High Dimensional Data and Statistical Learning

Lecture 5: Soft-Thresholding and Lasso

Weixing Song

Department of Statistics
Kansas State University

Outline

Outline

High Dimensional Linear Regression

We consider the following linear regression model

$$y_i = x_{i1}\beta_1 + \cdots + x_{ip}\beta_p + \varepsilon_i, \quad 1 \le i \le n.$$

Some notations:

Response: $\mathbf{y} = (y_1, \ldots, y_n)^T$;

Predictors: $\mathbf{x}_j = (x_{1j}, \ldots, x_{nj})^T$, $j = 1, 2, \ldots, p$;

Design Matrix: $\mathbf{X}_{n\times p} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$;

Regression Coefficients: $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$;

True Regression Coefficients: $\boldsymbol{\beta}^o = (\beta_1^o, \ldots, \beta_p^o)$;

Oracle Set: $\mathcal{O} = \{j : \beta_j^o \neq 0\}$;

Underlying Model Dimensions: $d^0 = \|\mathcal{O}\| = \#\{j : \beta_j^o \neq 0\}$.

Centering and Standardization

WLOG, we assume that the response and predictors are centered and the predictors are standardized as follows:

$$\sum_{i=1}^{n} y_i = 0, \quad \sum_{i=1}^{n} x_{ij} = 0, \quad \sum_{i=1}^{n} x_{ij}^2 = n,$$

for $j = 1, 2, \ldots, p$.

After the centering and standardization, there is no intercept in the model.

Each predictor is standardized to have the same magnitude in L_2. So the corresponding regression coefficients are comparable.

After model fitting, the results can be readily transformed back to the original scale.

We consider the penalized least squares (PLS) method:

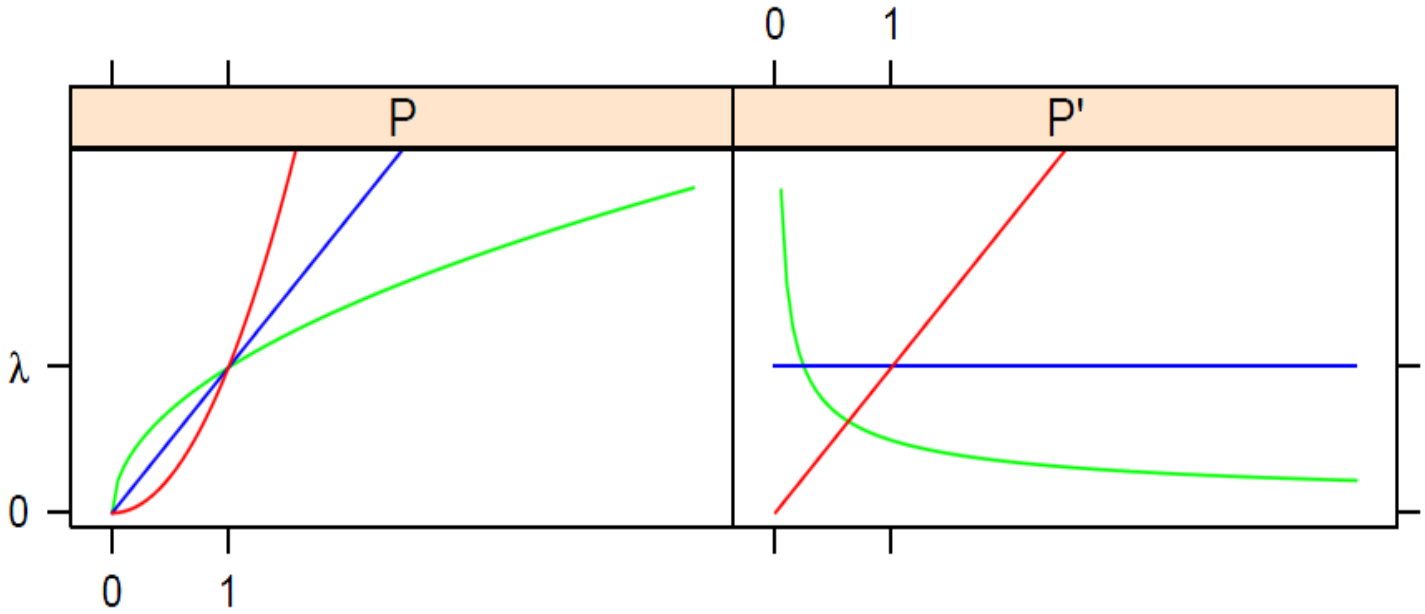
$$\min_{\boldsymbol{\beta}\in\mathbb{R}^p}\left\{\frac{1}{2n}\|\mathbf{y}-\mathbf{X}\boldsymbol{\beta}\|^2+\sum_{j=1}^{p}p_\lambda(|\beta_j|)\right\},$$

where $\|\cdot\|$ denotes the L_2-norm, $p_\lambda(\cdot)$ is a penalty function indexed by the regularized parameter $\lambda\geq 0$.

Some commonly used penalty functions:

- L_0-penalty (subset selection) and L_2-penalty (ridge regression);
- Bridge or L_γ penalty, $\gamma>0$. (Frank and Friedman, 1993);
- L_1 penalty or Lasso (Tibshirani, 1996);
- SCAD penalty (Fan and Li, 2001);
- MCP penalty (Zhang, 2010);
- Group penalties, bi-level penalties, ...

Plots of Bridge penalties:



$$p_\lambda(\theta) = \lambda|\theta|^\gamma\,,\ \gamma > 0.$$

bridge (γ=0.5)
lasso (γ=1)
ridge (γ=2)

Outline

Lasso

Lasso stands for "least absolute shrinkage and selection operator". There are two equivalent definitions.

- Minimizing the residual sum of squares subject to the sum of the absolute value of the coefficients being less than a constant:

$$\hat{\boldsymbol{\beta}} = \text{argmin} \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq t.$$

- Minimizing the penalized sum of squares:

$$\hat{\boldsymbol{\beta}} = \text{argmin} \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

Because of the nature of L_1 penalty or constraint, Lasso is able to estimate some coefficients as exactly 0 and hence performs variable selection.

The Lasso enjoys some of the favorable properties of both subset selection and ridge regression. It produces interpretable models and exhibits the stability of ridge regression.

The motivation for the Lasso came from an interesting proposal of Breiman (1993). Breiman's *non-negative Garotte* minimizes

$$\frac{1}{2n}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} c_j \hat{\beta}_j^{LS} x_{ij}) \quad \text{subject to} \quad c_j \geq 0, \sum_{j=1}^{p} c_j \leq t,$$

or

$$\frac{1}{2n}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} c_j \hat{\beta}_j^{LS} x_{ij}) + \lambda \sum_{j=1}^{p} c_j, \quad \text{subject to } c_j \geq 0.$$

The Garotte starts with the OLS estimates and shrinks them by non-negative factors whose sum is constrained.

The Garotte estimate depends on both the sign and the magnitude of OLS. In contrast, the Lasso avoids the explicit use of the OLS estimates.

Lasso is also closely related to the wavelet soft-thresholding method by Donoho and Johnstone (1994), forward statewise regression, and boosting methods.

Solution Path:

For each given λ, we solve the PLS problem. Therefore, for $\lambda \in [\lambda_{\min}, \lambda_{\max}]$, we have a solution path

$$\{\hat{\beta}_n(\lambda) : \lambda \in [\lambda_{\min}, \lambda_{\max}]\}.$$

To examine the solution path, we can plot each component of $\hat{\beta}_n(\lambda)$ versus λ.

In practice, we usually need to determine a value of λ, say, λ_*, and use $\hat{\beta}_n(\lambda_*)$ as the final estimator. This model selection step is usually done using some information criterion or cross validation techniques.

Thus it is important to have fast algorithms for computing the whole solution path or a grid of λ values.

There are multiple packages in R for computing the Lasso path: `ncvreg`, `glmnet` and `lars`,....

Note that the solution path can also be indexed by the constraint value t.

