# Chapter 4 Linear Models

LECTURER: ZHIHUA JIANG

# Content

# *3.1 Linear model*

- Linear model: linear function of attributes

$$f(\boldsymbol{x}) = w_1 x_1 + w_2 x_2 + \ldots + w_d x_d + b$$

- Matrix form: $f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$

$$\Big\Downarrow \quad \boldsymbol{\beta} = (\boldsymbol{w}; b), \quad \hat{\boldsymbol{x}} = (\boldsymbol{x}; 1)$$

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \longrightarrow \boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}$$

- Advantages:

simple model; basic model; good interpretability

# 3.2 Linear regression

- **Univariate linear regression**

$$f(x_i) = wx_i + b$$ such that $f(x_i) \approx y_i$

*Where $x_i$ is a scalar*

- **Multivariate linear regression**

$$f(\boldsymbol{x}_i) = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b$$ such that $f(x_i) \approx y_i$

*Where $x_i$ is a vector*

- **Generalized linear model**

$$y = g^{-1}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b)$$

*Where $g(\cdot)$ is a monotone differentiable function*

# *Univariate linear regression*

- How to determine *w* and *b*?
  a) To minimize the *MSE*

$$
\begin{aligned}
(w^*, b^*) &= \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} \left(f(x_i) - y_i\right)^2 \\
&= \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} (y_i - w x_i - b)^2 \ .
\end{aligned}
$$

# *Univariate linear regression*

- How to determine *w* and *b*? (cont.)

  b) Parameter estimation based on least square method

$$E_{(w,b)} = \sum_{i=1}^{m}(y_i - wx_i - b)^2$$

  - Differentiate with *w* and *b*, respective

$$\frac{\partial E_{(w,b)}}{\partial w} = 2\left(w\sum_{i=1}^{m}x_i^2 - \sum_{i=1}^{m}(y_i - b)\,x_i\right),$$

$$\frac{\partial E_{(w,b)}}{\partial b} = 2\left(mb - \sum_{i=1}^{m}(y_i - wx_i)\right),$$

$$\boxed{w = \frac{\sum_{i=1}^{m} y_i(x_i - \bar{x})}{\sum_{i=1}^{m} x_i^2 - \frac{1}{m}\left(\sum_{i=1}^{m} x_i\right)^2}, \qquad b = \frac{1}{m}\sum_{i=1}^{m}(y_i - wx_i)}$$

# Multivariate linear regression

- Given $D = \{(x_1, y_1), \ldots, (x_m, y_m)\}$,

$x_i = (x_{i1}; \ldots; x_{id}), y_i \in \mathbb{R} \rightarrow$ $m \times (d+1)$ matrix $X$

$$\hat{\boldsymbol{w}} = (\boldsymbol{w}; b)$$

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1d} & 1 \\ x_{21} & x_{22} & \ldots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \ldots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} & 1 \\ \boldsymbol{x}_2^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathrm{T}} & 1 \end{pmatrix}$$

$$\boldsymbol{y} = (y_1; y_2; \ldots; y_m)$$

# *Multivariate linear regression*

- Similarly, to minimize the MSE

$$\hat{\boldsymbol{w}}^* = \underset{\hat{\boldsymbol{w}}}{\arg\min}\, (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})^{\mathrm{T}} (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})$$

$$E_{\hat{\boldsymbol{w}}} = (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})^{\mathrm{T}} (\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}})$$

- Differentiate with $\hat{\boldsymbol{w}}$

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\,\mathbf{X}^{\mathrm{T}} (\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}) \xrightarrow[\text{positive definite matrix}]{\text{full-rank matrix, or}} \hat{\boldsymbol{w}}^* = (\mathbf{X}^{\mathrm{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathrm{T}}\boldsymbol{y}$$

# *Generalized linear model*

- Problem: how to let the linear prediction to approximate  some function of real labels?

