Linear Model Selection and Regularization

ddebarr@uw.edu

2017-02-09

"A machine learning researcher, a crypto-currency expert, and an Erlang programmer walk into a bar. Facebook buys the bar for $27 billion."
-- https://twitter.com/ML_Hipster

Course Outline

1. Introduction to Statistical Learning
2. Linear Regression
3. Classification
4. Resampling Methods
5. Linear Model Selection and Regularization
6. Moving Beyond Linearity
7. Tree-Based Methods
8. Support Vector Machines
9. Unsupervised Learning
10. Neural Networks and Genetic Algorithms

Agenda

Homework

Model Selection/Regularization

Focus of the Chapter is Linear Regression

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \epsilon$$

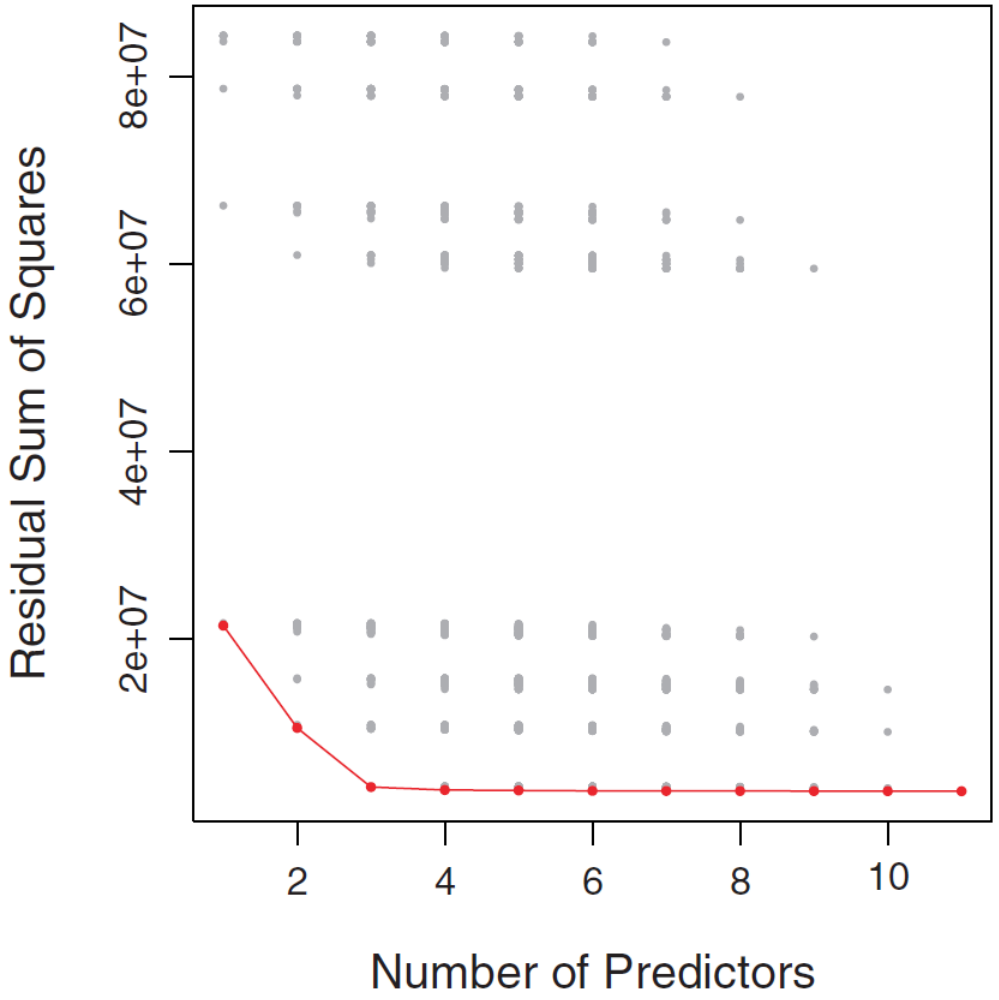
- Motivations
  - Better predictive accuracy; e.g. by using regularization (shrinkage; making the regression coefficients smaller) or dimensionality reduction
  - Better model interpretability; e.g. by using feature (subset) selection

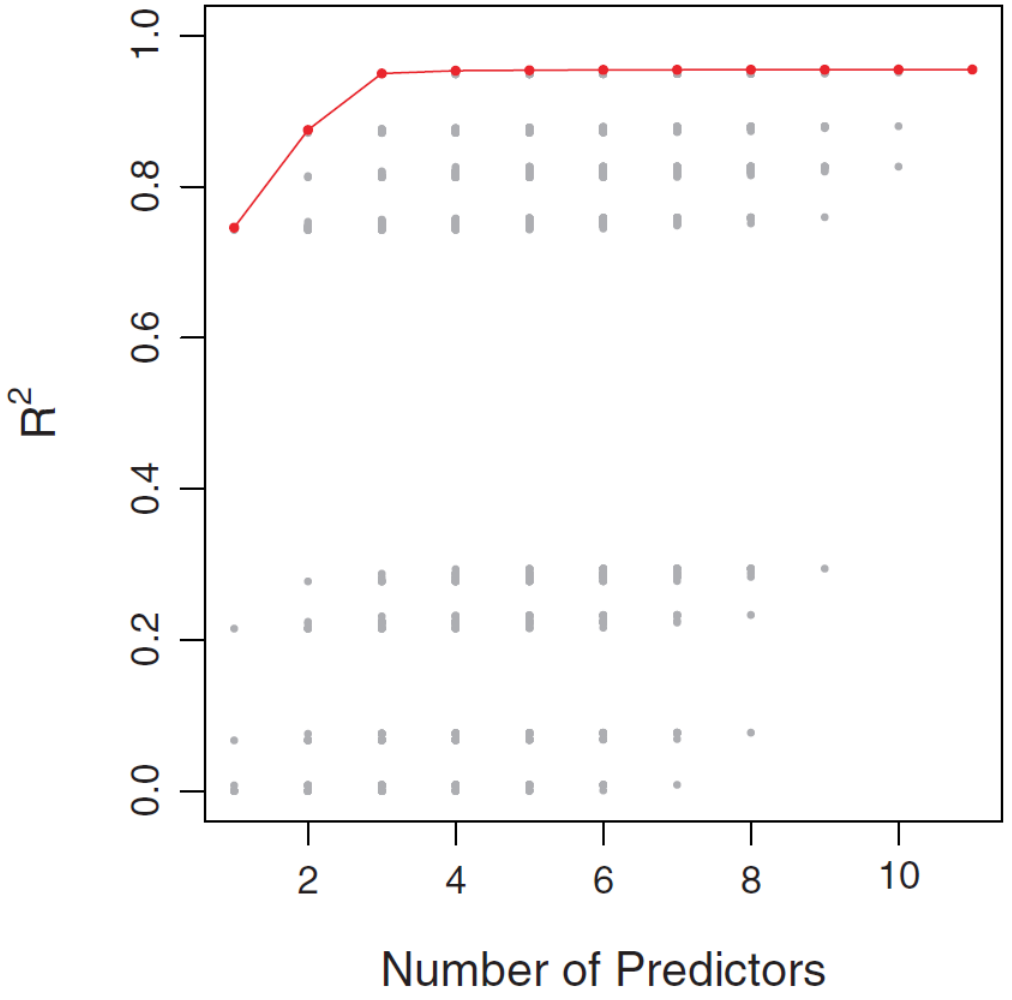
Best Subset Selection Algorithm

Algorithm 6.1 *Best subset selection*

1. Let $\mathcal{M}_0$ denote the *null model*, which contains no predictors. This model simply predicts the sample mean for each observation.
2. For $k = 1, 2, \ldots p$: $\sum_{k=1}^{p} \binom{p}{k} = 2^p$
   (a) Fit all $\binom{p}{k}$ models that contain exactly k predictors.
   (b) Pick the best among these $\binom{p}{k}$ models, and call it $\mathcal{M}_k$. Here *best* is defined as having the smallest RSS, or equivalently largest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Best Subset Selection Perf [2048 models]





Forward Stepwise Selection Algorithm

Algorithm 6.2 *Forward stepwise selection*

1. Let $\mathcal{M}_0$ denote the *null* model, which contains no predictors.
2. For $k = 0, \ldots, p-1$: $\quad 1+\sum_{k=0}^{p-1} p-k = 1+\frac{p(p+1)}{2}$
   (a) Consider all $p-k$ models that augment the predictors in $\mathcal{M}_k$ with one additional predictor.
   (b) Choose the *best* among these $p-k$ models, and call it $\mathcal{M}_{k+1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Best Subset versus Forward Stepwise Selection

They differ for the four variable model …

# Variables	Best subset	Forward stepwise
One	rating	rating
Two	rating, income	rating, income
Three	rating, income, student	rating, income, student
Four	cards, income, student, limit	rating, income, student, limit

Backward Stepwise Selection Algorithm

Algorithm 6.3 *Backward stepwise selection*

1. Let $\mathcal{M}_p$ denote the *full* model, which contains all p predictors.
2. For $k = p, p-1, \ldots, 1$: $1+\sum_{k=0}^{p-1} p-k = 1+\frac{p(p+1)}{2}$
   (a) Consider all k models that contain all but one of the predictors in $\mathcal{M}_k$, for a total of $k-1$ predictors.
   (b) Choose the *best* among these k models, and call it $\mathcal{M}_{k-1}$. Here *best* is defined as having smallest RSS or highest R^2.
3. Select a single best model from among $\mathcal{M}_0, \ldots, \mathcal{M}_p$ using cross-validated prediction error, C_p (AIC), BIC, or adjusted R^2.

Hybrid Approach

- Best Subset, Forward Stepwise, and Backward Stepwise Selection give similar but not identical results
- Hybrid
  - Variables are added to the model sequentially; however, after adding each new variable, the method may also remove any variables that no longer appear relevant
  - Attempts to mimic best subset selection while retaining the computational advantages of forward and backward stepwise selection

Two Common Approaches to Estimate Test Error

- Indirectly: adjusting the training set error by penalizing model complexity

- Directly: using either validation or cross validation [** use this **]

Penalized Error Estimates

- Colin Mallow's selection Criterion for a model with "p" predictors

$$C_p = \frac{1}{n}\left(\mathrm{RSS} + 2d\hat{\sigma}^2\right)$$

$\hat{\sigma}^2$ is an estimate of $Var(\varepsilon)$: $\frac{RSS_{all}}{n - p_{all}}$

- Hirotugu Akaike's Information Criterion (AIC) ["a-ka-ih-keh"]

$$\mathrm{AIC} = \frac{1}{n\hat{\sigma}^2}\left(\mathrm{RSS} + 2d\hat{\sigma}^2\right)$$

"d" is the number of dimensions (predictors)

- Bayesian Information Criterion (BIC)

$$\mathrm{BIC} = \frac{1}{n}\left(\mathrm{RSS} + \log(n)d\hat{\sigma}^2\right)$$

Practical Example for AIC and BIC

```
> # AIC versus BIC
> Credit = read.csv("http://www-bcf.usc.edu/~gareth/ISL/Credit.csv", row.names = 1)
> model = lm(Balance ~ Cards + Income + Student + Limit, data = Credit)
> AIC(model)
[1] 4822.701
> BIC(model)
[1] 4846.65
> residuals = model$residuals
> n = length(residuals)
> weights = rep.int(1, n)
> logLik(model)
'log Lik.' -2405.351 (df=6)
> log.likelihood = 0.5 * (sum(log(weights)) - n *
+                   (log(2 * pi) + 1 - log(n) + log(sum(weights * (residuals^2)))))
> log.likelihood
[1] -2405.351
> df = model$rank + 1    # +1 for dispersion parameter
> df
[1] 6
> -2 * log.likelihood + 2 * df
[1] 4822.701
> -2 * log.likelihood + log(n) * df
[1] 4846.65
```

Log Likelihood for Linear Regression

$$p\left(y_i \mid x_i; \beta, \sigma^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)$$

likelihood for response i

$$\prod_{i=1}^{n} p\left(y_i \mid x_i; \beta, \sigma^2\right) = \prod_{i=1}^{n}\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)\right)$$

likelihood for i.i.d. responses

$$\log\left(p\left(y_i \mid x_i; \beta, \sigma^2\right)\right) = \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)\right)$$

log likelihood for response i

$$\log\left(\prod_{i=1}^{n} p\left(y_i \mid x_i; \beta, \sigma^2\right)\right) = \sum_{i=1}^{n} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2}\frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)\right)$$

log likelihood for i.i.d. responses

$$= \sum_{i=1}^{n} \log\left(\left(2\pi\sigma^2\right)^{-(1/2)} \exp\left(-\frac{1}{2}\frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)\right)$$

$$= -\frac{1}{2}\sum_{i=1}^{n}\left(\log\left(2\pi\sigma^2\right) + \frac{\left(y_i - x_i^T\beta\right)^2}{\sigma^2}\right)$$

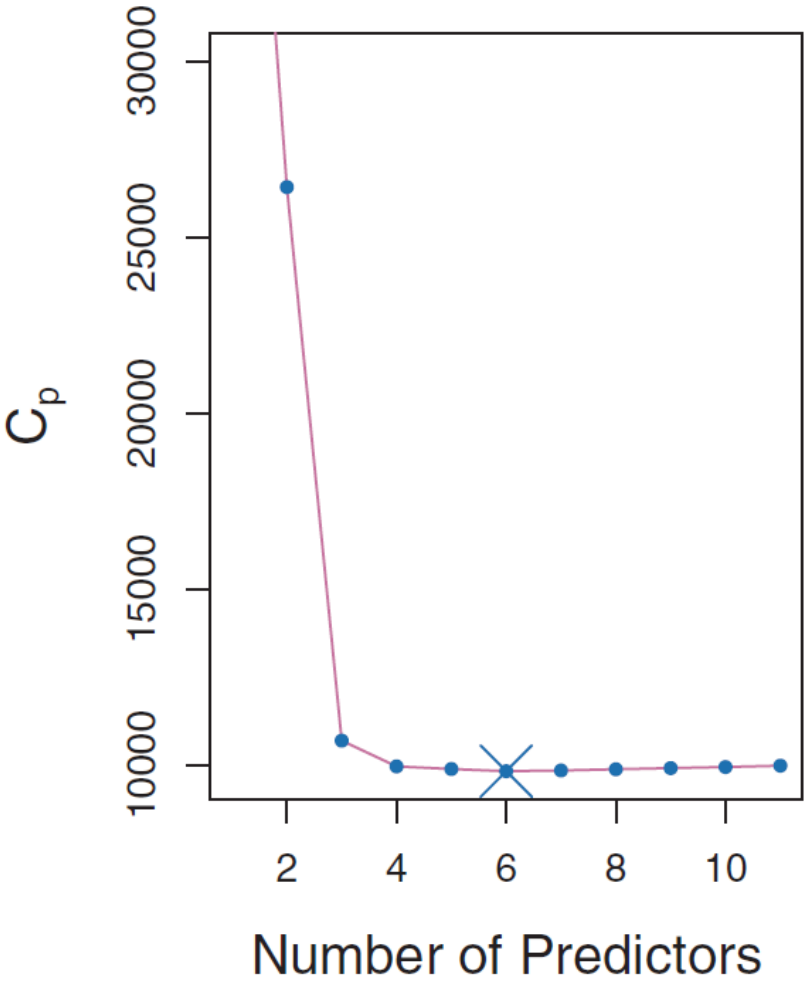
Adjusted R^2

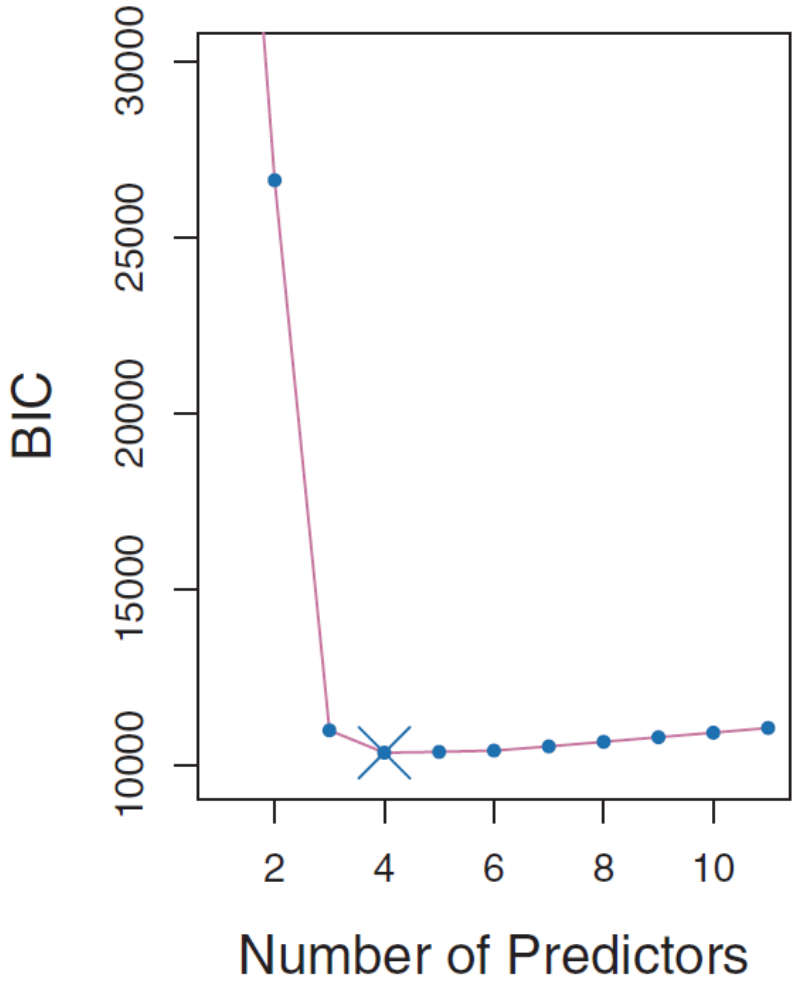
- Larger is better for Adjusted R^2

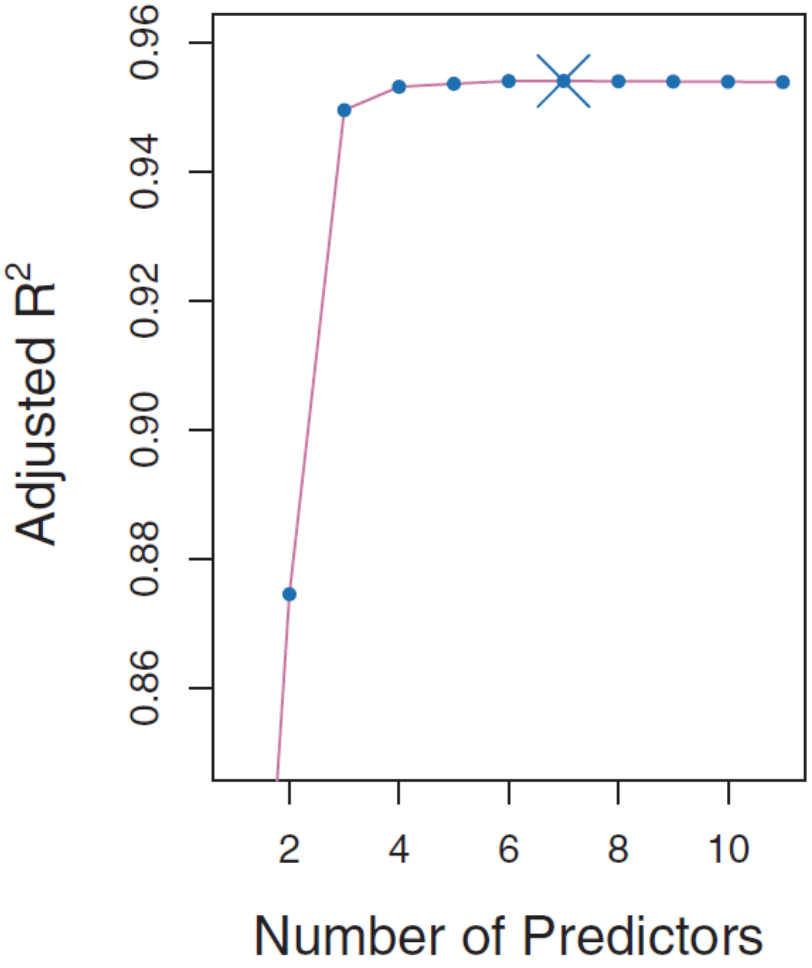
$$\text{Adjusted } R^2 = 1 - \frac{\text{RSS}/(n-d-1)}{\text{TSS}/(n-1)}$$

