Winter 2019

CSEN1022: Machine Learning

Discriminant Functions (1)

Seif Eldawlatly

Supervised Learning

- Definition

  The task of inferring a function from labeled data

- Typically involves two phases
  - Training phase: Infer the function from provided input vectors and their corresponding labels
  - Test phase: Use the inferred function to predict the label of a new input vector (different from input vectors used during training)

- Formally

  Given a training dataset of N observations $\{x_n\}$, where $n = 1, 2, \ldots, N$ together with the corresponding target values $\{t_n\}$, the goal is to predict the value of t for a new value of x

Linear Classification

- Classification

  Take an input $\mathbf{x}$ and assign it to one of K discrete classes

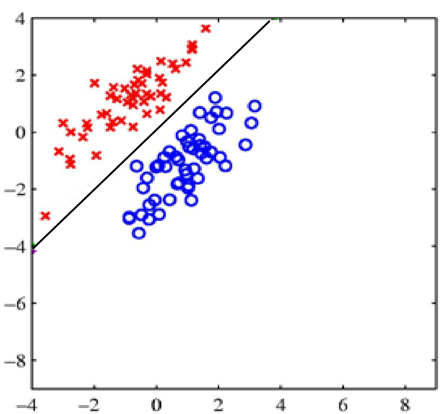
- Decision Boundary

  A boundary (could be linear or non-linear) between two decision regions

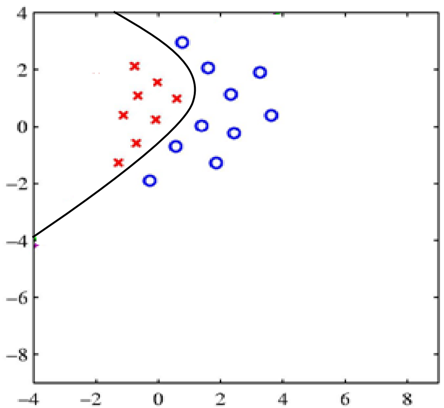
- Decision Regions:

  \- Red or Blue, 1 or -1, Friend or Enemy

Linearly Separable



Non-linearly Separable



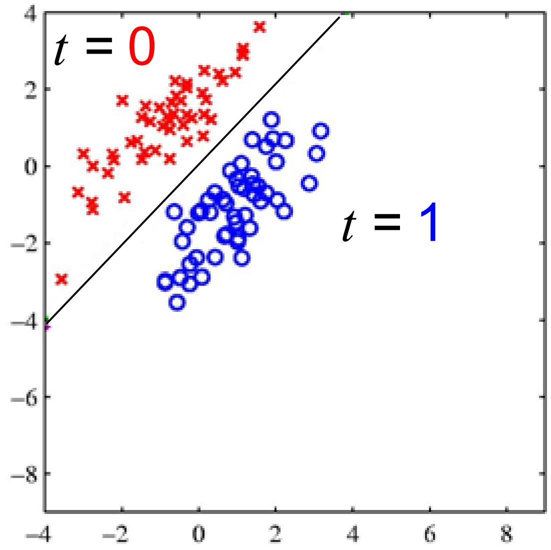
Linear Classification

- Classification Problem

  Goal: Determine the target value (label) for a data point

  Input vector: $\mathbf{x}$

  Target variable: t



- Two Classes (K = 2)

  $t \in \{0,1\}$ — t = 0 → Class C_1; t = 1 → Class C_2

- K Classes (K > 2)

  $$\mathbf{t}_i = [t_1, t_2, \ldots, t_K], \text{where } t_n = 1 \text{ for } \mathbf{x}_i \in C_n \text{ and } t_m = 0, m \neq n$$

  Example: $\mathbf{x}_i \in C_3$, $K = 5$ → $\mathbf{t}$ = [0, 0, 1, 0, 0]

Linear Classifiers

- We will discuss 3 major types of linear classifiers:
    - Discriminant Functions
    - Probabilistic Generative Models
    - Probabilistic Discriminative Models

