Linear Methods and Regularization

PS690 Computational Methods in Social Science

Jiawei Fu
Department of Political Science
Duke University

September 2, 2025

Overview

Linear Model in Low Dimension

- Given i.i.d. sample $D_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$
- Consider linear model $Y_i = X_i^T \beta_0 + \epsilon_i \Leftrightarrow Y = X\beta + \epsilon$,
  where $Y = (Y_1, Y_2, ..., Y_n) \in \mathbb{R}^n$ is the vector of responses, $X \in \mathbb{R}^{n \times p}$ is the matrix of predictor variables.
- The OLS estimator $\hat{\beta} = \arg min \sum_{i=1}^{n} (Y_i - X_i^T \beta)^2 = (X'X)^{-1} X'Y$
- We calculate the In-sample risk: $\mathbb{E}[\frac{1}{n} ||X\hat{\beta} - X\beta||^2]$
- We assume X and ϵ are independent, and $\epsilon \sim N(0, \sigma^2 I)$
- Then, in-sample risk is $\sigma^2 \frac{p}{n}$.
- In the low dimension case, $p \ll n$, the in-sample risk is negligible.

Linear Model in High Dimension

- What happened in high dimensions?
- First, when p is moderate , in-sample risk $\sigma^2 \frac{p}{n}$ is poor.
- Second, when $p > n$, $X'X$ is not invertible, and there are infinitely many solutions.
- Third, because parameter is larger than sample point, the model can fit training data perfectly through interpolation.
- This implies training error is zero but generalization error is large.
- We hope to restrict the complexity of the function.
- Solution: regularization. (In deep learning, the method called weight decay).
- We will see that regularized model has lower variance.

Ridge Regression

- Ridge regression shrinks the regression coefficients by imposing a l_2 penalty on their size.

$$\hat{\beta}^{ridge} = \arg\, min \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

$$s.t. \sum_{j=1}^{p} \beta_j^2 \leq t \quad (||\beta||_2^2 \leq t)$$

- By Lagrange, an equivalent way to write is

$$\hat{\beta}^{ridge} = \arg\, min \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

- λ and t, are tuning parameters that controls the amount of shrinkage.
- As λ increases, ridge estimates shrinks to 0. As λ decreases, it becomes OLS estimates.

Ridge

- Note: The ridge solutions are not equivariant under scaling of the inputs, and so one normally standardizes the inputs first.
- Write the objective function in the matrix form,

$$RSS(\gamma) = (\mathbf{y} - \mathbf{X}\beta)^T(\mathbf{y} - \mathbf{X}\beta) + \lambda\beta^T\beta$$

- Given strictly convex (for any $\lambda > 0$), the *unique* solution can be easily derived:

$$\hat{\beta}^{ridge} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$$

- The solution adds a positive constant to the diagonal of $\mathbf{X}^T\mathbf{X}$ before inversion. This makes the problem nonsingular (b/c the smallest eigenvalue is at least $\lambda > 0$), even if $\mathbf{X}^T\mathbf{X}$ is not of full rank.
- If inputs are orthonormal ($X'X = I$), the ridge estimates are just a scaled version of the least squares estimates: $\hat{\beta}^{ridge} = \frac{\hat{\beta}^{ols}}{1+\lambda}$.
- It is biased but has lower variance compared to OLS estimator.

LASSO

- LASSO shrinks the regression coefficients by imposing a l_1 penalty on their size.

$$\hat{\beta}^{lasso} = \arg\, min \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2$$

$$s.t. \sum_{j=1}^{p} |\beta_j| \leq t \quad (||\beta||_1 \leq t)$$

- By Lagrange, an equivalent way to write is

$$\hat{\beta}^{lasso} = \arg\, min \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- λ is a tuning parameter that controls the amount of shrinkage.

Sparsity of LASSO Aolution

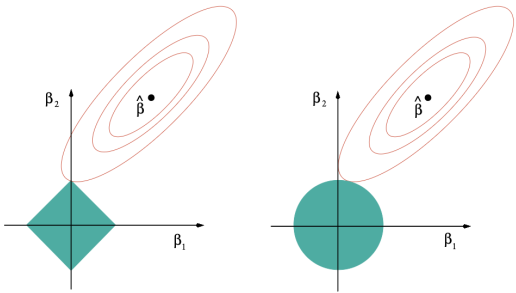


Figure: Ellipses are contours of objective function, centered at the ols estimates. Green parts denote constraints.

