# Lecture 3: Principal Components Analysis (PCA)

Reading: Sections 6.3.1, 10.1, 10.2, 10.4

STATS 202: Data mining and analysis

Sergio Bacallado
September 19, 2018

# Announcements

- Homework 1 is out; due next Thursday.
- Kaggle invitations have been sent. You have to create a Kaggle account *with your Stanford email* and join the competition using the invitation link on the class website.
- If you want to form a team, you should do so before making any submissions.

# The bias variance decomposition

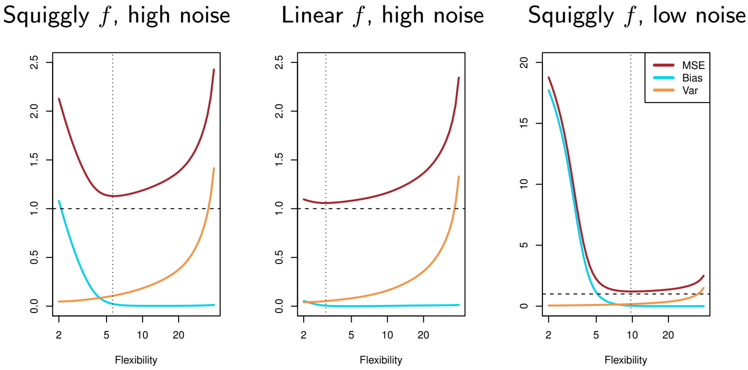
The inputs, $x_1, \ldots, x_n$ are fixed, a test point $x_0$ is also fixed.

$$y_i = f(x_i) + \varepsilon_i \qquad \varepsilon_i \text{ i.i.d, mean } 0.$$

A regression method fit to $(x_1, y_1), \ldots, (x_n, y_n)$ produces the estimate $\hat{f}$. Then, the Mean Squared Error at $x_0$ satisfies:

$$MSE(x_0) = E(y_0 - \hat{f}(x_0))^2 = \text{Var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{Var}(\varepsilon).$$

Both variance and squared bias are always positive, so to minimize the MSE, you must reach a tradeoff between bias and variance.



Squiggly $f$, high noise
Linear $f$, high noise
Squiggly $f$, low noise
MSE
Bias
Var
Flexibility
Flexibility
Flexibility
0.0
0.5
1.0
1.5
2.0
2.5
0
5
10
15
20
2
5
10
20


Figure 2.12

# Classification problems

In a classification setting, the output takes values in a discrete set.

For example, if we are predicting the brand of a car based on a number of variables, the function $f$ takes values in the set $\{\text{Ford, Toyota, Mercedes-Benz}, \ldots\}$.

# Classification problems

In a classification setting, the output takes values in a discrete set.

For example, if we are predicting the brand of a car based on a number of variables, the function $f$ takes values in the set $\{$Ford, Toyota, Mercedes-Benz$, \ldots\}$.

The model:

$$Y = f(X) + \varepsilon$$

becomes insufficient, as $X$ is not necessarily real-valued.

# Classification problems

In a classification setting, the output takes values in a discrete set.

For example, if we are predicting the brand of a car based on a number of variables, the function $f$ takes values in the set $\{\text{Ford, Toyota, Mercedes-Benz}, \ldots\}$.

The model:

$$\cancel{Y = f(X) + \varepsilon}$$

becomes insufficient, as $X$ is not necessarily real-valued.

# Classification problems

In a classification setting, the output takes values in a discrete set.

For example, if we are predicting the brand of a car based on a number of variables, the function $f$ takes values in the set $\{\text{Ford, Toyota, Mercedes-Benz}, \dots\}$.

