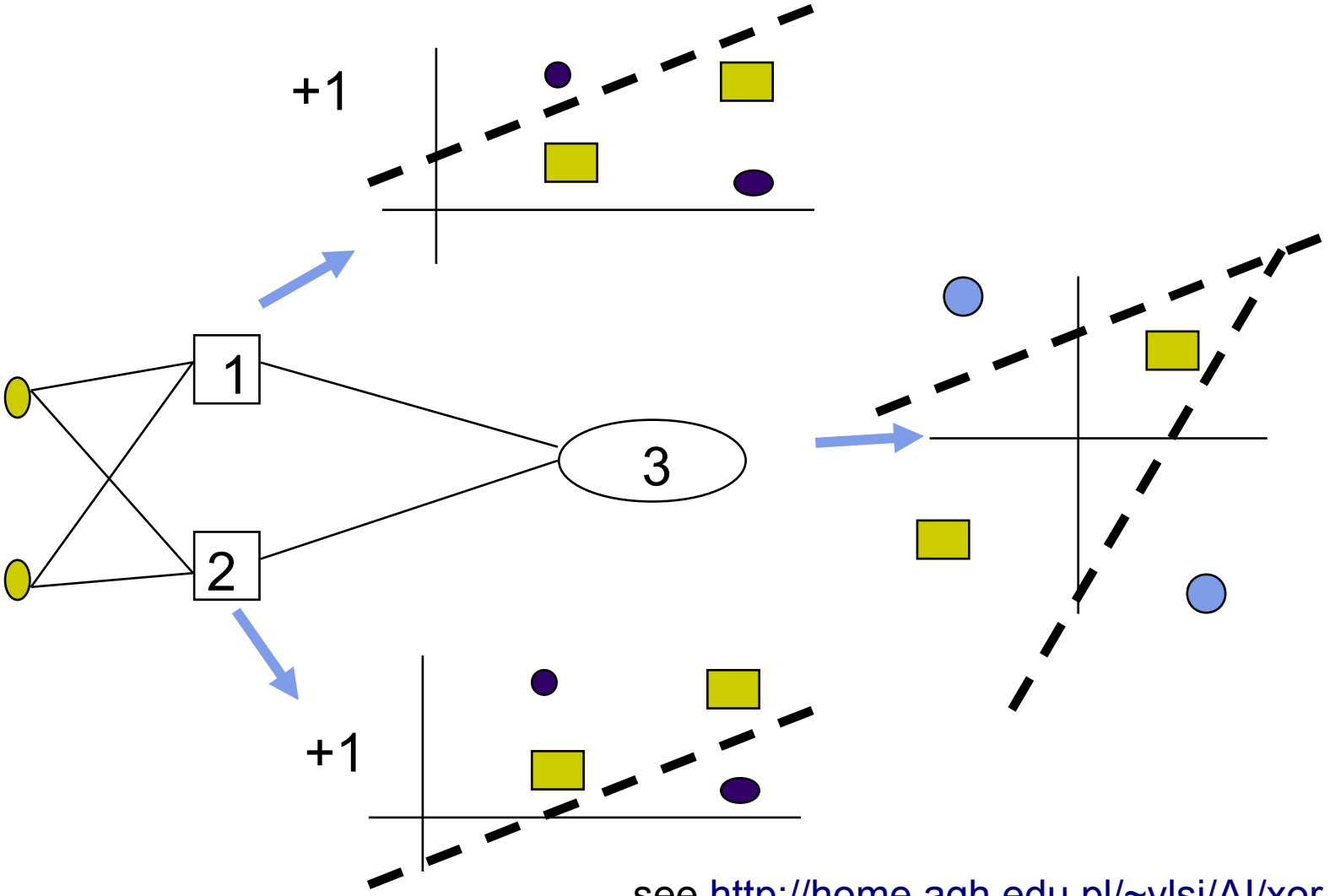
+1 XOR +1 = *false*

Impossible to separate
the classes by a single line!

