Unsupervised Learning
(ISLR 10.1-10.3)

Yingbo Li

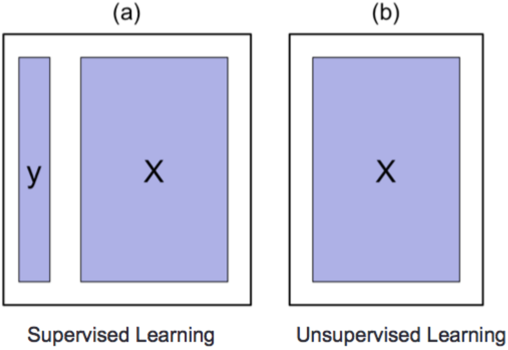
Southern Methodist University

STAT 4399

Outline

Supervised vs. Unsupervised Learning

- Supervised learning: both X and Y are known
- Unsupervised learning: only X is known



Unsupervised Learning

Goals: to discover interesting things:

- Clustering: discover unknown subgroups in data. Examples:
  - subgroups of breast cancer patients grouped by their gene expression measurements,
  - groups of shoppers characterized by their browsing and purchase histories,
- Data visualization: low dimensional representation of high-dimensional data

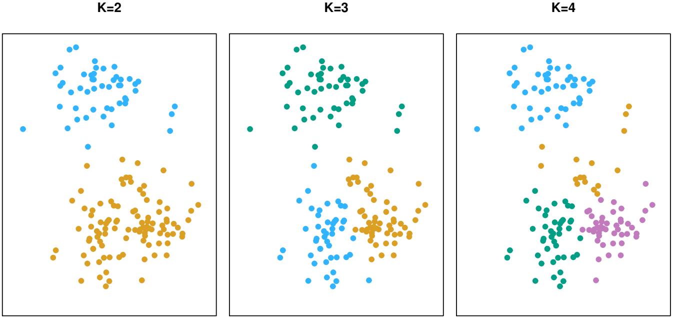
Challenge: more subjective (than supervised learning)
There is no simple goal for the analysis, such as prediction of a response.

Clustering

- *Clustering* refers to a very broad set of techniques for finding subgroups, or clusters, in a data set.
- A good clustering is one when the observations within a group are similar but between groups are very different.
- We must define what it means for two or more observations to be similar or different.
  - Distance in $\mathbb{R}^p$ Euclidian space
  - Other metrics?
- This is often a domain-specific consideration that must be made based on knowledge of the data being studied.
- There are many different types of clustering methods.

K-Means Clustering

- One must first specify the parameter K, number of clusters.
- Then the algorithm will assign each observation to exactly one of the K clusters.



There is no ordering of the clusters, so the cluster coloring is arbitrary.

How Does K-Means Work?

- We would like to partition that data set into K clusters

$$C_1, C_2, \ldots, C_K$$

  - Non-overlapping clusters
  - Each observation belongs to one cluster
  - If the ith observation belongs to the kth cluster, then $i \in C_k$
- The idea behind K-means clustering: *a good clustering is one for which the within-cluster variation is as small as possible*.
- The optimization problem of K-means:

$$\min_{C_1, \ldots, C_K} \sum_{k=1}^{K} W(C_k),$$

where $W(C_k)$ is the amount of difference among the points in C_k.

