CLUSTERING

Chapter 10

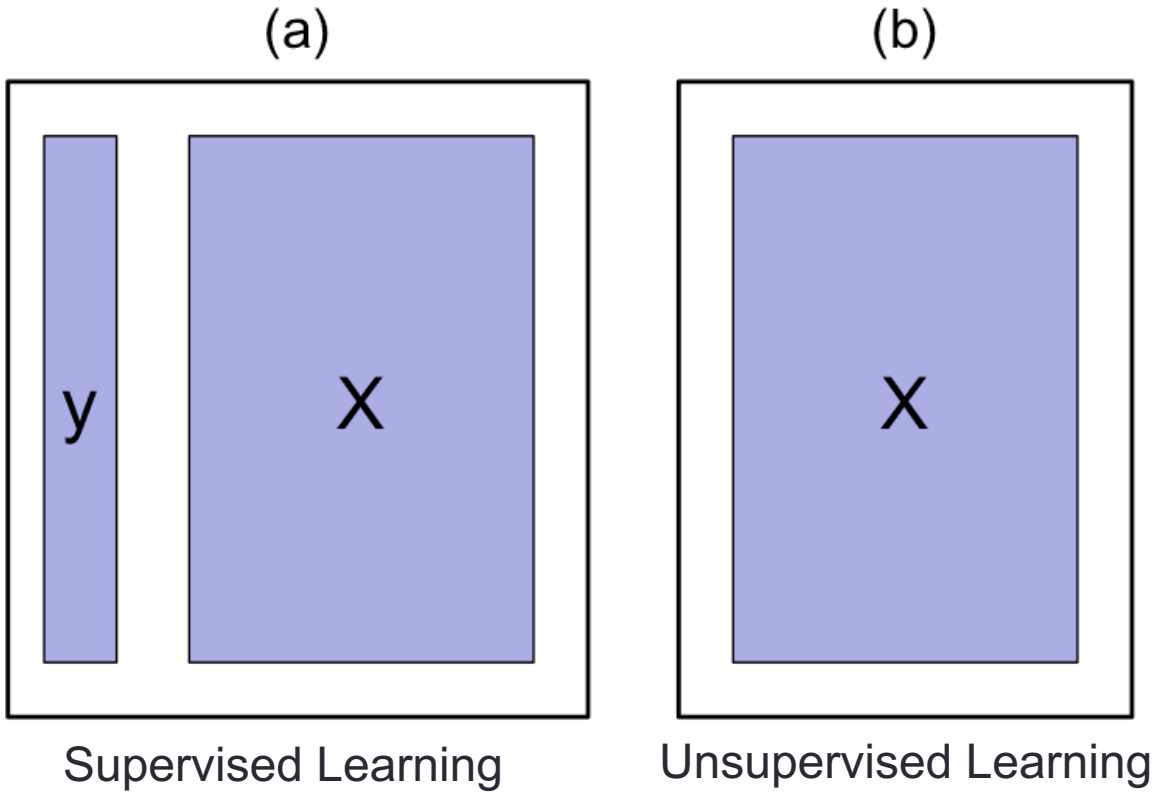
Outline

- What is Clustering?
- K-Means Clustering
- Hierarchical Clustering

WHAT IS CLUSTERING?

Supervised vs. Unsupervised Learning

- Supervised Learning: both X and Y are known
- Unsupervised Learning: only X



Clustering

- Clustering refers to a set of techniques for finding subgroups, or clusters, in a data set
  - *Be careful not to use the word "class" instead of cluster*

- Good clustering: when the observations within a group are similar but observations in differnt groups are very different

- For example, suppose we collect *p* measurements on each of *n* breast cancer patients. There may be different unknown types of cancer which we could discover by clustering the data

