Lecture 4: Principal Component Analysis and Regression

UNIVERSITY OF SAN FRANCISCO

James D. Wilson

MATH 373

Plan for this Lecture

- Dimension Reduction via Principal Component Analysis
- Principal Component Regression for High Dimensional settings
- Algorithms:
  - Principal Component Analysis (PCA)
  - Principal Component Regression (PCR)
- Warning: Linear Algebra heavy!

Reference: ISL Sections 6.3 - 6.4; 10.2

Regression Methods So Far

Setting: Continuous response y and predictors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p$

So Far: Linear regression using some (or all) of the original predictors:

$$y_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i, \qquad i = 1, \ldots, n$$

- Ordinary Least Squares: with model selection
- Shrinkage methods: Ridge, Lasso, Elastic Net

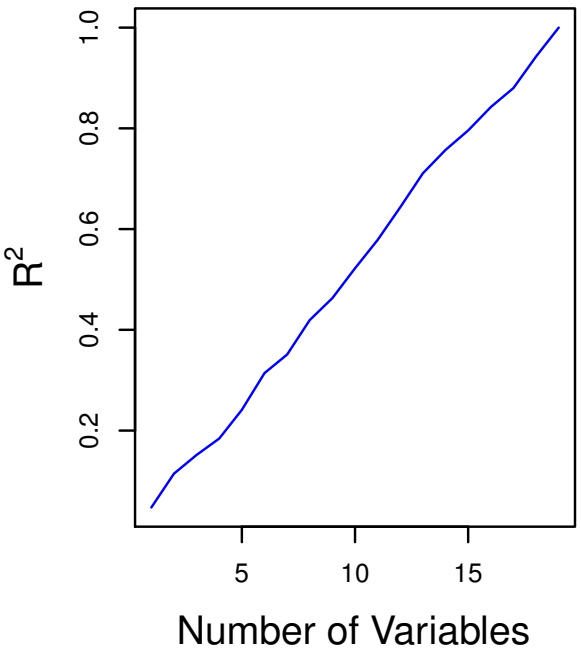
Regression Methods So Far

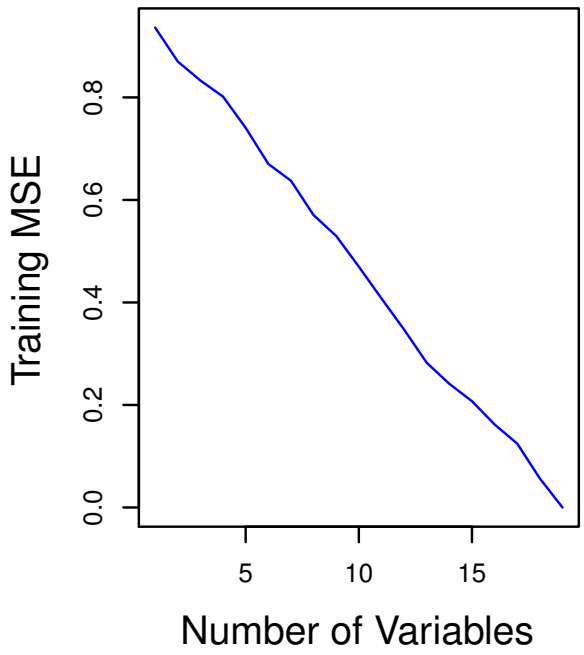
Potential concerns: High-dimensional setting ($p >> n$), or correlations among variables

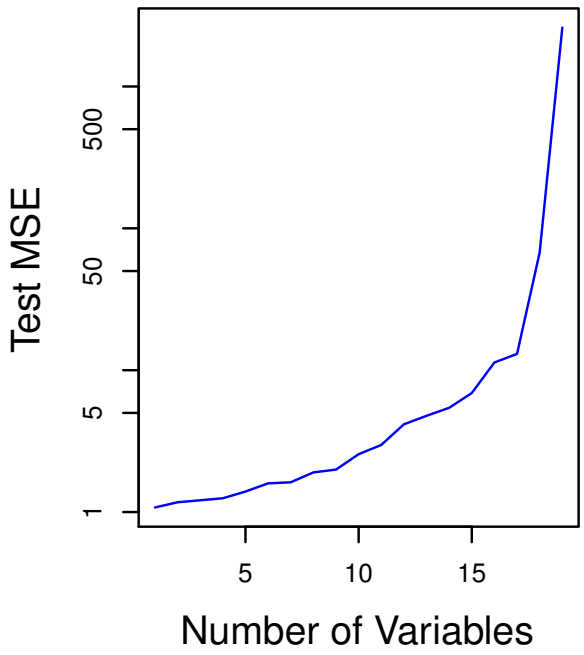
- $(X^T X)$ is no longer full rank (and therefore non-invertible)
- The curse of dimensionality refers to the fact that including more predictors **does not** improve the prediction of capabilities of a model even though the MSE in the training set may lead you to think otherwise.
- Note: Shrinkage and model selection tools *can* reduce p

Curse of Dimensionality

Example: $n = 20$ training observations







Dimension Reduction Methods

Aim: Reduce the dimension p to some $M < n$

Idea: Transform, or project, variables $\mathbf{x}_1, \ldots, \mathbf{x}_p$ to a lower dimensional space $\mathbf{z}_1, \ldots, \mathbf{z}_M$ where

$$\mathbf{z}_m = \sum_{j=1}^{p} \phi_{jm} \mathbf{x}_j$$

for some constants $\phi_{1m}, \phi_{2m}, \ldots, \phi_{pm}$, $m = 1, \ldots, M$

Dimension Reduction Methods

(New?) Model:

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i$$

Tis new, but there is a close relationship with the standard model:

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

where: $\beta_0 = \theta_0$, and $\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm}$ for $j = 1, \ldots, p$

Dimension Reduction Methods

Results:

- The estimated β_j coefficients must take form

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

- Bias is increased
- When $p > n$, selecting a value $M << p$ can significantly reduce variance of the coefficients
- When $M = p$, fitting the new model is equivalent to fitting OLS estimates for β

Using a Dimension Reduction Method

Requires 2 steps:

1. Transformation: transform $\mathbf{x}_1, \ldots, \mathbf{x}_p$ to $\mathbf{z}_1, \ldots, \mathbf{z}_M$
2. Fitting the model: Fit the model

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i$$

using least squares (i.e. minimizing the MSE in the training set)

Focus: principal component regression, which regresses y onto the *directions of highest variability* in the data matrix X. First, we will describe how to find these *directions of highest variability* in X via principal component analysis.

