Statistical Learning

Penalized Linear Regression: Part II

Spring 2024

Outline

- Some theoretical motivation
- Bias correction: Non-negative garrote, Adaptive Lasso
- Unbiased penalties: SCAD, MCP
- Penalties for special data structures: grouped lasso, fused lasso

Some Theoretical Motivations

What we have learned...

- Put a penalty term on the size of parameter estimates to control the complexity
- A unified framework is to minimize the objective function

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \sum_{j=1}^{p} P_\lambda(\beta_j),$$

  where $P_\lambda(\cdot)$ is a penalty function applied on the value of each parameter, and λ is a tuning parameter.

  - Lasso: $P_\lambda(\beta) = \lambda|\beta|$
  - Ridge: $P_\lambda(\beta) = \lambda\beta^2$
  - Best subset: $P_\lambda(\beta) = \lambda\mathbf{1}\{\beta \neq 0\}$
  - Elastic net: $P_\lambda(\beta) = \lambda_1|\beta| + \lambda_2\beta^2$

What we are missing...

- In a high-dimension setting, suppose that there are only a small subset of variables ($p_0 < p$) that are relevant to the outcome Y. In other words, the model is sparse.
- Denote this set of true nonzero variables as $\mathcal{S} = \{j : \beta_j \neq 0\}$, hence, $|\mathcal{S}| = p_0$.
- W.L.O.G, we assume that $\mathcal{S} = \{1, \ldots, p_0\}$, i.e.,

$$Y = \beta_1 X_1 + \ldots + \beta_{p_0} X_{p_0} + \epsilon.$$

- We want to know whether we can consistently select these variables, and estimate them

Oracle Properties

- Lets assume that there is an underlying covariance structure of X such that $\mathbf{X}^\mathsf{T}\mathbf{X}/n \to \mathbf{\Sigma}$, where $\mathbf{\Sigma}$ is positive definite.
- If we perform the OLS estimator on the set of variables $\{X_j : \beta_j \neq 0\}$, we could have

  $$\sqrt{n}(\widehat{\boldsymbol{\beta}}_{\mathcal{S}}^{\text{ols}} - \boldsymbol{\beta}_{\mathcal{S}}) \to_d \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{11}^{-1}),$$

  where $\mathbf{\Sigma}_{11}$ is the upper-left $p_0 \times p_0$ matrix of Σ, i.e.,

  $$\text{Cov}(X) = \mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{21} & \mathbf{\Sigma}_{22} \end{bmatrix}$$

- **Question:** Is it possible that our shrinkage estimator would (if we properly choose λ, which may depend on n)
  - identify the correct subset of predictors
  - has the optimal estimation rate (the rate of OLS estimator performed on the true nonzero set)

Oracle Properties

- Formally, these are called the oracle properties (Fan and Li, 2001):
  - Selection consistency:

$$\{j : \widehat{\beta}_j \neq 0\} = \mathcal{S}$$

  - Asymptotic normality with optimal estimation rate:

$$\sqrt{n}(\widehat{\boldsymbol{\beta}}_{\mathcal{S}} - \boldsymbol{\beta}_{\mathcal{S}}) \to_d \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{11}^{-1})$$

- Unfortunately, Lasso does not enjoy the oracle properties — Lasso can achieve either one of them, but not simultaneously.

Selection consistency

- It is again intuitive to investigate some simple cases — orthogonal design $\mathbf{X}^\mathsf{T}\mathbf{X}/n = \mathbf{I}$.
- In addition, we assume that the errors $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$
- Then applying Bonferroni and the property of Gaussian tail probability, we have

$$\mathsf{P}(\exists\, j \in \mathcal{S}^c, \text{ s.t. } \widehat{\beta}_j \neq 0) \leq 2 \exp\left\{-\frac{n\lambda^2}{2\sigma^2} + \log p\right\}$$

- Note 1. The probability is: any of the unimportant parameter was estimated to be nonzero.
- Note 2. For Gaussian tail, $\mathsf{P}(z > t) \leq \exp\{-t^2/2\sigma^2\}$.

Selection consistency

- This implies, for p fixed, if we set $\sqrt{n}\lambda \to \infty$,

$$\mathsf{P}(\exists\, j \in \mathcal{S}^c, \text{ s.t. } \widehat{\beta}_j \neq 0) \to 0.$$

- If we have $\sqrt{n}\lambda \to c$, then

$$\mathsf{P}(\exists\, j \in \mathcal{S}^c, \text{ s.t. } \widehat{\beta}_j \neq 0) \to 1 - \delta.$$

- For p diverging (what rate?), setting $\lambda = \mathcal{O}(\sigma\sqrt{\log p/n})$ will ensure a positive chance of eliminating all unimportant parameters.
- If p grows exponentially with n, it won't work...

