Machine Learning

Lecture 06

Min-Kuan Chang

minkuanc@nchu.edu.tw

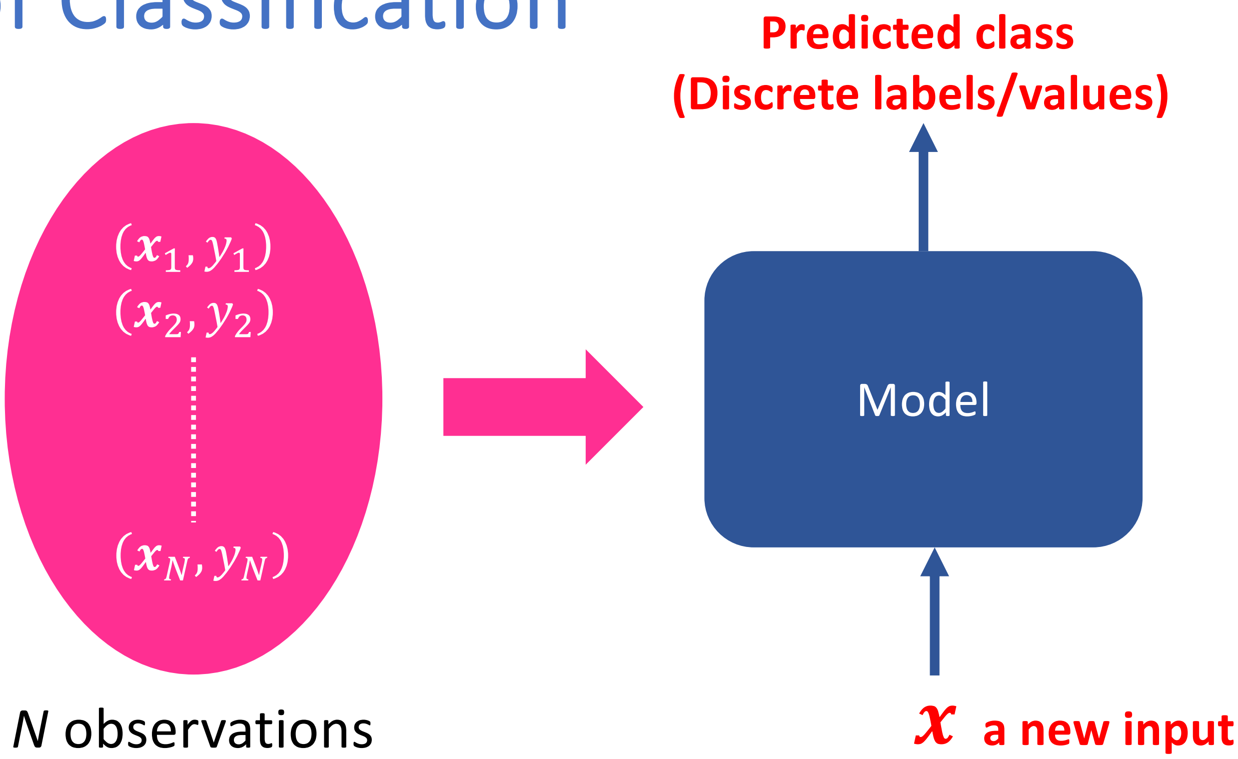
EE, College of EECS

Linear Models for Classification

Topic 01

Two-Class Classification

Goal of Classification



- The goal in classification is to take an input vector $\boldsymbol{x}$ and to assign it to one of K discrete classes C_k where $k = 1, 2, \cdots, K$
  - the classes are taken to be disjoint
    - each input is assigned to one and only one class
  - the input space is divided into decision regions whose boundaries are called decision boundaries or decision surfaces
- In this lecture, we consider linear models for classification
  - the decision surfaces are linear functions of the input vector
  - the decision surfaces are defined by $(D-1)$-dimensional hyperplanes within the D-dimensional input space
  - data sets whose classes can be separated exactly by linear decision surfaces are said to be linearly separable

Two-Class Classification

- In the two-class classification, a target has two possible labels or values. For example, $y_i \in \{C_1, C_2\}$ or $y_i \in \{-1,1\}$
- A discriminant is present to help classify an input
- The discriminant is to map an input to one of the classes, which is either C_1 or C_2(-1 or 1) in the case of the two-class classification
- The simplest form of discriminant is the linear discriminant function

判別

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

together with the decision boundary to assign a class to the input $\boldsymbol{x}$

Two-Class Classification

- In the two-class classification, the linear discriminant function maps an input $\boldsymbol{x}$ to one of the two classes, C_1 and C_2

$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0 \underset{C_2}{\overset{C_1}{\gtrless}} 0$$

- The decision boundary in the two-class classification

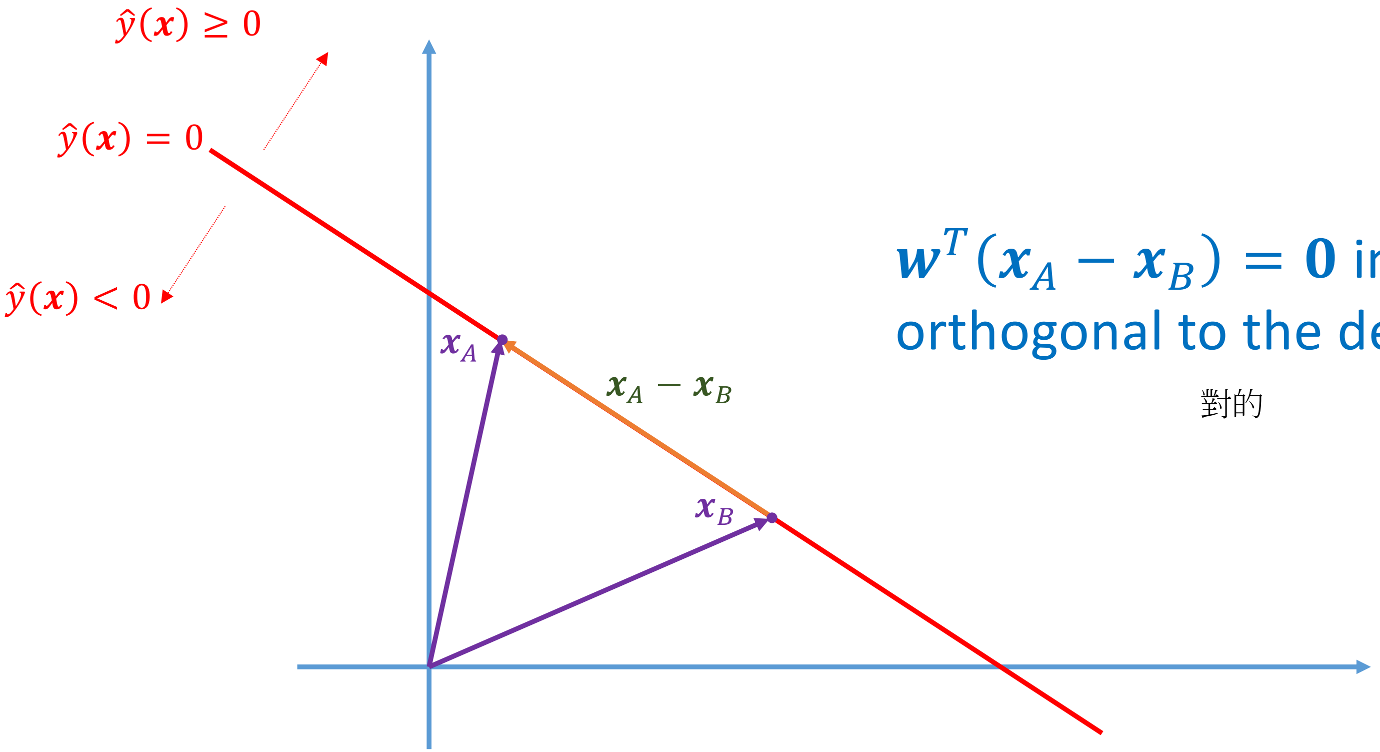
$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0 = 0$$

Two-Class Classification

- Geometry of the decision boundary
  - two inputs, say $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$ lie on the decision boundary

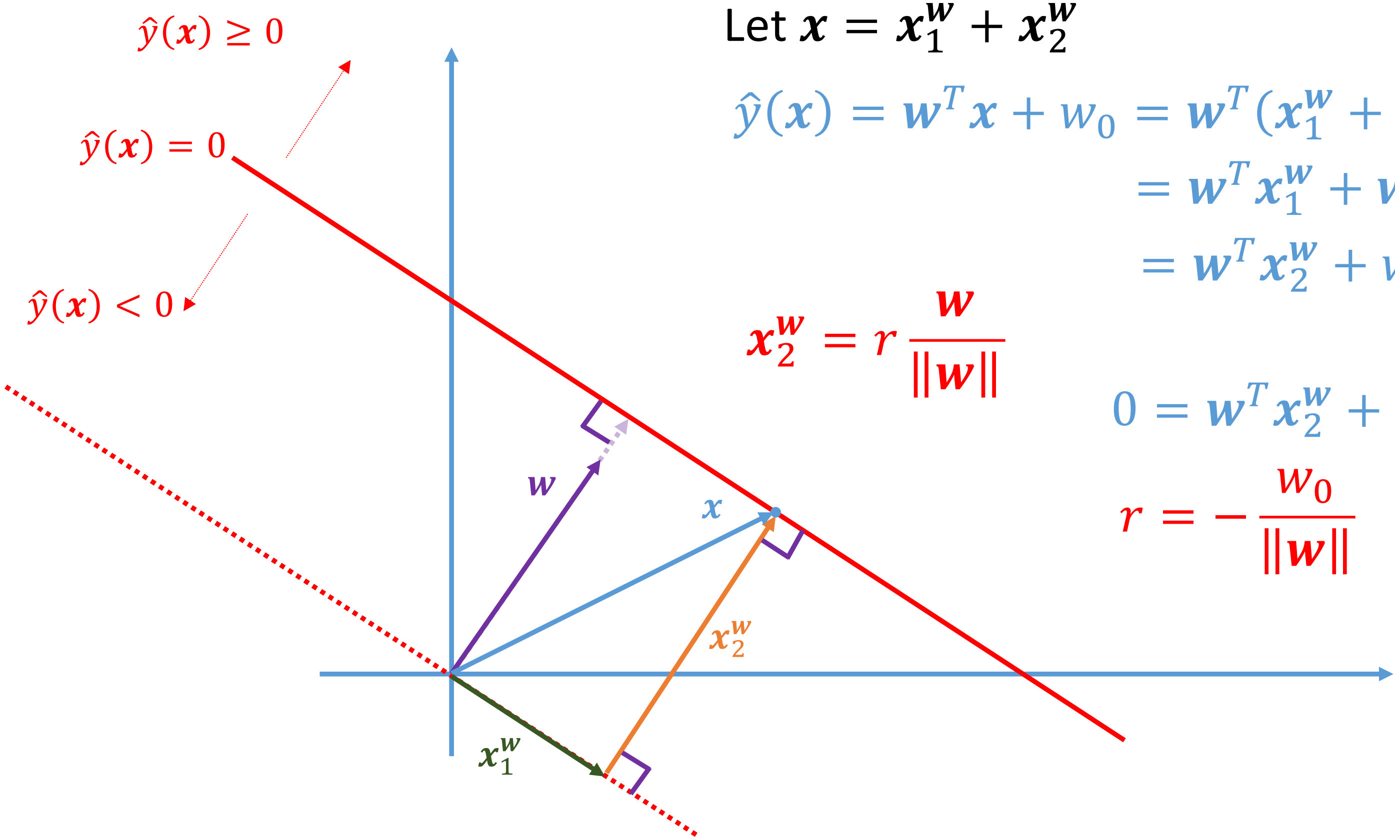
$$\boldsymbol{w}^T\boldsymbol{x}_A + w_0 = 0 \qquad \boldsymbol{w}^T\boldsymbol{x}_B + w_0 = 0$$

