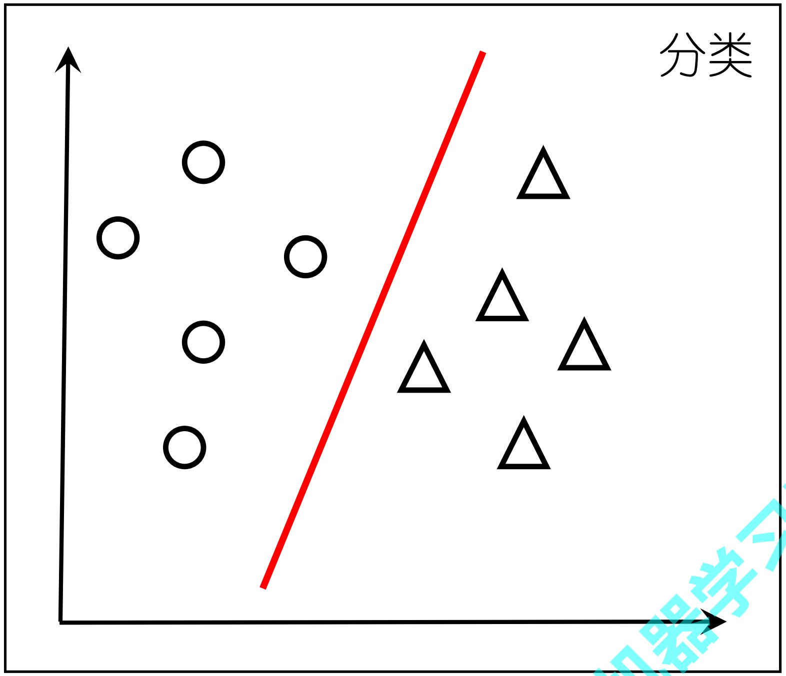
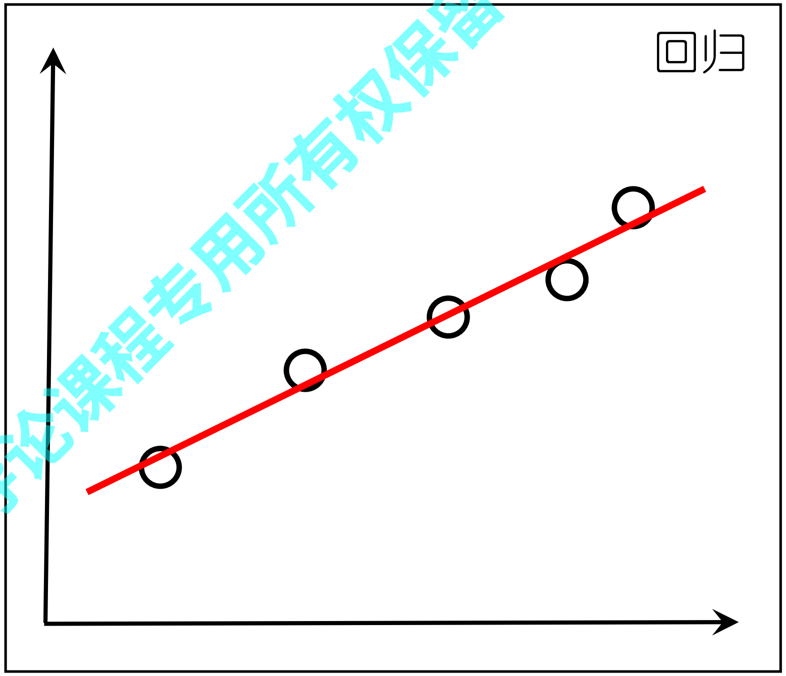
三、线性模型

主讲教师：周志华

线性模型





线性模型(linear model)试图学得一个通过属性的线性组合来进行预测的函数

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

向量形式： $f(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b$

简单、基本、可理解性好

线性回归 (linear regression)

