第五章 自变量的选择

Tianxiao Pang

Zhejiang University

November 19, 2019

内容

内容

内容

在第三章和第四章我们讨论了线性回归模型的估计方法和假设检验问题, 但应用回归分析处理实际问题时, 首先要解决的问题是模型的选择(model selection). 所谓模型的选择包含两方面的内容.

一是选择回归模型的类型, 即判断是用线性回归模型还是非线性回归模型来处理实际问题. 统计学上称之为回归模型的线性检验, 在有重复试验的情况下可以使用卡方拟合优度检验来处理这个问题. 本课程不讨论这部分内容.

二是在选定模型后, 自变量的选择问题(variable selection). 自变量选择过少或选择不当, 会使所建立的方程与实际有较大的偏离而无法使用. 自变量选择过多, 其后果是计算量增大、估计和预测的精度也会下降(见定理5.1.1和定理5.1.2).

自变量选择的后果

假设根据经验和专业知识, 初步确定一切可能对因变量y有影响的自变量共有p个, 记为$x_1, \cdots, x_p$. 相应的线性回归模型(矩阵形式)为

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{e}, \ \ \mathsf{E}(\boldsymbol{e}) = \boldsymbol{0}, \ \mathsf{Cov}(\boldsymbol{e}) = \sigma^2 \boldsymbol{I}_n, \qquad (5.1.1)$$

这里$\boldsymbol{X}$为$n \times (p+1)$的列满秩设计矩阵, 第一列元素全为1. 我们称(5.1.1)为全模型.

假设我们根据某些自变量选择的准则, 剔除了(5.1.1)中的一些对因变量影响较小的自变量, 不妨假设剔除了后$p-q$个自变量x_{q+1}, $\cdots$, x_p. 记

$$\begin{pmatrix} \boldsymbol{x}_1^{'} \\ \vdots \\ \boldsymbol{x}_n^{'} \end{pmatrix} = \boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_q & \boldsymbol{X}_t \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_{1q}^{'} & \boldsymbol{x}_{1t}^{'} \\ \vdots & \vdots \\ \boldsymbol{x}_{nq}^{'} & \boldsymbol{x}_{nt}^{'} \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \boldsymbol{\beta}_q \\ \boldsymbol{\beta}_t \end{pmatrix}.$$

则我们得到一个新模型

$$\boldsymbol{Y} = \boldsymbol{X}_q\boldsymbol{\beta}_q + \boldsymbol{e}, \ \ \mathsf{E}(\boldsymbol{e}) = \boldsymbol{0}, \ \mathsf{Cov}(\boldsymbol{e}) = \sigma^2\boldsymbol{I}_n, \qquad (5.1.2)$$

这里$\boldsymbol{X}_q$为$n \times (q+1)$的列满秩设计矩阵, $\boldsymbol{\beta}_q$为$q+1$维的列向量. 我们称(5.1.2)为选模型.

在全模型假设下, 回归系数$\boldsymbol{\beta}$和σ^2的最小二乘估计为

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{Y},\ \hat{\sigma}^2 = \frac{\boldsymbol{Y}'[\boldsymbol{I}_n - \boldsymbol{X}(\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}']\boldsymbol{Y}}{n-p-1}, \quad (5.1.3)$$

在$\boldsymbol{x}_0' = (\boldsymbol{x}_{0q}' \ \ \boldsymbol{x}_{0t}')$点上的预测为$\hat{y}_0 = \boldsymbol{x}_0'\hat{\boldsymbol{\beta}}$.

在选模型假设下, 回归系数$\boldsymbol{\beta}_q$和σ^2的最小二乘估计为

$$\widetilde{\boldsymbol{\beta}}_q = (\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q{}'\boldsymbol{Y},\ \widetilde{\sigma}_q^2 = \frac{\boldsymbol{Y}'[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q']\boldsymbol{Y}}{n-q-1} \quad (5.1.4)$$

在$\boldsymbol{x}_0' = (\boldsymbol{x}_{0q}' \ \ \boldsymbol{x}_{0t}')$点上的预测为$\widetilde{y}_{0q} = \boldsymbol{x}_{0q}'\widetilde{\boldsymbol{\beta}}_q$.

对$\hat{\boldsymbol{\beta}}$作相应的分块:

$$\hat{\boldsymbol{\beta}} = \begin{pmatrix} \hat{\boldsymbol{\beta}}_q \\ \hat{\boldsymbol{\beta}}_t \end{pmatrix}.$$

若$\widetilde{\boldsymbol{\theta}}$是未知参数$\boldsymbol{\theta}$的有偏估计, 那么协方差矩阵不能作为衡量估计精度之用, 更合理的度量标准为均方误差矩阵(MSEM: Mean Square Error Matrix).

定义

设$\boldsymbol{\theta}$是一列向量. $\widetilde{\boldsymbol{\theta}}$为$\boldsymbol{\theta}$的一个估计. 定义$\widetilde{\boldsymbol{\theta}}$的均方误差矩阵为

$$MSEM(\widetilde{\boldsymbol{\theta}}) = E[(\widetilde{\boldsymbol{\theta}} - \boldsymbol{\theta})(\widetilde{\boldsymbol{\theta}} - \boldsymbol{\theta})'].$$

不难推得:

$$\mathrm{MSEM}(\widetilde{\boldsymbol{\theta}}) = \mathrm{Cov}(\widetilde{\boldsymbol{\theta}}) + (\mathrm{E}(\widetilde{\boldsymbol{\theta}}) - \boldsymbol{\theta})(\mathrm{E}(\widetilde{\boldsymbol{\theta}}) - \boldsymbol{\theta})'. \qquad (5.1.5)$$

回忆分块矩阵求逆公式:

引理 (分块矩阵求逆公式)

设$\boldsymbol{A}$为非奇异的对称矩阵, 将其分块为

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{B} & \boldsymbol{C} \\ \boldsymbol{C}' & \boldsymbol{D} \end{pmatrix},$$

则当$\boldsymbol{B}^{-1}, \boldsymbol{D}^{-1}$都存在时有

$$\begin{aligned} \boldsymbol{A}^{-1} &= \begin{pmatrix} \boldsymbol{B}_1 & \boldsymbol{C}_1 \\ \boldsymbol{C}_1' & \boldsymbol{D}_1 \end{pmatrix} \\ &= \begin{pmatrix} (\boldsymbol{B} - \boldsymbol{C}\boldsymbol{D}^{-1}\boldsymbol{C}')^{-1} & -\boldsymbol{B}_1\boldsymbol{C}\boldsymbol{D}^{-1} \\ -\boldsymbol{D}^{-1}\boldsymbol{C}'\boldsymbol{B}_1 & \boldsymbol{D}^{-1} + \boldsymbol{D}^{-1}\boldsymbol{C}'\boldsymbol{B}_1\boldsymbol{C}\boldsymbol{D}^{-1} \end{pmatrix} \\ &= \begin{pmatrix} \boldsymbol{B}^{-1} + \boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1\boldsymbol{C}'\boldsymbol{B}^{-1} & -\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1 \\ -\boldsymbol{D}_1\boldsymbol{C}'\boldsymbol{B}^{-1} & (\boldsymbol{D} - \boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{C})^{-1} \end{pmatrix}. \end{aligned}$$

定理 (5.1.1, 对估计的影响)

假设全模型*(5.1.1)*正确, 则
(1)$E(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$; $E(\widetilde{\boldsymbol{\beta}}_q) = \boldsymbol{\beta}_q + \boldsymbol{G}\boldsymbol{\beta}_t$, 这里$\boldsymbol{G} = (\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q'\boldsymbol{X}_t$, 所以除了$\boldsymbol{\beta}_t = \boldsymbol{0}$或者$\boldsymbol{X}_q'\boldsymbol{X}_t = \boldsymbol{0}$外, $E(\widetilde{\boldsymbol{\beta}}_q) \neq \boldsymbol{\beta}_q$;
(2)$Cov(\hat{\boldsymbol{\beta}}_q) - Cov(\widetilde{\boldsymbol{\beta}}_q)$为非负定矩阵;
*(3)*当$Cov(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵时, $MSEM(\hat{\boldsymbol{\beta}}_q) - MSEM(\widetilde{\boldsymbol{\beta}}_q)$为非负定矩阵;
(4)$E(\widetilde{\sigma}_q^2) \geq E(\hat{\sigma}^2)$, 仅当$\boldsymbol{\beta}_t = \boldsymbol{0}$时等号成立.

证明: (1)$\mathsf{E}(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta}$是显然的. 现来考察$\widetilde{\boldsymbol{\beta}}_q$的均值. 根据(5.1.4),

$$\begin{aligned}
\mathsf{E}(\widetilde{\boldsymbol{\beta}}_q) &= (\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q{}'\mathsf{E}(\boldsymbol{Y}) \\
&= (\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q{}'\begin{pmatrix}\boldsymbol{X}_q & \boldsymbol{X}_t\end{pmatrix}\begin{pmatrix}\boldsymbol{\beta}_q \\ \boldsymbol{\beta}_t\end{pmatrix} \\
&= \begin{pmatrix}\boldsymbol{I}_{q+1} & \boldsymbol{G}\end{pmatrix}\begin{pmatrix}\boldsymbol{\beta}_q \\ \boldsymbol{\beta}_t\end{pmatrix} \\
&= \boldsymbol{\beta}_q + \boldsymbol{G}\boldsymbol{\beta}_t,
\end{aligned}$$

不难看出, 除了$\boldsymbol{\beta}_t = \boldsymbol{0}$或者$\boldsymbol{X}_q'\boldsymbol{X}_t = \boldsymbol{0}$外, $\mathsf{E}(\widetilde{\boldsymbol{\beta}}_q) \neq \boldsymbol{\beta}_q$.

(2) 记

$$\boldsymbol{X}'\boldsymbol{X} = \begin{pmatrix} \boldsymbol{X}_q' \\ \boldsymbol{X}_t' \end{pmatrix} (\boldsymbol{X}_q \ \ \boldsymbol{X}_t) = \begin{pmatrix} \boldsymbol{X}_q'\boldsymbol{X}_q & \boldsymbol{X}_q'\boldsymbol{X}_t \\ \boldsymbol{X}_t'\boldsymbol{X}_q & \boldsymbol{X}_t'\boldsymbol{X}_t \end{pmatrix} \triangleq \begin{pmatrix} \boldsymbol{B} & \boldsymbol{C} \\ \boldsymbol{C}' & \boldsymbol{D} \end{pmatrix},$$

这里$\boldsymbol{B} = \boldsymbol{X}_q'\boldsymbol{X}_q$, $\boldsymbol{C} = \boldsymbol{X}_q'\boldsymbol{X}_t$, $\boldsymbol{D} = \boldsymbol{X}_t'\boldsymbol{X}_t$. 又记

$$(\boldsymbol{X}'\boldsymbol{X})^{-1} = \begin{pmatrix} \boldsymbol{B}_1 & \boldsymbol{C}_1 \\ \boldsymbol{C}_1' & \boldsymbol{D}_1 \end{pmatrix}.$$

由

$$\mathsf{Cov}(\hat{\boldsymbol{\beta}}) = \mathsf{Cov}\begin{pmatrix} \hat{\boldsymbol{\beta}}_q \\ \hat{\boldsymbol{\beta}}_t \end{pmatrix} = \sigma^2(\boldsymbol{X}'\boldsymbol{X})^{-1} = \sigma^2 \begin{pmatrix} \boldsymbol{B}_1 & \boldsymbol{C}_1 \\ \boldsymbol{C}_1' & \boldsymbol{D}_1 \end{pmatrix}$$

知$\mathsf{Cov}(\hat{\boldsymbol{\beta}}_q) = \sigma^2\boldsymbol{B}_1$. 又$\mathsf{Cov}(\widetilde{\boldsymbol{\beta}}_q) = \sigma^2(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1} = \sigma^2\boldsymbol{B}^{-1}$, 所以

$$\mathsf{Cov}(\hat{\boldsymbol{\beta}}_q) - \mathsf{Cov}(\widetilde{\boldsymbol{\beta}}_q) = \sigma^2(\boldsymbol{B}_1 - \boldsymbol{B}^{-1}) = \sigma^2\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1\boldsymbol{C}'\boldsymbol{B}^{-1}$$

为非负定矩阵.

