



方差分析

[One-Way ANOVA](#)

[多重比较](#)

[Two-Way ANOVA](#)

考试

One-way ANOVA

基于方差分析表判断假设是否成立

方差分析表：

来源	平方和SS	自由度df	均方和MS	F值
因子A	SS_A	$a - 1$	$MS_A = \frac{SS_A}{a-1}$	$F_A = \frac{MS_A}{MS_E}$
误差E	SS_E	$n - a$	$MS_E = \frac{SS_E}{n-a}$	
总和	SS_T	$n - 1$		

题设：给定了显著性水平和数据结构表

1. 首先计算 SS_T, SS_A, SS_E
2. 得到方差分析表，得到 F 比
3. 根据自由度，给出拒绝域
4. 判断 F 比是否落入拒绝域中，验证假设是否成立

方差分析的基本假设

1. 方差齐性：Bartlett 检验、Levene 检验
2. 正态性：Q-Q 图、Shapiro-Wilk 检验
3. 独立性：列联表的独立性检验

方差分析的三个定理

定理 1：

$$\frac{SS_E}{\sigma^2} \sim \chi^2(n - a)$$

定理 2：

$$E(SS_A) = (a-1)\sigma^2 + m \sum_{i=1}^a \alpha_i^2$$

$$\frac{SS_A}{\sigma^2} \stackrel{H_0}{\sim} \chi^2(a-1)$$

定理 3:

$$SS_A \perp SS_E$$

多重比较

为什么使用多重比较

因为 One-way ANOVA 只能判断是否有不相等，但不知道是哪一对不相等，多重比较就是为了解决这一问题的。

tukey 方法

首先用蒙特卡罗方法计算 t 化极差分布

Algorithm 1 t 化极差统计量的蒙特卡洛分布

Require: 水平数目 a , t 分布的自由度 df , 重复次数 N ;

Ensure: t 化极差统计量的 N 个观测值

- 1: for $n = 1, 2, \dots, N$ do
 - 2: 从标准正态分布 $N(0, 1)$ 产生 a 个随机数: x_1, x_2, \dots, x_a ;
 - 3: 将 a 个数据进行排序, 令 x_{\max} 为最大值, x_{\min} 为最小值;
 - 4: 从自由度为 df 的 χ^2 分布产生一个随机数 y ;
 - 5: 计算 $q_n = (x_{\max} - x_{\min}) / \sqrt{y/df}$;
-

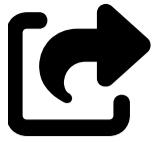
实际计算中需要比较每一组样本均值的差与临界值的大小:

$$c = q_{1-\alpha}(a, df)\hat{\sigma} / \sqrt{m}$$

Two-way ANOVA

基于方差分析表判断假设是否成立

来源	平方和 SS	自由度 df	均方和 MS	F 值
因子 A	SS_A	$a-1$	$MS_A = \frac{SS_A}{a-1}$	$F_A = \frac{MS_A}{MS_E}$
因子 B	SS_B	$b-1$	$MS_B = \frac{SS_B}{b-1}$	$F_B = \frac{MS_B}{MS_E}$
交互效应 AB	SS_{AB}	$(a-1)(b-1)$	$MS_{AB} = \frac{SS_{AB}}{(a-1)(b-1)}$	$F_{AB} = \frac{MS_{AB}}{MS_E}$
误差 E	SS_E	$ab(m-1)$	$MS_E = \frac{SS_E}{ab(m-1)}$	
总和	SS_T	$n-1$		



One-Way ANOVA

二样本独立 t 检验

假设检验问题

$$H_0: \mu_1 = \mu_2 \quad vs \quad H_1: \mu_1 \neq \mu_2$$

检验统计量

$$t = \frac{\bar{x}_1 - \bar{x}_2}{s_w \sqrt{\frac{1}{m_1} + \frac{1}{m_2}}} \stackrel{H_0}{\sim} t(m_1 + m_2 - 2)$$

合方差

$$\begin{aligned} s_w^2 &= (m_1 + m_2 - 2)^{-1} \left((m_1 - 1)s_1^2 + (m_2 - 1)s_2^2 \right) \\ &= \frac{(m_1 - 1)}{(m_1 + m_2 - 2)} \cdot s_1^2 + \frac{(m_2 - 1)}{(m_1 + m_2 - 2)} \cdot s_2^2 \end{aligned}$$

拒绝域法

$$W = \{|t| \geq t_{1-\alpha/2}(2(m-1))\}$$

p 值法

$$p = 2P(t > |t_0|)$$

模型

模型定义

定义

1. 响应变量：记为 y
2. 因子：记为 a
3. 重复次数：记为 m

数据结构

水平	观测到的响应				总和	均值
1	y_{11}	y_{12}	\cdots	y_{1m}	$y_{1\cdot}$	$\bar{y}_{1\cdot}$
2	y_{21}	y_{22}	\cdots	y_{2m}	$y_{2\cdot}$	$\bar{y}_{2\cdot}$
\vdots	\vdots	\vdots		\vdots	\vdots	\vdots
a	y_{a1}	y_{a2}	\cdots	y_{am}	$y_{a\cdot}$	$\bar{y}_{a\cdot}$
汇总					$y_{\cdot\cdot}$	$\bar{y}_{\cdot\cdot}$

均值模型假设

$$H_0: \mu_1 = \mu_2 = \dots = \mu_a$$

H_1 ：存在在两种水平 i, j 下的均值不相等，即 $\mu_i \neq \mu_j$.

均值模型

$$y_{ij} = \mu_i + \varepsilon_{ij}, \begin{cases} i = 1, 2, \dots, a \\ j = 1, 2, \dots, m \end{cases}$$

效应模型假设

$$H_0: \alpha_1 = \alpha_2 = \dots = \alpha_a = 0$$

H_1 ：存在第 i 个水平不为零，即 $\alpha_i \neq 0$.

效应模型

$$y_{ij} = \underline{\mu + \alpha_i} + \varepsilon_{ij}, \begin{cases} i = 1, 2, \dots, a \\ j = 1, 2, \dots, m \end{cases}$$

效应模型约束

$$\sum_{i=1}^n \alpha_i = 0$$

随机误差假定

$$\varepsilon_{ij} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2)$$

数据分布

$$y_{ij} \sim N(\mu + \alpha_i, \sigma^2)$$

平方和分解

平方和分解公式

$$\begin{aligned} \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_{..})^2 &= \sum_{i=1}^a \sum_{j=1}^m ((\bar{y}_i - \bar{y}_{..}) + (y_{ij} - \bar{y}_{i.}))^2 \\ &= m \sum_{i=1}^a (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_{i.})^2 \\ &\quad + 2 \sum_{i=1}^a \sum_{j=1}^m (\bar{y}_i - \bar{y}_{..}) (y_{ij} - \bar{y}_{i.}) \\ &= m \sum_{i=1}^a (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_{i.})^2 \end{aligned}$$

$$SS_T = SS_A + SS_E$$

交叉项为 0，因为有

$$\sum_{j=1}^m (y_{ij} - \bar{y}_i) = y_i - m\bar{y}_i = y_i - y_i = 0$$

符号规定

$$SS_A = m \sum_{i=1}^a (\bar{y}_{i\cdot} - \bar{y}_{\cdot\cdot})^2$$

$$SS_E = \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_{i\cdot})^2$$

定理证明

定理 1

$$\frac{SS_E}{\sigma^2} \sim \chi^2(n - a)$$

证明：

$$\begin{aligned} SS_E &= \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_i)^2 \\ &= \sum_{i=1}^a \sum_{j=1}^m \left((\mu + \alpha_i + \varepsilon_{ij}) - m^{-1} \sum_{j=1}^m (\mu + \alpha_i + \varepsilon_{ij}) \right)^2 \\ &= \sum_{i=1}^a \sum_{j=1}^m \left((\mu + \alpha_i + \varepsilon_{ij}) - \left(\mu + \alpha_i + m^{-1} \sum_{j=1}^m \varepsilon_{ij} \right) \right)^2 \\ &= \sum_{i=1}^a \sum_{j=1}^m (\varepsilon_{ij} - \bar{\varepsilon}_{i\cdot})^2 \end{aligned}$$

$$\frac{1}{\sigma^2} \sum_{j=1}^m (\varepsilon_{ij} - \bar{\varepsilon}_i)^2 \sim \chi^2(m - 1), \quad j = 1, 2, \dots, m$$

$$\frac{SS_E}{\sigma^2} = \frac{1}{\sigma^2} \sum_{i=1}^a \sum_{j=1}^m (\varepsilon_{ij} - \bar{\varepsilon}_i)^2 \sim \chi^2(a(m - 1)) = \chi^2(n - a)$$

定理 2

$$E(SS_A) = (a - 1)\sigma^2 + m \sum_{i=1}^a \alpha_i^2$$

$$\frac{SS_A}{\sigma^2} \stackrel{H_0}{\sim} \chi^2(a - 1)$$

证明：

$$\begin{aligned}
SS_A &= m \sum_{i=1}^a (\bar{y}_i \cdot -\bar{y}..)^2 \\
&= m \sum_{i=1}^a \left(\frac{1}{m} \sum_{j=1}^m (\mu + \alpha_i + \varepsilon_{ij}) - \frac{1}{n} \sum_{i=1}^a \sum_{j=1}^m (\mu + \alpha_i + \varepsilon_{ij}) \right)^2 \\
&= m \sum_{i=1}^a (\alpha_i + \bar{\varepsilon}_i \cdot -\bar{\varepsilon}..)^2 \\
&= m \sum_{i=1}^a \left(\alpha_i^2 + (\bar{\varepsilon}_i \cdot -\bar{\varepsilon}..)^2 + 2\alpha_i (\bar{\varepsilon}_i \cdot -\bar{\varepsilon}..) \right) \\
&= m \sum_{i=1}^a \alpha_i^2 + m \sum_{i=1}^a (\bar{\varepsilon}_{i.} - \bar{\varepsilon}..)^2 + 2m \sum_{i=1}^a \alpha_i (\bar{\varepsilon}_i \cdot \bar{\varepsilon}..)
\end{aligned}$$

因为有

$$\varepsilon_{ij} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2)$$

所以

$$\begin{aligned}
\bar{\varepsilon}_i &= m^{-1} \sum_{j=1}^m \varepsilon_{ij} \sim N(0, \sigma^2 m^{-1}) \\
\bar{\varepsilon}.. &= n^{-1} \sum_{i=1}^a \sum_{j=1}^m \varepsilon_{ij} \sim N(0, \sigma^2 n^{-1})
\end{aligned}$$

交叉项期望

$$E \left(2m \sum_{i=1}^a \alpha_i (\bar{\varepsilon}_{i.} - \bar{\varepsilon}..) \right) = 2m \sum_{i=1}^a \alpha_i E(\bar{\varepsilon}_i \cdot \bar{\varepsilon}..) = 0$$

于是有

$$E(SS_A) = m \sum_{i=1}^a \alpha_i^2 + m E \left(\sum_{i=1}^a (\bar{\varepsilon}_i - \bar{\varepsilon}..)^2 \right)$$

因为

$$\bar{\varepsilon}.. = \frac{1}{n} \sum_{i=1}^a \sum_{j=1}^m \varepsilon_{ij} = \frac{1}{a} \sum_{i=1}^a \bar{\varepsilon}_i$$

所以

$$(\sigma^2 m^{-1})^{-1} \sum_{i=1}^a (\bar{\varepsilon}_i - \bar{\varepsilon}..)^2 \sim \chi^2(a-1)$$

由卡方分布的期望可知

$$E(SS_A) = m \sum_{i=1}^a \alpha_i^2 + (a-1)\sigma^2$$

当 $\alpha_1 = \alpha_2 = \dots = \alpha_a = 0$ ，有

$$\frac{SS_A}{\sigma^2} = \frac{\sum_{i=1}^a (\bar{\varepsilon}_{i.} - \bar{\varepsilon}..)^2}{\sigma^2/m} \sim \chi^2(a-1)$$

定理 3

$$SS_A \perp SS_E$$

证明：

$$SS_A = m \sum_{i=1}^a (\alpha_i + \bar{\varepsilon}_{i\cdot} - \bar{\varepsilon}_{..})^2$$

$$SS_E = \sum_{i=1}^a \sum_{j=1}^m (\varepsilon_{ij} - \bar{\varepsilon}_{i\cdot})^2$$

由定理可得，

$$\sum_{j=1}^m (\varepsilon_{ij} - \bar{\varepsilon}_{i\cdot})^2 \perp \bar{\varepsilon}_i$$

又因为

$$\varepsilon_{ij} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2)$$

因此独立。

检验

检验统计量

$$F_A = \frac{SS_A/(a-1)}{SS_E/(n-a)}$$

拒绝域

$$F_A \geq F_{1-\alpha}(a-1, n-a)$$

p 值

$$p_A = P(F \geq F_A)$$

方差分析表

来源	平方和SS	自由度df	均方和MS	F值
因子A	SS_A	$a-1$	$MS_A = \frac{SS_A}{a-1}$	$F_A = \frac{MS_A}{MS_E}$
误差E	SS_E	$n-a$	$MS_E = \frac{SS_E}{n-a}$	
总和	SS_T	$n-1$		

参数估计

点估计

$$y_{ij} \stackrel{\text{独立}}{\sim} N(\mu + \alpha_i, \sigma^2) \quad i = 1, 2, \dots, a, j = 1, 2, \dots, m$$

似然函数

$$L(\mu, \alpha_1, \alpha_2, \dots, \alpha_a, \sigma^2) = \prod_{i=1}^a \prod_{j=1}^m \left\{ \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(y_{ij} - \mu - \alpha_i)^2}{2\sigma^2} \right\} \right\}$$

对数似然函数

$$\begin{aligned} l(\mu, \alpha_1, \alpha_2, \dots, \alpha_a, \sigma^2) \\ = \ln L(\mu, \alpha_1, \alpha_2, \dots, \alpha_a, \sigma^2) \\ = -\frac{n}{2} \ln(2\pi\sigma^2) - \sum_{i=1}^a \sum_{j=1}^m \frac{(y_{ij} - \mu - \alpha_i)^2}{2\sigma^2} \end{aligned}$$

似然方程组

$$\begin{cases} \frac{\partial l}{\partial \mu} = \frac{1}{\sigma^2} \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \mu - \alpha_i) = 0 \\ \frac{\partial l}{\partial \alpha_i} = \frac{1}{\sigma^2} \sum_{j=1}^m (y_{ij} - \mu - \alpha_i) = 0, \quad i = 1, 2, \dots, a \\ \frac{\partial l}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \mu - \alpha_i)^2 = 0 \end{cases}$$

效应模型约束

$$\sum_{i=1}^a \alpha_i = 0$$

极大似然估计

$$\begin{cases} \hat{\mu} = \bar{y}_{..}, \\ \hat{\alpha}_i = \bar{y}_{i\cdot} - \bar{y}_{..}, \quad i = 1, 2, \dots, a \\ \hat{\sigma}_{\text{MLE}}^2 = \frac{1}{n} \sum_{i=1}^a \sum_{j=1}^m (y_{ij} - \bar{y}_{i\cdot})^2 = \frac{SS_E}{n} \end{cases}$$

针对均值模型的有

$$\hat{\mu}_i = \bar{y}_{i\cdot}$$

方差的无偏估计

$$\hat{\sigma}^2 = \frac{SS_E}{n-a} = MS_E$$

区间估计

$$\bar{y}_i = \frac{1}{m} \sum_{j=1}^m y_{ij} = \mu + \alpha_i + \bar{\varepsilon}_i \sim N\left(\underline{\mu + \alpha_i}_{\mu_i}, \sigma^2 m^{-1}\right)$$

$$\frac{SS_E}{\sigma^2} \sim \chi^2(n - a)$$

又有诸均值与 SS_E 独立

$$\frac{\sqrt{m}(\bar{y}_{i\cdot} - \mu_i)}{\sqrt{SS_E/(n-a)}} \sim t(n-a), \quad i = 1, 2, \dots, a$$

于是有置信区间

$$\mu_i \in [\bar{y}_{i\cdot} - t_{1-\alpha/2}(n-a)\hat{\sigma}, \bar{y}_{i\cdot} + t_{1-\alpha/2}(n-a)\hat{\sigma}]$$



多重比较

水平均值差的置信区间

问题：想要知道那些水平的均值是不相等的。

枢轴量法

对于

$$\bar{y}_i \sim N(\mu_i, \sigma^2 m^{-1}) \quad \bar{y}_{i'} \sim N(\mu_{i'}, \sigma^2 m^{-1})$$

两者独立，于是有

$$\bar{y}_i \sim N(\mu_i, \sigma^2 m^{-1}) \quad \text{和} \quad \bar{y}_{i'} \sim N(\mu_{i'}, \sigma^2 m^{-1})$$

σ^2 未知，于是有

$$\hat{\sigma}^2 = \frac{SS_E}{n - a}$$

于是有枢轴量

$$\frac{(\bar{y}_i - \bar{y}_{i'}) - (\mu_i - \mu_{i'})}{\sqrt{\frac{2}{m}} \hat{\sigma}} \sim t(n - a)$$

置信区间

$$(\bar{y}_i - \bar{y}_{i'}) \pm \sqrt{\frac{2}{m}} \hat{\sigma} \cdot t_{1-\alpha/2}(n - a)$$

可以构造检验问题

$$H_0 : \mu_i = \mu_{i'} \quad vs \quad H_0 : \mu_i \neq \mu_{i'}$$

由于因子有 a 种水平

$$\binom{a}{2} = \frac{a(a-1)}{2}$$

那么置信水平较两个的就有放松，我们可以得到如下不太紧的界

$$\begin{aligned} P(\cap_{i=1}^k A_i) &\leq P(A_1) = 1 - \alpha \\ P(\cap_{i=1}^k A_i) &= 1 - P(\cup_{i=1}^k \bar{A}_i) \\ &\geq 1 - \sum_{i=1}^k P(\bar{A}_i) = 1 - k(1 - (1 - \alpha)) \\ &= 1 - k\alpha \end{aligned}$$

Bonferroni 方法

我们把每个事件发生的概率提高

$$t_{1-\alpha/2}(n-a) \rightarrow t_{1-\alpha/(a(a-1))}(n-a)$$

于是有

$$P\left(\bigcap_{i=1}^{a(a-1)/2} A_i\right) \geq 1 - a(a-1)/2 \cdot \frac{\alpha}{a(a-1)/2} = 1 - \alpha$$

但这样精度很差

多重比较问题

在 $a(a > 2)$ 个水平均值中同时比较任意两个水平均值间有无明显差异的问题称为多重比较。

检验问题

$$H_0^{ii'} : \mu_i = \mu_{i'}, \quad 1 \leq i < i' \leq a$$

于是拒绝域为 (至少一个不成立)

$$W = \bigcup_{1 \leq i < i' \leq a} \{|\bar{y}_{i\cdot} - \bar{y}_{i'}| \geq c_{ii'}\}$$

其中 $c_{ii'}$ 是临界值, 由原假设成立时 $P(W) = \alpha$ 确定。

由于各个水平下重复次数均相等, 基于对称性一个很自然的要求是 $c_{ii'}$ 是相等的, 我们记为 c 。

Tukey 方法

考虑多重比较的检验问题

$$H_0^{ii'} : \mu_i = \mu_{i'}, \quad 1 \leq i < i' \leq a$$

原假设成立时有

$$\mu_1 = \mu_2 = \dots = \mu_a = \mu$$

我们有

$$\begin{aligned}
P(W) &= P\left(\bigcup_{1 \leq i < i' \leq a} \{|\bar{y}_i - \bar{y}_{i'}| \geq c\}\right) \\
&= 1 - P\left(\bigcap_{1 \leq i < i' \leq a} \{|\bar{y}_i - \bar{y}_{i'}| < c\}\right) \\
&= 1 - P\left(\max_{1 \leq i < i' \leq a} |\bar{y}_i - \bar{y}_{i'}| < c\right) \\
&= P\left(\max_{1 \leq i < i' \leq a} |\bar{y}_i - \bar{y}_{i'}| \geq c\right) \\
&= P\left(\max_{1 \leq i < i' \leq a} \left| \frac{(\bar{y}_i - \mu) - (\bar{y}_{i'} - \mu)}{\hat{\sigma}/\sqrt{m}} \right| \geq \frac{c}{\hat{\sigma}/\sqrt{m}}\right) \\
&= P\left(\max_i \frac{\bar{y}_i - \mu}{\hat{\sigma}/\sqrt{m}} - \min_i \frac{\bar{y}_i - \mu}{\hat{\sigma}/\sqrt{m}} \geq \frac{c}{\hat{\sigma}/\sqrt{m}}\right)
\end{aligned}$$

t 化极差统计量

$$q(a, df) = \max_i \frac{\bar{y}_i - \mu}{\hat{\sigma}/\sqrt{m}} - \min_i \frac{\bar{y}_i - \mu}{\hat{\sigma}/\sqrt{m}}$$

其中

$$\frac{\bar{y}_i - \mu}{\hat{\sigma}/\sqrt{m}} \sim t(n - a)$$

给出蒙特卡洛算法

Algorithm 1 t 化极差统计量的蒙特卡洛分布

Require: 水平数目 a , t 分布的自由度 df , 重复次数 N ;

Ensure: t 化极差统计量的 N 个观测值

- 1: for $n = 1, 2, \dots, N$ do
 - 2: 从标准正态分布 $N(0, 1)$ 产生 a 个随机数: x_1, x_2, \dots, x_a ;
 - 3: 将 a 个数据进行排序, 令 x_{\max} 为最大值, x_{\min} 为最小值;
 - 4: 从自由度为 df 的 χ^2 分布产生一个随机数 y ;
 - 5: 计算 $q_n = (x_{\max} - x_{\min}) / \sqrt{y/df}$;
-

拒绝域

$$P(W) = P(q(a, df) \geq \sqrt{m}c/\hat{\sigma}) = \alpha$$

可推出

$$c = q_{1-\alpha}(a, df)\hat{\sigma}/\sqrt{m}$$

实际计算中需要比较每一组样本均值的差与临界值的大小。



Two-Way ANOVA

主效应与交互效应

这里考虑有两个因子的情况。

因子效应：当某一因子的水平改变时导致了响应变量发生变化。

主效应：某一因子对响应变量的直接影响。

交互效应：间接影响。

模型及假设

数据结构

		因子 B		
		1	2	...
因子 A	1	$y_{111}, y_{112}, \dots, y_{11m}$	$y_{121}, y_{122}, \dots, y_{12m}$	\dots
	2	$y_{211}, y_{212}, \dots, y_{21m}$	$y_{221}, y_{222}, \dots, y_{22m}$	\dots
	\vdots			
	a	$y_{a11}, y_{a12}, \dots, y_{a1m}$	$y_{a21}, y_{a22}, \dots, y_{a2m}$	\dots
				$y_{ab1}, y_{ab2}, \dots, y_{abm}$

均值模型

$$y_{ijk} = \mu_{ij} + \varepsilon_{ijk} \begin{cases} i = 1, 2, \dots, a \\ j = 1, 2, \dots, b \\ k = 1, 2, \dots, m \end{cases}$$

效应模型

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk} \begin{cases} i = 1, 2, \dots, a \\ j = 1, 2, \dots, b \\ k = 1, 2, \dots, m \end{cases}$$

效应模型的约束

$$\begin{aligned} \sum_{i=1}^a \alpha_i &= 0 \\ \sum_{j=1}^b \beta_j &= 0 \\ \sum_{i=1}^a (\alpha\beta)_{ij} &= \sum_{j=1}^b (\alpha\beta)_{ij} = 0 \end{aligned}$$

注意交互效应不是乘积，是一个组合。

假定实验误差满足

$$\varepsilon_{ijk} \sim N(0, \sigma^2)$$

综上，效应模型一般满足

$$y_{ijk} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \varepsilon_{ijk}, \quad \varepsilon_{ijk} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2)$$

$$i = 1, 2, \dots, a, j = 1, 2, \dots, b, k = 1, 2, \dots, m \\ s.t.$$

$$\sum_{i=1}^a \alpha_i = 0, \quad \sum_{j=1}^b \beta_j = 0$$

$$\sum_{i=1}^a (\alpha\beta)_{ij} = \sum_{j=1}^b (\alpha\beta)_{ij} = 0$$

检验

有三组检验

判断因子 A 是否对响应变量有影响

$$H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_a = 0 \quad vs \quad H_1 : \text{因子 } A \text{ 至少存在一个水平 } \alpha_i \neq 0$$

判断因子 B 是否对响应变量有影响

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_b = 0 \quad vs \quad H_1 : \text{因子 } B \text{ 至少存在一个水平 } \beta_j \neq 0$$

判断因子 A、B 是否对响应变量有交互影响

$$H_0 : \text{对于任意 } i = 1, 2, \dots, a, j = 1, 2, \dots, b, (\alpha\beta)_{ij} = 0 \text{ 均成立} \\ H_1 : \text{因子 } A \text{ 和 } B \text{ 至少存在一组水平组合 } (\alpha\beta)_{ij} \neq 0.$$

符号定义

$$y_{i..} = \sum_{j=1}^b \sum_{k=1}^m y_{ijk}, \quad \bar{y}_{i..} = \frac{1}{bm} y_{i..} \\ y_{.j} = \sum_{i=1}^a \sum_{k=1}^m y_{ijk}, \quad \bar{y}_{.j} = \frac{1}{am} y_{.j} \\ y_{ij} = \sum_{k=1}^m y_{ijk}, \quad \bar{y}_{ij} = \frac{1}{m} y_{ij} \\ y_{...} = \sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^m y_{ijk}, \quad \bar{y}_{...} = \frac{1}{n} y_{...}$$

平方和分解

$$\begin{aligned}
\sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^m (y_{ijk} - \bar{y}_{...})^2 &= \sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^m ((\bar{y}_{i..} - \bar{y}_{...}) + (\bar{y}_{.j.} - \bar{y}_{...}) \\
&\quad + (\bar{y}_{ij.} - \bar{y}_{i..} - \bar{y}_{.j.} + \bar{y}_{...}) + (y_{ijk} - \bar{y}_{ij.}))^2 \\
&= bm \sum_{i=1}^a (\bar{y}_{i..} - \bar{y}_{...})^2 + am \sum_{j=1}^b (\bar{y}_{.j.} - \bar{y}_{...})^2 \\
&\quad + m \sum_{i=1}^a \sum_{j=1}^b (\bar{y}_{ij.} - y_{i..} - \bar{y}_{.j.} + \bar{y}_{...})^2 \\
&\quad + \sum_{i=1}^a \sum_{j=1}^b \sum_{k=1}^m (y_{ijk} - \bar{y}_{ij.})^2
\end{aligned}$$

可以简记为

$$SS_T = SS_A + SS_B + SS_{AB} + SS_E$$

引理: Cochran 定理

假设

$$\mathbf{x} = (x_1, x_2, \dots, x_n)'$$

其中每个元素满独立同分布的标准正态分布, 令

$$\sum_{i=1}^n \mathbf{x}' \mathbf{x} = \sum_{i=1}^k \mathbf{x}' \mathbf{B}_i \mathbf{x}$$

其中

$$\text{rank}(\mathbf{B}_i) = r_i$$

于是我们有 $\mathbf{x}' \mathbf{B}_i \mathbf{x}$ 是相互独立的卡方随机变量 $\chi^2(r_i)$ 当且仅当 \mathbf{B}_i 是对称矩阵且

$$\sum_{i=1}^k r_i = n$$

这里我们只要找到四个对称矩阵, 使得这些偏差平方和可以写成二次型的结构。这里, 矩阵的秩对应了卡方分布的自由度。

我们可以得到以下自由度

效应	自由度
因子A	$a - 1$
因子B	$b - 1$
交互效应AB	$(a - 1)(b - 1)$
误差	$ab(m - 1)$
总和	$abm - 1$

均方: 偏差平方和除以相应的自由度。

均方的性质

期望性质

$$\begin{aligned}
 E(MS_A) &= E\left(\frac{SS_A}{a-1}\right) = \sigma^2 + \frac{bm \sum_{i=1}^a \alpha_i^2}{a-1} \\
 E(MS_B) &= E\left(\frac{SS_B}{b-1}\right) = \sigma^2 + \frac{am \sum_{j=1}^b \beta_j^2}{b-1} \\
 E(MS_{AB}) &= E\left(\frac{SS_{AB}}{(a-1)(b-1)}\right) = \sigma^2 + \frac{m \sum_{i=1}^a \sum_{j=1}^b (\alpha\beta)_{ij}^2}{(a-1)(b-1)} \\
 E(MS_E) &= E\left(\frac{SS_E}{ab(m-1)}\right) = \sigma^2
 \end{aligned}$$

分布性质

$$\begin{aligned}
 \frac{MS_A}{MS_E} &\sim F(a-1, ab(m-1)) \\
 \frac{MS_B}{MS_E} &\sim F(b-1, ab(m-1)) \\
 \frac{MS_{AB}}{MS_E} &\sim F((a-1)(b-1), ab(m-1))
 \end{aligned}$$

方差分析表

来源	平方和SS	自由度df	均方和MS	F值
因子A	SS_A	$a-1$	$MS_A = \frac{SS_A}{a-1}$	$F_A = \frac{MS_A}{MS_E}$
因子B	SS_B	$b-1$	$MS_B = \frac{SS_B}{b-1}$	$F_B = \frac{MS_B}{MS_E}$
交互效应AB	SS_{AB}	$(a-1)(b-1)$	$MS_{AB} = \frac{SS_{AB}}{(a-1)(b-1)}$	$F_{AB} = \frac{MS_{AB}}{MS_E}$
误差E	SS_E	$ab(m-1)$	$MS_E = \frac{SS_E}{ab(m-1)}$	
总和	SS_T	$n-1$		

方差分析部分到此结束★★, °*:..☆(￣▽￣)/\$:*.°★*。



线性回归

✓ 一元线性回归

💡 多元线性回归

考试

一元回归分析

解

$$\begin{cases} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 = l_{xx}^{-1} l_{xy} \end{cases}$$

定理

定理 1

如果有

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2)$$

那么有

$$\begin{aligned} \hat{\beta}_0 &\sim N\left(\beta_0, \left(\frac{1}{n} + \frac{\bar{x}^2}{l_{xx}}\right) \sigma^2\right), \hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma^2}{l_{xx}}\right); \\ \text{Cov}(\hat{\beta}_0, \hat{\beta}_1) &= -\frac{\bar{x}}{l_{xx}} \sigma^2 \end{aligned}$$

类似于方差分析

偏差平方和: 无模型的残差

$$SS_T = \sum_{i=1}^n (y_i - \bar{y})^2 = l_{yy}$$

回归平方和: 有模型的效应 (有模型与无模型的差距)

$$SS_R = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$$

残差平方和: 有模型的残差

$$SS_E = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

有平方和分解

$$SS_T = SS_R + SS_E$$

定理 2

$$\begin{aligned} E(SS_R) &= \sigma^2 + \beta_1^2 l_{xx} \\ E(SS_E) &= (n-2)\sigma^2 \end{aligned}$$

定理 3

若诸 y_i 相互独立, 有

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2), i = 1, 2, \dots, n$$

则有

1. $SS_E/\sigma^2 \sim \chi^2(n-2)$
2. 若 H_0 成立, 则有 $SS_R/\sigma^2 \sim \chi^2(1)$
3. SS_R 与 SS_E, \bar{y} 独立。

一元回归分析的方差分析表

来源	平方和	自由度	均方	F值
回归	SS_R	1	$MS_R = SS_R$	$F_0 = \frac{SS_R}{SS_E/(n-2)}$
误差	SS_E	$n-2$	$MS_E = \frac{SS_E}{n-2}$	
总和	SS_T	$n-1$		

定理 4

如果 y 是相互独立的且 y_i 是正态分布随机变量, 即 $y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2)$, 那么, 对给定的 x_0 , 有

$$\hat{y}_0 = \hat{\beta}_0 + \hat{\beta}_1 x_0 \sim N\left(\beta_0 + \beta_1 x_0, \left(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}\right) \sigma^2\right).$$

多元回归分析

解

$$\hat{\beta}_{\text{LS}} = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}$$

$$\hat{\mathbf{y}} = \mathbf{X} \hat{\beta} = \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} = \mathbf{H} \mathbf{y}$$

我们称 \mathbf{H} 为帽子矩阵。

定理

定理一：帽子矩阵的性质

1. n 阶对称矩阵
2. 幂等矩阵
3. 迹为 $p + 1$ (对称幂等矩阵秩和迹相等)

定理二：最小二乘估计的性质

1. $E(\hat{\beta}) = \beta$
2. $\text{Var}(\hat{\beta}) = \sigma^2 (\mathbf{X}' \mathbf{X})^{-1}$

定理三：最小二乘估计与残差线性不相关

即证

$$\text{Cov}(\hat{\beta}, e) = \mathbf{0}$$

我们同时也可以得到一个推论：最小二乘估计与残差平方和独立。

规范化处理

中心化

中心化的因变量与未中心化的因变量之间的关系

$$\mathbf{y}^* = \mathbf{y} - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{y} = (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}$$

中心化的自变量与未中心化的自变量之间的关系

$$\begin{aligned} \mathbf{X}_c &= \mathbf{X}_o - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{X}_o = \left(\mathbf{I}_n - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \right) \mathbf{X}_o \\ &= (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{X}_o \end{aligned}$$

$$\hat{\beta}_{c, \text{slope}} = \hat{\beta}_{\text{slope}}$$

标准化

$$\begin{aligned} x_{ij}^{**} &= \frac{x_{ij}^*}{\sqrt{L_{jj}}} = \frac{x_{ij} - \bar{x}_j}{\sqrt{L_{jj}}}, \quad i = 1, 2, \dots, n; j = 1, 2, \dots, p. \\ y_i^{**} &= \frac{y_i^*}{\sqrt{L_{yy}}}, \quad i = 1, 2, \dots, n. \end{aligned}$$

其中, L_{jj} 是自变量 x_j 的离差平方和

$$L_{jj} = \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2$$

L_{yy} 是因变量 y 的离差平方和

$$L_{yy} = \sum_{i=1}^n (y_i - \bar{y})^2$$

$$\begin{aligned}\mathbf{y}^{**} &= \left(\frac{y_1 - \bar{y}}{\sqrt{L_{yy}}}, \dots, \frac{y_n - \bar{y}}{\sqrt{L_{yy}}} \right)' = \frac{1}{\sqrt{L_{yy}}} \mathbf{y}^* \\ \mathbf{X}_s &= \left(\frac{1}{\sqrt{L_{11}}} \mathbf{x}_1^*, \dots, \frac{1}{\sqrt{L_{pp}}} \mathbf{x}_p^* \right) \\ &= \mathbf{X}_c \mathbf{L}.\end{aligned}$$

$$\hat{\boldsymbol{\beta}}_{s, \text{slope}} = \frac{1}{\sqrt{L_{yy}}} \mathbf{L}^{-1} \hat{\boldsymbol{\beta}}_{c, \text{slope}}$$

F 检验

原假设

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_p = 0$$

备择假设

$$H_1 : \text{存在 } \beta_j \text{ 不为零}, j = 1, 2, \dots, p.$$

离差平方和

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

记为

$$SS_T = SS_R + SS_E$$

检验统计量

$$F_0 = \frac{SS_R/p}{SS_E/(n-p-1)}$$

定理四：偏差平方和的分布

在正态假设下，有

1. $SS_E/\sigma^2 \sim \chi^2(n-p-1)$ ，其中 $SS_E = \mathbf{e}'\mathbf{e}$
2. SS_E 和 SS_R 独立
3. 在 H_0 下， $SS_R/\sigma^2 \sim \chi^2(p)$

方差分析表

来源	平方和	自由度	均方	F值	p值
回归	SS_R	p	$MS_R = \frac{SS_R}{p}$	$F_0 = \frac{MS_R}{MS_E}$	$p_0 = P(F \geq F_0)$
误差	SS_E	$n - p - 1$	$MS_E = \frac{SS_E}{(n-p-1)}$		
总和	SS_T	$n - 1$			

估计与预测

定理五：参数估计的性质

在正态假设下，即 $\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n)$ ，有

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2 (\mathbf{X}' \mathbf{X})^{-1})$$

复相关系数

拟合优度可以用来度量回归方程对样本观测值的拟合程度。（ x 的波动能够解释多少 y 的波动）

在多元线性回归中，定义样本决定系数为

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T}$$

定理六：预测值的无偏性

\hat{y}_0 是 y_0 的无偏预测，即

$$E(\hat{y}_0) = E(y_0)$$

定理七：最小方差无偏估计

假定 $\hat{\boldsymbol{\beta}}$ 是 $\boldsymbol{\beta}$ 的 OLS，对于任意常数向量 \mathbf{c} ， $\mathbf{c}'\hat{\boldsymbol{\beta}}$ 是 $\mathbf{c}'\boldsymbol{\beta}$ 的最小方差无偏估计 (BLUE)。

预测值的分布

分布

$$\hat{y}_0 \sim N(\mathbf{x}'_0 \boldsymbol{\beta}, \sigma^2 \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0)$$

$$\hat{y}_0 - y_0 \sim N(0, \sigma^2 (1 + \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0))$$



一元线性回归

背景

回归分析是由 19 世纪英国统计学家 **高尔顿**首先提出的。

高尔顿研究父子升高的统计规律，发现有**回归方程**：

$$\hat{y} = 33.73 + 0.516x$$

在实际问题中，感兴趣的变量 y 与易于获得的变量 x 之间存在紧密关联，但又不由 x 而唯一确定的，这种关系成为**统计关系**。

当给定 x 的取值时， y 的取值是无法唯一确定的，我们通常可以认为是一个**随机变量**。

给定 x 时，称 y 的**条件数学期望**为 y 关于 x 的回归函数。

$$f(x) = E(y | x)$$

对于**线性模型**，可以把 f 看作 x 的**线性方程**。

对于**神经网络**，可以把 f 看作 x 的**非线性方程**。

对于**深度学习**，可以把 f 看作 x 的**多个非线性方程的复合**。

模型

一元线性回归

$$y = \beta_0 + \beta_1 x + \varepsilon$$

其**数学模型**为

$$y = \beta_0 + \beta_1 x$$

随机误差用来概括由于人们认识的局限性和客观原因导致的种种偶然因素。

通常假定：

$$\begin{cases} E(\varepsilon) = 0 \\ \text{Var}(\varepsilon) = \sigma^2 < \infty \end{cases}$$

回归方程

$$E(y | x) = \beta_0 + \beta_1 x$$

由于参数是未知的，因此需要通过样本估计，假定样本是**独立观测**的，即随机变量**独立同分布**。

注意：这里仅讨论样本是确定的。随即情况请见附录。

模型版

$$y = \beta_0 + \beta_1 x + \varepsilon$$

数据版

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, i = 1, 2, \dots, n$$

一元线性经验回归方程

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

当给定取值时，称

$$\hat{y}_0 = \hat{\beta}_0 + \hat{\beta}_1 x_0$$

为回归值，拟合值或预测值。

参数估计

最小二乘估计

定义残差

$$e_i = y_i - E(y_i | x_i) = y_i - \beta_0 - \beta_1 x_i$$

偏差平方和

$$\begin{aligned} Q(\beta_0, \beta_1) &= \sum_{i=1}^n (y_i - E(y_i))^2 \\ &= \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2. \end{aligned}$$

最小二乘估计

$$\begin{aligned} (\hat{\beta}_0, \hat{\beta}_1) &= \arg \min_{\beta_0, \beta_1} Q(\beta_0, \beta_1) \\ &= \arg \min_{\beta_0, \beta_1} \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2 \end{aligned}$$

求驻点：一阶导数为 0

$$\frac{\partial Q}{\partial \beta_0} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) = 0 \quad (1)$$

$$\frac{\partial Q}{\partial \beta_1} = -2 \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i) x_i = 0 \quad (2)$$

正规方程组

$$\begin{cases} n\beta_0 + n\bar{x}\beta_1 = n\bar{y} \\ n\bar{x}\beta_0 + \sum_{i=1}^n x_i^2\beta_1 = \sum_{i=1}^n x_i y_i \end{cases}$$

于是，估计为

$$\begin{cases} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \end{cases}$$

记号

$$\begin{aligned} l_{xx} &= \sum_{i=1}^n (x_i - \bar{x})^2 = \sum_{i=1}^n x_i^2 - n(\bar{x})^2 \\ l_{xy} &= \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = \sum_{i=1}^n x_i y_i - n\bar{x} \cdot \bar{y}, \end{aligned}$$

那么有

$$\begin{cases} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 = l_{xx}^{-1} l_{xy} \end{cases}$$

应该计算二阶偏导

$$\begin{aligned} \left| \begin{pmatrix} 2n & 2n\bar{x} \\ 2n\bar{x} & 2\sum_{i=1}^n x_i^2 \end{pmatrix} \right| &= 4n \sum_{i=1}^n x_i^2 - 4n^2(\bar{x})^2 \\ &= 4n \sum_{i=1}^n (x_i - \bar{x})^2 \\ &> 0 \end{aligned}$$

于是证明这确实是极小值。

极大似然估计

依赖似然函数

$$L(\theta; x_1, x_2, \dots, x_n) = \prod_{i=1}^n f(x_i; \theta)$$

最大似然估计

$$\hat{\theta} = \arg \max_{\theta} L(\theta; x_1, x_2, \dots, x_n)$$

分布假定

$$\varepsilon \sim N(0, \sigma^2)$$

$$\varepsilon_i \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2), \quad i = 1, 2, \dots, n$$

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2), \quad i = 1, 2, \dots, n$$

我们给出密度函数

$$f(y_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{1}{2\sigma^2} [y_i - (\beta_0 + \beta_1 x_i)]^2 \right\}$$

似然函数为 (注意有三个参数)

$$\begin{aligned} L(\beta_0, \beta_1, \sigma^2) &= \prod_{i=1}^n f_i(y_i) \\ &= (2\pi\sigma^2)^{-\frac{n}{2}} \exp \left\{ -\frac{1}{2\sigma^2} \sum_{i=1}^n [y_i - (\beta_0 + \beta_1 x_i)]^2 \right\} \end{aligned}$$

易证其对数似然和似然函数的最大值点是相同的, 于是有

$$\ln L(\beta_0, \beta_1, \sigma^2) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2$$

对系数求导后有

$$\begin{cases} \hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x} \\ \hat{\beta}_1 = l_{xx}^{-1} l_{xy} \end{cases}$$

发现

$$\arg \max_{\beta_0, \beta_1} \ln L(\beta_0, \beta_1, \sigma^2) \Leftrightarrow \arg \min_{\beta_0, \beta_1} Q(\beta_0, \beta_1)$$

我们同样可以对方差求导

$$\frac{\partial \ln L}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} \sum_{i=1}^n (y_i - (\beta_0 + \beta_1 x_i))^2 = 0$$

于是最大似然估计有

$$\begin{aligned} \hat{\sigma}_{\text{ML}}^2 &= \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \\ &= \frac{1}{n} \sum_{i=1}^n (y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2 \end{aligned}$$

实际上, 我们更加需要无偏估计, 即

$$\begin{aligned} \hat{\sigma}^2 &= \frac{1}{n-2} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \\ &= \frac{1}{n-2} \sum_{i=1}^n (y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i))^2. \end{aligned}$$

