BIOST 546: Machine Learning for Biomedical Big Data

Ali Shojaie

Lecture 9: Dimension Reduction - Part I
Spring 2017

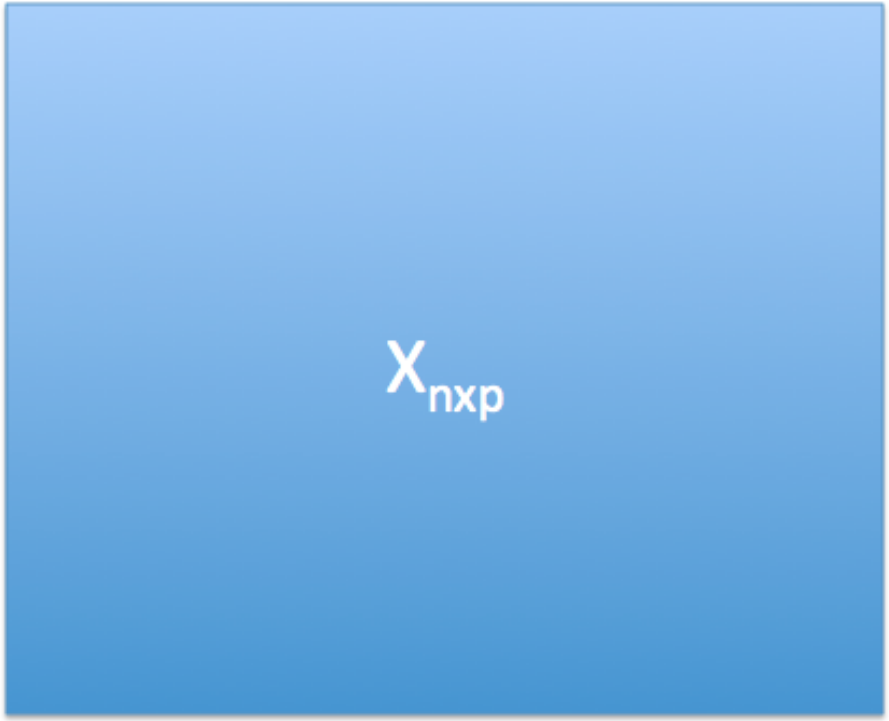
Recap

- High-dimensional inference
- Adjusting for multiple comparisons
  - FWER
  - FDR
  - Permutation-based approaches
  - SAM

Today's Class

- Dimension reduction methods
- PCA

Unsupervised Learning: A Reminder



- n number of observations
- p number of variables/features
- no *response* variable
- In biological applications, often $p \gg n$

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?
- Can individuals' SNP profiles be used to learn about their ethnic/racial backgrounds?

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?
- Can individuals' SNP profiles be used to learn about their ethnic/racial backgrounds?
- Can we find cancer subtypes based on gene/metabolic expression patterns?

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?
- Can individuals' SNP profiles be used to learn about their ethnic/racial backgrounds?
- Can we find cancer subtypes based on gene/metabolic expression patterns?
- What's the best way to visualize a high dimensional genomic data?

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?
- Can individuals' SNP profiles be used to learn about their ethnic/racial backgrounds?
- Can we find cancer subtypes based on gene/metabolic expression patterns?
- What's the best way to visualize a high dimensional genomic data?
- How to find "interesting patterns" in the data?

Unsupervised Learning Examples

- Do genes/samples in microarray expression data form interesting groups?
- Can individuals' SNP profiles be used to learn about their ethnic/racial backgrounds?
- Can we find cancer subtypes based on gene/metabolic expression patterns?
- What's the best way to visualize a high dimensional genomic data?
- How to find "interesting patterns" in the data?
- Which genes/protiens/metabolites are "associated" with the disease (this is really not an unsupervised learning question, but somewhat related...)?

Unsupervised Learning Methods

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)
  - PCA: Principal Component Analysis
  - MDS: Multi-Dimensional Scaling
  - Sparse PCA, Kernel PCA, ICA, Manifold Learning, ...

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)
  - PCA: Principal Component Analysis
  - MDS: Multi-Dimensional Scaling
  - Sparse PCA, Kernel PCA, ICA, Manifold Learning, ...
- Cluster Analysis: Find **similar groups** of variables/samples; this can be the final goal of the analysis, or a preliminary step

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)
  - PCA: Principal Component Analysis
  - MDS: Multi-Dimensional Scaling
  - Sparse PCA, Kernel PCA, ICA, Manifold Learning, ...
- Cluster Analysis: Find **similar groups** of variables/samples; this can be the final goal of the analysis, or a preliminary step
  - Hierarchical Clustering: Agglomerative (bottom-up) clustering
  - Partition-based Methods: K-means, Model-based clustering, Spectral clustering
  - Self Organizing Maps, bi-clustering, ...

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)
  - PCA: Principal Component Analysis
  - MDS: Multi-Dimensional Scaling
  - Sparse PCA, Kernel PCA, ICA, Manifold Learning, ...
- Cluster Analysis: Find **similar groups** of variables/samples; this can be the final goal of the analysis, or a preliminary step
  - Hierarchical Clustering: Agglomerative (bottom-up) clustering
  - Partition-based Methods: K-means, Model-based clustering, Spectral clustering
  - Self Organizing Maps, bi-clustering, ...
- Multiple Hypothesis Testing: Find genes/proteins/metabolites that are associated with the response

Unsupervised Learning Methods

- Dimension Reduction: Find **low dimensional representation** of data; this is a very useful tool for **discovering patterns** in the data, and also improving performance of regression and prediction methods (recall PCR)
  - PCA: Principal Component Analysis
  - MDS: Multi-Dimensional Scaling
  - Sparse PCA, Kernel PCA, ICA, Manifold Learning, ...
- Cluster Analysis: Find **similar groups** of variables/samples; this can be the final goal of the analysis, or a preliminary step
  - Hierarchical Clustering: Agglomerative (bottom-up) clustering
  - Partition-based Methods: K-means, Model-based clustering, Spectral clustering
  - Self Organizing Maps, bi-clustering, ...
- Multiple Hypothesis Testing: Find genes/proteins/metabolites that are associated with the response
  - Family wise error rates (Bonferroni correction)
  - False discovery rate control (FDR)

Challenges in Unsupervised Learning

- Unlike supervised learning, there is no direct method for calculating p-values, or performing cross validation

Challenges in Unsupervised Learning

- Unlike supervised learning, there is no direct method for calculating p-values, or performing cross validation
- Comparison and selection of method becomes more difficult

Challenges in Unsupervised Learning

- Unlike supervised learning, there is no direct method for calculating p-values, or performing cross validation
- Comparison and selection of method becomes more difficult
- Difficult to validate the results of analysis

Challenges in Unsupervised Learning

- Unlike supervised learning, there is no direct method for calculating p-values, or performing cross validation
- Comparison and selection of method becomes more difficult
- Difficult to validate the results of analysis
- In high dimensional settings, choices for displaying results of analysis are limited

Challenges in Unsupervised Learning

- Unlike supervised learning, there is no direct method for calculating p-values, or performing cross validation
- Comparison and selection of method becomes more difficult
- Difficult to validate the results of analysis
- In high dimensional settings, choices for displaying results of analysis are limited
- Unsupervised methods are often based on notions of "similarity" or "distance". When $p \gg n$, these become less informative/accurate

Why Dimension Reduction?

When dealing with high dimensional omics data ($p \gg n$)

- Data visualization becomes very difficult (cannot draw 2D scatterplots for large p).

Why Dimension Reduction?

When dealing with high dimensional omics data ($p \gg n$)

- Data visualization becomes very difficult (cannot draw 2D scatterplots for large p).
- Prediction accuracy of traditional statistical models reduces.

Why Dimension Reduction?

When dealing with high dimensional omics data ($p \gg n$)

- Data visualization becomes very difficult (cannot draw 2D scatterplots for large p).
- Prediction accuracy of traditional statistical models reduces.
- High dimensional data often have high degrees of redundancy (correlation among features).

Why Dimension Reduction?

When dealing with high dimensional omics data ($p \gg n$)

- Data visualization becomes very difficult (cannot draw 2D scatterplots for large p).
- Prediction accuracy of traditional statistical models reduces.
- High dimensional data often have high degrees of redundancy (correlation among features).
- Many features may be uninformative for the particular problem under study (noise features).

Why Dimension Reduction?

When dealing with high dimensional omics data ($p \gg n$)

- Data visualization becomes very difficult (cannot draw 2D scatterplots for large p).
- Prediction accuracy of traditional statistical models reduces.
- High dimensional data often have high degrees of redundancy (correlation among features).
- Many features may be uninformative for the particular problem under study (noise features).
- Dimension reduction ideally allows us retain information on most important features of the data, while reducing noise and simplifying visualization & analysis.

What is Dimension Reduction?

- Map the data into a new low-dimensional space, where important characteristics of the data are preserved.

What is Dimension Reduction?

- Map the data into a new low-dimensional space, where important characteristics of the data are preserved.
- The new space often gives a (linear or non-linear) transformation of the original data.

What is Dimension Reduction?

- Map the data into a new low-dimensional space, where important characteristics of the data are preserved.
- The new space often gives a (linear or non-linear) transformation of the original data.
- Visualization and analysis (clustering/prediction/...) is then performed in the new space.

What is Dimension Reduction?

- Map the data into a new low-dimensional space, where important characteristics of the data are preserved.
- The new space often gives a (linear or non-linear) transformation of the original data.
- Visualization and analysis (clustering/prediction/...) is then performed in the new space.
- In some cases, (especially for non-linear transformations) interpretation becomes difficult.

