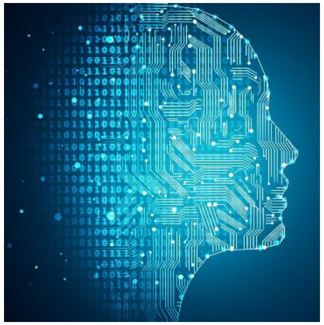
Principles of Machine Learning

Hierarchical Clustering

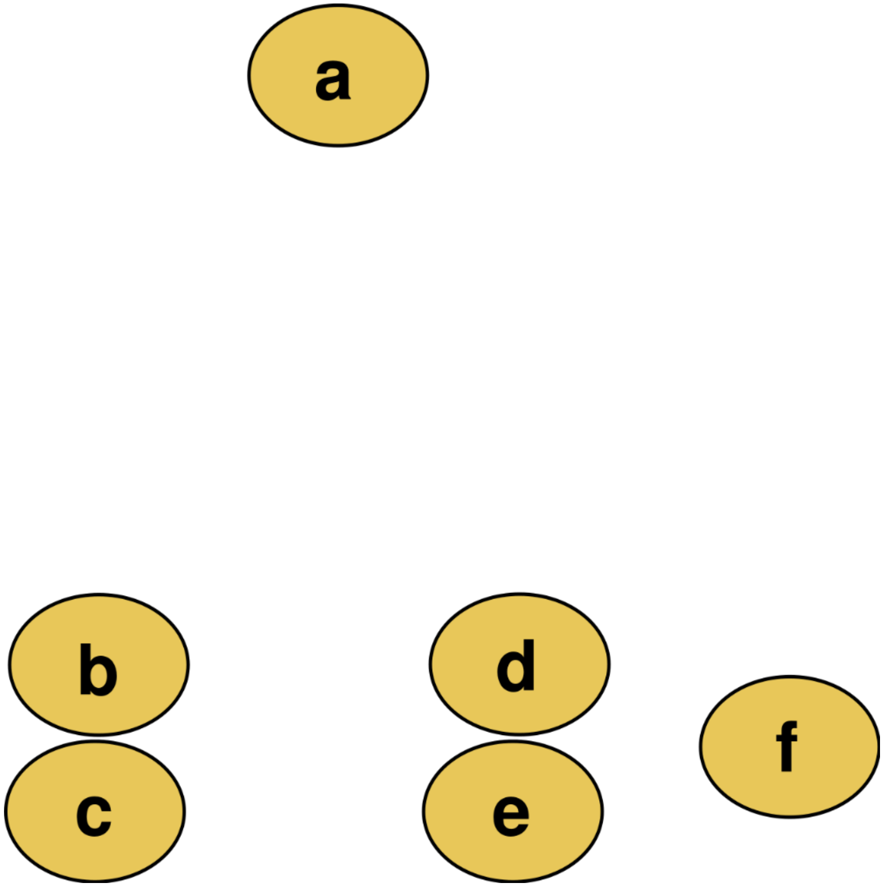
Kevin Moon (kevin.moon@usu.edu)

STAT/CS 5810/6655



Clusters at different scales

- How would you cluster this data?
- K-means (and many other methods) produce a single partition (clustering) of a dataset
- **Hierarchical clustering** produces a hierarchy of clusterings