定理 1

如果有

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2)$$

那么有

$$\hat{\beta}_0 \sim N \left(\beta_0, \left(\frac{1}{n} + \frac{\bar{x}^2}{l_{xx}} \right) \sigma^2 \right), \hat{\beta}_1 \sim N \left(\beta_1, \frac{\sigma^2}{l_{xx}} \right);$$

$$\text{Cov} \left(\hat{\beta}_0, \hat{\beta}_1 \right) = -\frac{\bar{x}}{l_{xx}} \sigma^2$$

展开回归系数

$$\begin{aligned}\hat{\beta}_1 &= l_{xx}^{-1} l_{xy} = l_{xx}^{-1} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \\ &= l_{xx}^{-1} \left(\sum_{i=1}^n (x_i - \bar{x}) y_i - \sum_{i=1}^n (x_i - \bar{x}) \bar{y} \right) \\ &= l_{xx}^{-1} \left(\sum_{i=1}^n (x_i - \bar{x}) y_i \right) \\ &= \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} y_i\end{aligned}$$

$$\begin{aligned}\hat{\beta}_0 &= \bar{y} - \hat{\beta}_1 \bar{x} \\ &= \frac{1}{n} \sum_{i=1}^n y_i - \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} y_i \bar{x} \\ &= \sum_{i=1}^n \left(\frac{1}{n} - \frac{\bar{x}(x_i - \bar{x})}{l_{xx}} \right) y_i\end{aligned}$$

我们在这里已经能够得到回归系数满足正态分布的性质了，因为可以发现回归系数其实是**正态分布的加权求和**。

进一步考虑**期望与方差**

$$\begin{aligned}E(\hat{\beta}_1) &= \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} E(y_i) = \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} (\beta_0 + \beta_1 x_i) \\ &= \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} \beta_0 + \sum_{i=1}^n \frac{x_i (x_i - \bar{x})}{l_{xx}} \beta_1 \\ &= \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} \beta_0 + \sum_{i=1}^n \frac{(x_i - \bar{x}) x_i}{l_{xx}} \beta_1 - \sum_{i=1}^n \frac{\bar{x} (x_i - \bar{x})}{l_{xx}} \beta_1 \\ &= \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} \beta_0 + \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{l_{xx}} \beta_1 = \beta_1 \\ \text{Var}(\hat{\beta}_1) &= \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{l_{xx}} \right)^2 \text{Var}(y_i) = \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{(l_{xx})^2} \sigma^2 = \frac{\sigma^2}{l_{xx}}\end{aligned}$$

$$\begin{aligned}E(\hat{\beta}_0) &= E(\bar{y}) - E(\hat{\beta}_1) \bar{x} \\ &= \frac{1}{n} \sum_{i=1}^n (\beta_0 + \beta_1 x_i) - \beta_1 \bar{x} \\ &= \beta_0 + \beta_1 \bar{x} - \beta_1 \bar{x} \\ &= \beta_0 \\ \text{Var}(\hat{\beta}_0) &= \sum_{i=1}^n \left(\frac{1}{n} - \frac{(x_i - \bar{x}) \bar{x}}{l_{xx}} \right)^2 \text{Var}(y_i) \\ &= \sigma^2 \sum_{i=1}^n \left(\frac{1}{n^2} - \frac{2(x_i - \bar{x}) \bar{x}}{nl_{xx}} + \frac{(x_i - \bar{x})^2 \bar{x}^2}{l_{xx}^2} \right) \\ &= \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{l_{xx}} \right)\end{aligned}$$

证明中用了**加一项 0** 的技巧，配凑出一些比较好的性质。

由于各项 y 独立，有

$$\begin{aligned}\text{Cov}(\hat{\beta}_0, \hat{\beta}_1) &= \text{Cov}\left(\sum_{i=1}^n \left(\frac{1}{n} - \frac{(x_i - \bar{x})\bar{x}}{l_{xx}}\right) y_i, \sum_{i=1}^n \frac{(x_i - \bar{x})}{l_{xx}} y_i\right) \\ &= \sum_{i=1}^n \left(\frac{1}{n} - \frac{(x_i - \bar{x})\bar{x}}{l_{xx}}\right) \frac{(x_i - \bar{x})}{l_{xx}} \sigma^2 \\ &= -\frac{\bar{x}}{l_{xx}} \sigma^2\end{aligned}$$

其中交叉项肯定为 0。

我们可以从中给出一些**推论**：

1. 我们给出了无偏估计 $\hat{\beta}_0$ 与 $\hat{\beta}_1$
2. 除了 $\bar{x} = 0$ 外， $\hat{\beta}_0$ 与 $\hat{\beta}_1$ 相关。
3. 提高估计精度：增加样本量或者是增加 l_{xx} ，要求数据分散。

显著性检验

概述

主要检测 β_1 是否为 0

检验问题如下

$$H_0 : \beta_1 = 0 \quad vs \quad H_1 : \beta_1 \neq 0$$

F 检验

类似于方差分析

偏差平方和：无模型的残差

$$SS_T = \sum_{i=1}^n (y_i - \bar{y})^2 = l_{yy}$$

回归平方和：有模型的效应（有模型与无模型的差距）

$$SS_R = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$$

残差平方和：有模型的残差

$$SS_E = \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

有平方和分解

定理 2

$$\begin{aligned} E(SS_R) &= \sigma^2 + \beta_1^2 l_{xx} \\ E(SS_E) &= (n-2)\sigma^2 \end{aligned}$$

首先计算 SS_R 的期望

$$SS_R = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 = \sum_{i=1}^n \left((\hat{\beta}_0 + \hat{\beta}_1 x_i) - (\hat{\beta}_0 + \hat{\beta}_1 \bar{x}) \right)^2 = \hat{\beta}_1^2 l_{xx}$$

$$\begin{aligned} E(SS_R) &= E(\hat{\beta}_1^2) l_{xx} = \left(\text{Var}(\hat{\beta}_1) + (E(\hat{\beta}_1))^2 \right) l_{xx} \\ &= \left(\frac{\sigma^2}{l_{xx}} + \beta_1^2 \right) l_{xx} \\ &= \sigma^2 + \beta_1^2 l_{xx} \end{aligned}$$

然后计算 SS_E 的期望

$$\begin{aligned} SS_E &= \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n \left(\beta_0 + \beta_1 x_i + \varepsilon_i - \hat{\beta}_0 - \hat{\beta}_1 x_i \right)^2 \\ &= \sum_{i=1}^n \left((\beta_0 - \hat{\beta}_0) + (\beta_1 - \hat{\beta}_1) x_i + \varepsilon_i \right)^2 \\ &= \sum_{i=1}^n \left((\beta_0 - \hat{\beta}_0)^2 + (\beta_1 - \hat{\beta}_1)^2 x_i^2 + \varepsilon_i^2 + 2(\beta_0 - \hat{\beta}_0)(\beta_1 - \hat{\beta}_1) \right. \\ &\quad \left. + 2(\beta_0 - \hat{\beta}_0)\varepsilon_i + 2(\beta_1 - \hat{\beta}_1)x_i\varepsilon_i \right) \end{aligned}$$

$$\begin{aligned} E(SS_E) &= n \text{Var}(\hat{\beta}_0) + \text{Var}(\hat{\beta}_1) \sum_{i=1}^n x_i^2 + n \text{Var}(\varepsilon_i) + 2n \text{Cov}(\hat{\beta}_1, \hat{\beta}_0) \\ &\quad - 2 \sum_{i=1}^n E(\hat{\beta}_0 \varepsilon_i) - 2 \sum_{i=1}^n x_i E(\hat{\beta}_1 \varepsilon_i). \end{aligned}$$

由于有

$$\hat{\beta}_0 = \sum_i \left(\frac{1}{n} - \frac{(x_i - \bar{x})}{l_{xx}} \right) y_i; \quad \hat{\beta}_1 = \sum_i \frac{(x_i - \bar{x})}{l_{xx}} y_i$$

有

$$\begin{aligned} E(\hat{\beta}_0 \varepsilon_i) &= E \left(\varepsilon_i \sum_j \left(\frac{1}{n} - \frac{(x_j - \bar{x})}{l_{xx}} \right) y_j \right) \\ &= E \left(\varepsilon_i \sum_j \left(\frac{1}{n} - \frac{(x_j - \bar{x})}{l_{xx}} \right) (\beta_0 + \beta_1 x_j + \varepsilon_j) \right) \\ &= E \left(\varepsilon_i^2 \left(\frac{1}{n} - \frac{(x_i - \bar{x})}{l_{xx}} \right) \right) \\ &= \left(\frac{1}{n} - \frac{(x_i - \bar{x})}{l_{xx}} \right) \sigma^2. \end{aligned}$$

$$\begin{aligned}
E(\hat{\beta}_1 \varepsilon_i) &= E\left(\varepsilon_i \sum_j \frac{(x_j - \bar{x})}{l_{xx}} y_j\right) \\
&= E\left(\varepsilon_i \sum_j \frac{(x_j - \bar{x})}{l_{xx}} (\beta_0 + \beta_1 x_j + \varepsilon_j)\right) \\
&= \frac{(x_i - \bar{x})}{l_{xx}} \sigma^2
\end{aligned}$$

$$\begin{aligned}
\sum_{i=1}^n E(\hat{\beta}_0 \varepsilon_i) &= \sum_{i=1}^n \left(\frac{1}{n} - \frac{(x_i - \bar{x}) \bar{x}}{l_{xx}} \right) \cdot \sigma^2 = \sigma^2 \\
\sum_{i=1}^n E(\hat{\beta}_1 \varepsilon_i) &= \sum_{i=1}^n x_i \frac{(x_i - \bar{x})}{l_{xx}} \sigma^2 = \sigma^2
\end{aligned}$$

$$\begin{aligned}
E(SS_E) &= n \cdot \left(\frac{1}{n} + \frac{\bar{x}^2}{l_{xx}} \right) \sigma^2 + \frac{\sigma^2}{l_{xx}} \cdot \sum_i x_i^2 + n\sigma^2 - 2n \cdot \frac{\bar{x}^2}{l_{xx}} \sigma^2 - 2\sigma^2 - 2\sigma^2 \\
&= \sigma^2 \left(1 + \frac{\sum x_i^2}{l_{xx}} - \frac{n\bar{x}^2}{l_{xx}} + n - 4 \right) \\
&= (n-2)\sigma^2.
\end{aligned}$$

定理 3

若诸 y_i 相互独立, 有

$$y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2), i = 1, 2, \dots, n$$

则有

1. $SS_E/\sigma^2 \sim \chi^2(n-2)$
2. 若 H_0 成立, 则有 $SS_R/\sigma^2 \sim \chi^2(1)$
3. SS_R 与 SS_E, \bar{y} 独立。

证明

构造正交矩阵 A

$$A = \{a_{ij}\}_{n \times n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,p} \\ \vdots & \vdots & & \vdots \\ a_{n-2,1} & a_{n-2,2} & \cdots & a_{n-2,p} \\ \frac{x_1 - \bar{x}}{\sqrt{l_{xx}}} & \frac{x_2 - \bar{x}}{\sqrt{l_{xx}}} & \cdots & \frac{x_n - \bar{x}}{\sqrt{l_{xx}}} \\ \frac{1}{\sqrt{n}} & \frac{1}{\sqrt{n}} & \cdots & \frac{1}{\sqrt{n}} \end{pmatrix}$$

单位阵满足

$$\sum_k a_{i,k} a_{j,k} = 0, \quad 1 \leq i < j \leq n-2;$$

$$\frac{1}{\sqrt{n}} \sum_j a_{i,j} = 0$$

$$\sum_j a_{i,j} \frac{x_j - \bar{x}}{\sqrt{l_{xx}}} = 0$$

令 $z = Ay$ ，其中 $z = (z_1, z_2, \dots, z_n)'$ 满足

$$z_i = \sum_j a_{ij} y_j, \quad i = 1, \dots, n-2$$

$$z_{n-1} = \frac{\sum_j (x_j - \bar{x}) y_j}{\sqrt{l_{xx}}} = \frac{\sum_j (x_j - \bar{x})(y_j - \bar{y})}{\sqrt{l_{xx}}} = \frac{l_{xy}}{\sqrt{l_{xx}}} = \sqrt{l_{xx}} \cdot \frac{l_{xy}}{l_{xx}} = \sqrt{l_{xx}} \hat{\beta}_1$$

$$z_n = \frac{1}{\sqrt{n}} \sum_j y_j = \sqrt{n} \bar{y}$$

这里可以发现两个很好的等式

$$z_i^2 = SS_E, \quad i = 1, \dots, n-2$$

$$z_{n-1}^2 = SS_R = l_{yy}$$

那么， z 符合 n 维正态分布，且均值和方差分别为

$$\begin{aligned} E(z_i) &= E\left(\sum_j a_{ij} y_j\right) = \sum_j a_{ij} (\beta_0 + \beta_1 x_j) \\ &= \beta_0 \sum_j a_{ij} + \beta_1 \sum_j a_{ij} x_j = 0, \quad i = 1, \dots, n-2; \\ E(z_{n-1}) &= \sqrt{l_{xx}} \cdot \beta_1; \\ E(z_n) &= \sqrt{n} (\beta_0 + \beta_1 \bar{x}); \\ \text{Var}(z) &= \text{Var}(Ay) = A \text{Var}(y) A' = A \sigma^2 I_n A' = \sigma^2 I_n. \end{aligned}$$

我们可以得到以下结论：

1. 各分量相互独立
2. 前 $n-2$ 个分量独立同分布（正态情况下不相关与独立等价），且分布为 $N(0, \sigma^2)$
3. z_{n-1} 的分布为 $N(\sqrt{l_{xx}} \beta_1, \sigma^2)$
4. z_n 的分布为 $N(\sqrt{n}(\beta_0 + \beta_1 \bar{x}), \sigma^2)$

因为

$$\begin{aligned} \sum_{i=1}^n z_i^2 &= z' z = y' A' A y = y' y = \sum_i y_i^2 = SS_T + n \bar{y}^2, \\ z_{n-1} &= \sqrt{l_{xx}} \hat{\beta}_1 = \sqrt{SS_R}, \\ z_n &= \sqrt{n} \bar{y}. \end{aligned}$$

所以有

$$SS_T + n \bar{y}^2 = \sum_{i=1}^{n-2} z_i^2 + SS_R + n \bar{y}^2$$

即有

$$SS_T = \sum_{i=1}^{n-2} z_i^2 + SS_R$$

因此有

$$SS_E = SS_T - SS_R = \sum_{i=1}^{n-2} z_i^2 \sim \chi^2(n-2)$$

在 $\beta_1 = 0$ 时, 因为

$$z_{n-1} \sim N(0, \sigma^2)$$

所以有

$$\frac{SS_R}{\sigma^2} = \left(\frac{z_{n-1}}{\sigma} \right)^2 \sim \chi^2(1)$$

因为 SS_E 与前 $n-2$ 个 z_i 有关, SS_R 仅与 z_{n-1} 有关, \bar{y} 仅与 z_n 有关, 因此 SS_E 、 SS_R 和 \bar{y} 三者相互独立。

因为 $\hat{\beta}_1$ 仅与 SS_R 有关, 所以 $\hat{\beta}_1$ 、 SS_E 和 \bar{y} 三者相互独立。

可以构造

$$F_0 = \frac{SS_R}{SS_E/(n-2)}$$

进行检验, 在 $\beta_1 = 0$ 时, 有拒绝域

$$F_0 \geq F_{1-\alpha}(1, n-2)$$

一元回归分析的方差分析表

来源	平方和	自由度	均方	F值
回归	SS_R	1	$MS_R = SS_R$	$F_0 = \frac{SS_R}{SS_E/(n-2)}$
误差	SS_E	$n-2$	$MS_E = \frac{SS_E}{n-2}$	
总和	SS_T	$n-1$		

t 检验

由于

$$\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma^2}{l_{xx}}\right) \quad \frac{SS_E}{\sigma^2} \sim \chi^2(n-2)$$

且 $\hat{\beta}_1$ 相互独立, 因此在 H_0 为真时, 有

$$t_0 = \frac{\hat{\beta}_1}{\hat{\sigma}/\sqrt{l_{xx}}} \sim t(n-2)$$

其中 $\hat{\sigma} = \sqrt{SS_E/(n-2)}$ 。

拒绝域

$$W = \{|t_0| > t_{1-\alpha/2}(n-2)\}$$

相关系数检验

$$H_0 : \rho = 0 \quad v.s. \quad H_1 : \rho \neq 0$$

样本相关系数

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{l_{xy}}{\sqrt{l_{xx}l_{yy}}},$$

拒绝域

$$P(W) = P(|r| \geq c) = \alpha$$

三种检验的关系

t 和 F 的关系

$$t_0^2 = \left(\frac{\hat{\beta}_1}{\hat{\sigma}/\sqrt{l_{xx}}} \right)^2 = \frac{\hat{\beta}_1^2 l_{xx}}{SS_E/(n-2)} = \frac{SS_R}{SS_E/(n-2)} = F_0$$

第三个等式成立的原因是

$$\begin{aligned} SS_R &= \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 = \sum_{i=1}^n (\hat{\beta}_0 + \hat{\beta}_1 x_i - \bar{y})^2 \\ &= \sum_{i=1}^n [\bar{y} + \hat{\beta}_1 (x_i - \bar{x}) - \bar{y}]^2 = \sum_{i=1}^n [\hat{\beta}_1 (x_i - \bar{x})]^2 = \hat{\beta}_1^2 l_{xx} \end{aligned}$$

r 和 F 的关系

$$\begin{aligned} r^2 &= \left(\hat{\beta}_1 \sqrt{\frac{l_{xx}}{l_{yy}}} \right)^2 = \hat{\beta}_1^2 \frac{l_{xx}}{l_{yy}} = \frac{SS_R}{SS_T} = \frac{SS_R}{SS_R + SS_E} \\ &= \frac{SS_R / (SS_E/(n-2))}{SS_R / (SS_E/(n-2)) + (n-2)} = \frac{F_0}{F_0 + (n-2)} \end{aligned}$$

分位数

$$r_{1-\alpha/2}(n-2) = \sqrt{\frac{F_{1-\alpha}(1, n-2)}{F_{1-\alpha}(1, n-2) + (n-2)}}$$

样本决定系数

$$r^2 = \frac{SS_R}{SS_T}$$

估计与预测

当给定 x_0 时, 我们关心的是

$$y_0 = \beta_0 + \beta_1 x_0 + \varepsilon_0$$

1. 估计: 我想得到 $E(y_0)$, 一个点估计。
2. 预测: 我想得到 y_0 , 因为 y_0 是一个连续随机变量, 因此是一个区间估计。

关于期望的点估计

定理 4

如果 y 是相互独立的且 y_i 是正态分布随机变量, 即 $y_i \sim N(\beta_0 + \beta_1 x_i, \sigma^2)$, 那么, 对给定的 x_0 , 有

$$\hat{y}_0 = \hat{\beta}_0 + \hat{\beta}_1 x_0 \sim N\left(\beta_0 + \beta_1 x_0, \left(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}\right) \sigma^2\right).$$

证明

$$\begin{aligned} E(\hat{y}_0) &= E(\hat{\beta}_0) + E(\hat{\beta}_1) x_0 = \beta_0 + \beta_1 x_0 \\ \text{Var}(\hat{y}_0) &= \text{Var}(\hat{\beta}_0 + \hat{\beta}_1 x_0) \\ &= \text{Var}(\hat{\beta}_0) + \text{Var}(\hat{\beta}_1) x_0^2 + 2 \text{Cov}(\hat{\beta}_0, \hat{\beta}_1) x_0 \\ &= \left(\frac{1}{n} + \frac{\bar{x}^2}{l_{xx}}\right) \cdot \sigma^2 + \frac{x_0^2}{l_{xx}} \cdot \sigma^2 - \frac{2\bar{x}x_0}{l_{xx}} \cdot \sigma^2 \\ &= \left(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}\right) \sigma^2 \end{aligned}$$

所以

$$\hat{y}_0 \sim N\left(\beta_0 + \beta_1 x_0, \left(\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}\right) \sigma^2\right).$$

但 σ^2 未知, 因此用其估计去替代

$$SS_E / \sigma^2 \sim \chi^2(n - 2)$$

$$\hat{\sigma}^2 = \frac{SS_E}{n - 2}$$

同时, 我们注意到

$$\hat{y}_0 = \bar{y} + \hat{\beta}_1 (x_0 - \bar{x}) \rightarrow \hat{y}_0 \perp SS_E$$

于是有

$$\frac{(\hat{y}_0 - E(y_0)) / \sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})}{l_{xx}} \sigma}}{\sqrt{\frac{SS_E}{\sigma^2} / (n - 2)}} = \frac{\hat{y}_0 - E(y_0)}{\hat{\sigma} \sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}}} \sim t(n - 2)$$

因此有置信区间

$$[\hat{y}_0 - \delta_0, \hat{y}_0 + \delta_0]$$

其中

$$\delta_0 = t_{1-\alpha/2}(n - 2) \hat{\sigma} \sqrt{\frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}}$$

关于预测值的区间估计

我们知道

$$y_0 = E(y_0) + \varepsilon_0$$

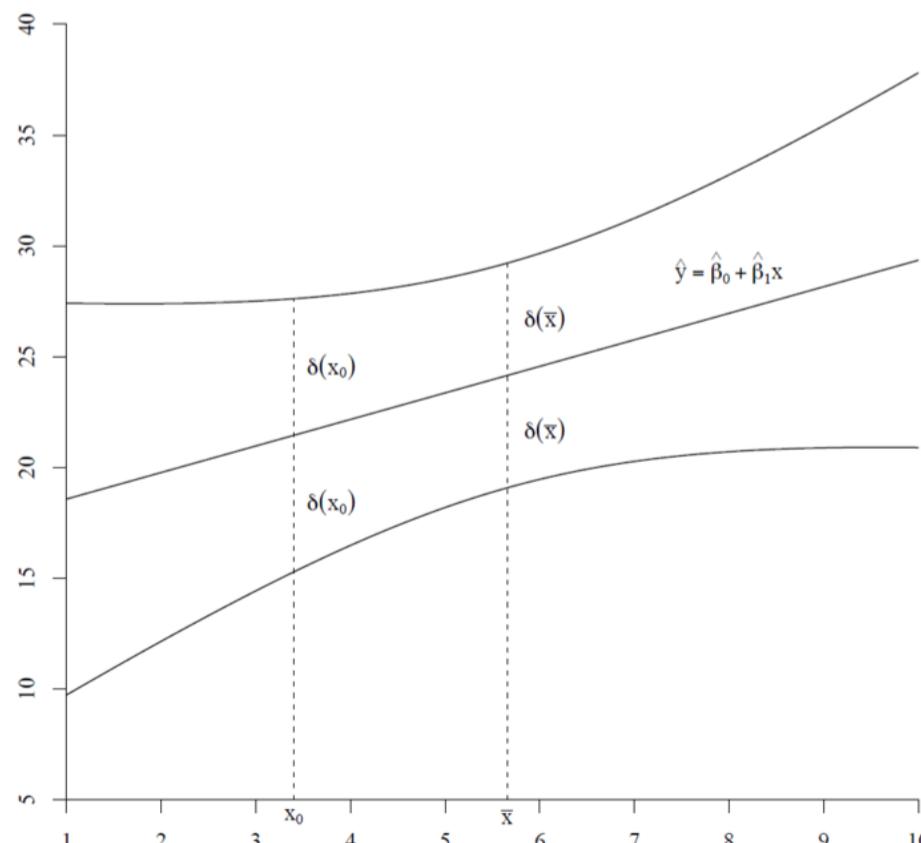
因此，预测区间拓展为

$$[\hat{y}_0 - \delta_1, \hat{y}_0 + \delta_1]$$

其中

$$\delta_1 = t_{1-\alpha/2}(n - 2) \hat{\sigma} \sqrt{1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{l_{xx}}}$$

下图展示了精确预测区间长度



我们看到随着 x 离均值越远， $\delta(x)$ 越大。

因此，如何使得预测比较准确有几种方法：

1. 尽量不要预测数据点范围以外的内容，这种情况下精度很低

2. 适当增加样本量
3. 尽量使得样本分布均匀



多元线性回归

模型与假设

线性回归模型：

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon$$

随机误差有：

$$\begin{cases} E(\varepsilon) = 0 \\ \text{Var}(\varepsilon) = \sigma^2 \end{cases}$$

实际中，我们可以写出如下方程组：

$$\begin{cases} y_1 = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_p x_{1p} + \varepsilon_1 \\ y_2 = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_p x_{2p} + \varepsilon_2 \\ \vdots \\ y_n = \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_p x_{np} + \varepsilon_n \end{cases}$$

令

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad \mathbf{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

$$\begin{aligned} \boldsymbol{\beta} &= (\beta_0, \beta_1, \beta_2, \cdots, \beta_p)' \\ \boldsymbol{\varepsilon} &= (\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_n)' \end{aligned}$$

线性回归模型的矩阵形式

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

注：

1. $\boldsymbol{\beta}$ 是 $p + 1$ 维的
2. \mathbf{X} 的大小为 $n \times (p + 1)$

问：如何估计回归系数 $\boldsymbol{\beta}$

基本假定：

1. 设计矩阵 X

a. 是确定性变量，不是随机变量

b. 列满秩

$$\text{rank}(X) = p + 1 < n$$

2. 随机误差是零均值和等方差的 (高斯-马尔可夫假设)

$$E(\varepsilon_i) = 0, \quad i = 1, 2, \dots, n$$

$$\text{Cov}(\varepsilon_i, \varepsilon_j) = \begin{cases} \sigma^2, & i = j \\ 0, & i \neq j \end{cases} \quad i, j = 1, 2, \dots, n$$

3. 假定随机误差项服从正态分布

$$\begin{cases} \varepsilon_i \sim N(0, \sigma^2), & i = 1, 2, \dots, n \\ \varepsilon_1, \varepsilon_2, \dots, \varepsilon_n \text{ 相互独立} \end{cases}$$

因变量 \mathbf{y} 的期望向量和协方差矩阵

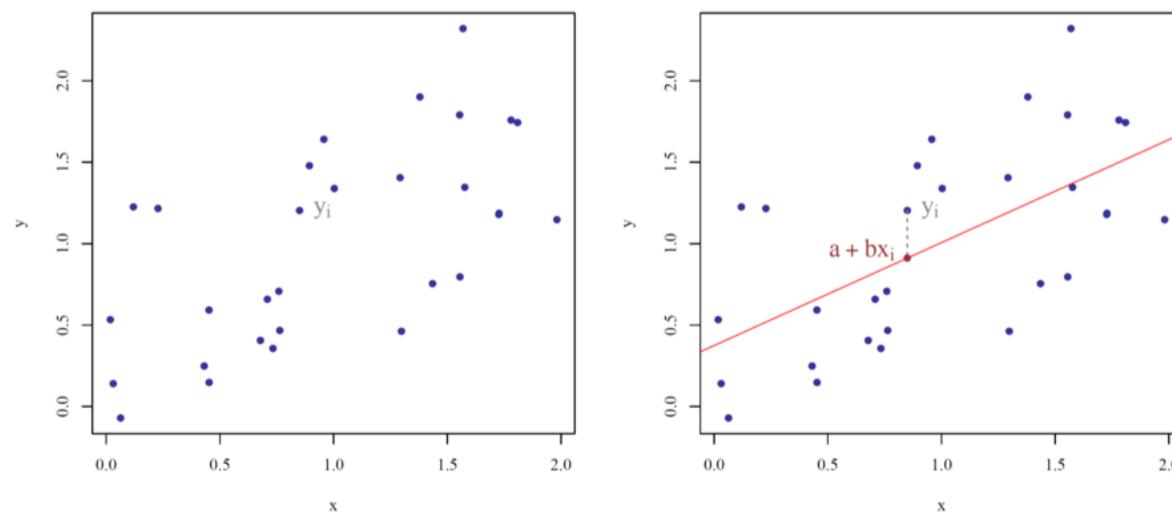
$$\begin{aligned} E(\mathbf{y}) &= \mathbf{X}\boldsymbol{\beta} \\ \text{Var}(\mathbf{y}) &= \sigma^2 \mathbf{I}_n \end{aligned}$$

参数估计

最小二乘估计

定义离差：

$$e_i = y_i - \mathbf{x}'_i \boldsymbol{\beta}$$



对于线性模型，离差平方和定义为

$$Q(\boldsymbol{\beta}) = \sum_{i=1}^n (y_i - \mathbf{x}'_i \boldsymbol{\beta})^2$$

最小二乘估计为

$$\hat{\boldsymbol{\beta}}_{\text{LS}} = \arg \min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta})$$

离差平方和可写为

$$\begin{aligned} Q(\beta) &= \sum_{i=1}^n (y_i - \mathbf{x}'_i \beta)^2 \\ &= \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \\ &= (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) \\ &= \beta' \mathbf{X}' \mathbf{X} \beta - 2\beta' \mathbf{X}' \mathbf{y} + \mathbf{y}' \mathbf{y} \end{aligned}$$

补充知识：对于一个 p 维列向量

$$\mathbf{x} = (x_1, x_2, \dots, x_p)'$$

线性函数求导

$$\frac{\partial(\mathbf{x}' \mathbf{a})}{\partial \mathbf{x}} = \frac{\partial(\mathbf{a}' \mathbf{x})}{\partial \mathbf{x}} = \mathbf{a}$$

二次型求导

$$\frac{\partial(\mathbf{x}' \mathbf{B} \mathbf{x})}{\partial \mathbf{x}} = (\mathbf{B} + \mathbf{B}') \mathbf{x}$$

特别地，如果 \mathbf{B} 是一个对称矩阵，那么

$$\frac{\partial(\mathbf{x}' \mathbf{B} \mathbf{x})}{\partial \mathbf{x}} = 2\mathbf{B}\mathbf{x}$$

具体计算方法：

对 β 求导，可得

$$\frac{\partial Q(\beta)}{\partial \beta} = 2\mathbf{X}' \mathbf{X} \beta - 2\mathbf{X}' \mathbf{y}$$

令导数为 0，而且 $\mathbf{X}' \mathbf{X}$ 是满秩的

$$\hat{\beta}_{\text{LS}} = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}$$

说明：

1. 非奇异矩阵， $|\mathbf{X}' \mathbf{X}| \neq 0$ 。
2. 由线性代数基本结论可知， $\text{rank}(\mathbf{X}) \geq \text{rank}(\mathbf{X}' \mathbf{X})$ 。也就是说 $\text{rank}(\mathbf{X}' \mathbf{X}) = p + 1$ ，那么 $\text{rank}(\mathbf{X}) \geq p + 1$
3. 另一方面，设计矩阵 \mathbf{X} 为 $n \times (p + 1)$ 阶矩阵，于是 $n \geq p + 1$

回归向量定义为

$$\hat{\mathbf{y}} = \mathbf{X} \hat{\beta}$$

其中

$$\hat{y}_i = \mathbf{x}'_i \hat{\beta}, i = 1, 2, \dots, n$$

将向量展开

$$\hat{\mathbf{y}} = \mathbf{X} \hat{\beta} = \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} = \mathbf{H} \mathbf{y}$$

我们称 \mathbf{H} 为帽子矩阵。

定理一：帽子矩阵的性质

1. n 阶对称矩阵
2. 幂等矩阵
3. 迹为 $p + 1$ (对称幂等矩阵秩和迹相等)

证明略。

残差定义为

$$\begin{aligned} \mathbf{e} &= \mathbf{y} - \hat{\mathbf{y}} & (1) \\ &= \mathbf{y} - \mathbf{H} \mathbf{y} & (2) \\ &= (\mathbf{I} - \mathbf{H}) \mathbf{y} & (3) \end{aligned}$$

回归值和残差垂直

$$\hat{\mathbf{y}}' \mathbf{e} = (\mathbf{H} \mathbf{y})' ((\mathbf{I} - \mathbf{H}) \mathbf{y}) = \mathbf{y}' \mathbf{H}' (\mathbf{I} - \mathbf{H}) \mathbf{y} = 0$$

残差的协方差阵

$$\begin{aligned} \text{Var}(\mathbf{e}) &= \text{Cov}(\mathbf{e}, \mathbf{e}) \\ &= \text{Cov}((\mathbf{I} - \mathbf{H}) \mathbf{y}, (\mathbf{I} - \mathbf{H}) \mathbf{y}) \\ &= (\mathbf{I} - \mathbf{H}) \text{Cov}(\mathbf{y}, \mathbf{y}) (\mathbf{I} - \mathbf{H})' \\ &= \sigma^2 (\mathbf{I} - \mathbf{H}) \mathbf{I}_n (\mathbf{I} - \mathbf{H})' \\ &= \sigma^2 (\mathbf{I} - \mathbf{H}) \end{aligned}$$

由此，我们可以构造误差项方差的估计

$$\hat{\sigma}^2 = \frac{1}{n - p - 1} (\mathbf{e}' \mathbf{e}) = \frac{1}{n - p - 1} \sum_{i=1}^n e_i^2$$

注： $\mathbf{I} - \mathbf{H}$ 的迹为 $n - p - 1$

极大似然估计

我们有

$$\mathbf{y} \sim N_n (\mathbf{X} \beta, \sigma^2 \mathbf{I}_n)$$

那么其联合密度函数为

$$f(\mathbf{y}; \boldsymbol{\beta}, \sigma^2) = \frac{1}{(2\pi)^{n/2} |\sigma^2 \mathbf{I}_n|^{1/2}} \exp \left\{ -\frac{1}{2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\sigma^2 \mathbf{I}_n)^{-1} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right\}$$

参数的似然函数为

$$L(\boldsymbol{\beta}, \sigma^2) = (2\pi)^{-n/2} (\sigma^2)^{-n/2} \exp \left\{ -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right\}$$

我们可以转化成对数似然函数，因为

$$\begin{aligned} (\hat{\boldsymbol{\beta}}_{\text{ML}}, \hat{\sigma}_{\text{ML}}^2) &= \arg \max_{(\boldsymbol{\beta}, \sigma^2)} L(\boldsymbol{\beta}, \sigma^2) \\ &= \arg \max_{(\boldsymbol{\beta}, \sigma^2)} \ln(L(\boldsymbol{\beta}, \sigma^2)) \end{aligned}$$

所以

$$\ln L(\boldsymbol{\beta}, \sigma^2) = -\frac{n}{2} \ln(2\pi) - \frac{n}{2} \ln(\sigma^2) - \frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$$

对参数求偏导，即

$$\begin{cases} \frac{\partial \ln L(\boldsymbol{\beta}, \sigma^2)}{\partial \boldsymbol{\beta}} = -\frac{1}{\sigma^2} (\mathbf{X}' \mathbf{X} \boldsymbol{\beta} - \mathbf{X}' \mathbf{y}) = 0 \\ \frac{\partial \ln L(\boldsymbol{\beta}, \sigma^2)}{\partial \sigma^2} = -\frac{n}{2\sigma^2} + \frac{1}{2\sigma^4} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = 0 \end{cases}$$

极大似然估计为

$$\begin{cases} \hat{\boldsymbol{\beta}}_{\text{ML}} = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} \\ \hat{\sigma}_{\text{ML}}^2 = \frac{1}{n} (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{\text{ML}})' (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{\text{ML}}) = \frac{1}{n} \mathbf{e}' \mathbf{e} \end{cases}$$

说明：

1. $\hat{\boldsymbol{\beta}}_{\text{ML}} = \hat{\boldsymbol{\beta}}_{\text{LS}}$
2. $\hat{\sigma}_{\text{ML}}^2$ 不是无偏估计，但是是相合估计

参数估计

多元概率论常用结论

$$\begin{aligned} \mathbb{E}(\mathbf{Ax} + \mathbf{c}) &= \mathbf{A} \mathbb{E}(\mathbf{x}) + \mathbf{c} \\ \text{Var}(\mathbf{Ax} + \mathbf{c}) &= \mathbf{A} \text{Var}(\mathbf{x}) \mathbf{A}' \\ \text{Cov}(\mathbf{Ax}, \mathbf{By}) &= \mathbf{A} \text{Cov}(\mathbf{x}, \mathbf{y}) \mathbf{B}' \end{aligned}$$

定理二：最小二乘估计的性质

1. $E(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$
2. $\text{Var}(\hat{\boldsymbol{\beta}}) = \sigma^2 (\mathbf{X}' \mathbf{X})^{-1}$

证明

$$\begin{aligned}
E(\hat{\beta}) &= E\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}\right) = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E(\mathbf{y}) \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'E(\mathbf{X}\beta + \varepsilon) \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'(\mathbf{X}\beta + E(\varepsilon)) \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{X}\beta = \beta
\end{aligned}$$

$$\begin{aligned}
\text{Var}(\hat{\beta}) &= \text{Var}\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}\right) \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\text{Var}(\mathbf{y})\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right)' \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\text{Var}(\mathbf{X}\beta + \varepsilon)\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right)' \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'(\sigma^2\mathbf{I}_n)\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1} \\
&= \sigma^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1} \\
&= \sigma^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}
\end{aligned}$$

定理三：最小二乘估计与残差线性不相关

即证

$$\text{Cov}(\hat{\beta}, \mathbf{e}) = \mathbf{0}$$

我们同时也可以得到一个推论：最小二乘估计与残差平方和独立。

证明

$$\begin{aligned}
\text{Cov}(\hat{\beta}, \mathbf{e}) &= \text{Cov}\left(\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}, (\mathbf{I} - \mathbf{H})\mathbf{y}\right) \\
&= \sigma^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\left(\mathbf{I}_n - \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\right) \\
&= \sigma^2 \cdot 0 \\
&= 0
\end{aligned}$$

在此总结一下残差和哪些量线性不相关：

1. 回归值和残差
2. 最小二乘估计和残差

中心化和标准化

矩阵的知识

$$\left(\begin{array}{cc} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{array}\right)^{-1} = \left(\begin{array}{cc} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{array}\right)$$

$$\mathbf{E}_{11} = \mathbf{A}^{-1} + \mathbf{A}^{-1} \mathbf{B} (\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B})^{-1} \mathbf{C} \mathbf{A}^{-1} \quad (4)$$

$$\mathbf{E}_{12} = -\mathbf{A}^{-1} \mathbf{B} (\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B})^{-1} \quad (5)$$

$$\mathbf{E}_{21} = -(\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B})^{-1} \mathbf{C} \mathbf{A}^{-1} \quad (6)$$

$$\mathbf{E}_{22} = (\mathbf{D} - \mathbf{C} \mathbf{A}^{-1} \mathbf{B})^{-1} \quad (7)$$

原本的模型

$$y = \mathbf{x}' \boldsymbol{\beta} + \varepsilon$$

我们这里写做

$$\hat{y} = \hat{\beta}_{\text{intercept}} + \mathbf{x}' \hat{\boldsymbol{\beta}}_{\text{slope}}$$

原始数据集

$$\begin{cases} \mathbf{y} = (y_1, \dots, y_n)' \\ \mathbf{X} = (\mathbf{1}_n, \mathbf{X}_o), \quad \mathbf{X}_o = (\mathbf{x}_1, \dots, \mathbf{x}_p) \end{cases}$$

最小二乘估计为

$$\begin{aligned} \hat{\boldsymbol{\beta}} &= (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} \\ &= \begin{pmatrix} n & \mathbf{1}'_n \mathbf{X}_o \\ \mathbf{X}'_o \mathbf{1}_n & \mathbf{X}'_o \mathbf{X}_o \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{1}'_n \\ \mathbf{X}'_o \end{pmatrix} \mathbf{y} \\ &= \begin{pmatrix} n^{-1} + n^{-2} \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n & -n^{-1} \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \\ -n^{-1} \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n & \mathbf{A}_o \end{pmatrix} \begin{pmatrix} \mathbf{1}'_n \\ \mathbf{X}'_o \end{pmatrix} \mathbf{y} \\ &= \begin{pmatrix} n^{-1} \mathbf{1}'_n + n^{-2} \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n - n^{-1} \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \\ -n^{-1} \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n + \mathbf{A}_o \mathbf{X}'_o \end{pmatrix} \mathbf{y} \end{aligned}$$

其中

$$\mathbf{A}_o = (\mathbf{X}'_o \mathbf{X}_o - n^{-1} \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n \mathbf{X}_o)^{-1}$$

中心化

$$\begin{aligned} x_{ij}^* &= x_{ij} - \bar{x}_j, \quad \bar{x}_j = n^{-1} \sum_{i=1}^n x_{ij} \\ y_i^* &= y_i - \bar{y}, \quad \bar{y} = n^{-1} \sum_{i=1}^n y_i \end{aligned}$$

令

$$\begin{cases} \mathbf{y}^* = (y_1^*, \dots, y_n^*)' \\ \mathbf{X}_c = (\mathbf{x}_1^*, \dots, \mathbf{x}_p^*) \\ \mathbf{X}^* = (\mathbf{1}_n, \mathbf{X}_c) \end{cases}$$

其中 $\mathbf{x}_j^* = (x_{1j}^*, \dots, x_{nj}^*)'$

那么其最小二乘估计为

$$\begin{aligned}
\hat{\beta}_c &= ((\mathbf{X}^*)' \mathbf{X}^*)^{-1} (\mathbf{X}^*)' \mathbf{y}^* \\
&= \begin{pmatrix} n & \mathbf{1}'_n \mathbf{X}_c \\ \mathbf{X}'_c \mathbf{1}_n & \mathbf{X}'_c \mathbf{X}_c \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{1}'_n \\ \mathbf{X}'_c \end{pmatrix} \mathbf{y}^* \\
&= \begin{pmatrix} n^{-1} + n^{-2} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n & -n^{-1} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \\ -n^{-1} \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n & \mathbf{A}_c \end{pmatrix} \begin{pmatrix} \mathbf{1}'_n \\ \mathbf{X}'_c \end{pmatrix} \mathbf{y}^* \\
&= \begin{pmatrix} n^{-1} \mathbf{1}'_n + n^{-2} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n - n^{-1} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \mathbf{X}'_c \\ -n^{-1} \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n + \mathbf{A}_c \mathbf{X}'_c \end{pmatrix} \mathbf{y}^*
\end{aligned}$$

其中

$$\mathbf{A}_c = (\mathbf{X}'_c \mathbf{X}_c - n^{-1} \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n \mathbf{X}_c)^{-1}$$

中心化的因变量与未中心化的因变量之间的关系

$$\mathbf{y}^* = \mathbf{y} - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{y} = (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}$$

其中

$$\mathbf{H}_{1n} = \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n$$

是对称幂等矩阵

中心化的自变量与未中心化的自变量之间的关系

$$\begin{aligned}
\mathbf{X}_c &= \mathbf{X}_o - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{X}_o = \left(\mathbf{I}_n - \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \right) \mathbf{X}_o \\
&= (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{X}_o
\end{aligned}$$

而且

$$\begin{aligned}
\mathbf{1}'_n (\mathbf{I}_n - \mathbf{H}_{1n}) &= \mathbf{1}'_n - \mathbf{1}'_n \mathbf{H}_{1n} \\
&= \mathbf{1}'_n - \mathbf{1}'_n \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n = 0
\end{aligned}$$

回归常数与回归系数

$$\hat{\beta}_c = \left(\hat{\beta}_{c, \text{intercept}}, \hat{\beta}_{c, \text{slope}} \right)'$$

回归常数

$$\begin{aligned}
\hat{\beta}_{c, \text{intercept}} &= (n^{-1} \mathbf{1}'_n + n^{-2} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n - n^{-1} \mathbf{1}'_n \mathbf{X}_c \mathbf{A}_c \mathbf{X}'_c) \mathbf{y}^* \\
&= n^{-1} \mathbf{1}'_n \mathbf{y}^* \\
&= n^{-1} \mathbf{1}'_n (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y} = 0
\end{aligned}$$

