STA 4241 Lecture, Week 8

Overview of what we will cover

- **More on shrinkage approaches**
  - Lasso regression
  - Comparison of ridge and lasso
- Approaches based on dimension reduction
  - Introduction to principle components analysis
  - Principle components regression
  - Partial least squares

Ridge regression

- Ridge regression has one important drawback for large values of p
- While the coefficients are shrunk towards zero, none are exactly zero
  - Unless $\lambda = \infty$ and all coefficients are zero
- This leads to a model that is difficult to interpret
- We might prefer a model that has certain covariates removed
  - The j^{th} element of $\widehat{\boldsymbol{\beta}}$ is exactly zero for some values of j

The lasso

- The lasso is a more recent shrinkage approach that aims to solve this issue
  - Still perform well in terms of prediction
  - Select important variables
- The lasso estimate, which we will denote by $\widehat{\boldsymbol{\beta}}_\lambda^L$ is the value of $\boldsymbol{\beta}$ that minimizes

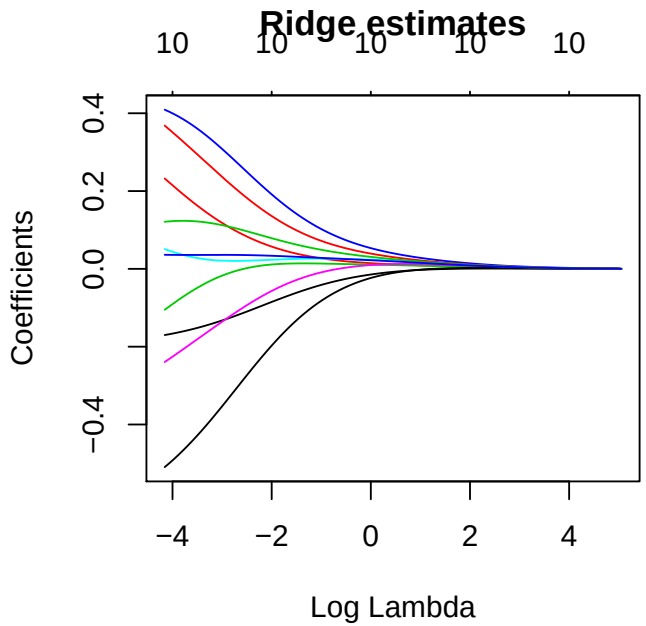
$$\sum_{i=1}^{n}\left(Y_i - \beta_0 - \sum_{j=1}^{p}\beta_j X_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j|$$

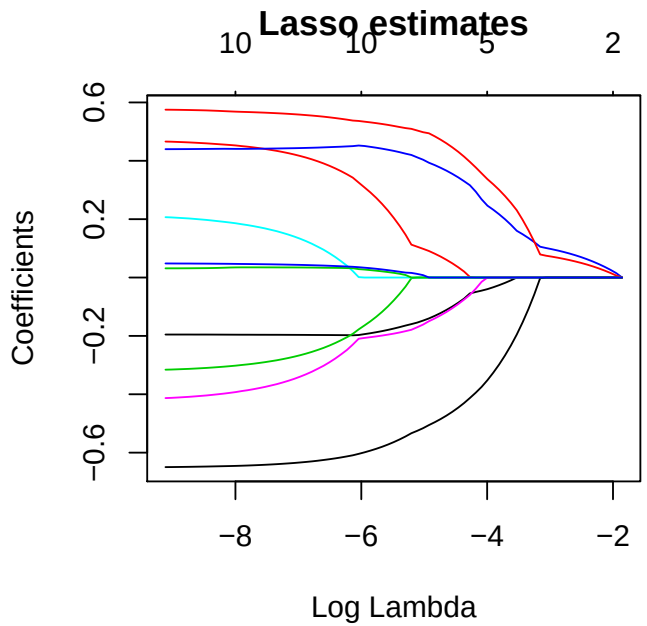
The lasso

- This looks extremely similar to ridge regression
  - Penalize the absolute values instead of squared values
- How does this estimator perform variable selection?
  - Why are some coefficients exactly zero?
- The full mathematical justification for why the lasso solution is sparse is beyond the scope of this class, but now we can gain some intuition for it

Example on mice data

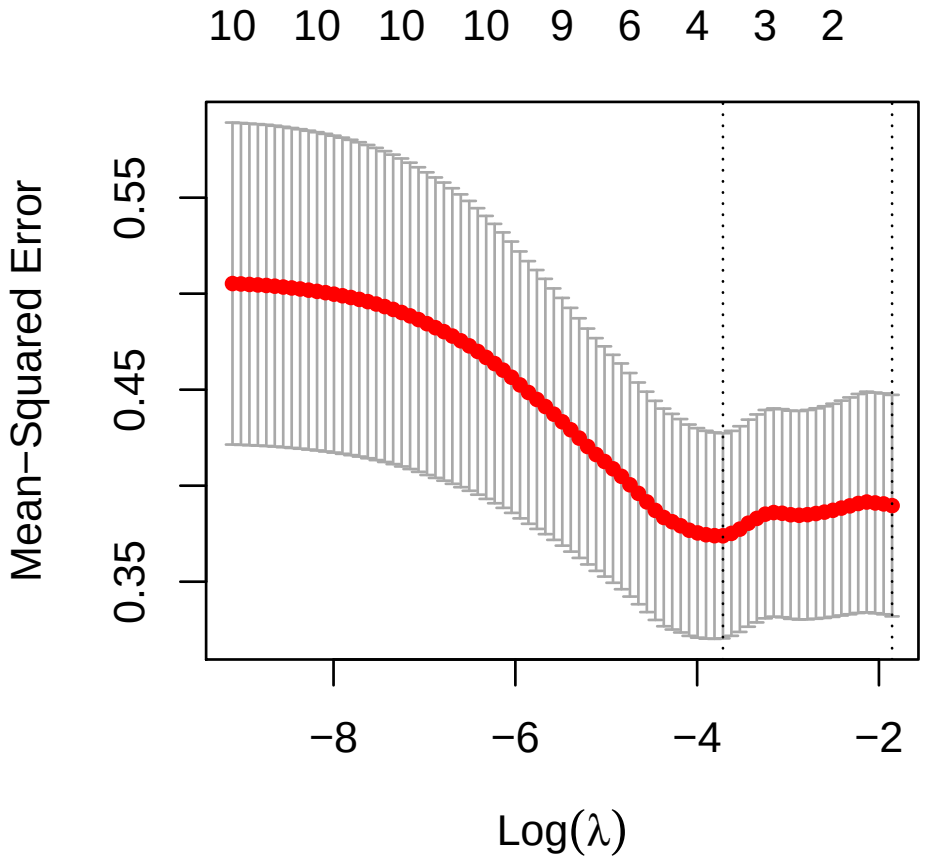
- Here are the coefficients from both ridge and lasso regression on the mice data
  - Lasso solution has estimates exactly equal to zero





Example on mice data

- Just as with ridge regression we need to choose the tuning parameter via cross validation



Alternative formulation of the problem

- To gain additional intuition for the difference between lasso and ridge regression, we can look at an alternative way to write these estimators as solutions to constrained optimization problems
- The lasso estimator can be equivalently described by

$$\widehat{\boldsymbol{\beta}}_\lambda^L = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min} \left\{ \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq s.$$

- Minimize the squared residuals while keeping the coefficients small

Alternative formulation of the problem

- Interestingly, both ridge regression and best subset selection can be written in this way