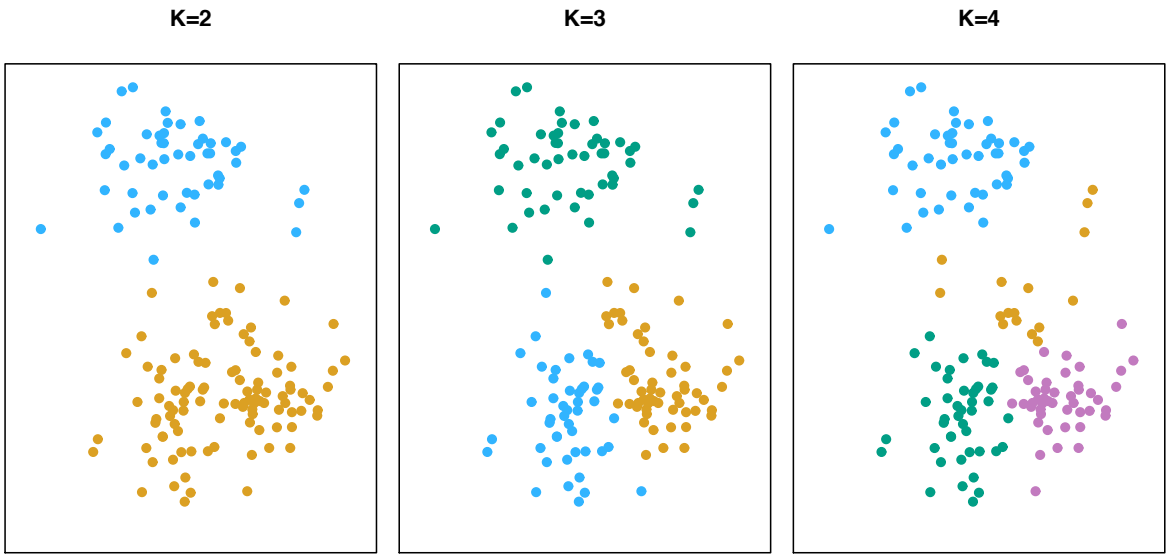
Different Clustering Methods

- There are many different types of clustering methods
- We will concentrate on two of the most commonly used approaches
  - K-Means Clustering
  - Hierarchical Clustering
- The objective is to have a
  - minimal *intra-cluster* - “within-cluster-variation”, i.e. the elements within a cluster should be as similar as possible
  - maximum *inter-cluster* – “center-to-center” distance, i.e. the cluster centers should be as far apart as possible

K-MEANS CLUSTERING

K-Means Clustering

- To perform K-means clustering, one must first specify the desired number of clusters K
- Then the K-means algorithm will assign each observation to exactly one of the K clusters

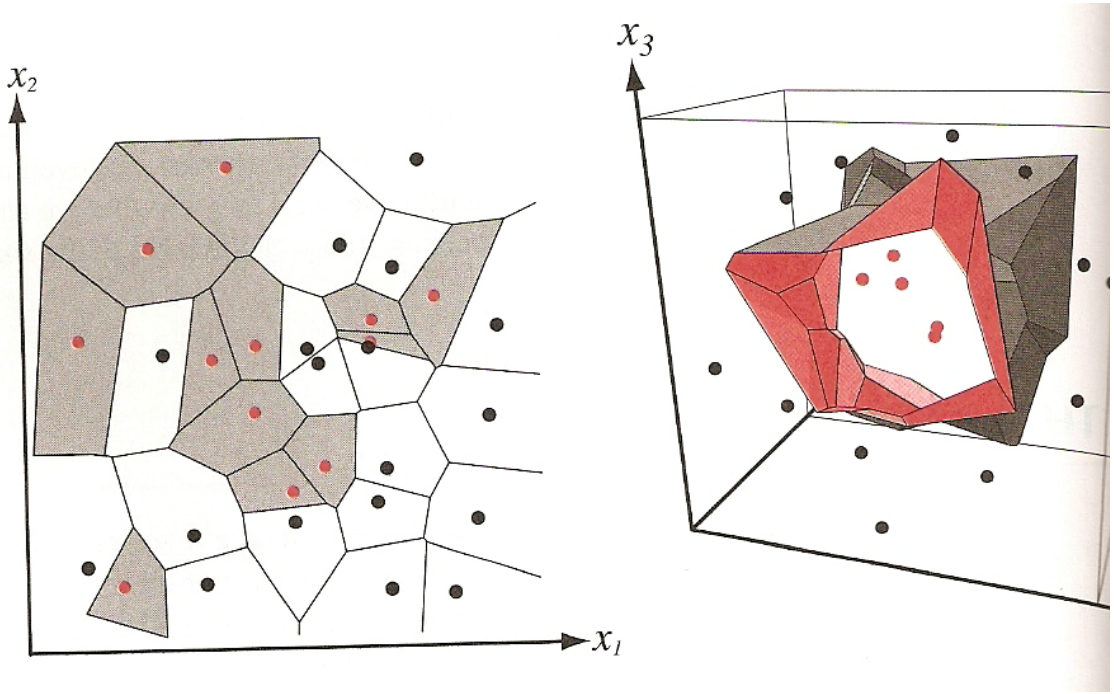


How does K-Means work?

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

$$C_1, \ldots, C_K$$

  - Each observation belongs to **one** of the K clusters
- K-Means results in a Voronoi Tessellation of the input space in $\Re^n$
  - A tessellation is a tiling/segmenting of the input space
  - Each segment/region is a Voronoi Cell and indicates which part of the input space "belongs" to which cluster center



K-Means Clustering Algorithm - Book

- Initial Step: Randomly assign each observation to one of K clusters such that

$$\forall i \in Obs, \exists k \in Clusters \mid i \in C_k \quad \text{(every observation belongs to a cluster)}$$

$$\forall l, m \in Clusters, l \neq m \rightarrow C_j \cap C_m = \varnothing \quad \text{(clusters are mutually exclusive)}$$

- Iterate until the cluster assignments stop changing:
  - For each of the K clusters, compute the cluster centroid. The kth cluster centroid is the mean of the observations assigned to the kth cluster

