

---

# 第三章：线性模型

---

---

# 目录

---

- 线性回归
  - 最小二乘法
  
- 二分类任务
  - 对数几率回归
  - 线性判别分析
  
- 多分类任务
  - 一对一
  - 一对其余
  - 多对多
  
- 类别不平衡问题

# 基本形式

---

## □ 线性模型一般形式

$$f(\mathbf{x}) = w_1x_1 + w_2x_2 + \dots + w_dx_d + b$$

$\mathbf{x} = (x_1; x_2; \dots; x_d)$ 是由属性描述的示例，其中 $x_i$ 是 $\mathbf{x}$ 在第  $i$ 个属性上的取值

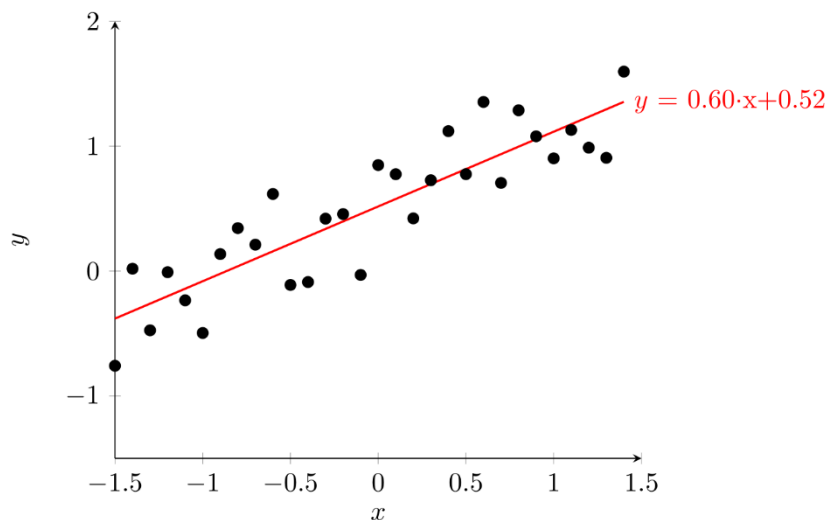
## □ 向量形式

$$f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + b$$

其中  $\mathbf{w} = (w_1; w_2; \dots; w_d)$

# 线性模型优点

- 形式简单、易于建模
- 可解释性
- 非线性模型的基础
  - 引入层级结构或高维映射



- 一个例子
  - 综合考虑色泽、根蒂和敲声来判断西瓜好不好
  - 其中根蒂的系数最大，表明根蒂最要紧；而敲声的系数比色泽大，说明敲声比色泽更重要

$$f_{\text{好瓜}}(x) = 0.2 \cdot x_{\text{色泽}} + 0.5 \cdot x_{\text{根蒂}} + 0.3 \cdot x_{\text{敲声}} + 1$$

# 线性回归

---

□ 给定数据集  $D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$

其中  $\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id})$   $y_i \in \mathbb{R}$

□ 线性回归 (**linear regression**) 目的

- 学得一个线性模型以尽可能准确地预测实值输出标记

$$f(\mathbf{x}_i) \simeq y_i$$

□ 离散属性处理

- 有“序”关系
  - 连续化为连续值
- 无“序”关系
  - 有 $k$ 个属性值，则转换为 $k$ 维向量

# 线性回归

---

## □ 单一属性的线性回归目标

$$f(x) = wx_i + b \quad \text{使得} \quad f(x_i) \simeq y_i$$

## □ 参数/模型估计：最小二乘法 (**least square method**)

$$\begin{aligned} (w^*, b^*) &= \arg \min_{(w, b)} \sum_{i=1}^m (f(x_i) - y_i)^2 \\ &= \arg \min_{(w, b)} \sum_{i=1}^m (y_i - wx_i - b)^2 \end{aligned}$$

# 线性回归 - 最小二乘法

## □ 最小化均方误差

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

## □ 分别对 $w$ 和 $b$ 求导, 可得

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

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

# 线性回归 - 最小二乘法

□ 得到闭式 (**closed-form**) 解

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

$$\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$$b = \frac{1}{m} \sum_{i=1}^m (y_i - wx_i)$$

其中

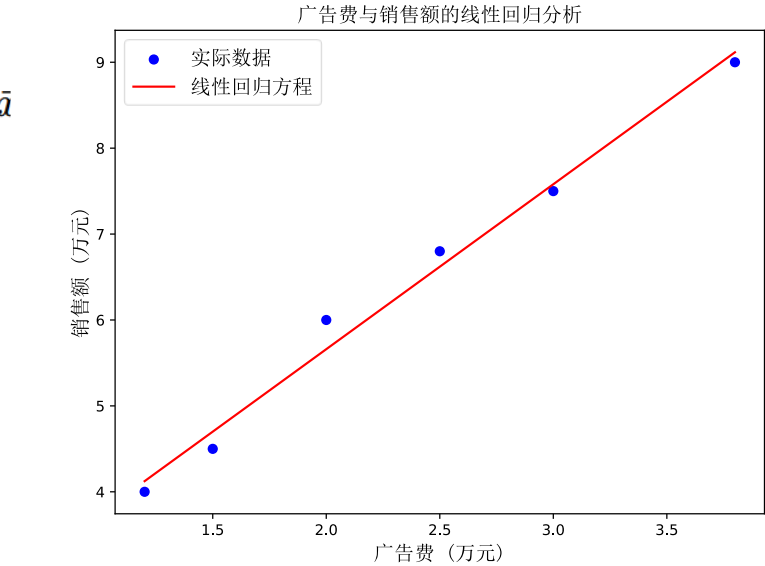
$$\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i$$



# 线性回归 - 最小二乘法

广告费与销售额的线性回归分析（利用最小二乘法建立销售额(y)与广告费(x)的线性回归方程，并预测广告费为3.5万元时的销售额）

广告费(x)	1.2	1.5	2.0	2.5	3.0	3.8
销售额(y)	4.0	4.5	6.0	6.8	7.5	9.0



$$= 2.3333$$

$$\frac{8}{-} = 6.3$$

$$wx_i)$$

$$\approx 1.82$$

$$w = \frac{\sum_{i=1}^m y_i(x_i - \bar{x})}{\sum_{i=1}^m x_i^2 - \frac{1}{m} (\sum_{i=1}^m x_i)^2}$$

$$w = \frac{97.25 - 6 \times 2.3333 \times 6.3}{37.38 - \frac{1}{6} \times (14.0)^2}$$

$$w = \frac{9.05}{4.71} \approx 1.92$$

$y = 1.92x + 1.82$  $y = 1.92 \times 3.5 + 1.82 = 8.54$

# 多元线性回归

---

## □ 给定数据集

$$D = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_m, y_m)\}$$

$$\mathbf{x}_i = (x_{i1}; x_{i2}; \dots; x_{id}) \quad y_i \in \mathbb{R}$$

## □ 多元线性回归目标

$$f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i + b \quad \text{使得} \quad f(\mathbf{x}_i) \simeq y_i$$