Estimation consistency

- As we discussed earlier, the shrinkage (bias) is a magnitude of λ, so, as long as $\lambda \to 0$, we have estimation consistency.
- However, this does not imply $\sqrt{n}$ rate of consistency (rate of OLS), especially if λ has to satisfy $\sqrt{n}\lambda \to \infty$ to achieve selection consistency...
- This sounds like a dilemma, we cannot have selection consistency and $\sqrt{n}$ rate simultaneously in a high-dimensional setting
- A compromise is to take $\sqrt{n}\lambda \to c$ in high dimensional settings.
- Another solution? Use a bias correction procedure or an unbiased penalty.

Remark

- When the design matrix is not orthogonal, things still work similarly.
- The irrepresentable condition.
- An enormous amount of papers on this topic. Things to read if interested:
    - Zhao and Yu (2006)
    - Meinshausen and Yu (2009)
    - Lv and Fan (2009)
    - Zhang (2010)

Bias Correction and Unbiased Penalties

Bias Correction

- Bias correction:
    - Non-negative garrote
    - Adaptive Lasso
- Unbiased penalty:
    - SCAD
    - MCP

Non-negative Garrote

- Non-negative Garrote was proposed by Breiman (1994)
- Suppose we can have an initial estimate: $\widehat{\boldsymbol{\beta}}^{\text{ols}}$
- We can perform a model by shrinking the coefficients:

$$\underset{d_1,\ldots,d_p}{\text{minimize}} \quad \frac{1}{2n}\left\|\mathbf{y} - \sum_{j=1}^{p} d_j \widehat{\beta}_j^{\text{ols}} \mathbf{x}_j\right\|^2 + \lambda \sum_{j=1}^{p} d_j,$$

subject to $d_j \geq 0$ for all j.

Non-negative Garrote

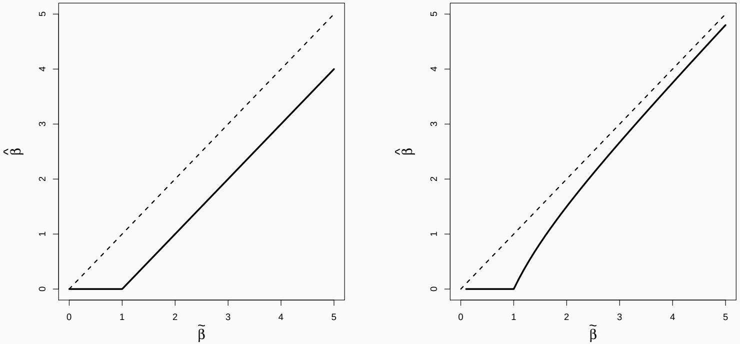
- Final estimate $\widehat{\beta}_j^{\text{ng}} = \widehat{d}_j \widehat{\beta}_j^{\text{ols}}$
- In orthogonal designs, the optimal d_j's are

$$d_j = \left(1 - \frac{n\lambda}{(\widehat{\beta}_j^{\text{ols}})^2}\right)_+$$

  which can be shrink to exactly 0 if $\widehat{\beta}_j^{\text{ols}}$ is small.

- For $\widehat{\beta}_j^{\text{ols}}$ sufficiently large, d_j is almost 1, which reduces the bias.

Comparison: Lasso vs. Garrote



Comparing Lasso shrinkage with non-negative garrote shrinkage

Adaptive Lasso

- Adaptive Lasso was proposed by Zou (2006)
- Suppose we can have an initial estimate $\widetilde{\boldsymbol{\beta}}$ that is $\sqrt{n}$ consistent
- Adjust the Lasso penalty based on how large $\widetilde{\boldsymbol{\beta}}$ is

$$\widehat{\boldsymbol{\beta}} = \underset{\beta_1,\ldots,\beta_p}{\arg\min} \quad \frac{1}{2n}\Big\|\mathbf{y} - \sum_{j=1}^{p} \beta_j \mathbf{x}_j\Big\|^2 + \lambda \sum_{j=1}^{p} \frac{1}{|\widetilde{\beta}_j|^\gamma} |\beta_j|,$$

  for a pre-chosen $\gamma > 0$.