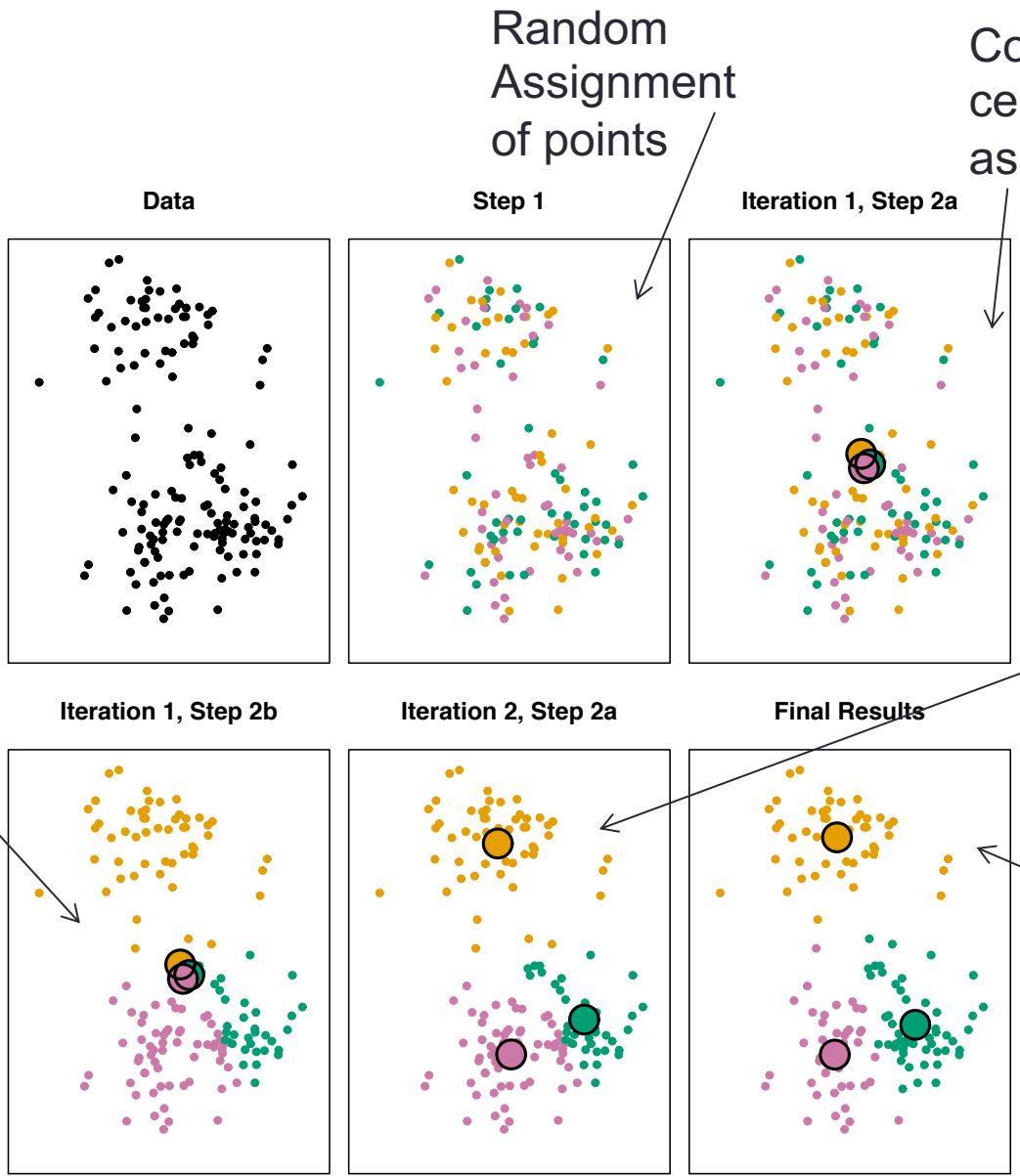
$$\forall k, \forall i \in C_k, \forall j \in p, \text{Centroid}_{k,j} = \frac{\sum_i X_{i,j}}{\mid C_k \mid}$$

(cluster centroid is mean of each feature for the observations belonging to it)

  - Assign each observation to the cluster whose centroid is closest (where “closest” is defined using Euclidean distance.

$$k_i = argmin_k \parallel x_i - Centroid_k \parallel^2$$

K-Means Algorithm - Visualized



K-Means Clustering Algorithm Alternative Initialization

- Alternative Initial Step:
  Randomly select k starting centroids
- Iterate until the cluster centroids each change very little:
  - Assign each observation to the cluster whose centroid is closest (where "closest" is defined using Euclidean distance.

$$k_i = argmin_k \, || \, x_i - Centroid_k \, ||^2$$

  - For each of the K clusters, update the cluster centroid. The kth cluster centroid is the mean of the observations assigned to the kth cluster

$$\forall k, \forall i \in C_k, \forall j \in p, \text{Centroid}_{k,j} = \frac{\sum_i X_{i,j}}{|C_k|}$$

(cluster centroid is mean of each feature for the observations belonging to it)

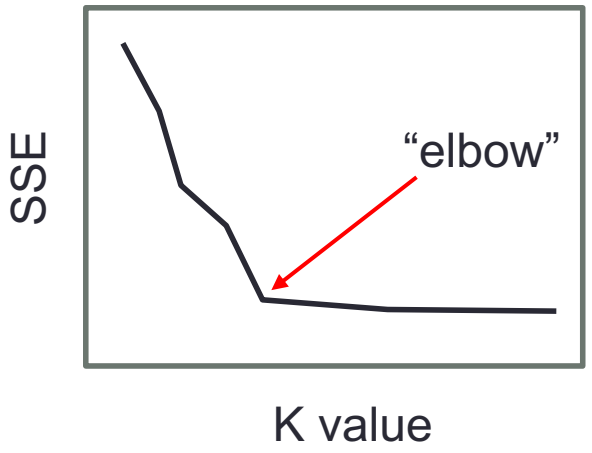
K-Means Considerations: *K*

- K-Means Achieves the property:

