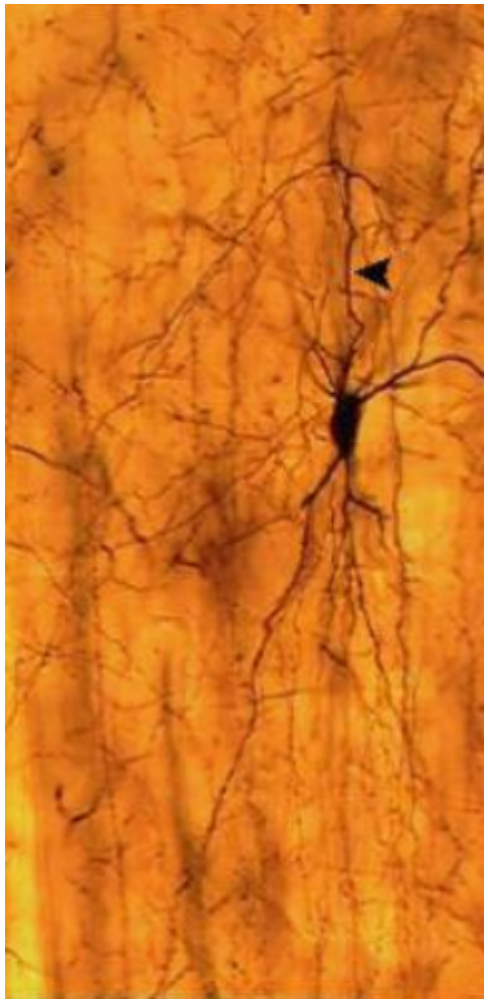
Freie Universität Berlin



Neural Networks, Part 1

Data Visualization and Data Mining

Talk 1 by Jana Cavojska

Instructors:

Prof. Dr. Agnès Voisard

Daniel Kressner

Overview

1. **What are neural networks?**
2. A single perceptron
3. Perceptron training
4. A simple network
5. Learning via backpropagation
6. Convolutional neural networks
7. 2nd talk: Visualization

1. What are Neural Networks?

- **classifiers**:
Input data → Class of input data

- consist of layers of neurons (**perceptrons**)

- different **architectures** for different types of problems

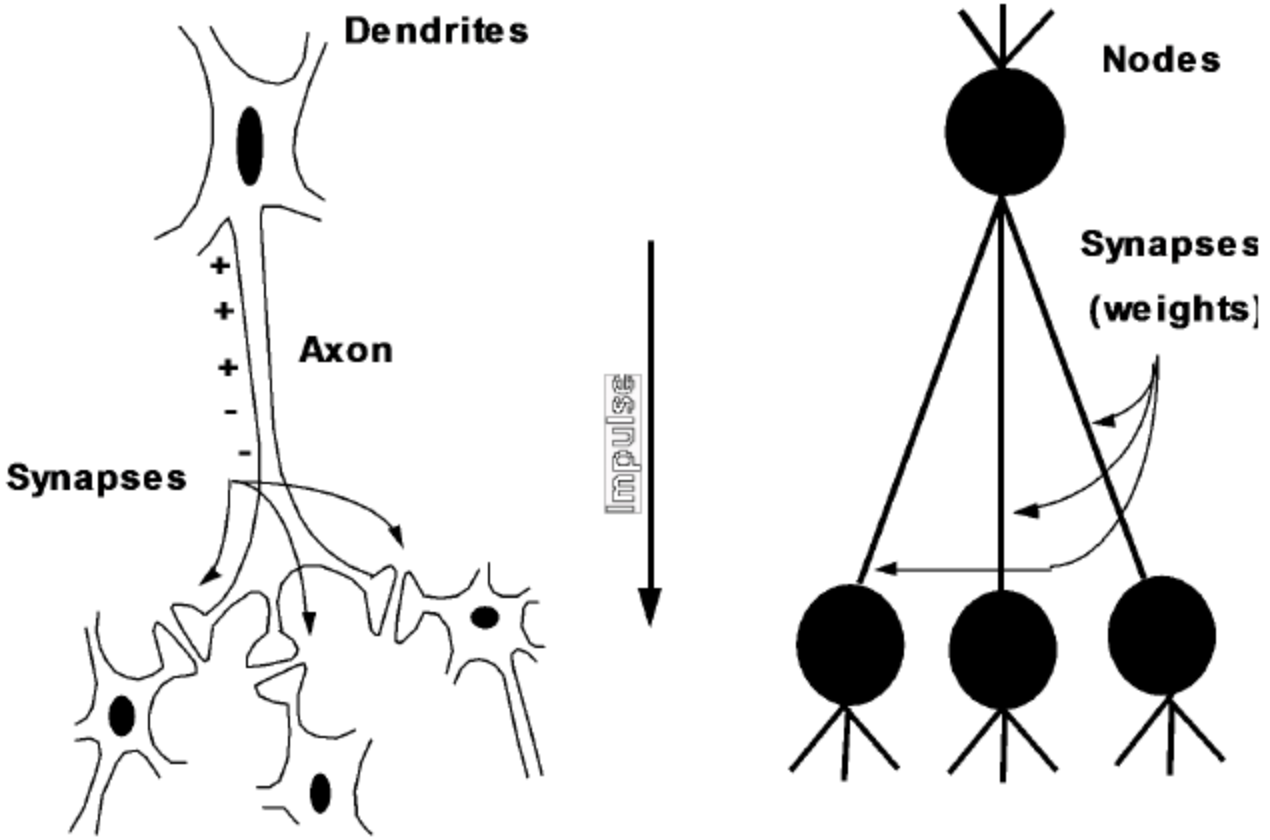
How did it all start?

Geoffrey E. Hinton's
2007 paper:

„Learning multiple layers of Representation“

In:
Trends in Cognitive Sciences
- TRENDS COGN SCI ,
vol. 11, no. 10, pp. 428-434,
2007

Neuron

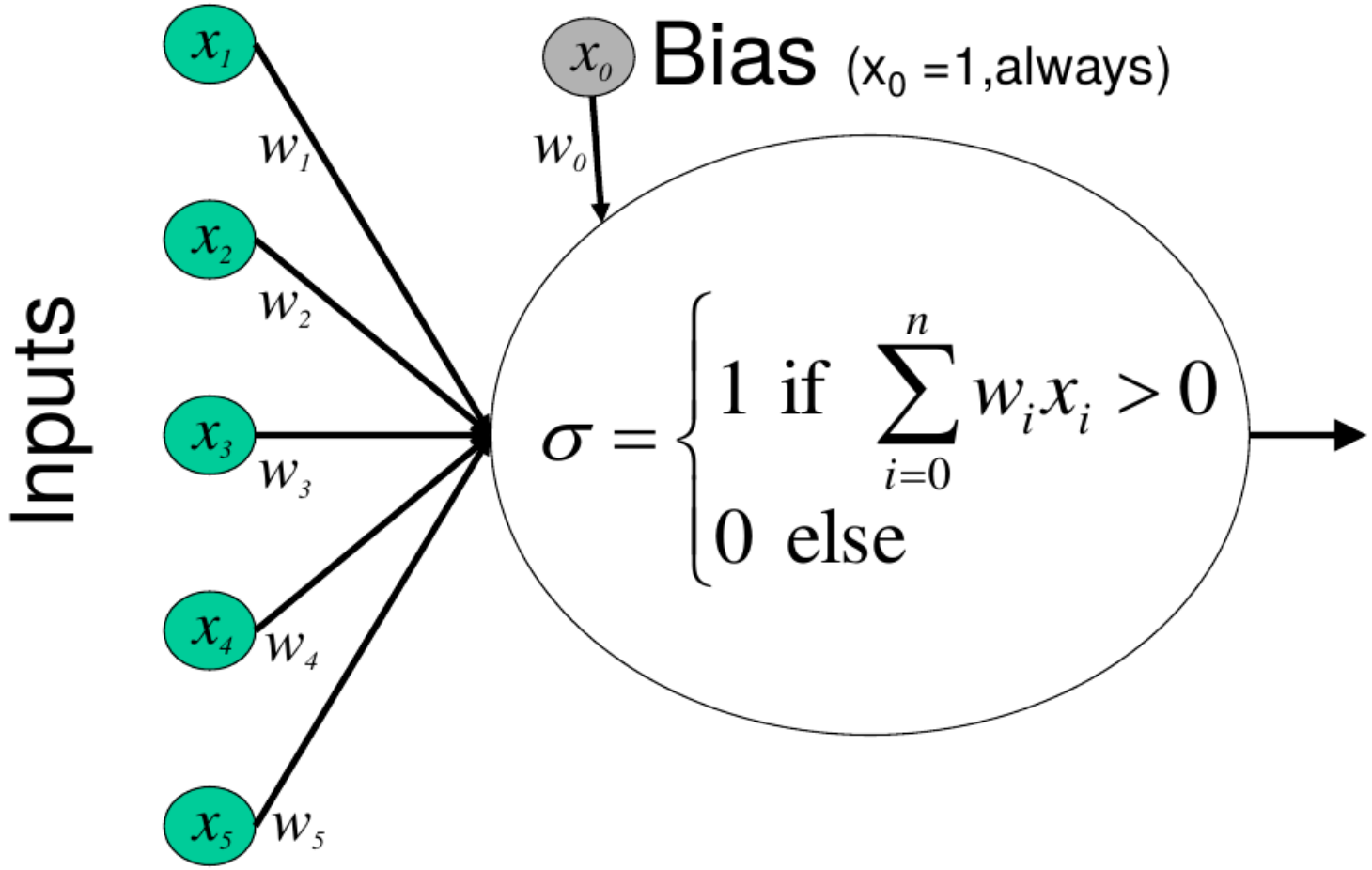


Source: [2]

Overview

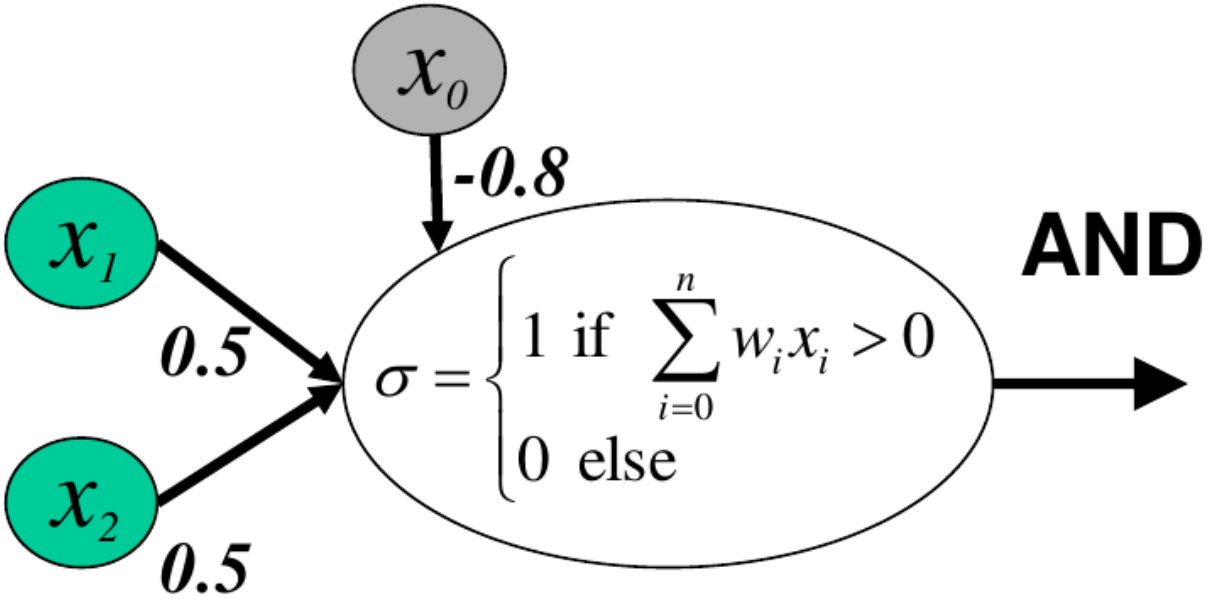
1. What are neural networks?
2. **A single perceptron**
3. Perceptron training
4. A simple network
5. Learning via backpropagation
6. Convolutional neural networks
7. 2nd talk: Visualization

