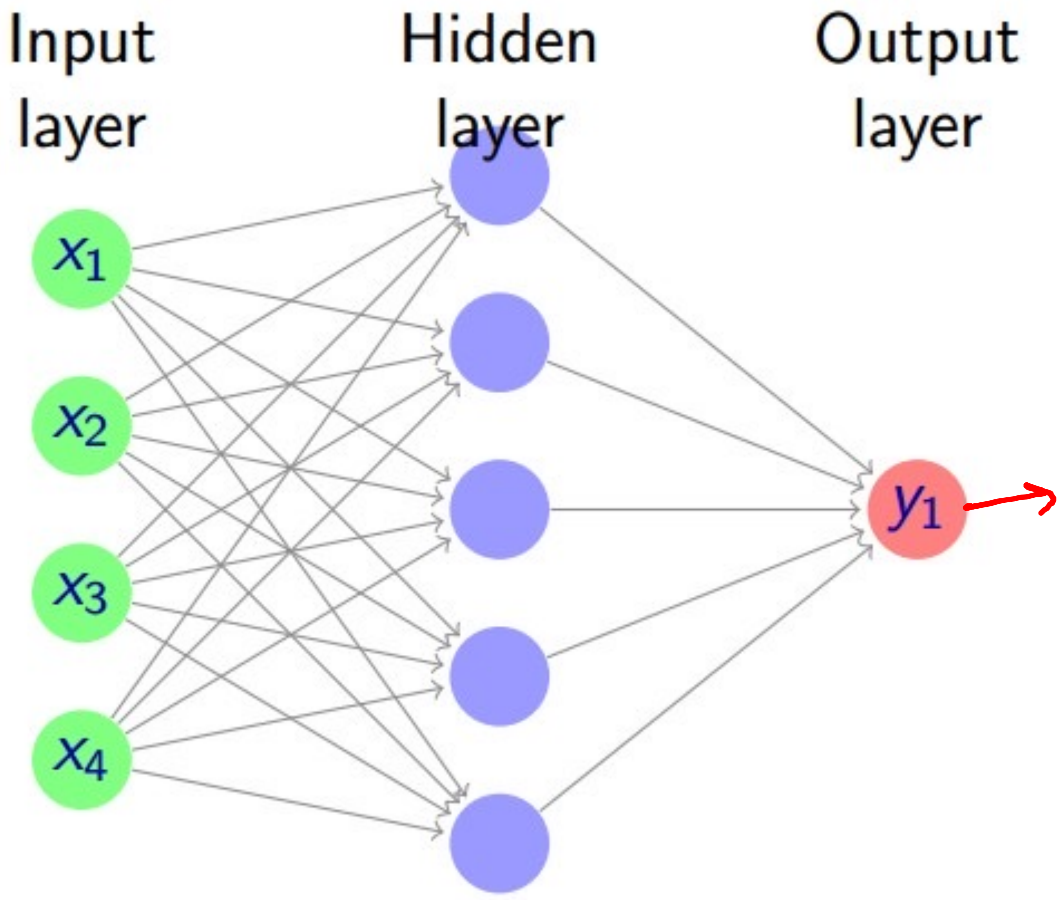
Lecture 15-b

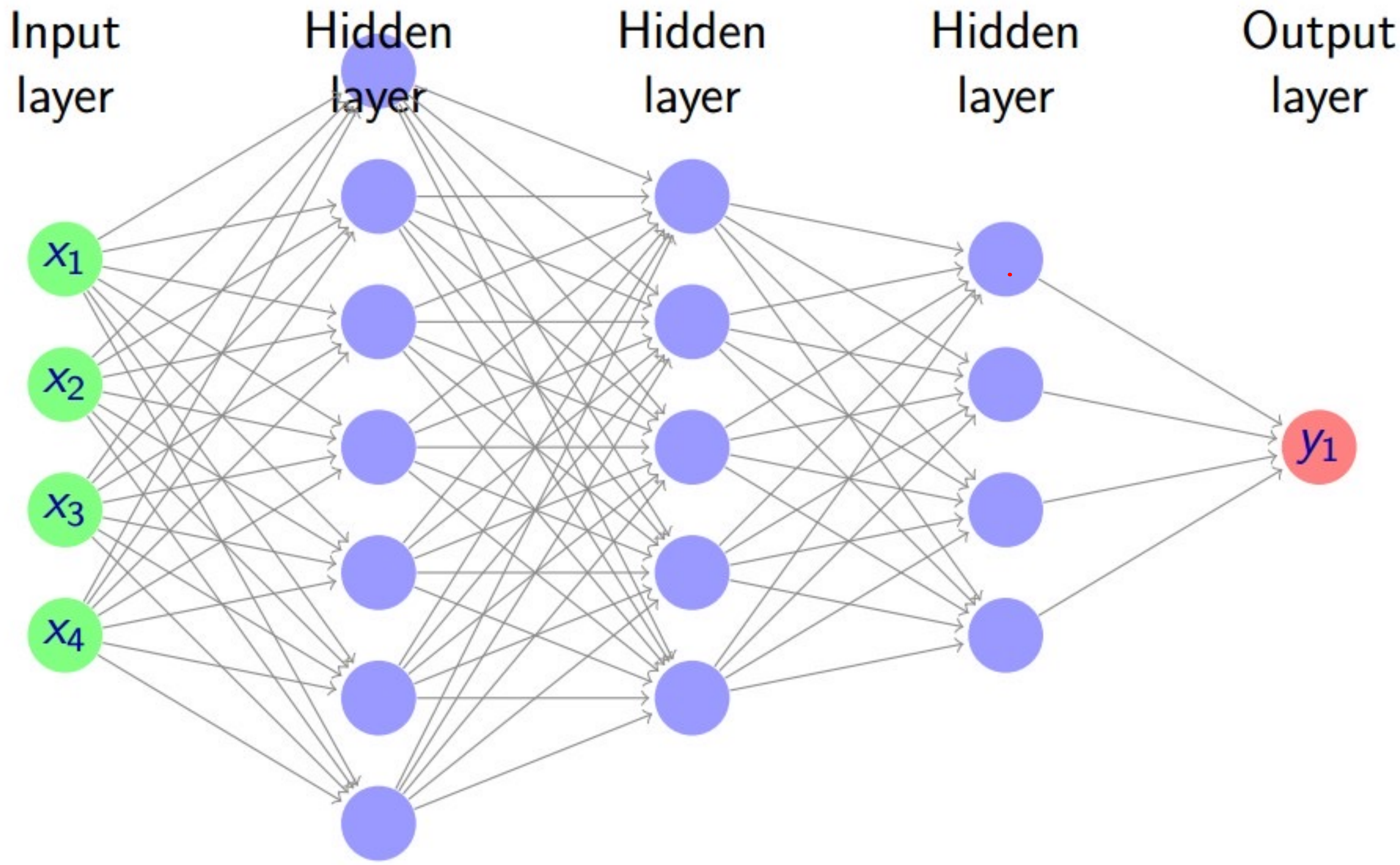
Deep Learning
Perceptron
Feed Forward Networks
Backpropagation

Slides from U Waterloo

Feedforward Neural Network



Feedforward Deep Networks



Feedforward Deep Networks

- Feedforward deep networks, a.k.a. multilayer perceptrons (MLPs), are parametric functions composed of several parametric functions.
- Each layer of the network defines one of these sub-functions.
- Each layer (sub-function) has multiple inputs and multiple outputs.
- Each layer composed of many units (scalar output of the layer).
- We sometimes refer to each unit as a feature.
- Each unit is usually a simple transformation of its input.
- The entire network can be very complex.

$x \in \mathbb{R}^d$ $\beta \in \mathbb{R}^d$ $\beta_0 \rightarrow$ bias parameter.

