Neural Networks

Foundations of Data Analysis

04/10/2023

Topic Outline

Basics of neural network theory and practice:

- supervised and unsupervised learning.

Learning

- Supervised Learning
  - Recognizing hand-written digits, pattern recognition, regression.
  - Labeled examples
    (input , desired output)
  - Neural Network models: perceptron, support vector machine.
- Unsupervised Learning
  - Find similar groups of documents in the web, content addressable memory, clustering.
  - Unlabeled examples
    (different realizations of the input alone)
  - Neural Network models: self organizing maps

Topic Outline

Most popular Neural Network models:

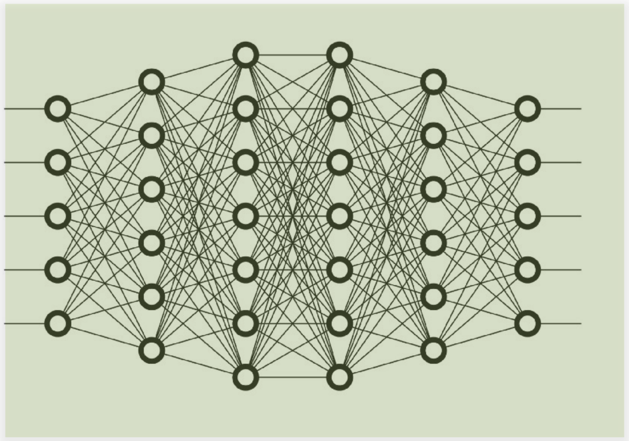
- architectures
- learning algorithms
- optimization

Neural Networks

- A NN is a machine learning approach inspired by the way in which the brain performs a particular learning task:

Neural Networks

Modeling



Human Intelligence

Brain Neural Network

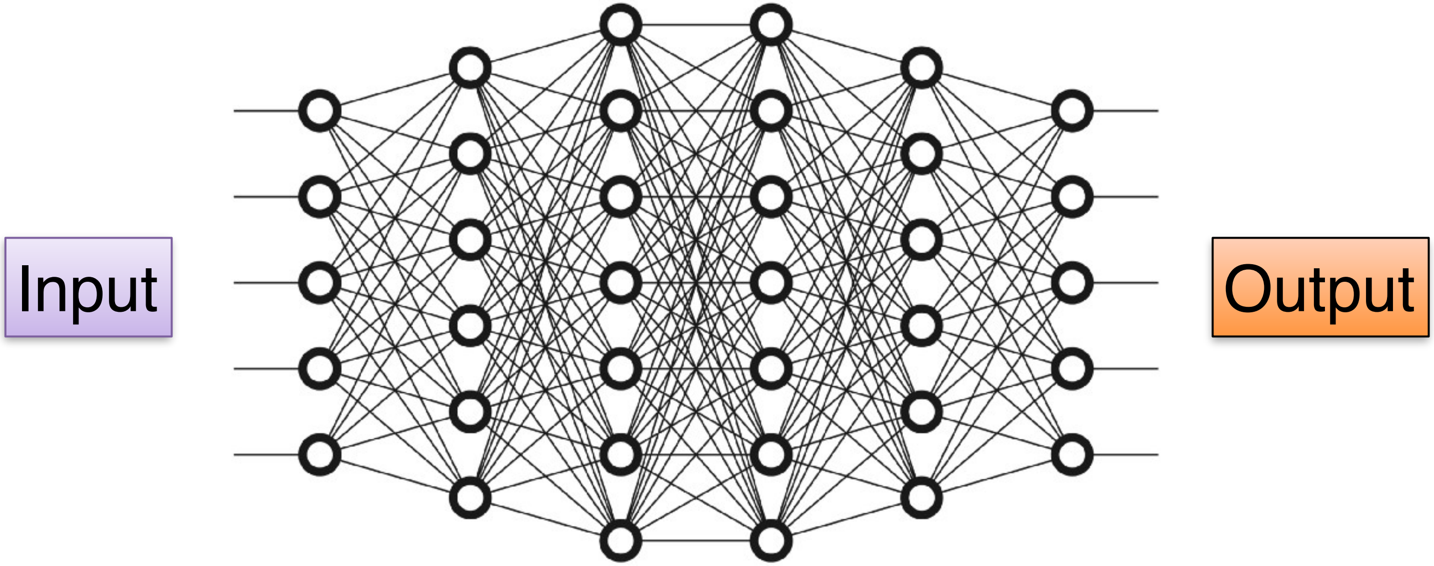
Human Learning

Machine/Artificial Intelligence

Deep Neural Network

Machine Learning

Neural Networks



Input

Output

- Inter neuron connection strengths (weights) are used to store the acquired information (the training examples).

- During the learning process the weights are modified in order to correctly model the particular learning task on the training examples.