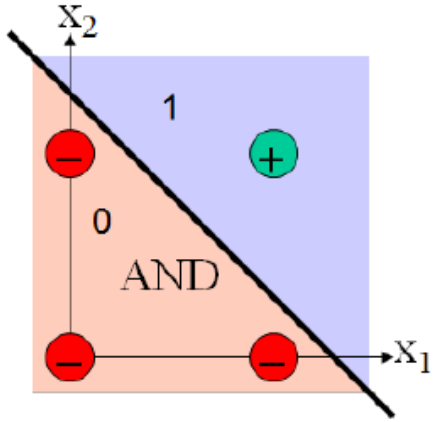
A Single Perceptron



Source: [2]

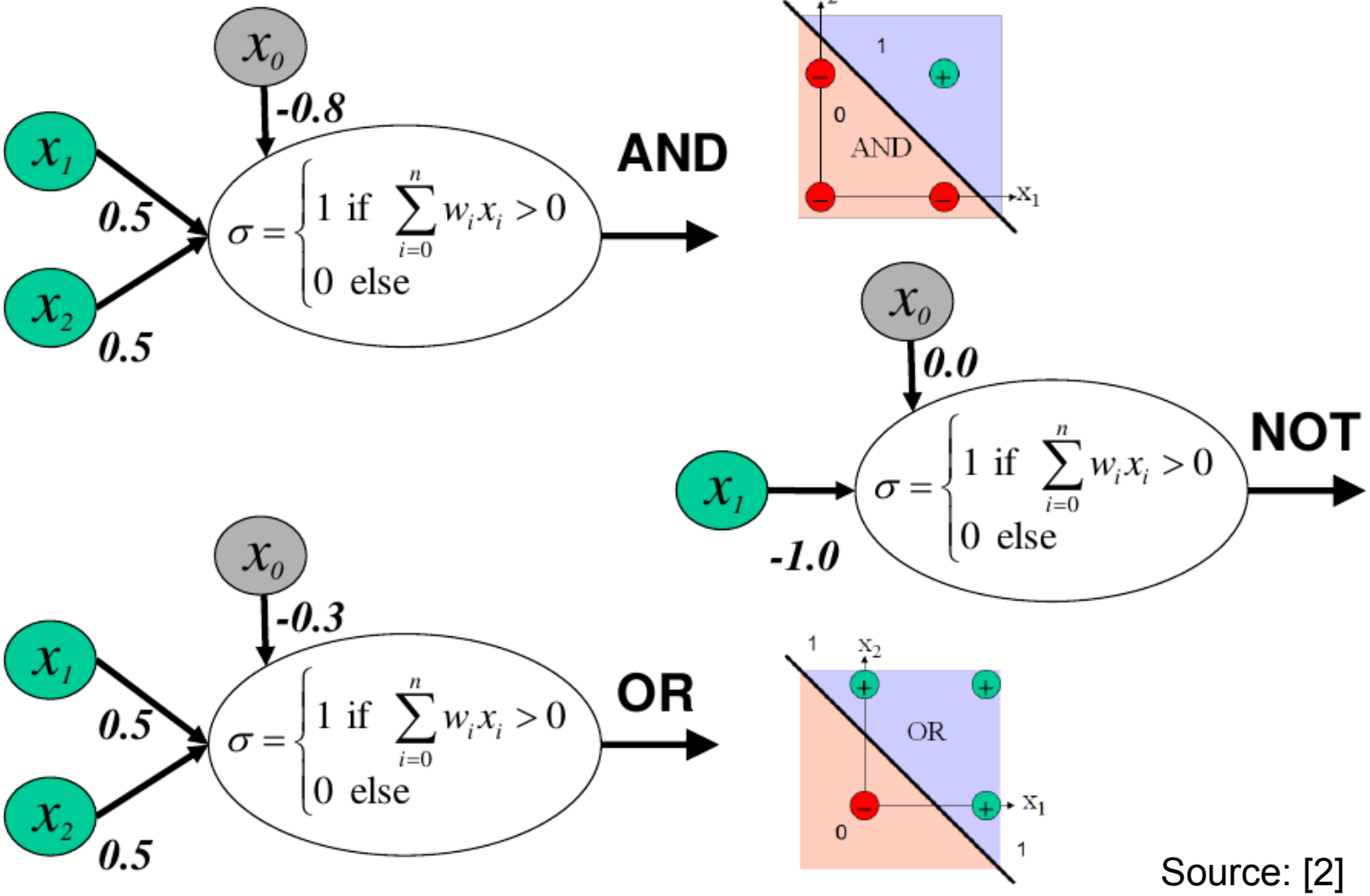
Logical Operators





Source: [2]

Logical Operators

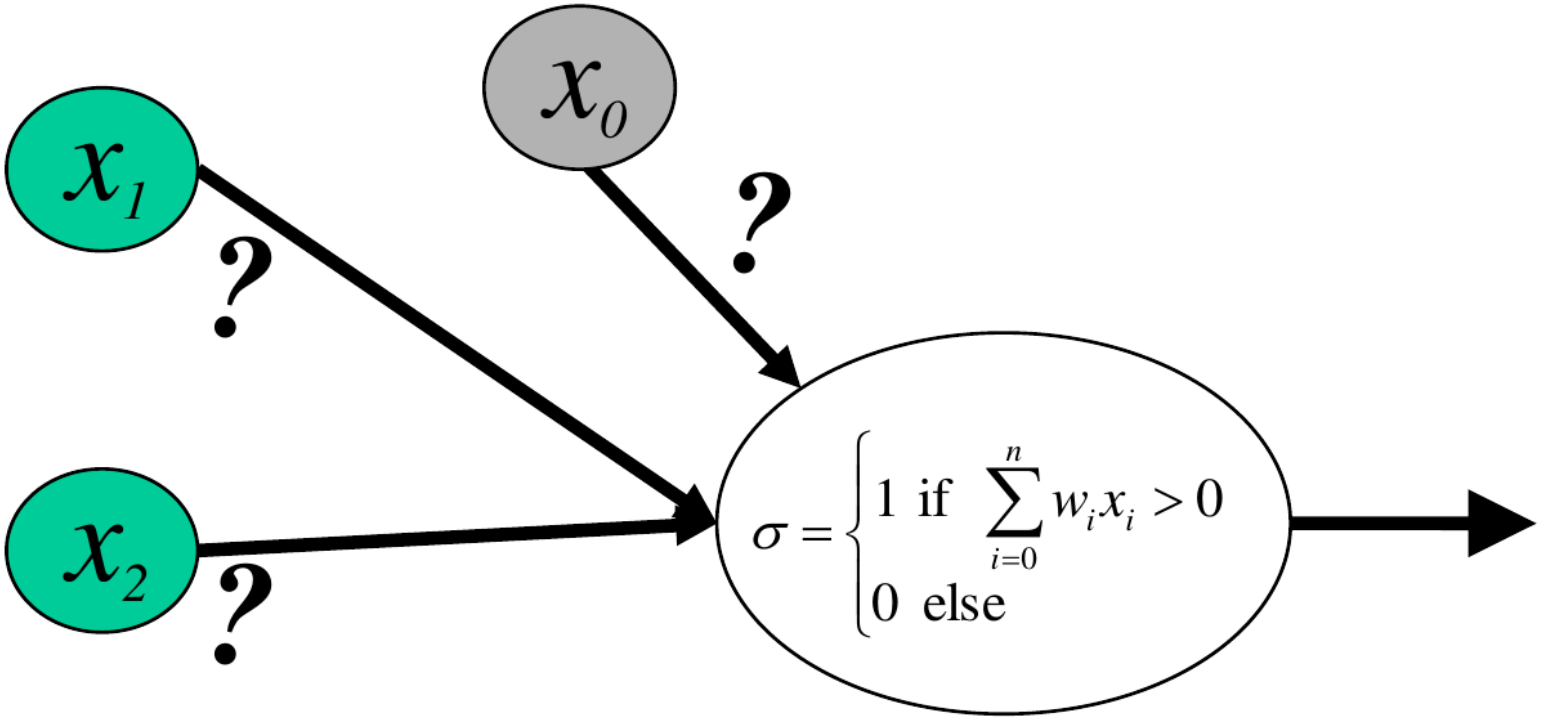


Source: [2]

Overview

1. What are neural networks?
2. A single perceptron
3. **Perceptron training**
4. A simple network
5. Learning via backpropagation
6. Convolutional neural networks
7. 2nd talk: Visualization

Perceptron Training?



Source: [2]

Perceptron Training

$$w_i \leftarrow w_i + \Delta w_i$$

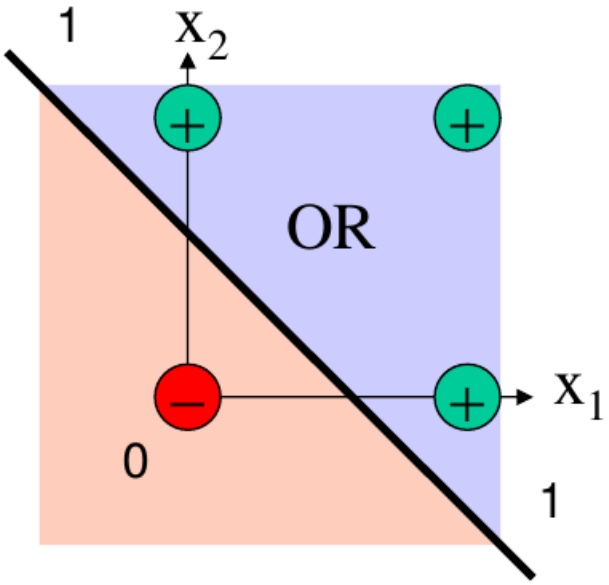
where

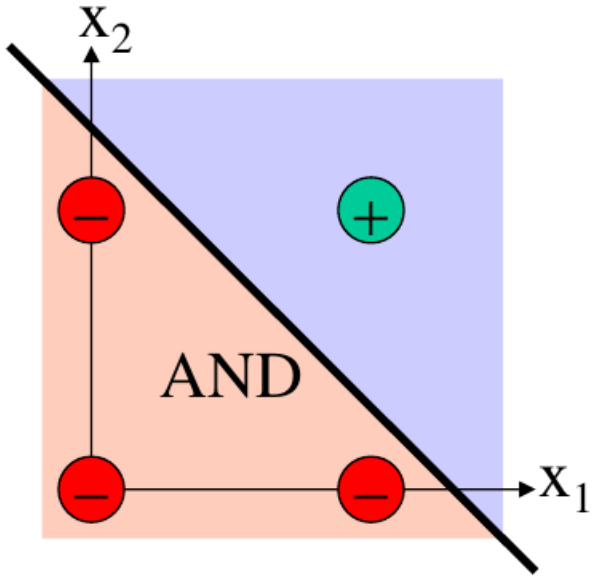
$$\Delta w_i = \eta (t - o) x_i$$

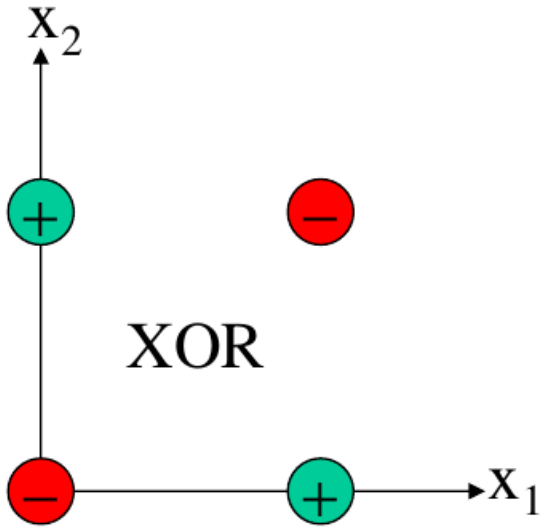
learning rate — target value — perceptron output — input value