# 多元线性回归

□ 把  $w$  和  $b$  吸收入向量形式  $\hat{w} = (w; b)$ , 数据集表示为

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

$$\mathbf{y} = (y_1; y_2; \dots; y_m)$$

# 多元线性回归 - 最小二乘法

---

□ 最小二乘法 (least square method)

$$\hat{\boldsymbol{w}}^* = \arg \min_{\hat{\boldsymbol{w}}} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})^T (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})$$

令  $E_{\hat{\boldsymbol{w}}} = (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})^T (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{w}})$  , 对 $\hat{\boldsymbol{w}}$ 求导得到

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\boldsymbol{X}^T (\boldsymbol{X}\hat{\boldsymbol{w}} - \boldsymbol{y})$$

令上式为零可得 $\hat{\boldsymbol{w}}$ 最优解的闭式解

# 多元线性回归 – 满秩讨论

□  $\mathbf{X}^T \mathbf{X}$  是满秩矩阵或正定矩阵, 则

$$\hat{w}^* = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

其中  $(\mathbf{X}^T \mathbf{X})^{-1}$  是  $\mathbf{X}^T \mathbf{X}$  的逆矩阵, 线性回归模型为

$$f(\hat{x}_i) = \hat{x}_i^T (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y}$$

□  $\mathbf{X}^T \mathbf{X}$  不是满秩矩阵

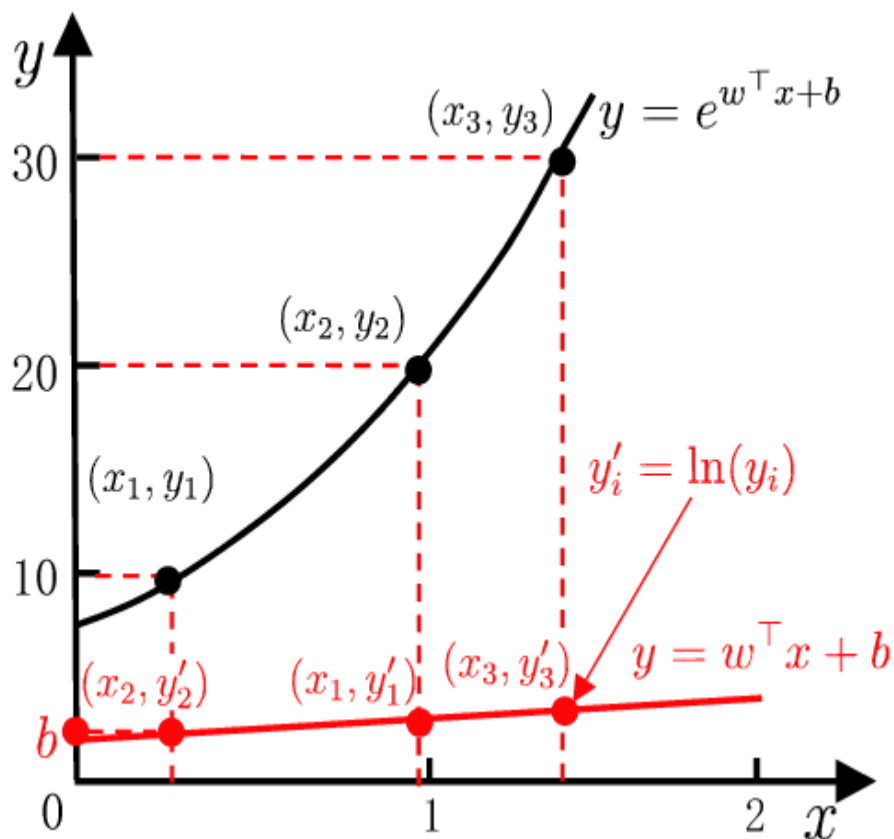
- 根据归纳偏好选择解 (参见1.4节)
- 引入正则化 (参加6.4节, 11.4节)

