



Modeling VIII

Introduction



- Big Data
 - Large Sample Size
 - Large Number of Variables
 - Traditional Methods are Difficult to Implement
 - Depends on the Available Technology
- Goal: Explore Approaches for Quick Filtering of Predictors
- Tutorial 15
 - Download Rmd
 - Install Package `> library(glmnet)`
 - Knit the Document
 - Read the Introduction

Introduction



My Data
is Bigger than
Your Data

Linear Model



- Consider the Following:
$$y_i = \beta_0 + X_{1i}\beta_1 + \dots + X_{pi}\beta_p + \epsilon_i$$
where $i = 1, 2, 3, \dots, n$

- Matrix Representation

$$\mathbf{y} = \beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

$$\text{where } \mathbf{y} = [y_1, y_2, \dots, y_n]',$$

$$\boldsymbol{\beta} = [\beta_1, \beta_2, \dots, \beta_p]',$$

$$\boldsymbol{\epsilon} = [\epsilon_1, \epsilon_2, \dots, \epsilon_n]',$$

and

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{21} & \dots & X_{p1} \\ X_{12} & X_{22} & \dots & X_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ X_{1n} & X_{2n} & \dots & X_{pn} \end{bmatrix}$$

Linear Model



- Information About Model Matrix

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{21} & \cdots & X_{p1} \\ X_{12} & X_{22} & \cdots & X_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ X_{1n} & X_{2n} & \cdots & X_{pn} \end{bmatrix}$$

This Matrix Should Be Standardized

- Once Standardized, The Intercept β_0 is Unnecessary in the Model
- For Interpretability, the Response Vector \mathbf{y} Can Also Be Standardized

Part 1: Simulate and Meditate



- Run Chunk 1
 - Simulating Response From a Linear Model
 - All Predictor Variables in X are Standardized `> rnorm()`
 - What is n ?
 - What is p ?
 - What do We Know About the True Signal We Want to Detect?

Sparse

Part 1: Simulate and Meditate

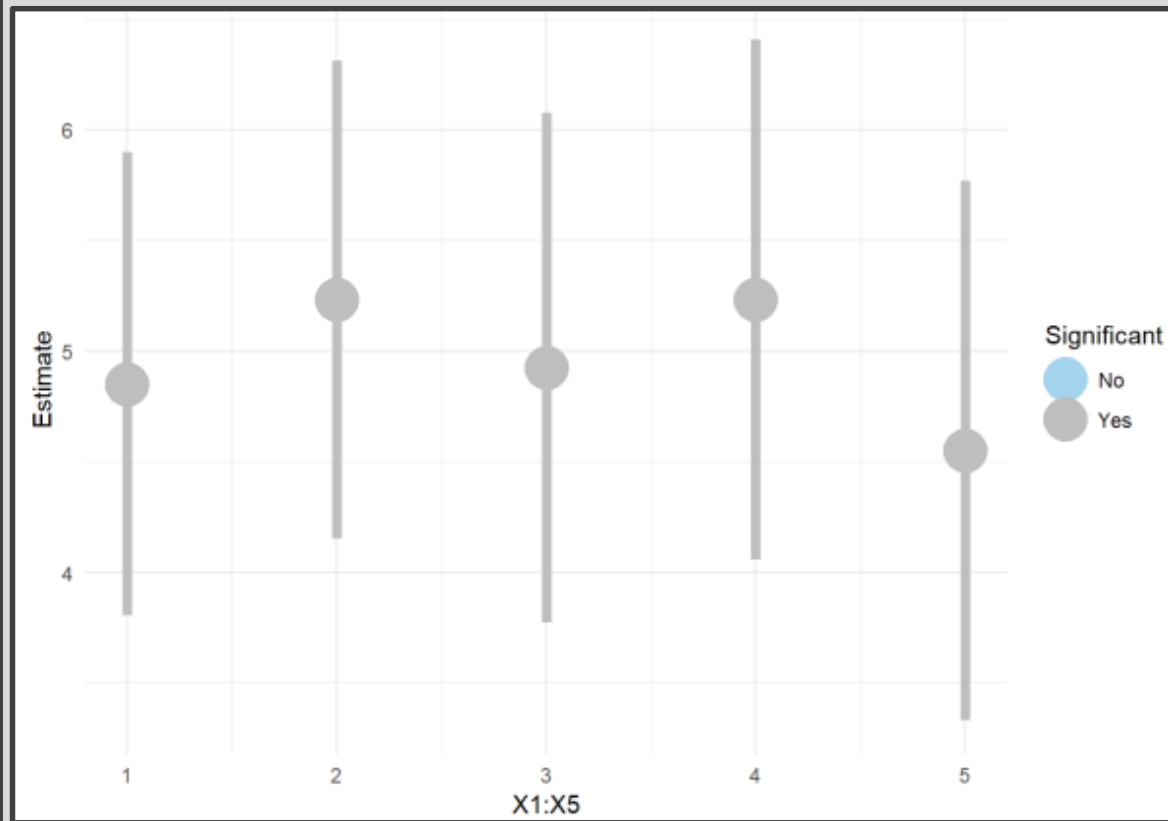


- Run Chunk 2
 - Fitting Naïve Linear Model
 - Obtaining Confidence Intervals for Parameters `> confint(lm.model)`
 - Figure Info
 - Show the Estimated Coefficients of Linear Model
 - Show Confidence Intervals for These Coefficients
 - What Does the Color Aesthetic Being Used For?