Source: [2]

Linear Separability





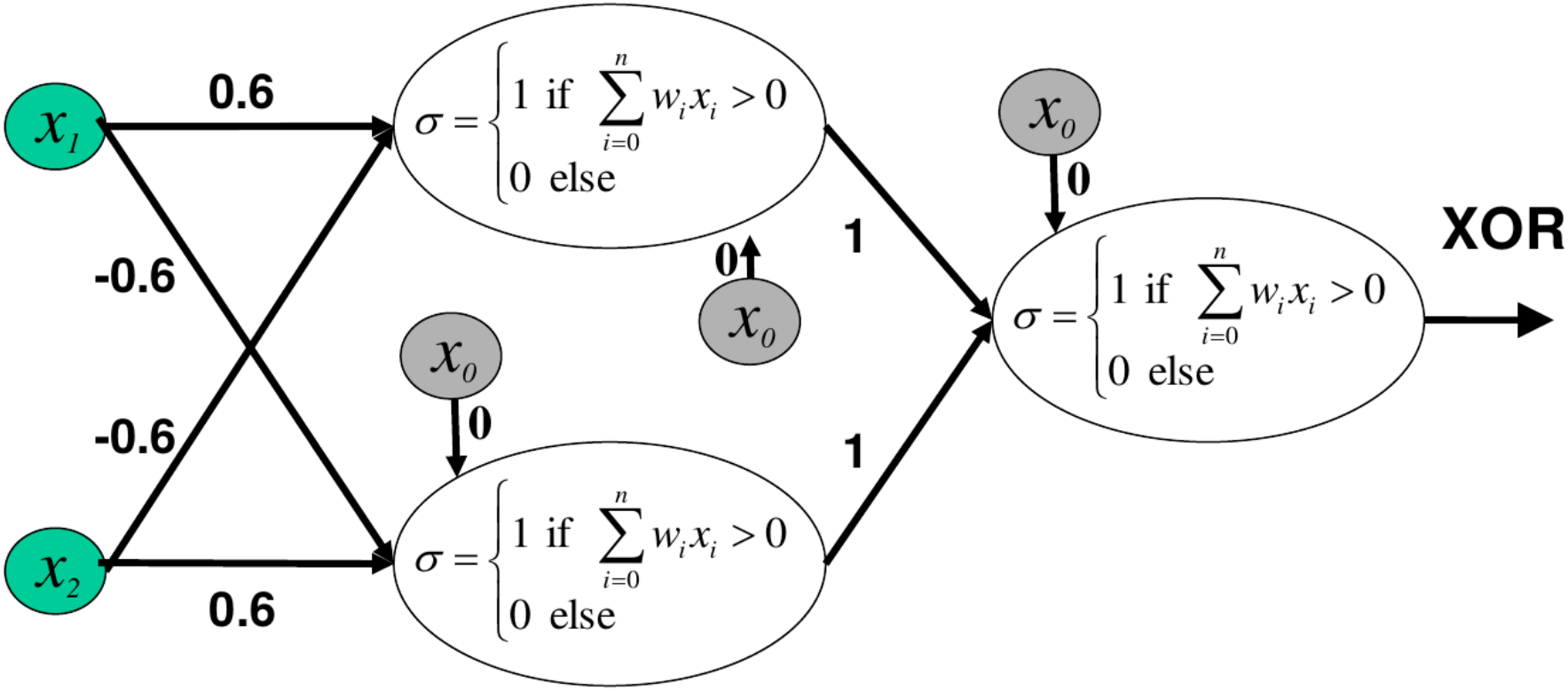


Source: [2]

Overview

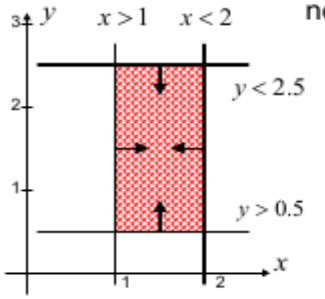
1. What are neural networks?
2. A single perceptron
3. Perceptron training
4. **A simple network**
5. Learning via backpropagation
6. Convolutional neural networks
7. 2nd talk: Visualization

XOR


x_1
x_2
0.6
0.6
-0.6
-0.6
$\sigma = \begin{cases} 1 \text{ if } \sum_{i=0}^{n} w_i x_i > 0 \\ 0 \text{ else} \end{cases}$
x_0
0
x_0
0
x_0
0
1
1
$\sigma = \begin{cases} 1 \text{ if } \sum_{i=0}^{n} w_i x_i > 0 \\ 0 \text{ else} \end{cases}$
XOR

Source: [2]

Multiple Neurons



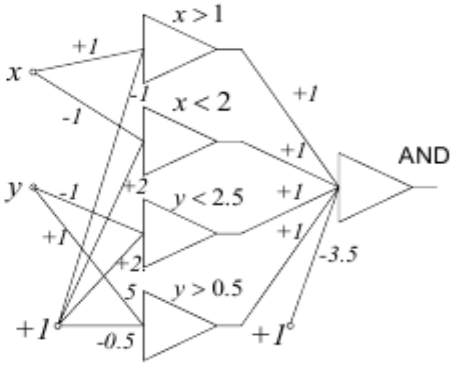
neuron equations:

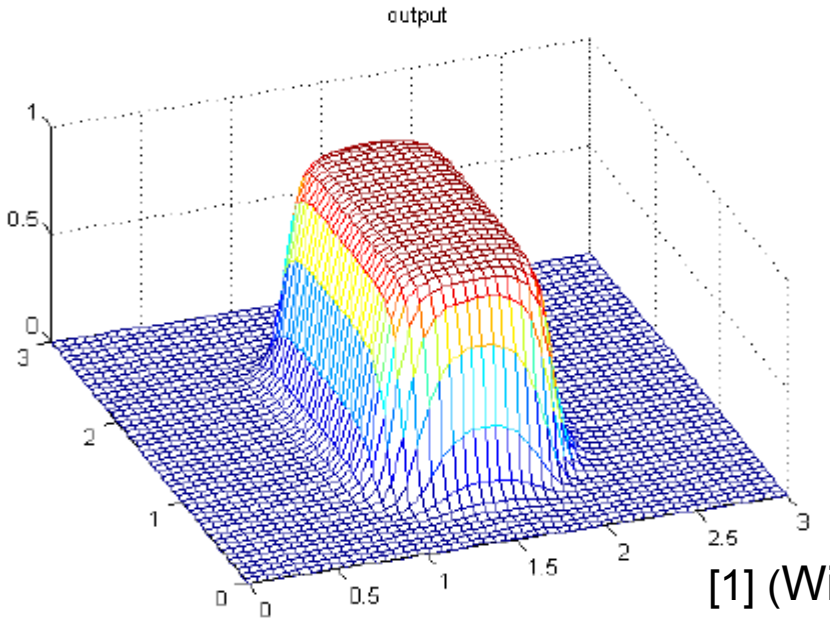
$x - 1 > 0$

$-x + 2 > 0$

$-y + 2.5 > 0$

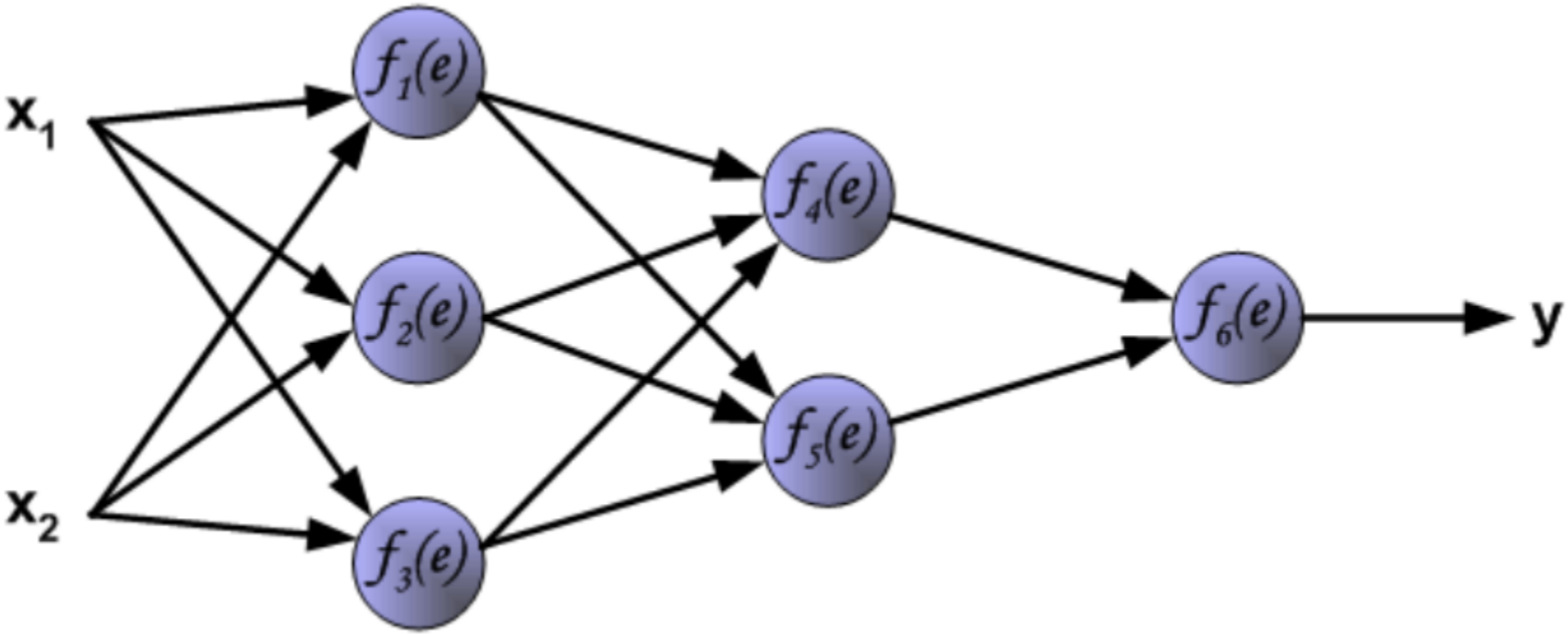
$y - 0.5 > 0$





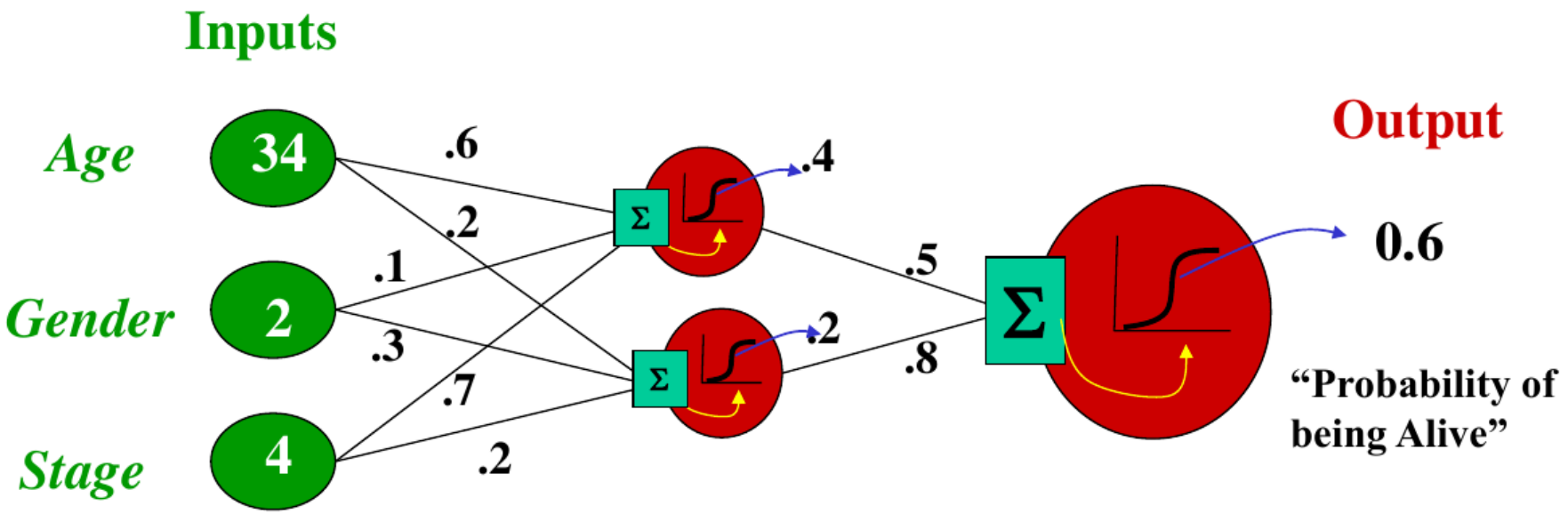
[1] (Wilamowski, 2003)

