



Statistical Learning

Lecture 3: Shrinkage and Dimension Reduction

Yi He

January 16, 2023

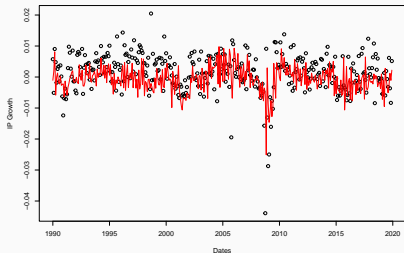
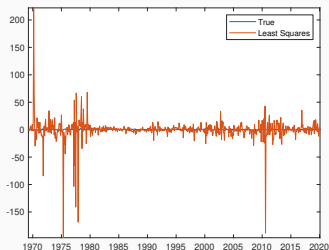
Plan for Today

1. Introduction
2. LASSO Regression
3. Ridge Regression
4. LASSO VS Ridge Regression
5. Principal Component Regression

Introduction

FRED-MD Data Set

- Response = monthly growth rate US industrial production index $\log(IP_t/IP_{t-1})$
- Features = one-month lagged observations of $p \approx 100$ economic variables from FRED-MD database
- Rolling-window forecasts with size $n = 120$



Lecture 2: Selecting $d = 9$ variables improves prediction performance

Revisiting Best Subset Selection

- Target y_i , features $x_i = (x_{i,1}, \dots, x_{i,p})^T$, $i = 1, \dots, n$
- β_0 intercept, $\beta = (\beta_1, \dots, \beta_p)^T$ linear regression coefficients
- Finding the best subset with d predictors by minimizing mean squared error

$$\underset{\beta_0, \beta}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^n \left(y_i - \beta_0 - x_i^T \beta \right)^2$$

under the complexity constraint that

$$\sum_{j=1}^p \mathbb{1}(\beta_j \neq 0) = \sum_{j=1}^p \mathbb{1}(\text{sign}(\beta_j) \neq 0) \leq d.$$

Shall we take the magnitude of β_j into account?

LASSO Regression

L_1 Norm Constraint

- Target y_i , $i = 1, \dots, n$
- **Standardized** features $x_i = (x_{i,1}, \dots, x_{i,p})^T$

Minimize the mean squared error

$$\underset{\beta_0, \beta}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^n \left(y_i - \beta_0 - x_i^T \beta \right)^2$$

... but under a different complexity constraint that

$$\underbrace{\sum_{j=1}^p |\beta_j|}_{\substack{L_1 \text{ norm of} \\ \beta = (\beta_1, \dots, \beta_p)^T}} \leq b, \text{ where } b > 0 \text{ is some given "budget".}$$

- We do not 'weight' coefficients thanks to the standardization
- The subdomain becomes *convex**: can be solved efficiently

LASSO Regression

Tute Q5: For every given $b > 0$, by using Lagrange multiplier method, we can reformulate the constrained optimization problem into a unconstrained optimization problem

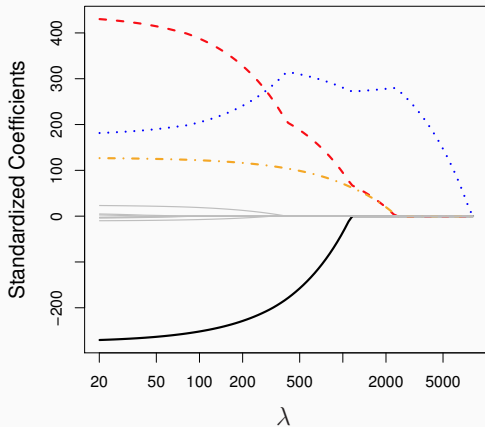
$$\underset{\beta_0, \beta}{\text{minimize}} \left\{ \frac{1}{n} \sum_{i=1}^n \left(y_i - \beta_0 - \mathbf{x}_i^T \beta \right)^2 + \underbrace{2\lambda \sum_{j=1}^p |\beta_j|}_{\text{penalty for complexity}} \right\},$$

where

- $\lambda = \lambda(b) \geq 0$ is some hyperparameter that depends on b
- The factor 2 is included for convention
- From now on we choose $\lambda > 0$ instead of $b > 0$.
- No closed-form but numerical solution can be effectively found, even simultaneously for many different λ 's.