Lasso: Examples

Example 1: Consider the linear regression model

$$y_i = \sum_{j=1}^{p} x_{ij}\beta_j + \varepsilon_i, \quad 1 \le i \le n,$$

where $\varepsilon \sim N(0, 1.5^2)$, $p = 100$.

Generate z_{ij}'s and w independently from $N(0, 1)$, let

$$x_{ij} = z_{ij} + w, \quad 1 \le j \le 4, \quad x_{i5} = z_{i5} + 2w, \quad x_{i6} = z_{i6} + w,$$

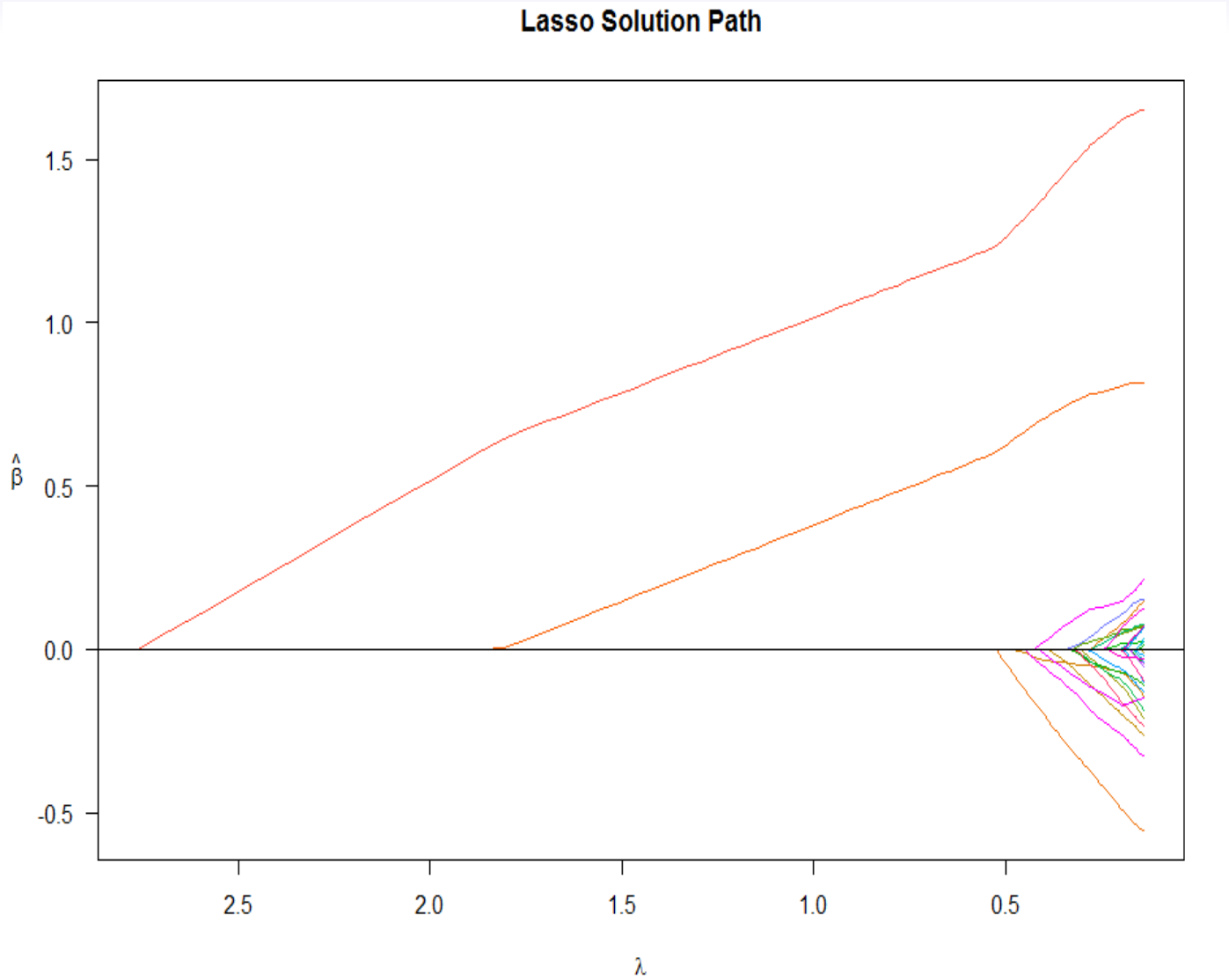
and $x_{ij} = z_{ij}$ for $j \ge 7$.

Let $(\beta_1, \beta_2, \beta_3) = (2, 1, -1)$ and $\beta_j = 0$ for $j \ge 4$.

To implement the lasso estimation for the above example, we use the R-package `ncvreg`. A sample R-code is

```
library(ncvreg);

n=100;
p=100;
sigma=1.5;
tau=1;

Z=matrix(rnorm(n*p,0,tau),nrow=n);
w=rnorm(n,0,tau);
X=matrix(0,nrow=n,ncol=p);

for(j in 1:6)
  {
   X[,j]=Z[,j]+ifelse(j==5,2,1)*w;
  }
X[,7:100]=Z[,7:100];

beta=as.vector(c(2,1,-1,rep(0,97)));
e=rnorm(n,0,sigma);
y=X%*%beta+e;

fit=ncvreg(X,y,penalty=c("lasso"))
plot(fit,main="Lasso Solution Path")
```

Remark: The `ncvreg` pacakge is built upon the work by Breheny and Huang (2011).

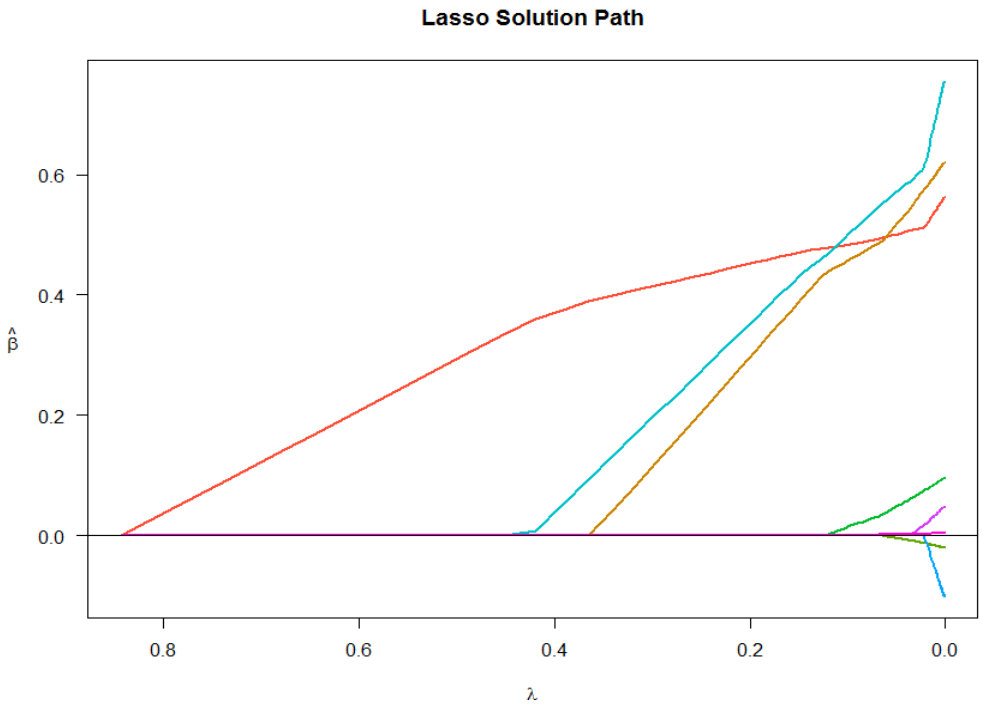

Lasso Solution Path
1.5
1.0
0.5
0.0
-0.5
β̂
2.5
2.0
1.5
1.0
0.5
λ

Example 2 (Prostate Data): The data is from a study by by Stamey et al. (1989) to examine the association between prostate specific antigen (PSA) and several clinical measures that are potentially associated with PSA in men who were about to receive a radical prostatectomy. The variables are as follows:

- lcavol: Log cancer volume
- lweight: Log prostate weight
- age: The man's age
- lbph: Log of the amount of benign hyperplasia
- svi: Seminal vesicle invasion; 1=Yes, 0=No
- lcp: Log of capsular penetration
- gleason: Gleason score
- pgg45: Percent of Gleason scores 4 or 5
- lpsa: Log PSA

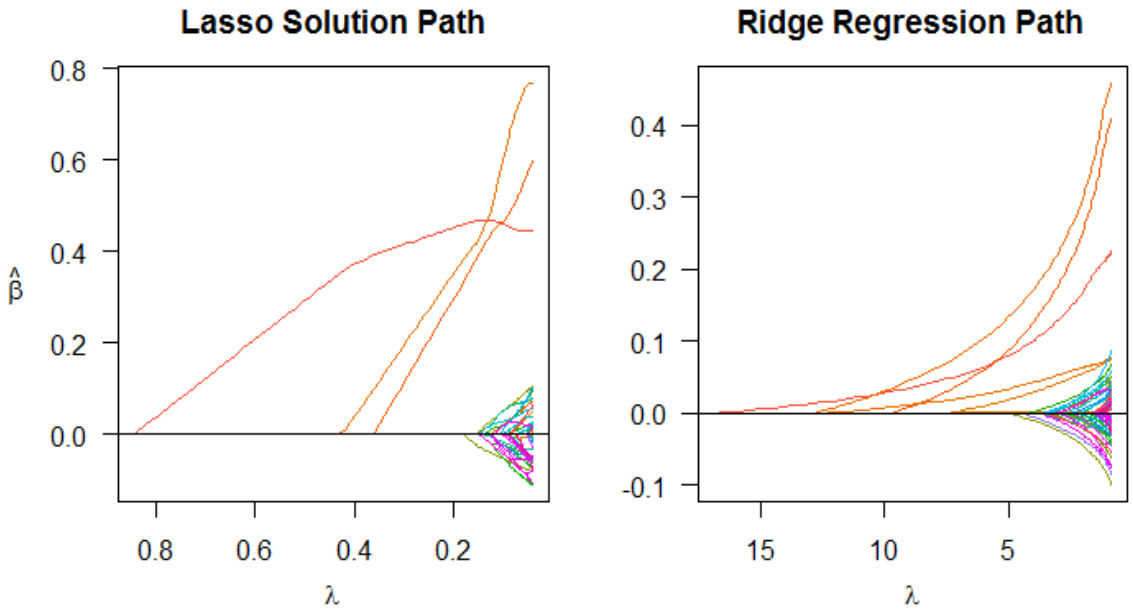
To implement the lasso estimation for the above example, we use the R-package **ncvreg**. A sample R-code is

```
library(glmnet)
data(prostate);
X=prostate[,1:8];
y=prostate$lpsa;
fit=ncvreg(X,y,penalty=c("lasso"));
plot(fit,main="Lasso Solution Path");
```



To compare Lasso and Ridge, we also add some artificial noise variables to the model.

```
data(prostate);
X=prostate[,1:8];
y=prostate$lpsa;
X1=matrix(rnorm(length(y)*100),nrow=length(y))
X=cbind(X,X1);
par(mfrow=c(1,2));
fitL=ncvreg(X,y,penalty=c("lasso"));
plot(fitL,main="Lasso Solution Path");
fitR=ncvreg(X,y,alpha=0.05);
plot(fitR,main="Ridge Regression Path")
```



Outline

Orthogonal Design in PLS

Insight about the nature of the penalization methods can be gleaned from the orthogonal design case.

When the design matrix multiplied by $n^{-1/2}$ is orthonormal, i.e., $\mathbf{X}^T\mathbf{X} = nI_{p\times p}$, the penalized least squares problem reduces to the minimization of

$$\frac{1}{2n}\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{\text{LSE}}\|^2 + \frac{1}{2}\|\hat{\boldsymbol{\beta}}_{\text{LSE}} - \boldsymbol{\beta}\|^2 + \sum_{j=1}^{p} p_\lambda(|\beta_j|),$$

where $\hat{\boldsymbol{\beta}}_{\text{LSE}} = n^{-1}\mathbf{X}^T\mathbf{y}$ is the OLS estimate.

Now the optimization problem is separable in β_j's. It suffices to consider the univariate PLS-problem

$$\hat{\theta}(z) = \text{argmin}_{\theta\in\mathbb{R}} \left\{ \frac{1}{2}(z-\theta)^2 + p_\lambda(|\theta|) \right\}. \tag{1}$$

Characterization of PLS Estimators

Antoniadis and Fan (2001)

Let $p_\lambda(\cdot)$ be a nonnegative, nondecreasing, and differentiable function in $(0, \infty)$. Further, assume that the function $-\theta - p'_\lambda(\theta)$ is strictly unimodal on $(0, \infty)$. Then we have the following results.

(a). The solution to the minimization problem (1) exists and is unique. It is antisymmetric: $\hat{\theta}(-z) = -\hat{\theta}(z)$.

(b). The solution satisfies

$$\hat{\theta}(z) = \begin{cases} 0 & \text{if } |z| \le p_0, \\ z - \operatorname{sgn}(z) p'_\lambda(\hat{\theta}(z)) & \text{if } |z| > p_0, \end{cases}$$

where $p_0 = \min_{\theta \ge 0}\{\theta + p'_\lambda(\theta)\}$. Moreover, $|\hat{\theta}(z)| \le |z|$.

(c). If $p'_\lambda(\cdot)$ is nonincreasing, then for $|z| > p_0$, we have

$$|z| - p_0 \le |\hat{\theta}(z)| \le |z| - p'_\lambda(|z|).$$

(d). When $p'_\lambda(\theta)$ is continuous on $(0, \infty)$, the solution $\hat{\theta}(z)$ is continuous if and only if the minimum of $|\theta| + p'_\lambda(|\theta|)$ is attained at point zero.

(e). If $p'_\lambda(|z|) \to 0$ as $|z| \to \infty$, then

$$\hat{\theta}(z) = z - p'_\lambda(|z|) + o(p'_\lambda(|z|)).$$

Proof: Denote $l(\theta)$ as the function in (1).

(a)-(b). Note that $l(\theta)$ tends to infinity as $|\theta| \to \infty$. Thus, minimizers do exist.

When $z = 0$, it is clear that $\hat{\theta}(z) = 0$ is the unique minimizer.

WLOG, assume that $z > 0$. Then for all $\theta > 0$, $l(-\theta) > l(\theta)$. Hence $\hat{\theta}(z) \geq 0$. Note that for $\theta > 0$,

$$l'(\theta) = \theta - z + p'_\lambda(\theta).$$

When $z < p_0$, the function l is strictly increasing on $(0, \infty)$ because the derivative function is positive. Hence $\hat{\theta}(z) = 0$.

Now assume that $z > p_0$. When the function $l'(\theta)$ is strictly increasing, there is at most one zero-crossing, and hence the solution is unique.

If $l'(\theta)$ has a valley on $(0, \infty)$ (Why?), there are two possible zero-crossings for the function l' on $(0, \infty)$. The larger one is the minimizer because the derivative function at that point is increasing. Hence, the solution is unique and satisfies

$$\hat{\theta}(z) = z - p'_\lambda(\hat{\theta}(z)) \leq z. \tag{2}$$

Proof(continued):

(c). From the above, it is easy to see that $\hat{\theta}(z) \leq z - p'_\lambda(z)$ when $p'_\lambda(\cdot)$ is nonincreasing.

Let θ_0 be the minimizer of $\theta + p'_\lambda(\theta)$ over $[0, \infty)$. Then, from the preceding argument, $\hat{\theta}(z) > \theta_0$ for $z > p_0$. If $p_\lambda(\cdot)$ is nonincreasing, then

$$p'_\lambda(\hat{\theta}(z)) \leq p'_\lambda(\theta_0) \leq \theta_0 + p'_\lambda(\theta_0) = p_0.$$

This and (2) prove result (c).

