Lecture 11: Sparsity

Applied Multivariate Analysis

Math 570, Fall 2014

Xingye Qiao
Department of Mathematical Sciences
Binghamton University

E-mail: qiao@math.binghamton.edu

Outline

1. Linear Regression and LASSO

2. Computing of LASSO

3. Other Loss Functions and Penalties

The next section would be . . . . . .

1 Linear Regression and LASSO

2 Computing of LASSO

3 Other Loss Functions and Penalties

Large p small n

- ... when there are more parameters than observations.
- Also known as High-dimensional, Low-sample size problem.
- Traditional methods fail
  - Linear regression: $(\mathbf{X}\mathbf{X}')$ is not invertible
  - LDA: sample covariance $\mathbf{S}$ is not invertible
  - Classification: infinitely many hyperplanes that can perfectly separate two classes (even if there is no distributional difference between the two classes).

Sparsity

- . . . is an assumption: most of the parameters bear no significance to the model; the number of parameters that matter is very sparse (small).
- Can be reasonable and desirable sometimes.
- Does not always work (for example, functional data)
- Can bring computational advantage (will see)

Linear Regression

- We consider the canonical linear regression setting,

$$\boldsymbol{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{1}$$

- $\boldsymbol{Y} = (y_1, \ldots, y_n)^T$ is the vector of the response variable
- $\mathbf{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^T$ is the n by p design matrix where each $\boldsymbol{x}_i \in \mathbb{R}^p$ (we follow this conventional setup in this lecture)
- $\boldsymbol{\varepsilon}$ is an n-dimensional vector for the random error
- $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^T$ is the coefficient vector. The ordinary least square solution to the linear regression problem minimizes the residual sum of squared, that is,

$$\boldsymbol{\beta}_{OLS} = \operatorname*{argmin}_{\boldsymbol{\beta}^*} \sum_{i=1}^{n} (Y_i - \boldsymbol{x}_i^T \boldsymbol{\beta}^*)^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}.$$

Sparsity assumption suggests that $\|\boldsymbol{\beta}\|_0 := \{\#\text{ of nonzero } \beta_j\text{'s}\}$ should be a small number, leading to minimizing the penalized RSS

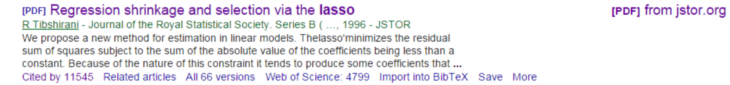
$$\sum_{i=1}^{n}(Y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\|\boldsymbol{\beta}\|_0$$

- $\|\boldsymbol{\beta}\|_0 = \sum_{j=1}^{p} p(\beta_j) = \sum_j \mathbb{1}_{\{\beta_j \neq 0\}}$. For each β_j, $p(\beta_j)$ is a constant 1 except when $\beta_j = 0$. Hence it is not continuous w.r.t. β_j.
- $\|\boldsymbol{\beta}\|_0$ can be relaxed to L_1 norm of $\boldsymbol{\beta}$, i.e., $\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} |\beta_j|$

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n}(Y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\|\boldsymbol{\beta}\|_1$$

Solution to this problem is called LASSO (least absolute shrinkage and selection operator). $\|\boldsymbol{\beta}\|_1$ is called L_1 (norm) penalty or often just the lasso penalty.

Tibshirani's lasso formulation



[PDF] Regression shrinkage and selection via the **lasso**

R Tibshirani - Journal of the Royal Statistical Society. Series B (..., 1996 - JSTOR

We propose a new method for estimation in linear models. Thelasso'minimizes the residual sum of squares subject to the sum of the absolute value of the coefficients being less than a constant. Because of the nature of this constraint it tends to produce some coefficients that ...