Discriminant Functions

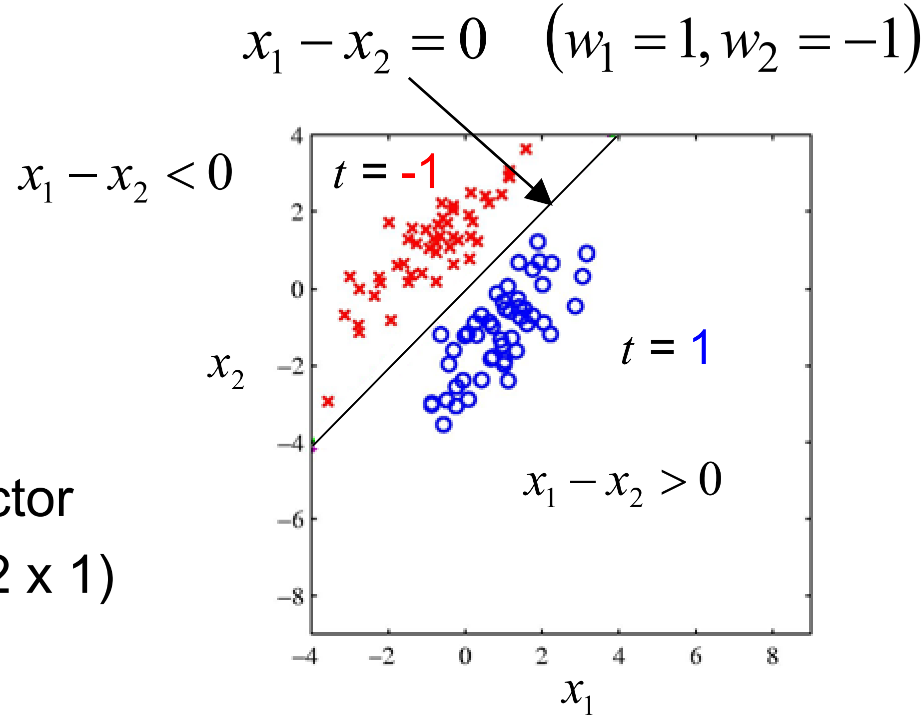
- For the case of two classes

$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

$\mathbf{x}$: Input vector

$\mathbf{w}$: Weight vector

w_0: bias



- For this example, $\mathbf{w}$ is a (2 x 1) vector and each input vector $\mathbf{x}$ is also a (2 x 1) vector

$$y(\mathbf{x}) = \begin{bmatrix} w_1 & w_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + w_0$$

If the total number of input vectors is 100, then the input dataset consists of $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, …, $\mathbf{x}_{100}$, where $\mathbf{x}_i = [x_{i1}, x_{i2}]$, $i = 1{:}100$

- Decision Surface is a hyperplane

Discriminant Functions

- In this type of methods, the decision boundary is linear but the classification decision is always non-linear

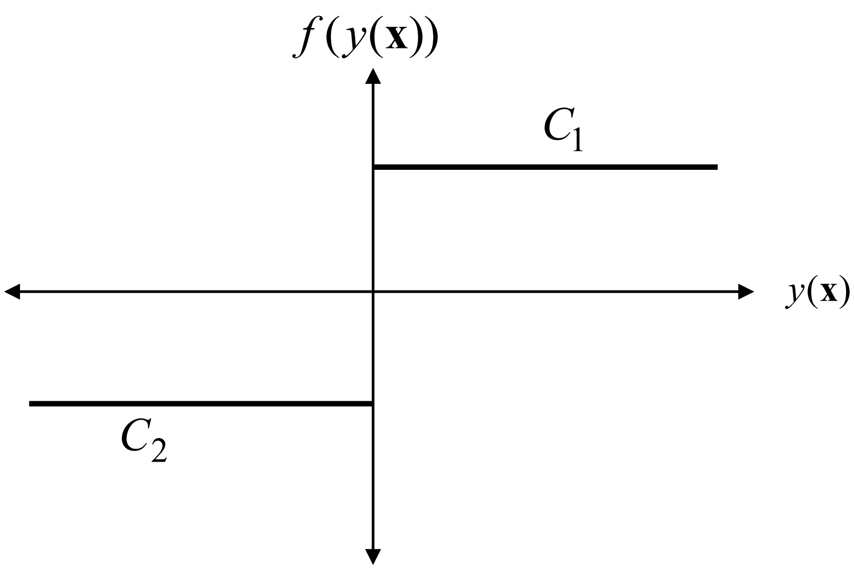
$$\text{if } y(\mathbf{x}) > 0, C = C_1$$
$$\text{if } y(\mathbf{x}) < 0, C = C_2$$