We will use slightly different notation:

$$\begin{aligned} P(X, Y) &: \text{joint distribution of } (X, Y), \\ P(Y \mid X) &: \text{conditional distribution of } X \text{ given } Y, \\ \hat{y}_i &: \text{prediction for } x_i. \end{aligned}$$

# Loss function for classification

There are many ways to measure the error of a classification prediction. One of the most common is the 0-1 loss:

$$E(\mathbf{1}(y_0 \neq \hat{y}_0))$$

Like the MSE, this quantity can be estimated from training and test data by taking a sample average:

$$\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}(y_i \neq \hat{y}_i)$$

# Bayes classifier

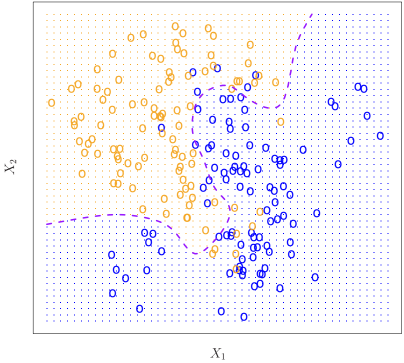



Figure 2.13

In practice, we never know the joint probability $P$. However, we can assume that it exists.

The Bayes classifier assigns:

$$\hat{y}_i = \text{argmax}_j \;\; P(Y = j \mid X = x_i)$$

It can be shown that this is the best classifier under the 0-1 loss.

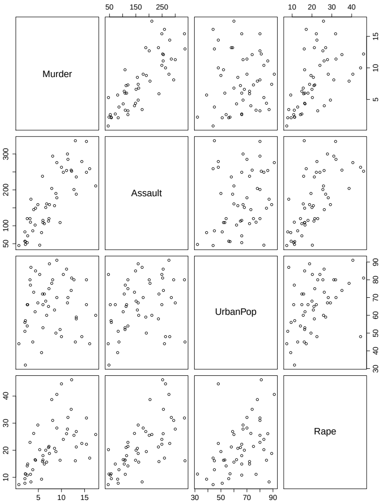
# Principal Components Analysis

- This is the most popular unsupervised procedure ever.
- Invented by Karl Pearson (1901).
- Developed by Harold Hotelling (1933).
- **What does it do?** It provides a way to visualize high dimensional data, summarizing the most important information.

# Principal Components Analysis

- This is the most popular unsupervised procedure ever.
- Invented by Karl Pearson (1901).
- Developed by Harold Hotelling (1933). $\leftarrow$ Stanford pride!
- **What does it do?** It provides a way to visualize high dimensional data, summarizing the most important information.

# What is PCA good for?



# What is PCA good for?

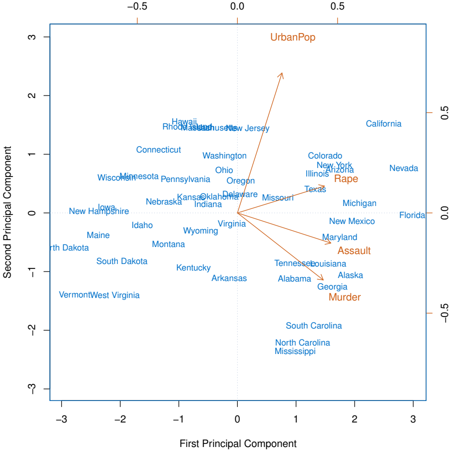
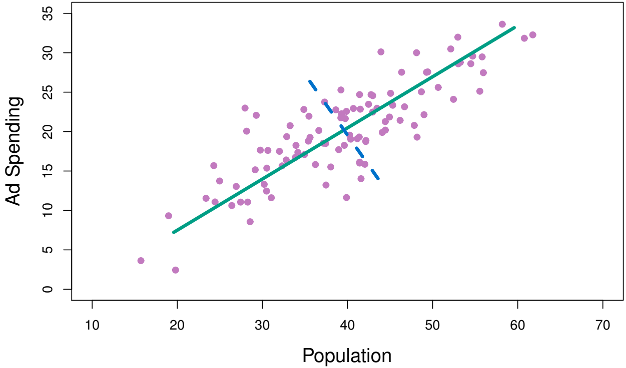



Figure 10.1

# What is the first principal component?

It is the vector which passes the closest to a cloud of samples, in terms of squared Euclidean distance.



i.e. The green direction minimizes the average squared length of the dotted lines.

