Unsupervised learning: clustering and PCA

Yifei Sun

Department of Biostatistics
Mailman School of Public Health
Columbia University

Unsupervised Learning

- Most of the course focused on supervised learning
  - Observe both features $X_1, X_2, \ldots, X_p$ and a response Y
  - The goal is then to predict Y using $X_1, X_2, \ldots, X_p$
- We now focus on unsupervised learning
  - Observe only the features $X_1, X_2, \ldots, X_p$

The Goals of Unsupervised Learning

- The goal is to discover interesting things about the measurements
  - Is there an informative way to visualize the data?
  - Can we discover subgroups among the variables or among the observations?
- We discuss two methods
  - Clustering - discovering unknown subgroups
  - Principal components analysis

Why Unsupervised Learning

- Unsupervised learning is more subjective than supervised learning
- There is no simple goal for the analysis, such as prediction
- But techniques for unsupervised learning are of growing importance
  - Subgroups of breast cancer patients grouped by their gene expression measurements
  - Groups of shoppers characterized by their browsing and purchase histories
- Advantage: easier to obtain unlabeled data, from a lab instrument or a computer, than labeled data which can require human intervention

PCA vs Clustering

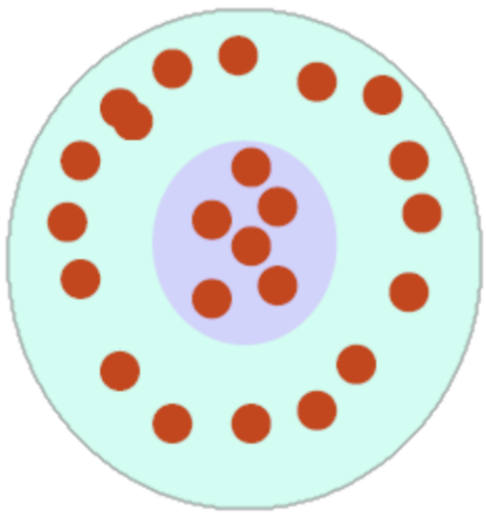
- PCA looks for a low-dimensional representation of the observations that explains a good fraction of the variance
- Clustering looks for homogeneous subgroups among the observations

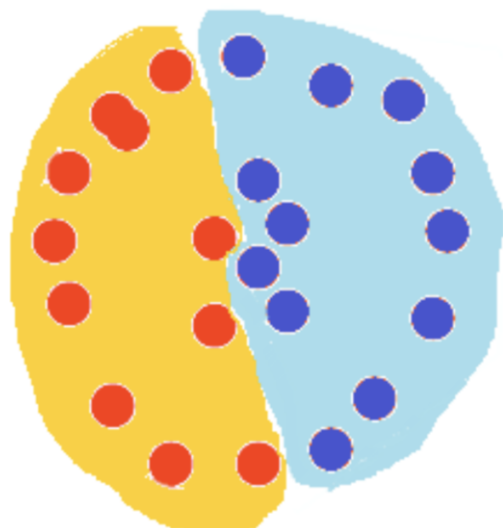
Clustering

- Clustering refers to a very broad set of techniques for finding subgroups or clusters in a data set
- We seek a partition of the data into distinct groups so that the observations within each group are quite similar each other

What do we need for clustering?

- Proximity measure
  - Dissimilarity measure: small if x_i, x_j are similar
- Criterion function to evaluate a clustering