Part 1: Simulate and Meditate



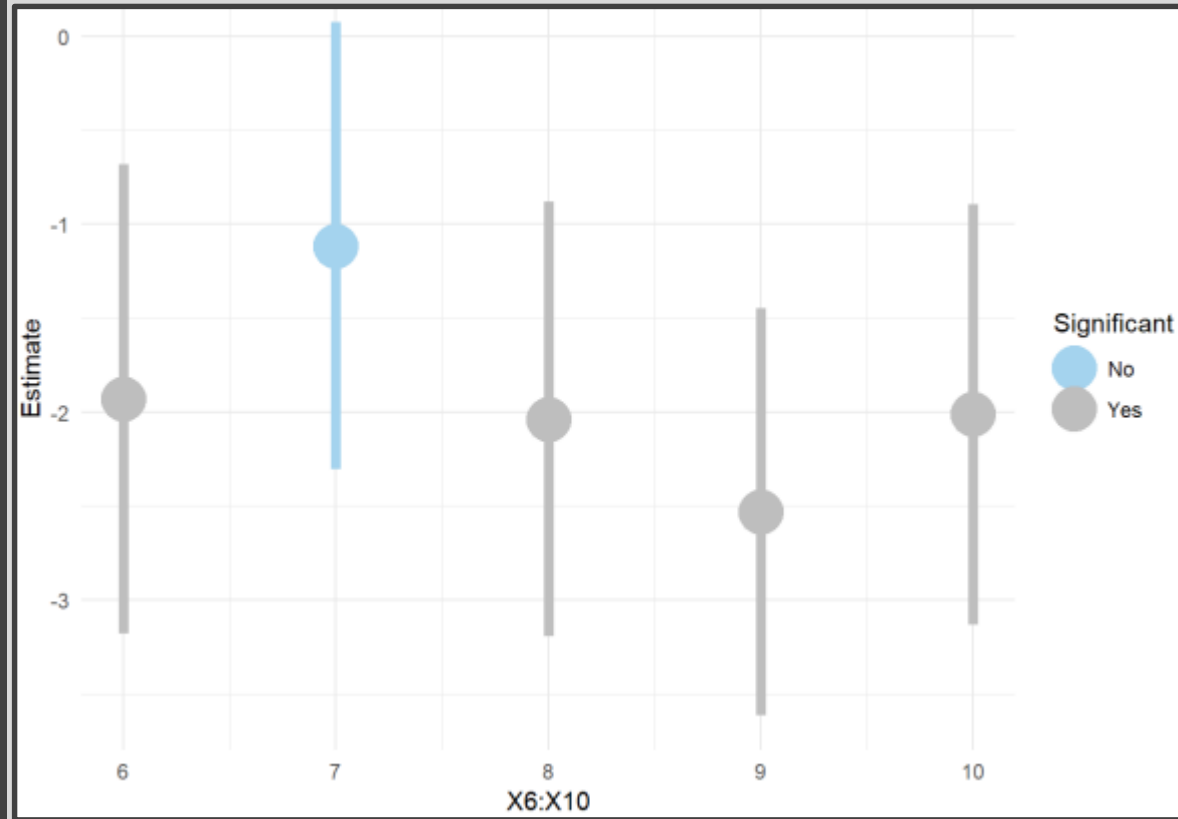
- Chunk 2 (Continued)
 - Knit the Document and Observe the 3 Graphics
 - Figure 1