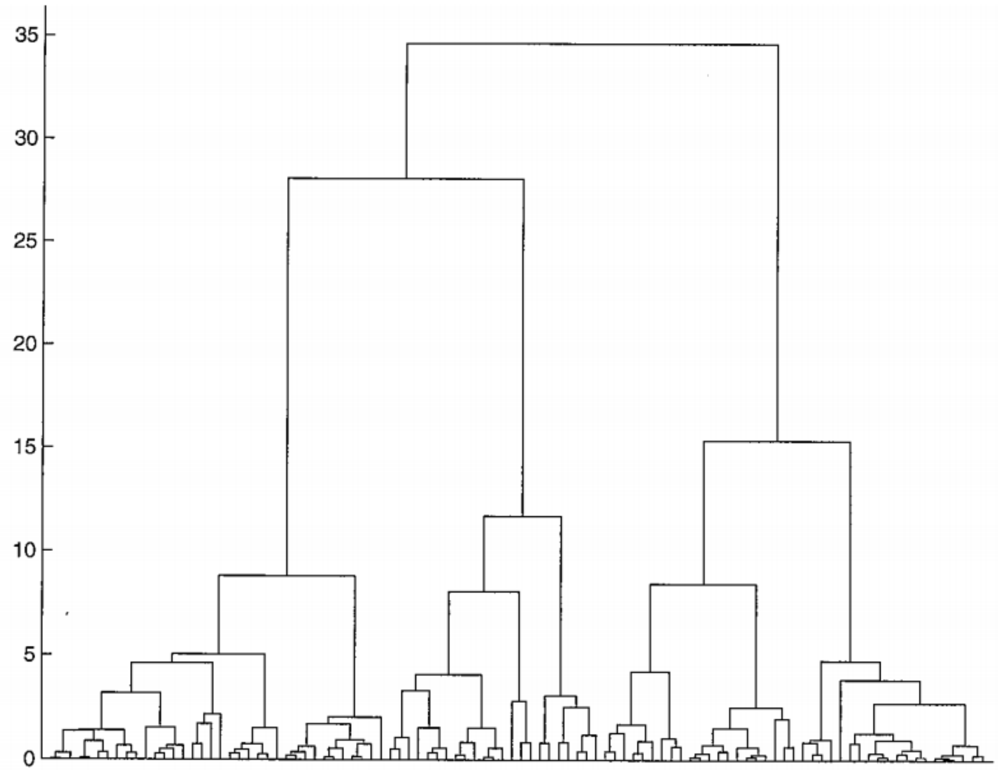


Hierarchical Clustering

- Data $\boldsymbol{x}_1, \dots, \boldsymbol{x}_n \in \mathbb{R}^d$
- A hierarchical clustering has n levels with each level corresponding to a different partition or cluster map
- The levels are hierarchical:
  - Level n: $\{\boldsymbol{x}_1\}, \{\boldsymbol{x}_2\}, \dots, \{\boldsymbol{x}_n\}$
  - Level 1: $\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\}$
  - Level k, $1 \leq k \leq n$: formed by merging two clusters at level $k+1$

Dendrograms

- Hierarchical clusterings can be represented graphically using a **dendrogram**
- Horizontal axis: organization of clusters (not unique)
- Vertical axis: dissimilarity of child clusters



Advantages and algorithms

Advantages of hierarchical clustering (over other methods like k-means)

- Clusters may exist at multiple scales
  - I.e. clusters may have subclusters
- Do not need to specify # of clusters in advance

Two main algorithms for hierarchical clustering

- Agglomerative (bottom-up)
- Divisive (top-down)

Dissimilarities

- Hierarchical clustering algorithms require a **dissimilarity matrix** as input

$$D = \left[d_{ij}\right]_{i,j=1}^{n}, \qquad d_{ij} = d\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right)$$

- The dissimilarity matrix is used to define a dissimilarity between two <u>clusters</u>
  - Multiple ways to do this
- **Example**: average dissimilarity between clusters A and B

$$d_{avg}(A, B) = \frac{1}{|A||B|} \sum_{\boldsymbol{x} \in A} \sum_{\boldsymbol{y} \in B} d(\boldsymbol{x}, \boldsymbol{y})$$

Agglomerative Hierarchical Clustering

- Denote $\mathcal{H}_k$ = set of clusters at level k

Algorithm

- Initialize $\mathcal{H}_n = \{\{\boldsymbol{x}_1\}, \{\boldsymbol{x}_2\}, \dots, \{\boldsymbol{x}_n\}\}$
- For $k = n - 1$ down to 1
  - Select clusters $A, B \in \mathcal{H}_{k+1}$ for which $d(A, B)$ is minimal
  - Set $\mathcal{H}_k$ to be $\mathcal{H}_{k+1}$ with A and B deleted and $A \cup B$ added
- End