Network architectures

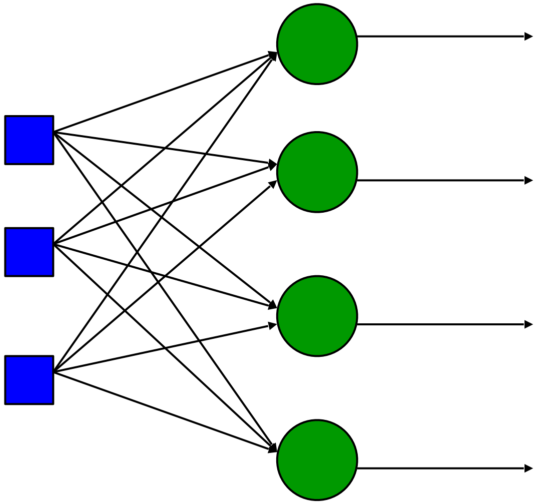
- Three different classes of network architectures
  - single-layer feed-forward
  - multi-layer feed-forward
  - recurrent

  neurons are organized in acyclic layers

- The architecture of a neural network is linked with the learning algorithm used to train

Single Layer Feed-forward

Input nodes layer

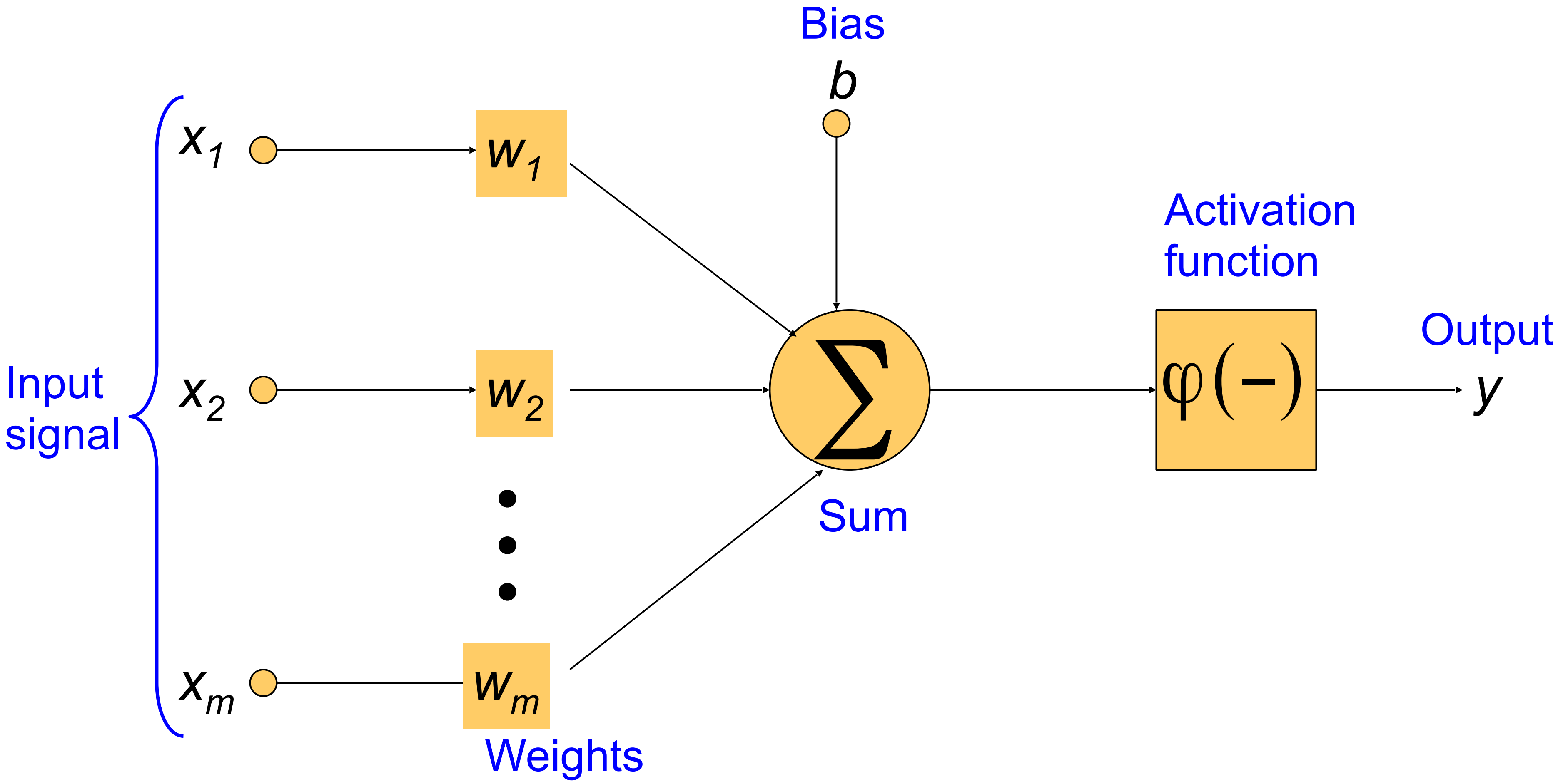


Output nodes layer

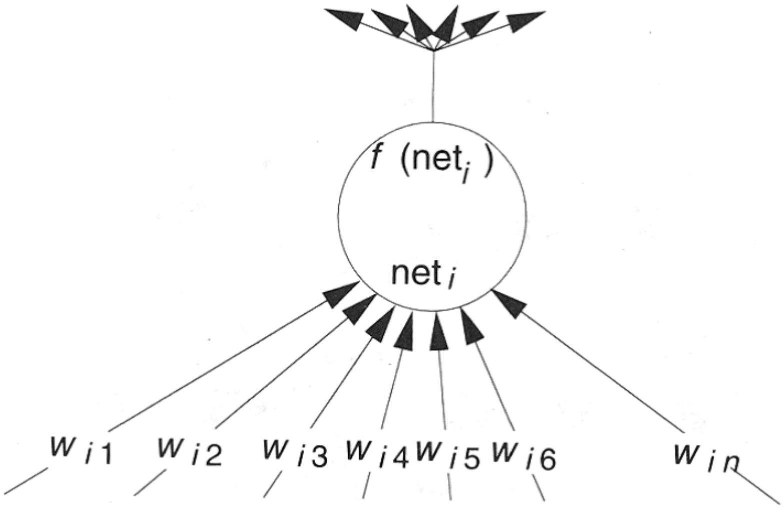
The Neuron

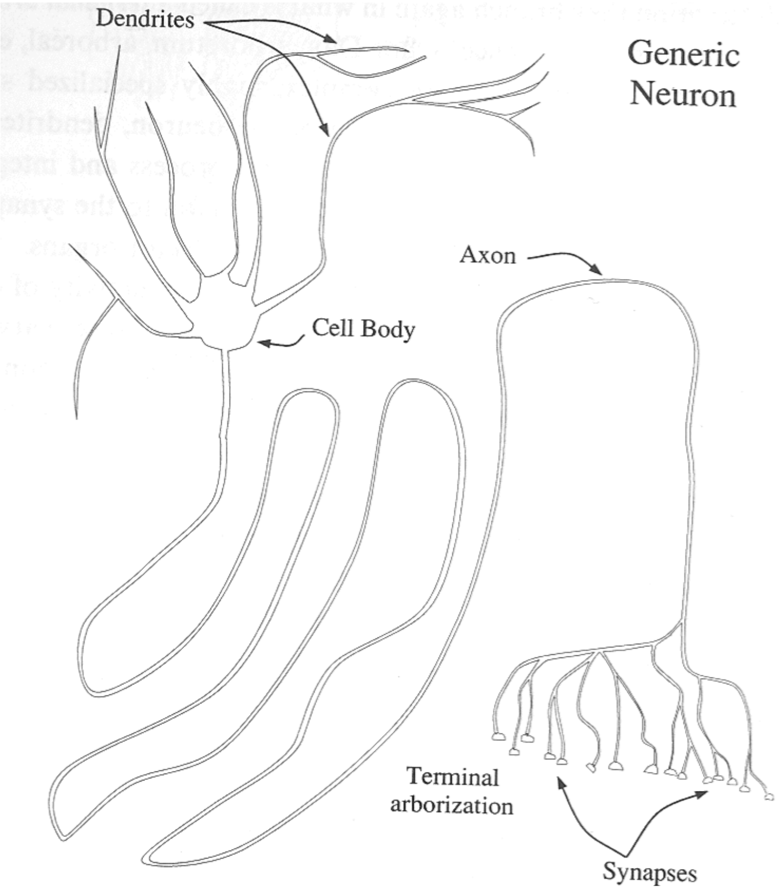
- The neuron is the basic information processing unit of a NN. It consists of :
  - A set of connecting links, each link characterized by a weight: W_1, W_2, …, W_m
  - An add function (linear combiner) computes the weighted sum of the inputs: $u = \sum_{j=1}^{m} w_j x_j$
  - Activation function for limiting the amplitude of the output of the neuron. $y = \varphi(u + b)$

The Neuron


Bias
b
x_1
w_1
Activation
function
Output
Input
signal
x_2
w_2
$\sum$
$\varphi(-)$
y
Sum
x_m
w_m
Weights