  Ridge:

$$\widehat{\boldsymbol{\beta}}_{\lambda}^{R} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{p}}{\arg\min} \left\{ \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \beta_j^2 \leq s.$$

  Best subset:

$$\widehat{\boldsymbol{\beta}}_{\lambda}^{Sub} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{p}}{\arg\min} \left\{ \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} 1(\beta_j \neq 0) \leq s.$$

Alternative formulation of the problem

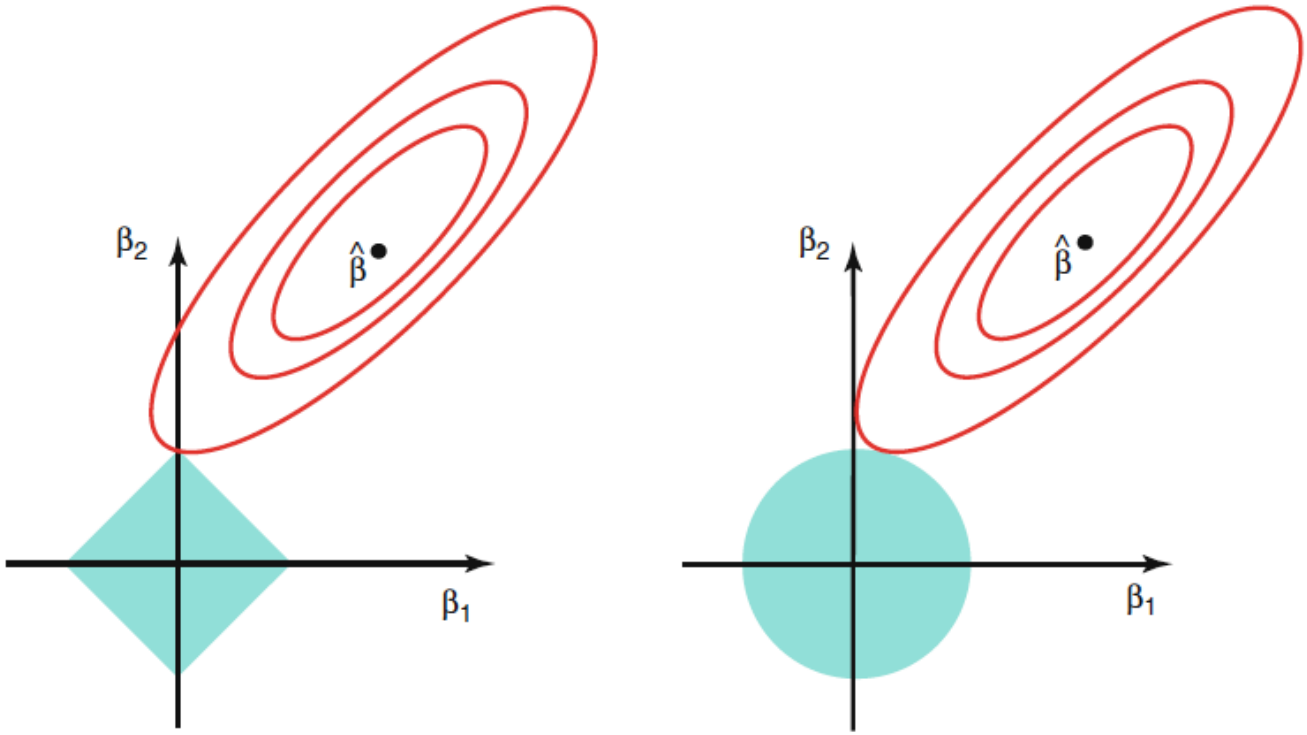
- All three estimators minimize the squared residuals subject to a constraint on the $\boldsymbol{\beta}$ vector
- Three different constraints
  - ℓ_0, ℓ_1, and ℓ_2 contraints
- It is clear how the best subset constraint leads to sparse solutions
  - Constraint is directly on the number of nonzero coefficients
- But why does the lasso and not the ridge?

Why the lasso performs variable selection

- To simplify let's think of the situation where we have two covariates and therefore are only constraining β_1 and β_2
- Ridge constraint is that $\beta_1^2 + \beta_2^2 < t$
  - A circle around the origin of radius $\sqrt{t}$
- Lasso constraint is that $|\beta_1| + |\beta_2| < t$
  - Diamond around the origin
- Note that if t is large enough, the constraints will contain the least squares estimates and we will simply obtain $\widehat{\beta}^{ols}$

Why the lasso performs variable selection

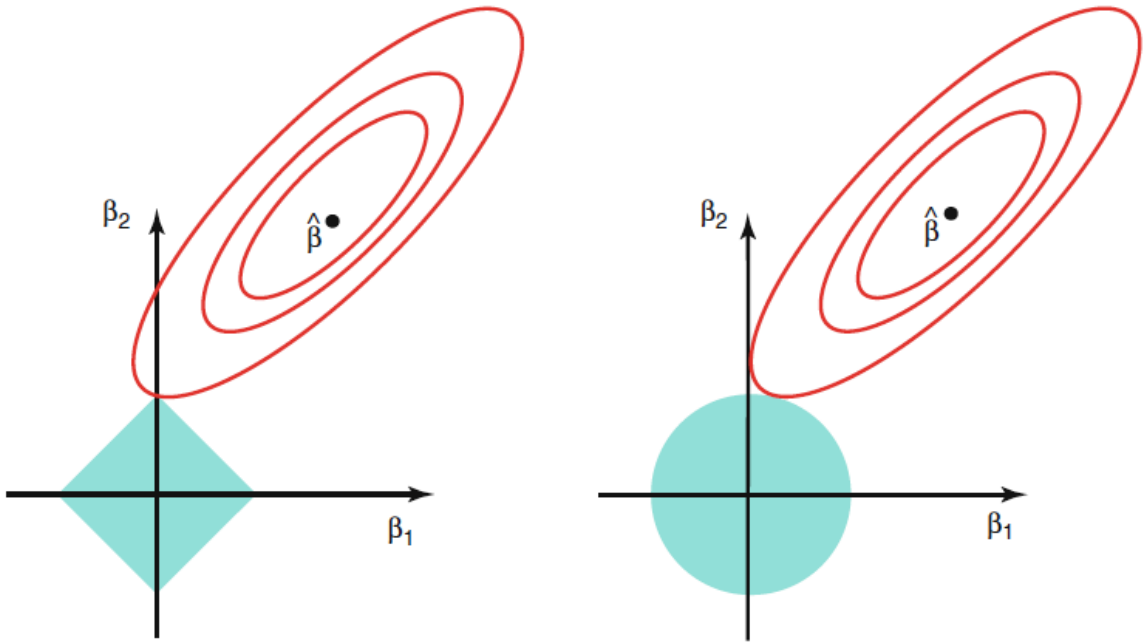
- Ellipses are the contours of the sum of squared errors (RSS), which is minimized at $\widehat{\beta}^{ols}$
- Blue regions represent the constraint regions



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Why the lasso performs variable selection

- The lasso or ridge solutions are the points where the RSS first hits the constraint region
- If the constraint is small enough, then the RSS will first hit the lasso constraint at a corner, thereby making one (or more) coefficient exactly zero
  - Not true for the ridge solutions



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

The lasso solution

- As with ridge regression, it can be helpful to examine the lasso solutions under a simplified setting
- Similar to lab, let's assume we have orthonormal covariates, i.e. $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}_p$
- In this setting, the lasso solutions have a nice form
  - Allow us to examine their bias, etc.