(3)由公式(5.1.5)以及结论(1)可知

$$\mathrm{MSEM}(\widetilde{\boldsymbol{\beta}}_q) = \sigma^2(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1} + \boldsymbol{G}\boldsymbol{\beta}_t\boldsymbol{\beta}_t{}'\boldsymbol{G}' = \sigma^2\boldsymbol{B}^{-1} + \boldsymbol{G}\boldsymbol{\beta}_t\boldsymbol{\beta}_t{}'\boldsymbol{G}',$$
$$\mathrm{MSEM}(\hat{\boldsymbol{\beta}}_q) = \sigma^2\boldsymbol{B}_1.$$

注意到$\boldsymbol{G} = \boldsymbol{B}^{-1}\boldsymbol{C}$, 所以当$\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵时,

$$\begin{aligned}
& \mathrm{MSEM}(\hat{\boldsymbol{\beta}}_q) - \mathrm{MSEM}(\widetilde{\boldsymbol{\beta}}_q) \\
= \;& \sigma^2\boldsymbol{B}_1 - \sigma^2\boldsymbol{B}^{-1} - \boldsymbol{G}\boldsymbol{\beta}_t\boldsymbol{\beta}_t{}'\boldsymbol{G}' \\
= \;& \sigma^2\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1\boldsymbol{C}'\boldsymbol{B}^{-1} - \boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{\beta}_t\boldsymbol{\beta}_t{}'\boldsymbol{C}'\boldsymbol{B}^{-1} \\
= \;& \boldsymbol{B}^{-1}\boldsymbol{C}(\sigma^2\boldsymbol{D}_1 - \boldsymbol{\beta}_t\boldsymbol{\beta}_t{}')\boldsymbol{C}'\boldsymbol{B}^{-1} \\
= \;& \boldsymbol{B}^{-1}\boldsymbol{C}(\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t')\boldsymbol{C}'\boldsymbol{B}^{-1}
\end{aligned}$$

为非负定矩阵.

(4)$\mathsf{E}(\hat{\sigma}^2) = \sigma^2$是已知的. 而

$$
\begin{aligned}
\mathsf{E}(\widetilde{\sigma}_q^2) &= \frac{1}{n-p-1}\mathsf{E}\Big\{\boldsymbol{Y}^{'}[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}]\boldsymbol{Y}\Big\} \\
&= \frac{1}{n-q-1}\mathsf{tr}\Big\{[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}]\mathsf{E}(\boldsymbol{Y}\boldsymbol{Y}^{'})\Big\} \\
&= \frac{1}{n-q-1}\mathsf{tr}\Big\{[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}](\sigma^2\boldsymbol{I}_n + \boldsymbol{X}\boldsymbol{\beta}\boldsymbol{\beta}^{'}\boldsymbol{X}^{'})\Big\} \\
&= \frac{1}{n-q-1}\Big\{(n-q-1)\sigma^2 + \boldsymbol{\beta}^{'}\boldsymbol{X}^{'}[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}]\boldsymbol{X}\boldsymbol{\beta}\Big\} \\
&= \sigma^2 + \frac{1}{n-q-1}\boldsymbol{\beta}_t^{'}\boldsymbol{X}_t^{'}[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}]\boldsymbol{X}_t\boldsymbol{\beta}_t \\
&= \sigma^2 + \frac{1}{n-q-1}\boldsymbol{\beta}_t^{'}(\boldsymbol{D} - \boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{C})\boldsymbol{\beta}_t \\
&= \sigma^2 + \frac{1}{n-q-1}\boldsymbol{\beta}_t^{'}\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_t \\
&\geq \sigma^2 = \mathsf{E}(\hat{\sigma}^2).
\end{aligned}
$$

记全模型下的预测偏差为$z_0 = y_0 - \hat{y}_0 = y_0 - \boldsymbol{x}_0'\hat{\boldsymbol{\beta}}$, 选模型下的预测偏差为$z_{0q} = y_0 - \widetilde{y}_{0q} = y_0 - \boldsymbol{x}_{0q}'\widetilde{\boldsymbol{\beta}}_q$.

定理 (5.1.2, 对预测的影响)

假设全模型*(5.1.1)*正确, 则
(1)$E(z_0) = 0$, $E(z_{0q}) = \boldsymbol{x}_{0t}'\boldsymbol{\beta}_t - \boldsymbol{x}_{0q}'\boldsymbol{G}\boldsymbol{\beta}_t$, 所以一般情形下, $\widetilde{y}_{0q}$为有偏预测;
(2)$Var(z_0) \geq Var(z_{0q})$;
*(3)*当$Cov(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵时, $MSE(\hat{y}_0) - MSE(\widetilde{y}_{0q}) \geq 0$.

证明: (1) $\mathsf{E}(z_0) = 0$是显然的. 现考察$\mathsf{E}(z_{0q})$. 由定理5.1.1中的结论(1)可知

$$\begin{aligned}
\mathsf{E}(z_{0q}) &= \boldsymbol{x}_0'\boldsymbol{\beta} - \boldsymbol{x}_{0q}'\mathsf{E}(\widetilde{\boldsymbol{\beta}}_q) \\
&= \boldsymbol{x}_0'\boldsymbol{\beta} - \boldsymbol{x}_{0q}'(\boldsymbol{\beta}_q + \boldsymbol{G}\boldsymbol{\beta}_t) \\
&= \boldsymbol{x}_{0t}'\boldsymbol{\beta}_t - \boldsymbol{x}_{0q}'\boldsymbol{G}\boldsymbol{\beta}_t,
\end{aligned}$$

所以一般情形下$\widetilde{y}_{0q}$是有偏预测.

(2) 首先, 容易看出

$$\mathsf{Var}(z_0) = \mathsf{Var}(y_0 - \hat{y}_0) = \mathsf{Var}(y_0) + \mathsf{Var}(\hat{y}_0) = \sigma^2(1 + \boldsymbol{x}_0^{'}(\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{x}_0).$$

而

$$\begin{aligned} \mathsf{Var}(z_{0q}) &= \mathsf{Var}(y_0 - \widetilde{y}_{0q}) = \mathsf{Var}(y_0) + \mathsf{Var}(\widetilde{y}_{0q}) \\ &= \sigma^2(1 + \boldsymbol{x}_{0q}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{0q}). \end{aligned}$$

注意到

$$\boldsymbol{x}_{0q}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{0q} = \boldsymbol{x}_{0q}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q}$$

以及

$$\begin{aligned} \boldsymbol{x}_0^{'}(\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{x}_0 &= (\boldsymbol{x}_{0q}^{'} \ \ \boldsymbol{x}_{0t}^{'}) \begin{pmatrix} \boldsymbol{B}_1 & \boldsymbol{C}_1 \\ \boldsymbol{C}_1^{'} & \boldsymbol{D}_1 \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_{0q} \\ \boldsymbol{x}_{0t} \end{pmatrix} \\ &= \boldsymbol{x}_{0q}^{'}\boldsymbol{B}_1\boldsymbol{x}_{0q} + \boldsymbol{x}_{0q}^{'}\boldsymbol{C}_1\boldsymbol{x}_{0t} + \boldsymbol{x}_{0t}^{'}\boldsymbol{C}_1^{'}\boldsymbol{x}_{0q} + \boldsymbol{x}_{0t}^{'}\boldsymbol{D}_1\boldsymbol{x}_{0t}, \end{aligned}$$

马上推得

$$
\begin{aligned}
& \mathrm{Var}(z_0) - \mathrm{Var}(z_{0q}) \\
= \; & \sigma^2[\boldsymbol{x}_0^{'}(\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{x}_0 - \boldsymbol{x}_{0q}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{0q}] \\
= \; & \sigma^2[\boldsymbol{x}_{0q}^{'}(\boldsymbol{B}_1 - \boldsymbol{B}^{-1})\boldsymbol{x}_{0q} + \boldsymbol{x}_{0q}^{'}\boldsymbol{C}_1\boldsymbol{x}_{0t} + \boldsymbol{x}_{0t}^{'}\boldsymbol{C}_1^{'}\boldsymbol{x}_{0q} + \boldsymbol{x}_{0t}^{'}\boldsymbol{D}_1\boldsymbol{x}_{0t}] \\
= \; & \sigma^2[\boldsymbol{x}_{0q}^{'}\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} \\
& \quad -\boldsymbol{x}_{0q}^{'}\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1\boldsymbol{x}_{0t} - \boldsymbol{x}_{0t}^{'}\boldsymbol{D}_1\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} + \boldsymbol{x}_{0t}^{'}\boldsymbol{D}_1\boldsymbol{x}_{0t}] \\
= \; & \sigma^2[\boldsymbol{x}_{0q}^{'}\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{D}_1(\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t}) - \boldsymbol{x}_{0t}^{'}\boldsymbol{D}_1(\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})] \\
= \; & \sigma^2(\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})^{'}\boldsymbol{D}_1(\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t}) \geq 0.
\end{aligned}
$$

(3) 首先, 容易看出

$$\begin{aligned}
&\mathrm{MSE}(\hat{y}_0) = \mathrm{E}(\hat{y}_0 - y_0)^2 = \mathrm{E}(z_0^2) = \mathrm{Var}(z_0),\\
&\mathrm{MSE}(\widetilde{y}_{0q}) = \mathrm{E}(\widetilde{y}_{0q} - y_0)^2 = \mathrm{E}(z_{0q}^2) = \mathrm{Var}(z_{0q}) + [\mathrm{E}(z_{0q})]^2.
\end{aligned}$$

由(1)的证明可得

$$\begin{aligned}
[\mathrm{E}(z_{0q})]^2 &= (\boldsymbol{x}_{0t}'\boldsymbol{\beta}_t - \boldsymbol{x}_{0q}'\boldsymbol{G}\boldsymbol{\beta}_t)^2\\
&= (\boldsymbol{x}_{0t}' - \boldsymbol{x}_{0q}'\boldsymbol{G})\boldsymbol{\beta}_t\boldsymbol{\beta}_t'(\boldsymbol{x}_{0t}' - \boldsymbol{x}_{0q}'\boldsymbol{G})'\\
&= (\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})'\boldsymbol{\beta}_t\boldsymbol{\beta}_t'(\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t}).
\end{aligned}$$

所以当$\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵时, 根据(2)的证明过程可知

$$\begin{aligned}
&\mathrm{MSE}(\hat{y}_0) - \mathrm{MSE}(\widetilde{y}_{0q}) = \mathrm{Var}(z_0) - \mathrm{Var}(z_{0q}) - [\mathrm{E}(z_{0q})]^2\\
=\ &(\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})'(\sigma^2\boldsymbol{D}_1 - \boldsymbol{\beta}_t\boldsymbol{\beta}_t')(\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})\\
=\ &(\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t})'(\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t')(\boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{x}_{0q} - \boldsymbol{x}_{0t}) \geq 0.
\end{aligned}$$

总结:

(1) 即使全模型正确, 剔除一部分自变量后, 可使得剩余的那部分自变量的回归系数的LSE的方差减少, 但此时的估计一般为有偏估计, 若被剔除的自变量对因变量影响较小或难于掌握(用$\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵来刻画), 则剔除这些自变量后可使得剩余自变量的回归系数的LSE的精度(用均方误差来刻画)有所提高.