Principal Component Analysis

Idea: Reduce the dimension of the $n \times p$ matrix X.

How: Project X onto an $M < p$ dimensional space $\{\mathbf{z}_1, \ldots, \mathbf{z}_M\}$ so that

- $\mathbf{z}_i^T \mathbf{z}_j = 0$, for all $i \neq j$ (orthogonality)
- $\mathbf{z}_i^T \mathbf{z}_i = 1$, for all $i = 1, \ldots, M$
- $\mathbf{z}_1, \ldots, \mathbf{z}_k$ explain the most variability in X possible for a k-dimensional space subject to the above 2 constraints. Equivalently, the projection of X is as close as possible to the subspace of $\mathbf{z}_1, \ldots, \mathbf{z}_k$

Principal Component Analysis

Fact: It turns out that $\mathbf{z}_m$ can be expressed as linear combinations of the columns of X:

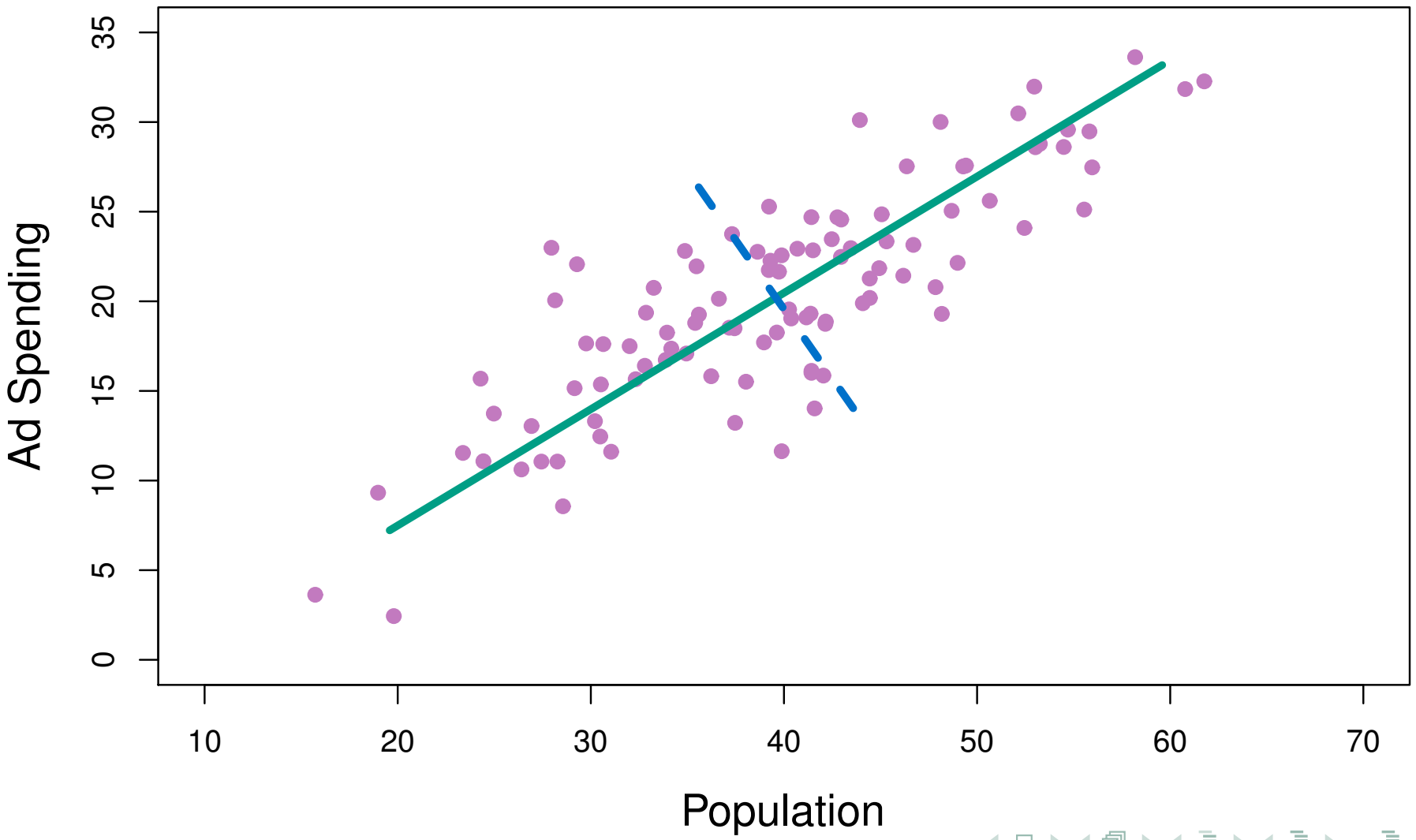
$$\mathbf{z}_m = \sum_{j=1}^{p} \phi_{jm} \mathbf{x}_j$$

where $\sum_{j=1}^{p} \phi_{jm}^2 = 1$

Terminology:

- $\mathbf{z}_1, \ldots, \mathbf{z}_M$ are called the principal components (PCs)
- $z_{11}, \ldots, z_{n1}$ are the principal component scores of the first PC
- $\phi_{11}, \ldots, \phi_{p1}$ are the principal component loadings of the first PC

Visualization of PCs



Principal Component Analysis

- This is the first unsupervised method we've discussed. That is, there is no response affecting our analysis of the data X.
- We are focusing now on the transformation of X to a lower-dimensional space
- **Big Question**: How do we choose the transformation?!
- **Big Question** (for PCA): How do we *maximize* the variation in X?

Empirical Covariance Matrix

Data matrix: X

The covariance of variable j and variable k is

$$\text{Cov}(\mathbf{x}_j, \mathbf{x}_k) = \frac{1}{n}\sum_{i=1}^{n}(x_{ij} - \overline{\mathbf{x}}_j)(x_{ik} - \overline{\mathbf{x}}_k) = \frac{1}{n}\sum_{i=1}^{n} x_{ij}\, x_{ik} - \overline{\mathbf{x}}_j\, \overline{\mathbf{x}}_k$$

where $\overline{\mathbf{x}}_j$ is a scalar: $\overline{\mathbf{x}}_j = \sum_{i=1}^{n} x_{ij}$

Definition: Empirical Covariance Matrix (for variables)

$$\Sigma := \{\text{Cov}(\mathbf{x}_j, \mathbf{x}_k) : 1 \leq j, k \leq p\}$$

Matrix Formula for Covariance

Goal: Express Σ in terms of the Gram matrix $X^T X$.

