第6-5章 Variable Selection

部分Slides参考

Tibshirani: www-stat.stanford.edu/~tibs/ftp/**lassotalk**.pdf

Chapter 3 of Hastie, Tibshirani and Friedman: Elementary of Statistical Learning

Chapter2 of Buhlmann, Statistics for high-dimensional data

从线性回归谈起

- 令

$$Y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \qquad i = 1, 2, \cdots, n$$

- 其中

$$E(\epsilon_i) = 0, var(\epsilon_i) = \sigma^2$$

- 如果将Y和X_j去中心化，则截距项 $\beta_0 = 0,$ 于是模型可以简化为

$$Y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \qquad i = 1, 2, \cdots, n$$

从线性回归谈起

- 记

$$Y=\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, X=\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \cdots & \cdots & \cdots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}, \beta=\begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}$$

- 其中X称之为设计矩阵，于是有

$$\begin{cases} Y = X\beta + \epsilon \\ \epsilon \sim N(0, \sigma^2 I_n) \end{cases}$$

回归的含义

- 给定数据 $(x_i, y_i), i = 1, \cdots, n.$ 我们希望得到协变量X和响应变量Y的函数关系

$$y_i = m(x_i) + \epsilon_i$$

- 另一个含义：如果把变量X,Y都看成随机变量，而把数据看成随机变量的实现, 那么m(x)就是X=x下Y的期望, 即

$$m(x) = E(Y|X = x)$$

- 于是回归的目的就是对函数m(x)进行估计

最小二乘估计

- 定义残差平方和

$$RSS = \sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 = (Y - X\beta)^T(Y - X\beta)$$

- 那么最小二乘估计

$$\hat{\beta} = \operatorname*{Argmax}_{\beta} RSS$$

最小二乘求解

- 求导

$$\frac{\partial RSS}{\partial \beta_j} = -2\sum_{j=1}^{n} x_{ij}(y_i - \sum_{k=1}^{p} x_{ik}\beta_k) = 0$$

$$\sum_{i=1}^{n} x_{ij}(y_i - \sum_{k=1}^{p} x_i k\beta_k) = 0$$

$$\sum_{k=1}^{p}(\sum_{i=1}^{n} x_{ij}x_{ik}) = \sum_{i=1}^{n} x_{ij}y_i, \quad j = 1, 2, \cdots p$$

- 写成矩阵形式即

$$(X^T X)_{p\times p}\beta = X^T Y$$

最小二乘解

- 如果矩阵X^TX可逆，则

$$\hat{\beta} = (X^TX)^{-1}X^TY$$

- 那么回归函数

$$m(x) = X\hat{\beta} = X(X^TX)^{-1}X^TY = \hat{Y}$$

帽子矩阵

- 定义帽子矩阵

$$L_{n\times n} = X(X^TX)^{-1}X^T$$

- L满足

$$L^T = L, L^2 = L$$

- 那么

$$m(X) = \begin{pmatrix} m(x_1) \\ \vdots \\ m(x_n) \end{pmatrix} = L_{n\times n}Y_{n\times 1}$$

最小二乘估计的统计性质

- 无偏性(Unbiased)

$$E(\hat{\beta}) = (X^TX)^{-1}X^TEY = (X^TX)^{-1}X^TX\beta = \beta$$

- 方差(条件：ε_i i.i.d, $Var(\varepsilon)=\sigma^2 I_n$)

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

- 故

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

Gauss-Markov定理

- Gauss-Markov定理：所有线性无偏估计中，参数β的最小二乘估计具有最小方差。
- 考虑参数估计的均方误差

$$
\begin{aligned}
MSE(\theta) &= E(\tilde{\theta} - \theta)^2 \\
&= Var(\tilde{\theta}) + [E(\tilde{\theta}) - \theta]^2
\end{aligned}
$$

- 估计的均方误差和预测误差相关

$$E(Y - X_0^T\tilde{\beta})^2 = \sigma^2 + E(X_0^T\beta - X_0^T\tilde{\beta})^2 = \sigma^2 + MSE(X_0^T\tilde{\beta}))$$

- 一种可能性：有偏的估计可能具有更小的均方误差

最小二乘估计的统计性质

- 方差的无偏估计

$$
\begin{aligned}
\hat{\sigma}^2 &= \frac{1}{N-p-1}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2 \\
&= \frac{1}{N-p-1}\sum_{i=1}^{N}\hat{\epsilon}_i^2
\end{aligned}
$$

- 这里自由度减少了p+1个是因为估计了p个回归系数，加上中心化又减少了一个自由度。

平方和分解

- 平方和分解

$$TSS = \sum_{i=1}^{n}(y_i - \overline{y})^2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 + \sum_{i=1}^{n}(\hat{y}_i - \overline{y})^2$$
$$\triangleq RSS + MSS$$

- 其中TSS总偏差平方和; RSS残差平方和(由随机误差引起); MSS回归平方和(由回归的好坏决定);