Methods of Dimension Reduction

- Principal Component Analysis (PCA)
- Multi-Dimensional Scaling (MDS)
- Kernel PCA, Sparse PCA, Manifold Learning, ...

Principal Component Analysis (PCA)

Recall: PCR provides improvements for regression models in high dimensional settings.

Principal Component Analysis (PCA)

Recall: PCR provides improvements for regression models in high dimensional settings.

- The idea in PCA is similar to PCR, but PCA is unsupervised!

Principal Component Analysis (PCA)

Recall: PCR provides improvements for regression models in high dimensional settings.

- The idea in PCA is similar to PCR, but PCA is unsupervised!
- There is no response variable y, and the goal is data visualization or pattern discovery.

Principal Component Analysis (PCA)

Recall: PCR provides improvements for regression models in high dimensional settings.

- The idea in PCA is similar to PCR, but PCA is unsupervised!
- There is no response variable y, and the goal is data visualization or pattern discovery.
- In some cases, PCA is also used directly to find subclasses of observations with heterogenous properties (admixture populations, population stratification, ...).

Examples of PCA Applications

PCA is widely used in population genetics and genome-wide association studies: **to correct for stratification**.

nature genetics

Principal components analysis corrects for stratification in genome-wide association studies

Alkes L Price[1,2], Nick J Patterson[2], Robert M Plenge[2,3], Michael E Weinblatt[3], Nancy A Shadick[3] & David Reich[1,2]

Population stratification—allele frequency differences between cases and controls due to systematic ancestry differences—can cause spurious associations in disease studies. We describe a method that enables explicit detection and correction of population stratification on a genome-wide scale. Our method uses principal components analysis to explicitly model ancestry differences between cases and controls. The resulting correction is specific to a candidate marker's variation in frequency across ancestral populations, minimizing spurious associations while maximizing power to detect true associations. Our simple, efficient approach can easily be applied to disease studies with hundreds of thousands of markers.

Examples of PCA Applications

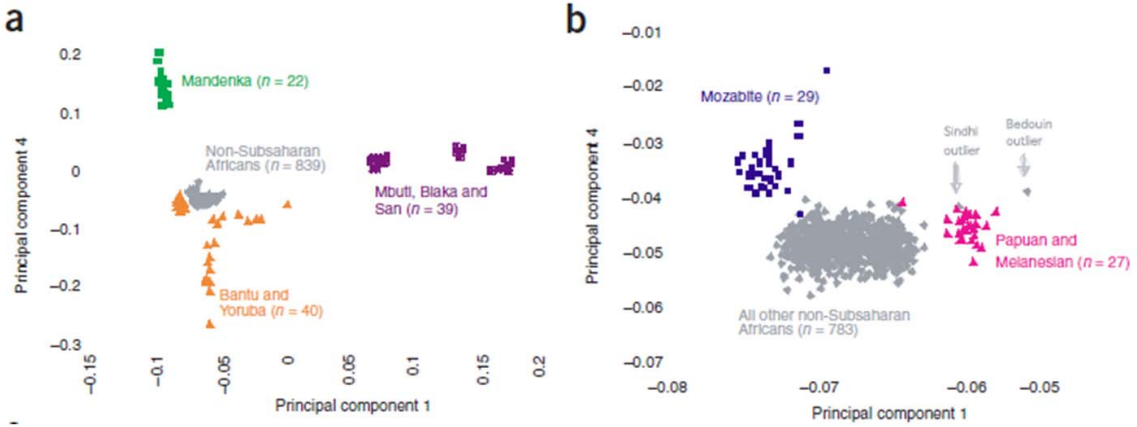
PCA is widely used in population genetics and genome-wide association studies: for the **study of human migration patterns**.

Principal component analysis of genetic data

David Reich, Alkes L Price & Nick Patterson

Principal component analysis (PCA) has been a useful tool for analysis of genetic data, particularly in studies of human migration. A new study finds evidence that the observed geographic gradients, traditionally thought to represent major historical migrations, may in fact have other interpretations.

Principal component analysis (PCA) has been used for several decades to study human population migrations, resulting in remarkable inferences about history. On page 646 of this issue, John Novembre and Matthew Stephens[1] show that the geographic gradients that emerge when PCA is applied to genetic data—and that are sometimes interpreted as highly suggestive of major historical migrations—can also have other explanations. We suggest guidelines for scientists interested in using PCA in genetic



Examples of PCA Applications

PCA is widely used in population genetics and genome-wide association studies: in the **study of admixture populations**.

Principal Component Analysis under Population Genetic Models of Range Expansion and Admixture

Olivier François,*[1] Mathias Currat,[2] Nicolas Ray,[3,4,5] Eunjung Han,[5] Laurent Excoffier,[3,4] and John Novembre[6,7]

[1]Laboratoire Techniques de l'Ingénierie Médicale et de la Complexité, Faculty of Medicine, University Joseph Fourier, Grenoble Institute of Technology, Centre National de la Recherche Scientifique UMR5525, La Tronche, France

[2]Laboratory of Anthropology, Genetics and Peopling history, Department of Anthropology and Ecology, University of Geneva, Geneva, Switzerland

[3]Computational and Molecular Population Genetics Lab, Institute of Ecology and Evolution, University of Berne, Berne, Switzerland

[4]Swiss Institute of Bioinformatics, Lausanne, Switzerland

[5]EnviroSPACE laboratory, Climate Change and Climate Impacts, Institute for Environmental Sciences, University of Geneva, Carouge, Switzerland

[6]Department of Ecology and Evolutionary Biology, University of California

[7]Interdepartmental Program in Bioinformatics, University of California-Los Angeles

Examples of PCA Applications

PCA is widely used in population genetics and genome-wide association studies: to **discover SNP sets** for pathway analysis

Genetic Epidemiology 26: 11–21 (2004)

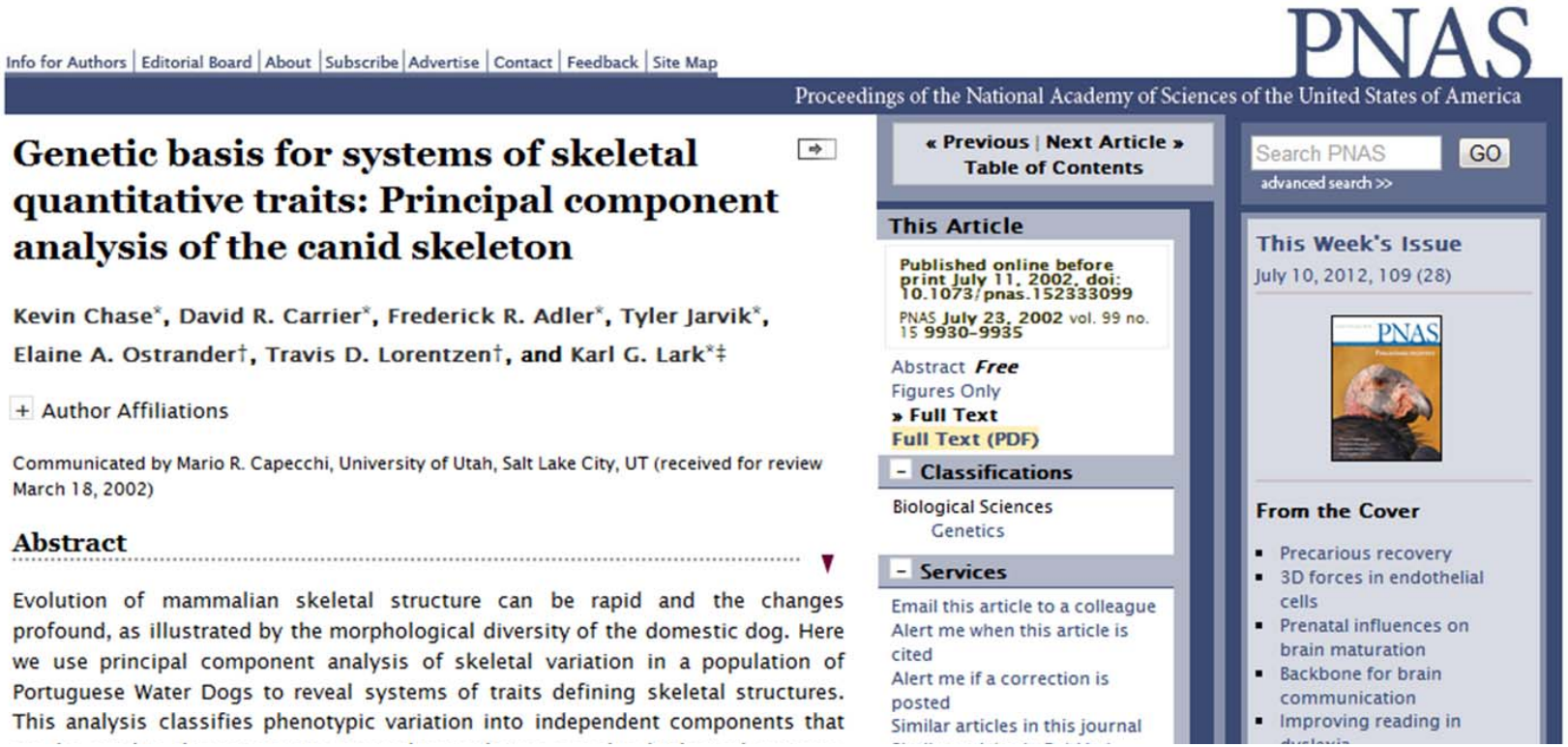
Principal Component Analysis for Selection of Optimal SNP-Sets That Capture Intragenic Genetic Variation

Benjamin D. Horne[1,2] and Nicola J. Camp[1]

[1]Genetic Epidemiology Division, Department of Medical Informatics, University of Utah, Salt Lake City, Utah
[2]Cardiovascular Department, LDS Hospital, Salt Lake City, Utah

Examples of PCA Applications

It is also used in a variety of other applications...



