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:

  $$y_i = \beta_0 + X_{1i}\beta_1 + \ \ldots + X_{pi}\beta_p + \epsilon_i$$

  where $i = 1,2,3,\ldots,n$

- Matrix Representation

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

  where $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]'$,

  $$\boldsymbol{\beta} = \left[\beta_1, \beta_2, \ldots, \beta_p\right]',$$

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

  and

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

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 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

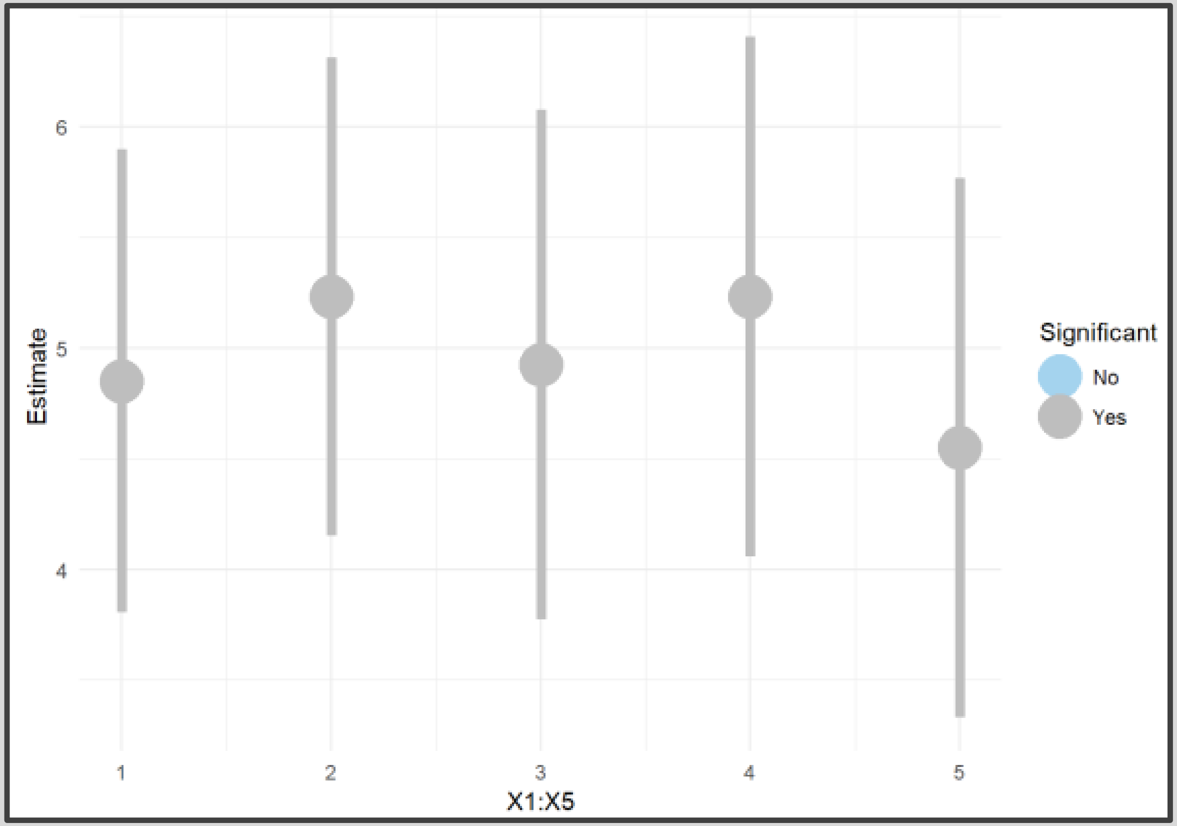
- Run Chunk 2
  - Fitting Naïve Linear Model
  - Obtaining Confidence Intervals for Parameters

```
> 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?

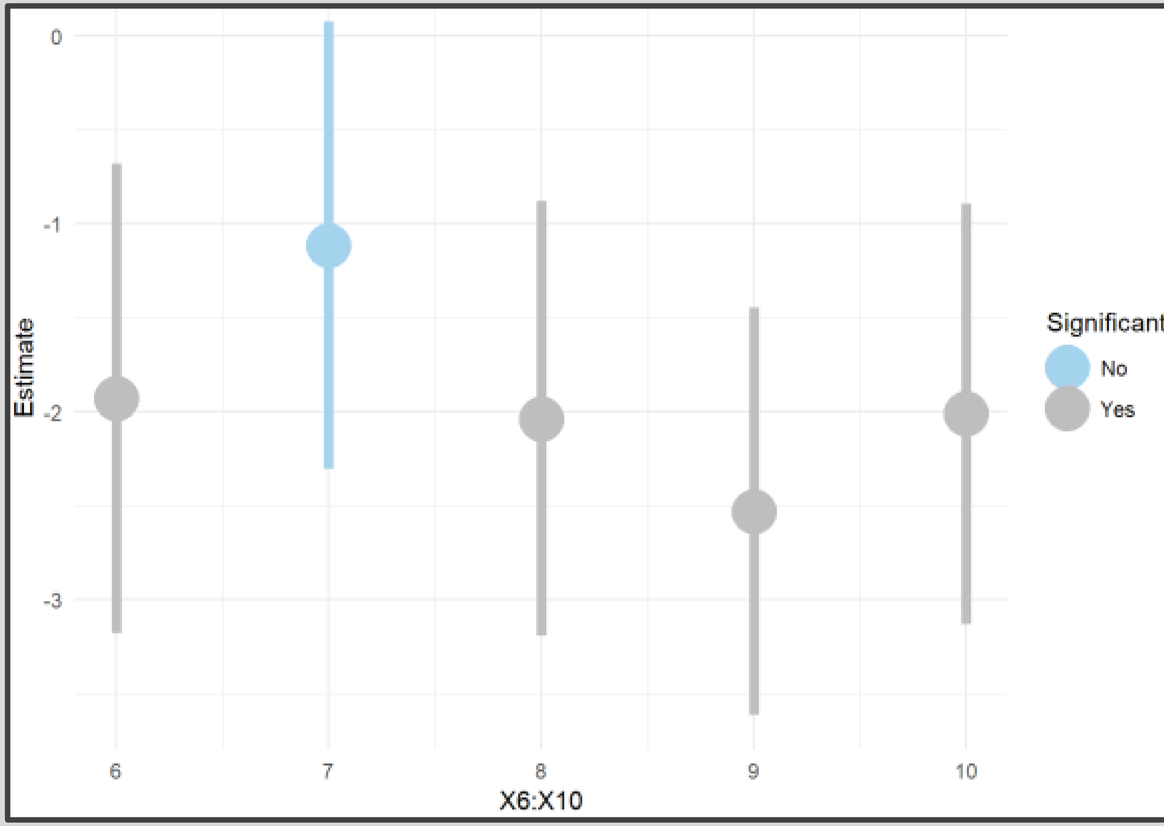
Part 1: Simulate and Meditate

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



Part 1: Simulate and Meditate

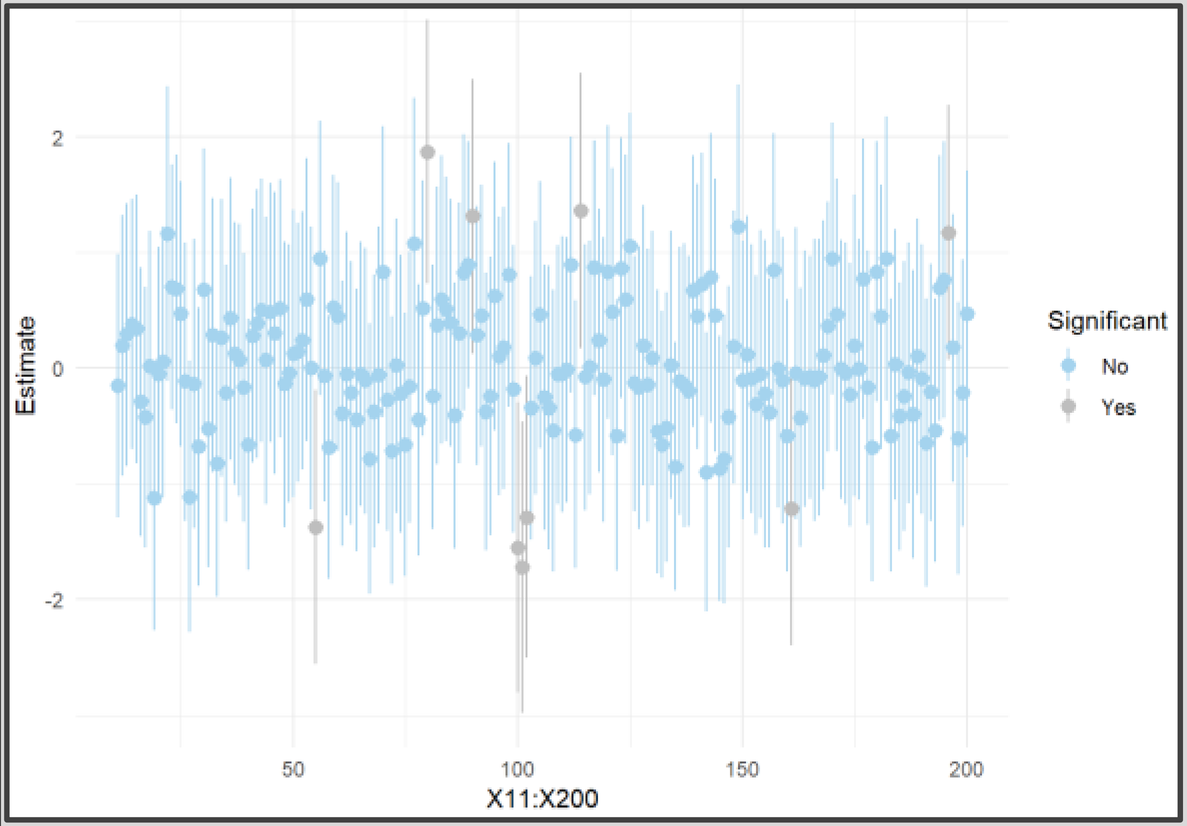
- Chunk 2 (Continued)
  - Figure 2



- 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

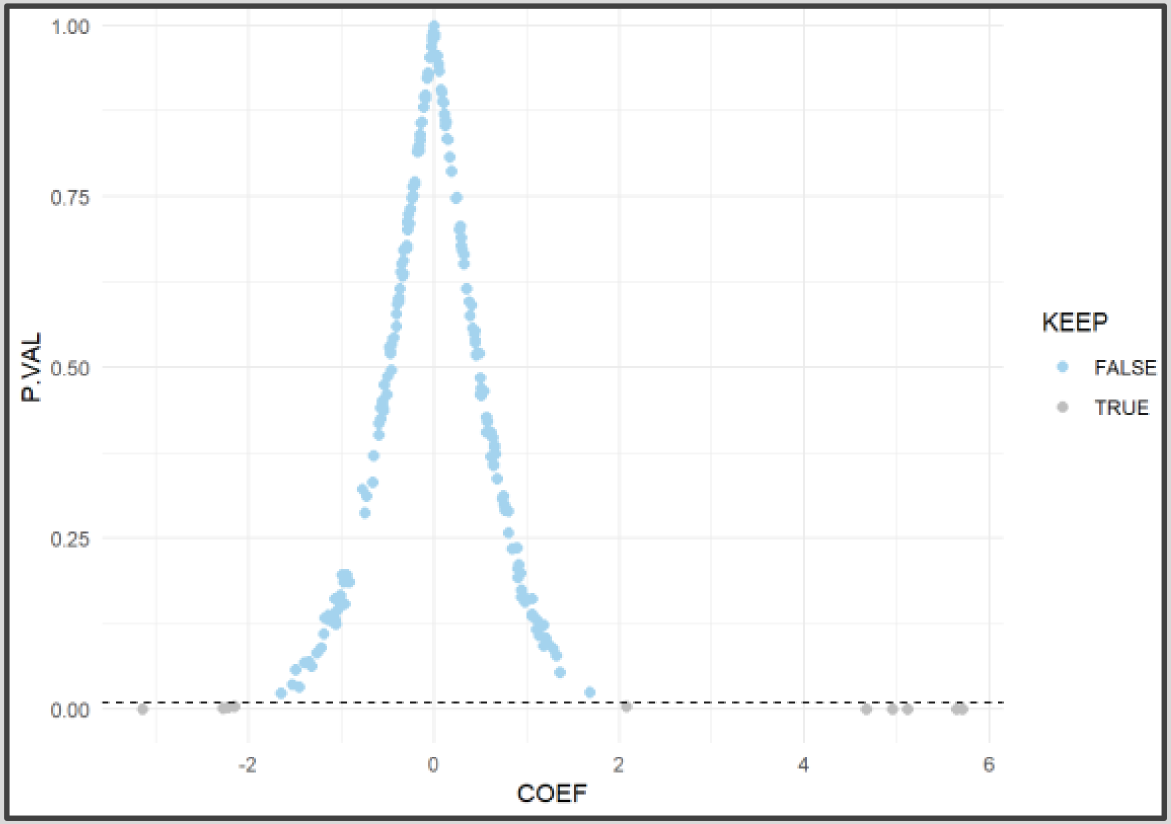
- Obtaining P-Values