Dissimilarity measures

- $x_i = (x_{i1}, \ldots, x_{ip})$
- Euclidean distance

$$d(x_i, x_{i'}) = \sqrt{\sum_{j=1}^{p} (x_{ij} - x_{i'j})^2}$$

- Manhattan distance

$$d(x_i, x_{i'}) = \sum_{j=1}^{p} |x_{ij} - x_{i'j}|$$

- More generally, Minkowski distance

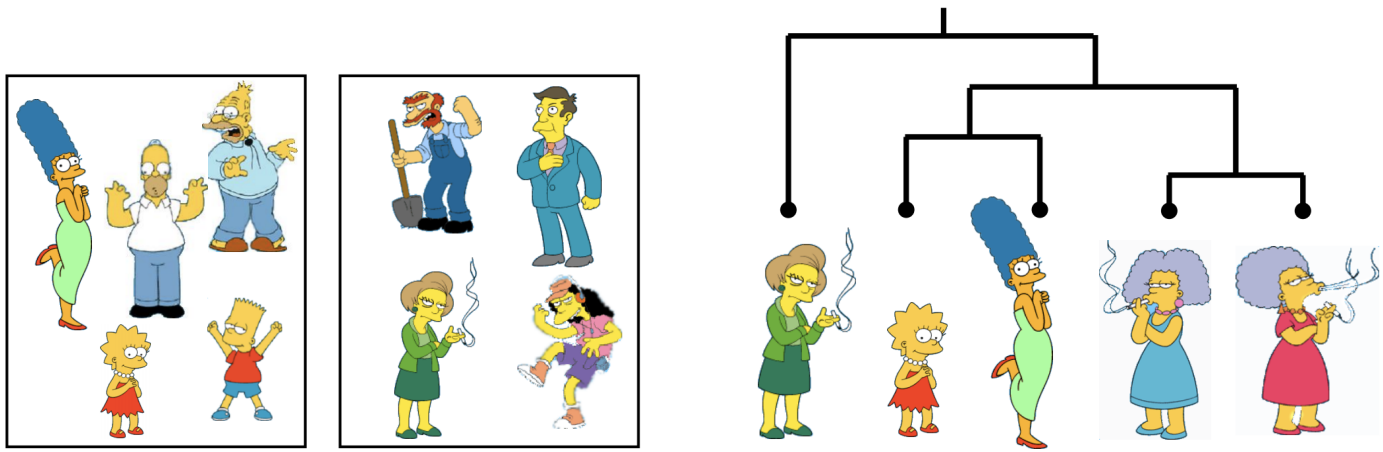
$$d_q(x_i, x_{i'}) = \left(\sum_{j=1}^{p} |x_{ij} - x_{i'j}|^q \right)^{\frac{1}{q}}$$

Cluster Evaluation

- Intra-cluster cohesion
  - How near the data points in a cluster are to the cluster centroid
  - Sum of squared error is a commonly used measure
- Inter-cluster separation
  - Different cluster centroids should be far away from one another
- In many applications, expert judgments are still the key

Two Clustering Methods

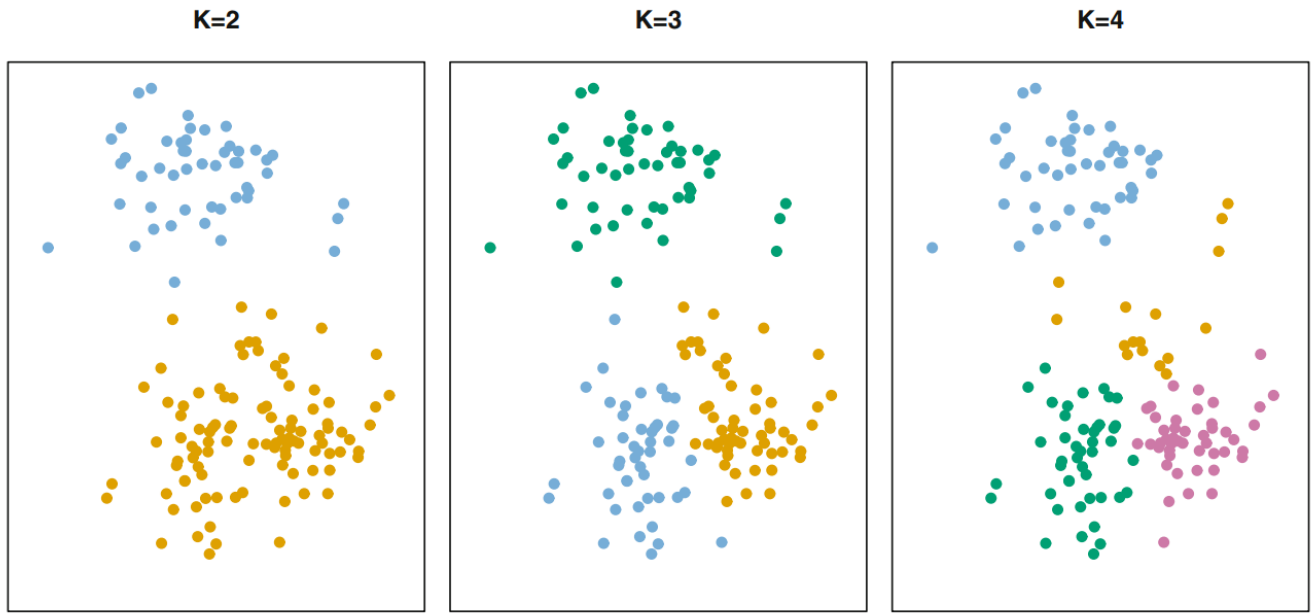
- In K-means clustering, we seek to partition the observations into a pre-specified number of clusters
- In hierarchical clustering, we do not know in advance how many clusters we want
  - A tree-like visual representation of the observations - dendrogram
  - View at once the clusterings obtained for each possible number of clusters



From Eamonn Keogh, "Measurement of similarity and clustering"

K-means Clustering

A simulated data set with 150 observations in 2-dimensional space. Panels show the results of applying K-means clustering with different values of K, the number of clusters.