- Smaller is better for C_p, AIC, and BIC

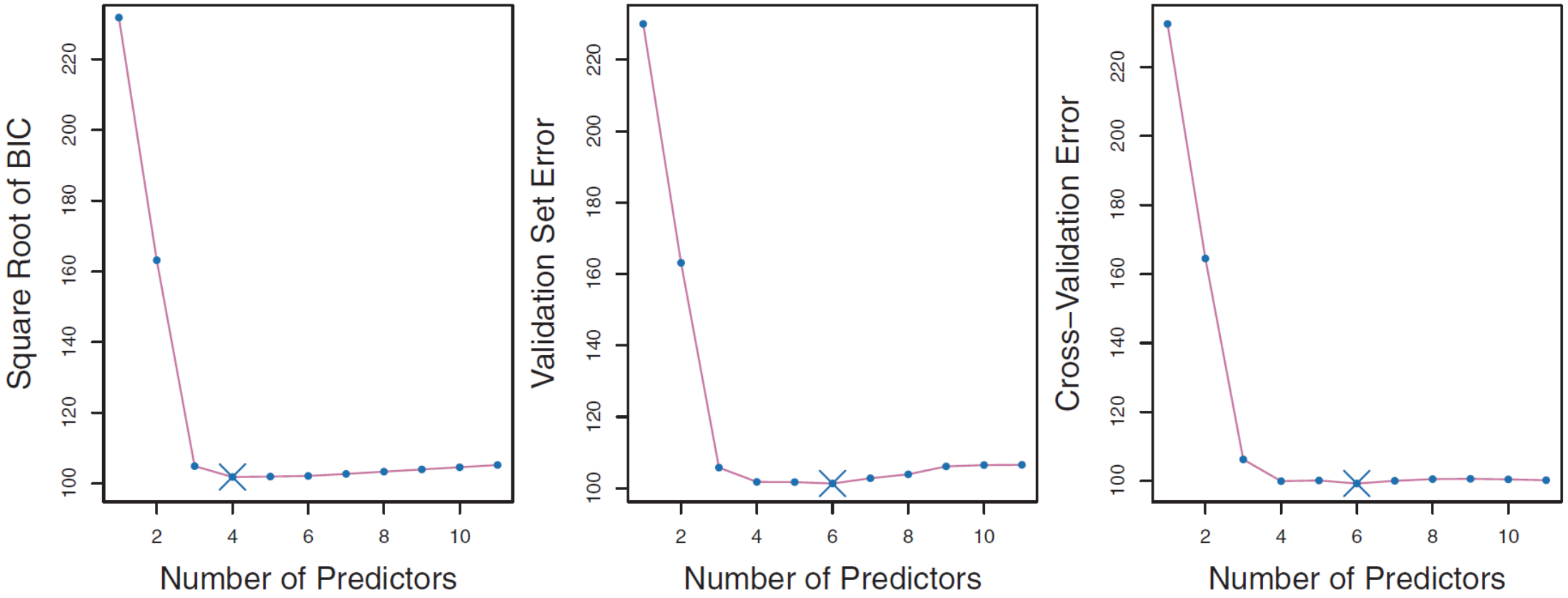
C_p versus BIC versus Adjusted R^2







BIC versus Validation versus Cross Validation



How many predictors would you use?

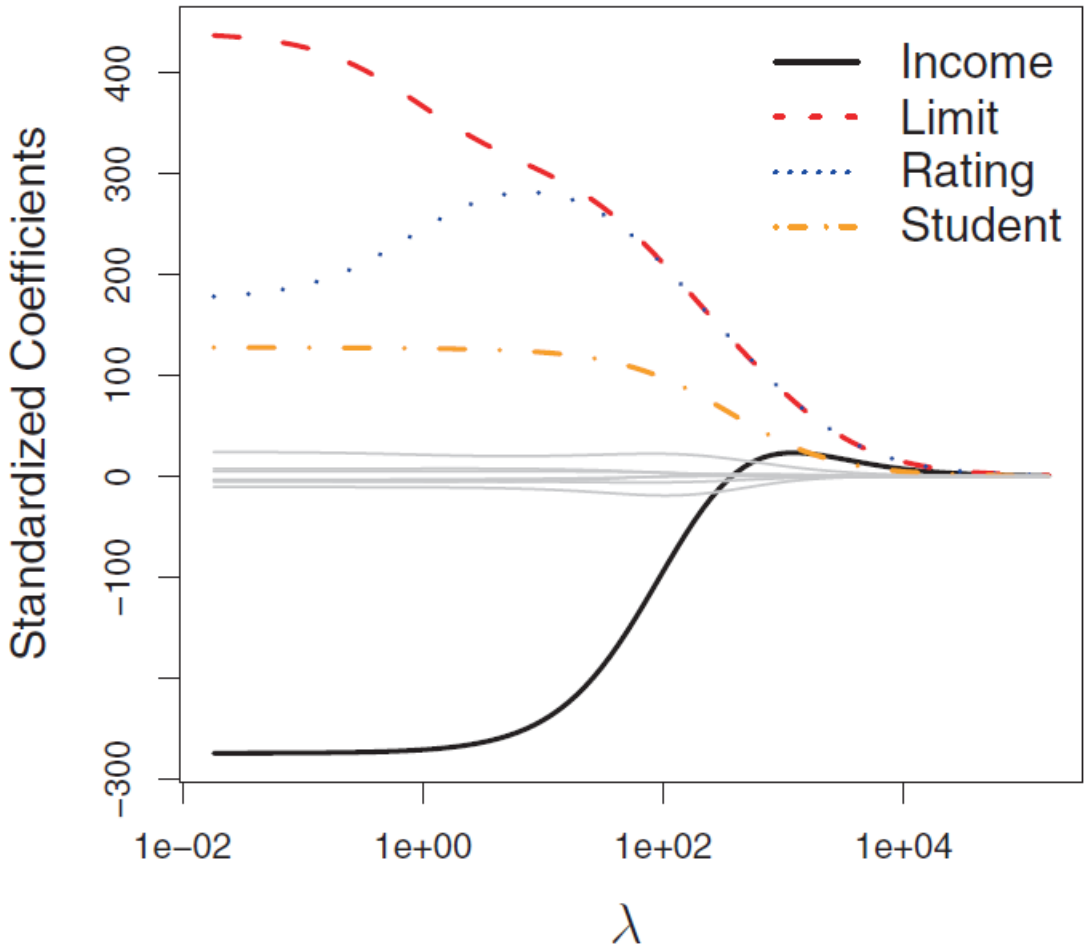
Ridge Regression

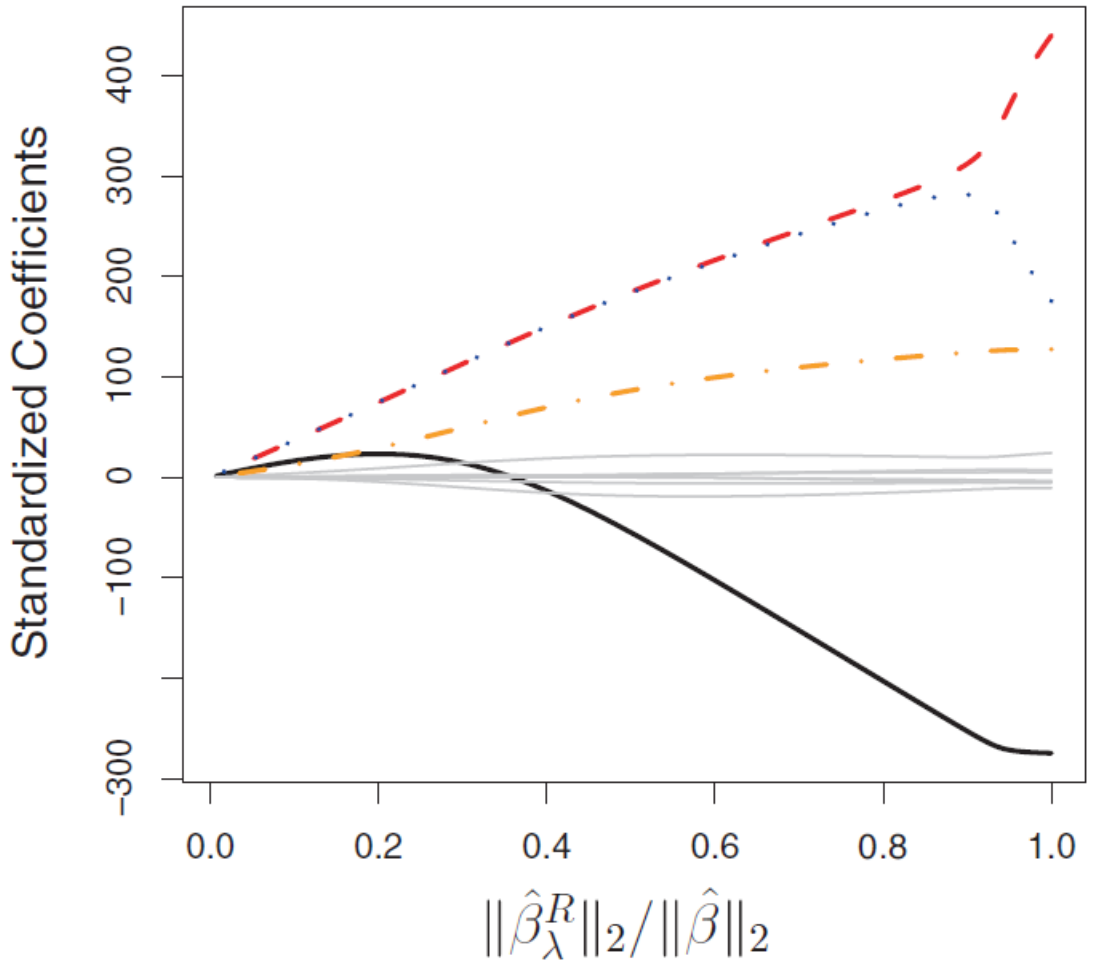
- The “lambda” (second) term is called a shrinkage (also known as regularization) penalty
- The “lambda” term is a tuning parameter
  - As lambda goes towards zero, the l2 (“el two”) norm of the regression coefficients gets larger and we get a least squares fit
  - As lambda goes towards infinity, the l2 norm of the regression coefficients gets smaller and we eventually get a null model (high bias; low variance)

$$\|\beta\|_2 = \sqrt{\sum_{j=1}^{p} {\beta_j}^2}$$

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = \mathrm{RSS} + \lambda\sum_{j=1}^{p}\beta_j^2$$

Ridge Regression for the Credit Data Set





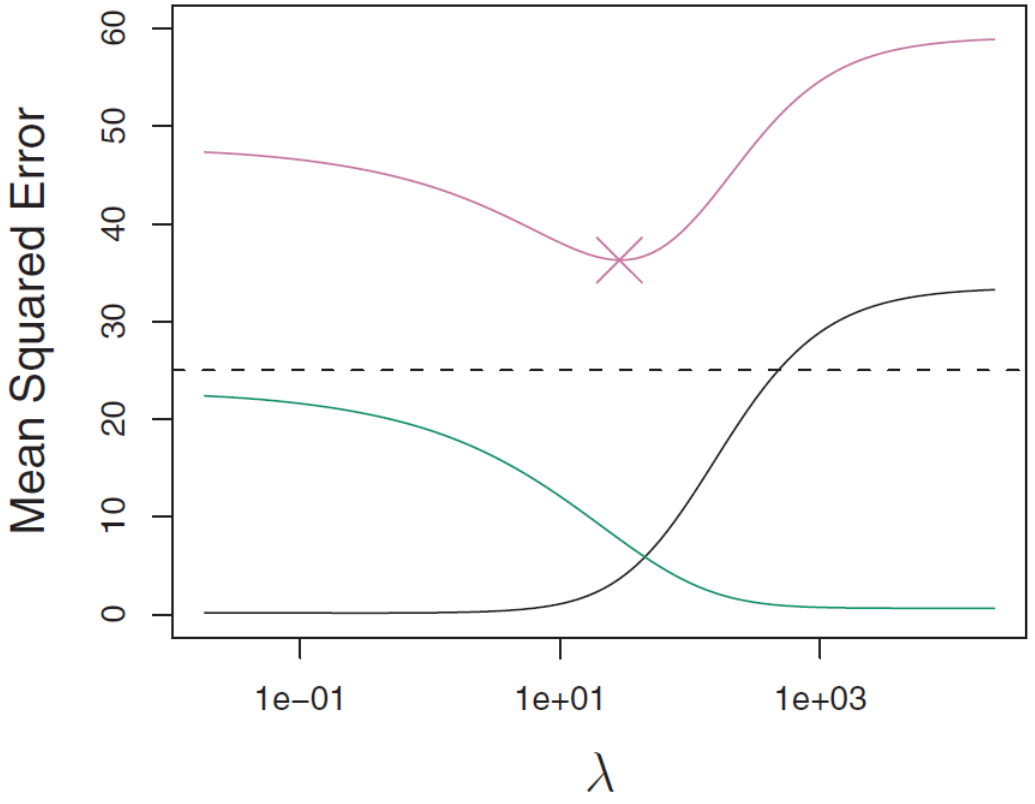
Standardizing (Scaling) the Predictors

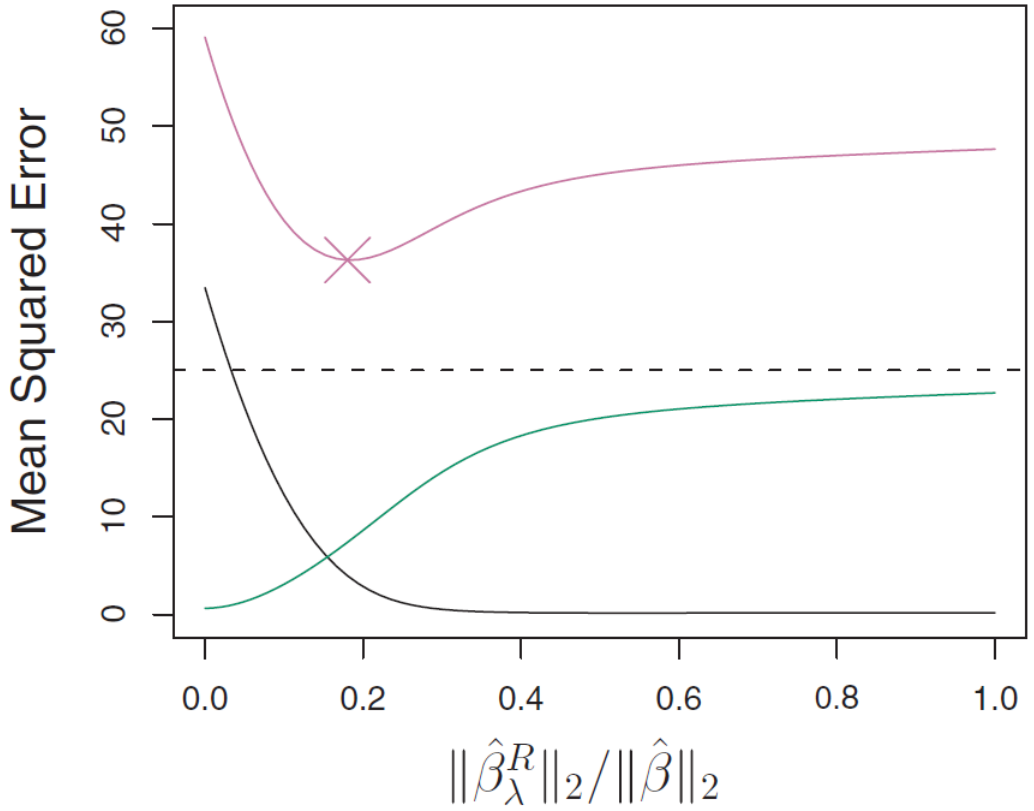
- The new unit of measure is the standard deviation for the predictor