回归的评价--判定系数

- R^2

$$R^2 = \frac{MSS}{TSS} = 1 - \frac{RSS}{TSS}$$

- 自由度调整R^2

$$\overline{R}^2 = 1 - \frac{RSS/(N-p-1)}{TSS/(N-1)}$$

回归方程的检验

- 检验问题

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_p = 0$$
$$H_A : \beta_1, \cdots, \beta_p \text{ At least one not } 0$$

- 如果随机误差满足：$\epsilon_i \sim N(0, \sigma^2), i.i.d$, 则在H_0下

$$F = \frac{MSS/p}{RSS/(N-p-1)} \sim F(p, N-p-1)$$

回归系数的检验

- 检验问题

$$H_0^i : \beta_i = 0$$
$$H_A^i : \beta_i \neq 0$$

- 如果随机误差满足：$\epsilon_i \sim N(0, \sigma^2), i.i.d$ ，则在H^i_0下

$$F_i = \frac{P_i}{RSS/(N-p-1)} \sim F(1, N-p-1)$$
$$t_i = \frac{\hat{\beta}_i/\sqrt{l^{ii}}}{\sqrt{RSS/(N-p-1)}} \sim t(N-p-1)$$

回归系数的检验

- 其中分子是偏回归平方和，是全部变量的回归平方和与去掉第i个变量之后的回归平方和的差

$$P_i = MSS - MSS(i)$$

- 可以证明

$$P_i = \hat{\beta}_i^2 / l^{ii}$$

- 其中 l^{ii} 为L的逆的第i个对角元素。

$$L = X^T X$$

Variable Selection Problem

- A common problem is that there is a large set of candidate predictor variables.
- Goal is to choose a small subset from the larger set so that the resulting regression model is **simple**, yet have **good predictive ability**.

Two basic Methods of Selecting Predictors

- **Stepwise regression**: **Enter and remove predictors, in a stepwise manner, until there is no justifiable reason to enter or remove more.**
- **Best subsets regression**: Select the subset of predictors that do the best at meeting some well-defined objective criterion.

Stepwise Regression: the Idea

- Start with no predictors in the "**stepwise model**."
- At each step, enter or remove a predictor based on partial *F*-tests (that is, the *t*-tests).
- Stop when no more predictors can be justifiably entered or removed from the stepwise model.

Drawbacks of Stepwise Regression

- The final model is not guaranteed to be optimal in any specified sense.
- The procedure yields a single final model, although in practice there are often several equally good models.
- It doesn’t take into account a researcher’s knowledge about the predictors.

Stepwise Regression Methods

- Three broad categories:
  - Forward selection
  - Backward elimination
  - Stepwise regression

Forward Selection

- Start the model with intercept term only
- Add one regressor with largest F value for testing significance of candidate regressor with $F > F_{IN} = F_{\alpha,1,p}$.
- Choose a regressor with largest partial F-statistic,
- If $F>F_{IN}$, then x_2 is added.
- Procedure terminates either when the partial F-stastic at a particular step does not exceed F_{IN} or when the last candidate regressor is added.

Backward Elimination

- Start with a model with all K candidate regressors.
- The partial F-statistic is computed for each regressor, and drop a regressor which has the smallest F-statistic and < F_{OUT}.
- Stop when all partial F-statistics > F_{OUT}.

Stepwise Regression

- A modification of forward selection.
- A regressor added at an earlier step may be redundant. Hence this variable should be dropped from the model.
- Two cutoff values: F_{OUT} and F_{IN}
- Usually choose $F_{IN} > F_{OUT}$: more difficult to add a regressor than to delete one.

Stepwise Regression

- A modification of forward selection.
- A regressor added at an earlier step may be redundant. Hence this variable should be dropped from the model.
- Two cutoff values: F_{OUT} and F_{IN}
- Usually choose $F_{IN} > F_{OUT}$: more difficult to add a regressor than to delete one.

Stepwise regression: Step #1

1. Fit each of the one-predictor models, that is, regress *y* on x_1, regress *y* on x_2, … regress *y* on x_{p-1}.
2. The first predictor put in the stepwise model is the predictor that has the **largest partial F -value** ($F > F_{IN}$).
3. If no partial F-value$>F_{IN}$, stop.

Stepwise regression:
Step #2

1. Suppose x_1 was the "best" one predictor.
2. Fit each of the two-predictor models with x_1 in the model, that is, regress y on (x_1, x_2), regress y on (x_1, x_3), …, and y on (x_1, x_{p-1}).
3. The second predictor put in stepwise model is the predictor that has the **largest partial F-value** ($F>F_{IN}$).
4. If no partial F-value$>F_{IN}$, stop.

Stepwise regression:
Step #2 (continued)

1. Suppose x_2 was the "best" second predictor.
2. Step back and check again partial *F*-value for x_1. If the partial F-value<F_{OUT}, remove x_1 from the stepwise model.

Stepwise regression: Step #3

1. Suppose both x_1 and x_2 made it into the two-predictor stepwise model.
2. Fit each of the three-predictor models with x_1 and x_2 in the model, that is, regress y on (x_1, x_2, x_3), regress y on (x_1, x_2, x_4), …, and regress y on (x_1, x_2, x_p).