- In other words, we iteratively merge the two least dissimilar clusters until we have one cluster

Linkage

- **Linkage function**: formula that relates point dissimilarities to cluster dissimilarities
- **Examples**: next slide

Linkage Function Examples

- Average linkage

$$d_{avg}(A,B) = \frac{1}{|A||B|}\sum_{\boldsymbol{x}\in A}\sum_{\boldsymbol{y}\in B} d(\boldsymbol{x},\boldsymbol{y})$$

- Single linkage

$$d_{min}(A,B) = \min_{\substack{\boldsymbol{x}\in A \\ \boldsymbol{y}\in B}} d(\boldsymbol{x},\boldsymbol{y})$$

- Complete linkage

$$d_{max}(A,B) = \max_{\substack{\boldsymbol{x}\in A \\ \boldsymbol{y}\in B}} d(\boldsymbol{x},\boldsymbol{y})$$

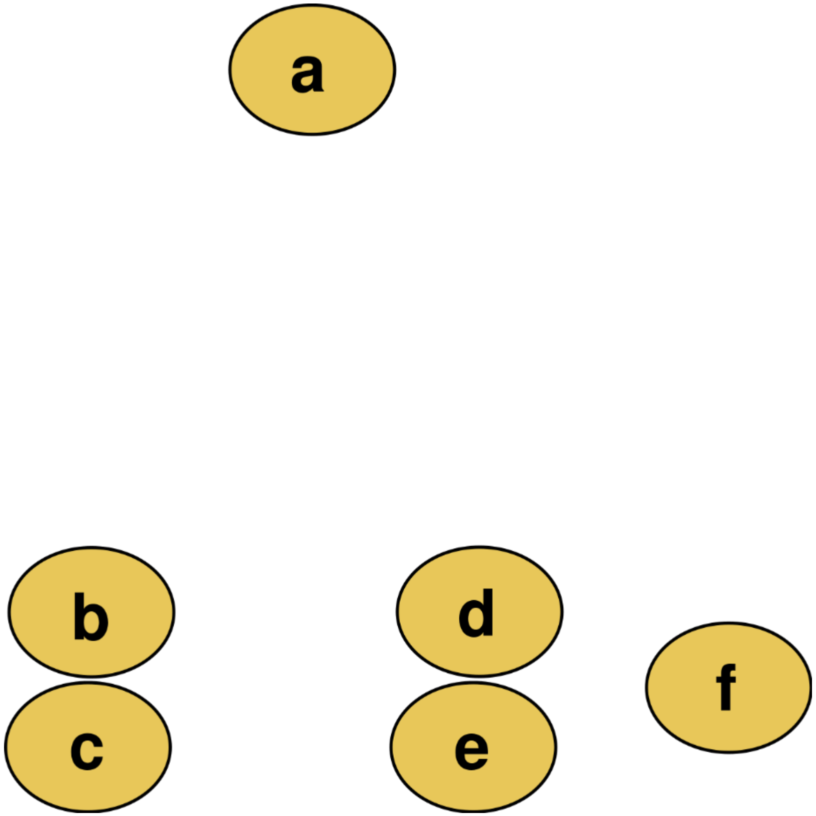
- Centroid linkage

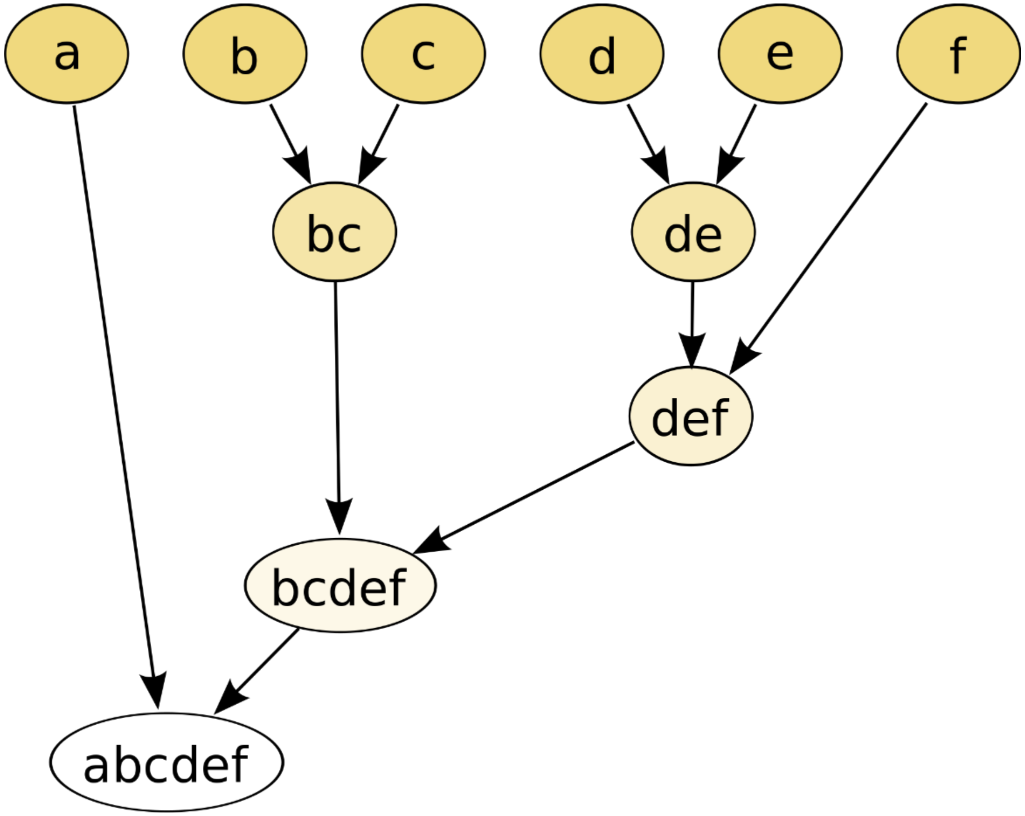
$$d_{cent}(A,B) = \|\overline{\boldsymbol{x}}_A - \overline{\boldsymbol{x}}_B\|$$

- Ward's linkage

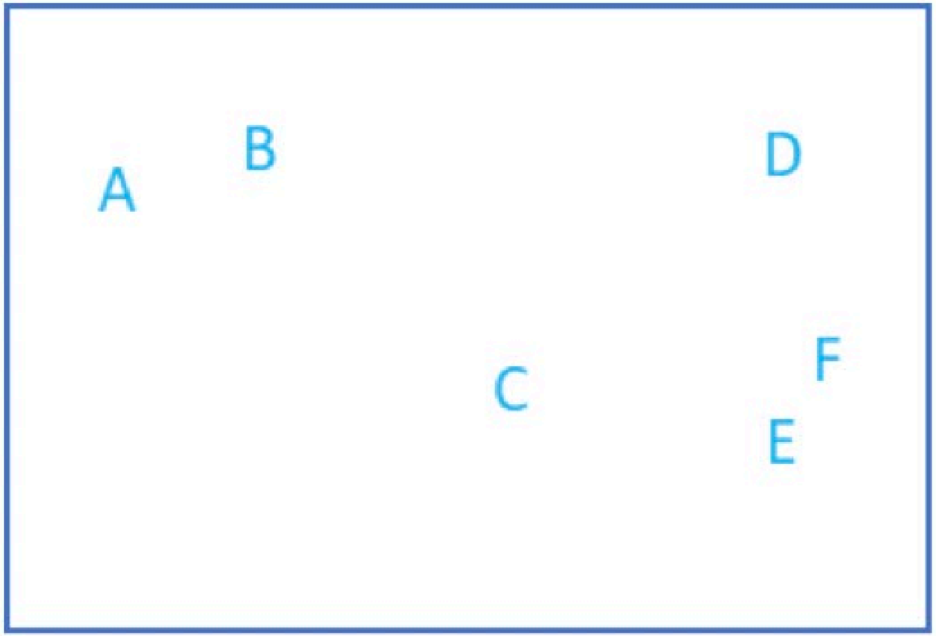
$$d_{Ward}(A,B) = \sqrt{\frac{n_A n_B}{n_A + n_B}}\|\overline{\boldsymbol{x}}_A - \overline{\boldsymbol{x}}_B\|$$