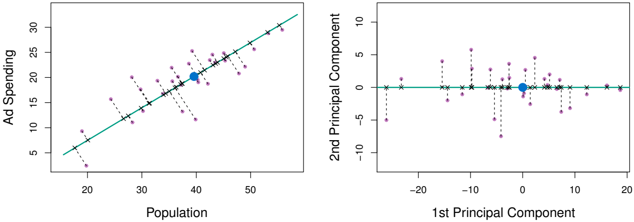
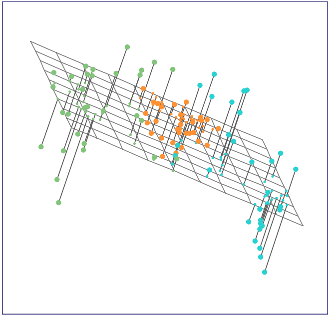



Figure 6.15

## What does this look like with 3 variables?

The first two principal components span a plane which is closest to the data.



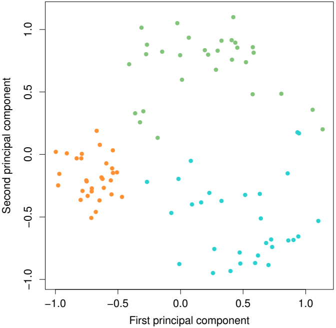



Figure 10.2

# A second interpretation

The projection onto the first principal component is the one with the **highest variance**.

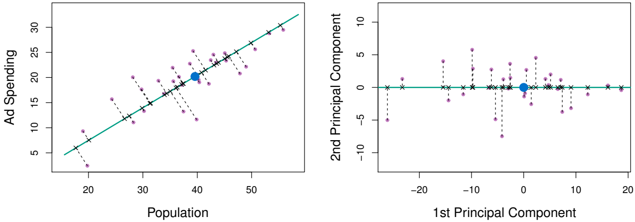



Figure 6.15

# How do we say this in math?

Let $\mathbf{X}$ be a data matrix with $n$ samples, and $p$ variables. From each variable, we subtract the mean of the column; i.e. we **center** the variables.

To find the first principal component $\phi_1 = (\phi_{11}, \ldots, \phi_{p1})$, we solve the following optimization

$$\max_{\phi_{11},\ldots,\phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \right\}$$

$$\text{subject to} \sum_{j=1}^{p} \phi_{j1}^2 = 1.$$

# How do we say this in math?

Let $\mathbf{X}$ be a data matrix with $n$ samples, and $p$ variables. From each variable, we subtract the mean of the column; i.e. we **center** the variables.

To find the first principal component $\phi_1 = (\phi_{11}, \ldots, \phi_{p1})$, we solve the following optimization

$$\max_{\phi_{11},\ldots,\phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \right\}$$

$$\text{subject to} \sum_{j=1}^{p} \phi_{j1}^2 = 1.$$

Variance of the $n$ samples projected onto $\phi_1$.

# How do we say this in math?

Let $\mathbf{X}$ be a data matrix with $n$ samples, and $p$ variables. From each variable, we subtract the mean of the column; i.e. we **center** the variables.