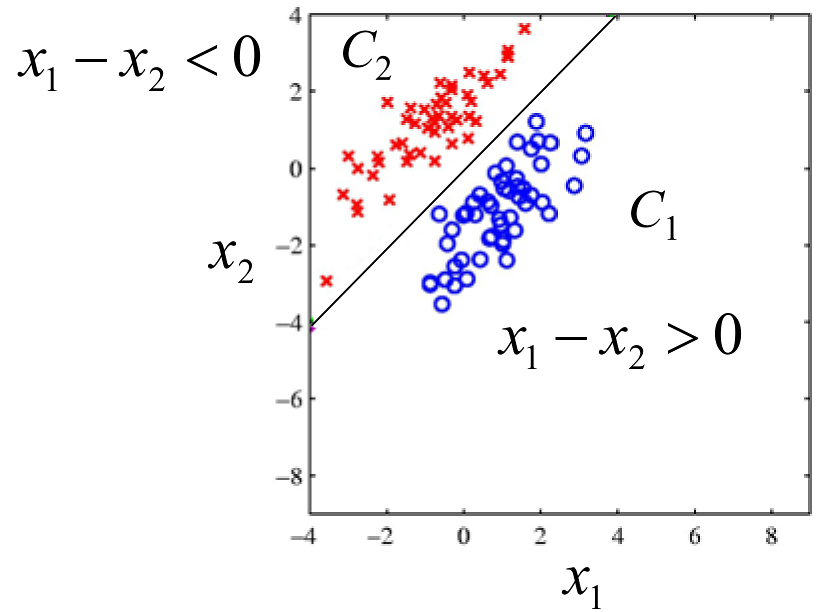
$\downarrow$

Non-linear Function

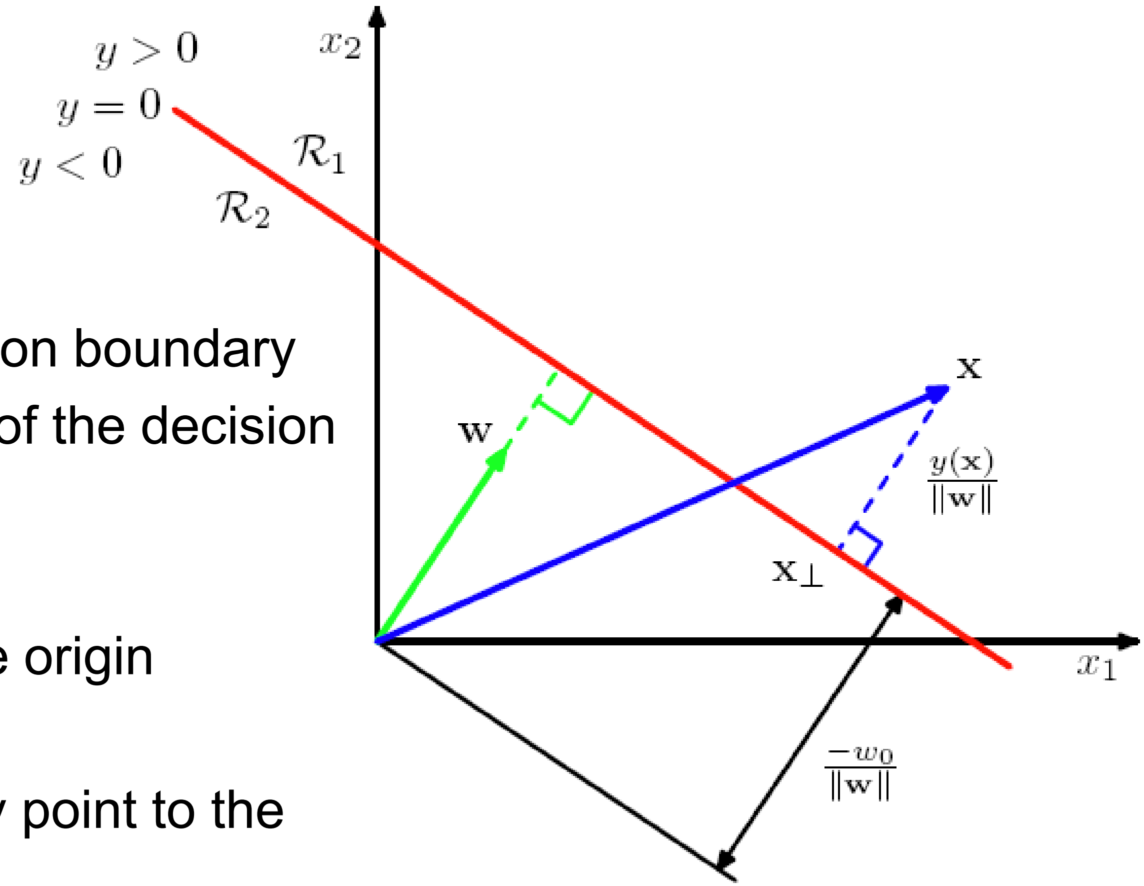
$$Decision = f(y(\mathbf{x}))$$

(Generalized Linear Model)





Discriminant Function Properties



$\mathbf{w}$ is orthogonal to the decision boundary so it defines the orientation of the decision boundary