$$y = g^{-1}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b)$$

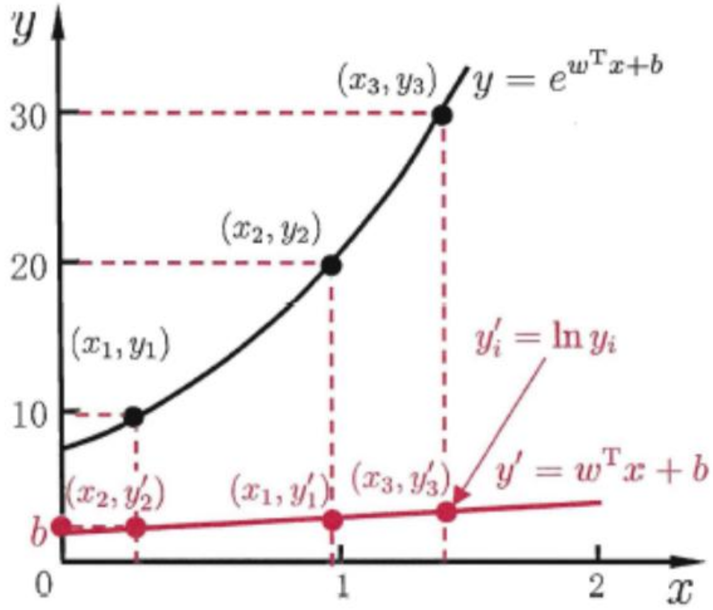
*Where g(·) is* the link function (*monotone & differentiable*),

*$g^{-1}(\cdot)$ is the inverse function*

- Example: lc ssion (对数线性回归)

$$\ln y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \Rightarrow y = e^{W^T X + b}$$

# Example: log-linear regression

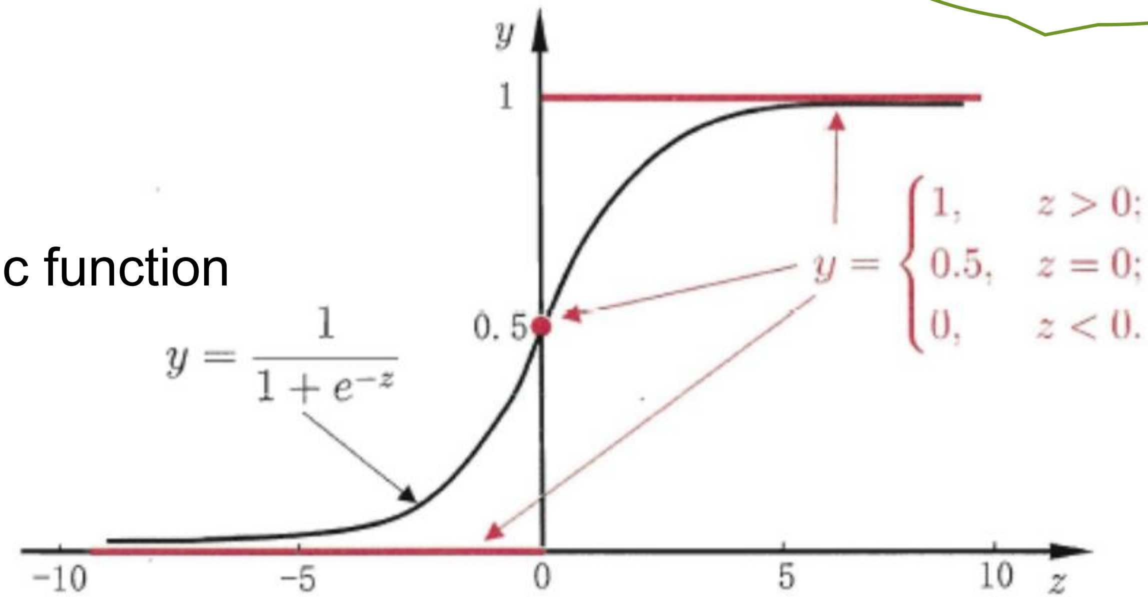


# *3.3 logistic regression*

- Problem: use the linear regression for the classification problem?
- Solution: *Generalized linear model*
- Candidates of $g(\cdot)$:
  a) Unit-step function (单位阶跃函数)
  b) Logistic function (对数几率函数)