(2) 当全模型正确时, 用选模型作预测, 则预测一般是有偏的, 但预测偏差的方差减小. 若被剔除的自变量对因变量影响较小或难于掌握(用$\mathrm{Cov}(\hat{\boldsymbol{\beta}}_t) - \boldsymbol{\beta}_t\boldsymbol{\beta}_t'$为非负定矩阵来刻画), 则剔除这些自变量后可使得预测的精度(用均方误差来刻画)有所提高.

因此在应用回归分析去处理实际问题时, 无论从回归系数的估计的角度看, 还是从预测的角度看, 对那些与因变量关系不大或难于掌握的自变量从模型中剔除都是有利的. 回归方程中自变量的选择要做到少而精.

自变量选择的准则

统计学家从数据与模型的拟合程度、预测精度等不同角度出发提出了多种回归自变量的选择准则, 它们都是对回归自变量的所有不同子集进行比较, 然后从中挑选一个"最优"的, 且绝大部分选择准则都是与残差平方和有关. 但是我们不能直接把残差平方和当成是自变量选择的一个准则, 理由如下.

记选模型(5.1.2)的残差平方和为RSS_q, 则

$$\mathrm{RSS}_q = \boldsymbol{Y}^{'}[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}]\boldsymbol{Y},$$

其中

$$\boldsymbol{X}_q = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1q} \\ 1 & x_{21} & \cdots & x_{2q} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \cdots & x_{nq} \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_{1q}^{'} \\ \boldsymbol{x}_{2q}^{'} \\ \vdots \\ \boldsymbol{x}_{nq}^{'} \end{pmatrix}$$

当在选模型(5.1.2)中再增加一个自变量, 不妨记为x_{q+1}, 相应的设计矩阵为

$$\boldsymbol{X}_{q+1} = (\boldsymbol{X}_q \ \ \boldsymbol{x}_{q+1}), \ \ \boldsymbol{x}_{q+1} = (x_{1,q+1}, \cdots, x_{n,q+1})^{'}.$$

相应的残差平方和为

$$\mathsf{RSS}_{q+1} = \boldsymbol{Y}^{'}[\boldsymbol{I}_n - \boldsymbol{X}_{q+1}(\boldsymbol{X}_{q+1}^{'}\boldsymbol{X}_{q+1})^{-1}\boldsymbol{X}_{q+1}^{'}]\boldsymbol{Y}.$$

由分块矩阵求逆公式得

$$\begin{aligned}(\boldsymbol{X}_{q+1}^{'}\boldsymbol{X}_{q+1})^{-1} &= \begin{pmatrix} \boldsymbol{X}_q^{'}\boldsymbol{X}_q & \boldsymbol{X}_q^{'}\boldsymbol{x}_{q+1} \\ \boldsymbol{x}_{q+1}^{'}\boldsymbol{X}_q & \boldsymbol{x}_{q+1}^{'}\boldsymbol{x}_{q+1} \end{pmatrix}^{-1} \\ &\triangleq \begin{pmatrix} (\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1} + \boldsymbol{a}\boldsymbol{b}\boldsymbol{a}^{'} & \boldsymbol{c} \\ \boldsymbol{c}^{'} & \boldsymbol{b} \end{pmatrix},\end{aligned}$$

其中$\boldsymbol{a} = (\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}\boldsymbol{x}_{q+1}, \boldsymbol{c} = -\boldsymbol{a}\boldsymbol{b}$,

$$\boldsymbol{b}^{-1} = \boldsymbol{x}_{q+1}^{'}\boldsymbol{x}_{q+1} - \boldsymbol{x}_{q+1}^{'}\boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'}\boldsymbol{x}_{q+1}.$$

于是

$$\begin{aligned}
& \boldsymbol{X}_{q+1}(\boldsymbol{X}_{q+1}^{'}\boldsymbol{X}_{q+1})^{-1}\boldsymbol{X}_{q+1}^{'} \\
= \; & (\boldsymbol{X}_q \;\; \boldsymbol{x}_{q+1}) \begin{pmatrix} \boldsymbol{X}_q^{'}\boldsymbol{X}_q & \boldsymbol{X}_q^{'}\boldsymbol{x}_{q+1} \\ \boldsymbol{x}_{q+1}^{'}\boldsymbol{X}_q & \boldsymbol{x}_{q+1}^{'}\boldsymbol{x}_{q+1} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{X}_q^{'} \\ \boldsymbol{x}_{q+1}^{'} \end{pmatrix} \\
= \; & \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'} + \boldsymbol{X}_q\boldsymbol{a}\boldsymbol{b}\boldsymbol{a}^{'}\boldsymbol{X}_q^{'} + \boldsymbol{x}_{q+1}\boldsymbol{c}^{'}\boldsymbol{X}_q^{'} \\
& + \boldsymbol{X}_q\boldsymbol{c}\boldsymbol{x}_{q+1}^{'} + \boldsymbol{x}_{q+1}\boldsymbol{b}\boldsymbol{x}_{q+1}^{'}.
\end{aligned}$$

从而(注意$\boldsymbol{b}$其实是一非负常数)

$$\begin{aligned}
& \boldsymbol{X}_{q+1}(\boldsymbol{X}_{q+1}^{'}\boldsymbol{X}_{q+1})^{-1}\boldsymbol{X}_{q+1}^{'} - \boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'} \\
= \; & \boldsymbol{b}[(\boldsymbol{X}_q\boldsymbol{a} - \boldsymbol{x}_{q+1})\boldsymbol{a}^{'}\boldsymbol{X}_q^{'} - (\boldsymbol{X}_q\boldsymbol{a} - \boldsymbol{x}_{q+1})\boldsymbol{x}_{q+1}^{'}] \\
= \; & \boldsymbol{b}(\boldsymbol{X}_q\boldsymbol{a} - \boldsymbol{x}_{q+1})(\boldsymbol{X}_q\boldsymbol{a} - \boldsymbol{x}_{q+1})^{'}
\end{aligned}$$

为非负定矩阵. 所以

$$\mathsf{RSS}_{q+1} - \mathsf{RSS}_q = \boldsymbol{Y}^{'}[\boldsymbol{X}_q(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q^{'} - \boldsymbol{X}_{q+1}(\boldsymbol{X}_{q+1}^{'}\boldsymbol{X}_{q+1})^{-1}\boldsymbol{X}_{q+1}^{'}]\boldsymbol{Y} \leq 0.$$

因此当自变量子集扩大时, 残差平方和随之减少, 如果按"RSS越小越好"的原则选择自变量, 则选入回归方程的自变量将越来越多, 最后将把所有自变量选入回归方程. 可见, 残差平方和不能直接用作选择自变量的准则.

自变量选择的几个常见准则:

(1) 平均残差平方和准则(RMS_q).

由于RSS_q随q增大而下降, 为了防止选取过多的自变量, 一个常见的做法是对RSS_q乘上一个随q增加而上升的函数, 作为惩罚因子. 于是定义

$$\mathrm{RMS}_q = \frac{\mathrm{RSS}_q}{n-q-1}.$$

我们按RMS_q越小越好的原则选择自变量, 并称其为平均残差平方和准则或RMS_q准则.

(2) 调整后的R^2准则.

判定系数$R_q^2 = \mathsf{ESS}_q/\mathsf{TSS}$度量了数据与模型的拟合程度, 自然希望它越大越好. 但根据定义$R_q^2 = 1 - \mathsf{RSS}_q/\mathsf{TSS}$, 我们不能直接把$R_q^2$作为选择自变量的准则, 否则将把所有自变量选入模型. 为了克服以上缺点, 我们引入调整后的判定系数

$$\begin{aligned}\bar{R}_q^2 &= 1 - \frac{\mathsf{RSS}_q/(n-q-1)}{\mathsf{TSS}/(n-1)} = 1 - \frac{n-1}{n-q-1}\frac{\mathsf{RSS}_q}{\mathsf{TSS}} \\ &= 1 - \frac{n-1}{n-q-1}(1-R_q^2).\end{aligned}$$

易见$\bar{R}_q^2 \le R_q^2$, 且$\bar{R}_q^2$并不一定随着自变量个数的增加而增加. 这是因为, 尽管$1-R_q^2$随着自变量的个数的增加而减少, 但是$(n-1)/(n-q-1)$随着q的增加而增加, 这就使得$\bar{R}_q^2$并不一定随q的增大而增大.

我们选择使$\bar{R}_q^2$达到最大的自变量子集.

(3) C_p准则.

C_p准则是C.L.Mallows于1964年提出的, 它是从预测的观点出发提出来的. 对于选模型(5.1.2), C_p统计量定义为

$$C_p = \frac{\mathrm{RSS}_q}{\hat{\sigma}^2} - [n - 2(q+1)], \tag{5.2.1}$$

这里RSS_q是选模型(5.1.2)的残差平方和, $\hat{\sigma}^2$为全模型(5.1.1)中σ^2的最小二乘估计. 我们按"C_p越小越好"的准则来选择自变量.

获得(5.2.1)的想法如下: 假设全模型为真, 但为了提高预测的精度, 用选模型(5.1.2)去做预测, 很自然地, 要求n个预测值与期望值的相对偏差平方和的期望值

$$\Gamma_q \triangleq \mathsf{E}\Big\{\sum_{i=1}^{n}\Big(\frac{\widetilde{y}_{iq} - \mathsf{E}(y_i)}{\sigma}\Big)^2\Big\} = \mathsf{E}\Big\{\frac{1}{\sigma^2}\sum_{i=1}^{n}(\boldsymbol{x}_{iq}'\widetilde{\boldsymbol{\beta}}_q - \boldsymbol{x}_i'\boldsymbol{\beta})^2\Big\}$$

达到最小.