(d). It is clear that continuity of the solution $\hat{\theta}(z)$ at the point $z = p_0$ if and only if the minimum of the function $|\theta| + p'_\lambda(|\theta|)$ is attained at 0. The continuity at other locations follow directly from the monotonicity and continuity of the function $\theta + p'_\lambda(\theta)$ in the interval $(0, \infty)$.

(e). This follows directly from (2). ☹ If $p_\lambda(z)$ is twice differentiable for large z values, and $p''_\lambda(|z|) \to 0$ as $|z| \to \infty$, then from (2)

$$\hat{\theta}(z) = z - p'_\lambda(\hat{\theta}(z)) = z - p'_\lambda(z) - (\hat{\theta}(z) - z)p''_\lambda(\tilde{z}),$$

where $\tilde{z}$ is between z and $\hat{\theta}(z)$. Since $z \to \infty$ implies $\hat{\theta}(z) \to \infty$, so the above equation $p''_\lambda(\tilde{z}) \to 0$ as $z \to \infty$, this in turn leads to $|\hat{\theta}(z) - z| \leq c|p'_\lambda(z)|$ for some constant. Therefore,

$$|\hat{\theta}(z) - z + p'_\lambda(z)| = |\hat{\theta}(z) - z| \cdot |p''_\lambda(\tilde{z})| \leq c|p'_\lambda(z)| \cdot |p''_\lambda(\tilde{z})| = o(p'_\lambda(z)).$$

The above theorem implies that the PLS estimator $\hat{\theta}(z)$ possesses the properties:

Sparsity if $\min_{t\geq 0}\{t + p'_\lambda(t)\} > 0$;

Approximate unbiasedness if $p'_\lambda(t) \to 0$ for large t;

Continuity if and only if $\operatorname{argmin}_{t\geq 0}\{t + p'_\lambda(t)\} = 0$.

In general for penalty functions, the singularity at the origin (i.e. $p'_\lambda(0+) > 0$) is needed for generating sparsity in variable selection and the concavity is needed to reduce the bias.

These conditions are applicable for general PLS problems and more.

Homework: Prove Antoniadis and Fan (2001)'s theorem for $z < 0$.

Lasso: Orthogonal Design

Under orthogonal design, i.e., $\mathbf{X}^T\mathbf{X} = nI$, Lasso estimation can be greatly simplified as discussed above. The problem becomes solving

$$\hat{\beta}_j = \text{argmin}_{\beta_j} \left\{ \frac{1}{2}(\hat{\beta}_j^{\text{LSE}} - \beta_j)^2 + \lambda|\beta_j| \right\}.$$

The Lasso estimator is given by

$$\hat{\beta}_j = \mathcal{S}(\hat{\beta}_j^{\text{LSE}}; \lambda),$$

where $\mathcal{S}(\cdot; \lambda)$ is the soft-thresholding operator

$$\mathcal{S}(z; \lambda) = \text{sgn}(z)(|z| - \lambda)_+ = \begin{cases} z - \lambda, & \text{if } z > \lambda, \\ 0 & \text{if } |z| \leq \lambda, \\ z + \lambda, & \text{if } z < -\lambda, \end{cases}$$

Homework: Verify the expression of $\hat{\beta}_j$ based on Antoniadis and Fan (2001)'s theorem.

Hard Thresholding

The solution to

$$\text{argmin}_\theta \left\{ \frac{1}{2}(z-\theta)^2 + \lambda^2 - (|\theta| - \lambda)^2 I(|\theta| < \lambda) \right\}$$

is given by $\hat{\theta} = H(z; \lambda)$, where

$$\mathcal{H}(z; \lambda) = zI(|z| > \lambda) = \begin{cases} z, & \text{if } |z| > \lambda, \\ 0, & \text{if } |z| \leq \lambda, \end{cases}$$

is called the hard thresholding operator.

The proof of the above minimizer is based upon Antoniadis and Fan (2001)'s theorem.

This corresponding to the best subset selection. Note that the best subset selection of size k reduces to choosing the k largest coefficients in absolute value and setting the rest to 0.

Ridge/Non-negative Garotte Estimators

Under orthogonal design, the ridge regression estimator is given by

$$\frac{1}{1+\lambda}\hat{\beta}_j^{\mathrm{LSE}}.$$

Under orthogonal design, the non-negative garotte solution is given by

$$\hat{\beta}_j^{\mathrm{NG}} = \left(1 - \frac{\lambda}{(\hat{\beta}_j^{\mathrm{LSE}})^2}\right)_+ \hat{\beta}_j^{\mathrm{LSE}}.$$

For: Note that the target function in the non-negative garotte procedure can be written as

$$\frac{1}{2n}\|\mathbf{y} - \mathbf{X}\mathrm{diag}(\hat{\boldsymbol{\beta}})\mathbf{c}\|_2^2 + \lambda\|\mathbf{c}\|.$$

Let $c_j^* = c_j\hat{\beta}_j^{\mathrm{LSE}}$. Then the target function becomes

$$\frac{1}{2n}\|\mathbf{y} - \mathbf{X}\mathbf{c}^*\|_2^2 + \lambda\sum_{j=1}^{p}|(\hat{\beta}_j^{\mathrm{LSE}})^{-1}c_j^*|$$

which is equivalent to $2^{-1}\|\hat{\boldsymbol{\beta}}^{\mathrm{LSE}} - \mathbf{c}^*\|_2^2 + \lambda\sum_{j=1}^{p}|(\hat{\beta}_j^{\mathrm{LSE}})^{-1}c_j^*|$

Therefore, minimizing the above target function is amount to find the minimizer of

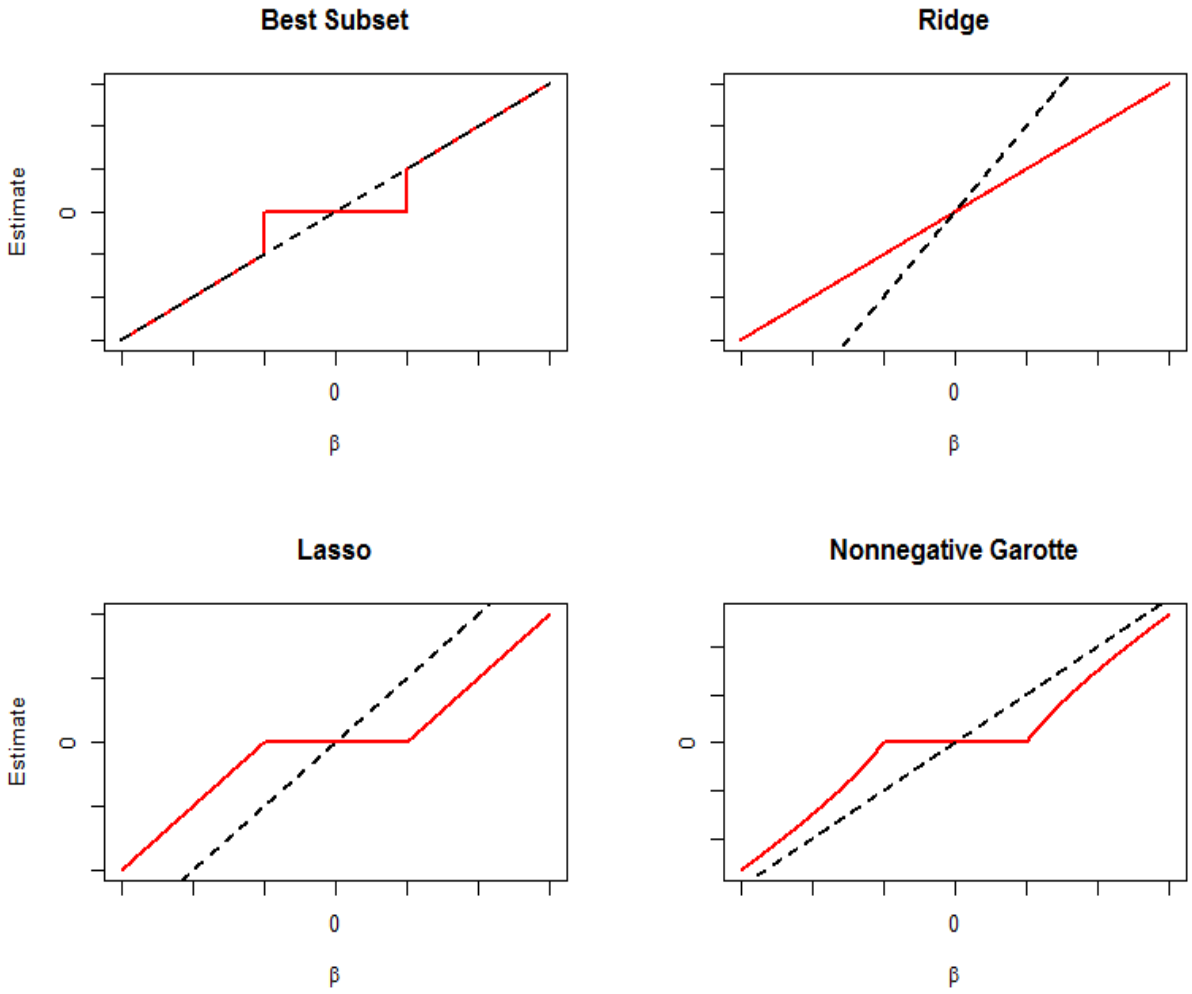
$$\frac{1}{2}(\hat{\beta}_j^{\mathrm{LSE}} - c_j^*)^2 + \lambda|(\hat{\beta}_j^{\mathrm{LSE}})^{-1}| \cdot |c_j^*|.$$