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

离散属性的处理：若有“序”(order)，则连续化；
否则，转化为 k 维向量

令均方误差最小化，有
$$(w^*, b^*) = \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} \left(f\left(x_i\right) - y_i\right)^2$$
$$= \underset{(w,b)}{\arg\min} \sum_{i=1}^{m} \left(y_i - wx_i - b\right)^2$$

对 $E_{(w,b)} = \sum_{i=1}^{m} \left(y_i - wx_i - b\right)^2$ 进行最小二乘参数估计

线性回归

分别对 w 和 b 求导:

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

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

令导数为 0, 得到闭式(closed-form)解:

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

多元(multi-variate)线性回归

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

$$\boldsymbol{x}_i = (x_{i1}; x_{i2}; \ldots; x_{id}) \qquad y_i \in \mathbb{R}$$

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

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} & 1 \\ x_{21} & x_{22} & \cdots & x_{2d} & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{md} & 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_1^{\mathrm{T}} & 1 \\ \boldsymbol{x}_2^{\mathrm{T}} & 1 \\ \vdots & \vdots \\ \boldsymbol{x}_m^{\mathrm{T}} & 1 \end{pmatrix} \qquad \boldsymbol{y} = (y_1; y_2; \ldots; y_m)$$

多元线性回归

同样采用最小二乘法求解，有

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

令 $E_{\hat{\boldsymbol{w}}} = \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)^{\mathrm{T}} \left(\boldsymbol{y} - \mathbf{X}\hat{\boldsymbol{w}}\right)$ ，对 $\hat{\boldsymbol{w}}$ 求导：

$$\frac{\partial E_{\hat{\boldsymbol{w}}}}{\partial \hat{\boldsymbol{w}}} = 2\mathbf{X}^{\mathrm{T}} \left(\mathbf{X}\hat{\boldsymbol{w}} - \boldsymbol{y}\right)$$ 令其为零可得 $\hat{\boldsymbol{w}}$

然而，麻烦来了：涉及矩阵求逆！

- 若 $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 满秩或正定，则 $\hat{\boldsymbol{w}}^* = \left(\mathbf{X}^{\mathrm{T}}\mathbf{X}\right)^{-1} \mathbf{X}^{\mathrm{T}}\boldsymbol{y}$
- 若 $\mathbf{X}^{\mathrm{T}}\mathbf{X}$ 不满秩，则可解出多个 $\hat{\boldsymbol{w}}$

