STAT 5630, Fall 2019

Clustering

Xiwei Tang, Ph.D. <xt4yj@virginia.edu>

University of Virginia
October 29, 2019

Unsupervised Learning

Unsupervised Learning

- Learn without the response variable Y, only $\{x_i\}_{i=1}^n$
- Goal: learn patterns in X
- Examples
  - Estimate the density, covariance, graph (network), etc. of X — could be difficult high-dimensional problems
  - Dimension reduction: identify low-dimensional manifolds within the feature space $\mathcal{X}$ that represent high data density.
  - Cluster analysis: try to identify multiple regions of the feature space that contains modes of density.
- Oftentimes, there is no clear measure of success for unsupervised learning.

Cluster Analysis

Cluster Analysis

- Group the dataset into subsets so that those within each subset are more closed related (similar) to each other than those objects assigned to other subsets. Each subset is called a cluster
- Flat clustering vs. hierarchical clustering: flat clustering divides the dataset into k cluster; and hierarchical clustering is to arrange the clusters into a natural hierarchy.
- Clustering results are crucially dependent on the measure of similarity (or distance) between the “points” to be clustered.

Distance Matrices

- Dissimilarity Matrix is distance matrices that gives small values for a pair of similar (closer) subjects.
- The distance between two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$ is given by

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

- The most commonly used measurement (for continuous features) is the Euclidian distance:

$$d_j(x_{ij}, x_{i'j}) = (x_{ij} - x_{i'j})^2,$$

hence $D(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\|_2^2$.

Distance Measures

- The Euclidian distance can be related to the correlation (angle) between the two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$.
- In high-dimensional settings, we sometimes center and standardize (to mean 0 and sd 1) the vectors $\boldsymbol{x}_i$ for each subject.
- Then the correlation defined as

$$\rho(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \frac{\sum_j (x_{ij} - \overline{x}_i)(x_{i'j} - \overline{x}_{i'})}{\sqrt{\sum_j (x_{ij} - \overline{x}_i)^2 \sum_j (x_{i'j} - \overline{x}_{i'})^2}}$$

  is related to the distance measure through the relationship

$$D(x_{ij}, x_{i'j}) = 2(1 - \rho(\boldsymbol{x}_i, \boldsymbol{x}_{i'}))$$

- Note: this standardization is only useful in high-dimensional settings.

Clustering Algorithms

- We want to form K clusters, indexed by $k \in \{1, \ldots, K\}$.
- Let $C(i)$: denote the cluster index assigned to the ith observation.
- Goal: search for the function $C : \{1, \ldots, n\} \to \{1, \ldots, K\}$ to minimize the overall within cluster distance:

$$W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i), C(i')=k} d(x_i, x_{i'}).$$

Clustering Algorithms

- Similarly, we could maximize the between cluster distance

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

- However, the total distance can be broke down into

$$T = \frac{1}{2} \sum_{i=1}^{n} \sum_{i'=1}^{n} d_{ii'} = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \left[\sum_{C(i')=k} d_{ii'} + \sum_{C(i')\neq k} d_{ii'} \right]$$
$$= W(C) + B(C)$$

- The total distance is fixed for a given set of data.

Minimizing $W(C) \iff$ maximizing $B(C)$

Clustering Algorithms

- Given a specific distance measure $d(\cdot, \cdot)$, several algorithms can be used to find the clusters
  - Combinatorial algorithms
  - K-means clustering
  - Hierarchical clustering

Combinatorial Algorithms

- For small n and K, we could minimize W by brute-force search.
- However, this is not feasible for large n and K, since the number of distinct assignments can be extremely large:

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

- For example $S(10, 4) = 34,105$; $S(19, 4) \approx 10^{10}$.
- It calls for more efficient algorithms: may not be optimal but a reasonably good suboptimal partition.

K-means Clustering

- Idea: an iterative greedy descent algorithm
  - Initialization: A random partition of K clusters is specified.
  - Iterative step: Update the cluster registration towards a direction that the within cluster distance is reduced.
  - Stopping rule: When no improvement can be reached, terminate the algorithm
- Convergence is guaranteed, but not necessarily to global optima.

K-means Clustering

- Consider an enlarged optimization problem:

$$\min_{C,\{m_k\}_{k=1}^K} \sum_{k=1}^{K} \sum_{C(i)=k} \|x_i - m_k\|^2$$

- This problem is NP-hard for ≥ 2 dimensions.
- Instead, consider an algorithm that alternatively update the two components:
  - C, the cluster assignments
  - $\{m_k\}_{k=1}^K$: the cluster means
- We will do an iterative update by:
  1) Fixing C, find the best $\{m_k\}_{k=1}^K$
  2) Fixing $\{m_k\}_{k=1}^K$, find the best C