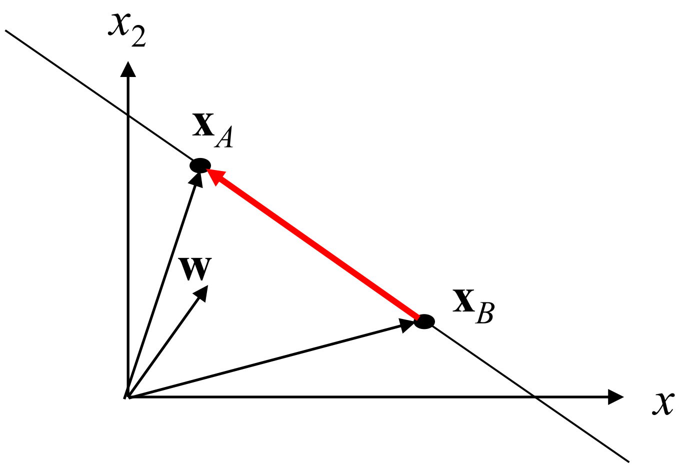
$\frac{-w_0}{\|\mathbf{w}\|}$ is the distance from the origin

$\frac{y(\mathbf{x})}{\|\mathbf{w}\|}$ is the distance from any point to the decision boundary

Discriminant Function Properties

- $\mathbf{w}$ is orthogonal to the decision boundary

We need to prove that the dot product between $\mathbf{w}$ and the direction of the decision boundary is 0.



Assume any 2 points $\mathbf{x}_A$ and $\mathbf{x}_B$ on the decision boundary. These 2 points must satisfy the following equation since they are on the decision boundary

$$y(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + w_0 = 0$$

Therefore, $\mathbf{w}^T\mathbf{x}_A + w_0 = 0$

$$\mathbf{w}^T\mathbf{x}_B + w_0 = 0$$

By subtracting the last 2 equations we get, $\mathbf{w}^T(\mathbf{x}_A - \mathbf{x}_B) = 0$

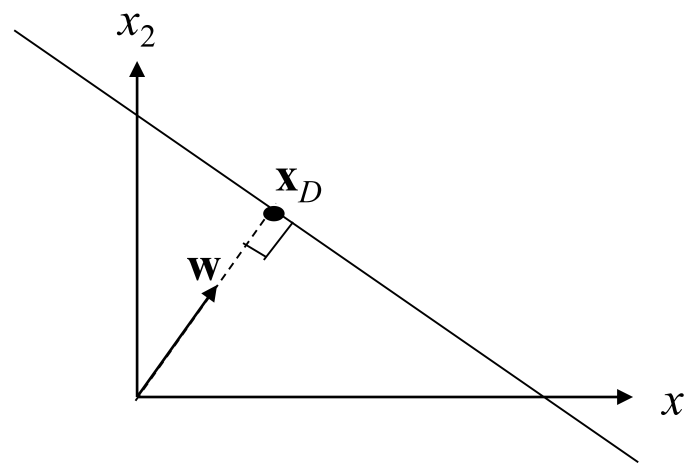
Since $\mathbf{x}_A$ - $\mathbf{x}_B$ is the vector from the point $\mathbf{x}_B$ to the point $\mathbf{x}_A$, then such vector is in the same direction as the decision boundary.

Since $\mathbf{w}^T(\mathbf{x}_A - \mathbf{x}_B)$ is the dot product between $\mathbf{w}$ and the vector that is in the same direction of the decision boundary and we proved that it is 0, therefore $\mathbf{w}$ is orthogonal to the decision boundary given that the such dot product is equal to $\mathbf{w}\cdot\mathbf{u} = \|\mathbf{w}\|.\|\mathbf{u}\|\cos\theta$ which will be 0 if $\cos\theta = 0$ which will happen if the angle is 90° (in the non-trivial case)

Discriminant Function Properties

- $\frac{-w_0}{\|\mathbf{w}\|}$ is the distance between the decision boundary and the origin

We need to find the norm of the vector $\mathbf{x}_D$



Since the point $\mathbf{x}_D$ is on the decision boundary, then it satisfies $\mathbf{w}^T\mathbf{x}_D + w_0 = 0$ (1)

Since the dot product of $\mathbf{w}$ and $\mathbf{x}_D$ is equal to

$$\mathbf{w}\cdot\mathbf{x}_D = \mathbf{w}^T\mathbf{x}_D = \|\mathbf{w}\|.\|\mathbf{x}_D\|\cos\theta$$