Class label

Unit-step function




Logistic function

Linear prediction

# Logistic function

- Odds(几率)

  y: probability of x being a positive example

  1-y: probability of x being a negative example

  y/(1-y): relative probability of x being a positive example

- Logistic function: any order differentiable
  x fu nathematic
  ties

$$y = \frac{1}{1+e^{-z}} \quad \Rightarrow \quad y = \frac{1}{1+e^{-(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b)}} \quad \Rightarrow \quad \ln\frac{y}{1-y} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b$$

# Logistic function

- Posterior probability estimation

$$\ln \frac{y}{1-y} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \quad \Longrightarrow \quad \ln \frac{p(y=1 \mid \boldsymbol{x})}{p(y=0 \mid \boldsymbol{x})} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b .$$

$$p(y=1 \mid \boldsymbol{x}) = \frac{e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}} ,$$

$$p(y=0 \mid \boldsymbol{x}) = \frac{1}{1+e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}} .$$

$\boxed{?}$ $\Longleftarrow$ maximum likelihood method

$$\prod_{i=1}^{m} p(y_i \mid x_i; w, b)$$

# Logistic function

- Log likelihood function:

$$\ell(\boldsymbol{w}, b) = \sum_{i=1}^{m} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}, b)$$

$$\boldsymbol{\beta} = (\boldsymbol{w}; b), \quad \hat{\boldsymbol{x}} = (\boldsymbol{x}; 1)$$

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \longrightarrow \boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}.$$

$$p_1(\hat{\boldsymbol{x}}; \boldsymbol{\beta}) = p(y = 1 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}),$$

$$p_0(\hat{\boldsymbol{x}}; \boldsymbol{\beta}) = p(y = 0 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}) = 1 - p_1(\hat{\boldsymbol{x}}; \boldsymbol{\beta})$$

$$\Downarrow$$

$$p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}, b) = y_i p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta})$$

# Logistic function

- Likelihood function (cont.)

$$p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}, b) = y_i p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta})$$

$$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b \longrightarrow \boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}$$

Max $l(w;b) \leftrightarrow$ Min $l(\beta)$

$$\boxed{\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left( -y_i \boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i + \ln \left( 1 + e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i} \right) \right)}$$

Newton method (update for the *t*+1 iteration)

$$\boldsymbol{\beta}^{t+1} = \boldsymbol{\beta}^{t} - \left( \frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}\, \partial \boldsymbol{\beta}^{\mathrm{T}}} \right)^{-1} \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}$$

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left( -y_i \boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i + \ln\left(1 + e^{\boldsymbol{\beta}^{\mathrm{T}} \hat{\boldsymbol{x}}_i}\right) \right)$$

Max $l(w,b) \leftrightarrow$ Min $l(\beta)$

$$l(w,b) = \sum_{i=1}^{m} \ln\ p(y_i \mid x_i; w, b)$$

$$= \sum_{i=1}^{m} \ln[y_i p_1(\hat{x}_i; \beta) + (1 - y_i) p_0(\hat{x}_i; \beta)]$$

$$= \sum_{i=1}^{m} \ln[y_i \frac{e^{\beta^T \hat{x}_i}}{1 + e^{\beta^T \hat{x}_i}} + (1 - y_i) \frac{1}{1 + e^{\beta^T \hat{x}_i}}]$$

