Lecture 22: Clustering Analysis

Hao Helen Zhang

Outlines

- Unsupervised Learning
  - Introduction
  - Various Learning Problems
- Clustering Analysis
  - Introduction
  - Optimal Clustering
- Various Clustering Algorithms
  - K-means Algorithm
  - K-medoids Algorithm
  - Hierarchical Clustering

What Is Unsupervised Learning

Learning patterns from data without a teacher.

- Data: $\{\mathbf{x}_i\}_{i=1}^n$
- No y_i's are available.

Various unsupervised learning problems:

- cluster analysis
- dimension reduction
- density estimation problems (useful for low-dimensional problems)

In contrast to supervised learning, there is no clear measure of success for unsupervised learning.

About Supervised Learning

Learning with a teacher: given a training set $\{\mathbf{x}_i, y_i\}_{i=1}^n$ to learn the relationship between $\mathbf{x}$ and y.

- regression, classification, ...
- Once a model $f(\mathbf{x})$ is obtained, one can use it for prediction $\hat{y} = f(\mathbf{x})$.
- Measure of success: accuracy of prediction. For example,

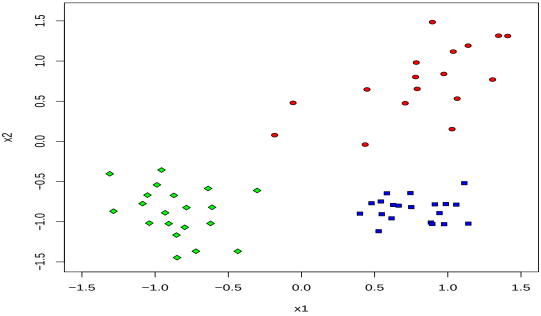
$$L(y, \hat{y}) = (y - \hat{y})^2.$$

"student": the prediction value $\hat{y}_i$
"teacher": the correct answer and/or an error associated with the student's answer.

Cluster Analysis

Goal: to find multiple regions of the **X**-space that contains modes of $P(\mathbf{x})$.

- Can we represent $P(\mathbf{x})$ by a mixture of simpler densities representing distinct types or classes of observations?



Dimension Reduction

Goal: to describe the association among variables as a function of a smaller set of "latent" variables.

- Principle component analysis (PCA)
- Independent component analysis (ICA)
  - used for blind source separation
  - Assume $\mathbf{X} = A\mathbf{S}$, where $\mathbf{S}$ is p-vector containing independent components. The goal is to identify A and $\mathbf{S}$.
- multidimensional scaling (MDS)
- self-organizing maps (SOP)
- principal curves
- ....

Density Estimation

Goal: estimate the density distribution of $X_1, \cdots, X_p$.

- histogram
- Gaussian mixture provides a crude model
- kernel density estimation, smoothing splines

Challenges:

- Density estimation can be infeasible for high dimensional problems.

Cluster Analysis Introduction

Goal: Group or segment the dataset (a collection of objects) into subsets, such that those within each subset are more closely related to one another than those assigned to different subsets.

- Each subset is called a *cluster*

Two types of clustering: flat and hierarchical clustering:

- *flat clustering* divides the dataset into k clusters
- *hierarchical clustering* is to arrange the clusters into a natural hierarchy.
  - involves successively grouping the clusters themselves
  - At each level of the hierarchy, clusters within the same group are more similar to each other than those in different groups.

Proximity and Dissimilarity Matrices

Clustering results are crucially dependent on the measure of similarity or distance between the "points" to be clustered.

- One can use either a similarity or dissimilarity matrix.
- Similarities can be converted to dissimilarities using a monotone-decreasing function.

Given two points (objects), $\mathbf{x}_i$ and $\mathbf{x}_{i'}$, there are two levels of dissimilarity:

- Attribute dissimilarity: $d_j(\mathbf{x}_{ij}, \mathbf{x}_{i'j})$ quantifies the dissimilarity in their jth attribute.
- Object dissimilarity: $d(\mathbf{x}_i, \mathbf{x}_{i'})$ quantifies the overall dissimilarity between two points (objects).