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

- In *K-means* clustering, we must specify K for the number of clusters we desire
  - If we know how many clusters we want, then we can select K
  - What if we don't know? How do we determine a "best" K?
    - Elbow method – compute SSE for several K values and look for the "elbow":

$$SSE=\sum_{k=1}^{k}\sum_{x\in C_k}(x-C_k)^2$$



  - Other Potential (Automatic) Solutions:
    - χ-Means (Bayesian Information Criteria)
    - G-Means (Anderson-Darling)
    - PG-Means (Kolmogorov-Smirnov test)

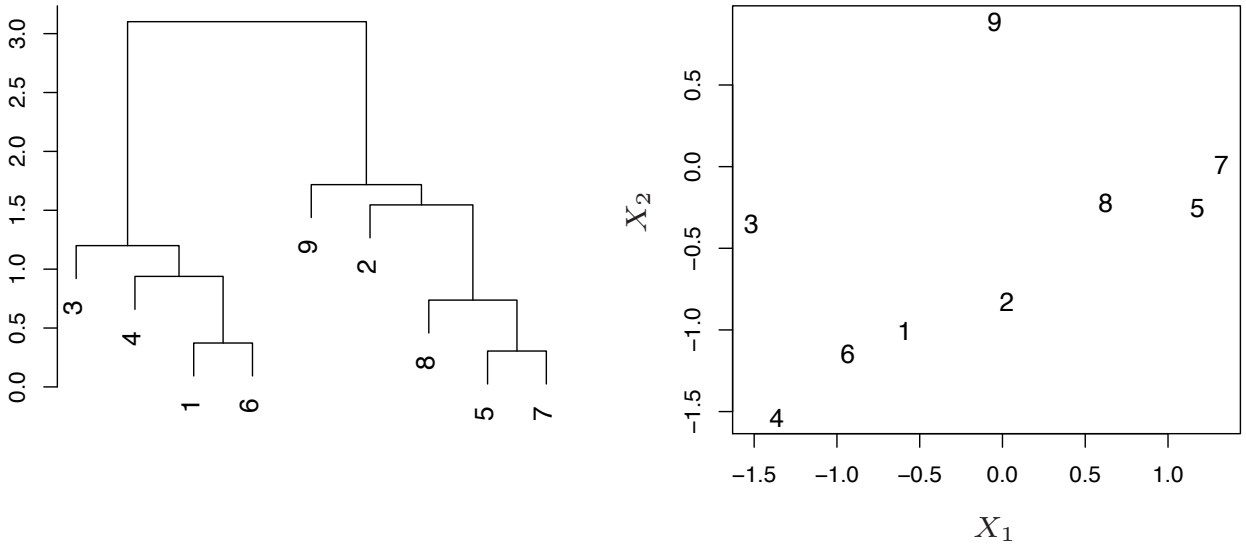
HIERARCHICAL CLUSTERING

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 Dendogram

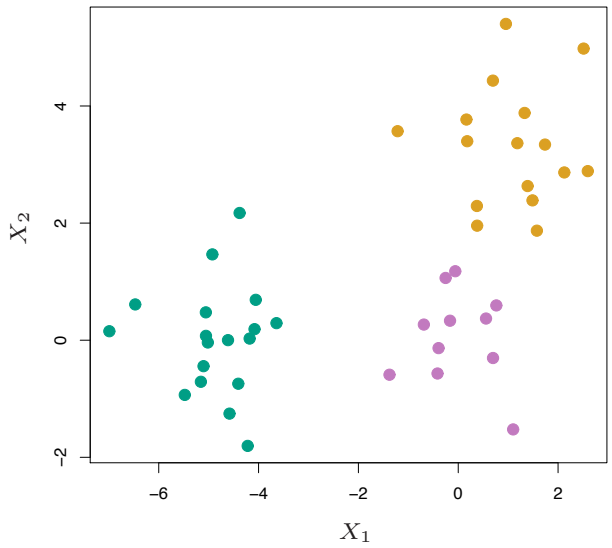
Dendograms

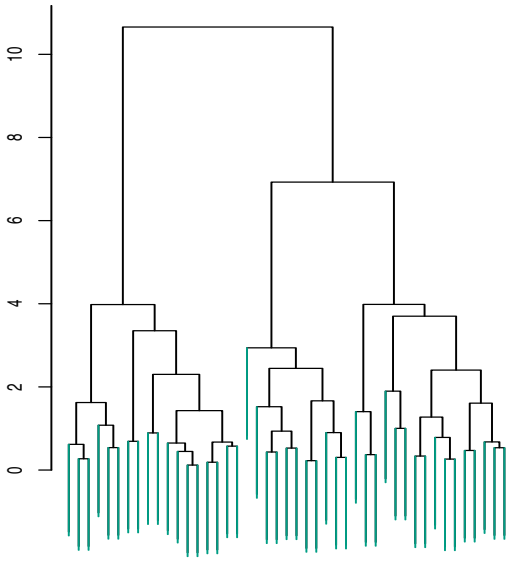
- First join closest points (5 and 7)
- Height of fusing/merging (on vertical axis) indicates how similar the points are
- After the points are fused they are treated as a single observation and the algorithm continues