Fact:

$$\Sigma = n^{-1} X^T X - \mathbf{u}\,\mathbf{u}^T$$

where $\mathbf{u}^T = (\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_p)$ is the vector of column means of X.

Matrix Formula for Covariance

Assumption: the vector of column means $\mathbf{u} = 0$. (column center the data)

Then, $\Sigma = n^{-1} X^T X$.

Properties of Σ

1. Σ is $p \times p$, symmetric, and non-negative definite
2. $\text{rank}(\Sigma) = \text{rank}(X^T X) = \text{rank}(X) \leq \min(n, p)$
3. Σ has real eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$.
4. If $p > n$ then $\text{rank}(\Sigma) < p$ and Σ is not invertible.

Total Variation and the Trace of Σ

Total Variation of X

Basic facts about the trace imply

$$\begin{aligned}
\sum_{k=1}^{p} \lambda_k &= \operatorname{tr}(\Sigma) = \frac{1}{n}\operatorname{tr}(X^T X) \\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} x_{ij}^2 \\
&= \sum_{j=1}^{p}\left(\frac{1}{n}\sum_{i=1}^{n} x_{ij}^2\right) = \sum_{j=1}^{p} \operatorname{Var}(\mathbf{x}_j) \\
&= \text{The } \textit{total variation} \text{ of } X
\end{aligned}$$

Variation and the Trace of Σ

Magnitude of the Samples

$$\begin{aligned}
\sum_{k=1}^{p} \lambda_k &= \operatorname{tr}(\Sigma) = \frac{1}{n}\operatorname{tr}(X^T X) \\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p} x_{ij}^2 \\
&= \frac{1}{n}\sum_{i=1}^{n} \|x_i\|^2 \\
&:= \text{The average squared norm of the samples}
\end{aligned}$$

Approximating a Set of Vectors

Given: Vectors $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^p$ centered so that $\sum_i \mathbf{u}_i = \mathbf{0}$ (Think about the rows of X once X is column-centered)

Goal: Find a low dimensional summary of $\mathbf{u}_1, \ldots, \mathbf{u}_n$, more precisely, a subspace V of $\mathbb{R}^p$ such that

- $\dim(V)$ much less than p (and n)
- projection of $\mathbf{u}_j$ onto V is close to $\mathbf{u}_j$

Note: Smallest subspace of $\mathbb{R}^p$ *containing* $\{\mathbf{u}_i\}$ is

$$V = \text{span}(\mathbf{u}_1, \ldots, \mathbf{u}_n) \quad \text{with} \quad \dim(V) \leq n$$

Approximating a Set of Vectors

Consider approximating subspace V of dimension 1, equivalently,

$$V = \{\alpha \mathbf{u}_0 : \alpha \in \mathbb{R}\} \text{ some } \mathbf{u}_0 \in \mathbb{R}^p \text{ with } \|\mathbf{u}_0\| = 1.$$

Definition: The projection of $\mathbf{u}$ onto V is $(\mathbf{u}^T \mathbf{u}_0)\, \mathbf{u}_0$.

Two (Complementary) Goals:

1. Find $\mathbf{u}_0$ to maximize $\text{Var}(\{\mathbf{u}_1^T \mathbf{u}_0, \ldots, \mathbf{u}_n^T \mathbf{u}_0\})$

1. Find $\mathbf{u}_0$ to minimize $n^{-1} \sum_{i=1}^n \|\mathbf{u}_i - (\mathbf{u}_i^T \mathbf{u}_0)\, \mathbf{u}_0\|^2$

Variance of Projections

By definition of the variance:

$$\begin{aligned}
\text{Var}(\{\mathbf{u}_1^T\mathbf{u}_0,\ldots,\mathbf{u}_n^T\mathbf{u}_0\}) &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_i^T\mathbf{u}_0)^2 - \Big[\frac{1}{n}\sum_{i=1}^{n}\mathbf{u}_i^T\mathbf{u}_0\Big]^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_i^T\mathbf{u}_0)^2 - \Big[\frac{1}{n}\mathbf{u}_0^T\Big(\sum_{i=1}^{n}\mathbf{u}_i\Big)\Big]^2 \\
&= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_i^T\mathbf{u}_0)^2
\end{aligned}$$

The last equality follows since $\sum_i \mathbf{u}_i = \mathbf{0}$ (data are centered).

Sum of Squares Fit

Consider sum of squares. Completing square of ith term gives

$$\begin{aligned} \|\mathbf{u}_i - (\mathbf{u}_i^T \mathbf{u}_0)\,\mathbf{u}_0\|^2 &= \|\mathbf{u}_i\|^2 - 2\,(\mathbf{u}_i^T \mathbf{u}_0)^2 + (\mathbf{u}_i^T \mathbf{u}_0)^2\,\|\mathbf{u}_0\|^2 \\ &= \|\mathbf{u}_i\|^2 - (\mathbf{u}_i^T \mathbf{u}_0)^2 \end{aligned}$$

Therefore

$$\begin{aligned} \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{u}_i - (\mathbf{u}_i^T \mathbf{u}_0)\,\mathbf{u}_0\|^2 &= \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{u}_i\|^2 - \frac{1}{n}\sum_{i=1}^{n} (\mathbf{u}_i^T \mathbf{u}_0)^2 \\ &= \frac{1}{n}\sum_{i=1}^{n} \|\mathbf{u}_i\|^2 - \mathrm{Var}(\{\mathbf{u}_i^T \mathbf{u}_0\}) \qquad (1) \end{aligned}$$

Conclusion: *Choosing* $\mathbf{u}_0$ *to minimize the sum of squares fit is equivalent to maximizing variance of projection lengths.*

Empirical Covariance Matrix

Define $X = n \times p$ matrix with rows $\mathbf{u}_1^T, \ldots, \mathbf{u}_n^T$.