Minsky-Pappert's Blunder 2/2

- They have previous correct conclusion incorrectly generalized for all ANNs. The history showed, that they are responsible for more than two decades delay in the ANN investigations.
- **Linear non-separability!**

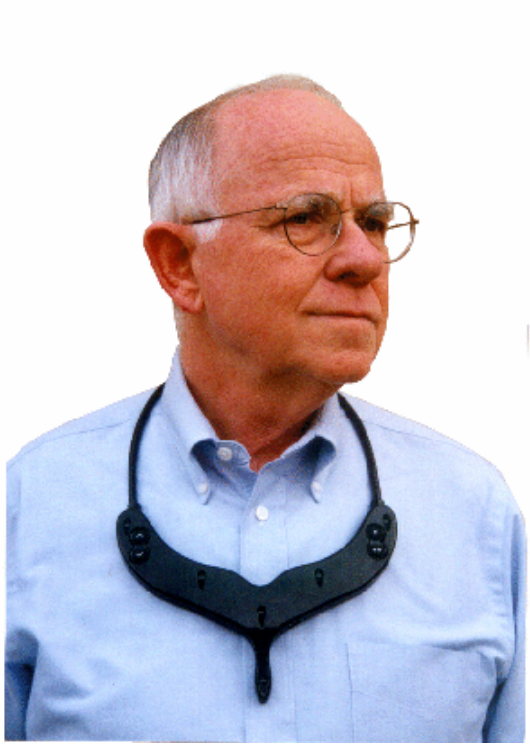
XOR Solution - Add a Layer



see http://home.agh.edu.pl/~vlsi/AI/xor_t/en/main.htm

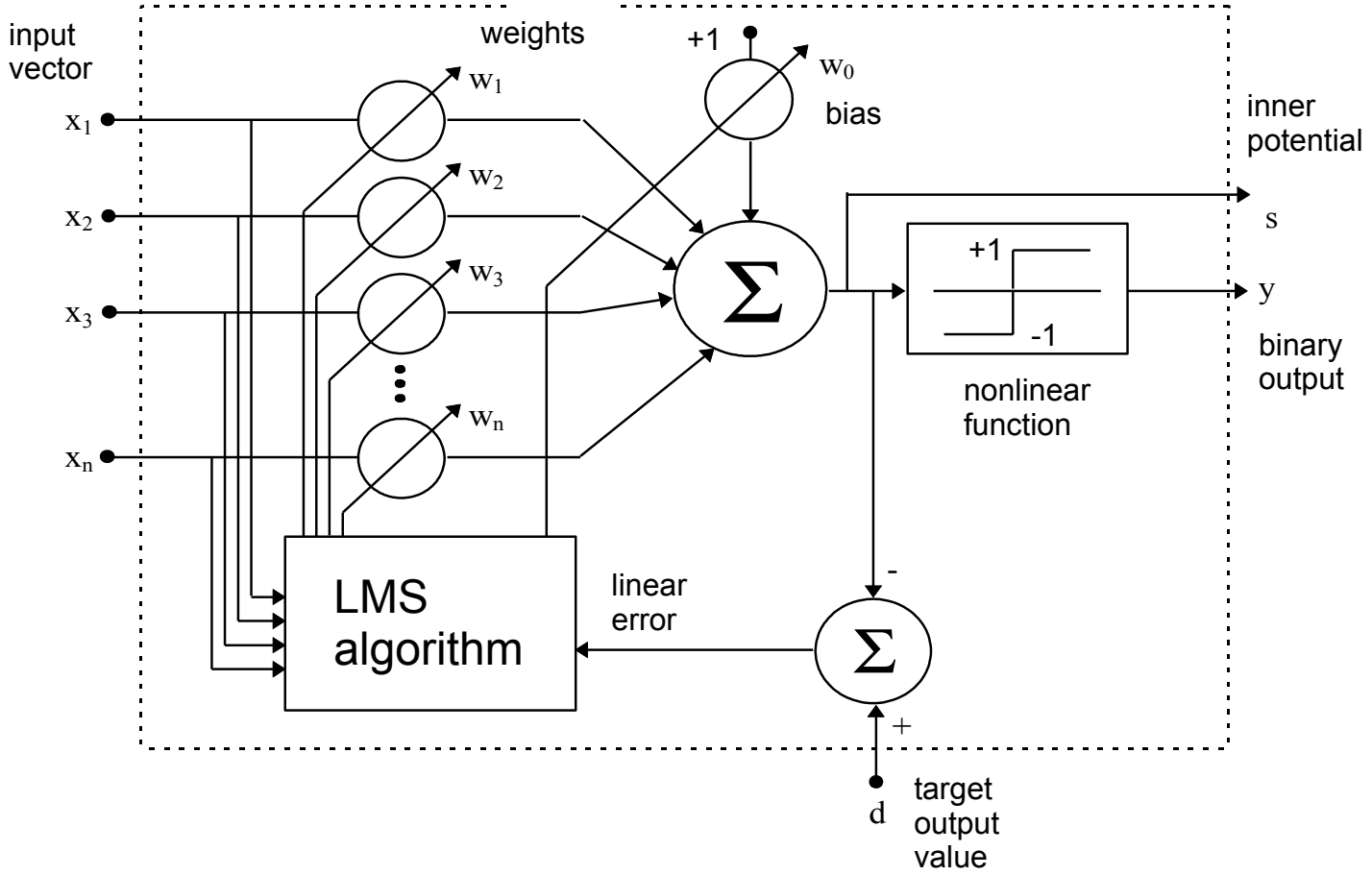
ADALINE

- Bernard Widrow, Stanford university, 1960.
- Single perceptron.
- Bipolar output: +1,-1.
- Binary, bipolar or real valued input.
- New learning algorithm:
  - LMS/delta rule.



- Application: echo cancellation in long distance communication circuits (still used).

ADALINE - **AdA**ptive **Li**near **Ne**uron



http://www.learnartificialneuralnetworks.com/perceptronadaline.html

ADALINE Learning Algorithm: LMS/Delta Rule

$$w_i(t+1) = w_i(t) + \eta\, e(t)\, x_i(t)$$

$$e(t) = d(t) - s(t)$$

where

w_i	weight of i-th input	x_i	value of i-th input
η	learning rate <0;1)	s	neurons inner potential
		d	desired output

Least Mean Square also called Widrow-Hoff's Delta rule.