由于

$$\begin{aligned}
\mathbf{A}_c &= (\mathbf{X}'_c \mathbf{X}_c - n^{-1} \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n \mathbf{X}_c)^{-1} \\
&= (\mathbf{X}'_o (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{X}_o \\
&\quad - n^{-1} \mathbf{X}'_c (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{1}_n \mathbf{1}'_n (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{X}_o)^{-1} \\
&= (\mathbf{X}'_o (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{X}_o)^{-1} = \mathbf{A}_o
\end{aligned}$$

另一方面, 回归系数

$$\begin{aligned}\hat{\beta}_{c, \text{slope}} &= \left(-n^{-1} \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n + \mathbf{A}_c \mathbf{X}'_c \right) \mathbf{y}^* \\ &= \mathbf{A}_c \mathbf{X}'_c (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y} \\ &= \mathbf{A}_o \mathbf{X}'_o (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}\end{aligned}$$

而

$$\begin{aligned}\hat{\beta}_{\text{slope}} &= \left(-n^{-1} \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n + \mathbf{A}_o \mathbf{X}'_o \right) \mathbf{y} \\ &= \mathbf{A}_o \mathbf{X}'_o (\mathbf{I}_n - n^{-1} \mathbf{1}_n \mathbf{1}'_n) \mathbf{y} \\ &= \mathbf{A}_o \mathbf{X}'_o (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}\end{aligned}$$

于是

$$\hat{\beta}_{c, \text{slope}} = \hat{\beta}_{\text{slope}}$$

那么采用中心化的数据得到的经验回归方程为

$$\hat{y}^* = (\mathbf{x}^*)' \hat{\beta}_{\text{slope}}$$

标准化

$$\begin{aligned}x_{ij}^{**} &= \frac{x_{ij}^*}{\sqrt{L_{jj}}} = \frac{x_{ij} - \bar{x}_j}{\sqrt{L_{jj}}}, \quad i = 1, 2, \dots, n; j = 1, 2, \dots, p. \\ y_i^{**} &= \frac{y_i^*}{\sqrt{L_{yy}}}, \quad i = 1, 2, \dots, n.\end{aligned}$$

其中, L_{jj} 是自变量 x_j 的离差平方和

$$L_{jj} = \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2$$

L_{yy} 是因变量 y 的离差平方和

$$L_{yy} = \sum_{i=1}^n (y_i - \bar{y})^2$$

令

$$\begin{aligned}\mathbf{y}^{**} &= \left(\frac{y_1 - \bar{y}}{\sqrt{L_{yy}}}, \dots, \frac{y_n - \bar{y}}{\sqrt{L_{yy}}} \right)' = \frac{1}{\sqrt{L_{yy}}} \mathbf{y}^* \\ \mathbf{X}_s &= \left(\frac{1}{\sqrt{L_{11}}} \mathbf{x}_1^*, \dots, \frac{1}{\sqrt{L_{pp}}} \mathbf{x}_p^* \right)' \\ &= \mathbf{X}_c \mathbf{L}.\end{aligned}$$

其中

$$\mathbf{L} = \text{diag} \left\{ \frac{1}{\sqrt{L_{11}}}, \dots, \frac{1}{\sqrt{L_{pp}}} \right\}$$

最小二乘估计

$$\hat{\boldsymbol{\beta}}_s = \left(\hat{\boldsymbol{\beta}}_{s, \text{intercept}}, \hat{\boldsymbol{\beta}}_{s, \text{slope}} \right)' = \left(0, \hat{\boldsymbol{\beta}}_{s, \text{slope}} \right)'$$

回归系数

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{s, \text{slope}} &= (\mathbf{X}'_s \mathbf{X}_s)^{-1} \mathbf{X}'_s \mathbf{y}^{**} \\ &= (\mathbf{L} \mathbf{X}'_c \mathbf{X}_c \mathbf{L})^{-1} \mathbf{L} \mathbf{X}'_c \frac{1}{\sqrt{L_{yy}}} \mathbf{y}^* \\ &= \mathbf{L}^{-1} (\mathbf{X}'_c \mathbf{X}_c)^{-1} \mathbf{L}^{-1} \mathbf{L} \mathbf{X}'_c \frac{1}{\sqrt{L_{yy}}} \mathbf{y}^* \\ &= \frac{1}{\sqrt{L_{yy}}} \mathbf{L}^{-1} (\mathbf{X}'_c \mathbf{X}_c)^{-1} \mathbf{X}'_c \mathbf{y}^* \end{aligned}$$

注意到

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{c, \text{slope}} &= (-n^{-1} \mathbf{A}_c \mathbf{X}'_c \mathbf{1}_n \mathbf{1}'_n + \mathbf{A}_c \mathbf{X}'_c) \mathbf{y}^* \\ &= \mathbf{A}_c \mathbf{X}'_c (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}^* \\ &= (\mathbf{X}'_c \mathbf{X}_c)^{-1} \mathbf{X}'_c (\mathbf{I}_n - \mathbf{H}_{1n}) (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y}^* \\ &= (\mathbf{X}'_c \mathbf{X}_c)^{-1} \mathbf{X}'_c \mathbf{y}^* \end{aligned}$$

因此

$$\hat{\boldsymbol{\beta}}_{s, \text{slope}} = \frac{1}{\sqrt{L_{yy}}} \mathbf{L}^{-1} \hat{\boldsymbol{\beta}}_{c, \text{slope}}$$

其中每一个分量为

$$\hat{\beta}_{sj} = \frac{\sqrt{L_{jj}}}{\sqrt{L_{yy}}} \hat{\beta}_{cj} = \frac{\sqrt{L_{jj}}}{\sqrt{L_{yy}}} \hat{\beta}_j, \quad j = 1, 2, \dots, p.$$

显著性检验

显著性检验：因变量与自变量是否存在显著的线性关系。

有两种检验方法：

1. F 检验：检验回归方程的显著性。
2. t 检验：检验回归系数的显著性。

除了显著性检验，我们这里简单介绍一下常用的指标用于衡量线性回归的拟合优度。

F 检验

原假设

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0$$

备择假设

$$H_1 : \text{存在 } \beta_j \text{ 不为零}, j = 1, 2, \dots, p.$$

如果原假设为真，则表明用线性回归模型刻画数据的关系是不合适的。

离差平方和

$$\sum_{i=1}^n (y_i - \bar{y})^2 = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 + \sum_{i=1}^n (y_i - \hat{y}_i)^2$$

记为

$$SS_T = SS_R + SS_E$$

拟合值 $\hat{y}_i = \mathbf{x}'_i \hat{\beta}$

偏差 $e_i = y_i - \hat{y}_i$

检验统计量

$$F_0 = \frac{SS_R/p}{SS_E/(n-p-1)}$$

定理四：偏差平方和的分布

在正态假设下，有

1. $SS_E/\sigma^2 \sim \chi^2(n-p-1)$ ，其中 $SS_E = \mathbf{e}'\mathbf{e}$
2. SS_E 和 SS_R 独立
3. 在 H_0 下， $SS_R/\sigma^2 \sim \chi^2(p)$

多元统计知识补充：

1. 假设一个 n 维随机变量

$$\mathbf{x} \sim N_n(\boldsymbol{\mu}, \sigma^2 \mathbf{I}_n)$$

如果 \mathbf{C} 是一个对称矩阵, 且 $\text{rank}(\mathbf{C}) = r$, 那么二次型

$$\mathbf{x}' \mathbf{C} \mathbf{x} / \sigma^2 \sim \chi^2(r, \delta)$$

其中

$$\delta = \frac{1}{\sigma^2} \boldsymbol{\mu}' \mathbf{C} \boldsymbol{\mu}$$

当且仅当

$$\mathbf{C}^2 = \mathbf{C} \quad \text{且} \quad \text{rank}(\mathbf{C}) = r (r \leq n)$$

2. \mathbf{A} 为 n 阶对称矩阵, \mathbf{B} 为 $m \times n$ 矩阵, 那么

$$\mathbf{B} \mathbf{A} = \mathbf{0}_{m \times n}$$

当且仅当 $\mathbf{B} \mathbf{x}$ 和 $\mathbf{x}' \mathbf{A} \mathbf{x}$ 相互独立

3. \mathbf{A}, \mathbf{B} 均为 n 阶对称矩阵, 则

$$\mathbf{A} \mathbf{B} = \mathbf{0}_{n \times n}$$

当且仅当 $\mathbf{x}' \mathbf{A} \mathbf{x}$ 和 $\mathbf{x}' \mathbf{B} \mathbf{x}$ 相互独立

第一部分证明

由于残差

$$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = (\mathbf{I} - \mathbf{H}) \mathbf{y}$$

残差平方和

$$SS_E = \mathbf{e}' \mathbf{e} = \mathbf{y}' (\mathbf{I} - \mathbf{H}) \mathbf{y}$$

由于

$$\mathbf{y} \sim N_n(\mathbf{X} \boldsymbol{\beta}, \sigma^2 \mathbf{I}_n)$$

$$\text{rank}(\mathbf{I} - \mathbf{H}) = n - (p + 1)$$

那么

$$\frac{1}{\sigma^2} \mathbf{E}(\mathbf{y})' (\mathbf{I} - \mathbf{H}) \mathbf{E}(\mathbf{y}) = \frac{1}{\sigma^2} \boldsymbol{\beta}' \mathbf{X}' (\mathbf{I} - \mathbf{H}) \mathbf{X} \boldsymbol{\beta} = 0$$

因此

$$SS_E/\sigma^2 \sim \chi^2(n-p-1)$$

第二部分证明

由于

$$\begin{aligned} SS_R &= \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 = \sum_{i=1}^n (\mathbf{x}'_i \hat{\beta} - \bar{\mathbf{x}}' \hat{\beta})^2 \\ &= \hat{\beta}' \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})' \hat{\beta} \\ &= \mathbf{y}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{A} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} \end{aligned}$$

其中

$$\mathbf{A} = \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})'$$

残差平方和

$$SS_E = \mathbf{e}' \mathbf{e} = \mathbf{y}' (\mathbf{I} - \mathbf{H}) \mathbf{y}$$

容易得到

$$\mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{A} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' (\mathbf{I} - \mathbf{H}) = \mathbf{0}$$

那么有 SS_E 和 SS_R 独立 (多元统计分析补充三的结论)。

第三部分证明

因为

$$\bar{\mathbf{x}}' = (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{X}$$

我们有

$$\begin{aligned} \mathbf{A} &= \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})' \\ &= \sum_{i=1}^n \mathbf{x}_i \mathbf{x}'_i - n \bar{\mathbf{x}} \bar{\mathbf{x}}' \\ &= \mathbf{X}' \mathbf{X} - (\mathbf{1}'_n \mathbf{1}_n) \mathbf{X}' \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{X} \\ &= \mathbf{X}' \mathbf{X} - \mathbf{X}' \mathbf{1}_n (\mathbf{1}'_n \mathbf{1}_n)^{-1} \mathbf{1}'_n \mathbf{X} \\ &= \mathbf{X}' (\mathbf{I}_n - \mathbf{H}_{\mathbf{1}_n}) \mathbf{X} \end{aligned}$$

所以

$$\begin{aligned} SS_R &= \mathbf{y}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' (\mathbf{I}_n - \mathbf{H}_{\mathbf{1}_n}) \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} \\ &= \mathbf{y}' \mathbf{H} (\mathbf{I}_n - \mathbf{H}_{\mathbf{1}_n}) \mathbf{H} \mathbf{y} \end{aligned}$$

由于

$$\mathbf{X} = (\mathbf{1}_n, \mathbf{X}_o)$$

因此有

$$\begin{aligned}\mathbf{H}\mathbf{H}_{1n} &= \begin{pmatrix} \mathbf{1}_n & \mathbf{X}_o \end{pmatrix} \begin{pmatrix} n^{-1} + n^{-2}\mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n & -n^{-1}\mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \\ -n^{-1}\mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n & \mathbf{A}_o \end{pmatrix} \begin{pmatrix} \mathbf{1}'_n \\ \mathbf{X}'_o \end{pmatrix} (n^{-1}\mathbf{1}_n \mathbf{1}'_n) \\ &= (n^{-1}\mathbf{1}_n \mathbf{1}'_n + n^{-2}\mathbf{1}_n \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n - n^{-1}\mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o \mathbf{1}_n \mathbf{1}'_n \\ &\quad - n^{-1}\mathbf{1}_n \mathbf{1}'_n \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o + \mathbf{X}_o \mathbf{A}_o \mathbf{X}'_o) (n^{-1}\mathbf{1}_n \mathbf{1}'_n) \\ &= (n^{-1}\mathbf{1}_n \mathbf{1}'_n)^2 \\ &= \mathbf{H}_{1n}\end{aligned}$$

于是

$$\begin{aligned}SS_R &= \mathbf{y}' \mathbf{H} (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{H} \mathbf{y} \\ &= \mathbf{y}' (\mathbf{H} - \mathbf{H}_{1n}) \mathbf{y}\end{aligned}$$

令

$$\mathbf{B} = (\mathbf{H} - \mathbf{H}_{1n})$$

容易验证其是对称幂等矩阵，且

$$\text{rank}(\mathbf{B}) = \text{tr}(\mathbf{B}) = \text{tr}(\mathbf{H}) - \text{tr}(\mathbf{H}_{1n}) = (p+1) - 1 = p$$

在原假设成立时

$$\beta_1 = \beta_2 = \dots = \beta_p = 0$$

于是

$$\mathbf{y} \sim N(\beta_0 \mathbf{1}_n, \sigma^2 \mathbf{I})$$

根据多元统计分析补充一的结论，有

$$\frac{SS_R}{\sigma^2} = \frac{\mathbf{y}' \mathbf{B} \mathbf{y}}{\sigma^2} \stackrel{H_0}{\sim} \chi^2(p, \delta)$$

其中

$$\begin{aligned}\delta &= \frac{1}{\sigma^2} (\beta_0 \mathbf{1}_n)' \mathbf{B} (\beta_0 \mathbf{1}_n) = \frac{\beta_0^2}{\sigma^2} \mathbf{1}'_n (\mathbf{H} - \mathbf{H}_{1n}) \mathbf{1}_n \\ &= \frac{\beta_0^2}{\sigma^2} \text{tr}(\mathbf{1}'_n (\mathbf{H} - \mathbf{H}_{1n}) \mathbf{1}_n) = \frac{\beta_0^2}{\sigma^2} \text{tr}((\mathbf{H} - \mathbf{H}_{1n}) \mathbf{1}_n \mathbf{1}'_n) \\ &= \frac{\beta_0^2}{\sigma^2} \text{tr}(n(\mathbf{H} - \mathbf{H}_{1n}) \mathbf{H}_{1n}) = 0\end{aligned}$$

那么，检验统计量

$$F_0 = \frac{SS_R/p}{SS_E/(n-p-1)} \stackrel{H_0}{\sim} F(p, n-p-1)$$

方差分析表

来源	平方和	自由度	均方	F值	p值
回归	SS_R	p	$MS_R = \frac{SS_R}{p}$	$F_0 = \frac{MS_R}{MS_E}$	$p_0 = P(F \geq F_0)$
误差	SS_E	$n - p - 1$	$MS_E = \frac{SS_E}{(n-p-1)}$		
总和	SS_T	$n - 1$			

给定显著性水平 α

1. 当 $F_0 > F_{1-\alpha}(p, n - p - 1)$
2. 当 $p_0 = P(F \geq F_0) < \alpha$

拒绝原假设 H_0

说明

$$SS_T = SS_R + SS_E$$

即

$$\mathbf{y}' (\mathbf{I}_n - \mathbf{H}_{1n}) \mathbf{y} = \mathbf{y}' (\mathbf{H} - \mathbf{H}_{1n}) \mathbf{y} + \mathbf{y}' (\mathbf{I}_n - \mathbf{H}) \mathbf{y}$$

且

$$SS_R \perp SS_E$$

t 检验

对于 n 维随机变量

$$\mathbf{x} \sim N_n(\boldsymbol{\mu}, \Sigma)$$

我们有

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b} \sim N_m(\mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{A}\Sigma\mathbf{A}')$$

定理五：参数估计的性质

在正态假设下，即 $\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}_n)$ ，有

$$\hat{\boldsymbol{\beta}} \sim N(\boldsymbol{\beta}, \sigma^2 (\mathbf{X}'\mathbf{X})^{-1})$$

证明：

由定理二可知 $\hat{\boldsymbol{\beta}}$ 的期望和方差

期望

$$E(\hat{\beta}) = \beta$$

方差

$$\text{Var}(\hat{\beta}) = \sigma^2 (\mathbf{X}' \mathbf{X})^{-1}$$

由定理五可知

$$\hat{\beta} = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}$$

那么 $\hat{\beta}$ 就符合多元正态分布，因此有

$$\hat{\beta} \sim N(\beta, \sigma^2 (\mathbf{X}' \mathbf{X})^{-1})$$

多元统计知识补充：

假设随机向量 $\mathbf{x} = (x_1, x_2, \dots, x_p)'$ 是一个 p 维正态分布

$$N(\mu, \Sigma)$$

其中

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{pmatrix} = \begin{pmatrix} E(x_1) \\ E(x_2) \\ \vdots \\ E(x_p) \end{pmatrix} \quad \Sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \cdots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{pmatrix}$$

那么，每一个分量 x_j 也服从正态分布，即

$$x_j \sim N(\mu_j, \sigma_{jj})$$

根据定理五可知

$$\hat{\beta} \sim N(\beta, \sigma^2 (\mathbf{X}' \mathbf{X})^{-1})$$

记 c_{ij} 表示 $(\mathbf{X}' \mathbf{X})^{-1}$ 中第 $(i+1, j+1)$ 元素， $i, j = 0, 1, \dots, p$

根据多元统计的知识，我们有

$$\hat{\beta}_j \sim N(\beta_j, c_{jj} \sigma^2), j = 0, 1, \dots, p$$

检验统计量

$$t_j = \frac{\hat{\beta}_j}{\sqrt{c_{jj}} \hat{\sigma}}$$

其中

$$\hat{\sigma}^2 = \frac{1}{n-p-1} SS_E$$

当原假设 $H_{0j} : \beta_j = 0$ 为真时

$$t_j \sim t(n - p - 1)$$

给定显著性水平 α ，当

$$|t_j| \geq t_{1-\alpha/2}(n - p - 1)$$

拒绝原假设，并认为 β_j 显著不为 0

两者检验方式对比

一元线性回归中，回归系数显著性的 t 和 F 检验等价。

多元线性回归中， t 检验代表的是某一个自变量对因变量的线性回归效果，而 F 检验则代表整体自变量对因变量的线性回归效果。

复相关系数

拟合优度可以用来度量回归方程对样本观测值的拟合程度。（ x 的波动能够解释多少 y 的波动）

在多元线性回归中，定义样本决定系数为

$$R^2 = \frac{SS_R}{SS_T} = 1 - \frac{SS_E}{SS_T}$$

注：

1. 取值范围为 $[0, 1]$
2. 越大，说明拟合效果越好

样本复相关系数为

$$R = \sqrt{R^2} = \sqrt{\frac{SS_R}{SS_T}}$$

与样本相关系数的区别

1. 复相关系数的符号恒为正，相关系数的符号有正有负；
2. 复相关系数衡量作为一个整体自变量与因变量的线性关系
3. 相关系数衡量单个随机变量与因变量的线性关系

置信区间与预测

概述

给定

$$\mathbf{x}_0 = (1, x_{01}, x_{02}, \dots, x_{0p})'$$

我们关心

$$y_0 = \mathbf{x}'_0 \boldsymbol{\beta} + \varepsilon_0$$

基本假定

$$E(\varepsilon_0) = 0 \quad \text{Var}(\varepsilon_0) = \sigma^2$$

也就是说

$$E(y_0) = \mathbf{x}'_0 \boldsymbol{\beta} \quad \text{Var}(y_0) = \sigma^2$$

基于数据，我们可以得到对参数的最小二乘估计

$$\hat{\boldsymbol{\beta}} = (\hat{\beta}_0, \hat{\beta}_1, \dots, \hat{\beta}_p)'$$

点预测

预测值

$$\hat{y}_0 = \mathbf{x}'_0 \hat{\boldsymbol{\beta}} = \hat{\beta}_0 + x_{01} \hat{\beta}_1 + \dots + x_{0p} \hat{\beta}_p$$

怎么评价，考虑均方误差

$$E(\hat{y}_0 - y_0)^2$$

注意到 \hat{y}_0, y_0 都为随机变量。由于

$$\begin{aligned} & (\hat{y}_0 - y_0)^2 \\ &= (\hat{y}_0 - E(\hat{y}_0) + E(\hat{y}_0) - E(y_0) + E(y_0) - y_0)^2 \\ &= (\hat{y}_0 - E(\hat{y}_0))^2 + 2(\hat{y}_0 - E(\hat{y}_0))(E(\hat{y}_0) - E(y_0)) \\ &\quad + (E(\hat{y}_0) - E(y_0))^2 + 2(E(\hat{y}_0) - E(y_0))(E(y_0) - y_0) \\ &\quad + (E(y_0) - y_0)^2 + 2(\hat{y}_0 - E(\hat{y}_0))(E(y_0) - y_0) \\ &= (\hat{y}_0 - E(\hat{y}_0))^2 + 2(\hat{y}_0 - E(\hat{y}_0))(E(\hat{y}_0) - E(y_0)) \\ &\quad + (E(\hat{y}_0) - E(y_0))^2 + 2(E(\hat{y}_0) - E(y_0))(-\varepsilon_0) \\ &\quad + \varepsilon_0^2 + 2(\hat{y}_0 - E(\hat{y}_0))(-\varepsilon_0) \end{aligned}$$

那么

$$E(\hat{y}_0 - y_0)^2 = \text{Var}(\hat{y}_0) + \text{Bias}^2(\hat{y}_0) + \text{Var}(\varepsilon_0)$$

考虑 $\text{Bias}^2(\hat{y}_0) = 0$ ，即 $E(\hat{y}_0) = E(y_0)$

定理六：预测值的无偏性

\hat{y}_0 是 y_0 的无偏预测，即

$$E(\hat{y}_0) = E(y_0)$$

证明：因为最小二乘估计的无偏性，我们有

$$E(\hat{y}_0) = E(\mathbf{x}'_0 \hat{\boldsymbol{\beta}}) = \mathbf{x}'_0 E(\hat{\boldsymbol{\beta}}) = \mathbf{x}'_0 \boldsymbol{\beta} = E(y_0)$$

我们可以将 \hat{y}_0 写为

$$\hat{y}_0 = \mathbf{x}'_0 \hat{\beta} = \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}$$

我们发现 \hat{y}_0 是 \mathbf{y} 的线性函数，因此 \hat{y}_0 是 y_0 的线性预测。

那么在 y_0 的所有线性预测中， \hat{y}_0 是不是最好的呢？

定理七：最小方差无偏估计

假定 $\hat{\beta}$ 是 β 的 OLS，对于任意常数向量 \mathbf{c} ， $\mathbf{c}' \hat{\beta}$ 是 $\mathbf{c}' \beta$ 的最小方差无偏估计 (BLUE)。

证明：

假设 $\mathbf{d}' \mathbf{y}$ 是 $\mathbf{c}' \beta$ 的任意一个线性无偏估计，即对于一切 β ，有

$$\mathbf{c}' \beta = E(\mathbf{d}' \mathbf{y}) = E(\mathbf{d}' (\mathbf{X} \beta + \varepsilon)) = \mathbf{d}' \mathbf{X} \beta$$

于是

$$\mathbf{d}' \mathbf{X} = \mathbf{c}'$$

由此

$$\text{Var}(\mathbf{d}' \mathbf{y}) = \mathbf{d}' \text{Var}(\mathbf{y}) \mathbf{d} = \sigma^2 \mathbf{d}' \mathbf{d}$$

$$\begin{aligned} \text{Var}(\mathbf{c}' \hat{\beta}) &= \text{Var}(\mathbf{c}' (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}) \\ &= \sigma^2 \mathbf{c}' (\mathbf{X}' \mathbf{X})^{-1} \mathbf{c} \\ &= \sigma^2 \mathbf{d}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{d}. \end{aligned}$$

从而

$$\begin{aligned} \text{Var}(\mathbf{d}' \mathbf{y}) - \text{Var}(\mathbf{c}' \hat{\beta}) &= \sigma^2 \mathbf{d}' \mathbf{d} - \sigma^2 \mathbf{d}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{d} \\ &= \sigma^2 \mathbf{d}' (\mathbf{I}_n - \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}') \mathbf{d} \\ &= \sigma^2 \mathbf{d}' (\mathbf{I}_n - \mathbf{H}) \mathbf{d} \\ &\geq 0 \end{aligned}$$

最后一个不等式成立的原因是因为 $\mathbf{I}_n - \mathbf{H}$ 是半正定的。

具体证明如下：

$$\mathbf{d}' (\mathbf{I}_n - \mathbf{H}) \mathbf{d} = \mathbf{d}' (\mathbf{I}_n - \mathbf{H})' (\mathbf{I}_n - \mathbf{H}) \mathbf{d} \quad (8)$$

$$= ((\mathbf{I}_n - \mathbf{H}) \mathbf{d})' (\mathbf{I}_n - \mathbf{H}) \mathbf{d} \quad (9)$$

$$= \|(\mathbf{I}_n - \mathbf{H}) \mathbf{d}\|_2^2 \geq 0 \quad (10)$$

因此得证。

推论： y_0 的一切线性无偏估计中， \hat{y}_0 方差最小。

预测值的分布

分布

$$\hat{y}_0 \sim N \left(\mathbf{x}'_0 \boldsymbol{\beta}, \sigma^2 \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0 \right)$$

$$\hat{y}_0 - y_0 \sim N \left(0, \sigma^2 \left(1 + \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0 \right) \right)$$

根据定理四可知

$$\frac{SS_E}{\sigma^2} \sim \chi^2(n - p - 1)$$

于是

$$\frac{\hat{y}_0 - y_0}{\hat{\sigma} \sqrt{1 + \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0}} \sim t_{n-p-1}$$

期望水平为 $1 - \alpha$ 的置信区间

$$\hat{y}_0 \pm t_{1-\frac{\alpha}{2}}(n - p - 1) \hat{\sigma} \sqrt{\mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0}$$

预测值水平为 $1 - \alpha$ 的置信区间

$$\hat{y}_0 \pm t_{1-\frac{\alpha}{2}}(n - p - 1) \hat{\sigma} \sqrt{1 + \mathbf{x}'_0 (\mathbf{X}' \mathbf{X})^{-1} \mathbf{x}_0}$$



变量选择

预测时，我们常常希望预测值的 RMSE 比较小

1. 丢失重要变量，导致拟合不足，偏差大。
2. 容纳过多不重要变量，导致模型过拟合，泛化能力差。

我们可以使用全子集归纳法选择合适的自变量，需要考虑 $2^p - 1$ 个回归模型

基本定义：

我们有 p 个自变量，定义为全模型，即

$$\mathbf{y} = \mathbf{X}_p \boldsymbol{\beta}_p + \boldsymbol{\epsilon}$$

从上述自变量中挑选出 p_1 个自变量，构造回归模型，定义为选模型

$$\mathbf{y} = \mathbf{X}_{p_1} \boldsymbol{\beta}_{p_1} + \boldsymbol{\epsilon}$$

为了简化起见，不妨认为 x_1, x_2, \dots, x_{p_1} 就是 x_1, x_2, \dots, x_p 中的前 p_1 个。

自变量选择：全模型与选模型

1. 如果全模型正确，错误地使用了选模型，认为是欠拟合
2. 如果选模型正确，错误地使用了全模型，认为是过拟合

影响

全模型

$$\begin{aligned}\hat{\boldsymbol{\beta}}_p &= (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p \mathbf{y} \\ \hat{\sigma}_p^2 &= \frac{1}{n - p - 1} S S_E^p \\ &= \frac{1}{n - p - 1} (\mathbf{y} - \hat{\mathbf{y}}_p)' (\mathbf{y} - \hat{\mathbf{y}}_p)\end{aligned}$$

预测值

$$\hat{\mathbf{y}}_0 = \mathbf{x}'_{p,0} \hat{\boldsymbol{\beta}}_p$$

选模型

$$\begin{aligned}
\hat{\beta}_{p1} &= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{y} \\
\hat{\sigma}_{p1}^2 &= \frac{1}{n - p_1 - 1} S S_E^{p1} \\
&= \frac{1}{n - p_1 - 1} (\mathbf{y} - \hat{\mathbf{y}}_{p1})' (\mathbf{y} - \hat{\mathbf{y}}_{p1})
\end{aligned}$$

预测值

$$\hat{\mathbf{y}}_0 = \mathbf{x}'_{p1,0} \hat{\beta}_{p1}$$

欠拟合

模型

全模型为真

$$\begin{aligned}
\mathbf{y} &= \mathbf{X}_p \beta_p + \boldsymbol{\varepsilon} \\
&= (\mathbf{X}_{p1} \quad \mathbf{Z}) \begin{pmatrix} \beta_{p1} \\ \boldsymbol{\gamma} \end{pmatrix} + \boldsymbol{\varepsilon} \\
&= \mathbf{X}_{p1} \beta_{p1} + \mathbf{Z} \boldsymbol{\gamma} + \boldsymbol{\varepsilon}
\end{aligned}$$

错误地使用了选模型

$$\mathbf{y} = \mathbf{X}_{p1} \beta_{p1} + \boldsymbol{\varepsilon}$$

假定丢失了重要的自变量

$$\begin{aligned}
\text{rank}(\mathbf{X}_p) &> \text{rank}(\mathbf{X}_{p1}) \\
\boldsymbol{\gamma} &\neq \mathbf{0}'_{p-p1}
\end{aligned}$$

参数估计

考虑 $\hat{\beta}_{p1}$ 期望

$$\begin{aligned}
E(\hat{\beta}_{p1}) &= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} E(\mathbf{y}) \\
&= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} E(\mathbf{X}_{p1} \beta_{p1} + \mathbf{Z} \boldsymbol{\gamma} + \boldsymbol{\varepsilon}) \\
&= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} (\mathbf{X}_{p1} \beta_{p1} + \mathbf{Z} \boldsymbol{\gamma} + E(\boldsymbol{\varepsilon})) \\
&= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} (\mathbf{X}_{p1} \beta_{p1} + \mathbf{Z} \boldsymbol{\gamma}) \\
&= \beta_{p1} + (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} \boldsymbol{\gamma}
\end{aligned}$$

因此通常而言有偏，如果

$$\mathbf{X}'_{p1} \mathbf{Z} = 0$$

则其为无偏估计

考虑 $\hat{\sigma}_{p1}^2$ 参数

注意到

$$SS_E^p = \mathbf{y}' (\mathbf{I} - \mathbf{H}_{\mathbf{X}_p}) \mathbf{y} \quad SS_E^{p1} = \mathbf{y}' (\mathbf{I} - \mathbf{H}_{\mathbf{X}_{p1}}) \mathbf{y}$$

其中

$$\begin{aligned} \mathbf{H}_{\mathbf{X}_p} &= \mathbf{X}_p (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p \\ &= (\mathbf{X}_{p1} \quad \mathbf{Z}) \left(\begin{array}{cc} \mathbf{X}'_{p1} \mathbf{X}_{p1} & \mathbf{X}'_{p1} \mathbf{Z} \\ \mathbf{Z}' \mathbf{X}_{p1} & \mathbf{Z}' \mathbf{Z} \end{array} \right)^{-1} \left(\begin{array}{c} \mathbf{X}'_{p1} \\ \mathbf{Z}' \end{array} \right) \end{aligned}$$

而

$$\begin{aligned} \mathbf{H}_{\mathbf{X}_p} &= (\mathbf{X}_{p1} \quad \mathbf{Z}) \left(\begin{array}{cc} \mathbf{X}'_{p1} \mathbf{X}_{p1} & \mathbf{X}'_{p1} \mathbf{Z} \\ \mathbf{Z}' \mathbf{X}_{p1} & \mathbf{Z}' \mathbf{Z} \end{array} \right)^{-1} \left(\begin{array}{c} \mathbf{X}'_{p1} \\ \mathbf{Z}' \end{array} \right) \\ &= (\mathbf{X}_{p1} \quad \mathbf{Z}) \left(\begin{array}{cc} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{array} \right) \left(\begin{array}{c} \mathbf{X}'_{p1} \\ \mathbf{Z}' \end{array} \right) \\ &= \mathbf{H}_{\mathbf{X}_{p1}} + \mathbf{H}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{H}_{\mathbf{X}_{p1}} \\ &\quad - \mathbf{H}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' - \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{H}_{\mathbf{X}_{p1}} + \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \\ &= \mathbf{H}_{\mathbf{X}_{p1}} + \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \end{aligned}$$

其中

$$\begin{aligned} \mathbf{D} &= (\mathbf{Z}' \mathbf{Z} - \mathbf{Z}' \mathbf{X}_{p1} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z})^{-1} = (\mathbf{Z}' (\mathbf{I} - \mathbf{H}_{\mathbf{X}_{p1}}) \mathbf{Z})^{-1} = (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \\ \mathbf{A} &= (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} + (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{X}_{p1} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \\ \mathbf{B} &= -(\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \\ \mathbf{C} &= -(\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{X}_{p1} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \end{aligned}$$

又因为

$$\begin{aligned} SS_E^{p1} &= SS_E^p + \mathbf{y}' (\mathbf{H}_{\mathbf{X}_p} - \mathbf{H}_{\mathbf{X}_{p1}}) \mathbf{y} \\ &= SS_E^p + \mathbf{y}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{y} \end{aligned}$$

期望

$$\begin{aligned} E(SS_E^{p1}) &= E(SS_E^p) + E(\mathbf{y}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{y}) \\ &= (n - p - 1)\sigma^2 + E(\mathbf{y})' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} E(\mathbf{y}) \\ &\quad + \sigma^2 \text{tr}(\mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} (\mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z})^{-1} \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}}) \\ &= (n - p - 1)\sigma^2 + \boldsymbol{\gamma}' \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} \boldsymbol{\gamma} + (p - p_1)\sigma^2 \\ &= (n - p_1 - 1)\sigma^2 + \boldsymbol{\gamma}' \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} \boldsymbol{\gamma} \end{aligned}$$

注意到

$$E(\hat{\sigma}_{p1}^2) = \frac{1}{n - p_1 - 1} E(SS_E^{p1}) = \sigma^2 + \frac{\boldsymbol{\gamma}' \mathbf{Z}' \mathbf{N}_{\mathbf{X}_{p1}} \mathbf{Z} \boldsymbol{\gamma}}{n - p_1 - 1} > \sigma^2$$

因为之前假设

$$\text{rank}(\mathbf{X}_p) > \text{rank}(\mathbf{X}_{p1})$$

因此有偏

预测

全模型是正确的，因此预测值

$$\hat{y}_{0,T} = \mathbf{x}'_{p,0} \hat{\boldsymbol{\beta}}_p$$

期望和方差

$$E(\hat{y}_{0,T}) = \mathbf{x}'_{p,0} E(\hat{\boldsymbol{\beta}}_p) = \mathbf{x}'_{p1,0} \boldsymbol{\beta}_{p1} + \mathbf{z}'_0 \boldsymbol{\gamma}$$
$$\text{Var}(\hat{y}_{0,T}) = \sigma^2 \mathbf{x}'_{p,0} (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{x}_{p,0}$$

但我们错误的使用了选模型，此时预测值为

$$\hat{y}_0 = \mathbf{x}'_{p1,0} \hat{\boldsymbol{\beta}}_{p1}$$

期望

$$E(\hat{y}_0) = \mathbf{x}'_{p1,0} \left(\boldsymbol{\beta}_{p1} + (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} \boldsymbol{\gamma} \right)$$
$$= \mathbf{x}'_{p1,0} \boldsymbol{\beta}_{p1} + \mathbf{x}'_{p1,0} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} \boldsymbol{\gamma}$$

预测偏差

$$\left(\mathbf{x}'_{p1,0} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{X}'_{p1} \mathbf{Z} - \mathbf{z}'_0 \right) \boldsymbol{\gamma}$$

方差差异

$$\text{Var}(\hat{y}_{0,T}) = \sigma^2 (\mathbf{x}'_{p1,0}, \mathbf{z}'_0) \mathbf{A} (\mathbf{x}'_{p1,0}, \mathbf{z}'_0)'$$
$$= \sigma^2 \mathbf{x}'_{p1,0} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{x}_{p1,0}$$
$$+ \sigma^2 (\mathbf{L}' \mathbf{x}_{p1,0} - \mathbf{z}_0)' \mathbf{M} (\mathbf{L}' \mathbf{x}_{p1,0} - \mathbf{z}_0)$$
$$\geq \sigma^2 \mathbf{x}'_{p1,0} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{x}_{p1,0}$$
$$= \text{Var}(\hat{y}_0)$$

方差变小

过拟合

系数期望无偏

$$E(\hat{\boldsymbol{\beta}}_p) = (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p E(\mathbf{y})$$
$$= (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p \left(\begin{pmatrix} \mathbf{X}_{p1} & \mathbf{Z} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta}_{p1} \\ \mathbf{0} \end{pmatrix} + E(\boldsymbol{\varepsilon}) \right)$$
$$= (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p (\mathbf{X}_{p1} \quad \mathbf{Z}) \begin{pmatrix} \boldsymbol{\beta}_{p1} \\ \mathbf{0} \end{pmatrix}$$
$$= (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p \mathbf{X}_p \begin{pmatrix} \boldsymbol{\beta}_{p1} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \boldsymbol{\beta}_{p1} \\ \mathbf{0} \end{pmatrix}$$

参数估计无偏

$$\begin{aligned}
E(SS_E^p) &= E(\mathbf{y}' \mathbf{N}_{\mathbf{X}_p} \mathbf{y}) \\
&= E(\mathbf{y})' \mathbf{N}_{\mathbf{X}_p} E(\mathbf{y}) + \sigma^2 \operatorname{tr}(\mathbf{N}_{\mathbf{X}_p}) \\
&= (\boldsymbol{\beta}'_p \quad \mathbf{0}') \mathbf{X}'_p \mathbf{N}_{\mathbf{X}_p} \mathbf{X}_p \begin{pmatrix} \boldsymbol{\beta}_p \\ \mathbf{0} \end{pmatrix} + (n - p - 1)\sigma^2 \\
&= (n - p - 1)\sigma^2
\end{aligned}$$

$$\hat{\sigma}_p^2 = \frac{SS_E^p}{n - p - 1}$$

预测期望无偏

$$\begin{aligned}
E(\hat{y}_0) &= \mathbf{x}'_{p,0} E(\hat{\boldsymbol{\beta}}_p) = (\mathbf{x}'_{p1,0} \quad \mathbf{z}'_0) \begin{pmatrix} \boldsymbol{\beta}_{p1} \\ 0 \end{pmatrix} \\
&= \mathbf{x}'_{p1,0} \boldsymbol{\beta}_{p1} = E(\hat{y}_{0,T})
\end{aligned}$$

预测方差偏大

$$\begin{aligned}
\operatorname{Var}(\hat{y}_0) &= \operatorname{Var}(\mathbf{x}'_{p,0} (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{X}'_p \mathbf{y}) \\
&= \sigma^2 \mathbf{x}'_{p,0} (\mathbf{X}'_p \mathbf{X}_p)^{-1} \mathbf{x}_{p,0} \\
&= \sigma^2 \mathbf{x}'_{p,0} \begin{pmatrix} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} + \mathbf{L} \mathbf{M} \mathbf{L}' & -\mathbf{L} \mathbf{M} \\ -\mathbf{M} \mathbf{L}' & \mathbf{M} \end{pmatrix} \mathbf{x}_{p,0} \\
&= \sigma^2 \mathbf{x}'_{p1,0} (\mathbf{X}'_{p1} \mathbf{X}_{p1})^{-1} \mathbf{x}_{p1,0} \\
&\quad + \sigma^2 (\mathbf{L}' \mathbf{x}_{p1,0} - \mathbf{z}_0)' \mathbf{M} (\mathbf{L}' \mathbf{x}_{p1,0} - \mathbf{z}_0) \\
&\geq \operatorname{Var}(\hat{y}_{0,T})
\end{aligned}$$

准则

结论

残差平方和重要结论

$$SS_E^{p1+1} \leq SS_E^{p1}$$

决定系数增加

$$R_{p1+1}^2 \geq R_{p1}^2$$

准则

1. 修正决定系数

$$\tilde{R}^2 = 1 - \frac{n - 1}{n - p - 1} (1 - R^2)$$

2. 方差的无偏估计

$$\hat{\sigma}^2 = \frac{1}{n - p - 1} SS_E$$

3. 赤池信息量准则

$$AIC = -2 \ln(\text{模型最大似然}) + 2(\text{模型独立参数个数})$$

带入模型

$$\begin{aligned} AIC &= n \ln(2\pi) + n \ln\left(\frac{SS_E}{n}\right) + n + 2(p+2) \\ &\propto n \ln(SS_E/n) + 2(p+1) \end{aligned}$$

4. 贝叶斯信息量准则

$$BIC = -2 \ln(\text{模型最大似然}) + \ln(n)(\text{模型独立参数个数})$$

带入模型

$$BIC = n \ln(SS_E/n) + \ln(n)(p+1)$$

逐步回归

首先有前进法和后退法两种讨论

1. 选择一种变量选择的准则 (如 AIC 最小)
2. 放入/剔除一个自变量
3. 以此类推直到不满足准则

时间复杂度为 $O(p^2)$

但这两种都有点问题：

1. 前进法：进去了就出不来了
2. 后退法：一开始的计算量很大

逐步回归的基本思想是有进有出：

1. 首先采用前进法的思想，但引入每一个自变量后，需要对已选入的变量逐个确定，如果之前引入的自变量因为当前自变量的引入而导致模型不再优化时，需要将其从回归方程中剔除。
2. 直到加入其他任意一个自变量或者剔除任何一个自变量，模型都不再优化。



多重共线性

定义和原因

最小二乘估计

$$\hat{\beta} = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y}$$

首先，如果自变量线性相关，则无法估计。如果相关性很高，那么

$$|\mathbf{X}' \mathbf{X}| \approx 0$$

由于

$$\text{Var}(\hat{\beta}) = \sigma^2 (\mathbf{X}' \mathbf{X})^{-1}$$

那么虽然可以得到估计，但精度很低。

完全共线性出现的场景及解决：

分类变量 (one-hot coding)：有 K 类变量，只能设置 $K - 1$ 个虚拟变量。

多余属性：房间大小和居住面积（一般情况）；电影票价格、人数与订单总额等等。

诊断

方差因子扩大法

定义

自变量 $\mathbf{X} = (x_1, x_2, \dots, x_p)$ 做标准化，记为 \mathbf{X}_s

而标准化后的自变量相关系数矩阵 $\mathbf{X}_s' \mathbf{X}_s = (r_{ij})$

其逆矩阵

$$\mathbf{C} = (c_{ij}) = (\mathbf{X}_s' \mathbf{X}_s)^{-1}$$

称矩阵主对角线元素

$$\text{VIF}_j = c_{jj}$$

为自变量 x_j 的方差扩大因子。

由来（第一种解释）

对因变量和自变量进行标准化后系数的关系

$$\hat{\beta}_{s,j} = \frac{\sqrt{l_{jj}}}{\sqrt{l_{yy}}} \hat{\beta}_j$$

我们这里不对因变量进行标准化，有

$$\hat{\beta}_{s,j} = \sqrt{l_{jj}} \hat{\beta}_j$$

易证

$$\text{Var}(\hat{\beta}_j) = \frac{c_{jj}}{l_{jj}} \sigma^2, \quad j = 1, 2, \dots, p$$