Let $\Sigma = n^{-1} X^T X$ be $p \times p$ covariance matrix of X. Then,

$$\begin{aligned}
\text{Var}(\{\mathbf{u}_i^T \mathbf{u}_0\}) &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_i^T \mathbf{u}_0)^2 = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{u}_0^T \mathbf{u}_i)(\mathbf{u}_i^T \mathbf{u}_0) \\
&= \frac{1}{n}\sum_{i=1}^{n}\mathbf{u}_0^T(\mathbf{u}_i \mathbf{u}_i^T)\mathbf{u}_0 = \mathbf{u}_0^T\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{u}_i \mathbf{u}_i^T\right)\mathbf{u}_0 \\
&= \mathbf{u}_0^T\left(\frac{1}{n}X^T X\right)\mathbf{u}_0 = \mathbf{u}_0^T \Sigma \mathbf{u}_0
\end{aligned}$$

Best 1-Dimensional Subspace

Upshot: Variance of the projections $\{\mathbf{u}_1^T \mathbf{u}_0, \ldots, \mathbf{u}_n^T \mathbf{u}_0\}$ onto $\mathbf{u}_0$ is equal to $\mathbf{u}_0^T \Sigma \mathbf{u}_0$. Maximized when $\mathbf{u}_0$ is the leading eigenvector of Σ.

Let $\lambda_1 \geq \cdots \geq \lambda_p \geq 0$ be the eigenvalues of Σ, with corresponding orthonormal eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$.

Fact: The best one dimensional approximation for $\mathbf{u}_1, \ldots, \mathbf{u}_n$ is obtained by projecting the data vectors onto the line

$$V_1 = \text{span}\{\mathbf{v}_1\},$$

in the direction of the principal eigenvector of Σ.

Approximation Error of V_1

By (1) and fact that $\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_i\|^2 = \mathrm{tr}(\Sigma)$, approximation error is

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_i - (\mathbf{u}_i^T\mathbf{v}_1)\,\mathbf{v}_1\|^2 &= \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{u}_i\|^2 - \mathbf{v}_1^T\Sigma\,\mathbf{v}_1 \\ &= \mathrm{tr}(\Sigma) - \lambda_1 = \sum_{i=1}^{p}\lambda_i - \lambda_1 = \sum_{i=2}^{p}\lambda_i\end{aligned}$$

Residual error after projecting onto $\mathbf{v}_1$ is sum of the remaining eigenvalues 2 through p.

Higher Order PCs

For $1 \leq d \leq p$ the d-dimensional subspace V of $\mathbb{R}^p$ minimizing

$$\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{u}_i - \text{proj}_V(\mathbf{u}_i)\|^2$$

is $V_d = \text{span}\{\mathbf{v}_1, \ldots, \mathbf{v}_d\}$ = span of d leading eigenvectors of Σ.

In this case, the projection of $\mathbf{u}$ onto V_d is

$$\text{proj}_{V_d}(\mathbf{u}) = \sum_{j=1}^{d} (\mathbf{u}^T \mathbf{v}_j)\, \mathbf{v}_j$$

and the approximation error of V_d is given by

$$\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{u}_i - \text{proj}_{V_d}(\mathbf{u}_i)\|^2 = \sum_{i=d+1}^{p} \lambda_i$$

PCA: Bringing it all together

Recall: Given X, we'd like to project columns $\mathbf{x}_1, \ldots, \mathbf{x}_p$ down to a lower-dimensional space $\mathbf{z}_1, \ldots, \mathbf{z}_M$ where

$$\mathbf{z}_m = \sum_{j=1}^{p} \phi_{jm} \mathbf{x}_j$$

Let $\mathbf{u}_j$ = the jth row of X.

- Principal component loadings (directions): $\phi_i = (\phi_{11}, \phi_{21}, \ldots, \phi_{p1})$
  Use $\mathbf{v}_i$ = the ith eigenvector of $\Sigma = n^{-1} X^T X$

- Principal components: $\mathbf{z}_1, \ldots, \mathbf{z}_M$ where

$$\mathbf{z}_j = X \mathbf{v}_j$$

- Principal component scores: $z_{\ell,j} = \mathbf{u}_\ell^T \mathbf{v}_j$

PCA: Important Considerations

- Method is unsupervised. There is *no* dependence on any response data y.
- Before application, the columns of X must be centered.
- Principal components are uncorrelated (hence no multicollinearity)
- No rigorous way to choose the number of PCs to use, but there is a heuristic (see next slide).

Choosing the number of PCs

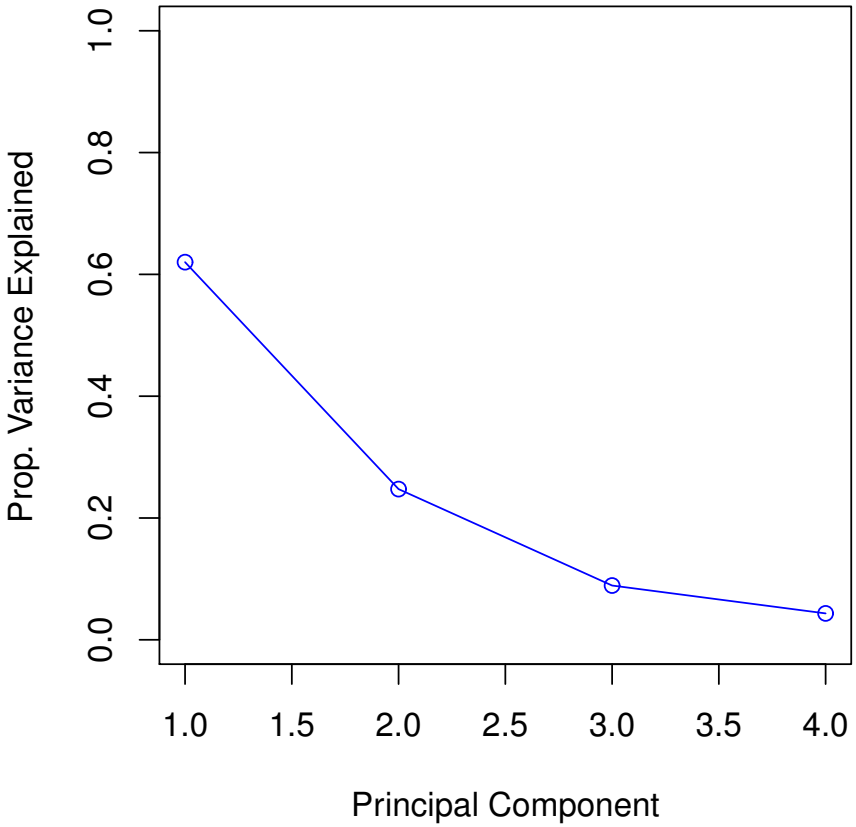
Definition: The percentage of variation (PVE) captured by the first d principal components, equivalently the subspace V_d, is