$$\boldsymbol{w}^T(\boldsymbol{x}_A - \boldsymbol{x}_B) = \mathbf{0}$$



$\boldsymbol{w}^T(\boldsymbol{x}_A - \boldsymbol{x}_B) = \boldsymbol{0}$ implies $\boldsymbol{w}$ is orthogonal to the decision boundary

對的



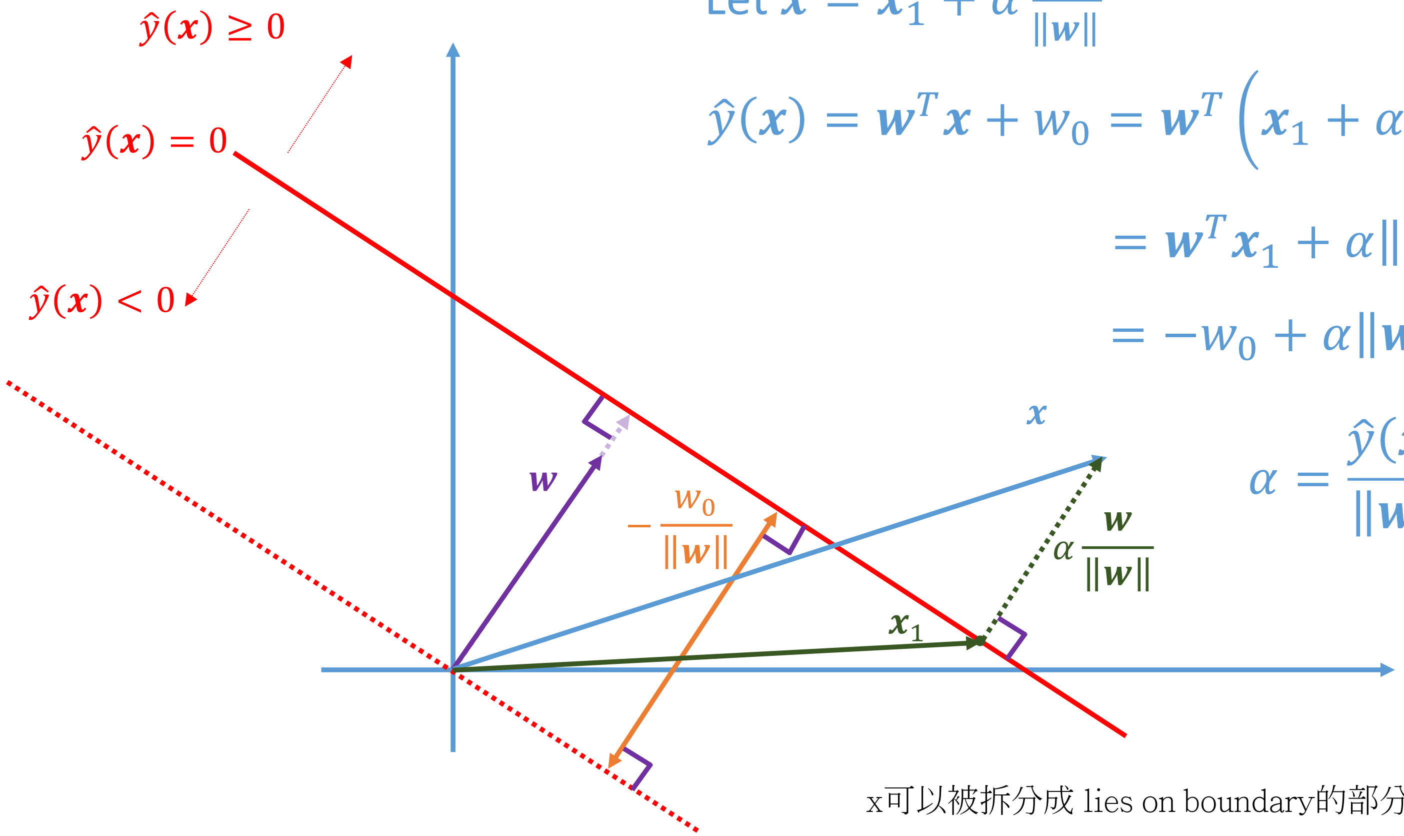
Let $\boldsymbol{x} = \boldsymbol{x}_1^{\boldsymbol{w}} + \boldsymbol{x}_2^{\boldsymbol{w}}$

$$
\begin{aligned}
\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} + w_0 &= \boldsymbol{w}^T(\boldsymbol{x}_1^{\boldsymbol{w}} + \boldsymbol{x}_2^{\boldsymbol{w}}) + w_0 \\
&= \boldsymbol{w}^T\boldsymbol{x}_1^{\boldsymbol{w}} + \boldsymbol{w}^T\boldsymbol{x}_2^{\boldsymbol{w}} + w_0 \\
&= \boldsymbol{w}^T\boldsymbol{x}_2^{\boldsymbol{w}} + w_0 = 0
\end{aligned}
$$

$$\boldsymbol{x}_2^{\boldsymbol{w}} = r\frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$$

$$0 = \boldsymbol{w}^T\boldsymbol{x}_2^{\boldsymbol{w}} + w_0 = r\|\boldsymbol{w}\| + w_0$$

$$r = -\frac{w_0}{\|\boldsymbol{w}\|}$$



Let $\boldsymbol{x} = \boldsymbol{x}_1 + \alpha \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|}$

$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + w_0 = \boldsymbol{w}^T \left(\boldsymbol{x}_1 + \alpha \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|} \right) + w_0$$

$$= \boldsymbol{w}^T \boldsymbol{x}_1 + \alpha \|\boldsymbol{w}\| + w_0$$

$$= -w_0 + \alpha \|\boldsymbol{w}\| + w_0 = \alpha \|\boldsymbol{w}\|$$

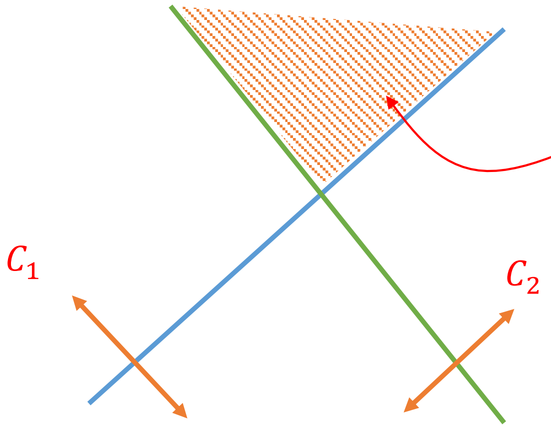
$$\alpha = \frac{\hat{y}(\boldsymbol{x})}{\|\boldsymbol{w}\|} \qquad \rho = |\alpha| = \frac{|\hat{y}(\boldsymbol{x})|}{\|\boldsymbol{w}\|}$$

The geometric margin at $\boldsymbol{x}$

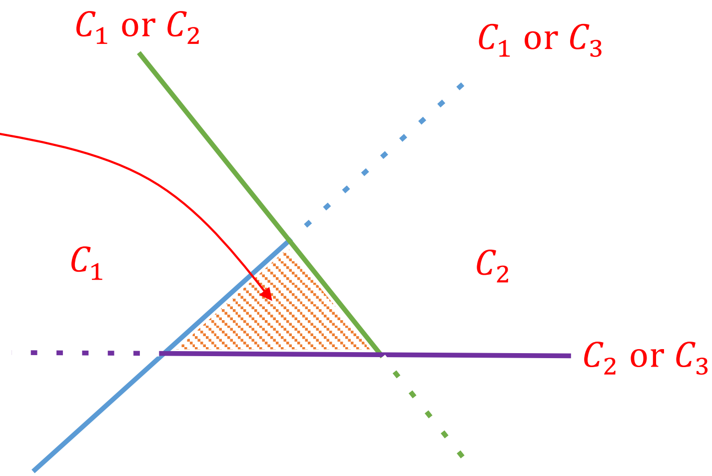
x可以被拆分成 lies on boundary的部分+正交於boundary的分量

