Introduction to Social Media Analytics (Lec 5)

Hao PENG
Department of Data Science
City University of Hong Kong
https://haoopeng.github.io/

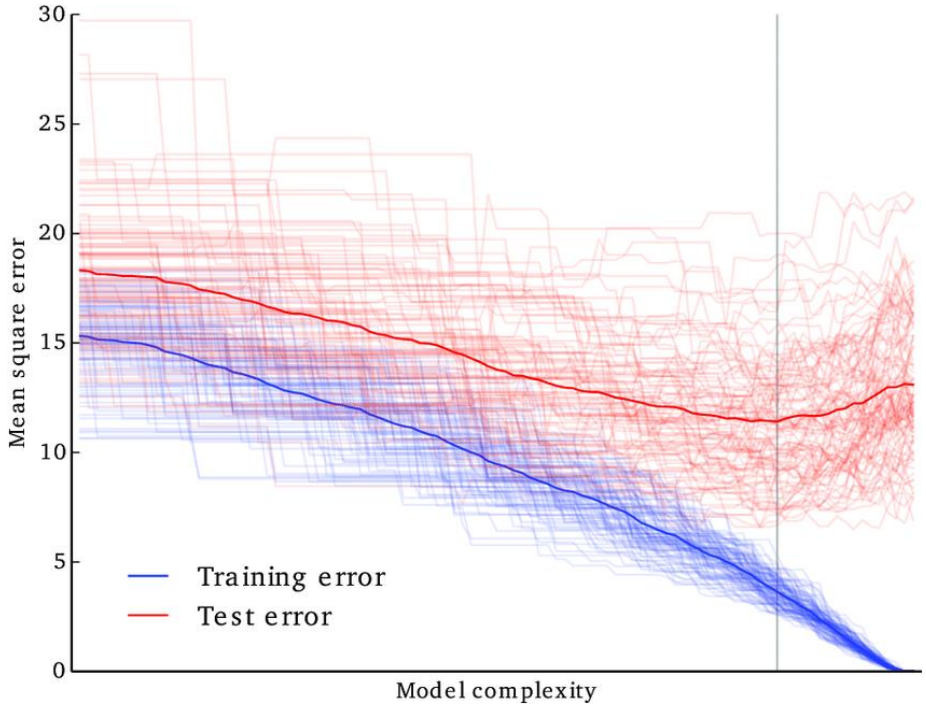
Two weeks ago …

- Supervised learning: linear regression
- Evaluate model performance with RMSE, R-squared
- Supervised classification: logit, naïve bayes, KNN
- True/False positive rate, Precision and Recall, F1-score

How to fine-tune and select the best model?

Model evaluation on test data

- Evaluating models on the test data (unseen by the model) is the best choice to compare model performance.
- Applicable to both regression and classification tasks.



What if there is no test data?
What if the training data is small?

The validation set approach

- Randomly divide the available set of observations into two equal parts, a training set and a validation set or hold-out set. Fit a model on the training set, and the fitted model is used to predict the responses for the observations in the validation set.

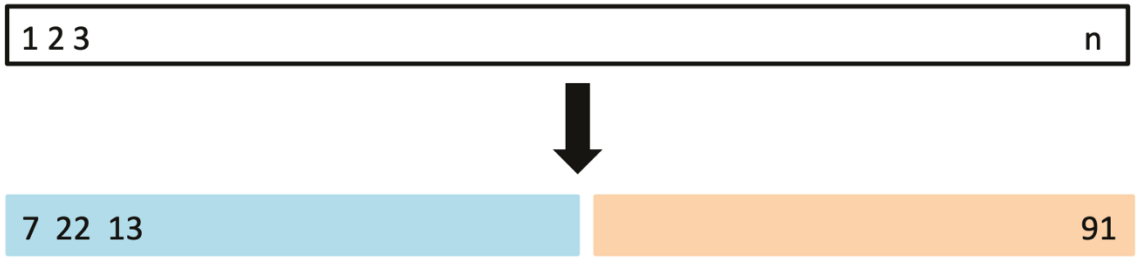


FIGURE 5.1. *A schematic display of the validation set approach. A set of n observations are randomly split into a training set (shown in blue, containing observations 7, 22, and 13, among others) and a validation set (shown in beige, and containing observation 91, among others). The statistical learning method is fit on the training set, and its performance is evaluated on the validation set.*

James, Gareth, et al. *An introduction to statistical learning: with applications in R*.

The validation set approach

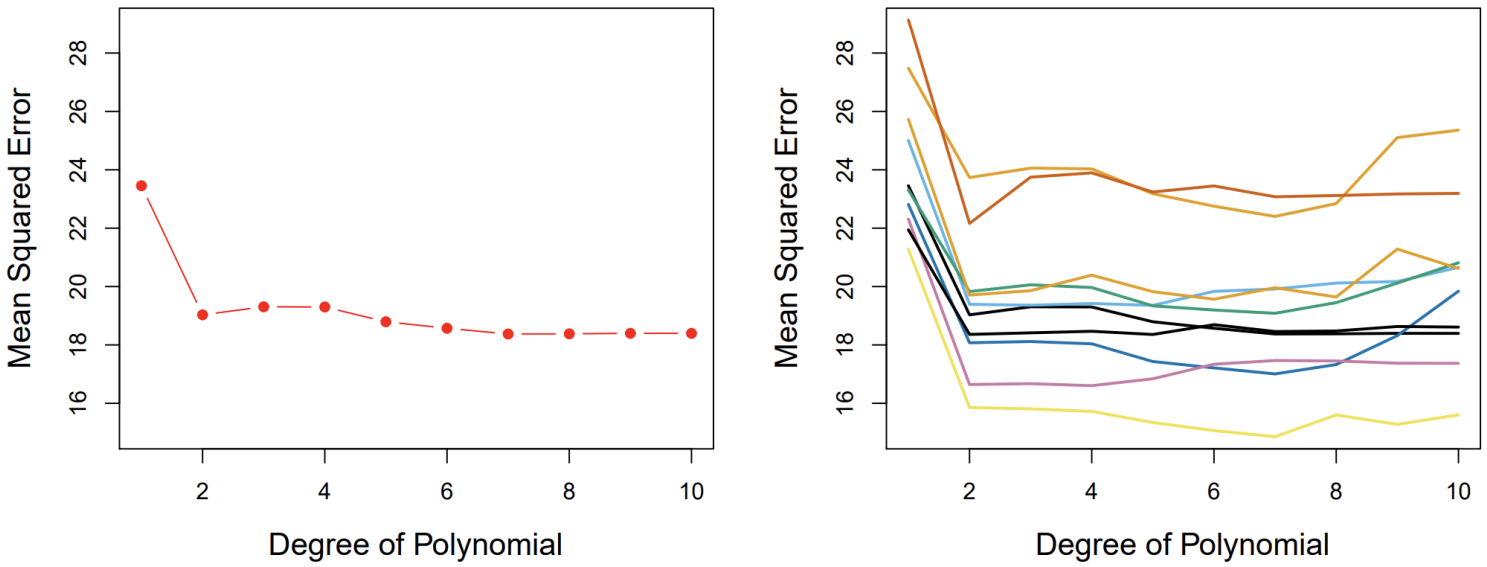


FIGURE 5.2. *The validation set approach was used on the* `Auto` *data set in order to estimate the test error that results from predicting* `mpg` *using polynomial functions of* `horsepower`. Left: *Validation error estimates for a single split into training and validation data sets.* Right: *The validation method was repeated ten times, each time using a different random split of the observations into a training set and a validation set. This illustrates the variability in the estimated test MSE that results from this approach.*

James, Gareth, et al. *An introduction to statistical learning: with applications in R.*

The validation set approach

- Drawbacks of this approach:
  - the estimate of the test error rate can be highly variable;
  - only half of the observations are used to train the model;
- Resampling methods: a refinement of the validation set approach that addresses the two issues.

Resampling methods

- Repeatedly draw samples from a training set and refit a model of interest (or compute certain estimates) on each sample to obtain additional information about the model (or estimates).
- Common methods include cross-validation and bootstrap.
- Useful when you have small sample size.

Cross-validation (CV)

- In most problems, there is no designated “test dataset” that is huge in size and set aside a priori. We can reserve part of training data as test.
- Cross-validation (CV) can be used to estimate the test error associated with a given statistical learning method:
  - to evaluate its performance (*model assessment*)
  - or to select the appropriate parameter (*model selection/tuning*)
- CV can be used for both classification and regression.
- A precursor of CV is the validation set approach that randomly divides the dataset into two equal parts.
- CV has a few variants; we only discuss the canonical version.

Leave-One-Out cross-validation (LOOCV)

- We first illustrate in the context of regression
- Suppose the training data contains $\{(x_1, y_1), \cdots, (x_n, y_n)\}$
- First, use $(n-1)$ observations $\{(x_2, y_2), \cdots, (x_n, y_n)\}$ to train and use the remaining observation (x_1, y_1) to evaluate the performance: $MSE_1 = (y_1 - \hat{y}_1)^2$
- Repeat this procedure by using (x_2, y_2) for the validation data, training on the $n-1$ observations $(x_1, y_1), (x_3, y_3), \cdots (x_n, y_n)$, and compute $MSE_2 = (y_2 - \hat{y}_2)^2$
- Repeat this approach n times produces n squared errors $MSE_1, \cdots, MSE_n$

- The LOOCV estimate for the test MSE is the average of these n estimates:

$$CV_{(n)} = \frac{1}{n} \sum_{i=1}^{n} MSE_i \, .$$

There is no need to randomly shuffle the training data for LOOCV. Why?