The lasso solution

- In this setting, it can be shown that the lasso solution satisfies

$$[\widehat{\beta}_\lambda^L]_j = \begin{cases} \mathbf{X}_j^T\mathbf{Y} - \lambda & : \mathbf{X}_j^T\mathbf{Y} > \lambda \\ \mathbf{X}_j^T\mathbf{Y} + \lambda & : \mathbf{X}_j^T\mathbf{Y} < -\lambda \\ 0 & : |\mathbf{X}_j^T\mathbf{Y}| \leq \lambda \end{cases}, \quad j = 1, \ldots, p$$

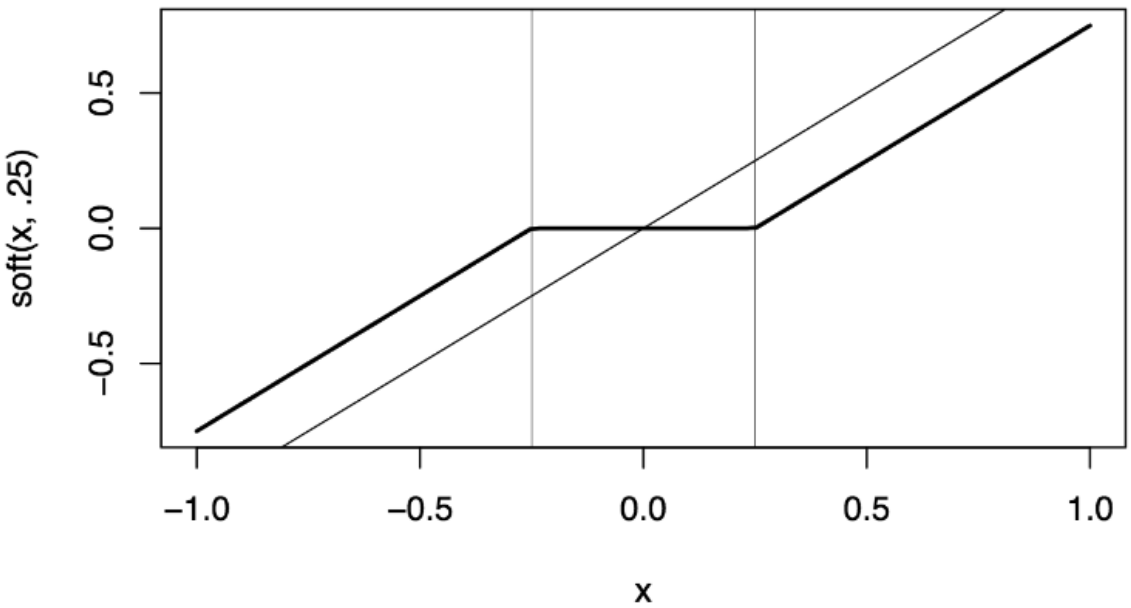
  This can also be written as

$$\implies [\widehat{\beta}_\lambda^L]_j = \mathrm{soft}(\mathbf{X}_j^T\mathbf{Y}, \lambda) \equiv \mathrm{max}(|\mathbf{X}_j^T\mathbf{Y}| - \lambda, 0)\mathrm{sign}(\mathbf{X}_j^T\mathbf{Y}), \quad j = 1, \ldots, p$$

  where soft is often referred to as the "soft-thresholding operator".

The lasso solution

- Here is an illustration of the soft-thresholding operator with threshold 0.25
- Values are shrunk by 0.25 towards 0 until they cross 0.25, where they become exactly zero



The lasso solution

- How does this compare with OLS and ridge regression estimators under orthonormal covariates?

  OLS:

$$\widehat{\beta}_j^{ols} = \mathbf{X}_j^T \mathbf{Y}$$

  Ridge:

$$\widehat{\beta}_j^{R} = \frac{\mathbf{X}_j^T \mathbf{Y}}{1+\lambda}$$

  Lasso:

$$\widehat{\beta}_j^{L} = \max(|\mathbf{X}_j^T \mathbf{Y}| - \lambda, 0)\mathrm{sign}(\mathbf{X}_j^T \mathbf{Y})$$

The lasso solution

- The OLS solution is simply the correlation between $\mathbf{X}_j$ and $\mathbf{Y}$
- Ridge takes the OLS solution and divides by a factor of $(1 + \lambda)$
  - Multiplicative shrinkage towards zero
- Lasso takes the OLS solution and subtracts a constant amount of λ
  - Additive shrinkage towards zero
  - Exactly equal to zero if correlation between $\mathbf{X}_j$ and $\mathbf{Y}$ is close to zero

The lasso solution

- In general, the lasso does not have a nice closed form solution like the ridge solution or OLS
- It is not as simple as taking the derivative and solving for zero
  - Absolute value is not differentiable at zero
  - Solving the optimization problem involves subgradients and the KKT conditions for optimization
- The lasso is typically found using an iterative algorithm
  - Update coefficients one at a time and continue until convergence

The lasso solution

Algorithm 1: Coordinate descent algorithm for the lasso

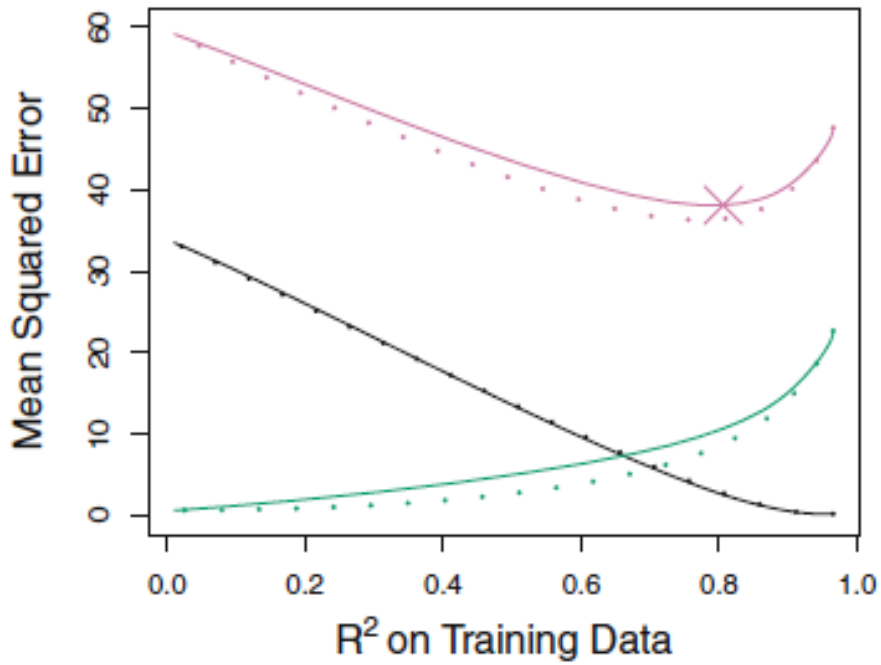
1. Initialize $\tilde{\beta} = (\tilde{\beta}_1, \tilde{\beta}_2, \ldots, \tilde{\beta}_p)$.
2. Compute $r = \mathbf{Y} - \mathbf{X}\tilde{\beta}$
3. Repeat until convergence:
    1. For $j = 1, \ldots, p$,
        1. Compute $r_j = r + \mathbf{X}_j\tilde{\beta}_j$
        2. Compute $\beta^+ = \text{soft}\left(\frac{r_j^T \mathbf{x}_j}{\mathbf{x}_j^T \mathbf{x}_j}, \lambda\right)$
        3. Set $[\tilde{\beta}]_j = \beta^+$
        4. Compute $r = r_j - \mathbf{X}_j\tilde{\beta}_j$