其中

$$l_{jj} = \sum_{i=1}^n (x_{ij} - \bar{x}_j)^2$$

说明：由于 $\text{VIF}_j = c_{jj}$ 越大，对应的回归系数的方差越大。

多重共线性（另一种解释）

我们将 x_j 作为因变量，与其余的 $p - 1$ 个自变量建立起多元回归模型。

可以证明

$$c_{jj} = \frac{1}{1 - R_j^2}$$

其中 R_j^2 表示其复决定系数

说明： R_j^2 度量了自变量 x_j 与其余自变量的线性相关程度。越大则说明方差扩大因子越大。

判断

对变量而言

$$\text{VIF}_j \geq c\text{VIF}$$

则说明自变量 x_j 与其余自变量之间存在多重共线性。

用平均数

$$\overline{\text{VIF}} = \frac{1}{p} \sum_{j=1}^p \text{VIF}_j$$

度量整个设计矩阵的多重共线性。

特征值判定法

判定方法

对 $\mathbf{X}'\mathbf{X}$ 特征值分解, 特征值从大到小排列, 计算

$$\kappa_j = \sqrt{\frac{\lambda_1}{\lambda_j}}, j = 1, 2, \dots, p$$

称其为特征值 λ_j 的条件数。

判定方法同上。

直观判定法

以下情况存在多重共线性:

1. 当增加或者剔除一个自变量, 其他自变量的回归系数估计值或显著性发生较大变化
2. 一些重要的自变量没有通过显著性检验或者标准误较大
3. 与因变量的简单相关系数绝对值很大但其没有通过显著性检验
4. 正负号与定性分析相反
5. 自变量间的相关系数较大

消除方法

1. 删去一些不重要的自变量
2. 增加样本量
3. 改进经典的 OLS

之后讨论岭回归和主成分回归时, 假定设计矩阵经过**标准化**, 而因变量未标准化。

岭回归

定义

为了矩阵求逆的方便, 我们采用

$$\mathbf{X}'\mathbf{X} + k\mathbf{I}, \quad k > 0$$

代替 $\mathbf{X}'\mathbf{X}$

称

$$\hat{\beta}(k) = (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1} \mathbf{X}'\mathbf{y}$$

为回归系数的岭回归估计, 其中 k 为岭参数。

注意：

1. $\mathbf{X}'\mathbf{X}$ 为自变量样本相关系数矩阵
2. 岭回归估计是关于回归参数的一个估计族

性质

定理一：岭回归估计是有偏的

证明

$$\begin{aligned} E(\hat{\boldsymbol{\beta}}(k)) &= E\left(\left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\mathbf{X}'\mathbf{y}\right) \\ &= \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\mathbf{X}'E(\mathbf{y}) \\ &= \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\mathbf{X}'\mathbf{X}\boldsymbol{\beta} \neq \boldsymbol{\beta} \end{aligned}$$

和 OLS 的关系

根据定义，有

$$\begin{aligned} \hat{\boldsymbol{\beta}}(k) &= \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\mathbf{X}'\mathbf{y} \\ &= \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\left(\mathbf{X}'\mathbf{X}\right)\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y} \\ &= \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}\right)^{-1}\left(\mathbf{X}'\mathbf{X}\right)\hat{\boldsymbol{\beta}} \end{aligned}$$

如果 k 与 \mathbf{y} 无关，则岭回归估计是 OLS 的线性变换。但 k 是由数据决定的，因此本质上说岭回归不是 OLS 的线性变换。

OLS 可以看做是岭回归的特例

$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}'\mathbf{X} + 0 \cdot \mathbf{I}\right)^{-1}\mathbf{X}'\mathbf{y} = \hat{\boldsymbol{\beta}}(0)$$

我们可以用均方误差比较两种估计。

定理二：岭回归估计 MSE 小于 OLS

存在 k ，使得

$$\text{MSE}(\hat{\boldsymbol{\beta}}(k)) \leq \text{MSE}(\hat{\boldsymbol{\beta}}(0))$$

证明：

我们先令

$$H(k) = \text{MSE}(\hat{\boldsymbol{\beta}}(k))$$

有

$$\begin{aligned}
H(k) &= \text{MSE}(\hat{\beta}(k)) = E(\hat{\beta}(k) - \beta)'(\hat{\beta}(k) - \beta) \\
&= E(\hat{\beta}(k) - E(\hat{\beta}(k)))'(\hat{\beta}(k) - E(\hat{\beta}(k))) \\
&\quad + (E(\hat{\beta}(k)) - \beta)'(E(\hat{\beta}(k)) - \beta) \\
&=: I_1(k) + I_2(k)
\end{aligned}$$

我们主要观察一下 0 的右邻域是否小于 $H(0)$ 即可, 于是我们可以转而分析导数

$$\frac{\partial H(k)}{\partial k} = \frac{\partial I_1(k)}{\partial k} + \frac{\partial I_2(k)}{\partial k}$$

先定义一些符号

$$\begin{aligned}
\hat{\beta}(k) &= (\mathbf{X}' \mathbf{X} + k\mathbf{I})^{-1} \mathbf{X}' \mathbf{y} \stackrel{\text{def}}{=} \mathbf{W}_k \mathbf{X}' \mathbf{y} \\
&= (\mathbf{X}' \mathbf{X} + k\mathbf{I})^{-1} (\mathbf{X}' \mathbf{X}) \hat{\beta} \stackrel{\text{def}}{=} \mathbf{W}_k^* \hat{\beta}
\end{aligned}$$

关系

$$\begin{aligned}
\mathbf{W}_k^* &= \mathbf{W}_k (\mathbf{X}' \mathbf{X}) \\
&= (\mathbf{X}' \mathbf{X} + k\mathbf{I})^{-1} (\mathbf{X}' \mathbf{X} + k\mathbf{I} - k\mathbf{I}) \\
&= \mathbf{I} - k (\mathbf{X}' \mathbf{X} + k\mathbf{I})^{-1} \\
&= \mathbf{I} - k \mathbf{W}_k
\end{aligned}$$

对 $\mathbf{X}' \mathbf{X}$ 特征值分解, 特征值从大到小排列, 有

$$(\mathbf{X}' \mathbf{X}) \mathbf{v}_j = \lambda_j \mathbf{v}_j, \quad j = 1, 2, \dots, p$$

等式两端同时加

$$(\mathbf{X}' \mathbf{X} + k\mathbf{I}) \mathbf{v}_j = (\mathbf{X}' \mathbf{X}) \mathbf{v}_j + k\mathbf{I} \cdot \mathbf{v}_j = \lambda_j \mathbf{v}_j + k\mathbf{I} \cdot \mathbf{v}_j = (\lambda_j + k) \mathbf{v}_j$$

那么

$$(\mathbf{X}' \mathbf{X} + k\mathbf{I})^{-1} \mathbf{v}_j = \frac{1}{\lambda_j + k} \mathbf{v}_j \Rightarrow \mathbf{W}_k \mathbf{v}_j = \frac{1}{\lambda_j + k} \mathbf{v}_j$$

同理, 有

$$(\mathbf{I} + k (\mathbf{X}' \mathbf{X})^{-1})^{-1} \mathbf{v}_j = \frac{\lambda_j}{\lambda_j + k} \mathbf{v}_j \Rightarrow \mathbf{W}_k^* \mathbf{v}_j = \frac{\lambda_j}{\lambda_j + k} \mathbf{v}_j$$

首先, 考虑

$$\begin{aligned}
I_1(k) &= E(\hat{\beta}(k) - E(\hat{\beta}(k)))'(\hat{\beta}(k) - E(\hat{\beta}(k))) \\
&= E(\mathbf{W}_k^* \hat{\beta} - \mathbf{W}_k^* \beta)'(\mathbf{W}_k^* \hat{\beta} - \mathbf{W}_k^* \beta) \\
&= E((\hat{\beta} - \beta)' (\mathbf{W}_k^*)' (\mathbf{W}_k^*) (\hat{\beta} - \beta)) \\
&= E(\boldsymbol{\varepsilon}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} (\mathbf{W}_k^*)' (\mathbf{W}_k^*) (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \boldsymbol{\varepsilon})
\end{aligned}$$

最后一个等式成立是因为

$$\begin{aligned}\hat{\beta} - \beta &= (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} - \beta = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' (\mathbf{X} \beta + \varepsilon) - \beta \\ &= \beta + (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \varepsilon - \beta = (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \varepsilon\end{aligned}$$

多元统计知识补充：

假设 \mathbf{A} 是对称矩阵， \mathbf{x} 是一个 p 维随机变量，有

$$E(\mathbf{x}' \mathbf{A} \mathbf{x}) = \text{tr}(\mathbf{A} \Sigma) + \boldsymbol{\mu}' \mathbf{A} \boldsymbol{\mu}.$$

其中 $\boldsymbol{\mu} = E(\mathbf{x})$ ， $\Sigma = \text{Var}(\mathbf{x})$

考虑第一部分

$$\begin{aligned}I_1(k) &= E\left(\varepsilon' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} (\mathbf{W}_k^*)' (\mathbf{W}_k^*) (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \varepsilon\right) \\ &= \sigma^2 \text{tr}\left((\mathbf{X}' \mathbf{X})^{-1} (\mathbf{W}_k^*)' (\mathbf{W}_k^*)\right) \\ &= \sigma^2 \text{tr}\left((\mathbf{X}' \mathbf{X})^{-1} (\mathbf{X}' \mathbf{X}) \mathbf{W}_k (\mathbf{I} - k \mathbf{W}_k)\right) \\ &= \sigma^2 (\text{tr}(\mathbf{W}_k) - k \text{tr}(\mathbf{W}_k^2)) \\ &= \sigma^2 \left(\sum_{j=1}^p \frac{1}{\lambda_j + k} - k \sum_{j=1}^p \frac{1}{(\lambda_j + k)^2}\right) \\ &= \sigma^2 \sum_{j=1}^p \frac{\lambda_j}{(\lambda_j + k)^2}\end{aligned}$$

所以

$$\frac{\partial I_1(k)}{\partial k} = -2\sigma^2 \sum_{j=1}^p \frac{\lambda_j}{(\lambda_j + k)^3} < 0$$

考虑第二部分

$$\begin{aligned}I_2(k) &= (E(\hat{\beta}(k)) - \beta)' (E(\hat{\beta}(k)) - \beta) \\ &= (\mathbf{W}_k^* \beta - \beta)' (\mathbf{W}_k^* \beta - \beta) \\ &= \beta' (\mathbf{W}_k^* - \mathbf{I})' (\mathbf{W}_k^* - \mathbf{I}) \beta \\ &= k^2 \beta' \mathbf{W}_k^2 \beta \\ &= k^2 \beta' \mathbf{V}' \mathbf{L} \mathbf{V} \beta \quad \text{令 } (\mathbf{W}_k^2 = \mathbf{V}' \mathbf{L} \mathbf{V}) \\ &= k^2 \boldsymbol{\alpha}' \mathbf{L} \boldsymbol{\alpha} \quad \text{令 } (\boldsymbol{\alpha} = \mathbf{V} \beta) \\ &= k^2 \sum_{j=1}^p \frac{\alpha_j^2}{(\lambda_j + k)^2}\end{aligned}$$

其中

$$\boldsymbol{\alpha} = \mathbf{V} \beta = (\alpha_1, \alpha_2, \dots, \alpha_p)'$$

与岭参数无关，由于

$$I_2(k) = k^2 \sum_{j=1}^p \frac{\alpha_j^2}{(\lambda_j + k)^2}$$

因此

$$\begin{aligned}\frac{\partial I_2(k)}{\partial k} &= 2 \sum_{j=1}^p \frac{k \alpha_j^2}{(\lambda_j + k)^2} - 2 \sum_{j=1}^p \frac{k^2 \alpha_j^2}{(\lambda_j + k)^3} \\ &= 2k \sum_{j=1}^p \frac{\lambda_j \alpha_j^2}{(\lambda_j + k)^3} \geq 0\end{aligned}$$

两者结合

$$\begin{aligned}\frac{\partial H(k)}{\partial k} &= \frac{\partial I_1(k)}{\partial k} + \frac{\partial I_2(k)}{\partial k} \\ &= -2\sigma^2 \sum_{j=1}^p \frac{\lambda_j}{(\lambda_j + k)^3} + 2k \sum_{j=1}^p \frac{\lambda_j \alpha_j^2}{(\lambda_j + k)^3}\end{aligned}$$

考虑 0 的右邻域

$$\left. \frac{\partial H(k)}{\partial k} \right|_{k=0} = \left. \frac{\partial I_1(k)}{\partial k} \right|_{k=0} + \left. \frac{\partial I_2(k)}{\partial k} \right|_{k=0} = -2\sigma^2 \sum_{j=1}^p \lambda_j^{-2} < 0$$

因此定理得证。

但

$$\begin{aligned}\frac{\partial H(k)}{\partial k} &= \frac{\partial I_1(k)}{\partial k} + \frac{\partial I_2(k)}{\partial k} \\ &= -2\sigma^2 \sum_{j=1}^p \frac{\lambda_j}{(\lambda_j + k)^3} + 2k \sum_{j=1}^p \frac{\lambda_j \alpha_j^2}{(\lambda_j + k)^3} \\ &= \sum_{j=1}^m \frac{2\lambda_j}{(\lambda_j + k)^3} (k \alpha_j^2 - \sigma^2)\end{aligned}$$

我们无法找到一个通用的解 k ，使得表达式最小（因为 β, σ^2 都是未知数）

另一角度

可以证明岭回归估计是最小化带有 L_2 正则项的离差平方和的解，即

$$\hat{\beta}(k) = \arg \min_{\beta} (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) + \lambda \beta' \beta$$

而这个又等价于

$$\min \quad (\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta) \quad \text{s.t.} \quad \beta' \beta \leq s$$

那么，我们发现：

1. 如果最小二乘估计满足约束，则 $\hat{\beta}(k) = \hat{\beta}$
2. 如果不满足约束，则有 $\hat{\beta}(k)' \hat{\beta}(k) \leq s < \hat{\beta}' \hat{\beta}$

第一个证明：符号对应 ($\theta = \beta$, $\mathbf{X}^T = \mathbf{X}'$)

$$\begin{aligned}
J(\theta) &= (\mathbf{y} - \mathbf{X}\theta)^T(\mathbf{y} - \mathbf{X}\theta) + \lambda^2\theta^T\theta \\
&= (\mathbf{y}^T - \theta^T\mathbf{X}^T)(\mathbf{y} - \mathbf{X}\theta) + \lambda^2\theta^T\theta \\
&= \mathbf{y}^T\mathbf{y} - \mathbf{y}^T\mathbf{X}\theta - \theta^T\mathbf{X}^T\mathbf{y} + \theta^T\mathbf{X}^T\mathbf{X}\theta + \lambda^2\theta^T\theta \\
&= \mathbf{y}^T\mathbf{y} - 2\theta^T\mathbf{X}^T\mathbf{y} + \theta^T\mathbf{X}^T\mathbf{X}\theta + \lambda^2\theta^T\theta \\
\frac{\partial}{\partial\theta}J(\theta) &= -2\mathbf{X}^T\mathbf{y} + 2\theta\mathbf{X}^T\mathbf{X} + 2\lambda^2\theta = 0 \\
\theta &= (\mathbf{X}^T\mathbf{X} + \lambda^2\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}
\end{aligned}$$

令 $k = \lambda^2$ ，则得到岭回归估计的闭式解。

第二个证明就是拉格朗日乘数法，可以推导到相同的闭式解。

定理三：岭回归估计是 OLS 的一种压缩

对任意 $k > 0$ ， $\|\hat{\beta}\| \neq 0$ ，总有

$$\|\hat{\beta}(k)\| < \|\hat{\beta}\|$$

证明：

对 $\mathbf{X}'\mathbf{X}$ 特征值分解，特征值从大到小排列，有

$$\mathbf{X}'\mathbf{X} = \mathbf{V}'\Lambda\mathbf{V}$$

回归模型

$$\begin{aligned}
\mathbf{y} &= \mathbf{X}\beta + \varepsilon \\
&= \mathbf{X}\mathbf{V}'\mathbf{V}\beta + \varepsilon \\
&=: \mathbf{Z}\alpha + \varepsilon
\end{aligned}$$

由于

$$\alpha = \mathbf{V}\beta \Rightarrow \beta = \mathbf{V}'\alpha$$

因此，其最小二乘估计

$$\begin{aligned}
\hat{\alpha} &= (\mathbf{Z}'\mathbf{Z})^{-1}\mathbf{Z}'\mathbf{y} = (\mathbf{V}\mathbf{X}'\mathbf{X}\mathbf{V}')^{-1}\mathbf{Z}'\mathbf{y} \\
&= (\mathbf{V}\mathbf{V}'\Lambda\mathbf{V}\mathbf{V}')^{-1}\mathbf{Z}'\mathbf{y} = \Lambda^{-1}\mathbf{Z}'\mathbf{y}
\end{aligned}$$

而 OLS 与其存在如下关系

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{V}'\Lambda^{-1}\mathbf{V}\mathbf{X}'\mathbf{y} = \mathbf{V}'\alpha$$

类似

$$\begin{aligned}
\hat{\alpha}(k) &= (\Lambda + k\mathbf{I})^{-1}\mathbf{Z}'\mathbf{y} \\
\hat{\beta}(k) &= \mathbf{V}'\hat{\alpha}(k)
\end{aligned}$$

所以

$$\|\hat{\beta}(k)\| = \|\hat{\alpha}(k)\| = \|(\Lambda + kI)^{-1}\Lambda\hat{\alpha}\| < \|\hat{\alpha}\| = \|\hat{\beta}\|$$

说明：

$\hat{\beta}(k)$ 是对 $\hat{\beta}$ 向原点的压缩，这是因为

$$\begin{aligned} \text{MSE}(\hat{\beta}) &= E \left((\hat{\beta} - \beta)' (\hat{\beta} - \beta) \right) \\ &= E \left(\hat{\beta}' \hat{\beta} \right) - \beta' \beta = E \|\hat{\beta}\|^2 - \|\beta\|^2 \end{aligned}$$

因此

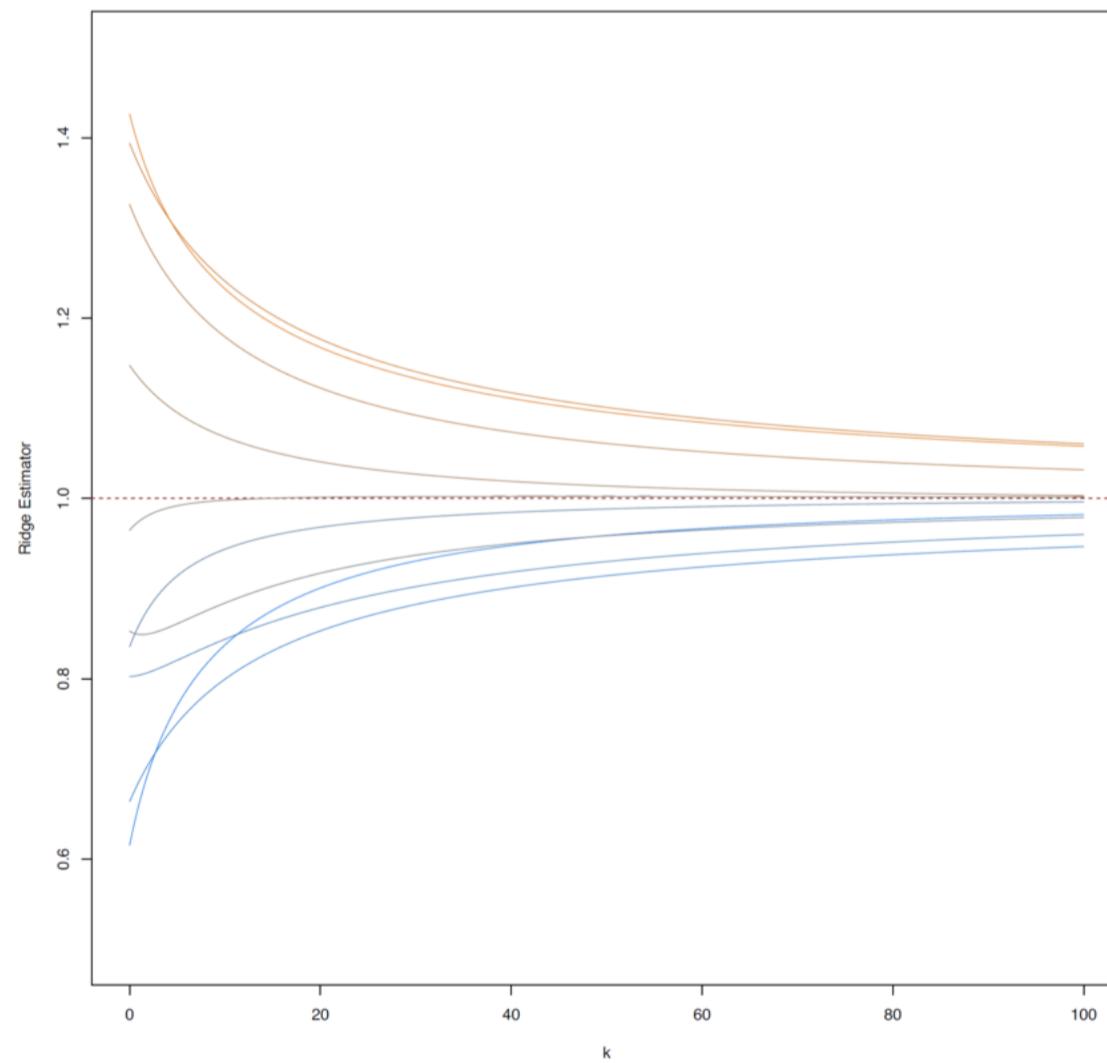
$$E \|\hat{\beta}\|^2 = \|\beta\|^2 + \text{MSE}(\hat{\beta}) = \|\beta\|^2 + \sigma^2 \sum_{j=1}^p \lambda_j^{-1}$$

当出现多重共线性时，第二项很大，因此应该对其进行合理压缩。

选择

岭迹法

$k - \hat{\beta}_j(k)$ 图像



原则：

1. 岭估计基本稳定
2. 符号合理
3. 残差平方和增大不多

VIF 法

岭回归估计方差

$$\begin{aligned}\text{Var}(\hat{\beta}(k)) &= \sigma^2 (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1} \mathbf{X}'\mathbf{X} (\mathbf{X}'\mathbf{X} + k\mathbf{I})^{-1} \\ &\stackrel{\text{def}}{=} \sigma^2 \mathbf{C}(k)\end{aligned}$$

主成分回归

主成分分析与回归分析的综合。

主成分分析

摘自我的实验报告内容：）

定理四：主成分定义

样本估计

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} = \begin{pmatrix} \mathbf{x}'_1 \\ \mathbf{x}'_2 \\ \vdots \\ \mathbf{x}'_n \end{pmatrix}$$

样本协方差阵

$$S = \frac{1}{n-1} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})' \stackrel{\text{def}}{=} (s_{kl})_{p \times p}$$

其中

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i \stackrel{\text{def}}{=} (\bar{x}_1, \dots, \bar{x}_p)'$$

$$s_{kl} = \frac{1}{n-1} \sum_{i=1}^n (x_{ik} - \bar{x}_k) (x_{il} - \bar{x}_l)$$

样本相关阵

$$R = (r_{kl})_{p \times p}$$

其中

$$r_{kl} = \frac{s_{kl}}{\sqrt{s_{kk} s_{ll}}}$$

其实只需要标准化后的矩阵就可以直接得到样本相关阵

$$\mathbf{X}^* = \begin{pmatrix} x_{11}^* & x_{12}^* & \cdots & x_{1p}^* \\ x_{21}^* & x_{22}^* & \cdots & x_{2p}^* \\ \vdots & \vdots & & \vdots \\ x_{n1}^* & x_{n2}^* & \cdots & x_{np}^* \end{pmatrix}$$

然后考虑 $(\mathbf{X}^*)' \mathbf{X}^*$ 的每一个元素

$$\begin{aligned} \sum_{i=1}^n (x_{ik}^*) (x_{il}^*) &= \sum_{i=1}^n \frac{x_{ik} - \bar{x}_k}{\sqrt{s_{kk}}} \frac{x_{il} - \bar{x}_l}{\sqrt{s_{ll}}} \\ &= \frac{1}{n-1} \sum_{i=1}^n \frac{(x_{ik} - \bar{x}_k) (x_{il} - \bar{x}_l)}{\sqrt{s_{kk} s_{ll}}} \\ &= r_{kl} \end{aligned}$$

动机

还是老样子，对 $\mathbf{X}' \mathbf{X}$ 特征值分解，特征值从大到小排列，有

$$\mathbf{X}' \mathbf{X} = \mathbf{V}' \Lambda \mathbf{V}$$

其中

$$\begin{aligned}\mathbf{\Lambda} &= \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_p\} \\ \mathbf{V}' &= (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p)\end{aligned}$$

易知

$$\mathbf{V}(\mathbf{X}'\mathbf{X})\mathbf{V}' = (\mathbf{X}\mathbf{V}')'(\mathbf{X}\mathbf{V}') = \mathbf{\Lambda}$$

令

$$\mathbf{Z} = \mathbf{X}\mathbf{V}'$$

那么

$$\mathbf{Z}'\mathbf{Z} = \mathbf{\Lambda}$$

我们有

$$\sum_{i=1}^n z_{ij} = 0, \quad \sum_{i=1}^n z_{ij}^2 = \lambda_j, \quad \sum_{i=1}^n z_{ij} z_{ik} = 0 (j \neq k)$$

定义

线性回归模型

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon} = \mathbf{X}\mathbf{V}'\mathbf{V}\boldsymbol{\beta} + \boldsymbol{\varepsilon} = \mathbf{Z}_{n \times p}\boldsymbol{\alpha}_{p \times 1} + \boldsymbol{\varepsilon}$$

拆分

$$\begin{aligned}\boldsymbol{\alpha} &= (\alpha_1, \dots, \alpha_k, \alpha_{k+1}, \dots, \alpha_p)' = (\boldsymbol{\alpha}'_1, \boldsymbol{\alpha}'_2)', \\ \mathbf{Z} &= (\mathbf{z}_1, \dots, \mathbf{z}_k, \mathbf{z}_{k+1}, \dots, \mathbf{z}_p) = (\mathbf{Z}_1, \mathbf{Z}_2).\end{aligned}$$

$$\mathbf{\Lambda} = \begin{pmatrix} \mathbf{\Lambda}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda}_2 \end{pmatrix}$$

其中

$$\mathbf{\Lambda}_1 = \text{diag}\{\lambda_1, \dots, \lambda_k\}, \quad \mathbf{\Lambda}_2 = \text{diag}\{\lambda_{k+1}, \dots, \lambda_p\}$$

可以知道, 因为 $|\mathbf{X}'\mathbf{X}| \approx 0$, 所以存在 k 使得后续的特征值均近似为 0, 因此有

$$\mathbf{y} = \mathbf{Z}_1\boldsymbol{\alpha}_1 + \mathbf{Z}_2\boldsymbol{\alpha}_2 + \boldsymbol{\varepsilon} = \mathbf{Z}_1\boldsymbol{\alpha}_1 + \boldsymbol{\varepsilon}$$

其估计

$$\hat{\boldsymbol{\alpha}}_1 = (\mathbf{Z}'_1\mathbf{Z}_1)^{-1}\mathbf{Z}'_1\mathbf{y} = \mathbf{\Lambda}_1^{-1}\mathbf{Z}'_1\mathbf{y}$$

因为

$$\boldsymbol{\beta} = \mathbf{V}'\boldsymbol{\alpha}$$

因此估计

$$\hat{\beta}_{\text{PC}} = \mathbf{V}' \begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} = \mathbf{V}'_1 \hat{\alpha}_1 = \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{Z}'_1 \mathbf{y}$$

称其为 β 的主成分估计

性质

和 OLS 的关系

$$\begin{aligned}\hat{\beta}_{\text{PC}} &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{Z}'_1 \mathbf{y} \\ &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{V}_1 \mathbf{X}' \mathbf{y} \\ &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{V}_1 \mathbf{X}' \mathbf{X} (\mathbf{X}' \mathbf{X})^{-1} \mathbf{X}' \mathbf{y} \\ &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{V}_1 \mathbf{X}' \mathbf{X} \hat{\beta} \\ &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{V}_1 \mathbf{V}' \mathbf{\Lambda} \mathbf{V} \hat{\beta} \\ &= \mathbf{V}'_1 \mathbf{\Lambda}_1^{-1} \mathbf{V}_1 (\mathbf{V}'_1, \mathbf{V}'_2) \begin{pmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{pmatrix} \begin{pmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \end{pmatrix} \hat{\beta} \\ &= \mathbf{V}'_1 \mathbf{V}_1 \hat{\beta}.\end{aligned}$$

定理五：主成分估计是有偏的

$$\begin{aligned}E(\hat{\beta}_{\text{PC}}) &= E(\mathbf{V}'_1 \mathbf{V}_1 \hat{\beta}) \\ &= \mathbf{V}'_1 \mathbf{V}_1 E(\hat{\beta}) \\ &= \mathbf{V}'_1 \mathbf{V}_1 \beta\end{aligned}$$

因为

$$\mathbf{I}_p = \mathbf{V}' \mathbf{V} = \mathbf{V}'_1 \mathbf{V}_1 + \mathbf{V}'_2 \mathbf{V}_2$$

因此

$$\mathbf{V}'_1 \mathbf{V}_1 \beta = (\mathbf{I} - \mathbf{V}'_2 \mathbf{V}_2) \beta \neq \beta$$

定理六：主成分估计 MSE 小于 OLS

由于

$$\hat{\beta}_{\text{PC}} = \mathbf{V}' \begin{pmatrix} \hat{\alpha}_1 \\ \mathbf{0} \end{pmatrix}$$

有

$$\begin{aligned}
\text{MSE}(\hat{\beta}_{\text{PC}}) &= E(\hat{\beta}_{\text{PC}} - \beta)'(\hat{\beta}_{\text{PC}} - \beta) \\
&= E\left(\mathbf{V}' \begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \mathbf{V}'\alpha\right)' \left(\mathbf{V}' \begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \mathbf{V}'\alpha\right) \\
&= E\left(\begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}\right)' \mathbf{V} \mathbf{V}' \left(\begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}\right) \\
&= E\left(\begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}\right)' \left(\begin{pmatrix} \hat{\alpha}_1 \\ 0 \end{pmatrix} - \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}\right)
\end{aligned}$$

那么

$$\begin{aligned}
\text{MSE}(\hat{\beta}_{\text{PC}}) &= E(\hat{\alpha}_1 - \alpha_1)'(\hat{\alpha}_1 - \alpha_1) + \|\alpha_2\|^2 \\
&= E(\varepsilon' \mathbf{Z}_1 \mathbf{\Lambda}_1^{-2} \mathbf{Z}_1' \varepsilon) + \|\alpha_2\|^2 \\
&= \sigma^2 \text{tr}(\mathbf{\Lambda}_1^{-1}) + \|\alpha_2\|^2 \\
&= \sigma^2 \sum_{j=1}^k \lambda_j^{-1} + \sum_{j=k+1}^p \alpha_j^2 \\
&= \text{MSE}(\hat{\beta}) + \left(\sum_{j=k+1}^p \alpha_j^2 - \sigma^2 \sum_{j=k+1}^p \lambda_j^{-1} \right)
\end{aligned}$$

由于存在多重共线性，导致括号内为负，所以、

$$\text{MSE}(\hat{\beta}_{\text{PC}}) < \text{MSE}(\hat{\beta})$$

定理七：主成分估计是 OLS 的一种压缩

$$\begin{aligned}
\|\hat{\beta}_{\text{PC}}\|^2 &= (\hat{\beta}_{\text{PC}})' \hat{\beta}_{\text{PC}} \\
&= (\mathbf{V}_1' \mathbf{V}_1 \hat{\beta})' (\mathbf{V}_1' \mathbf{V}_1 \hat{\beta}) \\
&= \hat{\beta}' \mathbf{V}_1' \mathbf{V}_1 \mathbf{V}_1' \mathbf{V}_1 \hat{\beta} \\
&= \hat{\beta}' \mathbf{V}_1' \mathbf{V}_1 \hat{\beta} \\
&\leq \hat{\beta}' \hat{\beta} \\
&= \|\hat{\beta}\|^2
\end{aligned}$$

选择

1. 给定一个阈值，使得

$$\sum_{j=1}^{k-1} \frac{\lambda_j}{p} < c_{\text{pc}}, \quad \sum_{j=1}^k \frac{\lambda_j}{p} \geq c_{\text{pc}}.$$

由此选择 k

2. 删除特征值接近于 0 的主成分

$$\lambda_k \geq c_0, \lambda_{k+1} < c_0$$

3. 均方误差确定 k

$$\sum_{j=1}^k \lambda_j^{-1} \leq 5k$$

論

聚类分析



聚类分析

考点

层次聚类

自下而上实例

假设有四个点 $ABCD$ ，距离矩阵如下，使用 simple linkage

$$\begin{pmatrix} 0 & 1 & 3 & 2 \\ 1 & 0 & 5 & 6 \\ 3 & 5 & 0 & 4 \\ 2 & 6 & 4 & 0 \end{pmatrix}$$

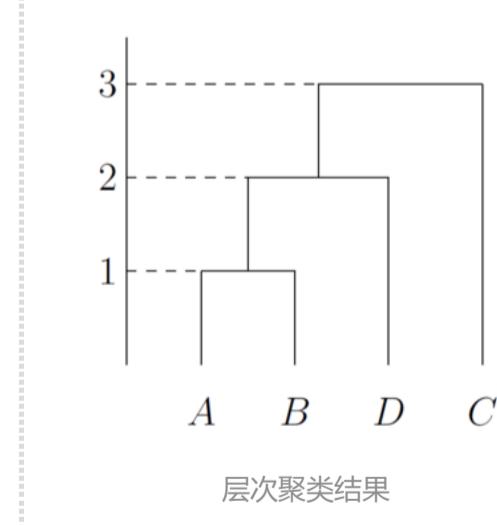
第一步，将 A 和 B 合并

$$\begin{pmatrix} 0 & 3 & 2 \\ 3 & 0 & 4 \\ 2 & 4 & 0 \end{pmatrix}$$

第二步，将 AB 和 D 合并

$$\begin{pmatrix} 0 & 3 \\ 3 & 0 \end{pmatrix}$$

最后聚成一类，结果



K-means

理解代码

每轮计算：

1. 样本距离，给每个数据一个标签

2. 计算新的均值向量

3. 重复 12 直到收敛

GMM

理解代码

考察 EM 算法 (详细可以查看西瓜书)。

每轮计算均值向量、协方差矩阵和混合系数。

E 步

$$\pi_{ik}^* = E(\delta_{ik} \mid \mathbf{x}_i) = P(\delta_{ik} = 1 \mid \mathbf{x}_i) = \frac{\pi_k \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{k=1}^K \pi_k \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}$$

$$\phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k) = (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -(\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) / 2 \right\}$$

M 步

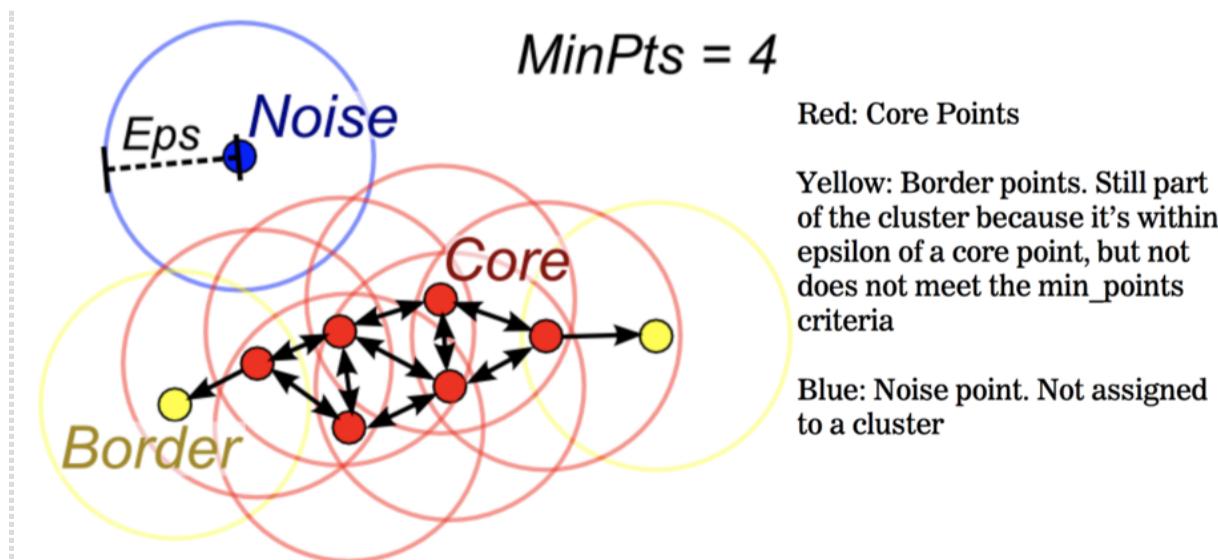
$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^n \pi_{ik}^* \mathbf{x}_i}{\sum_{i=1}^n \pi_{ik}^*}$$
$$\Sigma_k = \frac{\sum_{i=1}^n \pi_{ik}^* (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)'}{\sum_{i=1}^n \pi_{ik}^*}$$

$$\pi_k = \frac{1}{n} \sum_{i=1}^n \pi_{ik}^*$$

DBSCAN

理解代码

例子



算法

Algorithm 5 DBSCAN 聚类算法

Require: 样本集 $X = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)'$;
 邻域半径 ϵ ;
 可到达的最少样本个数 MinPts;

Ensure: DBSCAN 聚类的结果

- 1: 初始化: 类别序号 $k = 0$;
- 2: 标记每个样本点为未被访问过 $v_i = 0, i = 1, 2, \dots, n$;
- 3: **for** 从数据集 X 中随机选取一个未访问过的样本点 \mathbf{x}_i **do**
- 4: **if** 样本点 \mathbf{x}_i 是未访问过的, 即 $v_i = 0$ **then**
- 5: 标记点 \mathbf{x}_i 被访问过, 即 $v_i = 1$;
- 6: 计算 \mathbf{x}_i 的 ϵ 邻域, 即 $N_\epsilon(i) = \{\mathbf{x}_l \in X : \text{dist}(i, l) \leq \epsilon\}$;
- 7: **if** 样本点 \mathbf{x}_i 不是一个核心点, 即 $|N_\epsilon(i)| < \text{MinPts}$ **then**
- 8: 样本点 \mathbf{x}_i 是一个噪声;
- 9: **else**
- 10: $k = k + 1$;
- 11: 令 $G(i)$ 表示所有从样本点 \mathbf{x}_i 密度可达的样本点, 即 $G(i) = \{\mathbf{x}_l \in X : \text{点 } \mathbf{x}_l \text{ 可以从点 } \mathbf{x}_i \text{ 密度可达}\}$;
- 12: **for** 样本点 $\mathbf{x}_l \in G(i)$ **do**
- 13: **if** 样本点 \mathbf{x}_l 是未访问过的, 即 $v_l = 0$ **then**
- 14: 标记样本点 \mathbf{x}_l 为被访问过, 即 $v_l = 1$;
- 15: 将样本点 \mathbf{x}_k 归入类 \mathcal{C}_k 中;



聚类分析

聚类思想

动机

1. 识别从属特定总体的个体
2. 识别异常个体

核心问题

1. 相似性度量
2. 确定类别数目
3. 提取特征
4. 评价结果

基本定义

在样本空间上找到一组分割，使得类间距离尽可能大，类内距离尽可能小。

主要研究无标签数据集，因此是无监督学习。

数据集可以写成如下形式：

	1	2	...	j	...	p
1	x_{11}	x_{12}	...	x_{1j}	...	x_{1p}
2	x_{21}	x_{22}	...	x_{2j}	...	x_{2p}
\vdots	\vdots	\vdots		\vdots		\vdots
i	x_{i1}	x_{i2}	...	x_{ij}	...	x_{ip}
\vdots	\vdots	\vdots		\vdots		\vdots
n	x_{n1}	x_{n2}	...	x_{nj}	...	x_{np}

行代表样本，第 i 个样本

$$\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{ip})'$$

可以看作 p 维空间中的一个点。（我们这里把样本看成列向量，方便计算）

按行进行聚类，由此在数据集 X 中进行集群发现。

列代表特征，第 j 个特征

$$\mathbf{x}_j^* = (x_{1j}, x_{2j}, \dots, x_{nj})'$$

可以看作 n 维空间中的一个点。

按列进行聚类，由此在数据集 X 中进行降维。

距离定义

点间距离

考虑两个样本

$$\mathbf{x}_k = (x_{k1}, \dots, x_{kp})' \quad \mathbf{x}_l = (x_{l1}, \dots, x_{lp})'$$

连续变量

1. 欧式距离

$$\|\mathbf{x}_k - \mathbf{x}_l\|_2 = \sqrt{\sum_{j=1}^p (x_{kj} - x_{lj})^2}$$

2. 平方欧式距离

$$\|\mathbf{x}_k - \mathbf{x}_l\|_2^2 = \sum_{j=1}^p (x_{kj} - x_{lj})^2$$

3. 闵氏距离

$$\left(\sum_{j=1}^p (x_{kj} - x_{lj})^q \right)^{\frac{1}{q}}$$

4. 曼哈顿距离

$$\|\mathbf{x}_k - \mathbf{x}_l\|_1 = \sum_{j=1}^p |x_{kj} - x_{lj}|$$

5. 兰氏距离

$$\sum_{j=1}^p \frac{|x_{kj} - x_{lj}|}{|x_{kj}| + |x_{lj}|}$$

6. 切比雪夫距离

$$\|\mathbf{x}_k - \mathbf{x}_l\|_\infty = \max_j |x_{kj} - x_{lj}|$$

7. 马氏距离

$$\sqrt{(\mathbf{x}_k - \mathbf{x}_l)' \Sigma^{-1} (\mathbf{x}_k - \mathbf{x}_l)}$$

8. 皮尔逊线性相关系数距离

$$\text{Pearson r} = \frac{\sum_{j=1}^p (x_{kj} - \bar{x}_k)(x_{lj} - \bar{x}_l)}{\sqrt{\sum_{j=1}^p (x_{kj} - \bar{x}_k)^2 \sum_{j=1}^p (x_{lj} - \bar{x}_l)^2}}$$

$$\text{皮尔逊线性相关距离} = 1 - \text{Pearson r}$$

9. 余弦相似度

$$\cos \theta = \frac{\sum_{j=1}^p x_{kj} x_{lj}}{\sqrt{\sum_{j=1}^p x_{kj}^2 \sum_{j=1}^p x_{lj}^2}}$$

$$\text{余弦相关距离} = 1 - \cos \theta$$

10. 肯德尔秩相关系数

$$\text{协同对 (concordant pairs)} : (x_{kj} - x_{kj'}) (x_{lj} - x_{lj'}) > 0 \quad (1)$$

$$\text{不协同对 (discordant pairs)} : (x_{kj} - x_{kj'}) (x_{lj} - x_{lj'}) < 0 \quad (2)$$

$$\text{Kendall } \tau = \frac{n_c - n_d}{p(p-1)/2}$$

其中 n_c 表示协同对的个数, n_d 表示不协同对的个数。

$$\text{肯德尔相关距离} = 1 - \text{Kendall } \tau$$

11. 斯皮尔曼秩相关系数

$$x_{1j}, x_{2j}, \dots, x_{nj} \xrightarrow{\text{sort asc.}} r_{1j}, r_{2j}, \dots, r_{nj}$$