LOOCV

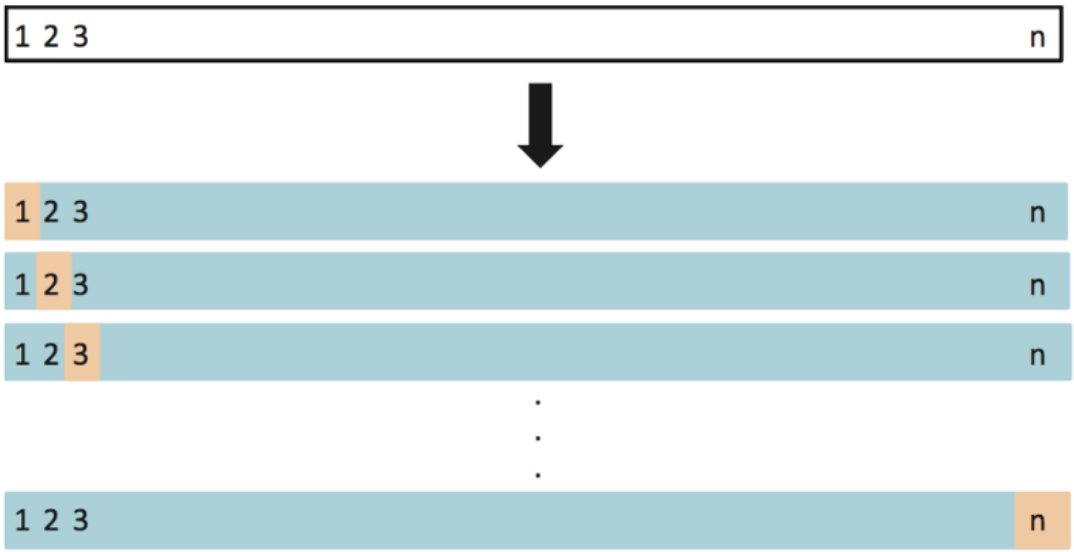


FIGURE 5.3. *A schematic display of LOOCV. A set of n data points is repeatedly split into a training set (shown in blue) containing all but one observation, and a validation set that contains only that observation (shown in beige). The test error is then estimated by averaging the n resulting MSE's. The first training set contains all but observation 1, the second training set contains all but observation 2, and so forth.*

James, Gareth, et al. *An introduction to statistical learning: with applications in R.*

More common: k-fold CV

- Computationally, LOOCV has the potential to be expensive to implement, since the model has to be fit n times
- k-fold CV: randomly divide the set of observations into k groups, or folds, of approximately equal size. The first fold is treated as a validation set, and the method is fit on the remaining $k-1$ folds. The mean squared error, MSE_1, is then computed on the observations in the hold-out fold
- This procedure is repeated k times; each time, a different group of observations is treated as a validation set
- This process results in k estimates of the test error, $MSE_1, MSE_2, \cdots, MSE_k$
- The k-fold CV estimate is computed by averaging these values,

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^{k} MSE_i$$

- We commonly use $CV_{(k)}$ to estimate the test error (model evaluation)

K-fold CV

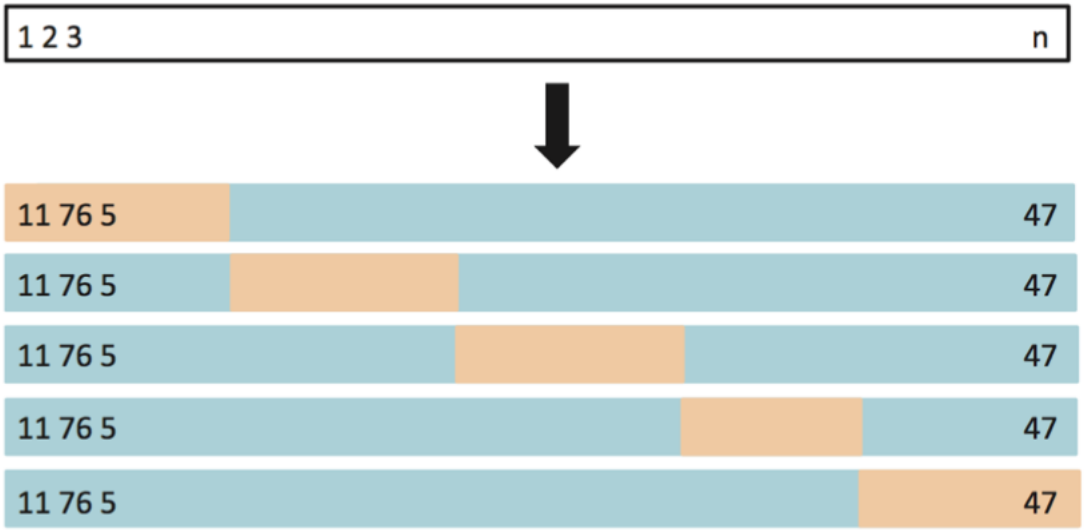


FIGURE 5.5. *A schematic display of 5-fold CV. A set of n observations is randomly split into five non-overlapping groups. Each of these fifths acts as a validation set (shown in beige), and the remainder as a training set (shown in blue). The test error is estimated by averaging the five resulting MSE estimates.*

LOOCV is a special case of k-fold CV in which $k = n$.

James, Gareth, et al. *An introduction to statistical learning: with applications in R*.

K-fold CV is very close to LOOCV

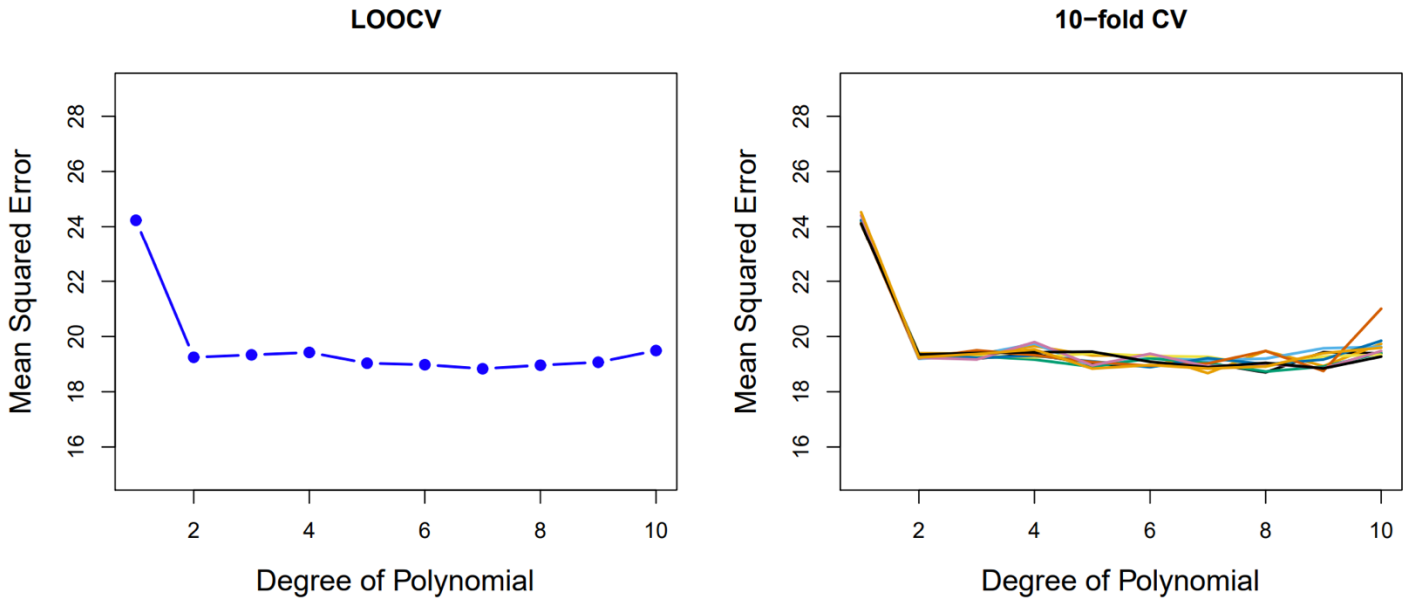


FIGURE 5.4. *Cross-validation was used on the* `Auto` *data set in order to estimate the test error that results from predicting* `mpg` *using polynomial functions of* `horsepower`. Left: *The LOOCV error curve.* Right: 10-*fold CV was run nine separate times, each with a different random split of the data into ten parts. The figure shows the nine slightly different CV error curves.*

Why is the variance of test errors in 10-fold CV much lower than that in the validation set approach?

More about k-fold CV

- The usual choices for *k* in practice are *k*=5 and *k*=10.
- Sometimes, when people have enough computing power, they do 10-fold CV multiple times and then take the average performance.
- The cross-validation criterion should be consistent with prediction evaluation criterion (group discussion on the following questions):

    - Think about the MSE in 10-fold CV for the regression we just introduced. Are they consistent with the out-of-sample MSE?

    - For classification, if we care about classification error, what CV criterion should we use? What if we care about the ROC?

k-fold CV for model tuning/selection

- Suppose in the Auto dataset, we are deciding between two linear models.
- model 1: use displacement to predict mpg
- model 2: use displacement and horsepower to predict mpg
- To choose between these two models using 5-fold CV, we calculate $CV_{(5)}$ for both models and choose the model with the better $CV_{(5)}$.
- The same idea extends to the case of many models.

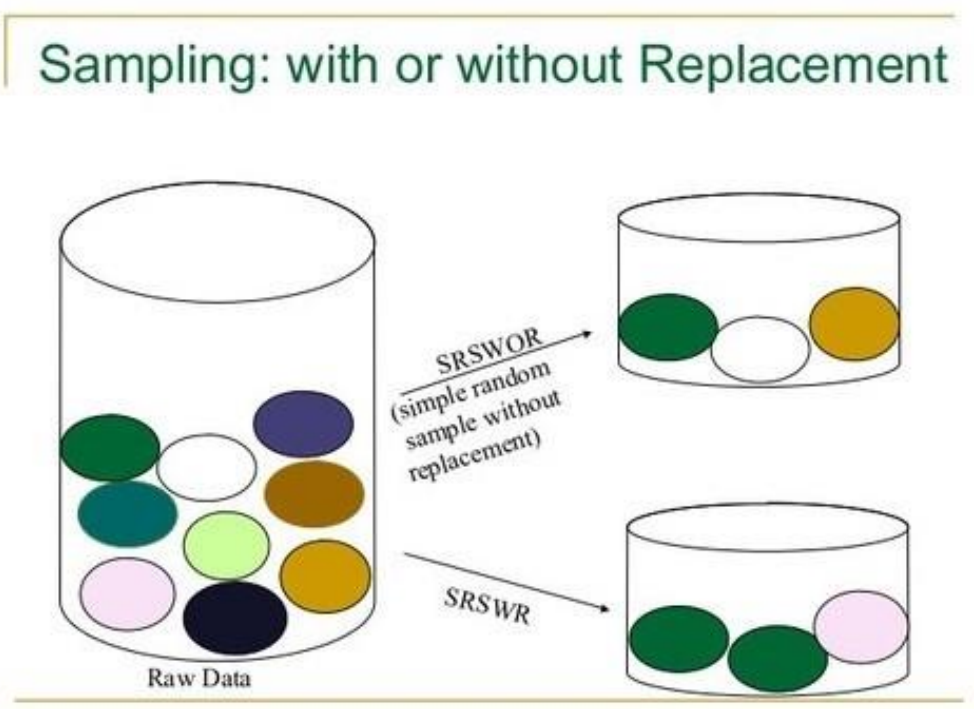
After you have chosen the best model, which of the 5 linear prediction equations do you use finally?