Interpretation

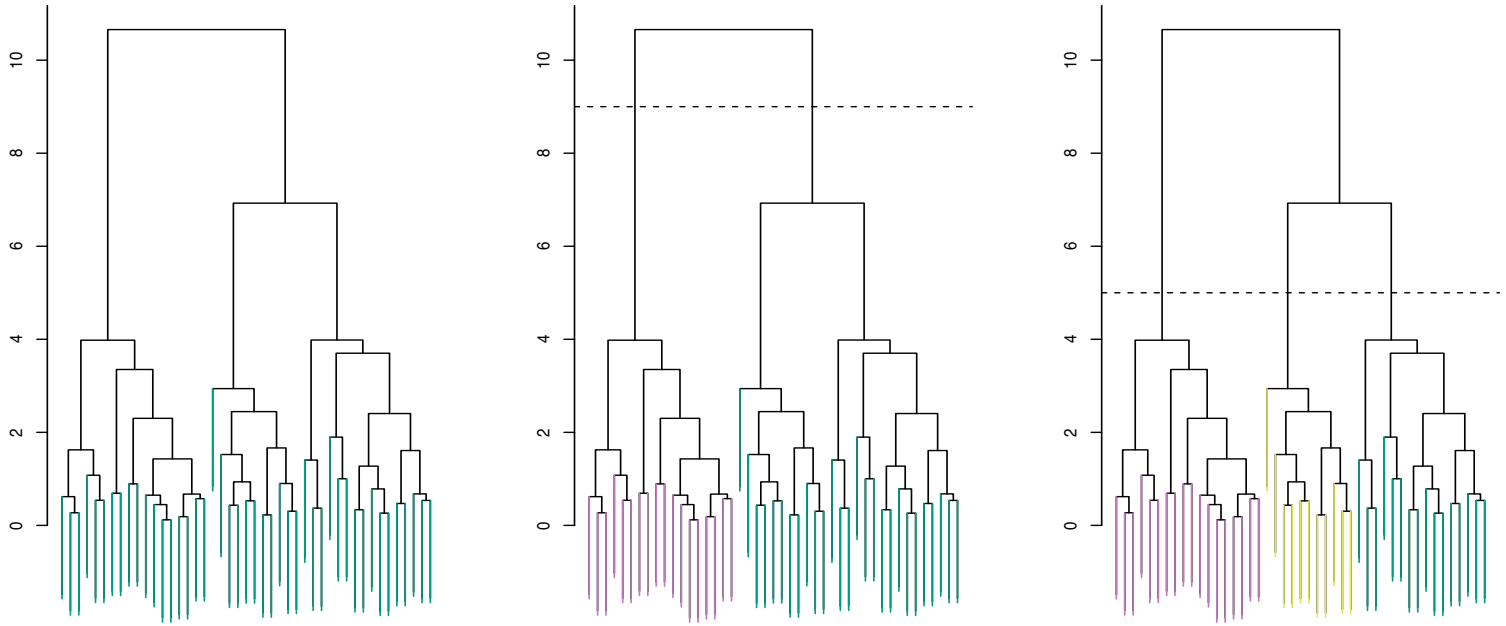
- Each “leaf” of the dendogram represents one of the 45 observations
- At the bottom of the dendogram, each observation is a distinct leaf. However, as we move up the tree, some leaves begin to fuse. These correspond to observations that are similar to each other.
- As we move higher up the tree, an increasing number of observations have fused. The earlier (lower in the tree) two observations fuse, the more similar they are to each other.
- Observations that fuse later are quite different





