Chapter 1
Perceptrons

Minlie Huang
aihuang@tsinghua.edu.cn
Dept. of Computer Science and Technology
Tsinghua University
http://coai.cs.tsinghua.edu.cn/hml/

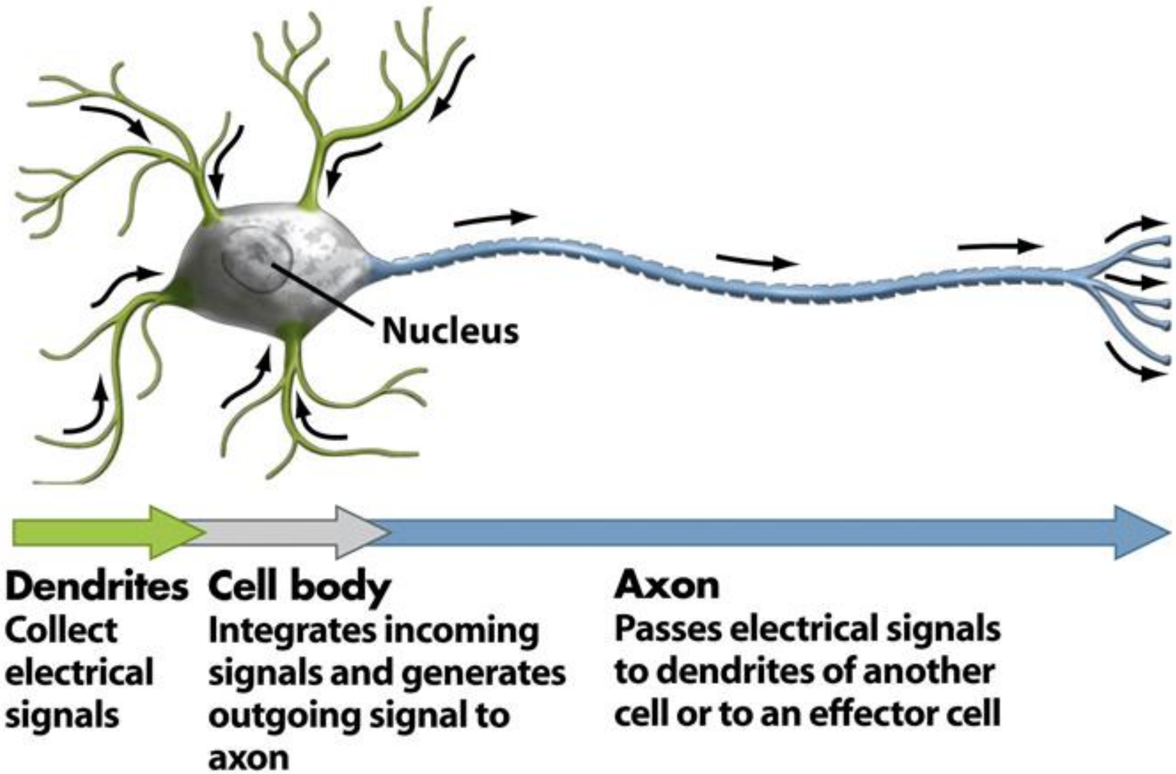


Figure 45-2b Biological Science, 2/e

Introduction

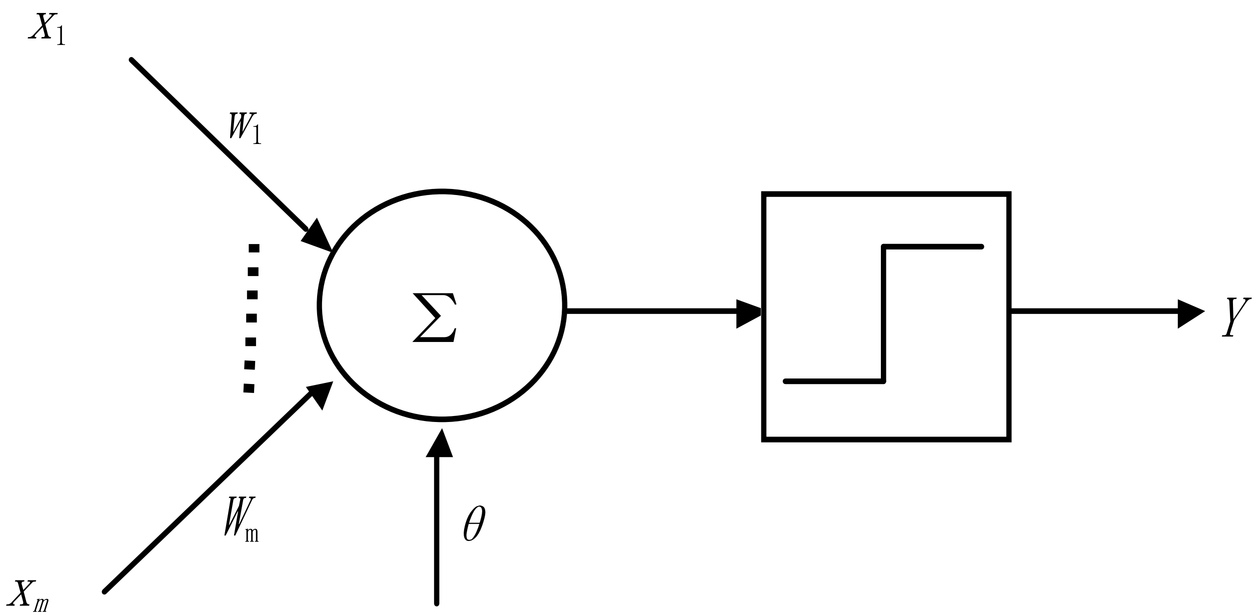
- Proposed by McCulloch and Pitts, and Hebb, Rosenblatt in 1957
- The simplest form of a neural network used for linearly separable problems