Therefore, lasso yields sparse solution vectors, having only some coordinates that are nonzero.

Coefficients Path

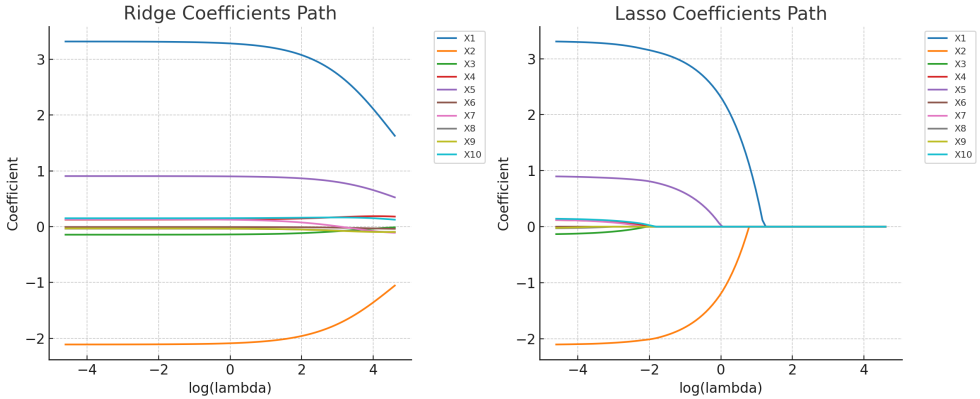


Figure: Coefficients Path for Ridge and LASSO

Solution

- There is always a unique fitted value $X\hat{\beta}$, because the squared error loss is strictly convex in $X\beta$.
- However, it not necessary for $\hat{\beta}$ when $X^T X$ is singular; for example, when $p > n$.
- Apply KKT condition, we solve the LASSO solution:

$$X^T(Y - X\hat{\beta}) = \lambda s$$

where s is a subgradient of the l_1 norm evaluated at $\hat{\beta}$,

$$s_j \in \begin{cases} +1 & \hat{\beta}_j > 0 \\ -1 & \hat{\beta}_j < 0 \\ [-1, 1] & \hat{\beta}_j = 0 \end{cases}$$

- Because $X\hat{\beta}$ is unique, the optimal subgradient s is always unique.
- It tells us that even in the case when lasso solutions are nonunique, any two solutions must always agree on the signs of common nonzero coefficients. In this sense, the lasso is already much better behaved than least squares when $d > n$.

Solution

- Let $A = supp(\hat{\beta})$ be set of active set.
- It is easy to see that

$$\hat{\beta}_A = (X_A^T X_A)^{-1}(X_A^T y - \lambda s_A)$$
$$\hat{\beta}_{-A} = 0$$

- We see that lasso shrink the OLS estimate by an amount $\lambda(X_A^T X_A)^{-1} s_A$.

LASSO

- People often minimize the average squared error:

$$\hat{\beta}^{lasso} = \arg min \frac{1}{2N} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

- This makes λ independent of sample size.
- The solution becomes $\hat{\beta}_A = (X_A^T X_A)^{-1}(X_A^T y - n\lambda s_A)$

LASSO

- Typically, We first standardize the predictors (mean zero, variance one) if covariates have difference units.
- How to find λ? Answer: Cross-validation.
- Split the data into training and validation data.
- For each λ, use training set to fit the model and predict values in the validation set.
- Compute RMSE and there are two options of λ
  1. λ_{min}: the λ gives the lowest CV error; low bias, but possibly high variance (more complex model, more predictors)
  2. λ_{1se}: the largest λ such that the CV error is within one standard error of the minimum; slightly more biased, but lower variance, better generalization.