Perceptron as a Single Layer Neuron





History of Perceptron

- Frank Rosenblatt
- 1928-1969

invented perceptron algorithm

History of Perceptron

- Marvin Minsky
- 1927-2016

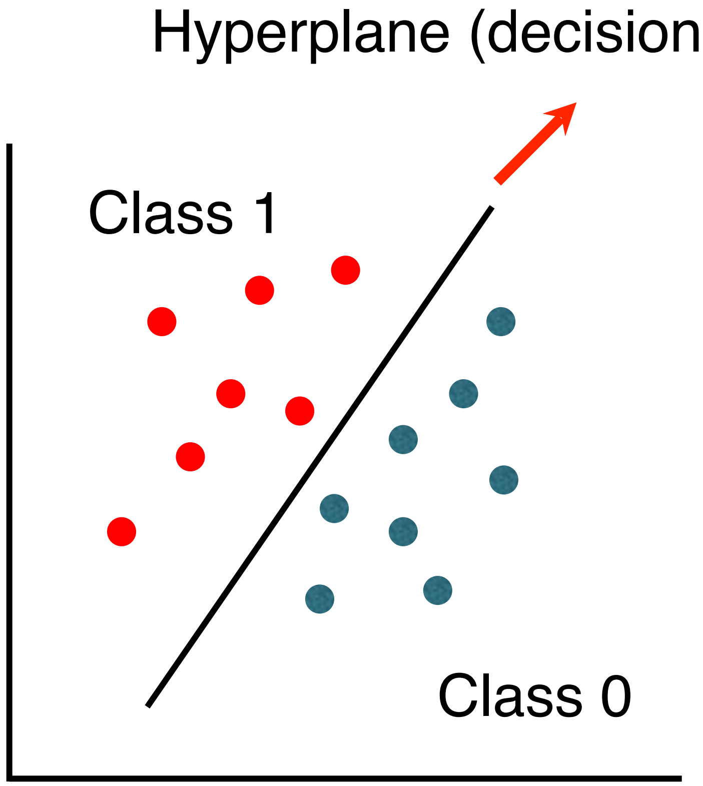
stated that a simple perceptron is limited to linearly separable problems

Perceptron Learning Algorithm

- First neural network learning model in the 1960’s
- Simple and limited (single layer model)

What is Perceptron?

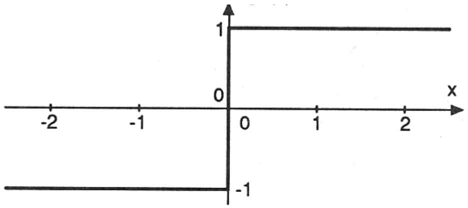
The goal of perceptron algorithm is to find a hyperplane that separates a set of data into two classes.



- Binary classifier
- Supervised learning

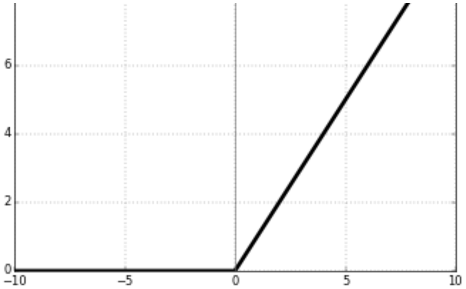
Activation Function

Outputs the label given an input or a set of inputs.



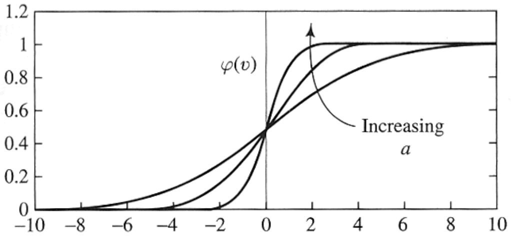
Step function

$$f(x) := \mathrm{sgn}(x) = \begin{cases} 1 & if \quad x \geq 0 \\ -1 & if \quad x < 0 \end{cases}$$



ReLU (rectified linear unit)