None! We refit the model with all the available training data.

Can we use k-fold CV to fine-tune parameters in a model?

Bootstrap in statistics

- Bootstrap: any procedure that relies on random sampling with replacement (from the original sample);
- Random sampling with vs. without replacement;



Bootstrap

- As a verb, bootstrap means “get oneself out of a situation using existing resources without extra help”.
- Bootstrapping is a procedure for estimating the distribution of an estimator by resampling (often with replacement) existing data.
- Bootstrap can be used to estimates the properties of a statistic. (e.g., mean, variance, confidence intervals, prediction error, etc.)
- This is a rather strange idea when we first look at it.
- However, it is one of the most influential ideas in statistics in the second half of the 20th century.
- Bootstrap is computationally heavy.
- Bootstrap has different versions (e.g., moving block bootstrap for time series data). We only discuss the simplest kind.

With bootstrap, how do we estimate the confidence interval of sample mean?

Bootstrap works in practice

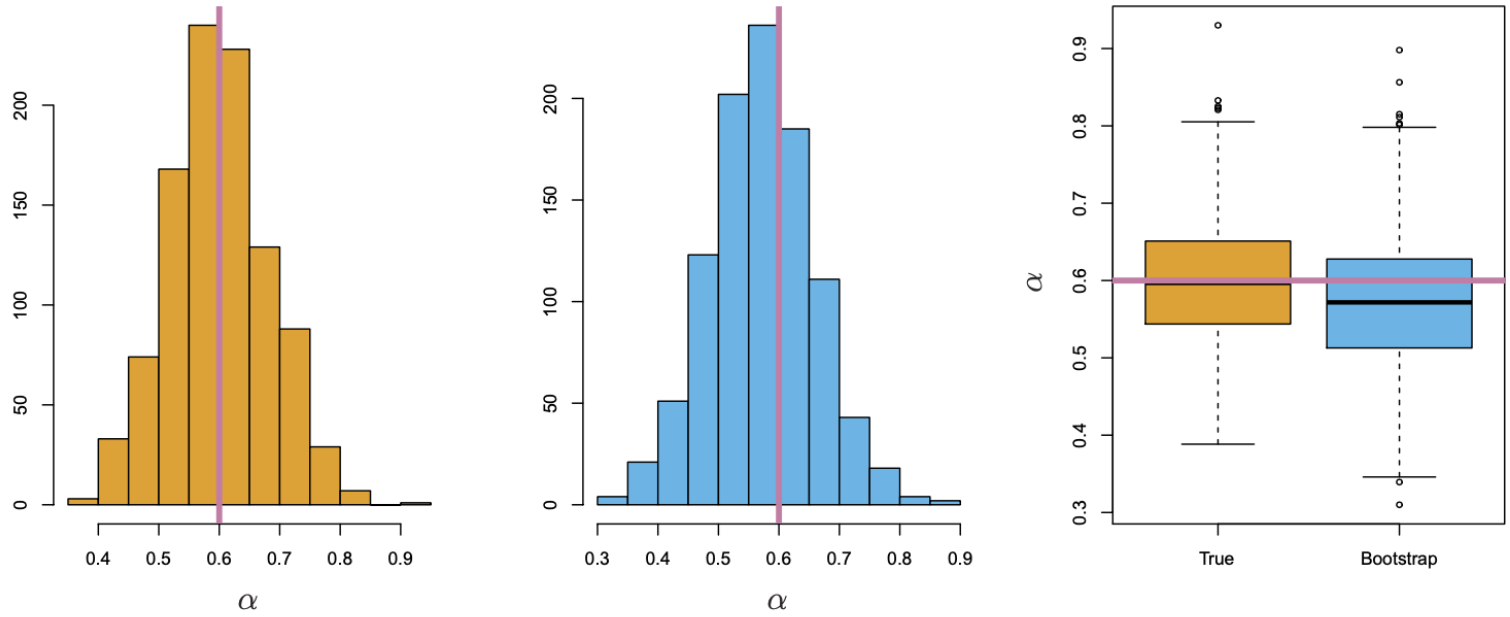
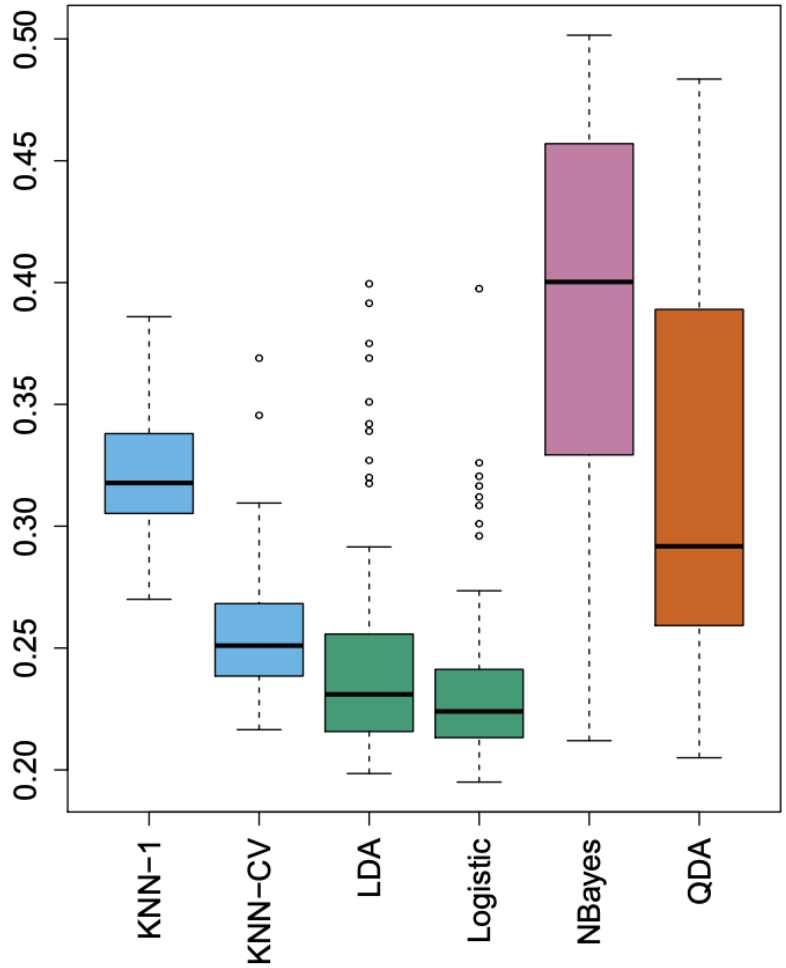


FIGURE 5.10. Left: *A histogram of the estimates of* α *obtained by generating 1,000 simulated data sets from the true population.* Center: *A histogram of the estimates of* α *obtained from 1,000 bootstrap samples from a single data set.* Right: *The estimates of* α *displayed in the left and center panels are shown as boxplots. In each panel, the pink line indicates the true value of* α.

Visualizing prediction errors



Unsupervised Learning

- No outcome variable, just a set of predictors or features measured on a set of samples.

- Difficult to know how well our models are doing.

- Less common compared with supervised models.
- Can be useful as a pre-processing step for supervised learning.

Unsupervised learning

We will talk about two topics:

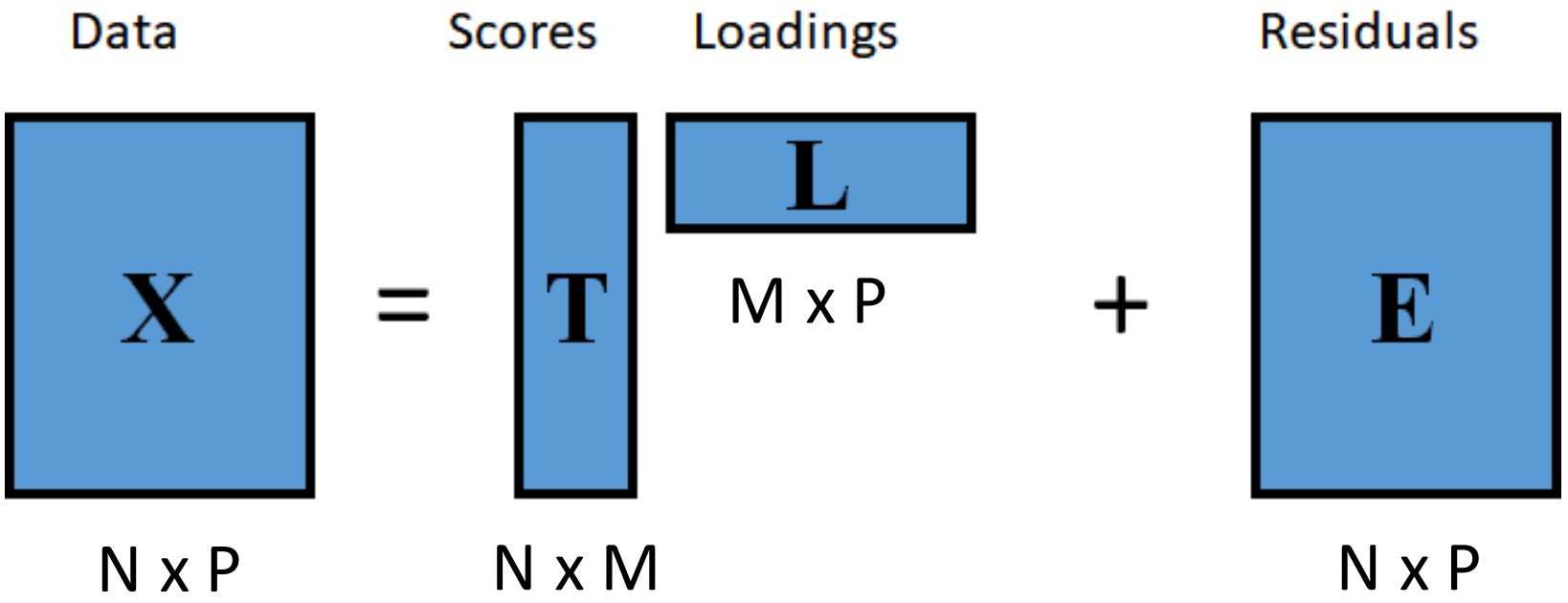
(1) *principal components analysis*: for data visualization or data pre-processing before other techniques are applied;