Info for Authors | Editorial Board | About | Subscribe | Advertise | Contact | Feedback | Site Map

PNAS

Proceedings of the National Academy of Sciences of the United States of America

Genetic basis for systems of skeletal quantitative traits: Principal component analysis of the canid skeleton

Kevin Chase*, David R. Carrier*, Frederick R. Adler*, Tyler Jarvik*, Elaine A. Ostrander†, Travis D. Lorentzen†, and Karl G. Lark*‡

+ Author Affiliations

Communicated by Mario R. Capecchi, University of Utah, Salt Lake City, UT (received for review March 18, 2002)

Abstract

Evolution of mammalian skeletal structure can be rapid and the changes profound, as illustrated by the morphological diversity of the domestic dog. Here we use principal component analysis of skeletal variation in a population of Portuguese Water Dogs to reveal systems of traits defining skeletal structures. This analysis classifies phenotypic variation into independent components that

« Previous | Next Article »
Table of Contents

This Article

Published online before print July 11, 2002, doi: 10.1073/pnas.152333099
PNAS July 23, 2002 vol. 99 no. 15 9930–9935

Abstract *Free*
Figures Only
» Full Text
Full Text (PDF)

Classifications

Biological Sciences
Genetics

Services

Email this article to a colleague
Alert me when this article is cited
Alert me if a correction is posted
Similar articles in this journal

Search PNAS GO
advanced search >>

This Week's Issue
July 10, 2012, 109 (28)

From the Cover

- Precarious recovery
- 3D forces in endothelial cells
- Prenatal influences on brain maturation
- Backbone for brain communication
- Improving reading in

PCA: An Overview

- Data: n observations living in a p-dimensional space.

PCA: An Overview

- Data: n observations living in a p-dimensional space.
- Not all p dimensions are equally useful, especially when $p \gg n$.

PCA: An Overview

- Data: n observations living in a p-dimensional space.
- Not all p dimensions are equally useful, especially when $p \gg n$.
- Many are either completely redundant (correlated features) or uninformative (noise features).

PCA: An Overview

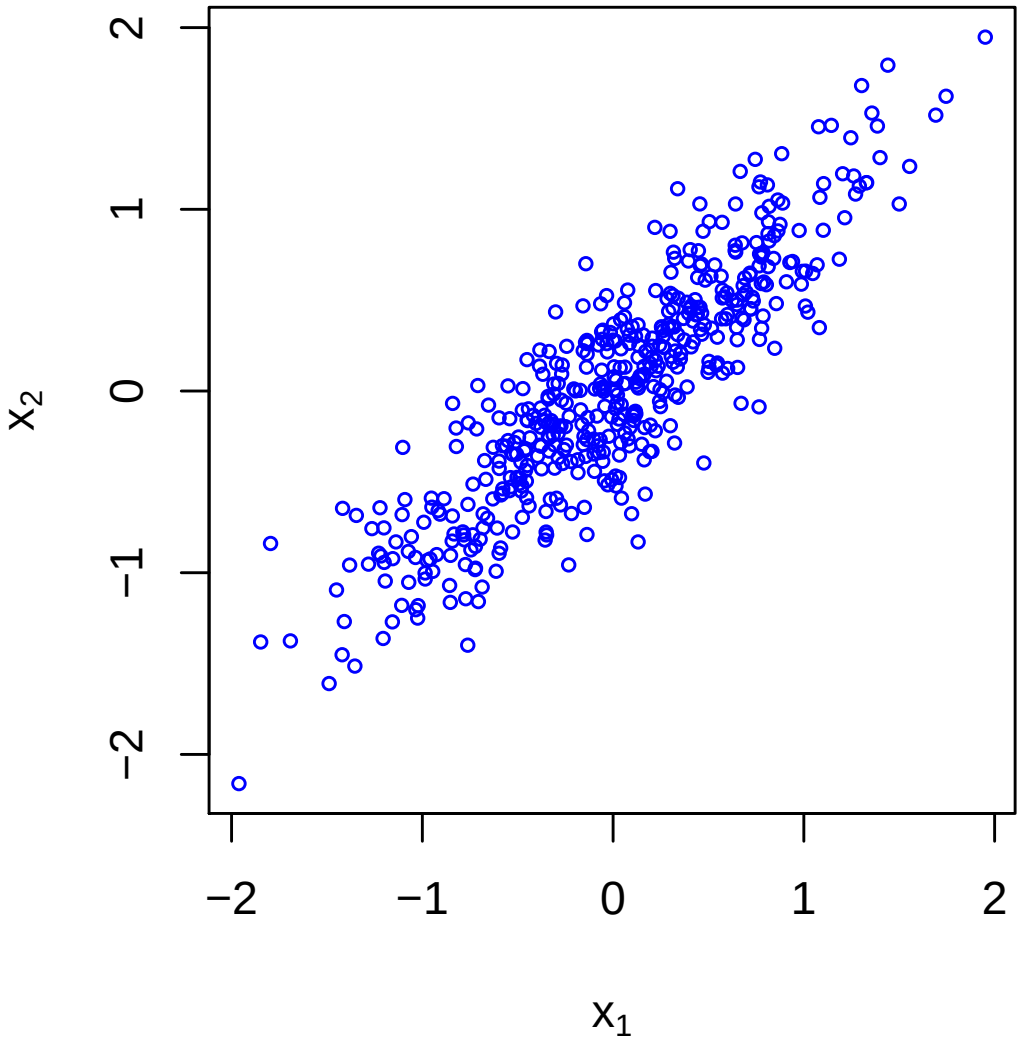
- Data: n observations living in a p-dimensional space.
- Not all p dimensions are equally useful, especially when $p \gg n$.
- Many are either completely redundant (correlated features) or uninformative (noise features).
- Need low-dimensional representation of the variables that captures most of the "information" in the data.

PCA: An Overview

- Data: n observations living in a p-dimensional space.
- Not all p dimensions are equally useful, especially when $p \gg n$.
- Many are either completely redundant (correlated features) or uninformative (noise features).
- Need low-dimensional representation of the variables that captures most of the "information" in the data.
- To maximize the information retained, we need to minimize the redundancy, and to do this, we look for low-dimensional representations that capture most of the variation in the data.

PCA: The Main Idea

Question: What is a good 1-dim representation of the data?



PCA: The Main Idea

Some Possibilities:

PCA: The Main Idea

Some Possibilities:

- Use one of the variables (e.g. X_1).

PCA: The Main Idea

Some Possibilities:

- Use one of the variables (e.g. X_1).
- Better idea: use a linear combination of the variables; i.e. a weighted average of the variables.

$$Z_1 = w_1 X_1 + w_2 X_2 = Xw$$

PCA: The Main Idea

Some Possibilities:

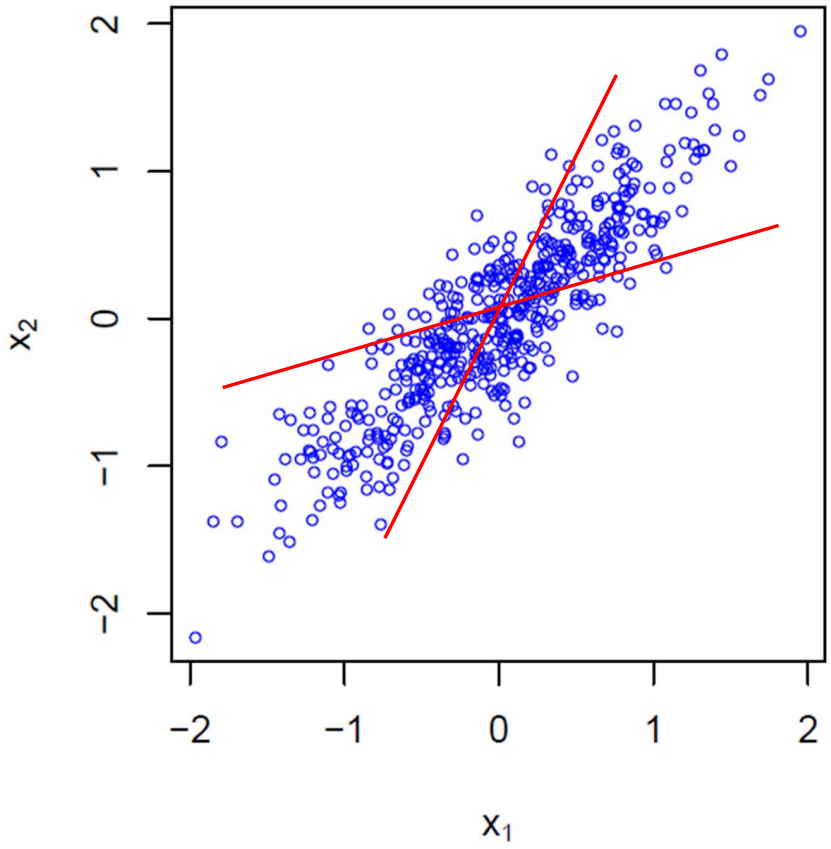
- Use one of the variables (e.g. X_1).
- Better idea: use a linear combination of the variables; i.e. a weighted average of the variables.

$$Z_1 = w_1 X_1 + w_2 X_2 = Xw$$

Question: what is a good choice for the weights w_i?

