北京大学

学士学位论文

题目 关于回归模型选择与参数估计的综述

姓　　名　李康

学　　号　00601045

院　　系　数学科学学院

专　　业　概率统计

研究方向　应用统计

指导教师　姚远

2010.5.30

Literature Review about Model Selection and Estimation in Regression

Li Kang

Supervisor: Yao Yuan

School of Mathematical Sciences, Peking University

June, 2010

Submitted in total fulfilment of the requirements for the degree of Bachelor in Probability and Statistics

版权声明

摘　　要

在线性回归模型中，我们需要从一大堆因变量中选出重要的变量来拟合响应变量值并对新的数据进行预测。传统的方法包括：向前选择，向后选择，主成分回归等。但我们发现，在模型中加入惩罚项后，得到的估计稀疏性更好，预测模型的均方误差更小，从而得到更满意的回归模型。其中最有名的为 Lasso 模型，由此提出了一个新的算法：LARS。LARS 计算效率更高，且修订的 LARS 可以解决 Lasso 估计，或于 Stagewise 方法一致。但有时需要将变量进行组合分群形成新的因子，对因子进行回归。为解决此提出三种算法：Group Lasso、Group LARS、Group Nonnegative Garrote，这需要组内变量是正交的；对于非正交一般情况，有一个算法可以有效解决. 另一方面，当惩罚项是 CAP 模式时，仍然可以进行估计。且在某些情况下，这种模型得到的结果要优于以上模型。

关键词： 线性回归，惩罚回归，变量选择，解路径

Abstract

In a linear model, we hope to select a parsimonious set from a large collection of possible covariates for the efficient prediction of a response variable. The classical techniques include *Forward Selection, Backward Elimination and Principle Component Regression*. However, we have found that penalized loss function minimization yielded sparse solutions and improved on the predictive performance. The most popular method is *Lasso*. We introduce a new algorithm called *LARS*, which is computationally efficient. And a simple modification of it implements the *Lasso*. we also consider the problem of selecting grouped variables, which can be solved by *Group Lasso, Group LARS, Group Nonnegative Garrote* if the each factor is orthonormalized. Otherwise, we derive a efficient algorithm that yields solutions sparse at both the group and individual feature levels. Finally, we introduce the CAP family for grouped and hierarchical variable selection.

Keywords: linear regression, penalized regression, variable selection, coefficient paths

目　　录

第一章　引言

在现实问题中，当我们要用因变量去预测响应变量时，最常用的是建立线性模型。好模型的评价主要有两个标准: 预测的准确性和模型的复杂度。这就需要我们在尽量不怎么降低预测准确性的条件下，简化模型，即寻找重要的因变量。通常的做法包括：向前选择，向后选择，子集选择等。但这些方法不是计算复杂，就是可能会严重夸大结论的显著性。所以需要寻找新的模型选择和参数估计的方法。

具体的说，对于一般线性模型：

$$Y = \sum_{i=1}^{p} x_i \beta_i + \epsilon = X\beta + \epsilon \tag{1.1}$$

其中 Y 是$n \times 1$向量，x_i 代表第 i 个因变量，X 为 $n \times p$ 矩阵，β 为参数向量，$\epsilon \sim N_n(0, \sigma^2 I)$. 设我们的参数估计为 $\hat{\beta}$，则预测向量 $\hat{\mu} = X\hat{\beta}$.

基于上面的模型，首先介绍一种向前逐步回归方法(Stagewise)：开始 $\hat{\mu} = 0$，计算当前的相关系数 $\hat{c} = c(\hat{\mu}) = X'(Y - \hat{\mu})$，然后在相关系数最大的那个分量上前进一小步，即令

$$\hat{j} = argmax|\hat{c}_j| \quad , \quad \hat{\mu} \to \hat{\mu} + \kappa \cdot sign(\hat{c}_{\hat{j}}) \cdot x_{\hat{j}} \tag{1.2}$$

其中 κ 为常数。如此往复循环。

其实，模型变量的选择就是将其中一些变量的系数置为零，也就是将参数 β 稀疏化。为此，Tibshirani提出Lasso方法（1996）：在参数绝对值总和受限的情况下，最小化残差平方和。即最小化

$$S(\hat{\beta}) = \|Y - \hat{\mu}\| = \sum_{i=1}^{n} (y_i - \hat{\mu}_i)^2$$

当

$$T(\hat{\beta}) = \sum_{j=1}^{p} |\hat{\beta}_j| \leq t. \tag{1.3}$$

对于这种受约束的模型，通常解法是根据 Karush-Kuhn-Tucker(KKT) 条件进行迭代求解。对于固定的 t，解在(1.3)的边界上达到，参数中只有一部分非零。t 足够大时，得到的估计将于最小二乘估计（OLS）一致。

上面两种方法的定义虽然看起来完全不同，但它们所得到的结果是几乎一致的。2003年，Efron等指出逐步回归和 Lasso 都可以由一种叫做“Least Angle Regression”(LARS)的方法变形得到，且LARS算法清晰的结构不仅适于高效率的计算，而且也利于理论分析。在第二章我们将会详细介绍 LARS 的有关步骤和性质。

模型（1.1）是对因变量进行回归，但有时候因变量可能是具有多个水平的因子，或我们需要将原始变量进行变形分类，然后进行回归。如果我们将原始变量形成的一个新的类变量看做一个因子，那我们的因子线性模型如下：

$$Y = \sum_{j=1}^{J} X_j \beta_j + \epsilon = X\beta + \epsilon \tag{1.4}$$

其中 Y 是 $n \times 1$ 向量，X_j 代表第 j 个因子，X_j 为 $n \times p_j$ 矩阵，β_j 为 p_j 维参数向量，$\epsilon \sim N_n(0, \sigma^2 I)$.

这样，模型选择的任务就是选出重要的因子，而不是单个变量。Ming Yuan和Yi Lin将单个变量选择的方法将以推广，在2004年给出三种主要的选择方法：Group Lasso，Group LARS，Group Nonnegative Garrote. 但它们都需要对每一个 X_j 进行正交化，即 $X_j' X_j = I_{p_j}, j = 1, \ldots, J.$ 此外，我们虽然挑出了重要因子，保证了 β 的稀疏性，却不能保证 β_j 的稀疏性。Friedman等提出了一种算法（2010），在惩罚项中加入对 β_j 内部的L_1惩罚，即：

$$\min_{\beta \in R^p} \left(\|Y - \sum_{j=1}^{J} X_j \beta_j\|_2^2 + \lambda_1 \sum_{j=1}^{J} \|\beta_j\|_2 + \lambda_2 \|\beta\|_1 \right) \tag{1.5}$$

有效的解决了稀疏性问题，且不要求 X_j 正交。这在第三章中将一一介绍。

由以上可以看出，在线性模型中加入惩罚项可以有效的提取出影响响应变量的重要信息。Lasso中引入参数的 L_1 范数是最常见的模型。其实可以将此推广到其他惩罚项模型，对于（1.4）回归，加上对因子间和因子内的惩罚项，可以增加更多的信息。Peng Zhao等提出 CAP 惩罚（2009），并给出了 CAP 惩罚模型下的参数估计的算法。发现对于 $J >> n$ 的情形，CAP具有更好的表现。第四章将进行讨论。

在第五章中，我们将针对第二章和第四章内的算法进行计算机模拟。

第二章　LARS算法

2.1　LARS算法阐述

对于模型（1.1），假定

$$\sum_{i=1}^{n} y_i = 0,\ \sum_{i=1}^{n} x_{ij} = 0,\ \sum_{i=1}^{n} x_{ij}^2 = 1,\ j = 1, 2, \ldots, p.$$