K-Means Algorithm

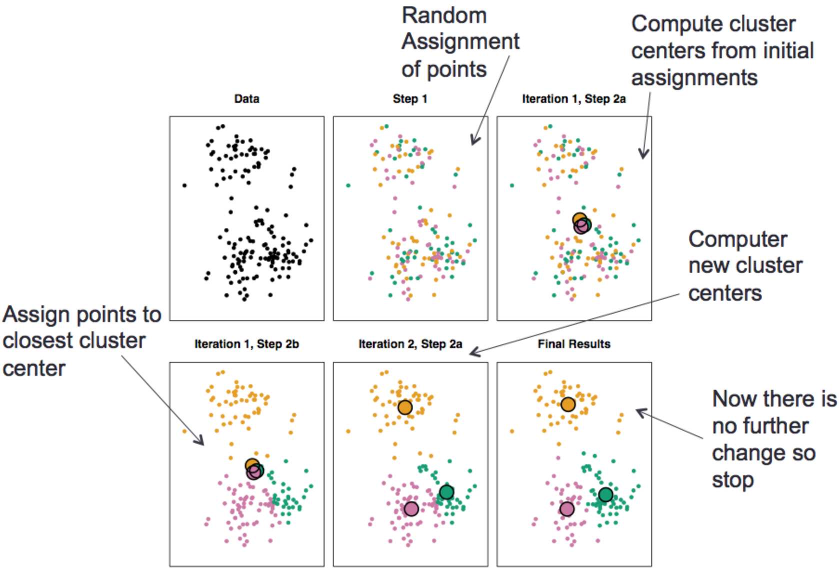
- Squared Euclidean distance

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

- K-Means Algorithm
  1. Initial assignment:
     Randomly assign each observation to one of K clusters.
  2. Iterate until the cluster assignments stop changing:
     1. For each cluster, compute the cluster centroid $(\bar{x}_{k1}, \ldots, \bar{x}_{kp})$.
     2. Assign each observation to the cluster whose centroid is closest (in Euclidean distance).

- This algorithm is guaranteed to decrease the value of the objective $\sum_{k=1}^{K} W(C_k)$ at each step.

An Illustration of the K-Means Algorithm

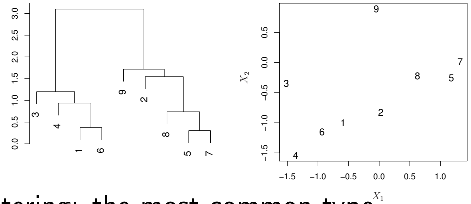


Different starting values

- The K-means algorithm can get stuck in "local optimums".
- Hence, it is important to run the algorithm multiple times with random starting points to find a good solution
- Those labeled in red all achieved the same best solution, with an objective value of 235.8

Hierarchical Clustering

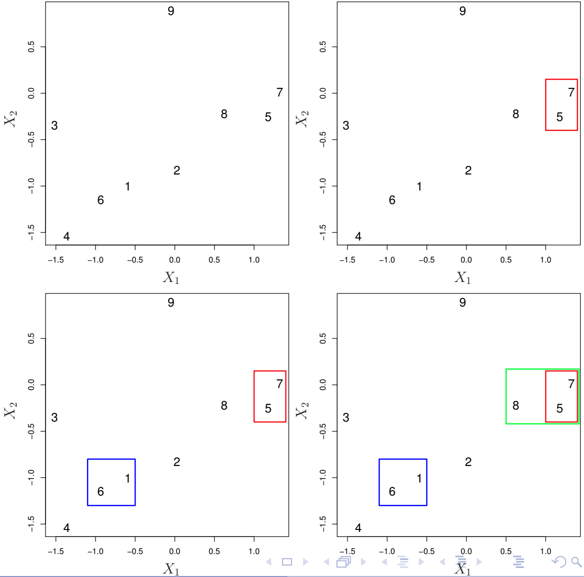
- K-means clustering requires choosing the number of clusters. If we don't want to do that, an alternative is to use hierarchical clustering.
- Hierarchical clustering has an added advantage that it produces a tree based representation of the observations, called a *Dendrogram*.



- Bottom-up clustering: the most common type
  - Start with each point in its own cluster.
  - Calculate a measure of *dissimilarity* between all clusters
  - Identify the closest two clusters and merge them.
  - Repeat.
  - Ends when all points are in a single cluster.

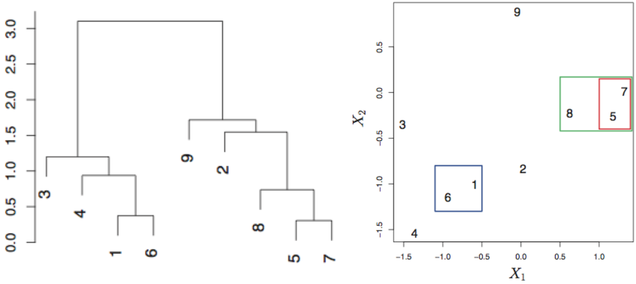
Builds a Hierarchy in a "Bottom-up" Fashion

- Start with 9 clusters
- Fuse 5 and 7
- Fuse 6 and 1
- Fuse the (5,7) cluster with 8.
- Continue until all observations are fused.