Cited by 11545 Related articles All 66 versions Web of Science: 4799 Import into BibTeX Save More

[PDF] from jstor.org

Originally, Robert Tibshirani's lasso was defined as

$$\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} (Y_i - \boldsymbol{x}_i^T \boldsymbol{\beta})^2$$

$$\text{subject to } \|\boldsymbol{\beta}\|_1 \leq t$$

- Without the constraint, there are infinitely many $\boldsymbol{\beta}$ that can make the objective function 0, hence the space to search for $\boldsymbol{\beta}$ is too large.
- The constraint $\|\boldsymbol{\beta}\|_1 \leq t$ makes the space to search from small. It has the effect to shrink some coefficients to 0.

Orthonormal example

- To show how LASSO works, we consider a simple example with $\mathbf{X}^T\mathbf{X} = \mathbb{I}_p$, i.e., the orthonormal case.
- Let $X_{(j)}$ be the jth column of $\mathbf{X}$. The orthonormal design says $\|X_{(j)}\| = 1$ and $X'_{(j)}X_{(j')} = 0$.
- Recall that in this case $\boldsymbol{\beta}_{OLS} = \mathbf{X}^T\boldsymbol{Y}$
- $\frac{1}{2}\sum_{i=1}^n (Y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\|\boldsymbol{\beta}\|_1$ can be written in matrices:

$$Q(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|_1$$

$$\begin{aligned}\frac{\partial Q(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} &= -\mathbf{X}^T(\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda\text{sign}(\boldsymbol{\beta}) \\ &= -\boldsymbol{\beta}_{OLS} + \boldsymbol{\beta} + \lambda\text{sign}(\boldsymbol{\beta})\end{aligned}$$

$$
\begin{aligned}
\frac{\partial Q(\boldsymbol{\beta})}{\partial \beta_j} &= -\beta_j^{OLS} + \beta_j + \lambda \text{sign}(\beta_j) \\
&= \begin{cases} -\beta_j^{OLS} + \beta_j + \lambda, & \text{if } \beta_j > 0 \\ -\beta_j^{OLS} + \beta_j + c\lambda, & \text{if } \beta_j = 0 \\ -\beta_j^{OLS} + \beta_j - \lambda, & \text{if } \beta_j < 0 \end{cases}
\end{aligned}
$$

where $-1 < c < 1$.
Hence, setting it to 0, we have that the critical point β_j^* is

$$
\beta_j^* = \begin{cases} \beta_j^{OLS} - \lambda, & \text{if } \beta_j^* > 0 \Leftrightarrow \text{if } \beta_j^{OLS} > \lambda \\ \beta_j^{OLS} - c\lambda = 0 & \text{if } \beta_j^* = 0 \Leftrightarrow \text{if } |\beta_j^{OLS}| < \lambda \\ \beta_j^{OLS} + \lambda, & \text{if } \beta_j^* < 0 \Leftrightarrow \text{if } \beta_j^{OLS} < -\lambda \end{cases}
$$

- Let's be more careful on the case $|\beta_j^{OLS}| < \lambda$, since Q is not differentiable at $\beta_j^* = 0$.
- Assume that $0 < \beta_j^{OLS} < \lambda$, then

$$\frac{\partial}{\partial \beta_j} Q(\boldsymbol{\beta}) = -\beta_j^{OLS} + \beta_j + \lambda \text{sign}(\beta_j)$$
$$\begin{cases} < 0, \text{ as } \beta_j < 0 \\ > 0, \text{ as } \beta_j > 0 \end{cases}$$
$$\Rightarrow \beta_j = 0 \text{ is the minimizing point}$$

- Similarly, assume that $-\lambda < \beta_j^{OLS} < 0$, then

$$\frac{\partial}{\partial \beta_j} Q(\boldsymbol{\beta}) = -\beta_j^{OLS} + \beta_j + \lambda \text{sign}(\beta_j)$$
$$\begin{cases} < 0, \text{ as } \beta_j < 0 \\ > 0, \text{ as } \beta_j > 0 \end{cases}$$
$$\Rightarrow \beta_j = 0 \text{ is the minimizing point}$$