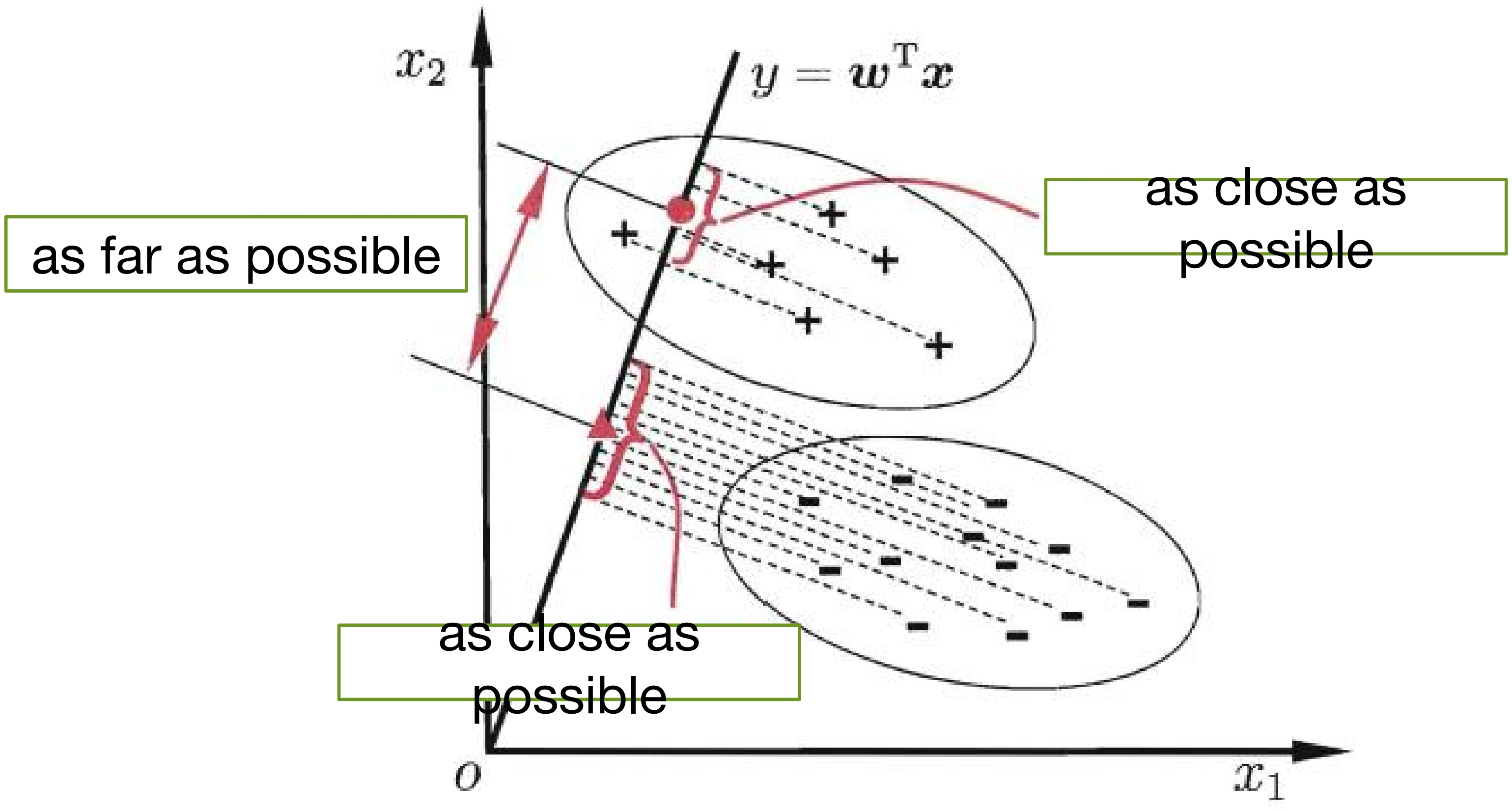
$$= \sum_{i=1}^{m} \ln \frac{y_i e^{\beta^T \hat{x}_i} + (1 - y_i)}{1 + e^{\beta^T \hat{x}_i}}$$

$$= \begin{cases} y_i = 1, & \sum_{i=1}^{m} \ln \frac{e^{\beta^T \hat{x}_i}}{1 + e^{\beta^T \hat{x}_i}} = \boxed{\sum_{i=1}^{m} (\beta^T \hat{x}_i - \ln(1 + e^{\beta^T \hat{x}_i}))} \\ y_i = 0, & \sum_{i=1}^{m} \ln \frac{1}{1 + e^{\beta^T \hat{x}_i}} = \boxed{\sum_{i=1}^{m} (-\ln(1 + e^{\beta^T \hat{x}_i}))} \end{cases}$$

# 3.4 Linear discriminant analysis (LDA)

- Idea:
  - Cast the samples onto a straight line
  - Project the similar samples as close as possible
  - Project the dissimilar samples as far as possible
  - For a new sample, determine the class according to the relative position of its projection point.