Comparison with ridge regression

- So the lasso has a clear advantage over ridge regression
  - Performs variable selection automatically
- But what about prediction performance?
- Which estimator is best for prediction performance?
  - It depends
  - Neither estimator universally outperforms the other one
  - Degree of sparsity plays a role

Comparison with ridge regression

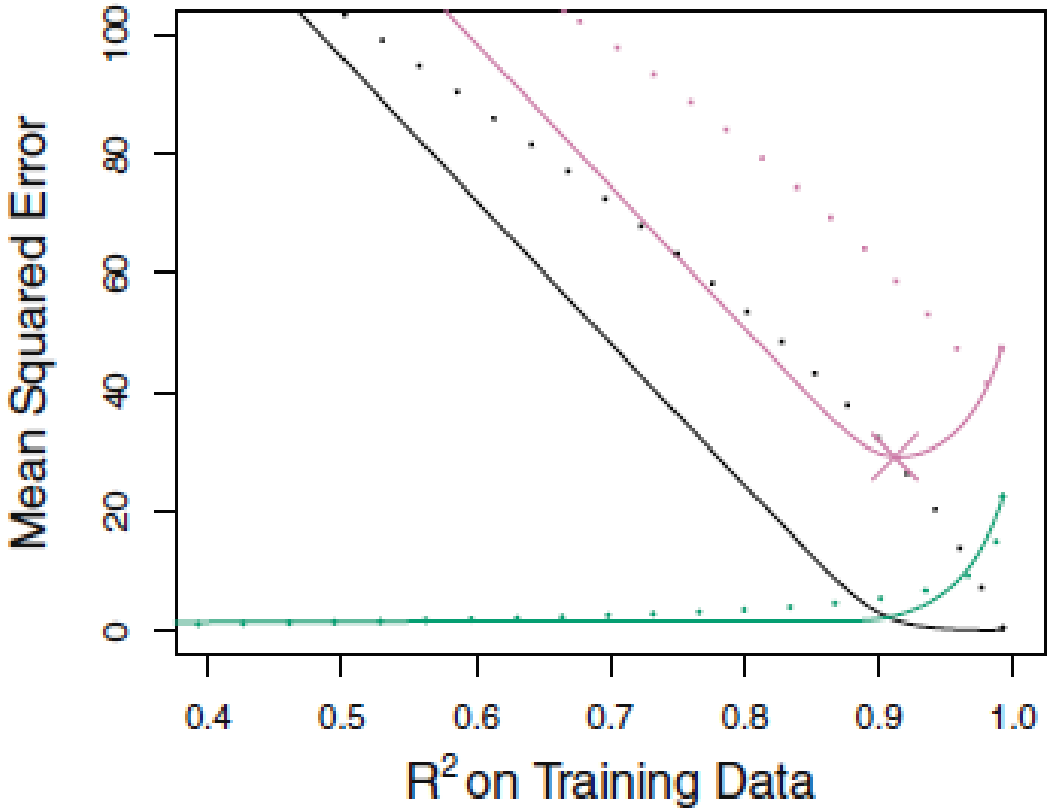
- Here is a simulated example from the textbook
- Black is the squared bias, green is the variance, purple is the MSE
- Solid line is lasso and dotted line is ridge



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Comparison with ridge regression

- Here is a different simulated example where the lasso outperforms the ridge



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Comparison with ridge regression

- These two examples highlight that there are situations where both the lasso or ridge can be preferred
- The first example was one in which all 45 predictors were associated with the outcome
- The second example had only 2 of 45 predictors that were important for predicting the outcome
- Not surprising that lasso does better in the sparse scenario

Problems with the lasso

- The lasso has been shown to have some nice theoretical properties, however, it can be improved upon in certain ways
- There are three common issues with the lasso
  1. Highly correlated predictors
  2. Bias of the estimated coefficients
  3. Inference for the coefficients is challenging
- If two predictors are highly correlated, lasso will tend to select one of them and drop the other

Fixes for the lasso issues

- Some of these issues can be addressed with different penalties
- The elastic net penalty minimizes the following

$$\sum_{i=1}^{n}\left(Y_i - \beta_0 - \sum_{j=1}^{p}\beta_j X_{ij}\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| + \frac{\gamma}{2}\sum_{j=1}^{p}\beta_j^2$$

- This addresses the issue of correlated covariates
- Elastic net will jointly shrink coefficients for highly correlated variables instead of simply selecting one
- Two tuning parameters to choose instead of one

Fixes for the lasso issues

- To fix the bias issue, the adaptive lasso has been proposed, which minimizes

$$\sum_{i=1}^{n}\left(Y_i - \beta_0 - \sum_{j=1}^{p}\beta_j X_{ij}\right)^2 + \lambda\sum_{j=1}^{p} w_j|\beta_j|$$

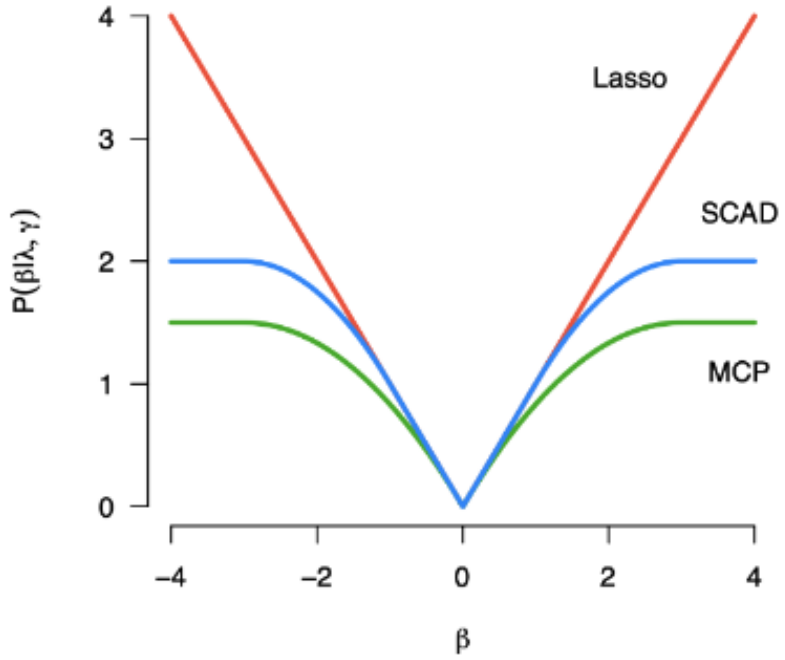
- w_j are weights that determine how much to shrink each coefficient
- If we have an initial estimate of $\boldsymbol{\beta}$ called $\widetilde{\boldsymbol{\beta}}$ then we can set