- Note: the penalty is essentially $\frac{\lambda}{|\widetilde{\beta}_j|^\gamma}$, which will be different for each β_j. Large $\widetilde{\beta}_j$ means small penalty, which reduces the bias

Adaptive Lasso

- The adaptive Lasso and the Non-negative Garrote are almost the same. If we take $\gamma = 1$ and use $\widehat{\boldsymbol{\beta}}^{\text{ols}}$ as the initial estimator $\widetilde{\boldsymbol{\beta}}$ in then adaptive Lasso, then we are solving for

$$\widehat{\boldsymbol{\beta}} = \underset{\beta_1,\ldots,\beta_p}{\arg\min} \quad \frac{1}{2n}\Big\|\mathbf{y} - \sum_{j=1}^{p} \beta_j \mathbf{x}_j\Big\|^2 + \lambda \sum_{j=1}^{p} \frac{1}{|\widehat{\beta}_j^{\text{ols}}|}|\beta_j|,$$

which is equivalent to treating $\frac{|\beta_j|}{|\widehat{\beta}_j^{\text{ols}}|}$ as d_j, and rescale each $\mathbf{x}_j$

$$\text{minimize} \quad \frac{1}{2n}\Big\|\mathbf{y} - \sum_{j=1}^{p} \frac{|\beta_j|}{|\widehat{\beta}_j^{\text{ols}}|}\big(\widehat{\beta}_j^{\text{ols}}\mathbf{x}_j\big)\Big\|^2 + \lambda \sum_{j=1}^{p} \frac{|\beta_j|}{|\widehat{\beta}_j^{\text{ols}}|},$$

if we require $\widehat{\beta}_j$ to have the same sign as $\widehat{\beta}_j^{\text{ols}}$.

Adaptive Lasso

- The adaptive Lasso can be easily implemented using existing R packages, such as glmnet, if we simply get an initial estimator and rescale each covariate.
- We don't have to use $\widehat{\beta}_j^{\text{ols}}$ as the initial guess. In practice, when $p > n$, we can use the lasso estimates as the initial value.
- Adaptive Lasso Algorithm (using Lasso as initial estimator, and use $\gamma = 1$)
  1. Fit a Lasso model and obtain $\widehat{\beta}_j^{\text{lasso}}$'s
  2. Rescale covariates $\mathbf{x}_j^* = \mathbf{x}_j \cdot \widehat{\beta}_j^{\text{lasso}}$
  3. Refit the Lasso model using $\mathbf{X}^*$ without standardizing the columns and obtain $\widehat{\beta}_j^*$'s
  4. Recover the original parameter estimates $\widehat{\beta}_j^* \cdot \widehat{\beta}_j^{\text{lasso}}$ for all j.

Unbiased Penalties

- The above two approaches are two-stage approaches
- The motivation was to adaptively choose the penalty level for each of the parameter estimates
- Is there a direct approach, which is known as the pathwise approach is to let the weights change i.e., penalty function $P_\lambda(|\beta|)$, that has the oracle property?
- Fan and Li (2001) suggest three properties that a penalty function should have
  - Unbiasedness: $P'(|\beta|) = 0$ for large $|\beta|$
  - Sparsity: $P'(0+) > 0$
  - Continuity: the minimum of $|\beta| + P'(|\beta|)$ is attained at 0

Unbiased Penalties

- Interpretations of the three properties
  - Unbiasedness: The resulting estimator is nearly unbiased when the true unknown parameter is large to avoid unnecessary modeling bias.
  - Sparsity: The resulting estimator is a thresholding rule, which automatically sets small estimated coefficients to zero to reduce model complexity.
  - Continuity: The resulting estimator is continuous in data to avoid instability in model prediction.

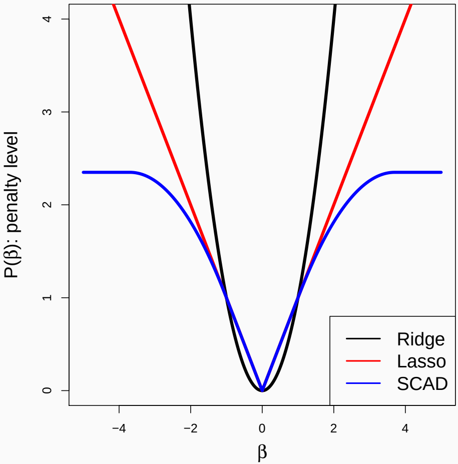
SCAD

- Smoothly clipped absolute deviation (SCAD) penalty (Fan and Li, 2001)
- Solve for the penalized loss function using the following penalty, with parameters $\lambda > 0$ and $\gamma > 2$

$$P(\beta) = \begin{cases} \lambda|\beta| & \text{if} \quad |\beta| \leq \lambda \\ \frac{2\gamma\lambda|\beta| - \beta^2 - \lambda^2}{2(\gamma - 1)} & \text{if} \quad \lambda < |\beta| < \gamma\lambda \\ \frac{\lambda^2(\gamma+1)}{2} & \text{if} \quad |\beta| \geq \gamma\lambda \end{cases}$$