- ***Perceptron convergence theorem***
- One neuron for two-class problems, multiple neurons for multi-class problems (multi-layer perceptron)
- Theoretically, multilayer perceptron can be used to solve any classification and regression problem with BP learning algorithm

Overview

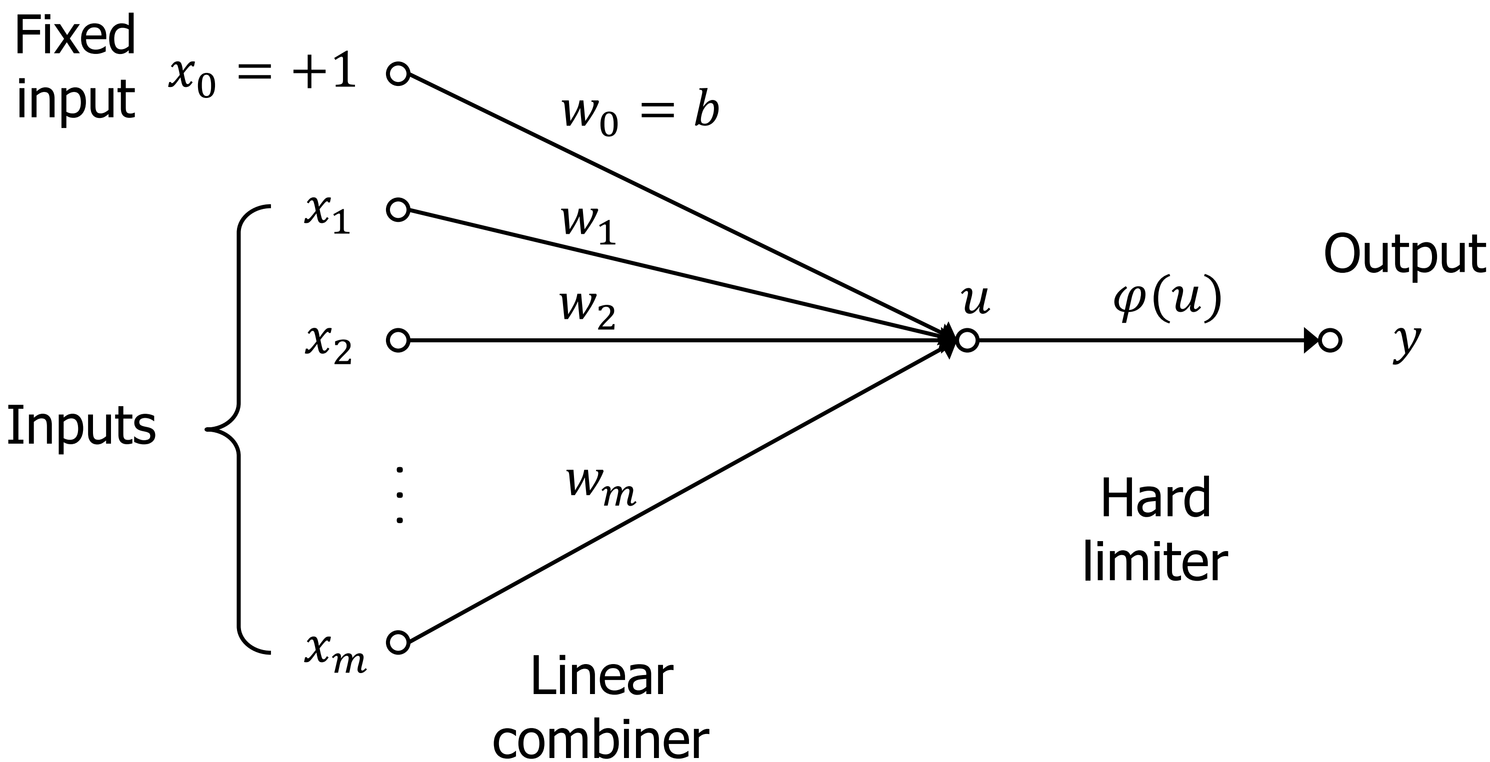
- Single layer Perceptron
- Multilayer Perceptron
  - BP Learning algorithm
  - Others