Stepwise regression: Step #3 (continued)

1. The third predictor put in stepwise model is the predictor that has the **largest partial *F*-value** ($F>F_{IN}$).
2. If no partial F-value$>F_{IN}$, stop.
3. Step back and check partial F-value for x_1 and x_2. If either partial F-value$<F_{OUT}$, remove the predictor from the stepwise model.

Stepwise Regression: Stopping the Procedure

- The procedure is stopped when adding an additional predictor does not yield a **partial F-value>F_{IN}.**

Lasso Model

- Lasso: Least Absolute Shrinkage and Selection Operator
- Minimize

$$\min_{\beta} \sum_{i=1}^{n} \frac{1}{2}(y_i - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- Equivalent to minimizing sum of squares with constraint (Lagarangian function)

$$\sum_{j=1}^{p} |\beta_j| \leq s$$

Lasso Explanation

- The bound "s" is a tuning parameter. When "s" is large enough, the constraint has no effect and the solution is just the usual multiple linear least squares regression of y on x_1, x_2, ...x_p.
- However when for smaller values of s (s>=0) the solutions are shrunken versions of the least squares estimates. Often,some of the coefficients b_j are zero. Choosing "s" is like choosing the number of predictors to use in a regression model, and cross-validation is a good tool for estimating the best value for "s".

http://statweb.stanford.edu/~tibs/lasso/simple.html

Ridge Regression

- Minimize

$$\min_{\beta} \frac{1}{2} \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} \beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p} |\beta_j^2|$$

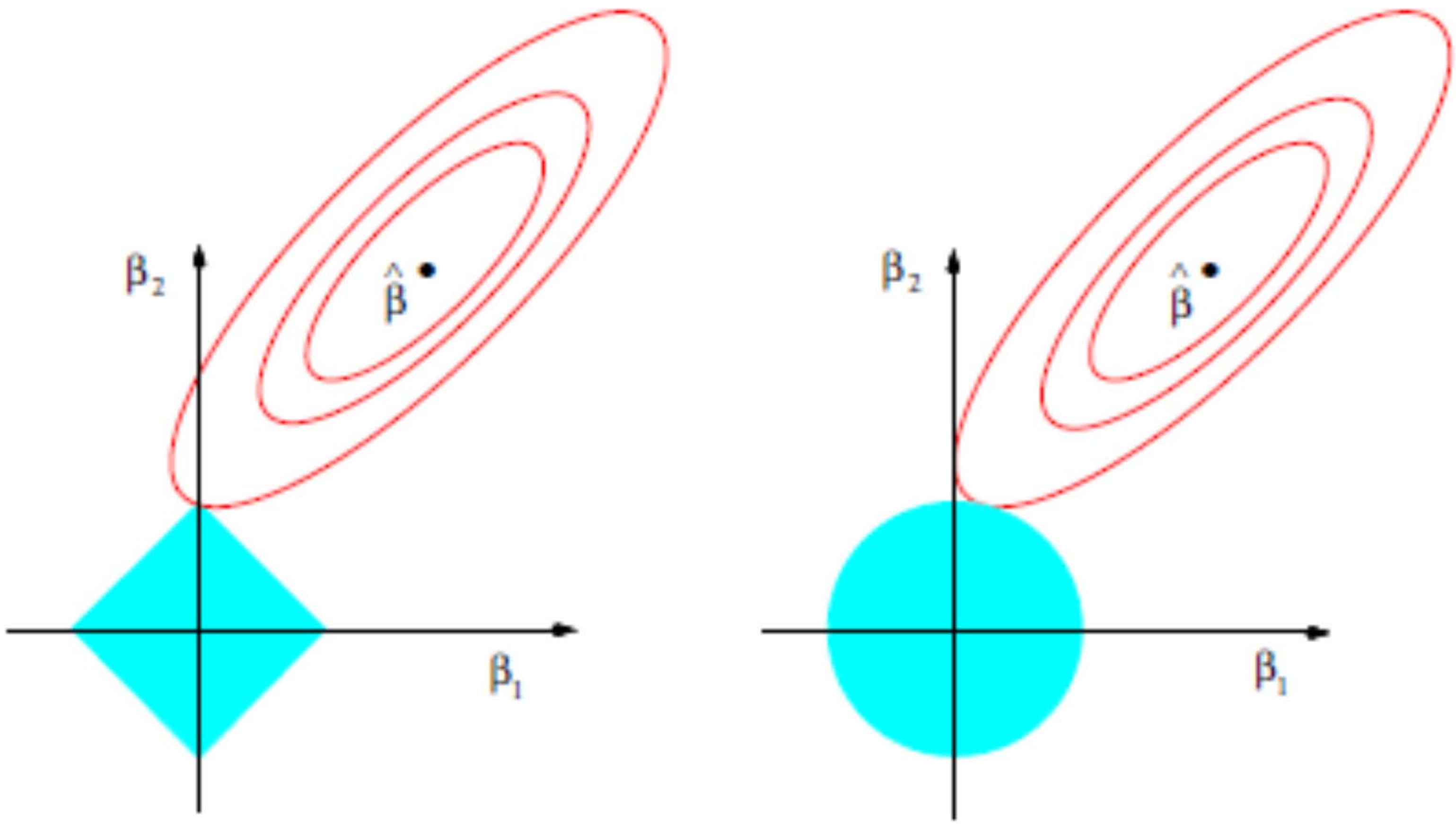
- Equivalent to minimizing sum of squares with constraint

$$\sum_{j=1}^{p} |\beta_j|^2 \leq s$$

- Close-form solution

$$\beta^{\text{ridge}} = (X^T X + \lambda I)^{-1} X^T Y$$

Picture of Lasso and Ridge Regression



Algorithms for Lasso

- Standard convex optimizer
- Least angle regression (LAR) - Efron et al 2004- computes
- Entire path of solutions. State-of-the-Art until 2008
- Pathwise coordinate descent---New