开始置估计 $\hat{\mu}$ 为零，找到与 Y 相关性最大的变量，不妨设为 x_i，加入模型，$\hat{\mu}$ 沿着相关方向前进，直到有其他变量与残差的相关性与 x_i 相同，设为 x_j。这时，残差在 x_i, x_j 所张成平面上的投影与这两个变量的夹角相同。$\hat{\mu}$ 再沿着此投影方向前进，直到有第三个变量与现在的残差的相关性和前两个变量相同。$\hat{\mu}$ 前进的方向变为残差在这三个变量所张平面的投影方向。继续如此下去，变量会按照重要性一一加入模型。举个例子，当 $X = (x_1, x_2)$ 时，如图2.1。

设 $\mathcal{A}$ 为{1,2,...,p}的子集,定义矩阵

$$X_{\mathcal{A}} = (\cdots s_j x_j \cdots)_{j \in \mathcal{A}}, \tag{2.11}$$

其中 $s_j = \pm 1$. 令

$$\mathcal{G}_{\mathcal{A}} = X_{\mathcal{A}}' X_{\mathcal{A}},\ A_{\mathcal{A}} = (1_{\mathcal{A}}' \mathcal{G}_{\mathcal{A}}^{-1} 1_{\mathcal{A}})^{-\frac{1}{2}}, \tag{2.12}$$

则等角向量为：

$$u_{\mathcal{A}} = X_{\mathcal{A}} w_{\mathcal{A}},\ w_{\mathcal{A}} = A_{\mathcal{A}} \mathcal{G}_{\mathcal{A}}^{-1} 1_{\mathcal{A}} \tag{2.13}$$

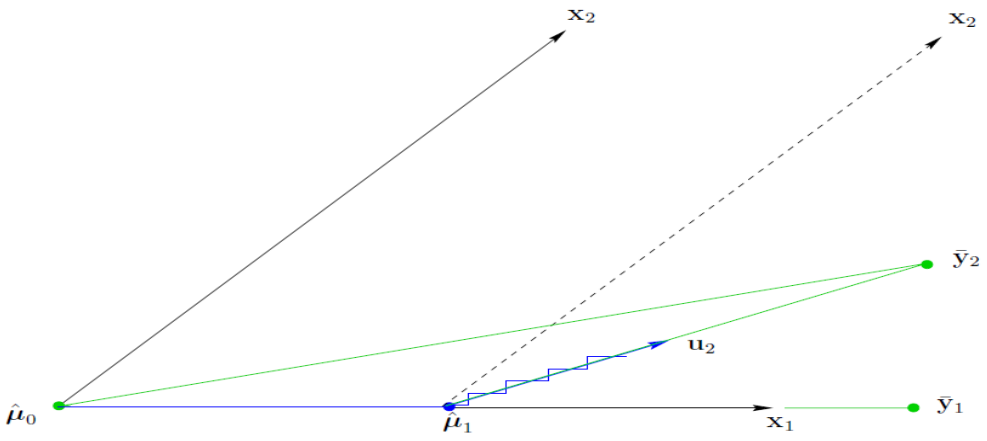


图 2.1: LARS:$\bar{y_2}$是y在x_1, x_2所张平面的投影，从$\hat{\mu}_0$出发，前进到$\hat{\mu}_1$，再沿着u_2前进。

容易推导，$u_{\mathcal{A}}$ 满足：

$$X'_{\mathcal{A}}u_{\mathcal{A}} = A_{\mathcal{A}}1_{\mathcal{A}},\ \|u_{\mathcal{A}}\|^2 = 1.$$

LARS算法的详细步骤为：假设 $\hat{\mu}_{\mathcal{A}}$ 是当前的LARS估计，$\hat{c}_j = X'(y - \hat{\mu}_{\mathcal{A}})$ 是当前的相关系数，$\mathcal{A}$ 是当前活跃集：

$$\hat{C} = \max_j\{|\hat{c}_j|\},\ \mathcal{A} = \{j : |\hat{c}_j| = \hat{C}\}. \tag{2.14}$$

对于 $j \in \mathcal{A}$，令

$$s_j = sign\{\hat{c}_j\} \tag{2.15}$$

利用(2.11)－(2.13)式计算 $X_{\mathcal{A}}$, $A_{\mathcal{A}}$, $u_{\mathcal{A}}$ 且计算内积向量

$$a \equiv X'u_{\mathcal{A}} \tag{2.16}$$

下一步更新 $\hat{\mu}_{\mathcal{A}}$:

$$\hat{\mu}_{\mathcal{A}_+} = \hat{\mu}_{\mathcal{A}} + \hat{\gamma}u\mathcal{A}, \tag{2.17}$$

其中

$$\hat{\gamma} = \min{}^+_{j\in\mathcal{A}^c}\left\{\frac{\hat{C} - \hat{c}_j}{A_{\mathcal{A}} - a_j},\ \frac{\hat{C} + \hat{c}_j}{A_{\mathcal{A}} + a_j}\right\}; \tag{2.18}$$

将向量看做空间中的一个点，经过简单的分析，可以得出第 $k(k < p)$步LARS得到的估计 $\hat{\mu}_k$ 在 $\hat{\mu}_{k-1}$ 与 Y 在 $\mathcal{A}_k$ 所张平面上的投影 $\bar{y}_k$ 的连线上，但不会达到 $\bar{y}_k$.

2.2 修订的LARS算法

向前逐步回归，Lasso，和 LARS 都是进行变量选择的。在这一节，我们将指出它们之间有很大的相似性。将 LARS 算法做简单的修改，就可以分别与 Lasso 和向前逐步回归相一致。并且这种修订的 LARS 具有更高的计算效率。

2.2.1 LARS-Lasso算法

令 $\hat{\beta}$ 是 Lasso 问题(1.2)的解，则 $\hat{\mu} = X\hat{\beta}$，$\hat{c}_j = x_j^{'}(y - \hat{\mu})$，Efron等在“Least Angle Regression ”引理8中指出，必有如下性质：

$$\text{sign}(\hat{\beta}_j) = \text{sign}(\hat{c}_j) = s_j \tag{2.21}$$

. 设当前活跃集是$\mathcal{A}$, $\hat{\mu}_{\mathcal{A}}$ 与 $\hat{\mu}$ 相同。令 $w_{\mathcal{A}} = A_{\mathcal{A}}\mathcal{G}_{\mathcal{A}}^{-1}1_{\mathcal{A}}$, $\hat{d}$是一个 p 维向量，当 $j \in \mathcal{A}$ 时，$\hat{d}_j = s_j w_{\mathcal{A}j}$, 其余为 0。接着考虑在其等角方向 $u_{\mathcal{A}}$ 上前进 γ. 有：

$$\mu(\gamma) = \hat{\mu}_{\mathcal{A}} = X\beta(\gamma),\ \beta_j(\gamma) = \hat{\beta}_j + \gamma\hat{d}_j$$

. 可以看出，$\beta_j(\gamma)$ 在 $\gamma_j = -\hat{\beta}_j/\hat{d}_j,\ j \in \mathcal{A}$ 处变号。由于准则(2.21), γ 应在 $\tilde{\gamma} = \min_{\gamma_j>0}$ 处截止。

由以上分析，得算法：

如果 $\tilde{\gamma} < \hat{\gamma}$，LARS中的步长取为 $\gamma = \tilde{\gamma}$，且将 $\tilde{j}(\gamma_{\tilde{j}} = \tilde{\gamma})$ 从活跃集中去除，即

$$\hat{\mu}_{\mathcal{A}_+} = \hat{\mu}_{\mathcal{A}} + \tilde{\gamma} u\mathcal{A},\ \mathcal{A}_+ = \mathcal{A} - \{\tilde{j}\}. \tag{2.22}$$

这与 Lasso 的关系由下面的结论保证:

定理2.1.