Ridge:  $\min_w ||Xw - y||_2^2 + \alpha ||w||_2^2$

Lasso:  $\min_w \frac{1}{2n_{\text{samples}}} ||Xw - y||_2^2 + \alpha ||w||_1$

# 对数线性回归

□ 输出标记的对数为线性模型逼近的目标



$$\ln y = \mathbf{w}^T \mathbf{x} + b$$



$$y = \mathbf{w}^T \mathbf{x} + b$$

# 线性回归 - 广义线性模型

---

## □ 一般形式

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

## □ $g(\cdot)$ 称为联系函数 (**link function**)

- 单调可微函数

## □ 对数线性回归是 $g(\cdot) = \ln(\cdot)$ 时广义线性模型的特例

# 二分类任务

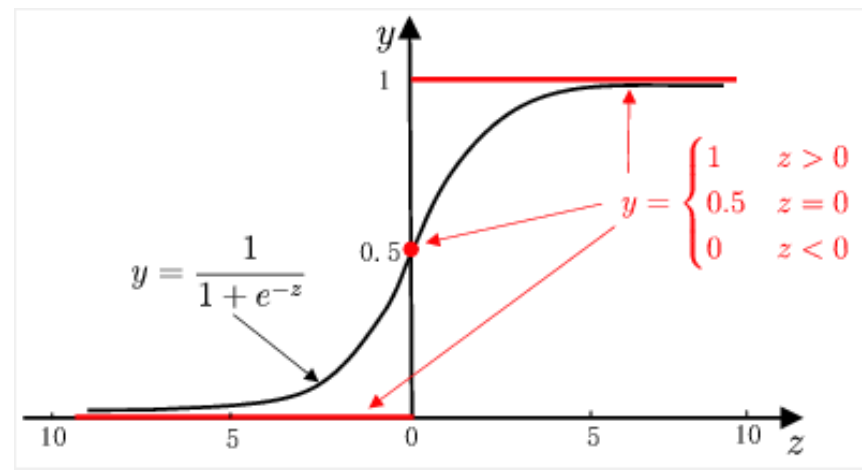
## □ 预测值与输出标记

$$z = \mathbf{w}^T \mathbf{x} + b \quad y \in \{0, 1\}$$

## □ 寻找函数将分类标记与线性回归模型输出联系起来

## □ 最理想的函数——单位阶跃函数

$$y = \begin{cases} 0, & z < 0; \\ 0.5, & z = 0; \\ 1, & z > 0, \end{cases}$$



- 预测值大于零就判为正例，小于零就判为反例，预测值为临界值零则可任意判别



# 二分类任务

## □ 单位阶跃函数缺点

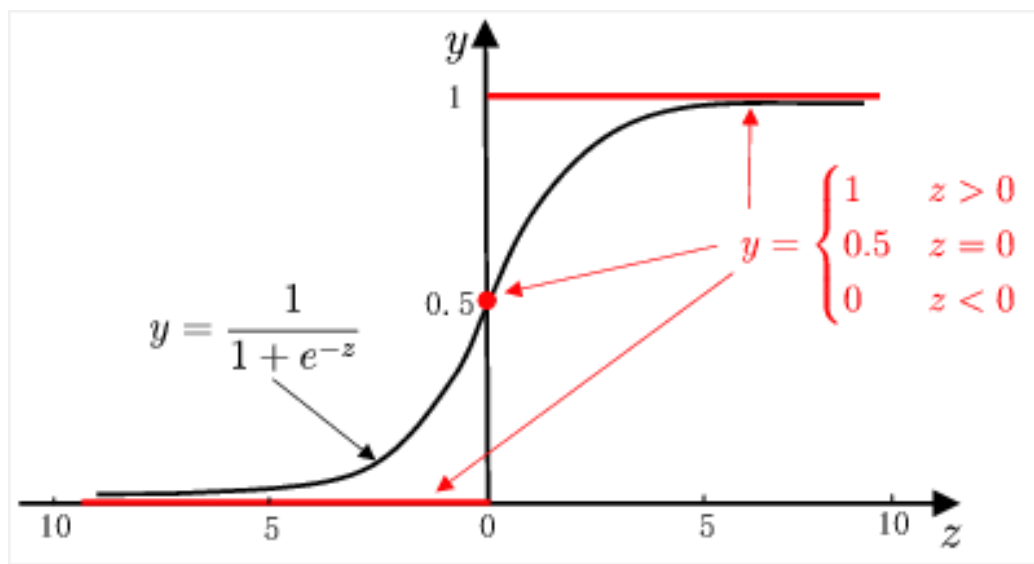
- 不连续

## □ 替代函数——对数几率函数 (**logistic function**)

- 单调可微、任意阶可导

单位阶跃函数与对数几率函数的比较

$$y = \frac{1}{1 + e^{-z}}$$



# 对数几率回归

## □ 运用对数几率函数

$$y = \frac{1}{1 + e^{-z}} \quad \text{变为} \quad y = \frac{1}{1 + e^{-(w^T x + b)}}$$

## □ 对数几率 (log odds)

- 样本作为正例的相对可能性的对数

$$\ln \frac{y}{1 - y}$$

## □ 对数几率回归优点

- 无需事先假设数据分布
- 可得到“类别”的近似概率预测
- 可直接应用现有数值优化算法求取最优解

# 对数几率回归 - 极大似然法

---

## □ 对数几率

$$\ln \frac{p(y = 1 \mid \mathbf{x})}{p(y = 0 \mid \mathbf{x})} = \mathbf{w}^T \mathbf{x} + b$$

显然有

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

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

# 对数几率回归 - 极大似然法

---

## □ 极大似然法 (**maximum likelihood**)

- 给定数据集

$$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^m$$

- 最大化样本属于其真实标记的概率
  - 最大化对数似然函数

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

# 对数几率回归 - 极大似然法

□ 转化为最小化负对数似然函数求解

- 令  $\beta = (w; b)$ ,  $\hat{x} = (x; 1)$ , 则  $w^T x + b$  可简写为  $\beta^T \hat{x}$

- 再令

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

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

则似然项可重写为

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

- 故等价形式为要最小化

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

# 对数几率回归 - 极大似然法

$$\begin{aligned}\ell(\boldsymbol{\beta}) &= \sum_{i=1}^m \ln ([p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta})]^{y_i} [p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta})]^{1-y_i}) \\&= \sum_{i=1}^m [y_i \ln (p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta})) + (1 - y_i) \ln (p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta}))] \\&= \sum_{i=1}^m \{y_i [\ln (p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta})) - \ln (p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta}))] + \ln (p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta}))\} \\&= \sum_{i=1}^m \left[ y_i \ln \left( \frac{p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta})}{p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta})} \right) + \ln (p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta})) \right] \\&= \sum_{i=1}^m \left[ y_i \ln \left( e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i} \right) + \ln \left( \frac{1}{1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i}} \right) \right] \\&= \sum_{i=1}^m \left( y_i \boldsymbol{\beta}^T \hat{\mathbf{x}}_i - \ln(1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i}) \right)\end{aligned}$$

# 对数几率回归

□ 求解得

$$\beta^* = \arg \min_{\beta} \ell(\beta)$$

□ 牛顿法第 $t+1$ 轮迭代解的更新公式

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

其中关于  $\beta$  的一阶、二阶导数分别为

$$\frac{\partial \ell(\beta)}{\partial \beta} = - \sum_{i=1}^m \hat{\mathbf{x}}_i (y_i - p_1(\hat{\mathbf{x}}_i; \beta))$$

$$\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} = \sum_{i=1}^m \hat{\mathbf{x}}_i \hat{\mathbf{x}}_i^T p_1(\hat{\mathbf{x}}_i; \beta) (1 - p_1(\hat{\mathbf{x}}_i; \beta))$$

高阶可导连续凸函数，梯度下降法/牛顿法 [Boyd and Vandenberghe, 2004]

# 机器学习的三要素

## □ 模型

- 线性方法:  $f(\mathbf{x}, \theta) = \mathbf{w}^T \mathbf{x} + b$
- 广义线性方法:  $f(\mathbf{x}, \theta) = \mathbf{w}^T \phi(\mathbf{x}) + b$ 
  - 如果 $\phi(\mathbf{x})$ 为可学习的非线性基函数,  $f(\mathbf{x}, \theta)$ 就等价于神经网络。

## □ 学习准则

- 期望风险

$$\mathcal{R}(f) = \mathbb{E}_{(\mathbf{x}, y) \sim p(\mathbf{x}, y)} [\mathcal{L}(f(\mathbf{x}), y)],$$