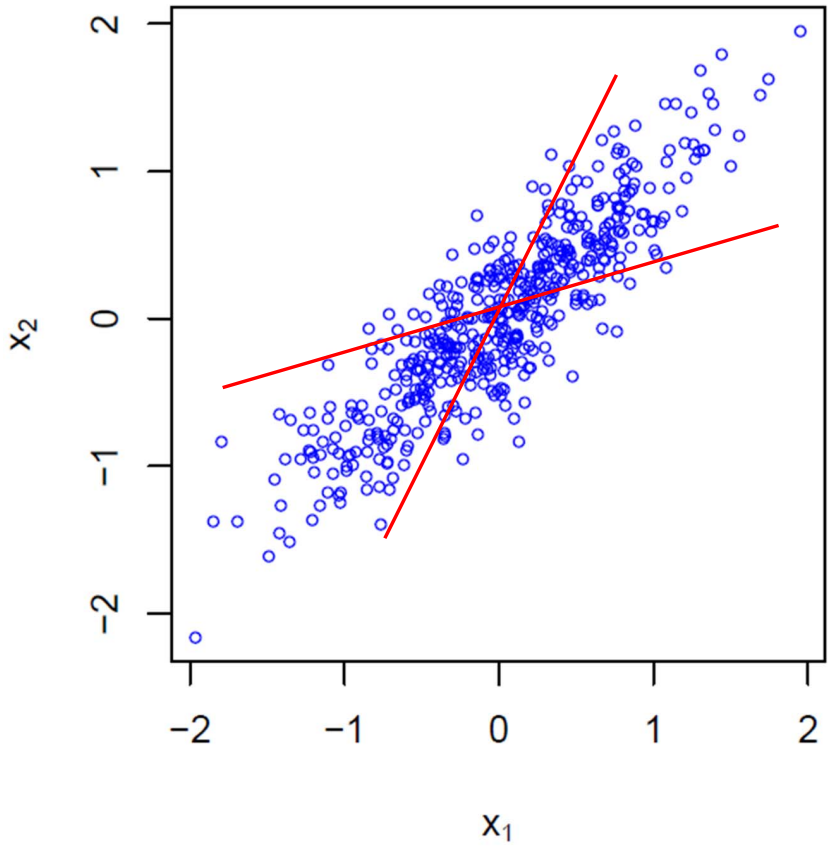
PCA: The Main Idea

Many possibilities, but which one is a good choice?



PCA: The Main Idea

Many possibilities, but which one is a good choice?



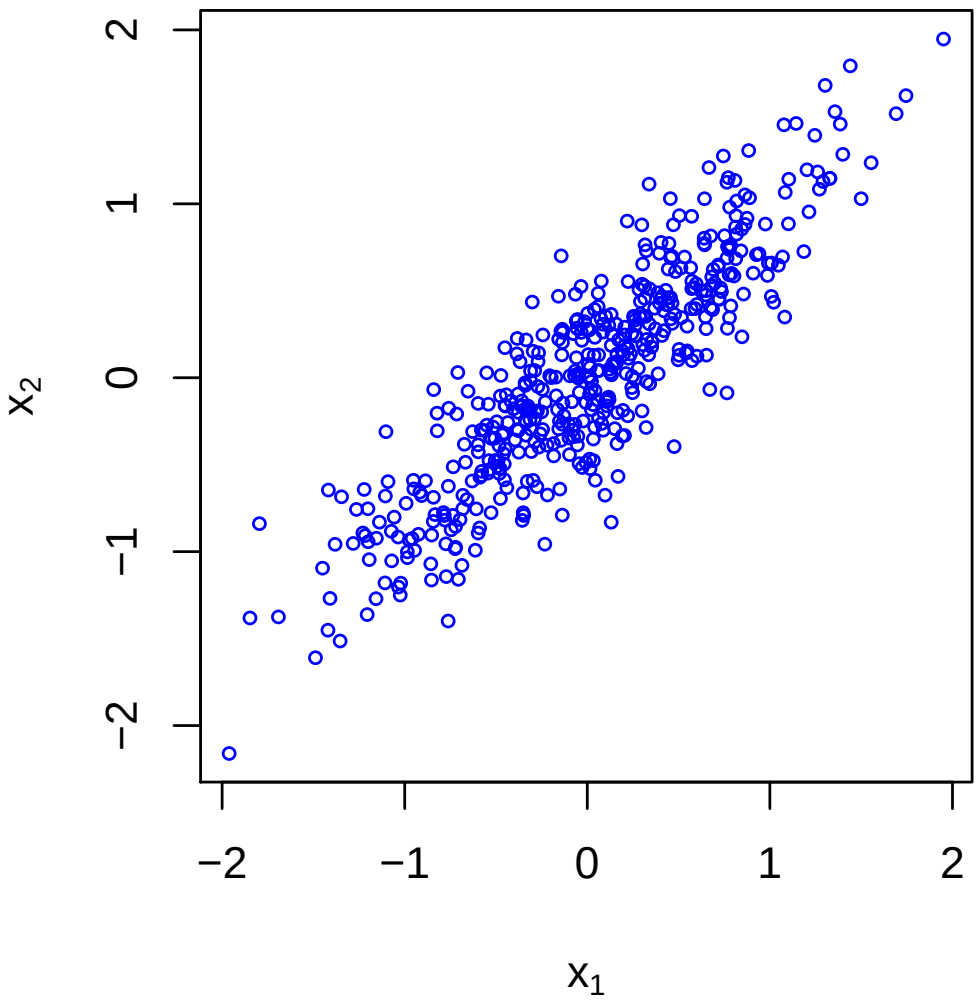
Need some criterion for a principled choice of the weights.

The Criterion for Principal Components

- In PCA, we try to find the direction with maximum variance.

The Criterion for Principal Components

- In PCA, we try to find the direction with maximum variance.



The Criterion for Principal Components

- In PCA, we try to find the direction with maximum variance.

The Criterion for Principal Components

- In PCA, we try to find the direction with maximum variance.
- *Formally*, we find the vector of weights w using the following **criterion**:

$$\begin{aligned} \underset{w}{\text{maximize}} \quad & \mathrm{Var}(Xw) \\ \underset{w}{\text{maximize}} \quad & w^{\mathsf{T}}\mathrm{Var}(X)w \\ \underset{w}{\text{maximize}} \quad & w^{\mathsf{T}}\Sigma w \end{aligned}$$

where $\Sigma = \mathrm{Cov}(X)$ is the covariance matrix of the X.

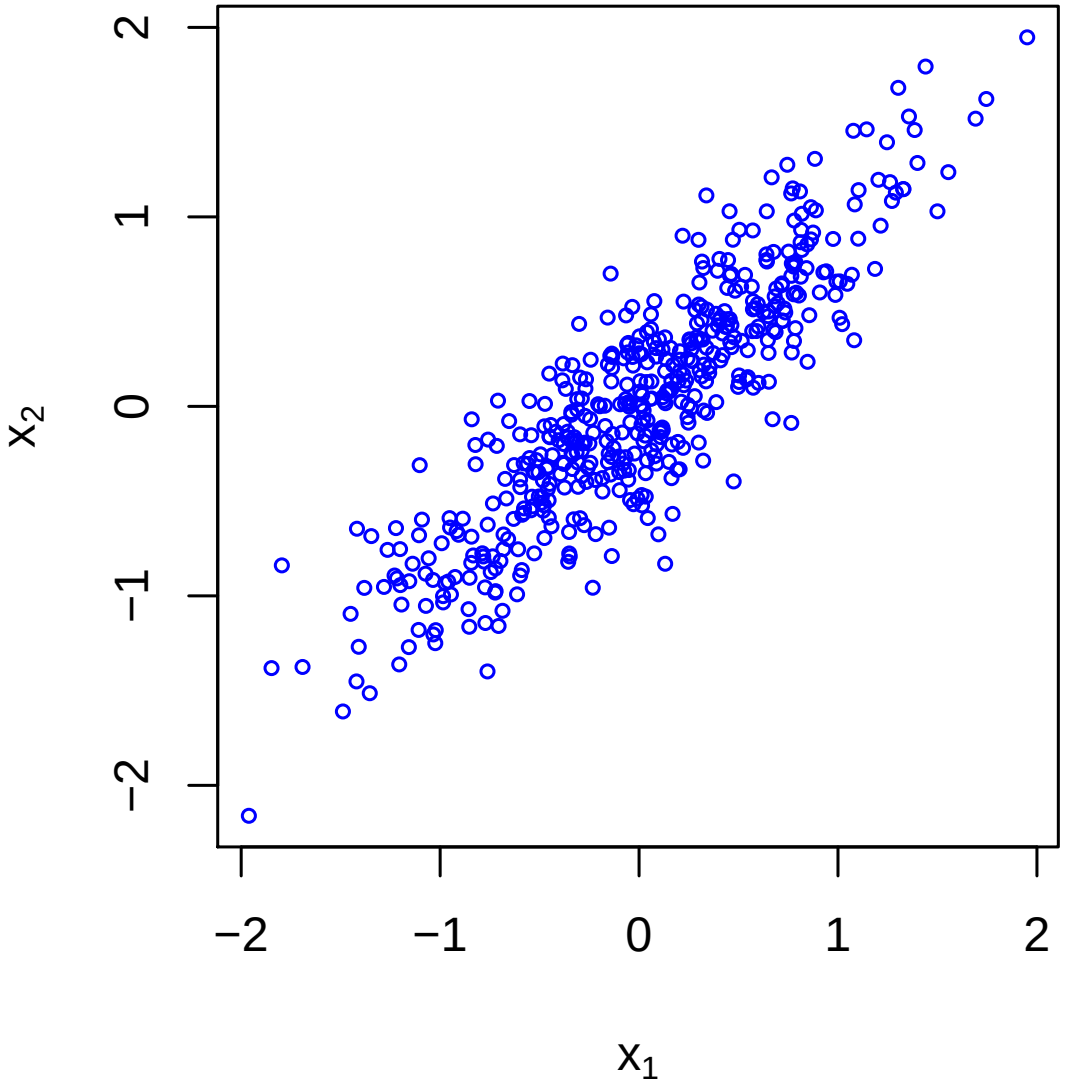
The Criterion for Principal Components

- In PCA, we try to find the direction with maximum variance.
- *Formally*, we find the vector of weights w using the following **criterion**:

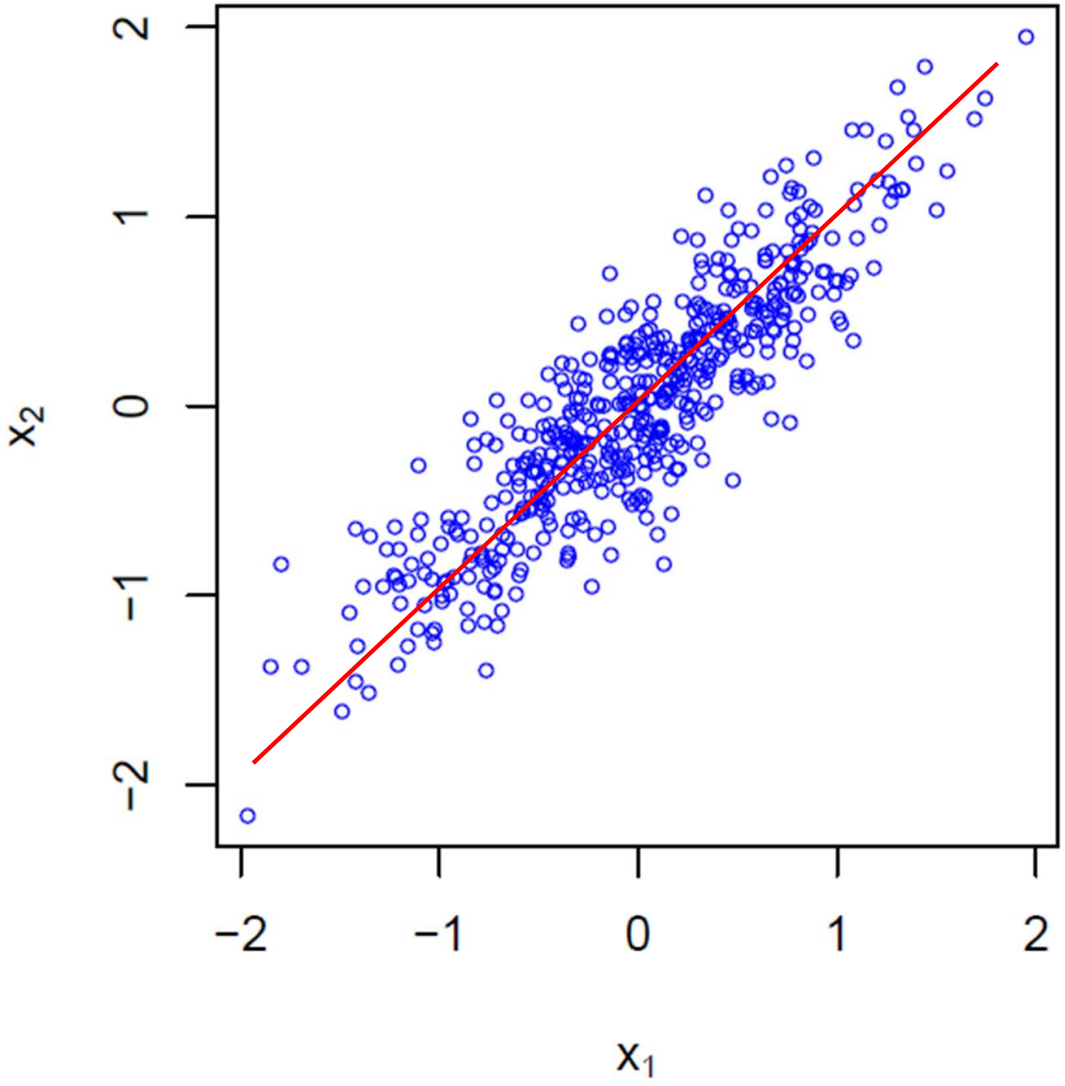
$$\begin{aligned} \underset{w}{\text{maximize}} \quad & \mathrm{Var}(Xw) \\ \underset{w}{\text{maximize}} \quad & w^{\mathsf{T}}\mathrm{Var}(X)w \\ \underset{w}{\text{maximize}} \quad & w^{\mathsf{T}}\Sigma w \end{aligned}$$

  where $\Sigma = \mathrm{Cov}(X)$ is the covariance matrix of the X.
- I.e., the interesting direction according to the PCA criterion is the one that captures the majority of the variance in the data.

The 1-Dimensional PCA Solution



The 1-Dimensional PCA Solution



- But, what if we need another direction:

$$Z_2 = v_1 X_1 + v_2 X_2 = Xv$$

- But, what if we need another direction:

$$Z_2 = v_1 X_1 + v_2 X_2 = Xv$$

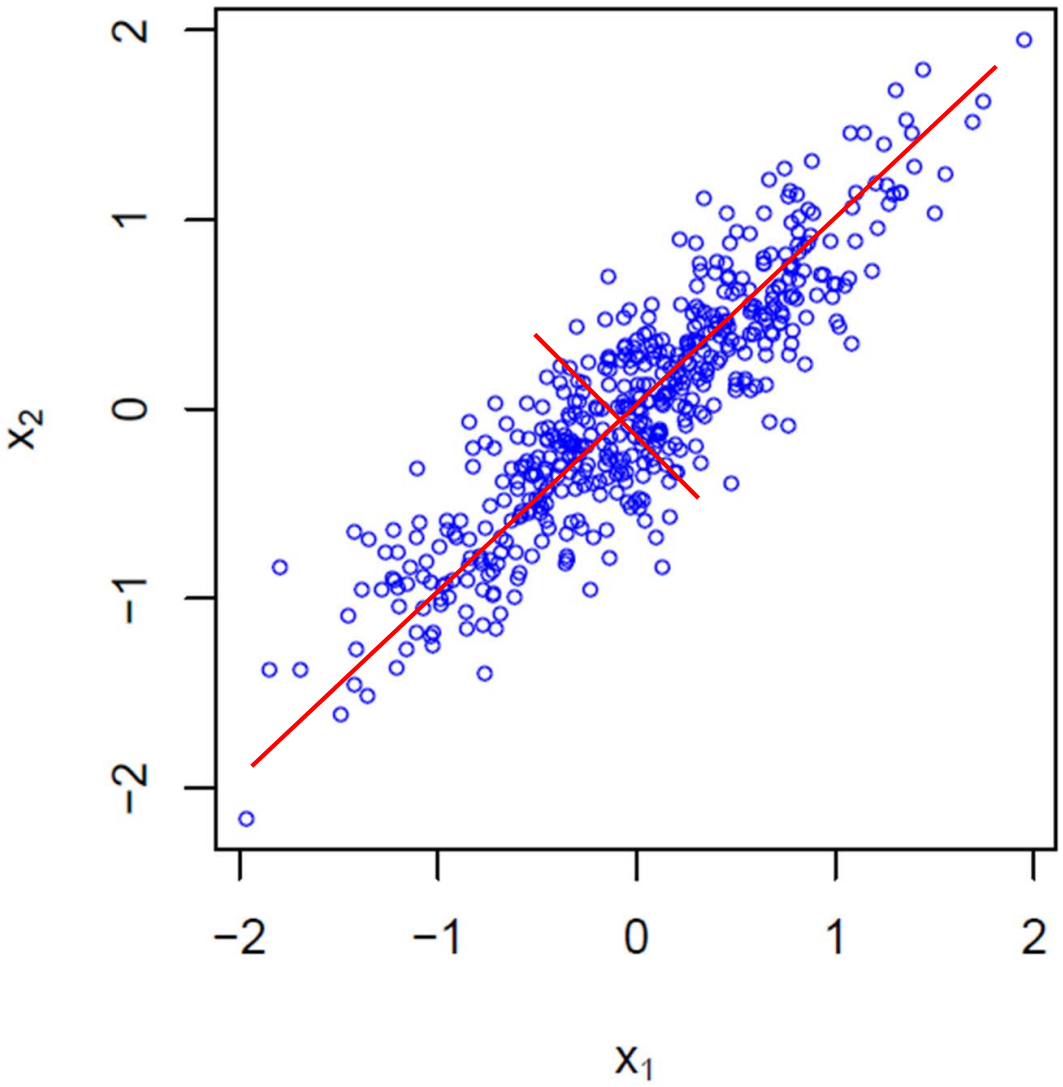
- A systematic way to find additional principal components (PC's), is to choose subsequent linear combinations orthogonal/perpendicular to previous ones.
- This means that we want to choose v to be orthogonal to w, but to explain the majority of variability in the data.