将原始数值用秩替代, 然后类似于皮尔逊线性相关系数

$$\text{Spearman } \rho = \frac{\sum_{j=1}^p (r_{kj} - \bar{r}_{kj})(r_{lj} - \bar{r}_{lj})}{\sqrt{\sum_{j=1}^p (r_{kj} - \bar{r}_{kj})^2 \sum_{j=1}^p (r_{lj} - \bar{r}_{lj})^2}}$$

$$\text{斯皮尔曼相关距离} = 1 - \text{Spearman } \rho$$

混合变量

变量相似度

$$s_j = s_j(x_{kj}, x_{lj})$$

变量间距离

$$d_j \triangleq 1 - s_j$$

1. 定性变量

$$s_j = s_j(x_{kj}, x_{lj}) = \begin{cases} 1 & \text{如果 } x_{kj} \text{ 和 } x_{lj} \text{ 相同} \\ 0 & \text{如果 } x_{kj} \text{ 和 } x_{lj} \text{ 不同} \end{cases}$$

2. 定量变量

$$s_j = s_j(x_{kj}, x_{lj}) = 1 - \frac{|x_{kj} - x_{lj}|}{R_j}$$

3. 定序变量

$$s_j = s_j(x_{kj}, x_{lj}) = 1 - \frac{|r_{kj} - r_{lj}|}{\max_k r_{kj} - \min_k r_{kj}}$$

样本相似度定义

$$s(\mathbf{x}_k, \mathbf{x}_l) = \frac{\sum_{j=1}^p s_j(x_{kj}, x_{lj}) \delta(x_{kj}, x_{lj}) w_j}{\sum_{j=1}^p \delta(x_{kj}, x_{lj}) w_j}$$

其中

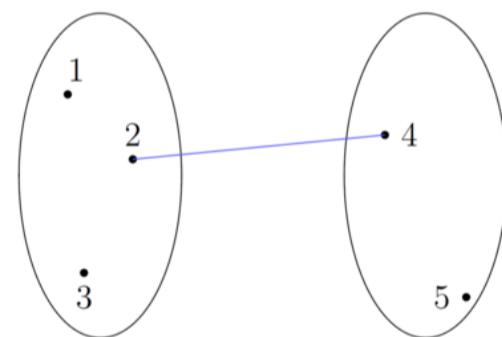
$$\delta(x_{kj}, x_{lj}) = \begin{cases} 0, & \text{如果 } x_{kj} \text{ 或 } x_{lj} \text{ 存在缺失观测} \\ 1, & \text{其他} \end{cases}$$

w_j 表示权重, 一般取值为1, 但是如果事先知道第 j 个特征尤其重要, 可以增加相应的权重。

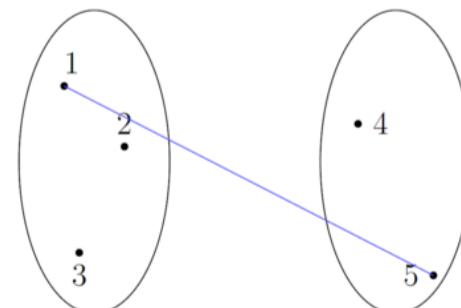
类间距离

举例

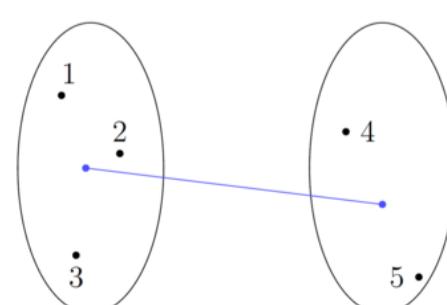
1. simple linkage



2. complete linkage



3. centroid linkage



我们可以统一的关联准则，这样就可以方便计算合并后两两族间距离。

$$d(k \cup l, i) = \alpha_k d(k, i) + \alpha_l d(l, i) + \beta d(k, l) + \gamma |d(k, i) - d(l, i)|$$

名称	α_k	α_l	β	γ
最短距离法 (single-linkage)	0.5	0.5	0	-0.5
最长距离法 (complete-linkage)	0.5	0.5	0	0.5
类平均法 UPGMA (average linkage)	$\frac{n_k}{n_k+n_l}$	$\frac{n_l}{n_k+n_l}$	0	0
加权类平均法 WPGMA (McQuitty 法)	0.5	0.5	0	0
中位数法 WPGMC (median linkage)	0.5	0.5	-0.25	0
中心法 UPGMC (centroid linkage)	$\frac{n_k}{n_k+n_l}$	$\frac{n_l}{n_k+n_l}$	$-\frac{n_k n_l}{(n_k+n_l)^2}$	0
Ward 法 (minimum variance)	$\frac{n_k+n_l}{n_k+n_l+n_i}$	$\frac{n_l+n_i}{n_k+n_l+n_i}$	$-\frac{n_i}{n_k+n_l+n_i}$	0

此方法可以很好的运用于自下而上的层次聚类中。

聚类方法

层次聚类

动机

试图在不同层次对数据集进行划分，从而形成树形的聚类结构。

形式

1. 自下而上：每个样本各自分到一个类中，之后将类间距离最近的两类关联，并建立一个新的类，反复此过程直到所有的样本聚合至一个类中。
2. 自上而下：将所有样本归到一个类中，之后将在类中相距最远的样本记为两个新的类，基于这两个类，将未进行聚类的点逐一比较其与两个新的类的距离，这样所有样本划分成了两类，在每一个类中重复此过程直到每个样本点各自分到一个类中。

我们通常采用最简单的形式，即自下而上，使用 simple linkage

自下而上实例

假设有四个点 $ABCD$ ，距离矩阵如下，使用 simple linkage

$$\begin{pmatrix} 0 & 1 & 3 & 2 \\ 1 & 0 & 5 & 6 \\ 3 & 5 & 0 & 4 \\ 2 & 6 & 4 & 0 \end{pmatrix}$$

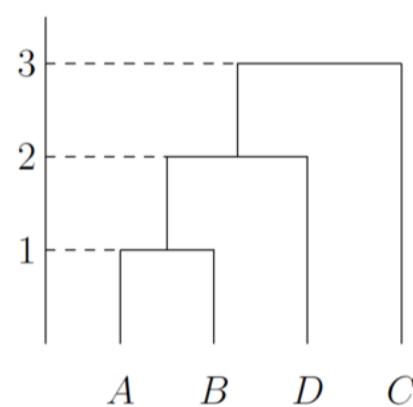
第一步，将 A 和 B 合并

$$\begin{pmatrix} 0 & 3 & 2 \\ 3 & 0 & 4 \\ 2 & 4 & 0 \end{pmatrix}$$

第二步，将 AB 和 D 合并

$$\begin{pmatrix} 0 & 3 \\ 3 & 0 \end{pmatrix}$$

最后聚成一类，结果



层次聚类结果

k 均值聚类

动机

To find groups in the data, with the number of groups represented by the variable K

一种快速聚类的算法，可以看成一种特殊的 GMM。

概述

输入

1. 聚类数目 K
2. 样本集 $X = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)'$

目标

给出一个划分，使得

$$\min W(\mathcal{C}) \quad (3)$$

$$s.t. \quad C_k \cap C_l = \emptyset, \quad \bigcup_{k=1}^K C_k = X \quad (4)$$

划分

$$\mathcal{C} = \{C_1, C_2, \dots, C_K\}$$

对应一个聚类结果，我们希望该划分**最优**，使得类内距离足够小，类间距离足够大。

我们通常采用**平方欧氏距离**

$$d_{kl} = d(k, l) = \|\mathbf{x}_k - \mathbf{x}_l\|_2^2 = \sum_{j=1}^p (x_{kj} - x_{lj})^2$$

由此，我们可以定义损失函数。

损失函数

$$W(\mathcal{C}) = \sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{x}_i - \mathbf{m}_k\|^2$$

其中 \mathbf{m}_k 表示第 k 类的均值或中心

实际上， k 均值聚类就是解决一个最优化问题：

$$\mathcal{C}^* = \arg \min_{\mathcal{C}} W(\mathcal{C})$$

由于是 NP-hard 问题，因此需要使用迭代法：

1. 确定 k 个类的中心 \mathbf{m}_k ，将样本逐一分配到其最近的中心所对应的类中，得到一个聚类结果。
2. 更新每个类的样本均值，作为类更新后的中心；重复此过程，直到收敛为止。

说明

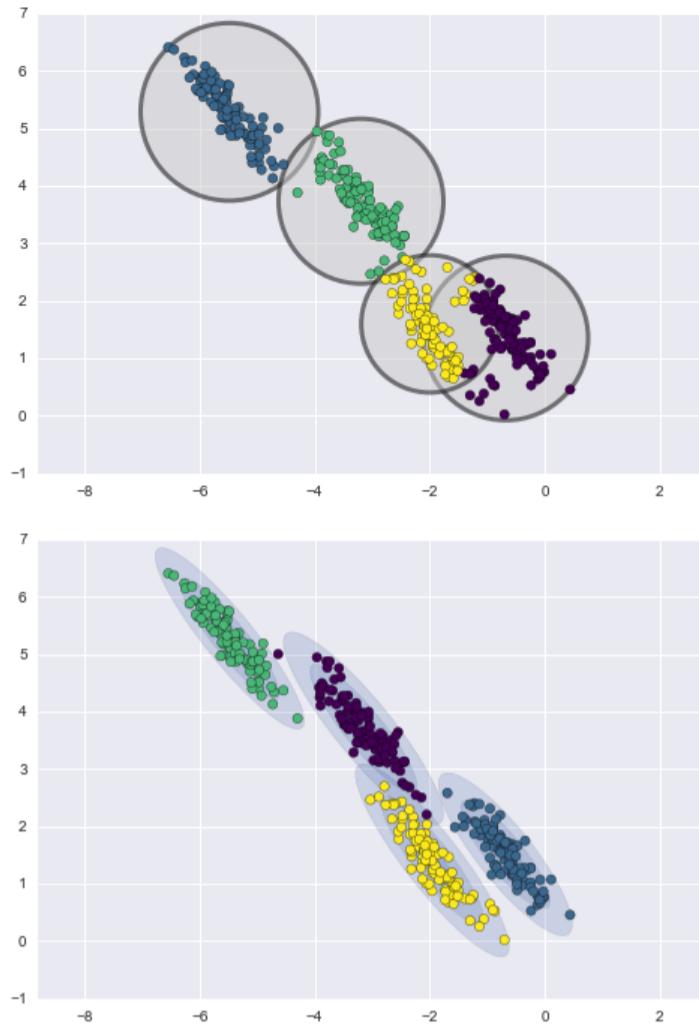
1. 收敛可以设置为聚类结果不变。
2. 复杂度是 $O(pnK)$ ，其中 p 表示特征个数， n 表示样本个数， K 是聚类数目。
3. 如果**非凸**， K 均值聚类算法**难以收敛**。
4. 有时候可以放宽收敛限制，比如设定一个较为宽松的阈值。

高斯混合模型 GMM

动机

修正了一些 k-means 的不足：

- 可以 fit 更加 oblong 的 cluster



- 从硬聚类调整到了软聚类

我们可以给出了一组评分，一般我们可以选择评分最大的作为其所属的类。但我们可以同时输出多个，增加正确的概率。

概述

核心：分布的正态性假定

假定第 i 个样本来自第 k 类正态分布

$$N_p(\boldsymbol{\mu}_k, \Sigma_k)$$

密度函数

$$f(\mathbf{x}_i) = (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right\}$$

当我们可以确定第 i 个样本是来自于第 k 个高斯分布总体时，我们可以构造变量

$$\delta_{ik} = \begin{cases} 1, & \text{当第 } i \text{ 个样本 } x_i \text{ 属于第 } k \text{ 个总体;} \\ 0, & \text{当第 } i \text{ 个样本 } x_i \text{ 不属于第 } k \text{ 个总体.} \end{cases}$$

于是 $\boldsymbol{\delta}_i = (\delta_{i1}, \delta_{i2}, \dots, \delta_{iK})'$ 满足

1. 独立同分布的随机向量
2. 服从多维分布 $M(1, \pi_1, \pi_2, \dots, \pi_K)$
3. $\pi_k = P(\delta_{ik} = 1)$ 且满足

$$0 < \pi_k < 1, \quad \sum_{i=1}^K \pi_k = 1$$

$\boldsymbol{\delta}_i = (\delta_{i1}, \delta_{i2}, \dots, \delta_{iK})'$ 的密度公式

$$f(\boldsymbol{\delta}_i) = \prod_{k=1}^K (\pi_k)^{\delta_{ik}}, i = 1, 2, \dots, n$$

给定 $\boldsymbol{\delta}_i$ 后, \mathbf{x}_i 的密度函数

$$f(\mathbf{x}_i | \boldsymbol{\delta}_i) = \prod_{k=1}^K \left((2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right\} \right)^{\delta_{ik}}$$

样本 $\{\mathbf{x}_i, \boldsymbol{\delta}_i\}$ 联合密度函数

$$\begin{aligned} & \prod_{i=1}^n f(\mathbf{x}_i, \boldsymbol{\delta}_i) \\ &= \prod_{i=1}^n f(\boldsymbol{\delta}_i) \cdot f(\mathbf{x}_i | \boldsymbol{\delta}_i) \\ &= \prod_{i=1}^n \prod_{k=1}^K \left(\pi_k (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right\} \right)^{\delta_{ik}} \end{aligned}$$

上式是未知参数

$$\boldsymbol{\theta} = (\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \Sigma_1, \dots, \Sigma_K)$$

的似然函数。

但实际上我们仅仅能够观测到样本 $\{\mathbf{x}_i\}, i = 1, 2, \dots, n$, 而无法观测到 $\boldsymbol{\delta}_i$

样本 \mathbf{x}_i 的密度函数为

$$f(x_i) = \sum_{k=1}^K \pi_k (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -\frac{1}{2} (x_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (x_i - \boldsymbol{\mu}_k) \right\}$$

这个密度函数是由 K 个正态分布的密度函数加权组合而成的, 常称为高斯混合模型。

前置知识

$$\frac{\partial \ln \det(X)}{\partial X} = X^{-1}$$

$$\frac{\partial \text{tr}(AX^{-1}B)}{\partial X} = - (X^{-1}BAX^{-1})'$$

EM 算法

变量 δ_{ik} 无法观测到, 于是将其作为潜变量。未知参数 $\boldsymbol{\theta}$ 的对数似然为

$$\begin{aligned}
l(\boldsymbol{\theta}) &= \ln L(\boldsymbol{\theta}) \\
&\propto -\frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \delta_{ik} \left((\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) + \ln |\Sigma_k| \right) + \sum_{i=1}^n \sum_{k=1}^K \delta_{ik} \ln (\pi_k) \\
&= Q_0(\boldsymbol{\theta})
\end{aligned}$$

E步

目的: 将潜变量 δ_{ik} 的期望 π_{ik}^* 代入 $Q_0(\boldsymbol{\theta})$

$$\begin{aligned}
Q_0(\boldsymbol{\theta}) &= -\frac{1}{2} \sum_{i=1}^n \sum_{k=1}^K \pi_{ik}^* \left((\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) + \ln |\Sigma_k| \right) + \sum_{i=1}^n \sum_{k=1}^K \pi_{ik}^* \ln (\pi_k) \\
&=: Q_1(\boldsymbol{\theta}) + Q_2(\boldsymbol{\theta})
\end{aligned}$$

其中

$$\pi_{ik}^* = E(\delta_{ik} | \mathbf{x}_i) = P(\delta_{ik} = 1 | \mathbf{x}_i) = \frac{\pi_k \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}{\sum_{k=1}^K \pi_k \phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k)}$$

$$\phi(\mathbf{x}_i; \boldsymbol{\mu}_k, \Sigma_k) = (2\pi)^{-p/2} |\Sigma_k|^{-1/2} \exp \left\{ -(\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) / 2 \right\}$$

M步

目的: 求 $Q_0(\boldsymbol{\theta})$ 的最大值而确定未知参数的估计。

我们发现

1. $Q_1(\boldsymbol{\theta})$ 仅和未知参数 $\{\boldsymbol{\mu}_k, \Sigma_k\}_{k=1}^K$ 有关
2. $Q_2(\boldsymbol{\theta})$ 仅和未知参数 $\{\pi_k\}_{k=1}^K$ 有关

于是我们可以分别确定最大值点。

先对 $Q_1(\boldsymbol{\theta})$ 分别对参数求导

$$\frac{\partial Q_1(\boldsymbol{\theta})}{\partial \boldsymbol{\mu}_k} = -\frac{1}{2} \sum_{i=1}^n \pi_{ik}^* \Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) = 0 \quad (5)$$

$$\frac{\partial Q_1(\boldsymbol{\theta})}{\partial \Sigma_k} = \sum_{i=1}^n \pi_{ik}^* \left(\Sigma_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)' \Sigma_k^{-1} + \Sigma_k^{-1} \right) = 0 \quad (6)$$

由此解得

$$\begin{aligned}
\boldsymbol{\mu}_k &= \frac{\sum_{i=1}^n \pi_{ik}^* \mathbf{x}_i}{\sum_{i=1}^n \pi_{ik}^*} \\
\Sigma_k &= \frac{\sum_{i=1}^n \pi_{ik}^* (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)'}{\sum_{i=1}^n \pi_{ik}^*}
\end{aligned}$$

再对 $Q_2(\boldsymbol{\theta})$ 求导, 注意到

$$\sum_{k=1}^K \pi_k = 1, \quad 0 < \pi_k < 1, k = 1, 2, \dots, K$$

运用拉格朗日乘子法

$$Q_2^*(\boldsymbol{\theta}) = \sum_{i=1}^n \sum_{k=1}^K \pi_{ik}^* \ln(\pi_k) - \lambda \left(\sum_{k=1}^K \pi_k - 1 \right)$$

求导，使得导函数为 0

$$\frac{\partial Q_2^*(\boldsymbol{\theta})}{\partial \pi_k} = \sum_{i=1}^n \frac{\pi_{ik}^*}{\pi_k} - \lambda = 0$$

解得

$$\pi_k = \frac{1}{n} \sum_{i=1}^n \pi_{ik}^*$$

1

循环 EM 两步直到收敛。

至此，EM 算法结束。关于其收敛性有如下结论：**EM 算法可以保证收敛到一个稳定点，但是却不能保证收敛到全局的极大值点，因此它是局部最优的算法。**

DBSCAN

动机

To separate clusters of high density from clusters of low density.

概念

1. ϵ 邻域：如果

$$N_\epsilon(k) = \{\mathbf{x}_l \in X \mid d(k, l) \leq \epsilon\}$$

则称 \mathbf{x}_k 的 ϵ 邻域。

2. 核心点：如果 $|N_\epsilon(k)| \geq \text{MinPts}$ ，那么，称 \mathbf{x}_k 为核心点。

3. 密度直达、密度可达和密度连接

a. 密度直达： $\mathbf{x}_l \in N_\epsilon(k)$ 且 $|N_\epsilon(k)| \geq \text{MinPts}$ ，那么称 \mathbf{x}_l 可以从点 \mathbf{x}_k 直接密度可达或密度直达。

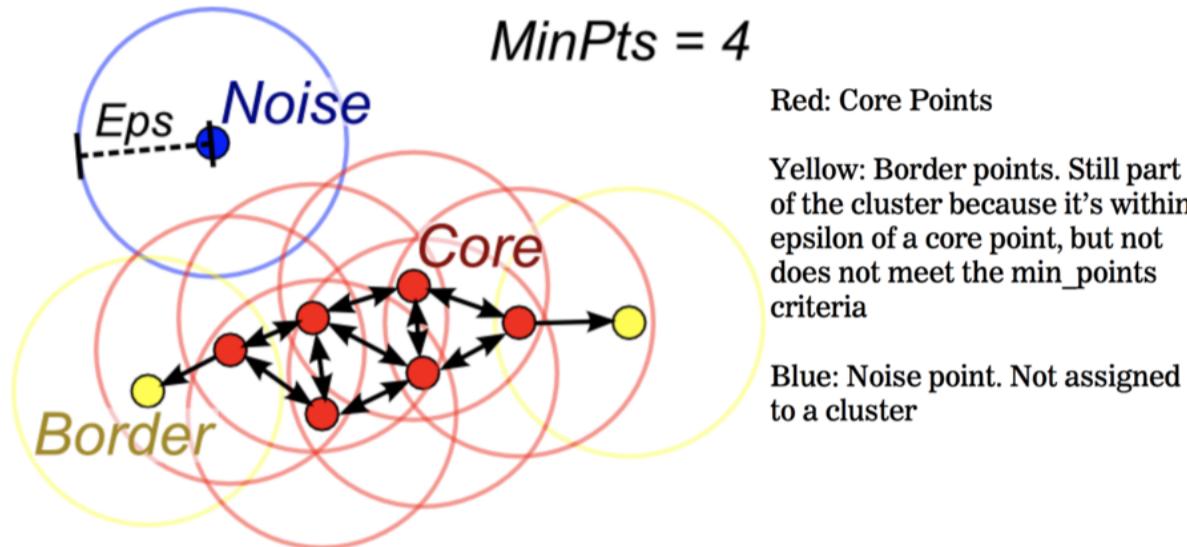
b. 密度可达：存在点列 $\mathbf{x}_{k0} = \mathbf{x}_k, \mathbf{x}_{k1}, \mathbf{x}_{k2}, \dots, \mathbf{x}_{kn} = \mathbf{x}_l \in X$ 使得 \mathbf{x}_{i+1} 可以从点 \mathbf{x}_i 直接密度可达，那么称 \mathbf{x}_l 可以从点 \mathbf{x}_k 密度可达。

c. 密度连接：存在一个点 \mathbf{x}_i 使得点 \mathbf{x}_k 和 \mathbf{x}_l 均从点 \mathbf{x}_i 密度可达，那么，称点 \mathbf{x}_k 和 \mathbf{x}_l 密度连接。

4. 类

- a. 最大性：如果任意两个点，一个在集合内，且两点可以密度可达，则另一个点属于该集合。
 - b. 连接性：集合内任意两个点都是密度连接的。
 - c. 核心点一定满足最大性和连接性。
5. 噪声：不属于任何一类的点。

例子



算法

Algorithm 5 DBSCAN 聚类算法

Require: 样本集 $X = (x_1, x_2, \dots, x_n)'$;
 邻域半径 ϵ ;
 可到达的最少样本个数 $MinPts$;

Ensure: DBSCAN 聚类的结果

- 1: 初始化：类别序号 $k = 0$;
- 2: 标记每个样本点为未被访问过 $v_i = 0, i = 1, 2, \dots, n$;
- 3: **for** 从数据集 X 中随机选取一个未访问过的样本点 x_i **do**
- 4: **if** 样本点 x_i 是未访问过的，即 $v_i = 0$ **then**
- 5: 标记点 x_i 被访问过，即 $v_i = 1$;
- 6: 计算 x_i 的 ϵ 邻域，即 $N_\epsilon(i) = \{x_l \in X : dist(i, l) \leq \epsilon\}$;
- 7: **if** 样本点 x_i 不是一个核心点，即 $|N_\epsilon(i)| < MinPts$ **then**
- 8: 样本点 x_i 是一个噪声;
- 9: **else**
- 10: $k = k + 1$;
- 11: 令 $G(i)$ 表示所有从样本点 x_i 密度可达的样本点，即 $G(i) = \{x_l \in X : \text{点 } x_l \text{ 可以从点 } x_i \text{ 密度可达}\}$;
- 12: **for** 样本点 $x_l \in G(i)$ **do**
- 13: **if** 样本点 x_l 是未访问过的，即 $v_l = 0$ **then**
- 14: 标记样本点 x_l 为被访问过，即 $v_l = 1$;
- 15: 将样本点 x_l 归入类 C_k 中;

聚类方法的评价

聚类有效性是评价聚类结果的方式

评价有内外区分：即是否使用外部信息（知道样本标签）

这里假定硬聚类（一个样本有且只有一个类），如果没有聚类成功的点，则把其归为一类

外部聚类有效性

概述

对于 n 个测试样本

$$\mathbf{x}_i (i = 1, 2, \dots, n)$$

假定分类结果为

$$\mathcal{C} = \{C_1, C_2, \dots, C_K\}$$

满足

$$C_k \cap C_l = \emptyset, \quad \bigcup_{k=1}^K C_i = X$$

真实分类结果为

$$\mathcal{P} = \{P_1, P_2, \dots, P_{K'}\}$$

满足

$$P_k \cap P_l = \emptyset, \quad \bigcup_{k=1}^{K'} P_i = X$$

可能性矩阵 (Contingency matrix)

	P_1	P_2	\dots	$P_{K'}$	求和
C_1	n_{11}	n_{12}	\dots	$n_{1K'}$	$n_{1\cdot}$
C_2	n_{21}	n_{22}	\dots	$n_{2K'}$	$n_{2\cdot}$
\vdots	\vdots	\vdots		\vdots	\vdots
C_K	n_{K1}	n_{K2}	\dots	$n_{KK'}$	$n_{K\cdot}$
求和	$n_{\cdot 1}$	$n_{\cdot 2}$	\dots	$n_{\cdot K'}$	n

可以计算

$$p_{ij} = \frac{n_{ij}}{n}, \quad p_i = \frac{n_{i\cdot}}{n}, \quad p_j = \frac{n_{\cdot j}}{n}$$

注意：该矩阵进行行变换指标不变，因为我把第一类分到第二类，第二类分到第一类是没有问题的，因为我们只评价聚类有效性。

熵 (Entropy, E)

$$E = - \sum_i p_i \left(\sum_j \frac{p_{ij}}{p_i} \ln \frac{p_{ij}}{p_i} \right)$$

纯度 (Purity, P)

$$P = \sum_i p_i \left(\max_j \frac{p_{ij}}{p_i} \right)$$

内部聚类有效性

紧密度 (compactness) : 同一类内不同个体之间紧密关联的度量。

1. 方差可以提现数据的紧密度，低方差代表紧密度好。
2. 可以依赖距离。

区分度 (separation) : 不同类间区别程度的度量。

1. 两个类中心的距离, 或者取两类最短距离。
2. 依赖密度。

常见指标

均方标准差

$$RMSSTD = \left(\frac{\sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{x}_i - \mathbf{m}_k\|^2}{p \sum_{k=1}^K (n_k - 1)} \right)^{1/2}$$

R 平方 (RS)

$$RS = 1 - \frac{\sum_{k=1}^K \sum_{i \in C_k} \|\mathbf{x}_i - \mathbf{m}_k\|^2}{\sum_{i=1}^n \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2}$$

参数选择

轮廓法 (Silhouette)

基本思想: 同类相似, 异类不同。

我们首先可以定义轮廓值。

轮廓值可以衡量一个样本相对其他类中的样本而言与本类样本的相似度, 取值范围为 $[-1, 1]$ 。我们可以认为轮廓值较高表示该样本被很好地聚到其所属的类, 而不和其他类相似。如果大部分的样本具有较高的轮廓值, 那么聚类的结果是恰当的, 反之则不合适。

我们一样假定分类结果为

$$\mathcal{C} = \{C_1, C_2, \dots, C_K\}$$

对于第 i 个样本, 不妨假定其属于第 k 类, n_k 为第 k 类样本量。我们可以定义 $a(i)$ 为第 i 个样本与同类其他样本的平均距离

$$a(i) = \frac{1}{n_k - 1} \sum_{j \in C_k, j \neq i} \text{dist}(i, j)$$

注意如果类只有一个样本, 则 $a(i) = 0$ 。

对于第 i 个样本和另一个样本量为 $n_{k'}$ 的类 $\mathcal{C}_{k'}$, 我们可以令 $d(i, \mathcal{C}_{k'})$ 为第 i 个样本与第 k' 类所有样本的平均距离

$$d(i, \mathcal{C}_{k'}) = \frac{1}{n_{k'}} \sum_{j \in \mathcal{C}_{k'}} \text{dist}(i, j)$$

我们定义 $b(i)$ 为第 i 个样本与不属于同一类的所有样本的最近平均距离

$$b(i) = \min_{k' \neq k} d(i, \mathcal{C}_{k'})$$

于是可以定义轮廓值 $s(i)$

$$s(i) = \frac{b(i) - a(i)}{\max(a(i) - b(i))}$$

那么，对于所有 n 个样本，可以计算**平均轮廓值**，用来度量聚类数目 K 是否合适。

由此，以**最大的平均轮廓值所对应的 K** 作为最优聚类数目，这种方法称为**轮廓法**。

CH (Calinski-Harabasz) 指数

与方差分析中的 F 检验统计量类似。

我们一样假定分类结果为

$$\mathcal{C} = \{C_1, C_2, \dots, C_K\}$$

我们分别考虑类间平方和 $B(K)$ 和类内平方和 $W(K)$ ，我们有

$$\text{CH}(K) = \frac{B(K)/(K-1)}{W(K)/(n-K)}$$

其中

$$W(K) = \sum_{k=1}^K \sum_{i \in \mathcal{C}_k} \|\mathbf{x}_i - \bar{\mathbf{x}}_k\|^2, \quad \bar{\mathbf{x}}_k = n_k^{-1} \sum_{i \in \mathcal{C}_k} \mathbf{x}_i$$
$$B(K) = \sum_{k=1}^K n_k \|\bar{\mathbf{x}}_k - \bar{\mathbf{x}}\|^2, \quad \bar{\mathbf{x}} = K^{-1} \sum_{k=1}^K \bar{\mathbf{x}}_k$$

最优聚类数目

$$k_{opt} = \arg \max_k \text{CH}(K)$$

附录

www.turingfinance.com
<http://www.turingfinance.com/clustering-countries-real-gdp-growth-part2/#quality>



李航统计学习方法

主要讨论第一篇：监督学习

 **统计学习方法概论**

 **感知机**

 **KNN**

 **朴素贝叶斯**

 **决策树**

 **logistic回归和最大熵模型**

 **SVM**

 **提升方法**

 **EM算法**

 **隐马尔可夫模型**

感知机

两种：原始与对偶版本都要会。

算法 2.1 (感知机学习算法的原始形式)

输入：训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, 其中 $x_i \in \mathcal{X} = \mathbf{R}^n$, $y_i \in \mathcal{Y} = \{-1, +1\}$, $i = 1, 2, \dots, N$; 学习率 η ($0 < \eta \leq 1$);

输出： w, b ; 感知机模型 $f(x) = \text{sign}(w \cdot x + b)$ 。

(1) 选取初值 w_0, b_0 ;

(2) 在训练集中选取数据 (x_i, y_i) ;

(3) 如果 $y_i(w \cdot x_i + b) \leq 0$,

$$w \leftarrow w + \eta y_i x_i$$

$$b \leftarrow b + \eta y_i$$

(4) 转至 (2), 直至训练集中没有误分类点。 ■

算法 2.2 (感知机学习算法的对偶形式)

输入: 线性可分的数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, 其中 $x_i \in \mathbf{R}^n$, $y_i \in \{-1, +1\}$, $i = 1, 2, \dots, N$; 学习率 η ($0 < \eta \leq 1$);

输出: α, b ; 感知机模型 $f(x) = \text{sign} \left(\sum_{j=1}^N \alpha_j y_j x_j \cdot x + b \right)$, 其中 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)^T$ 。

(1) $\alpha \leftarrow 0, b \leftarrow 0$;

(2) 在训练集中选取数据 (x_i, y_i) ;

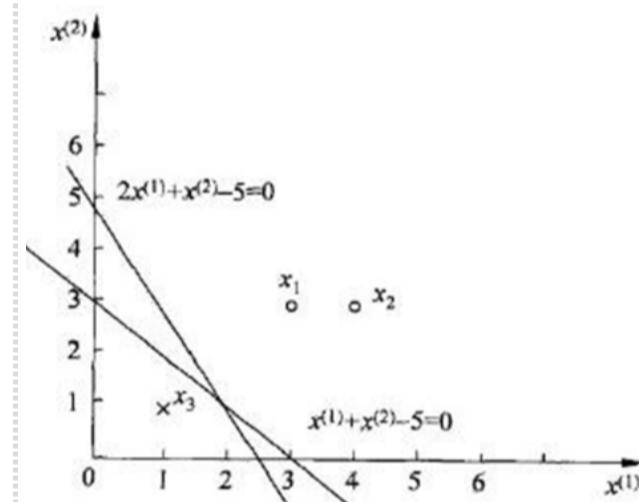
(3) 如果 $y_i \left(\sum_{j=1}^N \alpha_j y_j x_j \cdot x_i + b \right) \leq 0$,

$$\alpha_i \leftarrow \alpha_i + \eta$$

$$b \leftarrow b + \eta y_i$$

(4) 转至 (2) 直到没有误分类数据。 ■

例子



$$x_1 = (3, 3)^T, \quad x_2 = (4, 3)^T, \quad x_3 = (1, 1)^T$$

取超参数 $\eta = 1$

取初值 $w_0 = 0, b_0 = 0$

第一轮:

1. 选择 $x_1 = (3, 3)^T, y_1 (w_0 \cdot x_1 + b_0) = 0$, 没有被正确分类, 更新参数。

a. $w_1 = w_0 + y_1 x_1 = (3, 3)^T$

b. $b_1 = b_0 + y_1 = 1$

c. 线性模型 $w_1 \cdot x + b_1 = 3x^{(1)} + 3x^{(2)} + 1$

2. 选择 $x_2 = (4, 3)^T, y_2 (w_1 \cdot x_2 + b_1) > 0$, 被正确分类。

3. 选择 $x_3 = (1, 1)^T, y_3 (w_1 \cdot x_3 + b_1) < 0$, 没有被正确分类, 更新参数。

a. $w_2 = w_1 + y_3 x_3 = (2, 2)^T$

b. $b_2 = b_1 + y_3 = 0$

c. $w_2 \cdot x + b_2 = 2x^{(1)} + 2x^{(2)}$

如此循环, 具体计算结果如下表所示

迭代次数	误分类点	w	b	$w \cdot x + b$
0		0	0	0
1	x_1	$(3, 3)^T$	1	$3x^{(1)} + 3x^{(2)} + 1$
2	x_3	$(2, 2)^T$	0	$2x^{(1)} + 2x^{(2)}$
3	x_3	$(1, 1)^T$	-1	$x^{(1)} + x^{(2)} - 1$
4	x_3	$(0, 0)^T$	-2	-2
5	x_2	$(3, 3)^T$	-1	$3x^{(1)} + 3x^{(2)} - 1$
6	x_3	$(2, 2)^T$	-2	$2x^{(1)} + 2x^{(2)} - 2$
7	x_3	$(1, 1)^T$	-3	$x^{(1)} + x^{(2)} - 3$
8	0	$(1, 1)^T$	-3	$x^{(1)} + x^{(2)} - 3$

朴素贝叶斯

MLE 和 MAP 估计都要会。

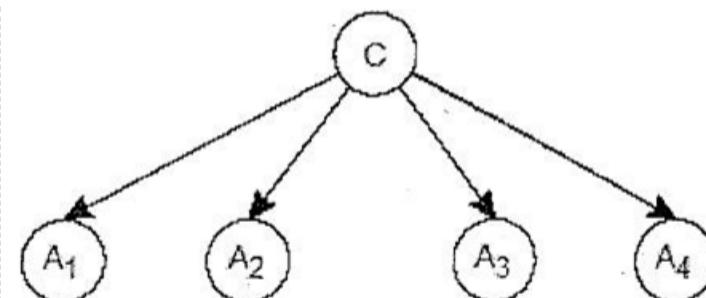
例子

给定数据集

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

给定条件

$\langle \text{Outlook} = \text{sunny}, \text{Temperature} = \text{cool}, \text{Humidity} = \text{high}, \text{Wind} = \text{strong} \rangle$



预计计算

$$c(x) = \arg \max_{c \in \{\text{yes, no}\}} P(c) P(\text{sunny} | c) P(\text{cool} | c) P(\text{high} | c) P(\text{strong} | c)$$

计算: (这里 $\lambda = 1$)

$$P(yes) = (9+1)/(14+2) = 10/16 \quad (1)$$

$$P(sunny \mid yes) = (2+1)/(9+3) = 3/12 \quad (2)$$

$$P(cool \mid yes) = (3+1)/(9+3) = 4/12 \quad (3)$$

$$P(high \mid yes) = (3+1)/(9+2) = 4/11 \quad (4)$$

$$P(strong \mid yes) = (3+1)/(9+2) = 4/11 \quad (5)$$

$$P(no) = (5+1)/(14+2) = 6/16 \quad (6)$$

$$P(sunny \mid no) = (3+1)/(5+3) = 4/8 \quad (7)$$

$$P(cool \mid no) = (1+1)/(5+3) = 2/8 \quad (8)$$

$$P(high \mid no) = (4+1)/(5+2) = 5/7 \quad (9)$$

$$P(strong \mid no) = (3+1)/(5+2) = 4/7 \quad (10)$$

$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = 0.0069$$

$$P(no)P(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = 0.0191$$

因此选 no。

支持向量机

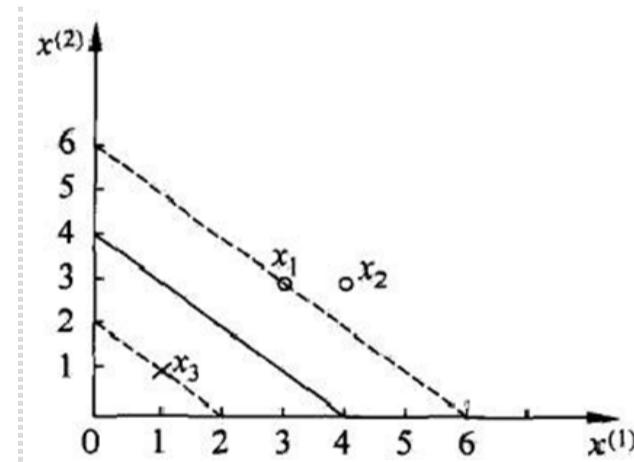
(线性可分) 二分类的 SVM 原始与对偶形式都要会。

原始形式最优化问题

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (11)$$

$$\text{s.t. } y_i (w \cdot x_i + b) - 1 \geq 0, \quad i = 1, 2, \dots, N \quad (12)$$

例子 1



$$\min_{w,b} \frac{1}{2} (w_1^2 + w_2^2) \quad (13)$$

$$\text{s.t. } 3w_1 + 3w_2 + b \geq 1 \quad (14)$$

$$4w_1 + 3w_2 + b \geq 1 \quad (15)$$

$$-w_1 - w_2 - b \geq 1 \quad (16)$$

$$w_1 = w_2 = \frac{1}{2}, \quad b = -2$$

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

其中

$$x_1 = (3, 3)^T \quad x_3 = (1, 1)^T$$

对偶问题

$$\min_{\alpha} \quad \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (17)$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (18)$$

$$\alpha_i \geq 0, \quad i = 1, 2, \dots, N \quad (19)$$

设对偶问题的解为

$$\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_l^*)^T$$

则原始问题的解为

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i \quad (20)$$

$$b^* = y_j - \sum_{i=1}^N \alpha_i^* y_i (x_i \cdot x_j) \quad (21)$$

例子 2

$$\min_{\alpha} \quad \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (22)$$

$$= \frac{1}{2} (18\alpha_1^2 + 25\alpha_2^2 + 2\alpha_3^2 + 42\alpha_1\alpha_2 - 12\alpha_1\alpha_3 - 14\alpha_2\alpha_3) - \alpha_1 - \alpha_2 - \alpha_3 \quad (23)$$

$$\text{s.t.} \quad \alpha_1 + \alpha_2 - \alpha_3 = 0 \quad (24)$$

$$\alpha_i \geq 0, \quad i = 1, 2, 3 \quad (25)$$

带入约束

$$s(\alpha_1, \alpha_2) = 4\alpha_1^2 + \frac{13}{2}\alpha_2^2 + 10\alpha_1\alpha_2 - 2\alpha_1 - 2\alpha_2$$

求条件极值，注意在偏导数为 0 的地方不满足约束，因此在边界上取得

$$s\left(0, \frac{2}{13}\right) = -\frac{2}{13}$$

$$s\left(\frac{1}{4}, 0\right) = -\frac{1}{4}$$

因此最小值在 $(\frac{1}{4}, 0)$ 取得，由公式

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i \quad (26)$$

$$b^* = y_j - \sum_{i=1}^N \alpha_i^* y_i (x_i \cdot x_j) \quad (27)$$

计算可得

$$\begin{aligned} w_1^* &= w_2^* = \frac{1}{2} \\ b^* &= -2 \end{aligned}$$

分离超平面

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

分类决策函数

$$f(x) = \text{sign} \left(\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 \right)$$

隐马尔可夫模型

例子 1

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = (0.2, 0.4, 0.4)^T$$

$$O = (\text{红}, \text{白}, \text{红})$$

计算初值

$$\begin{aligned} \alpha_1(1) &= \pi_1 b_1(o_1) = 0.10 \\ \alpha_1(2) &= \pi_2 b_2(o_1) = 0.16 \\ \alpha_1(3) &= \pi_3 b_3(o_1) = 0.28 \end{aligned}$$

递推计算

$$\begin{aligned}
\alpha_2(1) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i1} \right] b_1(o_2) = 0.154 \times 0.5 = 0.077 \\
\alpha_2(2) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i2} \right] b_2(o_2) = 0.184 \times 0.6 = 0.1104 \\
\alpha_2(3) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i3} \right] b_3(o_2) = 0.202 \times 0.3 = 0.0606 \\
\alpha_3(1) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i1} \right] b_1(o_3) = 0.04187 \\
\alpha_3(2) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i2} \right] b_2(o_3) = 0.03551 \\
\alpha_3(3) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i3} \right] b_3(o_3) = 0.05284
\end{aligned}$$

终止

例子 2

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = (0.2, 0.4, 0.4)^T$$

$$O = (\text{红}, \text{白}, \text{红})$$

求最优路径

1. 初始化: 在 $t = 1$ 时, 对每一个状态 i , $i = 1, 2, 3$, 求状态 i 观测 o_1 为红的概率, 记为: $\delta_1(i)$

$$\delta_1(i) = \pi_i b_i(o_1) = \pi_i b_i(\text{红}), i = 1, 2, 3$$

$$\delta_1(1) = 0.10, \quad \delta_1(2) = 0.16, \quad \delta_1(3) = 0.28$$

$$\psi_1(i) = 0, i = 1, 2, 3$$

2. 在 $t = 2$ 时, 对每一个状态 i , $i = 1, 2, 3$, 求在 $t = 1$ 时状态为 j 观测 o_1 为红并在 $t = 2$ 时状态为 i 观测 O_2 为白的路径的最大概率, 记为: $\delta_2(i)$

$$\begin{aligned}
\delta_2(1) &= \max_{1 \leq j \leq 3} [\delta_1(j) a_{j1}] b_1(o_2) \\
&= \max_j \{0.10 \times 0.5, 0.16 \times 0.3, 0.28 \times 0.2\} \times 0.5 \\
&= 0.028
\end{aligned}$$

$$\begin{aligned}
\psi_2(1) &= 3 \\
\delta_2(2) &= 0.0504, \quad \psi_2(2) = 3 \\
\delta_2(3) &= 0.042, \quad \psi_2(3) = 3
\end{aligned}$$