利用算法(2.22)，且假定一次 $\mathcal{A}$ 中的元素变化只有一个，则可以找到 Lasso 所有的解.

2.2.2 LARS-Stagewise算法

回顾向前逐步回归的方法(1.2)，假设在估计 $\hat{\mu}$ 后，以 κ 为步长走了N步，且 N_j 为总共以 x_j 为方向走的步数，$j = 1, 2, \ldots, p$. 经过简单分析，若 $j \notin \mathcal{A}, N_j = 0$. 令

$$P \equiv (N_1, N_2, \ldots, N_p)/N$$

则新的估计为：

$$\mu = \hat{\mu} + N\kappa X_{\mathcal{A}} P_{\mathcal{A}} \tag{2.23}$$

而在 LARS 中为：

$$\mu_{\mathcal{A}} + \gamma X_{\mathcal{A}} w_{\mathcal{A}},\ w_{\mathcal{A}} = A_{\mathcal{A}} \mathcal{G}_{\mathcal{A}}^{-1} 1_{\mathcal{A}} \tag{2.24}$$

比较上面两式，发现若要二者相同，必须 $u_{\mathcal{A}} \in \mathcal{C}_{\mathcal{A}}$, 其中

$$\mathcal{C}_{\mathcal{A}} = \left(v = \sum_{j \in \mathcal{A}} s_j x_j P_j,\ P_j \geq 0 \right).$$

若不，则自然想到将 $u_{\mathcal{A}}$ 投影到 $\mathcal{C}_{\mathcal{A}}$上 即距离最近的点。由此得算法：

在LARS算法中将 $u_{\mathcal{A}}$ 替换成 $u_{\mathcal{B}}$，$u_{\mathcal{B}}$ 为 $u_{\mathcal{A}}$ 投影到 $\mathcal{C}_{\mathcal{A}}$上 的单位向量。

此修订的算法由下面定理保证：

定理2.2.

在上面算法下，可以找到向前逐步回归所有的解.

由以上定理，可知前后两步估计：

$$\hat{\mu}_{\mathcal{A}_+} - \hat{\mu}_{\mathcal{A}} = \hat{\gamma} u_{\mathcal{B}} = \hat{\gamma} X_{\mathcal{B}} w_{\mathcal{B}}$$

因为$w_{\mathcal{B}}$非负，所以有

$$sign(\hat{\beta}_{+j} - \hat{\beta}_j) = s_j,\ j \in \mathcal{B}$$

最后，我们比较一下这一节提到的三种方法：

- 向前逐步回归：相邻两步 $\hat{\beta}_j$ 改变量的符号与 $\hat{c}_j = x_j'(y - \hat{\mu})$ 相同。
- Lasso: $\hat{\beta}_j$ 与 $\hat{c}_j$ 符号相同。
- LARS:没有符号限制。

2.3 自由度和C_p估计

LARS，Lasso，向前逐步回归都可以将因变量逐一的加入模型，那么到什么时候为止呢？这需要我们给出准确的数学估计。类似于经典的 AIC、BIC 准则，我们针对LARS算法提出了 C_p 准则如下。

设$y \sim (\mu, \sigma^2 I)$, $\hat{\mu} = g(y)$是对 μ 的估计。有：

$$(\hat{\mu}_i - \mu_i)^2 = (y_i - \hat{\mu}_i)^2 - (y_i - \mu_i)^2 + 2(\hat{\mu}_i - \mu_i)(y_i - \mu_i) \tag{2.31}$$

上式对i求和并取期望得：

$$E\left\{\frac{\|\hat{\mu} - \mu\|^2}{\sigma^2}\right\} = E\left\{\frac{\|y - \hat{\mu}\|^2}{\sigma^2} - n\right\} + 2\sum_{i=1}^{n} cov(\hat{\mu}_i,\ y_i)/\sigma^2. \tag{2.32}$$

由此，我们定义自由度为：

$$df_{\mu,\,\sigma^2} = \sum_{i=1}^{n} cov(\hat{\mu}_i,\ y_i)/\sigma^2. \tag{2.33},$$

由(2.32)式，取 C_p 估计如下：

$$C_p\hat{\mu} = \frac{\|y - \hat{\mu}\|^2}{\sigma^2} - n + 2df_{\mu,\sigma^2}. \tag{2.34}$$

如果 $\sigma^2, df_{\mu,\sigma^2}$ 已知，则 $C_p(\hat{\mu})$ 是 $E\{\|\hat{\mu} - \mu\|^2/\sigma^2\}$ 的无偏估计。对于线性估计 $\hat{\mu} = My$, 则 $df_{\mu,\,\sigma^2} = traceM$, 这与OLS自由度的定义是一致的。当这些都未知时，可以用OLS的估计 $\bar{\mu}$, $\bar{\sigma}^2$ 代替。自由度的计算可以采用自助法得到数值结果。这里不再赘述。但对于 LARS 算法第 k 步得到的 $\hat{\mu}_k$，有如下定理：

定理2.3.

当因变量 $x_1, x_2, \ldots, x_p$ **相互正交时，则** $df\{\hat{\mu}_k\} = k.$

又若所有可能子集 $\mathcal{A}$, 满足：

$$\mathcal{G}_{\mathcal{A}}^{-1} 1_{\mathcal{A}} > 0, \tag{2.34}$$

即每一个元素都大于零。有如下定理：

定理2.4.

在条件(2.34)**下，** $df\{\hat{\mu}_k\} = k.$

LARS、Lasso、逐步向前回归都满足条件(2.34)；但 Lasso 算法(2.22)，可能会超过 p 步，一个合理的猜想是第 k 步的自由度取为当前估计系数不为零的个数。

2.4 一些性质

对于LARS: 第 k-1 步中，设 x_k 是加入活跃集中的唯一变量，w_k, u_k 如(2.13)定义。则 w_k 的第 k 个元素符号与 $c_{kk} = x_k'(y - \hat{\mu}_{k-1})$ 相同。$\hat{\beta}_k (\hat{\mu}_k = X\hat{\beta}_k)$ 的第 k 个元素符号与 c_{kk} 相同。

对于Lasso: 设 $\mathcal{T}$ 是关于t的开区间，下确界为 t_0，t 在 $\mathcal{A}$ 内变化时，$\mathcal{A}$ 不变。则 Lasso 估计 $\hat{\mu}(t)$ 满足

$$\hat{\mu}(t) = \hat{\mu}(t_0) + A_{\mathcal{A}}(t - t_0)u_{\mathcal{A}}.$$

由此看出，$\hat{\mu}$, $\hat{\beta}$, $\hat{C}$ 关于t都是分段线性的。

当 $p < n$ 时，LARS 算法的时间复杂度是 $O(p^3 + np^2)$，与最小二乘拟合一致。当$p \gg n$时，LARS在吸收 n-1 个变量后停止，时间复杂度为$O(n^3)$.

