# Neural Network

- *Mimics the functionality of a brain.*
- *A neural network is a graph with neurons (nodes, units etc.) connected by links.*

# Neural Network: Neuron

x_0

w_0 synapse

axon from a neuron

$w_0 x_0$

dendrite

cell body

$w_1 x_1$

$$\sum_i w_i x_i + b$$

f

$f\left(\sum_i w_i x_i + b\right)$

output axon

activation function

$w_2 x_2$

Dendrite

Axon from another cell

Axonal arborization

Synapse

Axon

Nucleus

Synapses

Cell body or Soma
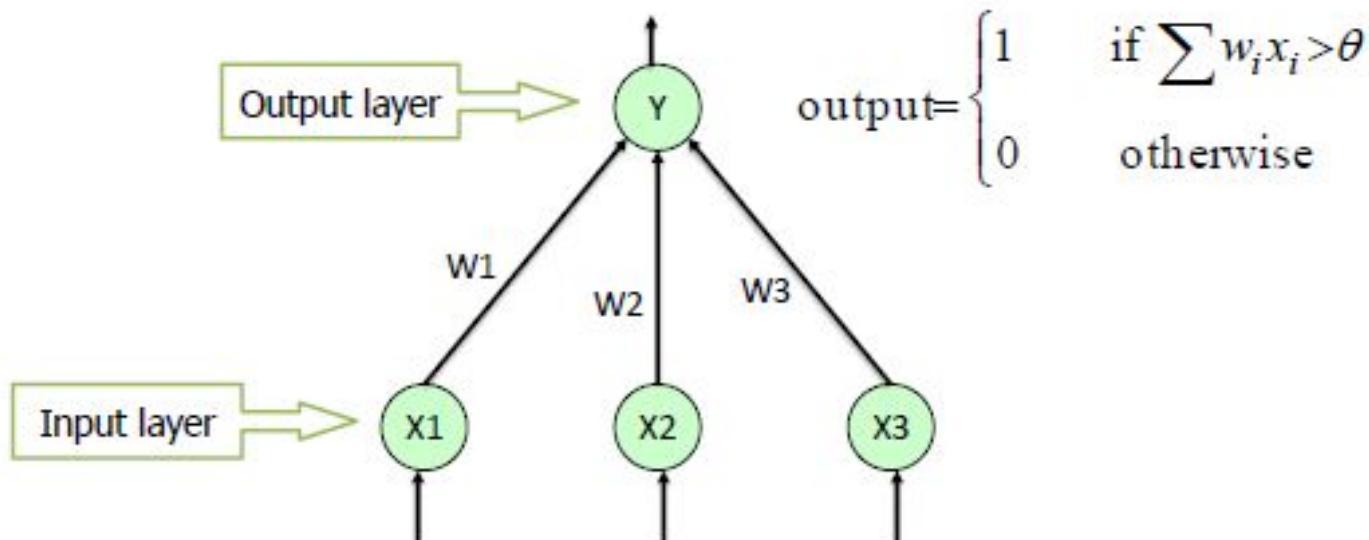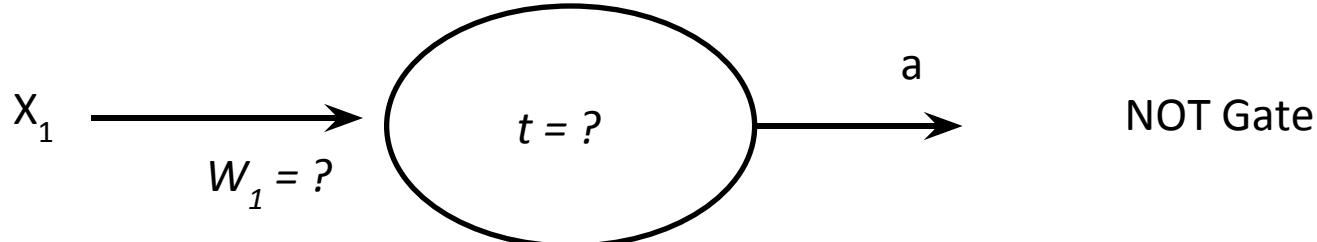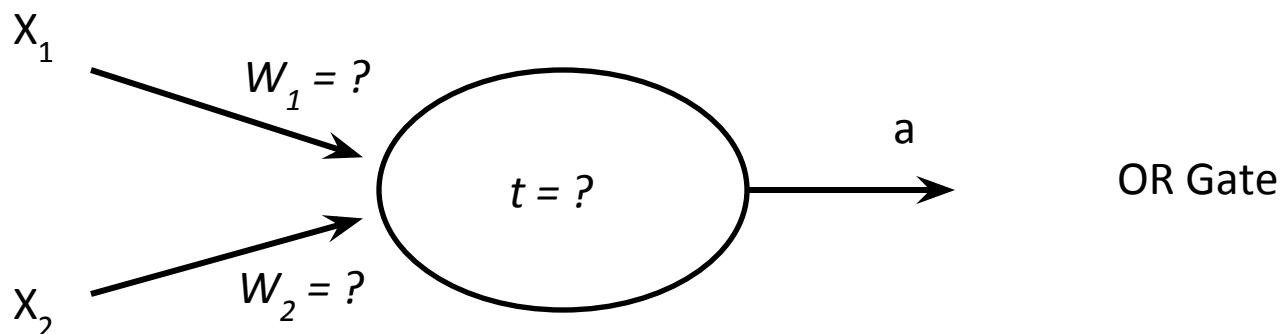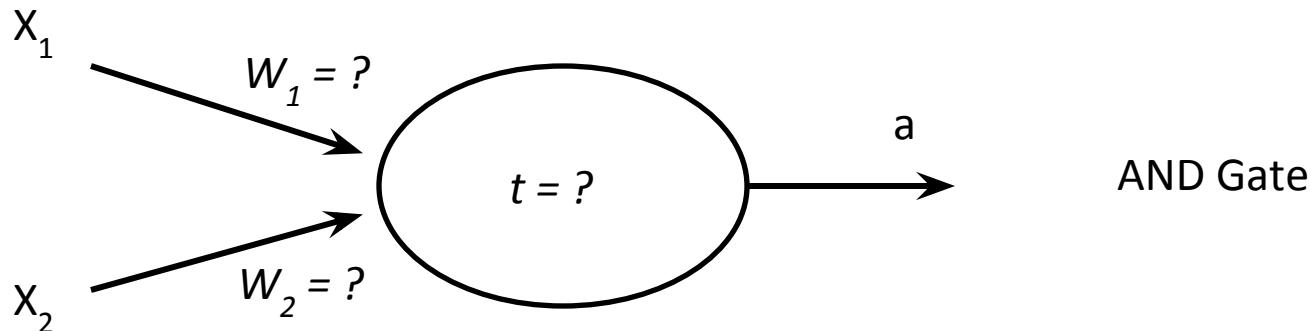
Neural Network: Perceptron