Hence the solution is

$$c_j^* = \mathcal{S}(\hat{\beta}_j^{\mathrm{LSE}}; \lambda|\hat{\beta}_j^{\mathrm{LSE}}|^{-1}).$$

Transformation back to c_j gives $\hat{\beta}_j^{\mathrm{NG}}$.

Comparison of Four Estimators



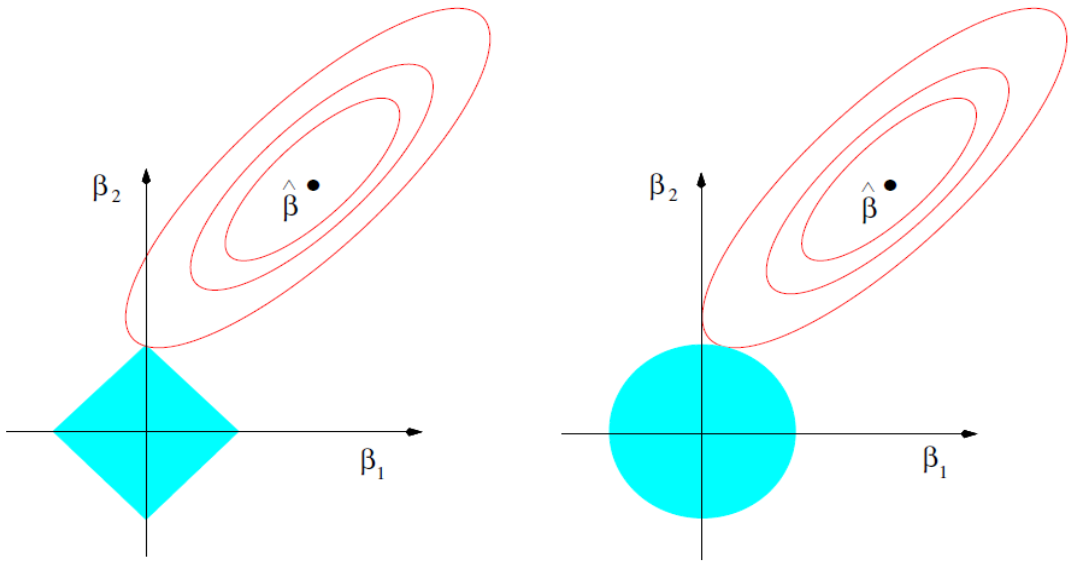
The Geometry of Lasso

The RSS term $\|\mathbf{y} - \mathbf{X}\beta\|$ equals the following quadratic function plus a constant

$$(\beta - \hat{\beta}_{\text{LSE}})^T \mathbf{X}^T \mathbf{X} (\beta - \hat{\beta}_{\text{LSE}}).$$

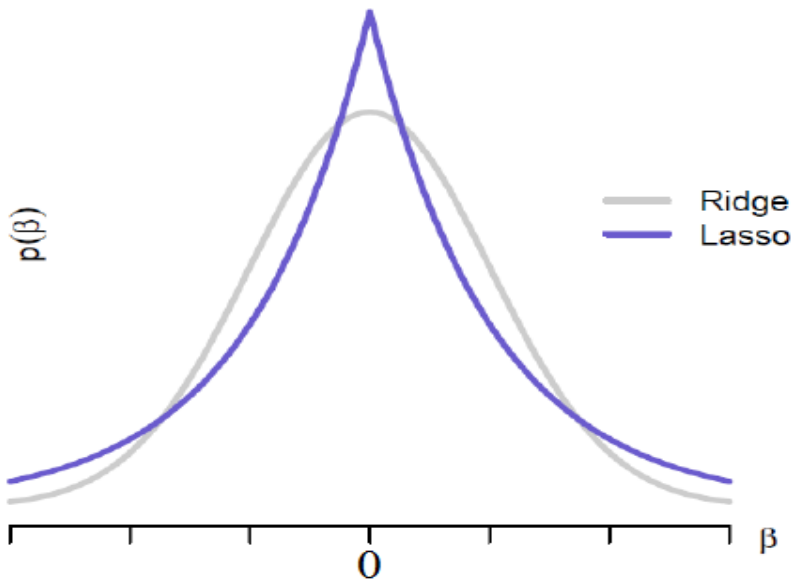
The contour of the above function is elliptical. These ellipsoids are centered at the LSE.

For Lasso, the L_1-constraint region is a diamond; for ridge, the L_2-constraint region is a disk.



Another way of comparing Lasso and ridge is from a Bayesian perspective.

In ridge, the prior of β is normal distribution; in Lasso, the prior of β is Laplace distribution.



Outline

How to obtain Lasso solution for the general case?

Lasso is a convex programming problem. Several algorithms have been proposed for computing L_1-penalized estimates.

- Coordinate descent. See Fu (1998), Friedman et al. (2007).
- Convex optimization algorithms. See Osborne et al. (2000a, b).
- Least angle regression (LARs). See Efron et al. (2004).
- Others .....

Here we first focus on the coordinate descent algorithm, which is simple, stable, and efficient for a variety of high-dimensional models.

Coordinate descent algorithms optimize a target function with respect to a single parameter at a time, iteratively cycling through all parameters until convergence is reached.

They are ideal for problems that have a simple closed form solution in a single dimension but lack one in higher dimensions.

Coordinate Descent Algorithm: Derivation

Given current values for the regression coefficients $\beta_k = \tilde{\beta}_k$, $k \neq j$. Define

$$L_j(\beta_j; \lambda) = \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{k \neq j} x_{ik} \tilde{\beta}_k - x_{ij} \beta_j \right)^2 + \lambda |\beta_j|.$$

Denote

$$\tilde{y}_{ij} = \sum_{k \neq j} x_{ik} \tilde{\beta}_k, \quad \tilde{r}_{ij} = y_i - \tilde{y}_{ij}, \quad \tilde{z}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij} \tilde{r}_{ij}.$$

$\tilde{r}_{ij}$ are called the partial residuals with respect to the j-th covariate.