Cross-validation

- We first randomly divide the full dataset into some number of groups $K > 1$. Typical choices of K might be 5 or 10, and sometimes N.
- We fix one group as the validation set, and designate the remaining K-1 groups as the training set.
- We then apply the lasso to the training data for a range of different λ values, and we use each fitted model to predict the responses in the validation set, recording the mean-squared prediction errors for each value of λ.
- This process is repeated a total of K times, with each of the K groups getting the chance to play the role of the validation data, with the remaining K-1 groups used as training data.
- These K estimates of prediction error are averaged for each value of λ, thereby producing a cross-validation error curve.

Cross-validation

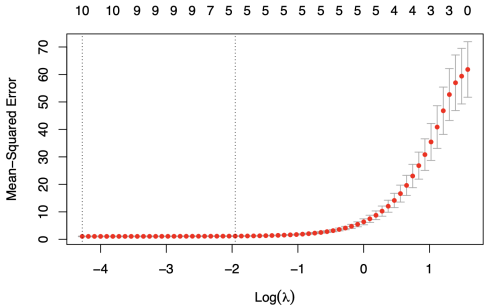


Figure: From [Hastie et al., 2015]

Theoretical Results for the Lasso

- It is useful to go through one formal result and proof of lasso to have a general picture what the result looks like in ML.
- We want to ask how close $X\hat{\beta}$ and $X\beta_0$.
- In other words, how $\frac{1}{n}||X\hat{\beta} - X\beta_0||$ behaves?
- In most cases, we are looking for the upper bound, with some probability statement.
- We will show that, for any $\delta > 0$,

$$\frac{1}{n}||X\hat{\beta} - X\beta_0||_2^2 \leq 4\sigma||\beta_0||_1\sqrt{\frac{2(\log ep/\delta)}{n}}$$

  with probability at least $1 - \delta$.
- It is known by "slow rate" result, which is on the order of $\sqrt{\log p/n}$. Recall for linear regression model, the in-sample risk is $\frac{\sigma^2 p}{n}$. This slow rate creates some problems; we will see it in Double ML.

Slow rate

- Start from what we have: the objective function.
- Because $\hat{\beta}$ us the optimal solution, we must have

$$\frac{1}{2}||Y - X\hat{\beta}||_2^2 + \lambda||\hat{\beta}||_1 \leq \frac{1}{2}||Y - X\beta_0||_2^2 + \lambda||\beta_0||_1$$

- Organize terms so that close to what we want:

$$\begin{aligned} ||X\hat{\beta} - X\beta_0||_2^2 &\leq 2 < Y - X\beta_0, X\hat{\beta} - X\beta_0 > +2\lambda(||\beta_0||_1 - ||\hat{\beta}||_1) \\ &= 2 < \epsilon, X\hat{\beta} - X\beta_0 > +2\lambda(||\beta_0||_1 - ||\hat{\beta}||_1) \end{aligned}$$

- $< a, b >= a^T b$
- $< \epsilon, X\hat{\beta} - X\beta_0 > \leq ||X^T\epsilon||_\infty ||\hat{\beta} - \beta_0||_1$ by Holder's inequality.

Slow rate

- Therefore, we have

$$||X\hat{\beta} - X\beta_0||_2^2 \leq 2||X^T\epsilon||_\infty||\hat{\beta} - \beta_0||_1 + 2\lambda(||\beta_0||_1 - ||\hat{\beta}||_1)$$

- How can we further simplify terms?
- By triangle inequality,

$$||X\hat{\beta} - X\beta_0||_2^2 \leq 2||X^T\epsilon||_\infty(||\hat{\beta}||_1 + ||\beta_0||_1) + 2\lambda(||\beta_0||_1 - ||\hat{\beta}||_1)$$

Slow rate

- ϵ is random.
- Assume $\max_j ||X_j||_2 \leq \sqrt{n}$(Note that we can always rescale to make this true.).
- Note that $X^T\epsilon$ is normal with mean zero and variance upper bound by $n\sigma^2$.
- We typically calculate the tail probability: for any $\delta > 0$,

$$\mathbb{P}(||X^T\epsilon||_\infty \geq \sigma\sqrt{2n(\log ep/\delta)} \geq 1 - \delta$$

- Therefore, taking $\lambda \geq 2\sigma\sqrt{2n(\log ep/\delta)}$, we get

$$||X\hat{\beta} - X\beta_0||_2^2 \leq 2\lambda(||\hat{\beta}||_1 + ||\beta_0||_1) + 2\lambda(||\beta_0||_1 - ||\hat{\beta}||_1) \leq 4\lambda||\beta_0||_1$$

  with probability at least $1 - \delta$.
- Note, for some problems (we will see in the double ML), λ should be carefully chosen so that lasso estimators have desirable performance.