- Network with only single layer.
- No hidden layers
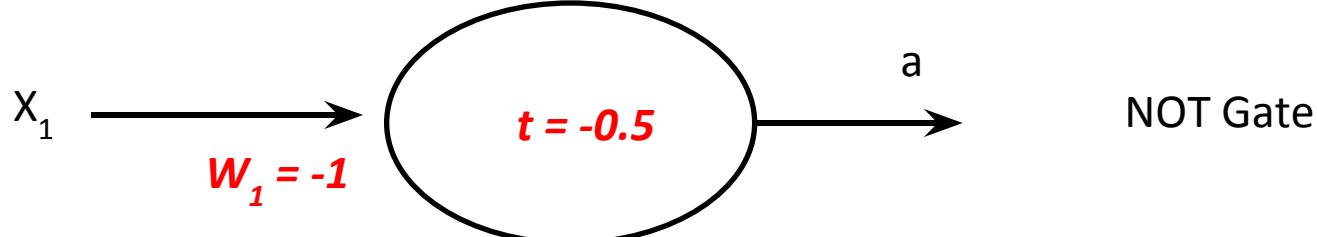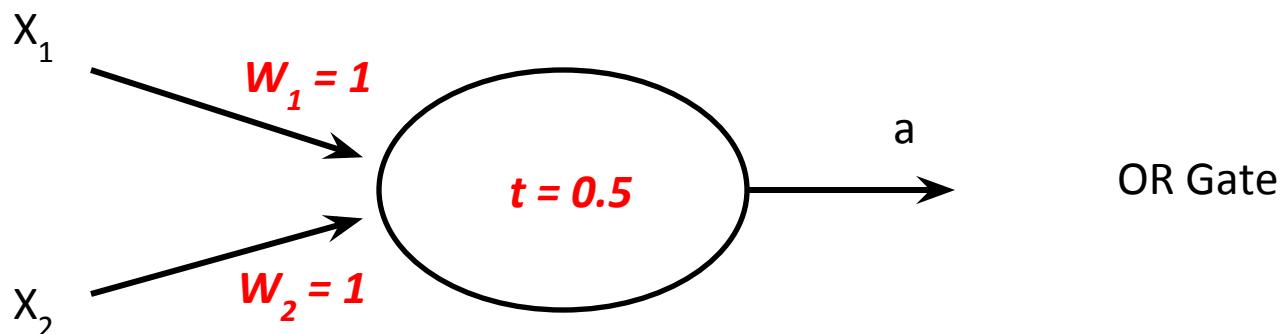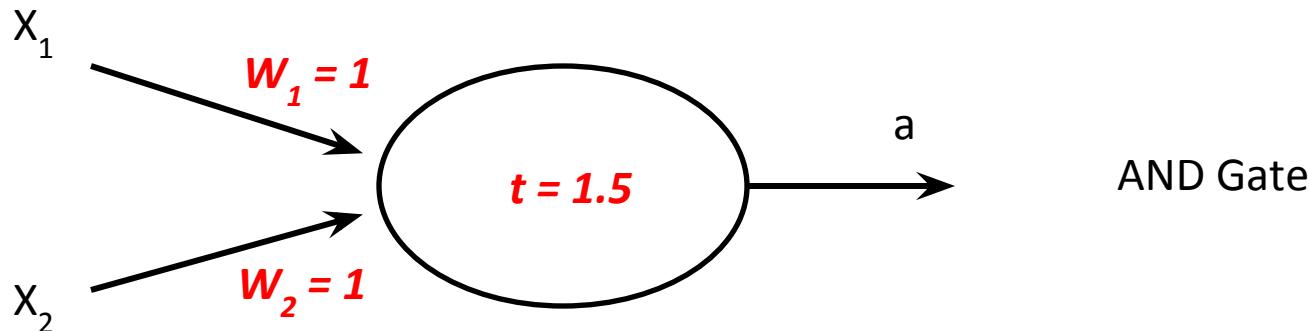
Single Layer Perceptron