Part 1: Simulate and Meditate



- Chunk 2 (Continued)
 - Figure 2

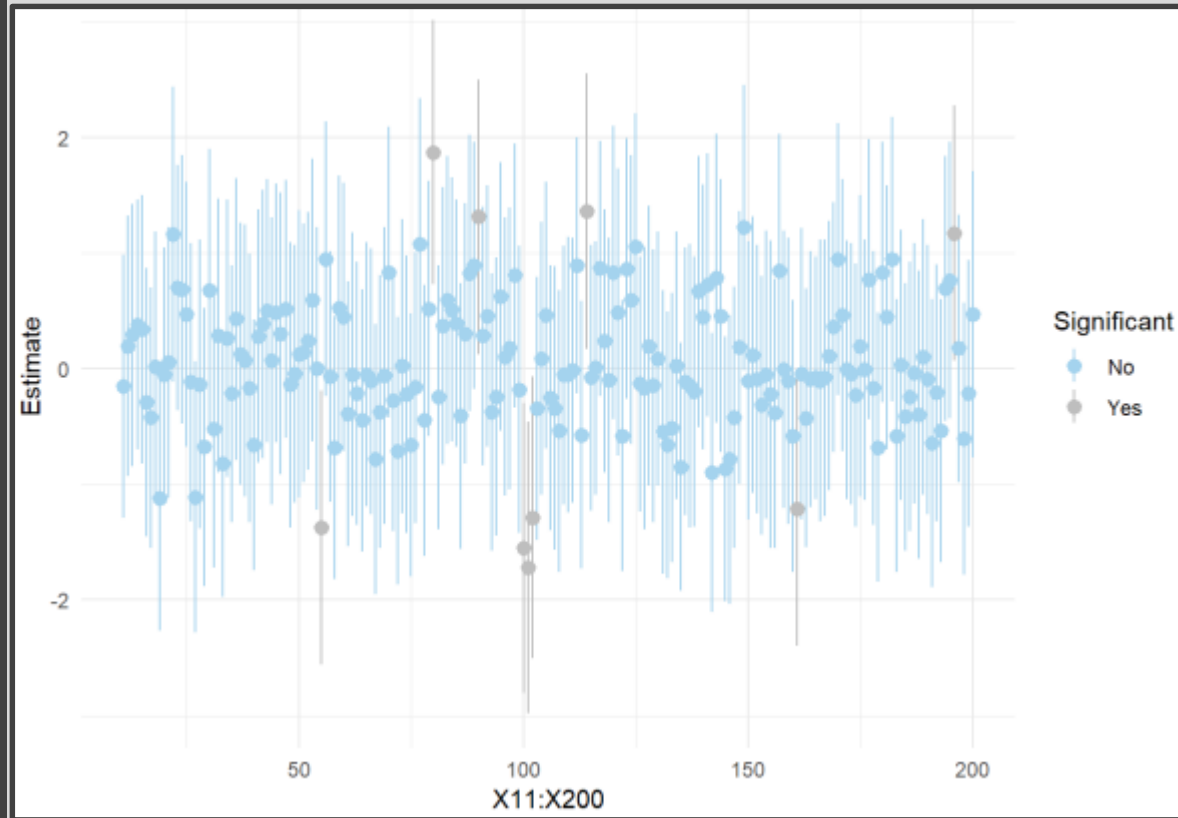


- What is the Problem?

Part 1: Simulate and Meditate



- Chunk 2 (Continued)
 - Figure 3



- What is the Problem?

Part 1: Simulate and Meditate



- Run Chunk 3
 - Regression for Each Predictor
 - Obtaining Coefficients

```
> coef(individual.mod)
(Intercept)          x.200
 0.1257668    -0.3200960
```

Save

- Obtaining P-Values

```
> summary(individual.mod)
```

Call:

```
lm(formula = y ~ ., data = SIM.DATA[, c(1, j + 1)])
```

Residuals:

Min	1Q	Median	3Q	Max
-47.252	-11.318	0.035	10.759	45.336

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.1258	0.7021	0.179	0.858
x.200	-0.3201	0.7230	-0.443	0.658