Details of K-means Clustering

Let $C_1, \ldots, C_K$ denote sets containing the indices of the observations in each cluster. These sets satisfy two properties:

- $C_1 \cup C_2 \cup \ldots \cup C_K = \{1, \ldots, n\}$
  Each observation belongs to at least one of the K clusters
- $C_k \cap C_{k'} = \emptyset$ for all $k \neq k'$
  No observation belongs to more than one cluster

For instance, if the ith observation is in the kth cluster, then $i \in C_k$

Details of K-means Clustering: continued

- Idea: A good clustering is one for which the within-cluster variation is as small as possible
- Within-cluster variation (WCV): a measure of the amount by which the observations within a cluster differ from each other
- Hence we want to solve the problem

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} WCV(C_k) \right\} \tag{1}$$

Partition the observations into K clusters such that the total WCV summed over all K clusters is as small as possible

How to define within-cluster variation?

- Euclidean distance

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

where $|C_k|$ denotes the number of observations in the kth cluster

- The optimization problem that defines K-means clustering is

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}$$

- Note that

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

where $\bar{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$ is the mean of feature j in cluster C_k

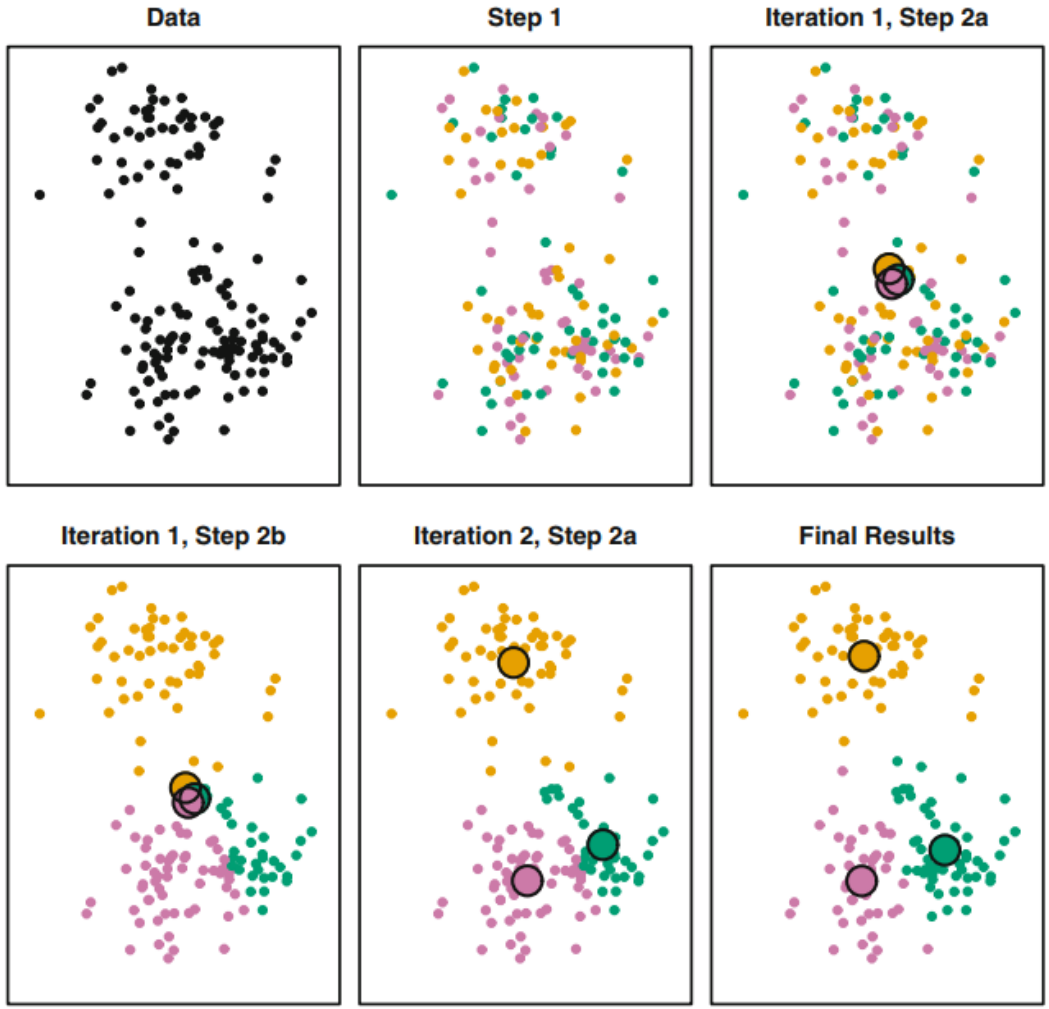
Clustering Algorithm

- Goal: find clusters so that within-cluster dissimilarity is minimized
- When $n = 19$, $K = 4$, the number of possible cluster assignment $\approx 10^{10}$
- Finding global optimal using enumeration is impossible
- Seek a good local optimum - iterative greedy descent