Multi-class Classification

- The goal is to assign an input to one of the K classes
- Adopting the binary classification leads to the ambiguity
  - an one-versus-the-rest classifier
  - an one-versus-one classifier



the ambiguous region



Multi-class Classification

- To avoid the ambiguity, a single *K*-class discriminant comprising *K* linear functions of the form

$$\hat{y}_k(\boldsymbol{x}) = \boldsymbol{w}_k^T \boldsymbol{x} + w_{k,0}$$

for $k = 1,2,\cdots,K$

- An input $\boldsymbol{x}$ is assigned to class *j* if

$$\hat{y}_j(\boldsymbol{x}) > \hat{y}_k(\boldsymbol{x}) \quad \text{or} \quad j = \arg \max_{k=1,2,\cdots,K} \hat{y}_k(\boldsymbol{x})$$

for $k = 1,2,\cdots,K$ and $k \neq j$

Linear Models for Classification

Topic 02

Support Vector Machine

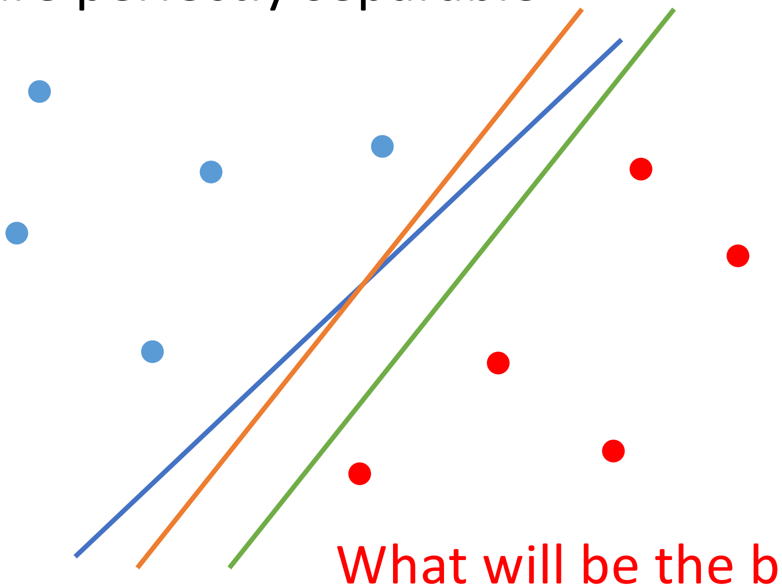
Separable Case - Part I

Linear model

Nonlinear model => "kernel"

The Separable Case

- We first assume that separable case in which the collected points of the two classes are perfectly separable



What will be the best decision boundary?

The Separable Case

- Recall the geometric margin,

$$\rho(\boldsymbol{x}) = \frac{|\hat{y}(\boldsymbol{x})|}{\|\boldsymbol{w}\|}$$

- The minimum geometric margin of $\hat{y}(\boldsymbol{x})$ given $\boldsymbol{w}$

$$\rho = \min_{n\in\{1,2,\cdots N\}} \frac{|\hat{y}(\boldsymbol{x}_n)|}{\|\boldsymbol{w}\|}$$

where $\boldsymbol{x}_n$ for $n = 1,2\cdots, N$ are the input points

[註解]

1. 模型：
＝ y_hat(x) = wx + w0，
是一個分隔面，將資料切分成兩區的平面。

2. minimum geometric margin：
在n個samples中，找到一個最小化y_hat(x)的sample，
白話一點就是找到一個sample離y_hat平面最近，
其計算出的值除以w的長度，
就是minimum geometric margin。

The Separable Case

- The safest way is to adjust $\boldsymbol{w}$ and w_0 so that ρ can be as large as possible

我們的主要目標是，將分隔平面(y_hat)離最近的資料越遠越好

- Given the $\boldsymbol{x}_n$ for $n = 1,2 \cdots, N$ and its corresponding target $y_i \in \{-1,1\}$, the maximum-margin optimization is

$$\rho_{max} = \max_{\boldsymbol{w},w_0} \rho = \max_{\boldsymbol{w},w_0} \min_{n} \frac{|\hat{y}(\boldsymbol{x}_n)|}{\|\boldsymbol{w}\|} = \max_{\boldsymbol{w},w_0} \min_{n} \frac{|\boldsymbol{w}^T\boldsymbol{x}_n + w_0|}{\|\boldsymbol{w}\|}$$

subject to

因此我們藉由調整w, w0，來最大化「離分隔面最近的資料」到「分隔面」的距離

$$y_n\hat{y}(\boldsymbol{x}_n) = y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 0$$

屬於 class = +1 類的資料點x，其 y_hat(x_n) > 0，其類別y_n = +1
屬於 class = -1 類的資料點則 y_hat(x) < 0，類別y_n = -1，

因此相乘必為正值，落在分隔面上則 = 0。

The Separable Case

- Note that $y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 0$ and $y_n \in \{-1,1\}$. This implies that

$$|\boldsymbol{w}^T\boldsymbol{x}_n + w_0| = y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0)$$

所以我們可以拿掉絕對值，用「所屬類別」乘以「模型預測值」來代替。

- The maximum-margin optimization becomes

$$\rho_{max} = \max_{\boldsymbol{w},w_0} \min_{n} \frac{y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0)}{\|\boldsymbol{w}\|}$$

The Separable Case

- Observing
  - when we make the rescaling $\boldsymbol{w} \rightarrow \kappa \boldsymbol{w}$ and $w_0 \rightarrow \kappa w_0$ for $\kappa > 0$

$$\frac{y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0)}{\|\boldsymbol{w}\|} = \frac{\kappa y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0)}{\kappa \|\boldsymbol{w}\|} = \frac{y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0)}{\|\boldsymbol{w}\|}$$

  - we can rescale $\boldsymbol{w}$ and w_0 such that

$$\min_{n=1,2\cdots,N} y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0) = 1$$

我們習慣將離分隔面最近的點之預測值設(rescale)為1

The Separable Case

- Based on this observation, the maximum-margin optimization can be rewritten as

由於最近的點的預測值的絕對值被定為1，
分子部分就能寫成1，我們僅需處理分母w的部分。

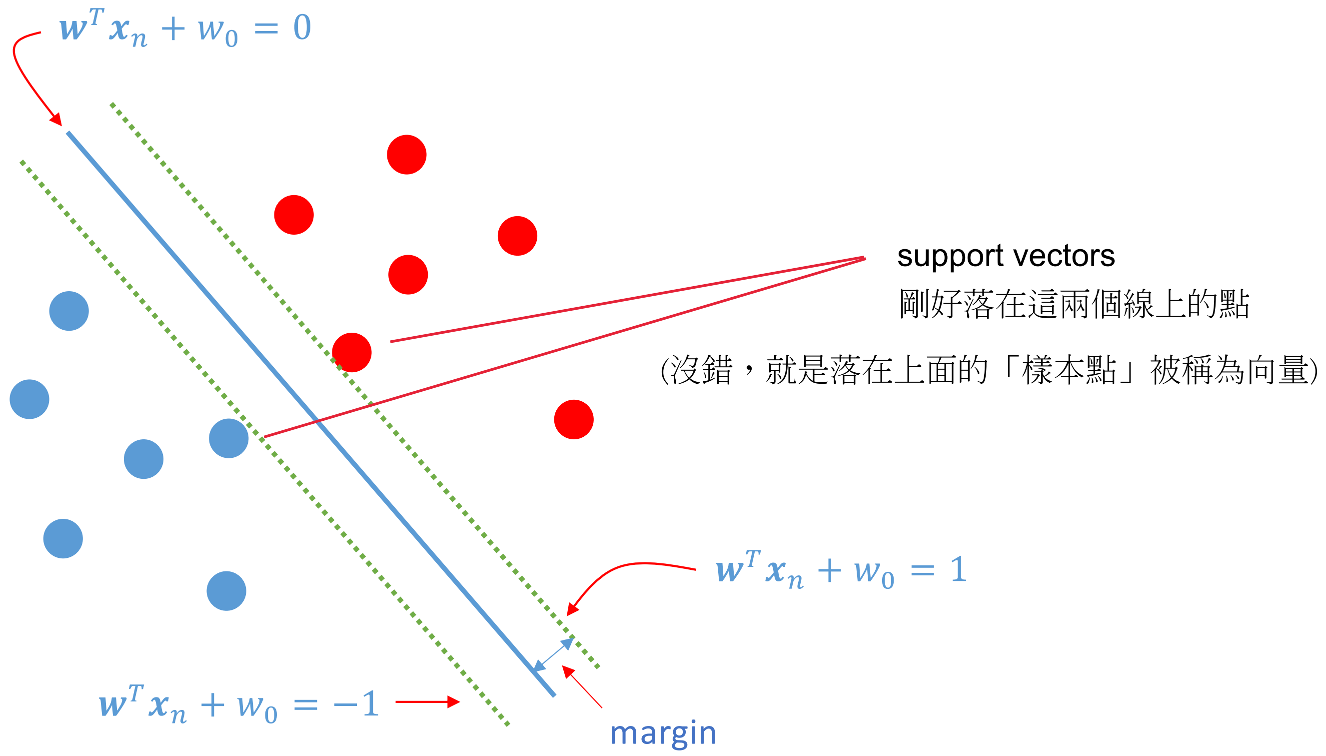
$$\max_{\boldsymbol{w},w_0} \frac{1}{||\boldsymbol{w}||}$$

such that $y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 1$ for $n = 1,2 \cdots, N$. This optimization problem is equivalent to

$$\min_{\boldsymbol{w},w_0} \frac{1}{2}||\boldsymbol{w}||^2$$

subject to $y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 1$

The Separable Case



The Separable Case

- This constraint optimization

$$\min_{\boldsymbol{w},w_0} \frac{1}{2}\|\boldsymbol{w}\|^2 \quad \text{subject to } y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 1$$

can be converted to

$$\min_{\boldsymbol{w},w_0} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n\left(1 - y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0)\right)$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \cdots, \alpha_N]^T$ and $\alpha_n \geq 0$ for $n = 1,2,\cdots,N$

The Separable Case

- Applying KKT conditions at the optimum, we have
  - $\frac{\partial}{\partial \boldsymbol{w}} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha}) = 0$

    求w

    $$w - \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n = 0$$

    此式很重要，後續會用來代換w

  - $\frac{\partial}{\partial w_0} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha}) = 0$