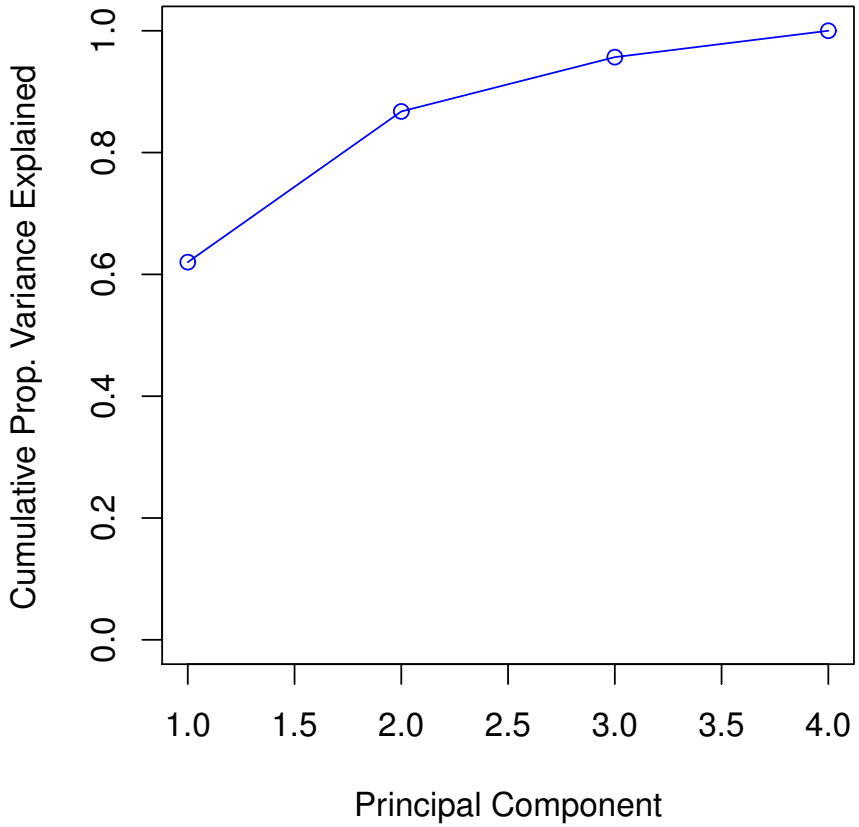
$$\frac{\sum_{i=1}^{d} \lambda_i}{\sum_{j=1}^{p} \lambda_j} \times 100$$

We choose the number of PCs by evaluating the scree plot: a plot of the number of PCs against the PVE.

Example of Scree Plot

Aim: capture a significant (enough) amount of variability in X





Example: TCGA Gene Expression Data

- Heat map of gene expression data from The Cancer Genome Atlas (TCGA)
  - Samples
    - 95 Luminal A breast tumors
    - 122 Basal breast tumors
  - Variables: 2000 randomly selected genes

Question: can the PCs of this gene expression data help distinguish cancer subtypes?

PCA on TCGA Expression Data

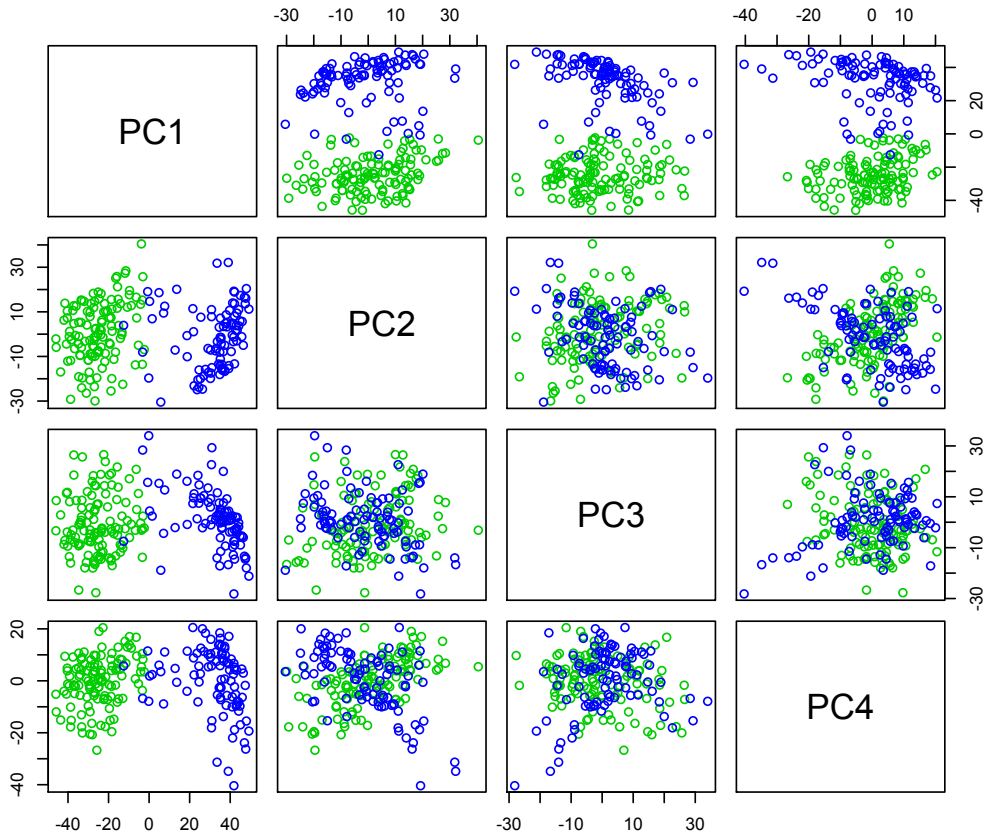
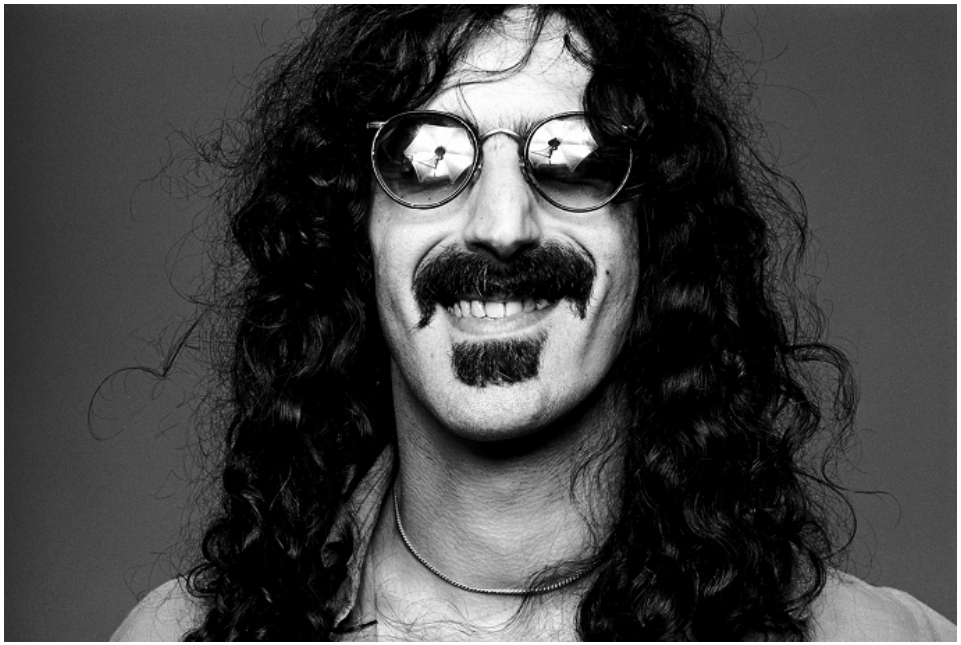