K-means Clustering Algorithm

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:
   - 2.1 For each of the K clusters, compute the cluster center, called centroid. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster
   - 2.2 Assign each observation to the cluster whose centroid is closest (where closest is defined using Euclidean distance)

Example


Data
Step 1
Iteration 1, Step 2a
Iteration 1, Step 2b
Iteration 2, Step 2a
Final Results

Properties of the Algorithm

- The algorithm is guaranteed to decrease the value of the objective function at each step
- However it is not guaranteed to give the global minimum

Example: Different Starting Values

K-means clustering performed six times on the data from previous figure

Strengths and Weaknesses of K-means

- Simple and fast: easy to understand and to implement
- Terminates at a local optimum - the global optimum is hard to find
- The algorithm is only applicable if the mean is defined
- The user needs to specify K
- Sensitive to outliers
- Sometimes sensitive to initial seeds
- The algorithm is not suitable for discovering clusters that are not hyper-ellipsoids

Hierarchical Clustering

- K-means clustering requires us to pre-specify the number of clusters
- Hierarchical clustering is an alternative approach which does not require that we commit to a particular choice of K
  - Agglomerative (bottom-up) algorithm: begin with each element as a separate cluster and merge them into successively larger clusters
  - Divisive (top-down) algorithm: begin with the whole set and proceed to divide it into successively smaller clusters

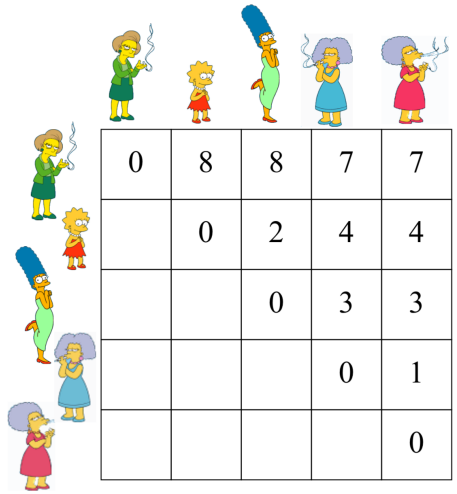
Hierarchical Clustering Algorithm (Agglomerative)

The approach in words:

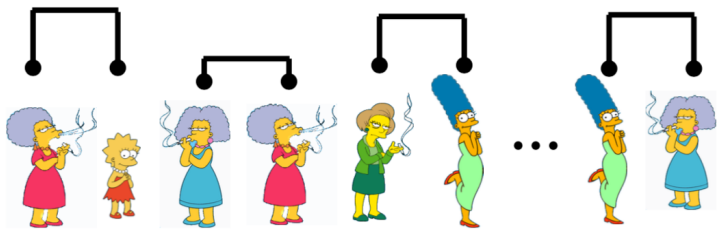
- Start with each point in its own cluster
- Identify the closest two clusters and merge them
- Repeat
- Ends when all points are in a single cluster

Most agglomerative algorithms possess monotonicity property: The dissimilarity between merged clusters is monotone increasing with the level of the merger

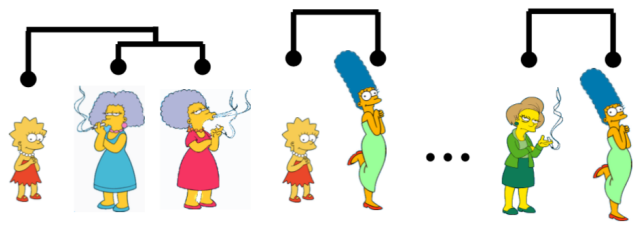
Hierarchical Clustering Algorithm (Agglomerative)



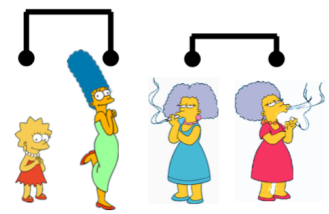
0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