(2) *clustering*: discovering unknown subgroups in data;

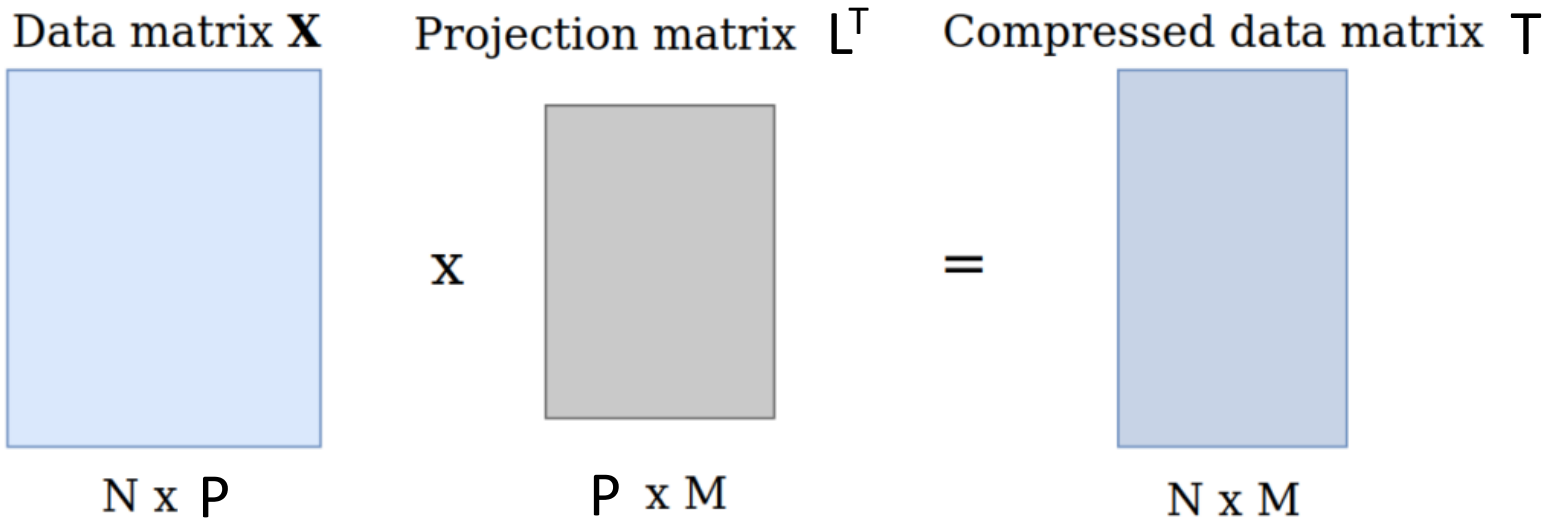
Principal Components Analysis

- Principal component analysis (PCA) is the process by which principal components are computed, and the subsequent use of these fewer components in understanding the data.
- The principal components are major directions in feature space along which the original data are highly variable.
- If we were to do scatterplot for every pair of variables when p = 10, how many scatterplot do we need? Limitation?
- We need a better method to visualize all observations in a single plot (often in a 2d-space) when p is large.

PCA as matrix factorization



PCA for data reduction



In practice, we want M << P (such as M = 2 or 3).

Principal Components Analysis

- Given a $n \times p$ dataset X , how do we compute the first principal component?
- First principal component of a set of features $X_1, X_2, \cdots, X_p$ is the *normalized* linear combination of p features:

$$Z_1 = \phi_{11} X_1 + \phi_{21} X_2 + \cdots + \phi_{p1} X_p \,, \text{ where } \sum_{j=1}^{p} \phi_{j1}^2 = 1 \,.$$

Such that Z_1 has the largest variance.

Principal Components Analysis

- Assume each variable has been centered to have mean zero.
- The first principal component vector solves the optimization problem:

$$\max_{\phi_{11},\cdots,\phi_{p1}} \left\{ \frac{1}{n} \sum_{i=1}^{n} z_{i1}^2 \right\}, \text{ subject to } \sum_{j=1}^{p} \phi_{j1}^2 = 1 .$$

- If we project n data points $x_1, \cdots, x_n$ on to the first PC, the projected values are $z_{11}, \cdots, z_{n1}$
- We refer to $z_{11}, \cdots, z_{n1}$ as the scores of the first principal component.
- We refer to $\phi_{11}, \cdots, \phi_{p1}$ as the loadings of the first principal component.
- The loading vector $\phi_1 = (\phi_{11}, \cdots, \phi_{p1})^T$ defines a direction in feature space along which the data vary the most.
- The second principal component is the linear combination of $X_1, \cdots, X_p$ that has the *maximal* variance out of all linear combinations that are ***uncorrelated*** with the first component Z_1.

Principal Components Analysis

- The second principal components scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12}x_{i1} + \phi_{22}x_{i2} + \cdots + \phi_{p2}x_{ip}$$

- "Z_2 and Z_1 are uncorrelated" is equivalent to "ϕ_1 is perpendicular to ϕ_2"
- The maximum number of PCs is min($n - 1$, p)
- Example USArrests data: For each of 50 states in the United States, the data set contains the number of arrests per 100, 000 residents for each of three crimes: Assault, Murder, and Rape. We also record UrbanPop (the percent of population living in urban areas) for each state. We use the following (4 x 2) vectors to project the data from (50 x 4) to (50 x 2):

	PC1	PC2
Murder	0.5358995	−0.4181809
Assault	0.5831836	−0.1879856
UrbanPop	0.2781909	0.8728062
Rape	0.5434321	0.1673186

Biplot for PCA

Biplot shows both the principal component scores and loadings.

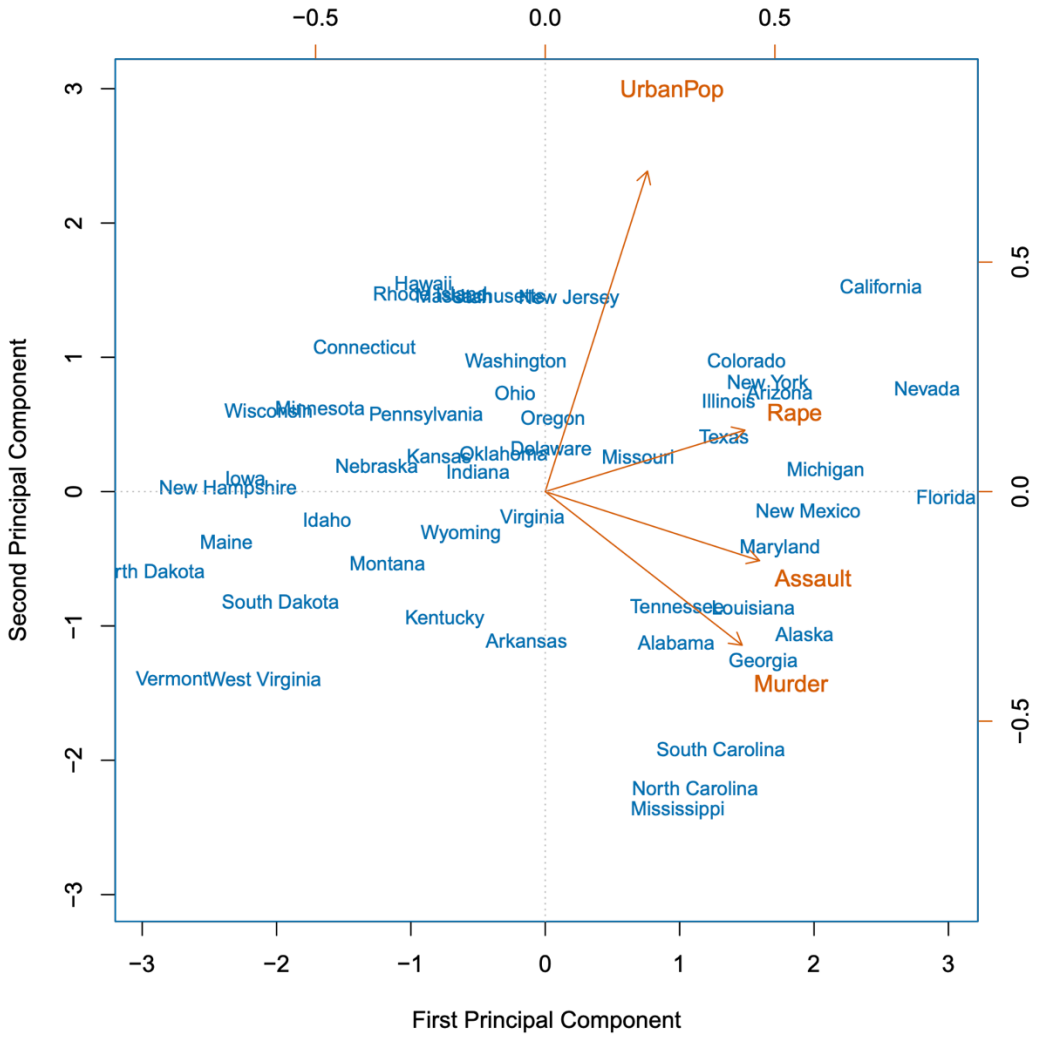
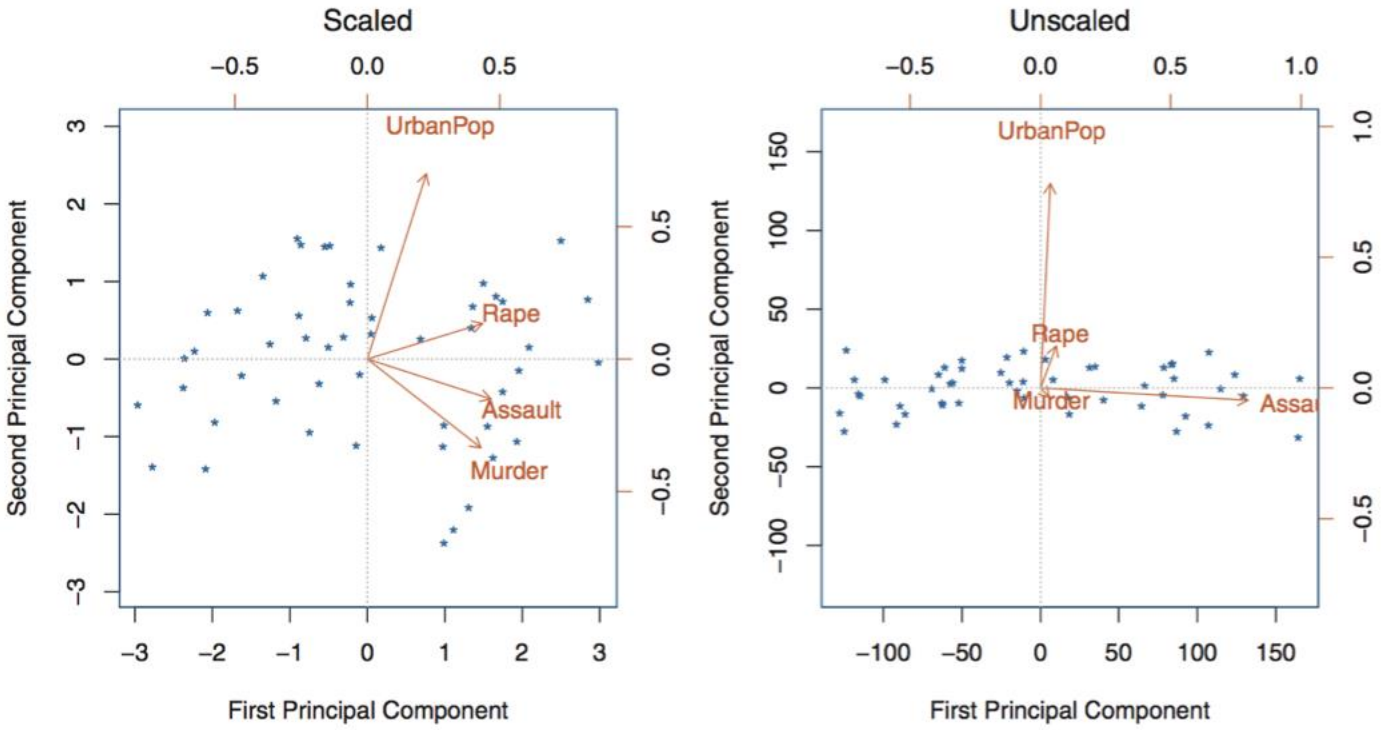