LASSO Solution under Orthonormal Design

Thus we have that under the orthonormal design, the solution to LASSO is

$$\boldsymbol{\beta}_{lasso} = S_\lambda(\boldsymbol{\beta}^{OLS})$$

where $S_\lambda(\cdot)$ is the soft-thresholding operator defined by

$$\begin{aligned} S_\lambda(x) &= \begin{cases} x - \lambda, & \text{if } x > \lambda \\ 0, & \text{if } |x| \leq \lambda \\ x + \lambda, & \text{if } x < -\lambda \end{cases} \\ &= \operatorname{sign}(x)(|x| - \lambda)_+ \end{aligned}$$

The next section would be . . . . . .

1 Linear Regression and LASSO

2 Computing of LASSO

3 Other Loss Functions and Penalties

Quadratic program with linear constraints.

- The lasso problem is essentially a quadratic programming problem with $2p$ linear constraint.
- Recast β_j as $\beta_j = \beta_j^+ - \beta_j^-$ with constraints that $\beta_j^+ > 0$ and $\beta_j^- > 0$ (hence $2p$ constraints)
- Replace $\|\boldsymbol{\beta}\|$ by $\sum_{j=1}^{p}(\beta_j^+ + \beta_j^-)$
- Very inefficient for large p.
- Must train $\boldsymbol{\beta}$ for each fixed λ
- Want to have a method to calculate the full solution path corresponding to a full span of λ values.

Least angle regression (LARS)

- Forward stepwise regression: an ancient idea. To add one variable at a time. At each step, identify the best variable to include in the model, and then update the least square fit sequentially.
- Least angle regression (LARS): a similar idea. But instead of exploiting the current variable at much as possible, LARS only fits the current variable to a certain level.

LARS–1

- At first step, identify the variable most correlated with the response.
- "Slowly" increase the coefficient for this variable (along the direction of the LS fit)
    - This causes the correlation between this variable and the residual of the fit to decrease (as the coefficient slowly increases).
    - . . . , until another variable has the same absolute correlation (with the residual) as the current one. At this point, both variables are in the *active* set.

LARS–2

- At each point when a new variable enters the active set (denoted as $\mathcal{A}$), "slowly" increase the coefficient of the variables in $\mathcal{A}$ along the direction of the LS fit of the current residual on $\mathcal{A}$, i.e.

  $$\boldsymbol{\beta}_{\mathcal{A}}(\alpha) = \boldsymbol{\beta}_{\mathcal{A}}(0) + \alpha\boldsymbol{\delta}$$

  where $\boldsymbol{\delta} = (\mathbf{X}_{\mathcal{A}}^T\mathbf{X}_{\mathcal{A}})^{-1}\mathbf{X}_{\mathcal{A}}^T\boldsymbol{r}$ and $\boldsymbol{r} = \boldsymbol{y} - \mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}_{\mathcal{A}}(0)$. Note that the variable which just entered the active set had coefficient 0.

  - An important feature of this process is that the correlations between the variables in $\mathcal{A}$ and the residual $\boldsymbol{r}(\alpha) := \boldsymbol{y} - \mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}_{\mathcal{A}}(\alpha)$ are all equal (to each other) and they decrease at the same time as $\alpha \uparrow$

- . . . , until there is another variable which has as much correlation with the residual $\boldsymbol{r}(\alpha)$ as the ones in the active set $\mathcal{A}$, at which point, the new variable enters the active set and the iterations restart.