Output layer $\Rightarrow$ Y

$$\text{output} = \begin{cases} 1 & \text{if } \sum w_i x_i > \theta \\ 0 & \text{otherwise} \end{cases}$$

W1

W2   W3

Input layer $\Rightarrow$ X1   X2   X3

Neural Network: Perceptron

X_1

$W_1 = ?$

$t = ?$

a

AND Gate

X_2

$W_2 = ?$

X_1

$W_1 = ?$

$t = ?$

a

OR Gate

X_2

$W_2 = ?$

X_1

$t = ?$

a

NOT Gate

$W_1 = ?$

Neural Network: Perceptron

X₁ — $W_1 = 1$ →
X₂ — $W_2 = 1$ →
$t = 1.5$ → a   AND Gate

X₁ — $W_1 = 1$ →
X₂ — $W_2 = 1$ →
$t = 0.5$ → a   OR Gate

X₁ — $W_1 = -1$ →
$t = -0.5$ → a   NOT Gate
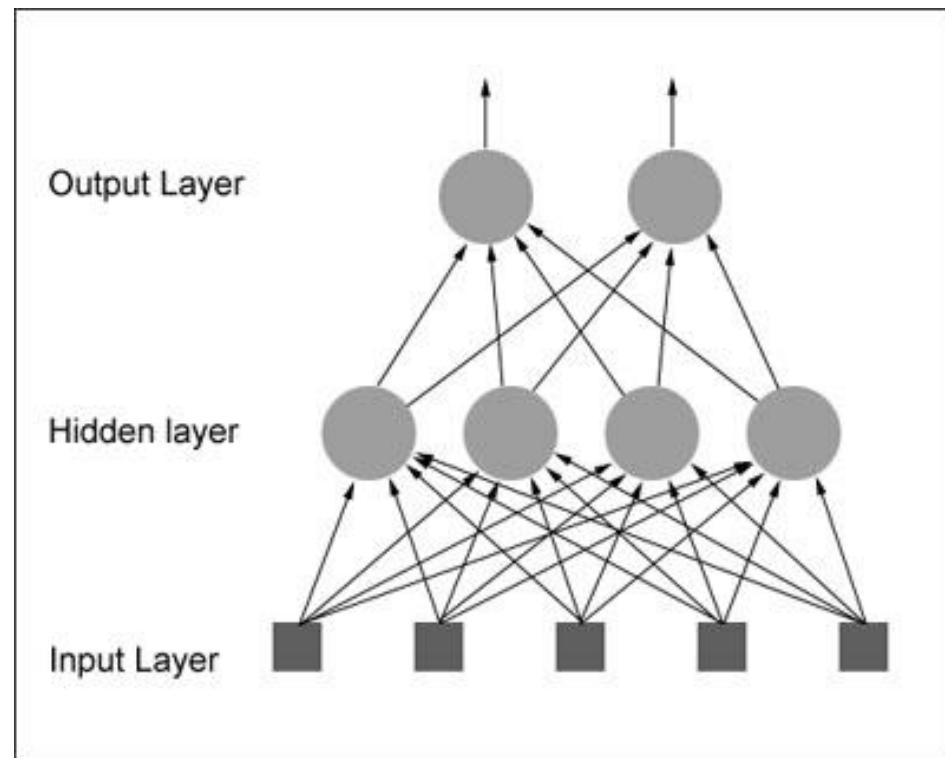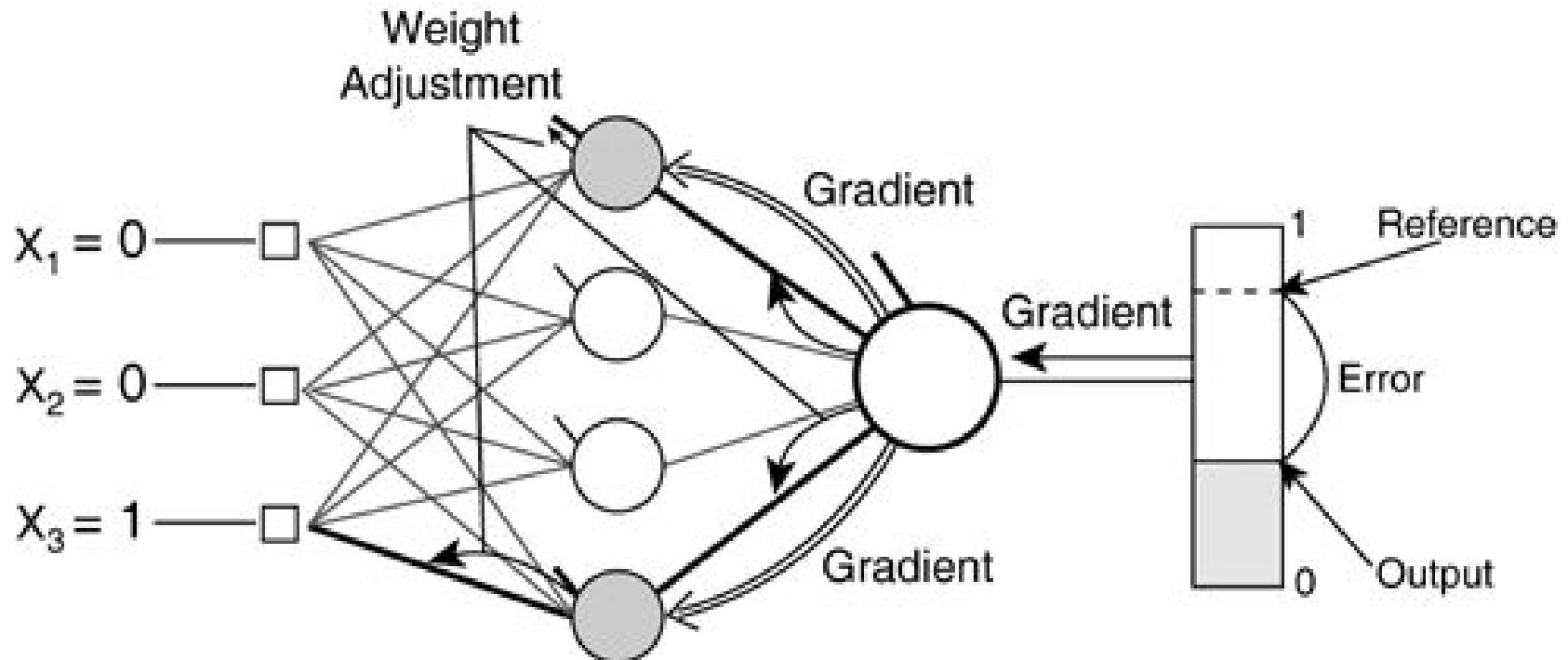
Neural Network: Multi Layer Perceptron (MLP) or Feed-Forward Network (FNN)