Fig. shows the first two principal components of the USArrests data.

The blue state names represent scores for the first two principal components.

The orange arrows indicate the first two principal component loading vectors (with axes on the top and right).

Center variables before PCA



Two principal component biplots for the USArrests data. Left: the same as previous one, with the variables scaled to have unit standard deviations. Right: principal components using unscaled data. Assault has by far the largest loading on the first principal component because it has the highest variance among the four variables. Usually, we standardize the variables to have 0 mean and SD 1 before doing PCA.

Proportion of variance explained

- We are interested in knowing the *proportion of variance explained* (PVE) by each principal component.
- The total variance present in a data set (assuming that the variables have been centered to have mean zero) is defined as:

$$\sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2$$

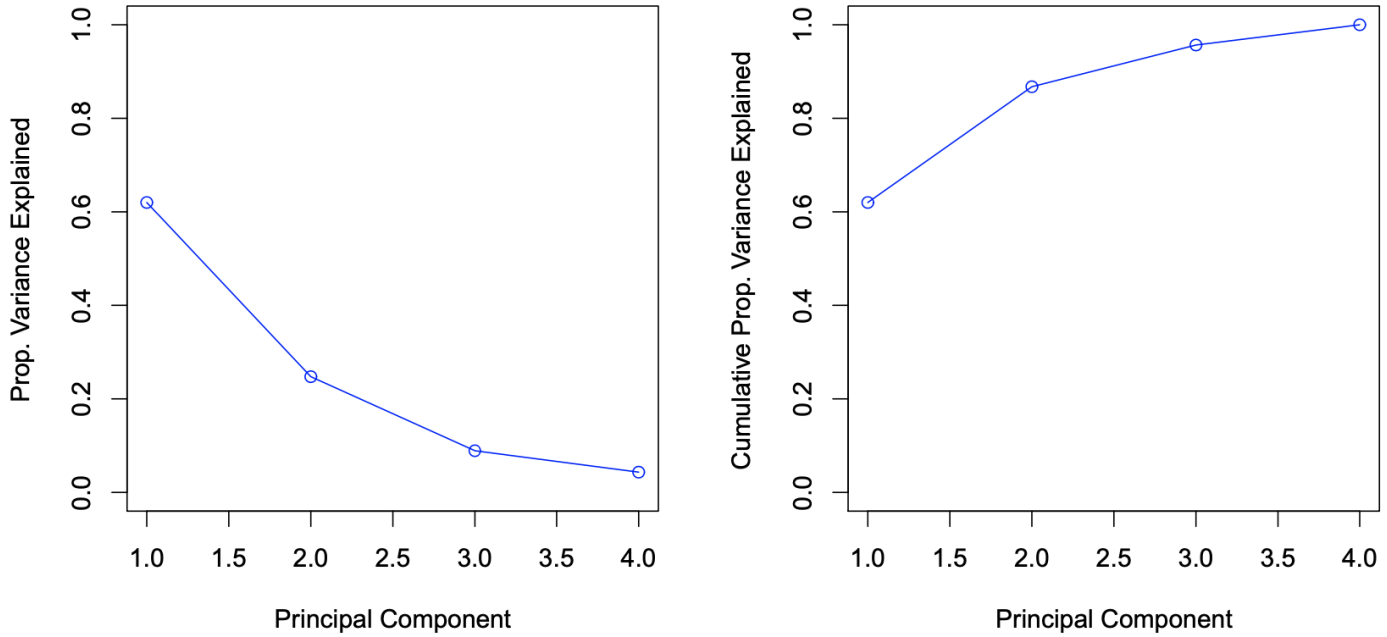
- The variance explained by the *m*th principal component is:

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

- The PVE of the *m*th principal component is given by:

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

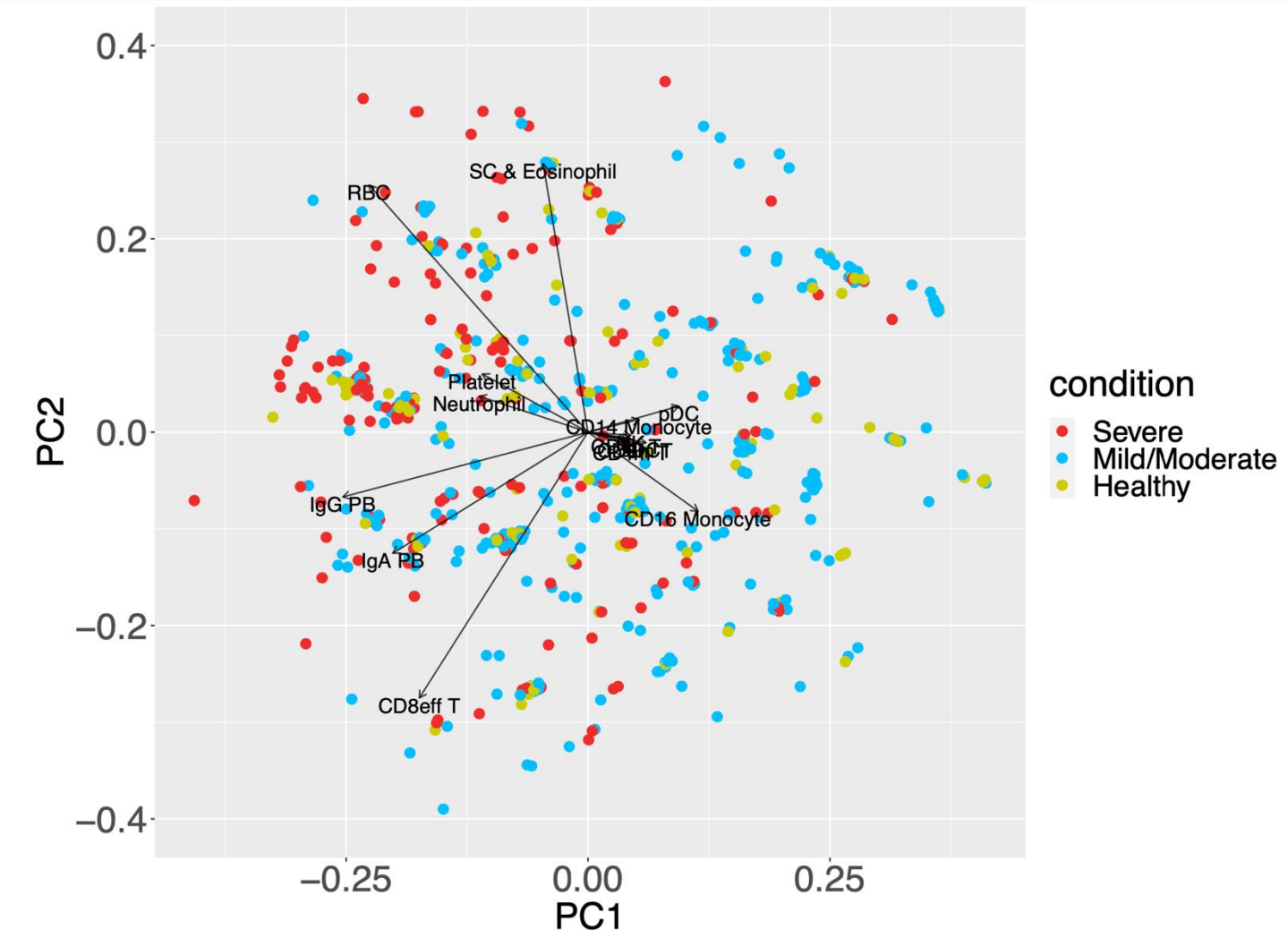
Proportion of variance explained



Left: *a scree plot depicting the proportion of variance explained by each of the four principal components in the* USArrests *data*. Right: *the cumulative proportion of variance explained by the four principal components in the* USArrests *data*.

The question of how many principal components are enough is inherently ill-defined. It depends on the specific area of application and the specific data set.