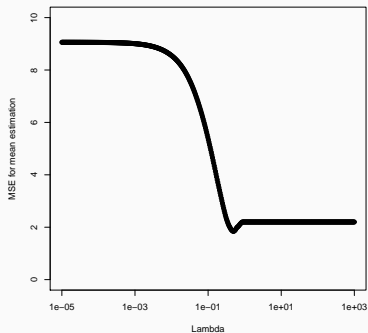
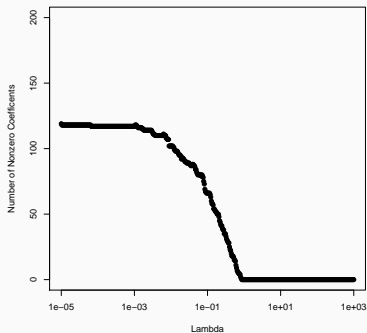
Example: Credit Data Set

- LASSO performs model selection: *sparse* learning
- + **Shrinkage** at the same time:



Warning: Sparsity as Illusion?

- Simulation example with $n = 100$, $p = 200$
- True β_j uniformly generated between -0.2 and 0.2: no zeros
- Best fitted model selected 18 variables



Ridge Regression

Quadratic Constraint

- Target y_i , $i = 1, \dots, n$
- **Standardized** features $x_i = (x_{i,1}, \dots, x_{i,p})^T$

What if we minimize the mean squared error

$$\underset{\beta_0, \beta}{\text{minimize}} \quad \frac{1}{n} \sum_{i=1}^n \left(y_i - \beta_0 - x_i^T \beta \right)^2$$

... now under a quadratic constraint that

$$\sum_{j=1}^p \beta_j^2 \leq b, \text{ where } b > 0 \text{ is some given "budget" ?}$$

Ridge Regression

Like for LASSO: for every given $b > 0$, by using Lagrange method, we can again reformulate the constrained optimization problem into a unconstrained optimization problem

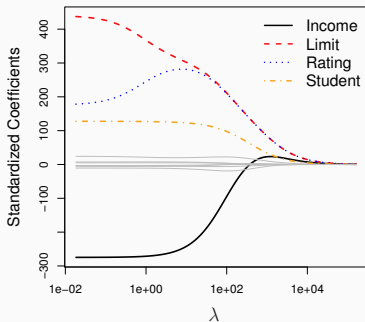
$$\underset{\beta_0, \beta}{\text{minimize}} \left\{ \frac{1}{n} \sum_{i=1}^n \left(y_i - \beta_0 - \sum_{j=1}^p x_{i,j} \beta_j \right)^2 + \underbrace{\lambda \sum_{j=1}^p \beta_j^2}_{\text{penalty for complexity}} \right\},$$

where

- $\lambda = \lambda(b) \geq 0$ is some hyper-parameter depending on b .
- From now on we choose $\lambda > 0$ rather than $b > 0$.
- Tute Q3: We can derive a closed-form solution

Credit Data Set: Ridge Regression

- Target: Balance
- features: income, limit, rating, student,...
- Shrinking towards zero as $\lambda \uparrow$ in general: variance \downarrow but bias \uparrow
- All variables are used, suitable for dense forecasting

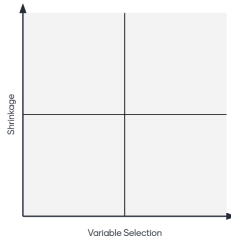


ISLR Figure 6.4,
left part

Go to www.menti.com and use the code 81 41 82 8

 Mentimeter

Does the method perform variable selection or/and shrinkage? 0=No, 1=Yes



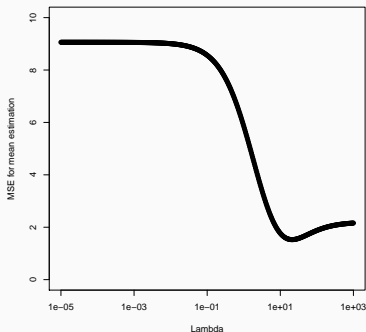
- 1 Forward Regression
- 2 LASSO Regression
- 3 Ridge Regression