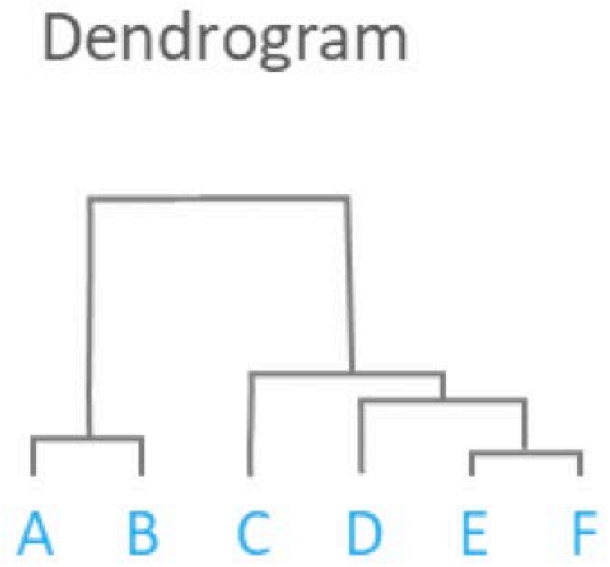
Example: single linkage results





Example: single linkage results

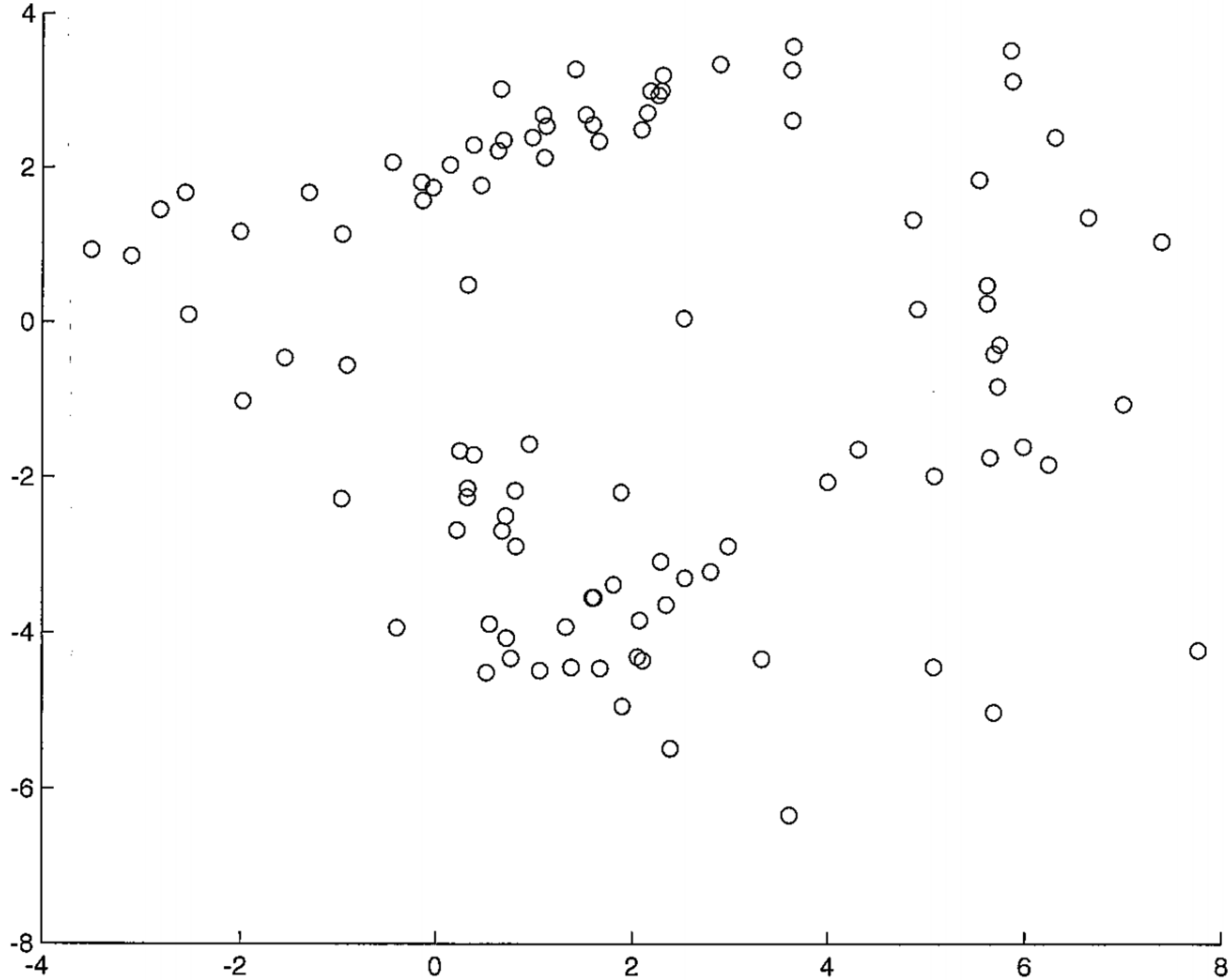




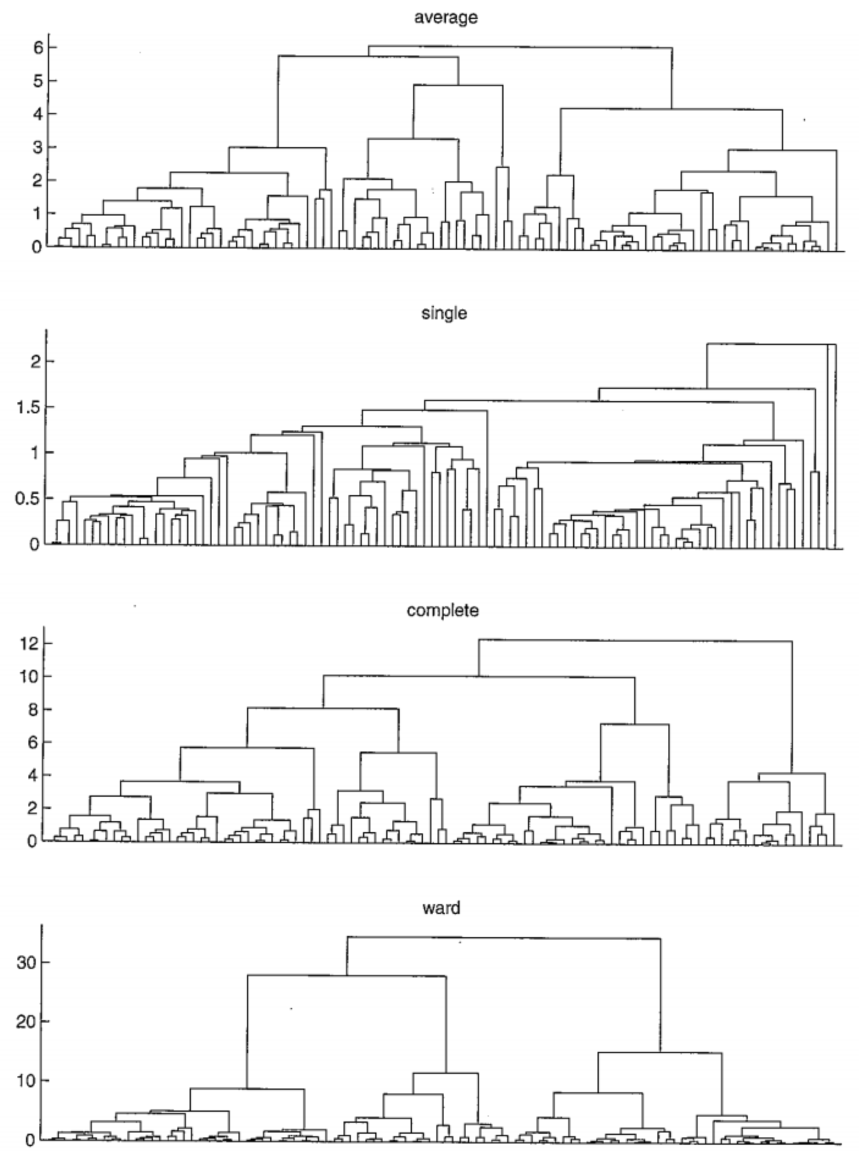
Remarks

- Centroid and Ward's linkage are not built out of an underlying point dissimilarity
- Average, single, and complete linkages can be applied to cluster non-Euclidean data as long as point dissimilarities are defined.
- The choice of linkage function has a major effect on the clustering
  - Also, there is often no clear choice for which is best to use

Dendrogram Comparisons



Dendrogram Comparisons