$$\tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}}$$

- Folks often center the predictors as well, by subtracting the mean

MSE Decomposition for the Credit Data Set





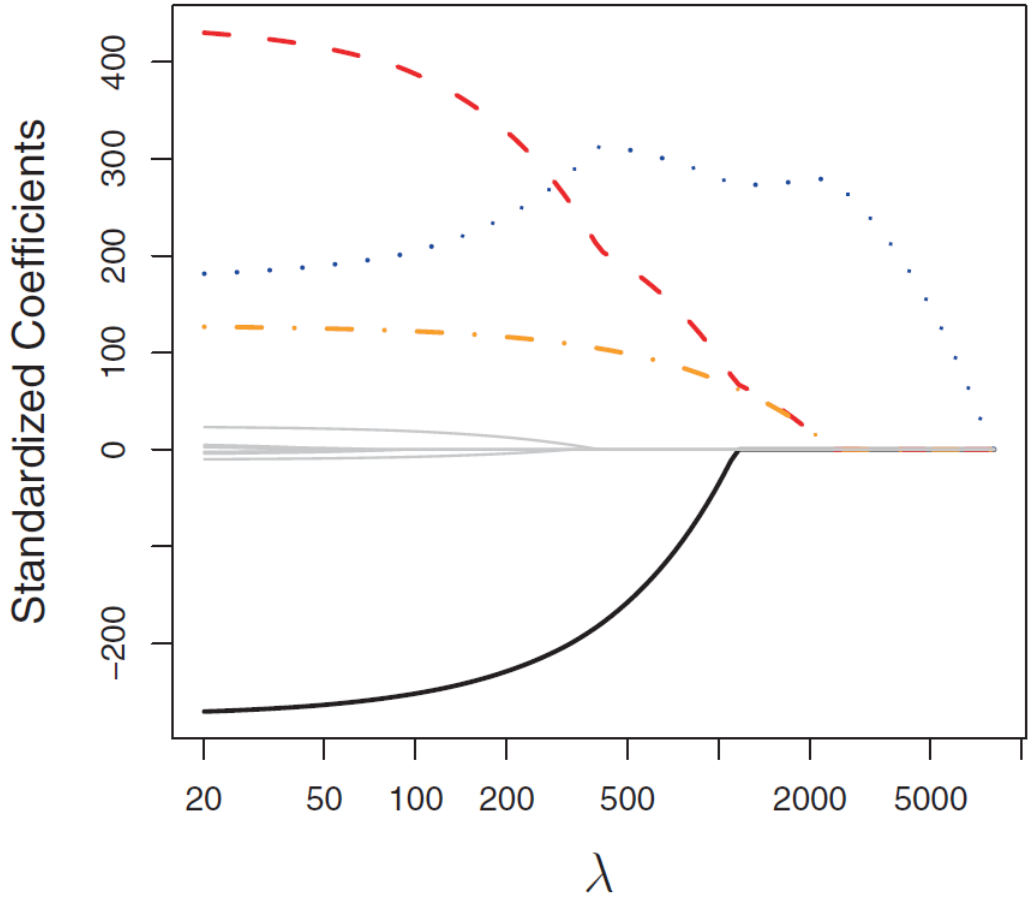
The Least Absolute Shrinkage and Selection Operator (LASSO)

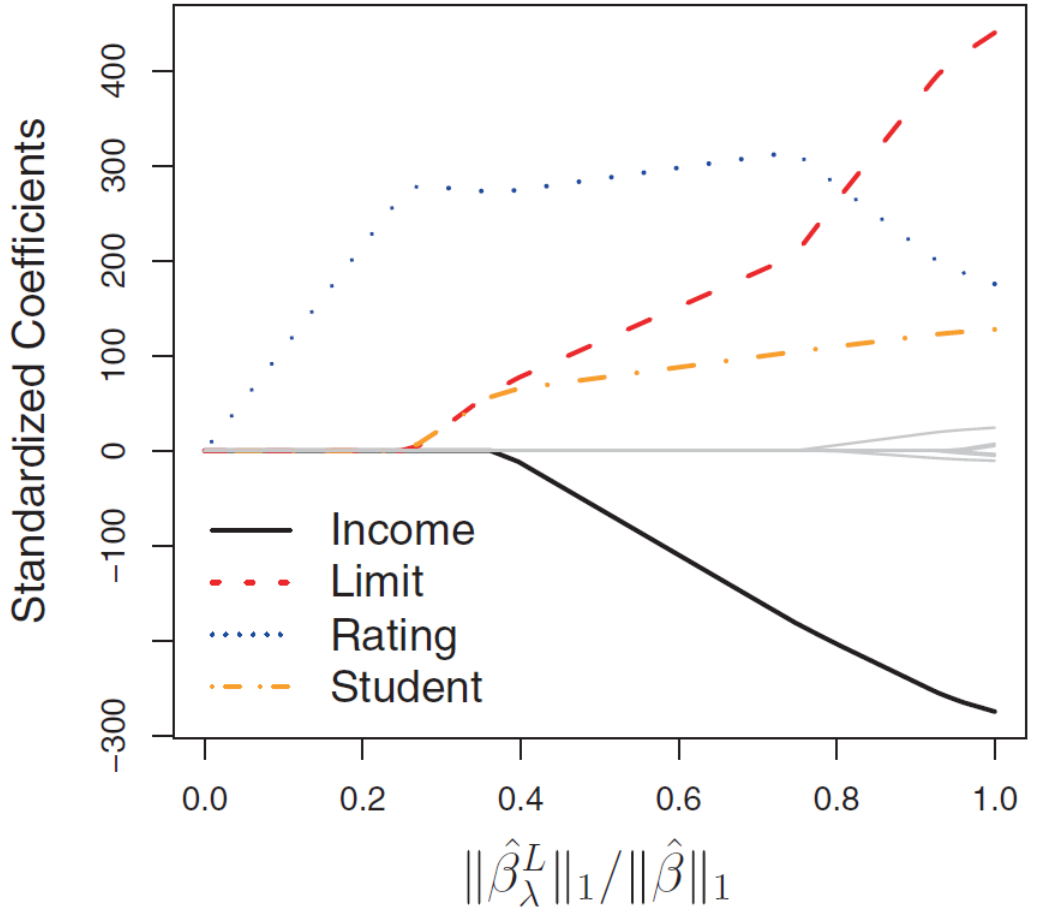
- The “lambda” (second) term is called a shrinkage (also known as regularization) penalty, because we’re using an l1 (“el 1”) norm to keep the regression coefficients small
- The “lambda” term is a tuning parameter
  - As lambda goes towards zero, the l1 (“el one”) norm of the regression coefficients gets larger and we get a least squares fit
  - As lambda goes towards infinity, the l1 norm of the regression coefficients gets smaller and we eventually get a null model (high bias; low variance)

$$\|\beta\|_1 = \sum |\beta_j|$$

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| = \mathrm{RSS} + \lambda \sum_{j=1}^{p} |\beta_j|$$

The Lasso for the Credit Data Set





Equivalent Formulation for Ridge Regression and the Lasso

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s$$

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s,$$

A Picture is Worth a Thousand Words

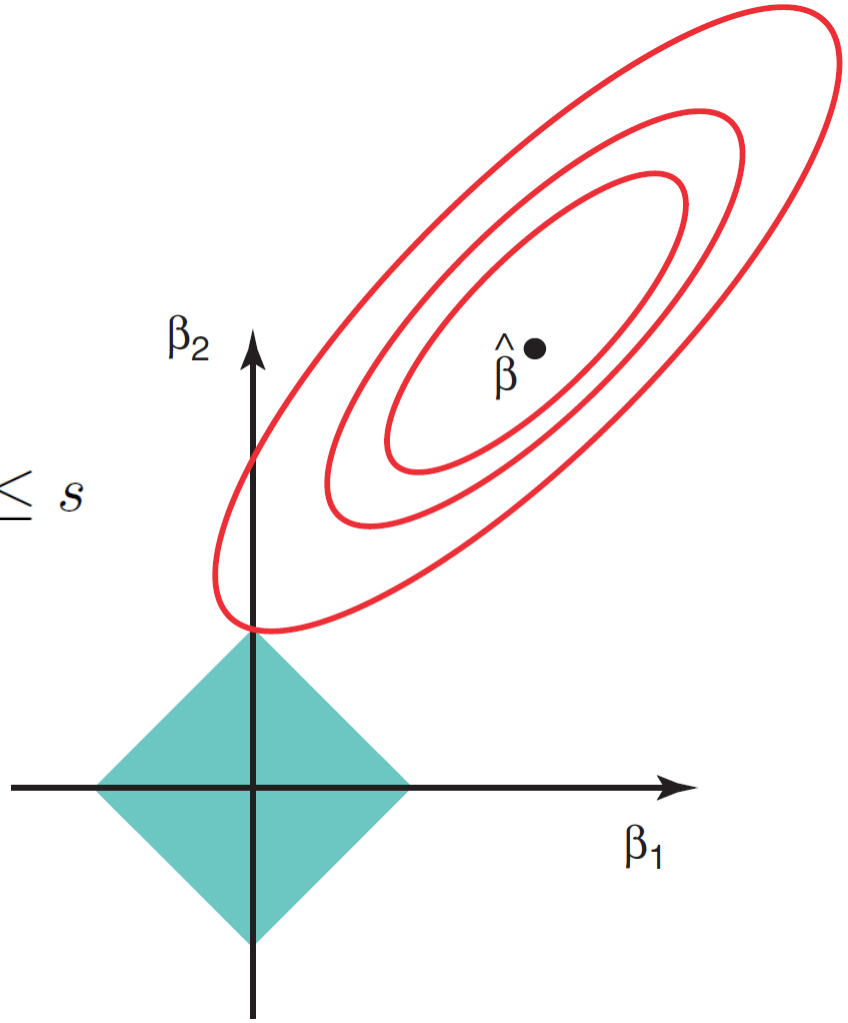
Blue marks the constrained search area

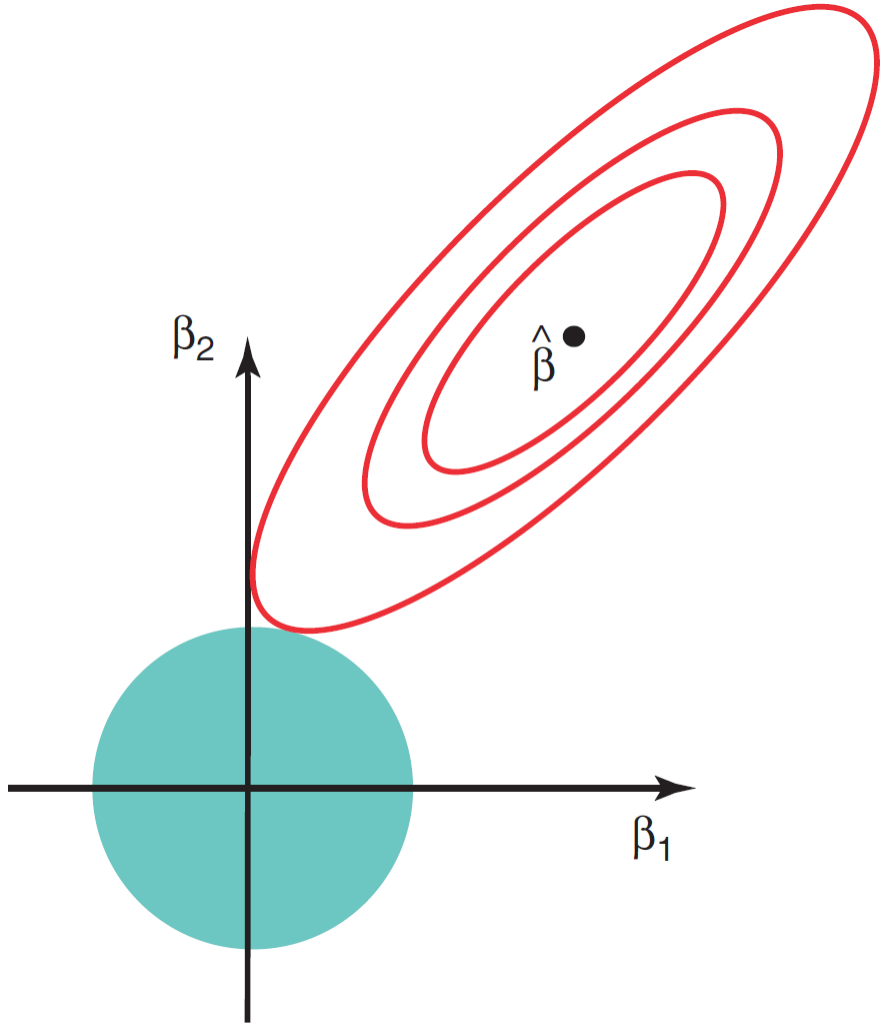
"Sometimes doing the right thing ain't doing the right thing" ☺

The Lasso

$$|\beta_1| + |\beta_2| \leq s$$

2D diamond
3D polyhedron
ND polytope



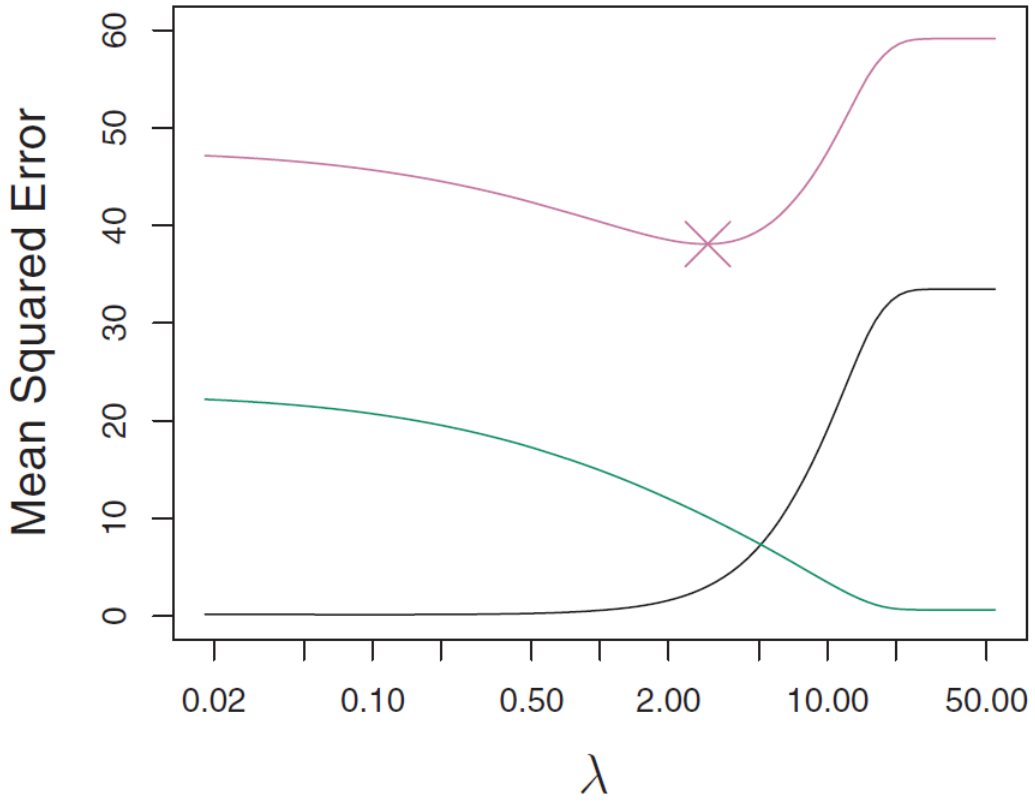


Ridge Regression

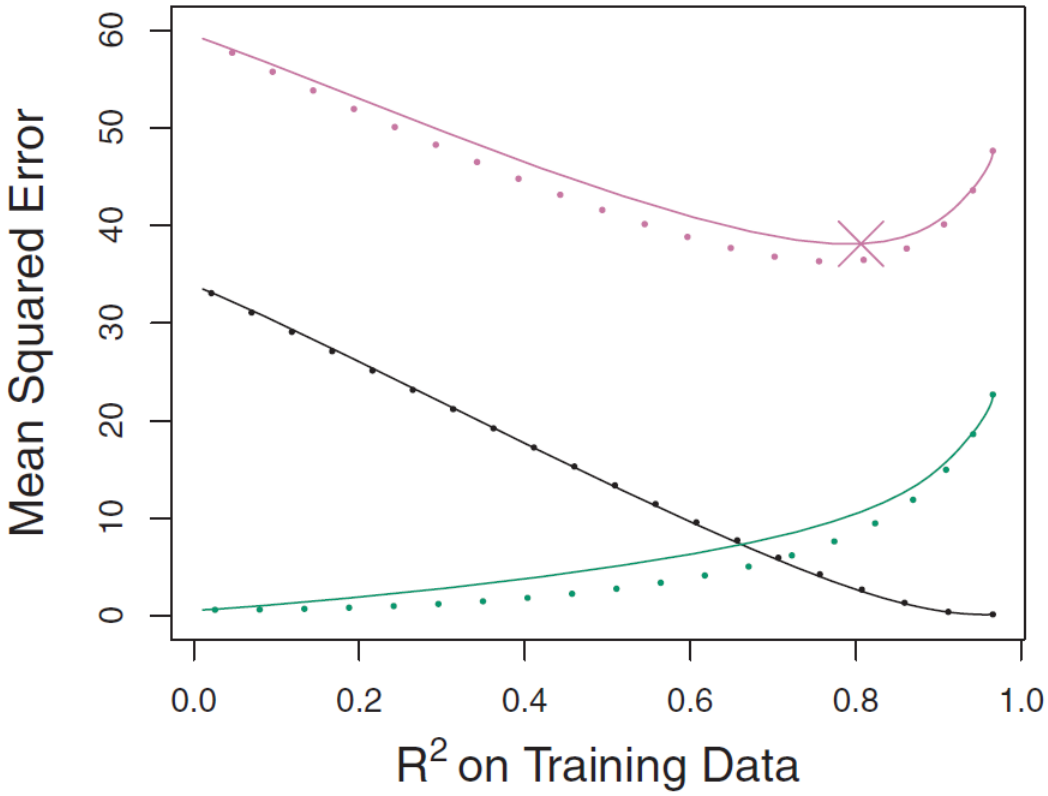
$$\beta_1^2 + \beta_2^2 \leq s$$

2D circle
3D sphere
ND hypersphere

Lasso versus Ridge Regression on Simulated Data: All 45 Features Related to the Response