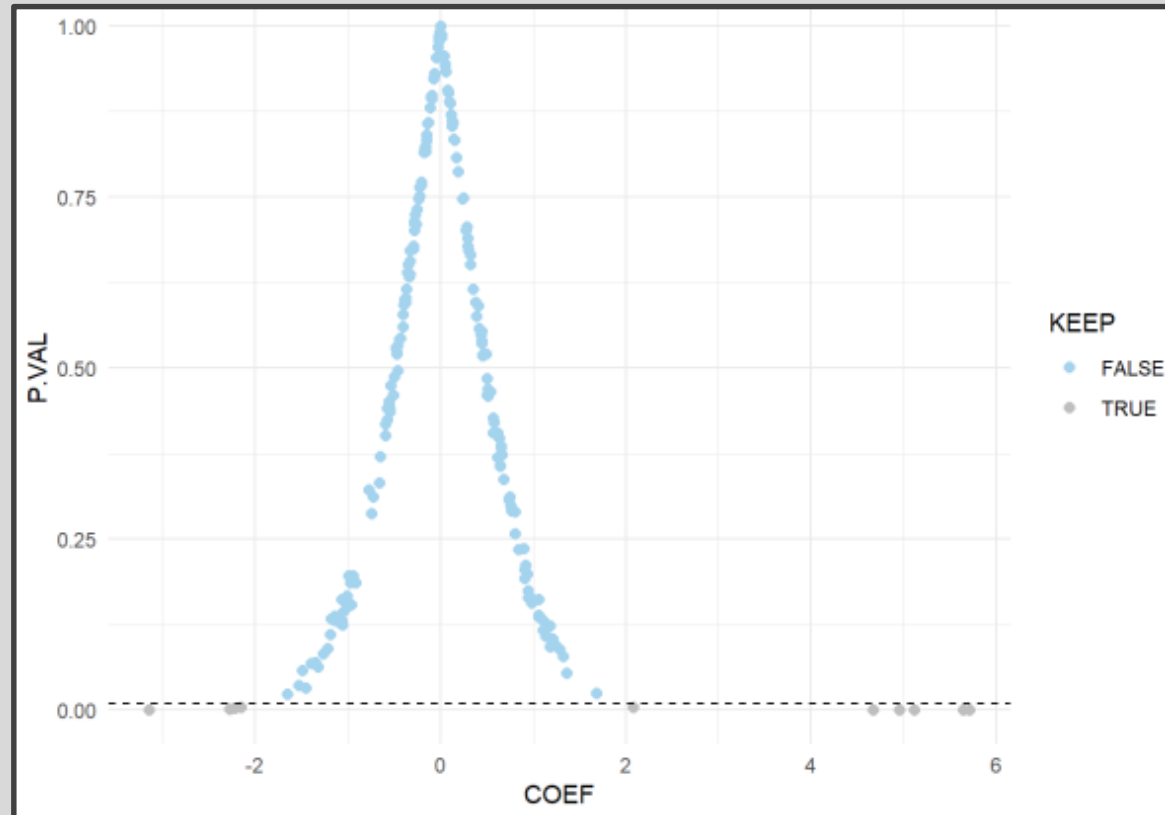
Save

Residual standard error: 15.66 on 498 degrees of freedom
Multiple R-squared: 0.0003934, Adjusted R-squared: -0.001614
F-statistic: 0.196 on 1 and 498 DF, p-value: 0.6582

Part 1: Simulate and Meditate



- Run Chunk 3
 - Figure Plots P-Values Against Coefficients



Part 1: Simulate and Meditate



- Run Chunk 3
 - Suppose We Were to Keep Only the Predictor Variables that Had P-Values < 0.01
 - Observe the Table

	P-Val > 0.01	P-Val < 0.01
Non-Zero	1%	4%
Zero	94%	1%

- 95% of Variables Ignored
- 5% of Variables Included
- Errors (What is Worse?)
 - We Will Ignore Variables that Are Important
 - We Will Include Variables that Are Irrelevant

Part 1: Simulate and Meditate



- Chunk 4
 - Try to Find the Smallest Cutoff Value So That We are Not Missing Important Variables
 - To Ensure We are Not Missing Important Variables, Should we Increase or Decrease the Original Cutoff (0.01)
 - What Cutoff Works?
 - Try Multiple Cutoffs and Observe the Table
 - Run the Code Inside the Chunk Until All 10 Important Variables are Retained for the Future

Part 1: Simulate and Meditate



- Chunk 4 (Continued)
 - Traditional Choice: 0.20
 - Output in Table

	P-Val > 0.01	P-Val < 0.01
Non-Zero	0%	5%
Zero	71%	24%

None of the Non-Zero Parameters Will Be Ignored

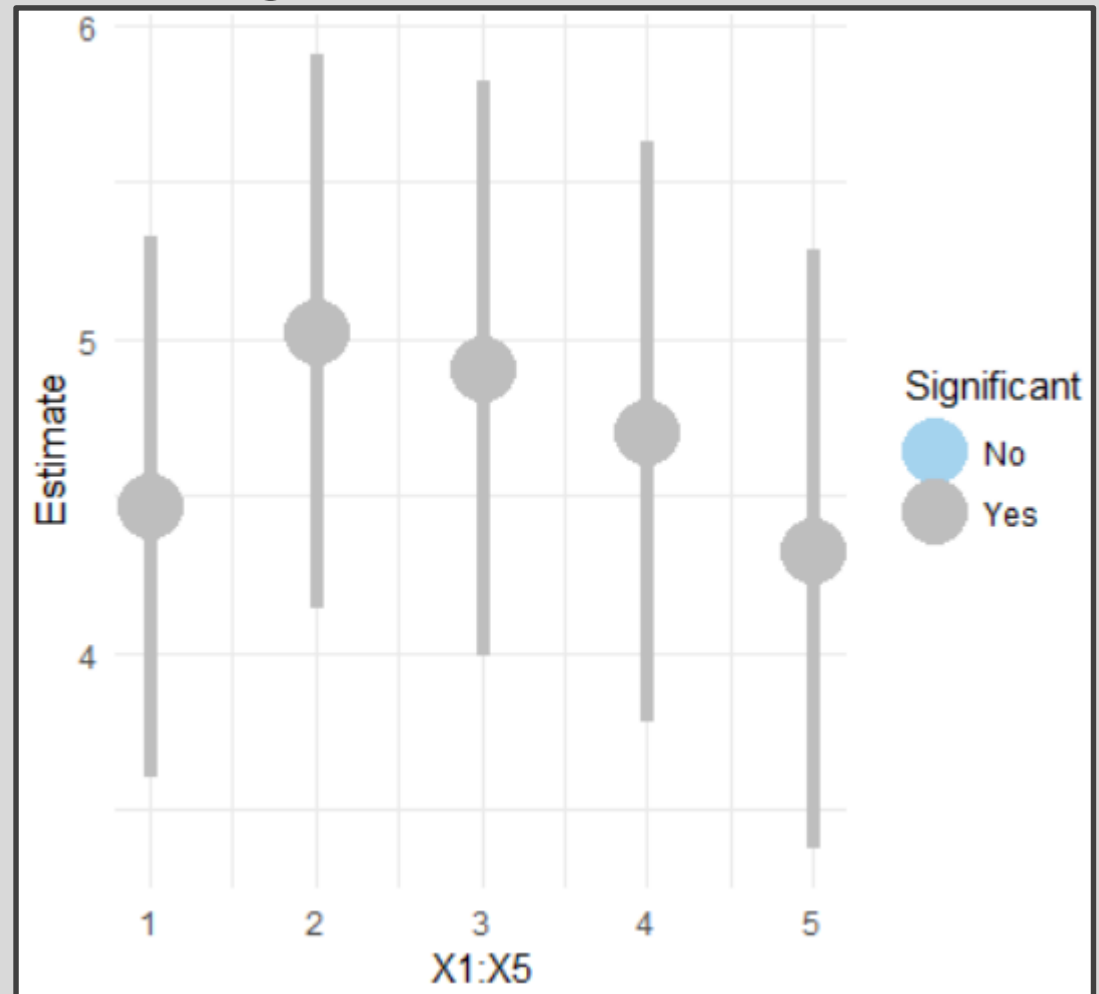
- Fit Linear Model for Variables Kept in Consideration

```
> lm(y~.,data=SIM.DATA[,c(1,which(KEEP)+1)])
```