```
> 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

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
```

Save

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 Futre

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%

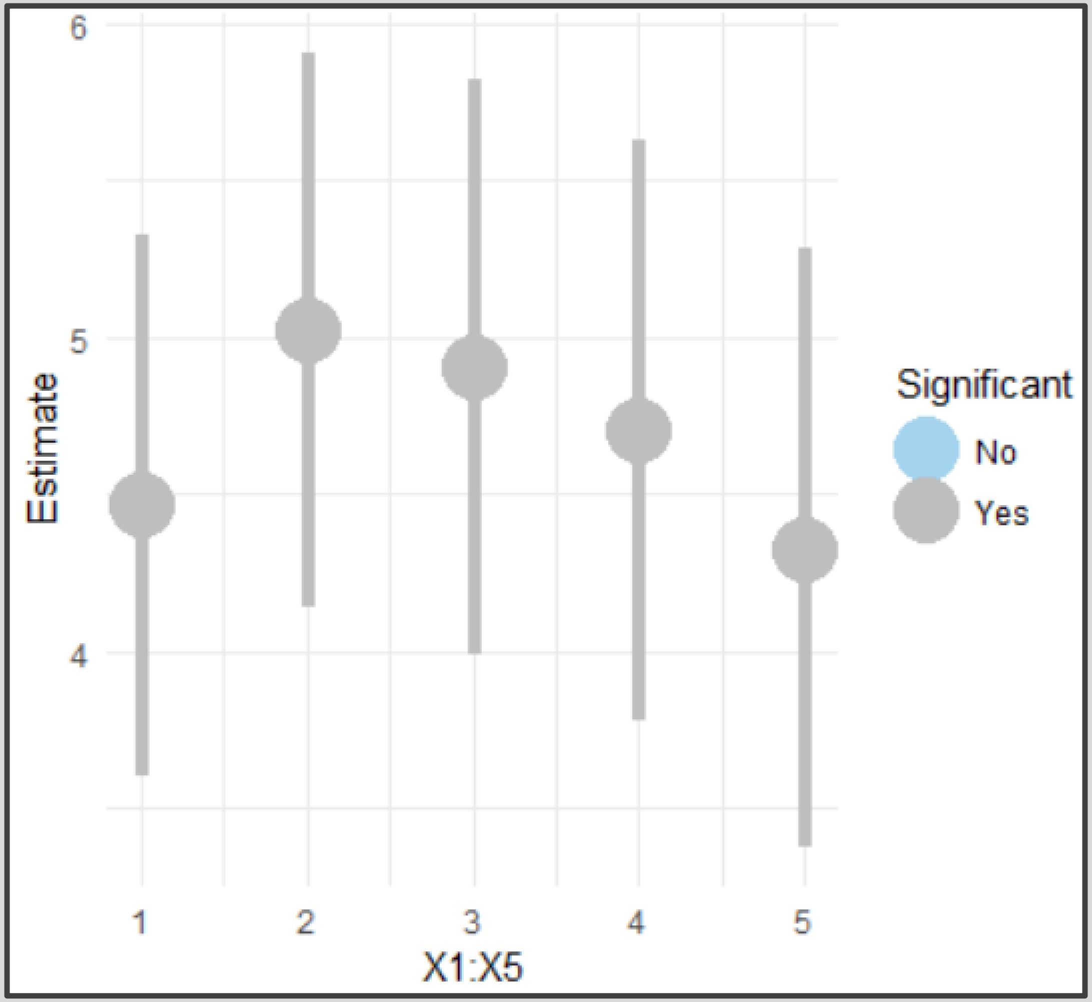
None of the Non-Zero Parameters Will Be Ignored

- Fit Linear Model for Variables Kept in Consideration

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

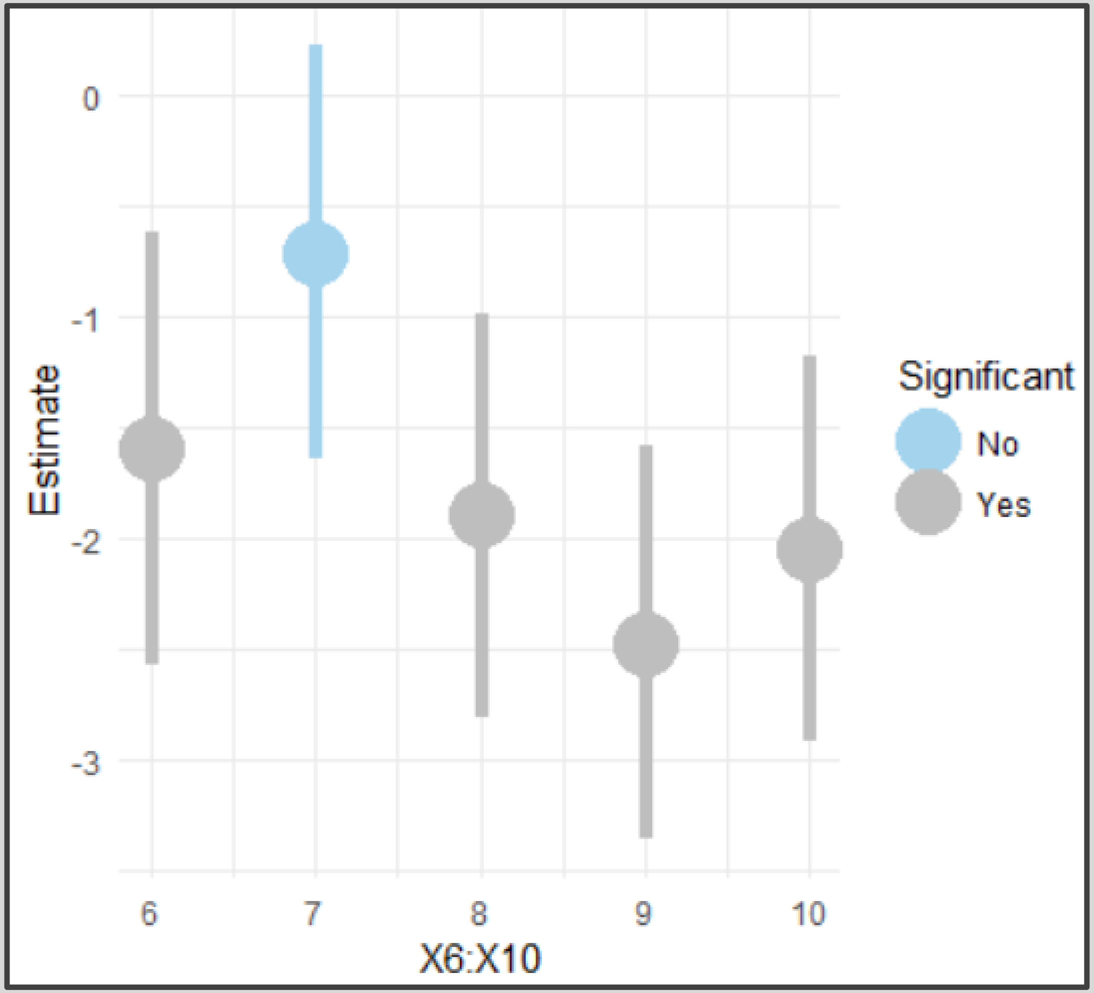