Example of a Simple Network



Source: [2]

Neural Network Model



Independent variables

Weights

Hidden Layer

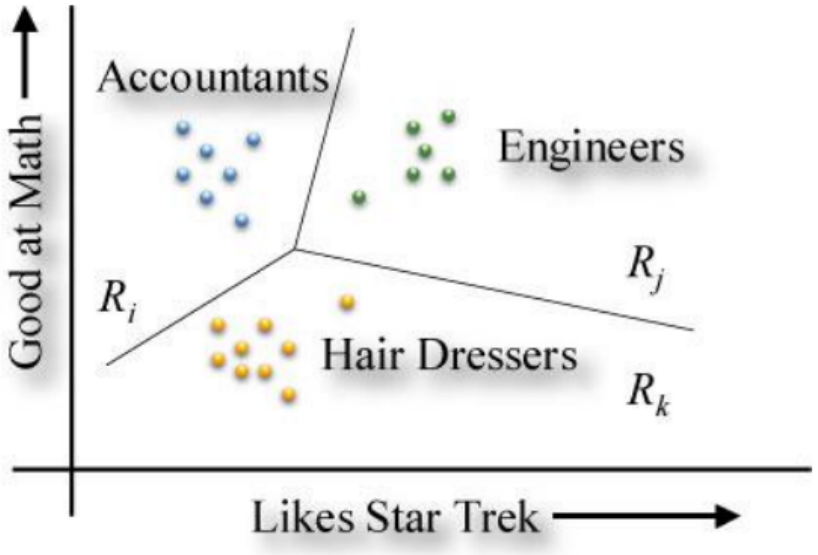
Weights

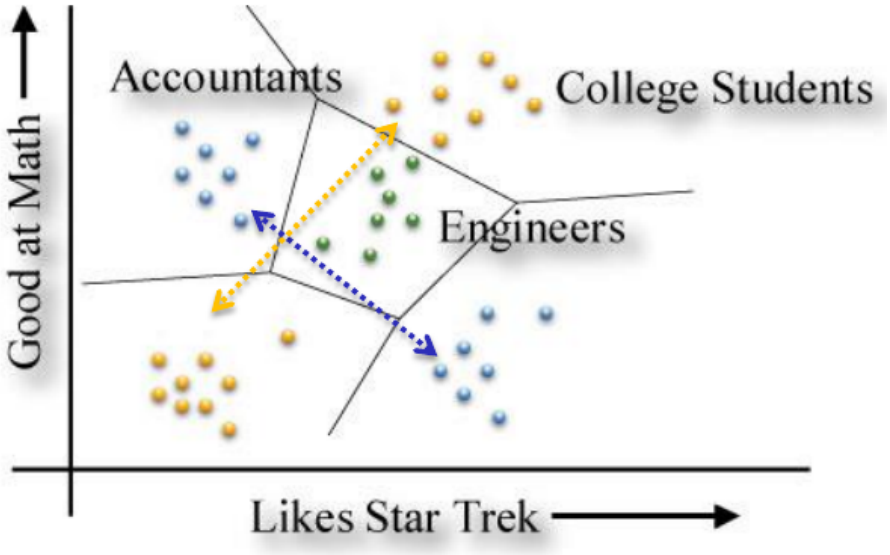
Dependent variable

Prediction

Source: [2]

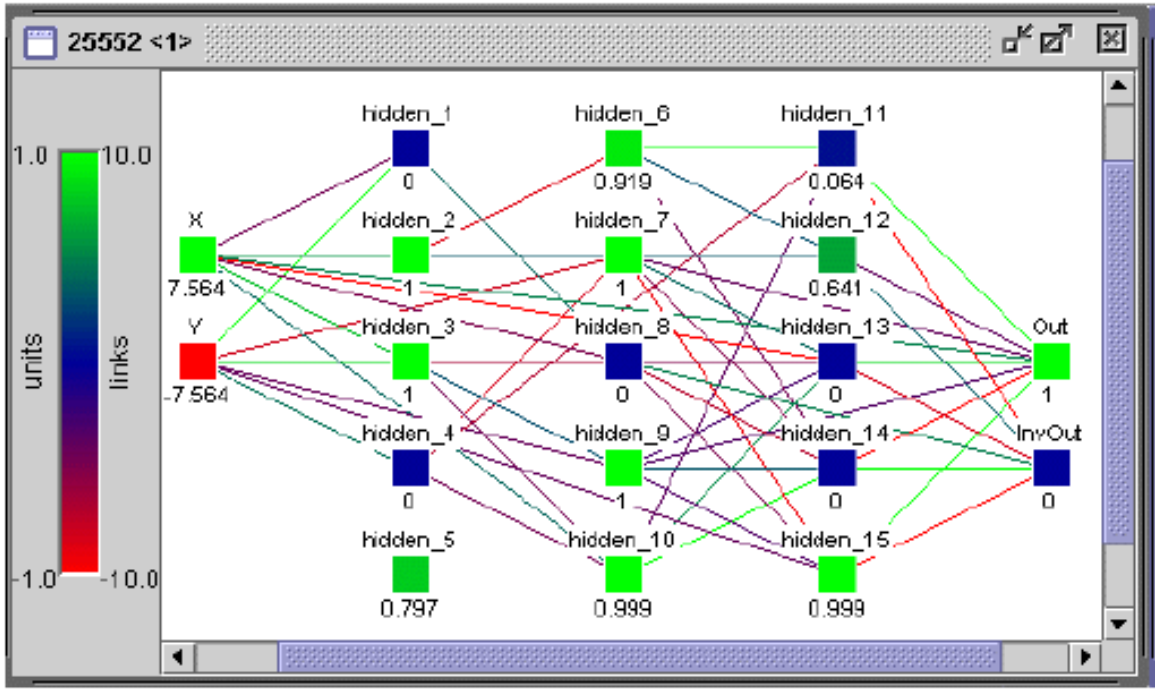
Being creative about Linear Separability

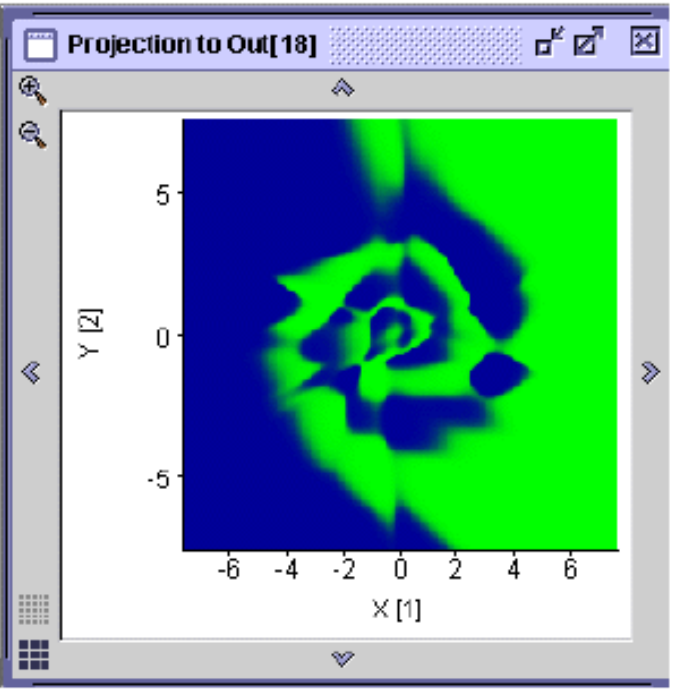




Source: [2]

Being creative about Linear Separability



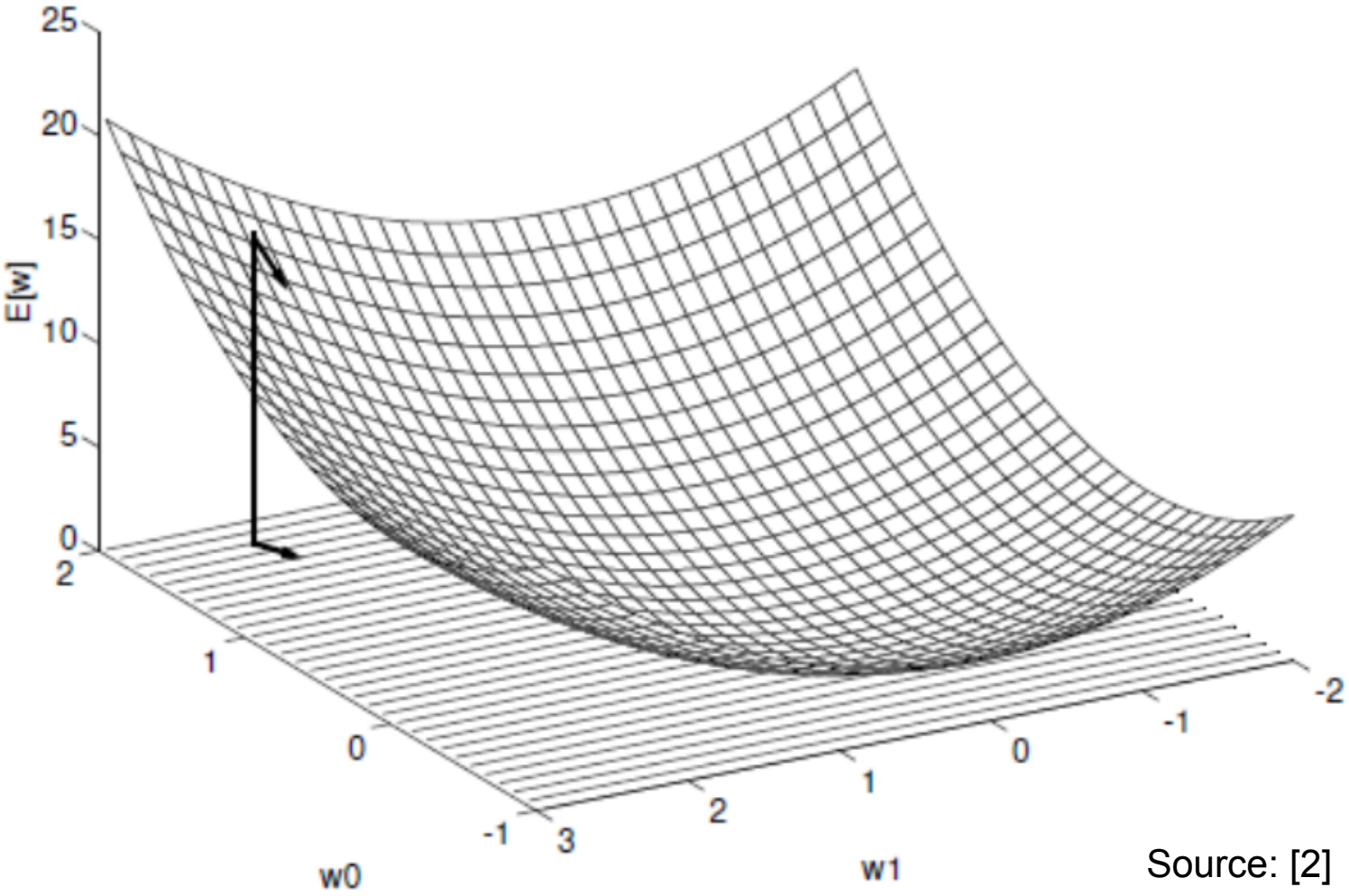


Source: [2]