    求w0

    $$\sum_{n=1}^{N} -\alpha_n y_n = 0$$

  - complementary slackness

    $$\alpha_n(1 - y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0)) = 0$$

The Separable Case

- Discussion:
  - the complementary slackness condition:
    - when $\alpha_n = 0$, $1 - y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) > 0$
    - when $\alpha_n > 0$, $1 - y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) = 0$
  - the complementary slackness condition together with $\boldsymbol{w} = \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n$:
    - when $\alpha_n > 0$, $\boldsymbol{x}_n$ will contribute to $\boldsymbol{w}$ and we call such $\boldsymbol{x}_n$ the support vector
    - when $\boldsymbol{x}_n$ is the support vector, $1 - y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) = 0$ and $\boldsymbol{x}_n$ lies on the marginal hyperplanes, $\boldsymbol{w}^T\boldsymbol{x}_n + w_0 = 1$ or $\boldsymbol{w}^T\boldsymbol{x}_n + w_0 = -1$
  - $\min_{\boldsymbol{w}, w_0} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha})$ is a strict convex optimization problem of $\boldsymbol{w}$
    - $\frac{\partial}{\partial \boldsymbol{w} \partial \boldsymbol{w}} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha})$ is positive definite
    - $J(\boldsymbol{w}, w_0, \boldsymbol{\alpha})$ is strictly convex
    - the optimal $\boldsymbol{w}$ is unique

Linear Models for Classification

Topic 03

Support Vector Machine

Separable Case - Part II

The Separable Case

Primal Problem

- Recall

$$\min_{\boldsymbol{w},w_0} J(\boldsymbol{w},w_0,\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{n=1}^{N} \alpha_n\left(1 - y_n\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right)\right)$$

- KKT conditions

$$w = \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n \qquad \sum_{n=1}^{N} \alpha_n y_n = 0 \qquad \alpha_n\left(1 - y_n\left(\boldsymbol{w}^T\boldsymbol{x}_n + w_0\right)\right) = 0\ \forall n \in \{1,2,\cdots,N\}$$

The Separable Case

- The dual problem

$$\max_{\boldsymbol{\alpha}} \left\{ \min_{\boldsymbol{w}, w_0} J(\boldsymbol{w}, w_0, \boldsymbol{\alpha}) \right\}$$

subject to $\sum_{n=1}^{N} \alpha_n y_n = 0$ and $\alpha_n \geq 0$ for $n = 1, 2, \cdots, N$

The Separable Case

$$\min_{\boldsymbol{w},w_0} J(\boldsymbol{w},w_0,\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \sum_{n=1}^{N}\alpha_n\left(1 - y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0)\right)$$

$$= \frac{1}{2}\left\|\sum_{n=1}^{N}\alpha_n y_n \boldsymbol{x}_n\right\|^2 + \sum_{n=1}^{N}\alpha_n - \sum_{n=1}^{N}\alpha_n y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0)$$

$$= \frac{1}{2}\left\|\sum_{n=1}^{N}\alpha_n y_n \boldsymbol{x}_n\right\|^2 + \sum_{n=1}^{N}\alpha_n - \sum_{n=1}^{N}\alpha_n y_n\left(\sum_{n=1}^{N}\alpha_n y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_n - w_0\sum_{n=1}^{N}\alpha_n y_n$$

$$= \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n y_n\,\alpha_{n'} y_{n'} \boldsymbol{x}_{n'}^T \boldsymbol{x}_n + \sum_{n=1}^{N}\alpha_n - \sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n y_n\,\alpha_{n'} y_{n'} \boldsymbol{x}_{n'}^T \boldsymbol{x}_n$$

$$= -\frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n y_n\,\alpha_{n'} y_{n'} \boldsymbol{x}_{n'}^T \boldsymbol{x}_n + \sum_{n=1}^{N}\alpha_n$$

The Separable Case

Dual Problem

- The dual problem becomes

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N} \alpha_n y_n \alpha_{n'} y_{n'} \boldsymbol{x}_{n'}{}^T \boldsymbol{x}_n + \sum_{n=1}^{N} \alpha_n$$

subject to $\sum_{n=1}^{N} \alpha_n y_n = 0$ and $\alpha_n \geq 0$ for $n = 1,2,\cdots,N$

This is a quadratic programming

The Separable Case

- Let $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \cdots, \alpha_N^*]$ be the solution to the dual problem. The class assigned to an input $\boldsymbol{x}$ is

$$\operatorname{sgn}(\hat{y}(\boldsymbol{x})) = \operatorname{sgn}\left(\left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x} + w_0\right)$$

- w_0 can be obtained from any support vector via

$$w_0 = \frac{1}{y_i} - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i = y_i - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i$$

The Separable Case

- The fact that

$$w_0 = \frac{1}{y_i} - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i = y_i - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i$$

gives an interesting result

$$\rho^2 = \frac{1}{\|\boldsymbol{\alpha}^*\|_1}$$

which can be obtained by multiplying both sides by $\alpha_i^* y_i$ for $\alpha_i^* > 0$ and taking the sum

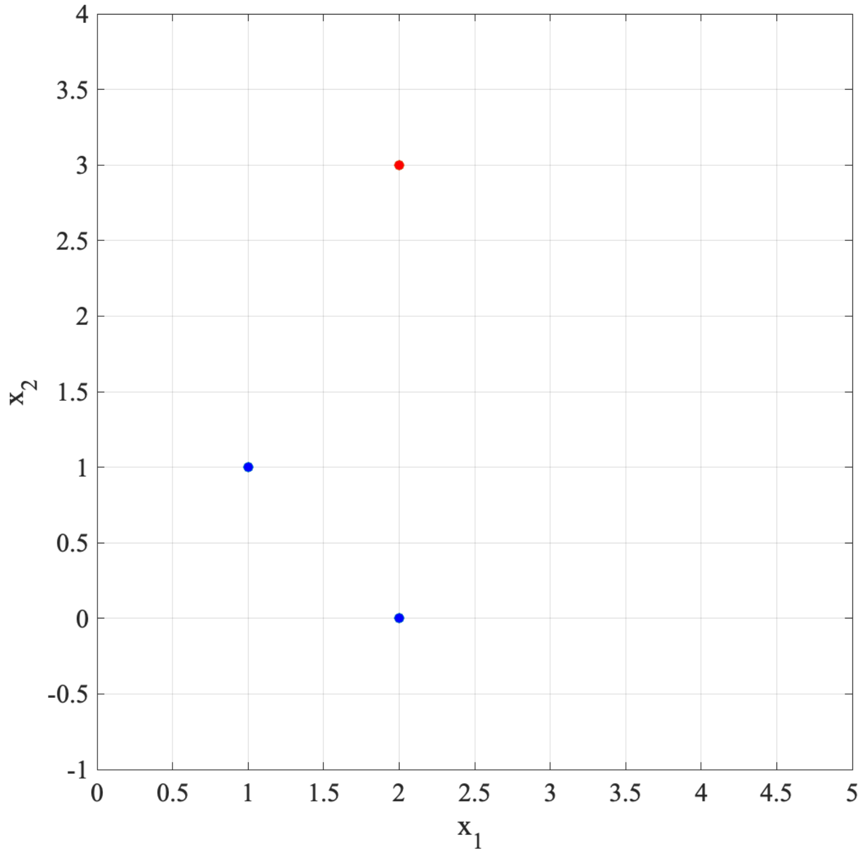
Example

- Class 1:

$$\boldsymbol{x}_1 = \begin{bmatrix}1\\1\end{bmatrix} \quad \boldsymbol{x}_2 = \begin{bmatrix}2\\0\end{bmatrix}$$

- Class 2:

$$\boldsymbol{x}_3 = \begin{bmatrix}2\\3\end{bmatrix}$$

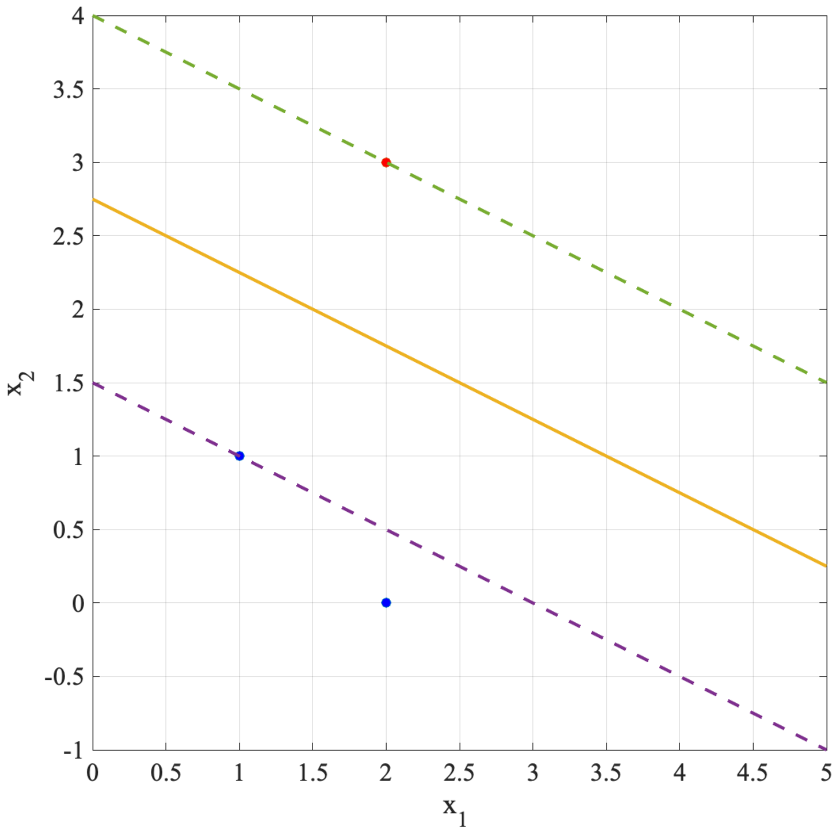


Example

- From the dual problem, using the algorithm for quadratic programming, we have $\alpha_1 = 0.8, \alpha_2 = 0, \alpha_3 = 0.8$
- $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $\begin{bmatrix} 2 \\ 3 \end{bmatrix}$ are support vectors
- This leads to $w_1 = 0.8$ and $w_2 = 1.6$. This indicates that $w_1 : w_2 = 1 : 2$. We let $\boldsymbol{w} = [a, 2a]^T$
- We know that

$$w^T \begin{bmatrix} 1 \\ 1 \end{bmatrix} + w_0 = -1 \text{ and } w^T \begin{bmatrix} 2 \\ 3 \end{bmatrix} + w_0 = 1 \Rightarrow a = 0.5, w_0 = -2.2$$