Part 2: Shrinkage Estimation

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



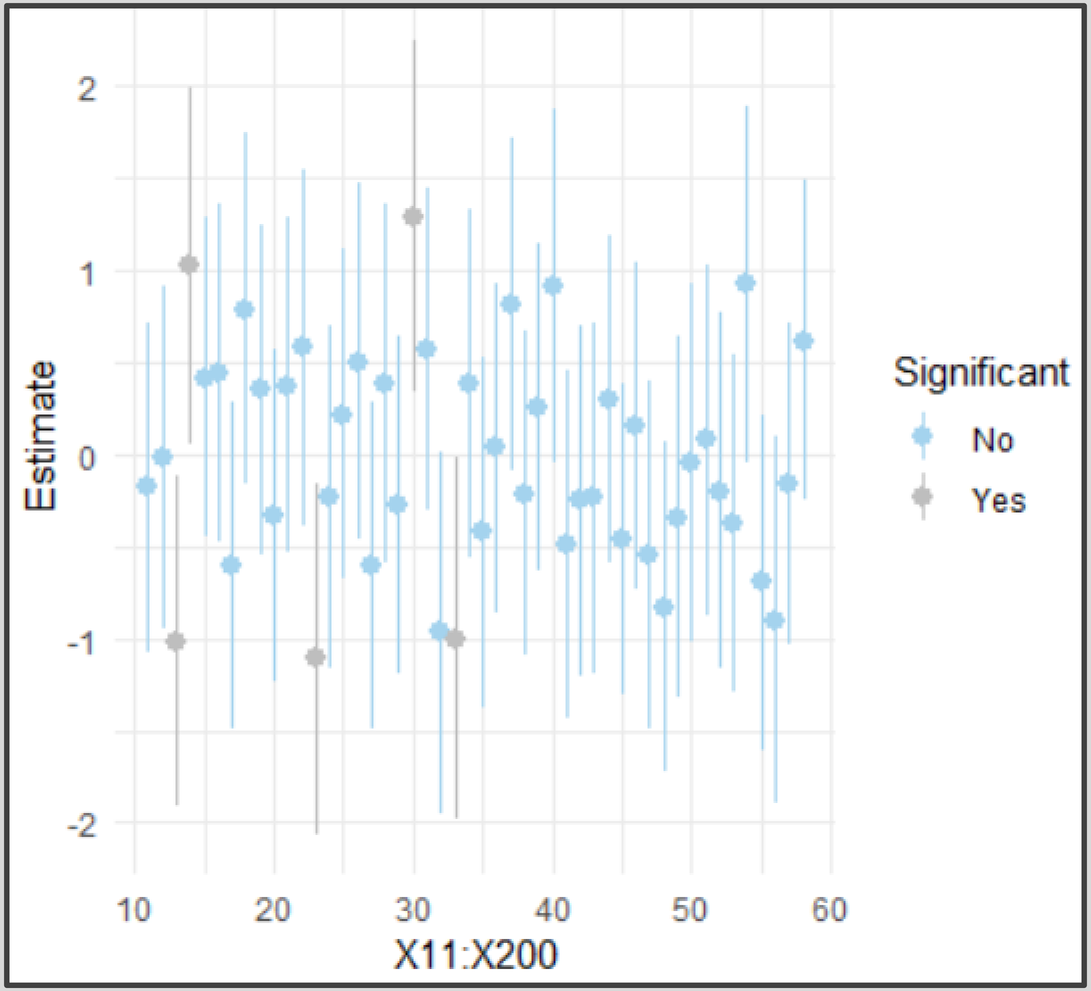
Part 1: Simulate and Meditate

- 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

$$SSE = \sum [y_i - (\beta_0 + \boldsymbol{x_i}'\boldsymbol{\beta})]^2$$

  - 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

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

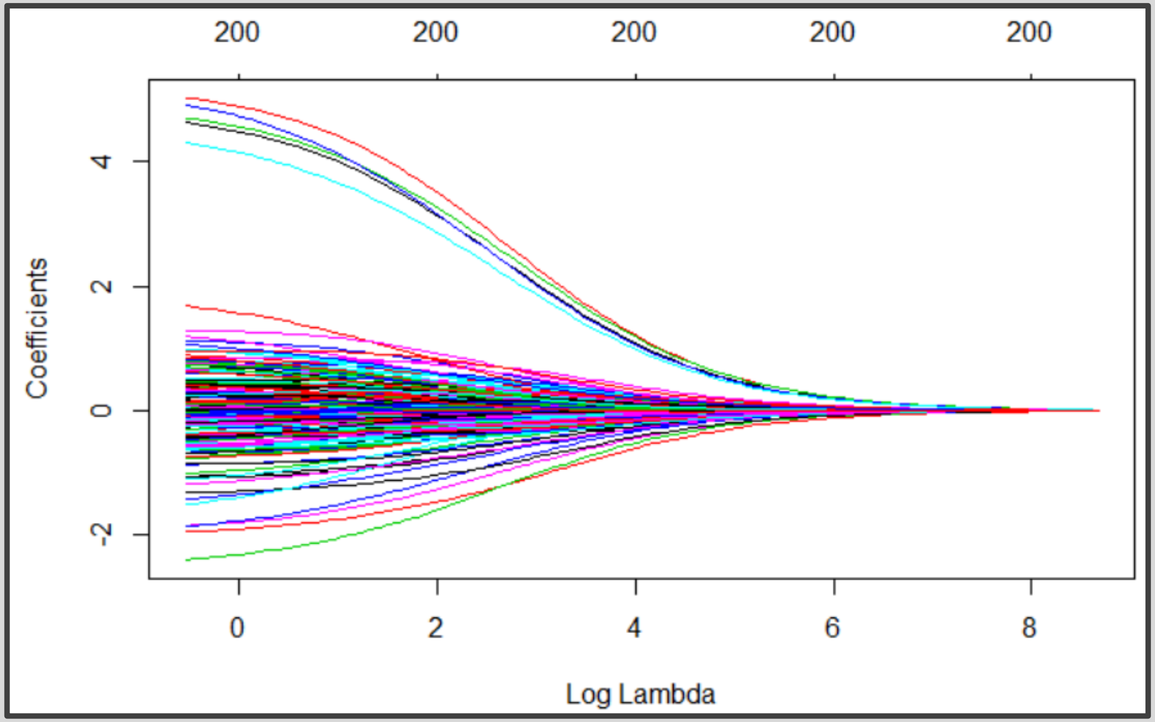
  - Variations
    - Ridge (1970): $\lambda = 1 \;\&\; \alpha = 0$