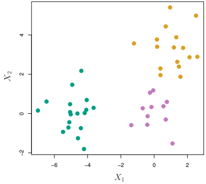
Dendrogram

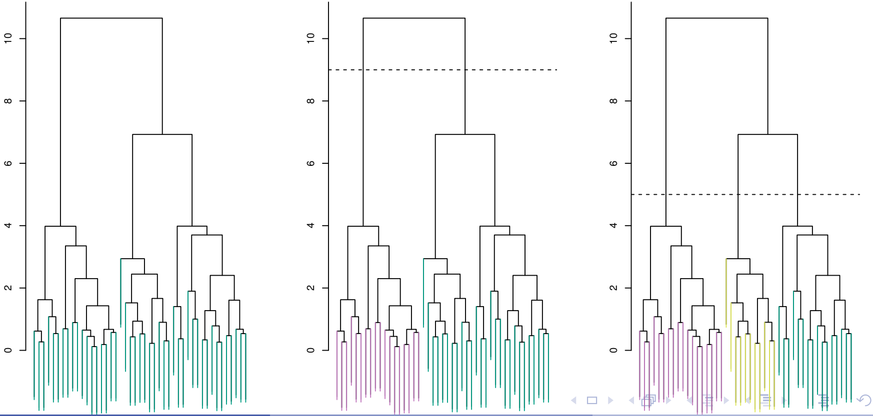
- Each "leaf" of the dendrogram represents one of the 9 observations
- Height of fusing (on vertical axis) indicates how similar the points are.
  - Obs. 9 is no more similar to 2 than it is to 8, 5, and 7, even though 9 and 2 are close together in terms of horizontal distance.
  - This is because 2, 8, 5, and 7 all fuse with 9 at the same height, approximately 1.8.



Choosing Clusters

- We draw lines across the dendrogram
- We can form any number of clusters depending on where we draw the break point.

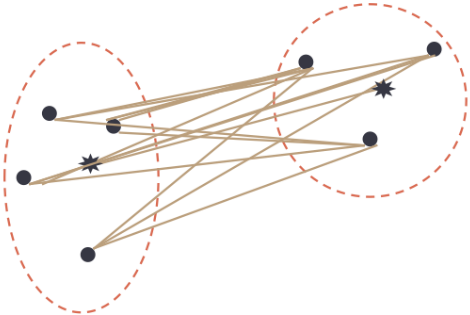




Dissimilarity

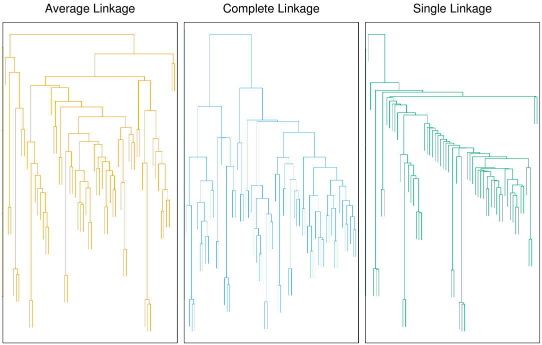
How do we define the *dissimilarity*, or *linkage*, between two clusters?

- Complete linkage: largest distance between observations
- Single linkage: smallest distance between observations
- Average linkage: average distance between all pairs of observations
- Centroid linkage: distance between centroids of the observations



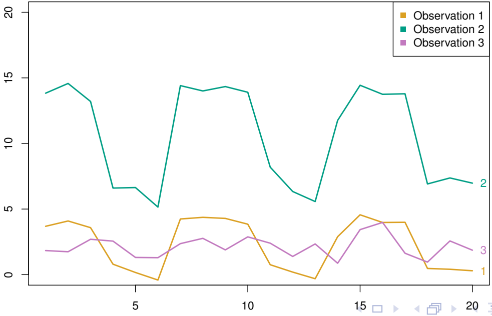
Linkage Can be Important

- Using the same data but difference linkage methods, the clustering results are very different!
- Complete and average linkage tend to yield evenly sized clusters, whereas single linkage tends to yield extended clusters to which single leaves are fused one by one.



Choice of Dissimilarity Measure

- So far, we have used Euclidean distance as the dissimilarity measure.
- An alternative is *correlation-based distance* which considers two observations to be similar if their features are highly correlated.
- In this example, we have 3 observations and $p = 20$ variables:
  - In terms of Euclidean distance obs. 1 and 3 are similar
  - However, obs. 1 and 2 are highly correlated so would be considered similar in terms of correlation measure



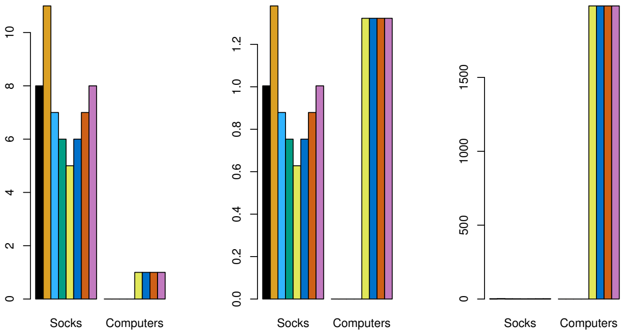
Online Shopping Example

- Suppose we record the number of purchases of each item (columns) for each customer (rows)
- Using Euclidean distance, customers who have purchases very little will be clustered together
- Using correlation measure, customers who tend to purchase the same types of products will be clustered together even if the magnitude of their purchase may be quite different