Example



$$\hat{y}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = 0.4x_1 + 0.8x_2 - 2.2$$

Linear Models for Classification

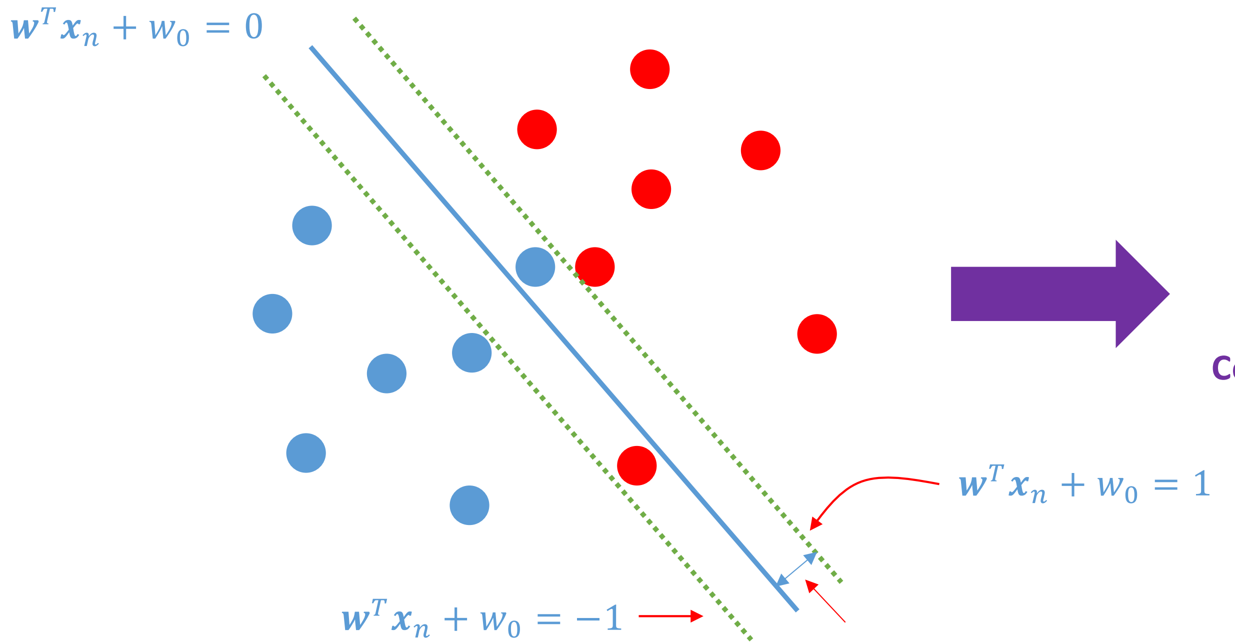
Topic 04

Support Vector Machine

Non-Separable Case

The Non-Separable Case

- In many cases, points of different classes are not always separable



$$y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \ngeq 1$$

Condition for being separable is violated

The Non-Separable Case

- Introducing the slack variables to relax the optimization problem in the separable case

$$y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 1 - \epsilon_n$$

where $\epsilon_n \geq 0$

  - ϵ_n measures the distance by which $\boldsymbol{x}_n$ violates $y_n(\boldsymbol{w}^T\boldsymbol{x}_n + w_0) \geq 1$
  - $\boldsymbol{x}_n$with $\epsilon_n > 0$ is considered as the outlier
  - those outlier with $0 < \epsilon_n < 1$ can be still be classified correctly.
  - for this reason, the margin ρ is called the soft margin as opposed to the hard margin in the separable case

The Non-Separable Case

- In the non-separable case, two conflicting objects are faced
  - we would like to limit the L_p-norm of these slack variables
    - results in fewer outliers
    - leads to the small margin
  - we would like to find the decision boundary with the margin as large as possible
    - causes more outliers
    - leads to the large L_p-norm of these slack variables

The Non-Separable Case

- Based on these two conflicting goals, the optimization becomes

$$\min_{\boldsymbol{w},w_0} \frac{1}{2}\|\boldsymbol{w}\|^2 + \gamma \sum_{n=1}^{N} \epsilon_n^q$$

subject to

$$y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0) \geq 1 - \epsilon_n, n = 1,2,\cdots,N$$

$$\epsilon_n \geq 0, n = 1,2,\cdots,N$$

The Non-Separable Case

- Introducing the Lagrange multipliers, we can convert the constrained optimization into unconstrained one
- When $q = 1$, The objective function becomes

$$J(\boldsymbol{w}, w_0, \boldsymbol{\epsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{w}\|^2 + \gamma \sum_{n=1}^{N} \epsilon_n + \sum_{n=1}^{N} \alpha_n \left(1 - \epsilon_n - y_n(\boldsymbol{w}^T \boldsymbol{x}_n + w_0)\right) - \sum_{n=1}^{N} \beta_n \epsilon_n$$

where $\boldsymbol{\epsilon} = [\epsilon_1, \cdots, \epsilon_N], \boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_N], \boldsymbol{\beta} = [\beta_1, \cdots, \alpha_N]$

$$\alpha_i \geq 0, \beta_i \geq 0, i = 1,2, \cdots, N$$

Primal Problem

The Non-Separable Case

- Applying KKT conditions at the optimum, we have
  - $\frac{\partial}{\partial \boldsymbol{w}} J(\boldsymbol{w}, w_0, \boldsymbol{\epsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta})$

$$w - \sum_{n=1}^{N} \alpha_n y_n \boldsymbol{x}_n = 0$$

  - $\frac{\partial}{\partial w_0} J(\boldsymbol{w}, w_0, \boldsymbol{\epsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\sum_{n=1}^{N} -\alpha_n y_n = 0$$

  - $\frac{\partial}{\partial \epsilon_i} J(\boldsymbol{w}, w_0, \boldsymbol{\epsilon}, \boldsymbol{\alpha}, \boldsymbol{\beta})$

$$\gamma - \alpha_i - \beta_i = 0$$

The Non-Separable Case

- Applying KKT conditions at the optimum, we have
  - complementary slackness

$$\alpha_n(1-\epsilon_n-y_n(\boldsymbol{w}^T\boldsymbol{x}_n+w_0))=0, n=1,2\cdots,N$$

$$\beta_n\epsilon_n=0, n=1,2\cdots,N$$

The Non-Separable Case

Dual Problem

- The dual problem becomes

$$\max_{\boldsymbol{\alpha}} -\frac{1}{2}\sum_{n=1}^{\boldsymbol{N}}\sum_{n'=1}^{\boldsymbol{N}} \alpha_n y_n \alpha_{n'} y_{n'} \boldsymbol{x}_{n'}{}^T \boldsymbol{x}_n + \sum_{n=1}^{\boldsymbol{N}} \alpha_n$$

subject to $\sum_{n=1}^{\boldsymbol{N}} \alpha_n y_n = 0$ and $0 \le \alpha_n \le \gamma$ for $n = 1,2,\cdots,N$

This is the same as the separable case!!
(They only differ in the constraint)

The Non-Separable Case

- Let $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \cdots, \alpha_N^*]$ be the solution to the dual problem. The class assigned to an input $\boldsymbol{x}$ is

$$\operatorname{sgn}(\hat{y}(\boldsymbol{x})) = \operatorname{sgn}\left(\left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x} + w_0\right)$$

- w_0 can be obtained from any support vector lying on a marginal hyperplane via

$$w_0 = \frac{1}{y_i} - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i = y_i - \left(\sum_{n=1}^{N} \alpha_n^* y_n \boldsymbol{x}_n\right)^T \boldsymbol{x}_i$$

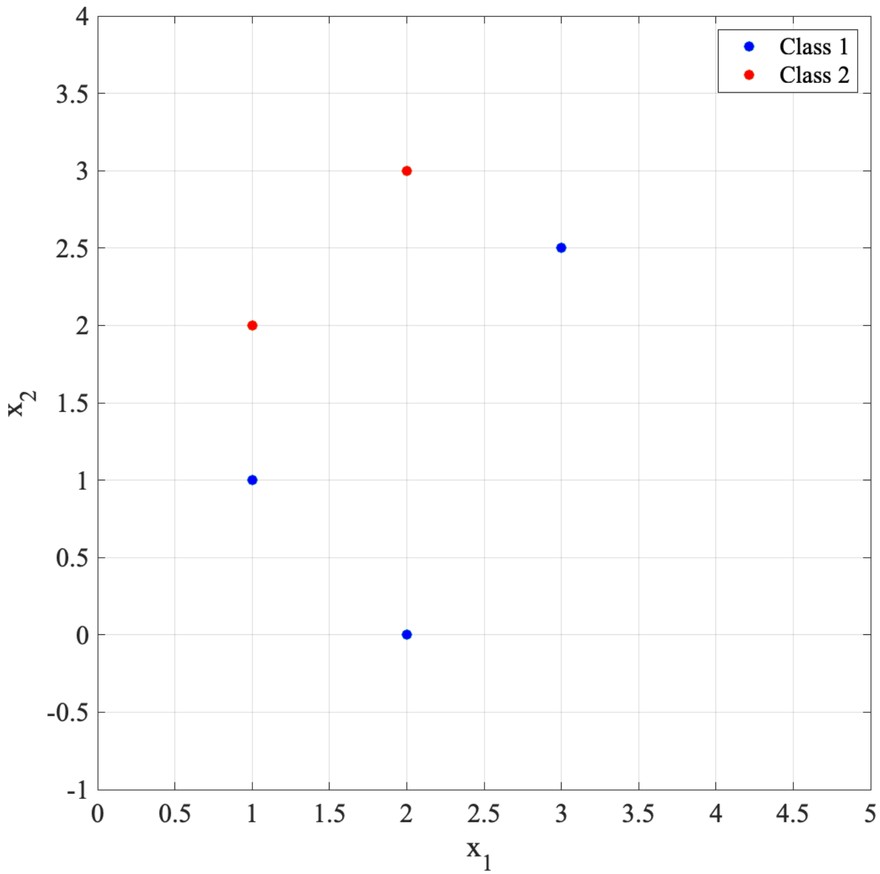
Example

- Class 1:

$$\boldsymbol{x}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \boldsymbol{x}_2 = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad \boldsymbol{x}_3 = \begin{bmatrix} 3 \\ 2.5 \end{bmatrix}$$