Statistical Inference of LASSO

- It is very difficult to characterize the distribution of the LASSO estimator.
- Because of penalty, some coefficients are exactly shrunk to zero with positive probability, even asymptotically.
- The asymptotic distribution is irregular and depends on tuning parameters and the true sparsity pattern.
- Moreover, it is post-selection inference, where the model is "found" by a data-driven selection process; however, Classical statistical theory assumes the fixed model.
- We will cover it in the next lecture.

Variable Selection

- Thus far, we have focused on bounds on either the l_2-error or the prediction error associated with a lasso solution.
- Now we are asking whether LASSO recover the true sparsity pattern: $\hat{\beta}$ has nonzero entris in the same position as the true β^*.
- We refer to this property as variable selection consistency or sparsistency.
- Note that it is possible for the l_2 error $||\hat{\beta} - \beta^*||_2$ to be quite small even if $||\hat{\beta}||$ and β^* have different supports, as long as $\hat{\beta}$ is nonzero for all "suitably large" entries of β^*, and not "too large" in positions where β^* is zero.

Variable Selection

- We first state some conditions.

Condition (Irrepresentability or Mutual Incoherence)

There must exist some $\gamma > 0$ *such that*

$$\max_{j \in S^c} ||(X_S^T X_S)^{-1} X_S^T x_j||_1 \leq 1 - \gamma$$

- In the most desirable case, the columns $\{x_j, j \in S^c\}$would all be orthogonal to the columns of X_S, and we would be guaranteed that $\gamma = 1$.
- In the high-dimensional setting, this complete orthogonality is not possible, but we can still hope for a type of "near orthogonality" to hold.

Variable Selection

Condition (Eigenvalue condition)

Submatrix X_S is well-behaved in the sense that

$$\lambda_{min}(X_S^T X_S/N) \geq C_{min} > 0$$

- The smallest eigenvalue measures collinearity.
- If this condition were violated, then the columns of X_S would be linearly dependent, and it would be impossible to estimate β^* even in the "oracle case" when the support set S were known.

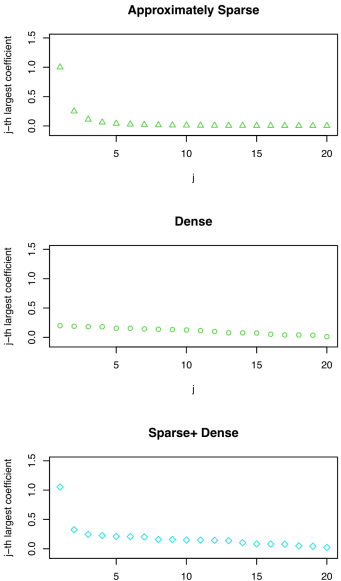
Variable Selection

Theorem

Suppose that the design matrix X satisfies the mutual condition with parameter $\gamma > 0$, and the columns are normalized, and the eigenvalue condition both hold. For a noise vector w with i.i.d. $N(0, \sigma^2)$ entries, consider the regularized lasso program with $\lambda_N \geq \frac{8\sigma}{\lambda}\sqrt{\frac{\log p}{N}}$. Then with probability greater than $1 - c_1 e^{-c_2 N \lambda_N^2}$, the lasso has the following properties

1. *No false inclusion: The unique optimal solution has its support $S(\hat{\beta})$ contained within the true support $S(\beta^*)$.*
2. *No false exclusion: The lasso solution includes all indices $j \in S(\beta^*)$ such that $|\beta_j^*| > \beta_{min}$, and hence is variable selection consistent as long as $\min_{j \in S} |\beta^*| > \beta_{min}$, where $\beta_{min} = \lambda_N[\frac{4\sigma}{\sqrt{C_{min}}} + ||(X_S^T X_S/N)^{-1}||_\infty]$.*