To find the first principal component $\phi_1 = (\phi_{11}, \ldots, \phi_{p1})$, we solve the following optimization

$$\max_{\phi_{11},\ldots,\phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \right\}$$

$$\text{subject to} \sum_{j=1}^{p} \phi_{j1}^2 = 1.$$

Projection of the $i$th sample onto $\phi_1$. Also known as **the score** $z_{i1}$

## How do we say this in math?

To find the second principal component $\phi_2 = (\phi_{12}, \ldots, \phi_{p2})$, we solve the following optimization

$$\max_{\phi_{12},\ldots,\phi_{p2}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j2} x_{ij} \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} \phi_{j2}^2 = 1 \quad \text{and} \quad \sum_{j=1}^{p} \phi_{j1}\phi_{j2} = 0.$$

# How do we say this in math?

To find the second principal component $\phi_2 = (\phi_{12}, \ldots, \phi_{p2})$, we solve the following optimization

$$\max_{\phi_{12},\ldots,\phi_{p2}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j2} x_{ij} \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} \phi_{j2}^2 = 1 \quad \text{and} \quad \sum_{j=1}^{p} \phi_{j1}\phi_{j2} = 0.$$

First and second principal components must be orthogonal.

# How do we say this in math?

To find the second principal component $\phi_2 = (\phi_{12}, \ldots, \phi_{p2})$, we solve the following optimization

$$\max_{\phi_{12},\ldots,\phi_{p2}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left( \sum_{j=1}^{p} \phi_{j2} x_{ij} \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} \phi_{j2}^2 = 1 \quad \text{and} \quad \sum_{j=1}^{p} \phi_{j1}\phi_{j2} = 0.$$

First and second principal components must be orthogonal.

Equivalent to saying that the scores $(z_{11}, \ldots, z_{n1})$ and $(z_{12}, \ldots, z_{n2})$ are uncorrelated.

# Solving the optimization

This optimization is fundamental in linear algebra. It is satisfied by either:

- The singular value decomposition (SVD) of $\mathbf{X}$:

  $$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\boldsymbol{\Phi}^T$$

  where the $i$th column of $\boldsymbol{\Phi}$ is the $i$th principal component $\phi_i$, and the $i$th column of $\mathbf{U}\boldsymbol{\Sigma}$ is the $i$th vector of scores $(z_{1i}, \ldots, z_{ni})$.

- The eigendecomposition of $\mathbf{X}^T\mathbf{X}$:

  $$\mathbf{X}^T\mathbf{X} = \boldsymbol{\Phi}\boldsymbol{\Sigma}^2\boldsymbol{\Phi}^T$$

# PCA in practice: The biplot

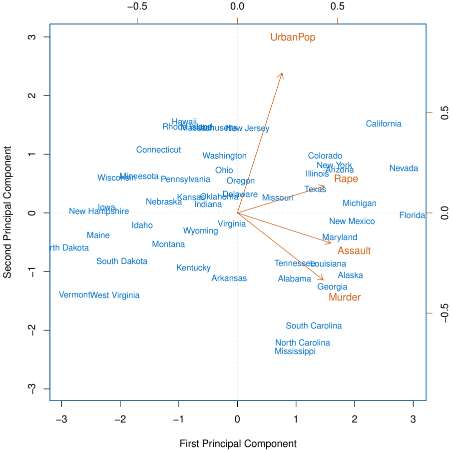



Figure 10.1

# Scaling the variables

Most of the time, we don't care about the absolute numerical value of a variable. We care about the value relative to the spread observed in the sample.

Before PCA, in addition to **centering** each variable, we also multiply it times a constant to make its variance equal to 1.