LASSO求解

- 定理： Denote the gradient of $(2n)^{-1}||Y - X\beta||_2^2$ by $G(\beta) = -X^T(Y - X\beta)/n$. Then a necessary and sufficient condition for $\hat{\beta}$ to the solution of lasso problem is:

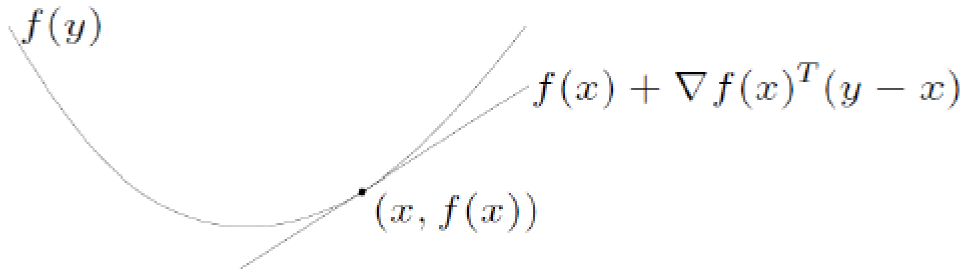
$$\begin{cases} G_j(\hat{\beta}) = -\text{sign}(\hat{\beta}_j)\lambda & \text{if } \hat{\beta}_j \neq 0 \\ |G_j(\hat{\beta})| \leq \lambda & \text{if } \hat{\beta}_j = 0 \end{cases}$$

- Moreover, if the solution is not unique (e.g. if n>p) and $G_j(\hat{\beta}) < \lambda$ for some solution $\hat{\beta}$, then $\hat{\beta}_j = 0$ for all solutions.

次梯度(Subgradient)

- 设凸函数 $f : \Omega \to R, \Omega$ 是R^n上的凸集, 如果函数在x_0点不可微，则满足

$$f(x) \geq f(x_0) + \nabla f(x_0)(x - x_0)$$



[Boyd & Vandenberghe]

- 这里 $\nabla f(x_0)$ 是梯度

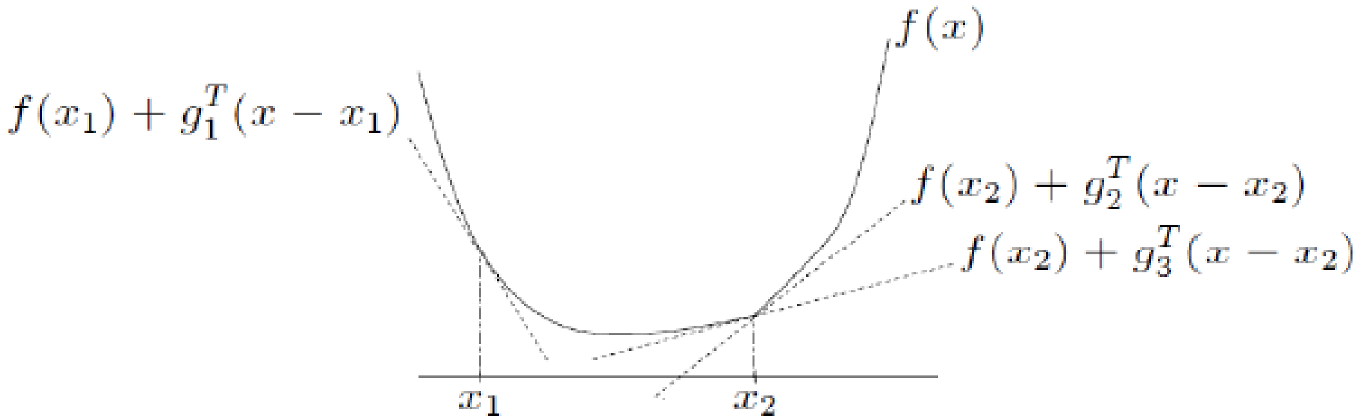
http://select.cs.cmu.edu/class/10725-S10/recitations/r7/Subgradients.pdf

次梯度(Subgradient)