Part 2: Shrinkage Estimation



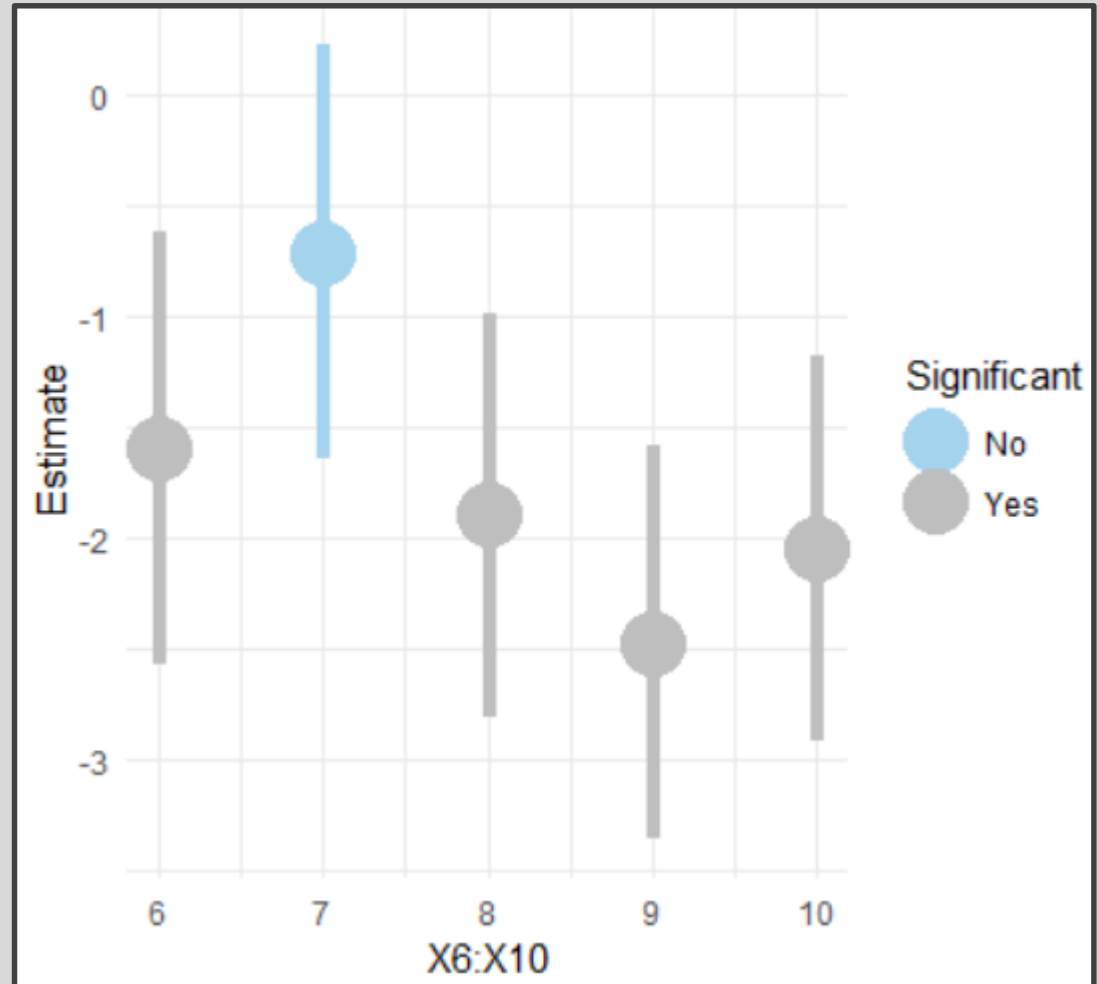
- Chunk 4 (Continued)
 - Suppose Cutoff is 0.2
 - Figure 1