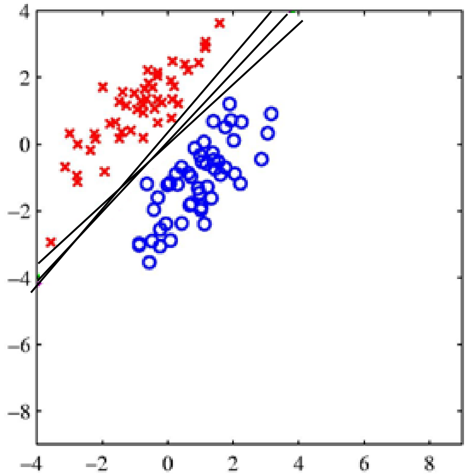
Since $\mathbf{w}$ and $\mathbf{x}_D$ are both in the same direction, therefore $\cos\theta = 1$ and

$$\mathbf{w}^T\mathbf{x}_D = \|\mathbf{w}\|.\|\mathbf{x}_D\| \qquad (2)$$

Since from equation (1), $\mathbf{w}^T\mathbf{x}_D = -w_0$

By substituting in (2), we get $\|\mathbf{x}_D\| = \frac{-w_0}{\|\mathbf{w}\|}$

Learning Classifier Parameters



$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

How to find $\mathbf{w}$ and w_0 ?

Least Squares

Fisher’s Linear Discriminant

Perceptron

Least Squares for Classification

- A Simple Solution

$$y(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$$

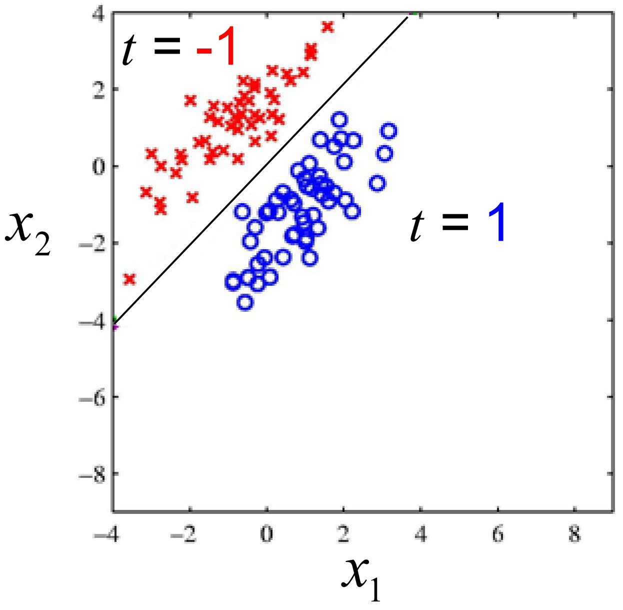
Goal: Find $\mathbf{w}$ and w_0 such that $y(\mathbf{x}) = t$

Let $\widetilde{\mathbf{w}} = [\mathbf{w}; w_0]$ $\quad\quad$ $\widetilde{\mathbf{x}} = [\mathbf{x}; 1]$

We define an error function as

$$E_D(\widetilde{\mathbf{w}}) = \frac{1}{2} \sum_{i=1}^{n} \left(\widetilde{\mathbf{w}}^T \widetilde{\mathbf{x}}_i - t_i \right)^2$$

where n is the total number of input vectors



- Least squares classifier tries to minimize the difference between the actual ($y(\mathbf{x})$) and desired (t) target values for all input vectors

Least Squares for Classification

- Let

$$\widetilde{\mathbf{w}} = [\mathbf{w}\,; w_0] \qquad \widetilde{\mathbf{x}} = [\mathbf{x}\,; 1]$$

$$E_D(\widetilde{\mathbf{w}}) = \frac{1}{2}\sum_{i=1}^{n}\left(\widetilde{\mathbf{w}}^T\widetilde{\mathbf{x}}_i - t_i\right)^2$$

To find $\widetilde{\mathbf{w}}$ such that error E_D is minimum, we need to take derivative of E_D with respect to $\widetilde{\mathbf{w}}$ and equate with zero

- For 2-dimensional input vectors, let

$$\widetilde{\mathbf{w}} = \begin{bmatrix} w_1 \\ w_2 \\ w_0 \end{bmatrix} \qquad \widetilde{\mathbf{X}} = \begin{bmatrix} x_{11} & x_{12} & 1 \\ x_{21} & x_{22} & 1 \\ \vdots & \vdots & \vdots \\ x_{n1} & x_{n2} & 1 \end{bmatrix} \qquad \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ \vdots \\ t_n \end{bmatrix}$$

Weights Vector   Input Data Matrix   Targets Vector

Least Squares for Classification

- The equation $E_D(\widetilde{\mathbf{w}}) = \frac{1}{2}\sum_{i=1}^{n}\left(\widetilde{\mathbf{w}}^T\widetilde{\mathbf{x}}_i - t_i\right)^2$ can be then written as

$$E_D(\tilde{\mathbf{w}}) = \frac{1}{2}\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}} - \mathbf{t}\right)^T\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}} - \mathbf{t}\right)$$

$$= \frac{1}{2}\left[\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right) - \left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\mathbf{t} - \mathbf{t}^T\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right) + \mathbf{t}^T\mathbf{t}\right]$$

$$= \frac{1}{2}\left[\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right) - \left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\mathbf{t} - \left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\mathbf{t} + \mathbf{t}^T\mathbf{t}\right]$$

$$= \frac{1}{2}\left[\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right) - 2\left(\tilde{\mathbf{X}}\tilde{\mathbf{w}}\right)^T\mathbf{t} + \mathbf{t}^T\mathbf{t}\right]$$