- Class 2:

$$\boldsymbol{x}_4 = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad \boldsymbol{x}_5 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

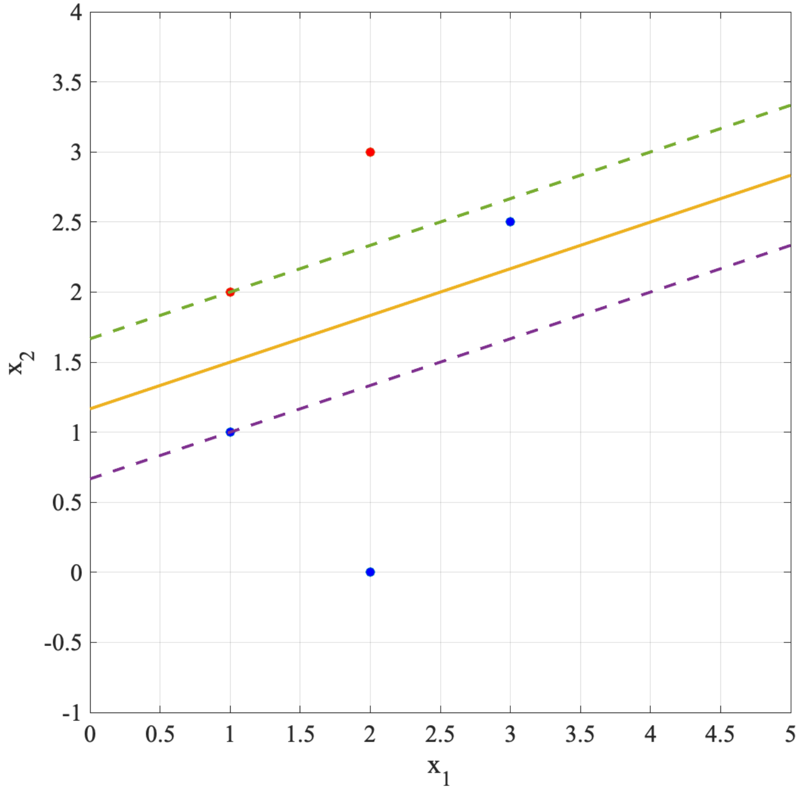


Example

- From the dual problem, using the algorithm for quadratic programming, when $\gamma = 1$, we have $\alpha_1 = 0.16, \alpha_2 = 0.16, \alpha_3 = 0, \alpha_4 = 0$ and $\alpha_5 = 0.32$
- $\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}2\\0\end{bmatrix}$ and $\begin{bmatrix}1\\2\end{bmatrix}$ are support vectors
- This leads to $w_1 = -0.16$ and $w_2 = 0.48$. This indicates that $w_1 : w_2 = -1:3$. We let $\boldsymbol{w} = [-a, 3a]^T$
- We know that

$$w^T \begin{bmatrix}1\\1\end{bmatrix} + w_0 = -1 \text{ and } w^T \begin{bmatrix}2\\3\end{bmatrix} + w_0 = 1 \Rightarrow a = \frac{2}{3}, w_0 = -1$$

The Non-Separable Case



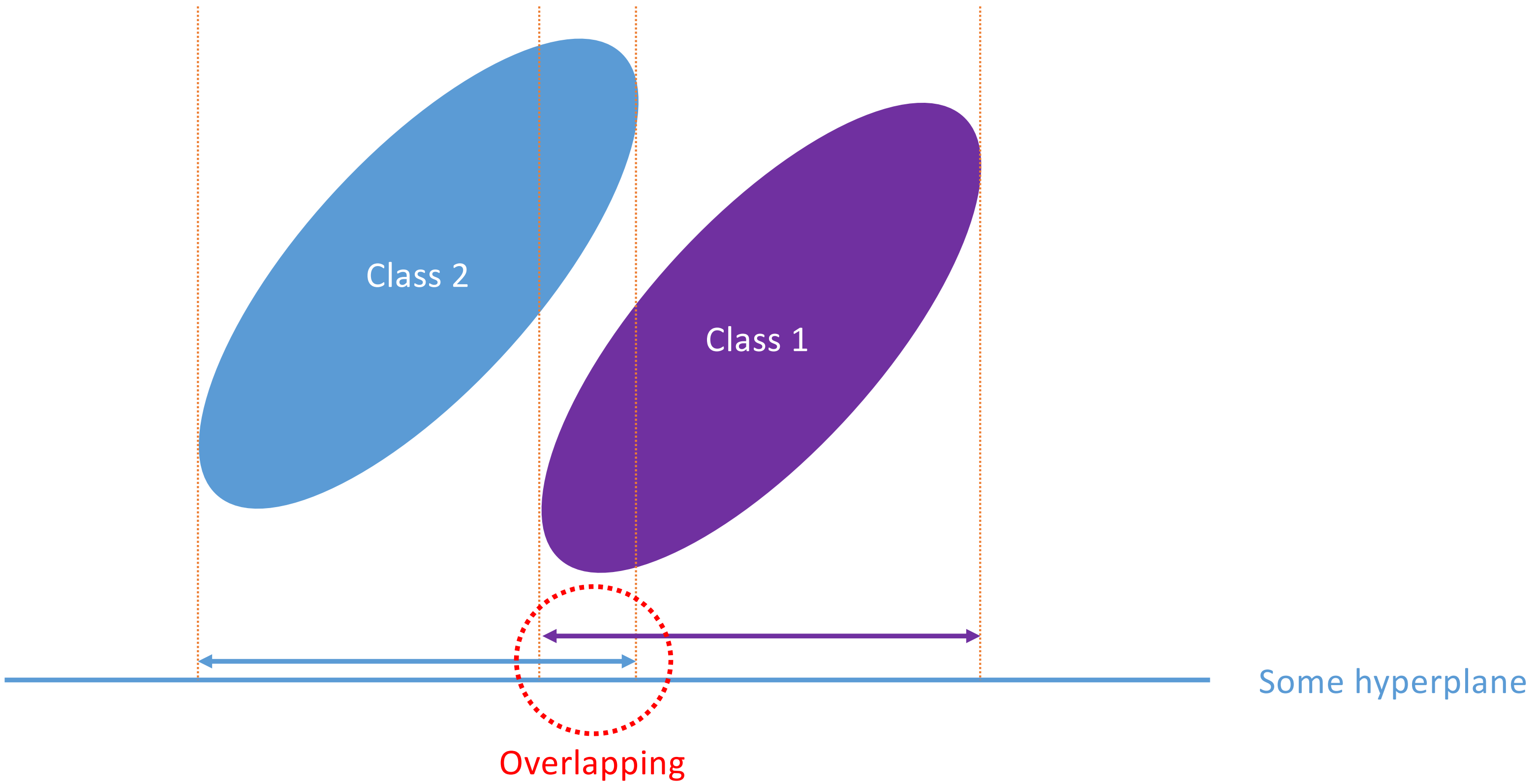
$$\hat{y}\left(\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}\right) = -\frac{2}{3}x_1 + \frac{4}{3}x_2 - 1$$

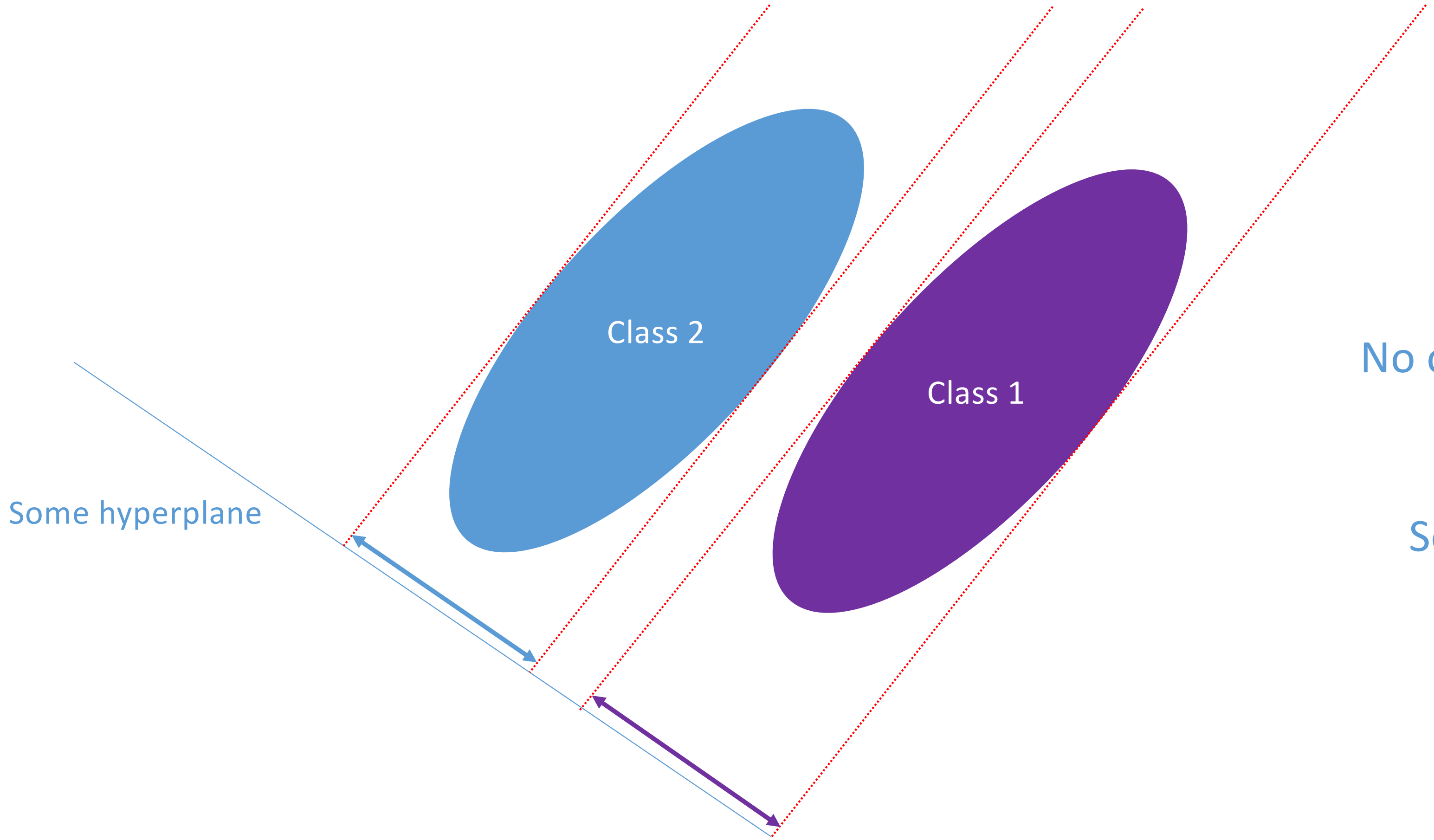
Linear Models for Classification

Topic 05

Fisher’s Linear Discriminant

- One way to view a linear classification model is in terms of dimensionality reduction
- The idea is to project the data point onto a hyperplane so that points of different classes can be separated as possible