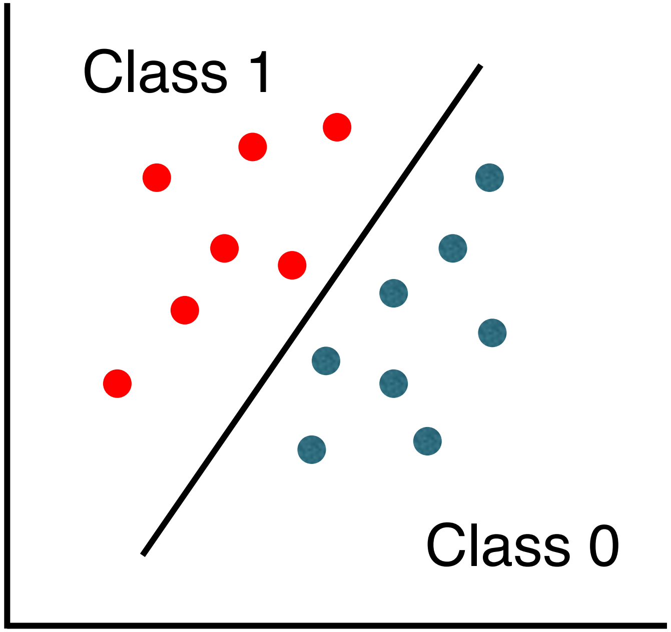
$$f(x) = max(0, x)$$



Sigmoid function

$$f(x) := \sigma(x) = \frac{1}{1 + e^{(-ax)}}$$

Perceptron

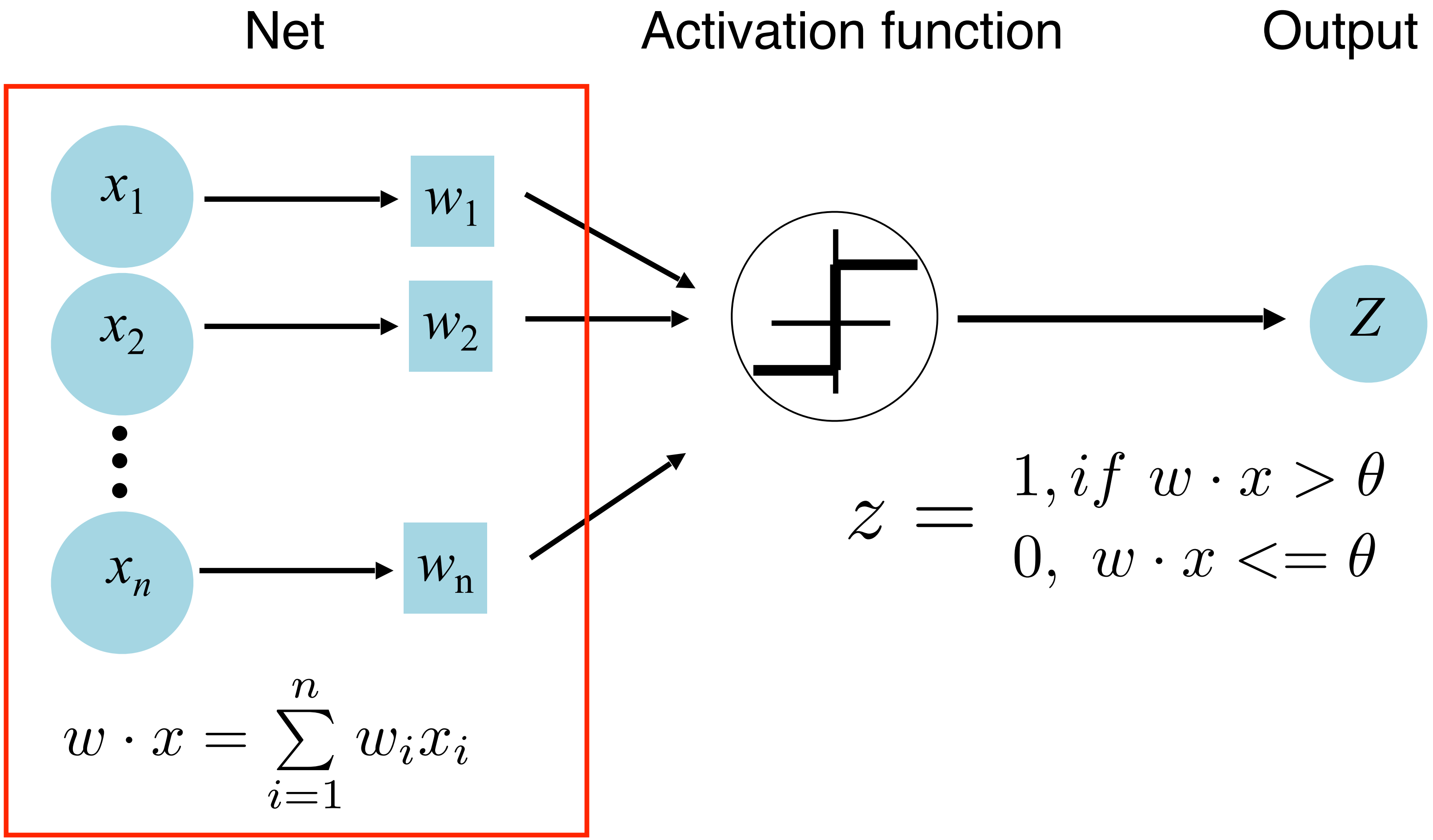


bias term

$$f(x) = \begin{array}{l} 1, if\ \ w \cdot x + \theta > 0 \\ 0, otherwise \end{array}$$

$$w \cdot x = \sum_{i=1}^{n} w_i x_i$$ (dot product)