    - Lasso (1996): $\lambda = 1 \;\&\; \alpha = 1$
    - Elastic Net (2005)

      $\lambda = 1 \;\&\; 0 < \alpha < 1$
  - Notice When
    - $\lambda = 0 \Rightarrow$ PSSE=SSE
    - As λ Gets Bigger, the Coefficients Approach 0

Part 2: Shrinkage Estimation and More Meditation

- Run Chunk 1
  - Ridge Penalty

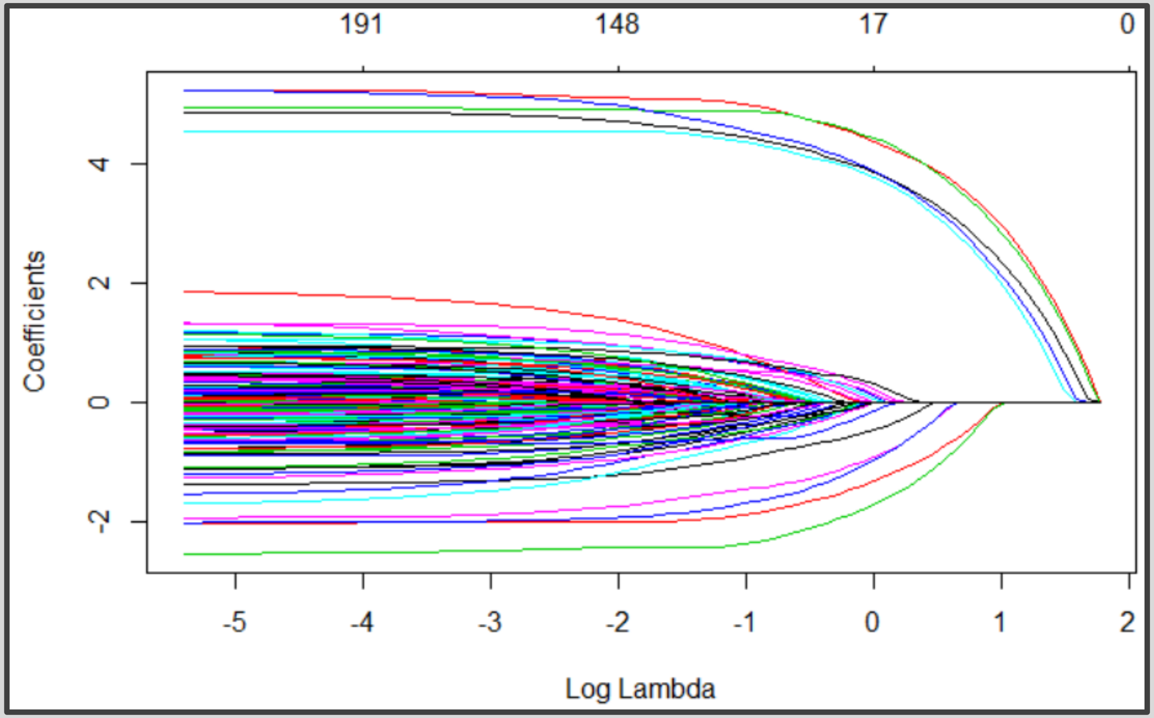
```
> ridge.mod=glmnet(x=as.matrix(SIM.DATA[,-1]),
+                  y=as.vector(SIM.DATA[,1]),
+                  alpha=0)