- Network with *n*+1 layers
- One output and *n* hidden layers.
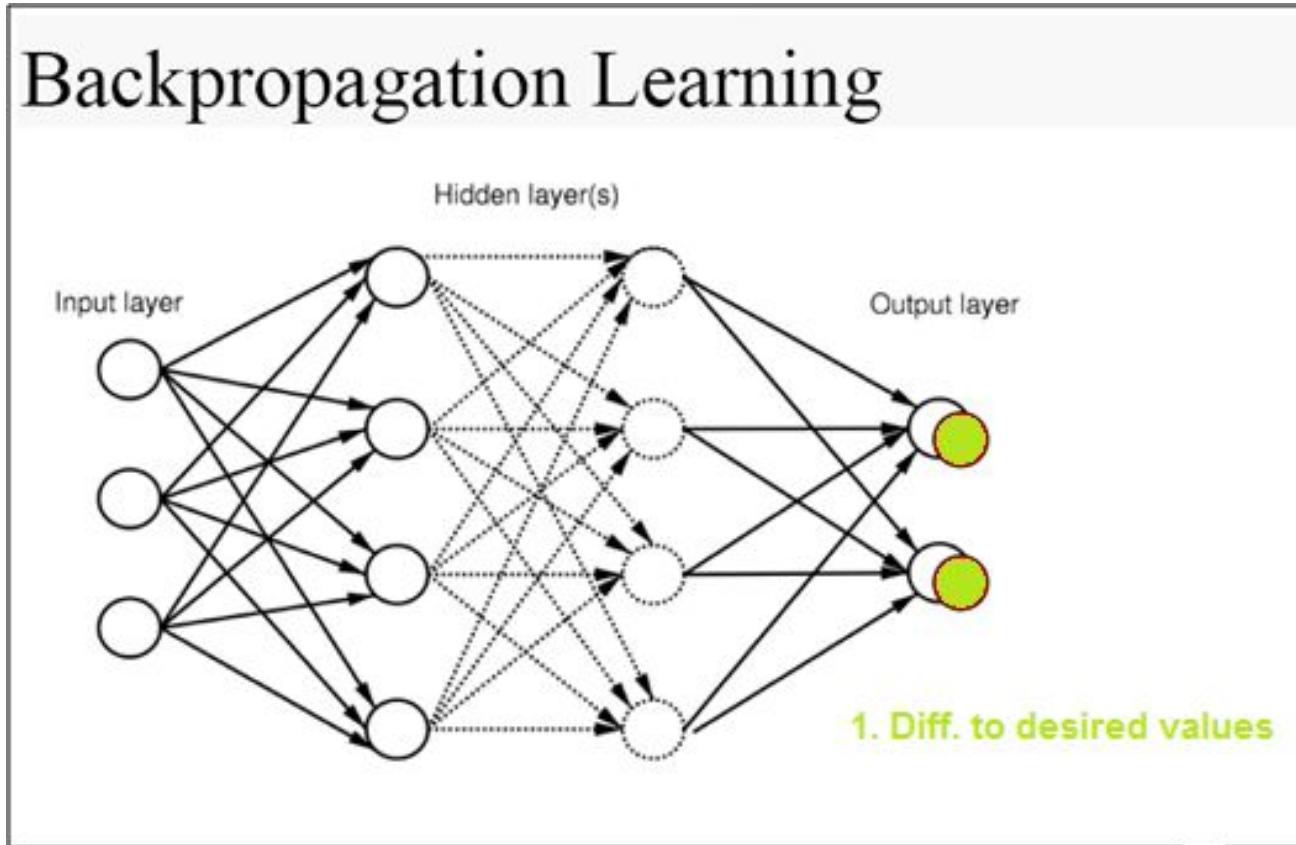