K-means Clustering

- Fixing C, we know the cluster label of each subject. For any set $\{i : C(i) = k\}$, finding the mean is

$$m_k = \arg\min_m \sum_{C(i)=k} \|x_i - m\|^2.$$

  This is simply finding the mean of all observations with $C(i) = k$.
- Fixing the cluster means $\{m_k\}_{k=1}^K$, to find the new cluster assignments, we simply recalculate the distance from an observation to each of the cluster mean.

$$C(i) = \arg\min_k \; d(x_i, \overline{x}_k)$$

K-means Clustering

- A K-means Clustering algorithm:
  1) Random split the dataset into K different subsets. Assign each subsets a cluster label. Then iterate between 2) and 3).
  2) Given cluster assignments C, calculate the cluster mean vectors $m_1, \ldots, m_K$.
  3) Given a current set of means $\{m_1, \ldots, m_K\}$, assign each observation to the closest current cluster mean.
- Note: We usually initiate the cluster labels randomly. However, this algorithm does not guarantee to converge to the global minimizer.
- The algorithm still has a descent property, which leads to a local minimizer.

Alternative Version

- K-medoids is an alternative version of K-means:
- Replace the second step by searching for the observation that minimizes the distance to all others in the cluster

$$i_k^* = \arg\min_{i:C(i)=k} \sum_{C(i')=k} D(x_i, x_{i'})$$

- Use $x_{i_k^*}$ as the “center” of cluster k.

Application

- An application of K-means: image segmentation (example taken from Prof. Domeniconi, GMU)
- Goal: partition an image into regions with homogeneous visual appearance (which could correspond to objects or parts of objects). This could save storage space.
- Image representation: each pixel is represented as a three dimensional point in RGB space, where

  R = intensity of red

  G = intensity of green

  B = intensity of blue

Image segmentation

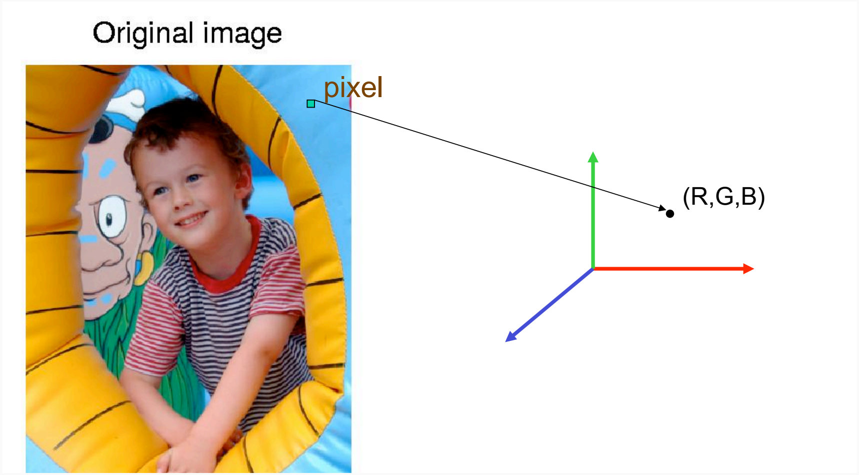


Image segmentation

Original image

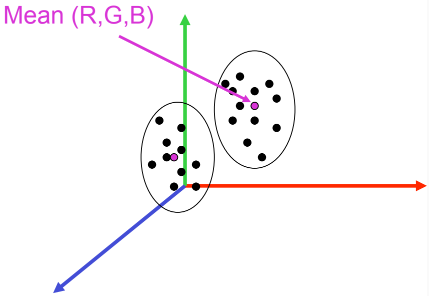
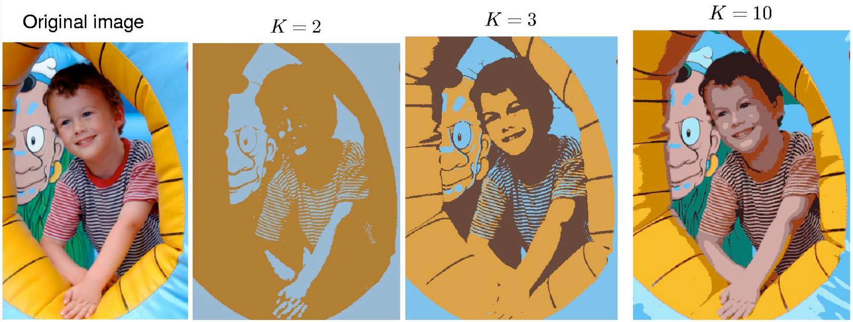