Perceptron

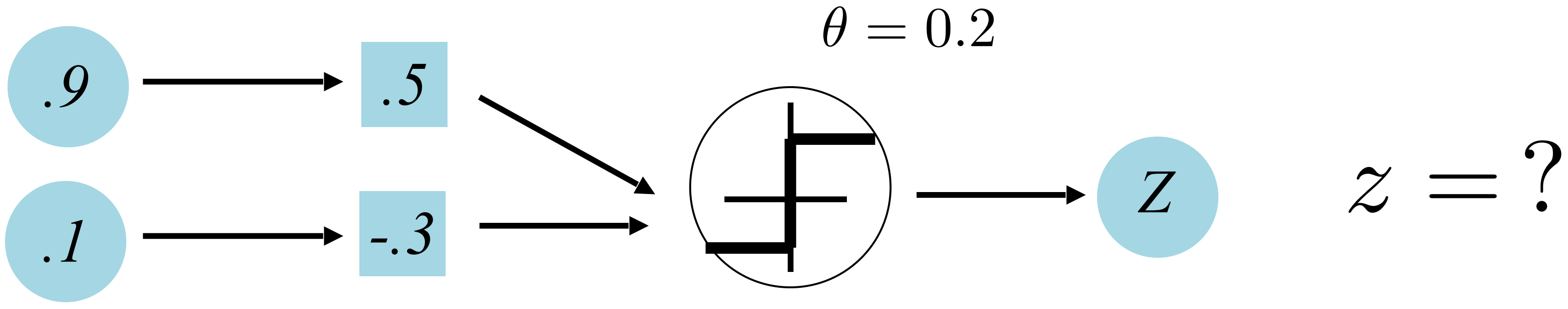


$$w \cdot x = \sum_{i=1}^{n} w_i x_i$$

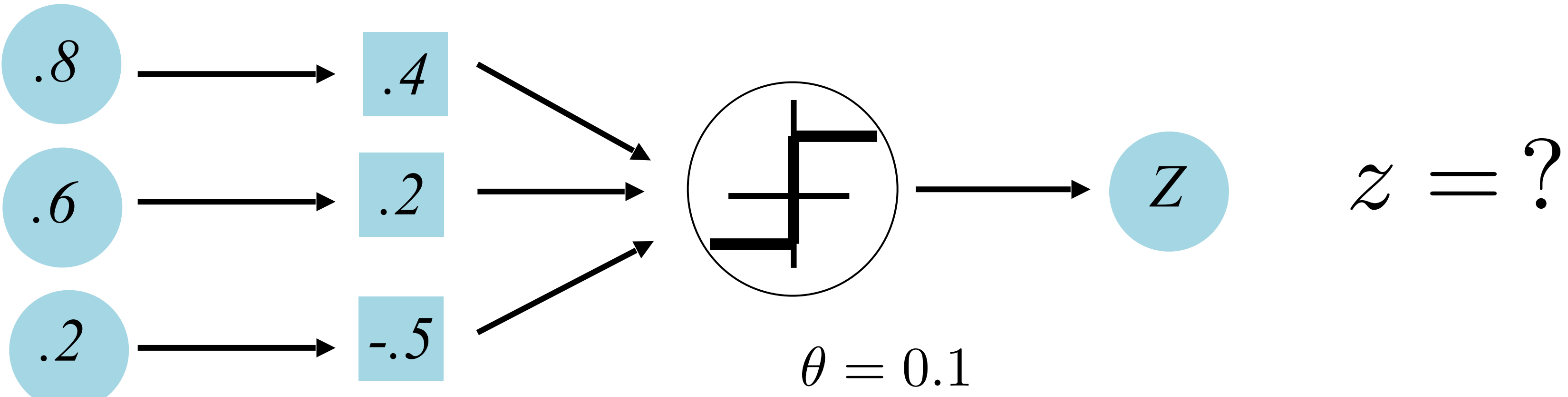
$$z = \begin{array}{l} 1, if \;\; w \cdot x > \theta \\ 0, \;\; w \cdot x <= \theta \end{array}$$

- Learning weights such that an objective function is maximized

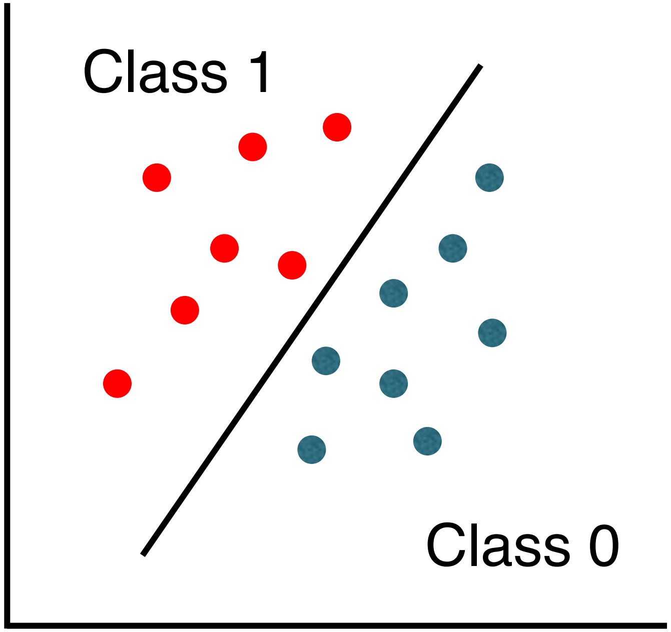
Examples



$$z = \begin{cases} 1, if \;\; w \cdot x > \theta \\ 0, \;\; w \cdot x <= \theta \end{cases}$$