Figure: Projections of Sample data onto the first four principal components of the TCGA dataset. Colors represent subtype of cancer: Luminal A and Basal

Example: Image Data



Data: $X = 396 \times 588$ matrix of pixel intensities

Question: Can we project columns of the image onto a low dimensional subspace and still reconstruct the image?

Image Reconstruction

d = 1, PVE = 80.42

d = 3, PVE = 88.91

d = 5, PVE = 92.99

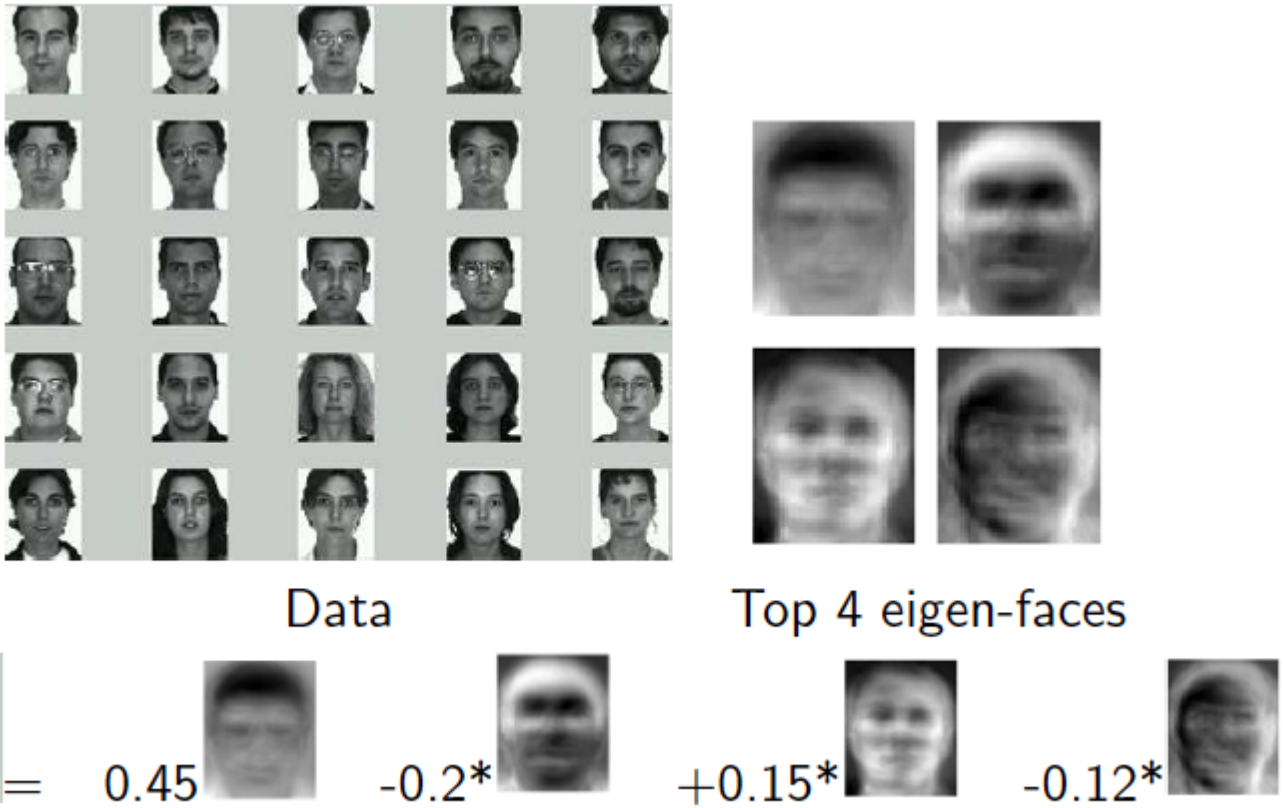
d = 10, PVE = 95.79

d = 20, PVE = 97.24

d = 40, PVE = 98.18

EigenFace, EigenGene, EigenEverything

Eigen-faces Example:



PCA on ... pancake recipes?!

From ”Cooking for Geeks: Real Science, Great Hacks, and Good Food”

Every pancake recipe is a scaled version of this eigen-pancake recipe:

- 1 1/2 cups flour
- 2 tablespoons sugar
- 2 teaspoons baking powder
- 1/2 teaspoon salt
- 2 tablespoons butter
- 11/4 cups milk
- 2 small eggs

Back to Regression

Standard Model:

$$y_i = \sum_{j=1}^{p} \beta_j x_{ij} + \epsilon_i$$

PCR Assumption: the directions in which $\mathbf{x}_1, \ldots, \mathbf{x}_p$ show the most variation are the directions that are associated with y.