Overview

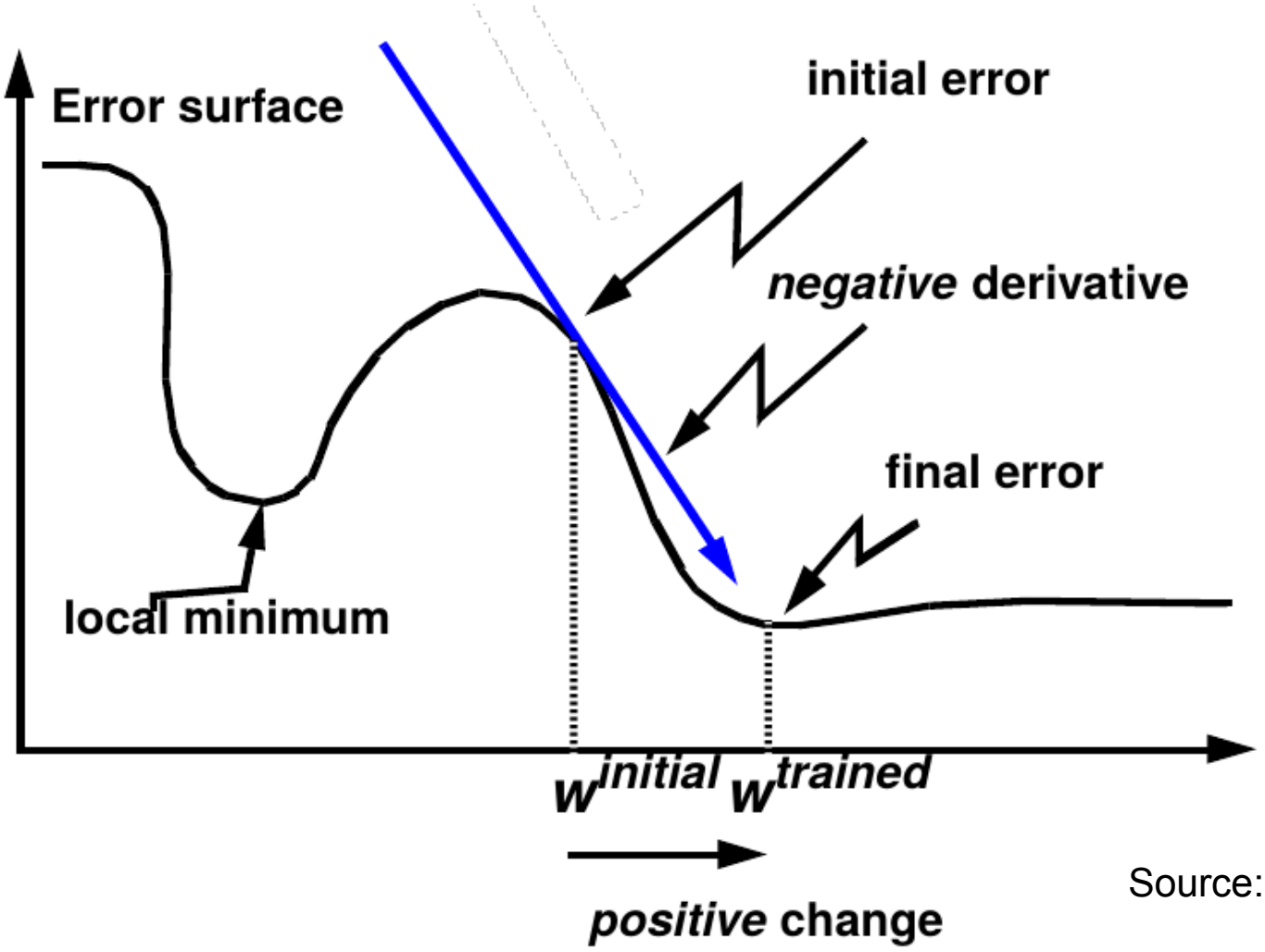
1. What are neural networks?
2. A single perceptron
3. Perceptron training
4. A simple network
5. **Learning via backpropagation**
6. Convolutional neural networks
7. 2nd talk: Visualization

Gradient Descent



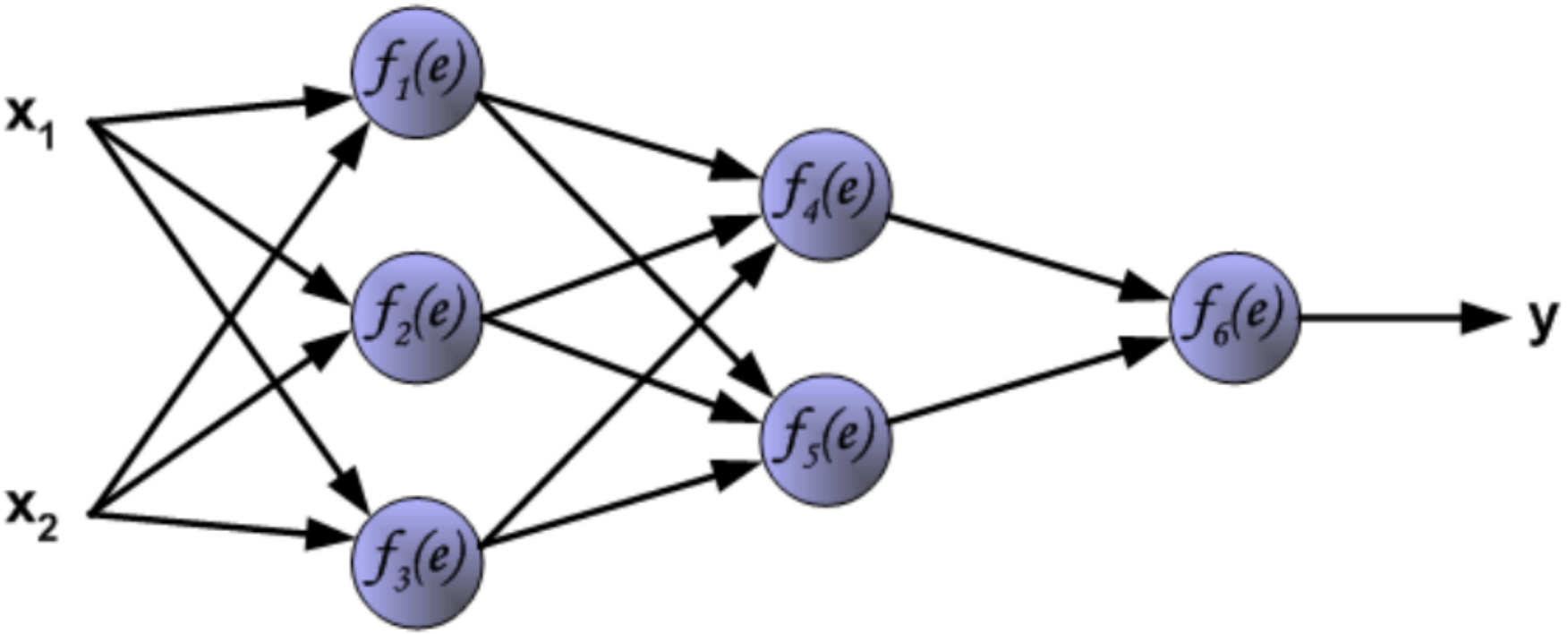
Source: [2]

Gradient Descent



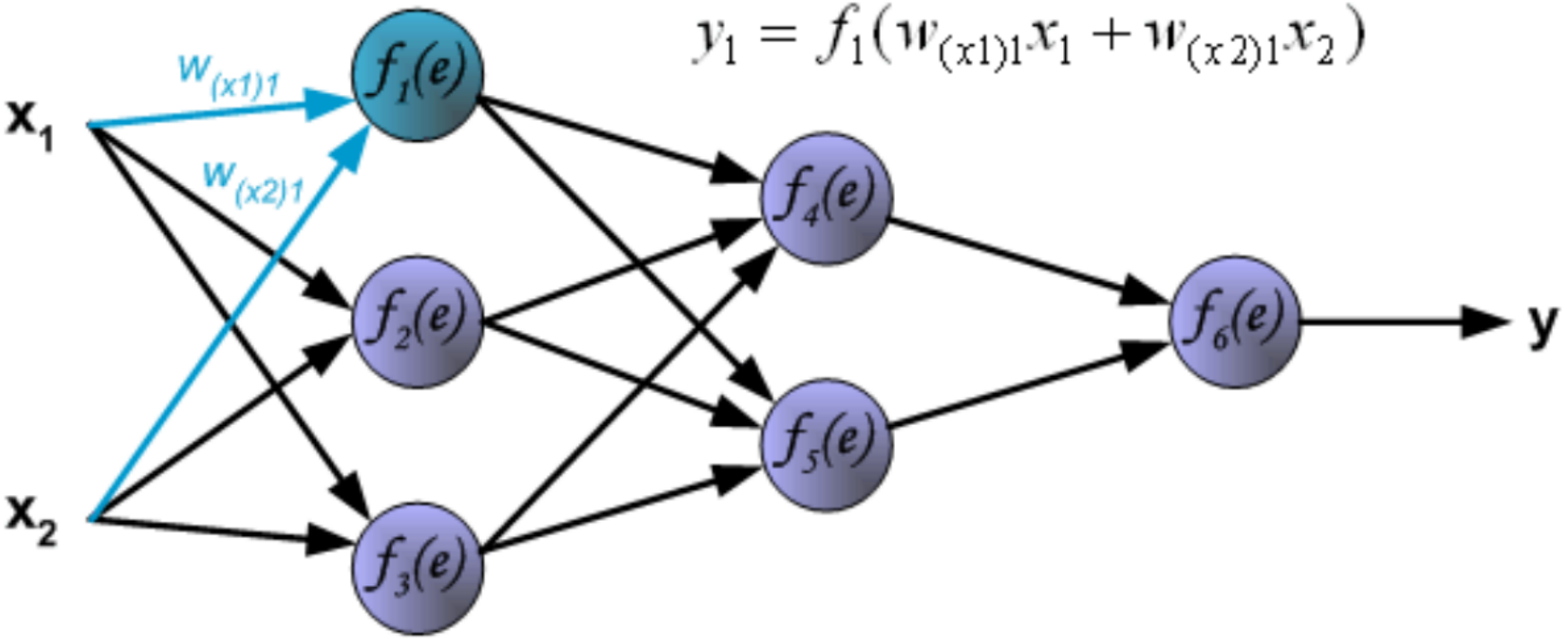
Source: [2]