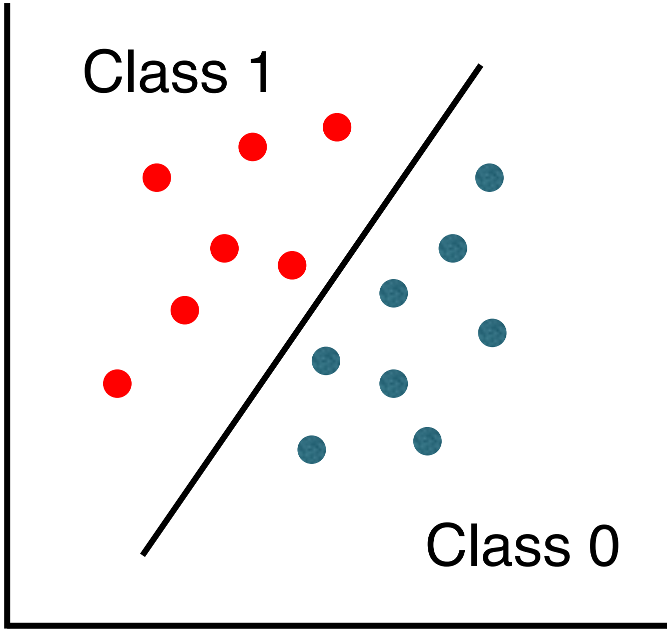
How to Learn Perceptron?



$$f(x) = \begin{cases} 1, if\ \ w \cdot x + \theta > 0 \\ 0, otherwise \end{cases}$$

w, θ are unknown parameters

How to Learn Perceptron?

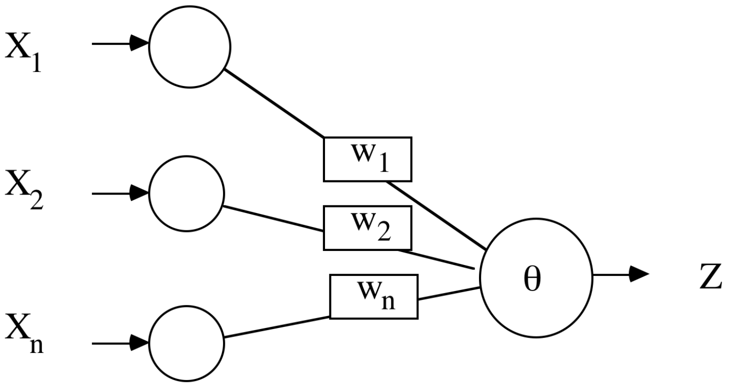


$$f(x) = \begin{cases} 1, if\ w \cdot x + \theta > 0 \\ 0, otherwise \end{cases}$$

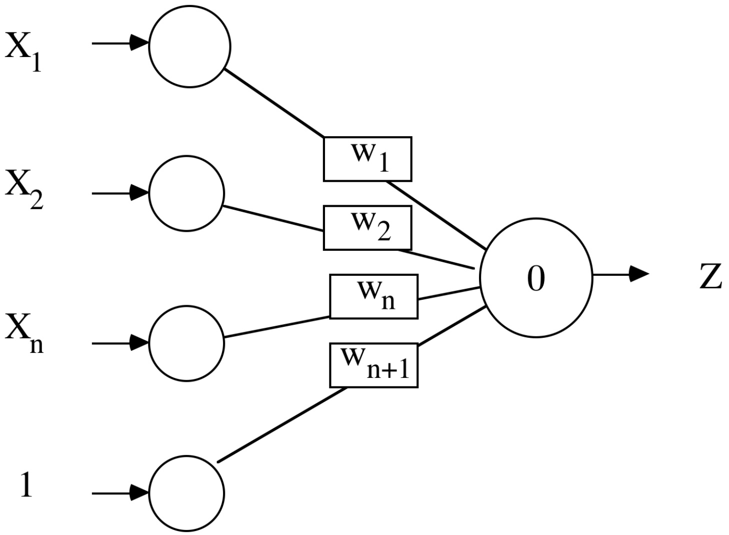
w, θ are unknown parameters

- In supervised learning the network has its output compared with known correct answers
  - Supervised learning
  - Learning with a teacher

Weight Versus Threshold



Do you need to adjust Theta? Yes, in most cases



where $w_{n+1} = -\theta$

Perceptron Learning Rules

- Consider linearly separable problems
- How to find appropriate weights
- Look if the output result *o* belongs to the desired class has the desired value *d* (give labels). For example, a loss function *L* as a sum-of-squared differences between the output and desired value.