- 设凸函数 $f:\Omega\to R,\Omega$ 是R^n上的凸集, 如果函数在x_0点不可微，如果存在向量g，使得

$$f(x)\geq f(x_0)+g^T(x-x_0)$$

则称向量g为x_0处的次梯度。

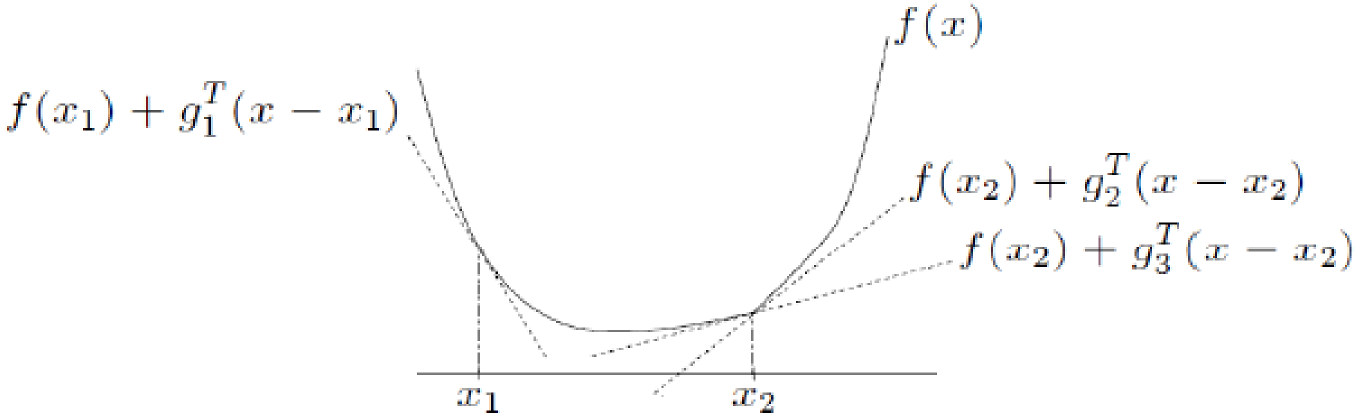


[Boyd & Vandenberghe]

http://select.cs.cmu.edu/class/10725-S10/recitations/r7/Subgradients.pdf

次微分

- 对于一个给定的点，可能不止一个这样的次梯度存在，而是一个次梯度集合，这样的集合称为次微分，记为 ∂f.



[Boyd & Vandenberghe]

http://select.cs.cmu.edu/class/10725-S10/recitations/r7/Subgradients.pdf

次微分例子

- 例： $f(x) = |x|$
- 在0点处不可微，由次梯度的定义

$$\begin{cases} f(x) - f(0) \geq g^T(x-0) \\ g^T \leq \dfrac{f(x)}{x} \in [-1,1] \end{cases}$$

http://select.cs.cmu.edu/class/10725-S10/recitations/r7/Subgradients.pdf

次微分性质

- 可加性: $\partial(f_1 + f_2) = \partial f_1 + \partial f_2$

- 伸缩性: $\partial(\alpha f) = \alpha \partial f \quad (\alpha > 0)$

- 仿射变换: $\partial f(Ax + b) = A^T \partial f(Ax + b)$

- 链锁法则: $\partial f(g(x)) = (\partial f)(g(x)) \partial g(x)$

次微分和优化

- 如果f是可微的凸函数

$$f(x^*) = \min_x f(x) \quad \Leftrightarrow \quad 0 = \nabla f(x^*)$$

- 如果f是不可微的凸函数

$$f(x^*) = \min_x f(x) \quad \Leftrightarrow \quad 0 \in \partial f(x^*)$$

KKT条件

- 对于下面的优化问题

$$
\begin{aligned}
&\min f(x) \\
&\text{subject to} : h_i(x) \leq 0, i = 1, \cdots, m \\
&\qquad\qquad\quad l_j(x) = 0, j = 1, \cdots, r
\end{aligned}
$$

- KKT 条件

  - $0 \in \partial f(x) + \sum_{i=1}^{m} u_i \partial h_i(x) + \sum_{j=1}^{r} v_j \partial \ell_j(x)$ (stationarity)
  - $u_i \cdot h_i(x) = 0$ for all i (complementary slackness)
  - $h_i(x) \leq 0, \ \ell_j(x) = 0$ for all i, j (primal feasibility)
  - $u_i \geq 0$ for all i (dual feasibility)

KTT for Lasso

Let's return the lasso problem: given response $y \in \mathbb{R}^n$, predictors $A \in \mathbb{R}^{n \times p}$ (columns $A_1, \ldots A_p$), solve

$$\min_{x \in \mathbb{R}^p} \frac{1}{2}\|y - Ax\|^2 + \lambda\|x\|_1$$

KKT conditions:

$$A^T(y - Ax) = \lambda s$$

where $s \in \partial\|x\|_1$, i.e.,

$$s_i \in \begin{cases} \{1\} & \text{if } x_i > 0 \\ \{-1\} & \text{if } x_i < 0 \\ [-1, 1] & \text{if } x_i = 0 \end{cases}$$