Standardizing the Variables

Consider an online shop that sells two items: socks and computers

- Left: in terms of quantity, socks have higher weight
- Center: after standardizing, socks and computers have equal weight
- Right: in terms of dollar sales, computers have higher weight



Practical Issues in Clustering

- Should the features first be standardized? i.e. Have the variables centered to have a mean of zero and standard deviation of one.
- In case of hierarchical clustering:
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
  - Where should we cut the dendogram in order to obtain clusters?
- In case of K-means clustering:
  - How many clusters should we look for the data?
- We should try several different choices, and look for the one with the most useful or interpretable solution. There is no single right answer!

Principal Components Analysis (PCA)

- PCA produces a low-dimensional representation of a dataset. It finds a sequence of linear combinations of the variables that have maximal variance, and are mutually uncorrelated.
- PCA can produce derived variables for use in supervised learning.
- For unsupervised learning, PCA serves as a tool for data visualization.

Principal Component Analysis

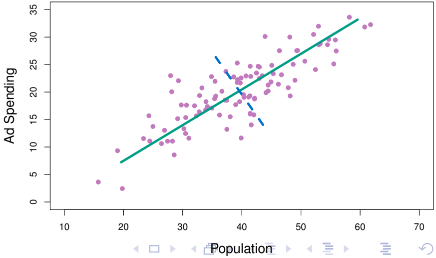
The *first principal component* of a set of predictors $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the predictors

$$Z_1 = \phi_{11}X_1 + \phi_{21}X_2 + \cdots + \phi_{p1}X_p$$

that has the largest variance.

- *Loadings*: $\phi_1 = (\phi_{11}, \phi_{21}, \ldots, \phi_{p1})$.
- By *normalized*, we mean that the ℓ_2 norm $\|\phi_1\|_2 = 1$.
- The loading vector ϕ_1 defines a direction in feature space along which the data vary the most.

- The population size and ad spending for 100 different cities
- The first principal component
- The second principal component



Computation of Principal Components

- Suppose we have a $n \times p$ data set $\mathbf{X}$, whose columns have been centered to have mean zero and variance one. Why?
- The optimization problem:

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

- *Scores* of the first component: for $i = 1, 2, \ldots, n$

$$z_{i1} = \phi_{11}x_{i1} + \phi_{21}x_{i2} + \cdots + \phi_{p1}x_{ip}$$

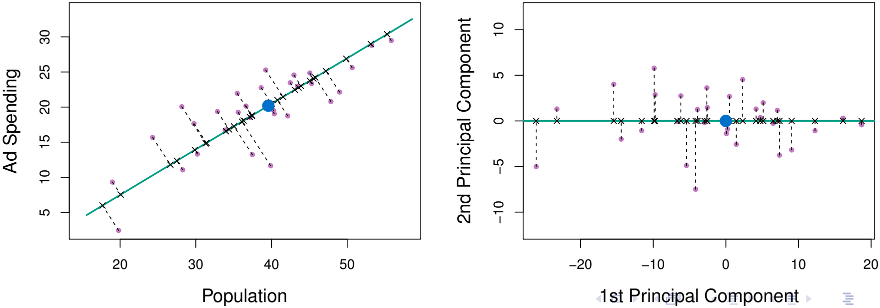
- If we project the n data points $x_1, \ldots, x_n$ onto the first principle component direction, the projected values are the scores $z_{11}, \ldots, z_{n1}$.

Further Principal Components

- The second principal component is the linear combination of $X_1, X_2, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.
- The second principal component scores $z_{12}, \ldots, z_{n2}$ take the form

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

- $\phi_2 = (\phi_{12}, \ldots, \phi_{p2})$ is the second principal component loading vector.



The USArrests Data

For each of the fifty 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`
- the percent of the population in each state living in urban areas `UrbanPop`

PCA loadings of the first two principal components

	PC1	PC2
Assault	0.58	-0.19
Murder	0.54	-0.42
Rape	0.54	0.17
UrbanPop	0.28	0.87

- PC1: overall rates of serious crimes
- PC2: the level of urbanization

Biplot for the First Two Principal Components

- Blue state names: the scores for the first two principal components
- Orange arrows: the first two principal component loading vectors