1. Transformation: transform $\mathbf{x}_1, \ldots, \mathbf{x}_p$ to $\mathbf{z}_1, \ldots, \mathbf{z}_M$.
   - Column center X
   - Set $\mathbf{z}_m$ = the mth eigenvector of $n^{-1} X^T X$
   - Choose $M \leq p$ using PVE

Principal Component Regression

2. Fitting the model: Fit the model

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i$$

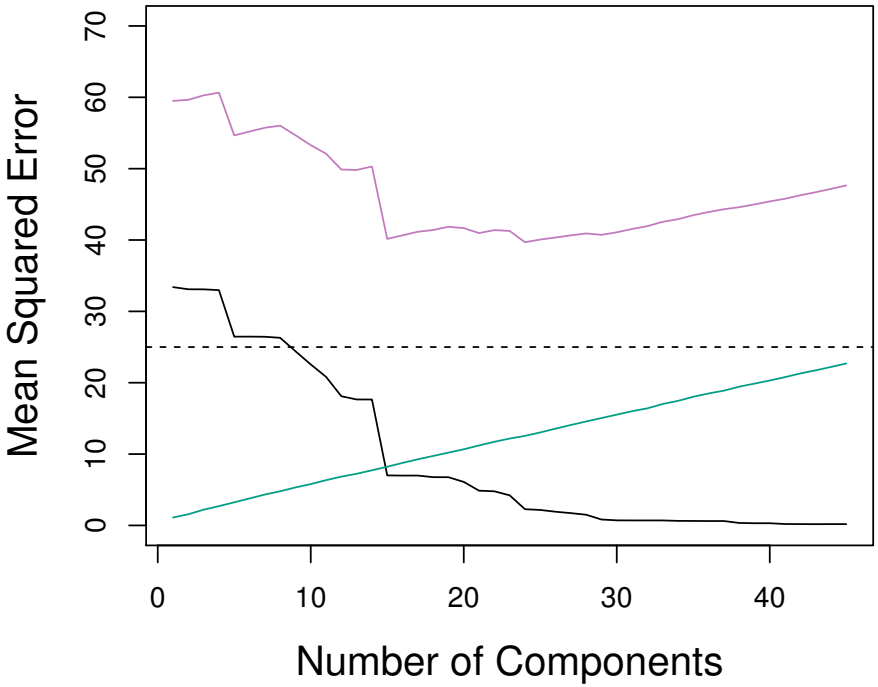
- Standard least squares regression can be used

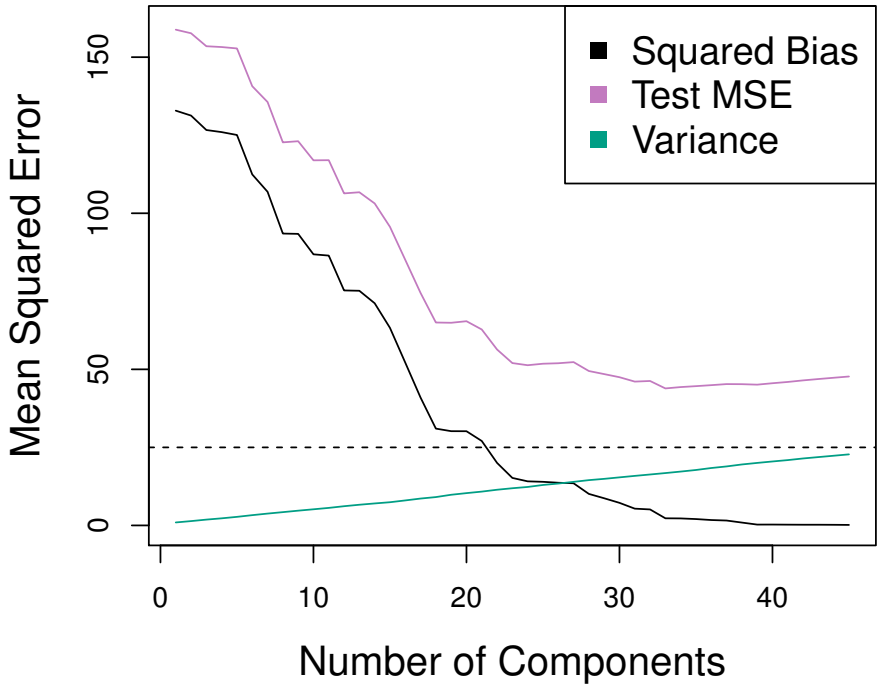
Training vs. Test set analysis

Training Set: Use to calculate the PC loadings $\phi_1, \ldots, \phi_M$. Then fit model by estimating $\theta_0, \ldots, \theta_M$.

Test Set: Project data onto PC loadings to obtain principal component scores. Then, evaluate goodness of fit of model using $\theta_0, \ldots, \theta_M$ from the Training data.