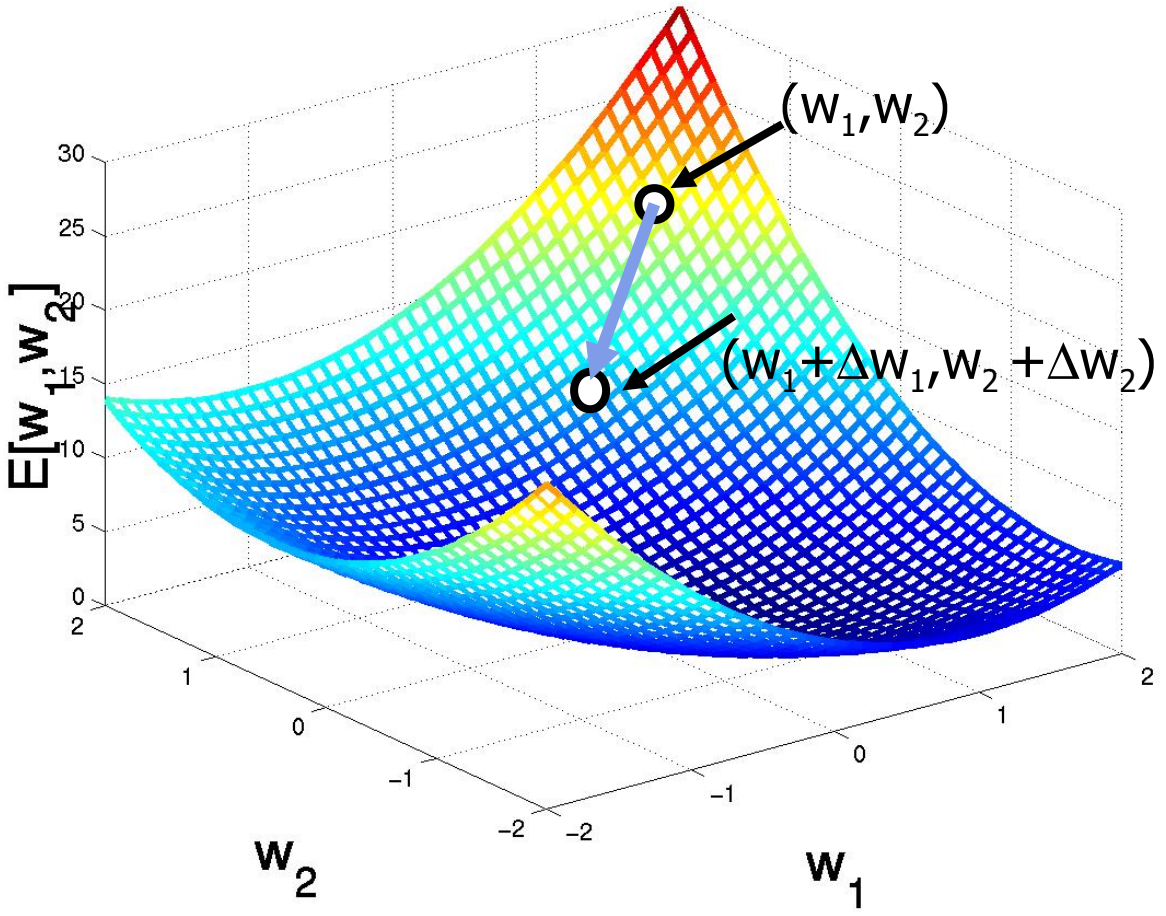
MP-perceptron vs. ADALINE

- Perceptron: $\Delta w_i = \eta(d-y)x_i$
- Delta rule: $\Delta w_i = \eta(d-s)x_i$
- y -vs- s (output or inner potential).
- Delta rule can asymptotically approach minimum of error for linearly non-separable problems. Perceptron NOT.
- Perceptron always converges to error free classifier of linearly separable problem, delta rule might not succeed!

LMS Algorithm … ?

- Goal: minimize error E,
- where $E = \Sigma(e_j(t)^2)$,
- j is for j-th input pattern.
- E is only a function of the weights.
- Gradient descent method.

Graphically: Gradient Descent



There Are Two Ways

- **Incremental mode**:
  - gradient descent over individual training examples.
- **Batch mode**:
  - gradient descent over the entire training data set.

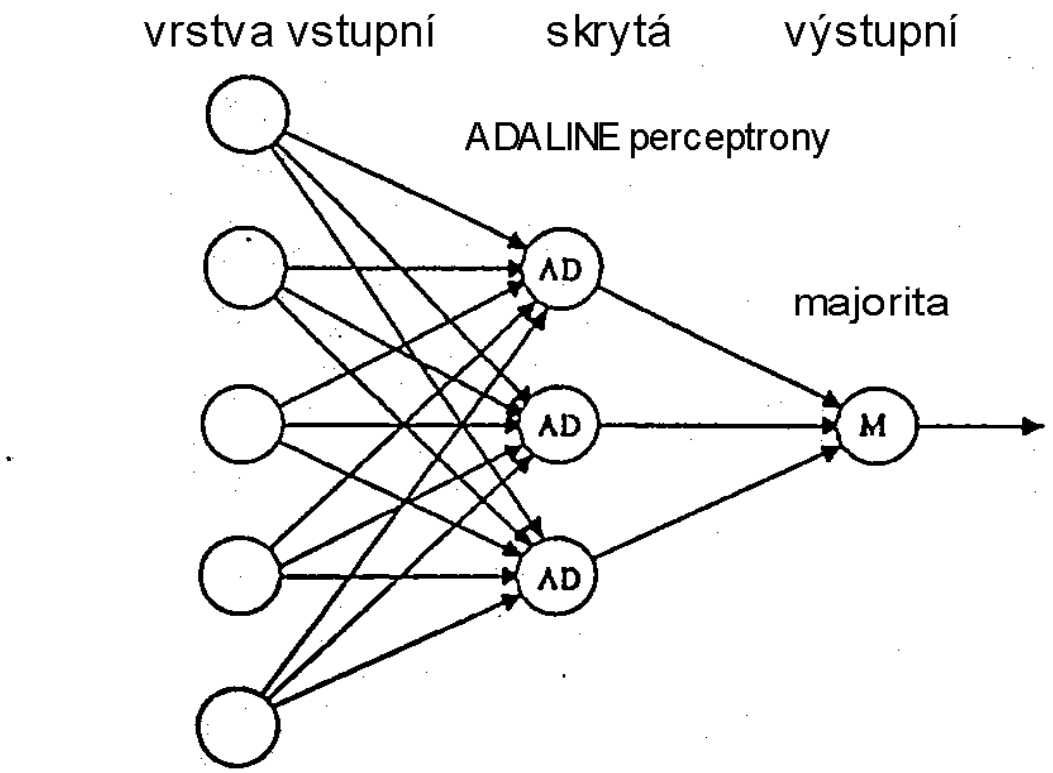
Multi-Class Classification? 1-of-N Encoding

- Perceptron can only handle two-class outputs. What can be done?
- Answer: for N-class problems, learn N perceptrons:
  - *Perceptron 1 learns "Output=1" vs "Output ≠ 1"*
  - *Perceptron 2 learns "Output=2" vs "Output ≠ 2"*
  - *:*
  - *Perceptron N learns "Output=N" vs "Output ≠ N"*
- To classify an output for a new input, just predict with each perceptron and find out which one puts the prediction the furthest into the positive region.