$$w_j = |\widetilde{\beta}_j|^{-1}$$

and reduce shrinkage of important covariates

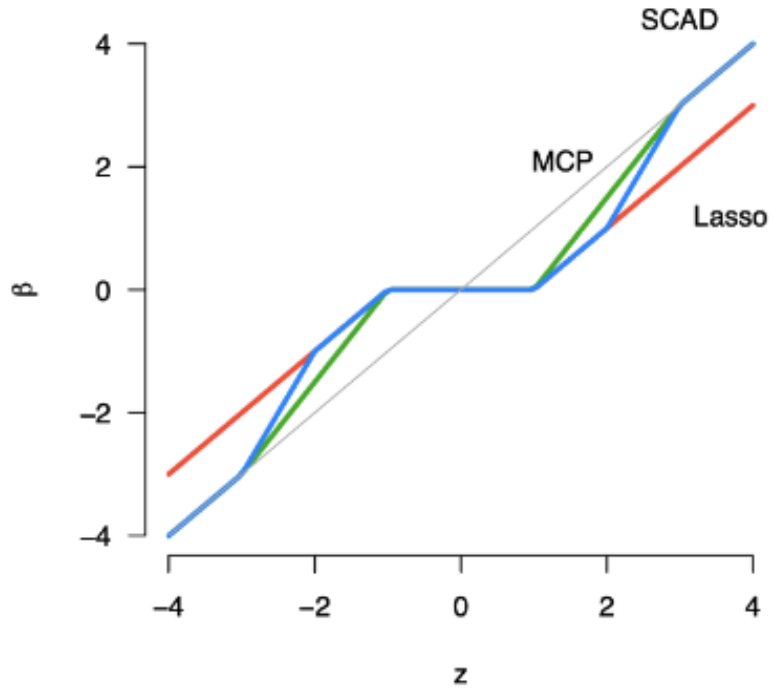
Fixes for the lasso issues

- An alternative is to create completely new penalties that do not penalize large coefficients as aggressively as lasso
- Here are the penalties as a function of β for lasso and two other penalties that aim to reduce bias of big coefficients



Fixes for the lasso issues

- We can see that the thresholding operator for these penalties acts differently than lasso
- Small coefficients still forced to zero, but large ones are not shrunk



Fixes for the lasso issues

- Until a few years ago, there was no way to perform inference on the lasso
  - No confidence intervals provided
  - Not ideal for statisticians or practitioners
- We won't cover inference with the lasso in detail, but there are two main ways to perform inference
  1. Conditional inference
     - Only perform inference on the nonzero lasso estimates
     - Performs inference one coefficient at a time
  2. De-biased lasso
     - Alter the lasso estimate so that it is asymptotically normal
     - Makes very strong assumptions

Overview of what we will cover

- More on shrinkage approaches
  - Review of ridge regression
  - Lasso regression
  - Comparison of ridge and lasso
- **Approaches based on dimension reduction**
  - Introduction to principle components analysis
  - Principle components regression
  - Partial least squares

Dimension reduction

- The previous approaches used the full set of p covariates and either shrunk their coefficients or found a subset of the covariates that were predictive of the outcome
- Dimension reducing techniques aim to find a transformed set of predictors
  - Smaller dimension than p
  - Retain as much information as possible
- Less interpretable, but can work very well in certain settings

Dimension reduction

- We aim to find new predictors $Z_1, \ldots, Z_m$ defined by

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

- These are linear combinations of our original predictors
- We then fit a regression model using least squares defined by

$$Y_i = \theta_0 + \sum_{m=1}^{M} \theta_m Z_{im} + \epsilon_i$$

Dimension reduction

- We've reduced the problem from estimating $(p+1)$ parameters to $(M+1)$ parameters
- Performance relies on well chosen values of $\phi_{jm}\ \forall\ j, m$
- Note that

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

where

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}$$

Dimension reduction

- So we've actually fit the same linear model, but under the constraint that the parameters take the form

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}$$

- This constraint can bias the estimated coefficients
- When $M < p$, this can lead to large decreases in the variance of the estimated coefficients
- If $M = p$, we recover the least squares estimates

Dimension reduction

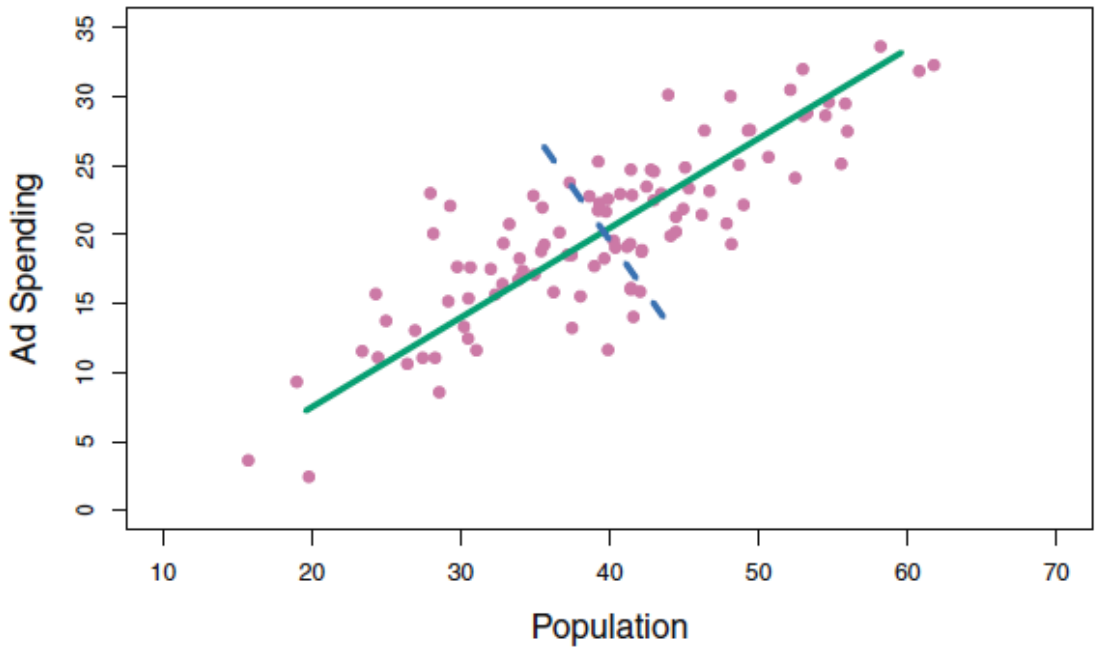
- All that we need to decide is how to choose the ϕ_{jm} parameters
- Our goal is to keep M as small as possible without hurting predictive performance
- Ideally we would have linear combinations of the covariates that contain all of the relevant information in the data
- We explore two approaches to choosing these weights
  1. Principle components analysis
  2. Partial least squares

Principle components analysis (PCA)

- Transforms a set of correlated variables into uncorrelated variables
- Uncorrelated variables are called the principle components (PCs)
- Very commonly used tool for dimension reduction
  - Reduce the number of variables without losing much information
  - Maximize the variability explained by the reduced variables
- This can improve statistical analysis in a number of ways
  - Data visualization / exploration
  - Improved prediction