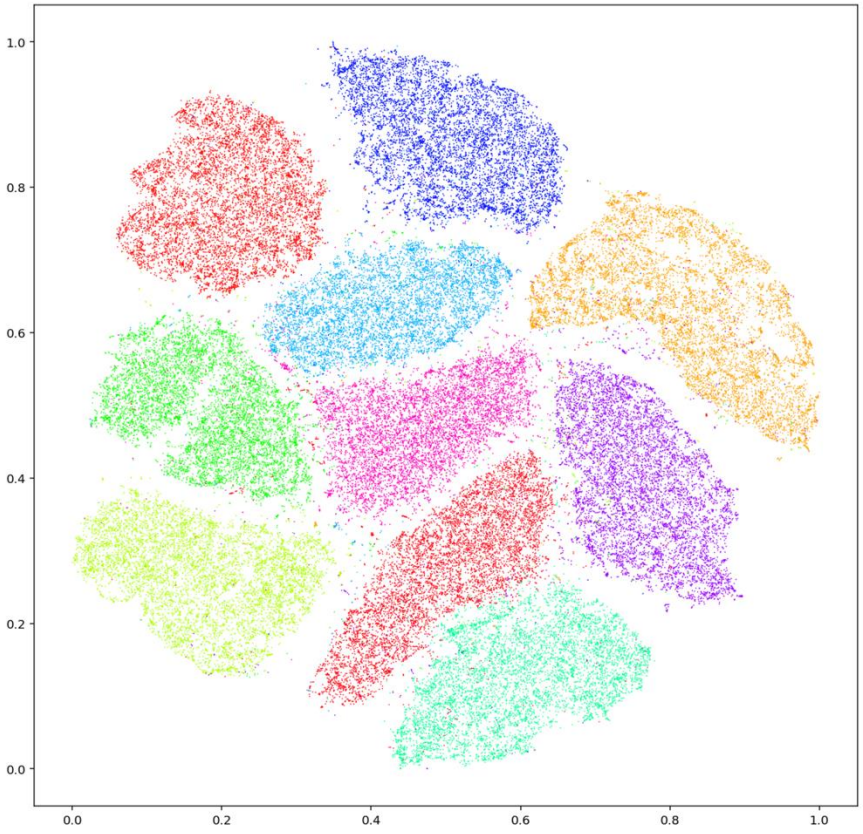
PCA does not always work well



Other dimension reduction method

t-SNE: t-distributed Stochastic Neighbor Embedding

MNIST dataset



T-SNE embeddings of MNIST dataset

https://scikit-learn.org/1.5/modules/generated/sklearn.manifold.TSNE.html

Introduction to clustering

- Clustering refers to a very broad set of techniques for finding subgroups, or clusters, in a dataset.
- Seek to partition observations into distinct groups so that observations within each group are similar with each other, while observations in different groups are different.
- What does it mean to be “different” and “similar”? Often it is a domain specific question.
- Sometimes (but not always), people do PCA (or other similar techniques) first and then do clustering on reduced dataset.

Different clustering approaches

There exist a great number of clustering methods.

We focus on *K-means clustering* and *hierarchical clustering.*

- K-means: partition all observations into a pre-specified number of clusters;

- hierarchical clustering: do not know in advance how many clusters we want; end up with a tree-like visual representation of observations, called a *dendrogram;*

K-means clustering

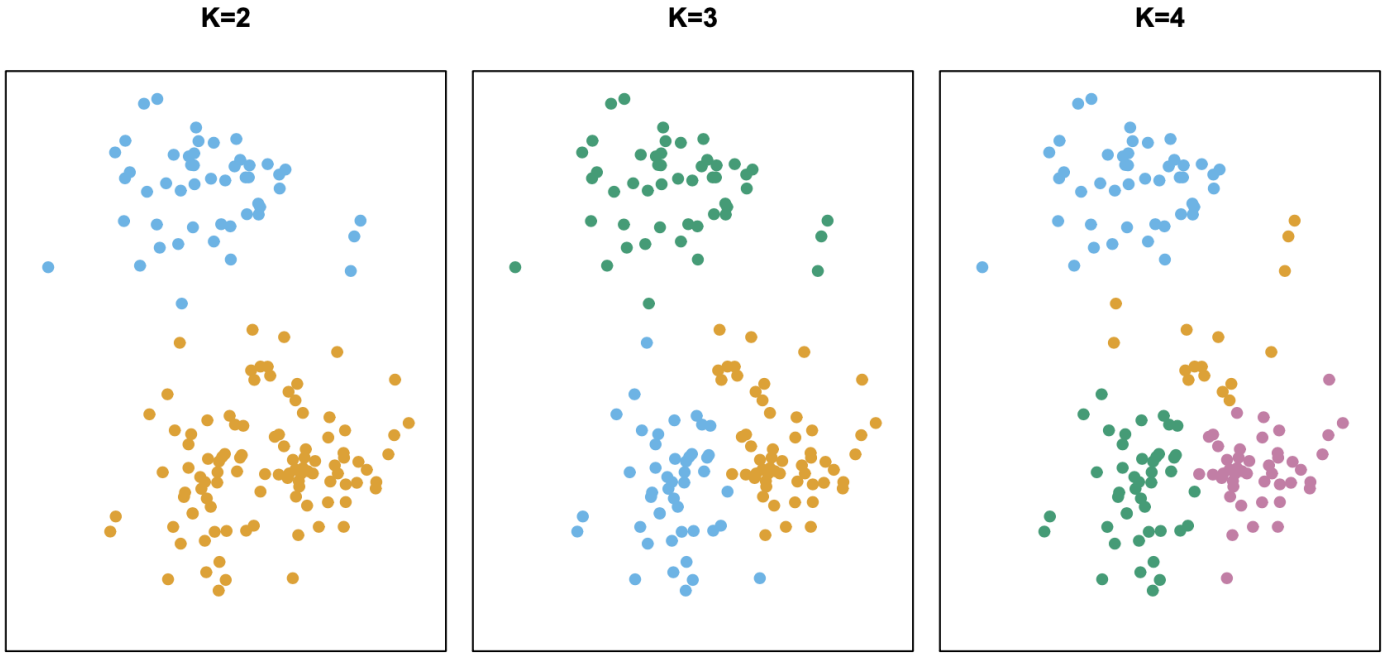
- Let $C_1, \cdots, C_K$ denote sets containing the indices of the observations in each cluster. These sets satisfy two properties:
  - $C_1 \cup C_2 \cdots \cup C_K = \{1, \cdots, n\}$
  - C_k and $C_{k'}$ have empty intersection, for different k and k'
- Idea for k-means clustering: a good clustering is one for which the within-cluster variation is as small as possible
- $W(C_k)$: a measure of the amount by which the observations within a cluster differ from each other, defined by

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

- K-means algorithm solves

$$\mathit{minimize}_{C_1,\cdots,C_K} \left\{ \sum_{k=1}^{K} W(C_k) \right\}$$

Examples



A simulated data set with 150 observations in two-dimensional space. Panels show the results of applying K*-means clustering with different* K*. The color of each observation indicates the cluster to which it was assigned using the* K*-means clustering algorithm. Note that there is no ordering of the clusters, so the cluster coloring is arbitrary.*

K-means algorithm

- Brutal force way is computationally infeasible
- The following algorithm finds an approximate solution

---

1. Randomly assign a number, from 1 to K, to each of the observations. These serve as initial cluster assignments for the observations.

2. Iterate until the cluster assignments stop changing:

   (a) For each of the K clusters, compute the cluster *centroid*. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster.

   (b) Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

K-means algorithm

- The Algorithm is guaranteed to decrease the value of the following objective at each step:

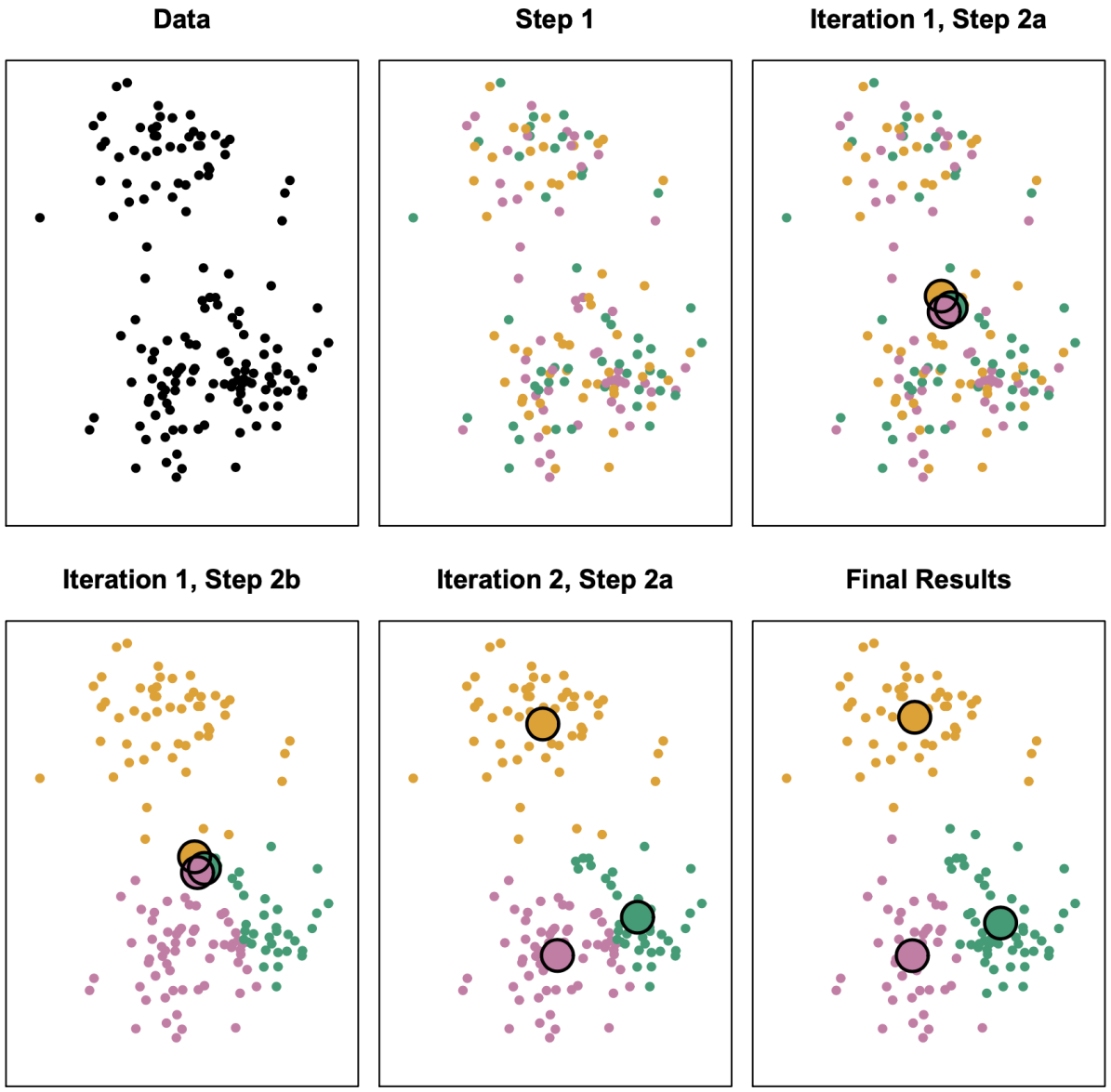
$$\mathit{minimize}_{C_1,\cdots,C_K} \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2$$

- To understand this, we need:

$$\frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} (x_{ij} - \bar{x}_{kj})^2 ,$$

where $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$

The progress of the K-means algorithm



K-means in practice

- Because the K-means algorithm finds a local rather than a global optimum, the results obtained will depend on the initial (random) cluster assignment of each observation.
- It is therefore recommended to run the algorithm multiple times using different random initial configurations.