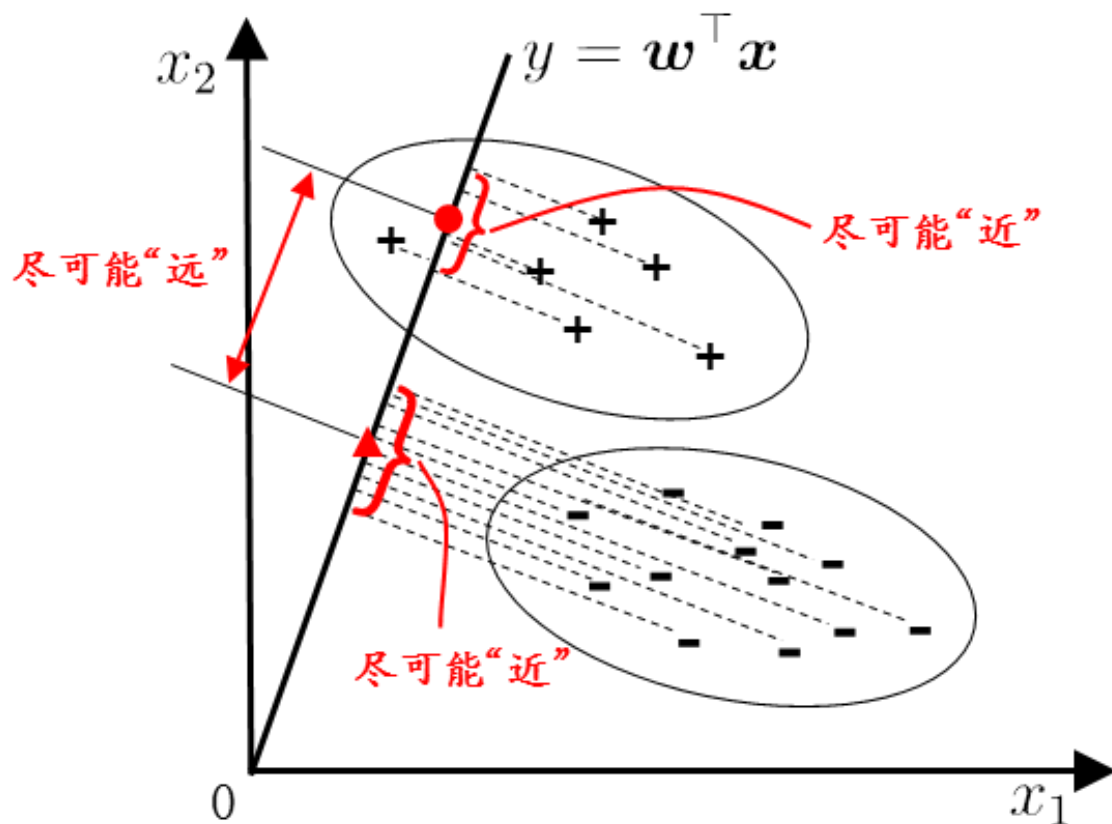
## □ 优化

- 梯度下降



# 二分类任务 - 线性判别分析

□ 线性判别分析 (Linear Discriminant Analysis) [Fisher, 1936]



LDA也可被视为一种  
监督降维技术

# 二分类任务 - 线性判别分析

## □ LDA的思想

- 欲使同类样例的投影点尽可能接近，可以让同类样例投影点的协方差尽可能小
- 欲使异类样例的投影点尽可能远离，可以让类中心之间的距离尽可能大

## □ 一些变量

- 第 $i$ 类示例的集合  $X_i$
- 第 $i$ 类示例的均值向量  $\mu_i$
- 第 $i$ 类示例的协方差矩阵  $\Sigma_i$
- 两类样本的中心在直线上的投影:  $w^T \mu_0$  和  $w^T \mu_1$
- 两类样本的协方差:  $w^T \Sigma_0 w$  和  $w^T \Sigma_1 w$

# 二分类任务 - 线性判别分析

## □ 最大化目标

$$\begin{aligned} J &= \frac{\|w^T \mu_0 - w^T \mu_1\|_2^2}{w^T \Sigma_0 w + w^T \Sigma_1 w} \\ &= \frac{w^T (\mu_0 - \mu_1) (\mu_0 - \mu_1)^T w}{w^T (\Sigma_0 + \Sigma_1) w} \end{aligned}$$

## □ 类内散度矩阵

$$S_w = \Sigma_0 + \Sigma_1$$

$$= \sum_{x \in X_0} (x - \mu_0) (x - \mu_0)^T + \sum_{x \in X_1} (x - \mu_1) (x - \mu_1)^T$$

## □ 类间散度矩阵

$$S_b = (\mu_0 - \mu_1) (\mu_0 - \mu_1)^T$$

# 二分类任务 - 线性判别分析

---

□ 广义瑞利商 (**generalized Rayleigh quotient**)

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

□ 令  $\mathbf{w}^T \mathbf{S}_w \mathbf{w} = 1$ , 最大化广义瑞利商等价形式为

$$\begin{aligned} \min_{\mathbf{w}} \quad & -\mathbf{w}^T \mathbf{S}_b \mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^T \mathbf{S}_w \mathbf{w} = 1 \end{aligned}$$

□ 运用拉格朗日乘子法

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

# 二分类任务 - 线性判别分析

## □ 同向向量

$$\underbrace{S_b \mathbf{w}}_{\text{同向向量}} = \lambda \underbrace{(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)}_{\text{同向向量}}$$

## □ 结果

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

## □ 求解

- 奇异值分解  $S_w = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$   $S_w^{-1} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$

## □ **LDA**的贝叶斯决策论解释

- 两类数据同先验、满足高斯分布且协方差相等时，**LDA**达到最优分类

# LDA推广 - 多分类任务

## □ 全局散度矩阵

$$\begin{aligned} \mathbf{S}_t &= \mathbf{S}_b + \mathbf{S}_w \\ &= \sum_{i=1}^m (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T \end{aligned}$$

## □ 类内散度矩阵

$$\mathbf{S}_w = \sum_{i=1}^N \mathbf{S}_{w_i}$$

其中

$$\mathbf{S}_{w_i} = \sum_{\mathbf{x} \in X_i} (\mathbf{x} - \boldsymbol{\mu}_i)(\mathbf{x} - \boldsymbol{\mu}_i)^T$$

## □ 求解得

$$\begin{aligned} \mathbf{S}_b &= \mathbf{S}_t - \mathbf{S}_w \\ &= \sum_{i=1}^N m_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T \end{aligned}$$

# LDA推广 - 多分类任务

## □ 优化目标

$$\max_{\mathbf{W}} \frac{\text{tr}(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{\text{tr}(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}$$

其中  $\mathbf{W} \in \mathbb{R}^{d \times (N-1)}$



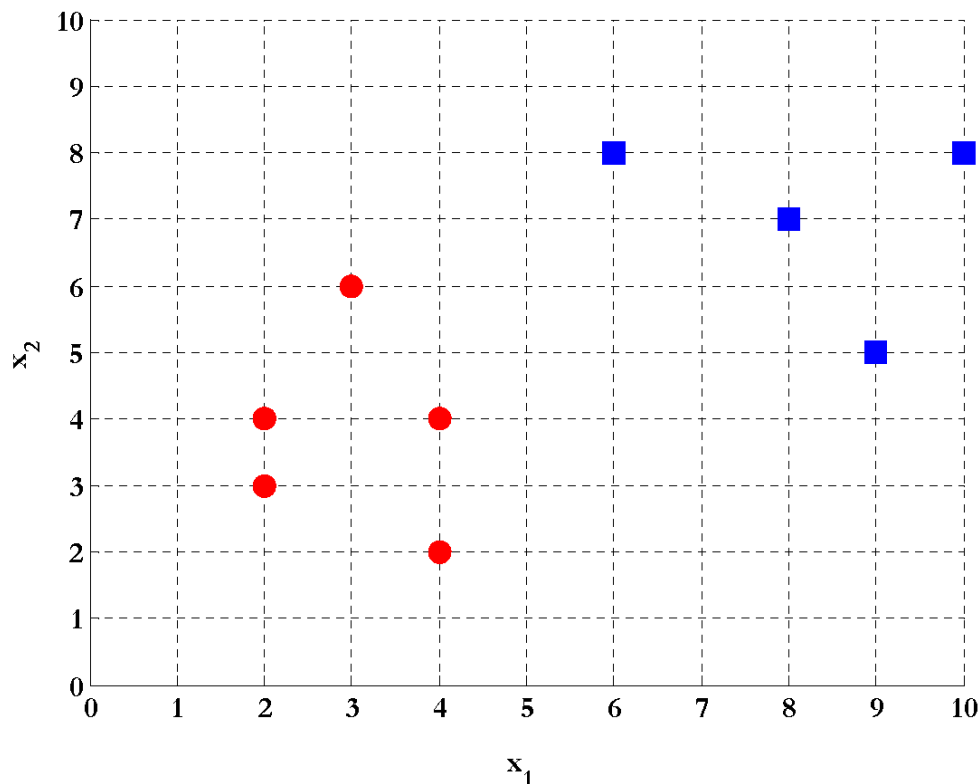
$$\mathbf{S}_b \mathbf{W} = \lambda \mathbf{S}_w \mathbf{W}$$