$$w^{new} = w^{old} - \eta \nabla_w L, \nabla_w L = \sum_i (o - d)x_i$$

η is called the learning rate, with $0 < \eta \leq 1$

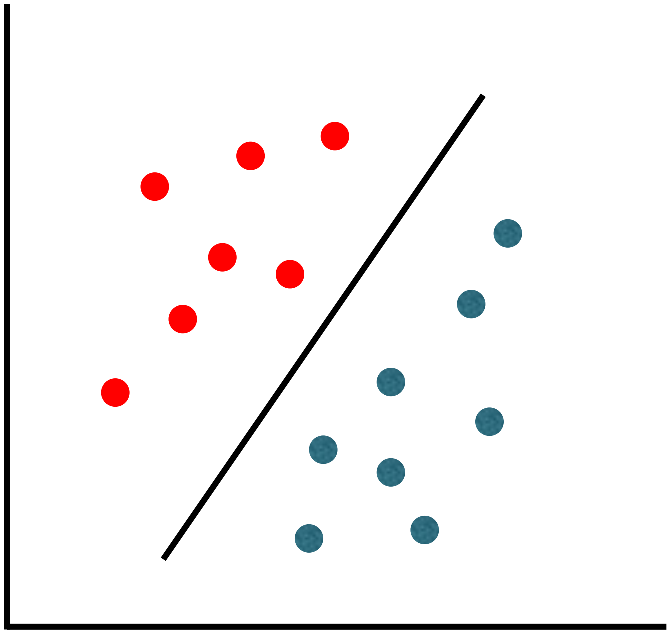
Perceptron Convergence Theorem: Guaranteed to find a solution in finite time if a solution exists

Perceptron Learning Rules

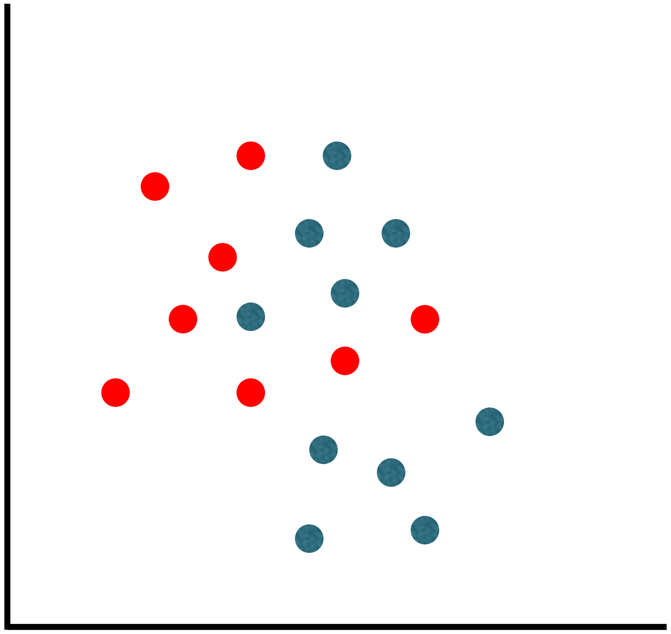
- The algorithm converges to the correct classification
    - if the training data is linearly separable

- When assigning a value to η we must keep in mind two conflicting requirements
    - Averaging of past inputs to provide stable weights estimates, which requires small η
    - Fast adaptation with respect to real changes in the underlying distribution of the process responsible for the generation of the input vector ***x***, which requires large η

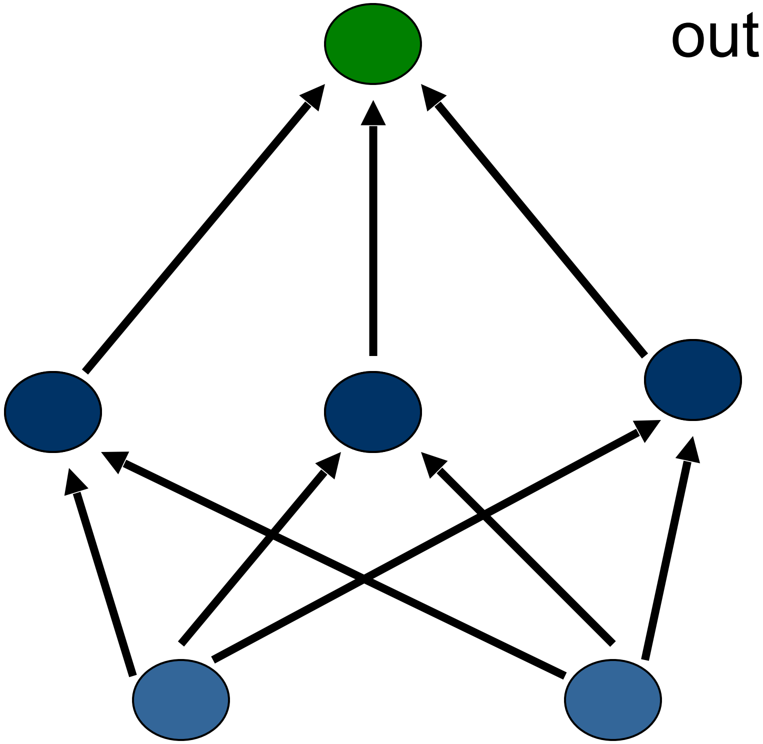
Linear Separability



Limited Functionality of Hyperplane



Multilayer Network



output layer

hidden layer

input layer

$$o_1 = \mathrm{sgn}(\sum_{i=0}^{n} w_{1i} x_i)$$

$$o_2 = \mathrm{sgn}(\sum_{i=0}^{n} w_{2i} x_i)$$