Choosing Clusters

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



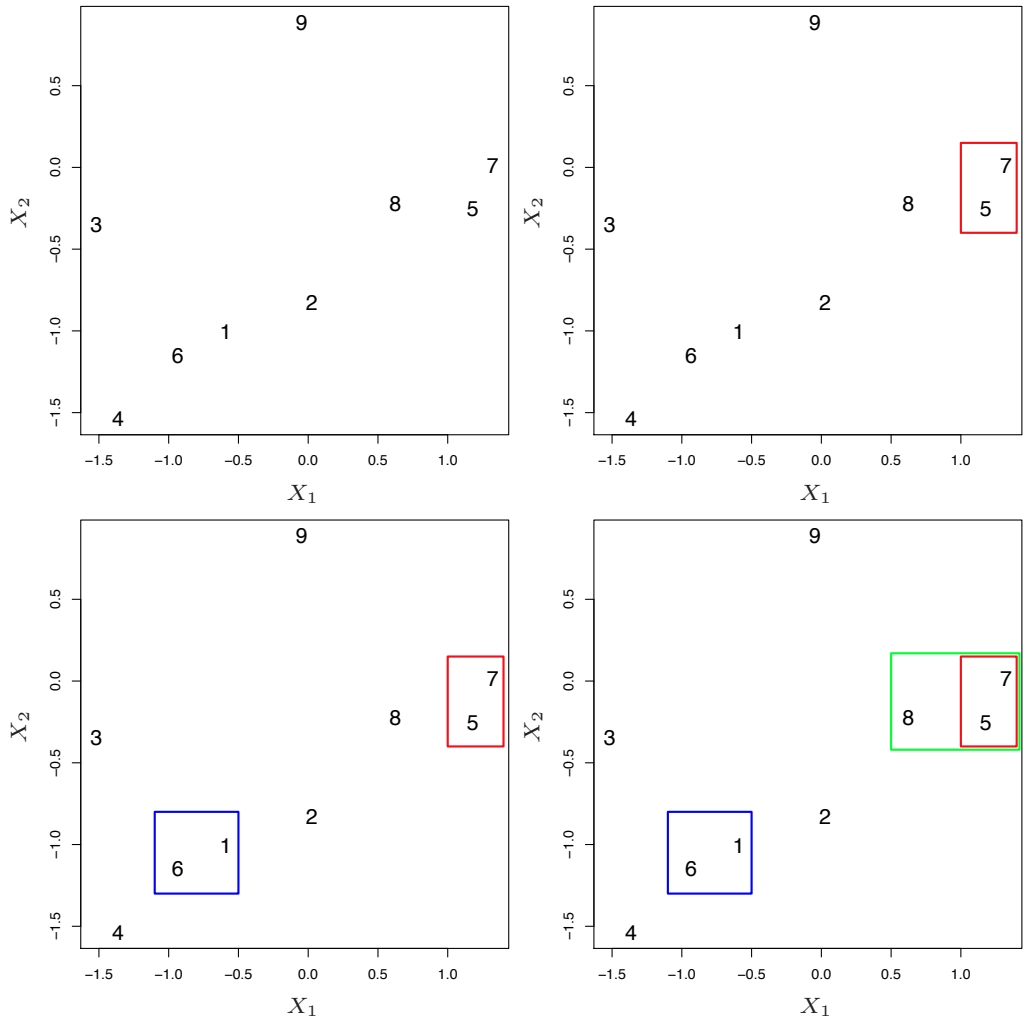
One Cluster | Two Clusters | Three Clusters

Algorithm (Agglomerative Approach)

- The dendogram is produced as follows:
  - Start with each point as a separate cluster (n clusters)
  - Calculate a measure of dissimilarity between all points/clusters
  - Fuse two clusters that are most similar so that there are now n-1 clusters
  - Fuse next two most similar clusters so there are now n-2 clusters
  - Continue until there is only 1 cluster

An Example

- 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.

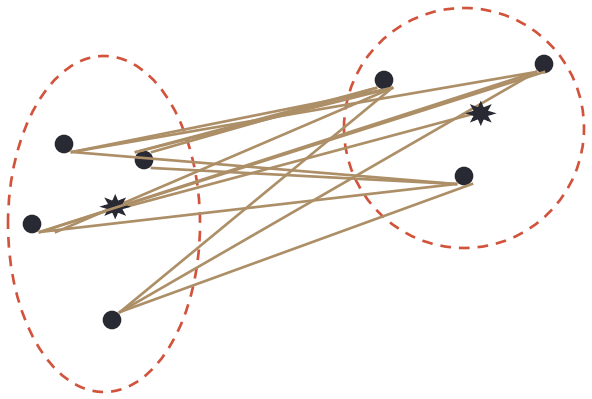


How do we define dissimilarity?

- Implementing hierarchical clustering involves one obvious issue
- How do we define the dissimilarity, or linkage, between the fused (5,7) cluster and 8?
- There are four options:
  - Complete Linkage
  - Single Linkage
  - Average Linkage
  - Centroid Linkage