$\mathbf{W}$  的闭式解则是  $\mathbf{S}_w^{-1} \mathbf{S}_b$  的 **N-1** 个最大广义特征值所对应的特征向量组成的矩阵

□ 多分类 **LDA** 将样本投影到 **N-1** 维空间，**N-1** 通常远小于数据原有的属性数，因此 **LDA** 也被视为一种监督降维技术

# 二分类LDA示例

- Compute the Linear Discriminant projection for the following two-dimensional dataset.
  - Samples for class  $\omega_1$  :  $\mathbf{X}_1=(x_1,x_2)=\{(4,2),(2,4),(2,3),(3,6),(4,4)\}$
  - Sample for class  $\omega_2$  :  $\mathbf{X}_2=(x_1,x_2)=\{(9,10),(6,8),(9,5),(8,7),(10,8)\}$



```
% samples for class 1
X1 = [4,2;
      2,4;
      2,3;
      3,6;
      4,4];

% samples for class 2
X2 = [9,10;
      6,8;
      9,5;
      8,7;
      10,8];
```



- The classes mean are :

$$\mu_1 = \frac{1}{N_1} \sum_{x \in \omega_1} x = \frac{1}{5} \left[ \begin{pmatrix} 4 \\ 2 \end{pmatrix} + \begin{pmatrix} 2 \\ 4 \end{pmatrix} + \begin{pmatrix} 2 \\ 3 \end{pmatrix} + \begin{pmatrix} 3 \\ 6 \end{pmatrix} + \begin{pmatrix} 4 \\ 4 \end{pmatrix} \right] = \begin{pmatrix} 3 \\ 3.8 \end{pmatrix}$$

$$\mu_2 = \frac{1}{N_2} \sum_{x \in \omega_2} x = \frac{1}{5} \left[ \begin{pmatrix} 9 \\ 10 \end{pmatrix} + \begin{pmatrix} 6 \\ 8 \end{pmatrix} + \begin{pmatrix} 9 \\ 5 \end{pmatrix} + \begin{pmatrix} 8 \\ 7 \end{pmatrix} + \begin{pmatrix} 10 \\ 8 \end{pmatrix} \right] = \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix}$$

```
% class means  
Mu1 = mean(X1) ' ;  
Mu2 = mean(X2) ' ;
```

- Covariance matrix of the first class:

$$\begin{aligned} S_1 &= \sum_{x \in \mathcal{O}_1} (x - \mu_1)(x - \mu_1)^T = \left[ \begin{pmatrix} 4 \\ 2 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 2 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 \\ &\quad + \left[ \begin{pmatrix} 2 \\ 3 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 3 \\ 6 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 4 \\ 4 \end{pmatrix} - \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} \right]^2 \\ &= \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix} \end{aligned}$$

```
% covariance matrix of the first class  
S1 = cov(X1);
```

- Covariance matrix of the second class:

$$\begin{aligned} S_2 &= \sum_{x \in \omega_2} (x - \mu_2)(x - \mu_2)^T = \left[ \begin{pmatrix} 9 \\ 10 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 6 \\ 8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 \\ &\quad + \left[ \begin{pmatrix} 9 \\ 5 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 8 \\ 7 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 + \left[ \begin{pmatrix} 10 \\ 8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^2 \\ &= \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix} \end{aligned}$$

```
% covariance matrix of the first class  
S2 = cov(X2) ;
```

- Within-class scatter matrix:

$$\begin{aligned} S_w = S_1 + S_2 &= \begin{pmatrix} 1 & -0.25 \\ -0.25 & 2.2 \end{pmatrix} + \begin{pmatrix} 2.3 & -0.05 \\ -0.05 & 3.3 \end{pmatrix} \\ &= \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix} \end{aligned}$$

```
% within-class scatter matrix  
Sw = S1 + S2 ;
```

- Between-class scatter matrix:

$$\begin{aligned} S_B &= (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \\ &= \left[ \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right] \left[ \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right]^T \\ &= \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix} \begin{pmatrix} -5.4 & -3.8 \end{pmatrix} \\ &= \begin{pmatrix} 29.16 & 20.52 \\ 20.52 & 14.44 \end{pmatrix} \end{aligned}$$