# Training: Back propagation algorithm
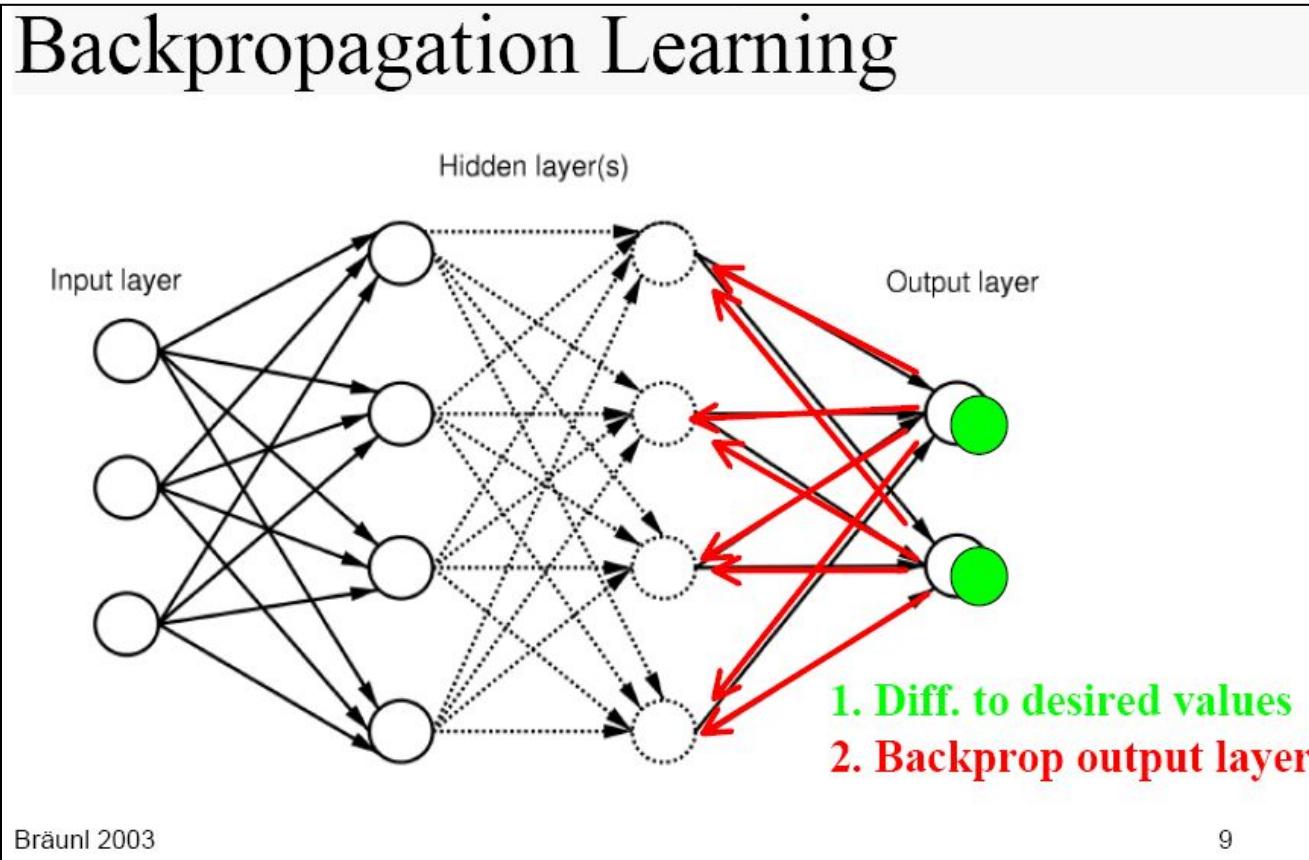
- Gradient decent algorithm