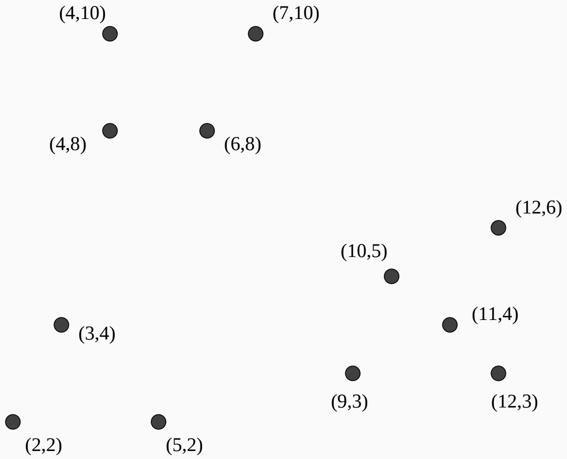
Image segmentation



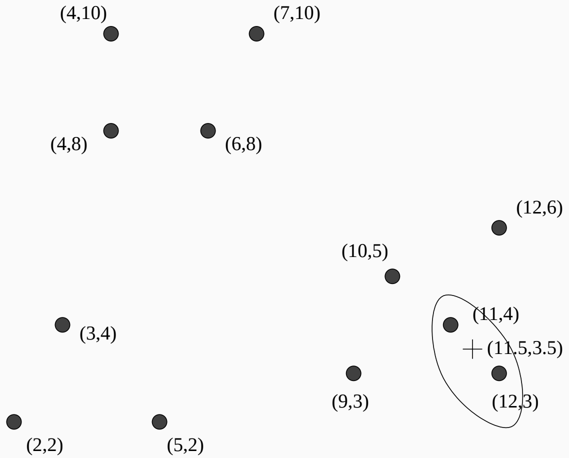
Hierarchical Clustering

- Choosing the number of clusters K can be difficult
- A hierarchical representation which
    - at the lowest level, each cluster contains a single observation.
    - at the highest level there is only one cluster containing all observations.
- Use dendrogram to display the clustering result.
- Two paradigms: agglomerative (bottom-up) and divisive (top-down).
- An example taken from Prof. Ullman, Stanford

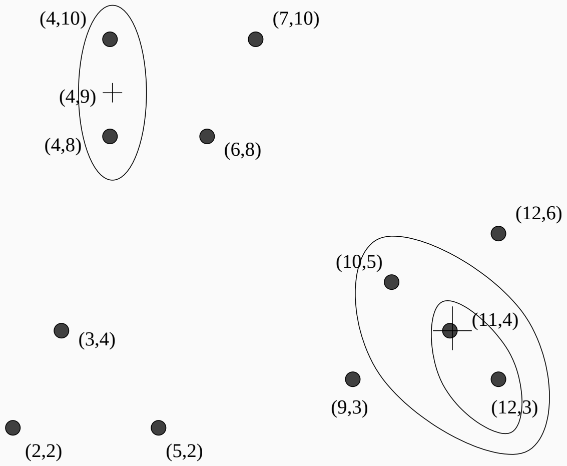
Example: Hierarchical Clustering



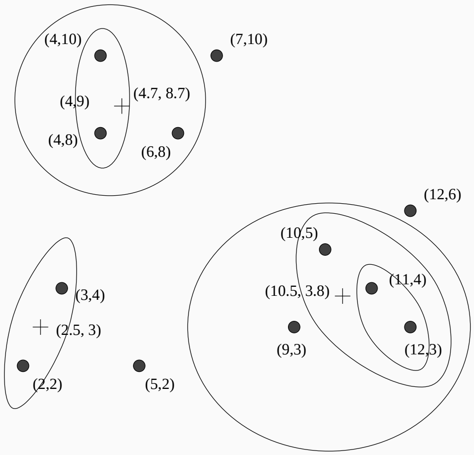
Example: Hierarchical Clustering



Example: Hierarchical Clustering

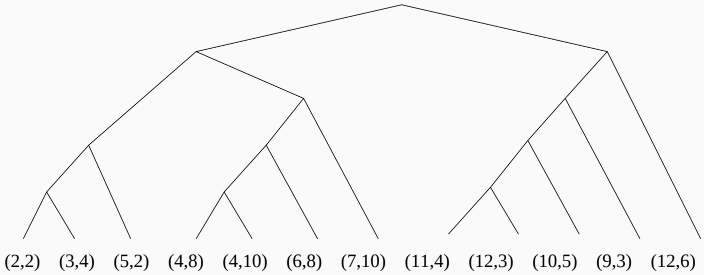


Example: Hierarchical Clustering



after several steps

Example: Hierarchical Clustering



A tree representation

Agglomerative Clustering (bottom-up)

- 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 clusters G and H: $d(G, H)$. Different choices:
  - Single linkage: the closest pair $d(G, H) = \min i \in G, i' \in H d_{ii'}$
  - Complete linkage: the furthest pair $d(G, H) = \max i \in G, i' \in H d_{ii'}$
  - Group Average: average dissimilarity $d(G, H) = \frac{1}{n_G n_H} \sum_{i \in G} \sum_{i' \in H} d_{ii'}$

Remark

- How many clusters we should choose?
- Sparse clustering for high-dimensional data?
- R implementation (base package):
    - kmeans
    - hclust