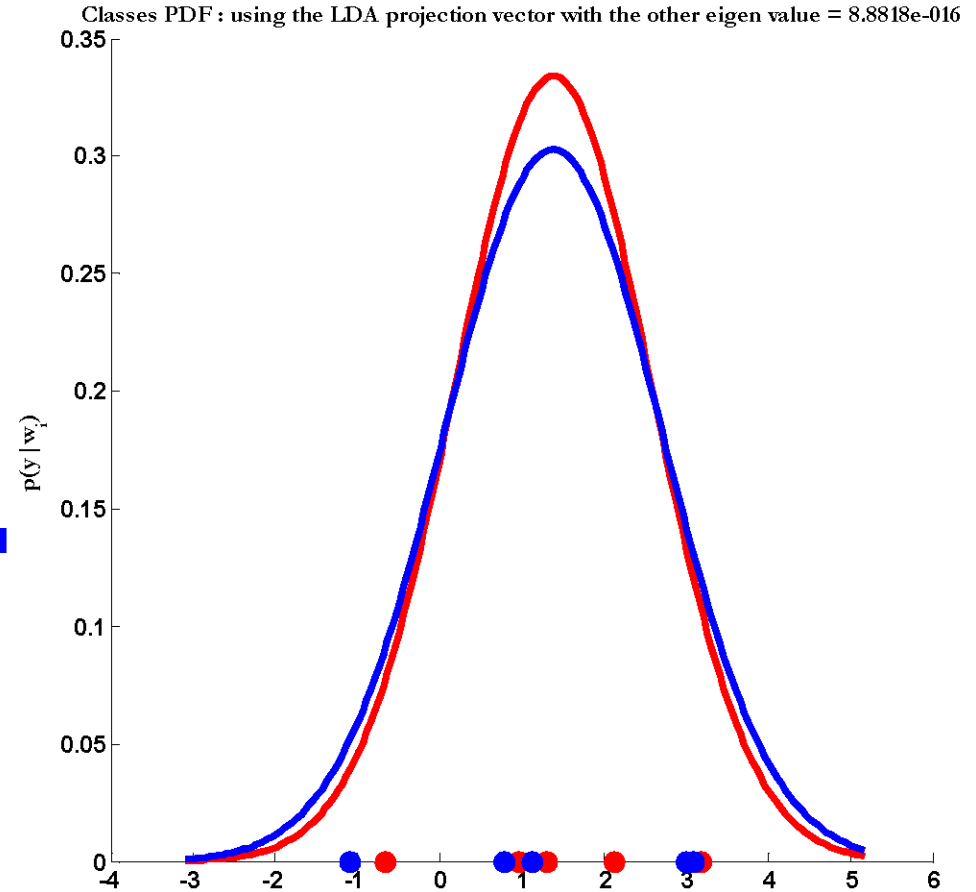
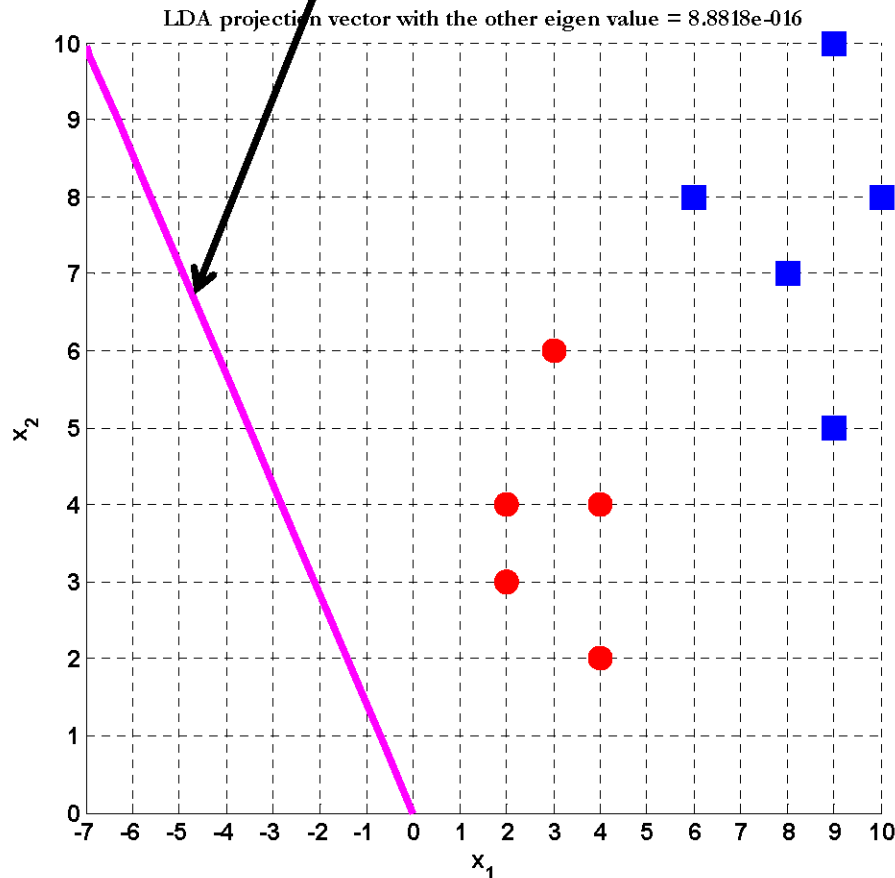
```
% between-class scatter matrix  
SB = (Mu1-Mu2) * (Mu1-Mu2) ' ;
```

---

$$\begin{aligned}w^* &= S_W^{-1}(\mu_1 - \mu_2) = \begin{pmatrix} 3.3 & -0.3 \\ -0.3 & 5.5 \end{pmatrix}^{-1} \left[ \begin{pmatrix} 3 \\ 3.8 \end{pmatrix} - \begin{pmatrix} 8.4 \\ 7.6 \end{pmatrix} \right] \\&= \begin{pmatrix} 0.3045 & 0.0166 \\ 0.0166 & 0.1827 \end{pmatrix} \begin{pmatrix} -5.4 \\ -3.8 \end{pmatrix} \\&= \begin{pmatrix} 0.9088 \\ 0.4173 \end{pmatrix}\end{aligned}$$