- At each time point, there is a value for $\boldsymbol{\beta}$. This corresponds to a value of $t = \sum_{j=1}^{p} |\beta_j|$
- At the very beginning, there was no variable in the active set. This corresponds to $t = 0$, or $\lambda = \infty$
- In the very end, all variables are in the active set. This corresponds to $t = \sum_{j=1}^{p} |\beta_j^{OLS}|$, or $\lambda = 0$

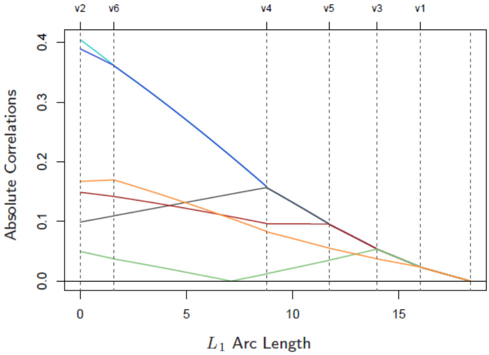
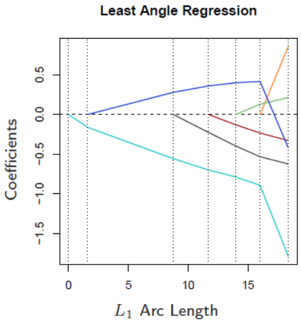
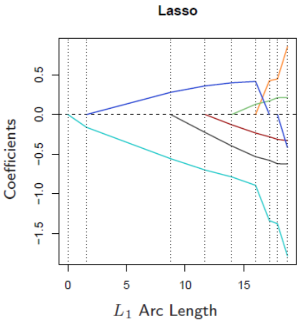


FIGURE 3.14. *Progression of the absolute correlations during each step of the LAR procedure, using a simulated data set with six predictors. The labels at the top of the plot indicate which variables enter the active set at each step. The step length are measured in units of L_1 arc length.*

LARS and LASSO

- The solutions to LARS and LASSO are very similar. With a small modification, the LARS algorithm produces the whole solution path of LASSO.
- Modification: if one variables in the active set has a correlation (with the residual) 0, then drop the variable out of the active set, recalculate the direction of increment $\boldsymbol{\delta} = (\mathbf{X}_{\mathcal{A}}^T \mathbf{X}_{\mathcal{A}})^{-1} \mathbf{X}_{\mathcal{A}}^T \boldsymbol{r}$





Properties of LARS

- We can exactly calculate the needed step size at the beginning of each step, and do not need to carefully take many tiny steps and recheck the correlation for many times.
- The solution path is piecewise linear (with respect to t). Only need to work out the values at turning points.
- Extremely fast.

Coordinate descent

Coordinate descent algorithm

1. Fix all the other variables, and vary the coefficient for one variable β_j to achieve the minimal of $Q(\beta_j|\boldsymbol{\beta}_{-j})$
2. Do the same for all the variables.
3. Iterate until convergence

This is actually pretty easy and fast.

$$Q(\boldsymbol{\beta}) = \frac{1}{2}\| \boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|_1$$

$$Q(\beta_j; \boldsymbol{\beta}_{-j}) = \frac{1}{2}\| \boldsymbol{Y} - \mathbf{X}_{(-j)}\boldsymbol{\beta}_{-j} - \mathbf{X}_{(j)}\beta_j\|^2 + \lambda\|\boldsymbol{\beta}_{-j}\|_1 + \lambda|\beta_j|$$

Here $\boldsymbol{Y} - \mathbf{X}_{(-j)}\boldsymbol{\beta}_{-j}$ is the new vector of residuals, and β_j is the only unknown variable. This is a one-dimensional simple linear regression model with the residual on the jth variable $\mathbf{X}_{(j)}$.

If each variable has been pre-normalized, then it is just as simple as

$$\beta_j(\lambda) = S_\lambda \left[\mathbf{X}_{(j)}^T\{\boldsymbol{Y} - \mathbf{X}_{(-j)}\boldsymbol{\beta}_{-j}\}\right]$$

Warm start

- For each j, the update can be done in just one line of command. We can quickly go over all the variables (call this one round), and we can do many rounds until the estimates do not change between two rounds.
- Often we want to calculate the solution for a range of λ, not just one λ
- Warm start: provide a reasonable guess for the correct solution at each stage.
    - Start with setting λ to be a large value, so that $\widehat{\boldsymbol{\beta}}(\lambda) = \mathbf{0}$ (we may want to choose the smallest value where this is true).
    - Decrease the λ by a little, and use the solution from the previous λ as the initial values (called "warm start").