Illustration in two dimensions

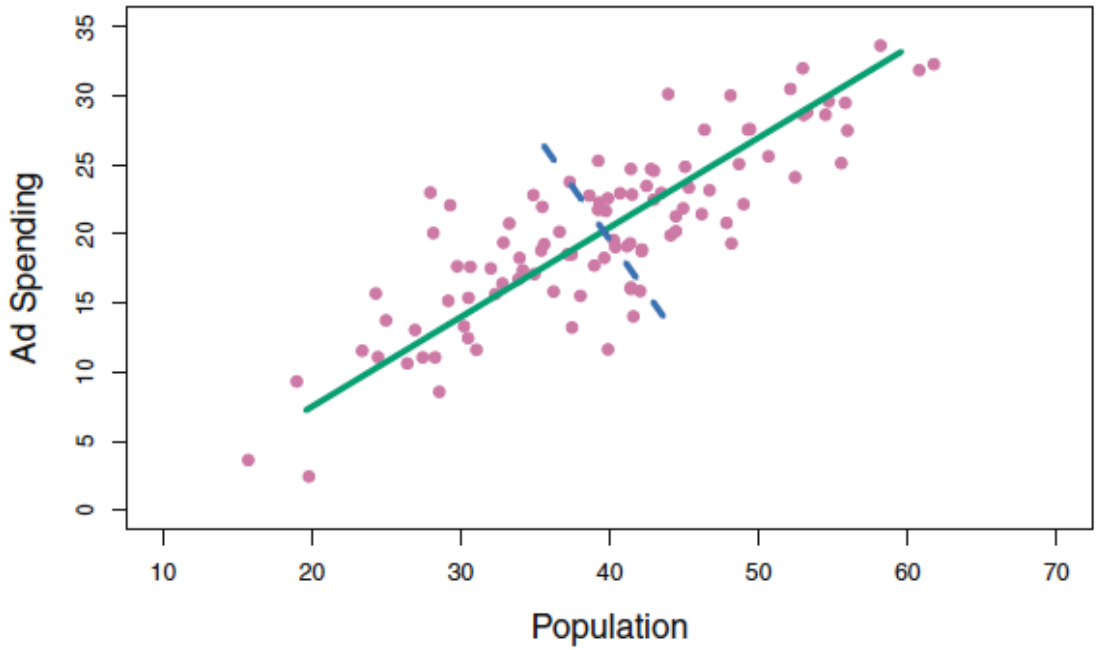
- Let's first examine PCs visually in two dimensions using an example from the textbook
- The green line shows the first PC and the blue line shows the second
  - Can obtain up to p PCs



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Illustration in two dimensions

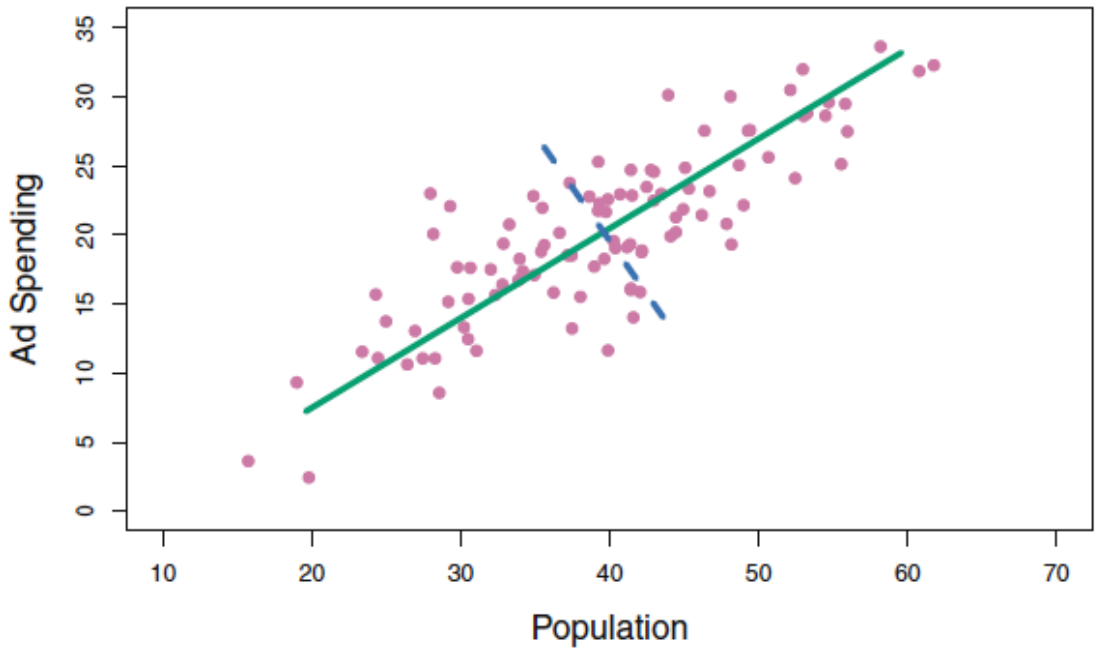
- The first PC is the line that is closest to the data
  - Minimizes the sum of perpendicular distances between data points and the line



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Illustration in two dimensions

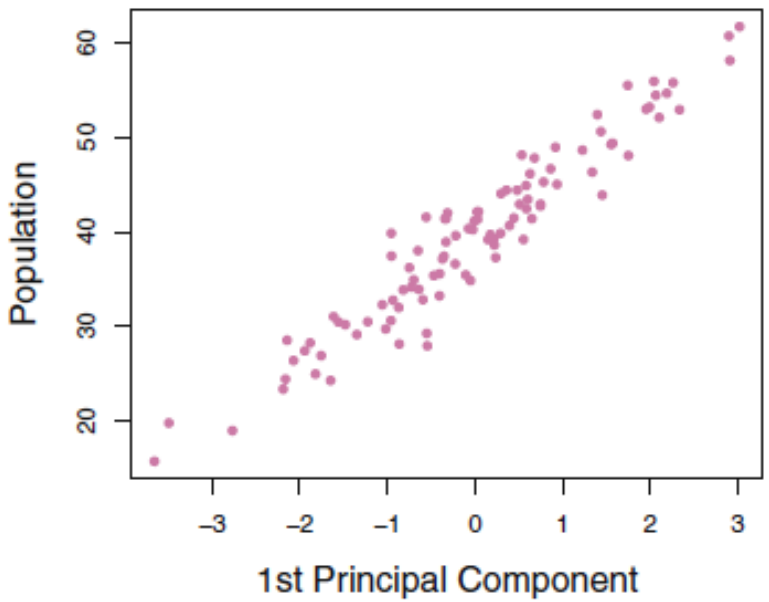
- The first PC is also the line such that if we project the observations onto this line, the projected observations will have the largest variability

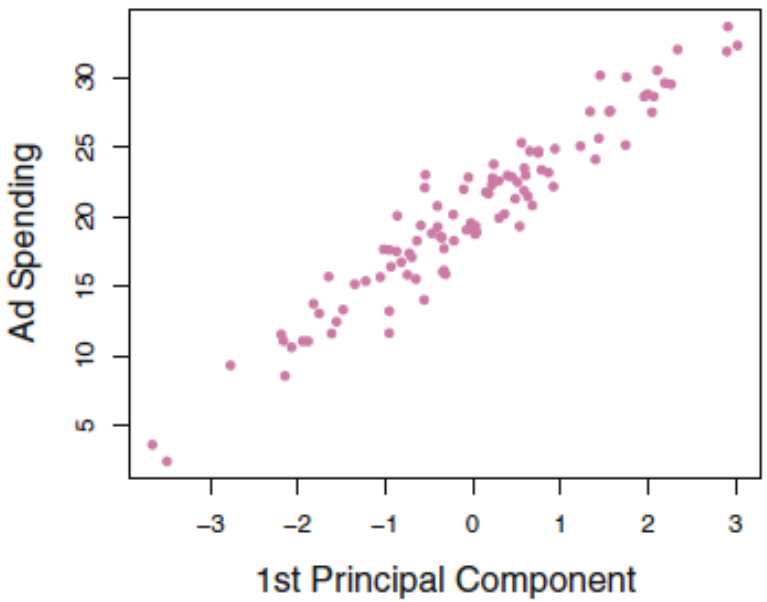


James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Illustration in two dimensions