Object Dissimilarities

A common choice of attribute dissimilarity is:

$$Squared\quad\text{distance}:\quad d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2.$$

- The object dissimilarity can be quantified by the sum of d_j's

$$D(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} d_j(x_{ij}, x_{i'j}).$$

  or a weighted sum

$$D(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} w_j d_j(x_{ij}, x_{i'j}) \quad \text{where} \sum_{j=1}^{p} w_j = 1.$$

- General distance-based dissimilarity:

$$d(\mathbf{x}_i, \mathbf{x}_{i'}) = L(\|\mathbf{x}_i - \mathbf{x}_{i'}\|).$$

Correlation-Based Dissimilarity

Alternatively, the following correlation can be used to measure the dissimilarity between subjects:

$$\rho(\mathbf{x}_i, \mathbf{x}_{i'}) = \frac{\sum_{j=1}^{p}(x_{ij} - \bar{x}_i)(x_{i'j} - \bar{x}_{i'})}{\sqrt{\sum_{j=1}^{p}(x_{ij} - \bar{x}_i)^2 \sum_{j=1}^{p}(x_{i'j} - \bar{x}_{i'})^2}},$$

where $\bar{x}_i = \sum_{j=1}^{p} x_{ij}/p$ is the average of variables (not observations).

- If inputs are standardized, then

$$\sum_{j=1}^{p}(x_{ij} - x_{i'j})^2 \propto 2(1 - \rho(\mathbf{x}_i, \mathbf{x}_{i'})).$$

- In the case, clustering based on correlation (similarity) is equivalent to that based on squared distance (dissimilarity).

Proximity Matrix

The proximity (or alikeness or affinity) matrix $D = (d_{ik})$:

- The *ik*-th element d_{ik} measuring the proximity between the *i*-th and the *k*th objects (or observations).
- D is of size $n \times n$:
- D is typically symmetric.
- If D is not symmetric, we can use $(D + D')/2$ can be applied.

Criterion for Optimal Clustering

- Each observation is uniquely labeled by an integer $i \in \{1, 2, \ldots, n\}$.
- k clusters: $k \in \{1, \ldots, K\}$.
- Define $k = C(i)$: the ith observation is assigned to the k-th cluster.

The optimal cluster C^* should satisfy that

- The total dissimilarity within clusters is minimized, i.e., we want to minimize $W(C)$:

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} d(\mathbf{x}_i, \mathbf{x}_{i'}).$$

Within-Class and Between-Class Dissimilarities

- Total dissimilarity:

$$T = \frac{1}{2}\sum_{i=1}^{n}\sum_{i'=1}^{n} d_{ii'} = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\Big(\sum_{C(i')=k} d_{ii'} + \sum_{C(i')\neq k} d_{ii'}\Big).$$

- Between-cluster dissimilarity:

$$B(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')\neq k} d_{ii'}.$$

- $W(C) = T - B(C)$.

Minimizing $W(C)$ is equivalent to maximizing $B(C)$.

Combinatorial Algorithms

- One needs to minimize W over all possible assignments of n points to K clusters.
- The number of distinct assignments is

$$S(n,K) = \frac{1}{K!}\sum_{k=1}^{K}(-1)^{K-k}\binom{K}{k}k^{n}.$$

- It is not feasible for large n and K.
- It calls for more efficient algorithms: may not be optimal but a reasonably good suboptimal partition.

K-means Method

One of the most popular iterative descent clustering methods.

- Works for the case when all variables are quantitative.
- Dissimilarity measure: the squared distance

$$d(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = ||\mathbf{x}_i - \mathbf{x}_{i'}||^2.$$

- The within-class scatter can be reduced to

$$W(C) = \sum_{k=1}^{K} n_k \sum_{C(i)=k} ||\mathbf{x}_i - \bar{\mathbf{x}}_k||^2,$$

where $\bar{\mathbf{x}}_k$ is the mean for cluster k and n_k is size of cluster k.

Optimization for K-means Clustering

Optimization Problem:

$$C^* = \min_{C} \sum_{k=1}^{K} n_k \sum_{C(i)=k} ||\mathbf{x}_i - \bar{\mathbf{x}}_k||^2,$$

where

$$\bar{\mathbf{x}}_S = \text{argmin}_m \sum_{i \in S} ||\mathbf{x}_i - m||^2.$$

- This can be solved using an iterative descent algorithm.

We actually need to solve the enlarged optimization problem:

$$\min_{C, \{m_k\}_1^K} \sum_{k=1}^{K} n_k \sum_{C(i)=k} ||\mathbf{x}_i - m_k||^2.$$

K-means Algorithm

Starts with the guesses for the K clusters:

1. (Clustering Step) Given K centroids $\{m_1, \ldots, m_k\}$, solve C by assigning each observation to the closest (current) cluster mean
$$C(i) = \text{argmin}_k ||\mathbf{x}_i - m_k||^2.$$
2. (Minimization Step)) For a given C, update center $\{m_1, \ldots, m_k\}$; that is the mean of k clusters.
3. Iterate steps 1 and 2 until the assignments do not change.