> plot(ridge.mod,xvar="lambda")
```



Part 2: Shrinkage Estimation and More Meditation

- Chunk 1 (Continued)
  - Lasso Penalty

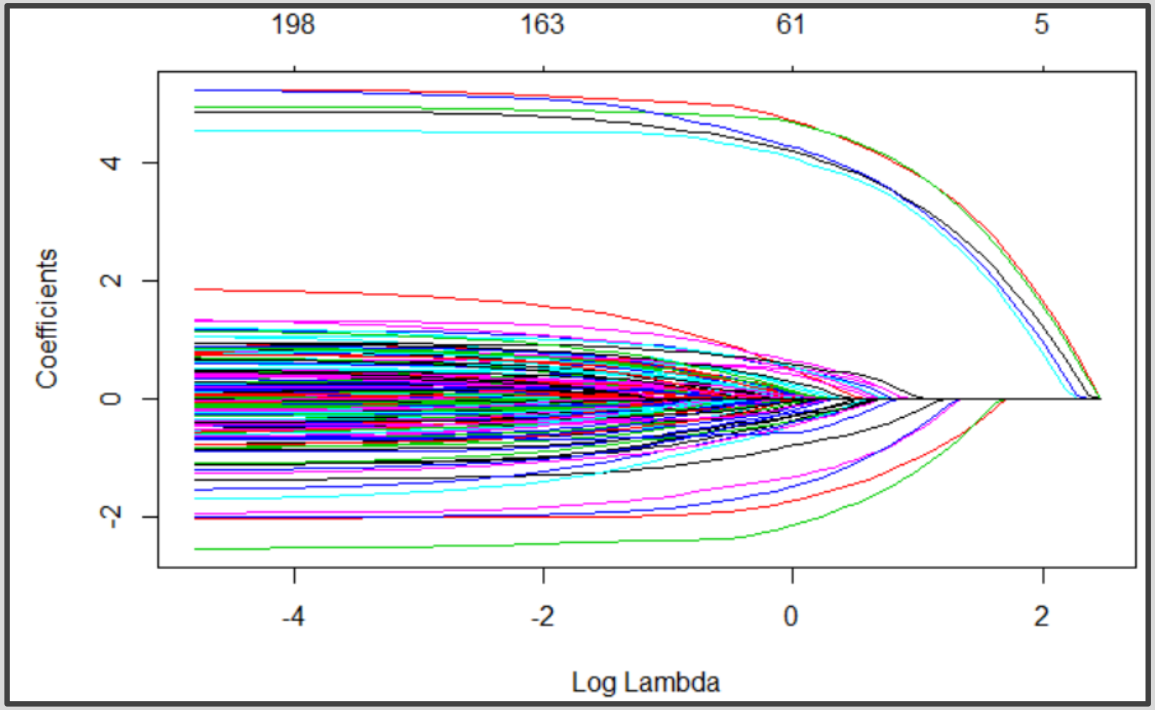
```
> lasso.mod=glmnet(x=as.matrix(SIM.DATA[,-1]),
+                  y=as.vector(SIM.DATA[,1]),
+                  alpha=1)
> plot(lasso.mod,xvar="lambda")
```



Part 2: Shrinkage Estimation and More Meditation

- Chunk 1 (Continued)
  - 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")
```