- The first PC is strongly correlated with both predictors

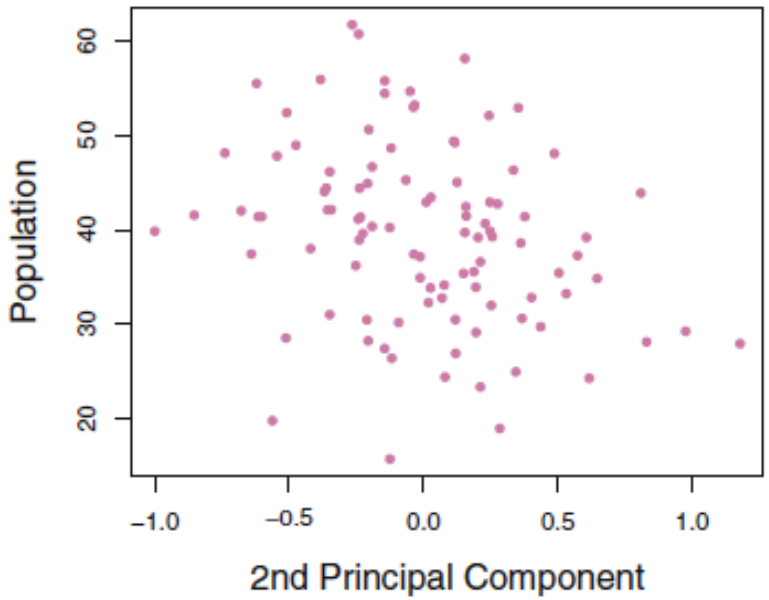


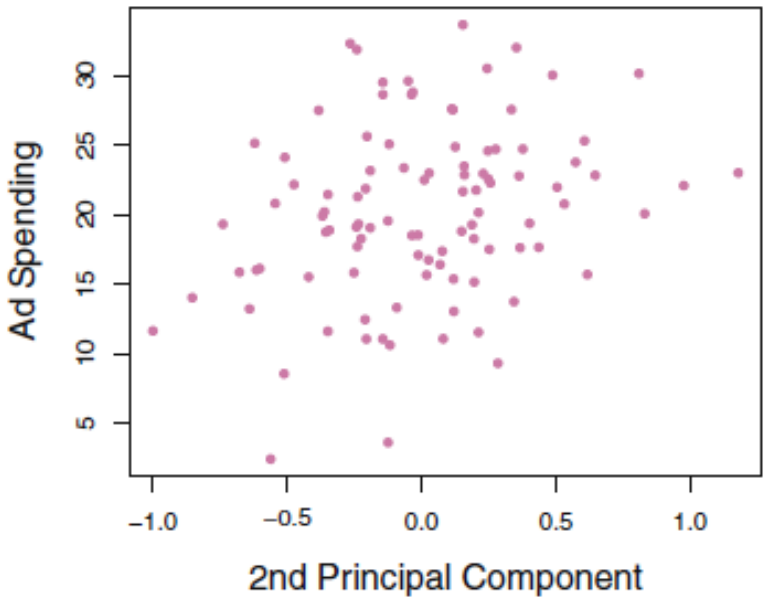


James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Illustration in two dimensions

- The second PC has very little correlation with the predictors in this case, because the first PC captured so much of the information in the two predictors





James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Mathematical formulation of PCA

- We will be using PCA to reduce the dimension of our covariates $\boldsymbol{X}$
- Suppose the correlation matrix of $\boldsymbol{X}$ is Σ
- PCA seeks to find linear combinations of $\boldsymbol{X}$ that explain the most variation in the original data
- These linear combinations need to be independent of each other

Finding the first PC

- Let ϕ be a p-dimensional vector
- Our goal is to maximize the variance of $\phi^T \boldsymbol{X}$
  - $\text{Var}(\phi^T \boldsymbol{X}) = \phi^T \boldsymbol{\Sigma} \phi$
- Why not just set ϕ to be large?
  - Instead restrict to vectors where $\phi^T \phi = 1$
- How do we find this ϕ?
  - Constrained maximization
  - Lagrange multipliers

Finding the first PC

- The method of Lagrange multipliers is a method for maximizing a function $f(x)$ subject to a constraint that $g(x) = 0$
- The idea is to create the Lagrangian function

$$f(x) - \lambda g(x)$$

- Then take the derivatives with respect to x and λ and solve for zero

Finding the first PC

- The Lagrangian function in our setting is

$$\boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\Sigma}\boldsymbol{\phi} - \lambda(\boldsymbol{\phi}^T\boldsymbol{\phi} - 1)$$

- If we take the derivative with respect to $\boldsymbol{\phi}$ and set equal to zero, we get the following

$$\begin{aligned} \boldsymbol{\Sigma}\boldsymbol{\phi} - \lambda\boldsymbol{\phi} &= 0 \\ \Leftrightarrow \boldsymbol{\Sigma}\boldsymbol{\phi} &= \lambda\boldsymbol{\phi} \end{aligned}$$

- This looks familiar!

Finding the first PC

- $\boldsymbol{\phi}$ is an eigenvector of $\boldsymbol{\Sigma}$
- λ is the corresponding eigenvalue
  - How do we choose which eigenvalue?
- We want to maximize $\boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\Sigma}\boldsymbol{\phi}$
  - Assuming all eigenvectors have been normalized so that $\boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\phi} = 1$
- We can see the following

$$\boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\Sigma}\boldsymbol{\phi} = \boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\lambda}\boldsymbol{\phi} = \lambda\boldsymbol{\phi}^{\boldsymbol{T}}\boldsymbol{\phi} = \lambda$$

- So we can choose λ to be the largest eigenvalue of $\boldsymbol{\Sigma}$
  - $\boldsymbol{\phi}$ is the corresponding eigenvector

Finding the first PC

- This process only gave us the first PC
- Finding the remaining PCs follows a similar procedure
- The one difference is that each subsequent PC has to be orthogonal to the previous PCs
- It turns out all of the PCs are simply the eigenvectors of $\mathbf{\Sigma}$
  - Ordered by the size of eigenvalues

PCA

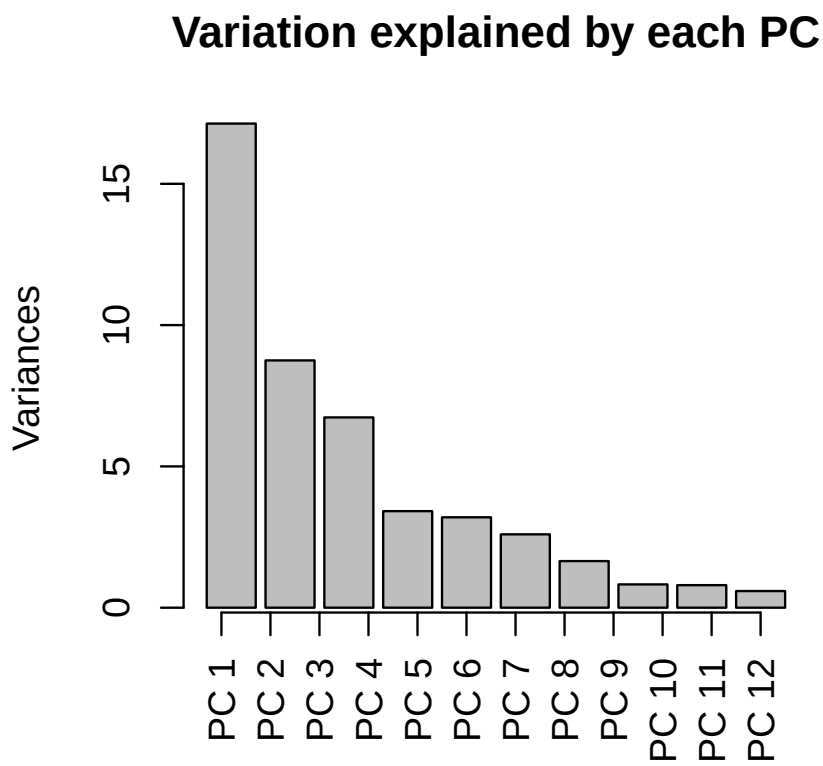
- Let $p = 50$ and $\boldsymbol{X}$ be a multivariate normal random variable with covariance

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.9 & 0.81 & \cdots & 0.9^{49} \\ 0.9 & 1 & 0.9 & \cdots & 0.9^{48} \\ 0.81 & 0.9 & 1 & \cdots & 0.9^{47} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0.9^{49} & 0.9^{48} & 0.9^{47} & \cdots & 1 \end{pmatrix}$$

- Many correlated variables
- Let's suppose we observe $n = 60$ independent copies of these random variables

PCA

- How much variation does each PC explain?