This can be regarded as a top-down procedure for clustering analysis. (see $\text{http://en.wikipedia.org/wiki/K-means}_{Clustering}$)

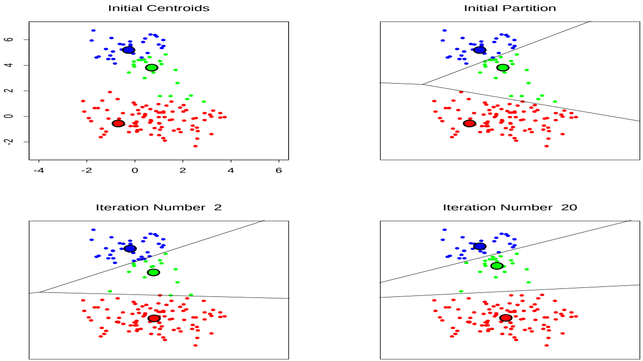


Figure 14.6: *Successive iterations of the K-means clustering algorithm for the simulated data of Figure 14.4.*

About K-means Algorithm

K-means often produces good results.

- The K-means algorithm is guaranteed to converge, since each of steps 1 and 2 reduced $W(C)$
  - The time complexity of K-means is $O(tKn)$, where t is the number of iterations.
- K-means finds a local optimum and may miss the global optimum.
- Different starting values lead to different clustering results.
  - should start the algorithm with many different random choices for the starting means, and choose the solution having smallest value of the $W(C)$.
- K-means does not work on categorical data; does not handle outliers well.

How to Choose K

In the K-means algorithm, one needs to specify K in advance.

- K is a tuning parameter. An inappropriate choice of K may yield poor results.
- Intuitively, one can try different K values and evaluate $W(C)$ on a test set.
  - However, $W(C)$ generally decreases with increasing K, since a large number of centers tend to fill the feature space densely and thus will be close to all data points.
  - So CV or test set does not work here.
- A heuristic approach: locating a kink in $W(C)$ for the optimal K^*.
- Use the *Gap* statistic (Tibshirani et al. 2001)

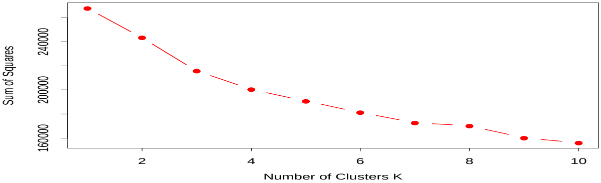


Figure 14.8: *Total within cluster sum of squares for K-means clustering applied to the human tumor microarray data.*

Motivations of K-medoids Algorithm

The K-means algorithm:

- assumes squared Euclidean distance in the minimization step
- requires all the inputs to be quantitative type

The K-medoids algorithm generalizes the K-means algorithm in the following ways:

- use arbitrarily defined similarities $D(\mathbf{x}_i, \mathbf{x}_{i'})$
- can be applied to any data described only by proximity matrices

Features of K-medoids Algorithm

Main ideas:

- Centers for each cluster are restricted to be one of the observations assigned to the cluster
  - there is no need to explicitly computer cluster centers
  - the computation cost is $O(n_k^2)$. (For K-means, the computation cost is $O(n_k)$.)
- Given a set of cluster centers $\{i_1, \cdots, i_k\}$, obtain the new assignments

  $$C(i) = \arg \min_{1 \le k \le K} d_{ii_k}.$$

  Requires the computation proportional to Kn.
- K-medoids is far more computationally intensive than K-means.

K-medoids Algorithm

Starts with the guesses for the K clusters:

1. (Clustering Step) Given $\{m_1, \ldots, m_k\}$, minimize the total error by assigning each observation to the closest center:

$$C(i) = \arg \min_{1 \le k \le K} D(\mathbf{x}_i, m_k).$$

2. (Minimization Step) For a given C, find the observation in the cluster k by minimizing the total distance to other points in the cluster k

$$i_k^* = \arg \min_{i:C(i)=k} \sum_{C(i')=k} D(\mathbf{x}_i, \mathbf{x}_{i'}).$$

3. Iterate steps 1 and 2 until the assignments do not change.

A heuristic search strategy to solve $\min_{C,\{i_k\}_1^K} \sum_{k=1}^K \sum_{C(i)=k} d_{ii_k}$.