第三章　聚类因子回归

在上一章中的 LARS 和 Lasso 算法在单个变量的选择中有良好的结果和很高的计算效率；但对于因子模型(1.4)，它们却不适用，有两个主要方面：(一)它们的选择是依赖于单个变量的影响力而不是整个因子的影响力，如果变量强而因子弱，则可能将弱因子选入模型；(二)因子内部的正交化方法不同，可能导致选择结果的不同。在本章中将介绍解决此类问题的方法。

3.1　Group Lasso

设 d 维向量 η, d 阶正定矩阵 K，定义 K－范数如下：

$$\|\eta\|_K = (\eta' K \eta)^{\frac{1}{2}}.$$

简记 $\|\eta\| = \|\eta\|_{I_d}$. 给定正定矩阵 $K_1, K_2, \ldots, K_J$, 则Group Lasso 寻求最小化下式的解：

$$\frac{1}{2}\left\| Y - \sum_{j=1}^{J} X_j \beta_j \right\|^2 + \lambda \sum_{j=1}^{J} \|\beta_j\|_{K_j} \tag{3.11}$$

在这里，我们选 $K_j = p_j I_{p_j}$. 这样，根据 KKT 条件，得以下结论：

命题3.1.

令 $K_j = p_j I_{p_j}, j = 1, \ldots, J$. $\beta = (\beta_1', \ldots, \beta_J')'$ **是**(3.11)**的解得充分必要条件是：**

$$-X_j'(Y - X\beta) + \frac{\lambda\sqrt{p_j}\beta_j}{\|\beta_j\|} = 0 \quad \forall \beta_j \neq 0 \tag{3.12}$$

$$\| - X_j'(Y - X\beta)\| \leq \lambda\sqrt{p_j} \quad \forall \beta_j = 0 \tag{3.13}$$

因为 $X_j' X_j = I_{P_j}$, 容易验证，(3.12)和(3.13)的解为

$$\beta_j = \left(1 - \frac{\lambda\sqrt{p_j}}{\|S_j\|}\right)_+ S_j \tag{3.13}$$

其中，$S_j = X_j'(Y - X\beta_{-j})$, $\beta_{-j} = (\beta_1', \ldots, \beta_{j-1}', 0', \beta_{j+1}', \ldots, \beta_J')$. 如此，可利用(3.13)式进行迭代求解。结果具有数值稳定性和收敛性。但计算负担比较大。

3.2 Group LARS

当 $p_1 = \cdots = p_J$ 时，LARS算法可以如下推广：定义 $\theta(r, X_j)$ 为向量 r 与 X_j 的列向量所张平面之间的夹角。则 $\cos^2\theta(r, X_j)$ 为用 X_j 回归 r 可解释的比例。因为 X_j 正交，可得 $\cos^2\theta(r, X_j) = \|X_j'r\|^2/\|r\|^2$. 这样，在第二章的 LARS 中，将相关系数的判断准则换为夹角的判断准则。即先找到与 Y 夹角最小的因子 X_{j_1}, 沿着 Y 在 X_j(列向量所张平面)上的投影方向前进，直到另一因子 X_{j_2} 满足

$$\|X_{j_1}'r\|^2 = \|X_{j_2}'r\|^2, \tag{3.21}$$

r为当前的残差向量。然后沿着 r 在 X_{j_1}和X_{j_2} 张平面上的投影方向前进，依此进行下去。

当 p_j 不全相等时，可用加权夹角准则，即在(3.21)式中两边分别除以 p_{j_1} 和 p_{j_2}。

完整的 Group LARS 算法如下：

(1)置$\beta^{[0]} = 0, k = 1,\ r^{[0]} = Y$.

(2)计算当前活跃集

$$\mathcal{A}_1 = \arg\max_j \|X_j'r^{[k-1]}\|^2/p_j.$$

(3)计算当前 $p = \sum p_j$ 维方向向量 γ:

$$\gamma_{\mathcal{A}_k^c} = 0, \gamma_{\mathcal{A}_k} = (X_{\mathcal{A}_k}'X_{\mathcal{A}_k})^- X_{\mathcal{A}_k}'r^{[k-1]}.$$

(4)对于每个 $j \notin \mathcal{A}_k$, 计算若 j 进入模型，需要前进的距离 α_j, 即满足

$$\|X_j'(r^{[k-1]} - \alpha_j X\gamma)\|^2/p_j = \|X_{j'}'(r^{[k-1]} - \alpha_j X\gamma)\|^2/p_j'$$

j' 为 $\mathcal{A}_k$ 中任一数。

(5)若 $\mathcal{A}_k \neq \{1, \ldots, J\}$, 令$\alpha = \min_{j\notin\mathcal{A}_k}\alpha_j \equiv \alpha_{j^*}$ 且 $\mathcal{A}_{k+1} = \mathcal{A} \cup \{j^*\}$. 否则，置 $\alpha = 1$.

(6) $\beta^{[k]} = \beta^{[k-1]} + \alpha\gamma,\ r^{[k]} = Y - X\beta^{[k]},\ k = k + 1$.返回(3)直到 $\alpha = 1$.

当 $\alpha = 1$ 时，达到一般的最小二乘估计，且一共需要经过J步。

3.3 Group Nonnegative Garrote

对于模型(1.1),Breiman在1995年提出了 nonnegative garrote，其估计 $\hat{\beta}_j = \hat{\beta}_j^{LS} d_j(\lambda)$, $\hat{\beta}_j^{LS}$ 是最小二乘估计，$d_j(\lambda)$ 是一个实数，使得：

$$d(\lambda) = \arg\min_d \frac{1}{2}\|Y - Zd\|^2 + \lambda \sum_{j=1}^{J} d_j,\ d_j \geq 0,\ \forall j. \tag{3.31}$$

其中 $Z_j = X_j \hat{\beta}_j^{LS}$.

对于模型(1.4),可以做同样的事情，重新定义 $d(\lambda)$ 如下：

$$d(\lambda) = \arg\min_d \frac{1}{2}\|Y - Zd\|^2 + \lambda \sum_{j=1}^{J} d_j,\ d_j \geq 0,\ \forall j. \tag{3.32}$$

nonnegative garrote 的解是分段线性的，类似于为解决 Lasso 对 LARS 作的修订，得到 Group Nonnegative Garrote 算法为：

(1)置 $d^{[0]} = 0, k = 1, r^{[0]} = Y$.

(2)计算当前活跃集

$$\mathcal{C}_1 = \arg\max_j Z_j' r^{[k-1]}/p_j$$

(3)计算当前p维方向向量 γ：

$$\gamma_{\mathcal{C}_k^c} = 0,\ \gamma_{\mathcal{C}_k} = (Z_{\mathcal{C}_k}' Z_{\mathcal{C}_k})^- Z_{\mathcal{C}_k}' r^{[k-1]}.$$

(4) 对于每个 $j \notin \mathcal{C}_k$, 计算若 j 进入模型，需要前进的距离 α_j, 即满足

$$Z_j'(r^{[k-1]} - \alpha_j Z\gamma)/p_j = Z_{j'}'(r^{[k-1]} - \alpha_j Z\gamma)/p_j'$$

j' 为 $\mathcal{C}_k$ 中任一数。

(5)对于每一个 $j \in \mathcal{C}_k$, 计算 $\beta_j = -d_j^{[k-1]}/\gamma_j, \alpha_j = \min(\beta_j, 1)$, 如果非负，则代表在 d_j 变为零之前能够前进的步长。

(6)如果 α_j 都非正或 $\min_{j:\alpha_j>0}\{\alpha_j\} > 1$, 置 $\alpha = 1$. 否则，令 $\alpha = \min_{j:\alpha_j>0}\{\alpha_j\} \equiv \alpha_{j^*}$. 且 $d^{[k]} = d^{[k-1]} + \alpha\gamma$. 如果 $j^* \notin \mathcal{C}_k$, 令 $\mathcal{C}_{k+1} = \mathcal{C} \cup \{j^*\}$, 否则 $\mathcal{C}_{k+1} = \mathcal{C} - \{j^*\}$.