Irrelevant Nonsense

Watch Me Whip Watch Me Lasso

Part 2: Shrinkage Estimation and More Meditation

- Tuning Parameters
  - Use Cross-Validation to Choose Tuning Parameters λ & α
  - Constraints
    - $\lambda > 0$
    - $0 \le \alpha \le 1$
  - Best Approach:
    - Divide Data Into Train & Test
    - Loop Over a Vector of Alpha
    - Find Best Lambda for Each Alpha Considered Using CV in Train
    - For Each Alpha and Best Lambda, Predict on Test and Select Alpha and Lambda that Minimize MSE

Part 2: Shrinkage Estimation and More Meditation

- Chunk 2
  - Illustration of 10 Fold CV
  - Finding Best Combination of Alpha and Lambda

	alpha	lambda	MSE
[1,]	0.0	25.073407	186.1875
[2,]	0.1	8.601355	149.4262
[3,]	0.2	4.719988	137.4652
[4,]	0.3	3.453454	133.7793
[5,]	0.4	2.590091	130.7873
[6,]	0.5	2.274097	130.5494
[7,]	0.6	1.895081	129.1495
[8,]	0.7	1.782728	130.0287
[9,]	0.8	1.559887	129.2083
[10,]	0.9	1.521754	131.2262
[11,]	1.0	1.247909	128.0857

Best: $\lambda = 1$ & $\alpha = 1.24$

Part 3: Less Meditation and More Application

- Built-In Data `> mpg`
    - n=234
    - Focus is on Modeling Hwy MPG
    - Subset Data to Include Only Wanted Covariates

year <int>	displ <dbl>	cyl <int>	drv <chr>	cty <int>	hwy <int>	fl <chr>	class <chr>
1999	1.8	4	f	18	29	p	compact
1999	1.8	4	f	21	29	p	compact
2008	2.0	4	f	20	31	p	compact
2008	2.0	4	f	21	30	p	compact
1999	2.8	6	f	16	26	p	compact
1999	2.8	6	f	18	26	p	compact

- There are p=7 Covariates
- Difficulty
    - Fitting all Combinations
    - Considering All 2-Way Interaction Terms

Part 3: Less Meditation and More Application

- Run Chunk 1
  - Creating Model Matrix
    - Up to 2-Way Interactions
    - Now, p=115
  - Model Selection is Difficult
  - Dividing Data into Train & Test is Not Advised (n=234)

- Run Chunk 2
  - Only a Few Options

alpha <dbl>	lambda <dbl>	CV.Error <dbl>
0.00	1.44063441	1.722966
0.25	0.55006214	1.620769
0.50	0.18956825	1.488094
0.75	0.10492193	1.456773
1.00	0.04942052	1.411025

Lowest Estimation of Prediction Error

Part 3: Less Meditation and More Application

- Chunk 2 (Continued)
  - Understanding cv.glmnet Object
    - $lambda = Contains Vector of Lambda Auto-Generated
    - $cvm = Cross Validated Estimate of Error for Each Lambda in $lambda
    - $lambda.min = The Lambda that Leads to Smallest CV Measure of Error
    - $lambda.1se = The Largest Value of Lambda Such That Error is Within 1 SD of the Error Using $lambda.min

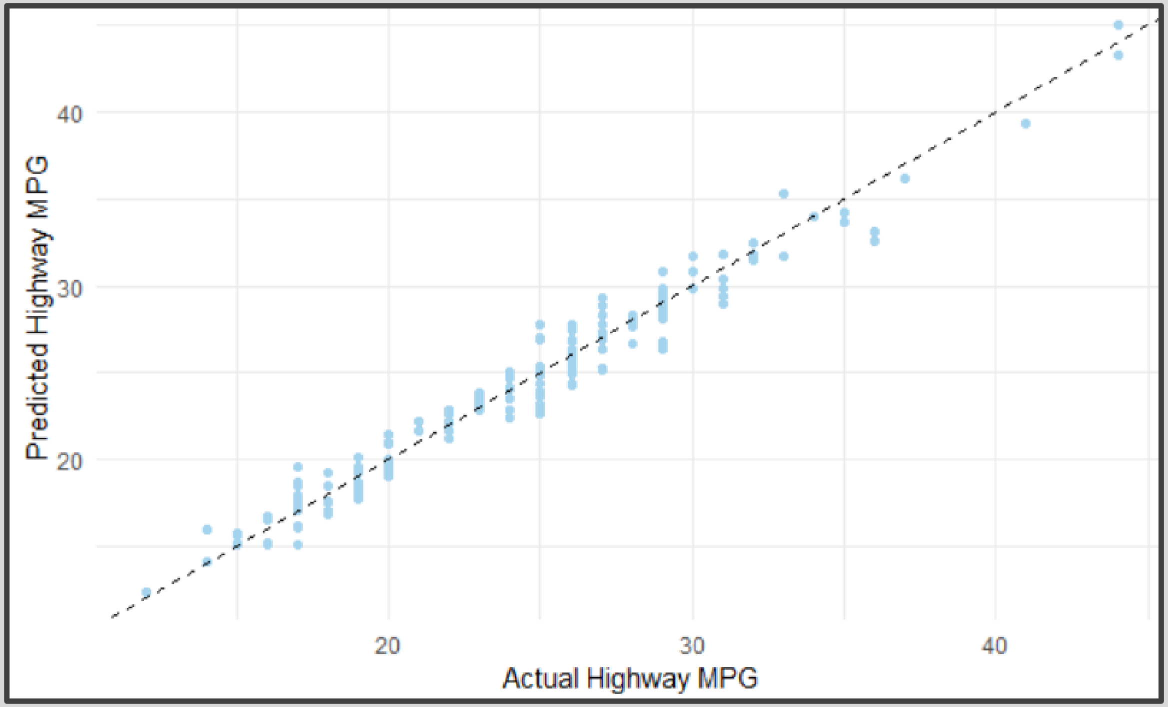
Part 3: Less Meditation and More Application