Data Type


Approximately Sparse
j−th largest coefficient
0.0
0.5
1.0
1.5
5
10
15
20
j
Dense
j−th largest coefficient
0.0
0.5
1.0
1.5
5
10
15
20
j
Sparse+ Dense
j−th largest coefficient
0.0
0.5
1.0
1.5
5
10
15
20

Generalized LASSO

- LASSO performs best in an approximately sparse setting.
- Ridge penalizes the large values of coefficients much more aggressively and small values much less aggressively; and thus is well suited for dense models.
- In general, we can consider any l_p penalty. For example, l_0: $||\beta||_0 = \sum_{j=1}^{d} 1\beta_j \neq 0$.
- This is most intuitive to select variables. However, it is not convex. Therefore, it is generally very hard to solve in practice except for very small d.
- Similar, we can choose different q:
  $\hat{\beta}^{ridge} = \arg min \sum_{i=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^q$

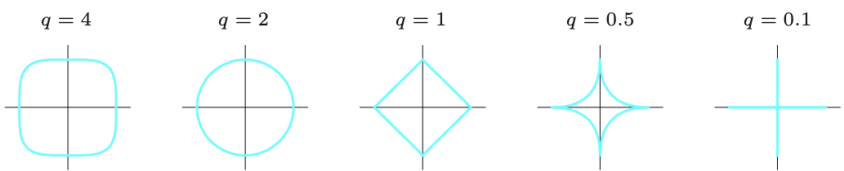


Figure: From [Hastie et al., 2009]

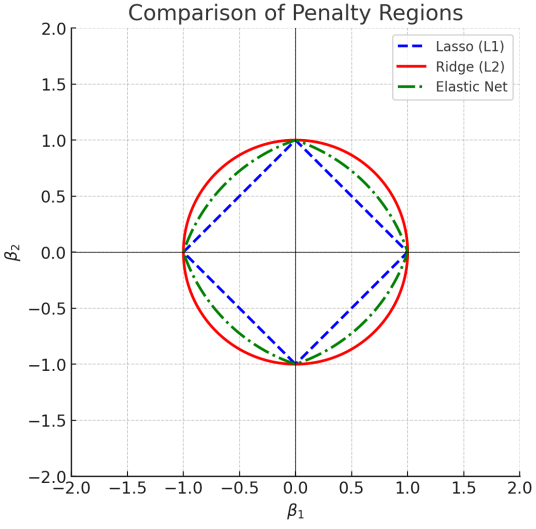
Elastic Net

- One step further, we can combine the advantage of both ridge (handle nonlinearity) and lasso (handle sparsity).

$$\arg min \frac{1}{N} \sum_{i=1}^{N} ||Y - X\beta||^2 + \lambda_2 ||\beta||^2 + \lambda_1 ||\beta|_1$$

- This is called elastic net regularization. It is invented to solve the high variability of lasso when covariates are highly correlated.
- The penalty can also be written as $\lambda \sum_{i=1}^{p} (\alpha \beta_j^2 + (1 - \alpha |\beta_j|))$
- Elastic Net can perform well in either the sparse or the dense scenario with appropriate tuning.

Elastic Net


Comparison of Penalty Regions
Lasso (L1)
Ridge (L2)
Elastic Net
β_1
β_2
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0

Lava

- Lava method is intended work well in sparse+dense setting: there are several large coefficients and many small coefficients that do not vanish quickly enough to satisfy approximately sparsity.
- It solves arg $min_{\beta:\beta=\delta+\xi} \frac{1}{N} \sum_{i=1}^{N} ||Y - X\beta||^2 + \lambda_2|\delta||^2 + \lambda_1||\xi|_1$
- Compared to the Elastic Net, the Lava method penalizes large and small coefficients much less aggressively – large coefficients are penalized like Lasso and small coefficients like Ridge.
- Like Ridge, Lava does not do variable selection.

References


Hastie, T., Tibshirani, R., Friedman, J., et al. (2009).
The elements of statistical learning.


Hastie, T., Tibshirani, R., and Wainwright, M. (2015).
Statistical learning with sparsity.
Monographs on statistics and applied probability, 143(143):8.