- But, what if we need another direction:

$$Z_2 = v_1 X_1 + v_2 X_2 = Xv$$

- A systematic way to find additional principal components (PC's), is to choose subsequent linear combinations orthogonal/perpendicular to previous ones.
- This means that we want to choose v to be orthogonal to w, but to explain the majority of variability in the data.
- In the case of 2-dimensional data, there is only one choice!! This is always the case for the last PC.
- For $p > 2$, there are many orthogonal vectors to choose from, and we need to find the one that explains the maximum variation in the data, and is orthogonal to the first one

The Full PCA Solution for 2 Dimensions



PCA: The General Problem

- Let $Z_1, Z_2, \ldots, Z_M$ represent $M \leq p$ linear combinations of the p predictors:

$$Z_m = \sum_{j=1}^{p} w_{mj} X_j.$$

PCA: The General Problem

- Let $Z_1, Z_2, \ldots, Z_M$ represent $M \leq p$ linear combinations of the p predictors:

$$Z_m = \sum_{j=1}^{p} w_{mj} X_j.$$

- $Z_1, \ldots, Z_M$ are chosen to be the principal components of the data:

$$w_m = \max_{w:\|w\|=1} \mathrm{Var}(Xw) \text{ and } w_m \perp \{w_1, \ldots w_{m-1}\}$$

PCA: The General Problem

- Let $Z_1, Z_2, \ldots, Z_M$ represent $M \leq p$ linear combinations of the p predictors:

$$Z_m = \sum_{j=1}^{p} w_{mj} X_j.$$

- $Z_1, \ldots, Z_M$ are chosen to be the principal components of the data:

$$w_m = \max_{w:\|w\|=1} \mathrm{Var}(Xw) \text{ and } w_m \perp \{w_1, \ldots w_{m-1}\}$$

- w_m's are called factor loadings or PC loadings

PCA: The General Problem

- Let $Z_1, Z_2, \ldots, Z_M$ represent $M \leq p$ linear combinations of the p predictors:

$$Z_m = \sum_{j=1}^{p} w_{mj} X_j.$$

- $Z_1, \ldots, Z_M$ are chosen to be the principal components of the data:

$$w_m = \max_{w:\|w\|=1} \mathrm{Var}(Xw) \text{ and } w_m \perp \{w_1, \ldots w_{m-1}\}$$

- w_m's are called factor loadings or PC loadings
- Z_m's are called principal components (PCs) or PC scores

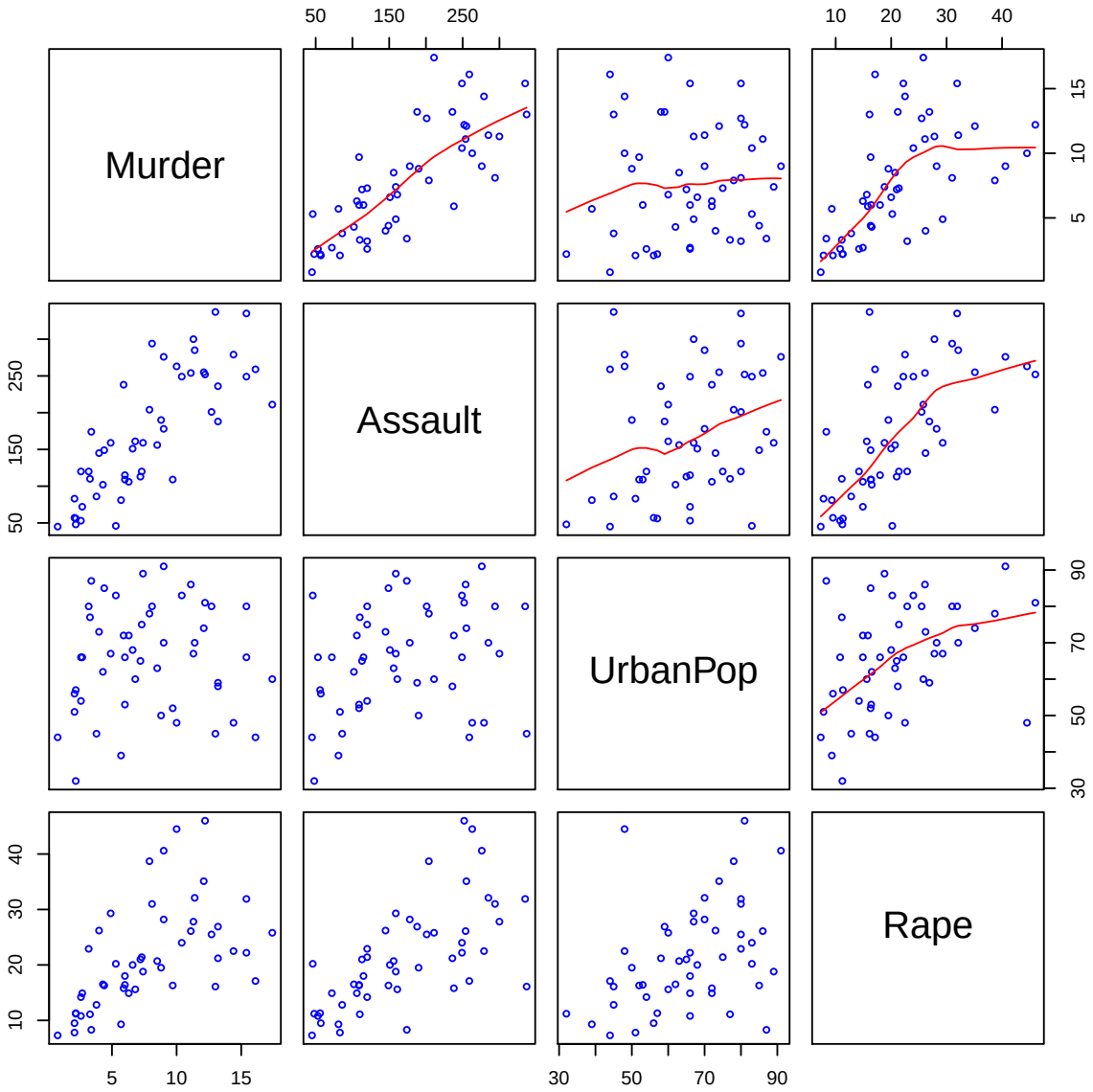
Example: PC Analysis of `USArrests` data

Data Description:

```
This data set contains statistics, in arrests per 100,000 residents
for "assault", "murder", and "rape" in each of the 50 US states in
1973. Also given is the "percent of the population living in urban areas".

A data frame with 50 observations on 4 variables:
[,1]  Murder     numeric    Murder arrests (per 100,000)
[,2]  Assault    numeric    Assault arrests (per 100,000)
[,3]  UrbanPop   numeric    Percent urban population
[,4]  Rape       numeric    Rape arrests (per 100,000)
```

Take a Look At the Data



Example: PC Analysis of `USArrests` data

What are the means for each of the variables?

Variable	Murder	Assault	UrbanPop	Rape
Mean	7.788	170.760	65.540	21.232

Example: PC Analysis of `USArrests` data

What are the means for each of the variables?

Variable	Murder	Assault	UrbanPop	Rape
Mean	7.788	170.760	65.540	21.232

What are the variances for each of the variables?

Variable	Murder	Assault	UrbanPop	Rape
Variances	18.97047	6945.16571	209.51878	87.72916

Example: PC Analysis of `USArrests` data

What are the means for each of the variables?

Variable	Murder	Assault	UrbanPop	Rape
Mean	7.788	170.760	65.540	21.232

What are the variances for each of the variables?

Variable	Murder	Assault	UrbanPop	Rape
Variances	18.97047	6945.16571	209.51878	87.72916

Vastly different means and variances, what happens if we fit PCA to this data?

Some Remarks

- In PCA, it is assumed that the variables are centered. So remember to always center the variables, before performing PCA.

Some Remarks

- In PCA, it is assumed that the variables are centered. So remember to always center the variables, before performing PCA.
- PCA works with both standardized (scaled) or unscaled data; however, the solutions may be different!

Some Remarks

- In PCA, it is assumed that the variables are centered. So remember to always center the variables, before performing PCA.
- PCA works with both standardized (scaled) or unscaled data; however, the solutions may be different!
- The PCA solutions are sensitive to scale of variables, and variables with larger variance will affect the results more.

Some Remarks

- In PCA, it is assumed that the variables are centered. So remember to always center the variables, before performing PCA.
- PCA works with both standardized (scaled) or unscaled data; however, the solutions may be different!
- The PCA solutions are sensitive to scale of variables, and variables with larger variance will affect the results more.
- This may not necessarily desirable in many applications, therefore, it is better to also standardize the variables.

Example: PC Analysis of `USArrests` data

```
> states <- row.names(USArrests)
> states[1:5]
[1] "Alabama"    "Alaska"     "Arizona"    "Arkansas"   "California"

> apply(USArrests, 2, mean)
  Murder  Assault UrbanPop     Rape
   7.788  170.760   65.540   21.232

> apply(USArrests, 2, var)
    Murder    Assault   UrbanPop       Rape
  18.97047 6945.16571  209.51878   87.72916

> pc.out <- prcomp(USArrests, scale=TRUE)
> print(pc.out$rot)
               PC1        PC2        PC3         PC4
Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
Rape     -0.5434321 -0.1673186  0.8177779  0.08902432
```

PC Analysis in R

- By default, `prcomp` centers the variables.

PC Analysis in R

- By default, `prcomp` centers the variables.
- The option `scale=TRUE` standardizes the variables to have standard deviation 1.

PC Analysis in R

- By default, `prcomp` centers the variables.
- The option `scale=TRUE` standardizes the variables to have standard deviation 1.
- The `rot` is shorthand for **rotation**, which reflects the fact that PC's are linear combinations of the original variables.