# Example: scaled vs. unscaled PCA

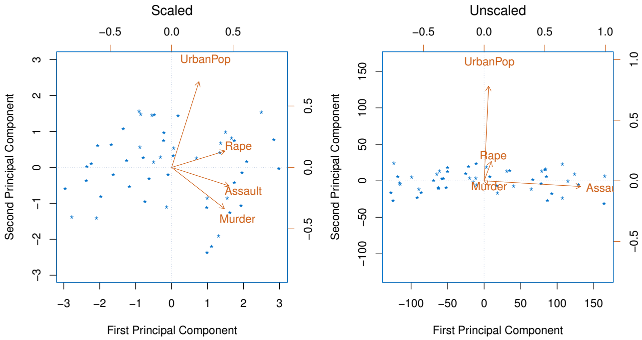


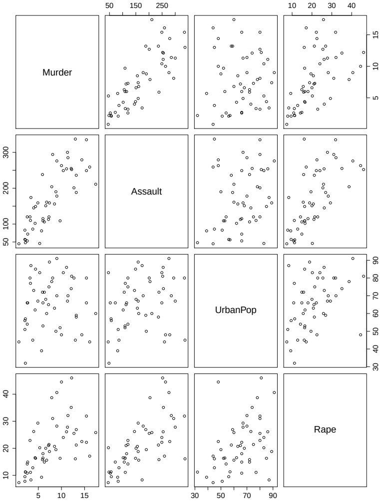

Figure 10.3

# Scaling the variables

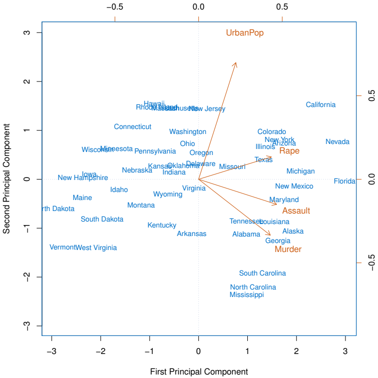
In special cases, we have variables measured in the same unit; e.g. gene expression levels for different genes.

Therefore, we care about the absolute value of the variables and we can perform PCA without scaling.

# How many principal components are enough?



# How many principal components are enough?




We said 2 principal components capture most of the relevant information. But how can we tell?

# The proportion of variance explained

We can think of the top **principal components** as directions in space in which the data vary the most.

The $i$th **score vector** $(z_{1i}, \ldots, z_{ni})$ can be interpreted as a *new* variable. The variance of this variable decreases as we take $i$ from 1 to $p$. However, the total variance of the score vectors is the same as the total variance of the original variables:

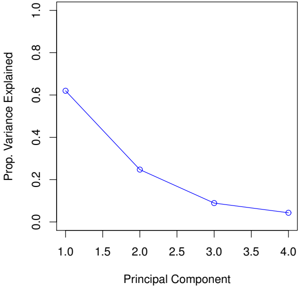
$$\sum_{i=1}^{p} \frac{1}{n} \sum_{j=1}^{n} z_{ji}^2 = \sum_{k=1}^{p} \mathsf{Var}(x_k).$$

We can quantify how much of the variance is captured by the first $m$ principal components/score variables.

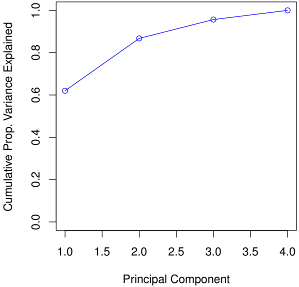
# The proportion of variance explained

The variance of the $m$th score variable is:

$$\frac{1}{n}\sum_{i=1}^{n} z_{im}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\phi_{jm}x_{ij}\right)^2 = \frac{1}{n}\Sigma_{mm}^2.$$




Scree plot



# Generalizations of PCA

PCA works under a Euclidean geometry in the space of variables. Often, the natural geometry is different:

- We expect some variables to be "closer" to each other that to other variables.
- Some correlations between variables would be more surprising than others.

Examples:

- Variables are pixel values, samples are different images of the brain. We expect neighboring pixels to have stronger correlations.
- Variables are rainfall measurements at different regions. We expect neighboring regions to have higher correlations.

# Generalizations of PCA

There are ways to include this knowledge in a PCA. See:

1. Susan Holmes. *Multivariate Analysis, the French way.* (2006).
2. Omar de la Cruz and Susan Holmes. *An introduction to the duality diagram*. (2011).
3. Stéphane Dray and Thibaut Jombart. *Revisiting Guerry's data: Introducing spatial constraints in multivariate analysis.* (2011).
4. Genevera Allen, Logan Grosenick, and Jonathan Taylor. *A Generalized Least Squares Matrix Decomposition*. (2011).