# Training: Back propagation algorithm
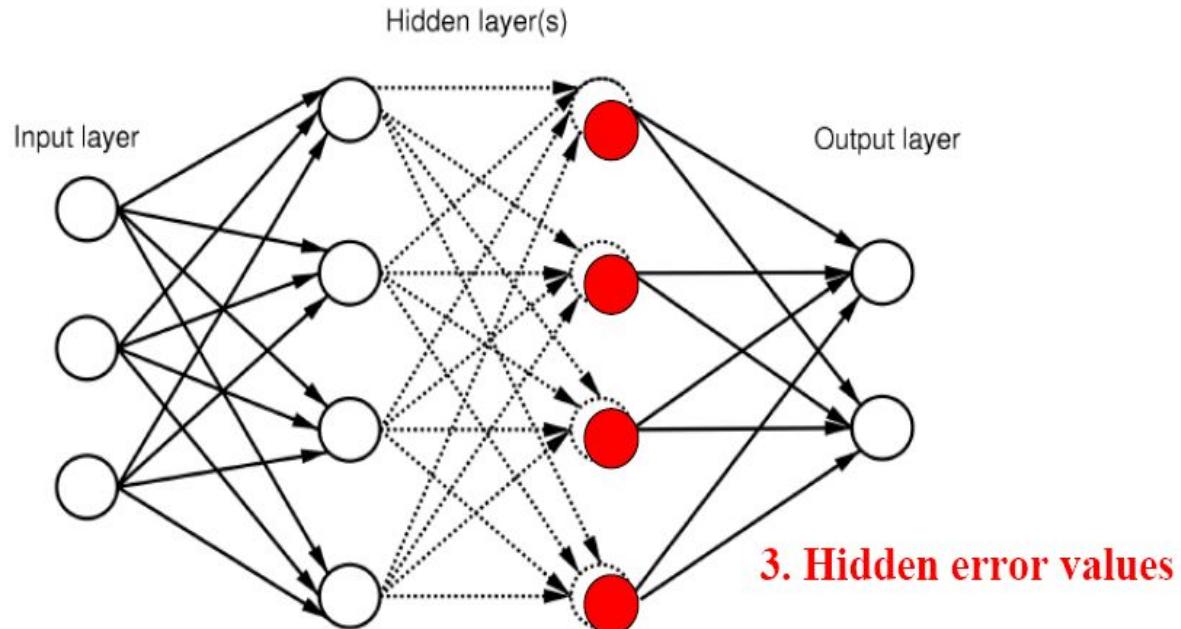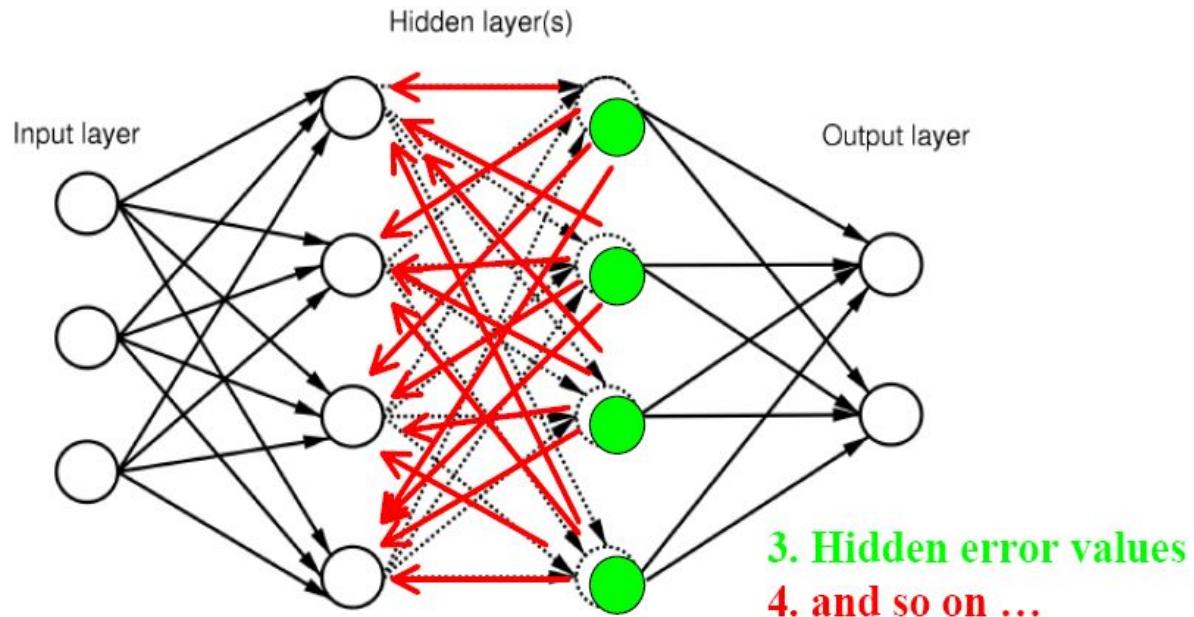
Training: Back propagation algorithm

Backpropagation Learning

Hidden layer(s)

Input layer

Output layer

1. Diff. to desired values
2. Backprop output layer

Bräunl 2003

9

Training: Back propagation algorithm

Backpropagation Learning

Input layer
Hidden layer(s)
Output layer

3. Hidden error values

Bräunl 2003

10

Training: Back propagation algorithm

Training: Back propagation algorithm

1. Initialize network with random weights

2. For all training cases (called examples):

   a. Present training inputs to network and calculate output

   b. For <u>all layers</u> (starting with output layer, back to input layer):

      i. Compare network output with correct output (error function)

      ii. Adapt weights in current layer