K-means produces different results

Hierarchical clustering

- Hierarchical clustering is an alternative approach to K-means clustering which does not require that we commit to a particular choice of *K.*
- We will discuss *bottom-up* or *agglomerative* clustering.
- Which is the most common type of hierarchical clustering. It refers to the fact that a dendrogram (generally depicted as an upside-down tree) is built starting from the leaves and combining clusters up to the trunk.

Hierarchical clustering algorithm

---

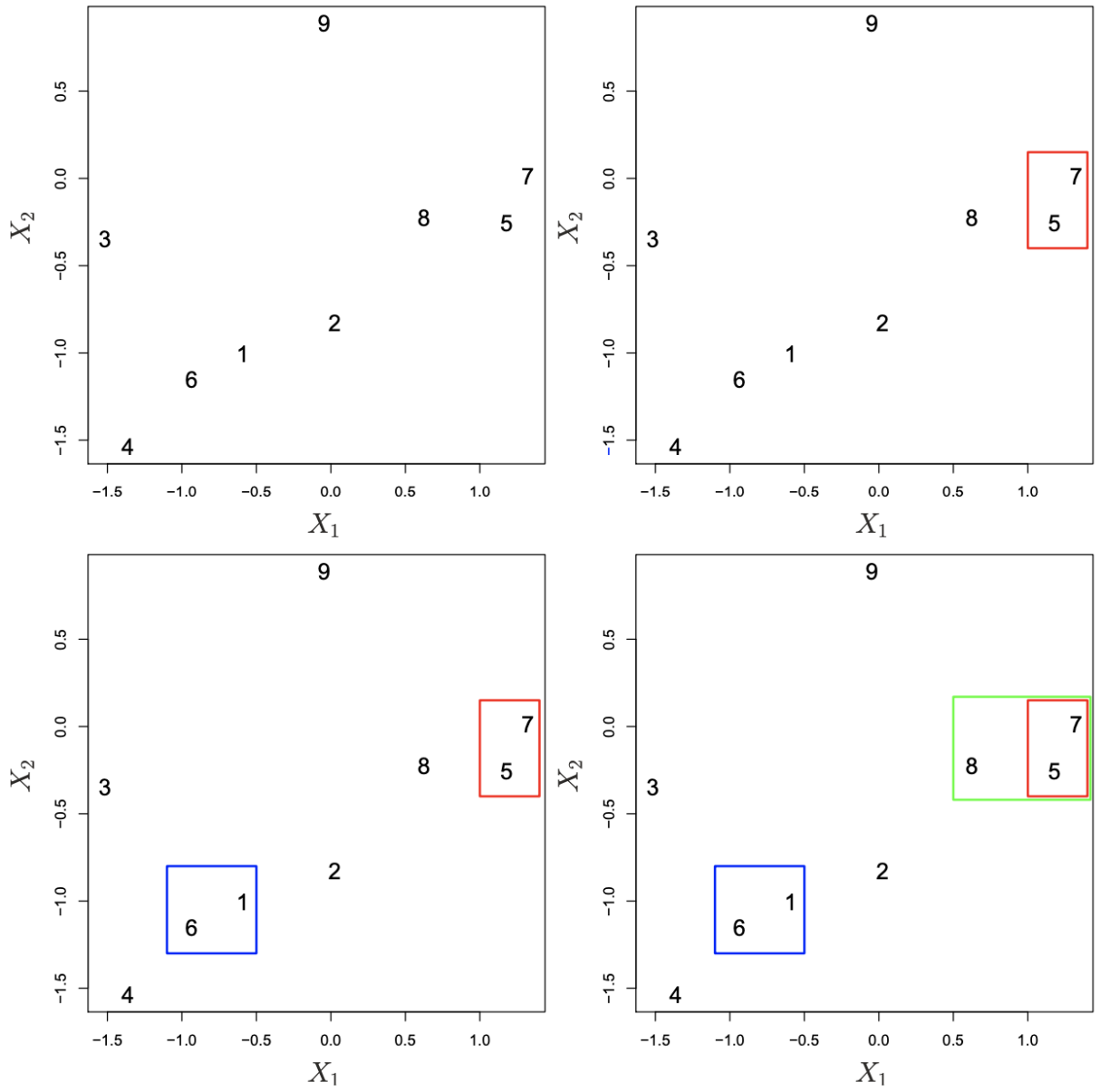
1. Begin with n observations and a measure (such as Euclidean distance) of all the $\binom{n}{2} = n(n-1)/2$ pairwise dissimilarities. Treat each observation as its own cluster.

2. For $i = n, n-1, \ldots, 2$:

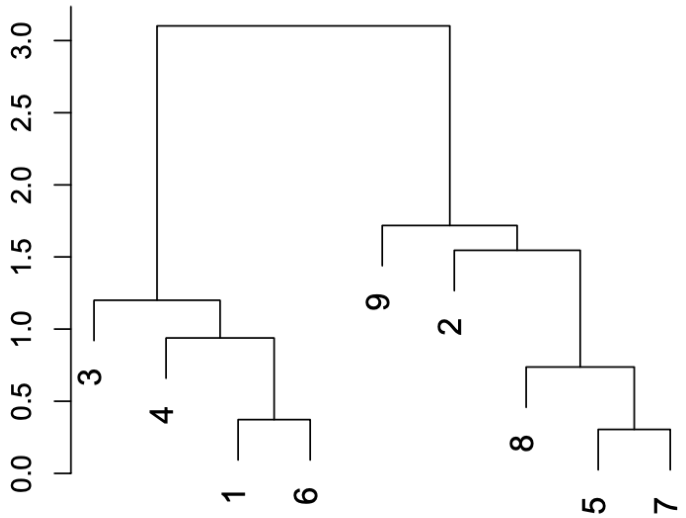
   (a) Examine all pairwise inter-cluster dissimilarities among the i clusters and identify the pair of clusters that are least dissimilar (that is, most similar). Fuse these two clusters. The dissimilarity between these two clusters indicates the height in the dendrogram at which the fusion should be placed.

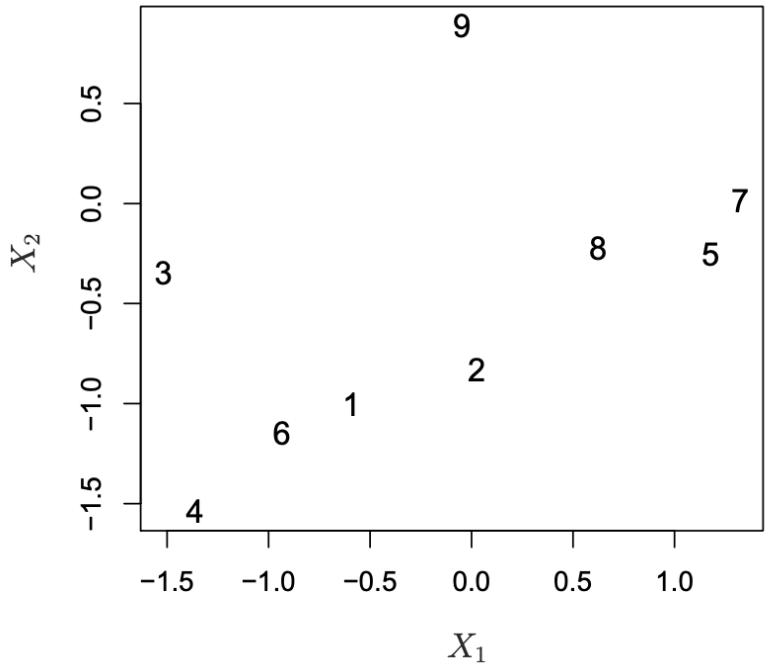
   (b) Compute the new pairwise inter-cluster dissimilarities among the $i-1$ remaining clusters.