Part 1: Simulate and Meditate



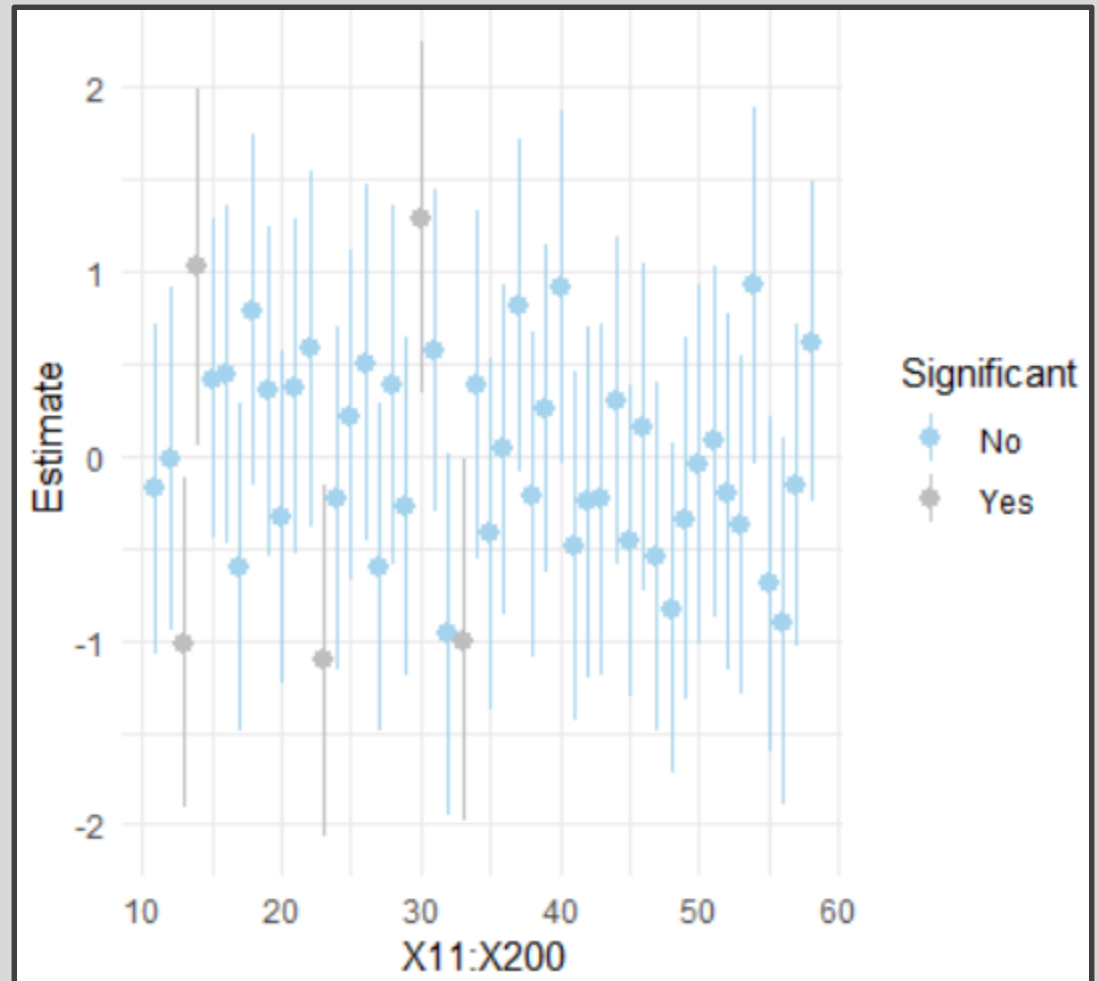
- Chunk 4 (Continued)
 - Figure 2



Part 1: Simulate and Meditate



- Chunk 4 (Continued)
 - Figure 2



Part 1: Simulate and Meditate



- Recap
 - Before Building Complex Models We are Performing a Simple Screening Procedure
 - Quick and Logical Approach
 - Problems
 - We May Lose Variables with Significant Interactions
 - We May Still Have Too Many
 - We May Retain Variables that are Highly Correlated
- Other Approach: Fit Full Model and Retain Variables with Sufficiently Small P-Values (<0.2)

Part 2: Shrinkage Estimation and More Meditation



- Classic Linear Model Estimation
 - Minimize Sum of Squared Error
$$SSE = \sum [y_i - (\beta_0 + x_i' \boldsymbol{\beta})]^2$$
 - Optimization: Find $\widehat{\beta}_0$ and $\widehat{\boldsymbol{\beta}}$ that Make SSE as Small as Possible
 - $\widehat{\beta}_0$ and $\widehat{\boldsymbol{\beta}}$ are Easily Found Using Matrix Representation
- Regularized Estimation
 - Produces Biased Estimates
 - Shrinks Coefficients Toward 0
 - Favors Smaller Models
 - May Lead to a Better Model for Out-of-Sample Prediction