# LDA - Projection

The projection vector corresponding to the **smallest** eigen value

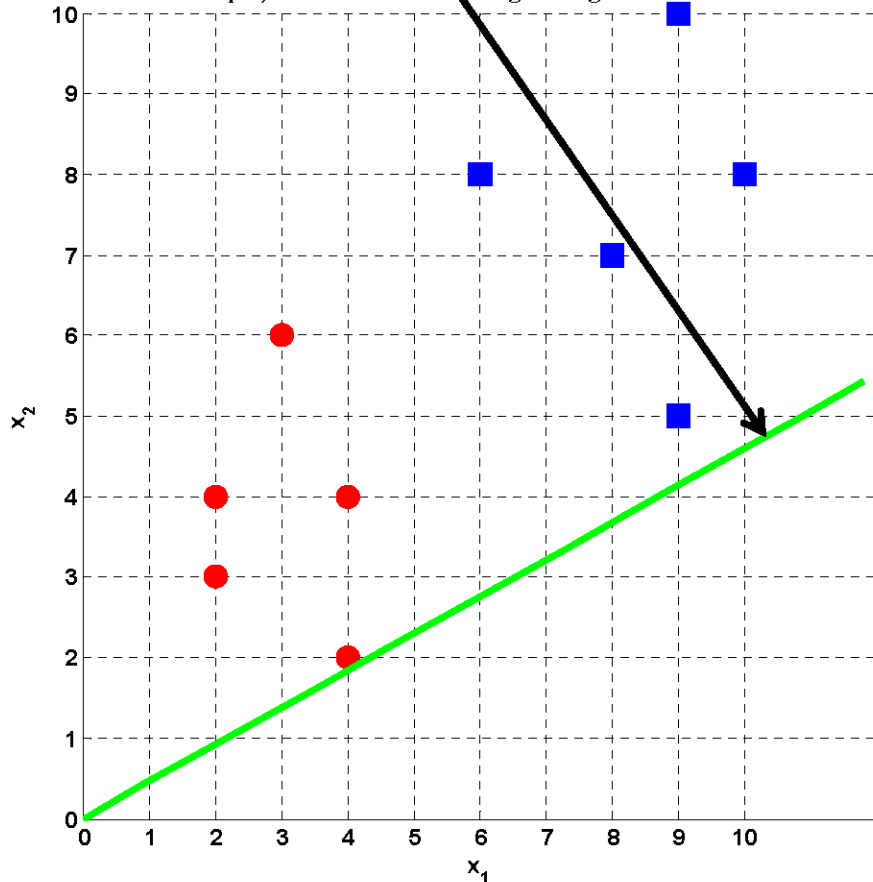


Using this vector leads to **bad separability** between the two classes

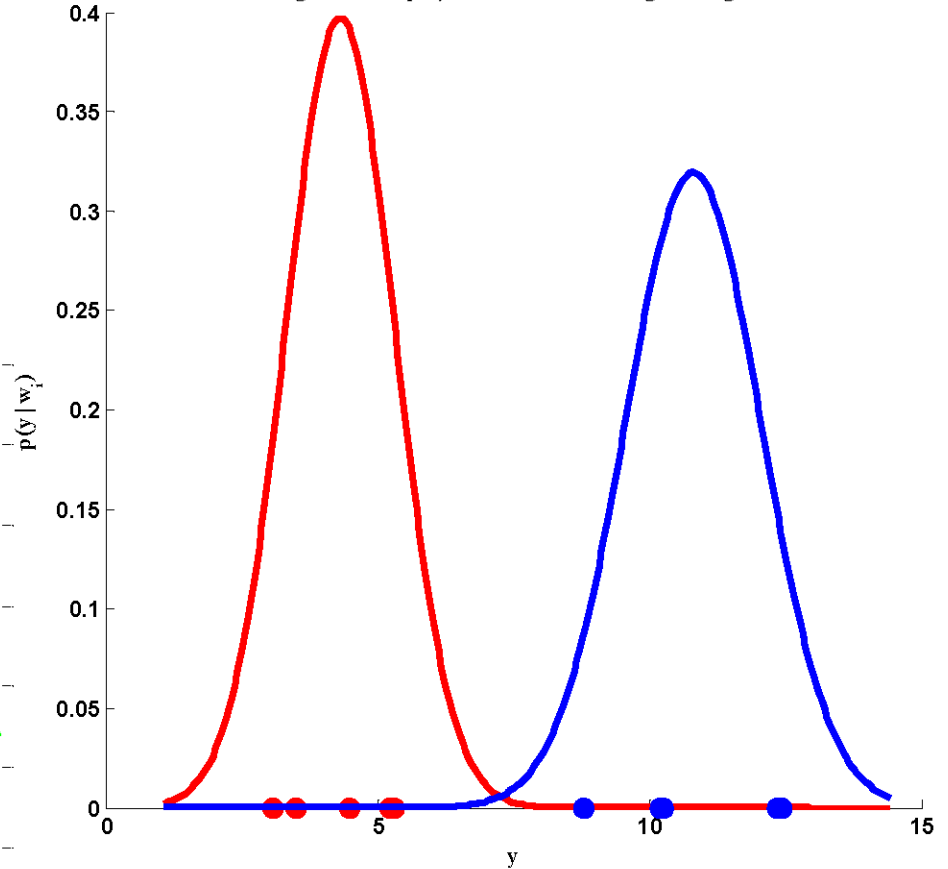
# LDA - Projection

The projection vector corresponding to the **highest** eigen value

LDA projection vector with the highest eigen value = 12.2007



Classes PDF : using the LDA projection vector with highest eigen value = 12.2007



Using this vector leads to **good separability** between the two classes



# 多分类学习

---

## □ 多分类学习方法

- 二分类学习方法推广到多类
- 利用二分类学习器解决多分类问题 (**常用**)
  - 对问题进行拆分, 为拆出的每个二分类任务训练一个分类器
  - 对于每个分类器的预测结果进行集成以获得最终的多分类结果

## □ 拆分策略

- 一对一 (**One vs. One, OvO**)
- 一对其余 (**One vs. Rest, OvR**)
- 多对多 (**Many vs. Many, MvM**)

# 多分类学习 - 一对一

---

## □ 拆分阶段

- **N**个类别两两配对
  - **$N(N-1)/2$**  个二类任务
- 各个二类任务学习分类器
  - **$N(N-1)/2$**  个二类分类器