写

$$\begin{aligned}
& \mathsf{E}\Big\{\frac{1}{\sigma^2}\sum_{i=1}^{n}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q-\boldsymbol{x}_i^{'}\boldsymbol{\beta})^2\Big\} \\
= & \frac{1}{\sigma^2}\sum_{i=1}^{n}\mathsf{E}\Big\{[\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q-\mathsf{E}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q)]+[\mathsf{E}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q)-\boldsymbol{x}^{'}\boldsymbol{\beta}]\Big\}^2 \\
= & \frac{1}{\sigma^2}\sum_{i=1}^{n}\Big\{\mathsf{E}[\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q-\mathsf{E}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q)]^2+[\mathsf{E}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q)-\boldsymbol{x}_i^{'}\boldsymbol{\beta}]^2\Big\} \\
\triangleq & \frac{1}{\sigma^2}(I_1+I_2).
\end{aligned}$$

易见

$$\begin{aligned}
I_1 = & \sum_{i=1}^{n}\mathsf{Var}(\boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q)=\sigma^2\sum_{i=1}^{n}\boldsymbol{x}_{iq}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{iq} \\
= & \sigma^2\mathsf{tr}\Big[(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\sum_{i=1}^{n}\boldsymbol{x}_{iq}\boldsymbol{x}_{iq}^{'}\Big]=(q+1)\sigma^2.
\end{aligned}$$

利用定理5.1.1中(1)的结论以及(4)中的证明过程, 得

$$
\begin{aligned}
I_2 &= \sum_{i=1}^{n}[\boldsymbol{x}_{iq}^{'}(\boldsymbol{\beta}_q + \boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{\beta}_t) - (\boldsymbol{x}_{iq}^{'}\boldsymbol{\beta}_q + \boldsymbol{x}_{it}^{'}\boldsymbol{\beta}_t)]^2 \\
&= \sum_{i=1}^{n}(\boldsymbol{x}_{iq}^{'}\boldsymbol{B}^{-1}\boldsymbol{C}\boldsymbol{\beta}_t - \boldsymbol{x}_{it}^{'}\boldsymbol{\beta}_t)^2 \\
&= \sum_{i=1}^{n}\boldsymbol{\beta}_t^{'}(\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{iq} - \boldsymbol{x}_{it})(\boldsymbol{x}_{iq}^{'}\boldsymbol{B}^{-1}\boldsymbol{C} - \boldsymbol{x}_{it}^{'})\boldsymbol{\beta}_t \\
&= \sum_{i=1}^{n}\boldsymbol{\beta}_t^{'}[\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{iq}\boldsymbol{x}_{iq}^{'}\boldsymbol{B}^{-1}\boldsymbol{C} - \boldsymbol{x}_{it}\boldsymbol{x}_{iq}^{'}\boldsymbol{B}^{-1}\boldsymbol{C} \\
&\qquad -\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{x}_{iq}\boldsymbol{x}_{it}^{'} + \boldsymbol{x}_{it}\boldsymbol{x}_{it}^{'}]\boldsymbol{\beta}_t \\
&= \boldsymbol{\beta}_t^{'}[\boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{B}\boldsymbol{B}^{-1}\boldsymbol{C} - \boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{C} - \boldsymbol{C}^{'}\boldsymbol{B}^{-1}\boldsymbol{C} + \boldsymbol{D}]\boldsymbol{\beta}_t \\
&= \boldsymbol{\beta}_t^{'}\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_t \\
&= (n-q-1)[\mathsf{E}(\widetilde{\sigma}_q^2) - \sigma^2].
\end{aligned}
$$

所以

$$\begin{aligned}\Gamma_q &= \mathsf{E}\Big\{\frac{1}{\sigma^2}\sum_{i=1}^{n}(\boldsymbol{x}_{iq}'\widetilde{\boldsymbol{\beta}}_q - \boldsymbol{x}_i'\boldsymbol{\beta})^2\Big\} \\ &= \frac{1}{\sigma^2}\Big\{(q+1)\sigma^2 + (n-q-1)[\mathsf{E}(\widetilde{\sigma}_q^2) - \sigma^2]\Big\} \\ &= \frac{\mathsf{E}(\mathsf{RSS}_q)}{\sigma^2} - [n-2(q+1)].\end{aligned}$$

因为$\mathsf{E}(\mathsf{RSS}_q)$与$\sigma^2$未知, 所以我们用$\mathsf{RSS}_q$代替$\mathsf{E}(\mathsf{RSS}_q)$以及用$\hat{\sigma}^2$代替$\sigma^2$即可得$C_p$统计量.

由定理5.1.1中的(4)的证明过程可知:

$$\Gamma_q = q + 1 + \frac{\boldsymbol{\beta}_t'\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_t}{\sigma^2}.$$

鉴于C_p统计量的重要性, 下面阐述C_p统计量的一些性质, 它们对于应用C_p统计量作自变量选择, 提供了理论依据.

定理 (5.2.1)

假设随机误差向量$\boldsymbol{e} \sim N(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_n)$, 则对选模型*(5.1.2)*的$C_p$统计量, 有

$$E(C_p) = q + 1 - t + \frac{n-p-1}{n-p-3}\left(t + \frac{\boldsymbol{\beta}_t' \boldsymbol{D}_1^{-1} \boldsymbol{\beta}_t}{\sigma^2}\right),$$

这里$\boldsymbol{D}_1 = (\boldsymbol{D} - \boldsymbol{C}'\boldsymbol{B}^{-1}\boldsymbol{C})^{-1}$, $\boldsymbol{B} = \boldsymbol{X}_q'\boldsymbol{X}_q$, $\boldsymbol{C} = \boldsymbol{X}_q'\boldsymbol{X}_t$, $\boldsymbol{D} = \boldsymbol{X}_t'\boldsymbol{X}_t$.

证明: 问题归结为计算$\mathsf{E}(\mathsf{RSS}_q/\hat{\sigma}^2)$. 对于全模型(5.1.1), 残差平方和$\mathsf{RSS} = (n-p-1)\hat{\sigma}^2$且

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

选模型(5.1.2)中的残差平方和RSS_q可看成是在假设$H: \boldsymbol{\beta}_t = \mathbf{0}$下模型的残差平方和. 因此, $\eta \triangleq \mathsf{RSS}_q - \mathsf{RSS}$与RSS相互独立(根据最小二乘法基本定理), 且

$$\begin{aligned}\mathsf{E}\Big(\frac{\mathsf{RSS}_q}{\hat{\sigma}^2}\Big) &= (n-p-1)\mathsf{E}\Big(\frac{\mathsf{RSS}_q}{\mathsf{RSS}}\Big)\\ &= (n-p-1)\Big[1+\mathsf{E}(\eta)\cdot\mathsf{E}\Big(\frac{1}{\mathsf{RSS}}\Big)\Big].\end{aligned}$$

记$k = n-p-1$, 由$\mathsf{RSS}/\sigma^2 \sim \chi^2(k)$得

$$\begin{aligned}\mathsf{E}\left(\frac{\sigma^2}{\mathsf{RSS}}\right) &= 2^{-\frac{k}{2}}\left[\Gamma\left(\frac{k}{2}\right)\right]^{-1}\int_0^{\infty} x^{-1}\cdot e^{-\frac{x}{2}}x^{\frac{k}{2}-1}dx \\ &= 2^{-\frac{k}{2}}\left[\Gamma\left(\frac{k}{2}\right)\right]^{-1}2^{\frac{k}{2}-1}\Gamma\left(\frac{k}{2}-1\right) \\ &= \frac{1}{k-2}=\frac{1}{n-p-3}.\end{aligned}$$

因此

$$\mathsf{E}\left(\frac{1}{\mathsf{RSS}}\right)=\frac{1}{\sigma^2}\cdot\frac{1}{n-p-3}.$$

由于假设$H:\boldsymbol{\beta}_t=\mathbf{0}$可以等价地写成线性假设$H:\boldsymbol{A}\boldsymbol{\beta}=\mathbf{0}$, 其中$\boldsymbol{A}=(\mathbf{0},\boldsymbol{I}_t)$, 这里的$\mathbf{0}$是$t\times(q+1)$的零矩阵. 显然$\mathsf{rk}(\boldsymbol{A})=t$. 同时注意到

$$\hat{\boldsymbol{\beta}}_t\sim N(\boldsymbol{\beta}_t,\sigma^2\boldsymbol{D}_1),$$

即

$$\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\hat{\boldsymbol{\beta}}_t\sim N(\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\boldsymbol{\beta}_t,\boldsymbol{I}_t).$$

所以由第四章的公式(4.1.8)及非中心卡方分布的定义2.4.1可知

$$\frac{\eta}{\sigma^2}=\frac{1}{\sigma^2}\hat{\boldsymbol{\beta}}_t'\boldsymbol{D}_1^{-1}\hat{\boldsymbol{\beta}}_t=(\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\hat{\boldsymbol{\beta}}_t)'(\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\hat{\boldsymbol{\beta}}_t)\sim\chi^2(t,\delta),$$

其中

$$\delta=(\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\boldsymbol{\beta}_t)'(\frac{1}{\sigma}\boldsymbol{D}_1^{-\frac{1}{2}}\boldsymbol{\beta}_t)=\frac{\boldsymbol{\beta}_t'\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_t}{\sigma^2}.$$

因此

$$\mathsf{E}(\eta)=\sigma^2\Big(t+\frac{\boldsymbol{\beta}_t'\boldsymbol{D}_1^{-1}\boldsymbol{\beta}_t}{\sigma^2}\Big).$$

现在, 已可推知(注意$p = q + t$)

$$\begin{aligned}
\mathsf{E}(C_p) &= \mathsf{E}\Big(\frac{\mathsf{RSS}_q}{\hat{\sigma}^2}\Big) - [n - 2(q+1)] \\
&= (n-p-1)\Big[1 + \frac{1}{n-p-3}\Big(t + \frac{\boldsymbol{\beta}_t' \boldsymbol{D}_1^{-1} \boldsymbol{\beta}_t}{\sigma^2}\Big)\Big] - [n - 2(q+1)] \\
&= q + 1 - t + \frac{n-p-1}{n-p-3}\Big(t + \frac{\boldsymbol{\beta}_t' \boldsymbol{D}_1^{-1} \boldsymbol{\beta}_t}{\sigma^2}\Big).
\end{aligned}$$

这个定理说明, C_p统计量不是

$$\Gamma_q = \frac{1}{\sigma^2}(I_1 + I_2) = q + 1 + \frac{\boldsymbol{\beta}_t' \boldsymbol{D}_1^{-1} \boldsymbol{\beta}_t}{\sigma^2}$$

的无偏估计. 但如果$n - p$较大, 使得

$$\frac{n-p-1}{n-p-3} \approx 1, \tag{5.2.2}$$

则$\mathsf{E}(C_p) \approx \Gamma_q$. 即$C_p$统计量是$\Gamma_q$的渐近无偏估计量. 根据$\Gamma_q$的意义, Γ_q越小越好, 所以我们应该选择具有最小C_p值的自变量子集.

推论

在定理*5.2.1*的条件下, 若$\beta_t = 0$, 则

$$C_p = (q + 1 - t) + tu,$$

等价地,

$$C_p - (q + 1) = t(u - 1),$$

其中$u \sim F(t, n - p - 1)$.