$\beta^T x + \beta_0$

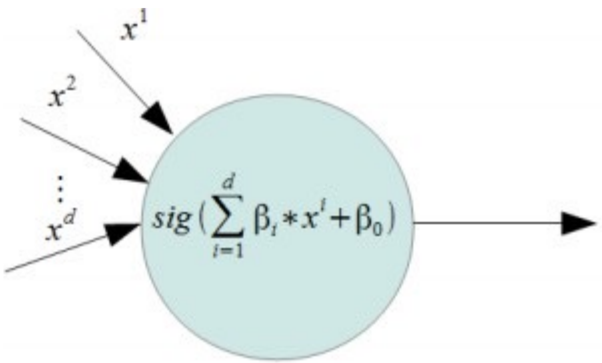
Perceptron

- The perceptron is the building block for neural networks.
- It was invented by Rosenblatt in 1957 at Cornell Labs, and first mentioned in the paper 'The Perceptron – a perceiving and recognizing automaton'.
- Perceptron computes a linear combination of factor of input and returns the sign.

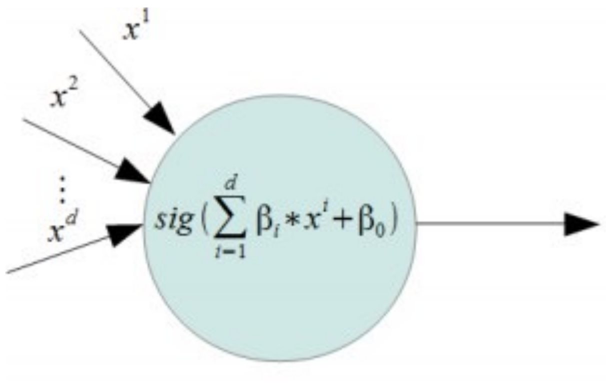
$\beta_1, \beta_2 \ldots \beta_d, \beta_0$

$\text{sign}(\beta^T x + \beta_0)$ 1, -1

$x, \beta \in \mathbb{R}^d$



Simple perceptron



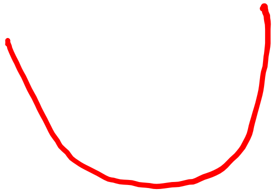
Simple perceptron

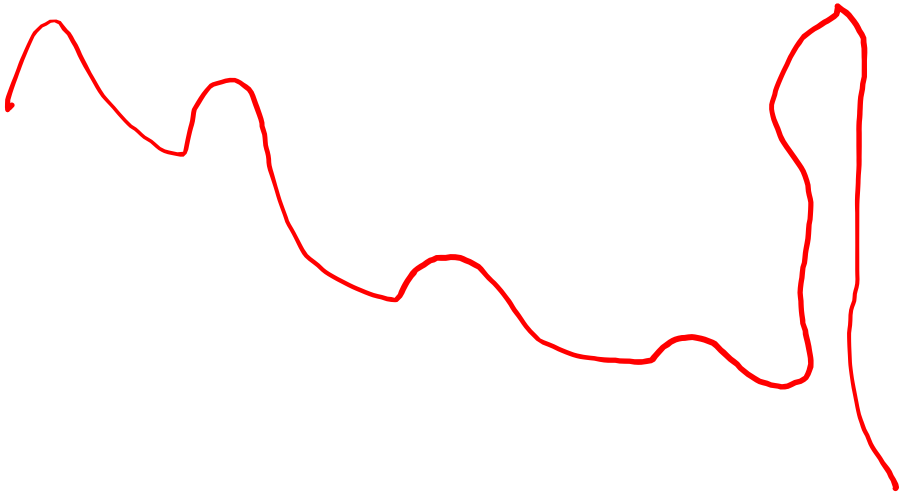
x^i is the i-th feature of a sample and β_i is the i-th weight. β_0 is defined as the bias. The bias alters the position of the decision boundary between the 2 classes. From a geometrical point of view, Perceptron assigns label "1" to elements on one side of $\beta^T x + \beta_0$ and label "-1" to elements on the other side

- define a cost function, $\phi(\boldsymbol{\beta}, \beta_0)$, as a summation of the distance between all misclassified points and the hyper-plane, or the decision boundary.
- To minimize this cost function, we need to estimate $\boldsymbol{\beta}, \boldsymbol{\beta_0}$.
  $\min_{\beta,\beta_0} \phi(\boldsymbol{\beta}, \beta_0) = \{\text{distance of all misclassified points}\}$

Score criterion

$\phi(\beta, \beta_0)$





Cost for error: $L(y, F)$

$$L(y, F) = |\, y - F \,|,\ (y - F)^2, \qquad y \in R$$

$y \in \{-1, 1\}$:

$$L(y, F) = \log(1 + e^{-yF}) \quad \text{logistic reg.}$$

$$L(y, F) = \max(0, 1 - yF) \quad \text{SVM}$$

ML Course.