- We next compute the derivative with respect to $\widetilde{\mathbf{w}}$ and equate with 0

$$\frac{\partial}{\partial\tilde{\mathbf{w}}}E_D(\tilde{\mathbf{w}}) = \frac{1}{2}\left[2\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\tilde{\mathbf{w}} - 2\tilde{\mathbf{X}}^T\mathbf{t}\right] = 0$$

$$\therefore \tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\tilde{\mathbf{w}} - \tilde{\mathbf{X}}^T\mathbf{t} = 0 \rightarrow \tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\tilde{\mathbf{w}} = \tilde{\mathbf{X}}^T\mathbf{t}$$

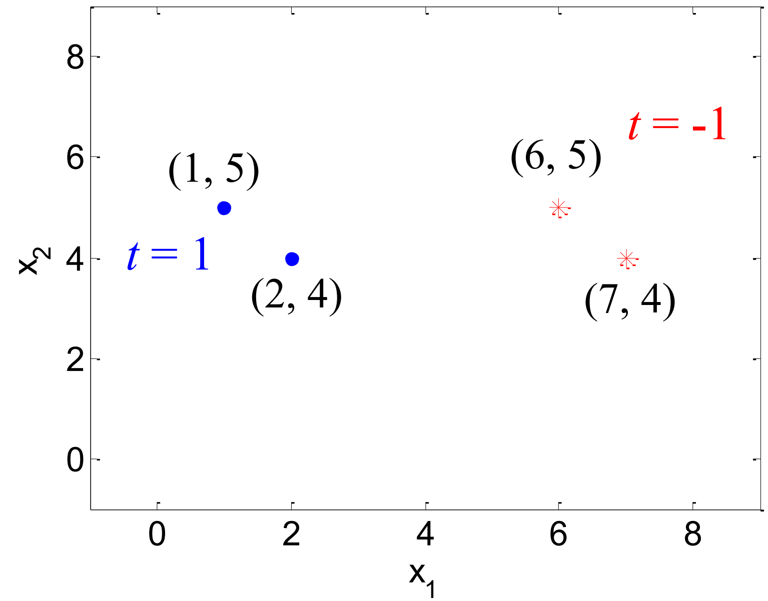
$$\boxed{\therefore \tilde{\mathbf{w}} = \left(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\right)^{-1}\tilde{\mathbf{X}}^T\mathbf{t}}$$

A Simple Example

- Consider the data given by

- The least squares solution is

$$\tilde{\mathbf{w}} = \left(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\right)^{-1}\tilde{\mathbf{X}}^T\mathbf{t}$$



$$\tilde{\mathbf{X}} = \begin{bmatrix} 1 & 5 & 1 \\ 2 & 4 & 1 \\ 6 & 5 & 1 \\ 7 & 4 & 1 \end{bmatrix}$$

$$\tilde{\mathbf{X}}^T\tilde{\mathbf{X}} = \begin{bmatrix} 1 & 2 & 6 & 7 \\ 5 & 4 & 5 & 4 \\ 1 & 1 & 1 & 1 \end{bmatrix}\begin{bmatrix} 1 & 5 & 1 \\ 2 & 4 & 1 \\ 6 & 5 & 1 \\ 7 & 4 & 1 \end{bmatrix} = \begin{bmatrix} 90 & 71 & 16 \\ 71 & 82 & 18 \\ 16 & 18 & 4 \end{bmatrix}$$

$$\left(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\right)^{-1} = \begin{bmatrix} 0.04 & 0.04 & -0.34 \\ 0.04 & 1.04 & -4.84 \\ -0.34 & -4.84 & 23.39 \end{bmatrix}$$

$$\left(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\right)^{-1}\tilde{\mathbf{X}}^T = \begin{bmatrix} -0.1 & -0.1 & 0.1 & 0.1 \\ 0.4 & -0.6 & 0.6 & -0.4 \\ -1.15 & 3.35 & -2.85 & 1.65 \end{bmatrix}$$