3. 同样 $t = 3$ 时

$$\delta_3(i) = \max_{1 \leq j \leq 3} [\delta_2(j)a_{ji}] b_i(o_3)$$

$$\psi_3(i) = \arg \max_{1 \leq j \leq 3} [\delta_2(j)a_{ji}]$$

$$\delta_3(1) = 0.00756, \quad \psi_3(1) = 2$$

$$\delta_3(2) = 0.01008, \quad \psi_3(2) = 2$$

$$\delta_3(3) = 0.0147, \quad \psi_3(3) = 3$$

4. 最优路径选择

$$P^* = \max_{1 \leq i \leq 3} \delta_3(i) = 0.0147$$

终点为

$$i_3^* = \arg \max_i [\delta_3(i)] = 3$$

依次反向寻找

在 $t = 2$ 时, $i_2^* = \psi_3(i_3^*) = \psi_3(3) = 3$
在 $t = 1$ 时, $i_1^* = \psi_2(i_2^*) = \psi_2(3) = 3$

因此, 最优路径为

$$I^* = (i_1^*, i_2^*, i_3^*) = (3, 3, 3)$$



统计学习方法概论

统计学习

统计学习的对象

data：计算机及互联网上的各种数字、文字、图像、视频、音频数据以及它们的组合。

数据的基本假设是**同类数据具有一定的统计规律性**。

统计学习的目的

用于对数据（特别是未知数据）进行**预测和分析**。

分类：

1. **Supervised learning** (这门课我们主要考虑监督学习)
2. Unsupervised learning
3. Semi-supervised learning
4. Reinforcement learning

监督学习：

1. 训练数据 training data
2. 模型 model —— 假设空间 hypothesis
3. 评价准则 evaluation criterion —— 策略 strategy
4. 算法 algorithm

统计学习的研究：

1. 统计学习方法
2. 统计学习理论（统计学习方法的有效性和效率和基本理论）
3. 统计学习应用

监督学习

概念：Instance, feature vector, feature space

输入实例 x 的特征向量：

$$x = \left(x^{(1)}, x^{(2)}, \dots, x^{(i)}, \dots, x^{(n)} \right)^T$$

$x^{(i)}$ 与 x_i 不同, 后者表示多个输入变量中的第 i 个

$$x_i = \left(x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(n)} \right)^T$$

训练集:

$$T = \{ (x_1, y_1), (x_2, y_2), \dots, (x_N, y_N) \}$$

输入变量和输出变量: 分类问题、回归问题、标注问题

联合概率分布

假设输入与输出的随机变量 X 和 Y 遵循联合概率分布 $P(X, Y)$

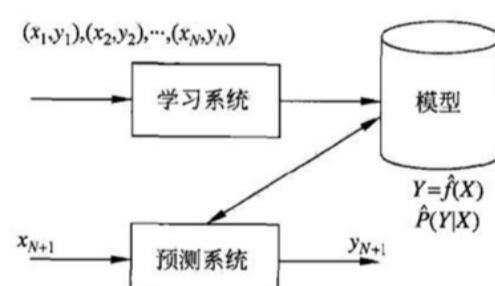
1. $P(X, Y)$ 为分布函数或分布密度函数
2. 对于学习系统来说, 联合概率分布是未知的,
3. 训练数据和测试数据被看作是依联合概率分布 $P(X, Y)$ 独立同分布产生的。

假设空间

监督学习目的是学习一个由输入到输出的映射, 称为模型

模式的集合就是**假设空间** (hypothesis space)

概率模型: 条件概率分布 $P(Y|X)$, 决策函数: $Y = f(X)$



$$y_{N+1} = \arg \max_{y_{N+1}} \hat{P}(y_{N+1} | x_{N+1})$$

$$y_{N+1} = \hat{f}(x_{N+1})$$

统计学习三要素

方法 = 模型 + 策略 + 算法

模型: 假设空间

决策函数的集合

$$\mathcal{F} = \{f \mid Y = f(X)\}$$

参数空间

$$\mathcal{F} = \{f \mid Y = f_\theta(X), \theta \in \mathbf{R}^n\}$$

条件概率的集合

$$\mathcal{F} = \{P \mid P(Y \mid X)\}$$

参数空间

$$\mathcal{F} = \{P \mid P_\theta(Y \mid X), \theta \in \mathbf{R}^n\}$$

策略 1：损失函数

损失函数：一次预测的好坏

风险函数：平均意义下模型预测的坏

0-1 损失函数 0-1 loss function

$$L(Y, f(X)) = \begin{cases} 1, & Y \neq f(X) \\ 0, & Y = f(X) \end{cases}$$

平方损失函数 quadratic loss function

$$L(Y, f(X)) = (Y - f(X))^2$$

绝对损失函数 absolute loss function

$$L(Y, f(X)) = |Y - f(X)|$$

对数损失函数 logarithmic loss function 或对数似然损失函数 loglikelihood loss function

$$L(Y, P(Y \mid X)) = -\log P(Y \mid X)$$

损失函数的期望

$$R_{\text{exp}}(f) = E_P[L(Y, f(X))] = \int_{\mathcal{X} \times \mathcal{Y}} L(y, f(x)) P(x, y) dx dy$$

风险函数 risk function 给出期望损失 expected loss

由 $P(x, y)$ 可以直接求出 $P(x|y)$ ，但不知道。

经验风险 empirical risk 给出经验损失 empirical loss

$$R_{\text{emp}}(f) = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i))$$

策略 2：经验风险最小化与结构风险最小化

经验风险最小化最优模型

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i))$$

当样本容量很小时，经验风险最小化学习的效果未必很好，会产生“过拟合 over-fitting。”

结构风险最小化 structure risk minimization，为防止过拟合提出的策略，等价于正则化 (regularization)，加入正则化项 regularizer，或罚项 penalty term：

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f)$$

算法

如果最优化问题有显式的解析式，算法比较简单

但通常解析式不存在，就需要数值计算的方法

模型评估与模型选择

训练误差，训练数据集的平均损失

$$R_{\text{emp}}(\hat{f}) = \frac{1}{N} \sum_{i=1}^N L(y_i, \hat{f}(x_i))$$

测试误差，测试数据集的平均损失

$$e_{\text{test}} = \frac{1}{N'} \sum_{i=1}^{N'} L(y_i, \hat{f}(x_i))$$

举例：损失函数是 0-1 损失时

$$e_{\text{test}} = \frac{1}{N'} \sum_{i=1}^{N'} I(y_i \neq \hat{f}(x_i))$$

测试数据集准确率

$$r_{\text{test}} = \frac{1}{N'} \sum_{i=1}^{N'} I(y_i = \hat{f}(x_i))$$

可以发现

$$r_{\text{test}} + e_{\text{test}} = 1$$

过拟合与模型选择

假设给定训练数据集

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

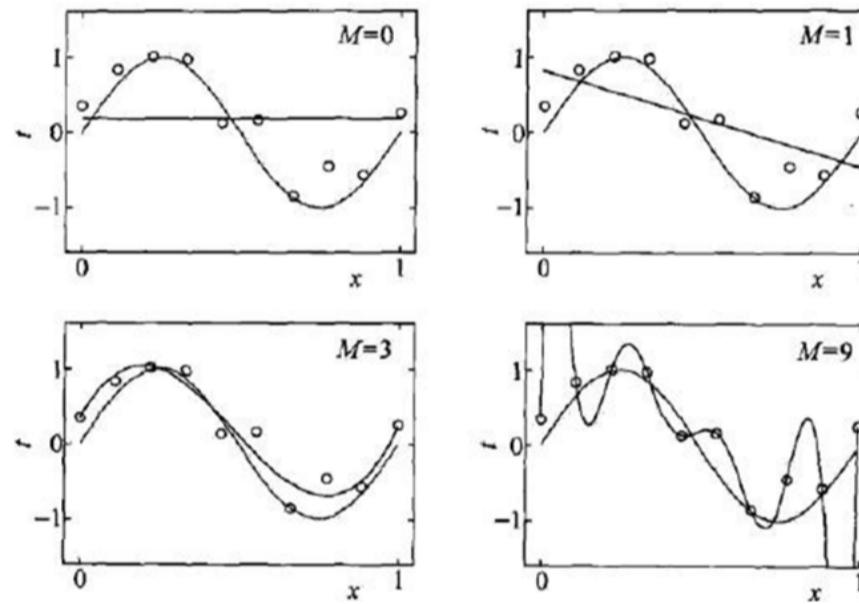
$$f_M(x, w) = w_0 + w_1 x + w_2 x^2 + \dots + w_M x^M = \sum_{j=0}^M w_j x^j$$

我们可以让**经验风险最小**

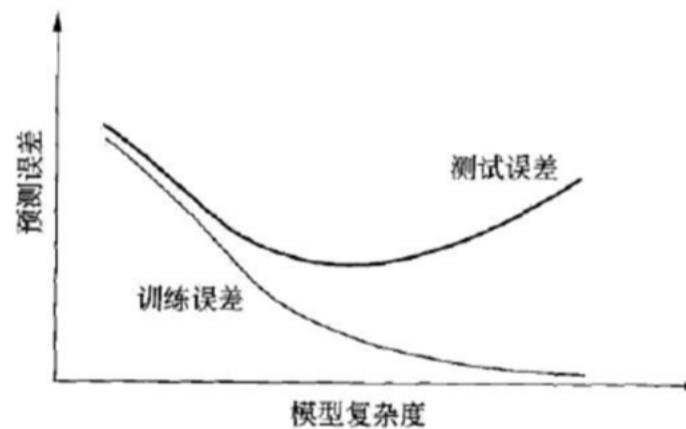
$$L(w) = \frac{1}{2} \sum_{i=1}^N (f(x_i, w) - y_i)^2 = \frac{1}{2} \sum_{i=1}^N \left(\sum_{j=0}^M w_j x_i^j - y_i \right)^2$$

$$w_j = \frac{\sum_{i=1}^N x_i y_i}{\sum_{i=1}^N x_i^{j+1}}, \quad j = 0, 1, 2, \dots, M$$

我们看到**训练误差**随着 M 增大变小，而**测试误差**先减小后增大。



因此，我们有



正则化与交叉验证

正则化一般形式

$$\min_{f \in \mathcal{F}} \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i)) + \lambda J(f)$$

常见回归问题中

$$L(w) = \frac{1}{N} \sum_{i=1}^N (f(x_i; w) - y_i)^2 + \frac{\lambda}{2} \|w\|^2$$

$$L(w) = \frac{1}{N} \sum_{i=1}^N (f(x_i; w) - y_i)^2 + \lambda \|w\|_1$$

数据划分:

1. 训练集 training set: 用于训练模型
2. 验证集 validation set: 用于模型选择
3. 测试集 test set: 用于最终对学习方法的评估

交叉验证 Cross Validation:

1. 简单交叉验证
2. K 折交叉验证 K-Fold Cross Validation
3. 留一交叉验证

泛化能力

泛化误差 generalization error

$$R_{\text{exp}}(\hat{f}) = E_P[L(Y, \hat{f}(X))] = \int_{\mathcal{X} \times \mathcal{Y}} L(y, \hat{f}(x)) P(x, y) dx dy$$

泛化误差上界

比较学习方法的泛化能力——比较泛化误差上界

结论

1. 样本容量增加, 泛化误差趋于 0
2. 假设空间容量越大, 泛化误差越大

二分类问题

$$X \in \mathbf{R}^n, \quad Y \in \{-1, +1\}$$

期望风险和经验风险

$$R(f) = E[L(Y, f(X))]$$

$$\hat{R}(f) = \frac{1}{N} \sum_{i=1}^N L(y_i, f(x_i))$$

经验风险最小化函数：

$$f_N = \arg \min_{f \in \mathcal{F}} \hat{R}(f)$$

定理 (泛化误差上界, 二分类问题) : 当假设空间是有限个函数的结合

$\mathcal{F} = \{f_1, f_2, \dots, f_d\}$, 对任意一个函数 f , 至少以概率 $1 - \delta$, 以下不等式成立:

$$R(f) \leq \hat{R}(f) + \varepsilon(d, N, \delta)$$

$$\varepsilon(d, N, \delta) = \sqrt{\frac{1}{2N} \left(\log d + \log \frac{1}{\delta} \right)}$$

生成模型与判别模型

监督学习的目的就是学习一个模型:

决策函数:

$$Y = f(X)$$

条件概率分布:

$$P(Y | X)$$

生成方法 Generative approach 对应生成模型: generative model

$$P(Y | X) = \frac{P(X, Y)}{P(X)}$$

应用

朴素贝叶斯法和隐马尔科夫模型

判别方法由数据直接学习决策函数 $f(X)$ 或条件概率分布 $P(Y | X)$ 作为预测的模型, 即判别模型。 **Discriminative approach 对应 discriminative model**

应用

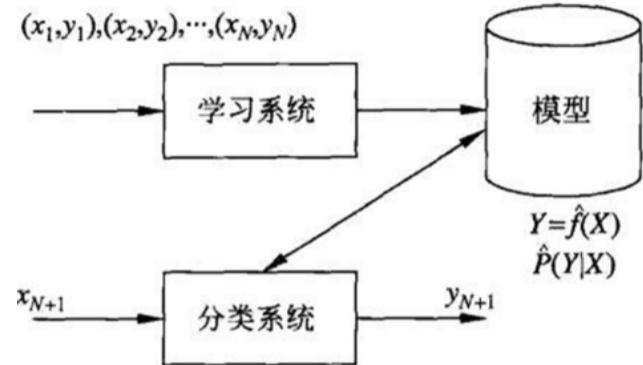
K 近邻法、感知机、决策树、logistic 回归模型、最大熵模型、支持向量机、提升方法和条件随机场。

各自优缺点:

生成方法: 可还原出联合概率分布 $P(X, Y)$, 而判别方法不能。生成方法的收敛速度更快, 当样本容量增加的时候, 学到的模型可以更快地收敛于真实模型; 当存在隐变量时, 仍可以使用生成方法, 而判别方法则不能用。

判别方法: 直接学习到条件概率或决策函数, 直接进行预测, 往往学习的准确率更高; 由于直接学习 $Y = f(X)$ 或 $P(Y|X)$, 可对数据进行各种程度上的抽象、定义特征并使用特征, 因此可以简化学习过程。

分类模型



一般的评价指标 (以混淆矩阵为基础的) 在算法书里讲过了。

分类问题

ROC: (receiver operating characteristic curve)

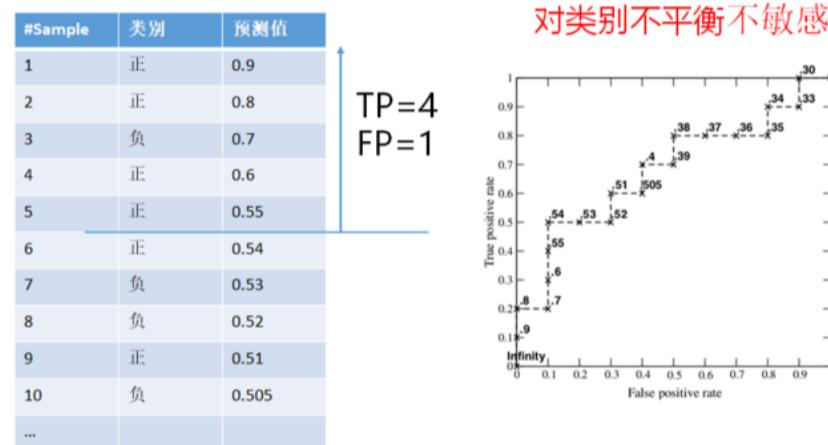
受试者工作特征曲线

AUC: Area Under the Curve

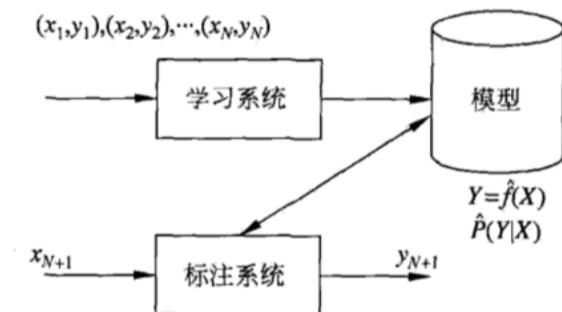
二分类

实值输出

对类别不平衡不敏感



标注模型



标注: tagging, 结构预测: structure prediction

输入: 观测序列, 输出: 标记序列或状态序列

学习和标注两个过程

训练集:

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

观测序列:

$$x_i = (x_i^{(1)}, x_i^{(2)}, \dots, x_i^{(n)})^T, i = 1, 2, \dots, N$$

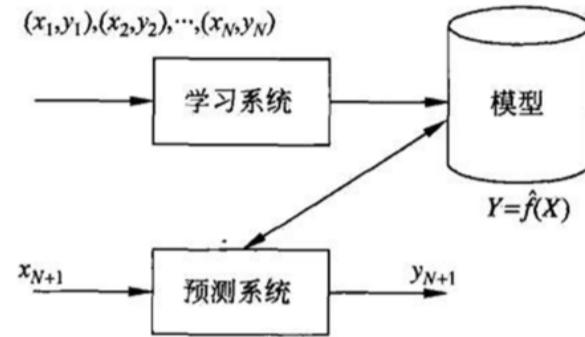
输出标记序列：

$$y_i = \left(y_i^{(1)}, y_i^{(2)}, \dots, y_i^{(n)} \right)^T$$

模型：条件概率分布

$$P \left(Y^{(1)}, Y^{(2)}, \dots, Y^{(n)} \mid X^{(1)}, X^{(2)}, \dots, X^{(n)} \right)$$

回归模型



回归模型是表示从输入变量到输出变量之间映射的函数

回归问题的学习等价于函数拟合

学习和预测两个阶段

一般使用**最小二乘**解决



感知机

模型

定义

输入空间是 $\mathcal{X} \subseteq \mathbf{R}^n$ ，输出空间是 $\mathcal{Y} = \{+1, -1\}$

输入 $x \in \mathcal{X}$ 表示实例的特征向量，对应于输入空间的点，输出 $y \in \mathcal{Y}$ 表示实例的类别，由输入空间到输出空间的函数

$$f(x) = \text{sign}(w \cdot x + b)$$

称为感知机。

模型参数：

1. 权值向量 $w \in \mathbf{R}^n$
2. 偏置 $b \in \mathbf{R}$

符号函数

$$\text{sign}(x) = \begin{cases} +1, & x \geq 0 \\ -1, & x < 0 \end{cases}$$

几何解释

线性方程

$$w \cdot x + b = 0$$

称为分离超平面。

学习策略

假设数据集是完全线性可分的，有两种自然的想法定义损失：

1. 误分类点的数目：不好求解。
2. 误分类点到超平面的总距离：比较好求解。

点到直线的距离：

$$d = \frac{|w \cdot x_0 + b|}{\|w\|}$$

误分类点

$$-y_i (w \cdot x_i + b) > 0$$

误分类点距离

$$-\frac{1}{\|w\|} y_i (w \cdot x_i + b)$$

总距离

$$-\frac{1}{\|w\|} \sum_{x_i \in M} y_i (w \cdot x_i + b)$$

因此定义损失函数

$$L(w, b) = - \sum_{x_i \in M} y_i (w \cdot x_i + b)$$

其中 M 为误分类点的集合。这个损失函数就是感知机学习的经验风险函数。

学习算法

$$\min_{w, b} L(w, b) = - \sum_{x_i \in M} y_i (w \cdot x_i + b)$$

首先任意选择一个超平面 w_0, b_0 ，然后通过随机梯度下降法极小化目标函数

$$\nabla_w L(w, b) = - \sum_{x \in M} y_i x_j \quad w \leftarrow w + \eta y_i x_i$$

$$\nabla_b L(w, b) = - \sum_{x \in M} y_s \quad b \leftarrow b + \eta y_i$$

直到训练集中没有误分类点。

算法 2.1 (感知机学习算法的原始形式)

输入: 训练数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, 其中 $x_i \in \mathcal{X} = \mathbf{R}^n$, $y_i \in \mathcal{Y} = \{-1, +1\}$, $i = 1, 2, \dots, N$; 学习率 η ($0 < \eta \leq 1$);

输出: w, b ; 感知机模型 $f(x) = \text{sign}(w \cdot x + b)$ 。

(1) 选取初值 w_0, b_0 ;

(2) 在训练集中选取数据 (x_i, y_i) ;

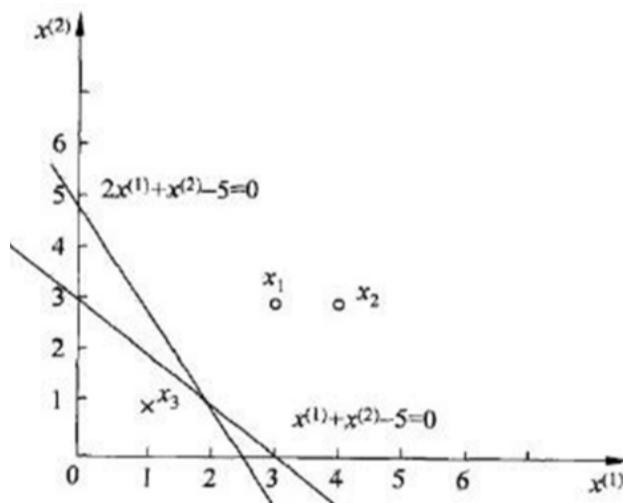
(3) 如果 $y_i(w \cdot x_i + b) \leq 0$,

$$w \leftarrow w + \eta y_i x_i$$

$$b \leftarrow b + \eta y_i$$

(4) 转至 (2), 直至训练集中没有误分类点。 ■

例子



$$x_1 = (3, 3)^T, \quad x_2 = (4, 3)^T, \quad x_3 = (1, 1)^T$$

取超参数 $\eta = 1$

取初值 $w_0 = 0$, $b_0 = 0$

第一轮:

1. 选择 $x_1 = (3, 3)^T$, $y_1 (w_0 \cdot x_1 + b_0) = 0$, 没有被正确分类, 更新参数。

a. $w_1 = w_0 + y_1 x_1 = (3, 3)^T$

b. $b_1 = b_0 + y_1 = 1$

c. 线性模型 $w_1 \cdot x + b_1 = 3x^{(1)} + 3x^{(2)} + 1$

2. 选择 $x_2 = (4, 3)^T$, $y_2 (w_1 \cdot x_2 + b_1) > 0$, 被正确分类。

3. 选择 $x_3 = (1, 1)^T$, $y_3 (w_1 \cdot x_3 + b_1) < 0$, 没有被正确分类, 更新参数。

a. $w_2 = w_1 + y_3 x_3 = (2, 2)^T$

b. $b_2 = b_1 + y_3 = 0$

c. 线性模型 $w_2 \cdot x + b_2 = 2x^{(1)} + 2x^{(2)}$

如此循环, 具体计算结果如下表所示

迭代次数	误分类点	w	b	$w \cdot x + b$
0		0	0	0
1	x_1	$(3, 3)^T$	1	$3x^{(1)} + 3x^{(2)} + 1$
2	x_3	$(2, 2)^T$	0	$2x^{(1)} + 2x^{(2)}$
3	x_3	$(1, 1)^T$	-1	$x^{(1)} + x^{(2)} - 1$
4	x_3	$(0, 0)^T$	-2	-2
5	x_2	$(3, 3)^T$	-1	$3x^{(1)} + 3x^{(2)} - 1$
6	x_3	$(2, 2)^T$	-2	$2x^{(1)} + 2x^{(2)} - 2$
7	x_3	$(1, 1)^T$	-3	$x^{(1)} + x^{(2)} - 3$
8	0	$(1, 1)^T$	-3	$x^{(1)} + x^{(2)} - 3$

收敛性

假设数据集是线性可分[1]的，则有

存在满足条件的 $\|\hat{w}_{\text{opt}}\| = 1$ 的超平面 $\hat{w}_{\text{opt}} \cdot \hat{x} = w_{\text{opt}} \cdot x + b_{\text{opt}} = 0$ ，且存在有 $\gamma > 0$

$$y_i (\hat{w}_{\text{opt}} \cdot \hat{x}_i) = y_i (w_{\text{opt}} \cdot x_i + b_{\text{opt}}) \geq \gamma$$

令 $R = \max_{1 \leq i \leq N} \|\hat{x}_i\|$ 感知机算法在训练集的误分类次数 k ，满足如下不等式

$$k \leq \left(\frac{R}{\gamma} \right)^2$$

[1] 数据集线性可分的充要条件是每个类别的凸包均不相交。

定理表明：

1. 误分类的次数 k 是有上界的，当训练数据集线性可分时，感知机学习算法原始形式迭代是收敛的。
2. 感知机算法存在许多解，既依赖于初值，也依赖迭代过程中误分类点的选择顺序。
3. 为得到唯一分离超平面，需要增加约束，如 SVM。
4. 线性不可分数据集，迭代震荡。

对偶形式

不失一般性，设初值为 0，则有：

$$\begin{aligned} w &\leftarrow w + \eta y_i x_i \quad \rightarrow \quad w = \sum_{i=1}^N \alpha_i y_i x_i \\ b &\leftarrow b + \eta y_i \quad \rightarrow \quad b = \sum_{i=1}^N \alpha_i y_i \end{aligned}$$

因此有对偶算法

算法 2.2 (感知机学习算法的对偶形式)

1

输入: 线性可分的数据集 $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, 其中 $x_i \in \mathbf{R}^n$, $y_i \in \{-1, +1\}$, $i = 1, 2, \dots, N$; 学习率 η ($0 < \eta \leq 1$);

输出: α, b ; 感知机模型 $f(x) = \text{sign} \left(\sum_{j=1}^N \alpha_j y_j x_j \cdot x + b \right)$, 其中 $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N)^T$ 。

(1) $\alpha \leftarrow 0, b \leftarrow 0$;

(2) 在训练集中选取数据 (x_i, y_i) ;

(3) 如果 $y_i \left(\sum_{j=1}^N \alpha_j y_j x_j \cdot x_i + b \right) \leq 0$,

$$\alpha_i \leftarrow \alpha_i + \eta$$

$$b \leftarrow b + \eta y_i$$

(4) 转至 (2) 直到没有误分类数据。 ■

第三步需要计算训练实例的内积, 因此可以先计算样本的 Gram 矩阵

$$G = [x_i \cdot x_j]_{N \times N}$$



KNN

KNN 算法

原理

在视觉里面有，这里简单介绍一下。

工作原理

1. 存在一个样本数据集合，也称作训练样本集，并且样本集中每个数据都存在标签，即我们知道样本集中每个数据与所属分类的对应关系。
2. 输入没有标签的新数据后，将新数据的每个特征与样本集中数据对应的特征进行比较，然后算法提取样本集中特征最相似数据（最近邻）的分类标签。
3. 一般来说，只选择样本数据集中前 N 个最相似的数据。 K 一般不大于 20，最后，选择 k 个中出现次数最多的分类，作为新数据的分类。

特点

优点

1. 精度高
2. 对异常值不敏感
3. 无输入数据假定

缺点

1. 时空复杂度高

使用数据范围

1. 数值型
2. 标称型

一般流程

收集数据：可以使用任何方法。

准备数据：距离计算所需要的数值，最后是结构化的数据格式。

分析数据：可以使用任何方法。

没有显式训练

测试算法：计算错误率。

使用算法：首先需要输入样本数据和结构化的输出结果，然后运行 k -近邻算法判定输入数据分别属于哪个分类，最后应用对计算出的分类执行后续的处理。

KNN model

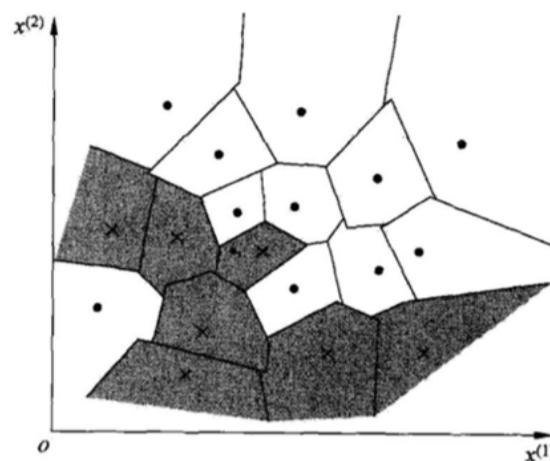


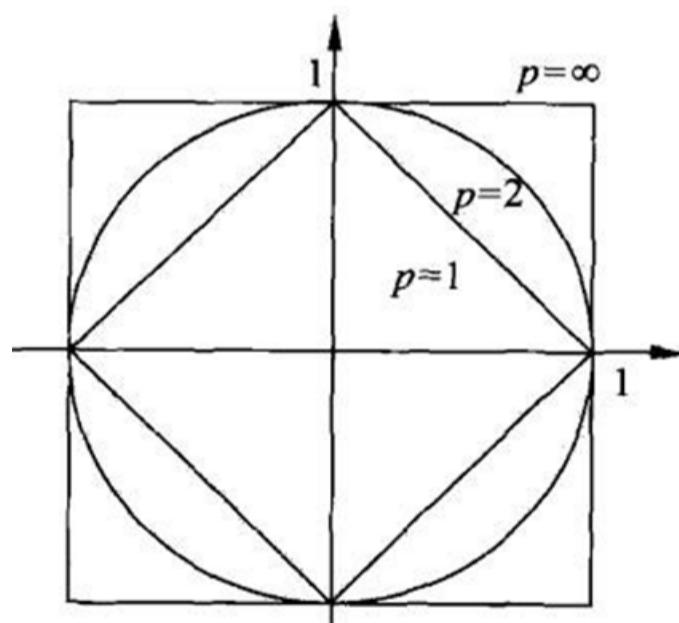
图 3.1 k 近邻法的模型对应特征空间的一个划分

这里 $K=1$ ，一共有两类。

距离度量

见统计方法中的聚类。一般使用 p 范数作为距离度量。

$$L_p(x_i, x_j) = \left(\sum_{l=1}^n |x_i^{(l)} - x_j^{(l)}|^p \right)^{\frac{1}{p}}$$



k 值选择

视觉里也写好了 QAQ。

1. 如果选择较小的 K 值

- a. “学习”的近似误差 (approximation error) 会减小，但 “学习”的估计误差 (estimation error) 会增大。
- b. 噪声敏感： K 值的减小就意味着整体模型变得复杂，容易发生过拟合。

2. 如果选择较大的 K 值

a. 减少学习的估计误差，但缺点是学习的近似误差会增大。

b. K 值的增大就意味着整体的模型**变得简单**。

注：近似误差指的是**偏差** (bias)，估计误差指的是**方差** (variance)。近似误差随着假设空间增大而减小，估计误差随着假设空间增大而增大。

k 值越小 \Leftrightarrow 单个样本影响越大 \Leftrightarrow 模型越复杂 \Leftrightarrow 假设空间越大
 \Rightarrow 近似误差越小(估计误差越大)，容易过拟合

k 值越大 \Leftrightarrow 单个样本影响越小 \Leftrightarrow 模型越简单 \Leftrightarrow 假设空间越小
 \Rightarrow 近似误差越大(估计误差越小)，容易欠拟合。

分类决策规则

多数表决规则 (经验风险最小化)

1. 分类函数: $f: \mathbf{R}^n \rightarrow \{c_1, c_2, \dots, c_K\}$

2. 损失函数: 0-1 损失

误判率:

$$P(Y \neq f(X)) = 1 - P(Y = f(X))$$

$$\frac{1}{k} \sum_{x_i \in N_k(x)} I(y_i \neq c_j) = 1 - \frac{1}{k} \sum_{x_i \in N_k(x)} I(y_i = c_j)$$

KNN 实现: kd 树

结构

kd (k-dimensional) 树的概念自 1975 年提出，试图解决的是在 k 维空间为数据集建立索引的问题。已知样本空间如何快速查询得到其近邻？唯有以空间换时间，**建立索引**便是计算机世界的解决之道。但是索引建立的方式各有不同，kd 树只是其中一种。它的思想如同分治法，即：**利用已有数据对 k 维空间进行切分**。

kd 树是二叉树，表示对 K 维空间的一个划分 (partition)。构造 Kd 树相当于不断地用垂直于坐标轴的超平面将 K 维空间切分，构成一系列的 K 维超矩形区域。Kd 树的每个结点对应于一个 K 维超矩形区域。

例子

$$T = \{(2, 3)^T, (5, 4)^T, (9, 6)^T, (4, 7)^T, (8, 1)^T, (7, 2)^T\}$$

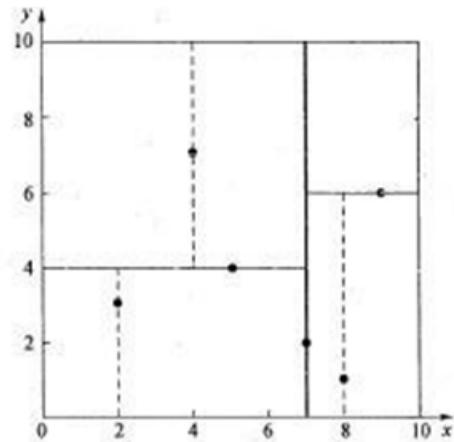
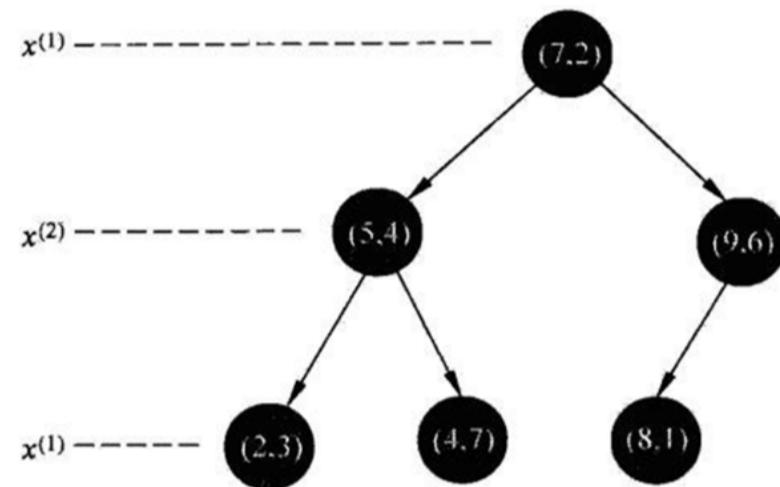
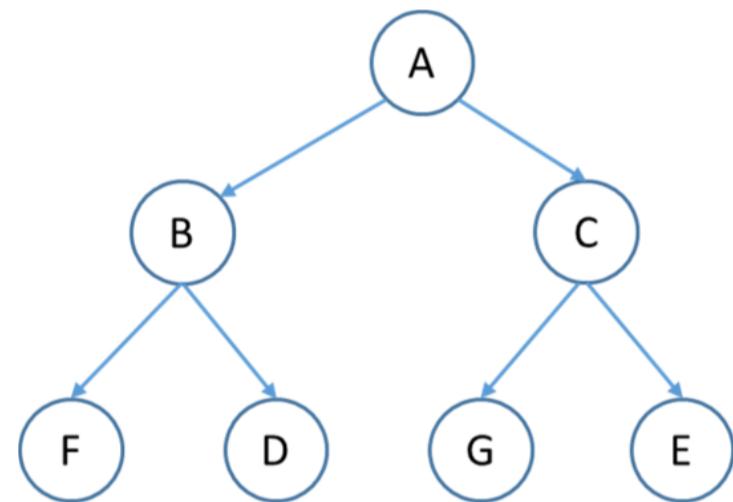
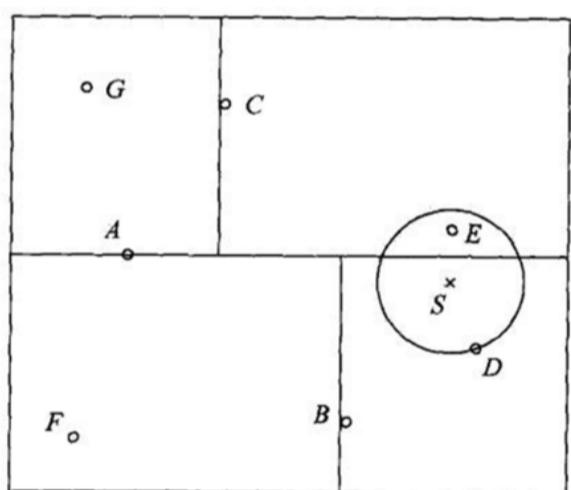


图1 二维数据k-d树空间划分示意图

每次选择中位数，知道每个 partition 上都有一个样本点。然后可以建立索引



搜索





朴素贝叶斯

学习与分类

基本方法

训练数据集

$$T = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$$

由联合概率分布产生 (i.i.d.)

朴素贝叶斯通过训练数据集学习联合概率分布

先验概率分布

$$P(Y = c_k), \quad k = 1, 2, \dots, K$$

条件概率分布：

$$P(X = x \mid Y = c_k) = P(X^{(1)} = x^{(1)}, \dots, X^{(n)} = x^{(n)} \mid Y = c_k), \quad k = 1, 2, \dots, K$$

注意：条件概率为指数级别的参数：

$$K \prod_{j=1}^n S_j$$

因此，我们需要条件独立性假设

$$\begin{aligned} P(X = x \mid Y = c_k) &= P(X^{(1)} = x^{(1)}, \dots, X^{(n)} = x^{(n)} \mid Y = c_k) \\ &= \prod_{j=1}^n P(X^{(j)} = x^{(j)} \mid Y = c_k) \end{aligned}$$

“朴素”贝叶斯名字由来，牺牲分类准确性。

注意：这里我们只需要估计这些参数：

$$K \sum_{j=1}^n S_j$$

贝叶斯定理

$$P(Y = c_k \mid X = x) = \frac{P(X = x \mid Y = c_k) P(Y = c_k)}{\sum_k P(X = x \mid Y = c_k) P(Y = c_k)}$$

代入上式，有

$$P(Y = c_k \mid X = x) = \frac{P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)}{\sum_k P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)}$$

贝叶斯分类器

$$y = f(x) = \arg \max_{c_k} \frac{P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)}{\sum_k P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)}$$

分母对所有 c_k 都相同，因此直接扔掉

$$y = \arg \max_{c_k} P(Y = c_k) \prod_j P(X^{(j)} = x^{(j)} \mid Y = c_k)$$

后验概率最大化的含义

朴素贝叶斯法将实例分到后验概率最大的类中，等价于期望风险最小化。（统计学习方法的目标）

假设选择 0-1 损失函数

期望风险函数

$$R_{\text{exp}}(f) = E[L(Y, f(X))]$$

注意二重积分可以拆成累次积分

$$\int [L(Y, f(X)) p(y \mid x) dy] p(x) dx$$

因此，可以取条件期望

$$R_{\text{exp}}(f) = E_X \sum_{k=1}^K [L(c_k, f(X))] P(c_k \mid X)$$

只需要对每个 x 逐个极小化

$$\begin{aligned} f(x) &= \arg \min_{y \in \mathcal{Y}} \sum_{k=1}^K L(c_k, y) P(c_k \mid X = x) \\ &= \arg \min_{y \in \mathcal{Y}} \sum_{k=1}^K P(y \neq c_k \mid X = x) \\ &= \arg \min_{y \in \mathcal{Y}} (1 - P(y = c_k \mid X = x)) \\ &= \arg \max_{y \in \mathcal{Y}} P(y = c_k \mid X = x) \end{aligned}$$

其中第三个等号成立的原因是：

推导出后验概率最大化准则

$$f(x) = \arg \max_{c_k} P(c_k \mid X = x)$$

问题

- 朴素贝叶斯的假设太强：条件独立性。因此实际上通常效果不好。
- 由于现实数据采样率不高和分布不均，因此估计概率效果不好。

参数估计

极大似然估计

可以使用 MLE 估计相应的概率：

先验概率的极大似然估计是：

$$P(Y = c_k) = \frac{\sum_{i=1}^N I(y_i = c_k)}{N}, k = 1, 2, \dots, K$$

设第 j 个特征 $x^{(j)}$ 可能取值的集合为 $\{a_{j1}, a_{j2}, \dots, a_{jS_j}\}$

条件概率的 MLE $j = 1, 2, \dots, n; l = 1, 2, \dots, S_j; k = 1, 2, \dots, K$

$$P(X^{(j)} = a_{jl} \mid Y = c_k) = \frac{\sum_{i=1}^N I(x_i^{(j)} = a_{jl}, y_i = c_k)}{\sum_{i=1}^N I(y_i = c_k)}$$

算法描述

输入：

- 训练数据集
- 第 i 个样本的第 j 个特征
- 第 j 个特征可能取的第 l 个值

输出： x 的分类

- 计算先验概率和条件概率

$$P(Y = c_k) = \frac{\sum_{i=1}^N I(y_i = c_k)}{N}, \quad k = 1, 2, \dots, K$$
$$P(X^{(j)} = a_{jl} \mid Y = c_k) = \frac{\sum_{i=1}^N I(x_i^{(j)} = a_{jl}, y_i = c_k)}{\sum_{i=1}^N I(y_i = c_k)}$$
$$j = 1, 2, \dots, n; \quad l = 1, 2, \dots, S_j; \quad k = 1, 2, \dots, K$$

- 对于给定的实例，计算

$$P(Y = c_k) \prod_{j=1}^n P\left(X^{(j)} = x^{(j)} \mid Y = c_k\right), \quad k = 1, 2, \dots, K$$

3. 确定 x 的分类

$$y = \arg \max_{c_k} P(Y = c_k) \prod_{j=1}^n P\left(X^{(j)} = x^{(j)} \mid Y = c_k\right)$$

贝叶斯估计

使用 MLE 导致有些时候出现概率极端的情况（由于样本不充分）因此采用贝叶斯估计（引入先验概率）

$$P_\lambda\left(X^{(j)} = a_{jt} \mid Y = c_k\right) = \frac{\sum_{i=1}^N I\left(x_i^{(j)} = a_{jt}, y_i = c_k\right) + \lambda}{\sum_{i=1}^N I(y_i = c_k) + S_j \lambda}$$

先验概率的贝叶斯估计

$$P_\lambda(Y = c_k) = \frac{\sum_{i=1}^N I(y_i = c_k) + \lambda}{N + K\lambda}$$

当 $\lambda = 1$ 时，为拉普拉斯平滑。

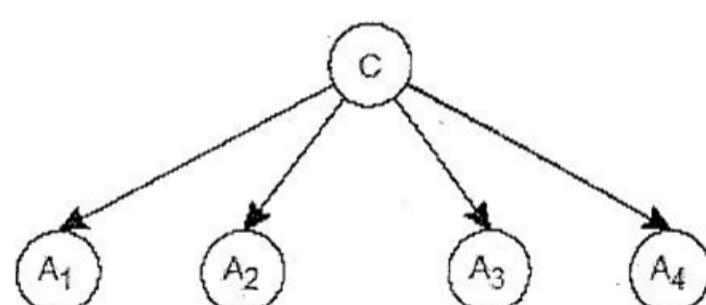
例子

给定数据集

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

给定条件

$\langle \text{Outlook} = \text{sunny}, \text{Temperature} = \text{cool}, \text{Humidity} = \text{high}, \text{Wind} = \text{strong} \rangle$



预计算

$$c(x) = \arg \max_{c \in \{\text{yes, no}\}} P(c)P(\text{sunny} \mid c)P(\text{cool} \mid c)P(\text{high} \mid c)P(\text{strong} \mid c)$$