- This penalty is non-convex. It coincides with the Lasso until $|\beta| = \lambda$, then smoothly transits to a quadratic function until $|\beta| = \gamma\lambda$, after which it remains constant for all $|\beta| > \gamma\lambda$.

Comparison



Comparing Lasso, Ridge and SCAD ($\gamma = 3.7$)

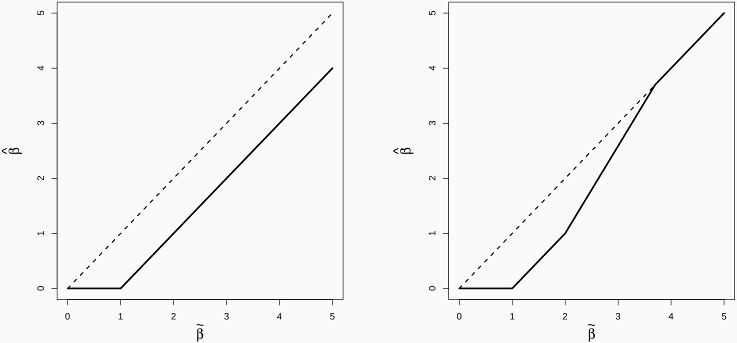
SCAD

- Again, in orthonormal case, we can get the SCAD solution from $\widehat{\boldsymbol{\beta}}^{\text{ols}}$

$$\widehat{\beta}_j^{\text{scad}} = \begin{cases} \text{sign}(\widehat{\beta}_j^{\text{ols}})\left(|\widehat{\beta}_j^{\text{ols}}| - \lambda\right)_+ & \text{if} \quad |\widehat{\beta}_j^{\text{ols}}| \leq 2\lambda \\ \frac{(\gamma-1)\widehat{\beta}_j^{\text{ols}} - \text{sign}(\widehat{\beta}_j^{\text{ols}})\gamma\lambda}{\gamma-2} & \text{if} \quad 2\lambda < |\widehat{\beta}_j^{\text{ols}}| \leq \gamma\lambda \\ \widehat{\beta}_j^{\text{ols}} & \text{if} \quad |\widehat{\beta}_j^{\text{ols}}| > \gamma\lambda \end{cases}$$

- What is the advantage? When $\widehat{\beta}_j^{\text{ols}}$ is sufficiently large, the SCAD estimator gives OLS solution.

Comparison: Lasso vs. SCAD



Comparing Lasso shrinkage with SCAD shrinkage

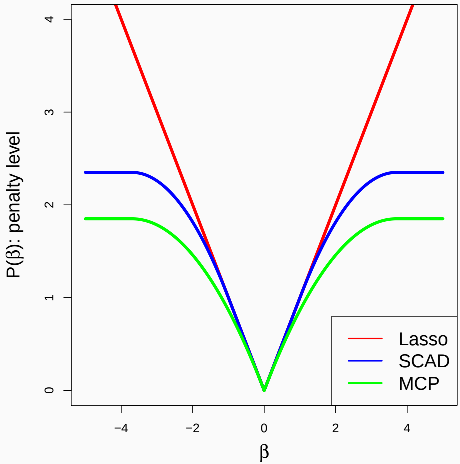
MCP

- Minimax concave penalty (MCP) is another unbiased penalty (Zhang, 2010)
- Exactly the same formulation of the penalized loss function, with a penalty term defined as

  $$P(\beta) = \begin{cases} \lambda|\beta| - \frac{\beta^2}{2\gamma} & \text{if} \quad |\beta| \leq \gamma\lambda \\ \frac{1}{2}\gamma\lambda^2 & \text{if} \quad |\beta| \geq \gamma\lambda \end{cases}$$

  for some $\gamma > 1$.
- The maximum concavity of this penalty function is $1/\gamma$, which is exactly controlled

Comparison



Comparing Lasso, SCAD ($\gamma = 3.7$), and MCP ($\gamma = 3.7$)