(7)置 $r^{[k]} = Y - Zd^{[k]}, k = k + 1$. 返回(3)直到 $\alpha = 1$.

当每一步(6)中 j^* 有且只有一个时，上述算法可以得到(3.32)的所有解。

nonnegative garrote 计算速度最快，但不能直接用于样本量小于因变量的情况下。Group LARS 和Group nonnegative garrote 的解是分段线性的，而Group

Lasso 的解是分段线性的充分必要条件为：其任一解 $\hat{\beta}$ 可以写成 $\hat{\beta}_j = c_j\beta_j^{LS}, j = 1,\ldots,J, c_1,\ldots,c_J$ 为标量. 这个条件在一般情况下是不成立的,所以 Group Lasso 的计算代价要比其他两种高。

3.4 C_p的估计

与上一章一样，为了得到最终的估计，需要进行 C_p 的估计。在高斯回归模型中，我们沿用(2.33)与(2.34).对于未知的参数，我们可以利用上一章的估计方法。当为正交设计时类似的，可以采用有如下近似：

对于 group Lasso,

$$\tilde{df}(\hat{\mu}(\lambda) \equiv X\beta) = \sum_j I(\|\beta_j\| > 0) + \sum_j \frac{\|\beta_j\|}{\|\beta_j^{LS}\|}(p_j - 1). \tag{3.41}$$

对于 group LARS,

$$\tilde{df}(\hat{\mu}_k \equiv X\beta^{[k]}) = \sum_j I(\|\beta_j^{[k]}\| > 0) + \sum_j \left(\frac{\sum_{l<k} \|\beta_j * [l+1] - \beta_j * [l]\|}{\sum_{l<J} \|\beta_j^{[l+1]} - \beta_j * [l]\|} \right)(p_j - 1). \tag{3.42}$$

对于 nonnegative garrote,

$$\tilde{df}(\hat{\mu}(\lambda) \equiv Zd) = 2\sum_j I(d_j > 0) + \sum_j d_j(p_j - 2). \tag{3.43}$$

我们有如下定理:

定理3.1.

若模型(1.4)**中**X**是正交的，则对于**(3.41)-(3.43)**的估计，**$df = E(\tilde{df})$.

但历史经验表明，对于非正交情形，以上估计也是可靠的。

3.5 非正交情形下

在本章第一节中，考虑了如下问题：

$$\min_{\beta \in R^P} \left(\|Y - \sum_{j=1}^{J} X_j\beta_j\|_2^2 + \lambda \sum_{j=1}^{J} \sqrt{p_j}\|\beta_j\| \right) \tag{3.51}$$

假若 X_j 内部是非正交的，则不能利用(3.13)式求解。这一节我们将解决一般问题下的 Group Lasso 求解方法。

假设X，Y已经中心化。为叙述简单，忽略权重p_j(有权重时推导类似)。(3.12)与(3.13)可以写为：

$$-X_j^T(Y-\sum_j X_j\beta_j)+\lambda s_l=0;\ j=1,2,\ldots,J, \tag{3.52}$$

其中 $s_j=\beta_j/\|\beta_j\|$ 如果 $\beta_j\neq 0$. 否则，$\|s_j\|_2\leq 1$。如果

$$\|X_j^T(Y-\sum_{k\neq j}X_k\hat{\beta}_k)\|<\lambda \tag{3.53}$$

那么 $\hat{\beta}_j=0$. 否则

$$\hat{\beta}_j=(X_j^TX_j+\lambda/\|\hat{\beta}_j\|)^{-1}X_j^Tr_j \tag{3.53}$$

其中 $r_j=Y-\sum_{k\neq j}X_k\hat{\beta}_k$.

如果 X_j 内部正交，则(3.53)与(3.13)一致。而正交会的不同将会导致非原始问题(3.51)的解。如： X_j 的 SVD 分解为 $X_j=UDV^T$, 将 X_j 正交化为 U，则 $X_j\beta_j=U[DV^T\beta_j]=U\beta_{j^*}$, 若要 $\|\beta_j^*\|=\|\beta_j\|$, 必须 $D=I$。所以需要重新考虑一种算法。

我们考虑更为严格的模型(1.5)。对于某一 l, 设 $X_l=Z=(Z_1,Z_2,\ldots,Z_k)$, 系数为 $\beta_l=\theta=(\theta_1,\theta_2,\ldots,\theta_k)$. $r=Y-\sum_{k\neq l}X_k\beta_k$. (1.5)式对 θ_j 求导得：

$$-Z_j^T(r-\sum_j Z_j\theta_j)+\lambda_1 s_j+\lambda_2 t_j=0,\ j=1,2,\ldots,k \tag{3.54}$$

其中 $s_j=\theta_j/\|\theta\|$，若 $\theta\neq 0$, 否则 $\|s\|_2\leq 1$. 若 $\theta_j\neq 0$, $t_j=\mathrm{sign}(\theta_j)$, 否则 $t_j\in[-1,1]$.

令$a=X_l^Tr$,则θ为零的充分必要条件为

$$\|s\|_2\leq 1,\ t_j\in[-1,1]$$

下，

$$a_j=\lambda_1 s_j+\lambda_2 t_j,\ j=1,2,\ldots,k \tag{3.55}$$

有解。

为了尽量使 θ 稀疏化，我们可以最小化：

$$J(t)=(1/\lambda_1^2)\sum_{j=1}^k(a_j-\lambda_2 t_j)^2=\sum_{j=1}^k s_j^2 \tag{3.56}$$

可知使上式最小的 t 为:

$$\hat{t}_j=\begin{cases}\frac{a_j}{\lambda_2} & |\frac{a_j}{\lambda_2}|\leq 1;\\ sign(\frac{a_j}{\lambda_2}) & |\frac{a_j}{\lambda_2}|>1.\end{cases}$$

若 $J(\hat{t}) \leq 1$, 则 $\theta = 0$. 否则，需最小化

$$\frac{1}{2}\sum_{i=1}^{N}(r_i - \sum_{j=1}^{k} Z_{ij}\theta_j)^2 + \lambda_1\|\theta\|_2 + \lambda_2\sum_{j=1}^{k}|\theta_j|. \tag{3.57}$$

上式对 θ_j 求导，可知：若 $\|Z_j^T(r - \sum_{k\neq j} Z_k\theta_k)\| < \lambda_2$, $\theta_j = 0$; 否则，上式可用最速下降法求解。综上，得具体算法如下：

(1)置 $\beta = \beta_0$ (β_0 为任意初始值), $l = 1$

(2)对于 l 因子,令 $r = Y - \sum_{k\neq l} X_k\beta_k$, $X_l = (Z_1, Z_2, \ldots, Z_k)$, $\beta_l = (\theta_1, \theta_2, \ldots, \theta_k)$, $w_j = (w_1, w_2, w_N) = r - \sum_{k\neq j} Z_k\theta_k$. 如果 $J(\hat{t}) \leq 1$(3.56式), 则 $\beta_l = 0$. 否则，对于 $j = 1, 2, \ldots, k$, 如果 $|Z_j^T w_j| < \lambda_2$, 则 $\theta_j = 0$;$|Z_j^T w_j| > \lambda_2$ 时，最小化

$$\frac{1}{2}\sum_{i=1}^{N}(w_i - \sum_{j=1}^{k} Z_{ij}\theta_j)^2 + \lambda_1\|\theta\|_2 + \lambda_2\sum_{j=1}^{k}|\theta_j|$$

可以采取一个一个最小化直到收敛的方法。

(3)对于 $l = 1, 2, \ldots, J$, 重复步骤(2),直到收敛。

对于 $\lambda_2 = 0$, 即退化为 group Lasso 模型，只需在上述步骤(2)中换为(3.53)判断 $\beta_l = 0$ 与否，且不需要判断 $|Z_j^T w_j| < \lambda_2$.