PC Analysis in R

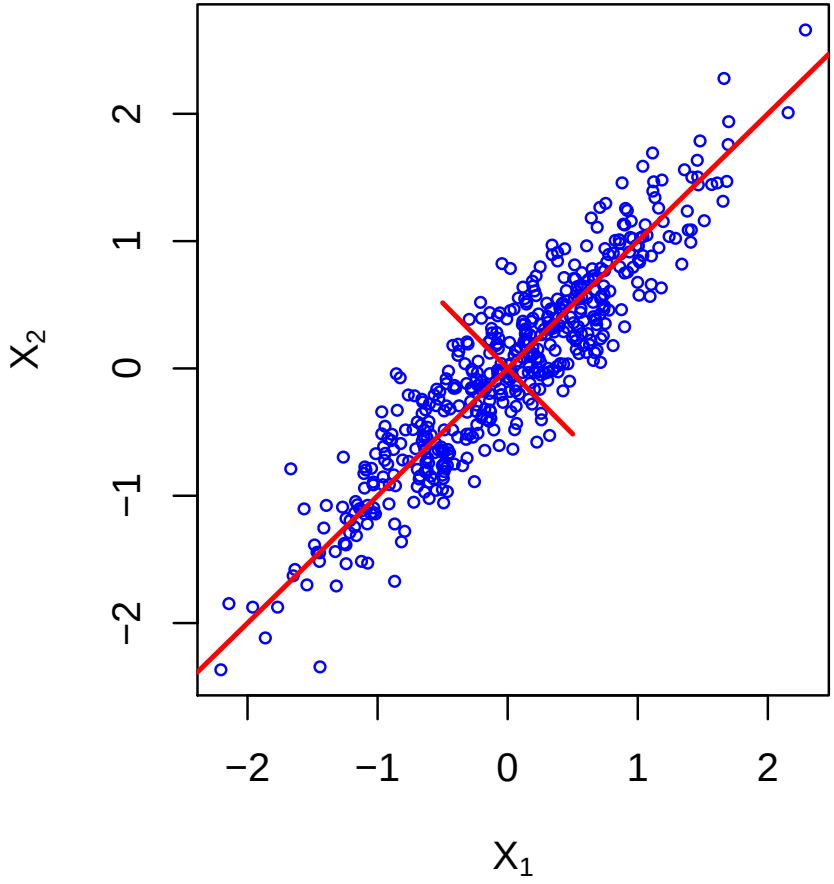
- By default, `prcomp` centers the variables.
- The option `scale=TRUE` standardizes the variables to have standard deviation 1.
- The `rot` is shorthand for **rotation**, which reflects the fact that PC's are linear combinations of the original variables.
- Each PC (column) in the above table gives the weights for the linear combination for calculating the mth PC. So the columns of the above table gives the *PC loadings*.

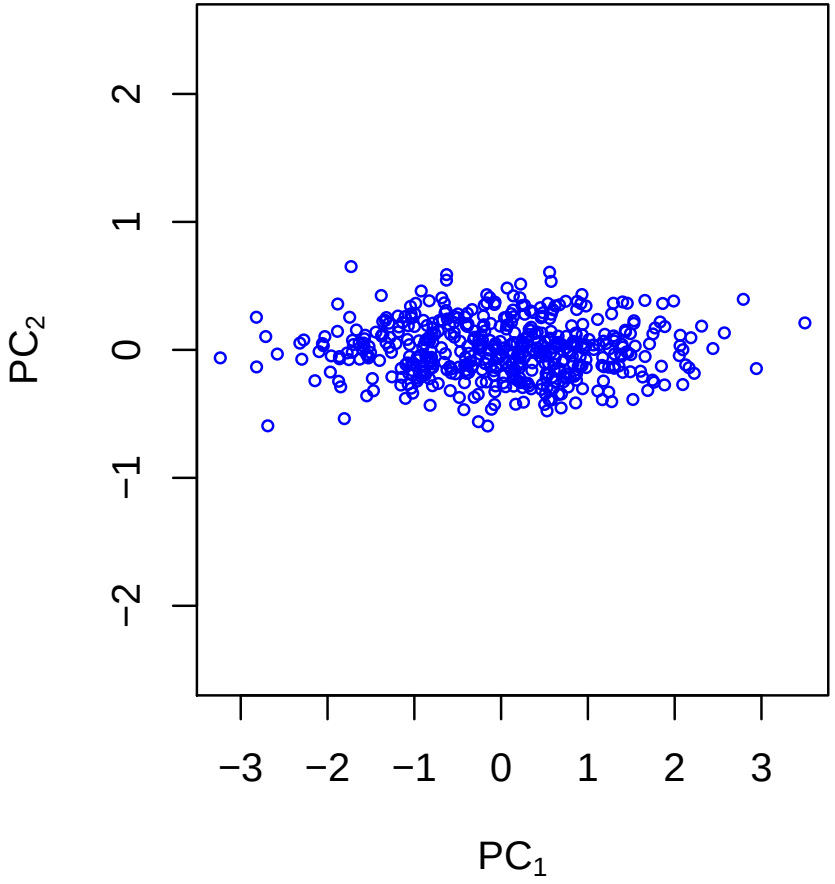
PC Loadings

```
> pc.out <- prcomp(USArrests, scale=TRUE)
> print(pc.out$rot)
               PC1        PC2        PC3         PC4
Murder   -0.5358995  0.4181809 -0.3412327  0.64922780
Assault  -0.5831836  0.1879856 -0.2681484 -0.74340748
UrbanPop -0.2781909 -0.8728062 -0.3780158  0.13387773
Rape     -0.5434321 -0.1673186  0.8177779  0.08902432
```

PC Loadings

- Observations can be plotted ("projected") in the space of PC's.
- Roughly, this is equivalent to rotating the axes and plotting the original data points in the new axes Z_1 and Z_2.





PC Loadings

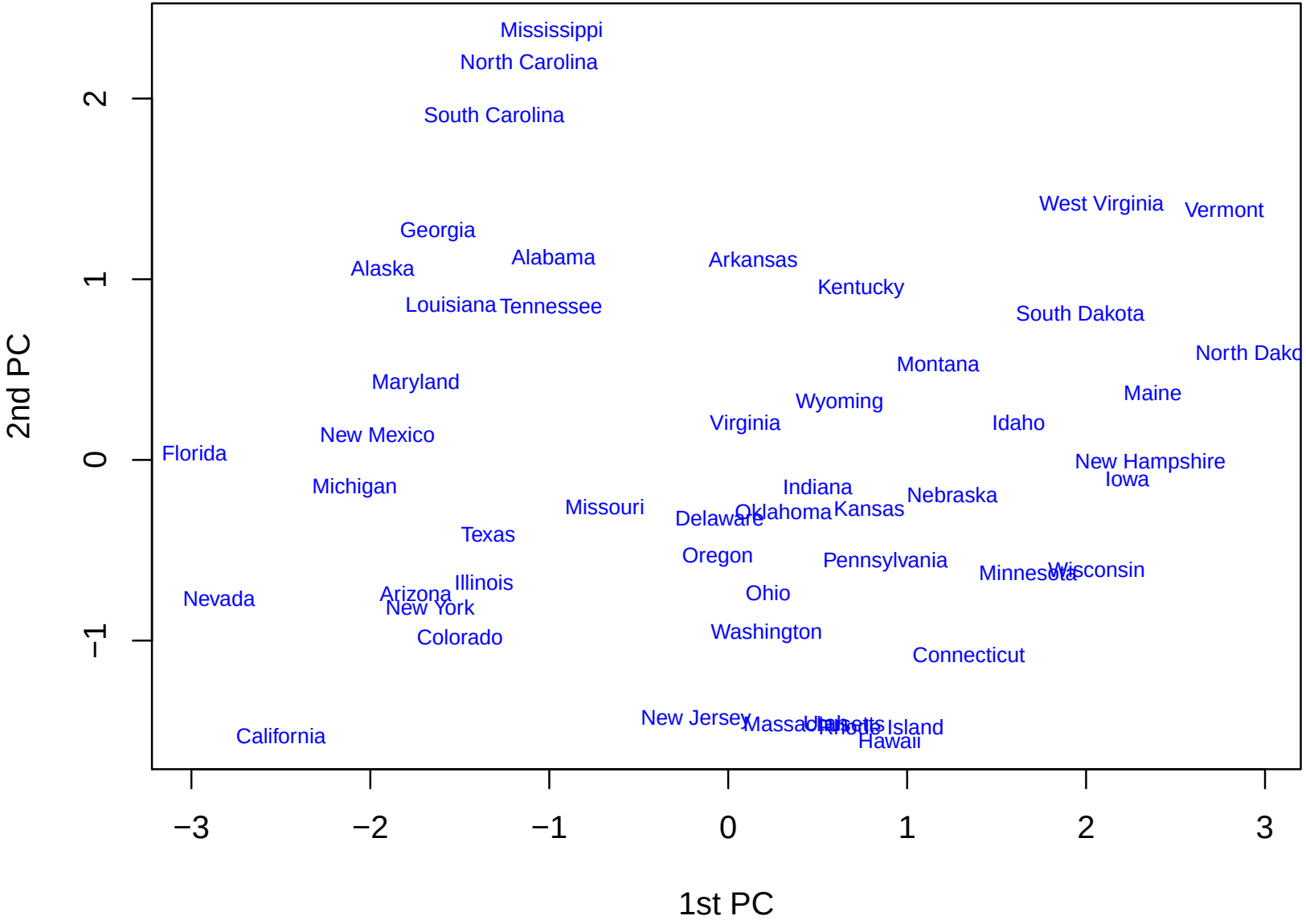
- Observations can be plotted ("projected") in the space of PC's.
- Roughly, this is equivalent to rotating the axes and plotting the original data points in the new axes Z_1 and Z_2.
- We can get these by setting `retx=TRUE` in the `prcomp` call:

```
pc.out <- prcomp(USArrests, scale=TRUE, retx=TRUE)
```