Derivatives of Penalties

- Let's consider the derivative at the positive side, i.e., $\beta > 0$
- The derivative of Lasso is λ for all $\beta > 0$
- The derivative of SCAD is

$$P'(\beta) = \begin{cases} \lambda & \text{if} \quad 0 < \beta \le \lambda \\ \frac{\gamma\lambda - \beta}{\gamma - 1} & \text{if} \quad \lambda < \beta < \gamma\lambda \\ 0 & \text{if} \quad \beta \ge \gamma\lambda \end{cases}$$

- The derivative of MCP is

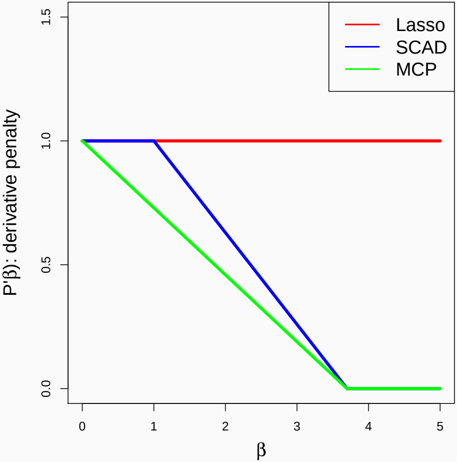
$$P'(\beta) = \begin{cases} \lambda - \frac{\beta}{\gamma} & \text{if} \quad 0 < \beta \le \gamma\lambda \\ 0 & \text{if} \quad \beta \ge \gamma\lambda \end{cases}$$

- All three penalties start out by applying the same rate of penalization λ at $0+$, which ensures sparsity

Derivatives of Penalties

- Lasso remains that rate, regardless of the size of β
- SCAD and MCP will smoothly relax the rate down to zero as the absolute value of β increases
- The difference between SCAD and MCP is that MCP relaxes the penalization rate immediately while with SCAD the rate remains flat for a while before decreasing
- Both SCAD and MCP satisfy our previously mentioned three requirements

Comparison



Comparing derivatives of Lasso, SCAD ($\gamma = 3.7$), and MCP ($\gamma = 3.7$)

Tuning Parameters

- Lasso involves one tuning parameter λ, which can be selected by cross-validation
- SCAD and MCP both have an extra tuning parameter γ
- Fan and Li (2001) recommended to use $\gamma = 3.7$ for SCAD
- Zhang (2010) used $\gamma = 2/(1 - \max_{j \neq k} |\mathbf{x}_j^\mathsf{T} \mathbf{x}_k|/n)$ in the simulation study

Penalties for Special Data Structures

Penalties for special data structures

- In many applications, the design matrix has some special structures
- We are going to discuss two cases:
  - Group Lasso: An X variable is categorical with more than 2 categories
  - Fused Lasso: X variables has a certain order and is highly correlated
  - Collaborative Lasso: Integrating data from difference sources

Group Lasso

- When a variable has multiple categories and is nominal, its effect cannot be described using just one parameter
- Suppose X_1 has three categories, and the corresponding parameters are a vector $\beta_1 = (\beta_{11}, \beta_{12}, \beta_{13})$
- If we apply a regular Lasso, the penalty term is

$$P(\beta_1) = \lambda \sum_{k=1}^{3} |\beta_{1k}|$$

- We may end up selecting a nonzero β_{13} but not the other two categories (thinking them as having the same effect)
- Another situation is when we have multiple outcomes, and we prefer the same variable to have nonzero parameters for all of them

Group Lasso

- Would it be possible to select all categories of X_1 as long as one of them is selected?
- The group Lasso penalty:

$$P(\beta_1) = \lambda \|\beta_1\|_2 = \lambda \sqrt{\beta_{11}^2 + \beta_{12}^2 + \beta_{13}^2}$$

- We apply this penalty to each group. Of course for group with size 1, it is just the Lasso.
- In general, if we have G different groups, the overall penalty is

$$P(\boldsymbol{\beta}) = \lambda \sum_{g=1}^{G} \|\boldsymbol{\beta}_{\mathcal{I}g}\|_2$$

where $\mathcal{I}g$ is the index set belonging to the g'th group.