Some algebra shows that

$$L_j(\beta_j; \lambda) = \frac{1}{2} (\beta_j - \tilde{z}_j)^2 + \lambda |\beta_j| + \frac{1}{2n} \sum_{i=1}^{n} \tilde{r}_{ij}^2 + \frac{1}{2} \tilde{z}_j^2.$$

Let $\tilde{\beta}_j$ denote the minimizer of $L_j(\beta_j; \lambda)$. We have

$$\tilde{\beta}_j = \mathcal{S}(\tilde{Z}_j; \lambda) = \operatorname{sgn}(\tilde{z}_j)(|\tilde{z}_j| - \lambda)_+,$$

where $S(\cdot; \lambda)$ is the soft-thresholding operator.

Coordinate Descent Algorithm

For any fixed λ,

1. Start with an initial value for $\boldsymbol{\beta} = \boldsymbol{\beta}^{(0)}$;
2. In the $s+1$-th iteration,
   (1). Let $j = 1$;
   (2). Calculate
   $$\tilde{z}_j = n^{-1} \sum_{i=1}^{n} x_{ij} r_i + \tilde{\beta}_j^{(s)},$$
   where $r_i = y_i - \tilde{y}_i = y_i - \sum_{j=1}^{p} x_{ij} \tilde{\beta}_j^{(s)}$ is the current residual.
   (3). Update $\tilde{\beta}_j^{(s+1)}$ using $\mathcal{S}(\tilde{z}_j; \lambda)$. If $j = p$, then exit step 2.
   (4). Update r_i using $r_i - (\tilde{\beta}_j^{(s+1)} - \tilde{\beta}_j^{(s)}) x_{ij}$ for all i.
   (5). Let $j \Leftarrow j + 1$, repeat (2)-(4).
3. Repeat step 2 for $s+1$ until convergence.

NOTE: The above algorithm is designed for the cases in which the predictors are standardized to have L_2-norm n.

The coordinate descent algorithm can be used repeatedly compute $\hat{\boldsymbol{\beta}}(\lambda)$ on a grid of λ values. Let $\lambda_{\max}$ be the smallest value for which all coefficients are 0, and $\lambda_{\min}$ be the minimum of λ.

We can use $\lambda_{\max} = \max_j |(\mathbf{x}_j^T \mathbf{x}_j)^{-1} \mathbf{x}_j^T \mathbf{y}|$ (Consider the orthogonal design case). If the design matrix is full rank, $\lambda_{\min}$ can be 0; otherwise, we use $\lambda_{\min} = \varepsilon \lambda_{\max}$ for some small ε, e.g., $\varepsilon = 10^{-4}$.

Let $\lambda_0 > \lambda_1 > \cdots > \lambda_k$ be a grid of decreasing λ-values, where $\lambda_0 = \lambda_{\max}$, and $\lambda_k = \lambda_{\min}$. Start at λ_0 for which $\hat{\boldsymbol{\beta}}$ has the solution 0 or close to 0, and proceed along the grid using the value of $\hat{\boldsymbol{\beta}}$ at the previous point of λ in the grid as the initial values for the current point. This is called warm start.

Homework:

(a). Verify the updating steps in the coordinate descent algorithm for Lasso.

(b). Write an R-program to implement the coordinate descent algorithm for computing the Lasso solution path.

Outline

Penalty Parameter Selection: CV

- Divide the data into V roughly equal part (5 to 10);
- For each $v = 1, 2, \ldots, V$, fit the model with parameter λ to the other $V - 1$ parts. Denote the resulting estimates by $\hat{\boldsymbol{\beta}}^{(-v)}$;
- Compute the prediction error (PE) in predicting the v-th part:

$$PE_v(\lambda) = \sum_{i \in v\text{-th part}} (y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}}^{(-v)}(\lambda))^2;$$

- Compute the overall cross-validation error

$$CV(\lambda) = \frac{1}{V} \sum_{v=1}^{V} PE_v(\lambda);$$

- Carry out the above steps for many values of λ and choose the value of λ that minimizes $CV(\lambda)$.

Penalty Parameter Selection: AIC and GCV

An **AIC** type criterion for choosing λ is

$$AIC(\lambda) = \log\left\{\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)\|^2/n\right\} + 2df(\lambda)/n,$$

where for the Lasso estimator, $df(\lambda) = \#$ of nonzero coefficients in the model fitted with λ.

A **Generalized Cross Validation (GCV)** criterion is defined as

$$GCV(\lambda) = \frac{n\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)\|^2}{(n - df(\lambda))^2}.$$

It can be seen that these two criteria are close to each other when $df(\lambda)$ is relatively small compared to n.

Penalty Parameter Selection: BIC

The GCV and AIC are reasonable criteria for tuning. However, they tend to select more variables than the true model contains.

Another criterion that is more aggressive in seeking a sparse model is the **Bayesian information criterion (BIC)**:

$$BIC(\lambda) = \log\left\{\|\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}(\lambda)\|^2/n\right\} + 2\log(n)df(\lambda)/n.$$

The tuning parameter λ is selected as the minimizer of $AIC(\lambda)$, $GIC(\lambda)$ or $BIC(\lambda)$.

Outline

Automatic model-building algorithms are notoriously familiar in the linear model literature: Forward Selection, Backward Elimination, Best Subset Selection and many others are used to produce good linear models for predicting a response $\mathbf{y}$ on a basis of some measured covariates $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p$.

The Lasso and Forward Stagewise regression will be discussed in this section.

They are both motivated by a unified approach called Least Angle Regression (LARs). LARs provides an unified explanation, fast implementation, and fast way to choose tuning parameter for Lasso and Forward Stagewise regression.

Forward Stepwise Selection

Given a collection of possible predictors, we select the one having the largest absolute correlation with the response y, say x_{j_1}, and perform linear regression of y on x_{j_1}. This leaves a residual vector orthogonal to x_{j_1}, now considered to be the response.

Regressing other predictors to x_{j_1} leads to $p-1$ residuals, now considered as new predictors.

Repeat the selection process by selecting the one from new predictors having the largest absolute correlation with the new response.

After k steps this results in a set of $x_{j_1}, \ldots, x_{j_k}$ that are then used in the usual way to construct a k-parameter linear model.

Forward Selection is an aggressive fitting technique that can be overly greedy, perhaps eliminating at the second step useful predictors that happen to be correlated with x_{j_1}.

Forward Stagewise Selection

Forward Stagewise is a much more cautious version of Forward Selection, which may take many tiny steps as it moves toward a final model.

We assume that the covariates have been standardized to have mean 0 and unit length and that the response has mean 0.

Forward Stagewise Selection

(1). It begins with $\hat{\mu} = \mathbf{X}\hat{\boldsymbol{\beta}} = 0$ and builds up the regression function in successive small steps.

(2). If $\hat{\mu}$ is the current Stagewise estimate, let $c(\hat{\mu})$ be the vector of current correlations $\hat{c} = c(\hat{\mu}) = \mathbf{X}^T(\mathbf{y} - \hat{\mu})$, so that $\hat{c}_j$ is proportional to the correlation between covariate x_j and the current residual vector.

(3). The next step of is taken in the direction of the greatest current correlation,

$$\hat{j} = \text{argmax}|\hat{c}_j|$$

and update $\hat{\mu}$ by

$$\hat{\mu} \Leftarrow \hat{\mu} + \varepsilon \cdot \text{sgn}(\hat{c}_{\hat{j}})x_{\hat{j}},$$

with ε a small constant.

Small is important here: the big choice $\varepsilon = |\hat{c}_{\hat{j}}|$ leads to the classic Forward Selection technique, which can be overly greedy, impulsively eliminating covarites which are correlated to $x_{\hat{j}}$.