Class 2
Class 1
Some hyperplane
Overlapping



No overlapping

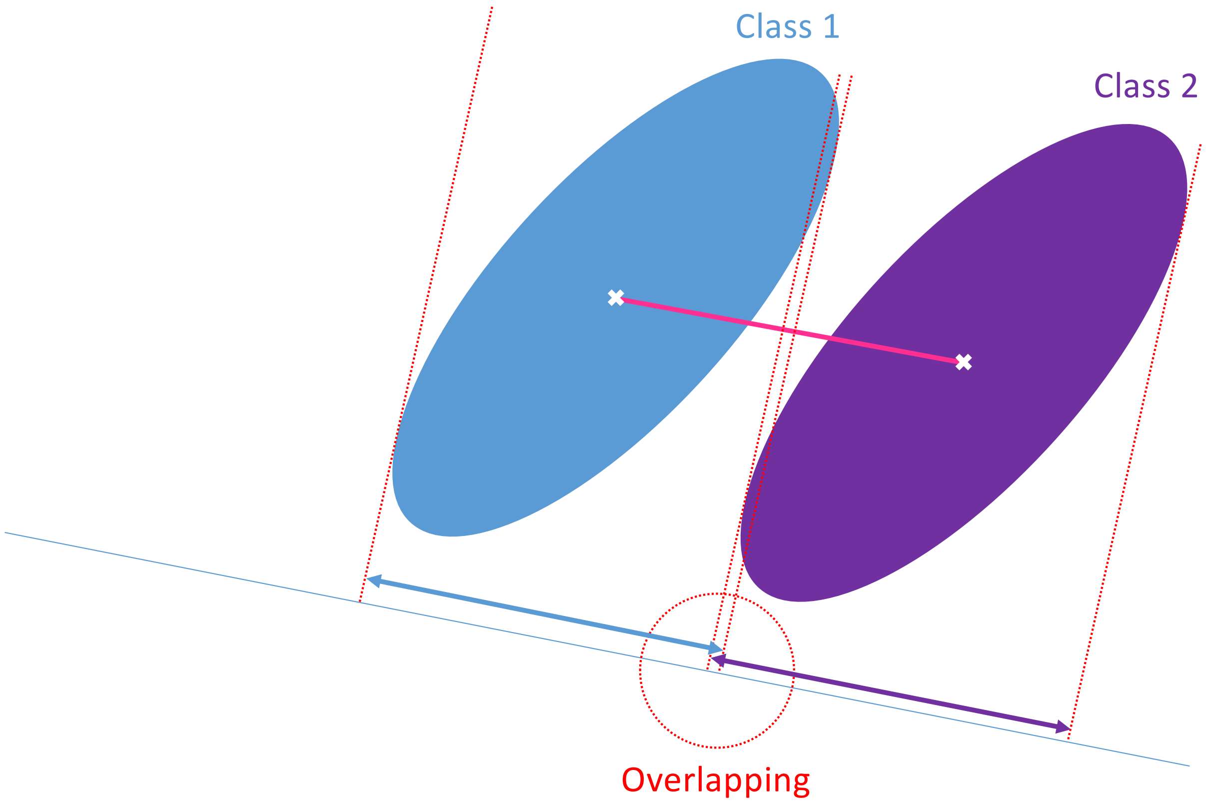
Separable

- The goal is to find a projecting vector $\boldsymbol{w}$ such that these points of these two classes to maximally separate these two classes
- The simplest measure of separation is the the separation of the means of two classes after the projection
- Thus, the goal can be

$$\boldsymbol{w} = \arg \max_{\boldsymbol{w},\, \|\boldsymbol{w}\|=1} m_2^p(\boldsymbol{w}) - m_1^p(\boldsymbol{w})$$

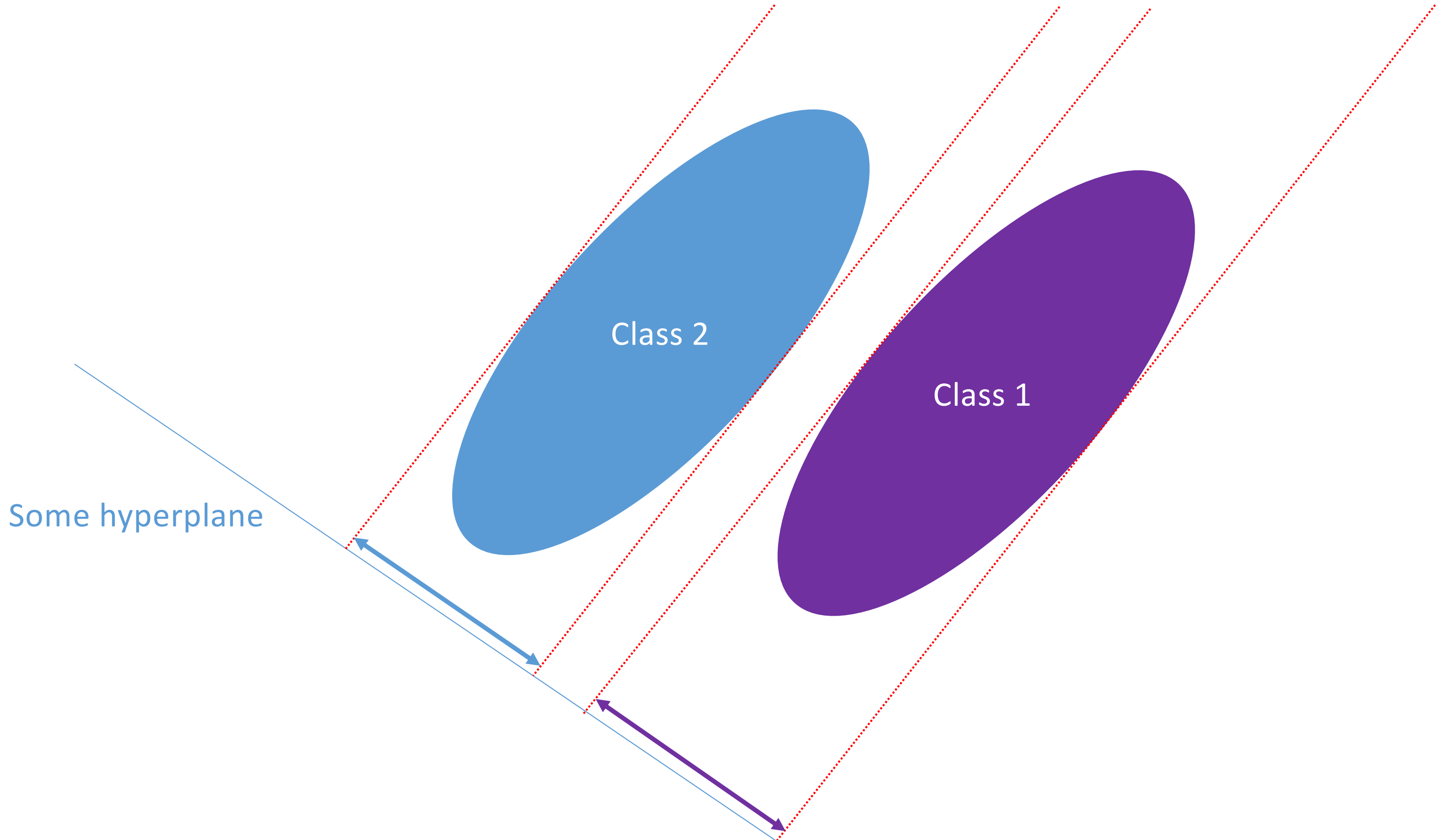
where $m_k^p(w) = \boldsymbol{w}^T \boldsymbol{m}_k$ and $\boldsymbol{m}_k$ is the mean vector of class C_k for $k = 1,2$

$$\boldsymbol{m}_k = \frac{1}{N_k} \sum_{\boldsymbol{x}_i \in C_k} \boldsymbol{x}_i$$



Maximal separation

Still overlapping



字幕預留位置

- The idea of Fisher's Linear Discriminant is to maximize the separation as possible while keeping the dispersion of data points after projection as small as possible
- The goal of Fisher's Linear Discriminant

$$\max_{\boldsymbol{w}} J(\boldsymbol{w}) = \frac{\left(m_2^p(\boldsymbol{w}) - m_1^p(\boldsymbol{w})\right)^2}{s_1^2(\boldsymbol{w}) + s_2^2(\boldsymbol{w})} = \frac{\boldsymbol{w}^T \boldsymbol{S}_B \boldsymbol{w}}{\boldsymbol{w}^T \boldsymbol{S}_w \boldsymbol{w}}$$

Rayleigh quotient

The between-class covariance matrix $\boldsymbol{S}_B = (\boldsymbol{m}_2 - \boldsymbol{m}_1)(\boldsymbol{m}_2 - \boldsymbol{m}_1)^{\boldsymbol{T}}$