第四章　CAP惩罚聚类因子模型

由上一章可以看出，L_1 范数惩罚可以保证参数估计的稀疏性。在这一章中我们介绍 CAP 惩罚项，它作为 Lasso 和岭回归德推广体现在两个方面：它是由多种范数惩罚形成；它允许各聚类因子之间有交叠。

设模型参数为 β, 观测数据为 Z, 损失函数为 $L(Z,\beta)$(假定L为凸), 则 L_r 范数惩罚下的参数估计为：

$$\hat{\beta}_r(\lambda) = \arg\min_{\beta}[L(Z,\beta) + \lambda T(\beta)]. \tag{4.1}$$

$$T(\beta) = \|\beta\|_r = (\sum_{j=1}^{J} |\beta_j|^r)^{\frac{1}{r}} \tag{4.2}$$

考虑 $T(\beta) \leq t$ 形成的区域，当 $0 < r < 1$ 时，区域不是凸的，(4.1)的解具有稀疏性。当 $r \geq 1$, 由 KKT 条件(4.1)的解满足：

$$\frac{\partial L}{\partial \beta_j} = -\lambda \cdot \text{sign}(\beta_j)\frac{|\beta_j|^{r-1}}{\|\beta\|_r^{r-1}},\ \beta_j \neq 0; \tag{4.3}$$

$$|\frac{\partial L}{\partial \beta_j}| \leq \lambda\frac{|\beta_j|^{r-1}}{\|\beta\|_r^{r-1}},\ \beta_j = 0. \tag{4.4}$$

由上面两式可以看出，对于 $1 < r \leq \infty$，可知估计 $\hat{\beta}_j = 0$ 的充要条件是 $\frac{\partial L}{\partial \beta_j}|_{\beta_j=0} = 0$. 这个条件在 L 是严格凸和 Z 的分布连续时是不成立的，这种情况下参数估计几乎都不为零。

4.1　CAP族

4.1.1　CAP惩罚

假定 p 个因变量 x_j 都应经标准化。令 $\mathcal{I} = \{1,\ldots,p\}$，记 K 个组为 $\mathcal{G}_k \subset \mathcal{I}, k = 1,\ldots,K$. 记范数参量为 $\gamma = (\gamma_0,\ldots,\gamma_K)$. 命 L_{γ_0} 为全局范数，$L_{\gamma_k}(k > 0)$ 为第 k 组内的范数。定义：

$$\beta_{\mathcal{G}_k} = (\beta_j)_{j\in\mathcal{G}_k}$$

$$N_k = \|\beta_{\mathcal{G}_k}\|_{\gamma_k},$$

$$N = (N_1, \ldots, N_K). \tag{4.11}$$

定义 CAP 惩罚为：

$$T_{\mathcal{G},\gamma}(\beta) = \|N\|_{\gamma_0}^{\gamma_0} = \left[\sum_k |N_k|^{\gamma_0}\right] \tag{4.12}$$

则 CAP 下的估计为：

$$\hat{\beta}_{\mathcal{G},\gamma}(\lambda) = \arg\min_{\beta}[L(Z,\beta) + \lambda T_{\mathcal{G},\gamma}(\beta)]. \tag{4.13}$$

类似上一章，将组看做因子，CAP惩罚可以进行因子选择和分等级的选择。

4.1.2 因子选择：无交叠情形

我们将 $x_j,\ j = 1, \ldots, p$ 进行无交叠分组，这样利用 CAP 选择就可以选出重要的因子而排除其他因子。在各因子间，用 L_{γ_0} 范数惩罚进行选择；在因子内部，L_{γ_k} 范数决定了 $\beta_{\mathcal{G}_k}$ 内部的关系。特别的，对于 $\gamma_0 = 1$, 保证了因子间的稀疏性。$\gamma_k > 1$ 描述了组内变量的凝聚程度，即它们之间的距离。形成组进行拟合可以减小预测误差。为了平衡组大小不同造成的各组在选择时地位的不同，可以在各组的惩罚项中根据组的大小加入权重。

4.1.3 分等级的选择：组窃套情形

在线性模型中，我们有时需要进行分等级的选择。如当我们考虑各因变量及它们的交叉项时，如果假定只有主变量进入模型时，才能考察其对应的交叉变量，这样主变量就比交叉变量具有优先等级。利用CAP惩罚可以利用组相互嵌套解决这种假定下的模型选择。这基于以下的结论：

定理4.1.

设 $\mathcal{I}_1,\ \mathcal{I}_2 \subset \{1, \ldots, p\}$, **若**:

- $\gamma_0 = 1,\ \gamma_k > 1,\ \forall k = 1, \ldots, K$
- $\mathcal{I}_1 \subset \mathcal{G}_k \Rightarrow \mathcal{I}_2 \subset \mathcal{G}_k,\ \forall k$
- $\exists k^*,\ s.t \mathcal{I}_2 \subset \mathcal{G}_{k^*},\ \mathcal{I}_1 \not\subset \mathcal{G}_{k^*}$

则当 $\beta_{\mathcal{I}_2} \neq 0,\ \beta_{\mathcal{I}_1} = 0$ **时**，$\frac{\partial}{\partial \beta_{\mathcal{I}_1}} T(\beta) = 0$

由以上定理可以看出，当 $\{Z : \frac{\partial}{\partial \beta_{\mathcal{I}_1}} L(Z,\beta)|_{\beta_{\mathcal{I}_1}=0} = 0\}$ 是零测集时，$\mathcal{I}_2$ 在 $\mathcal{I}_2$ 后面加入模型。等级关系可以用数的形式表示，用节点表示一个组，每一个节点下面

连的是比它低一等级的组。则 CAP 惩罚可以取为

$$T(\beta)=\sum_{k=1}^{K}\|(\beta_{\mathcal{G}_k},\beta_{alldescendantsof\mathcal{G}_k})\|. \tag{4.14}$$

4.2 CAP惩罚下的参数估计

在本节中需要解决两个问题：给定 λ 下模型中参数的估计和 λ 的选择。

4.2.1 参数估计

根据Boyd和Vandenberghe凸最优化的理论(2004),若(4.14)中的目标函数是凸的，则满足KKT条件的解就是(4.14)的解。我们有如下定理：

定理4.2.

如果 $\gamma_i \geq 1,\ \forall i=0,\ldots,K$, 则(4.12)是凸的。在若损失函数 $L(Z,\beta)$ 也是凸的，则(4.14)中的目标函数是凸的。

下面介绍一般算法和特殊情形下的算法(L_2 损失和 $\gamma_0=1,\ \gamma_k\equiv\infty$).