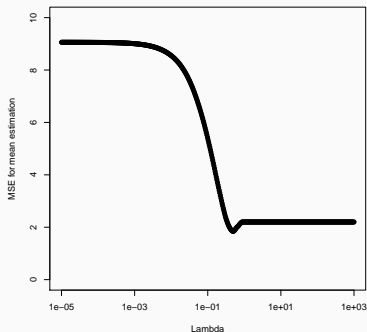
LASSO VS Ridge Regression

Ridge is Better for Dense Models

- Consider the dense model on page 7 with $n = 100$, $p = 200$
- True β_j uniformly generated between -0.2 and 0.2: no zeros
- MSE of mean estimation: best ridge/best lasso = 82.84%



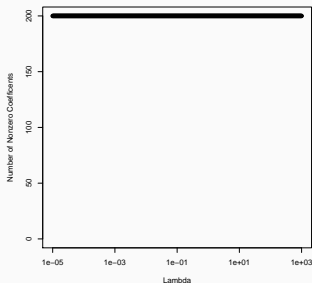
(a) Ridge



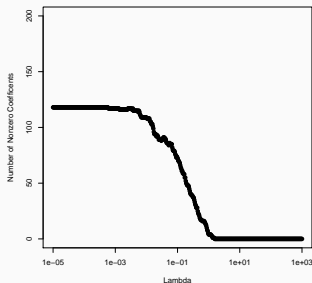
(b) LASSO

LASSO is Better for Sparse Models

- Simulated data from a **sparse** model with $n = 100$, $p = 200$
- The population model uses only $d = 10$ variables.
- All ridge estimates are non-zeros: no model selection
- MSE of regression function: best ridge/best lasso = 117.27%

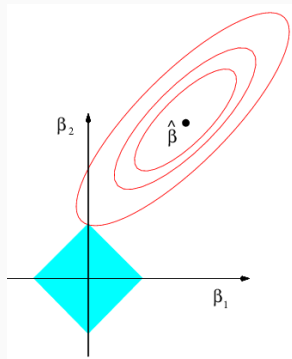


(c) Ridge



(d) LASSO

Why is LASSO Solution Sparse?



ISLR Figure 6.7, LASSO

Consider the case $p = 2$ and omit the intercept for simplicity.

Red = contour of MSE

$$\frac{1}{n} \sum_{i=1}^n (y_i - x_{i,1}\beta_1 - x_{i,2}\beta_2)^2$$

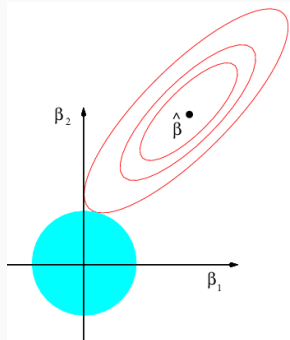
Blue = the diamond

$$|\beta_1| + |\beta_2| \leq b$$

Solution is a corner point of the diamond

- Good performance if the *true* model is sparse (or *approximately so*), and if predictors are not highly correlated

Why is Ridge Solution Dense?



ISLR Figure 6.7, Ridge

Consider the case $p = 2$ and omit the intercept for simplicity.

Red = contour of MSE

$$\frac{1}{n} \sum_{i=1}^n (y_i - x_{i,1}\beta_1 - x_{i,2}\beta_2)^2$$

Blue = the ball

$$\beta_1^2 + \beta_2^2 \leq b$$

Tangent point on the sphere

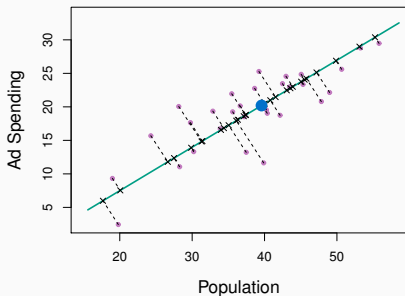
- No corner solution, usually more appropriate for dense models
- Bias variance trade-off via shrinkage
- Applicable in high dimensions with $p > n$