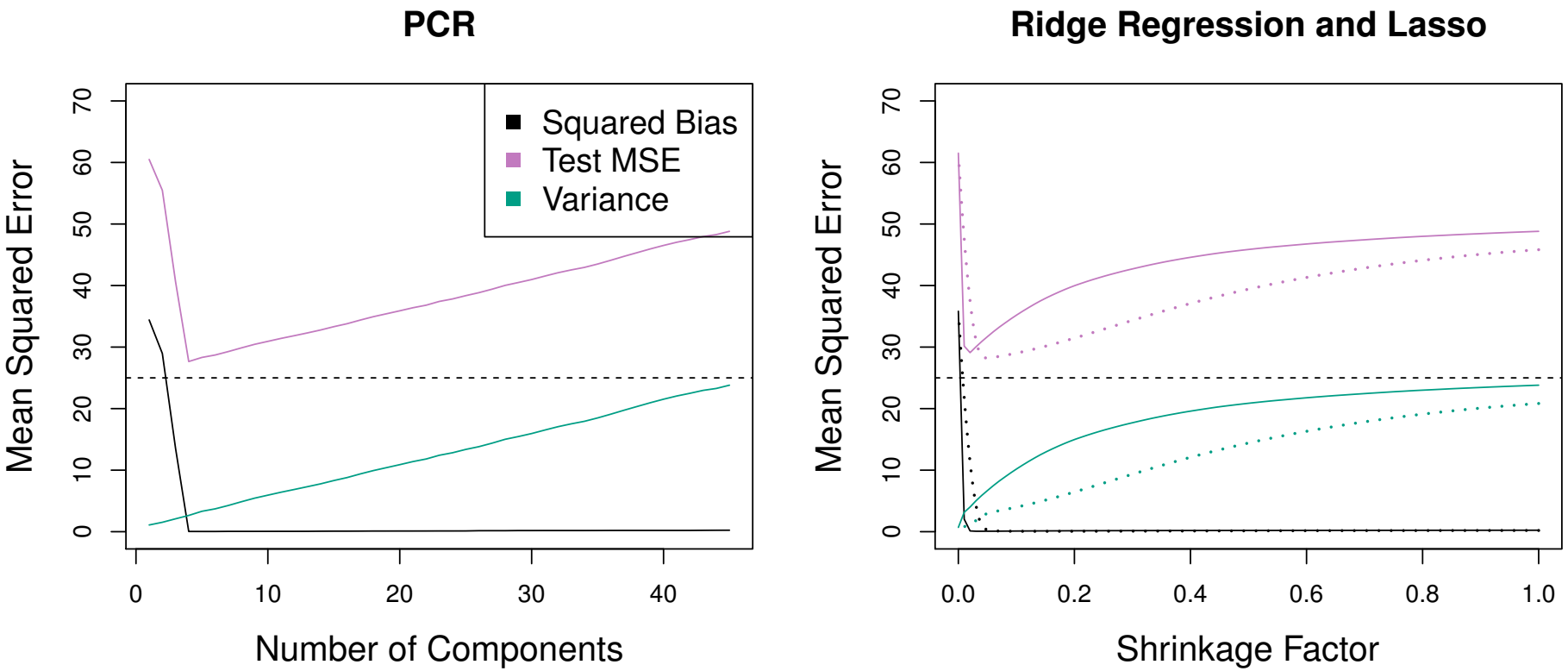
Example of PCR





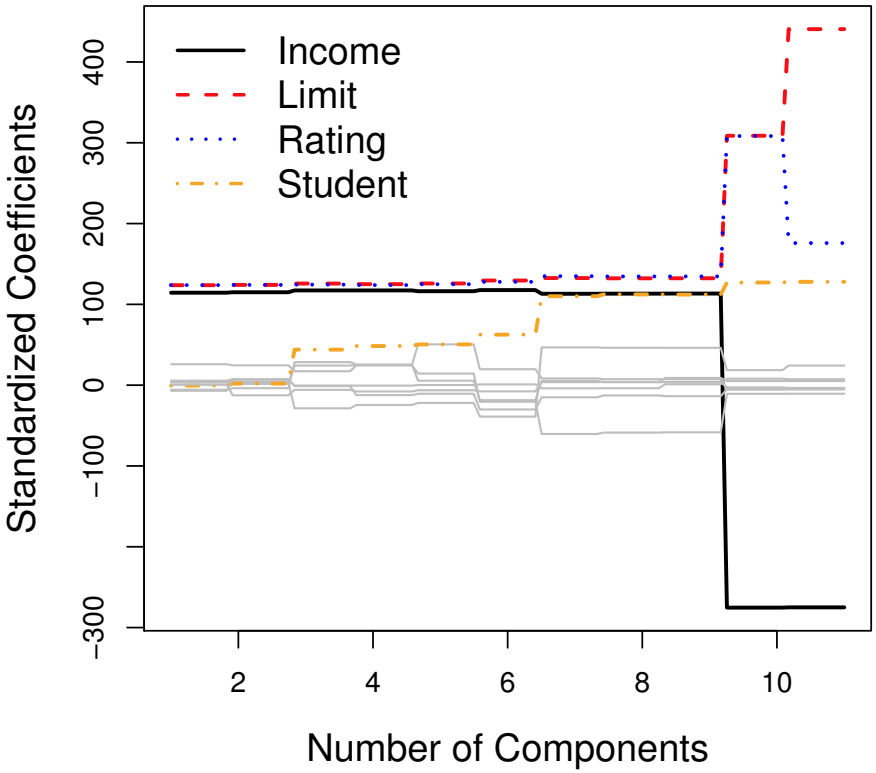
Two simulated data sets

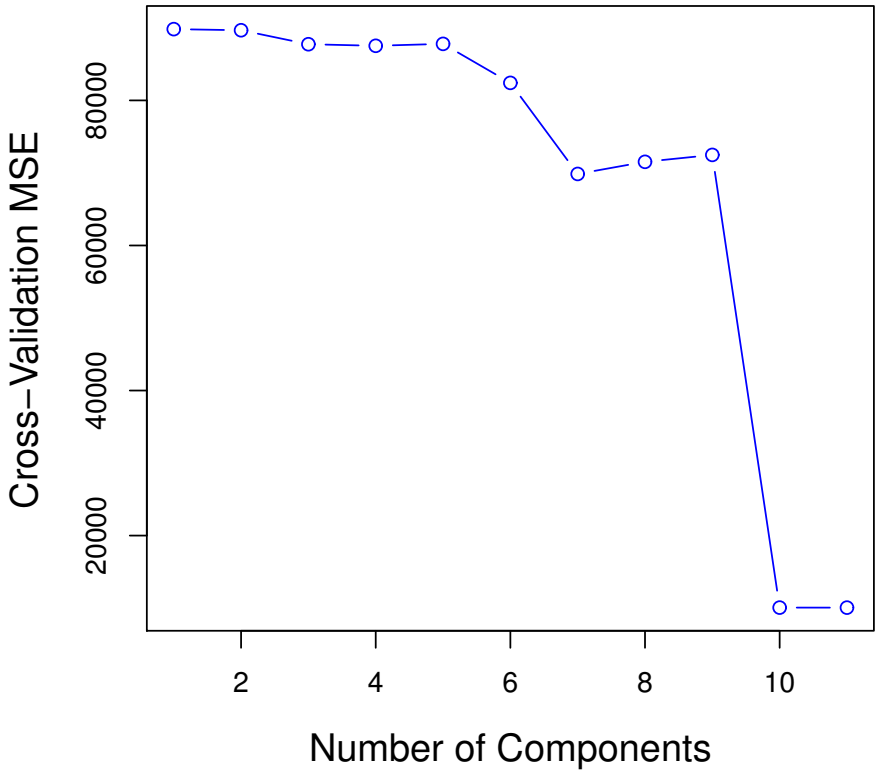
Example of PCR



Simulated data set where true predictors were first 5 PCs. (Right): dashed = Ridge, solid = Lasso

Example of PCR on Credit Data





Note the shrinkage effect! Closely related to Ridge regression.

Pros and Cons of PCR

Pros:

- Easily handles high dimensional data $p > n$
- Often leads to a drastic decrease in variance of parameter estimates
- Takes care of issues of correlations between variables (and multicollinearity)

Cons:

- Introduces bias to the parameter estimates
- Hard to interpret coefficients in regression

Implementing PCA and PCR with R

Next we'll show how to implement PCA and PCR in R.