# 3.4 Linear discriminant analysis (LDA)



# Goal of LDA

- Given dataset $D = \{(x_i, y_i)\}_{i=1}^{m}, y_i \in \{0,1\}$
  - $X_i$: sample set of the $i^{th}$ class
  - $\mu_i$: mean vector of the $i^{th}$ class
  - $\Sigma_i$: covariance matrix of the $i^{th}$ class
  - Projection points of the two centers in the line: $w^T \mu_0$ and $w^T \mu_1$
  - Covariance: $w^T\Sigma_0 w$ and $w^T\Sigma_1 w$
- Similar samples as close as possible → $w^T\Sigma_0 w + w^T\Sigma_1 w$ as small as possible
- Dissimilar samples as far as possible → $\|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1\|_2^2$ as large as possible

$$\|\boldsymbol{x}\|_2 := \sqrt{x_1^2 + \cdots + x_n^2}.$$

If the entries in the column vector

$$\mathbf{X} = \begin{bmatrix} X_1 \\ \vdots \\ X_n \end{bmatrix}$$

are random variables, each with finite variance, then the covariance matrix Σ is the matrix whose (*i*, *j*) entry is the covariance

$$\Sigma_{ij} = \mathrm{cov}(X_i, X_j) = \mathrm{E}[(X_i - \mu_i)(X_j - \mu_j)] = \mathrm{E}[X_i X_j] - \mu_i \mu_j,$$

where the operator E denotes the expected (mean) value of its argument, and

$$\mu_i = \mathrm{E}(X_i)$$

is the expected value of the *i*-th entry in the vector **X**. In other words,

$$\Sigma = \begin{bmatrix} \mathrm{E}[(X_1 - \mu_1)(X_1 - \mu_1)] & \mathrm{E}[(X_1 - \mu_1)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_1 - \mu_1)(X_n - \mu_n)] \\ \\ \mathrm{E}[(X_2 - \mu_2)(X_1 - \mu_1)] & \mathrm{E}[(X_2 - \mu_2)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_2 - \mu_2)(X_n - \mu_n)] \\ \\ \vdots & \vdots & \ddots & \vdots \\ \\ \mathrm{E}[(X_n - \mu_n)(X_1 - \mu_1)] & \mathrm{E}[(X_n - \mu_n)(X_2 - \mu_2)] & \cdots & \mathrm{E}[(X_n - \mu_n)(X_n - \mu_n)] \end{bmatrix}.$$

# 3.4 Goal of LDA

- Goal: maximize

$$J = \frac{\|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1\|_2^2}{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}} = \frac{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}(\boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1)\boldsymbol{w}}$$

$$\Downarrow$$

$$J = \frac{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}}$$

- Within-class scatter matrix

$$\mathbf{S}_w = \boldsymbol{\Sigma}_0 + \boldsymbol{\Sigma}_1 = \sum_{\boldsymbol{x}\in X_0} (\boldsymbol{x} - \boldsymbol{\mu}_0)(\boldsymbol{x} - \boldsymbol{\mu}_0)^{\mathrm{T}} + \sum_{\boldsymbol{x}\in X_1} (\boldsymbol{x} - \boldsymbol{\mu}_1)(\boldsymbol{x} - \boldsymbol{\mu}_1)^{\mathrm{T}}$$

- Between-class scatter matrix

$$\mathbf{S}_b = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}$$

# 3.4 Goal of LDA

- How to solve *w*?