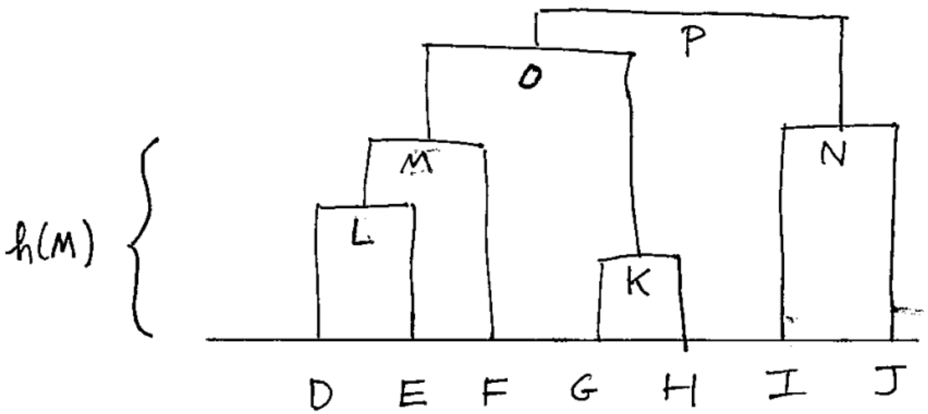
More Remarks

- Single linkage
  - Generates a minimal spanning tree
  - Sensitive to outliers: tends to merge them at the very end
  - Chaining: tends to produce elongated clusters
- Complete linkage
  - Discourages elongated clusters
  - Favors clusters with small diameter
- Average linkage
  - Compromise between single and complete
  - Affected by monotone scaling of d_{ij}
- Centroid linkage
  - Easy to compute
  - Dendrogram can be non-monotone
- Ward's linkage
  - Corrects centroid's monotonicity problem

Monotonicity

- Certain linkages have a monotonicity property
  - Allows us to assign a quantitative value to the height of nodes in the dendrogram
- More specifically, suppose a node was formed by merging two clusters A and B. Then the height of $A \cup B$ is defined to be
$$d(A, B)$$
- **Definition**: a linkage d is monotone if for any cluster $\{A \cup B\} \cup C$ produced by hierarchical clustering where A and B are merged first, we have
$$d(A \cup B, C) \geq d(A, B)$$

Example



Denote h = height

- $h(M) = d(L, F) = d(D \cup E, F) \geq d(D, E) = h(L)$
- $h(O) = d(M, K) = d(L \cup F, K) \geq d(L, F) = h(M)$
- $h(P) = d(O, N) \geq h(O)$ and $h(N)$