Part 2: Shrinkage Estimation and More Meditation



- Three Popular Methods
 - Download R Package

```
> library(glmnet)
```

- Penalized SSE

$$PSSE = SSE + \lambda[(1 - \alpha) \sum_{i=1}^p \beta_i^2 + \alpha \sum_{i=1}^p |\beta_i|]$$

- Variations

- Ridge (1970): $\lambda = 1$ & $\alpha = 0$
- Lasso (1996): $\lambda = 1$ & $\alpha = 1$
- Elastic Net (2005)
 $\lambda = 1$ & $0 < \alpha < 1$

- Notice When

- $\lambda = 0 \Rightarrow PSSE = SSE$
- As λ Gets Bigger, the Coefficients Approach 0

Irrelevant
Nonsense

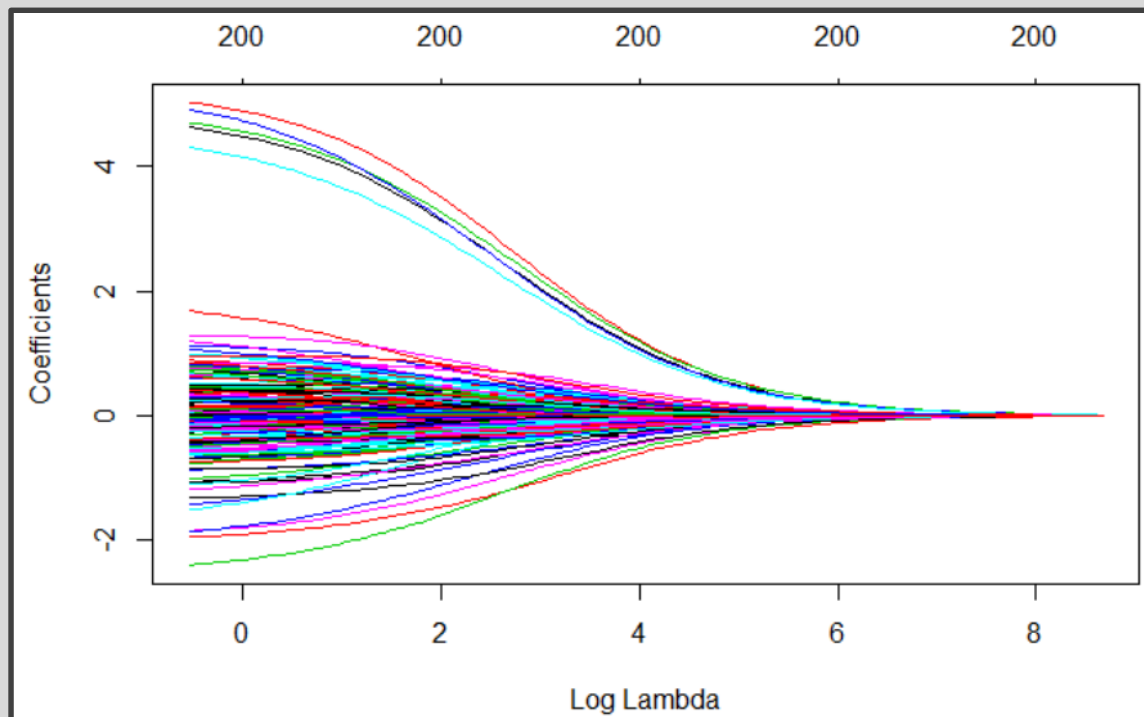


Watch Me
Whip
Watch Me
Lasso

Part 2: Shrinkage Estimation and More Meditation

- Run Chunk 1
 - Ridge Penalty

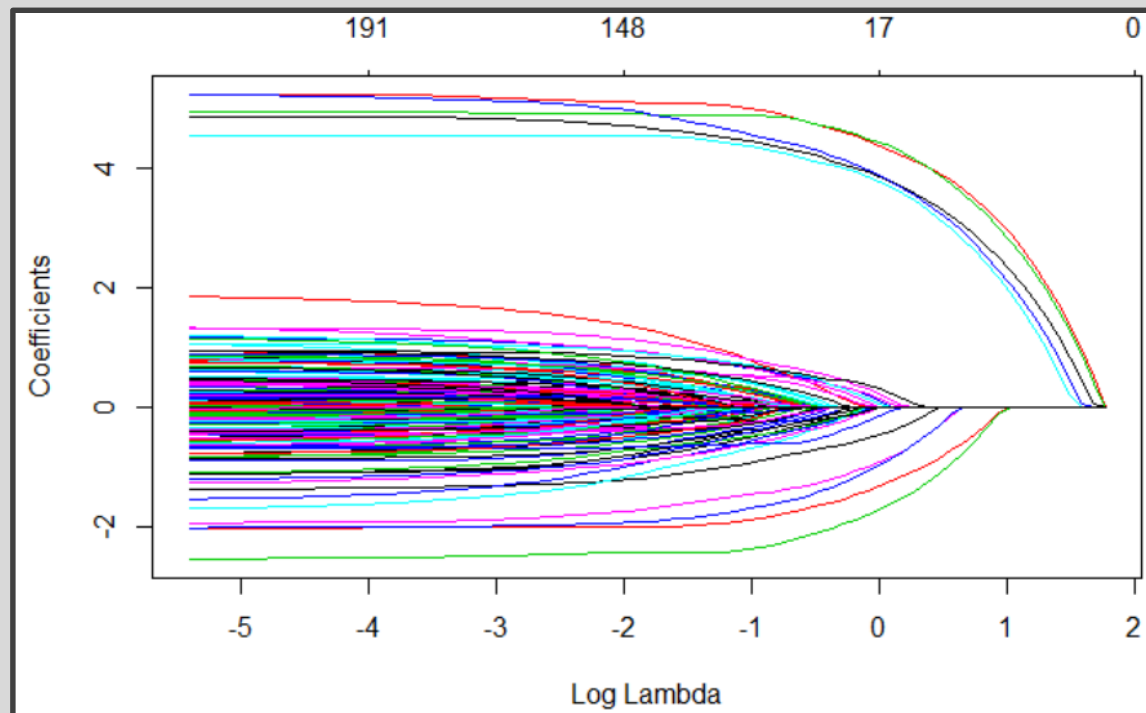
```