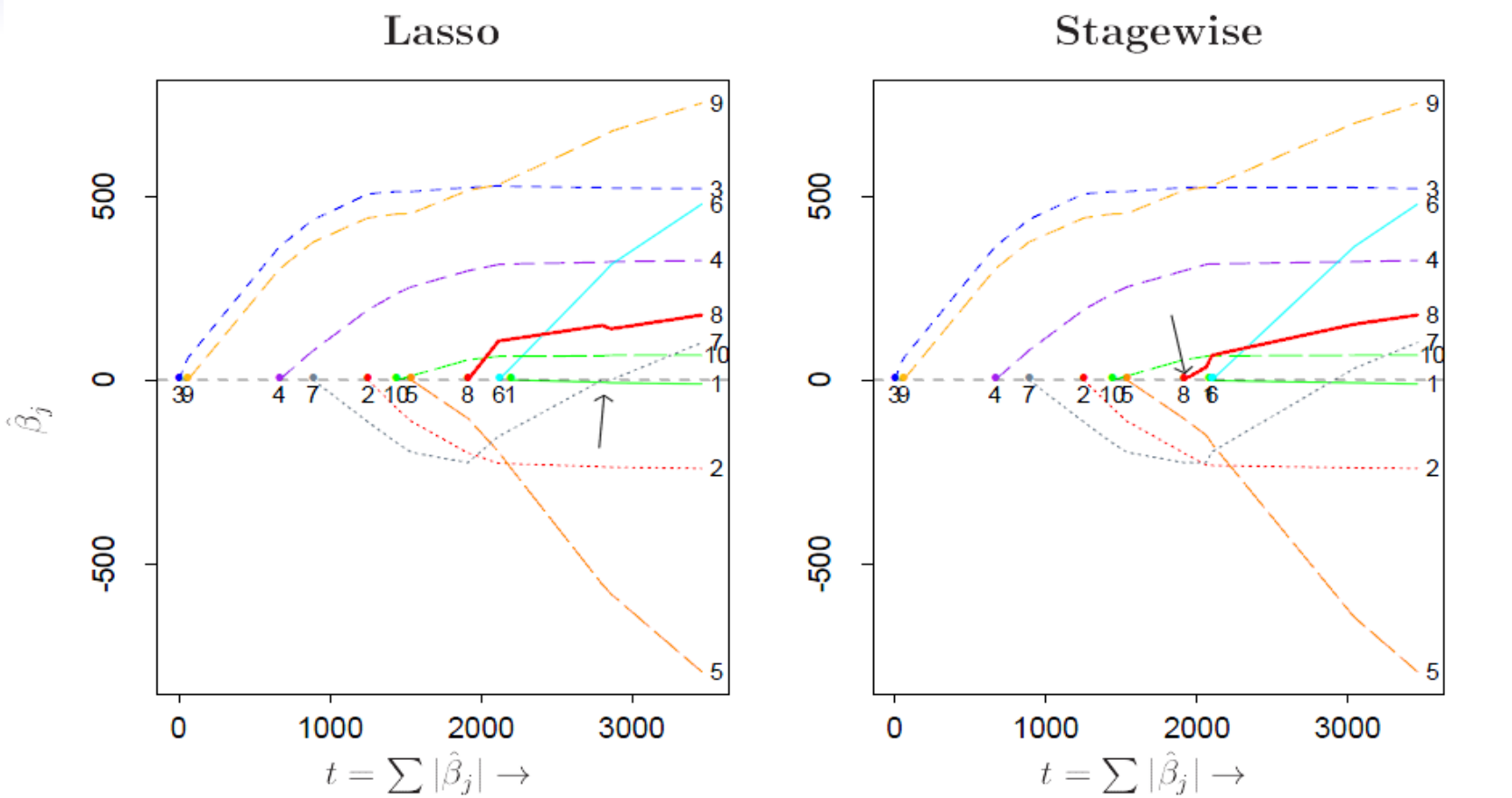
How does these related to Lasso?

Example (Diabetes Study): 442 diabetes patients were measured on 10 baseline variables (age, sex, body mass index, average blood pressure, and six blood serum measurements), as well as the response of interest, a quantitative measure of disease progression one year after baseline. A prediction model was desired for the response variables.

The $\mathbf{X}_{442\times 10}$ matrix has been standardized to have unit L_2-norm in each column and zero mean.

We use Lasso and Forward Stagewise regression to fit the model.

```
library(lars);
data(diabetes);
X=diabetes$x;
y=diabetes$y;
lasso=lars(X, y, type = c("lasso"));
stage=lars(X, y, type = c("forward.stagewise"));
par(mfrow=c(1,2));
plot(lasso);
plot(stage);
```



The two plots are nearly identical, but differ slightly for larger t as shown in the track of covariate 8.

Coincidence? or a general fact ...

Question: Are Lasso and infinitesimal forward stagewise identical?

Answer: With orthogonal design, yes; otherwise, similar.

Question: Why?

Answer: LARs provides answers to these questions, and an efficient way to compute the complete Lasso sequence of solutions.

LARs Algorithm: General Idea

Least Angle Regression is a stylized version of the stagewise procedure. Only p steps are required for the full set of solutions.

The LARs procedure works roughly as follows:

(1). Standardize the predictors to have mean 0 and unit L_2-norm, the response to have mean 0.

(2). As with classic Forward Selection, start with all coefficients equal to zero and find the predictor most correlated with $\mathbf{y}$, say x_{j_1}.

(3). Take the largest step possible in the direction of this predictor until some other predictor, say x_{j_2}, has as much correlation with the current residual.

(4). At this point, LARs parts company with Forward Selection. Instead of continuing along x_{j_1}, LARs proceeds in a direction equiangular between the two predictors until a third variable x_{j_3} earns its way into the most "correlated" set.

(5). LARs then proceeds equiangularly between x_{j_1}, x_{j_2}, x_{j_3}, i.e., along the "least angle direction", until a fourth variable enters.

(6). Continue in this way until all p predictors have been entered.

LARs Algorithm: Two-Predictor Case

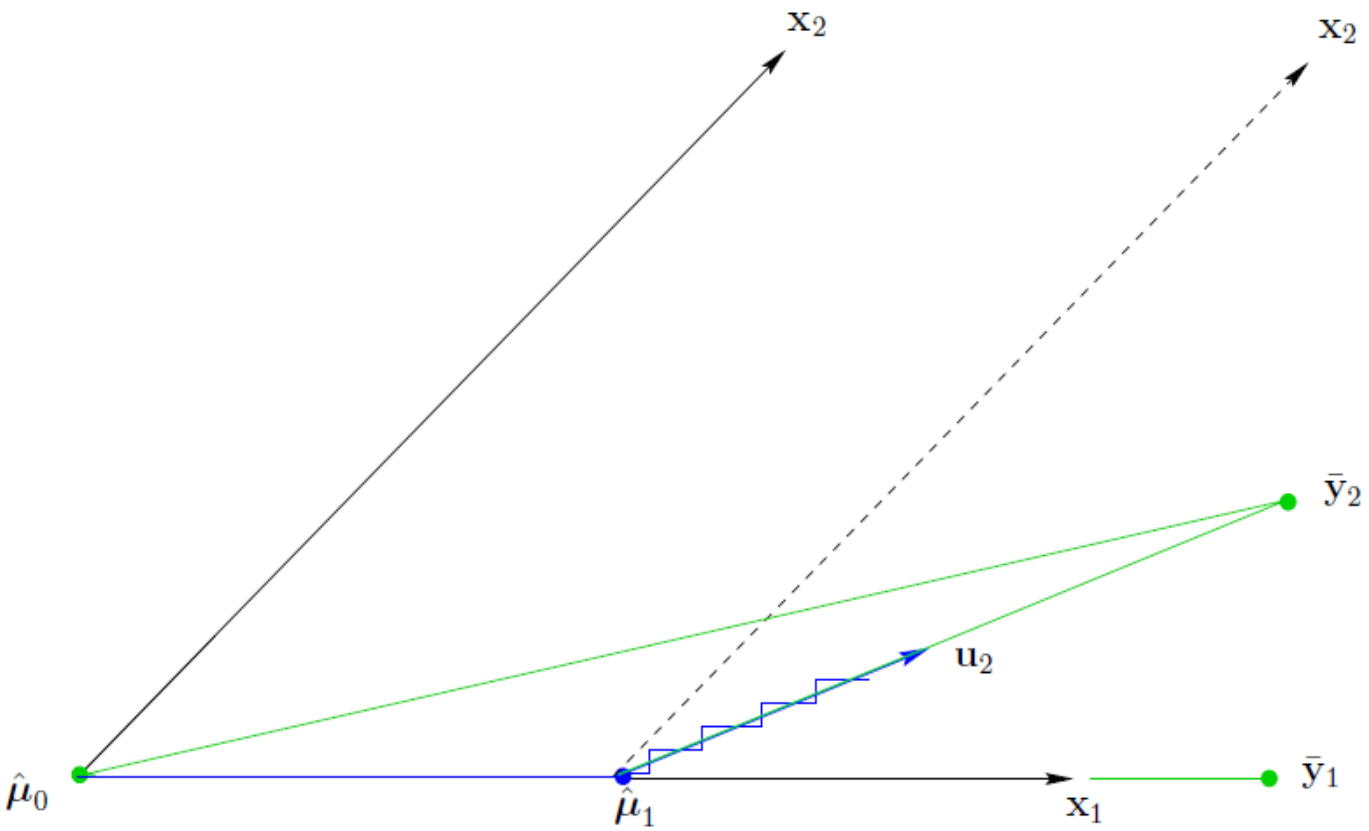
In a two-predictor case, the current correlations depend only on the projection into the linear space spanned by $\mathbf{x}_1$ and $\mathbf{x}_2$.