Proof: Single Linkage is Monotone

- Suppose HC produces the cluster $\{A \cup B\} \cup C$

$$d(A \cup B, C) = \min_{\substack{\boldsymbol{x} \in A \cup B \\ \boldsymbol{z} \in C}} d(\boldsymbol{x}, \boldsymbol{z})$$

$$= \min \left\{ \min_{\substack{\boldsymbol{x} \in A \\ \boldsymbol{z} \in C}} d(x, z), \min_{\substack{\boldsymbol{y} \in B \\ \boldsymbol{z} \in C}} d(\boldsymbol{y}, \boldsymbol{z}) \right\}$$

$$\geq \min_{\substack{\boldsymbol{x} \in A \\ \boldsymbol{y} \in B}} d(\boldsymbol{x}, \boldsymbol{y})$$

$$= d(A, B)$$

otherwise, A, B would have merged with C

Proof: Average Linkage is Monotone

$$d_{avg}(A \cup B, C) = \frac{1}{n_C(n_A + n_B)} \sum_{\boldsymbol{z} \in C} \sum_{\boldsymbol{x} \in A \cup B} d(\boldsymbol{z}, \boldsymbol{x})$$

$$= \frac{1}{n_C} \sum_{\boldsymbol{z} \in C} \left(\frac{1}{n_A + n_B} \left(\sum_{\boldsymbol{x} \in A} d(\boldsymbol{z}, \boldsymbol{x}) + \sum_{\boldsymbol{y} \in B} d(\boldsymbol{z}, \boldsymbol{y}) \right) \right)$$

$$= \frac{n_A}{n_A + n_B} d(A, C) + \frac{n_B}{n_A + n_B} d(B, C)$$

$$\geq \frac{n_A}{n_A + n_B} d(A, B) + \frac{n_A}{n_A + n_B} d(A, B)$$

$$= d(A, B)$$

Proof: Ward's Linkage is Monotone

- Proof is based on connection within-class scatter (recall the k-means lecture)

Global Criterion

- HC defines a cluster to be the output of a certain algorithm
- Can we view HC as an algorithm for (approximately) optimizing a global objective function?

Global Criterion

- Let $\mathcal{J}_k$ be an objective function that assesses the quality of a clustering into K clusters

Algorithm

- Initialize $\mathcal{H}_n = \{\{\boldsymbol{x}_1\}, \{\boldsymbol{x}_2\}, \dots, \{\boldsymbol{x}_n\}\}$
- For $k = n - 1$ down to 1
  - Find $A, B \in \mathcal{H}_{k+1}$ such that merging A and B to form $\mathcal{H}_k$ yields the smallest $\mathcal{J}_k$
- End

- Does this greedy algorithm ever coincide with HC?
  - Sometimes

Examples

- Complete linkage

$$\mathcal{T}_k(\mathcal{H}) = \max_{A \in \mathcal{H}} \left(\max_{\boldsymbol{x},\boldsymbol{y} \in A} d_{max}(\boldsymbol{x}, \boldsymbol{y}) \right)$$

  - Gives the max cluster diameter

- Ward's linkage

$$\mathcal{T}_k(\mathcal{H}) = \text{within cluster scatter (as in K-means)}$$

Additional Items

- HC can be used as initialization for other clustering methods
- Choosing k: Sometimes, we want to choose a specific level of clustering.
    - Options:
        - Same method as k-means (sum of within-cluster distances as a function of k)
        - Look for a large jump in the dendrogram
- Instability: HC is sensitive to perturbations of the data

Interpretation

- Dendrogram = summary of the algorithm, not a summary of the data
  - To what extent does the dendrogram represent the actual structure of the data?
- Model-based interpretation
  - HC may be viewed as a greedy method for maximum likelihood estimation of cluster parameters where different generative models correspond to different linkages

Further Reading

- Wikipedia on "Hierarchical Clustering"
- Kamvar, Klein, and Manning, "Interpreting and extending classical agglomerative clustering algorithms using a model-based approach."
- https://www.displayr.com/what-is-hierarchical-clustering/
- ISL Section 10.3.2
- ESL Section 14.3.12