Training using Backpropagation



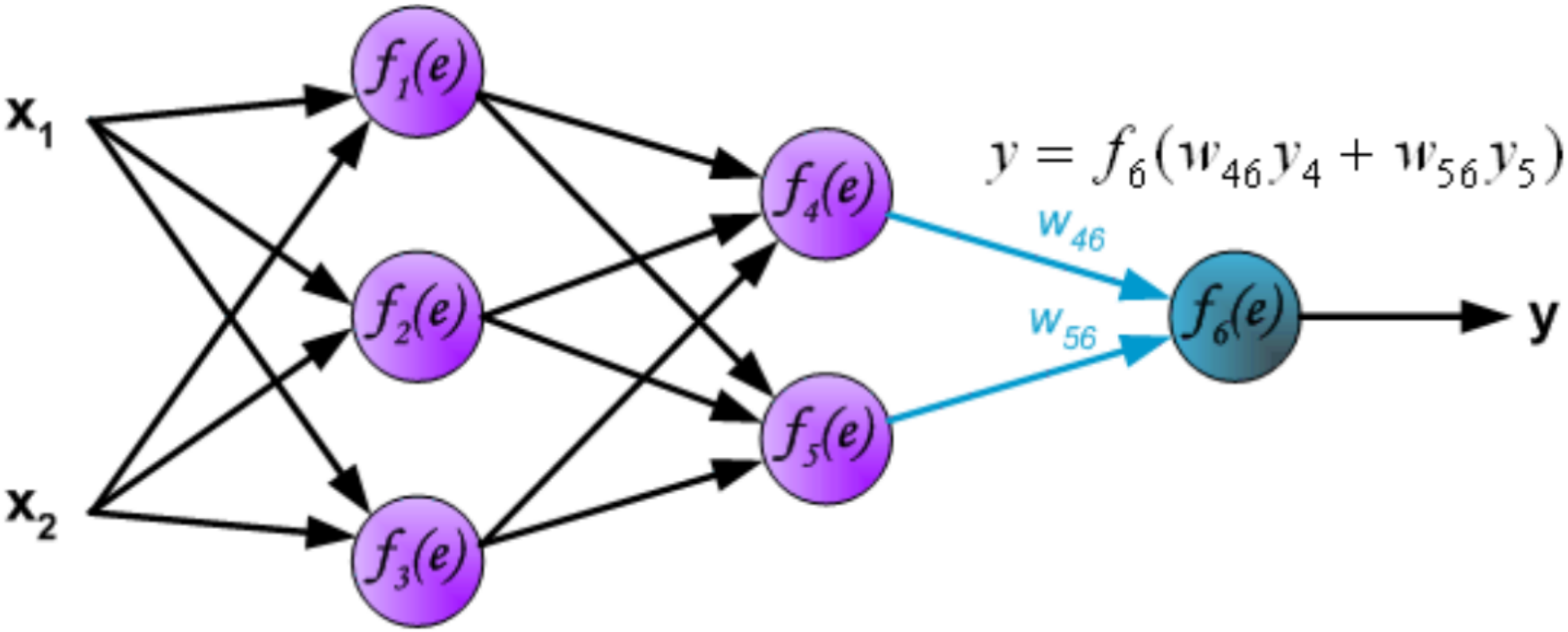
Source: [3]

Training using Backpropagation



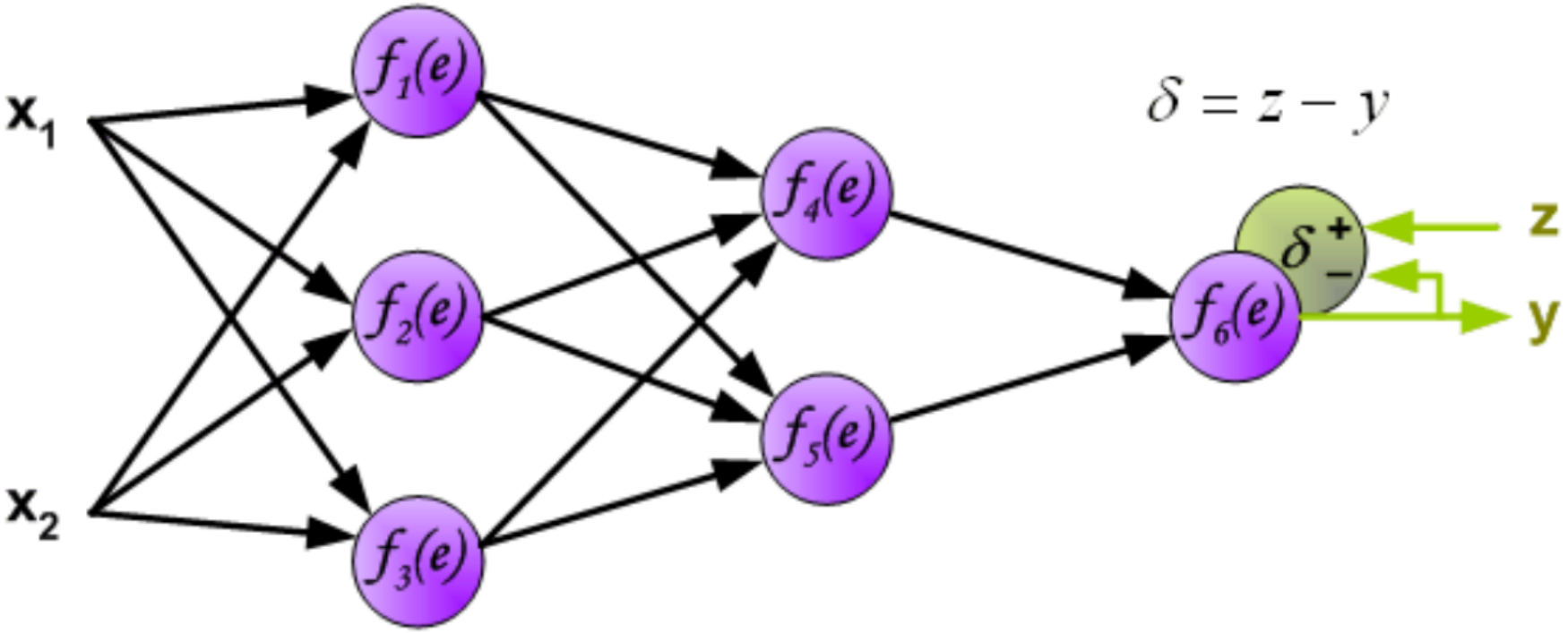
Source: [3]

Training using Backpropagation



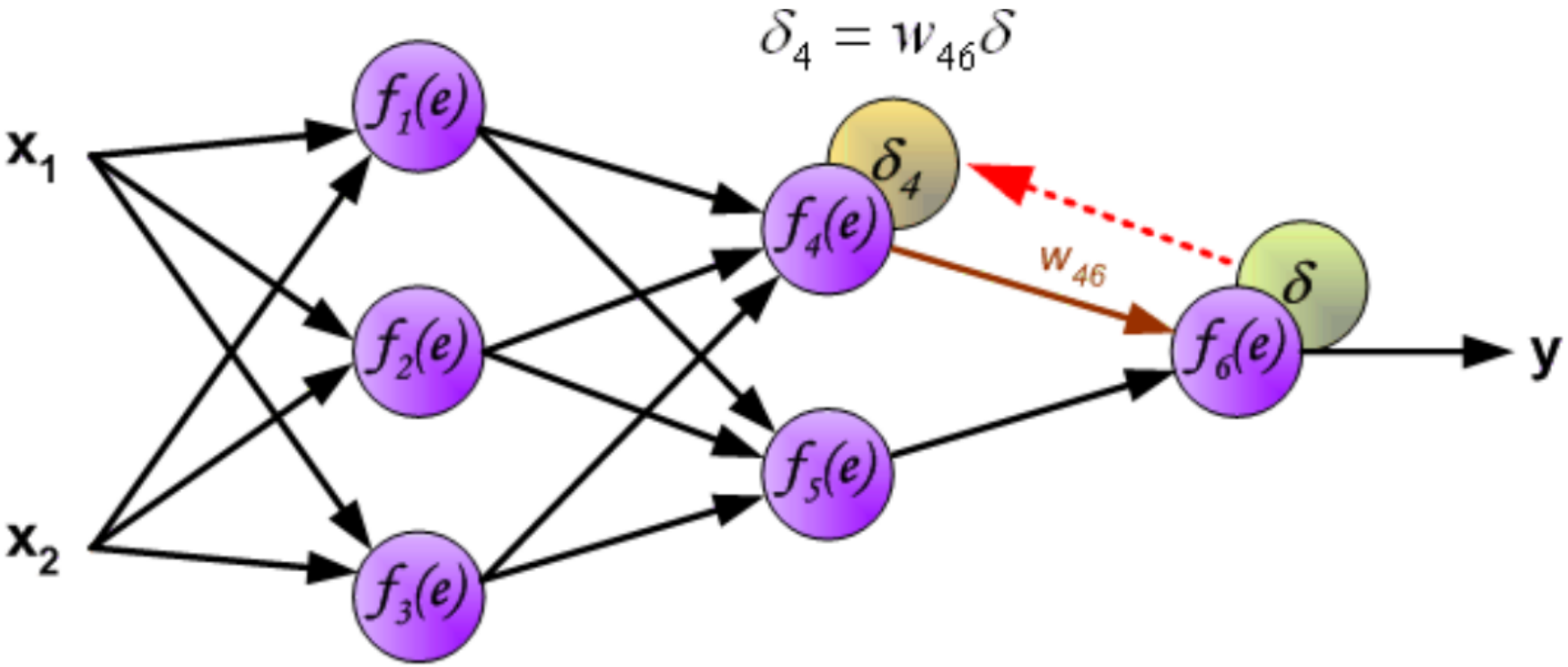
Source: [3]

Training using Backpropagation



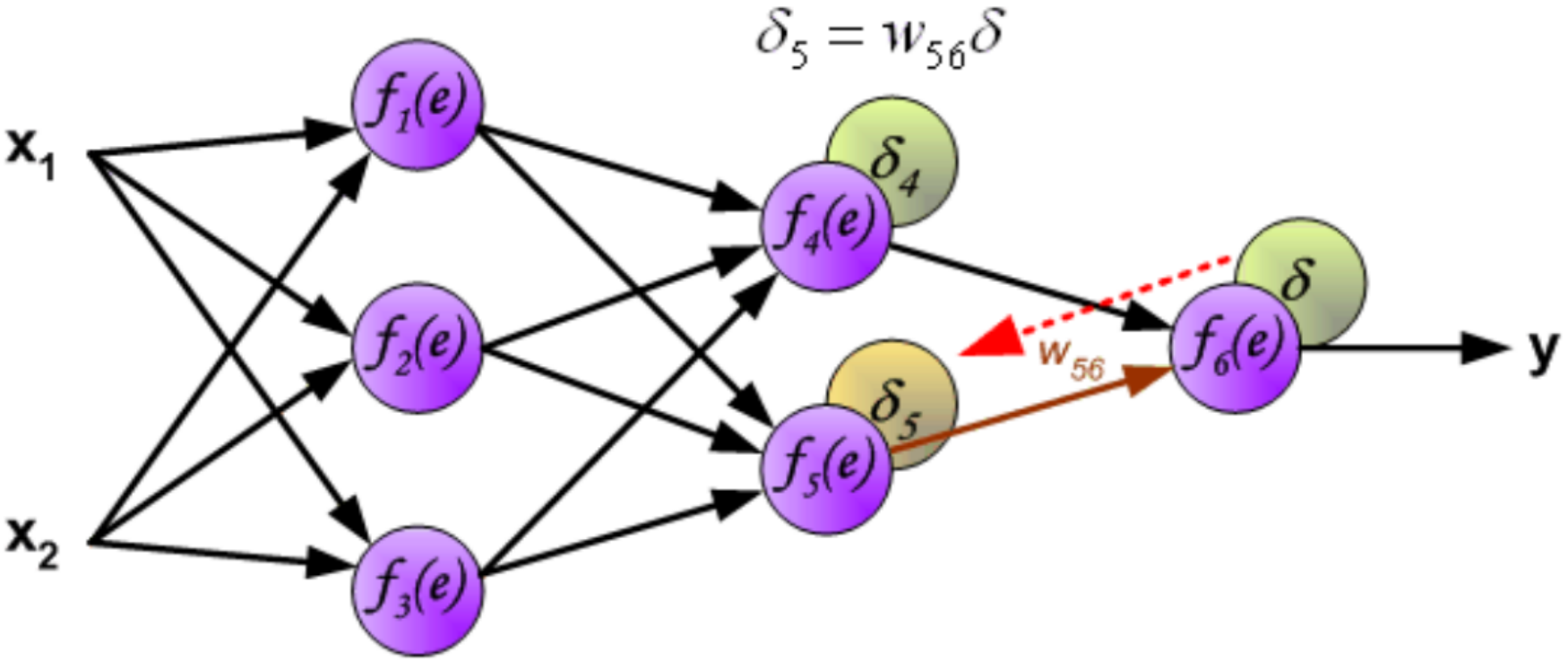
Source: [3]