> ridge.mod=glmnet(x=as.matrix(SIM.DATA[,-1]),  
+                  y=as.vector(SIM.DATA[,1]),  
+                  alpha=0)  
> plot(ridge.mod,xvar="lambda")
```



Part 2: Shrinkage Estimation and More Meditation

- Run Chunk 2
 - Lasso Penalty

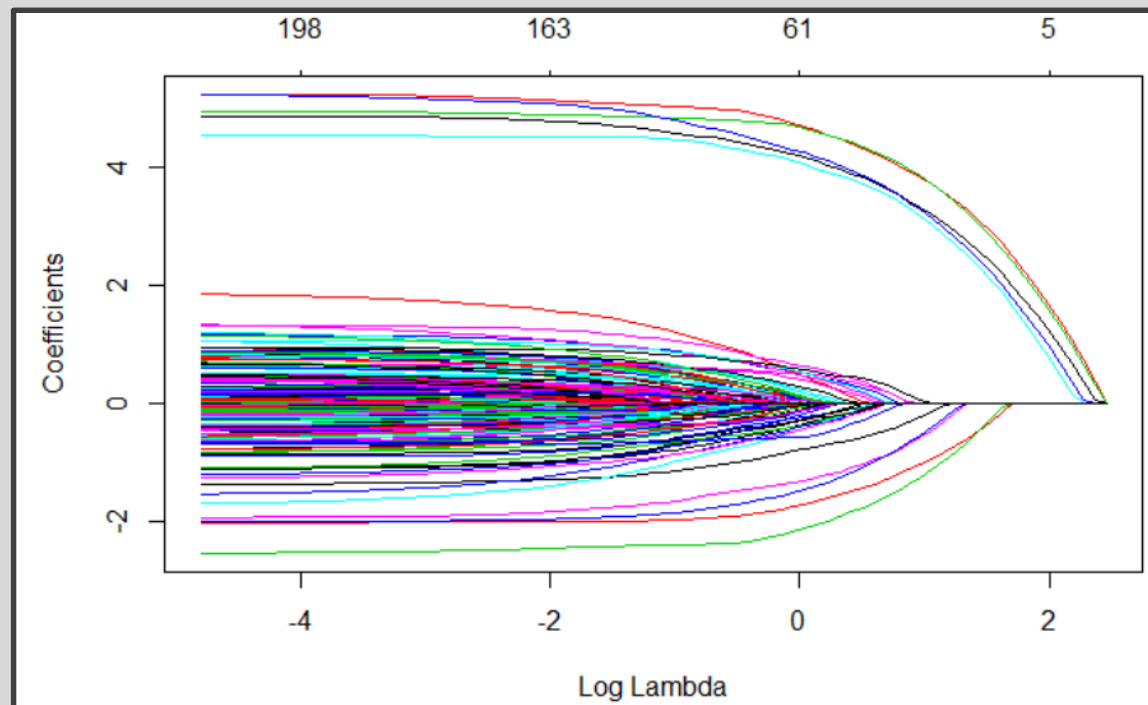
```
> lasso.mod=glmnet(x=as.matrix(SIM.DATA[,-1]),  
+                  y=as.vector(SIM.DATA[,1]),  
+                  alpha=1)  
> plot(lasso.mod,xvar="lambda")
```



Part 2: Shrinkage Estimation and More Meditation

- Run Chunk 3
 - Elastic Net Penalty

```
> enet.mod=glmnet(x=as.matrix(SIM.DATA[,-1]),  
+                  y=as.vector(SIM.DATA[,1]),  
+                  alpha=1/2)  
> plot(enet.mod,xvar="lambda")
```



Closing



Disperse
and Make
Reasonable
Decisions