Principal Component Regression

Data Projection

- Let $\phi = (\phi_1, \dots, \phi_p)^T$ be a given unit vector with $\|\phi\| = 1$.
- The projection of features $x_i = (x_{i,1}, \dots, x_{i,p})^T$ for i -th observation is given by $z_i\phi$ such that $\|x_i - z_i\phi\|^2$ is minimal.
- The solution is given by

$$z_i = \phi_1 x_{i,1} + \dots + \phi_p x_{i,p} = \phi^T x_i \in \mathbb{R}$$



ISLR Figure 6.15 (left)
 $\{z_i : i = 1, \dots, n\}$ is called a
component

Dimension Reduction

- The information from predictors may be best summarized by a few components in *orthogonal* directions.
- For forward selection and LASSO: the projections are onto coordinate basis and therefore we directly select variables.
- Now select the components (instead of variables) that summarize most of the *sample* information: we use *sample* variance to quantify the amount of sample information
- *First* principal component: component with largest *sample* variance, i.e., by solving

$$u_1 = \operatorname{argmax}_{\phi: \|\phi\|=1} \frac{1}{n} \sum_{i=1}^n (\phi^T (x_i - \bar{x}))^2$$

- Tute Q4: u_1 = first eigenvector of sample covariance matrix

Principal Components

Repeat the procedures we can extract more PCs:

- 2nd PC: maximizes sample variance again over the directions *orthogonal* to that of 1st PC

$$u_2 = \operatorname{argmax}_{\|\phi\|=1} \frac{1}{n} \sum_{i=1}^n (\phi^T (x_i - \bar{x}))^2$$

such that the scores $\phi^T (x_i - \bar{x})$ are *uncorrelated* with $u_1^T (x_i - \bar{x})$ in sample

- 3rd PC: maximizes sample variance again over the directions *orthogonal* to that of 1st and 2nd PCs
- ... up to p principal components

Principal Component Regression: Remarks

For training data:

- Demean and **standardize** the features
- Construct the m -th principal component

$$z_{i,m} = u_m^T x_i = \sum_{j=1}^p u_{m,j} x_{i,j}$$

for $m = 1, \dots, M$ where M is small.

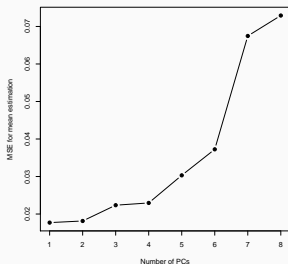
- Ordinary least squares estimation, using the PCs as features.
- The dimension of features is reduced to M

When making predictions:

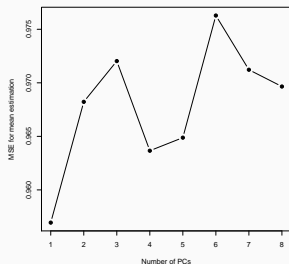
- transform the new observation of features using
- ... the **training** mean and **training** standard deviation
- use the component weights from **training** data
- Use these new PC scores and the OLS coefficients

Simulation Example: Factor Model

- PCR is particularly useful when the data are highly correlated and the overlapping information are important to the target
- For example, consider the factor model
- $x_{i,j} = \theta_k f_i + e_{i,k}$, $f_i \sim N(0, 1)$, $e_{i,k} \sim N(0, 0.01)$, $\theta_k \in [-0.2, 0.2]$
- $y_i = 0.1 f_i + \varepsilon_i$, $\varepsilon_i \sim N(0, 1)$



(a) True Means(Oracle)



(b) 5 fold CV

FRED-MD: Correlations Between Predictions

	Forward	Ridge	LASSO	PCR
Forward	1.0000000	0.8214825	0.9250341	0.5726788
Ridge	0.8214825	1.0000000	0.9067102	0.8396510
LASSO	0.9250341	0.9067102	1.0000000	0.6995926
PCR	0.5726788	0.8396510	0.6995926	1.0000000