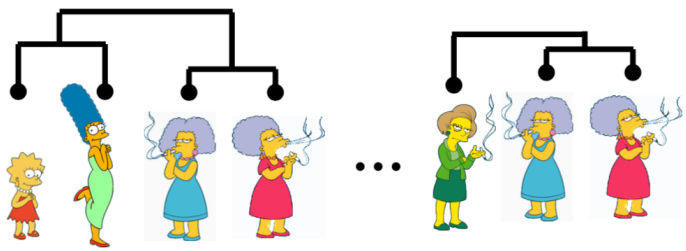


Choose the best

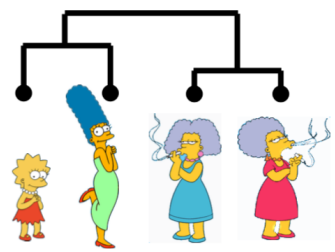


Choose the best



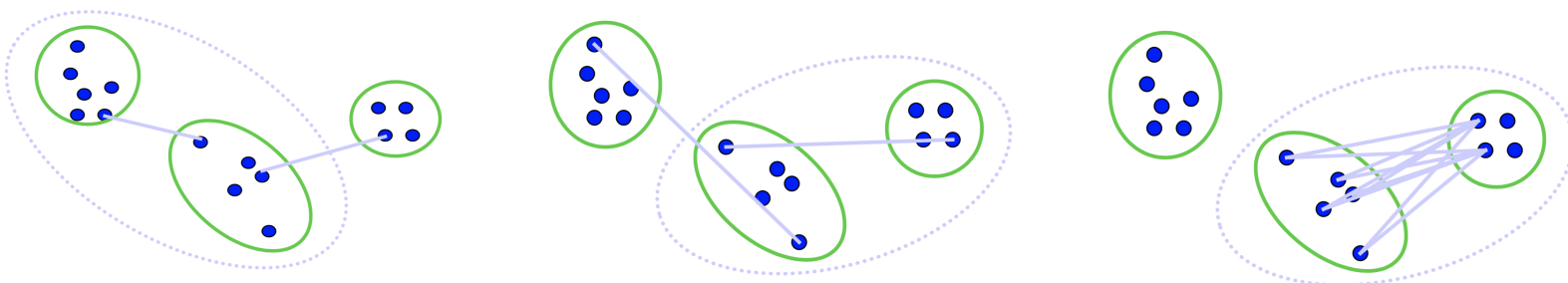


Choose the best

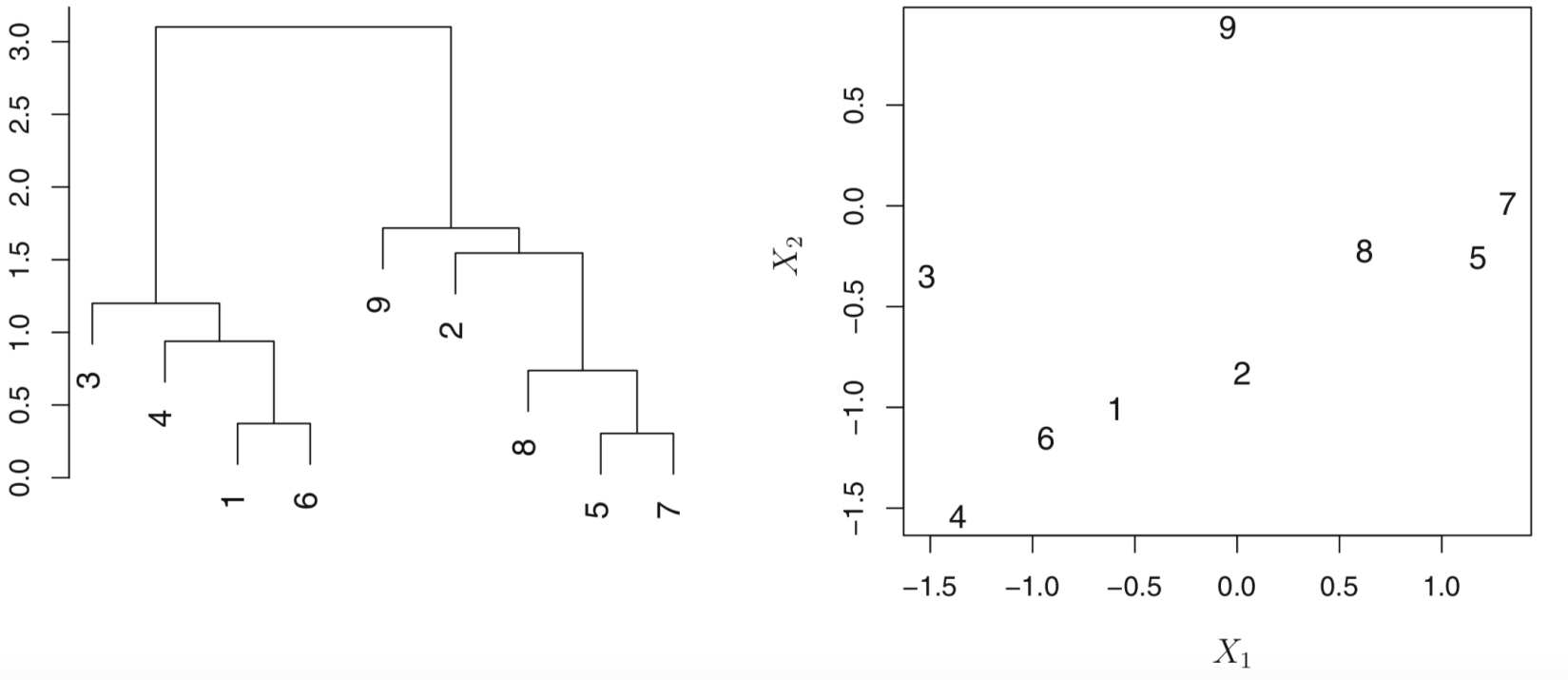


Common Ways to Measure Cluster Distance

- Single linkage: Minimal inter-cluster dissimilarity
  (Potentially long and skinny clusters)
- Complete linkage: Maximal inter-cluster dissimilarity
  (Compact clusters)
- Average linkage: Mean inter-cluster dissimilarity
  (Robust against noise)
- Centroid linkage: Dissimilarity between the centroids



Example



- We draw conclusions about the similarity of two observations based on the location on the **vertical** axis
- The height of each node is proportional to the value of the intergroup dissimilarity between its two daughters

Principal Components Analysis (PCA)

- PCA produces a low-dimensional representation of a dataset
- We want this transform to preserve the main structure in the feature space: looking for directions in the feature space along which the data exhibit an interesting trend
- PCA finds a sequence of linear combinations of the variables that have maximal variance, and are mutually uncorrelated
- Why PCA?
  - Produce derived variables for use in supervised learning (e.g., PCR)