- The first 10 PCs contain well over 90% of the information from the original 50 variables!

Principle components regression

- What if we want to use $\boldsymbol{X}$ to predict an outcome Y?

  Using original variables:

$$Y_i = \beta_0 + \sum_{j=1}^{p} X_{ij}\beta_j + \epsilon_i$$

  Using PCs:

$$Y_i = \theta_0 + \sum_{m=1}^{M} Z_{im}\theta_m + \epsilon_i$$

  where $Z_{im} = \sum_{j=1}^{p} \phi_{jm} X_{ij}$ and the weights $(\phi_{1m}, \ldots, \phi_{pm})$ are given by the m^{th} eigenvector of the covariance matrix of $\boldsymbol{X}$

Principle components regression

- Let's compare the predictive performance between these two approaches on a simulated example with covariates $\boldsymbol{X}$ defined previously
- Let's keep the top 10 principle components

Approach	Average squared prediction error
PCs	2.87
Original	7.18

Principle components regression

- There are two ways one can choose the number of PCs to include
  - Cross-validation
  - Keep the number of PCs that contain some percentage (say 90%) of the variability in the data
- Cross-validation is preferable if our goal is predictive performance
- Note that principle components regression does not perform variable selection because each PC contains information from all p covariates

Principle components regression

- As with the shrinkage approaches, it is recommended to standardize variables before performing PCA
- There is one major pitfall with combining PCA and regression
  - PCA is unsupervised!
  - PCA does not use the outcome whatsoever
- There is no reason to believe that the 95% of the variability that my PCs accounted for is the variability in $\boldsymbol{X}$ that helps explain Y

Partial least squares

- Partial least squares (PLS) is an alternative approach to dimension reduction that incorporates the outcome
  - Supervised approach
- Very similar to principle components regression in that it finds

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

and then regresses the outcome against these predictors

Partial least squares

- The key difference is how the weights $\{\phi_{jm}\}$ are found
- Let's find the first PLS direction
- For each covariate, X_j fit the following simple linear regression model

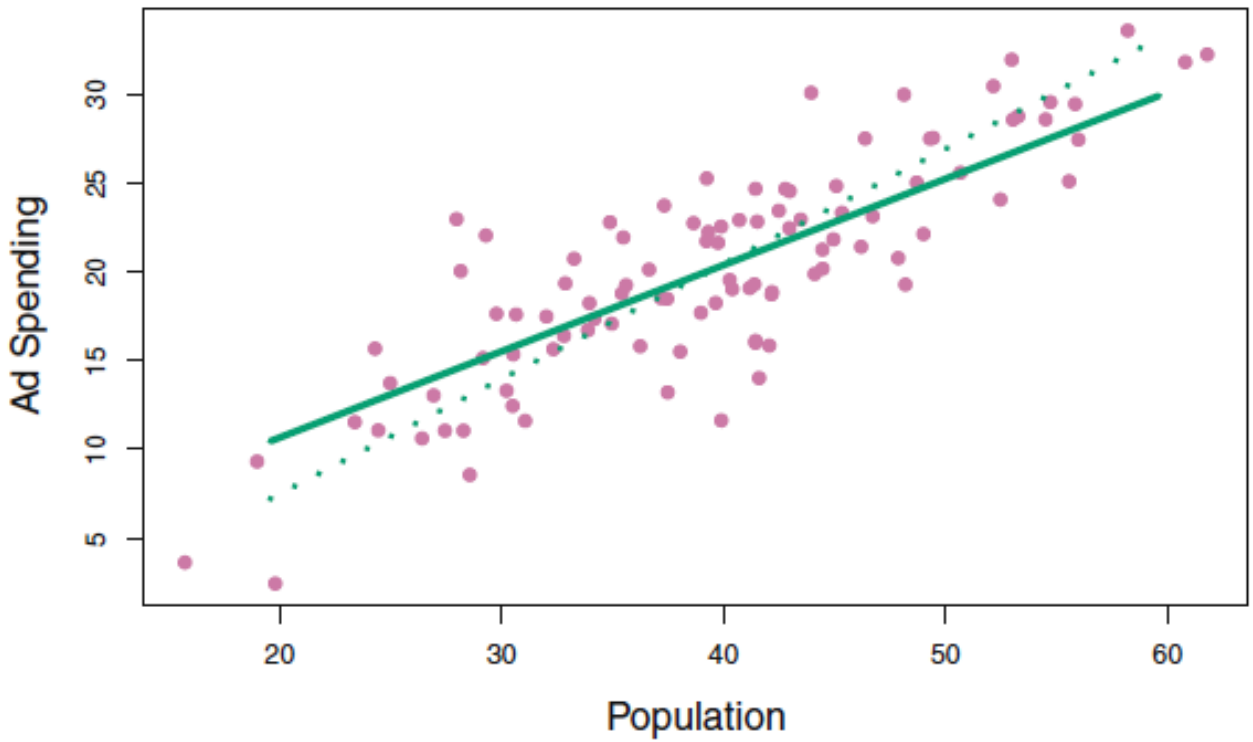
$$E(Y|X_j) = \gamma_{0j} + \gamma_{1j}X_j$$

- Set $\phi_{j1} = \gamma_{1j}$ and create the first transformed variable as

$$Z_1 = \sum_{j=1}^{p} \phi_{j1}X_j$$

Partial least squares

- The first PLS direction is the solid line while the first PCA direction is the dotted line
- PLS has less change in the ad dimension, which indicates that population is more associated with the outcome



James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An introduction to statistical learning. New York: springer.

Partial least squares

- To find the second PLS direction we first regress each covariate X_j against Z_1 using

$$E(X_j|Z_1) = \alpha_{0j} + \alpha_{1j}Z_1$$

- We then take the residuals from this model

$$X^*_{ij} = X_{ij} - \alpha_{0j} + \alpha_{1j}Z_{i1}$$

- This is the information in X_j that is not already captured by Z_1

Partial least squares

- We then perform the same operation as for the first set of PLS weights, but with these residuals

$$E(Y|X_j^*) = \gamma_{0j} + \gamma_{1j}X_j^*$$

- Set $\phi_{j2} = \gamma_{12}$ and create the second transformed variable as

$$Z_2 = \sum_{j=1}^{p} \phi_{j2}X_j$$

- This process can be repeated until M PLS directions are found

Partial least squares

- PLS can help reduce bias of principle components regression by including outcome information
- Many times this comes at a cost of increased variance
  - Doesn't necessarily perform better than PCA
  - Depends on the specific problem
- These methods can be particularly helpful when covariates are highly correlated
- If covariates are not very correlated, then ridge/lasso will likely perform better