Now we read off important fact: if $|A_i^T(y - Ax)| < \lambda$, then $x_i = 0$

... we'll return to this problem shortly

Orthonormal Design

- Orthonormal design

$$p = n, \frac{X^T X}{n} = I_{p \times p}$$

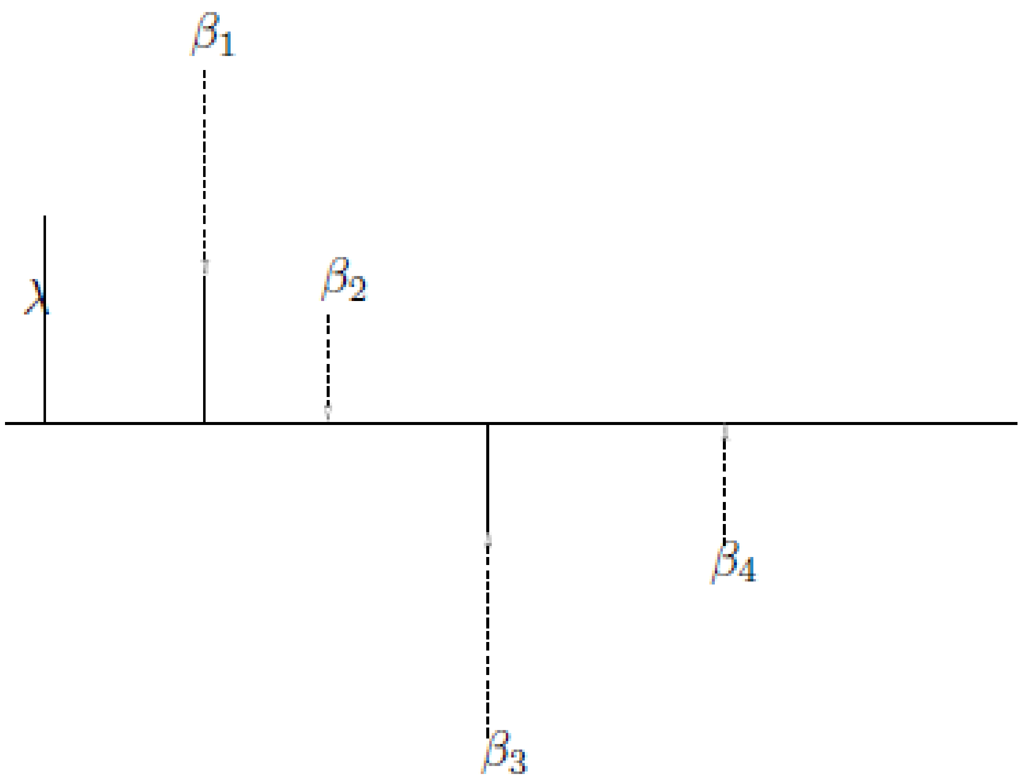
- Then the lasso estimator is the soft-thresholding estimator

$$\hat{\beta}_j(\lambda) = sgn(Z_j)(|Z_j| - \lambda/2)_+$$

$$Z_j = \frac{(X^T Y)_j}{n}$$

- Where $(x)_+=\max(x,0)$.

Soft-Thresholding



LARS (Least Angle Regression and Shrinkage)

- Least angle regression is like a more "democratic" version of forward stepwise regression.
- The least angle regression procedure follows the same general scheme as forward selection does, but doesn't add a predictor fully into the model.
- The coefficient of that predictor is increased only until that predictor is no longer the one most correlated with the residual r. Then some other competing predictor is invited to "join the club".

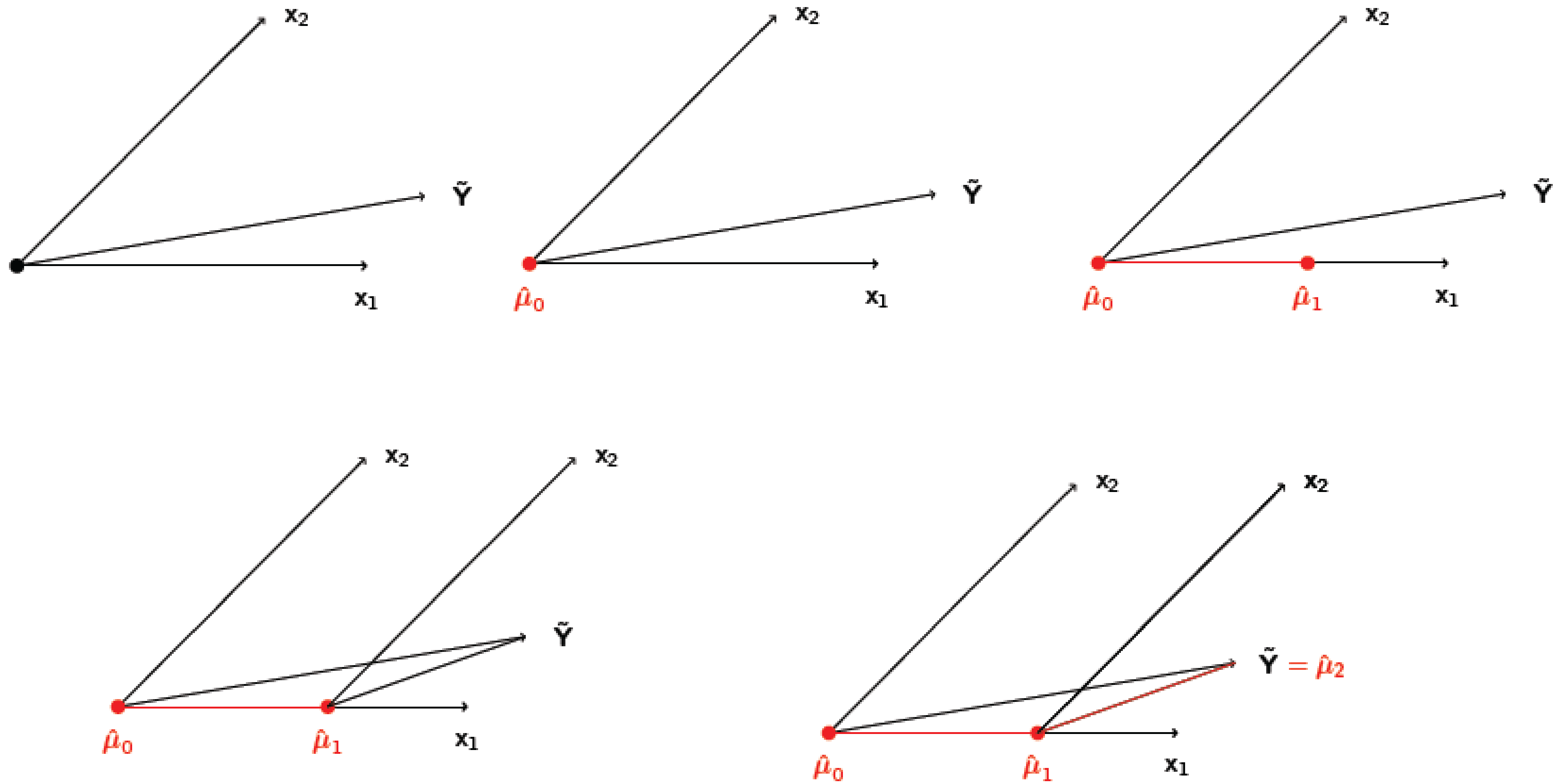
http://statweb.stanford.edu/~tibs/lasso/simple.html