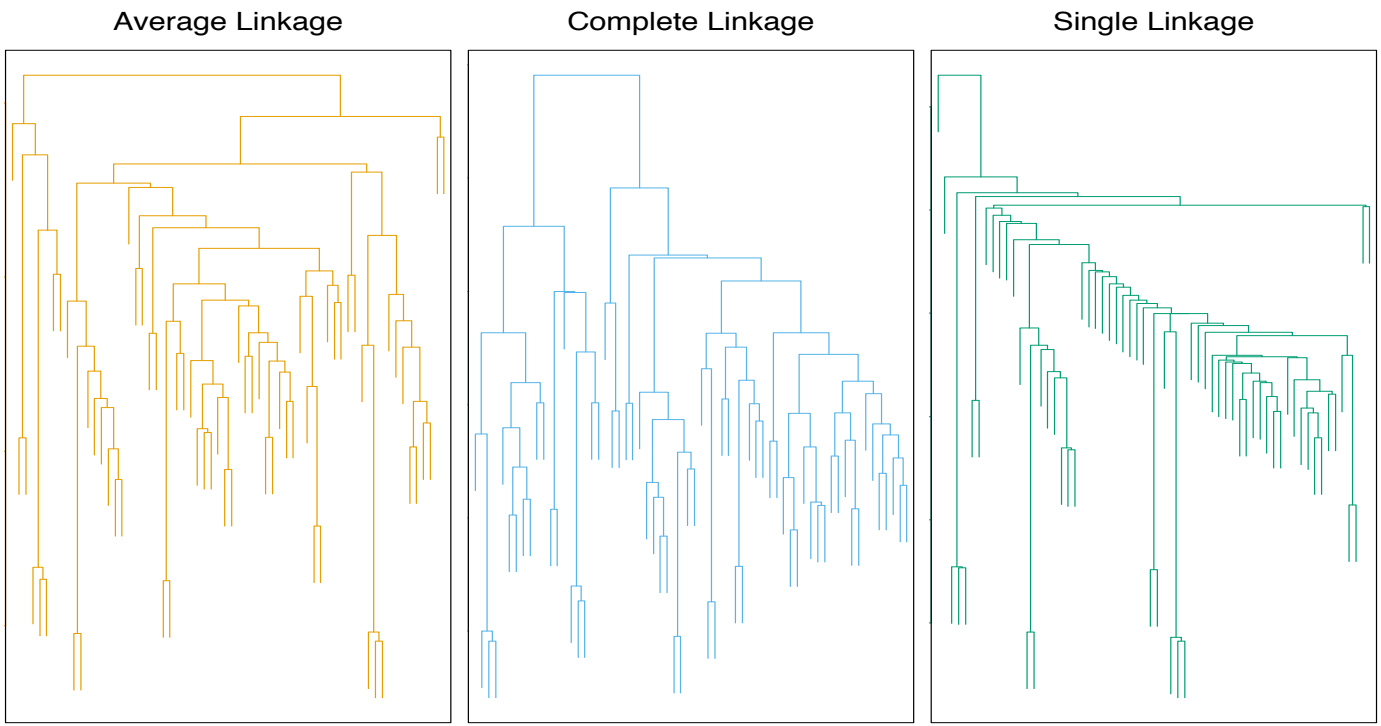
Linkage Methods: Distance Between Clusters

- **Complete Linkage:** Largest distance between observations
- **Single Linkage:** Smallest distance between observations
- **Average Linkage:** Average distance between observations
- **Centroid:** distance between centroids of the observations



Linkage Can be Important

- Here we have three clustering results for the same data
- The only difference is the linkage method but the 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.



Exercise

- Suppose that we have 5 observations, for which we compute a similarity (distance) matrix as follows:

	A	B	C	D	E
A	0				
B	9	0			
C	3	7	0		
D	6	5	9	0	
E	11	10	2	8	0

- On the basis of the similarity matrix, sketch the dendogram that results from hierarchically clustering these 5 observations using complete linkage.

FINAL THOUGHTS

Practical Issues in Clustering

- In order to perform clustering, some decisions must be made:
  - Should the features first be normalized? 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?
- In practice, we try several different choices, and look for the one with the most useful or interpretable solution.