此时需求助于归纳偏好，或引入 正则化 (regularization) ⟶ 第6、11章

线性模型的变化

对于样例 $(\boldsymbol{x}, y)$, $y \in \mathbb{R}$, 若希望线性模型的预测值逼近真实标记,

则得到线性回归模型 $y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$

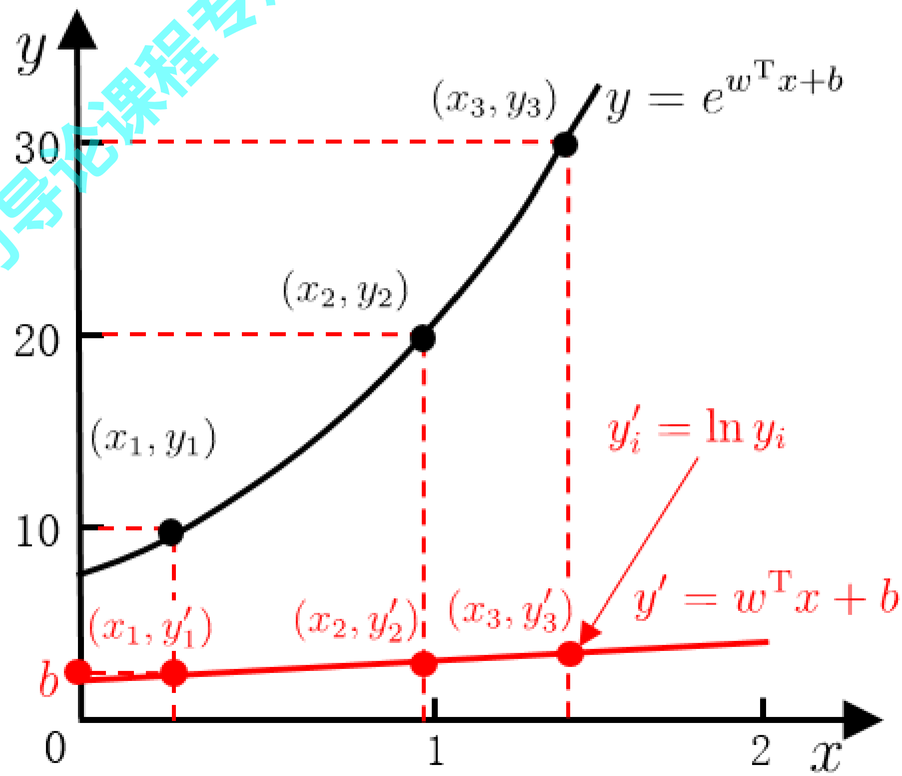
令预测值逼近 y 的衍生物？

若令 $\ln y = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$

则得到对数线性回归

(log-linear regression)

实际是在用 $e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}$ 逼近 y



广义(generalized)线性模型

一般形式：$y = g^{-1}\left(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b\right)$

单调可微的 联系函数 (link function)

令 $g(\cdot) = \ln(\cdot)$ 则得到 对数线性回归

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

…… ……

二分类任务

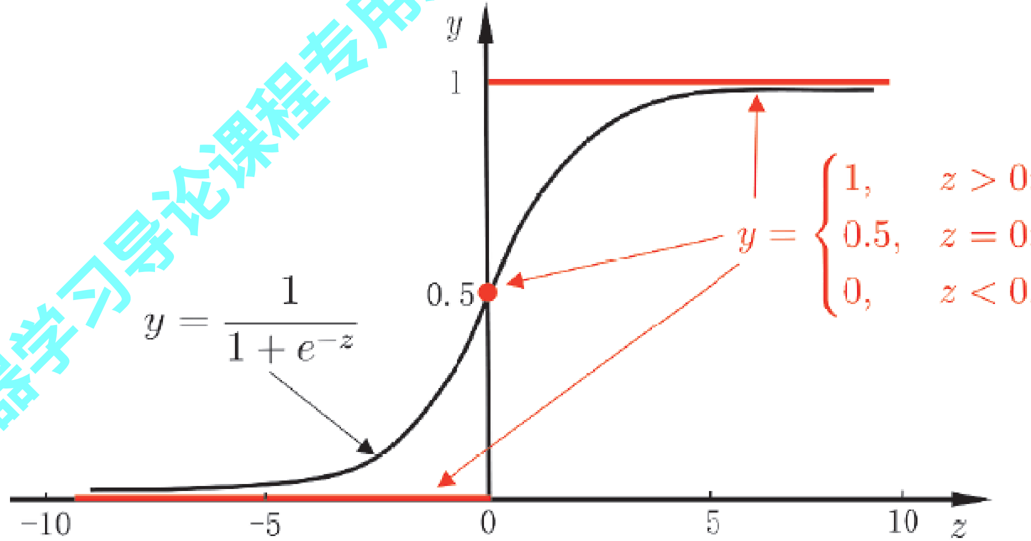
线性回归模型产生的实值输出 $z = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$

期望输出 $y \in \{0, 1\}$

找 z 和 y 的联系函数

理想的 “单位阶跃函数”
(unit-step function)

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



性质不好，
需找 “替代函数”
(surrogate function)

常用
单调可微、任意阶可导

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

对数几率函数
(logistic function)
简称 “对率函数”

对率回归

以对率函数为联系函数：

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

即：

$$\ln \frac{y}{1-y} = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$$

几率(odds)，反映了 $\boldsymbol{x}$ 作为正例的相对可能性

“对数几率”
(log odds, 亦称 logit)

“对数几率回归” (logistic regression)
简称 “对率回归”

- 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

注意：它是
分类学习算法！

求解思路

若将 y 看作类后验概率估计 $p(y=1 \mid \boldsymbol{x})$，则

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

于是，可使用“极大似然法” → 第7章
(maximum likelihood method)