Simple layer Perceptron

Perceptron Unit



Single Layer



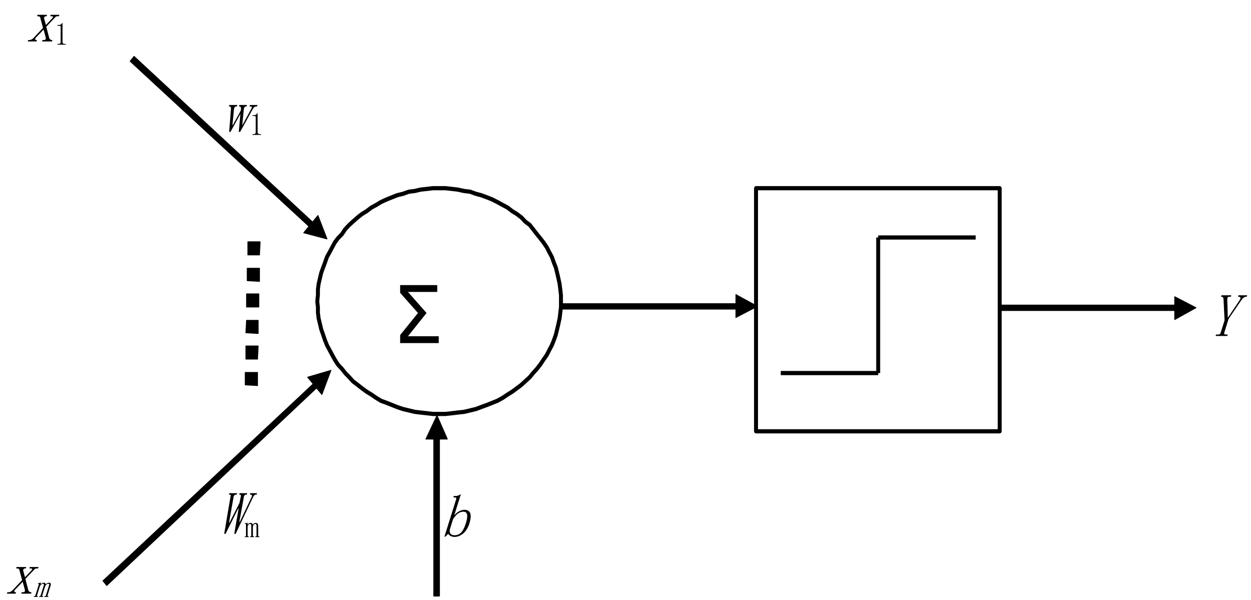
A model of neuron

- Inputs x_i, the m elements of $x(i)$ originate at different points in input space.
- Synaptic weights w_i,
- Weighted sum on inputs

$$u = w_0 x_0 + \ w_2 x_2 + \ldots + w_m x_m = \langle w, x \rangle$$

- The problem is how to design a multiple input — single output model of the unknown dynamical system by building it around a single linear or nonlinear neuron — control the adjustment of the weights.