证明: 记

$$u = \frac{\mathsf{RSS}_q - \mathsf{RSS}}{t\hat{\sigma}^2} = \frac{(\mathsf{RSS}_q - \mathsf{RSS})/t}{\mathsf{RSS}/(n-p-1)}.$$

易见u为假设$H : \boldsymbol{\beta}_t = \mathbf{0}$的$F$检验统计量. 所以, 若$\boldsymbol{\beta}_t = 0$, 则由最小二乘法基本定理知$u \sim F(t, n-p-1)$. 借助于$u$, C_p可表示为

$$\begin{aligned} C_p &= \left(\frac{\mathsf{RSS}_q - \mathsf{RSS}}{t\hat{\sigma}^2} + \frac{\mathsf{RSS}}{t\hat{\sigma}^2}\right)t - [n - 2(q+1)] \\ &= tu + \frac{(n-p-1)\hat{\sigma}^2}{\hat{\sigma}^2} - [n - 2(q+1)] \\ &= (q + 1 - t) + tu. \end{aligned}$$

我们来解释上述性质如何应用于自变量选择. 若$\boldsymbol{\beta}_t = \mathbf{0}$, 即选模型(5.1.2)是正确的, 那么从定理5.2.1知

$$\mathsf{E}(C_p) = q + 1 - t + \frac{n-p-1}{n-p-3}t,$$

若$n-p$较大使得(5.2.2)成立, 那么有

$$\mathsf{E}(C_p) \approx q + 1.$$

注意$q+1$其实是选模型的设计矩阵的秩. 这说明, 对于正确的选模型, 在平面直角坐标系中, 点$(q+1, C_p)$落在第一象限角平分线附近. 如果选模型不正确, 即$\boldsymbol{\beta}_t \neq \mathbf{0}$, 那么在条件(5.2.2)下有

$$\mathsf{E}(C_p) \approx q + 1 + \frac{\boldsymbol{\beta}_t' \boldsymbol{D}_1^{-1} \boldsymbol{\beta}_t}{\sigma^2} > q + 1,$$

此时点$(q+1, C_p)$将会向第一象限角平分线上方移动.

最后, 关于C_p统计量, 我们可以得到如下的自变量选择准则: 选择使得点$(q+1, C_p)$最接近第一象限角平分线且C_p值最小的选模型.

称直角坐标系中$(q+1, C_p)$的散点图为C_p图.

(4) AIC准则

极大似然原理是统计学中估计参数的一种重要的方法. Akaike把此方法加以修正, 提出了一种较为一般的模型选择准则, 称为Akaike信息量准则(AIC: Akaike Information Criterion).

对于一般的统计模型, 设$y_1, \cdots, y_n$是因变量的一个样本, 如果它们来自某个含k个参数的模型, 对应的似然函数的最大值记为$L_k(y_1, \cdots, y_n)$, 则选择使

$$\ln L_k(y_1, \cdots, y_n) - k \tag{5.2.3}$$

达到最大的模型. 下面我们把此准则应用于回归模型的自变量选择.

在选模型(5.1.2)中, 假设误差向量$\boldsymbol{e} \sim N(\boldsymbol{0}, \sigma^2 \boldsymbol{I}_n)$, 则$\boldsymbol{\beta}_q$与$\sigma^2$的似然函数为

$$L(\boldsymbol{\beta}_q, \sigma^2 | \boldsymbol{Y}) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{ -\frac{1}{2\sigma^2} \|\boldsymbol{Y} - \boldsymbol{X}_q \boldsymbol{\beta}_q\|^2 \right\}. \quad (5.2.4)$$

容易求得$\boldsymbol{\beta}_q$和σ^2的极大似然估计为

$$\bar{\boldsymbol{\beta}}_q = (\boldsymbol{X}_q' \boldsymbol{X}_q)^{-1} \boldsymbol{X}_q' \boldsymbol{Y},$$
$$\bar{\sigma}_q^2 = \frac{\mathsf{RSS}_q}{n} = \frac{\boldsymbol{Y}'[\boldsymbol{I}_n - \boldsymbol{X}_q(\boldsymbol{X}_q' \boldsymbol{X}_q)^{-1} \boldsymbol{X}_q']\boldsymbol{Y}}{n}.$$

代入(5.2.4)得对数似然函数的最大值为

$$\ln L(\bar{\boldsymbol{\beta}}_q, \bar{\sigma}_q^2 | \boldsymbol{Y}) = \frac{n}{2} \ln\left(\frac{n}{2\pi}\right) - \frac{n}{2} - \frac{n}{2} \ln(\mathsf{RSS}_q).$$

略去与q无关的项, 按照(5.2.3)得统计量$-\frac{n}{2}\ln(\mathrm{RSS}_q)-(q+1)$. 按AIC准则, 我们选择自变量子集使上式达到最大. 等价地, 记

$$\mathrm{AIC}=n\ln(\mathrm{RSS}_q)+2(q+1),$$

我们选择使上式达到最小的自变量子集.

注: Akaike(1976)和Haman(1979)基于Bayes方法提出了Bayes信息准则BIC:

$$\mathrm{BIC}=n\ln(\mathrm{RSS}_q)+(q+1)\ln n.$$

与AIC相比, BIC的惩罚加强了, 从而在选择变量进入模型上更加谨慎.

(5) J_p统计量准则

利用选模型进行预测, 预测偏差$y_0 - \boldsymbol{x}_{0q}^{'}\widetilde{\boldsymbol{\beta}}_q$的方差为

$$[1 + \boldsymbol{x}_{0q}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{0q}]\sigma^2.$$

因而在n个样本点上这些预测偏差的方差之和为

$$\begin{aligned}
\sum_{i=1}^{n}\mathsf{Var}(y_i - \boldsymbol{x}_{iq}^{'}\widetilde{\boldsymbol{\beta}}_q) &= \sigma^2\sum_{i=1}^{n}[1 + \boldsymbol{x}_{iq}^{'}(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\boldsymbol{x}_{iq}] \\
&= n\sigma^2 + \sigma^2\mathsf{tr}[(\boldsymbol{X}_q^{'}\boldsymbol{X}_q)^{-1}\sum_{i=1}^{n}\boldsymbol{x}_{iq}\boldsymbol{x}_{iq}^{'}] \\
&= (n + q + 1)\sigma^2.
\end{aligned}$$

由于σ^2未知, 所以用对应模型中σ^2的估计$\widetilde{\sigma}_q^2$代入就得到

$$\begin{aligned} J_p &= (n+q+1)\widetilde{\sigma}_q^2 \\ &= \frac{n+q+1}{n-q-1}\mathsf{RSS}_q. \end{aligned}$$

这里$(n+q+1)/(n-q-1)$起着惩罚的作用.

我们选择使J_p达到最小的自变量子集.

(6) 预测残差平方和PRESS$_q$(Predicted Residual Sum of Squares)准则

为了给出PRESS的定义和表达式, 我们略去q, 对全模型作推导. 考虑在建立回归方程时略去第i组数据, 此时记

$$\boldsymbol{Y}_{(i)} = \begin{pmatrix} y_1 \\ \vdots \\ y_{i-1} \\ y_{i+1} \\ \vdots \\ y_n \end{pmatrix}, \quad \boldsymbol{X}_{(i)} = \begin{pmatrix} \boldsymbol{x}_1^{'} \\ \vdots \\ \boldsymbol{x}_{i-1}^{'} \\ \boldsymbol{x}_{i+1}^{'} \\ \vdots \\ \boldsymbol{x}_n^{'} \end{pmatrix}, \quad \boldsymbol{e}_{(i)} = \begin{pmatrix} e_1 \\ \vdots \\ e_{i-1} \\ e_{i+1} \\ \vdots \\ e_n \end{pmatrix}.$$

相应的模型为

$$\boldsymbol{Y}_{(i)} = \boldsymbol{X}_{(i)}\boldsymbol{\beta} + \boldsymbol{e}_{(i)}.$$

此时$\boldsymbol{\beta}$的最小二乘估计为

$$\hat{\boldsymbol{\beta}}_{(i)} = (\boldsymbol{X}_{(i)}^{'}\boldsymbol{X}_{(i)})^{-1}\boldsymbol{X}_{(i)}^{'}\boldsymbol{Y}_{(i)}.$$

用$\boldsymbol{x}_i^{'}\hat{\boldsymbol{\beta}}_{(i)}$去预测第$i$个因变量, 预测偏差记为$\hat{e}_{(i)}$, 即

$$\hat{e}_{(i)} = y_i - \boldsymbol{x}_i^{'}\hat{\boldsymbol{\beta}}_{(i)}.$$

定义预测残差平方和为

$$\mathsf{PRESS} = \sum_{i=1}^{n}[\hat{e}_{(i)}]^2.$$

在第三章中我们已证明

$$\hat{\boldsymbol{\beta}}_{(i)} = \hat{\boldsymbol{\beta}} - \frac{(\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{x}_i\hat{e}_i}{1-h_{ii}},$$

所以

$$\hat{e}_{(i)} = y_i - \boldsymbol{x}_i^{'}\hat{\boldsymbol{\beta}} + \frac{\boldsymbol{x}_i^{'}(\boldsymbol{X}^{'}\boldsymbol{X})^{-1}\boldsymbol{x}_i\hat{e}_i}{1-h_{ii}} = \frac{\hat{e}_i}{1-h_{ii}}.$$

这里的$\hat{e}_i$是全数据情形下的第i个残差. 因此

$$\text{PRESS} = \sum_{i=1}^{n} \frac{\hat{e}_i^2}{(1-h_{ii})^2}.$$

若要计算PRESS_q, 只要将$\hat{e}_i$换成利用n组数据求得的选模型的回归方程的第i个残差, h_{ii}换成$\boldsymbol{X}_q(\boldsymbol{X}_q'\boldsymbol{X}_q)^{-1}\boldsymbol{X}_q'$(选模型的帽子矩阵)的第$i$个对角元即可.

我们选择使得PRESS_q达到最小的自变量子集.

例5.2.1 Hald水泥问题. 下面这组数据来自Hald的著作《Statistical Theory with Engineering Application》(1952).问题是考察含有如下四种化学成分:

x_1 : $3CaO \cdot Al_2O_3$的含量(%);

x_2 : $3CaO \cdot SiO_2$的含量(%);

x_3 : $4CaO \cdot Al_2O_3 \cdot Fe_2O_3$的含量(%);

x_4 : $2CaO \cdot SiO_2$的含量(%)

的水泥, 每一克所释放出的热量y与这四种成分含量之间的关系. 数据共13组.