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

最大化“对数似然”(log-likelihood)函数

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

求解思路

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

再令 $p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) = p(y = 1 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}) = \dfrac{e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}{1 + e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$

$p_0(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) = p(y = 0 \mid \hat{\boldsymbol{x}}; \boldsymbol{\beta}) = 1 - p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) = \dfrac{1}{1 + e^{\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b}}$

则似然项可重写为 $p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}_i, b) = y_i p_1(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\boldsymbol{x}}_i; \boldsymbol{\beta})$

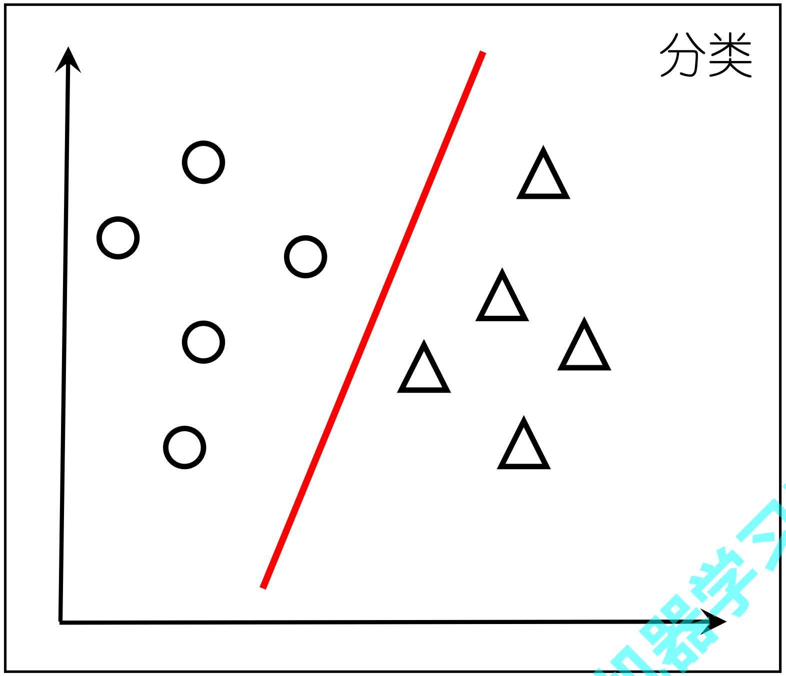
于是，最大化似然函数 $\ell(\boldsymbol{w}, b) = \sum_{i=1}^{m} \ln p(y_i \mid \boldsymbol{x}_i; \boldsymbol{w}, b)$

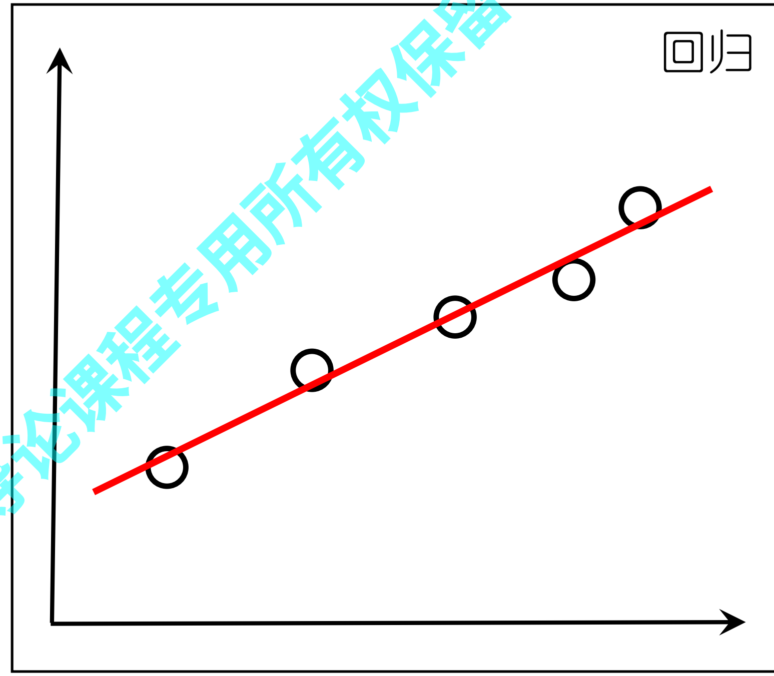
等价为最小化 $\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left(-y_i \boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i + \ln\left(1 + e^{\boldsymbol{\beta}^{\mathrm{T}}\hat{\boldsymbol{x}}_i}\right) \right)$