$$L(w, \lambda) = -w^T S_b w + \lambda(w^T S_w w - 1)$$

$$\frac{\partial L}{\partial w} = -2S_b w + 2\lambda S_w w = 0 \Rightarrow S_b w = \lambda S_w w$$

$$J = \frac{\boldsymbol{w}^{\mathrm{T}} \mathbf{S}_b \boldsymbol{w}}{\boldsymbol{w}^{\mathrm{T}} \mathbf{S}_w \boldsymbol{w}} \xrightarrow{\text{set } \boldsymbol{w}^{\mathrm{T}} \mathbf{S}_w \boldsymbol{w} = 1} \begin{aligned} \min_{\boldsymbol{w}} \quad & -\boldsymbol{w}^{\mathrm{T}} \mathbf{S}_b \boldsymbol{w} \\ \text{s.t.} \quad & \boldsymbol{w}^{\mathrm{T}} \mathbf{S}_w \boldsymbol{w} = 1 \end{aligned} \xrightarrow{\text{Lagrange multipliers}} \mathbf{S}_b \boldsymbol{w} = \lambda \mathbf{S}_w \boldsymbol{w}$$

[推导]：由公式 (3.36) 可得拉格朗日函数为

$$L(\boldsymbol{w},\lambda) = -\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w} + \lambda(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} - 1)$$

对 $\boldsymbol{w}$ 求偏导可得

$$\begin{aligned}\frac{\partial L(\boldsymbol{w},\lambda)}{\partial \boldsymbol{w}} &= -\frac{\partial(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w})}{\partial \boldsymbol{w}} + \lambda\frac{\partial(\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w} - 1)}{\partial \boldsymbol{w}} \\ &= -(\mathbf{S}_b + \mathbf{S}_b^{\mathrm{T}})\boldsymbol{w} + \lambda(\mathbf{S}_w + \mathbf{S}_w^{\mathrm{T}})\boldsymbol{w}\end{aligned}$$

由于 $\mathbf{S}_b = \mathbf{S}_b^{\mathrm{T}}, \mathbf{S}_w = \mathbf{S}_w^{\mathrm{T}}$，所以

$$\frac{\partial L(\boldsymbol{w},\lambda)}{\partial \boldsymbol{w}} = -2\mathbf{S}_b\boldsymbol{w} + 2\lambda\mathbf{S}_w\boldsymbol{w}$$

令上式等于 0 即可得

$$-2\mathbf{S}_b\boldsymbol{w} + 2\lambda\mathbf{S}_w\boldsymbol{w} = 0$$

$$\mathbf{S}_b\boldsymbol{w} = \lambda\mathbf{S}_w\boldsymbol{w}$$

由于我们想要求解的只有 $\boldsymbol{w}$，而 $\lambda$ 这个拉格朗乘子具体取值多少都无所谓，因此我们可以任意设定 $\lambda$ 来配合我们求解 $\boldsymbol{w}$。我们注意到

$$\mathbf{S}_b\boldsymbol{w} = (\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w}$$

如果我们令 $\lambda$ 恒等于 $(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)^{\mathrm{T}}\boldsymbol{w}$，那么上式即可改写为

$$\mathbf{S}_b\boldsymbol{w} = \lambda(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

将其代入 $\mathbf{S}_b\boldsymbol{w} = \lambda\mathbf{S}_w\boldsymbol{w}$ 即可解得

$$\boldsymbol{w} = \mathbf{S}_w^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$

# Lagrange multipliers (拉格朗日乘子法)

- Idea:

  $d$ variables, $k$ constraints → $d$+$k$ variables, 0 constraint

- Goal:

  $x$ is $d$-vector, minimize $f(x)$ s.t. $g(x)$=0

- n:

$$\boxed{L(\boldsymbol{x},\lambda)=f(\boldsymbol{x})+\lambda g(\boldsymbol{x})}, \xrightarrow{\text{Set partial derivative to be 0}} \begin{cases} \nabla f(\boldsymbol{x}^*)+\lambda\nabla g(\boldsymbol{x}^*)=0 \\ g(\boldsymbol{x})=0. \end{cases}$$