LARS Procedure

- Start with all coefficients b_j equal to zero.
- Find the predictor x_j most correlated with y
- Increase the coefficient b_j in the direction of the sign of its correlation with y. Take residuals (r=y-yhat) along the way. Stop when some other predictor x_k has as much correlation with r as x_j has.
- Increase (b_j, b_k) in their joint least squares direction, until some other predictor x_m has as much correlation with the residual r.
- Continue until: all predictors are in the model

http://statweb.stanford.edu/~tibs/lasso/simple.html

LARS Illustration

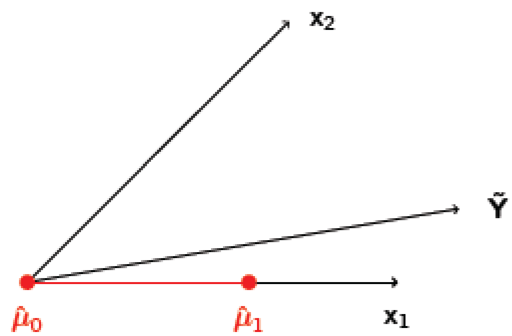


http://courses.cs.washington.edu/courses/cse599c1/13wi/slides/LARS-fusedlasso.pdf

LARS-Lasso Relationship

- Let $\mu(\gamma) = X\beta(\gamma)$
- We show that for active covariate j,

$$\text{sign}(\hat{\beta}_j) = \text{sign}(x_j'(y - \hat{\mu}))$$



LARS-Lasso Relationship

- Let $\mu(\gamma) = X\beta(\gamma)$ with $\beta_j(\gamma) = \hat{\beta}_j + \gamma\hat{d}_j$

- We showed that for active covariate *j*: $\text{sign}(\hat{\beta}_j) = \text{sign}(x_j'(y - \hat{\mu}))$

- $\beta_j(\gamma)$ changes sign at

- 1st sign change occurs at $\tilde{\gamma} = \min_{\gamma_j > 0}\{\gamma_j\}$ for covariate

Pathwise Coordinate Descent for the Lasso

- Coordinate descent: optimize one parameter (coordinate) at a time.
- How? suppose we had only one predictor. Solution is the soft-thresholded estimate

$$\mathrm{sign}(\hat{\beta})(\hat{\beta} - \lambda)_+$$

  where $\hat{\beta}$ is usual least squares estimate.
- Idea: with multiple predictors, cycle through each predictor in turn. We compute residuals and applying univariate soft-thresholding.

Pathwise Coordinate Descent for the Lasso

- Start with large value for λ (very sparse model) and slowly decrease it
- Most coordinates that are zero never become non-zero
- Coordinate descent code for Lasso is just 73 lines of Fortran!