The algorithm begins at $\mu_0 = 0$, then augments $\hat{\mu}_0$ in the direction of x_1 to $\hat{\mu}_1 = \hat{\mu}_0 + \hat{\gamma}_1 x_1$.

- Stagewise would choose γ_1 equal to some small value ε, and then repeat the process many times.
- Classic Forward Selection would take γ_1 large enough to make $\hat{\mu}_1$ equal the projection of y into $L(x_1)$.
- LARs uses an intermediate value of γ_1, the value that makes the residual equally correlated with x_1 and x_2.

The next LARs estimate is $\hat{\mu}_2 = \hat{\mu}_1 + \gamma_2 u_2$, with γ_2 chosen to make $\hat{\mu}_2 = \mathbf{X}\hat{\beta}_{\text{LSE}}$, where u_2 is the unit bisector of $\mathbf{x}_1$ and $\mathbf{x}_2$.

With $p > 2$ covariates, γ_2 would be smaller, leading to another change of direction.



The LARS algorithm in the case of $m = 2$ covariates; $\bar{\mathbf{y}}_2$ is the projection of $\mathbf{y}$ into $\mathcal{L}(\mathbf{x}_1, \mathbf{x}_2)$. Beginning at $\hat{\boldsymbol{\mu}}_o = \mathbf{0}$, the residual vector $\bar{\mathbf{y}}_2 - \hat{\boldsymbol{\mu}}_o$ has greater correlation with $\mathbf{x}_1$ than $\mathbf{x}_2$; the next LARS estimate is $\hat{\boldsymbol{\mu}}_1 = \hat{\boldsymbol{\mu}}_o + \hat{\gamma}_1 \mathbf{x}_1$, where $\hat{\gamma}_1$ is chosen such that $\bar{\mathbf{y}}_2 - \hat{\boldsymbol{\mu}}_1$ bisects the angle between $\mathbf{x}_1$ and $\mathbf{x}_2$; then $\hat{\boldsymbol{\mu}}_2 = \hat{\boldsymbol{\mu}}_1 + \hat{\gamma}_2 \mathbf{u}_2$, where $\mathbf{u}_2$ is the unit bisector; $\hat{\boldsymbol{\mu}}_2 = \bar{\mathbf{y}}_2$ in the case $m = 2$, but not for the case $m > 2$. The staircase indicates a typical Stagewise path. Here LARS gives the Stagewise track as $\epsilon \to 0$, but a modification is necessary to guarantee agreement in higher dimensions.

LARs in Action:

Least Angle Regreesion

(1). We begin at $\hat{\mu}_0 = 0$.

(2). Compute $\hat{\mathbf{c}} = \mathbf{X}^T(\mathbf{y} - \hat{\mu}_0)$, and $\hat{C} = \max_j |\hat{c}_j|$. Find out

$$\mathcal{A} = \{j : |\hat{c}_j| = \hat{C}\}, \quad s_j = \text{sgn}(\hat{c}_j), j = 1, 2, \ldots, p.$$

(3). Compute

$$\mathbf{X}_{\mathcal{A}} = (\cdots, s_j\mathbf{x}_j, \cdots), \quad \mathcal{G}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}}^T\mathbf{X}_{\mathcal{A}}, \quad A_{\mathcal{A}} = (\mathbf{1}_{\mathcal{A}}^T\mathcal{G}_{\mathcal{A}}^{-1}\mathbf{1}_{\mathcal{A}})^{-1/2}$$

and

$$w_{\mathcal{A}} = A_{\mathcal{A}}\mathcal{G}_{\mathcal{A}}^{-1}\mathbf{1}_{\mathcal{A}}, \quad \mathbf{u}_{\mathcal{A}} = \mathbf{X}_{\mathcal{A}}w_{\mathcal{A}}, \quad \mathbf{a} = \mathbf{X}^T\mathbf{u}_{\mathcal{A}}.$$

(4). Update $\hat{\mu}_0$ with

$$\hat{\mu} = \hat{\mu}_0 + \hat{\gamma}\mathbf{u}_{\mathcal{A}},$$

where

$$\hat{\gamma} = \min_{j \in \mathcal{A}^C}^{+} \left\{ \frac{\hat{C} - \hat{c}_j}{A_{\mathcal{A}} - a_j}, \frac{\hat{C} + \hat{c}_j}{A_{\mathcal{A}} + a_j} \right\},$$

and $\min^+$ indicates that the minimum is taken over only positive components.

Relationship between LARs, Lasso and Forward Stagewise Regression

Lasso and forward stagewise can be thought of as restricted versions of LARs.

For Lasso:

Start with LAR. If a coefficient crosses zero, stop. Drop that predictor, recompute the best direction and continue. This gives the Lasso path.

For forward stagewise:

Start with LAR. Compute best (equal angular) direction at each stage. If direction for any predictor j doesn't agree in sign with $\text{corr}(r, x_j)$, project direction into the "positive cone" and use the projected direction instead.

In other words, forward stagewise always moves each predictor in the direction of $\text{corr}(r, x_j)$.

The incremental forward stagewise procedure approximates these steps, one predictor at a time. As step size $\varepsilon \to 0$, we can show that it coincides with this modified version of LARs.

Outline

References

- Antoniadis, A. and J. Fan (2001). Regularization of Wavelet Approximations. *Journal of the American Statistical Association* 96, 939-967.
- Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics* 37(4), 373-384.
- Efron, B., T. Hastie, I. Johnstones, and R. Tibshirani (2004). Least angle regression. *The Annals of Statistics* 32(2), 407-499.
- Fan, J. and R. Li (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association* 96(456), 1348-1360.
- Frank, I. E. and J. H. Friedman (1993). A Statistical View of Some Chemometrics Regression Tools. *Technometrics* 35(2), 109-135.
- Friedman, J., T. Hastie, H. Höfling, and R. Tibshirani (2007). Pathwise coordinate optimization. *Annals of Applied Statistics* 2, 302-332.
- Huang, J., P. Breheny, and S. Ma (2012). A selective review of group selection in high dimensional models. *Statist. Sci.* 27(4), 481-499.
- Huang, J., S. Ma, H. Xie, and C.H. Zhang (2009, June). A group bridge approach for variable selection. *Biometrika* 96(2), 339-355.

- Osborne, M. R., B. Presnell, and B. A. Turlach (2000). On the LASSO and Its Dual. *Journal of Computational and Graphical Statistics* 9(2), 319-337.
- Park, Trevor, Casella, and George (2008, June). The Bayesian Lasso. *Journal of the American Statistical Association* 103(482), 681-686.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society (Series B)* 58, 267-288.
- Yuan, M. and Y. Lin (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 68(1), 49-67.
- Zhang, C.H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics* 38, 894-942.