表5.2.1: Hald水泥问题数据

序号	x_1	x_2	x_3	x_4	y
1	7	26	6	60	78.5
2	1	29	15	52	74.3
3	11	56	8	20	104.3
4	11	31	8	47	87.6
5	7	52	6	33	95.9
6	11	55	9	22	109.2
7	3	71	17	6	102.7
8	1	31	22	44	72.5
9	2	54	18	22	93.1
10	21	47	4	26	115.9
11	1	40	23	34	83.8
12	11	66	9	12	113.3
13	10	68	8	12	109.4

```
yx=read.table( “***.txt” )
x1=yx[, 1]
x2=yx[, 2]
x3=yx[, 3]
x4=yx[, 4]
y=yx[, 5]
cement=data.frame(x1,x2,x3,x4,y)
cement
X=matrix(c(x1,x2,x3,x4),nrow=13,byrow=F)
library(leaps)
adjr=leaps(X,y,int=T,method= “adjr2” )
adjr
adjr$which[which.max(adjr$adjr2),]
```

```
> X=matrix(c(x1,x2,x3,x4),nrow=13,byrow=F)
> library(leaps)
> adjr=leaps(X,y,int=T,method="adjr2")
> adjr
$which
      1     2     3     4
1 FALSE FALSE FALSE  TRUE
1 FALSE  TRUE FALSE FALSE
1  TRUE FALSE FALSE FALSE
1 FALSE FALSE  TRUE FALSE
2  TRUE  TRUE FALSE FALSE
2  TRUE FALSE FALSE  TRUE
2 FALSE FALSE  TRUE  TRUE
2 FALSE  TRUE  TRUE FALSE
2 FALSE  TRUE FALSE  TRUE
2  TRUE FALSE  TRUE FALSE
3  TRUE  TRUE FALSE  TRUE
3  TRUE  TRUE  TRUE FALSE
3  TRUE FALSE  TRUE  TRUE
3 FALSE  TRUE  TRUE  TRUE
4  TRUE  TRUE  TRUE  TRUE
```

Figure: $2^4 - 1 = 15$个自变量子集

```
$adjr2
 [1] 0.6449549 0.6359290 0.4915797 0.2209521 0.9744140 0.9669653 0.9223476
 [8] 0.8164305 0.6160725 0.4578001 0.9764473 0.9763796 0.9750415 0.9637599
[15] 0.9735634

> adjr$which[which.max(adjr$adjr2),]
    1     2     3     4
 TRUE  TRUE FALSE  TRUE
>
```

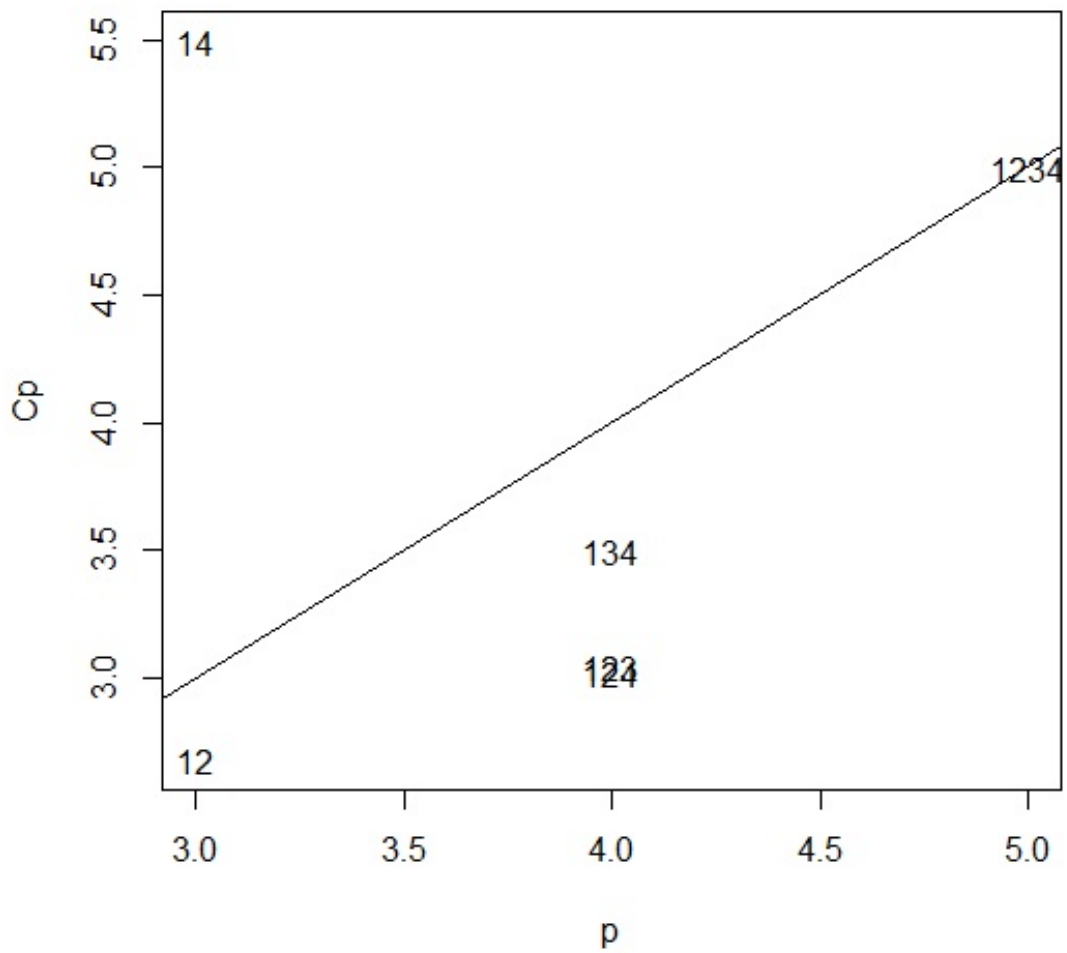
Figure: 调整后的R^2达到最大的自变量子集

利用调整后的R^2, 最终选择自变量子集: x_1, x_2, x_4.

```
library(faraway)
library(leaps)
cp=leaps(X,y,int=T,method= “Cp” )
cp
cp$which[which.min(cp$Cp),]
Cpplot(cp)
```

```
$Cp
 [1] 138.730833 142.486407 202.548769 315.154284   2.678242   5.495851
 [7]  22.373112  62.437716 138.225920 198.094653   3.018233   3.041280
[13]   3.496824   7.337474   5.000000

> cp$which[which.min(cp$Cp),]
    1     2     3     4
 TRUE  TRUE FALSE FALSE
> Cpplot(cp)
>
```



有些模型的C_p没有显示在图中, 因为它们的C_p值太大. 根据C_p准则, 最终选择自变量子集: x_1, x_2.

leaps中的method只有三种选择: method=c(“Cp”, “adjr2”, “r2”). 若需通过其它准则选择自变量, 我们可采用下面的方法.

```
yx=read.table(“***.txt”)
x1=yx[,1]
x2=yx[,2]
x3=yx[,3]
x4=yx[,4]
y=yx[,5]
cement=data.frame(x1,x2,x3,x4,y)
cement
library(leaps)
search.results=regsubsets(y~x1+x2+x3+x4,data=cement,
    method="exhaustive",nbest=15)
selection.criteria=summary(search.results)
selection.criteria
```

(未完, 待续, nbest指“number of subsets of each size to record”)

```
names(selection.criteria)
selection.criteria$which
n=length(cement[,1])
q=as.integer(row.names(selection.criteria$which))
R.sq=selection.criteria$rsq
AdjR.sq=selection.criteria$adjr2
rms=selection.criteria$rss/(n-q-1)
Cp=selection.criteria$cp
aic.f=n*log(selection.criteria$rss)+2*(q+1)
bic.f=n*log(selection.criteria$rss)+(q+1)*log(n)
var=as.matrix(selection.criteria$which[,2:5])
criteria.table=data.frame(cbind(q,rms,R.sq,AdjR.sq,Cp,aic.f,bic.f,
    var[,1],var[,2],var[,3],var[,4]),row.names=NULL)
names(criteria.table)=c("q","RMS","Rsq","aRsq","Cp","AIC",
    "BIC","x1","x2","x3","x4")
round(criteria.table,2)
```

```
> selection.criteria=summary(search.results)
> selection.criteria
Subset selection object
Call: regsubsets.formula(y ~ x1 + x2 + x3 + x4, data = cement, method = "exhaustive",
    nbest = 15)
4 Variables  (and intercept)
   Forced in Forced out
x1     FALSE      FALSE
x2     FALSE      FALSE
x3     FALSE      FALSE
x4     FALSE      FALSE
15 subsets of each size up to 4
Selection Algorithm: exhaustive
         x1  x2  x3  x4
1  ( 1 ) " " " " " " "*"
1  ( 2 ) " " "*" " " " "
1  ( 3 ) "*" " " " " " "
1  ( 4 ) " " " " "*" " "
2  ( 1 ) "*" "*" " " " "
2  ( 2 ) "*" " " " " "*"
2  ( 3 ) " " " " "*" "*"
2  ( 4 ) " " "*" "*" " "
2  ( 5 ) " " "*" " " "*"
2  ( 6 ) "*" " " "*" " "
3  ( 1 ) "*" "*" " " "*"
3  ( 2 ) "*" "*" "*" " "
3  ( 3 ) "*" " " "*" "*"
3  ( 4 ) " " "*" "*" "*"
4  ( 1 ) "*" "*" "*" "*"
```

```
> names(selection.criteria)
[1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"
> selection.criteria$which
   (Intercept)    x1    x2    x3    x4
1         TRUE FALSE FALSE FALSE  TRUE
1         TRUE FALSE  TRUE FALSE FALSE
1         TRUE  TRUE FALSE FALSE FALSE
1         TRUE FALSE FALSE  TRUE FALSE
2         TRUE  TRUE  TRUE FALSE FALSE
2         TRUE  TRUE FALSE FALSE  TRUE
2         TRUE FALSE FALSE  TRUE  TRUE
2         TRUE FALSE  TRUE  TRUE FALSE
2         TRUE FALSE  TRUE FALSE  TRUE
2         TRUE  TRUE FALSE  TRUE FALSE
3         TRUE  TRUE  TRUE FALSE  TRUE
3         TRUE  TRUE  TRUE  TRUE FALSE
3         TRUE  TRUE FALSE  TRUE  TRUE
3         TRUE FALSE  TRUE  TRUE  TRUE
4         TRUE  TRUE  TRUE  TRUE  TRUE
>
```

	q	RMS	Rsq	aRsq	Cp	AIC	BIC	x1	x2	x3	x4
1	1	80.35	0.67	0.64	138.73	92.20	93.33	0	0	0	1
2	1	82.39	0.67	0.64	142.49	92.52	93.65	0	1	0	0
3	1	115.06	0.53	0.49	202.55	96.86	97.99	1	0	0	0
4	1	176.31	0.29	0.22	315.15	102.41	103.54	0	0	1	0
5	2	5.79	0.98	0.97	2.68	58.76	60.46	1	1	0	0
6	2	7.48	0.97	0.97	5.50	62.09	63.78	1	0	0	1
7	2	17.57	0.94	0.92	22.37	73.20	74.89	0	0	1	1
8	2	41.54	0.85	0.82	62.44	84.38	86.08	0	1	1	0
9	2	86.89	0.68	0.62	138.23	93.97	95.67	0	1	0	1
10	2	122.71	0.55	0.46	198.09	98.46	100.16	1	0	1	0
11	3	5.33	0.98	0.98	3.02	58.32	60.58	1	1	0	1
12	3	5.35	0.98	0.98	3.04	58.36	60.62	1	1	1	0
13	3	5.65	0.98	0.98	3.50	59.07	61.33	1	0	1	1
14	3	8.20	0.97	0.96	7.34	63.92	66.18	0	1	1	1
15	4	5.98	0.98	0.97	5.00	60.29	63.11	1	1	1	1

DAAG package中的press(model)可以返回model的PRESS值.

```
> lm.reg=lm(y~x1+x2,data=cement)
> AIC(lm.reg)
[1] 64.31239
> library(DAAG)
> press(lm.reg)
[1] 93.88255
>
```

产生15个模型的press值:

```
library(DAAG)
mods=c("y~x1","y~x2","y~x3","y~x4","y~x1+x2","y~x1+x3",
"y~x1+x4","y~x2+x3","y~x2+x4","y~x3+x4","y~x1+x2+x3",
"y~x1+x2+x4","y~x1+x3+x4","y~x2+x3+x4","y~x1+x2+x3+x4")
P=numeric(15)
for (i in 1:15) {mod=lm(mods[i],cement)
P[i]=press(mod)}
P=data.frame("PRESS"=P,row.names=mods)
round(P,2)
```