Hierarchical Clustering

K-means does not give a linear ordering of objects within a cluster. This motivates the idea of hierarchical clustering:

- Produce hierarchical representations: the clusters at each level of the hierarchy are created by merging clusters at the next lower level.
- At the lowest level, each cluster contains a single observation.
- At the highest level there is only one cluster containing all observations.

It is very informative and highly interpretable to use the *dendrogram* to display the clustering result.

- Cutting the dendrogram horizontally at a particular height partitions the data into disjoint clusters represented by the vertical lines that intersect it.

Elements of Statistical Learning ©Hastie, Tibshirani & Friedman 2001 Chapter 14

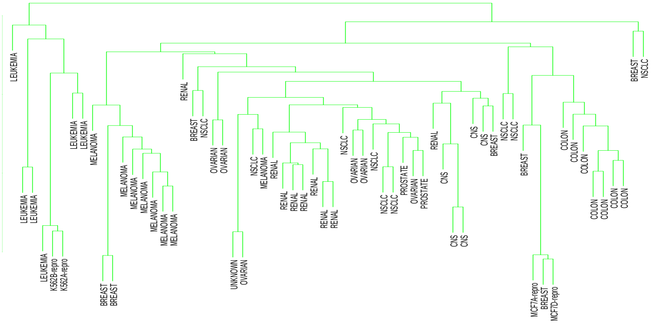


Figure 14.12: *Dendrogram from agglomerative hierarchical clustering with average linkage to the human tumor microarray data.*

Agglomerative Clustering

Two paradigms:

- *agglomerative* (bottom-up)
- *divisive* (top-down).

Key steps in the agglomerative clustering:

- Begin with every observation representing a singleton cluster.
- At each step, merge two "closest" clusters into one cluster and reduce the number of clusters by one.
- Need a measure of dissimilarity between two clusters.

Dissimilarity between Two Groups

In hierarchical clustering, it is important to define the dissimilarity between two clusters (group) G and H:

$d(G, H)$ is a function of the set of pairwise dissimilarities $d_{ii'}$, where $x_i \in G$ and $x_{i'} \in H$.

- Single linkage: the closest pair

$$d_{SL}(G, H) = \min_{i \in G, i' \in H} d_{ii'}.$$

- Complete linkage: the furthest pair

$$d_{CL}(G, H) = \max_{i \in G, i' \in H} d_{ii'}.$$

- Group Average: average dissimilarity

$$d_{GA}(G, H) = \frac{1}{n_G n_H} \sum_{i \in G} \sum_{i' \in A} d_{ii'}.$$

Group average clustering has a statistical consistency property

Elements of Statistical Learning ©Hastie, Tibshirani & Friedman 2001 Chapter 14

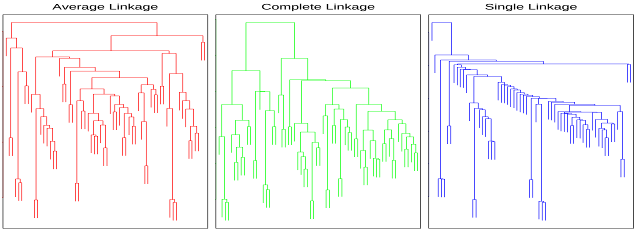


Figure 14.13: *Dendrograms from agglomerative hierarchical clustering of human tumor microarray data.*

Human Tumor Microarray Data

An example of high-dimensional clustering

- The data are $6,830 \times 64$ matrix of real numbers
- Row: the expression measurement for a gene.
- Column: a sample (labeled as “breast cancer” or “melaoma”)

In the next plot,

- we have arranged the genes (rows) and samples (columns) in ordering derived from hierarchical clustering
- the subtree with the tighter cluster is placed to the left (or bottom).

This two-way rearrangement of picture is more informative than displaying the data with randomly ordered rows and columns

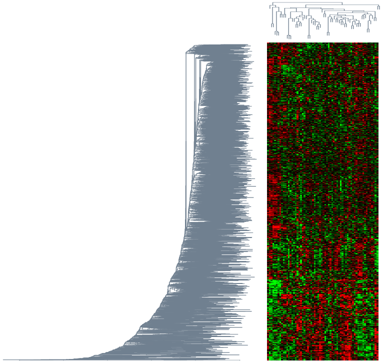


Figure 14.14: *DNA microarray data: average linka*
hierarchical clustering has been applied independent
to the rows (genes) and columns (samples), determi
ing the ordering of the rows and columns (see text
The colors range from bright green (negative, undere
pressed) to bright red (positive, overexpressed).

R functions

- K-means:
  - package stat function kmeans
- Hierarchical Clustering:
  - package stat function hclust.