4.2.1.1 BLasso 算法

BLasso算法由Zhao.P和Yu.B提出，具体步骤如下：

(1)给定很小的常数 $\epsilon>0,\xi>0$, 计算

$$(\hat{j},\hat{s}_{\hat{j}})=\arg\min_{s=\pm\epsilon,j}\sum_{i=1}^{n}L(Z_i;s1_j)$$

$$\hat{\beta}^0=\hat{s}_{\hat{j}}1_{\hat{j}}.$$

1_j 表示第 j 个为1，其他为0的向量。计算

$$\lambda^0=\frac{1}{\epsilon}\sum_{i=1}^{n}(L(Z_i;0)-L(Z_i;\hat{\beta}^0))$$

置当前活跃集为 $I_A^0=\{\hat{j}\}$, t=0.

(2)(向后和向前步)计算

$$\hat{j}=\arg\min_{j\in I_A^t}\sum_{i=1}^{n}L(Z_i;\hat{\beta}^t+s_j1_j)$$

其中 $s_j=-sign(\hat{\beta}_j^t)\epsilon$.

记 $\Gamma(\beta;\lambda)=L(Z;\beta)+\lambda T(\beta)$. 若 $\Gamma(\hat{\beta}^t+\hat{s}_{\hat{j}}1_{\hat{j}},\ \lambda^t)-\Gamma(\hat{\beta}^t,\ \lambda^t)\leq-\xi,$ 则：

$$\hat{\beta}^{t+1}=\hat{\beta}^t+\hat{s}_{\hat{j}}1_{\hat{j}},\ \lambda^{t+1}=\lambda^t.$$

否则，

$$(\hat{j}, \hat{s}) = \arg \min_{s=\pm\epsilon, j} \sum_{i=1}^{n} L(Z_i; \hat{\beta}^t s1_j),$$

$$\hat{\beta}^{t+1} = \hat{\beta}^t + \hat{s}1_{\hat{j}},$$

$$\lambda^{t+1} = \min[\lambda^t, \frac{1}{\epsilon}(L(Z; \hat{\beta}^t) - L(Z; \hat{\beta}^{t+1}) - \xi)],$$

$$I_A^{t+1} = I_A^t \cup \{\hat{j}\}.$$

(3)t=t+1, 重复(2)和(3)直到 $\lambda^t \leq 0$ 时停止。

BLasso 算法实际上是在损失函数最陡的梯度方向上前进，通过调整前进的步长可以在预测精确度和计算的时间代价之间进行选择。

4.2.1.2 分段线性路径

这一小节讨论 L_2 损失下，$\gamma_0 = 1$, $\gamma_k \equiv \infty$ 情形的参数估计。由Rosset等(2004)结论知，这种情况下的解路径是分段线性的。由此可以从一个点跳到另一个点，简化算法。下面分别介绍无交叠因子选择和嵌套分等级选择的参数估计(iCAP算法和hiCAP算法)。

icap算法：对于给定 λ 时，令第 k 组的相关系数为 $c_k = \|X'_{\mathcal{G}_k}(Y - X\beta)\|_1$, 活跃集 $\mathcal{A} = \{j : |c_j| = \max_{k=1,\ldots,K} |c_k|\}$. 在活跃集外，置参数为零。若 $\hat{\beta}$ 为解，则由 KKT 条件易知，每一个 $j \in \mathcal{G}_k$ 一定属于下面两个集合之一：$\mathcal{U}_k = \{j \in \mathcal{G}_k : X'_j(Y - X\hat{\beta}) = 0\}$ 和 $\mathcal{R}_k = \{j \in \mathcal{G}_k : \hat{\beta}_j = \|\hat{\beta}_{\mathcal{G}_k}\|_\infty\}$.
对于 $\lambda_0 = \max_{j\in 1,\ldots,K} \|X_{\mathcal{G}'_j} Y\|_1$, $\hat{\beta}(\lambda_0) = 0$. 从这一点出发，找到一个前进方向 $\Delta\hat{\beta}$ 使得对于很小的 $\delta > 0$, $\hat{\beta}(\lambda_0) + \delta\Delta\hat{\beta}$ 满足KKT条件。然后寻找下一个点：计算最小的 δ, 使下列情况至少发生一个

- $\mathcal{A}$ 增加或减少一个组,
- 存在k, $\mathcal{U}_k$ 和 $\mathcal{R}_k$ 中元素有转移，
- 在活跃集外的组，与当前残差的相关系数的符号发生变化;

如果不存在这样的 δ, 则导致非正规解。

hiCAP算法 对于分等级的选择，且因子具有树的结构，也可以根据 KKT 条件构建算法。$hiCAP$ 算法比较长，这里仅简单介绍一下思想。算法开始先构建无交叠的组，使：

- 每个组由一个子树构成;
- 将每一个子树看作一个超节点，则由这些超节点形成的超树必须满足在超节点

中 Y 与 X 的平均相关系数(带绝对值的)比其子代高。

因为根部的组具有最高的平均相关系数，所以从根部组的系数开始前进，保持：

• 每个超节点中 $Y - X\hat{\beta}$ 和 X 的相关系数(带绝对值的)不低于其子代;

• 每个超节点内的系数的绝对值不小于其子代。

在两个断点之间，找一个方向使 KKT 条件成立。断点根据以下特征寻找：

• 如果 $Y - X\hat{\beta}$ 与一个超节点包含的子树 $\mathcal{G}_a$ 的平均相关系数等同于其超节点，则 $\mathcal{G}_a$ 脱离成一个超节点。

• 如果超节点 a 的最大系数(带绝对值的)与其子超节点 b 相同，则它们合并成一个新的超节点。• 如果一个超节点的系数全为零，且其一个子代的平均相关系数(带绝对值的)与其相同，则它们合并。

4.2.2 λ的选择

我们选择用Sugiura的 AIC_c 准则，用k表示线性模型的维数，则：

$$AIC_c = \frac{n}{2}\log\left(\sum_{i=1}^{n}(Y_i - X_i\hat{\beta}(\lambda))^2\right) + \frac{n}{2}\left(\frac{1+\frac{k}{n}}{1-\frac{k+2}{n}}\right).$$

模型的选择是使上式最小化。

与前几章类似，自由度的估计则根据在某一点的自由参数的个数，特别的，对于 iCAP 算法，$\hat{df}(\lambda) = |\mathcal{A}(\lambda)| + \sum_{k\in\mathcal{A}(\lambda)} |\mathcal{U}_k(\lambda)|$.

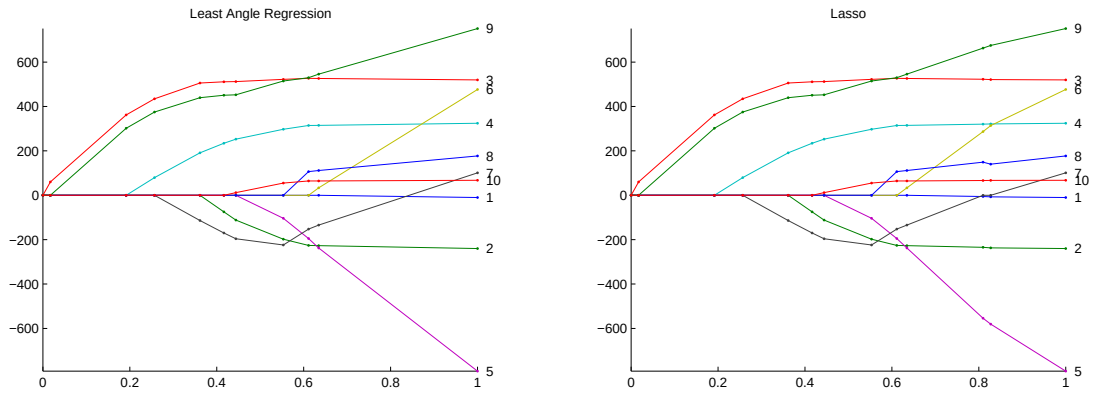
第五章　计算机模拟

在这一章中我们针对具体例子进行LARS，Lasso，BLasso，Group Lasso CAP，hiCAP算法的模拟，这样不仅对各种算法有个直观的认识，还可以进行相互的比较。