MADALINE (**Many ADALINE**)

- Processes binary signals.
- Bipolar encoding.
- One hidden, one output layer.
- Supervised learning
- Again B. Widrow.

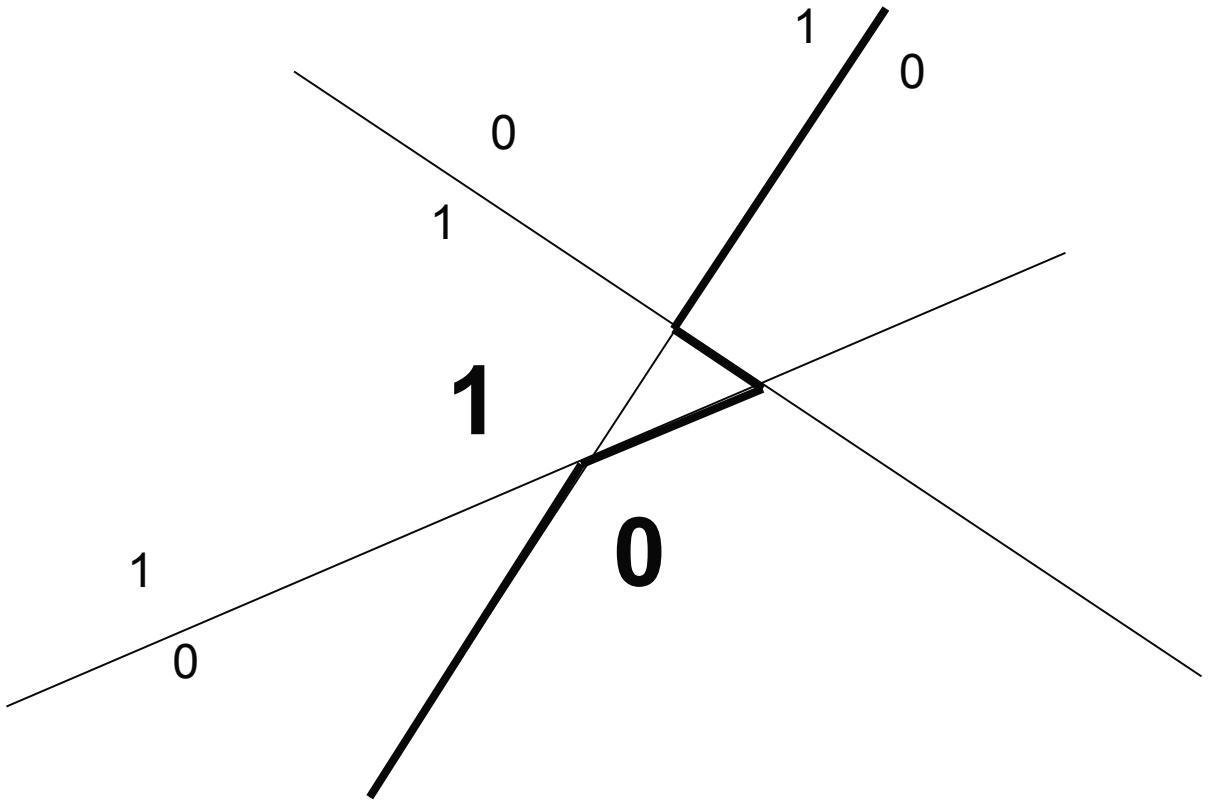
MADALINE: Architecture



Note: We learn only weights at ADALINE inputs.

MADALINE: Linearly Non-Separable Problems

- MADALINE can learn to linearly non-separable problems:



MADALINE Limitations

- MADALINE is unusable for complex tasks.
- ADALINEs learn independently, they are unable to separate the input space to form a complex decision boundary.

Next Lecture

- Multilayered Perceptron (MLP).
- Backpropagation learning algorithm.