高阶可导连续凸函数，可用经典的数值优化方法

如梯度下降法/牛顿法 [Boyd and Vandenberghe, 2004]

线性模型做“分类”



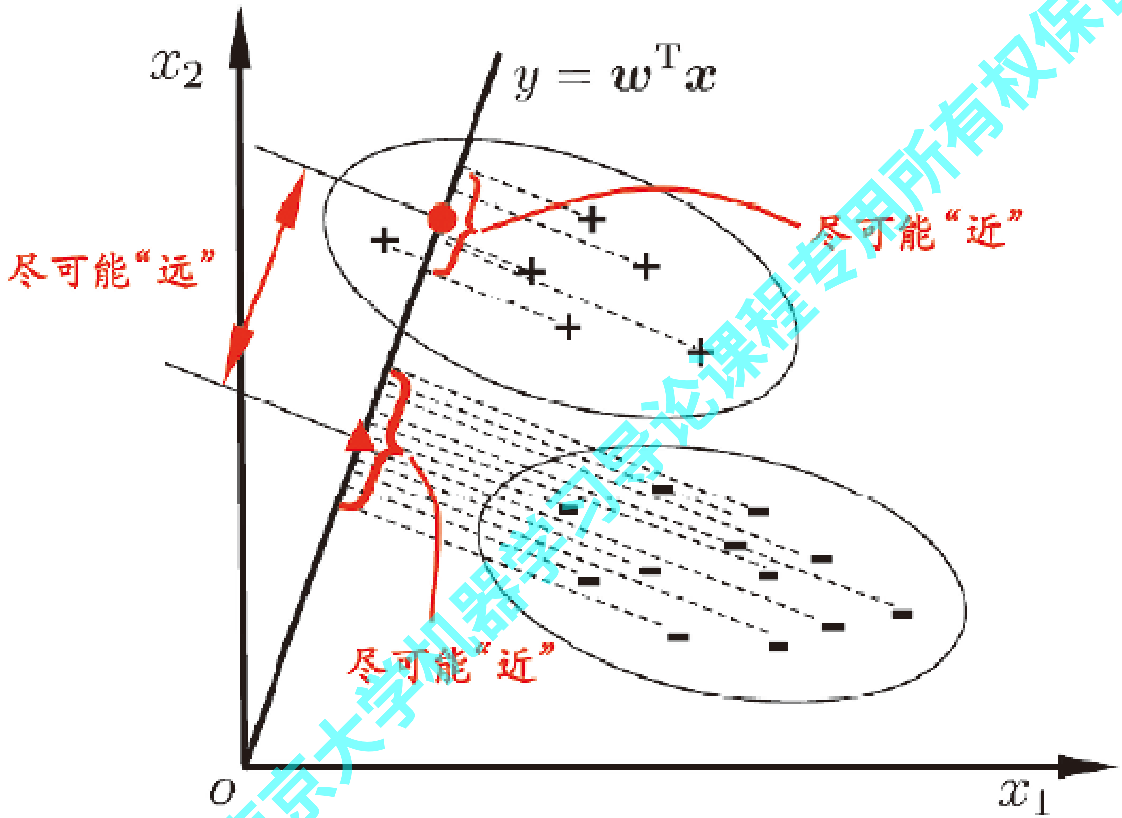


如何“直接”做分类？

广义线性模型；
通过“联系函数”

例如，对率回归

线性判别分析 (Linear Discriminant Analysis)