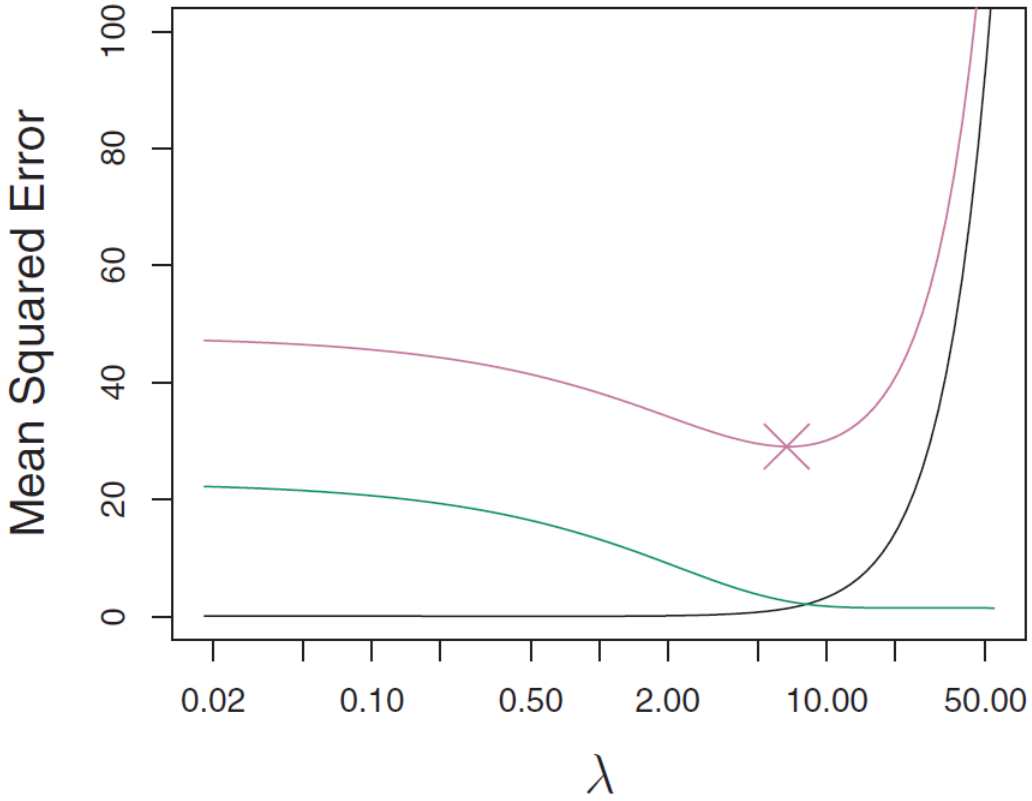


Lasso: MSE Decomposition

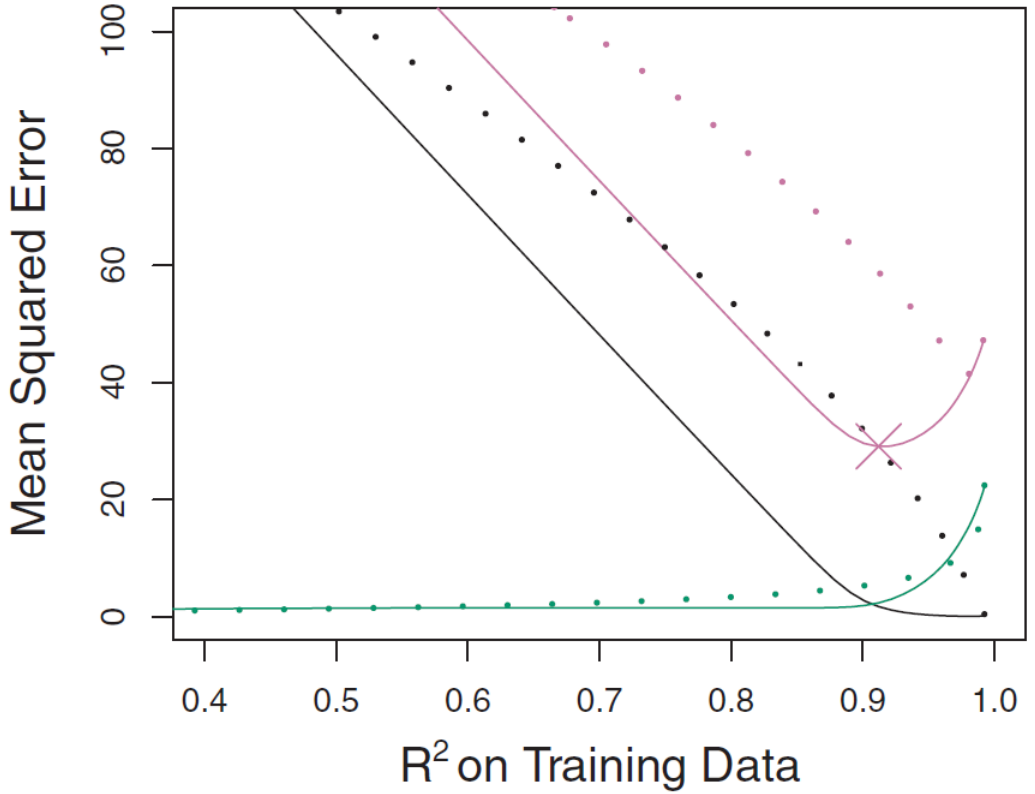


Lasso: solid
Ridge regression: dotted

Lasso versus Ridge Regression on Simulated Data: Only 2 out of 45 Features Related to the Response



Lasso: MSE Decomposition



Lasso: solid
Ridge regression: dotted

A Simple Special Case for Ridge Regression and the Lasso

n = p

X is the identity matrix

least squares

$$\sum_{j=1}^{p}(y_j - \beta_j)^2 \qquad \hat{\beta}_j = y_j$$

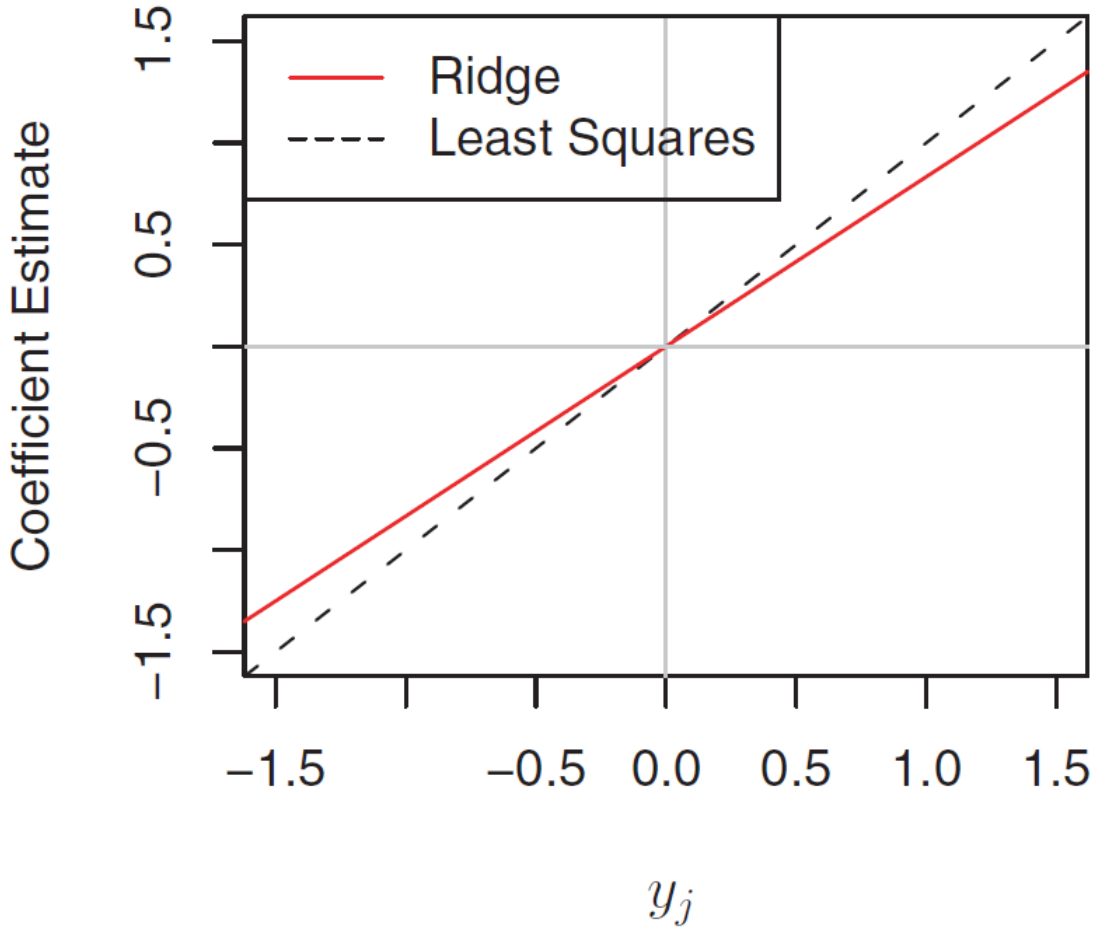
ridge regression

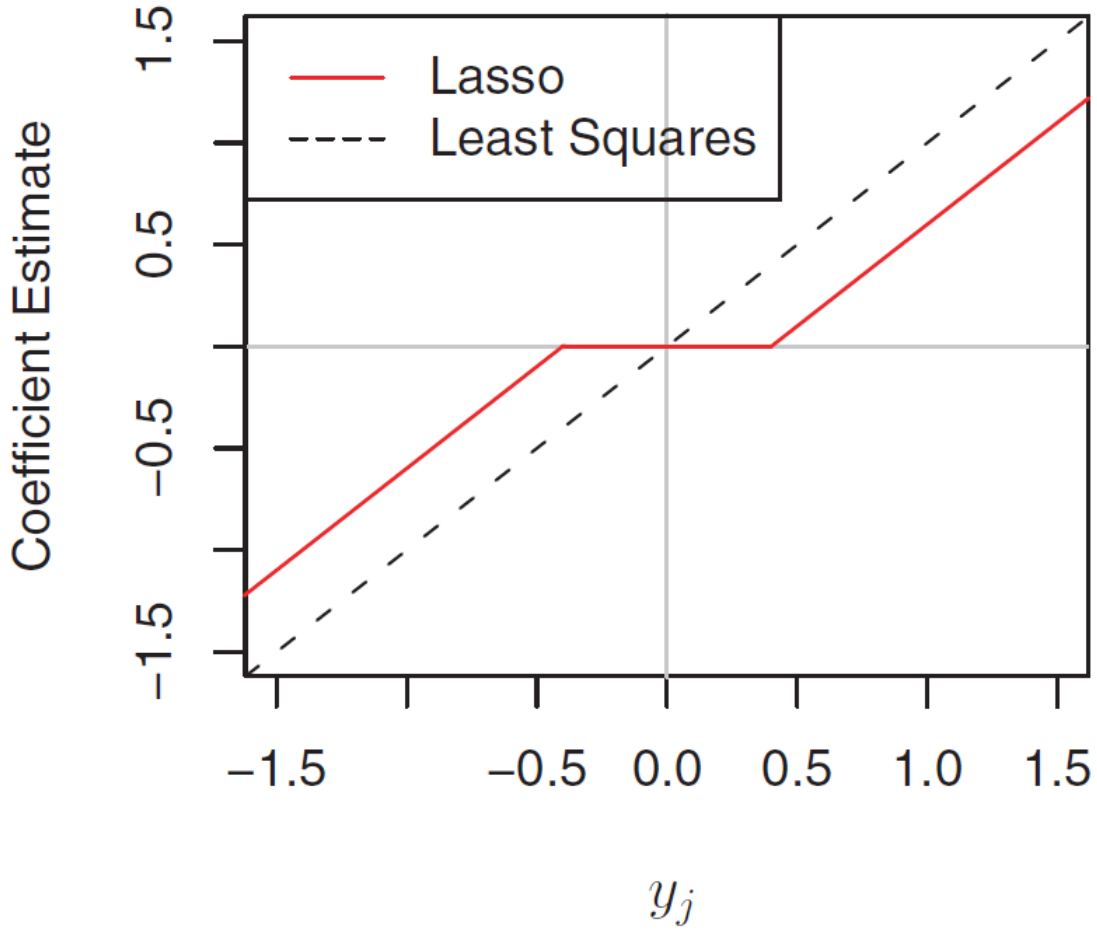
$$\sum_{j=1}^{p}(y_j - \beta_j)^2 + \lambda\sum_{j=1}^{p}\beta_j^2 \qquad \hat{\beta}_j^R = y_j/(1+\lambda)$$

lasso

$$\sum_{j=1}^{p}(y_j - \beta_j)^2 + \lambda\sum_{j=1}^{p}|\beta_j| \qquad \hat{\beta}_j^L = \begin{cases} y_j - \lambda/2 & \text{if } y_j > \lambda/2 \\ y_j + \lambda/2 & \text{if } y_j < -\lambda/2 \\ 0 & \text{if } |y_j| \le \lambda/2 \end{cases}$$

A Simple Special Case for Ridge Regression and the Lasso: the Picture Version ☺



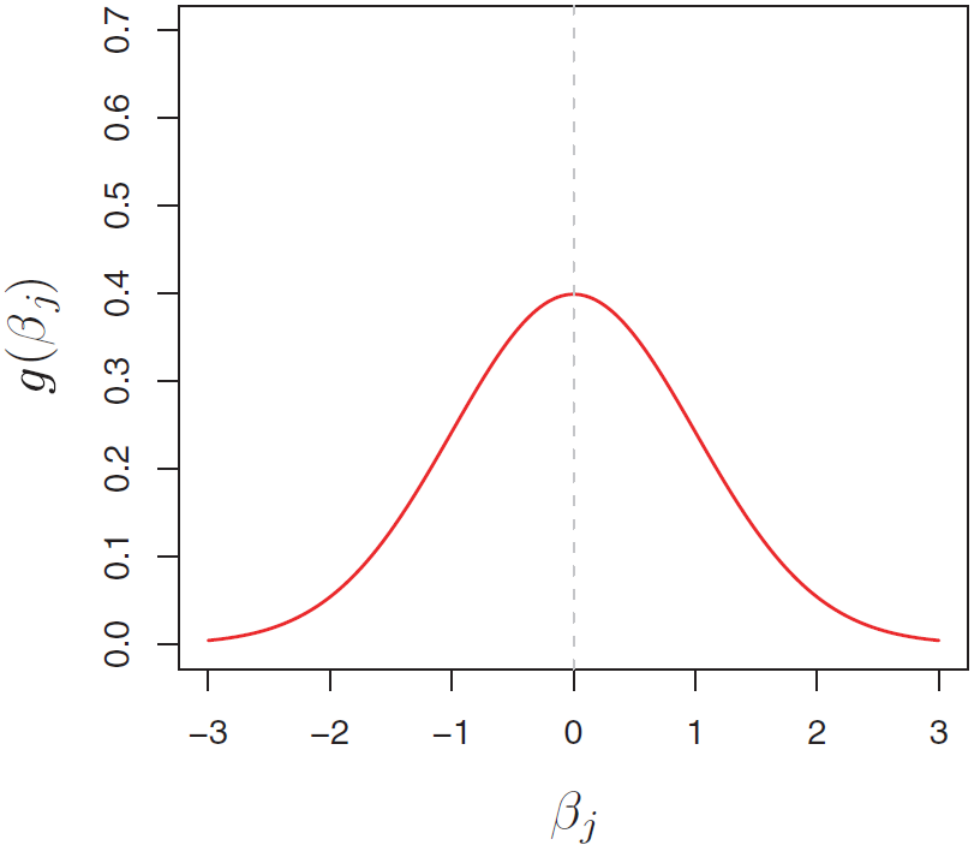


Both ridge regression and the lasso are shrinking the coefficients; but the lasso performs feature selection

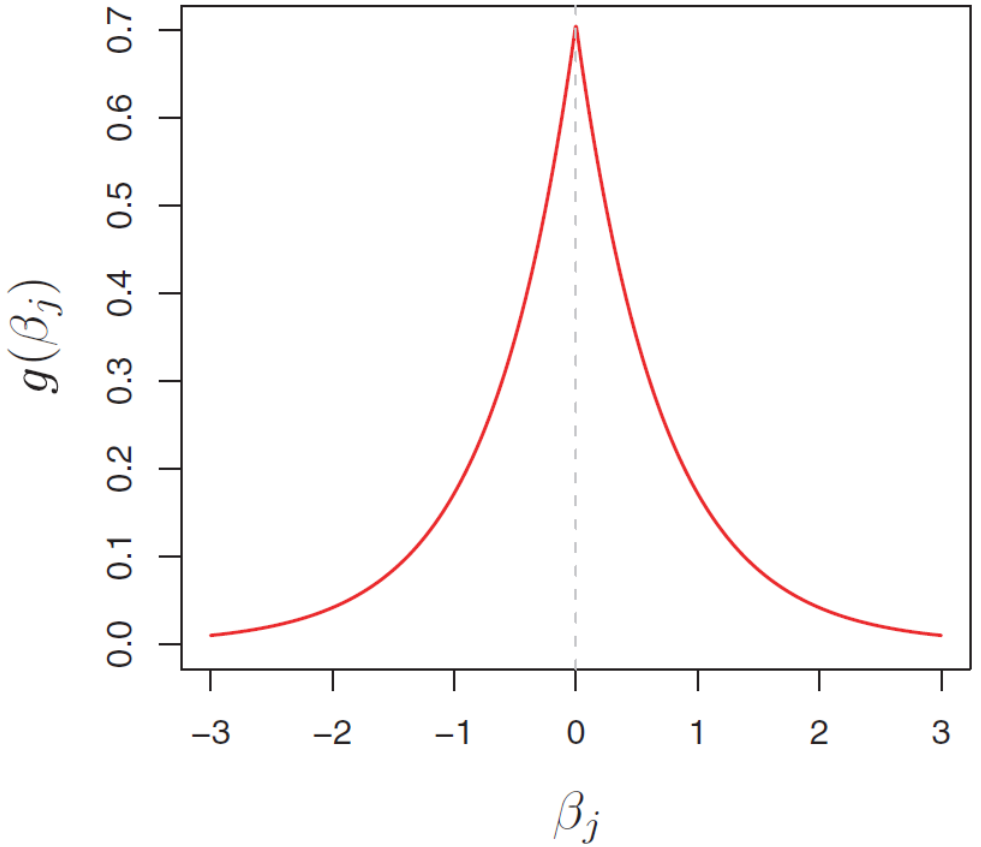
Bayesian Interpretation for Ridge Regression and the Lasso