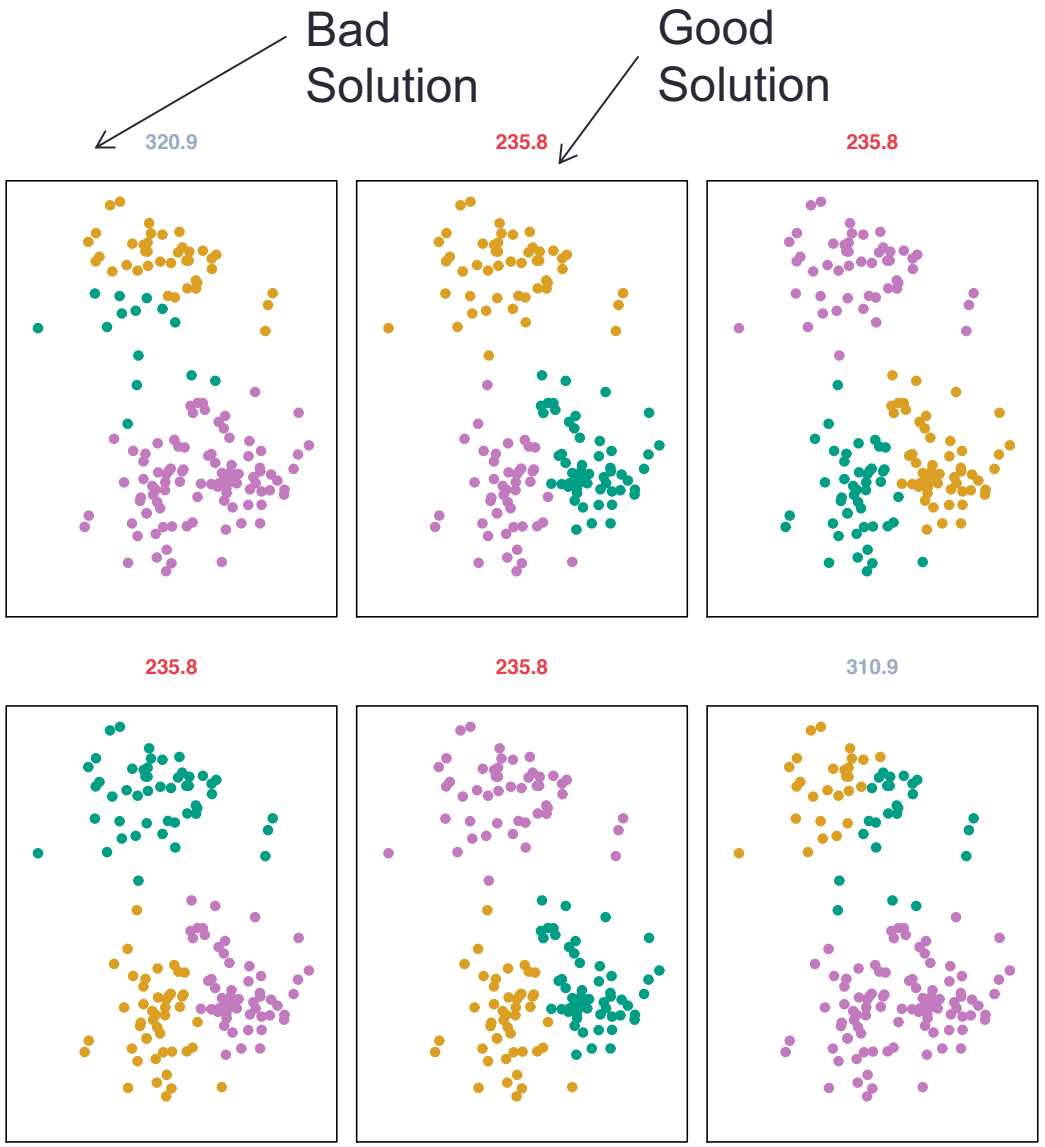
Using the results of clustering

- Most importantly, one must be careful about how the results of a clustering analysis are reported
- These results should not be taken as the absolute truth about a data set
- Rather, they should constitute a starting point for the developments of a scientific hypothesis and further study, preferably on independent data

Backup Slides

K-Means Considerations: Local Optimums

- The K-means algorithm can get stuck in “local optimums” and not find the best solution
- Hence, it is important to run the algorithm multiple times with random starting points to find a good solution

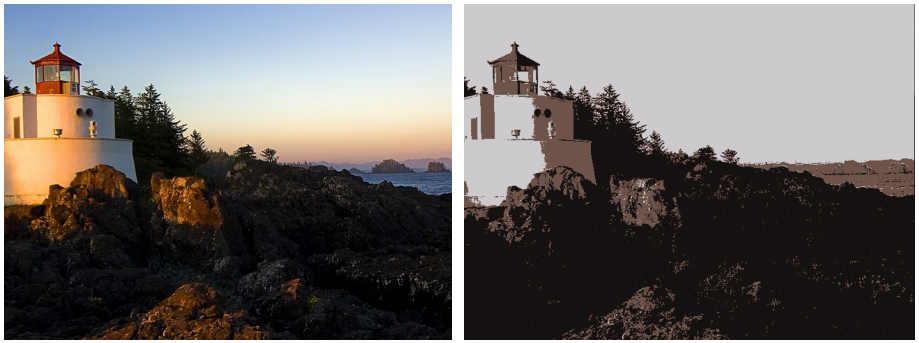


K-Means Considerations: Data

- Identification of potential cluster shape – non-convex shapes will perform poorly
  - Alternatives: CRYSTAL, DBSCAN,…
- Choice of similarity function
  - Continuous
    - Euclidean: $d(x_i, x_j) = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2}$
      - Normalize disparate axis dimensions x [0..100], y [0..1]
    - Mahalanobis: $d_M(\vec{x_i}, \vec{x_j}) = (\vec{x_i} - \vec{x_j})\sum^{-1}(\vec{x_i} - \vec{x_j})^T$
  - Discrete Binary - Jaccard index
  - Discrete - Dice/Czehanovsky-Sorensen measure
  - Mixed:
    - Gower similarity
    - Podani (Gower extended with ordinals)
  - Discrete and Mixed: Where is the cluster center?
    - k-Mediods

K-Mediods

1) **Initialize**: Randomly assign $\{C_1 \ldots C_K\}$ to K samples from $\{x_1 \ldots x_N\}$
2) **Assignment**: Assign each x_n to one of the $k=\{1 \ldots K\}$ cluster centers $\{C_1 \ldots C_K\}$ (distance based on similarity measure)
3) **Update**: For the given cluster assignment, update each C_k to the x_n in each cluster k that minimizes the error
4) **Repeat** until converged



Subsampling/ Vector Quantization

Image Summarization

Overview of Computational Complexities

- Use Big-Oh notation
  - Upper bounds on computational complexity
  - Performance bounds is based on:
    - *M*-Number of samples
    - *L*-Number of iterations
    - *K*-The number of clusters
    - *n*-Dimensionality of the data
- Performance Bounds:
  - K-Means – O(*nMKL*)
  - Soft K-Means – O(*nMKL*)
  - K-Mediods – O($nLKM_k^2$)

How good is our clustering?

- Evaluation without class labels (recall inter and intra-cluster optimizations)
  - Homogeneity
  - Separation
  - Silhouette Width
  - Davies Bouldin index
  - Dunn index - ratio between the minimal inter-cluster (center to center) distance to maximal intra-cluster (farthest point in cluster to farthest point in cluster) distance
- Evaluation with class labels: purity, F-measure, Rand index, Adjusted Rand index, Jaccard index, Fowled-Mallows index