由于将样例投影到一条直线（低维空间），因此也被视为一种“监督降维”技术 降维 ⟶ 第10章

LDA的目标

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

第 i 类示例的集合 X_i

第 i 类示例的均值向量 $\boldsymbol{\mu}_i$

第 i 类示例的协方差矩阵 $\boldsymbol{\Sigma}_i$

两类样本的中心在直线上的投影：$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0$ 和 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1$

两类样本的协方差：$\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w}$ 和 $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}$

同类样例的投影点尽可能接近 → $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_0\boldsymbol{w} + \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\Sigma}_1\boldsymbol{w}$ 尽可能小

异类样例的投影点尽可能远离 → $\|\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_0 - \boldsymbol{w}^{\mathrm{T}}\boldsymbol{\mu}_1\|_2^2$ 尽可能大

于是，最大化

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

LDA的目标

类内散度矩阵 (within-class scatter matrix)

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

类间散度矩阵 (between-class scatter matrix)

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

LDA的目标：最大化广义瑞利商 (generalized Rayleigh quotient)

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

$\boldsymbol{w}$ 成倍缩放不影响 J 值
仅考虑方向

求解思路

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

$$\min_{\boldsymbol{w}} \ -\boldsymbol{w}^{\mathrm{T}}\mathbf{S}_b\boldsymbol{w}$$
$$\text{s.t.}\ \boldsymbol{w}^{\mathrm{T}}\mathbf{S}_w\boldsymbol{w}=1$$

运用拉格朗日乘子法，有 $\mathbf{S}_b\boldsymbol{w}=\lambda\mathbf{S}_w\boldsymbol{w}$

$\mathbf{S}_b\boldsymbol{w}$ 的方向恒为 $\boldsymbol{\mu}_0-\boldsymbol{\mu}_1$，不妨令 $\mathbf{S}_b\boldsymbol{w}=\lambda(\boldsymbol{\mu}_0-\boldsymbol{\mu}_1)$

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

实践中通常是进行奇异值分解 $\mathbf{S}_w=\mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}}$ $\longrightarrow$ 附录A

然后 $\mathbf{S}_w^{-1}=\mathbf{V}\boldsymbol{\Sigma}^{-1}\mathbf{U}^{\mathrm{T}}$

推广到多类

假定有 N 个类

- 全局散度矩阵 $\mathbf{S}_t = \mathbf{S}_b + \mathbf{S}_w = \sum_{i=1}^{m} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$
- 类内散度矩阵 $\mathbf{S}_w = \sum_{i=1}^{N} \mathbf{S}_{w_i}$ $\quad \mathbf{S}_{w_i} = \sum_{\boldsymbol{x} \in X_i} (\boldsymbol{x} - \boldsymbol{\mu}_i)(\boldsymbol{x} - \boldsymbol{\mu}_i)^T$
- 类间散度矩阵 $\mathbf{S}_b = \mathbf{S}_t - \mathbf{S}_w = \sum_{i=1}^{N} m_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^T$

多分类LDA有多种实现方法：采用 $\mathbf{S}_b$, $\mathbf{S}_w$, $\mathbf{S}_t$ 中的任何两个

例如，$\max_{\mathbf{W}} \frac{\operatorname{tr}(\mathbf{W}^{\mathrm{T}} \mathbf{S}_b \mathbf{W})}{\operatorname{tr}(\mathbf{W}^{\mathrm{T}} \mathbf{S}_w \mathbf{W})} \Longrightarrow \mathbf{S}_b \mathbf{W} = \lambda \mathbf{S}_w \mathbf{W}$