$$p(\beta|X,Y) \propto f(Y|X,\beta)p(\beta|X) = f(Y|X,\beta)p(\beta)$$

Think Squared Error



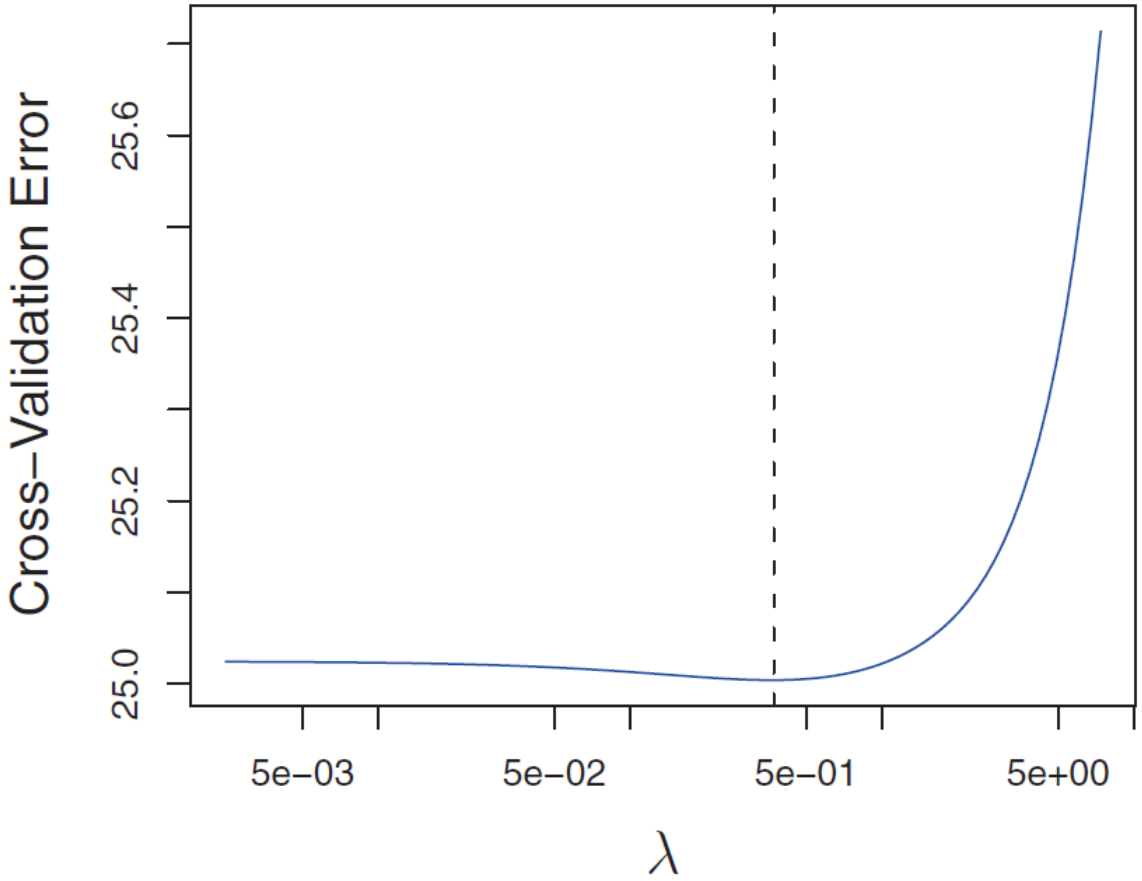
Ridge regression can be viewed as using a prior that has a Gaussian distribution with mode = 0

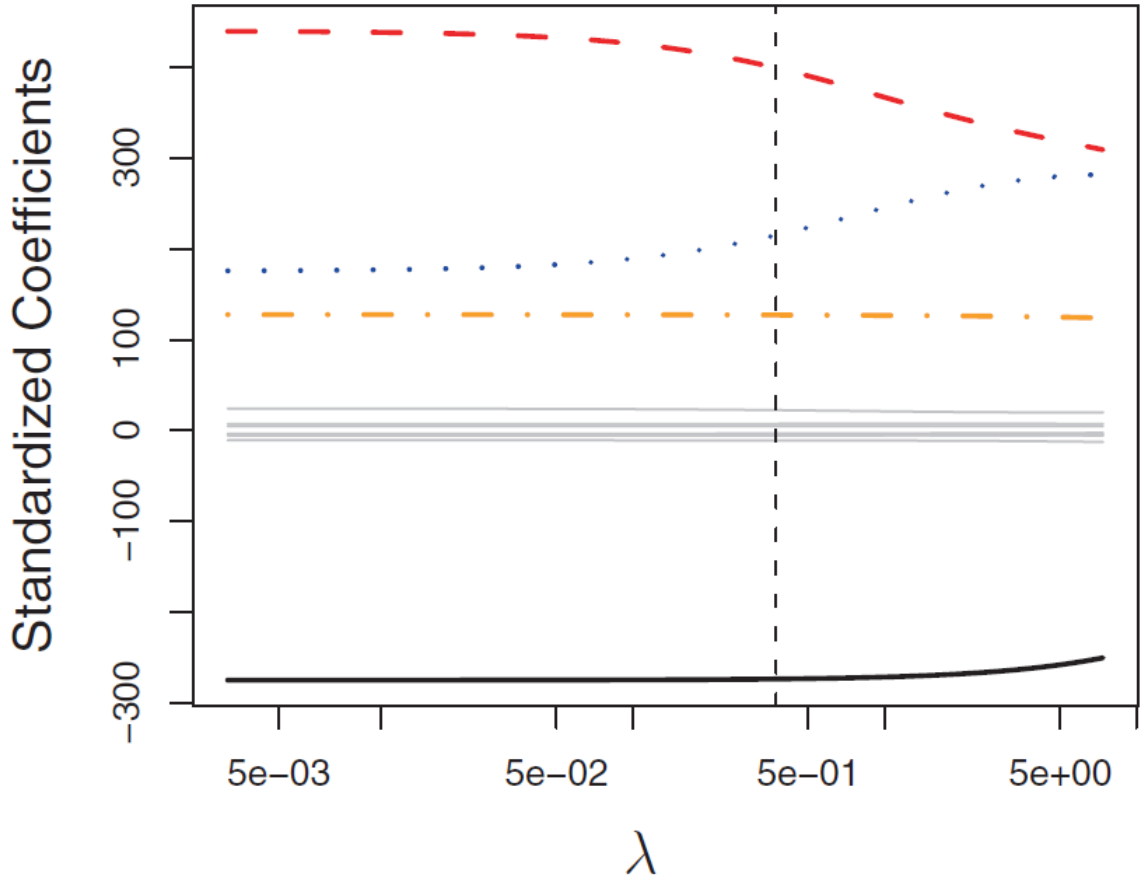


The lasso can be viewed as using a prior that has a Laplacian distribution with mode = 0

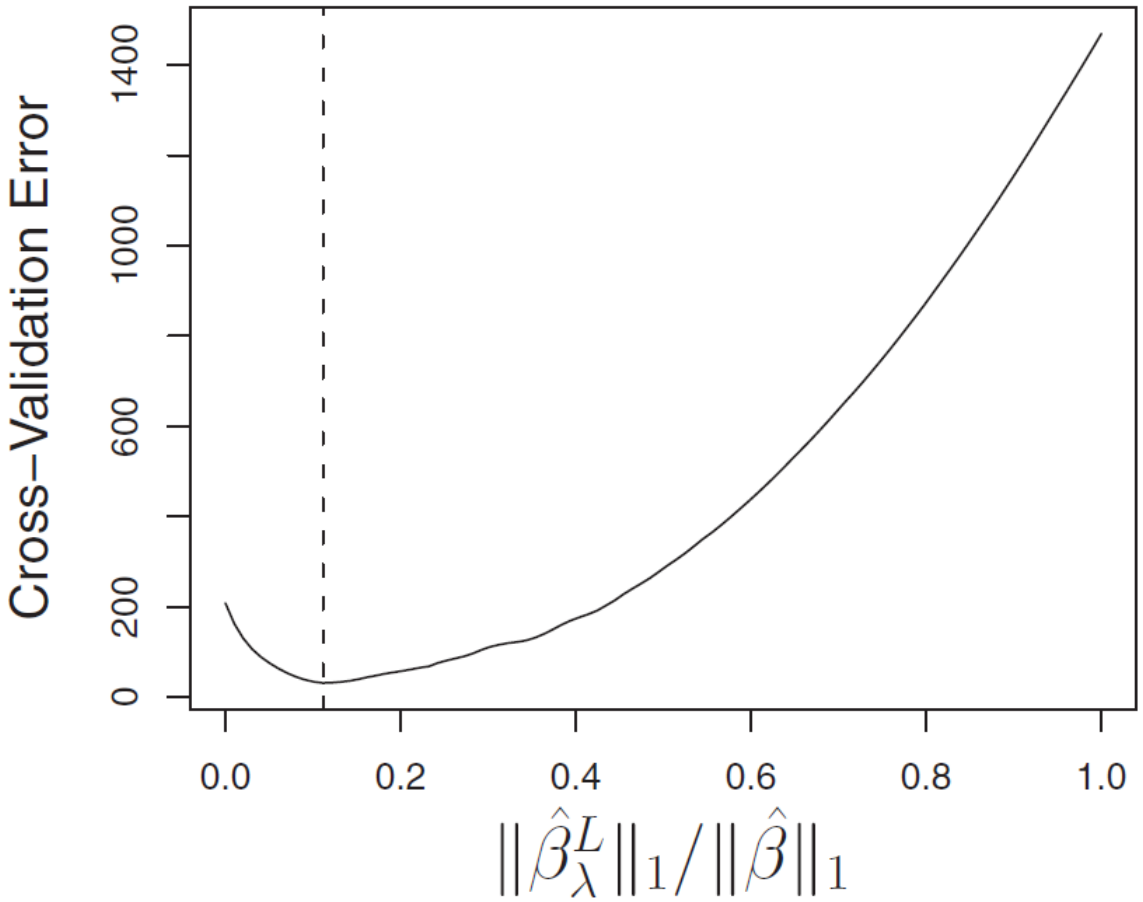
Think Absolute Error

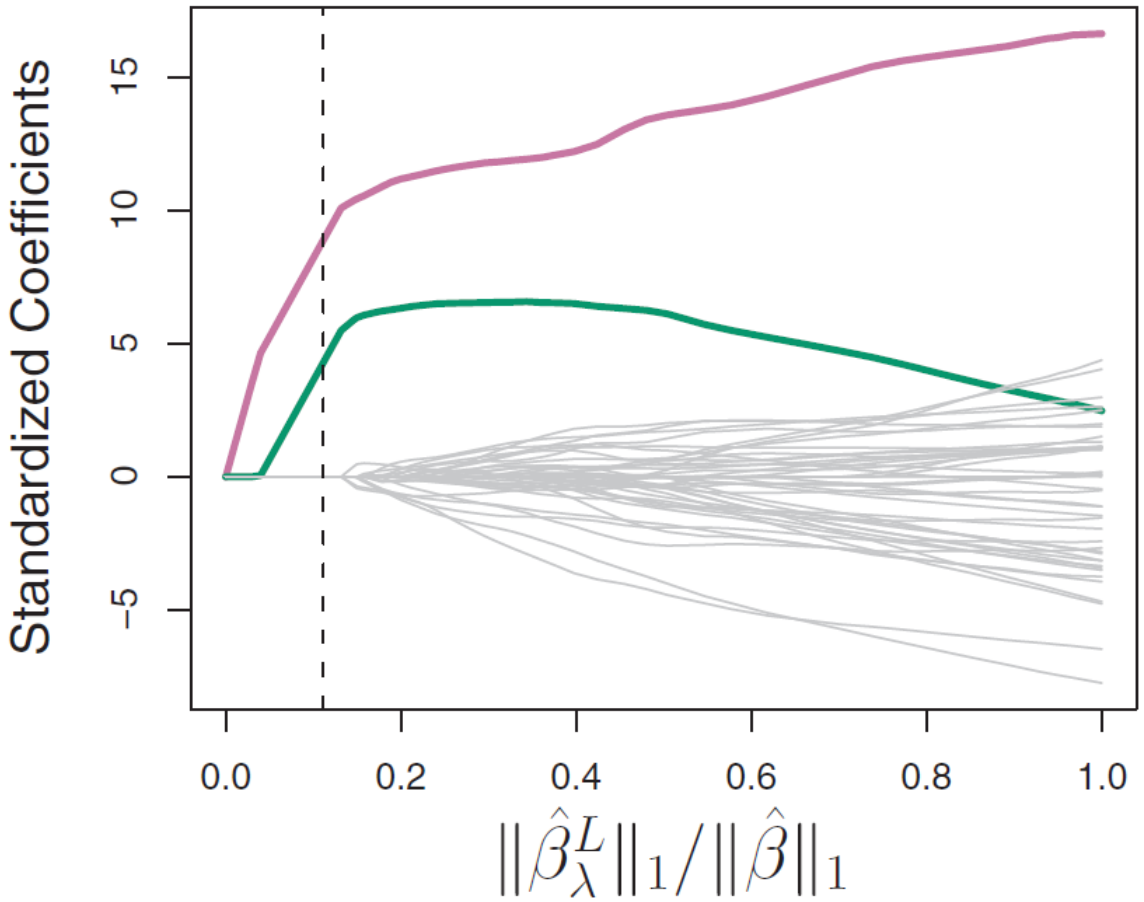
Selecting the Tuning Parameter: Ridge Regression for the Credit Data Set





Selecting the Tuning Parameter: the Lasso for the 2/45 Simulated Data Set





Dimensionality Reduction

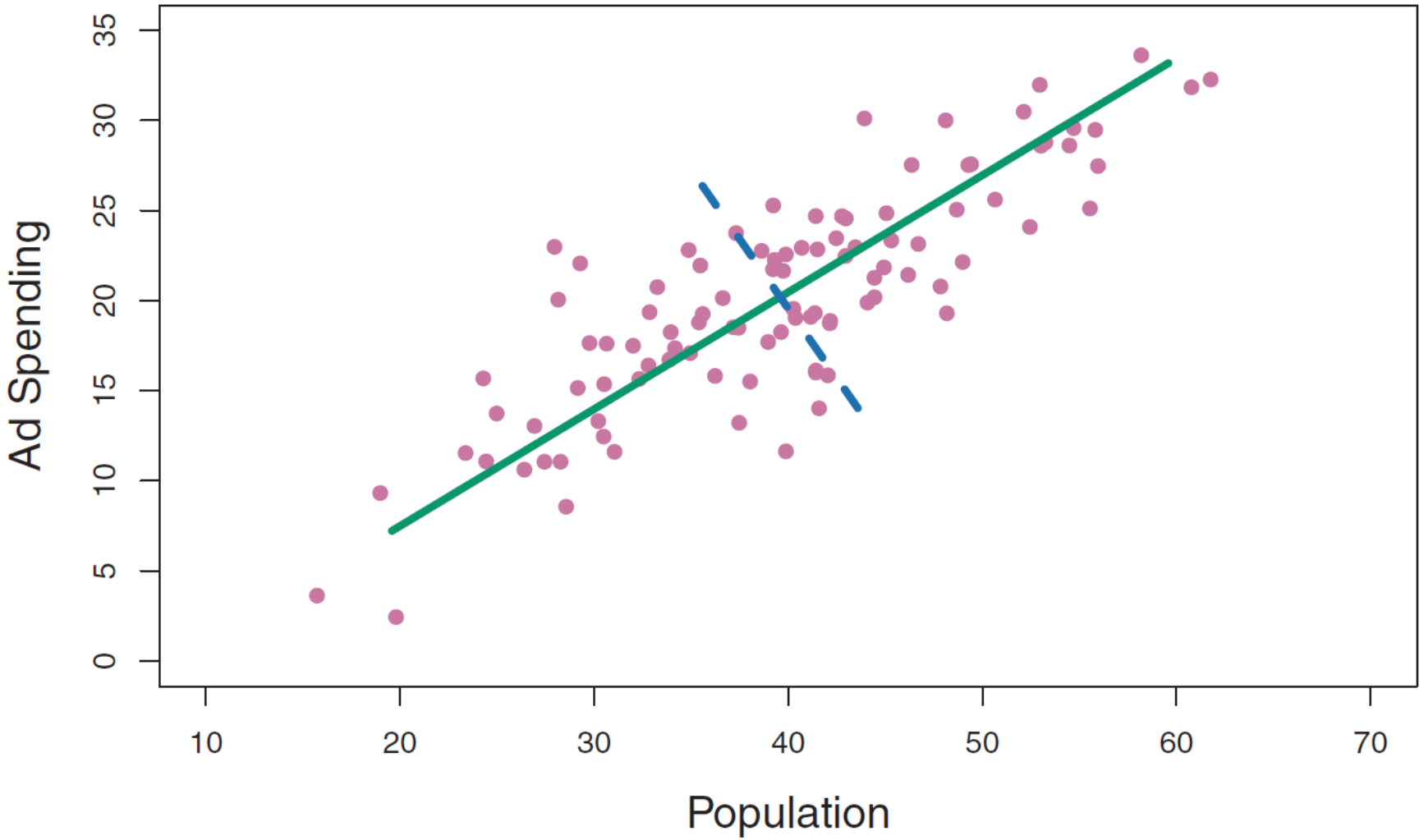
There will be “m” new predictors (replacing the old predictors), where each new predictor is a linear combination of the “p” old predictors [m < p]

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i, \quad i = 1, \ldots, n$$

$$\sum_{m=1}^{M} \theta_m z_{im} = \sum_{m=1}^{M} \theta_m \sum_{j=1}^{p} \phi_{jm} x_{ij} = \sum_{j=1}^{p} \sum_{m=1}^{M} \theta_m \phi_{jm} x_{ij} = \sum_{j=1}^{p} \beta_j x_{ij}$$