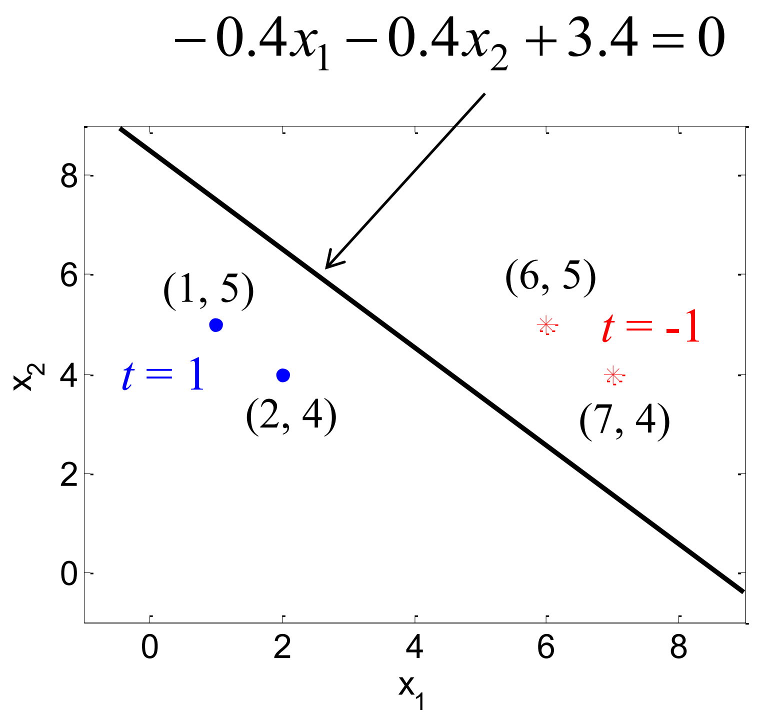
A Simple Example

$$\mathbf{t} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

$$\left(\widetilde{\mathbf{X}}^T \widetilde{\mathbf{X}}\right)^{-1} \widetilde{\mathbf{X}}^T \mathbf{t} = \begin{bmatrix} -0.1 & -0.1 & 0.1 & 0.1 \\ 0.4 & -0.6 & 0.6 & -0.4 \\ -1.15 & 3.35 & -2.85 & 1.65 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

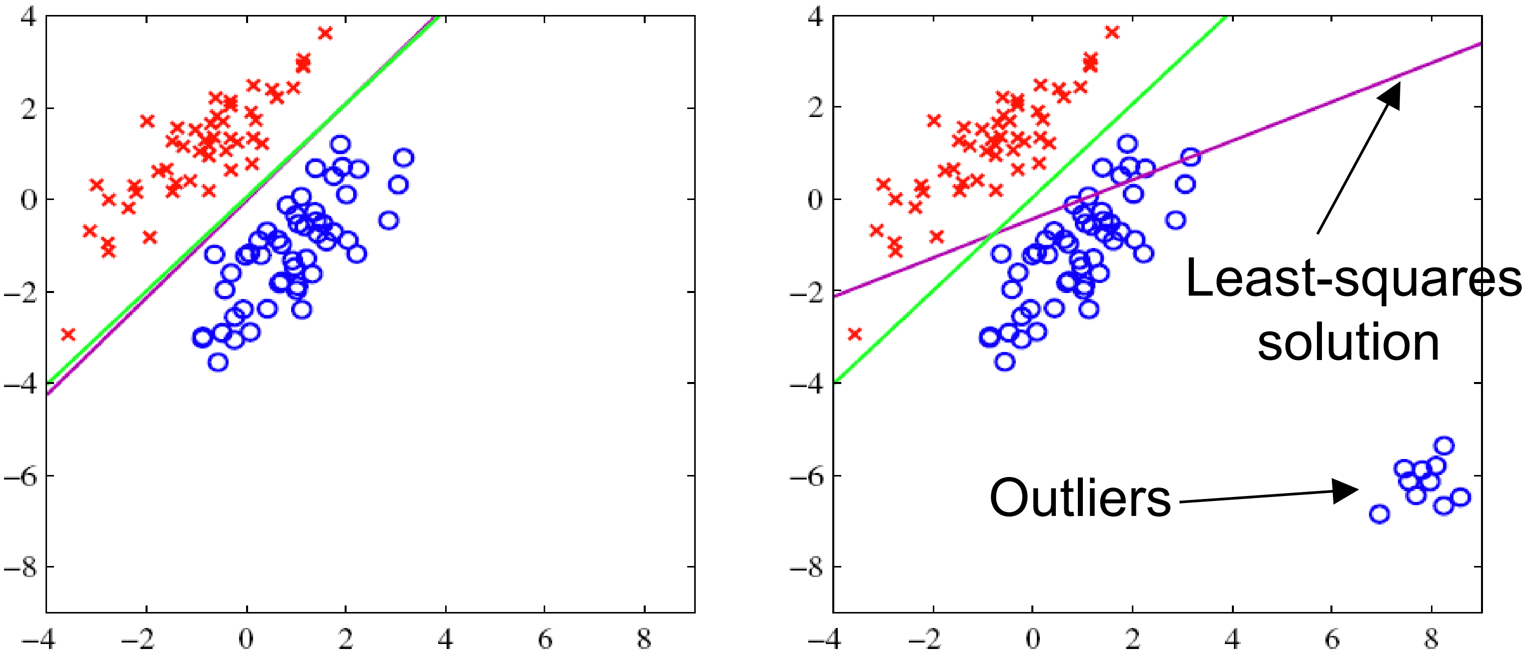
$$= \begin{bmatrix} -0.4 \\ -0.4 \\ 3.4 \end{bmatrix} = \widetilde{\mathbf{w}}$$