Training using Backpropagation



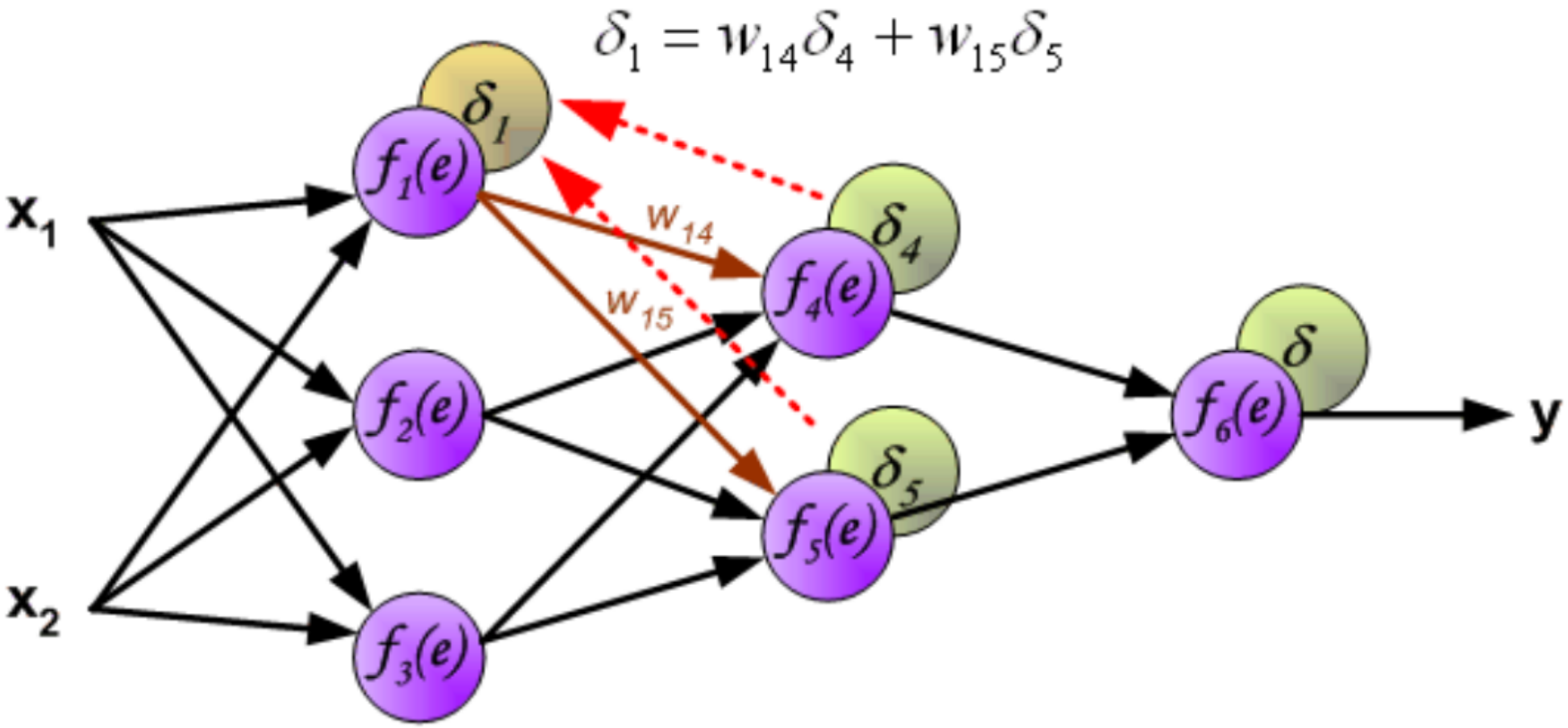
Source: [3]

Training using Backpropagation



Source: [3]

Training using Backpropagation

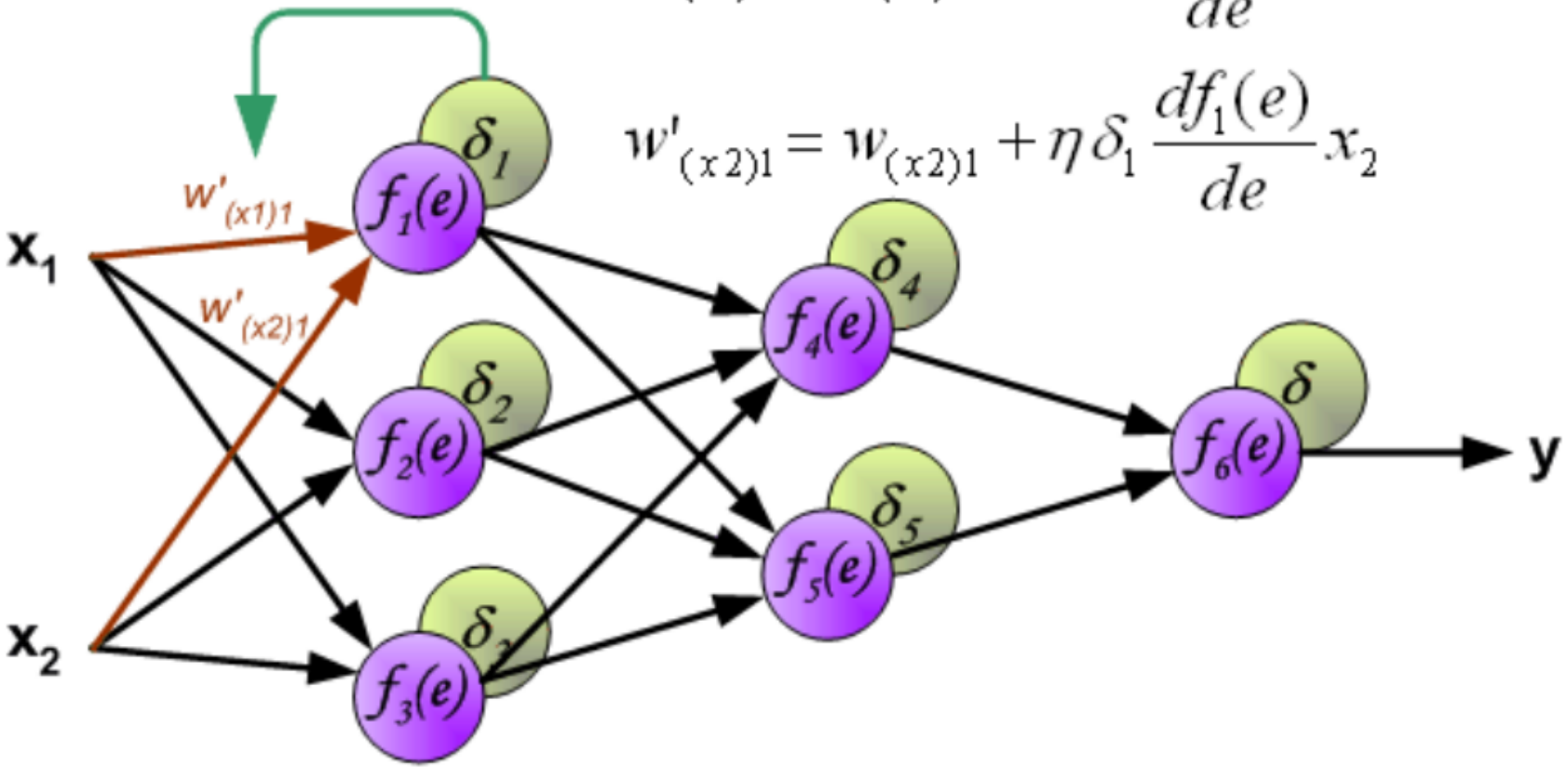


Source: [3]

Training using Backpropagation

$$w'_{(x1)1} = w_{(x1)1} + \eta \delta_1 \frac{df_1(e)}{de} x_1$$

$$w'_{(x2)1} = w_{(x2)1} + \eta \delta_1 \frac{df_1(e)}{de} x_2$$



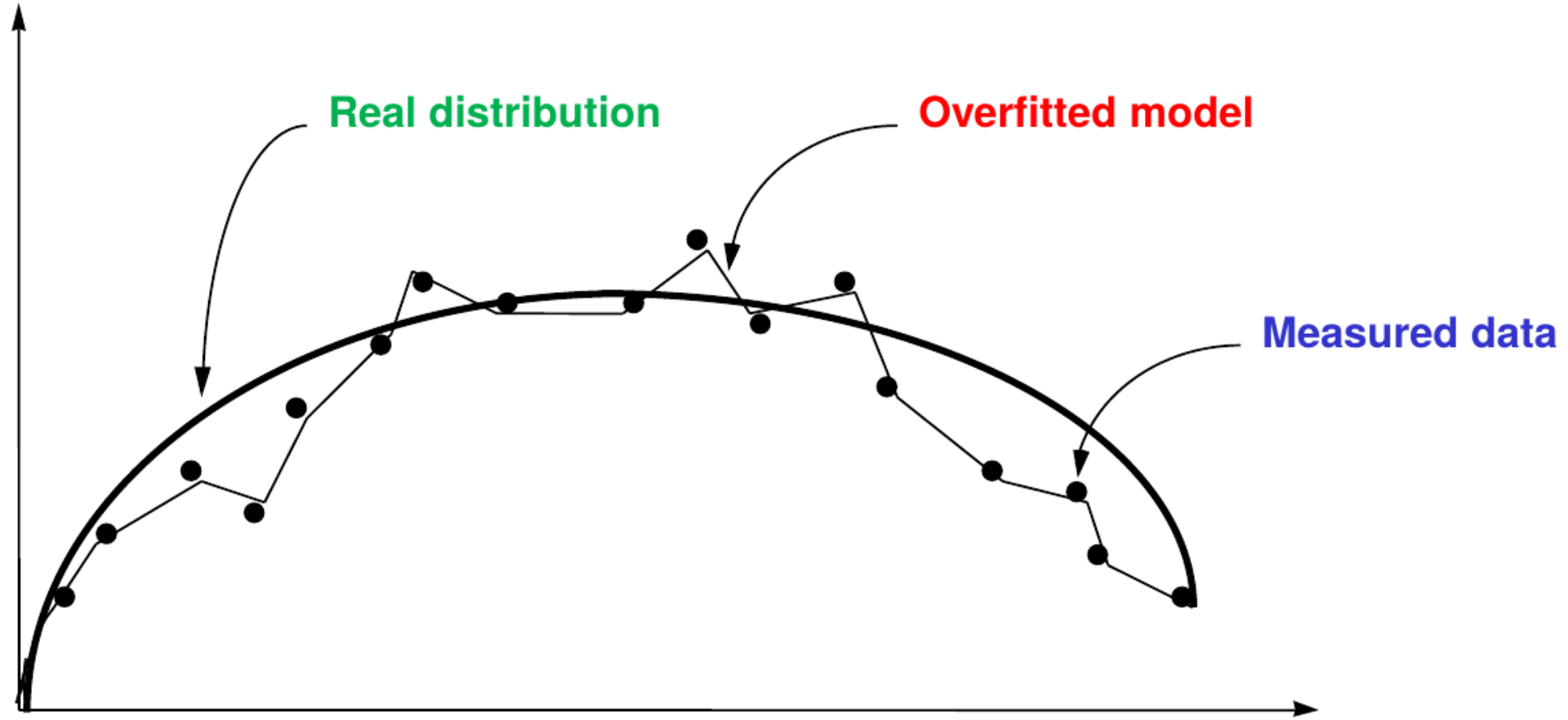
Source: [3]