Many many more

$\beta^T x + \beta_0$

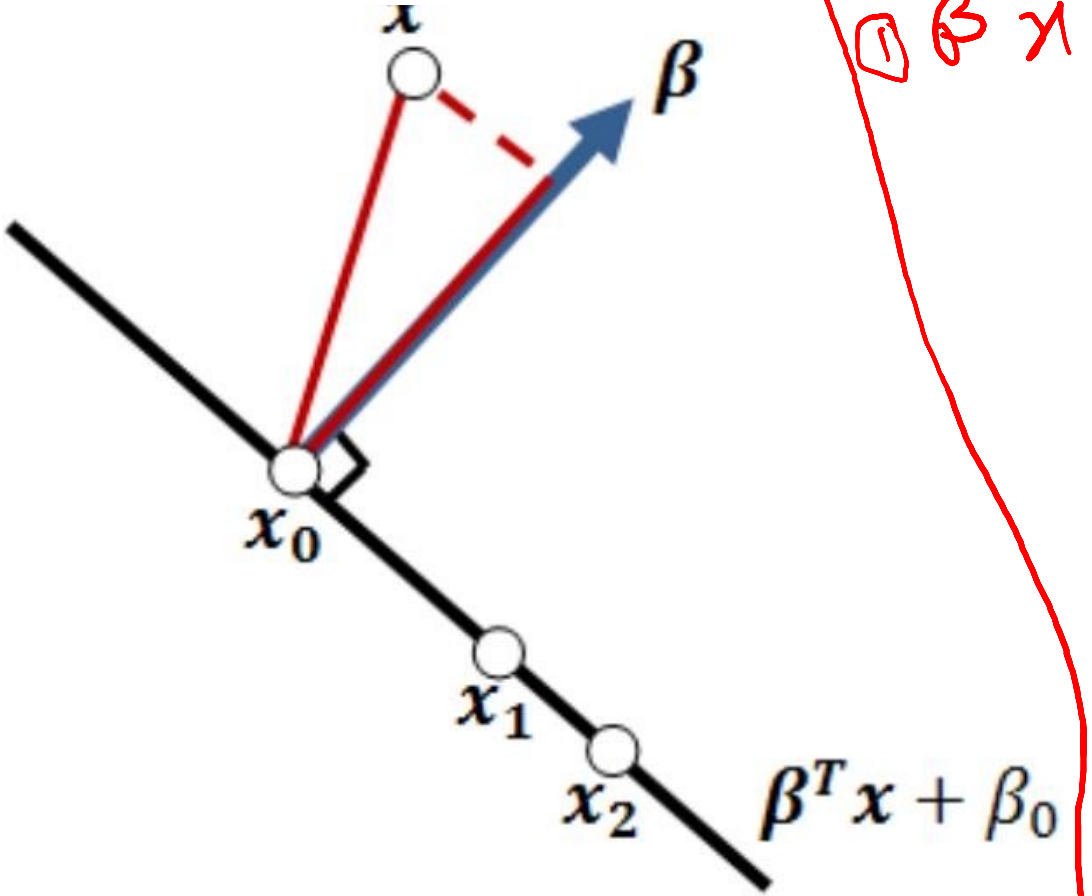
$\cancel{x - y + 1 = 0}$

$x \in R^2$

$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$x_1 - x_2 + 1 = 0$

$\beta = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad \beta_0 = 1$



① $\beta^T x_1 + \beta_0 = \beta^T x_2 + \beta_0$

$\beta^T (x_1 - x_2) = 0$

Distance between the point and the decision boundary hyperplane (black line).