$$y(\widetilde{\mathbf{x}}) = \widetilde{\mathbf{w}}^T \widetilde{\mathbf{x}} = -0.4x_1 - 0.4x_2 + 3.4$$



Least Squares for Classification

- Problems: Not robust to outliers

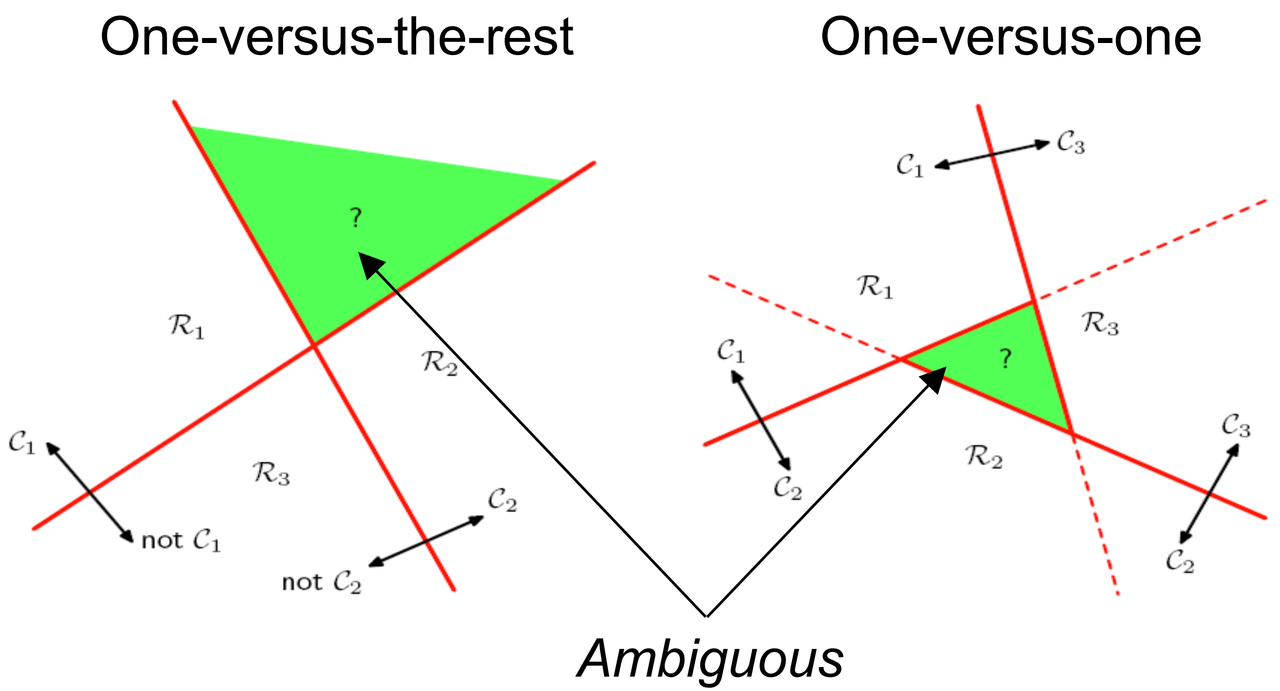


- Reason: The error function penalizes points that are too correct

$$E_D(\widetilde{\mathbf{w}}) = \frac{1}{2}\sum_{i=1}^{n}\left(\widetilde{\mathbf{w}}^T\widetilde{\mathbf{x}}_i - t_i\right)^2$$

K-class Discriminant Function

- To classify to multiple classes, a number of ways could be used



- Solution: Use K linear functions

$$y_k(\mathbf{x}) = \mathbf{w}_k^T \mathbf{x} + w_{k0}, k \in \{1, 2, \ldots, K\}$$

$$\text{if } \; y_k(\mathbf{x}) > y_j(\mathbf{x}) \; \text{for all } \; j \neq k, \; \text{then } \; C = C_k$$