计算: (这里 $\lambda = 1$)

$$P(\text{yes}) = (9 + 1)/(14 + 2) = 10/16 \quad (1)$$

$$P(\text{sunny} \mid \text{yes}) = (2 + 1)/(9 + 3) = 3/12 \quad (2)$$

$$P(\text{cool} \mid \text{yes}) = (3 + 1)/(9 + 3) = 4/12 \quad (3)$$

$$P(\text{high} \mid \text{yes}) = (3 + 1)/(9 + 2) = 4/11 \quad (4)$$

$$P(\text{strong} \mid \text{yes}) = (3 + 1)/(9 + 2) = 4/11 \quad (5)$$

$$P(\text{no}) = (5 + 1)/(14 + 2) = 6/16 \quad (6)$$

$$P(\text{sunny} \mid \text{no}) = (3 + 1)/(5 + 3) = 4/8 \quad (7)$$

$$P(\text{cool} \mid \text{no}) = (1 + 1)/(5 + 3) = 2/8 \quad (8)$$

$$P(\text{high} \mid \text{no}) = (4 + 1)/(5 + 2) = 5/7 \quad (9)$$

$$P(\text{strong} \mid \text{no}) = (3 + 1)/(5 + 2) = 4/7 \quad (10)$$

$$P(\text{yes})P(\text{sunny} \mid \text{yes})P(\text{cool} \mid \text{yes})P(\text{high} \mid \text{yes})P(\text{strong} \mid \text{yes}) = 0.0069$$

$$P(\text{no})P(\text{sunny} \mid \text{no})P(\text{cool} \mid \text{no})P(\text{high} \mid \text{no})P(\text{strong} \mid \text{no}) = 0.0191$$

因此选 no。



⑤ 决策树

决策树模型与学习

例子

分类模型的一般方法

- 1、**模型构建（归纳）** 通过对训练集合的归纳，建立分类模型。
- 2、**预测应用（推论）** 根据建立的分类模型，对测试集合进行测试。

模型

决策树是一种典型的分类方法

1. 首先对数据进行处理，利用归纳算法生成可读的规则和决策树，
2. 然后使用决策对新数据进行分析。

本质上决策树是通过一系列规则**对数据进行分类**的过程。

优点：

1. 推理过程**容易理解**，决策推理过程可以表示成 If Then 形式；
2. 推理过程完全依赖于**属性变量的取值特点**；
3. 可自动忽略目标变量没有贡献的属性变量，也为判断属性变量的重要性，减少变量的数目提供参考。 (**特征选择**)

和归纳算法

1. 决策树技术发现数据模式和规则的核心是**归纳算法**。
2. 归纳是**从特殊到一般**的过程。
3. 归纳推理从若干个事实中表征出的特征、特性和属性中，通过比较、总结、概括而得出一个**规律性的结论**。
4. 归纳推理论试图从对象的一部分或整体的特定的观察中获得一个完备且正确的描述。即从特殊事实到普遍性规律的结论。
5. 归纳对于认识的发展和完善具有重要的意义。人类知识的增长主要来源于归纳学习。
6. 归纳学习由于依赖于**检验数据**，因此又称为**检验学习**。
7. 归纳学习存在一个基本的假设：**假设如果能够在足够大的训练样本集中很好的逼近目标函数，则它也能在未见样本中很好地逼近目标函数**。该假定是归纳学习的有效性的前提条件。
8. 归纳过程就是在描述空间中进行搜索的过程。

- a. 归纳可分为**自顶向下**，**自底向上**和**双向搜索**三种方式。
- b. 自底向上法一次处理一个输入对象。将描述逐步一般化。直到最终的一般化描述。
- c. 自顶向下法对可能的一般性描述集进行搜索，试图找到一些满足一定要求的最优的描述。

从机器学习看分类及归纳等问题

从特殊的训练样例中归纳出一般函数是机器学习的中心问题；

从训练样例中进行学习通常被视为归纳推理。

1. 每个例子都是一个对偶（序偶） $(x, f(x))$ ，对每个输入的 x ，都有确定的输出 $f(x)$ 。
2. 学习过程将产生对目标函数 f 的不同逼近。 f 的每一个逼近都叫做一个**假设**。
3. 假设需要以某种形式表示。例如， $y = ax + b$ 。通过调整假设的表示，学习过程将产生出假设的不同变形。
4. 在表示中通常需要修改参数（如 a, b ）。
5. 从这些不同的变形中选择最佳的假设（或者说权值集合）。

方法的定义：使训练值与假设值-预测出的值之间的误差平方和 E 最小为最佳。

$$E = \sum_{x \in \text{Trainsample}} \left(V_{\text{train}}(x) - \hat{V}(x) \right)^2$$

算法

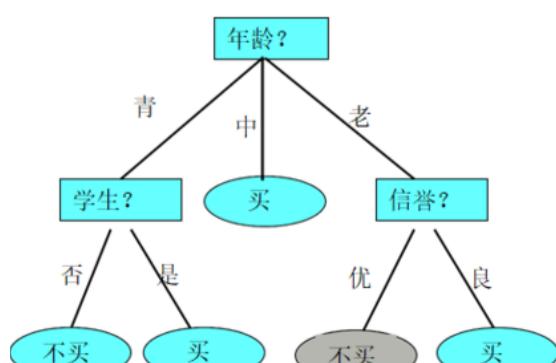
与决策树相关的重要算法包括：CLS, ID3, C4.5, CART

算法的发展过程

- Hunt, Marin 和 Stone 于 1966 年研制的 CLS 学习系统，用于学习单个概念。
- 1979 年 J.R. Quinlan 给出 ID3 算法，并在 1983 年和 1986 年对 ID3 进行了总结和简化，使其成为决策树学习算法的典型。
- Schlimmer 和 Fisher 于 1986 年对 ID3 进行改造，在每个可能的决策树节点创建缓冲区，使决策树可以递增式生成，得到 ID4 算法。
- 1988 年，Utgoff 在 ID4 基础上提出了 ID5 学习算法，进一步提高了效率。
- 1993 年，Quinlan 进一步发展了 ID3 算法，改进成 C4.5 算法。
- 另一类决策树算法为 CART，与 C4.5 不同的是，CART 的决策树由二元逻辑问题生成，每个树节点只有两个分枝，分别包括学习实例的正例与反例。

表示

决策树的基本组成部分：决策结点、分支和叶子。



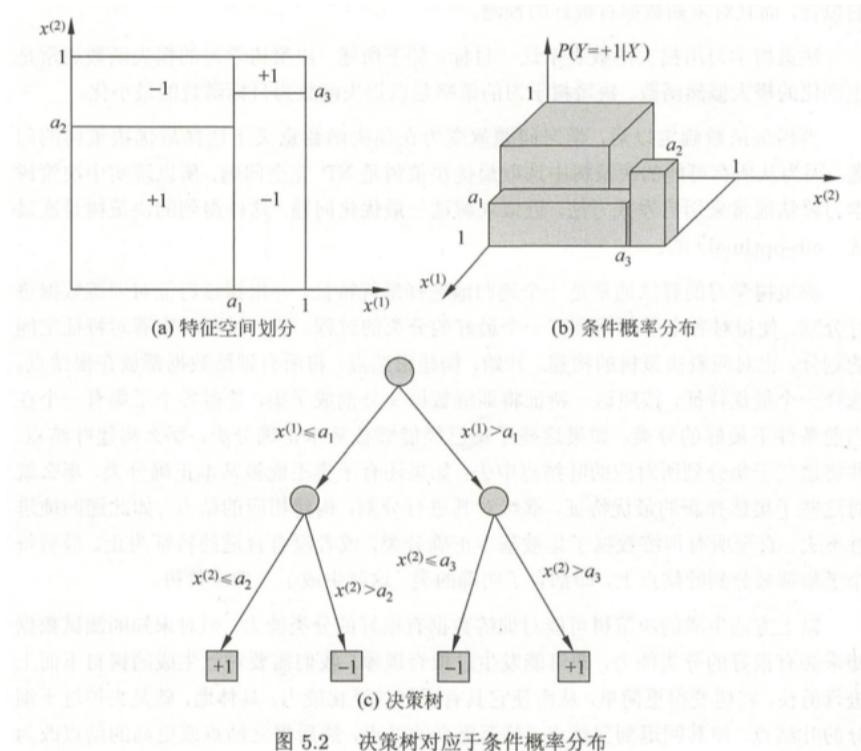
决策树中最上面的结点称为**根结点**。是整个决策树的开始。每个分支是一个新的**决策结点**，或者是树的**叶子**。

每个决策结点代表一个问题或者决策。通常对应待分类对象的属性。每个叶结点代表一种**可能的分类结果**。

在沿着决策树从上到下的遍历过程中，在每个结点都有一个测试。对每个结点上问题的不同测试输出导致不同的分枝，最后会达到一个叶子结点。这一过程就是利用决策树进行分类的过程，**利用若干个变量来判断属性的类别**。

条件概率分布

1. 决策树表示**给定特征条件下类的条件概率分布**。
2. 条件概率分布定义在特征空间的一个**划分** (partition) 上，将特征空间划分为**互不相交的单元** (cell) 或区域 (region)，并在每个单元定义一个类的概率分布就构成了一个**条件概率分布**。
3. 决策树的一条路径对应于划分中的一个单元。
4. 决策树所表示的条件概率分布由各个单元给定条件下类的条件概率分布组成



决策树学习本质上是从训练数据集中**归纳出一组分类规则，与训练数据集不相矛盾的决策树**。

能对训练数据进行正确分类的决策树可能有多个，也可能一个也没有，我们需要的是一个**与训练数据矛盾较小的决策树**，同时具有很好的**泛化能力**。

决策树学习是由**训练数据集估计条件概率模型**，基于特征空间划分的类的条件概率模型有无穷多个。

我们选择的条件概率模型应该不仅对训练数据有很好的拟合，而且对未知数据有**很好的预测**。

特征选择

决策树的 CLS 算法

CLS (ConceptLearning System) 算法

CLS 算法是早期的决策树学习算法。它是许多决策树学习算法的基础

CLS 基本思想

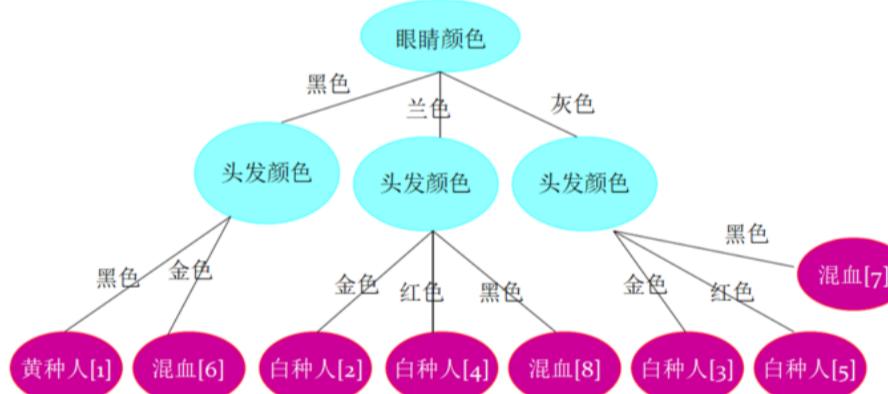
从一棵空决策树开始, 选择某一属性 (分类属性) 作为测试属性。该测试属性对应决策树中的决策结点。根据该属性的值的不同, 可将训练样本分成相应的子集:

如果该子集为空, 或该子集中的样本属于同一个类, 则该子集为叶结点,

否则该子集对应于决策树的内部结点, 即测试结点, 需要选择一个新的分类属性对子集进行划分, 直到所有的子集都为空 或者 属于同一类。

人员	眼睛颜色	头发颜色	所属人种
1	黑色	黑色	黄种人
2	蓝色	金色	白种人
3	灰色	金色	白种人
4	蓝色	红色	白种人
5	灰色	红色	白种人
6	黑色	金色	混血
7	灰色	黑色	混血
8	蓝色	黑色	混血

在这里, 我们先选眼睛颜色、再选头发颜色 (这是随意的, CLS 没有指定顺序)。



步骤:

1. 生成一颗空决策树和一张训练样本属性集
2. 若训练样本集 T 中所有的样本都属于同一类, 则生成结点 T , 并终止学习算法; 否则根据某种策略从训练样本属性表中选择属性 A 作为测试属性, 生成测试结点 A
3. 若 A 的取值为 v_1, v_2, \dots, v_m , 则根据 A 的取值的不同, 将 T 划分成 m 个子集 T_1, T_2, \dots, T_m
4. 从训练样本属性表中删除属性 A
5. 转步骤 2, 对每个子集递归调用 CLS

CLS 算法问题:

在步骤 3 中, 根据某种策略从训练样本属性表中选择属性 A 作为测试属性。没有规定采用何种测试属性。实践表明, 测试属性集的组成以及测试属性的先后对决策树的学习具有举足轻重的影响。

信息增益

Shannon 1948 年提出的信息论理论：

熵(entropy)：信息量大小的度量，即表示随机变量不确定性的度量。

熵的通俗解释：事件 a_i 的信息量 $I(a_i)$ 可如下度量：

$$I(a_i) = p(a_i) \log_2 \frac{1}{p(a_i)}$$

平均信息量

$$I(a_1, a_2, \dots, a_n) = \sum_{i=1}^n I(a_i) = \sum_{i=1}^n p(a_i) \log_2 \frac{1}{p(a_i)}$$

随机变量的熵

$$H(X) = - \sum_{i=1}^n p_i \log p_i$$

对数以 2 为底或以 e 为底(自然对数)，这时熵的单位分别称作比特(bit)或纳特(nat)，熵只依赖于 X 的分布，与 X 的取值无关。

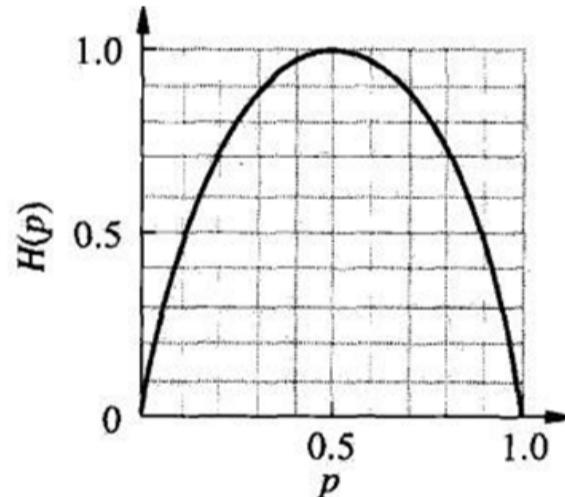
理论解释：熵越大，随机变量的不确定性越大。

0-1 分布

$$P(X = 1) = p, \quad P(X = 0) = 1 - p, \quad 0 \leq p \leq 1$$

熵

$$H(p) = -p \log_2 p - (1 - p) \log_2 (1 - p)$$



显然在 0.5 处熵最大。

考虑联合概率分布

$$P(X = x_i, Y = y_j) = p_{ij}, \quad i = 1, 2, \dots, n; \quad j = 1, 2, \dots, m$$

给出条件熵的定义：表示在已知随机变量 X 的条件下随机变量 Y 的不确定性，定义为 X 给定条件下 Y 的条件概率分布的熵 对 X 的数学期望：

$$\begin{aligned} H(Y | X) &= \sum_x p(x) H(Y | X = x) \\ &= - \sum_x p(x) \sum_y p(y | x) \log p(y | x) \end{aligned}$$

当熵和条件熵中的概率由数据估计（特别是极大似然估计）得到时，所对应的熵与条件熵分别称为经验熵（empirical entropy）和经验条件熵（empirical conditional entropy）

信息增益

$$g(D, A) = H(D) - H(D | A)$$

(Information gain)表示得知特征 X 的信息而使得类 Y 的信息的不确定性减少的程度

一般地，熵 $H(Y)$ 与条件熵 $H(Y|X)$ 之差称为互信息（mutual information）

决策树学习中的信息增益等价于训练数据集中类与特征的互信息

算法

记号

1. 设训练数据集为 D

$|D|$ 表示其样本容量，即样本个数

2. 设有 K 个类 C_k , $k = 1, 2, \dots, K$

$|C_k|$ 为属于类 C_k 的样本个数

3. 特征 A 有 n 个不同的取值 $\{a_1, a_2, \dots, a_n\}$ 根据特征 A 的取值将 D 划分为 n 个子集 $\{D_1, D_2, \dots, D_n\}$

$|D_i|$ 为 D_i 的样本个数

4. 记子集 D_i 中属于类 C_k 的样本集合为 D_{ik}

$|D_{ik}|$ 为 D_{ik} 的样本个数

决策树的生成

ID3 算法

ID3 算法是一种经典的决策树学习算法，由 Quinlan 于 1979 年提出。

ID3 算法主要针对属性选择问题。是决策树学习方法中最具影响和最为典型的算法。

- 该方法使用信息增益度选择测试属性。
- 当获取信息时，将不确定的内容转为确定的内容，因此信息 伴着不确定性。
- 从直觉上讲，小概率事件比大概率事件包含的信息量大。如果某件事情是“百年一见”则肯定比“习以为常”的事件包含的信息量大。

如何度量信息量的大小？答：信息增益

流程

实际使用

小结

C4.5 生成算法

决策树的剪枝

1. 在学习时过多考虑如何提高对训练数据的正确分类，从而构建出过于复杂的决策树，产生过拟合现象。解决方法是对已生成的决策树进行简化，称为剪枝。
2. 设树的叶结点个数为 $|T|$ ，每个叶结点有 N_t 个样本点，其中 k 类样本点有 N_{tk} 个，剪枝往往通过极小化决策树整体的损失函数。

$$C_\alpha(T) = \sum_{t=1}^{|T|} N_t H_t(T) + \alpha |T|$$

其中经验熵

$$H_t(T) = - \sum_k \frac{N_{tk}}{N_t} \log \frac{N_{tk}}{N_t}$$

损失函数就是使用正则化的极大似然估计进行模型选择。为什么是 MLE，理由如下：

$$\sum_{t=1}^{|T|} N_t H_t(T) = - \sum_{t=1}^{|T|} \sum_{k=1}^K N_t \frac{N_{tk}}{N_t} \log \frac{N_{tk}}{N_t} = - \sum_{t=1}^{|T|} \sum_{k=1}^K N_{tk} \log \frac{N_{tk}}{N_t} = - \log \prod_{t=1}^{|T|} \prod_{k=1}^K \left(\frac{N_{tk}}{N_t} \right)^{N_{tk}}$$

3. **剪枝算法**: 剪去某一子结点, 如果生成的新的整体树的**损失函数值**小于原树, 即 $C_\alpha(T_A) \leq C_\alpha(T_B)$, 则进行剪枝, 直到不能继续为止。具体可以由动态规划实现。

CART 算法

决策树面临的问题

理想的决策树有三种:

1. 叶子结点数最少
2. 叶子结点深度最小
3. 叶子结点数最少且叶子结点深度最小

我们发现上述的方法都是**贪心**实现的, 而达到最优是不可能的。

其次, 我们发现决策树容易过拟合, **泛化能力差**。

最后, 对连续属性有时候不太好做。在上文的算法, 我们需要将**连续属性离散化**。

CART 树

CART 既可以用于分类也可以用于回归。它假设决策树是**二叉树**, 内部结点特征的取值为“是”和“否”。递归地构建二叉树, 对回归树用**平方误差最小化准则**, 对分类树用**基尼指数最小化准则**。

CART 生成

回归树

设 Y 是连续变量, 给定训练数据集

假设已将输入空间划分为 M 各单元 $R_1, R_2 \dots R_m$, 并且每个单元 R_m 上有一个固定的输出 C_m , 回归树表示为:

$$f(x) = \sum_{m=1}^M c_m I(x \in R_m)$$

平方误差来表示预测误差, 用平方误差最小准则求解每个单元上的最优输出值

$$\sum_{x \in R_m} (y_i - f(x_i))^2$$

R_m 上的 C_m 的最优值

$$\hat{c}_m = \text{ave}(y_i \mid x_i \in R_m)$$

回归树的生成:

在训练数据集所在的输入空间中, 递归地将每个区域划分为两个子区域。选择第 j 个变量和它取的值 s 作为切分变量和切分点, 并定义两个区域

$$R_1(j, s) = \left\{ x \mid x^{(j)} \leq s \right\} \quad \text{和} \quad R_2(j, s) = \left\{ x \mid x^{(j)} > s \right\}$$

遍历变量 j ，对固定的 j 扫描切分点 s ，求解

$$\min_{j,s} \left[\min_{c_1} \sum_{x \in R_1(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{x \in R_2(j,s)} (y_i - c_2)^2 \right]$$

用选定的对 (j, s) 划分区域并决定相应的输出值

$$\hat{c}_m = \frac{1}{N_m} \sum_{x_i \in R_m(j,s)} y_i, \quad x \in R_m, \quad m = 1, 2$$

直到满足停止条件。

分类树

基尼指数：假设有 K 个类，样本属于第 k 类的概率为 p_k ，则概率分布的基尼指数为

$$\text{Gini}(p) = \sum_{k=1}^K p_k (1 - p_k) = 1 - \sum_{k=1}^K p_k^2$$

表示不确定性。

在特征 A 的条件下集合 D 的基尼指数定义为

$$\text{Gini}(D, A) = \frac{|D_1|}{|D|} \text{Gini}(D_1) + \frac{|D_2|}{|D|} \text{Gini}(D_2)$$

表示分割后集合 D 的不确定性。基尼指数越大，样本集合的**不确定性**也就越大。

分类树的生成：

从根结点开始，递归进行以下操作：设结点的训练数据集为 D ，对每个特征 A 和其可能取的每个值 a ，计算 $A = a$ 时的基尼指数，选择**基尼指数最小**的特征及其对应的切分点作为**最优特征与最优切分点**，生成两个子结点，直至满足**停止条件**。停止条件一般是结点中的样本个数小于阈值，或样本集的基尼指数小于阈值，或没有更多特征。

CART 剪枝

T_t 表示以 t 为根结点的子树， $|T_t|$ 是 T_t 的叶结点个数。可以证明当

$$\alpha = \frac{C(t) - C(T_t)}{|T_t| - 1}$$

T_t 与 t 有相同的损失函数值，且 t 的结点少。因此 t 比 T_t 更可取，对 T_t 进行剪枝。

并令 $a = \min(g(t))$ ，**自上而下**地访问内部节点 t ，如果有 $g(t) = a$ ，进行剪枝，并对 t 以**多数表决法**决定其类，得到子树 T ，如此循环地生成一串**子树序列**，直到新生成的 T 是由根结点单独构成的树为止。利用**交叉验证法**在子树序列中选取最优子树。

如果是**连续值**的情况，一般用**二分法**作为结点来划分。



logistic 回归和最大熵模型

logistic 回归

回归：广义线性模型 (generalized linear model)

分类：根据因变量的不同

1. 连续：多重线性回归
2. 二项分布：logistic 回归
3. poisson 分布：poisson 回归
4. 负二项分布：负二项回归

逻辑斯蒂分布

分布函数

$$F(x) = P(X \leq x) = \frac{1}{1 + e^{-(x-\mu)/\gamma}}$$

密度函数

$$f(x) = F'(x) = \frac{e^{-(x-\mu)/\gamma}}{\gamma (1 + e^{-(x-\mu)/\gamma})^2}$$



sigmoid: [0,1]

$$f(z) = \frac{1}{1 + \exp(-z)}$$

$$f'(z) = f(z)(1 - f(z))$$

sigmoid 可以扩展到多类

$$h_{W,b}(x) = f(W^T x) = f\left(\sum_{i=1}^n W_i x_i + b\right)$$

tanh: [0,1]

$$f(z) = \tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

$$f'(z) = 1 - (f(z))^2$$

二项逻辑斯蒂回归

由条件概率 $P(Y|X)$ 表示的分类模型

形式化为 logistic distribution

X 取实数, Y 取值 1,0

$$P(Y = 1 | x) = \frac{\exp(w \cdot x + b)}{1 + \exp(w \cdot x + b)}$$

$$P(Y = 0 | x) = \frac{1}{1 + \exp(w \cdot x + b)}$$

扩展向量 w 和 b 后简化公式

$$P(Y = 1 | x) = \frac{\exp(w \cdot x)}{1 + \exp(w \cdot x)}$$

$$P(Y = 0 | x) = \frac{1}{1 + \exp(w \cdot x)}$$

事件的几率 odds: 事件发生与事件不发生的概率之比 (the odds of experiencing an event) 为

$$\frac{p}{1 - p}$$

对数几率

$$\text{logit}(p) = \log \frac{p}{1 - p}$$

对逻辑斯蒂回归

$$\log \frac{P(Y = 1 | x)}{1 - P(Y = 1 | x)} = w \cdot x$$

也就是说在逻辑斯蒂回归模型中, 输出 $Y = 1$ 的对数几率是输入 x 的线性函数, 线性函数值越接近正无穷, 概率值就越接近 1, 反之则越接近 0。

极大似然函数估计

定义

$$P(Y = 1 | x) = \pi(x), \quad P(Y = 0 | x) = 1 - \pi(x)$$

二项分布的似然函数

$$\prod_{i=1}^N [\pi(x_i)]^{y_i} [1 - \pi(x_i)]^{1-y_i}$$

因此，对于给定的二分类训练数据集，对数似然函数为

$$\begin{aligned} L(w) &= \sum_{i=1}^N [y_i \log \pi(x_i) + (1 - y_i) \log (1 - \pi(x_i))] \\ &= \sum_{i=1}^N \left[y_i \log \frac{\pi(x_i)}{1 - \pi(x_i)} + \log (1 - \pi(x_i)) \right] \\ &= \sum_{i=1}^N [y_i (w \cdot x_i) - \log (1 + \exp(w \cdot x_i))] \end{aligned}$$

对 $L(w)$ 求最大值，得到 w 的估计值。

通常采用梯度下降法和拟牛顿法，学到的模型

$$P(Y = 1 \mid x) = \frac{\exp(\hat{w} \cdot x)}{1 + \exp(\hat{w} \cdot x)}$$

$$P(Y = 0 \mid x) = \frac{1}{1 + \exp(\hat{w} \cdot x)}$$

多项 logistic 回归

多项 logistic 回归模型

$$P(Y = k \mid x) = \frac{\exp(w_k \cdot x)}{1 + \sum_{k=1}^{K-1} \exp(w_k \cdot x)}, \quad k = 1, 2, \dots, K-1$$

$$P(Y = K \mid x) = \frac{1}{1 + \sum_{k=1}^{K-1} \exp(w_k \cdot x)}$$

最大熵模型

最大熵原理

学习概率模型时，在所有可能的概率模型(分布)中，熵最大的模型是最好的模型，表述为在满足约束条件的模型集合中选取熵最大的模型。

熵

$$H(P) = - \sum_x P(x) \log P(x)$$

取值范围

$$0 \leq H(P) \leq \log |X|$$

例子

假设随机变量 X 有 5 个取值 $\{A, B, C, D, E\}$ ，估计各个值的概率。

无先验概率

$$P(A) = P(B) = P(C) = P(D) = P(E) = \frac{1}{5}$$

先验

$$P(A) + P(B) = \frac{3}{10}$$

于是有

$$\begin{aligned} P(A) = P(B) &= \frac{3}{20} \\ P(C) = P(D) = P(E) &= \frac{7}{30} \end{aligned}$$

给定数据集

分布的经验函数

$$\tilde{P}(X = x, Y = y) = \frac{\nu(X = x, Y = y)}{N}$$

$$\tilde{P}(X = x) = \frac{v(X = x)}{N}$$

最大熵模型的定义

$$H(P) = - \sum_{x,y} \tilde{P}(x)P(y \mid x) \log P(y \mid x)$$

最大熵模型的学习

极大似然估计

模型学习的最优化方法



SVM

线性可分支持向量机与硬间隔最大化

线性可分支持向量机

分离超平面

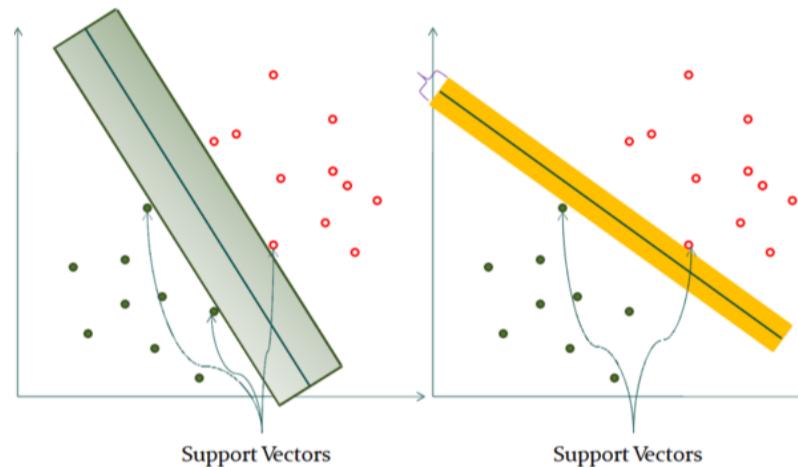
$$w^* \cdot x + b^* = 0$$

决策函数

$$f(x) = \text{sign}(w^* \cdot x + b^*)$$

训练数据集线性可分，且间隔最大

显然第一张图是我要找到的超平面



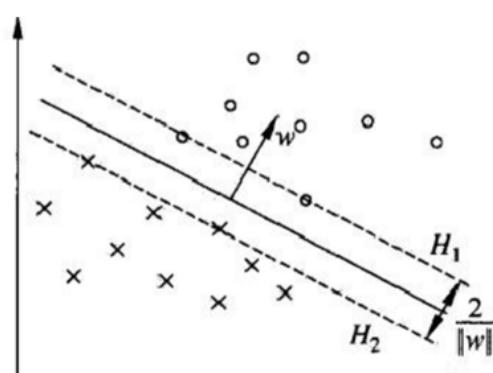
在线性可分情况下，训练数据集的样本点中与分离超平面距离最近的样本点的实例称为支持向量 (support vector)

原始形式最优化问题

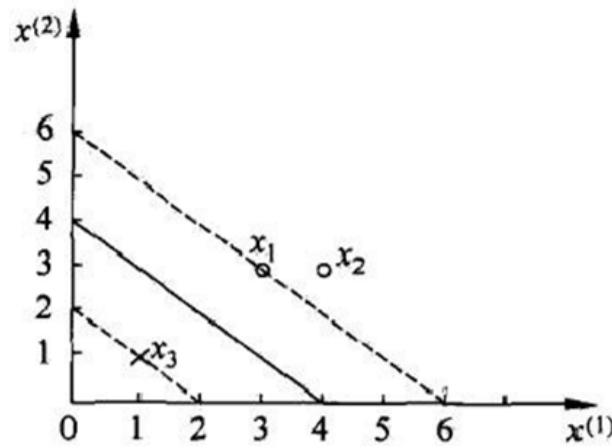
$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (1)$$

$$\text{s.t. } y_i (w \cdot x_i + b) - 1 \geq 0, \quad i = 1, 2, \dots, N \quad (2)$$

是凸二次规划问题



例子 1



$$\min_{w, b} \frac{1}{2} (w_1^2 + w_2^2) \quad (3)$$

$$\text{s.t. } 3w_1 + 3w_2 + b \geq 1 \quad (4)$$

$$4w_1 + 3w_2 + b \geq 1 \quad (5)$$

$$-w_1 - w_2 - b \geq 1 \quad (6)$$

$$w_1 = w_2 = \frac{1}{2}, \quad b = -2$$

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

其中

$$x_1 = (3, 3)^T \quad x_3 = (1, 1)^T$$

拉格朗日对偶

$$\min_{x \in \mathbf{R}^n} f(x) \quad (7)$$

$$\text{s.t. } c_i(x) \leq 0, \quad i = 1, 2, \dots, k \quad (8)$$

$$h_j(x) = 0, \quad j = 1, 2, \dots, l \quad (9)$$

拉格朗日函数

$$L(x, \alpha, \beta) = f(x) + \sum_{i=1}^k \alpha_i c_i(x) + \sum_{j=1}^l \beta_j h_j(x)$$

如果某个 x 违反约束条件

$$\theta_P(x) = \max_{\alpha, \beta: \alpha_i \geq 0} \left[f(x) + \sum_{i=1}^k \alpha_i c_i(x) + \sum_{j=1}^l \beta_j h_j(x) \right] = +\infty$$

因此有

$$\theta_P(x) = \begin{cases} f(x), & x \text{ 满足原始问题约束} \\ +\infty, & \text{其他} \end{cases}$$

考虑极小问题

$$\min_x \theta_P(x) = \min_x \max_{\alpha, \beta: \alpha_i \geq 0} L(x, \alpha, \beta)$$

与原问题等价

$$p^* = \min_x \theta_p(x)$$

对偶问题

定义

$$\theta_D(\alpha, \beta) = \min_x L(x, \alpha, \beta)$$

最大值问题

$$\max_{\alpha, \beta: \alpha_i \geq 0} \theta_D(\alpha, \beta) = \max_{\alpha, \beta: \alpha_i \geq 0} \min_x L(x, \alpha, \beta)$$

称为广义拉格朗日函数的极大极小问题

对偶问题最优值

$$d^* = \max_{\alpha, \beta: \alpha \geq 0} \theta_D(\alpha, \beta)$$

若原始问题和对偶问题都有最优值，则

$$d^* = \max_{\alpha, \beta: \alpha_i \geq 0} \min_x L(x, \alpha, \beta) \leq \min_x \max_{\alpha, \beta: \alpha_i \geq 0} L(x, \alpha, \beta) = p^*$$

等号在 KKT 条件成立时取得

$$\begin{aligned} \nabla_x L(x^*, \alpha^*, \beta^*) &= 0 \\ \nabla_\alpha L(x^*, \alpha^*, \beta^*) &= 0 \\ \nabla_\beta L(x^*, \alpha^*, \beta^*) &= 0 \end{aligned}$$

$$\begin{aligned} \alpha_i^* c_i(x^*) &= 0, \quad i = 1, 2, \dots, k \\ c_i(x^*) &\leq 0, \quad i = 1, 2, \dots, k \\ \alpha_i^* &\geq 0, \quad i = 1, 2, \dots, k \\ h_j(x^*) &= 0 \quad j = 1, 2, \dots, l \end{aligned}$$

学习的对偶算法

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^N \alpha_i$$

对偶问题

$$\max_{\alpha} \min_{w, b} L(w, b, \alpha)$$

KKT 条件

$$\nabla_w L(w, b, \alpha) = w - \sum_{i=1}^N \alpha_i y_i x_i = 0$$

$$\nabla_b L(w, b, \alpha) = \sum_{i=1}^N \alpha_i y_i = 0$$

得

$$w = \sum_{i=1}^N \alpha_i y_i x_i$$

$$\sum_{i=1}^N \alpha_i y_i = 0$$

那么广义拉格朗日函数化简为

$$L(w, b, \alpha) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i y_i \left(\left(\sum_{j=1}^N \alpha_j y_j x_j \right) \cdot x_i + b \right) + \sum_{i=1}^N \alpha_i$$

$$= -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^N \alpha_i$$

对偶问题

$$\max_{\alpha} -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^N \alpha_i$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0$$

$$\alpha_i \geq 0, \quad i = 1, 2, \dots, N$$

化为

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (10)$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (11)$$

$$\alpha_i \geq 0, \quad i = 1, 2, \dots, N \quad (12)$$

设对偶问题的解为

$$\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_l^*)^T$$

则原始问题的解为

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i \quad (13)$$

$$b^* = y_j - \sum_{i=1}^N \alpha_i^* y_i (x_i \cdot x_j) \quad (14)$$

分离超平面

$$\sum_{i=1}^N \alpha_i^* y_i (x \cdot x_i) + b^* = 0$$

决策函数为

$$f(x) = \text{sign} \left(\sum_{i=1}^N \alpha_i^* y_i (x \cdot x_i) + b^* \right)$$

例子 2

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (15)$$

$$= \frac{1}{2} (18\alpha_1^2 + 25\alpha_2^2 + 2\alpha_3^2 + 42\alpha_1\alpha_2 - 12\alpha_1\alpha_3 - 14\alpha_2\alpha_3) - \alpha_1 - \alpha_2 - \alpha_3 \quad (16)$$

$$\text{s.t. } \alpha_1 + \alpha_2 - \alpha_3 = 0 \quad (17)$$

$$\alpha_i \geq 0, \quad i = 1, 2, 3 \quad (18)$$

带入约束

$$s(\alpha_1, \alpha_2) = 4\alpha_1^2 + \frac{13}{2}\alpha_2^2 + 10\alpha_1\alpha_2 - 2\alpha_1 - 2\alpha_2$$

求条件极值，注意在偏导数为 0 的地方不满足约束，因此在边界上取得

$$s\left(0, \frac{2}{13}\right) = -\frac{2}{13}$$

$$s\left(\frac{1}{4}, 0\right) = -\frac{1}{4}$$

因此最小值在 $(\frac{1}{4}, 0)$ 取得，由公式

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i \quad (19)$$

$$b^* = y_j - \sum_{i=1}^N \alpha_i^* y_i (x_i \cdot x_j) \quad (20)$$

计算可得

$$\begin{aligned} w_1^* &= w_2^* = \frac{1}{2} \\ b^* &= -2 \end{aligned}$$

分离超平面

$$\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 = 0$$

分类决策函数

$$f(x) = \text{sign} \left(\frac{1}{2}x^{(1)} + \frac{1}{2}x^{(2)} - 2 \right)$$

线性支持向量机与软间隔最大化

动机

数据集大部分是线性不可分的

引入松弛因子

$$y_i (w \cdot x_i + b) \geq 1 - \xi_i$$

目标函数变为

$$\frac{1}{2} \|w\|^2 + C \sum_{j=1}^N \xi_i$$

C 为惩罚参数。

线性不可分的 SVM 学习问题

$$\min_{w, b, \xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i \quad (21)$$

$$\text{s.t. } y_i (w \cdot x_i + b) \geq 1 - \xi_i, \quad i = 1, 2, \dots, N \quad (22)$$

$$\xi_i \geq 0, \quad i = 1, 2, \dots, N \quad (23)$$

拉格朗日对偶

拉格朗日函数

$$L(w, b, \xi, \alpha, \mu) \equiv \frac{1}{2} \|w\|^2 + C \sum_{i=1}^N \xi_i - \sum_{i=1}^N \alpha_i (y_i (w \cdot x_i + b) - 1 + \xi_i) - \sum_{i=1}^N \mu_i \xi_i$$

使用 KKT 条件, 得到

$$\begin{aligned} w &= \sum_{i=1}^N \alpha_i y_i x_i \\ \sum_{i=1}^N \alpha_i y_i &= 0 \\ C - \alpha_i - \mu_i &= 0 \end{aligned}$$

因此对偶问题为

$$\max_{\alpha} -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) + \sum_{i=1}^N \alpha_i \quad (24)$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (25)$$

$$C - \alpha_i - \mu_i = 0 \quad (26)$$

$$\alpha_i \geq 0 \quad (27)$$

$$\mu_i \geq 0, \quad i = 1, 2, \dots, N \quad (28)$$

因为目标函数没有 μ_i ，因此有

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (29)$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (30)$$

$$0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N \quad (31)$$

原始问题的解

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i \quad (32)$$

$$b^* = y_j - \sum_{i=1}^N y_i \alpha_i^* (x_i \cdot x_j) \quad (33)$$

因此我们可以构造求解约束最优化问题

$$\min_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \quad (34)$$

$$\text{s.t.} \quad \sum_{i=1}^N \alpha_i y_i = 0 \quad (35)$$

$$0 \leq \alpha_i \leq C, \quad i = 1, 2, \dots, N \quad (36)$$

得到最优解

$$\alpha^* = (\alpha_1^*, \alpha_2^*, \dots, \alpha_N^*)^T$$

计算

$$w^* = \sum_{i=1}^N \alpha_i^* y_i x_i$$

并选择 α_i^* ，适合条件 $0 < \alpha_i^* < C$

$$b^* = y_j - \sum_{i=1}^N \alpha_i^* y_i (x_i \cdot x_j)$$

得到分离超平面

$$w^* \cdot x + b^* = 0$$

分类决策函数

$$f(x) = \text{sign}(w^* \cdot x + b^*)$$

注意我们的 KKT 条件

$$\begin{aligned}\nabla_w L(w^*, b^*, \xi^*, \alpha^*, \mu^*) &= w^* - \sum_{i=1}^N \alpha_i^* y_i x_i = 0 \\ \nabla_b L(w^*, b^*, \xi^*, \alpha^*, \mu^*) &= - \sum_{i=1}^N \alpha_i^* y_i = 0 \\ \nabla_\xi L(w^*, b^*, \xi^*, \alpha^*, \mu^*) &= C - \alpha^* - \mu^* = 0 \\ \alpha_i^* (y_i (w^* \cdot x_i + b^*) - 1 + \xi_i^*) &= 0 \\ \mu_i^* \xi_i^* &= 0 \\ y_i (w^* \cdot x_i + b^*) - 1 + \xi_i^* &\geq 0 \\ \xi_i^* &\geq 0 \\ \alpha_i^* &\geq 0 \\ i &= 1, 2, \dots, N\end{aligned}$$

讨论：

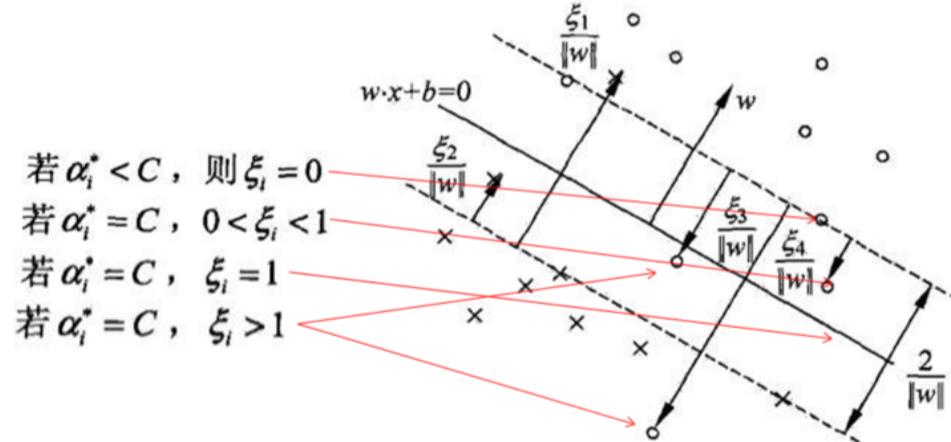
可以从模型复杂度的角度观察惩罚系数 C 的作用。

C 较大，导致模型训练错误率变小，复杂度变大，高方差。

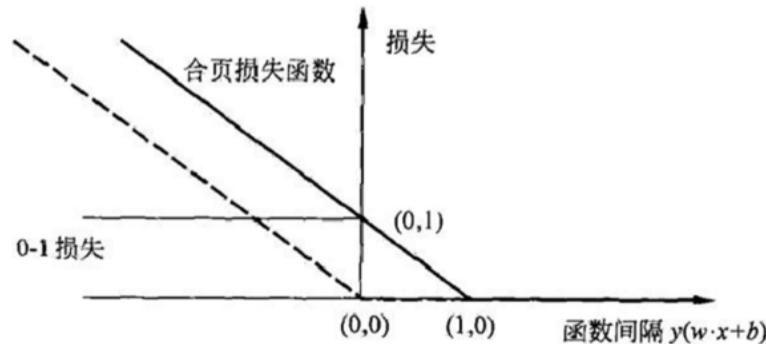
C 较小，导致模型训练错误率变大，复杂度变小，高偏差。

根据 KKT 条件，我们能够分析每个点的作用。

我们发现，只有在 boundary 上和那些不那么完美分类的点才对 w^* 有影响。

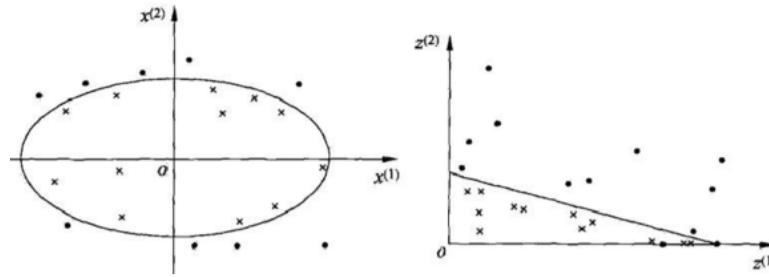


合页损失函数



非线性支持向量机与核函数

如果能用 \mathbb{R}^n 中的一个超曲面将正负例正确分开，则称这个问题为非线性可分问题。



用线性分类方法求解非线性分类问题分为两步：

1. 首先使用一个变换将原空间的数据映射到新空间。
2. 然后在新空间里用线性分类学习方法从训练数据中学习分类模型。

核函数

输入空间： \mathbb{R}^n

特征空间：希尔伯特空间

存在一个映射

$$\phi(x) : \mathcal{X} \rightarrow \mathcal{H}$$

使得对所有 $x, z \in \mathcal{X}$

$$K(x, z) = \phi(x) \cdot \phi(z)$$

则称 $K(x, z)$ 为核函数， $\phi(\cdot)$ 为映射函数，我们一般直接定义核函数

例子：试找出映射

取特征空间 $\mathcal{H} = \mathbf{R}^3$ ，记 $x = (x^{(1)}, x^{(2)})^T$, $z = (z^{(1)}, z^{(2)})^T$