Group Lasso

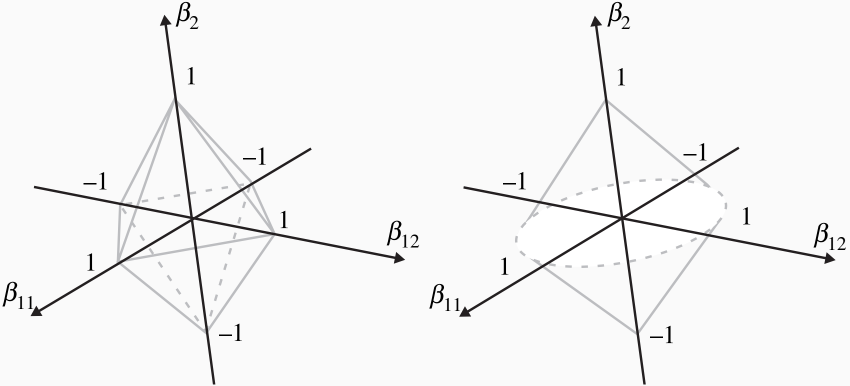
- It would be helpful to look at the derivatives:

$$\frac{\partial P(\beta_1)}{\partial \beta_{11}} = \frac{\lambda \beta_{11}}{\sqrt{\beta_{11}^2 + \beta_{12}^2 + \beta_{13}^2}}$$

  which is 0 if β_{11} is zero and any of the other two parameters are nonzero.

- This means that if one of the parameters in a group is selected, the penalty for others (if they are still 0) is 0, which allows them to start growing out of 0.

Geometry of Group Lasso



Individual Lasso vs. Group Lasso (Yuan and Lin, 2006)

Fused Lasso

- Nearby X_j's are highly correlated and they may contribute to the outcome together, meaning that if we believe X_j is related to Y, then X_{j-1} and X_{j+1} may also do. And their coefficients are likely to be the same or close.
- Recall that Lasso will have difficulty identifying all of them, since it is likely to pick one and conclude all signals.
- Fused Lasso takes advantage of this special ordering of X.

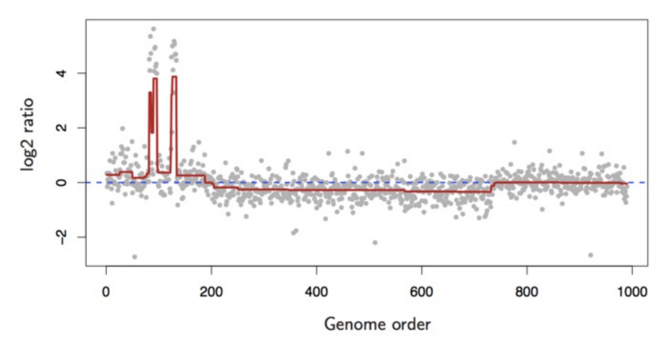
Fused Lasso

- Fused Lasso solves the optimization problem:

$$\arg\min_{\boldsymbol{\beta}} \frac{1}{2n}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|$$

- Encouraging nearly parameter estimates to be the same

Fused Lasso



Fused Lasso (Tibshirani et al. 2005) fitting

Collaborative Regression

- In many biomedical and biological studies, we may have information (covaraites) measured from different sources of collection, or different technology
- An integrative analysis tries to model an outcome using information from all sources, however, how to properly contribute the effect to different sources is the challenge.
- For example, since the signal from different sources can be highly correlated, including one source may push the signal from the other out.

Collaborative Regression

- One idea is to set up connections between different sources ("Collaborative Regression", by Gross and Tibshirani (2015)).
- For example, if X represents DNA methylation, while Z represents RNA expressions, and Y is cancer related measure, then we can have

$$\|\mathbf{y} - \mathbf{X}\beta\|^2 + \|\mathbf{y} - \mathbf{Z}\boldsymbol{\theta}\|^2 + \|\mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\theta}\|^2$$

- We can further add penalties to both $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ to force sparsity.

Collaborative Regression

- This problem can be easily solved using an augmentation of the design matrix:

$$\begin{pmatrix} \mathbf{X} & \mathbf{0} \\ \mathbf{0} & \mathbf{Z} \\ \mathbf{X} & -\mathbf{Z} \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ \boldsymbol{\theta} \end{pmatrix} \sim \begin{pmatrix} \mathbf{y} \\ \mathbf{y} \\ \mathbf{0} \end{pmatrix}$$

- Hence, this can be treated as a new “mega” design matrix and outcome vector, and solve for the solution of the parameters.

Concluding Remarks

- Penalization and variable selection
- Research direction in recent years
  - Debias of the Lasso estimator
  - Hypothesis testing and confidence intervals
  - Account for complicated data structures