- Run Chunk 3
  - Next
    - Use Best Alpha and Lambda
    - Observe the Non-Zero Coefficients
    - Plot Predictions and Errors
  - Table of Non-Zero Coefficients
    - Before p=115
    - Now p=28

```
## # A tibble: 29 x 2
##    Parameter              Estimate
##    <chr>                     <dbl>
##  1 Int                   -123.
##  2 year                     0.0660
##  3 cty                      0.799
##  4 fle                     -1.37
##  5 flr                     -0.0629
##  6 classpickup             -0.104
##  7 classsuv                -1.37
##  8 year:cyl                -0.0000392
##  9 year:drvf                0.0000955
## 10 year:cty                 0.0000565
## 11 year:classmidsize        0.0000259
## 12 year:classpickup        -0.000659
## 13 displ:drvr               0.127
## 14 displ:classmidsize       0.0317
## 15 displ:classsuv          -0.178
## 16 cyl:fle                 -0.143
## 17 cyl:flr                 -0.0973
## 18 cyl:classcompact         0.0462
## 19 cyl:classsuv            -0.0262
## 20 drvf:cty                 0.0466
## 21 drvr:cty                 0.0282
## 22 drvf:fld                 2.54
## 23 drvr:classsubcompact    -0.0754
## 24 cty:classminivan        -0.0574
## 25 cty:classpickup         -0.106
## 26 flr:classmidsize         0.488
## 27 flp:classsubcompact     -1.42
## 28 fld:classsuv            -0.552
## 29 flp:classsuv            -0.431
```

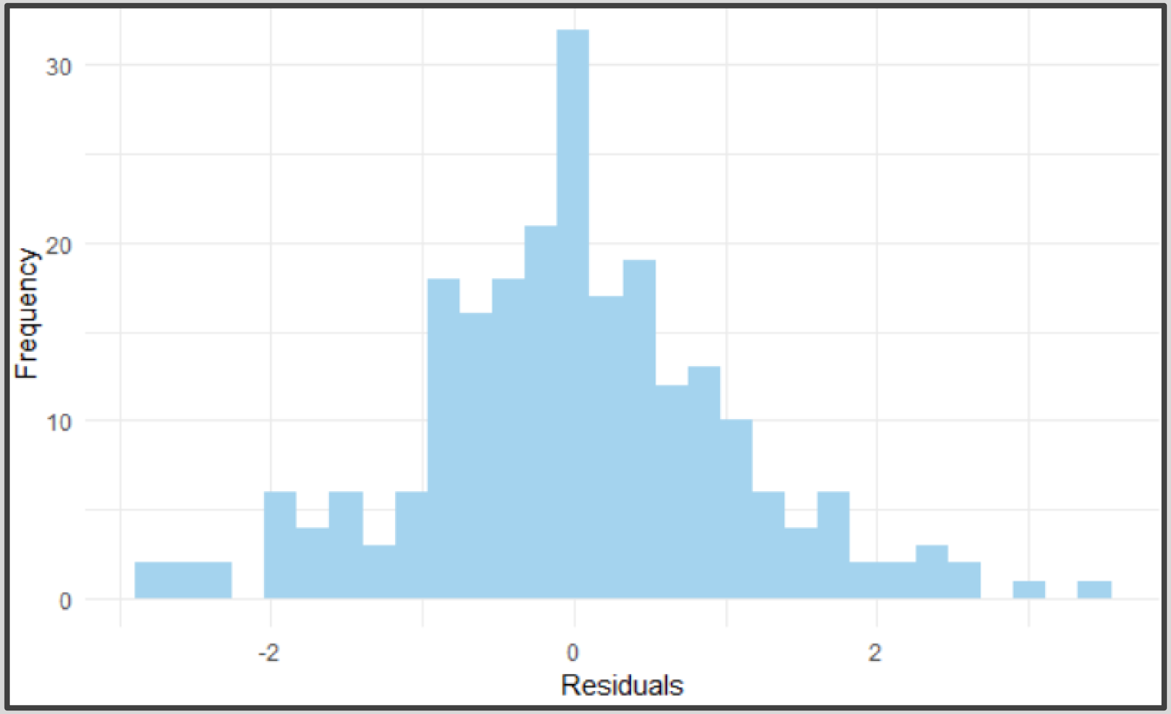
Part 3: Less Meditation and More Application

- Chunk 3 (Continued)
  - Comparing Predict and Actual



Part 3: Less Meditation and More Application

- Chunk 3 (Continued)
  - Distribution of Residuals



Closing

Disperse and Make Reasonable Decisions