The total within-class covariance matrix $\boldsymbol{S}_w = \sum_{n \in C_1} (\boldsymbol{x}_n - \boldsymbol{m}_1)(\boldsymbol{x}_n - \boldsymbol{m}_1)^{\boldsymbol{T}} + \sum_{n \in C_2} (\boldsymbol{x}_n - \boldsymbol{m}_2)(\boldsymbol{x}_n - \boldsymbol{m}_2)^{\boldsymbol{T}}$

- $J(\boldsymbol{w})$ is maximized when

$$(\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w})\boldsymbol{S}_w\boldsymbol{w} = (\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w})\boldsymbol{S}_B\boldsymbol{w} \rightarrow J(\boldsymbol{w})\boldsymbol{S}_w\boldsymbol{w} = \boxed{\boldsymbol{S}_B\boldsymbol{w}} \rightarrow \lambda\boldsymbol{w} = \boldsymbol{S}_w^{-1}\boldsymbol{S}_B\boldsymbol{w}$$

the generalized eigenvalue problem

always in the direction of $(\boldsymbol{m}_2 - \boldsymbol{m}_1)$

- Let us look at $\boldsymbol{S}_B\boldsymbol{w}$ carefully

$$\boldsymbol{S}_B\boldsymbol{w} = (\boldsymbol{m}_2 - \boldsymbol{m}_1)(\boldsymbol{m}_2 - \boldsymbol{m}_1)^T\,\boldsymbol{w} = \alpha(\boldsymbol{m}_2 - \boldsymbol{m}_1)$$

  - $\boldsymbol{S}_B\boldsymbol{w}$ is always in the same direction of $(\boldsymbol{m}_2 - \boldsymbol{m}_1)$

- Since both $\boldsymbol{w}^T\boldsymbol{S}_B\boldsymbol{w}$ and $\boldsymbol{w}^T\boldsymbol{S}_w\boldsymbol{w}$ are scalars, we have

$$\boldsymbol{S}_w\boldsymbol{w} \propto \boldsymbol{S}_B\boldsymbol{w} = \alpha(\boldsymbol{m}_2 - \boldsymbol{m}_1) \quad \Rightarrow \quad \boldsymbol{w} \propto \boxed{\boldsymbol{S}_w^{-1}(\boldsymbol{m}_2 - \boldsymbol{m}_1)}$$

Fisher's linear discriminant

- After the projecting vector $\boldsymbol{w}$ is determined, the decision rule can be

$$\hat{y}(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{x} \underset{C_2}{\overset{C_1}{\gtreqless}} \gamma_{th}$$

Example – Gaussian data

- Class 1:

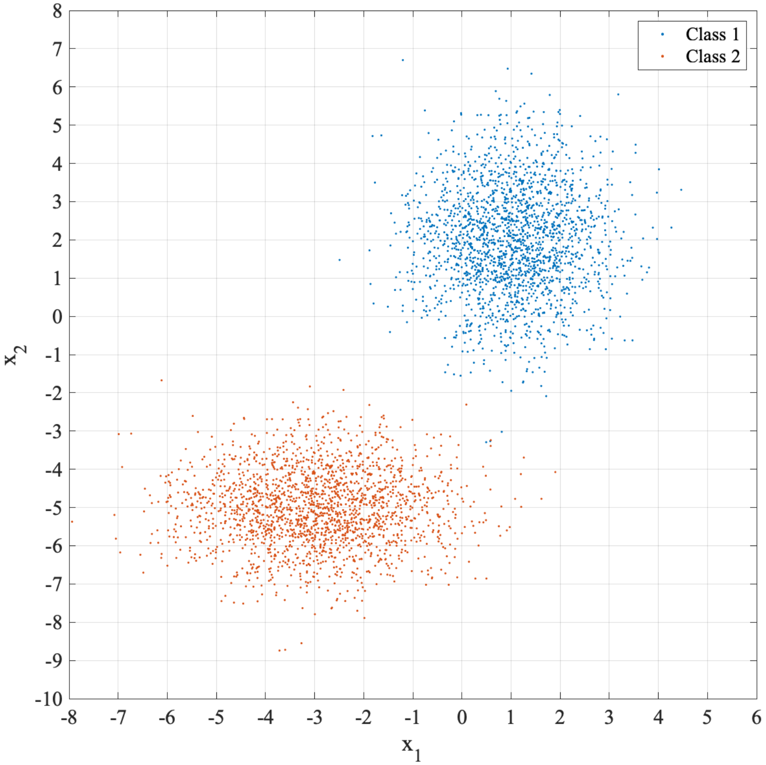
$$m_1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad K_1 = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}$$

- Class 2:

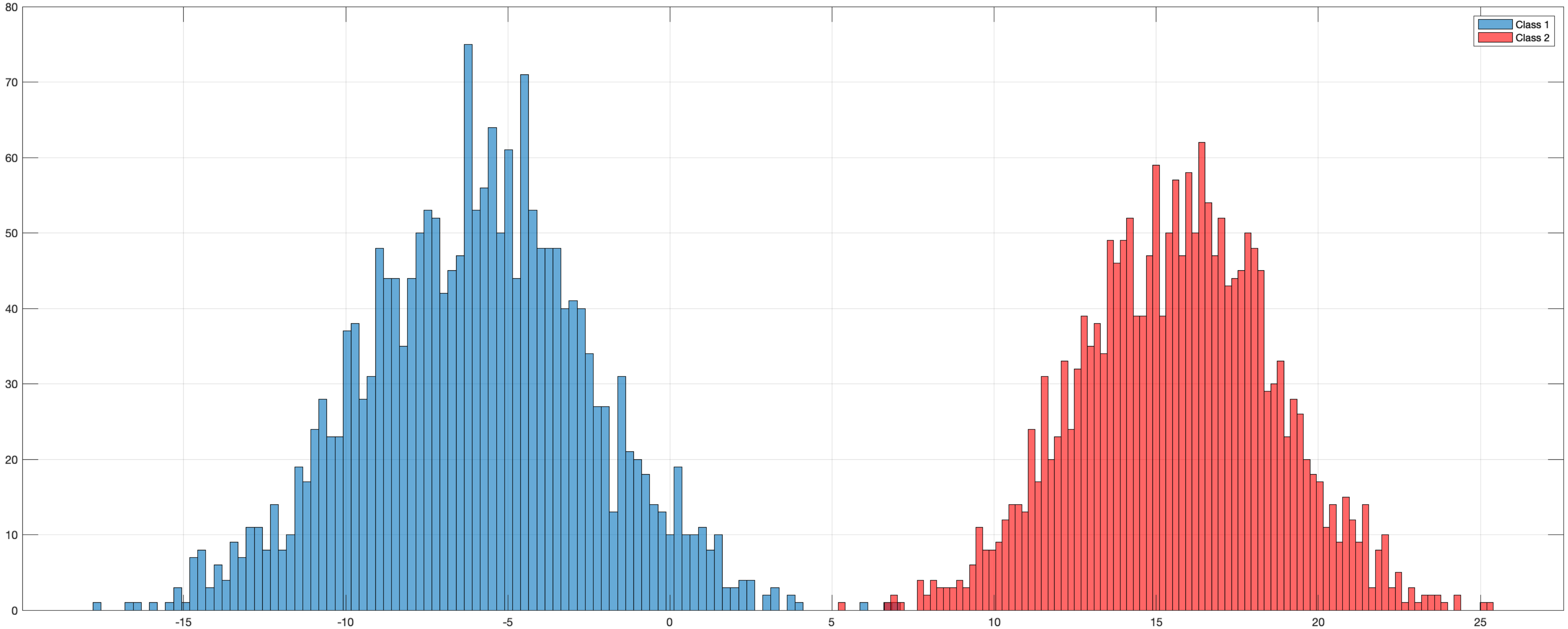
$$m_2 = \begin{bmatrix} -3 \\ -5 \end{bmatrix} \quad K_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}$$

- The projecting vector

$$w = S_w^{-1}(m_2 - m_1) = (K_2 + K_1)^{-1}(m_2 - m_1) = \begin{bmatrix} -4/3 \\ -7/3 \end{bmatrix}$$



Example – Gaussian data



After projection

字幕預留位置

Fisher's Discriminant for Multi-class Classification

- Assume that K classes are present and we would like to classify the targets into one of these classes in D-dimensional space, where $D > K$
- Now we introduce D' "features"

$$\hat{y}_i(\boldsymbol{x}) = \boldsymbol{w}_i^T \boldsymbol{x}, \qquad \text{for } i = 1,2,\cdots,D'$$

Dimension Reduction

$$\widehat{\boldsymbol{Y}} = \boldsymbol{W}^T \boldsymbol{x}$$

$$\boldsymbol{W} = [\boldsymbol{w}_1\ \boldsymbol{w}_2 \cdots \boldsymbol{w}_{D'}]$$

Fisher’s Discriminant for Multi-class Classification

- The within-class covariance matrix

$$\boldsymbol{S}_W = \sum_{k=1}^{K} \boldsymbol{S}_k \qquad \boldsymbol{S}_k = \sum_{n \in C_k} (\boldsymbol{x}_n - \boldsymbol{m}_k)(\boldsymbol{x}_n - \boldsymbol{m}_k)^T$$

$$\boldsymbol{m}_k = \frac{1}{N_k} \sum_{n \in C_k} \boldsymbol{x}_n$$

- The between-class covariance

$$\boldsymbol{S}_B = \sum_{k=1}^{K} N_k (\boldsymbol{m}_k - \boldsymbol{m})(\boldsymbol{m}_k - \boldsymbol{m})^{\mathrm{T}} \quad \boldsymbol{m} = \sum_{n=1}^{N} \boldsymbol{x}_n$$

- The total covariance matrix

$$\boldsymbol{S}_T = \boldsymbol{S}_W + \boldsymbol{S}_B$$

Fisher's Discriminant for Multi-class Classification

- The goal is similar to the two-class classification is to seek an optimal $\boldsymbol{W}$ to maximize

$$J(\boldsymbol{W}) = tr\{(\boldsymbol{W}^T \boldsymbol{S}_W \boldsymbol{W})^{-1}(\boldsymbol{W}^T \boldsymbol{S}_B \boldsymbol{W})\}$$

- The D' projecting vectors are those eigenvectors corresponding to the D' largest eigenvalues of

$$\boldsymbol{S}_W^{-1} \boldsymbol{S}_B$$

Q&A