  - Serves as a tool for data visualization

Principal Components Analysis: details

- The first principal component (PC) of $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

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

  that has the largest variance
- By normalized, we mean that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$
- Elements $\phi_{11}, \ldots, \phi_{p1}$ - loadings of the first PC
- The principal loading vector $\phi_1 = (\phi_{11}, \phi_{21}, \ldots, \phi_{p1})^T$
- We constrain the loadings so that their sum of squares is equal to one. Why?
- Scaling of the variables matters

Computation of Principal Components

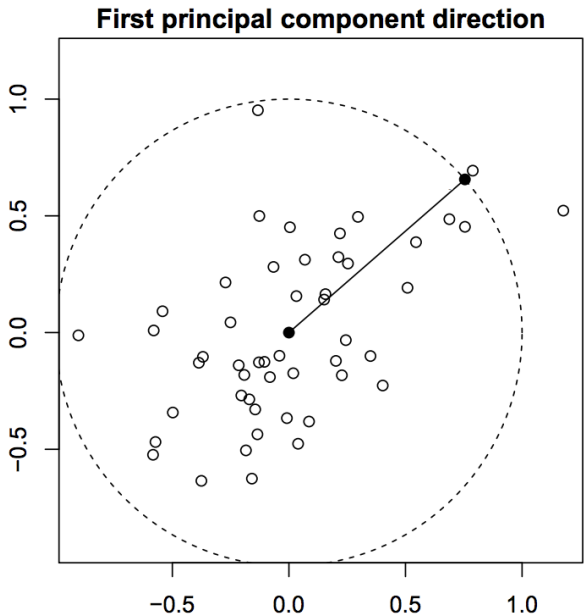
- Suppose we have a $n \times p$ data set $\boldsymbol{X}$
- Since we are only interested in variance, we assume that each of the variables in $\boldsymbol{X}$ has been centered to have mean zero
- We then look for the linear combination of the sample feature values of the form

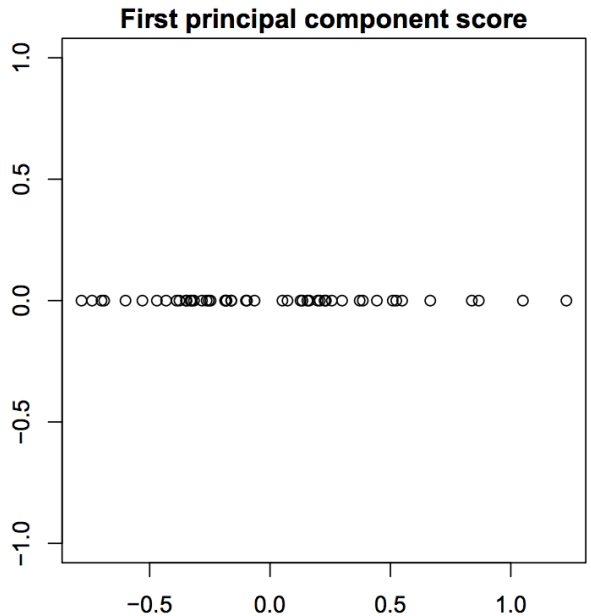
$$z_{i1} = \phi_{11} x_{i1} + \phi_{21} x_{i2} + \ldots + \phi_{p1} x_{ip} \tag{2}$$

for $i = 1, \ldots, n$ that has largest sample variance, subject to the constraint that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$