## □ 测试阶段

- 新样本提交给所有分类器预测
  - **$N(N-1)/2$**  个分类结果
- 投票产生最终分类结果
  - 被预测最多的类别为最终类别

# 多分类学习 - 一对其余

---

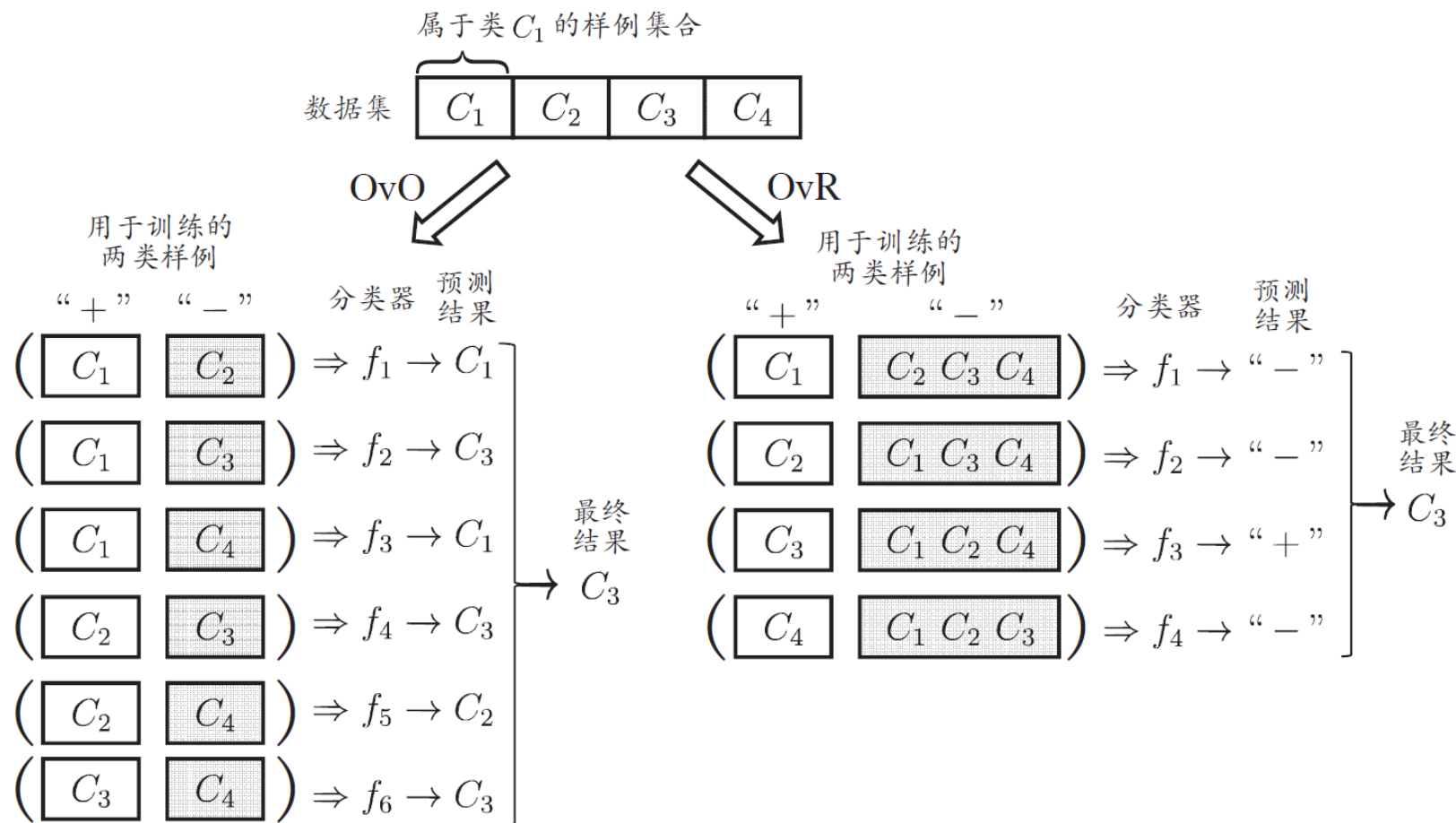
## □ 任务拆分

- 某一类作为正例，其他反例
  - **N** 个二类任务
- 各个二类任务学习分类器
  - **N** 个二类分类器

## □ 测试阶段

- 新样本提交给所有分类器预测
  - **N** 个分类结果
- 比较各分类器预测置信度
  - 置信度最大类别作为最终类别

# 多分类学习 - 两种策略比较



# 多分类学习 - 两种策略比较

## 一对一

- 训练 $N(N-1)/2$ 个分类器，存储开销和测试时间大
- 训练只用两个类的样例，训练时间短

## 一对其余

- 训练 $N$ 个分类器，存储开销和测试时间小
- 训练用到全部训练样例，训练时间长

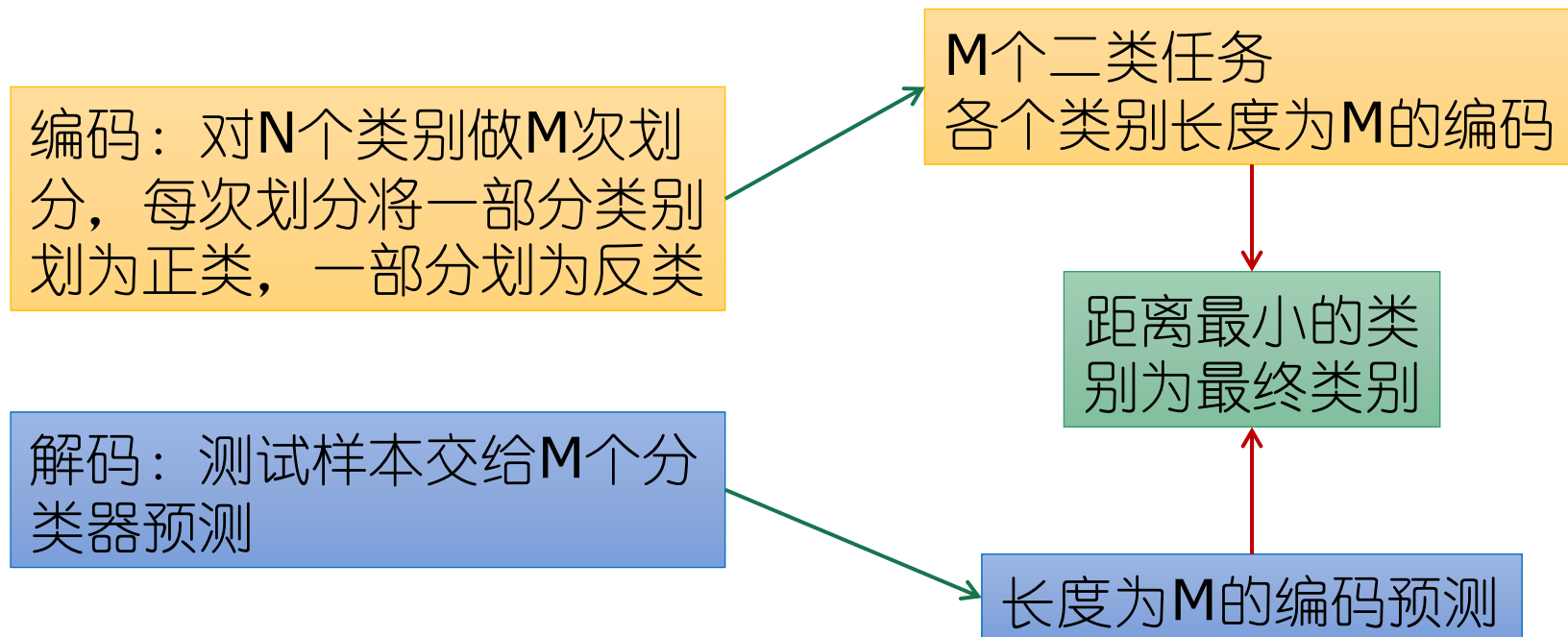
预测性能取决于具体数据分布，多数情况下两者差不多

# 多分类学习 - 多对多

## ❑ 多对多 (**Many vs Many, MvM**)

- 若干类作为正类, 若干类作为反类

## ❑ 纠错输出码 (Error Correcting Output Code, ECOC)



# 多分类学习 - 多对多

## ❑ 纠错输出码(Error Correcting Output Code, ECOC)

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	海明 距离	欧氏 距离
$C_1 \rightarrow$	-1	+1	-1	+1	+1	3	$2\sqrt{3}$
$C_2 \rightarrow$	+1	-1	-1	+1	-1	4	4
$C_3 \rightarrow$	-1	+1	+1	-1	+1	1	2
$C_4 \rightarrow$	-1	-1	+1	+1	-1	2	$2\sqrt{2}$
测试 示例 $\rightarrow$	-1	-1	+1	-1	+1		

(a) 二元 ECOC 码

[Dietterich and Bakiri,1995]

	$f_1$	$f_2$	$f_3$	$f_4$	$f_5$	$f_6$	$f_7$	海明 距离	欧氏 距离
$C_1 \rightarrow$	-1	-1	+1	+1	-1	+1	+1	4	4
$C_2 \rightarrow$	-1	0	0	0	+1	-1	0	2	2
$C_3 \rightarrow$	+1	+1	-1	-1	-1	+1	-1	5	$2\sqrt{5}$
$C_4 \rightarrow$	-1	+1	0	+1	-1	0	+1	3	$\sqrt{10}$
测试 示例 $\rightarrow$	-1	+1	+1	-1	+1	-1	+1		

(b) 三元 ECOC 码

[Allwein et al. 2000]

- ECOC编码对分类器错误有一定容忍和修正能力，编码越长、纠错能力越强
- 对同等长度的编码，理论上来说，任意两个类别之间的编码距离越远，则纠错能力越强

# 类别不平衡问题

## □ 类别不平衡 (**class imbalance**)

- 不同类别训练样例数相差很大情况 (正类为小类)

类别平衡正例预测  $\frac{y}{1-y} > 1$    $\frac{y}{1-y} > \frac{m^+}{m^-}$  正负类比例

## □ 再缩放

- 欠采样 (**undersampling**)
  - 去除一些反例使正反例数目接近 (**EasyEnsemble** [Liu et al.,2009])
- 过采样 (**oversampling**)
  - 增加一些正例使正反例数目接近 (**SMOTE** [Chawla et al.2002])
- 阈值移动 (**threshold-moving**)



# 优化提要

---

- 各任务下（回归、分类）各个模型优化的目标
  - 最小二乘法：最小化均方误差
  - 对数几率回归：最大化样本分布似然
  - 线性判别分析：投影空间内最小（大）化类内（间）散度
  
- 参数的优化方法
  - 最小二乘法：线性代数
  - 对数几率回归：凸优化梯度下降、牛顿法
  - 线性判别分析：矩阵论、广义瑞利商

# 总结

---

## □ 线性回归

- 最小二乘法（最小化均方误差）

## □ 二分类任务

- 对数几率回归
  - 单位阶跃函数、对数几率函数、极大似然法
- 线性判别分析
  - 最大化广义瑞利商

## □ 多分类学习

- 一对一
- 一对其余
- 多对多
  - 纠错输出码

## □ 类别不平衡问题

- 基本策略：再缩放