核函数

$$K(x, z) = (x \cdot z)^2$$

可以取

$$(x \cdot z)^2 = (x^{(1)}z^{(1)} + x^{(2)}z^{(2)})^2 = (x^{(1)}z^{(1)})^2 + 2x^{(1)}z^{(1)}x^{(2)}z^{(2)} + (x^{(2)}z^{(2)})^2$$

可以取

$$\phi(x) = \left((x^{(3)})^2, \sqrt{2}x^{(1)}x^{(2)}, (x^{(2)})^2 \right)^T$$

可以验证

$$\phi(x) \cdot \phi(z) = (x \cdot z)^2 = K(x, z)$$

核函数在 SVM 的应用

目标函数

$$W(\alpha) = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^N \alpha_i$$

决策函数

$$f(x) = \text{sign} \left(\sum_{i=1}^{N_s} a_i^* y_i \phi(x_i) \cdot \phi(x) + b^* \right) = \text{sign} \left(\sum_{i=1}^{N_s} a_i^* y_i K(x_i, x) + b^* \right)$$

正定核

1. 定义映射
2. 定义向量空间
3. 定义内积空间
4. 定义希尔伯特空间

充要条件

设 $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbf{R}$ 是对称函数, 则 $K(x, z)$ 为正定核函数的充要条件是对任意 $x_i \in \mathcal{X}, i = 1, 2, \dots, m$, $K(x, z)$ 对应的 Gram 矩阵

$$K = [K(x_i, x_j)]_{m \times m}$$

是半正定的。

常用核函数

1. 多项式核函数

$$K(x, z) = (x \cdot z + 1)^p$$

2. 高斯核函数

$$K(x, z) = \exp \left(-\frac{\|x - z\|^2}{2\sigma^2} \right)$$

非线性 SVM

$$f(x) = \text{sign} \left(\sum_{i=1}^N \alpha_i^* y_i K(x \cdot x_i) + b^* \right)$$

如果 K 是正定核, 则决策函数是凸二次规划问题。

序列最小最优化算法

支持向量机的学习问题可以形式化为求解凸二次规划问题。这样的凸二次规划问题具有全局最优解，并且有许多最优化算法可以用于这一问题的求解；

但是当训练样本容量很大时，这些算法往往变得非常低效，以致无法使用。所以，如何高效地实现支持向量机学习就成为一个重要的问题。

$$\min_{\alpha_1, \alpha_2} W(\alpha_1, \alpha_2) = \frac{1}{2} K_{11} \alpha_1^2 + \frac{1}{2} K_{22} \alpha_2^2 + y_1 y_2 K_{12} \alpha_1 \alpha_2 \quad (37)$$

$$- (\alpha_1 + \alpha_2) + y_1 \alpha_1 \sum_{i=3}^N y_i \alpha_i K_{i1} + y_2 \alpha_2 \sum_{i=3}^N y_i \alpha_i K_{i2} \quad (38)$$

$$\text{s.t.} \quad \alpha_1 y_1 + \alpha_2 y_2 = - \sum_{i=3}^N y_i \alpha_i = \zeta \quad (39)$$

$$0 \leq \alpha_i \leq C, \quad i = 1, 2 \quad (40)$$



提升方法

提升方法 AdaBoost

bagging & boosting

也称 bootstrap aggregating

- 抽样得到多个新数据集，每个大小和原数据集相等
- 对某个学习算法分别作用于每个数据集得到多个分类器
- 投票法：选择投票结果最多的类别作为最后的分类结果
 - 并行：各自投票 bagging
 - 串行：在前一个分类器预测不够准的情况下继续构建新的分类器 boosting

Adaboost

1. 每一轮如何改变训练数据的权值或概率分布？

Adaboost：提高那些被前一轮弱分类器错误分类样本的权值，降低那些被正确分类样本的权值。

2. 如何将弱分类器组合成一个强分类器？

Adaboost：加权多数表决，加大分类误差率小的弱分类器的权值，使其在表决中起较大的作用，减小分类误差率大的弱分类器的权值，使其在表决中起较小的作用。

初始化权重分布

$$D_1 = (w_{11}, \dots, w_{1i}, \dots, w_{1N}) \quad w_{1i} = \frac{1}{N}, \quad i = 1, 2, \dots, N$$

计算弱分类器错误率

$$e_m = P(G_m(x_i) \neq y_i) = \sum_{G_m(x_i) \neq y_i} w_{mi}$$

系数

$$\alpha_m = \frac{1}{2} \log \frac{1 - e_m}{e_m}$$

训练数据集权值分布

$$w_{m+1,i} = \begin{cases} \frac{w_{mi}}{Z_m} e^{-\alpha_m}, & G_m(x_i) = y_i(1) \\ \frac{w_{mi}}{Z_m} e^{\alpha_m}, & G_m(x_i) \neq y_i(2) \end{cases}$$

归一化因子

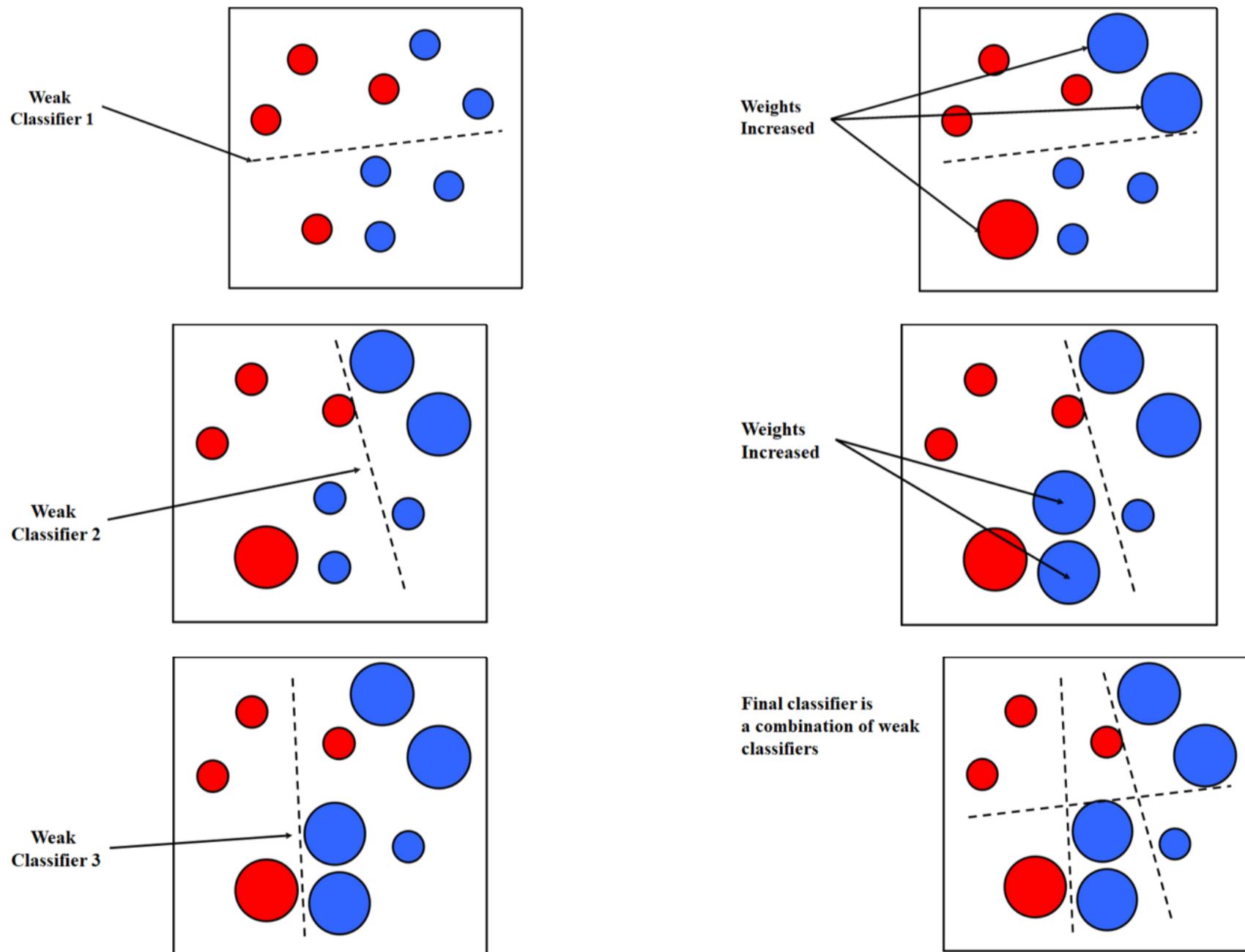
$$Z_m = \sum_{i=1}^N w_{mi} \exp(-\alpha_m y_i G_m(x_i))$$

构建弱分类器线性组合

$$f(x) = \sum_{m=1}^M \alpha_m G_m(x)$$

得到最终分类器

$$G(x) = \text{sign}(f(x)) = \text{sign} \left(\sum_{m=1}^M \alpha_m G_m(x) \right)$$



AdaBoost 算法的训练误差分析

AdaBoost 算法的解释

特殊的加法模型

1. 损失函数：指数函数
2. 学习算法：前向分布算法的二分类学习算法

不要求证明

加法模型：每一步只学习一个基函数及其系数

$$\min_{\beta_m, \gamma_m} \sum_{i=1}^N L \left(y_i, \sum_{m=1}^M \beta_m b(x_i; \gamma_m) \right)$$

提升树

1. 加法模型
2. 前向分布算法
3. 基函数：决策树

$$f_M(x) = \sum_{m=1}^M T(x; \Theta_m)$$

首先确定初始提升树

$$f_0(x) = 0$$

第 m 步模型

$$f_m(x) = f_{m-1}(x) + T(x; \Theta_m)$$

损失函数

1. 平方误差：回归
2. 指数损失：分类
3. 一般损失：一般问题



EM 算法

例子

伯努利分布

只有一个隐变量

似然函数

$$P(Y \mid \theta) = \sum_Z P(Z \mid \theta) P(Y \mid Z, \theta)$$

带入观测数据和未观测数据 (完全数据)

$$P(Y \mid \theta) = \prod_{j=1}^n [\pi p^{y_j} (1-p)^{1-y_j} + (1-\pi)q^{y_j} (1-q)^{1-y_j}]$$

选择初值

$$\theta^{(0)} = (\pi^{(0)}, p^{(0)}, q^{(0)})$$

计算每一步的估计值

$$\theta^{(i)} = (\pi^{(i)}, p^{(i)}, q^{(i)})$$

E 步 (因为只有一个隐变量, 所以只有一个隐变量求期望)

$$\mu^{(i+1)} = \frac{\pi^{(i)} (p^{(i)})^{y_j} (1-p^{(i)})^{1-y_j}}{\pi^{(i)} (p^{(i)})^{y_j} (1-p^{(i)})^{1-y_j} + (1-\pi^{(i)}) (q^{(i)})^{y_j} (1-q^{(i)})^{1-y_j}}$$

M 步 (对所有变量极大化似然)

$$\pi^{(i+1)} = \frac{1}{n} \sum_{j=1}^n \mu_j^{(i+1)}$$

$$p^{(i+1)} = \frac{\sum_{j=1}^n \mu_j^{(i+1)} y_j}{\sum_{j=1}^n \mu_j^{(i+1)}}$$

$$q^{(i+1)} = \frac{\sum_{j=1}^n (1 - \mu_j^{(i+1)}) y_j}{\sum_{j=1}^n (1 - \mu_j^{(i+1)})}$$

GMM

在前面的统计课里写过了。

首先要找到完全数据的似然函数

$$\begin{aligned} P(y, \gamma | \theta) &= \prod_{j=1}^N P(y_j, \gamma_{j1}, \gamma_{j2}, \dots, \gamma_{jK} | \theta) \\ &= \prod_{k=1}^K \prod_{j=1}^N [\alpha_k \phi(y_j | \theta_k)]^{\gamma_k} \\ &= \prod_{k=1}^K \alpha_k^n \prod_{j=1}^N [\phi(y_j | \theta_k)]^{\gamma_k} \\ &= \prod_{k=1}^K \alpha_k^{n_k} \prod_{j=1}^N \left[\frac{1}{\sqrt{2\pi}\sigma_k} \exp\left(-\frac{(y_j - \mu_k)^2}{2\sigma_k^2}\right) \right]^{\gamma_j} \end{aligned}$$

E步

对隐变量求期望

$$\begin{aligned} \hat{\gamma}_{jk} &= E(\gamma_{jk} | y, \theta) = P(\gamma_{jk} = 1 | y, \theta) \\ &= \frac{P(\gamma_{jk} = 1, y_j | \theta)}{\sum_{k=1}^K P(\gamma_{jk} = 1, y_j | \theta)} \\ &= \frac{P(y_j | \gamma_{jk} = 1, \theta) P(\gamma_{jk} = 1 | \theta)}{\sum_{k=1}^K P(y_j | \gamma_{jk} = 1, \theta) P(\gamma_{jk} = 1 | \theta)} \\ &= \frac{\alpha_k \phi(y_j | \theta_k)}{\sum_{k=1}^K \alpha_k \phi(y_j | \theta_k)}, \quad j = 1, 2, \dots, N; \quad k = 1, 2, \dots, K \end{aligned}$$

M步

Q函数

$$\begin{aligned} Q(\theta, \theta^{(i)}) &= E \left[\log P(y, \gamma | \theta) | \gamma, \theta^{(i)} \right] \\ &= E \left\{ \sum_{k=1}^K n_k \log \alpha_k + \sum_{j=1}^N \gamma_{jk} \left[\log \left(\frac{1}{\sqrt{2\pi}} \right) - \log \sigma_k - \frac{1}{2\sigma_k^2} (y_j - \mu_k)^2 \right] \right\} \\ &= \sum_{k=1}^K \left\{ \sum_{j=1}^N (E\gamma_{jk}) \log \alpha_k + \sum_{j=1}^N (E\gamma_{jk}) \left[\log \left(\frac{1}{\sqrt{2\pi}} \right) - \log \sigma_k - \frac{1}{2\sigma_k^2} (y_j - \mu_k)^2 \right] \right\} \\ &= \sum_{k=1}^K n_k \log \alpha_k + \sum_{k=1}^N \hat{\gamma}_{jk} \left[\log \left(\frac{1}{\sqrt{2\pi}} \right) - \log \sigma_k - \frac{1}{2\sigma_k^2} (y_j - \mu_k)^2 \right] \end{aligned}$$

EM就是E和M

EM就是极大化下限函数

K-means 是一种 Hard EM

从 EM 到广义 EM



隐马尔可夫模型

基本概念

隐马尔科夫模型的定义

隐马尔可夫模型是关于时序的概率模型。

描述由一个隐藏的马尔可夫链随机生成不可观测的状态随机序列 (state sequence), 再由各个状态生成一个观测而产生观测随机序列 (observation sequence) 的过程, 序列的每一个位置又可以看作是一个时刻。

记号

Q : 所有可能状态的集合

V : 所有可能观测的集合

$$Q = \{q_1, q_2, \dots, q_N\}, \quad V = \{v_1, v_2, \dots, v_M\}$$

I : 长度为 T 的状态序列

O : 对应的观测序列

$$I = (i_1, i_2, \dots, i_T), \quad O = (o_1, o_2, \dots, o_T)$$

初始概率分布

$$\pi_i = P(i_1 = q_i), \quad i = 1, 2, \dots, N$$

状态转移概率分布

$$A = [a_{ij}]_{N \times N}$$

$$a_{ij} = P(i_{t+1} = q_j \mid i_t = q_i), \quad i = 1, 2, \dots, N; j = 1, 2, \dots, N$$

观测概率分布

$$B = [b_j(k)]_{N \times M}$$

$$b_j(k) = P(o_t = v_k \mid i_t = q_j), \quad k = 1, 2, \dots, M; j = 1, 2, \dots, N$$

三要素

$$\lambda = (A, B, \pi)$$

基本假设：

1. 齐次马尔可夫性假设：隐马尔可夫链 t 的状态只和 $t-1$ 状态有关：

$$P(i_t | i_{t-1}, o_{t-1}, \dots, i_1, o_1) = P(i_t | i_{t-1}), \quad t = 1, 2, \dots, T$$

2. 观测独立性假设，观测只和当前时刻状态有关：

$$P(o_t | i_T, o_T, i_{T-1}, o_{T-1}, \dots, i_{t+1}, o_{t+1}, i_t, i_{t-1}, o_{t-1}, \dots, i_1, o_1) = P(o_t | i_t)$$

例子

盒子	1	2	3	4
红球	5	3	6	8
白球	5	7	4	2

转移规则：

1. 盒子 1：下一个一定是盒子 2
2. 盒子 2 或 3：下一个 0.4 左、0.6 右
3. 盒子 4：下一个 0.5 自身、0.5 左

观测序列：

$$O = \{\text{红, 红, 白, 白, 红}\}$$

状态集合

$$Q = \{\text{盒子}_1, \text{盒子}_2, \text{盒子}_3, \text{盒子}_4\}, \quad N = 4$$

观测集合

$$V = \{\text{红球, 白球}\} \quad M = 2$$

初始化概率分布

$$\pi = (0.25, 0.25, 0.25, 0.25)'$$

状态转移矩阵

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0.6 & 0 \\ 0 & 0.4 & 0 & 0.6 \\ 0 & 0 & 0.5 & 0.5 \end{bmatrix}$$

观测矩阵

$$B = \begin{bmatrix} 0.5 & 0.5 \\ 0.3 & 0.7 \\ 0.6 & 0.4 \\ 0.8 & 0.2 \end{bmatrix}$$

观测序列的生成过程

算法 10.1 (观测序列的生成)

输入: 隐马尔可夫模型 $\lambda = (A, B, \pi)$, 观测序列长度 T ;

输出: 观测序列 $O = (o_1, o_2, \dots, o_T)$.

- (1) 按照初始状态分布 π 产生状态 i_1
- (2) 令 $t = 1$
- (3) 按照状态 i_t 的观测概率分布 $b_{i_t}(k)$ 生成 o_t
- (4) 按照状态 i_t 的状态转移概率分布 $\{a_{i_t i_{t+1}}\}$ 产生状态 i_{t+1} , $i_{t+1} = 1, 2, \dots, N$
- (5) 令 $t = t + 1$; 如果 $t < T$, 转步 (3); 否则, 终止

隐马尔科夫模型的三个基本问题

一共有三种:

1. 概率计算问题

给定 λ, O , 计算 $P(O|\lambda)$

2. 学习问题

给定 O , 估计 λ 使得 $P(O|\lambda)$ 最大

3. 预测问题

给定 λ, O , 求使得 $P(O|\lambda)$ 最大的 I

概率计算算法

直接计算法

$$\begin{aligned} P(O | \lambda) &= \sum_I P(O | I, \lambda) P(I | \lambda) \\ &= \sum_{i_1, i_2, \dots, i_T} \pi_{i_1} b_{i_1}(o_1) a_{i_1 i_2} b_{i_2}(o_2) \cdots a_{i_{T-1} i_T} b_{i_T}(o_T) \end{aligned}$$

复杂度 $O(TN^T)$, 因为有重叠子问题, 可使用 DP 进行简化。

前向算法

记前向概率

$$\alpha_t(i) = P(o_1, o_2, \dots, o_t, i_t = q_i \mid \lambda)$$

初值

$$\alpha_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, N$$

从前往后递推

$$\alpha_{t+1}(i) = \left[\sum_{j=1}^N \alpha_t(j) a_{ji} \right] b_i(o_{t+1}), \quad i = 1, 2, \dots, N$$

终止

$$P(O \mid \lambda) = \sum_{i=1}^N \alpha_T(i)$$

复杂度 $O(N^2T)$ 。

例子

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = (0.2, 0.4, 0.4)^T$$

$$O = (\text{红}, \text{白}, \text{红})$$

计算初值

$$\begin{aligned} \alpha_1(1) &= \pi_1 b_1(o_1) = 0.10 \\ \alpha_1(2) &= \pi_2 b_2(o_1) = 0.16 \\ \alpha_1(3) &= \pi_3 b_3(o_1) = 0.28 \end{aligned}$$

递推计算

$$\begin{aligned}
\alpha_2(1) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i1} \right] b_1(o_2) = 0.154 \times 0.5 = 0.077 \\
\alpha_2(2) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i2} \right] b_2(o_2) = 0.184 \times 0.6 = 0.1104 \\
\alpha_2(3) &= \left[\sum_{i=1}^3 \alpha_1(i) a_{i3} \right] b_3(o_2) = 0.202 \times 0.3 = 0.0606 \\
\alpha_3(1) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i1} \right] b_1(o_3) = 0.04187 \\
\alpha_3(2) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i2} \right] b_2(o_3) = 0.03551 \\
\alpha_3(3) &= \left[\sum_{i=1}^3 \alpha_2(i) a_{i3} \right] b_3(o_3) = 0.05284
\end{aligned}$$

终止

后向算法

记后向概率

$$\beta_t(i) = P(o_{t+1}, o_{t+2}, \dots, o_T \mid i_t = q_i, \lambda)$$

初值

$$\beta_T(i) = 1, \quad i = 1, 2, \dots, N$$

从后往前递推

$$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad i = 1, 2, \dots, N$$

终止

$$P(O \mid \lambda) = \sum_{i=1}^N \pi_i b_i(o_1) \beta_1(i)$$

前向后向可以写作统一形式

$$P(O \mid \lambda) = \sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j), \quad t = 1, 2, \dots, T-1$$

一些概率与期望值的计算

给定 λ, O , 在时刻 t 处于状态 q_i

$$\gamma_t(i) = P(i_t = q_i \mid O, \lambda)$$

计算

$$\gamma_t(i) = P(i_t = q_i \mid O, \lambda) = \frac{P(i_t = q_i, O \mid \lambda)}{P(O \mid \lambda)}$$

$$\alpha_t(i)\beta_t(i) = P(i_t = q_i, O \mid \lambda)$$

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O \mid \lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^N \alpha_t(j)\beta_t(j)}$$

$$\xi_t(i, j) = P(i_t = q_i, i_{t+1} = q_j \mid O, \lambda)$$

计算

$$\xi_t(i, j) = \frac{P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}{P(O \mid \lambda)} = \frac{P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}{\sum_{i=1}^N \sum_{j=1}^N P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda)}$$

$$P(i_t = q_i, i_{t+1} = q_j, O \mid \lambda) = \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)$$

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) a_{ij} b_j(o_{t+1}) \beta_{t+1}(j)}$$

学习算法

监督学习

已知

$$\{(O_1, I_1), (O_2, I_2), \dots, (O_s, I_s)\}$$

转移概率估计

$$\hat{a}_{ij} = \frac{A_{ij}}{\sum_{j=1}^N A_{ij}}, \quad i = 1, 2, \dots, N; j = 1, 2, \dots, N$$

观测概率估计

$$\hat{b}_j(k) = \frac{B_{jk}}{\sum_{k=1}^M B_{jk}}, \quad j = 1, 2, \dots, N; k = 1, 2, \dots, M$$

初始状态概率估计：原始状态的频率

非监督学习

已知

$$\{O_1, O_2, \dots, O_s\}$$

求模型参数，实际上就是使用 EM 算法进行参数估计。

$$P(O \mid \lambda) = \sum_I P(O \mid I, \lambda)P(I \mid \lambda)$$

完全数据

$$(O, I) = (o_1, o_2, \dots, o_T, i_1, i_2, \dots, i_T)$$

其对数似然函数

$$\log P(O, I \mid \lambda)$$

Baum Welch 算法

E 步

$$Q(\lambda, \bar{\lambda}) = \sum_I \log P(O, I \mid \lambda)P(O, I \mid \bar{\lambda})$$

$$P(O, I \mid \lambda) = \pi_{i1} b_{i1}(o_1) a_{i1i2} b_{i2}(o_2) \cdots a_{iT-1iT} b_{iT}(o_T)$$

那么有

$$\begin{aligned} Q(\lambda, \bar{\lambda}) &= \sum_I \log \pi_i P(O, I \mid \bar{\lambda}) \\ &+ \sum_I \left(\sum_{t=1}^{T-1} \log a_{iit+1} \right) P(O, I \mid \bar{\lambda}) + \sum_I \left(\sum_{t=1}^T \log b_{it}(o_t) \right) P(O, I \mid \bar{\lambda}) \end{aligned}$$

M 步

对第一项

$$\sum_I \log \pi_{i0} P(O, I \mid \bar{\lambda}) = \sum_{i=1}^N \log \pi_i P(O, i_1 = i \mid \bar{\lambda})$$

有约束

$$\sum_{i=1}^N \pi_i = 1$$

因此使用拉格朗日乘子

$$\sum_{i=1}^N \log \pi_i P(O, i_1 = i \mid \bar{\lambda}) + \gamma \left(\sum_{i=1}^N \pi_i - 1 \right)$$

$$\frac{\partial}{\partial \pi_i} \left[\sum_{i=1}^N \log \pi_i P(O, i_1 = i \mid \bar{\lambda}) + \gamma \left(\sum_{i=1}^N \pi_i - 1 \right) \right] = 0$$

得到

$$P(O, i_1 = i \mid \bar{\lambda}) + \gamma \pi_i = 0$$

$$\gamma = -P(O \mid \bar{\lambda})$$

$$\pi_i = \frac{P(O, i_1 = i \mid \bar{\lambda})}{P(O \mid \bar{\lambda})}$$

第二项

$$\sum_I \left(\sum_{t=1}^{T-1} \log a_{i,i_{t+1}} \right) P(O, I \mid \bar{\lambda}) = \sum_{i=1}^N \sum_{j=1}^N \sum_{t=1}^{T-1} \log a_{ij} P(O, i_t = i, i_{t+1} = j \mid \bar{\lambda})$$

有约束

$$\sum_{j=1}^N a_{ij} = 1$$

同样利用拉格朗日乘子得到

$$a_{ij} = \frac{\sum_{t=1}^{T-1} P(O, i_t = i, i_{t+1} = j \mid \bar{\lambda})}{\sum_{t=1}^{T-1} P(O, i_t = i \mid \bar{\lambda})}$$

第三项

$$\sum_I \left(\sum_{t=1}^T \log b_{i_t}(o_t) \right) P(O, I \mid \bar{\lambda}) = \sum_{j=1}^N \sum_{t=1}^T \log b_j(o_t) P(O, i_t = j \mid \bar{\lambda})$$

有约束

$$\sum_{k=1}^M b_j(k) = 1$$

得到

$$b_j(k) = \frac{\sum_{t=1}^T P(O, i_t = j \mid \bar{\lambda}) I(o_t = v_k)}{\sum_{t=1}^T P(O, i_t = j \mid \bar{\lambda})}$$

将概率用 $\gamma_t(i)$, $\xi_t(i, j)$ 表示

$$a_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$b_j(k) = \frac{\sum_{t=1, o_t=v_k}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}$$

$$\pi_i = \gamma_1(i)$$

最后得到了我们想要的模型参数。

预测算法

近似算法

维特比算法

用动态规划解概率最大路径，一个路径对应一个状态序列。

最优路径具有这样的特性：如果最优路径在时刻 t 通过结点 i_t^* ，那么这一路径从结点 i_t^* 到终点 i_T^* 的部分路径，对于从 i_t^* 到 i_T^* 的所有可能的部分路径来说，必须是最优的。

只需从时刻 $t = 1$ 开始，递推地计算在时刻 t 状态为 i 的各条部分路径的最大概率，直至得到时刻 $t = T$ 状态为 i 的各条路径的最大概率，时刻 $t = T$ 的最大概率即为最优路径的概率 P^* ，最优路径的终结点 i_T^* 也同时得到。

之后，为了找出最优路径的各个结点，从终结点开始，由后向前逐步求得结点 i_{T-1}^*, \dots, i_i^* ，得到最优路径：

$$I^* = (i_1^*, i_2^*, \dots, i_T^*)$$

导入两个变量 δ 和 ψ ，定义在时刻 t 状态为 i 的所有单个路径 (i_1, i_2, \dots, i_t) 中概率最大值为：

$$\delta_t(i) = \max_{i_1, i_2, \dots, i_{t-1}} P(i_t = i, i_{t-1}, \dots, i_1, o_t, \dots, o_1 \mid \lambda), \quad i = 1, 2, \dots, N$$

由定义可得变量 δ 的递推公式：

$$\begin{aligned} \delta_{t+1}(i) &= \max_{i_1, i_2, \dots, i_t} P(i_{t+1} = i, i_t, \dots, i_1, o_{t+1}, \dots, o_1 \mid \lambda) \\ &= \max_{1 \leq j \leq N} [\delta_t(j) a_{ji}] b_i(o_{t+1}), \quad i = 1, 2, \dots, N; t = 1, 2, \dots, T-1 \end{aligned}$$

定义在时刻 t 状态为 i 的所有单个路径 $(i_1, i_2, \dots, i_{t-1}, i)$ 中概率最大的路径的第 $t-1$ 个节点为：

$$\psi_t(i) = \arg \max_{1 \leq j \leq N} [\delta_{t-1}(j) a_{ji}], \quad i = 1, 2, \dots, N$$

输入：模型 $\lambda = (A, B, \pi)$ ，观测 $O = (o_1, o_2, \dots, o_T)$

输出：最优路径 $I^* = (i_1^*, i_2^*, \dots, i_T^*)$

1. 初始化

$$\delta_1(i) = \pi_i b_i(o_1), \quad i = 1, 2, \dots, N$$

$$\Psi_1(i) = 0, \quad i = 1, 2, \dots, N$$

2. 递推 ($t = 2, 3, \dots, T$)

$$\delta_t(i) = \max_{1 \leq j \leq N} [\delta_{t-1}(j)a_{ji}] b_i(o_t), \quad i = 1, 2, \dots, N$$

$$\Psi_t(i) = \arg \max_{1 \leq j \leq N} [\delta_{t-1}(j)a_{ji}], \quad i = 1, 2, \dots, N$$

3. 终止

$$P^* = \max_{1 \leq i \leq N} \delta_T(i)$$

$$i_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$$

4. 最优路径回溯 ($t = T-1, T-2, \dots, 1$)

$$i_t^* = \Psi_{t+1}(i_{t+1}^*)$$

5. 求得最优路径

$$I^* = (i_1^*, i_2^*, \dots, i_T^*)$$

要考的

例子

$$A = \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \\ 0.7 & 0.3 \end{bmatrix}, \quad \pi = (0.2, 0.4, 0.4)^T$$

$$O = (\text{红}, \text{白}, \text{红})$$

求最优路径

1. 初始化: 在 $t = 1$ 时, 对每一个状态 i , $i = 1, 2, 3$, 求状态 i 观测 o_1 为红的概率, 记为: $\delta_1(i)$

$$\delta_1(i) = \pi_i b_i(o_1) = \pi_i b_i(\text{红}), i = 1, 2, 3$$

$$\delta_1(1) = 0.10, \quad \delta_1(2) = 0.16, \quad \delta_1(3) = 0.28$$

$$\psi_1(i) = 0, i = 1, 2, 3$$

2. 在 $t = 2$ 时, 对每一个状态 i , $i = 1, 2, 3$, 求在 $t = 1$ 时状态为 j 观测 o_1 为红并在 $t = 2$ 时状态为 i 观测 o_2 为白的路径的最大概率, 记为: $\delta_2(i)$

$$\begin{aligned} \delta_2(1) &= \max_{1 \leq j \leq 3} [\delta_1(j)a_{j1}] b_1(o_2) \\ &= \max_j \{0.10 \times 0.5, 0.16 \times 0.3, 0.28 \times 0.2\} \times 0.5 \\ &= 0.028 \end{aligned}$$

$$\begin{aligned} \psi_2(1) &= 3 \\ \delta_2(2) &= 0.0504, \quad \psi_2(2) = 3 \\ \delta_2(3) &= 0.042, \quad \psi_2(3) = 3 \end{aligned}$$

3. 同样 $t = 3$ 时

$$\delta_3(i) = \max_{1 \leq j \leq 3} [\delta_2(j)a_{ji}] b_i(o_3)$$

$$\psi_3(i) = \arg \max_{1 \leq j \leq 3} [\delta_2(j)a_{ji}]$$

$$\delta_3(1) = 0.00756, \quad \psi_3(1) = 2$$

$$\delta_3(2) = 0.01008, \quad \psi_3(2) = 2$$

$$\delta_3(3) = 0.0147, \quad \psi_3(3) = 3$$

4. 最优路径选择

1

$$P^* = \max_{1 \leq i \leq 3} \delta_3(i) = 0.0147$$

终点为

$$i_3^* = \arg \max_i [\delta_3(i)] = 3$$

依次反向寻找

在 $t = 2$ 时, $i_2^* = \psi_3(i_3^*) = \psi_3(3) = 3$
在 $t = 1$ 时, $i_1^* = \psi_2(i_2^*) = \psi_2(3) = 3$

因此, 最优路径为

1

$$I^* = (i_1^*, i_2^*, i_3^*) = (3, 3, 3)$$

1

1



聚类

真的没什么好讲的，都是前面学过的，连算法都是一模一样。

直接放伪代码

基于原型

K-means

```
输入: 样本集  $D = \{x_1, x_2, \dots, x_m\}$ ;  
聚类簇数  $k$ .  
过程:  
1: 从  $D$  中随机选择  $k$  个样本作为初始均值向量  $\{\mu_1, \mu_2, \dots, \mu_k\}$   
2: repeat  
3:   令  $C_i = \emptyset$  ( $1 \leq i \leq k$ )  
4:   for  $j = 1, \dots, m$  do  
5:     计算样本  $x_j$  与各均值向量  $\mu_i$  ( $1 \leq i \leq k$ ) 的距离:  $d_{ji} = \|x_j - \mu_i\|_2$ ;  
6:     根据距离最近的均值向量确定  $x_j$  的簇标记:  $\lambda_j = \arg \min_{i \in \{1, 2, \dots, k\}} d_{ji}$ ;  
7:     将样本  $x_j$  划入相应的簇:  $C_{\lambda_j} = C_{\lambda_j} \cup \{x_j\}$ ;  
8:   end for  
9:   for  $i = 1, \dots, k$  do  
10:    计算新均值向量:  $\mu'_i = \frac{1}{|C_i|} \sum_{x \in C_i} x$ ;  
11:    if  $\mu'_i \neq \mu_i$  then  
12:      将当前均值向量  $\mu_i$  更新为  $\mu'_i$   
13:    else  
14:      保持当前均值向量不变  
15:    end if  
16:   end for  
17: until 当前均值向量均未更新  
18: return 簇划分结果  
输出: 簇划分  $C = \{C_1, C_2, \dots, C_k\}$ 
```

LVQ

```
输入: 样本集  $D = \{(x_1, y_1), \dots, (x_m, y_m)\}$ ;  
原型向量个数  $q$ , 各原型向量预设的类别标记  $\{t_1, t_2, \dots, t_q\}$ ;  
学习率  $\eta \in (0, 1)$ .  
过程:  
1: 初始化一组原型向量  $\{p_1, p_2, \dots, p_q\}$   
2: repeat  
3:   从样本集  $D$  随机选取样本  $(x_j, y_j)$ ;  
4:   计算样本  $x_j$  与  $p_i$  ( $1 \leq i \leq q$ ) 的距离:  $d_{ji} = \|x_j - p_i\|_2$ ;  
5:   找出与  $x_j$  距离最近的原型向量:  $i^* = \arg \min_{i \in \{1, 2, \dots, q\}} d_{ji}$ ;  
6:   if  $y_j = t_{i^*}$  then  
7:      $p' = p_{i^*} + \eta \cdot (x_j - p_{i^*})$   
8:   else  
9:      $p' = p_{i^*} - \eta \cdot (x_j - p_{i^*})$   
10:  end if  
11:  将原型向量  $p_{i^*}$  更新为  $p'$   
12: until 满足停止条件  
13: return 当前原型向量  
输出: 原型向量  $\{p_1, p_2, \dots, p_q\}$ 
```

GMM

输入: 样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$;
 高斯混合成分个数 k .
 过程:
 1: 初始化高斯混合分布的模型参数 $\{(\alpha_i, \mu_i, \Sigma_i) \mid 1 \leq i \leq k\}$
 2: **repeat**
 3: **for** $j = 1, \dots, m$ **do**
 4: 根据(9.30)计算 \mathbf{x}_j 由各混合成分生成的后验概率, 即

$$\gamma_{ji} = p_{\mathcal{M}}(z_j = i \mid \mathbf{x}_j) \quad (1 \leq i \leq k)$$

 5: **end for**
 6: **for** $i = 1, \dots, k$ **do**
 7: 计算新均值向量: $\mu'_i = \frac{\sum_{j=1}^m \gamma_{ji} \mathbf{x}_j}{\sum_{j=1}^m \gamma_{ji}}$;
 8: 计算新协方差矩阵: $\Sigma'_i = \frac{\sum_{j=1}^m \gamma_{ji} (\mathbf{x}_j - \mu'_i)(\mathbf{x}_j - \mu'_i)^\top}{\sum_{j=1}^m \gamma_{ji}}$;
 9: 计算新混合系数: $\alpha'_i = \frac{\sum_{j=1}^m \gamma_{ji}}{m}$;
 10: **end for**
 11: 将模型参数 $\{(\alpha_i, \mu_i, \Sigma_i) \mid 1 \leq i \leq k\}$ 更新为 $\{(\alpha'_i, \mu'_i, \Sigma'_i) \mid 1 \leq i \leq k\}$
 12: **until** 满足停止条件
 13: $C_i = \emptyset \quad (1 \leq i \leq k)$
 14: **for** $j = 1, \dots, m$ **do**
 15: 根据(9.31)确定 \mathbf{x}_j 的簇标记 λ_j ;
 16: 将 \mathbf{x}_j 划入相应的簇: $C_{\lambda_j} = C_{\lambda_j} \cup \{\mathbf{x}_j\}$
 17: **end for**
 18: **return** 簇划分结果
 输出: 簇划分 $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$

基于密度

DBSCAN

输入: 样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$;
 邻域参数 $(\epsilon, \text{MinPts})$.
 过程:
 1: 初始化核心对象集合: $\Omega = \emptyset$
 2: **for** $j = 1, \dots, m$ **do**
 3: 确定样本 \mathbf{x}_j 的 ϵ -邻域 $N_\epsilon(\mathbf{x}_j)$;
 4: **if** $|N_\epsilon(\mathbf{x}_j)| \geq \text{MinPts}$ **then**
 5: 将样本 \mathbf{x}_j 加入核心对象集合: $\Omega = \Omega \cup \{\mathbf{x}_j\}$
 6: **end if**
 7: **end for**
 8: 初始化聚类簇数: $k = 0$
 9: 初始化未访问样本集合: $\Gamma = D$
 10: **while** $\Omega \neq \emptyset$ **do**
 11: 记录当前未访问样本集合: $\Gamma_{\text{old}} = \Gamma$;
 12: 随机选取一个核心对象 $\mathbf{o} \in \Omega$, 初始化队列 $Q = < \mathbf{o} >$;
 13: $\Gamma = \Gamma \setminus \{\mathbf{o}\}$;
 14: **while** $Q \neq \emptyset$ **do**
 15: 取出队列 Q 中的首个样本 \mathbf{q} ;
 16: **if** $|N_\epsilon(\mathbf{q})| \geq \text{MinPts}$ **then**
 17: 令 $\Delta = N_\epsilon(\mathbf{q}) \cap \Gamma$;
 18: 将 Δ 中的样本加入队列 Q ;
 19: $\Gamma = \Gamma \setminus \Delta$;
 20: **end if**
 21: **end while**
 22: $k = k + 1$, 生成聚类簇 $C_k = \Gamma_{\text{old}} \setminus \Gamma$;
 23: $\Omega = \Omega \setminus C_k$
 24: **end while**
 25: **return** 簇划分结果
 输出: 簇划分 $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$

基于层次

AGNES

输入: 样本集 $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m\}$;
聚类簇距离度量函数 $d \in \{d_{\min}, d_{\max}, d_{\text{avg}}\}$;
聚类簇数 k .

过程:

- 1: **for** $j = 1, \dots, m$ **do**
- 2: $C_j = \{\mathbf{x}_j\}$
- 3: **end for**
- 4: **for** $i = 1, \dots, m$ **do**
- 5: **for** $j = i, \dots, m$ **do**
- 6: $M(i, j) = d(C_i, C_j)$;
- 7: $M(j, i) = M(i, j)$
- 8: **end for**
- 9: **end for**
- 10: 设置当前聚类簇个数: $q = m$
- 11: **while** $q > k$ **do**
- 12: 找出距离最近的两个聚类簇 (C_{i^*}, C_{j^*}) ;
- 13: 合并 (C_{i^*}, C_{j^*}) : $C_{i^*} = C_{i^*} \cup C_{j^*}$;
- 14: **for** $j = j^* + 1, \dots, q$ **do**
- 15: 将聚类簇 C_j 重编号为 C_{j-1}
- 16: **end for**
- 17: 删除距离矩阵 M 的第 j^* 行与第 j^* 列;
- 18: **for** $j = 1, \dots, q-1$ **do**
- 19: $M(i^*, j) = d(C_{i^*}, C_j)$;
- 20: $M(j, i^*) = M(i^*, j)$
- 21: **end for**
- 22: $q = q - 1$
- 23: **end while**
- 24: **return** 簇划分结果

输出: 簇划分 $\mathcal{C} = \{C_1, C_2, \dots, C_k\}$



降维与度量学习

K 近邻

多维缩放

主成分分析

流形学习

度量学习