	PRESS
y~x1	1699.61
y~x2	1202.09
y~x3	2616.36
y~x4	1194.22
y~x1+x2	93.88
y~x1+x3	2218.12
y~x1+x4	121.22
y~x2+x3	701.74
y~x2+x4	1461.81
y~x3+x4	294.01
y~x1+x2+x3	90.00
y~x1+x2+x4	85.35
y~x1+x3+x4	94.54
y~x2+x3+x4	146.85
y~x1+x2+x3+x4	110.35

leaps package中的"regsubsets"也有图示法的变量选择功能.

```
yx=read.table( “***.txt” )
x1=yx[,1]
x2=yx[,2]
x3=yx[,3]
x4=yx[,4]
y=yx[,5]
cement=data.frame(x1,x2,x3,x4,y)
cement
library(leaps)
subsets=regsubsets(y~x1+x2+x3+x4,data=cement)
summary(subsets)
plot(subsets)
plot(subsets,scale="Cp")
plot(subsets,scale="adjr2")
```

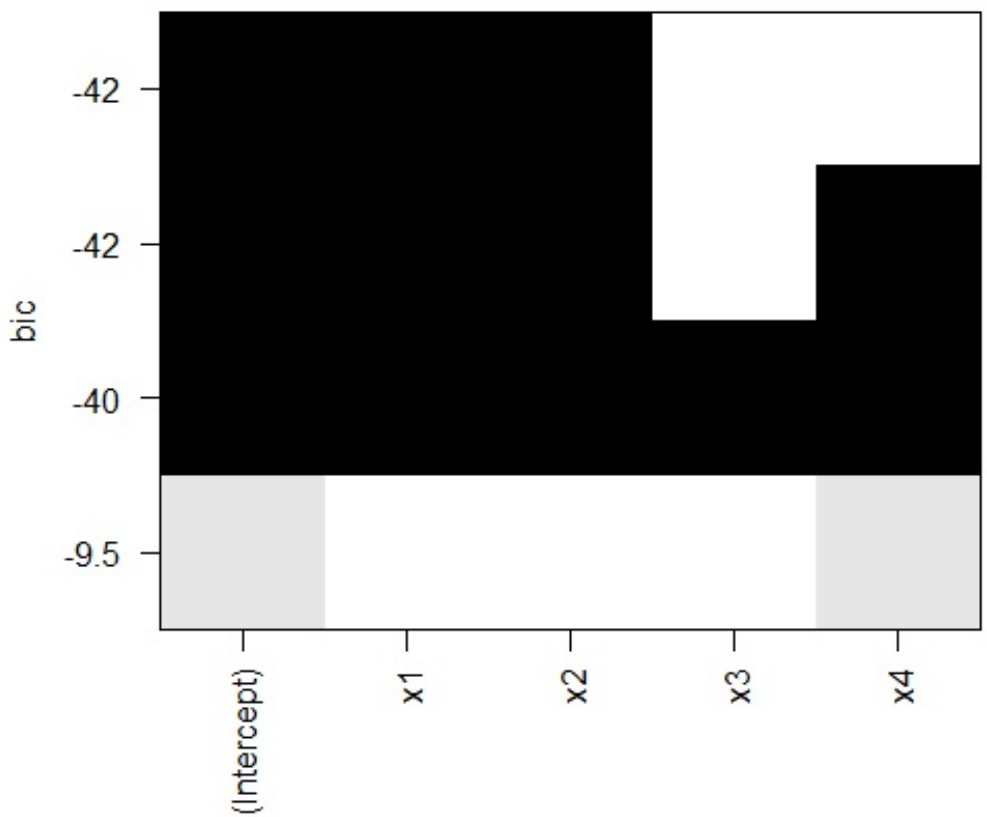
注: scale=" xx" where “xx” is either “C_p” , “adjr2” , “r2” or “bic” .

```
> library(leaps)
> subsets=regsubsets(y~x1+x2+x3+x4,data=cement)
> summary(subsets)
Subset selection object
Call: regsubsets.formula(y ~ x1 + x2 + x3 + x4, data = cement)
4 Variables  (and intercept)
   Forced in Forced out
x1     FALSE      FALSE
x2     FALSE      FALSE
x3     FALSE      FALSE
x4     FALSE      FALSE
1 subsets of each size up to 4
Selection Algorithm: exhaustive
         x1  x2  x3  x4
1  ( 1 ) " " " " " " "*"
2  ( 1 ) "*" "*" " " " "
3  ( 1 ) "*" "*" " " "*"
4  ( 1 ) "*" "*" "*" "*"
>
```

这结果告诉我们: 若只选择一个自变量, 应选入x_4; 若只选入两个自变量, 应选入x_1和x_2; 若选入三个自变量, 应选入x_1, x_2和x_4; 若选入四个自变量, 则全部选入.

注: plot (subsets, scale=“xx”)可显示变量选择示意图, 其中“xx” 可以是“Cp”, “adjr2”, “r2” 或“bic”, 默认是“bic” .

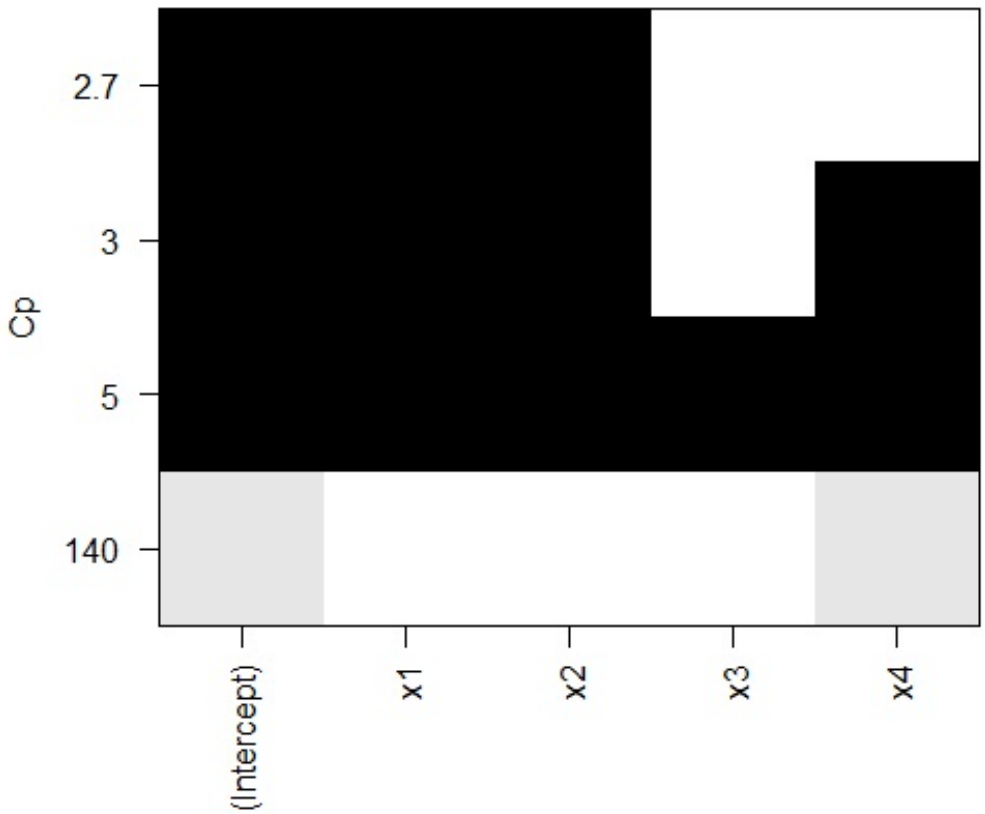
plot(subsets)产生下图:



图形解释: Each row in this graph represents a model; the shaded rectangles in the columns indicate the variables included in the given model. The numbers on the left margin are the values of Schwartz' Bayesian Information Criterion; not that the axis is not quantitative but is ordered. The darkness of the shading simply represents the ordering of the BIC values.

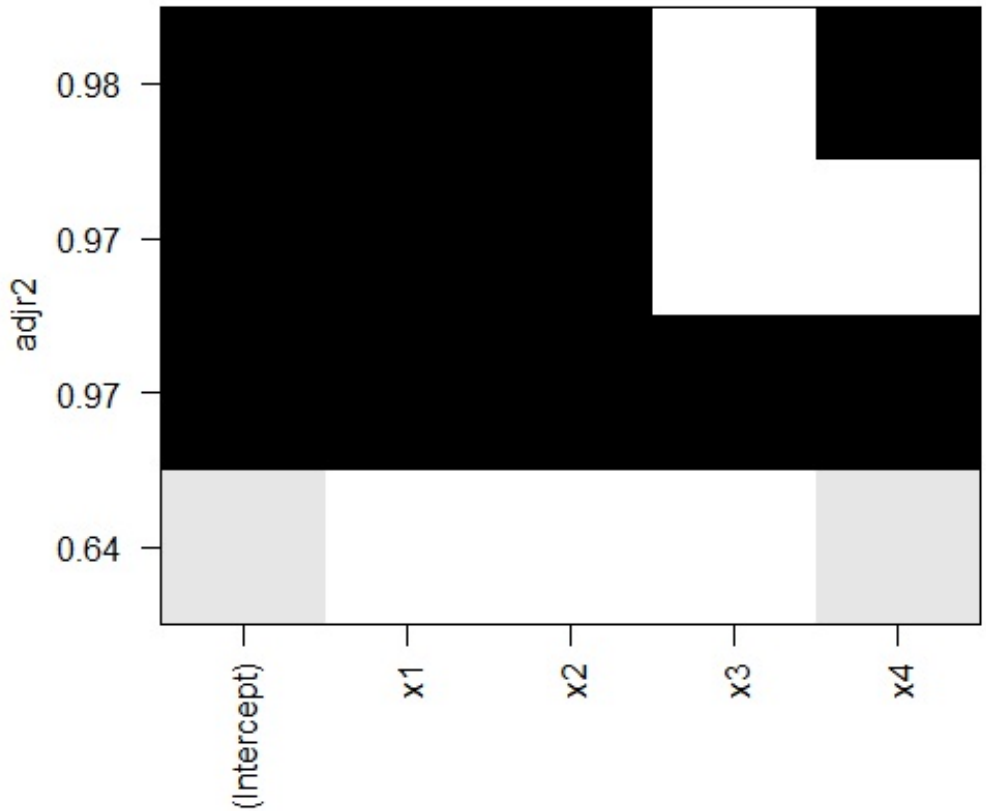
因此, 采用BIC准则, 应选入自变量x_1和x_2.

plot(subsets,scale="Cp")产生的图形是



因此, 采用C_p准则, 应选入自变量x_1和x_2.

plot(subsets,scale="adjr2")产生的图形是



因此, 采用调整后的R^2准则, 应选入自变量x_1, x_2和x_4.

自变量选择的方法

多元线性回归分析中, p个自变量的所有可能子集构成$2^p - 1$个线性回归方程. 当可供选择的自变量个数不太多时, 用前面介绍的自变量选择准则可挑选出"最优"的回归方程. 但是当自变量的个数较多时, 以上这些方法就不大实用了. 为此, 人们提出了一些较为简便、实用、快捷的选择最优回归方程的方法. 这些方法各有优缺点, 至今没有绝对最优的方法, 目前常用的方法有向前法、向后法和逐步回归法.