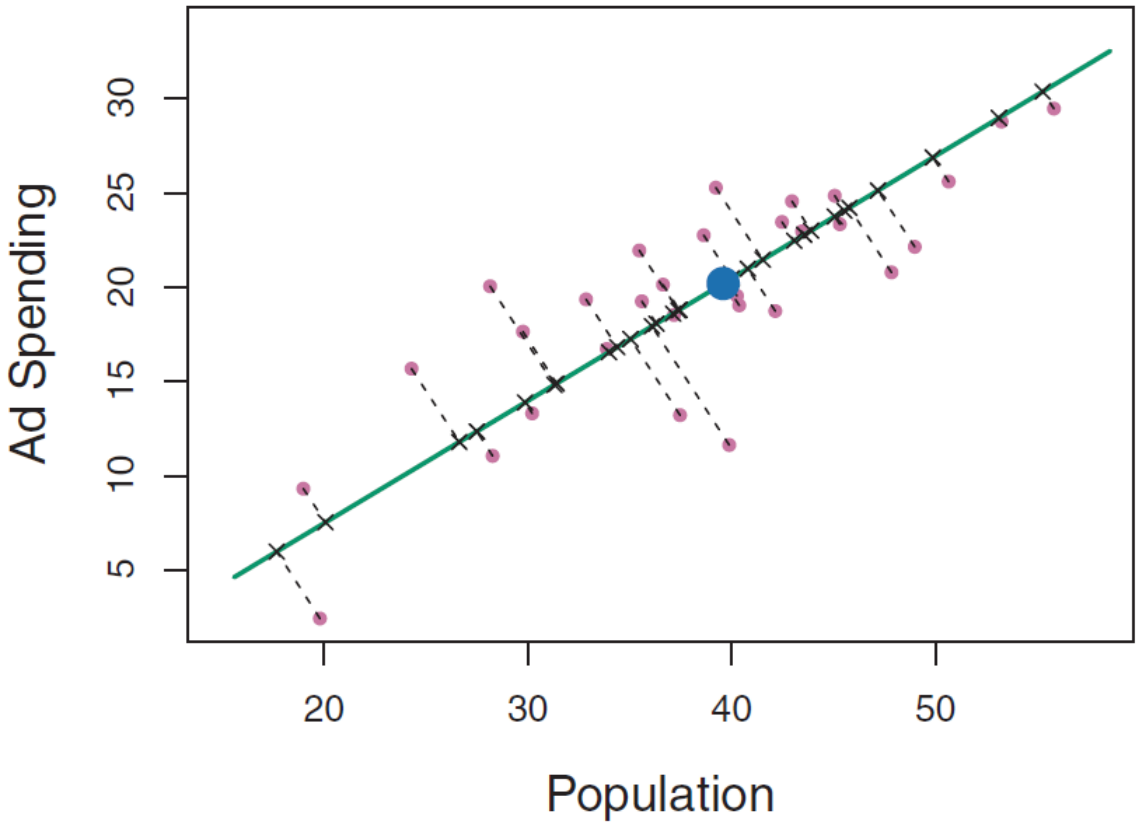
Principal Components Analysis

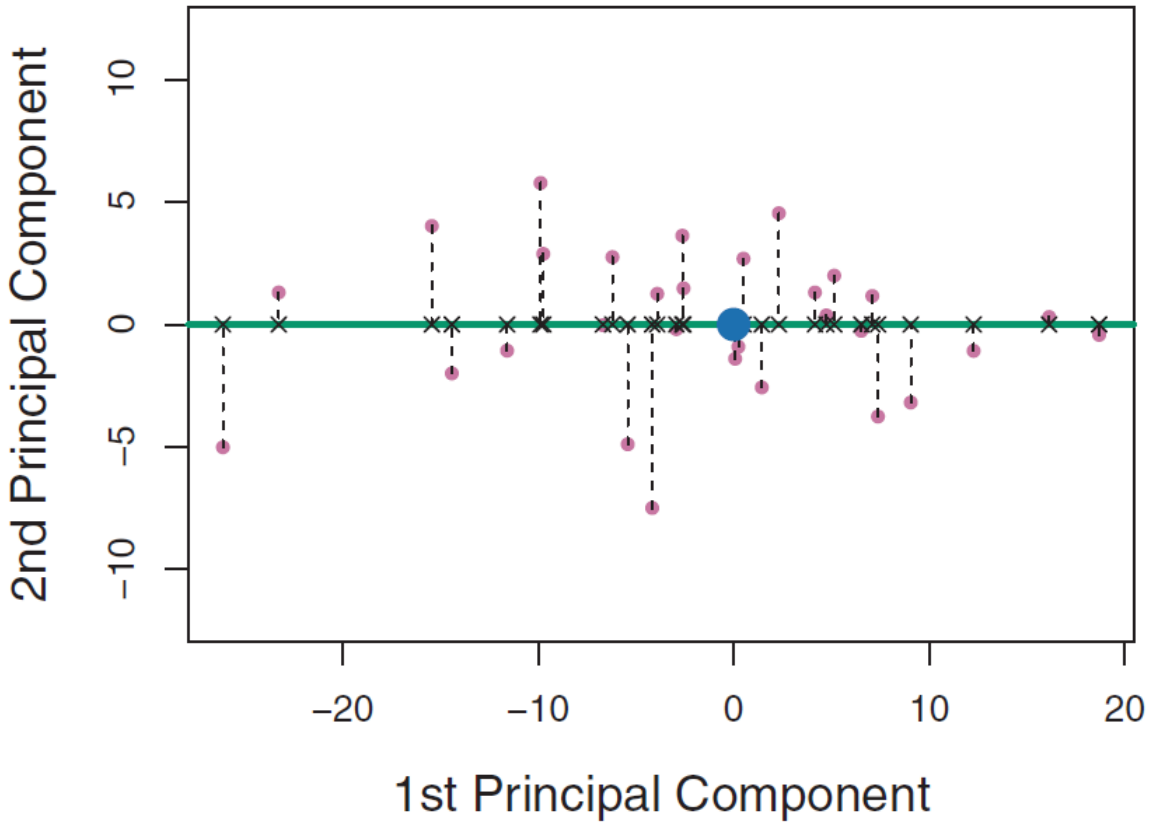


$$z_{i1} = 0.839 \times (\text{pop}_i - \overline{\text{pop}}) + 0.544 \times (\text{ad}_i - \overline{\text{ad}})$$

- The solid green line is the first principal component
  - the new axis for the first feature
  - the direction of maximum variance
- The dashed blue line is the second principal component
  - the new axis for the second feature [wait: what?!]
- The hope is that we are removing noise

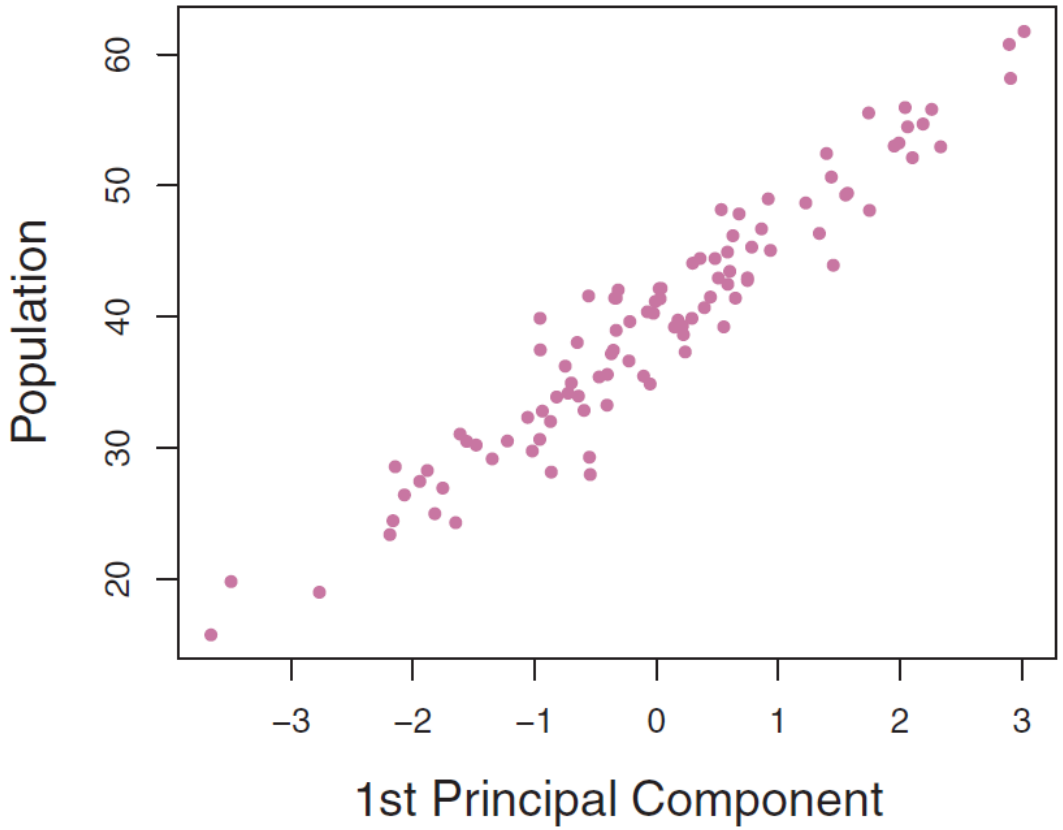
Example Projection

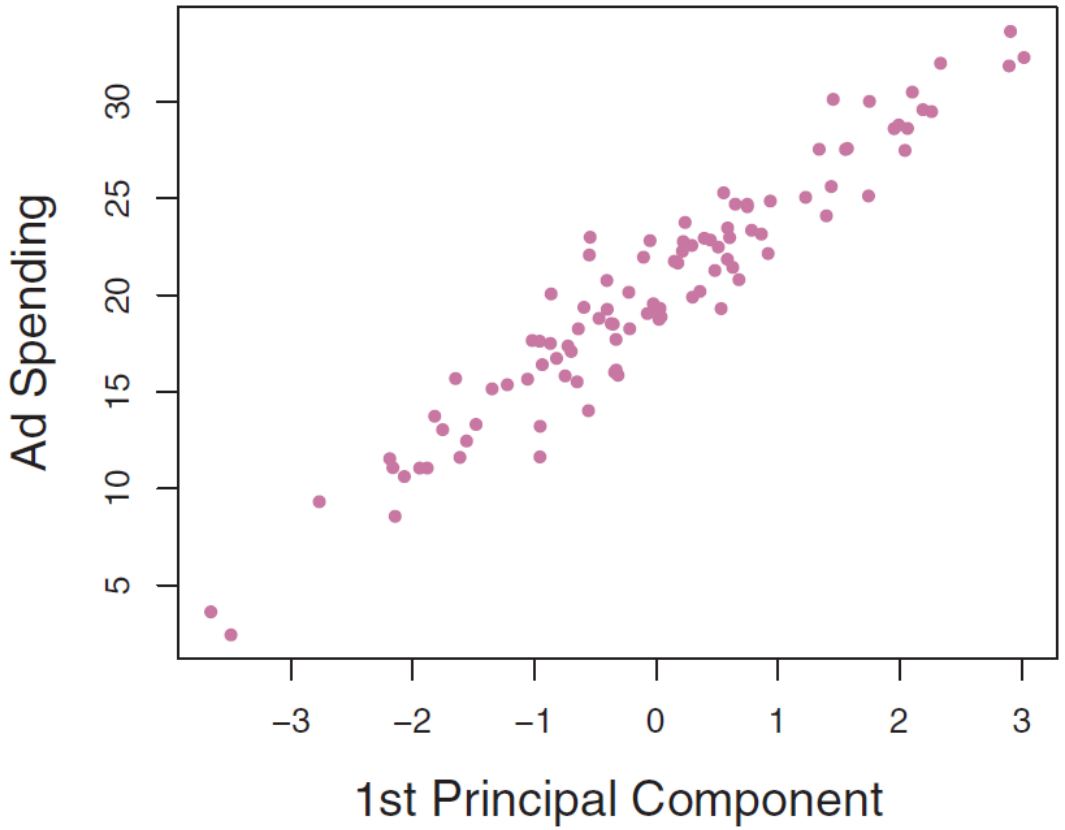




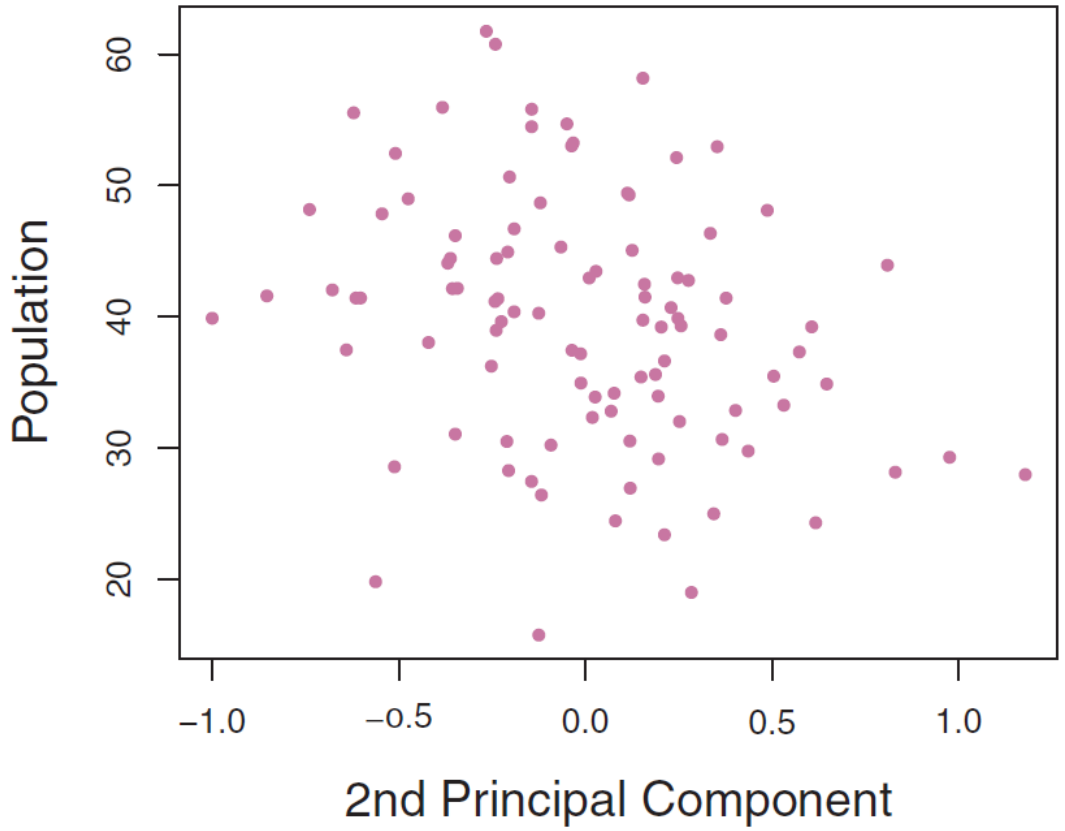
We're minimizing the squared distance to the new axis

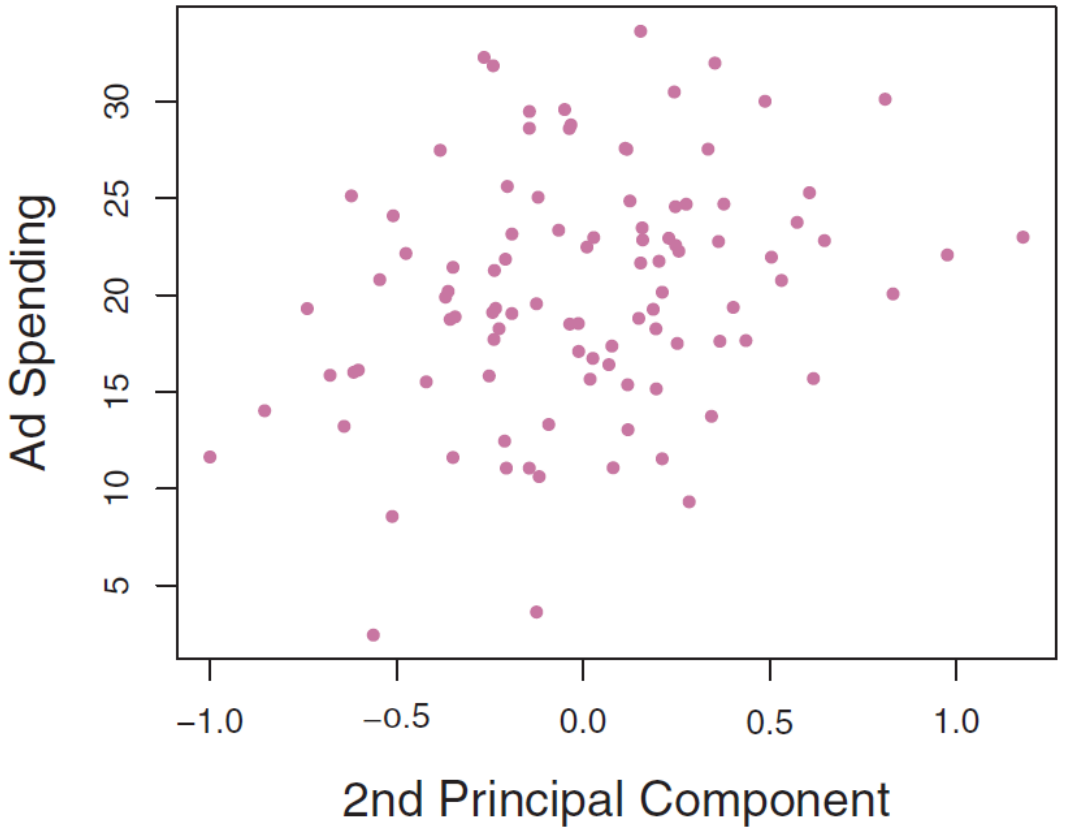
Relationship of the Old Predictors to the First New Predictor: Strong Relationships