Classifier



$$y_i = f(\sum w_{ij} x_j + b)$$

$$f(u_j) = \begin{cases} 1 & u_j \geq 0 \\ -1 & u_j \prec 0 \end{cases}$$

Supervised learning

- Teacher gives samples of inputs $x(n)$ and corresponding desired outputs $t(n)$
- Goal is to find weights which imitate the behavior of the teacher

Learning Rule

$$w_i(n+1) = w_i(n) + \eta(\mathrm{t}(n) - y(n))x_i(n)$$
$$i = 1, 2, \ldots, m$$

- If the *n*-th input $x(n)$ is correctly classified, i.e., *t(n)=y(n)*
  - Nothing happens

- Otherwise, *t(n)!=y(n)*
  - Update weights (two cases)

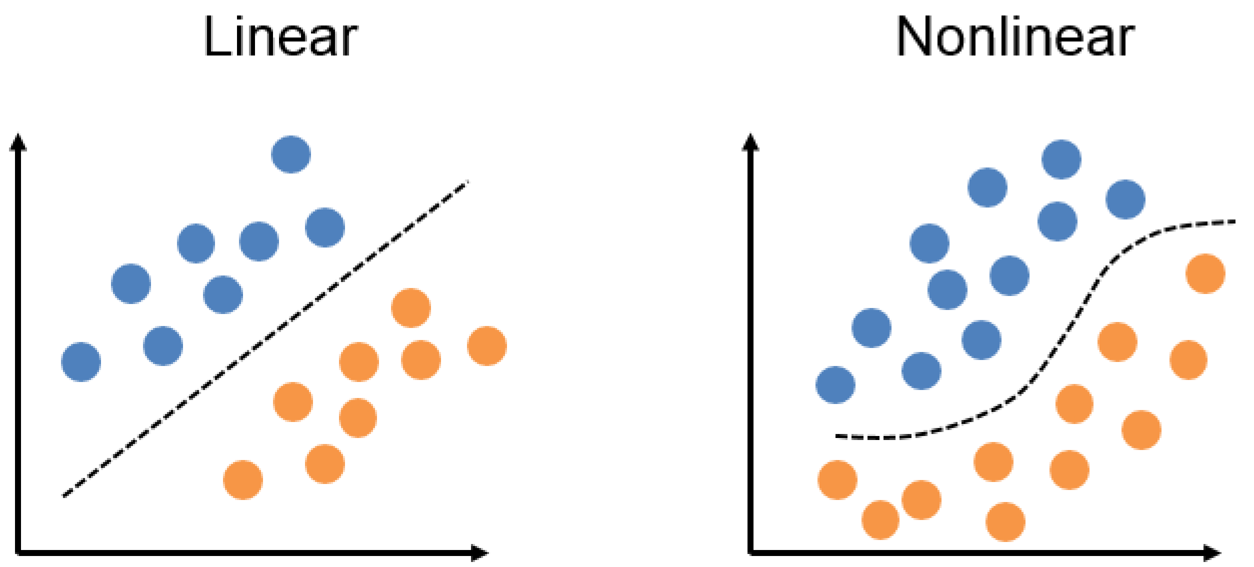
Adjust the Weights

- Adjust the weights
  - Start from randomly initialized weights
  - Update weights according to the rule
  - Stops when convergence or other condition is met

Perceptron convergence theorem

- For linearly separable problems, the algorithm **converges at finite steps**
- See proof (another pdf)

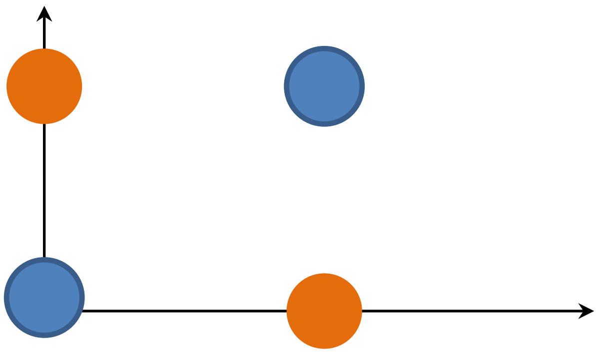
Linear and nonlinear data



Linear function can separate the data without any error

XOR Problem

- Some classifications are impossible
- A famous example: XOR problem
  - Class 1: (0, 0) and (1, 1)
  - Class 2: (1, 0) and (0, 1)
  - The classes are not linearly separable, i.e. there is no hyperplane (line in this case) separating the classes.



To be continued