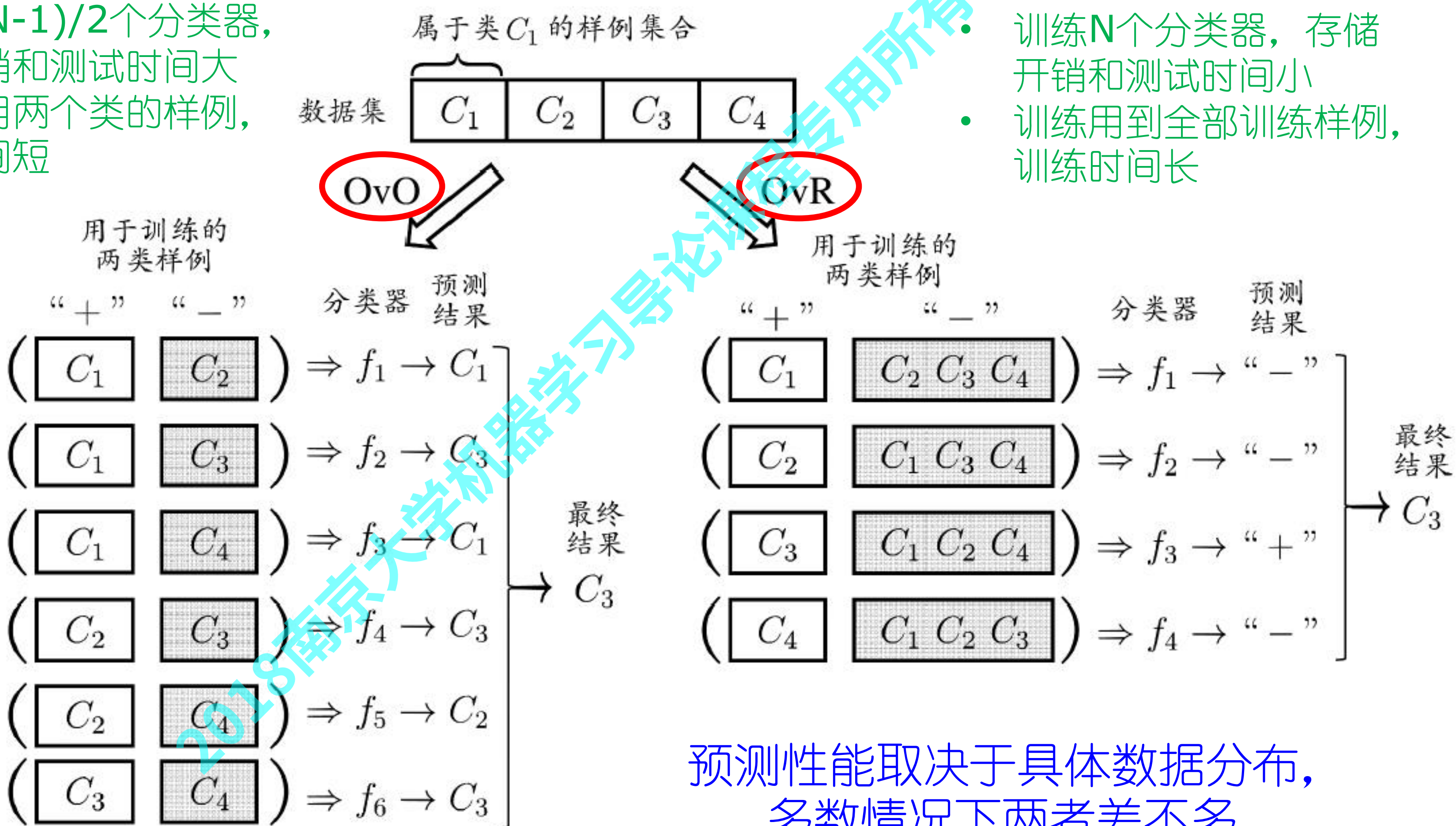
$\mathbf{W} \in \mathbb{R}^{d \times (N-1)}$

$\mathbf{W}$的闭式解是 $\mathbf{S}_w^{-1}\mathbf{S}_b$的 d' ($\leq N$-1) 个最大非零广义特征值对应的特征向量组成的矩阵

多分类学习

拆解法：将一个多分类任务拆分为若干个二分类任务求解

- 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



- 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

纠错输出码 (ECOC)

多对多(Many vs Many, MvM)：将若干类作为正类，若干类作为反类

一种常见方法：纠错输出码 (Error Correcting Output Code)