例一：Diabetes数据

病号	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	y
1	59	2	32.1	101	157	93.2	38	4	4.9	87	151
2	48	1	21.6	87	183	103.2	70	3	3.9	69	75
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
441	36	1	30.0	95	201	125.2	42	5	5.1	85	220
442	36	1	19.6	71	250	133.2	97	3	4.6	92	57

在上表中，一共有442个病人，$x_1 - x_{10}$ 代表病人的十个身体指标：年龄，性别，体重指数，平均血压，和六个血清测量数据；y代表病人的疾病情况。我们需要用 x 来建立模型拟合 y. 这里有两个目标：一是对于新来的病人，能够较准确的预测；二是提取出对 y 影响较大的变量。



(a) lars　　(b) lasso

图 5.1: 横坐标为当前系数的L_1和对OLS解L_1和的比值，纵坐标为系数的值。

图5.1是利用第二章的LARS算法和修订的LARS算法得出的，可以看出LARS和Lasso的解路径是相同的，即变量加入模型的顺序一致；且他们的解曲线也相似，除了 x_8 的末端有明显不同。

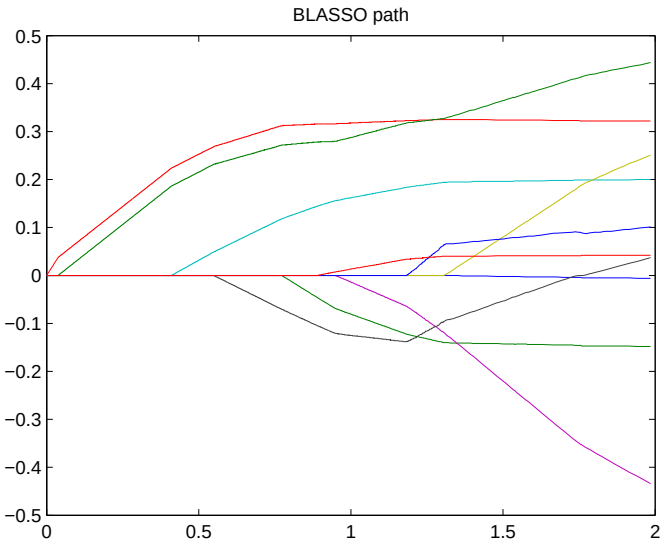
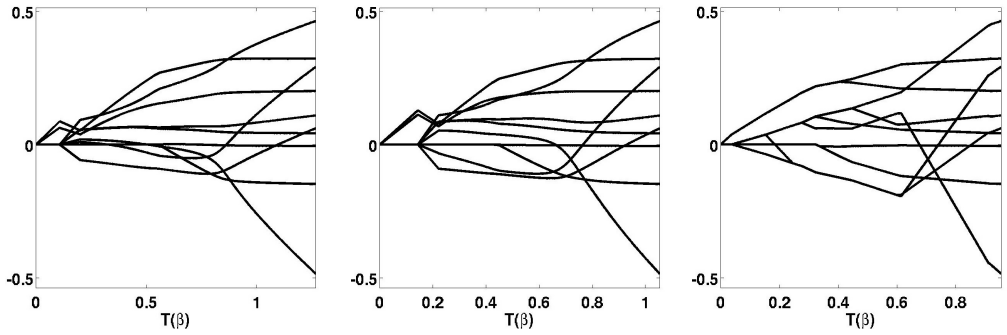


图 5.2: 横坐标为系数的L_1和$(\times 10^{-3})$，纵坐标为系数的值$(\times 10^{-3})$.

图5.2是利用BLasso算法对Lasso模型的求解。可以看出这与5.1(b)是相同的。



(a) (b) (c)

图 5.3: $\gamma_0 = 1$(a)GLasso($\gamma_k \equiv 2$); (b)CAP($\gamma_k \equiv 4$); (c)iCAP($\gamma_k \equiv \infty$);

将变量分为三组：$x_1 - x_2$; $x_3 - x_4$; $x_5 - x_10$。图5.3利用其他算法比较了各个解路径。

第六章　总结与讨论

本文从线性模型出发，首先介绍了 LARS 算法，我们看到 LARS 算法只用到因变量与响应变量的相关位置，却能经过简单调整解决系数带约束的 Lasso 模型。这反应了 Lasso 模型本身的特性：当损失函数为凸时，一般会在约束区域的尖角上取得解。然后，我们将原始变量进行分组，考虑因子线性模型。这将一般线性模型的解法将以推光就可以解决。为了保证解系数的稀疏性，对因子间与因子内的系数分别作限制是必要的。从以上可以看到，模型中加入惩罚项可以优化模型，最后介绍了 CAP 惩罚。CAP 惩罚囊括了 Lasso，岭回归等模型。它对解决的一般的因子选择和分等级的选择提供了一般的算法。

LARS 算法计算简便高效，用它来解决 Lasso 是很好的选择。但当因变量个数远大于观测个数时，一步活跃集中改变的不止一个，这样定理2.1不适用。而在生物实验或其他实际问题中，这种情况是常见的。CAP 的预测比 Lasso 更好，且可以建立有交叠的分组来进行等级选择，对于有交叉效应的问题是最好的选择。但这些分组和等级结构都是预先假定的，怎样根据数据建立这些以与 CAP 更好的配合，将是值得考虑的。

这些模型都假定因变量 x 是固定的，如果我们的观测 x 具有人为误差，则 x 是一个具有一定分布的随机变量。对于这样的模型，我们一般用贝叶斯后验分布求解。为了能够选出重要的变量，我们是否可以同 Lasso 一样对系数进行一定的限制，而使最后估计的某些系数为零？这是一个研究的问题。

参考文献

[1] Tibshirani，R.(1996)，Regression Shrinkage and Selection via the Lasso，*J. Royal. Statist. Soc. B.* **58**，267-288.

[2] Breiman，L.(1995)，Better Subset Regression Using the Nonnegative Garrote，*Technometrics*，**37**，373-384.

[3] Efron，B.，Johnstone，I.，Hastie，T. and Tibshirani，R.(2004)，Least Angle Regression，*Ann. Statist.*，**32**，407-499.

[4] Boyd，S. and Vandenberghe，L.(2004)，Convex Optimization. Cambridge University Press.

[5] Yuan，M. and Lin，Y.(2006)，Model Selection and Estimation in Regression with Grouped Variables，*Journal of the Royal Statistical Society, Series B.* **68**，49-67.

[6] Peng Zhao，Guilherme Rocha and Bin Yu.(2009)，The Composite Absolute Penalties Family for Grouped and Hierarchical Variable Selection，*Ann. Stat.*，**37**，3468-3497.

[7] Jerome Friedman，Trevor Hastie and Robert Tibshirani.(2010 preprint)，A Note on the Group Lasso and a Sparse Group Lasso.

致　　谢

值此论文完成之际，谨在此向多年来给予我关心和帮助的老师、同学、朋友和家人表示衷心的感谢！

……