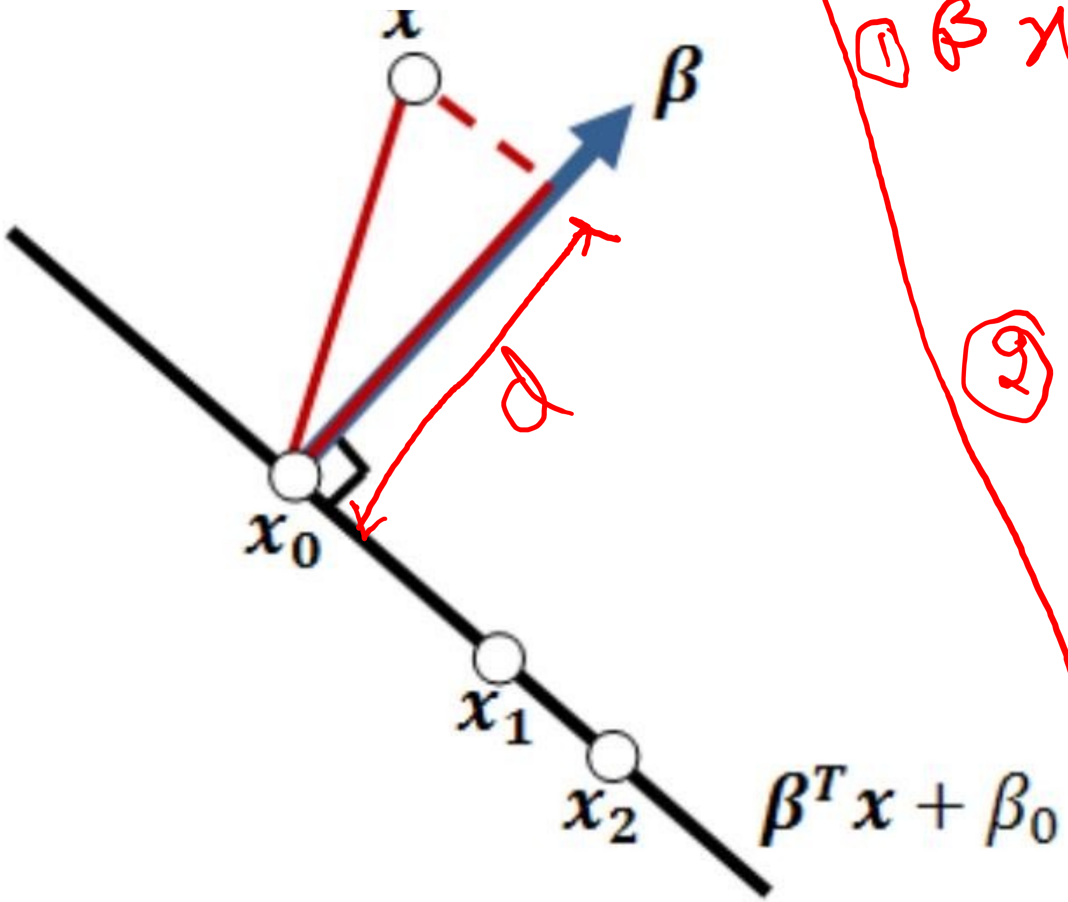
$\beta^T x + \beta_0$

$x \in R^2$

$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$x_1 - x_2 + 1 = 0$

$\beta = \begin{bmatrix} 1 \\ -1 \end{bmatrix} \quad \beta_0 = 1$



① $\beta^T x_1 + \beta_0 = \beta^T x_2 + \beta_0$

$\beta^T (x_1 - x_2) = 0$

② $\beta^T x_0 + \beta_0 = 0$

$\beta^T x_0 = -\beta_0$

③ $\beta^* = \dfrac{\beta}{||\beta||}$

$\beta^{*T}(x - x_0)$

Distance between the point and the decision boundary hyperplane (black line).

$$\frac{\beta^T x}{\|\beta\|} - \frac{\beta^T x_0}{\|\beta\|} = \frac{\beta^T x + \beta_0}{\|\beta\|}$$

Say x is misclassified. $x \rightarrow$ Class 1

label of x was $y = 1$

$\beta^T x + \beta_0 < 0 \longrightarrow$ belongs to class -1

$$y(\beta^T x + \beta_0) < 0$$

$$\boxed{-y(\beta^T x + \beta_0) > 0}$$

for any ~~point~~ mis-classified point x_i, with true label y_i, the distance

$$d_i = -y_i\left(\beta^T x_i + \beta_0\right)$$