LARS vs Coordinate descent

- LARS provides exact piecewise linear solution to LASSO, while the solution from Coordinate descent is on a grid of many discrete points.
- For large problem, Coordinate descent is faster.
- Piecewise linear path algorithm can be found useful for other problems, as long as the loss function is piecewise linear or quadratic in β, and the penalty function is piecewise linear in β
- Coordinate descent turns out to be useful for a broader problems in statistics and machine learning.

Computing: R

- LARS: use package ‘lars’
- Coordinate descent: use packages ‘lassoshotting’, ‘glmnet’, etc.

The next section would be . . . . . .

1 Linear Regression and LASSO

2 Computing of LASSO

3 Other Loss Functions and Penalties

Loss+Regularization

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \boldsymbol{x}_i^T\boldsymbol{\beta})^2 + \lambda\|\boldsymbol{\beta}\|_1$$

Lasso belongs to a more general family of loss function with a regularization term.

$$\min \frac{1}{n}\sum_{i=1}^{n} L(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i) + \lambda R(\boldsymbol{\beta})$$

- The loss function $L(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i)$ measures the goodness-of-fit of the model.
  - Squared loss: $(y_i - f(\boldsymbol{x}_i))^2$
  - Hinge loss: $(1 - y_i f(\boldsymbol{x}_i))_+$
  - Logistic loss: $\log(1 + \exp(-y_i f(\boldsymbol{x}_i)))$
  - Any negative log-likelihood

$$\min \frac{1}{n} \sum_{i=1}^{n} L(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i) + \lambda R(\boldsymbol{\beta})$$

- The regularization term $R(\boldsymbol{\beta})$ controls the complexity of the model. In particular, since the model above can be written as $\min \frac{1}{n} \sum_{i=1}^{n} L(\boldsymbol{\beta}, \boldsymbol{x}_i, y_i)$ subject to $R(\boldsymbol{\beta}) < t$, the addition of the constraint make the space to search $\boldsymbol{\beta}$ smaller.
- The regularization term is also called penalty term. Common examples of the penalty functions are
  - L_0 norm: $\|\boldsymbol{\beta}\|_0$ – best subset selection, AIC or BIC.
  - L_1 norm: $\|\boldsymbol{\beta}\|_1$ – lasso
  - L_2 norm: $\|\boldsymbol{\beta}\|_2^2$ – ridge regression
  - L_q norm: $\|\boldsymbol{\beta}\|_q^q = \sum_{j=1}^{p} |\beta_j|^q$ $(1 < q < 2)$ – bridge regression
  - SCAD
  - Elastic net: $c_1\|\boldsymbol{\beta}\|_1 + c_2\|\boldsymbol{\beta}\|_2^2$
  - MCP
  - ...
  - ...

Ridge regression

$$\min L(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \frac{\lambda}{2}\|\boldsymbol{\beta}\|_2$$

We have that

$$\frac{\partial L(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -\mathbf{X}^T(\boldsymbol{Y} - \mathbf{X}\boldsymbol{\beta}) + \lambda\boldsymbol{\beta}$$