Using Matrices

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,R} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,R} \\ \\ w_{S,1} & w_{S,2} & \cdots & w_{S,R} \end{bmatrix}$$

Source: [6]

Problem: Overfitting

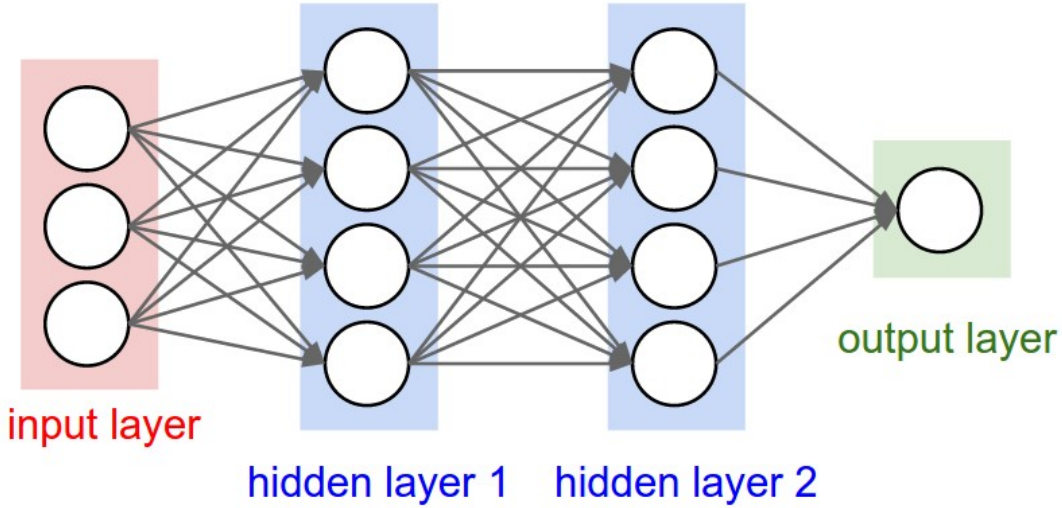


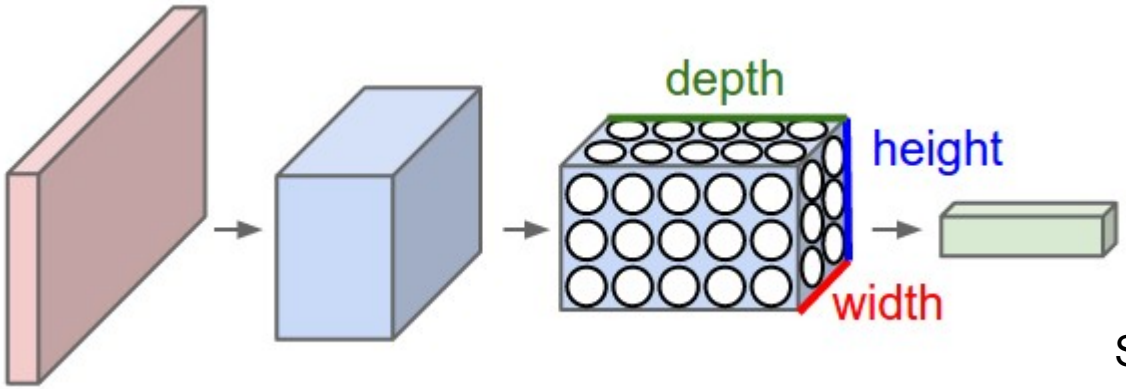
Source: [3]

Overview

1. What are neural networks?
2. A single perceptron
3. Perceptron training
4. A simple network
5. Learning via backpropagation
6. **Convolutional neural networks**
7. 2nd talk: Visualization

Convolutional Neural Networks (CNN)

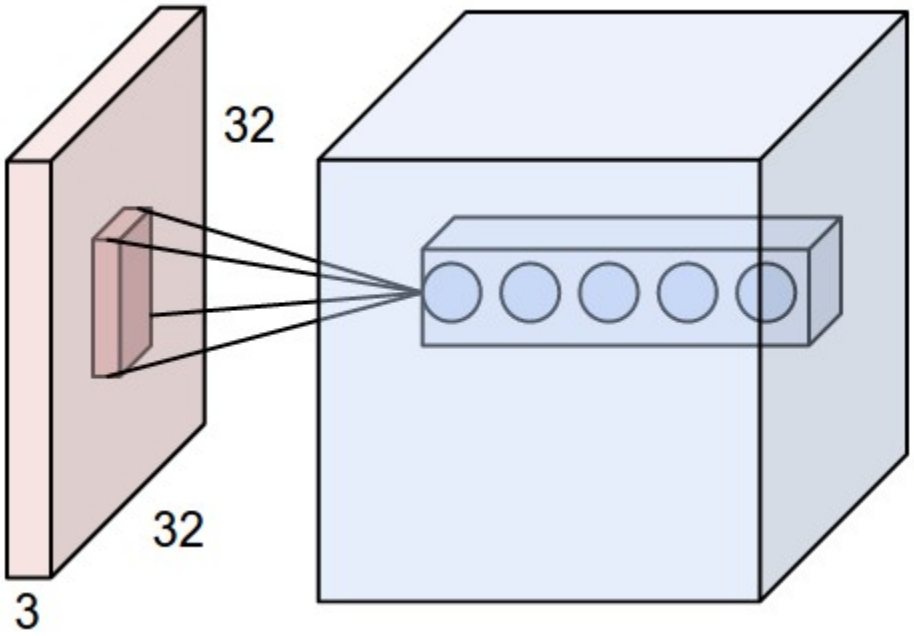




Source: [4]

Convolutional Neural Networks (CNN)

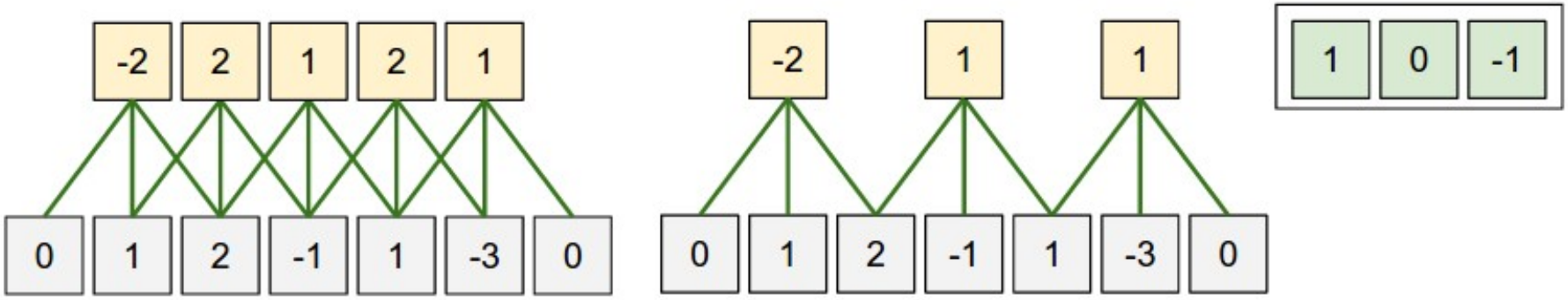
Complexity Reduction



Source: [4]

Convolutional Neural Networks (CNN)

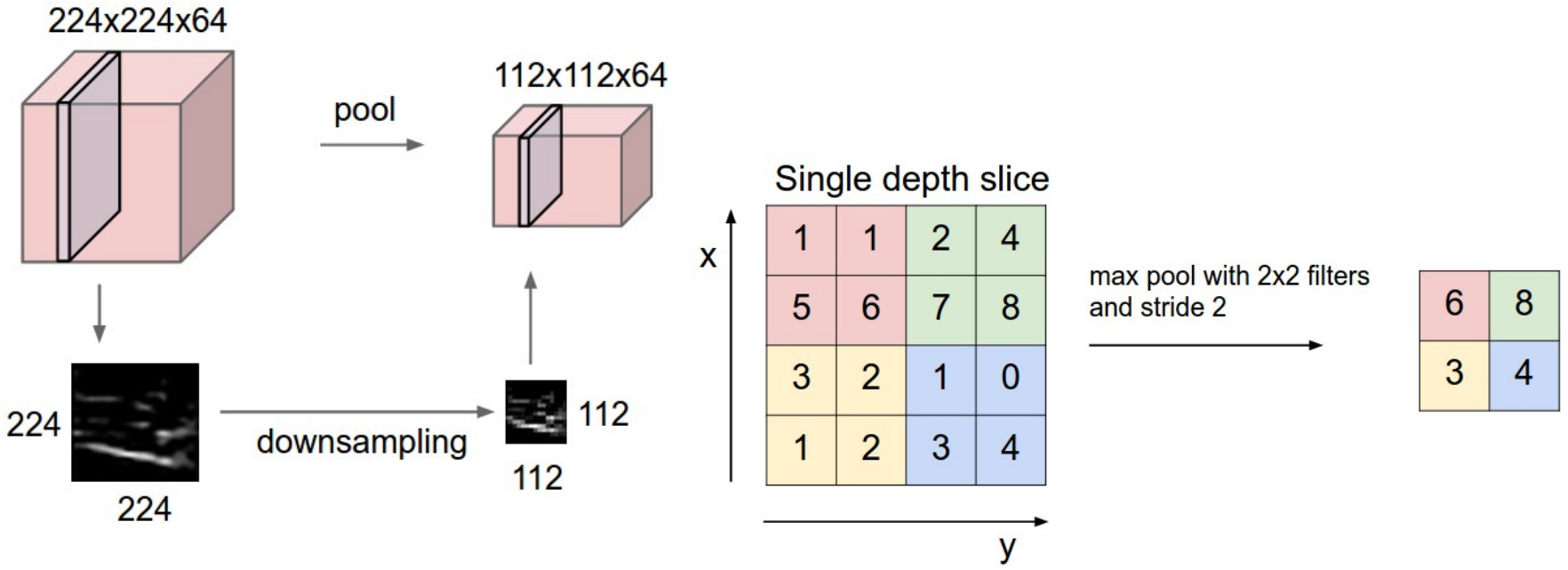
Complexity Reduction



Source: [4]

Convolutional Neural Networks (CNN)

Max-Pooling



Source: [4]

Convolutional Neural Networks (CNN)

Architecture examples

- LeNet (1990's)
- AlexNet (2012)
- ZF Net (2013)
- GoogLeNet (2014)
- VGGNet (2014)

Source: [4]

Overview

1. What are neural networks?
2. A single perceptron
3. Perceptron training
4. A simple network
5. Learning via backpropagation
6. Convolutional neural networks
7. **2nd talk: Visualization**

Why visualize Neural Networks?

- because they are too difficult to understand otherwise

Thanks!

Quellen

[1] Bogdan M. Wilamowski. „Neural Network Architectures and Learning“. In: Conference: Industrial Technology, 2003 IEEE International Conference on, Volume: 1

[2] „Vertiefung: Neural Networks for Secondary Structure Prediction“. URL: http://medicalbioinformatics.de/downloads/lectures/Algorithmische_BioInformatik/WS13-14/algbioinf_ws13-14_woche9_2.pdf.

[3] „Vertiefung: Neural Networks for Secondary Structure Prediction, Prediction Methods for Special Secondary Structures“. URL: http://medicalbioinformatics.de/downloads/lectures/Algorithmische_BioInformatik/WS13-14/algbioinf_ws13-14_woche10_1.pdf

[4] „Convolutional Neural Networks (CNNs / ConvNets)“. URL: http://cs231n.github.io/convolutional-networks/

[5] „Chapter 15: Visual Processing: Cortical Pathways“. URL: http://neuroscience.uth.tmc.edu/s2/chapter15.html.

[6] „Neural Network Architectures“. URL: http://de.mathworks.com/help/nnet/ug/neural-network-architectures.html

(Alle Weblinks zuletzt abgerufen am 18.11.2015 um 14:27 Uhr)