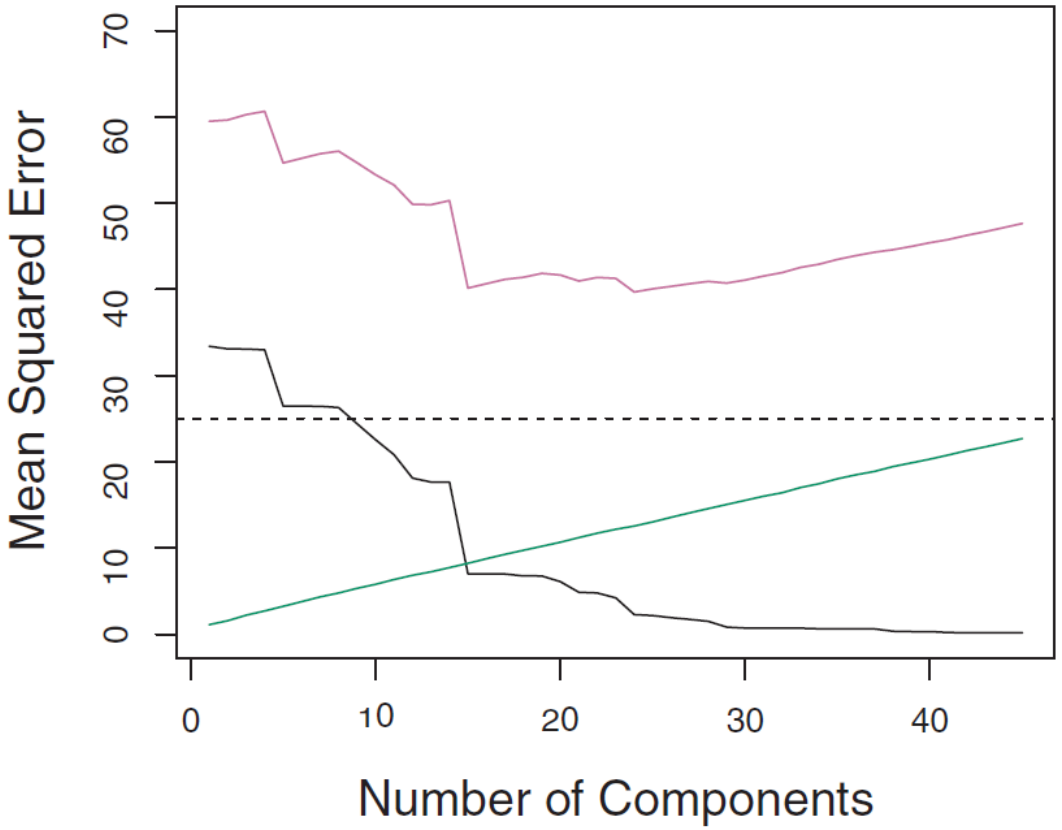
Relationship of the Old Predictors to the Second New Predictor: Weak Relationships

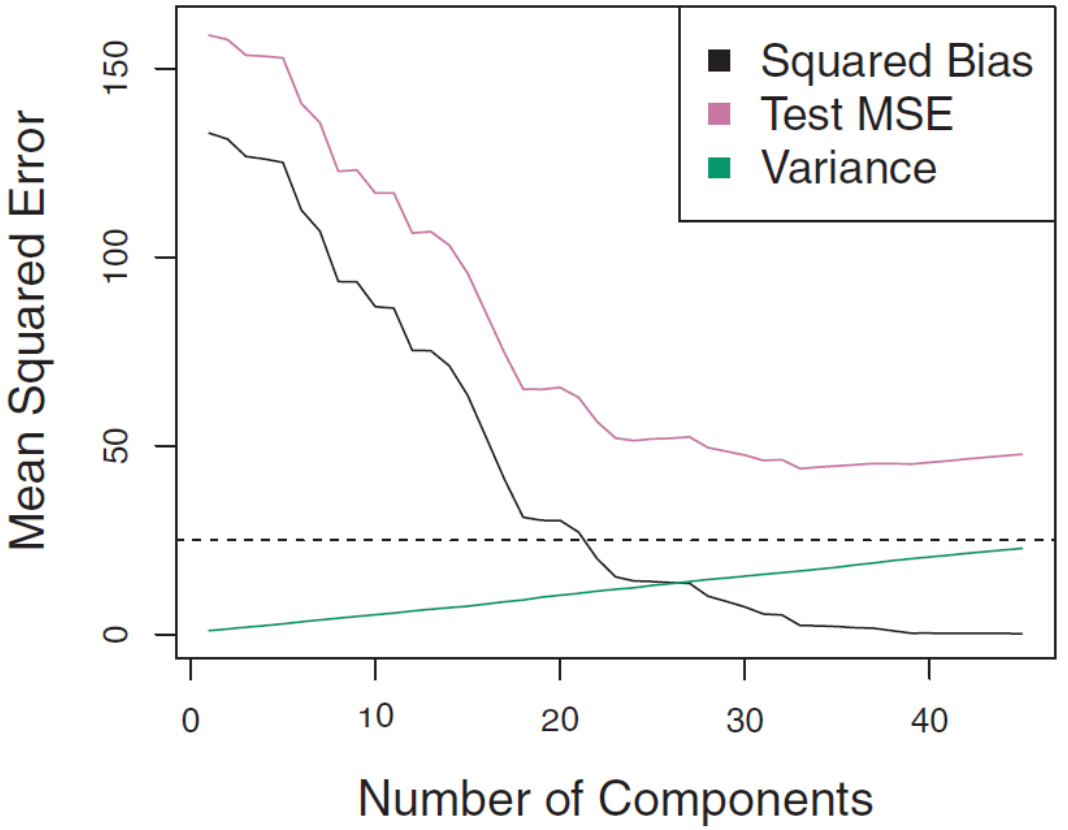




Principal Components Regression on the 45 Predictor Simulations

Maybe the two panels have been flipped in Figure 6.18? Test MSE being lower on the right would be consistent with all 45 predictors being related to the response

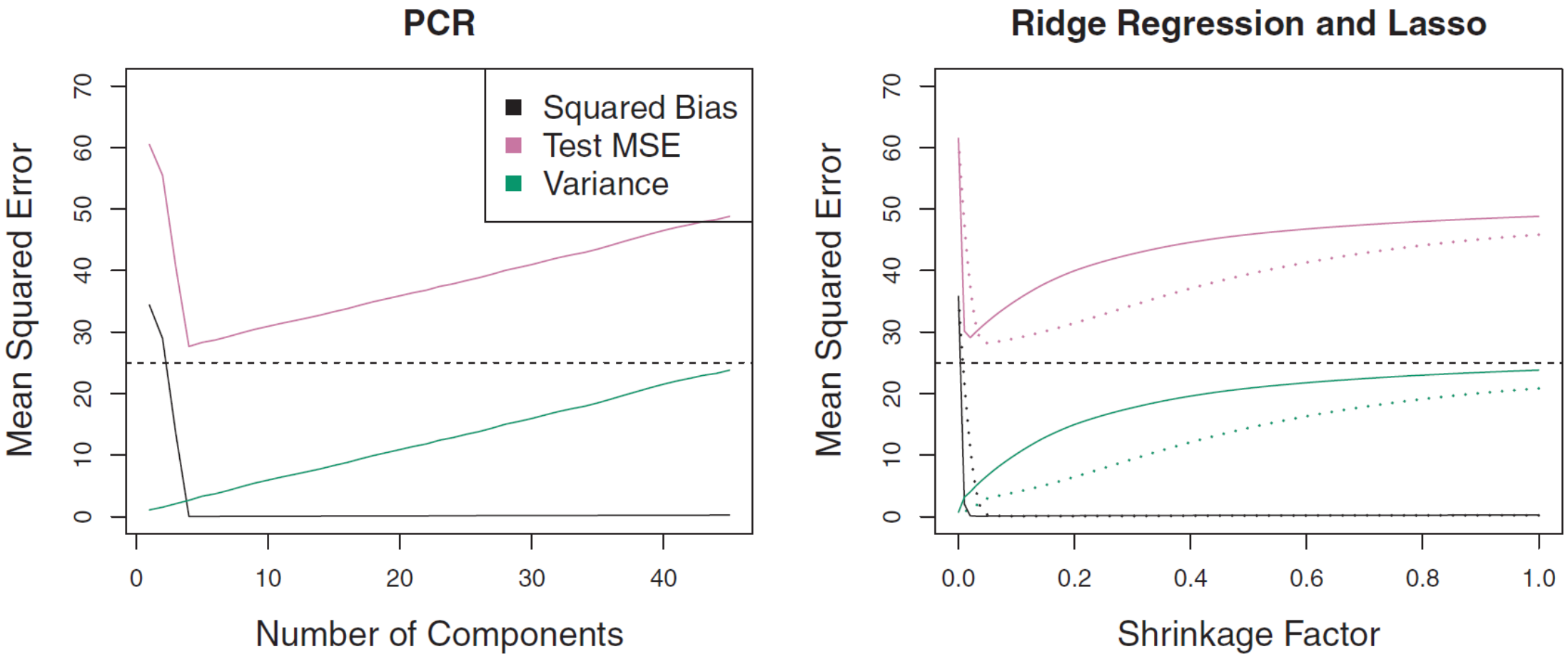




"All 45 predictors related to the response"

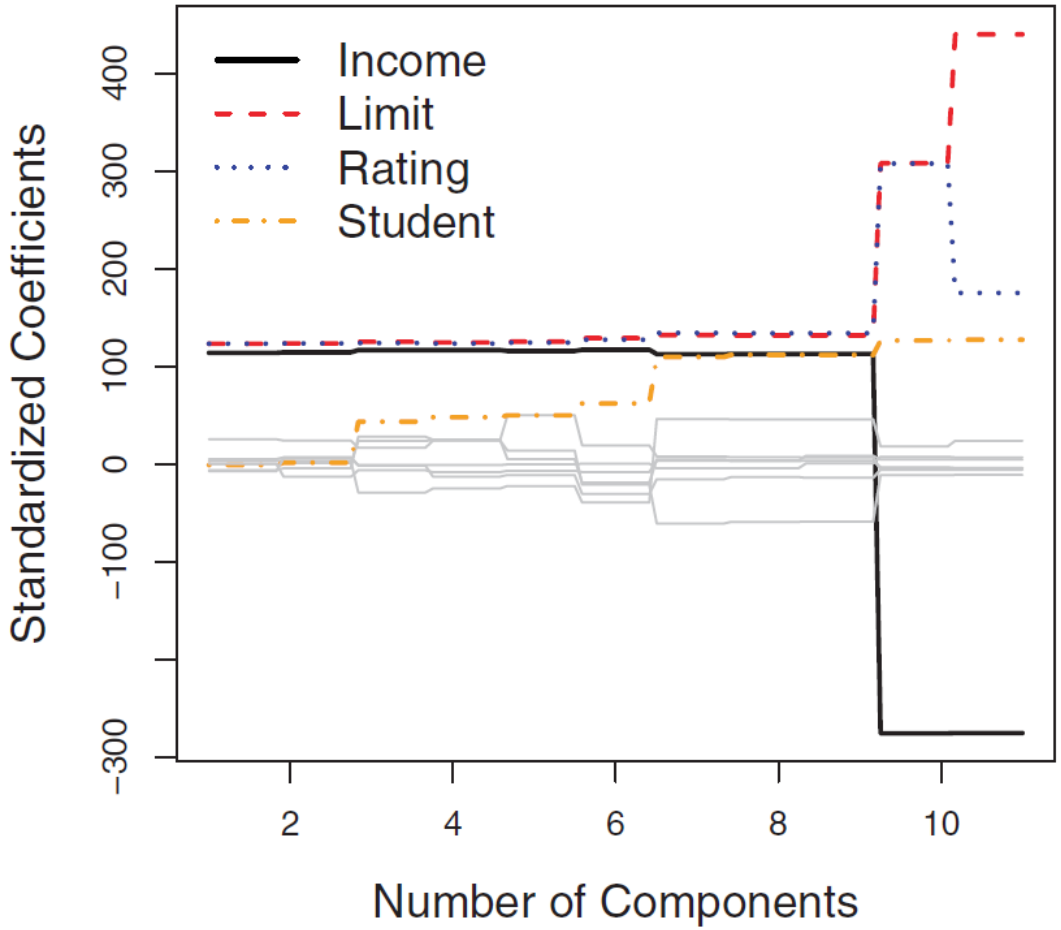
"Only 2 predictors related to the response"

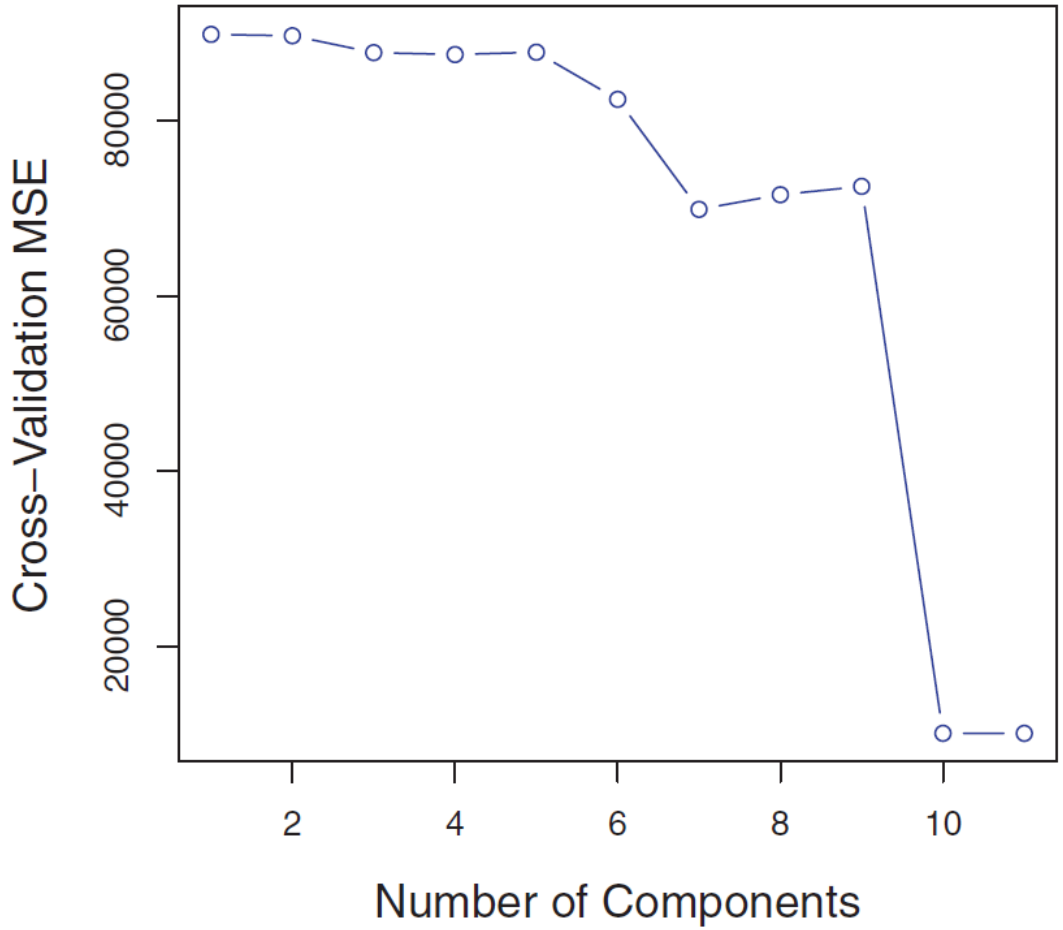
Principal Components Regression versus Ridge Regression and the Lasso



Simulated data set where the first 5 components contains all the information about the response

Principal Components Regression on the Credit Data Set





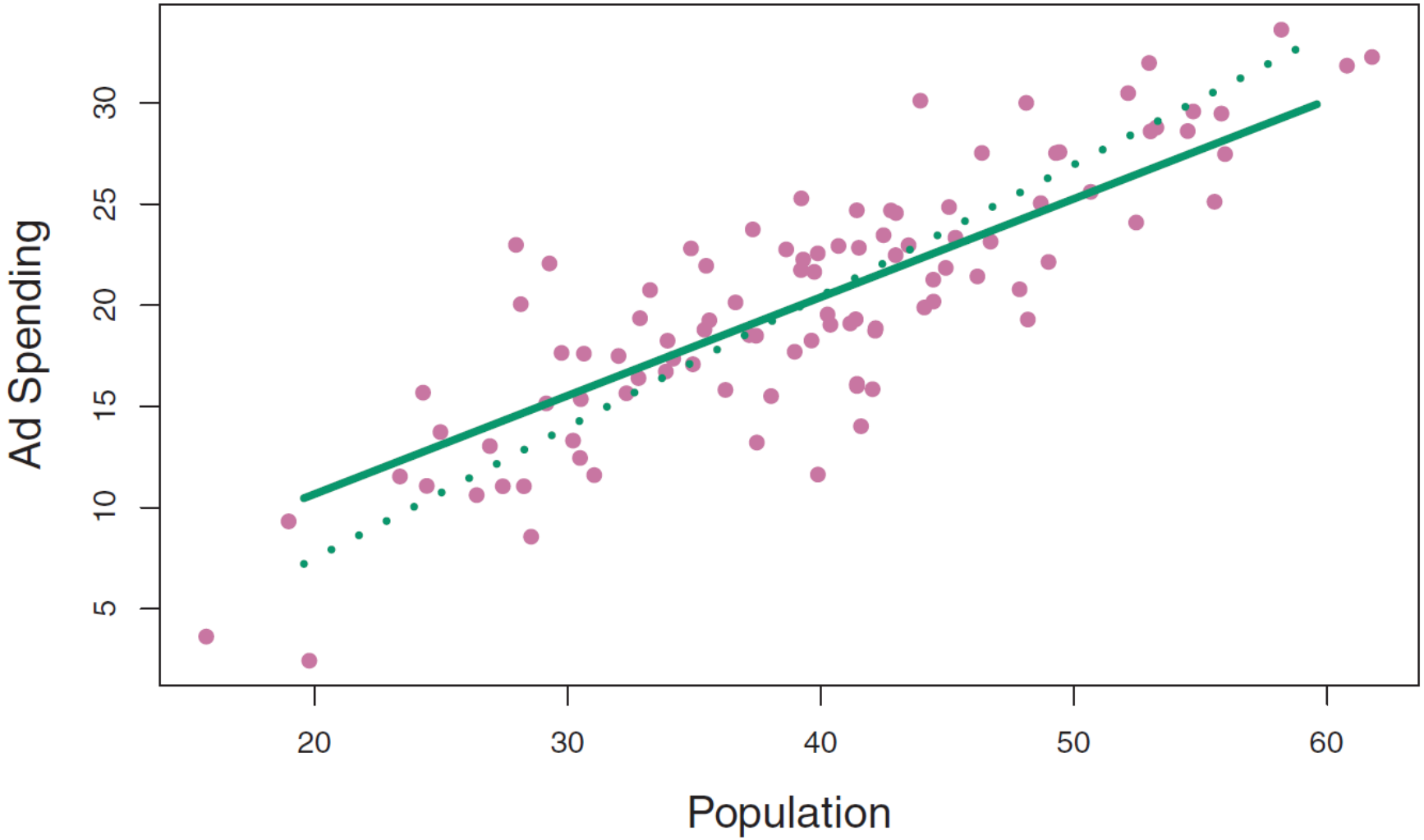
Partial Least Squares

Partial Least Squares (PLS) is a supervised alternative to Principal Components Regression (PCR): unlike PCR, PLS also uses the output variable to generate the new predictors