Geometry of PCA

- The loading vector ϕ_1 with elements $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most
- If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, z_{21}, \ldots, z_{n1}$
- Example: $p = 2$, $n = 50$





Further Principal Components

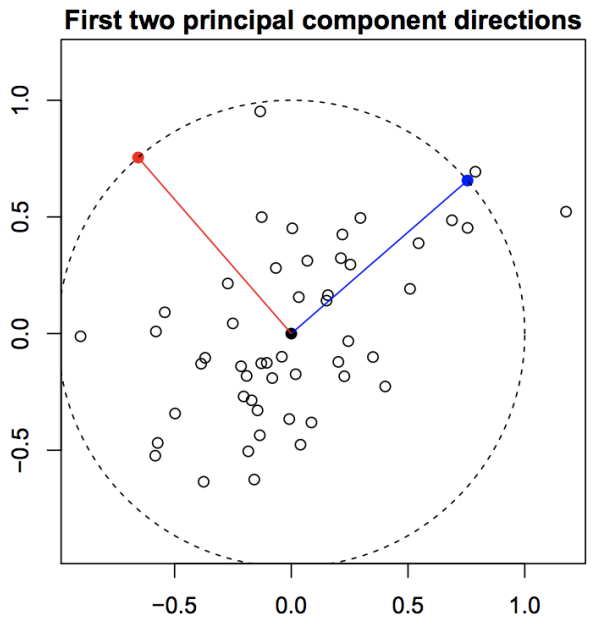
- The idea is to successively find **orthogonal** (perpendicular) directions of the highest variance
- The second PC is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are **uncorrelated** with Z_1
- The second PC scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

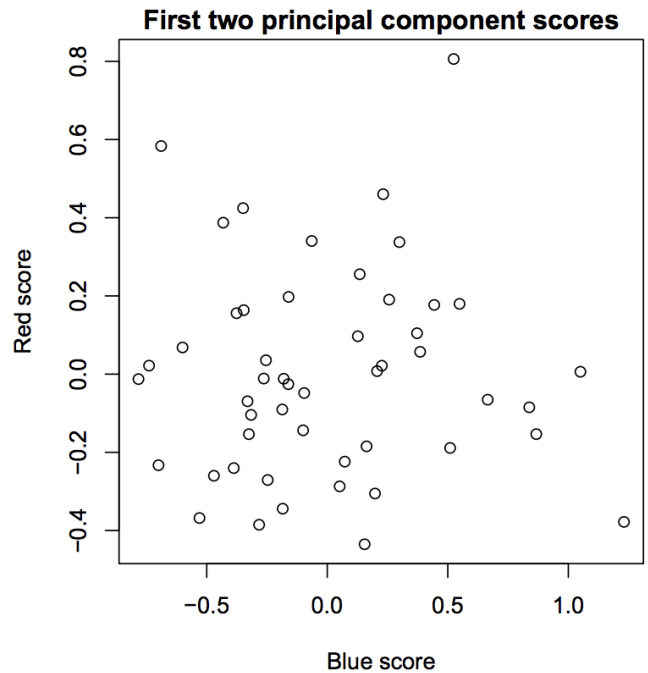
$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + \ldots + \phi_{p2} x_{ip},$$

  where ϕ_2 is the second PC loading vector, with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$
- Similarly, the kth PC is the linear combination of X that captures as much of the information as possible and is uncorrelated with the first $(k-1)$ PC

Further Principal Components: continued

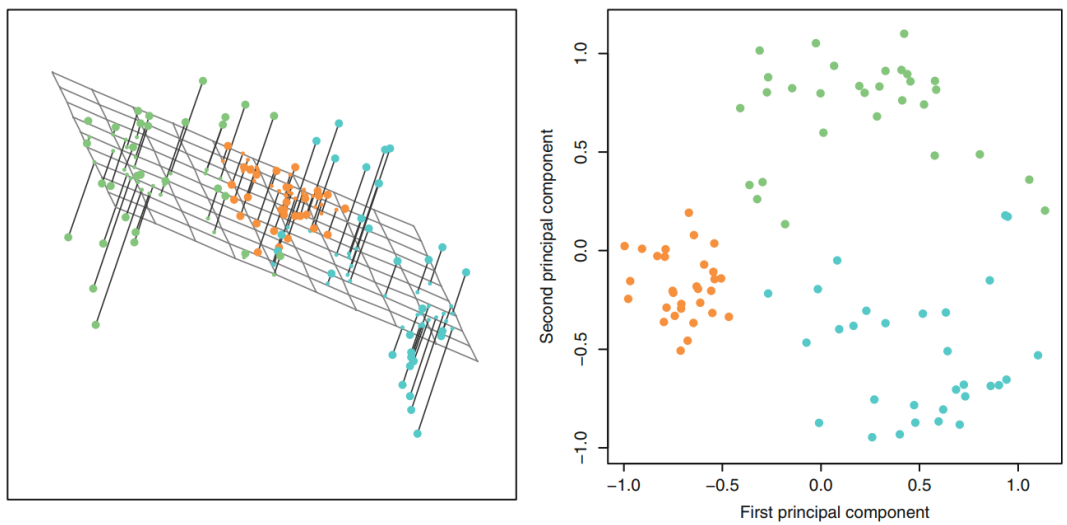
- Constraining Z_2 to be uncorrelated with Z_1 is equivalent to constraining the direction ϕ_2 to be orthogonal to the direction ϕ_1