- `pc.out$x` is a matrix of dimension 50×4, which has as its columns the PC score vectors
- Plotting the observations in the space of PC scores can reveal interesting relationships between them.

```
plot(pc.out$x[,1], pc.out$x[,2], type="n", xlab="1st PC", ylab="2nd P
text(pc.out$x[,1], pc.out$x[,2], labels=states)
```

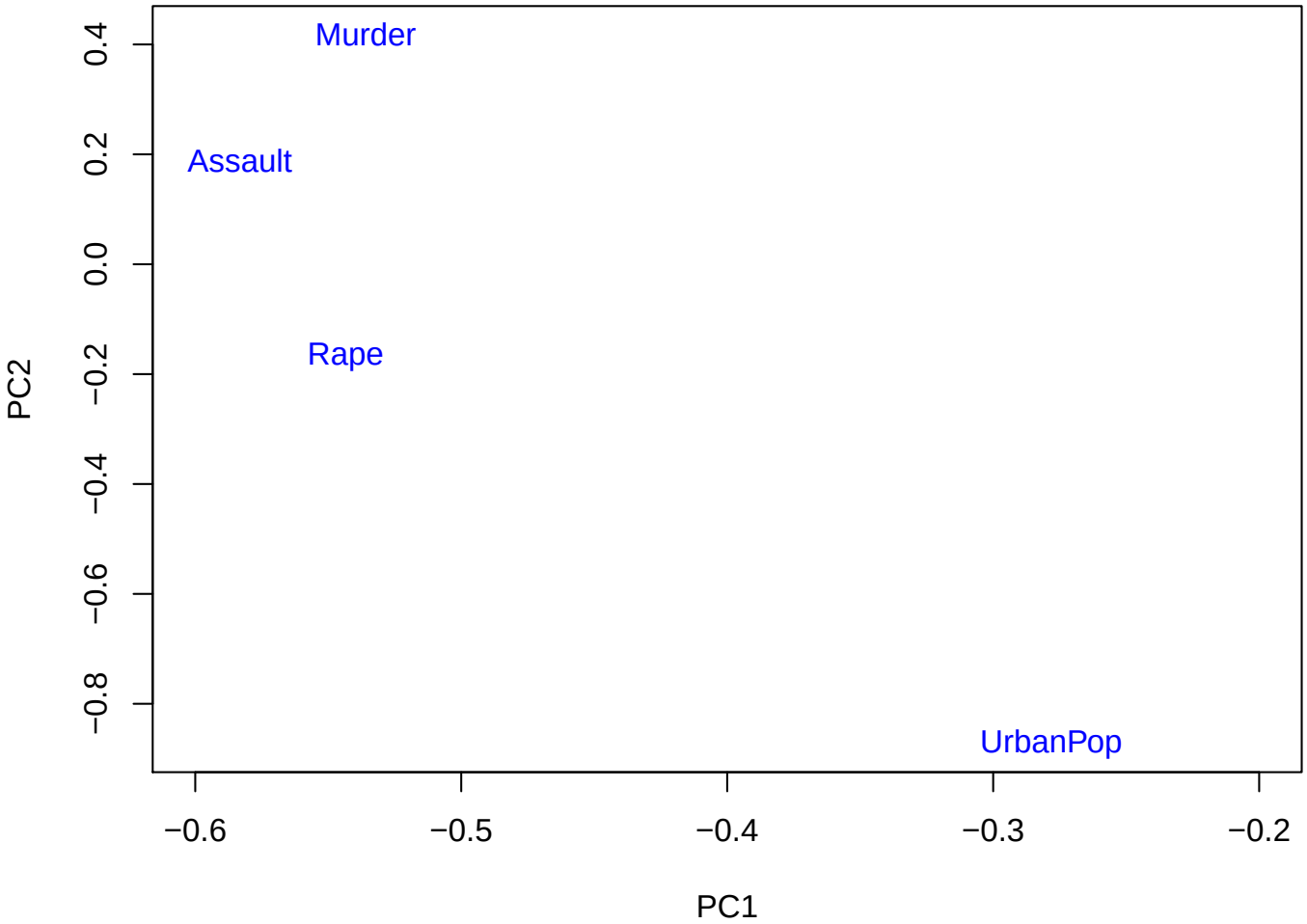
Plotting **Observations** in the Space of PC's



Plotting **Variables** in the Space of PC's

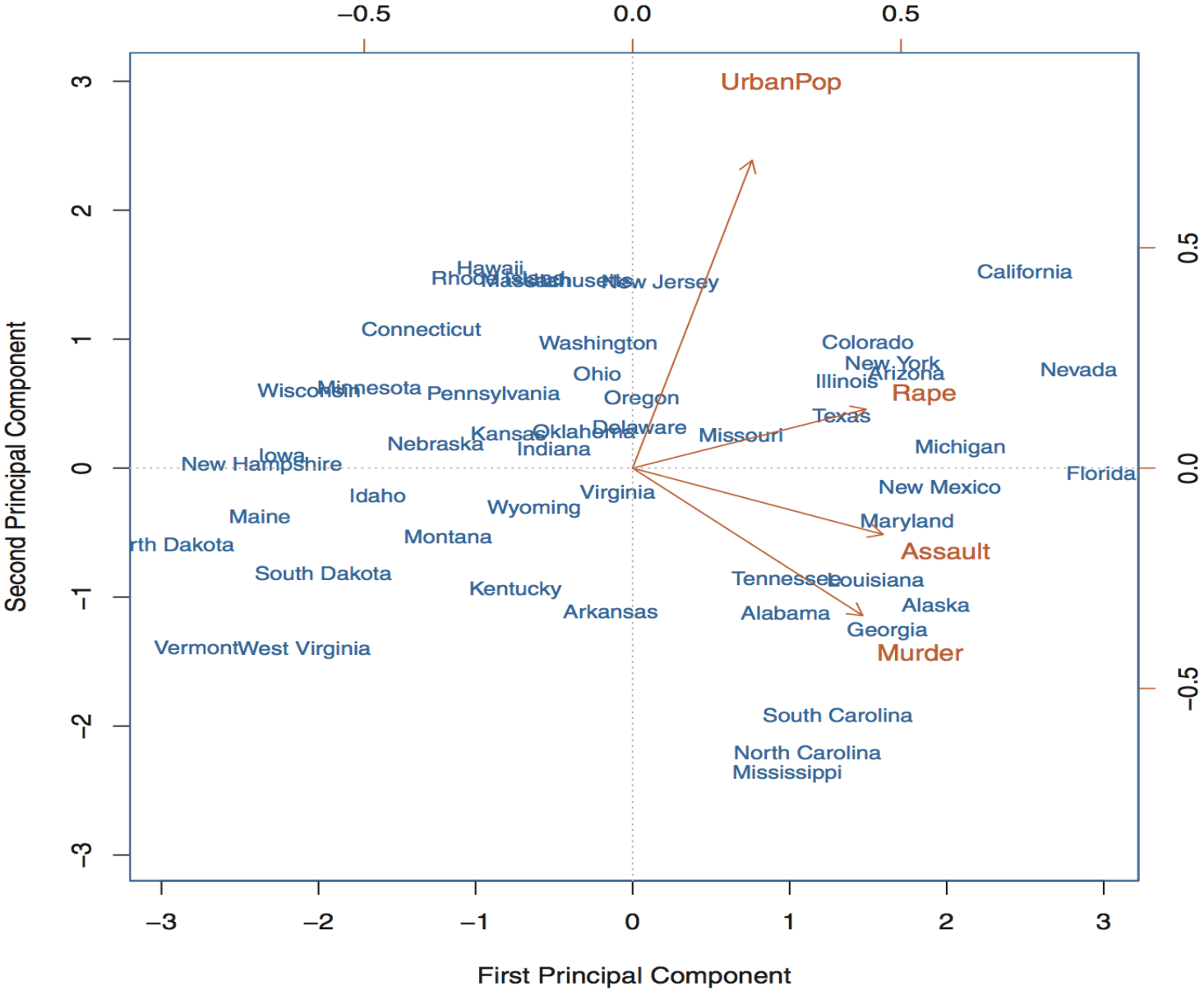
We can also plot the variables in the space of PC loadings:

```
plot(pc.out$rot, type="n")
text(pc.out$rot, names(USArrests), col=4)
```



Plotting **Both** Variables and Observations

We can do this using a `biplot()`



Recap

- PCA results in two main objects:

Recap

- PCA results in two main objects:
  1. PC loadings: these are the weights w_ms for the linear combinations of the original variables

Recap

- PCA results in two main objects:
  1. PC loadings: these are the weights w_ms for the linear combinations of the original variables
  2. PC scores (or PCs): these are the projected observations: $Z_m = \sum_{j=1}^{p} w_{mj} X_j$

Recap

- PCA results in two main objects:
  1. PC loadings: these are the weights w_ms for the linear combinations of the original variables
  2. PC scores (or PCs): these are the projected observations: $Z_m = \sum_{j=1}^{p} w_{mj} X_j$
- can plot observations in space of PC scores; these are found from `pc.out$x`

Recap

- PCA results in two main objects:
  1. PC loadings: these are the weights w_ms for the linear combinations of the original variables
  2. PC scores (or PCs): these are the projected observations: $Z_m = \sum_{j=1}^{p} w_{mj} X_j$
- can plot observations in space of PC scores; these are found from `pc.out$x`
- can plot variables in the space of PC loadings; these are found from `pc.out$rot`

Next Lecture

- PCA, continued
- MDS