Setting it to 0 and we have

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

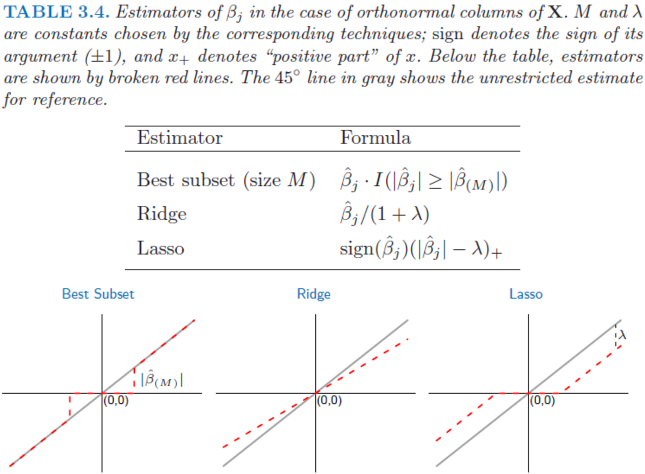


TABLE 3.4. *Estimators of β_j in the case of orthonormal columns of $\mathbf{X}$. M and λ are constants chosen by the corresponding techniques;* sign *denotes the sign of its argument (±1), and x_+ denotes "positive part" of x. Below the table, estimators are shown by broken red lines. The 45° line in gray shows the unrestricted estimate for reference.*

Estimator	Formula
Best subset (size M)	$\hat\beta_j \cdot I(\lvert\hat\beta_j\rvert \geq \lvert\hat\beta_{(M)}\rvert)$
Ridge	$\hat\beta_j/(1+\lambda)$
Lasso	$\text{sign}(\hat\beta_j)(\lvert\hat\beta_j\rvert - \lambda)_+$

- Best subset selection: hard-thresholding
- Ridge: proportional shrinkage
- lasso: soft-thresholding

Simple case with $p = 2$

The criterion $\sum_{i=1}^{n}(y_i - \beta' \boldsymbol{x}_i)^2 = \|\boldsymbol{Y} - \mathbf{X}\beta\|^2 =$ $(\boldsymbol{Y} - \mathbf{X}\beta)^T(\boldsymbol{Y} - \mathbf{X}\beta) = [\boldsymbol{\beta}_{OLS} - \beta]'(\mathbf{X}^T\mathbf{X})[\boldsymbol{\beta}_{OLS} - \beta] + c.$

- Hence we seek to find $\boldsymbol{\beta}$ to minimize some scaled distance to the "true" OLS estimator $\boldsymbol{\beta}_{OLS}$, subject to a constraint that $\boldsymbol{\beta}$ is coming from a restricted space specified by $R(\boldsymbol{\beta}) < t$. See the next two figures.
- When $R(\boldsymbol{\beta}) < t$ has a certain shape, $\boldsymbol{\beta}$ should tend to have a zero element.

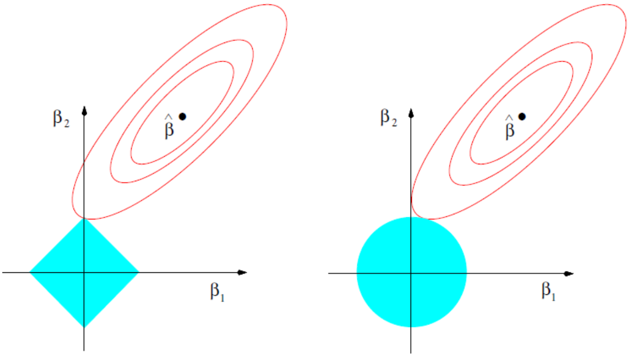


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions* $|\beta_1| + |\beta_2| \leq t$ *and* $\beta_1^2 + \beta_2^2 \leq t^2$*, respectively, while the red ellipses are the contours of the least squares error function.*

L_q norm penalty – bridge regression

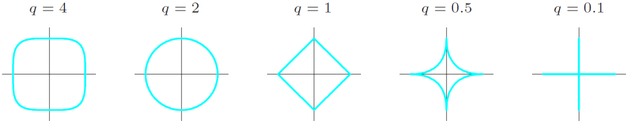


FIGURE 3.12. *Contours of constant value of* $\sum_i |\beta_j|^q$ *for given values of* q.

- Unless $q \leq 1$, there is no sparsity effect.
- The cases of $q \neq 1$ and $q \neq 2$ are difficult to calculate.