编码：对 N 个类别做 M 次划分，每次将一部分类别划为正类，一部分划为反类

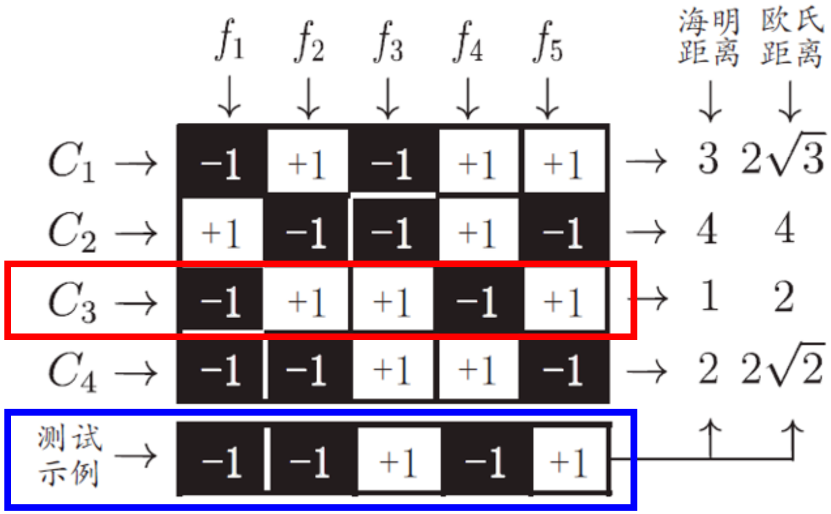
M 个二类任务；
(原)每类对应一个长为 M 的编码

距离最小的类为最终结果

解码：测试样本交给 M 个分类器预测

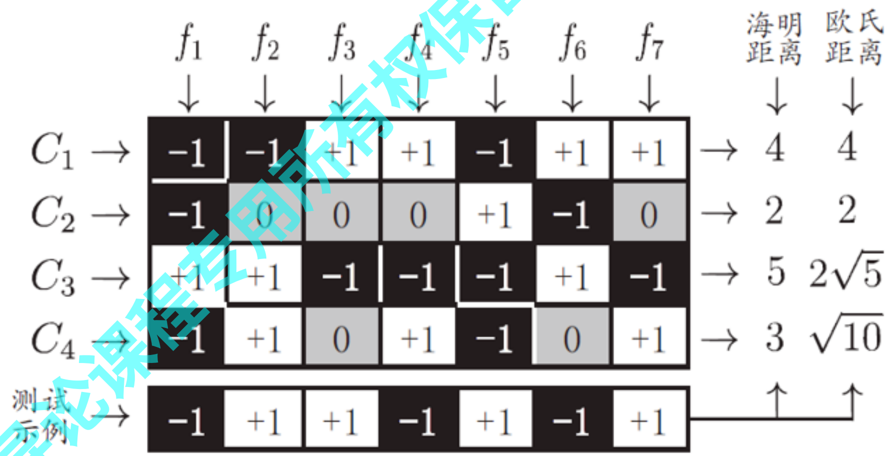
长为 M 的预测结果编码

纠错输出码



(a) 二元 ECOC 码

[Dietterich and Bakiri,1995]



(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 (class-imbalance)

不同类别的样本比例相差很大；“小类”往往更重要

基本思路：

$$\text{若}\quad \frac{y}{1-y} > 1 \quad \text{则}\quad \text{预测为正例.}$$

$$\text{若}\quad \frac{y}{1-y} > \frac{m^+}{m^-} \quad \text{则}\quad \text{预测为正例.}$$

基本策略

——“再缩放”(rescaling)：

$$\frac{y'}{1-y'} = \frac{y}{1-y} \times \frac{m^-}{m^+}$$

然而，精确估计 m^-/m^+ 通常很困难！

常见类别不平衡学习方法：

- 过采样 (oversampling)

  例如：SMOTE

- 欠采样 (undersampling)

  例如：EasyEnsemble

- 阈值移动 (threshold-moving)

前往第四站……