(1) 向前法(Forward).

向前法的思想是回归模型中的自变量由少到多, 每次增加一个, 直到没有可引入的自变量为止. 具体做法是:

Step1: 因变量y关于每个自变量$x_1, \cdots, x_p$分别建立一元线性回归方程, 因此得到p个回归方程. 分别计算这p个一元线性回归方程的p个回归系数的F检验值, 记为$F_1^{(1)}, \cdots, F_p^{(1)}$. 选其最大者, 记为

$$F_j^{(1)} = \max\{F_1^{(1)}, \cdots, F_p^{(1)}\}.$$

给定显著性水平α, 若$F_j^{(1)} > F_\alpha(1, n-2)$, 则首先将$x_j$引入回归方程. 不妨假设$x_j$就是$x_1$;

Step2: 将因变量y分别与$(x_1, x_2), (x_1, x_3), \cdots, (x_1, x_p)$建立$p-1$个二元线性回归方程, 对这$p-1$个二元线性回归方程中$x_2, \cdots, x_p$的回归系数进行$F$ 检验, 并计算F值, 记为$F_2^{(2)}, \cdots, F_p^{(2)}$. 选其最大者, 记为

$$F_j^{(2)} = \max\{F_1^{(2)}, \cdots, F_p^{(2)}\}.$$

给定显著性水平α, 若$F_j^{(2)} > F_\alpha(1, n-3)$, 则首先将$x_j$引入回归方程. 不妨假设$x_j$就是$x_2$;

Step3: 继续以上做法, 假设已确定选入q个自变量$x_1, \cdots, x_q$, 在建立$q+1$元线性回归方程时, 若所有的$x_{q+1}, \cdots, x_p$的F值均不大于$F_\alpha(1, n-q-2)$时, 变量选择结束. 这时得到的q元线性回归方程就是最终确定的回归方程.

向前法的缺点是: 不能反映引入新变量后的变化情况. 因为某个自变量可能刚开始时是显著的, 但是当引入其它自变量后它就变得不显著了, 但是没有机会将其剔除. 即一旦引入, 就是"终身制"的.

(2) 向后法(Backward).

向后法与向前法恰恰相反, 向后法的思想是选入的自变量个数由多到少, 每次剔除一个, 直到没有可剔除的自变量为止. 具体做法是:

Step1: 因变量y关于所有自变量$x_1, \cdots, x_p$建立一个p元线性回归方程, 分别计算p个回归系数的F检验值, 记为$F_1^{(p)}, \cdots, F_p^{(p)}$. 选其最小者, 记为

$$F_j^{(p)} = \min\{F_1^{(p)}, \cdots, F_p^{(p)}\}.$$

给定显著性水平β, 若$F_j^{(p)} \leq F_\beta(1, n-p-1)$, 则首先将$x_j$从回归方程中剔除. 不妨假设$x_j$就是$x_p$;

Step2: 因变量y关于自变量$x_1, \cdots, x_{p-1}$建立一个$p-1$元的线性回归方程, 分别计算$p-1$个回归系数的F检验值, 记为$F_1^{(p-1)}$, $\cdots$, $F_{p-1}^{(p-1)}$. 选其最小者, 记为

$$F_j^{(p-1)} = \min\{F_1^{(p-1)}, \cdots, F_{p-1}^{(p-1)}\}.$$

给定显著性水平β, 若$F_j^{(p-1)} \leq F_\beta(1, n-p)$, 则将$x_j$从回归方程中剔除. 不妨假设$x_j$就是$x_{p-1}$;

Step3: 继续以上做法, 假设已确定剔除了$p-q$个自变量$x_{q+1}, \cdots, x_q$, 在y关于$x_1, \cdots, x_q$的q元线性回归方程中, 若所有$x_1, \cdots, x_q$的F值均大于$F_\beta(1, n-q-1)$, 则变量选择结束. 这时得到的q元线性回归方程就是最终确定的回归方程.

向后法的缺点是: 一开始就把所有自变量引入回归方程, 这样的计算量很大. 另外, 自变量一旦被剔除, 将永远没有机会再重新进入回归方程, 即是"一棒子打死"的.

(3) 逐步回归法(Stepwise).

逐步回归法的基本思想是有进有出. 具体做法是: 将自变量一个一个地引入回归方程, 每引入一个自变量后, 都要对已选入的自变量进行逐个检验, 当先引入的自变量由于后引入的自变量而变得不再显著时, 就要将其剔除. 将这个过程反复进行下去, 直到既无显著的自变量引入回归方程, 也无不显著的自变量从回归方程中剔除为止. 这样就避免了向前法和向后法各自的缺点, 以保证最后得到的自变量子集是"最优"的自变量子集.

R软件中可以用step()做向前法、向后法、逐步回归, 但是它们是基于AIC准则的, 而不是基于p-value或F值.

向前法:

```
yx=read.table( “***.txt” )
x1=yx[, 1]
x2=yx[, 2]
x3=yx[, 3]
x4=yx[, 4]
y=yx[, 5]
cement=data.frame(x1,x2,x3,x4,y)
cement
min.model=lm(y~1,data=cement)
fwd.model=step(min.model,direction=”forward”,
                    scope=(~x1+x2+x3+x4))
summary(fwd.model)
```

```
Start:  AIC=71.44
y ~ 1

       Df Sum of Sq     RSS    AIC
+ x4    1   1831.90  883.87 58.852
+ x2    1   1809.43  906.34 59.178
+ x1    1   1450.08 1265.69 63.519
+ x3    1    776.36 1939.40 69.067
<none>              2715.76 71.444

Step:  AIC=58.85
y ~ x4

       Df Sum of Sq    RSS    AIC
+ x1    1    809.10  74.76 28.742
+ x3    1    708.13 175.74 39.853
<none>              883.87 58.852
+ x2    1     14.99 868.88 60.629

Step:  AIC=28.74
y ~ x4 + x1

       Df Sum of Sq    RSS    AIC
+ x2    1    26.789 47.973 24.974
+ x3    1    23.926 50.836 25.728
<none>              74.762 28.742

Step:  AIC=24.97
y ~ x4 + x1 + x2

       Df Sum of Sq    RSS    AIC
<none>              47.973 24.974
+ x3    1   0.10909 47.864 26.944
```

```
> summary(fwd.model)

Call:
lm(formula = y ~ x4 + x1 + x2, data = cement)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0919 -1.8016  0.2562  1.2818  3.8982

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  71.6483    14.1424   5.066 0.000675 ***
x4           -0.2365     0.1733  -1.365 0.205395
x1            1.4519     0.1170  12.410 5.78e-07 ***
x2            0.4161     0.1856   2.242 0.051687 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.309 on 9 degrees of freedom
Multiple R-squared:  0.9823,    Adjusted R-squared:  0.9764
F-statistic: 166.8 on 3 and 9 DF,  p-value: 3.323e-08
```

向前法的结果: 选入x_1, x_2和x_4.

向后法:

```
yx=read.table( “***.txt” )
x1=yx[, 1]
x2=yx[, 2]
x3=yx[, 3]
x4=yx[, 4]
y=yx[, 5]
cement=data.frame(x1,x2,x3,x4,y)
cement
full.model=lm(y~x1+x2+x3+x4,data=cement)
bwd.model=step(full.model,direction=”backward”)
summary(bwd.model)
```

```
> full.model=lm(y~x1+x2+x3+x4,data=cement)
> bwd.model=step(full.model,direction="backward")
Start:  AIC=26.94
y ~ x1 + x2 + x3 + x4

       Df Sum of Sq    RSS    AIC
- x3    1    0.1091 47.973 24.974
- x4    1    0.2470 48.111 25.011
- x2    1    2.9725 50.836 25.728
<none>               47.864 26.944
- x1    1   25.9509 73.815 30.576

Step:  AIC=24.97
y ~ x1 + x2 + x4

       Df Sum of Sq    RSS    AIC
<none>               47.97 24.974
- x4    1      9.93  57.90 25.420
- x2    1     26.79  74.76 28.742
- x1    1    820.91 868.88 60.629
```

```
> summary(bwd.model)

Call:
lm(formula = y ~ x1 + x2 + x4, data = cement)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0919 -1.8016  0.2562  1.2818  3.8982

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  71.6483    14.1424   5.066 0.000675 ***
x1            1.4519     0.1170  12.410 5.78e-07 ***
x2            0.4161     0.1856   2.242 0.051687 .
x4           -0.2365     0.1733  -1.365 0.205395
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.309 on 9 degrees of freedom
Multiple R-squared:  0.9823,    Adjusted R-squared:  0.9764
F-statistic: 166.8 on 3 and 9 DF,  p-value: 3.323e-08
```

向后法的结果: 选入x_1, x_2和x_4.

逐步回归法:

```
yx=read.table( “***.txt” )
x1=yx[, 1]
x2=yx[, 2]
x3=yx[, 3]
x4=yx[, 4]
y=yx[, 5]
cement=data.frame(x1,x2,x3,x4,y)
cement
min.model=lm(y~1,data=cement)
step.model=step(min.model,direction= “both” ,
                        scope=(~x1+x2+x3+x4))
summary(step.model)
```

```
> step.model=step(min.model,direction="both",scope=(~x1+x2+x3+x4))
Start:  AIC=71.44
y ~ 1

       Df Sum of Sq     RSS    AIC
+ x4    1   1831.90  883.87 58.852
+ x2    1   1809.43  906.34 59.178
+ x1    1   1450.08 1265.69 63.519
+ x3    1    776.36 1939.40 69.067
<none>              2715.76 71.444

Step:  AIC=58.85
y ~ x4

       Df Sum of Sq     RSS    AIC
+ x1    1    809.10   74.76 28.742
+ x3    1    708.13  175.74 39.853
<none>              883.87 58.852
+ x2    1     14.99  868.88 60.629
- x4    1   1831.90 2715.76 71.444
```

未完,待续.

```
Step:  AIC=28.74
y ~ x4 + x1

       Df Sum of Sq     RSS    AIC
+ x2    1     26.79   47.97 24.974
+ x3    1     23.93   50.84 25.728
<none>                74.76 28.742
- x1    1    809.10  883.87 58.852
- x4    1   1190.92 1265.69 63.519

Step:  AIC=24.97
y ~ x4 + x1 + x2

       Df Sum of Sq    RSS    AIC
<none>               47.97 24.974
- x4    1      9.93  57.90 25.420
+ x3    1      0.11  47.86 26.944
- x2    1     26.79  74.76 28.742
- x1    1    820.91 868.88 60.629
> 
```

```
> summary(step.model)

Call:
lm(formula = y ~ x4 + x1 + x2, data = cement)

Residuals:
    Min      1Q  Median      3Q     Max
-3.0919 -1.8016  0.2562  1.2818  3.8982

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  71.6483    14.1424   5.066 0.000675 ***
x4           -0.2365     0.1733  -1.365 0.205395
x1            1.4519     0.1170  12.410 5.78e-07 ***
x2            0.4161     0.1856   2.242 0.051687 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.309 on 9 degrees of freedom
Multiple R-squared:  0.9823,    Adjusted R-squared:  0.9764
F-statistic: 166.8 on 3 and 9 DF,  p-value: 3.323e-08
```

逐步回归法的结果: 选入x_1, x_2和x_4.