PCA Finds the Hyperplane Closest to the Observations

- The first PC loading vector has a special property: it defines the line in p-dimensional space that is closest to the n observations
- The notion of PC as a dimension that are closest to the n observations extends beyond just the first principal component
- For instance, the first two PCs of a data set span the plane that is closest to the n observations



Proportion Variance Explained

- To understand the strength of each component, we are interested in knowing the proportion of variance explained (PVE) by each one
- The total variance present in a data set (assuming that the variables have been centered to have mean zero) is defined as

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

and the variance explained by kth principal component is

$$Var(Z_k) = \frac{1}{n} \sum_{i=1}^{n} z_{ik}^2$$

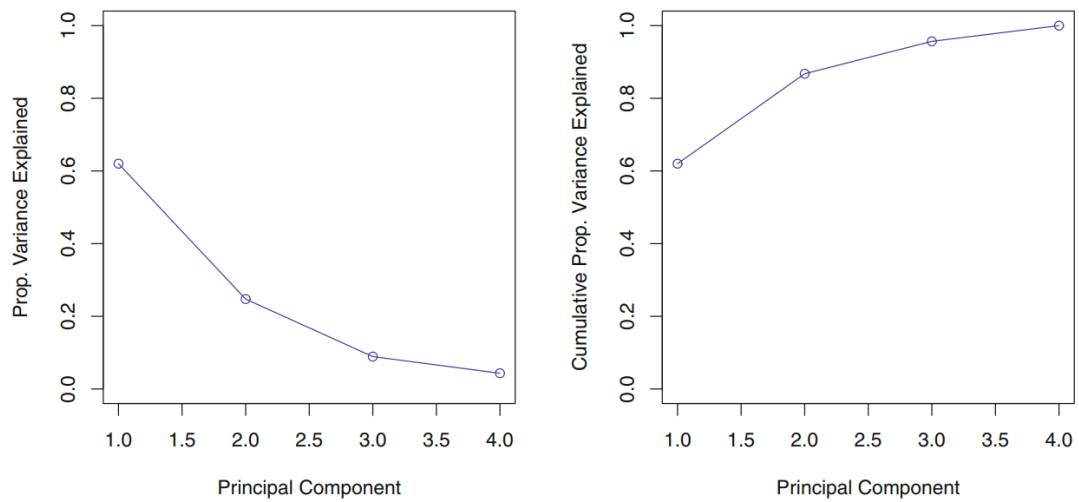
- When $n > p$, $\sum_{j=1}^{p} Var(X_j) = \sum_{k=1}^{p} Var(Z_k)$

Proportion Variance Explained: continued

- Therefore, the PVE of the kth principal component is given by the positive quantity between 0 and 1

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

- The PVEs sum to one
- We sometimes display the cumulative PVEs



Dimension Reduction via PC scores

- Dimension reduction via PCA can be achieved by taking the first k PC scores
- We can think of the first k PC scores as our new feature vectors
- Big saving if $k \ll p$
- How good are these features at capturing the structure of our old features?
  Look at the PVE as a function of k - scree plot

Dimension Reduction in Supervised Learning

- Principal component regression: dimension reduction + regression
- Let $Z_1, Z_2, \ldots, Z_M$ represent $M < p$ linear combinations of our original p predictors

$$Z_m = \sum_{j=1}^{p} \phi_{jm} X_j$$

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

- We can then fit the linear regression model

$$y_i = \theta_0 + \sum_{m=1}^{M} \theta_m z_{im} + \epsilon_i, i = 1, \ldots, n \tag{3}$$

using ordinary least squares

Principal Component Regression: Details

Notice that

$$\sum_{m=1}^{M} \theta_m z_{im} = \sum_{j=1}^{p} \beta_j x_{ij},$$

where

$$\beta_j = \sum_{m=1}^{M} \theta_m \phi_{jm} \tag{4}$$

- Model (3) is a special case of the original linear regression
- Dimension reduction serves to constrain the estimated β_j coefficients, since now they must take the form (4)
- Choice of M: cross-validaton