Hierarchical clustering in action

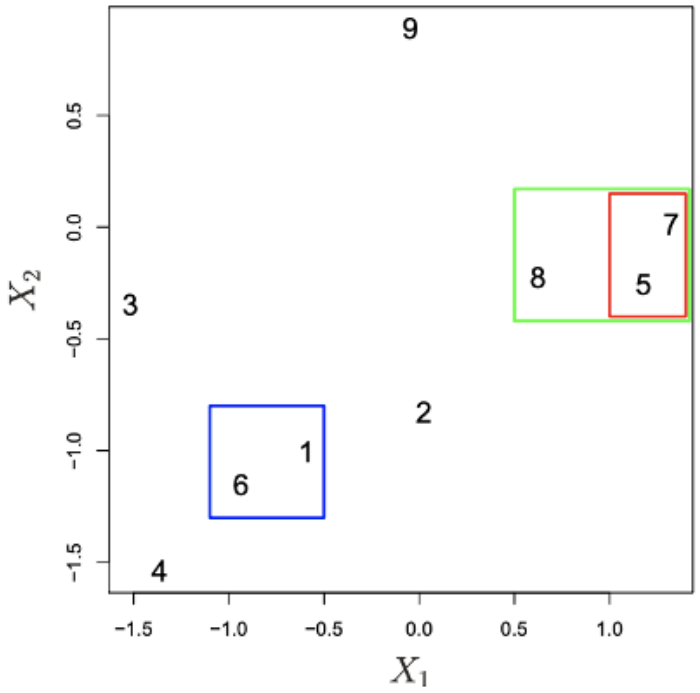


Hierarchical clustering in action





Hierarchical clustering algorithm

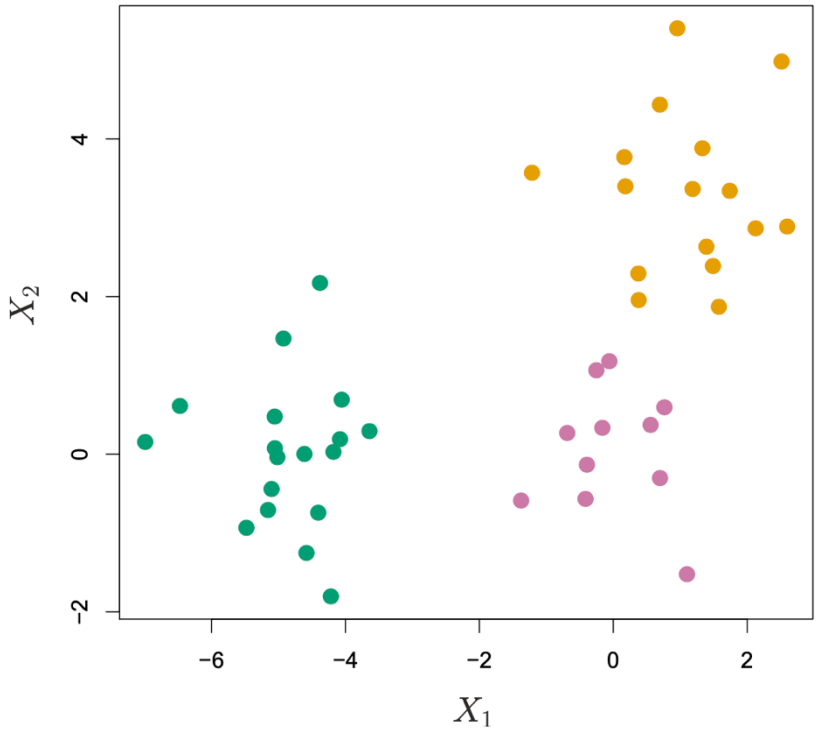


How can we calculate the distance between two clusters with multiple observations?

Four common types of linkage in hierarchical clustering

Linkage	*Description*
Complete	Maximal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities. Single linkage can result in extended, trailing clusters in which single observations are fused one-at-a-time.
Average	Mean intercluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

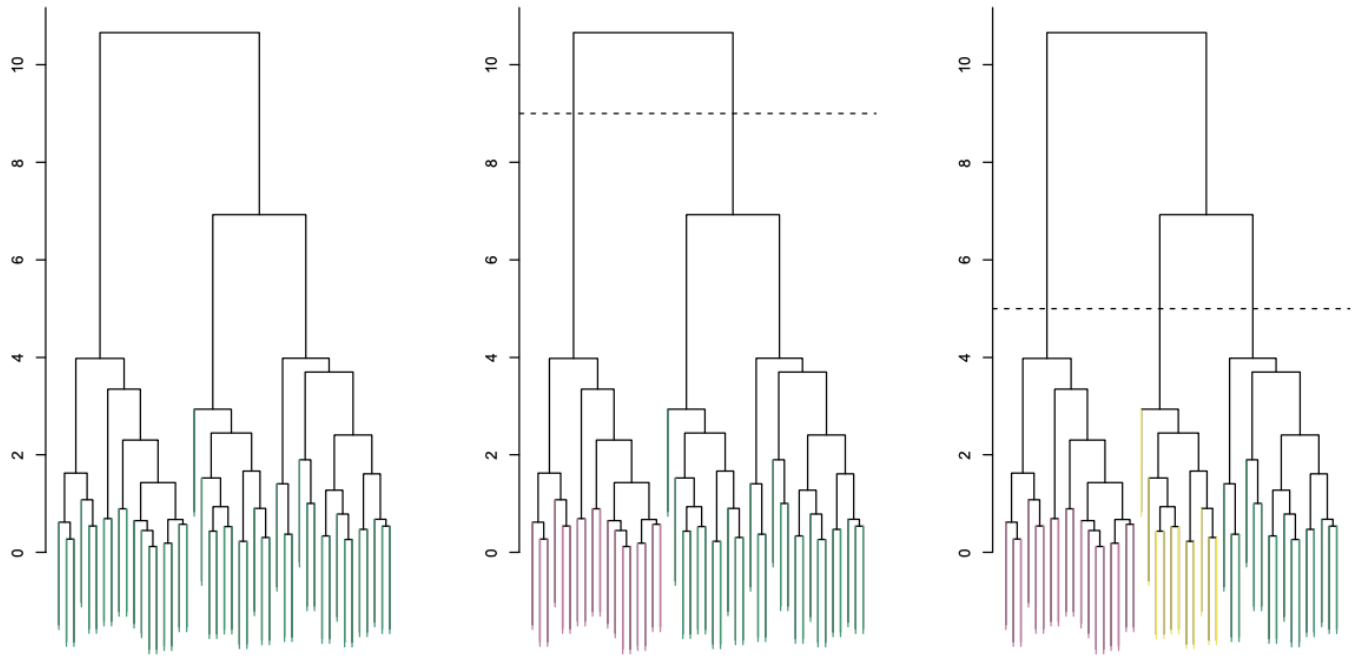
Hierarchical clustering example



Toy dataset:

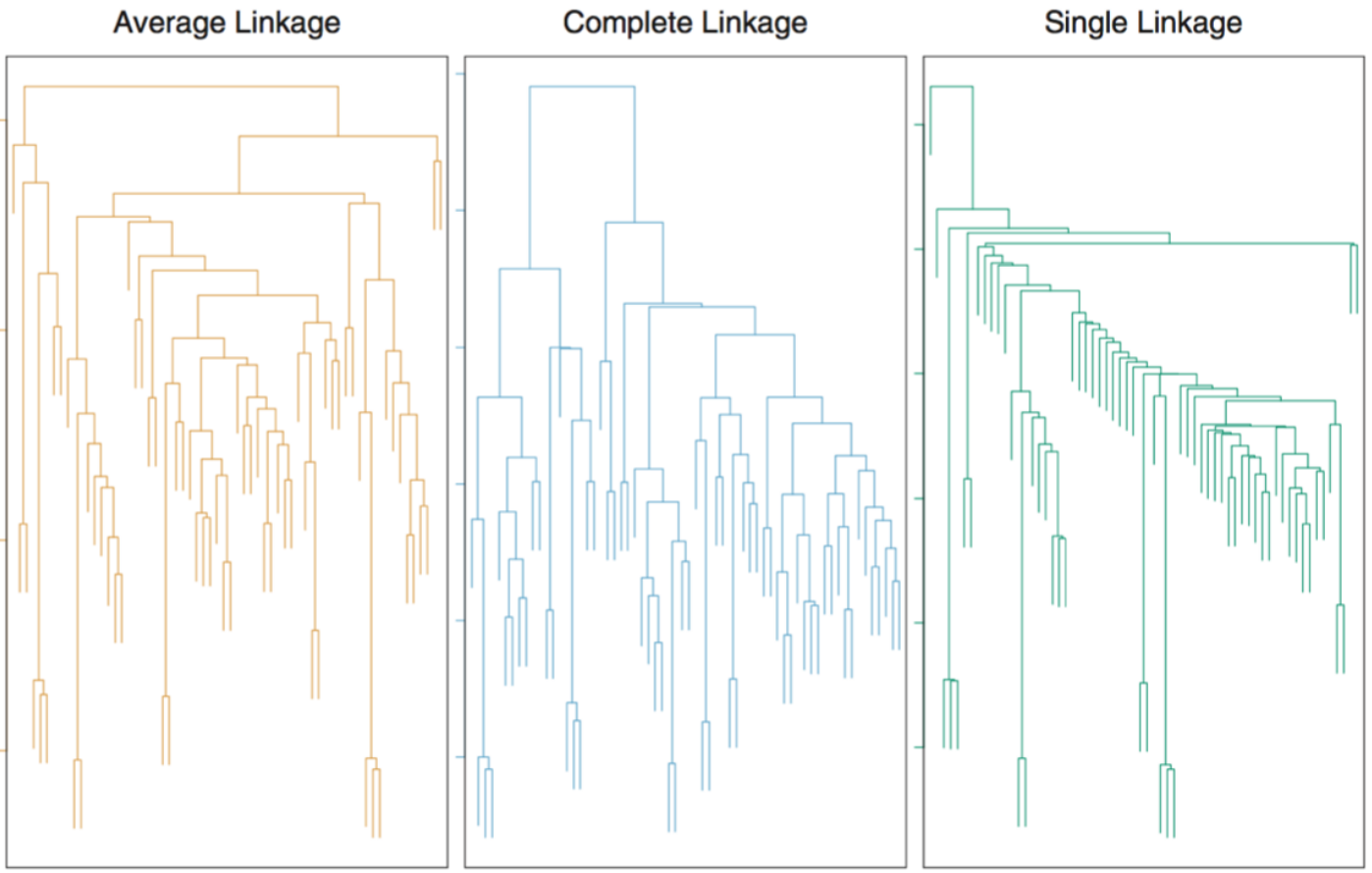
45 observations generated in two-dimensional space. There are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to discover the classes from the data by clustering the observations.

Dendrogram produced with different linkage



Left: *dendrogram obtained from hierarchically clustering the data from the previous figure with complete linkage and Euclidean distance.* Center: *the dendrogram from the left-hand panel, cut at a height of nine (indicated by the dashed line). This cut results in two distinct clusters, shown in different colors.* Right: *the dendrogram from the left-hand panel, now cut at a height of five. This cut results in three distinct clusters, shown in different colors.*

Dendrogram produced with different linkage



Average, complete, and single linkage applied to an example data set. Average and complete linkage tend to yield more balanced clusters.

Practical considerations

- Should the features first be rescaled in some way?
- K-means clustering, how many clusters should we look for in the data?
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
  - Where should we cut the dendrogram to obtain clusters?
- Answers: For these methods, there is no single right answer – any solution that exposes some interesting aspects of the data should be considered.
- In practice, we try several different choices and look for the one with the most useful or interpretable solution.
- Caution: clustering results should not be taken as the truth of a data set.

Course notes

- Text analysis V2 next week
- Midterm the week after
  - Closed book
  - 2-hour in class
  - 3 A4 CheatSheet
- Hw1 due before Midterm