SCAD Penalty

"Variable selection via nonconcave penalized likelihood and its oracle properties" J Fan, R Li - *Journal of the American Statistical Association*, 2001

- Fan and Li had concern about the bias of β_j for those true signal j's.
- For true signals, the bias is $\pm\lambda$
- In addition, they articulate three desirable properties for a penalized estimator
  1. sparsity,
  2. unbiasedness and
  3. continuity

SCAD Penalty

Fan and Li proposed to use $\lambda R(\boldsymbol{\beta}) = \sum_j p(|\beta_j|)$, where the function $p(t) : \mathbb{R}_+ \mapsto \mathbb{R}_+$ is defined as

$$p'(t) = \min\{1, (\gamma - t/\lambda)_+/(\gamma - 1)\}$$

- Smoothly Clipped Absolute Deviation – SCAD
- Satisfy the conditions
- Has some good theoretical property
- This penalty function is not convex

MCP

“Nearly unbiased variable selection under minimax concave penalty” CH Zhang - *The Annals of Statistics*, 2010

- Similar to SCAD, nonconvex.
- Improved theoretical properties.

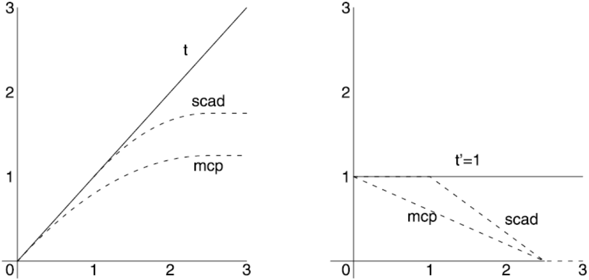


FIG. 1. *The ℓ_1 penalty $\rho_1(t) = t$ for the LASSO along with the MCP $\rho_2(t)$ and the SCAD penalty $\rho_3(t)$, $t > 0$, $\gamma = 5/2$. Left: penalties $\rho_m(t)$. Right: their derivatives $\dot{\rho}_m(t)$.*

Figure: Courtesy of Zhang 2010

Elastic net

- Motivated by the observation that when there are groups of variables that have similar effect, lasso tends to select only one of them and ignore the others.
- A compromise between the ℓ_1 and ℓ_2 norm penalties.
- $c_1\|\boldsymbol{\beta}\|_1 + c_2\|\boldsymbol{\beta}\|_2^2$

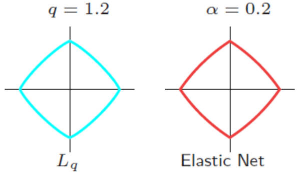


FIGURE 3.13. *Contours of constant value of $\sum_j |\beta_j|^q$ for $q = 1.2$ (left plot), and the elastic-net penalty $\sum_j(\alpha\beta_j^2+(1-\alpha)|\beta_j|)$ for $\alpha = 0.2$ (right plot). Although visually very similar, the elastic-net has sharp (non-differentiable) corners, while the $q = 1.2$ penalty does not.*

Computing with concave penalty

- Coordinate descent may not be very useful, since there is no clear closed-form solution.
- Often use local linear or quadratic approximations.

LQA

- Local Quadratic Approximation
- SCAD: Approximate $p(t)$ [which is concave], by

$$p(t^{(0)}) + \frac{1}{2}\frac{p'(t^{(0)})}{t^{(0)}}(t^2 - {t^{(0)}}^2)$$

Since the other terms are fixed, for each iteration, essentially the problem becomes a weighted ridge regression problem with weighted ℓ_2 penalty.

LLA

- Local Linear Approximation (Zou and Li 2008)
- Approximate $p(t)$ by

$$p(t^{(0)}) + p'(t^{(0)})(t - t^{(0)})$$

- Essentially a weighted lasso ℓ_1 penalty at each iteration.
- They suggest that we could stop after 1 step.

Other fruitful developments

- Weighted lasso
- Adaptive lasso
- Fused lasso
- Grouped lasso
- Relaxed lasso

Each of these can be a good topic to present/read.