Nonlinear Partial Least Squares (NIPALS)

```
X = scale(X, center = T, scale = T)
y = scale(y, center = T, scale = T)
New.Predictors = NULL
for (i in 1:m) {    # where m is the number of new predictors
   weights = t(X) %*% y
   weights = weights / sqrt((t(weights) %*% weights)[1,1])
   new.predictor = X %*% weights
   # lm(X[,j] ~ new.predictor)$coefficients[2]
   p = (t(X) %*% new.predictor) / (t(new.predictor) %*% new.predictor)[1,1]
   # lm(y ~ new.predictor)$coefficients[2]
   coefficient = (t(new.predictor) %*% y) / (t(new.predictor) %*% new.predictor)[1,1]
   X = X - new.predictor %*% t(p)
   y = y - new.predictor %*% coefficient
   New.Predictors = cbind(New.Predictors, new.predictor)
}
```

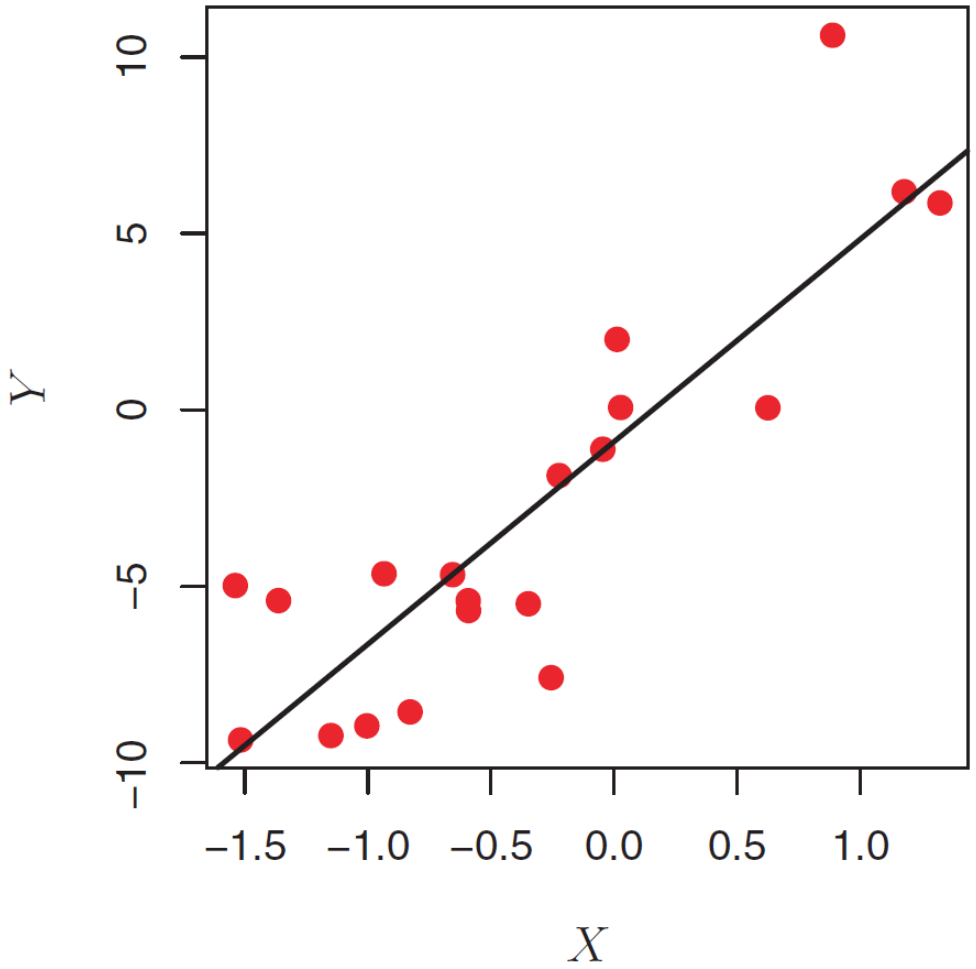
Partial Least Squares Direction (solid) versus the Principal Components Regression Direction (dotted)

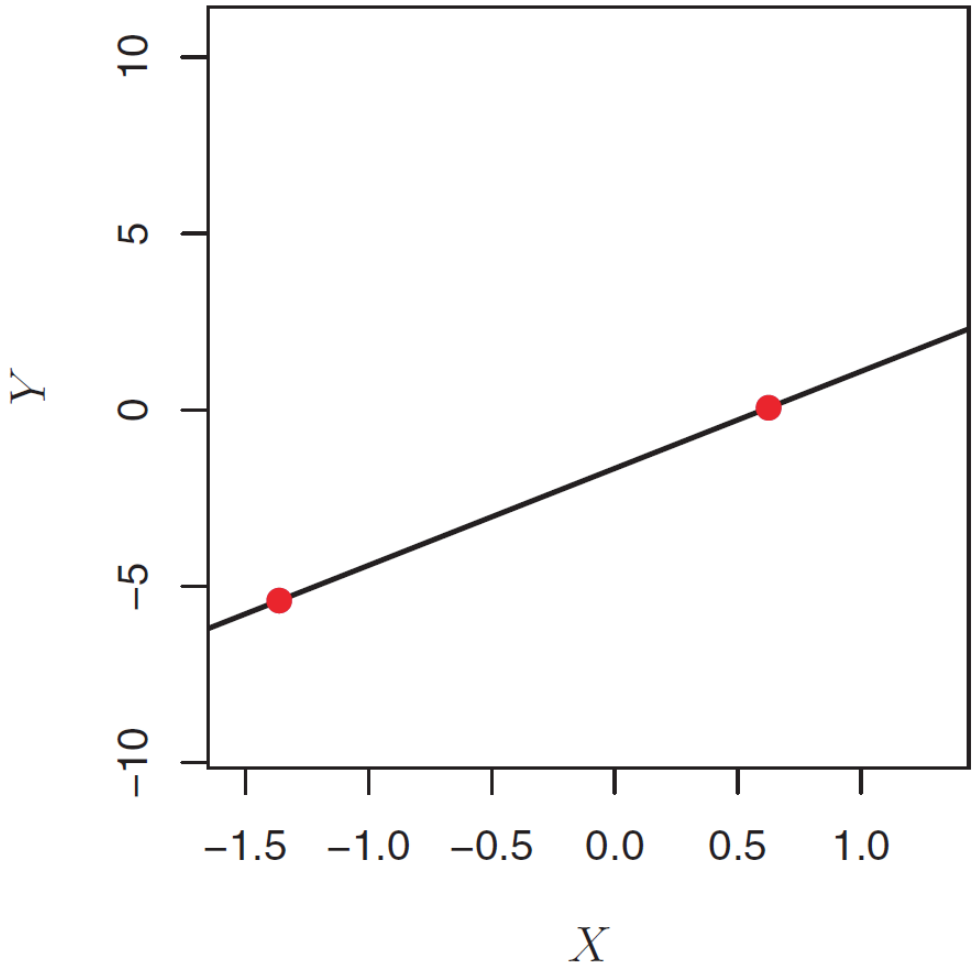


High-Dimensional Data

- $p > n$
- Examples:
  - Predicting blood pressure based on age, gender, Body Mass Index (BMI) and half a million different Single Nucleotide Polymorphisms [SNPs: mutations of DeoxyriboNucleic Acid (DNA) molecules]: $n = 200$ and $p \approx 500{,}000$
  - Understanding shopping behavior using search history [using binary indicators for the search terms]: $n \approx 1{,}000$ and p is the number of search terms

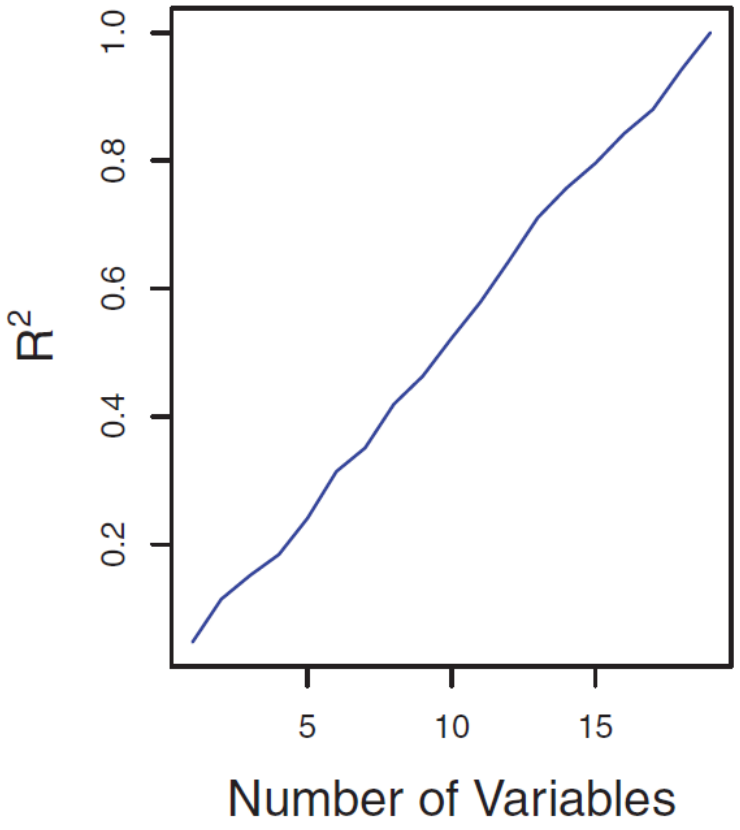
High-Dimensional Data: What Could Go Wrong?

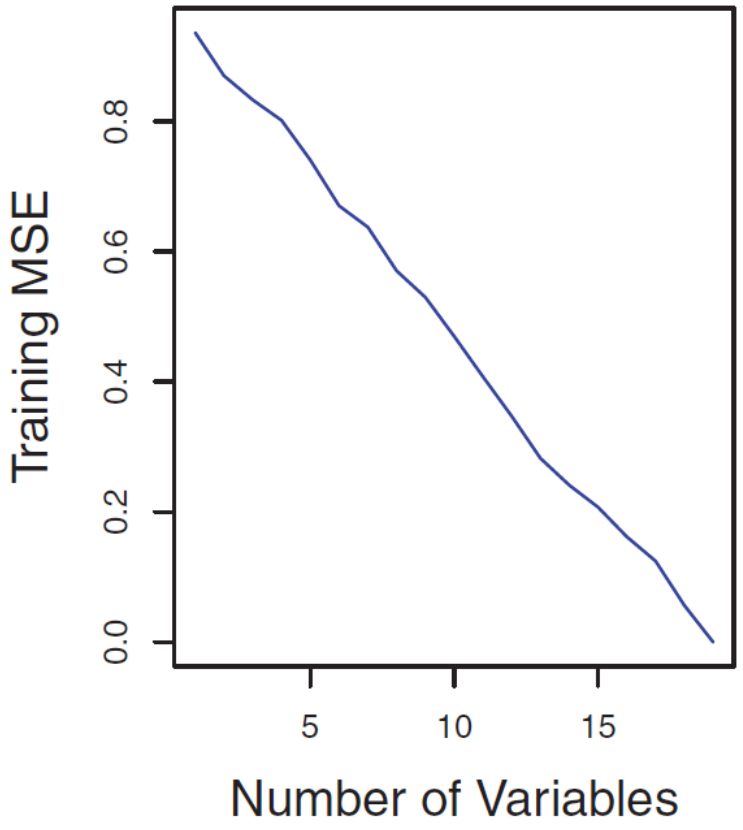


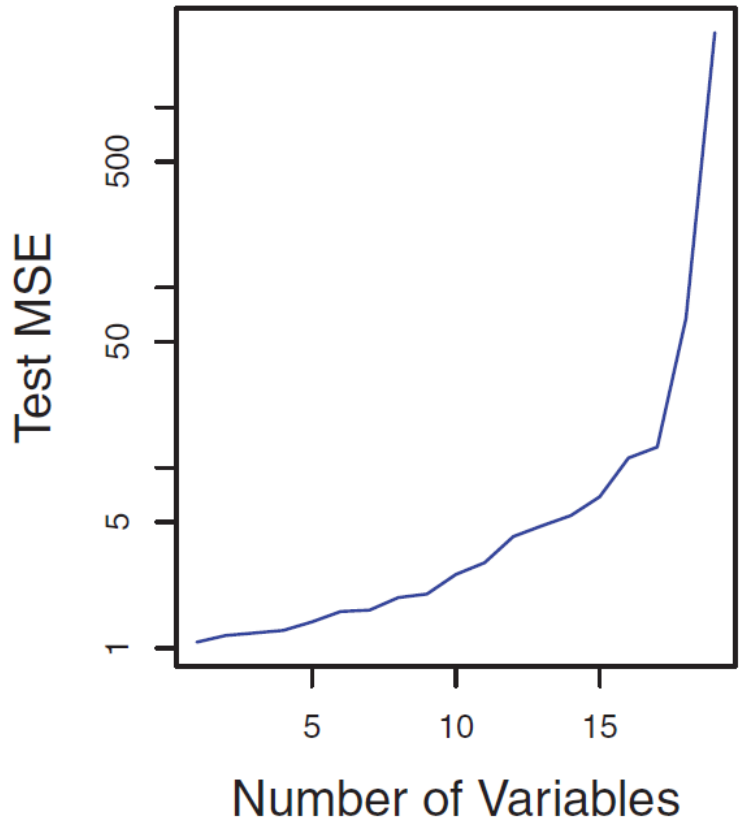


Example where n == p: can we trust the fit?

High-Dimensional Data: What Could Go Wrong?

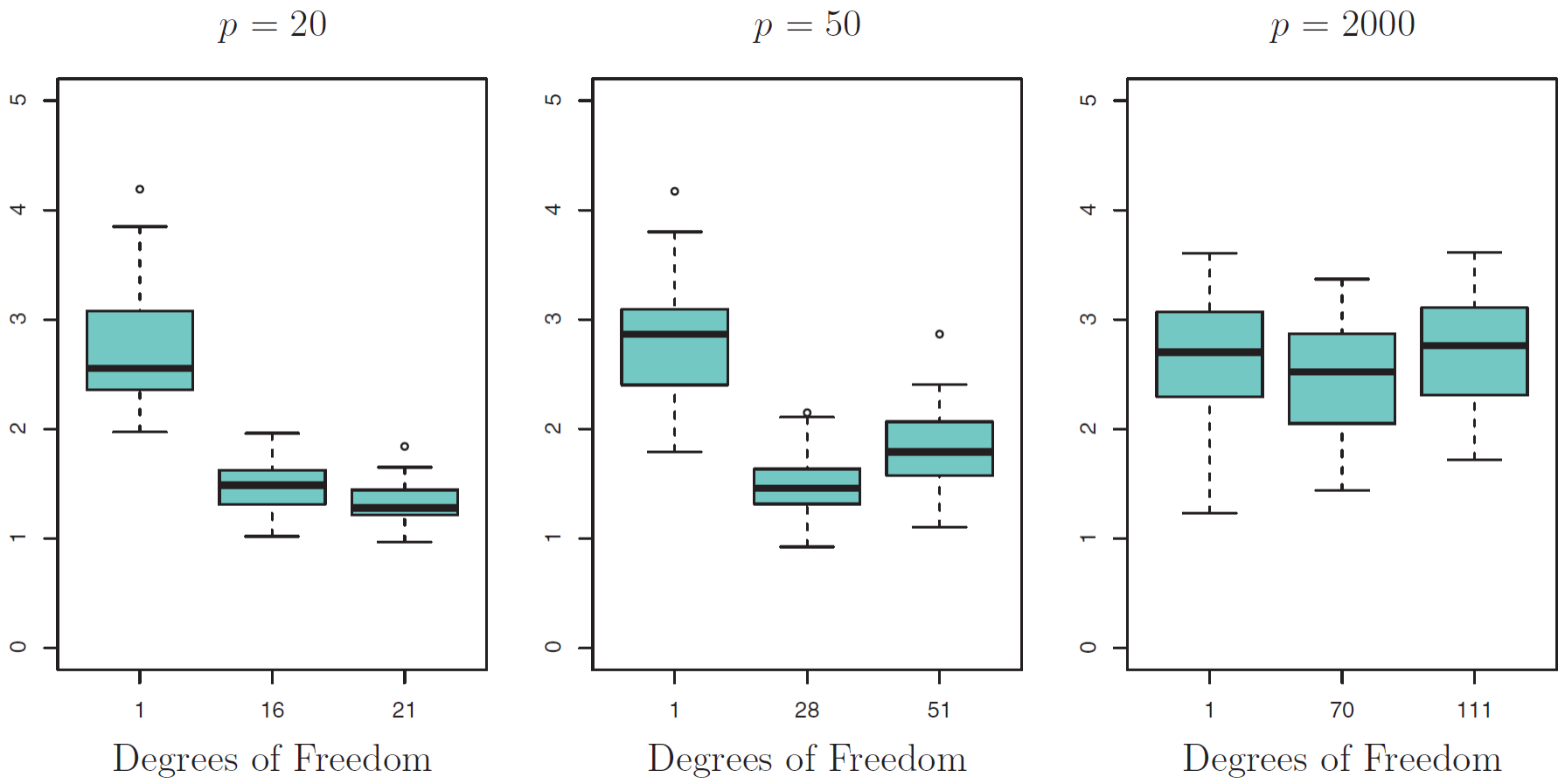






n = 20 observations with up to 20 *unrelated* variables: perfect fit!

Regression in High Dimensions



- n = 100
- p = 20 related variables
- Regularization plays a key role
- Appropriate model selection is crucial for good performance
- Test error tends to increase as the dimensionality increases [thanks to fitting noise: this is the curse of dimensionality]

Interpreting Results in High Dimensions

- Multicollinearity: it's possible to write a variable as a linear combination of all the other variables
  - We won't know exactly which variables are related to the outcome ☹
  - Example: forward stepwise selection picks out 17 of the half million Single Nucleotyde Polymorphisms (SNPs) for predicting high blood pressure: we cannot say this set of predictors is better than all the others
- We need to be careful about reporting error
  - As seen earlier, it's easy to obtain a useless model that has residuals of zero on the training data
  - Make sure the evaluation is based on validation or cross validation

Agenda

Homework

Model Selection/Regularization