Extension

- Pathwise coordinate descent can be generalized to many other models: logistic/multinomial for classification, graphical lasso for undirected graphs, fused lasso for signals.
- Its speed and simplicity are quite remarkable.
- glmnet R package now available on CRAN

Lasso理论性质总结

- 考虑如下的渐近问题

$$Y_{n;i} = \sum_{j=1}^{p_n} \beta^0_{n;j} x^{(j)}_{n;i} + \epsilon_{n;i}, \quad i = 1, \cdots, n; n = 1, 2, \cdots$$

- 这里允许 $p = p_n \gg n$
- 真实相关变量的稀疏性假设

$$|\beta^0|_1 = o\left(\sqrt{\frac{n}{\log p}}\right)$$

Lasso理论性质总结

- Slow rate of convergence

$$||X(\hat{\beta} - \beta^0)||_2^2 = O_p(||\beta^0||_1 \sqrt{\log(p)/n})$$

- 因此在稀疏性假设下得到了预测的相合性 (Consistency for prediction)

Lasso理论性质总结

- Fast convergence rate

$$||X(\hat{\beta} - \beta^0||_2^2/n = O_p(s_0\phi^{-2}\log(p)/n)$$
$$||\hat{\beta} - \beta_0||_q = O_p(s_0^{1/q}\phi^{-2}\sqrt{log(p)/n}), \quad q \in \{1, 2\}$$

- where s_0 equals the number of non-zero regression coefficients. ϕ^2 denotes a restricted eigenvalue of the design matrix X.

Lasso理论性质总结

- Variable screening property

$$Pr(\hat{S} \supseteq S_0) \to 1 \quad (p \geq n \to +\infty)$$

where

$$\hat{S} = \{j : \hat{\beta}_j \neq 0, j = 1, 2, \cdots, p\}$$
$$S_0 = \{j : \beta_j^0 \neq 0, j = 1, 2, \cdots, p\}$$

- 条件: beta-min condition

$$\min_{j \in S_0^C} |\beta_j^0| \gg \phi^{-2} \sqrt{s_0 \log(p)/n}$$

Lasso理论性质总结

- Consistent Variable selection property

$$Pr[\hat{S} = S_0] \to 1 \quad (p \geq n \to +\infty)$$

- 条件beta-min condition+ Irrepresentable condition

$$||\hat{\Sigma}_{2,1}\hat{\Sigma}_{1,1}^{-1} sign(\beta_1^0, \cdots, \beta_{s_0}^0)||_\infty \leq \theta, 0 < \theta < 1$$

- Irrepresentable condition fails to hold if the design matrix X is too much “ill-posed” and exhibits a too strong degree of dependence within “smaller” sub-matrices of X.
- 其中分块矩阵(s_0 相关变量| p-s_0不相关变量)

$$\hat{\Sigma} = n^{-1}X^TX = \begin{pmatrix} \hat{\Sigma}_{1,1} & \hat{\Sigma}_{1,2} \\ \hat{\Sigma}_{2,1} & \hat{\Sigma}_{2,2} \end{pmatrix}$$

Adaptive Lasso

- Zou Hui (2006) proposed a two stage procedure

$$\hat{\beta}_{adapt}(\lambda) = \underset{\beta}{\text{Argmin}} \left(\frac{1}{2n} ||Y - X\beta||_2^2) + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{|\hat{\beta}_{init,j}|} \right)$$

where $\hat{\beta}_{init}$ is an initial estimator

- It has the following obvious property

$$\hat{\beta}_{init,j} = 0 \quad \Rightarrow \quad \hat{\beta}_{adapt,j} = 0$$

Adaptive Lasso

- 通常情况下Lasso存在所谓的Overestimation现象，即以很大的概率选出的变量大大超过真实的相关变量

- 而在一个比较弱的条件下，Adaptive Lasso选出的变量集合是相合的

Elastic Net: Motivation

- For strong correlated covariates, Lasso may select one but typically not both of them.
- In term of sparsity, this is what we would like to do.
- In term of interpretation, we may want to have two even strongly correlated variables among the selected variables.

Elastic Net

- Zou and Hastie (2005) proposed a double penalization

$$\hat{\beta}_{naiveEN}(\lambda_1, \lambda_2) = \underset{\beta}{\mathrm{Argmin}} \left(\frac{1}{2n} ||Y - X\beta||_2^2 + \lambda_1 |\beta|_1 + \lambda_2 ||\beta||_2^2 \right)$$

$$\hat{\beta}_{EN}(\lambda_1, \lambda_2) = (1 + \lambda_2) \hat{\beta}_{naiveEN}$$

Group Lasso

- In some application, parameter vector is structured into groups

$$G_1, \cdots, G_q; \quad G_i \cap G_j = \emptyset (i \neq j)$$
$$\cup_{j=1}^q G_j = \{1, 2, \cdots, p\}$$
$$\beta = (\beta_{G_1}, \cdots, \beta_{G_q}), \beta_{G_j} = \{\beta_r : r \in G_j\}$$

- Group lasso penalty

$$\lambda_{j=1}^q m_j ||\beta_{G_j}||_2, \quad m_j = \sqrt{T_j}, T_j = |G_j|.$$

Group Lasso

- Group lasso estimator is a linear or generalized linear model is then defined as

$$\begin{cases} \hat{\beta}(\lambda) = \underset{\beta}{\operatorname{Argmin}}\, Q_\lambda(\beta) \\ Q_\lambda(\beta) = \dfrac{1}{n}\displaystyle\sum_{i=1}^{n} \rho_\beta(X_i, Y_i) + \lambda \sum_{j=q}^{q} m_j ||\beta_{G_j}||_2 \end{cases}$$