Clustering

Hierarchical clustering

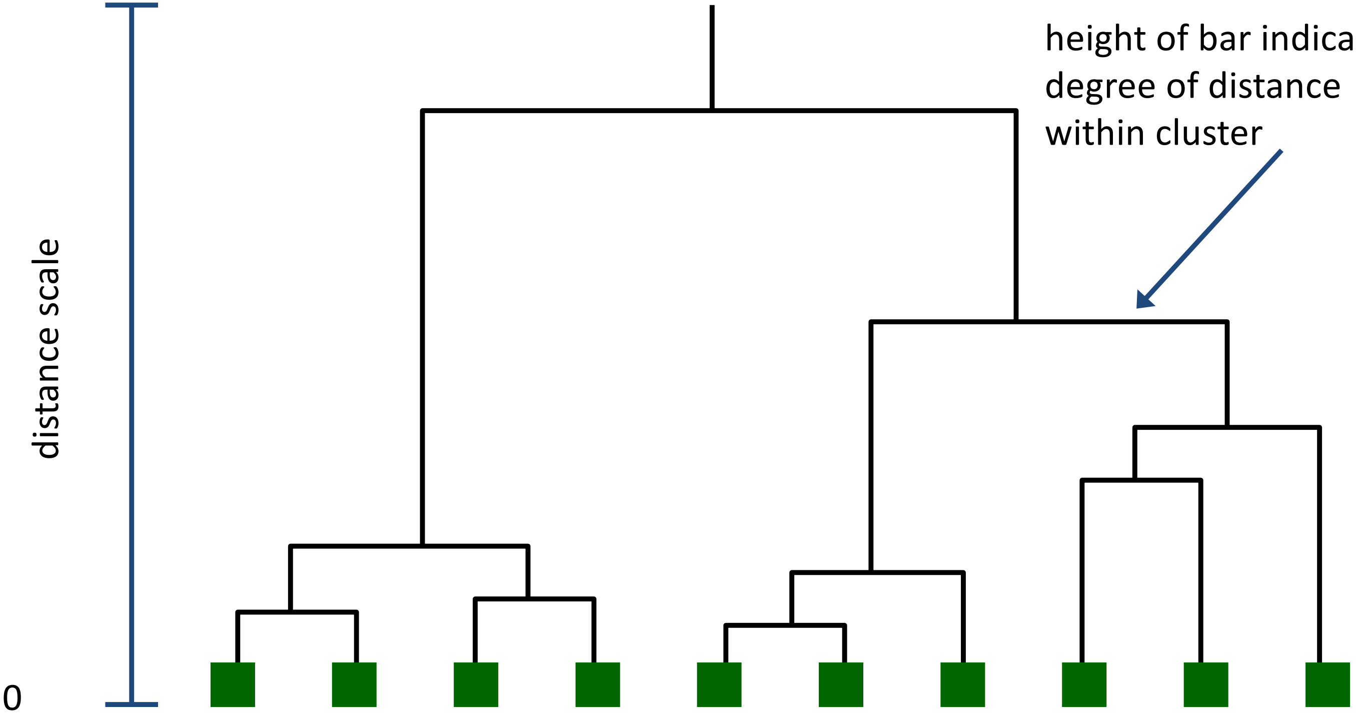
Outline

- Hierarchical vs. flat clustering
- Hierarchical clustering task definition
- Top-down vs. bottom up clustering
- Distances between clusters
- Computational complexity

Hierarchical vs. Flat clustering

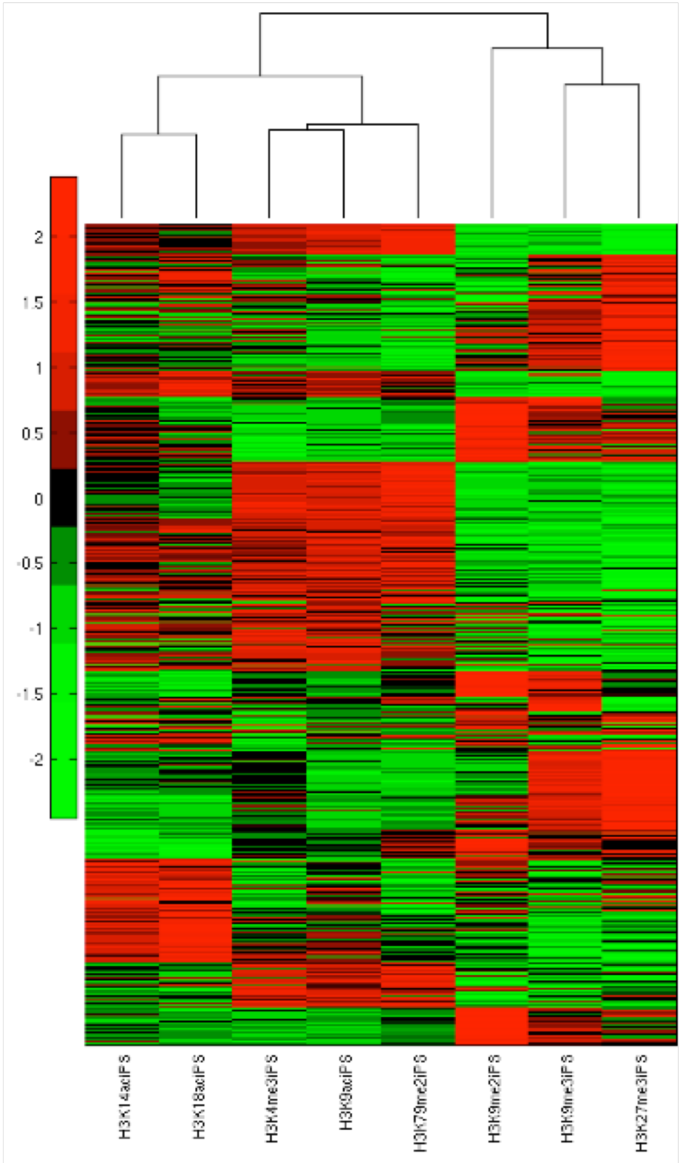
- Flat clustering (e.g., K-means and Gaussian Mixture Models)
  - Number of clusters, *K,* is pre-specified
  - Each object is assigned to one of these clusters
- Hierarchical clustering
  - Hierarchical relationships established between all objects
  - A threshold on the maximum dissimilarity can be used to convert a hierarchical clustering into a flat clustering
    - Multiple flat clusterings can be produced by varying the threshold

Hierarchical clustering



height of bar indicates degree of distance within cluster

leaves represent objects to be clustered (e.g. genes or samples)



Hierarchical clustering example

clustering of chromatin marks measured near genes in a particular cell type (induced pluripotent cell (iPS))

- Columns correspond to chromatin marks
- Eight marks
  - Five activating
    - H3K14
    - H3K18
    - H3K4me3
    - H3K9ac
    - H3K79me2
  - Three repressive
    - H3K9me2
    - H3K9me3
    - H3K27me3

Data from Sridharan et al.

Hierarchical clustering

- **Input**: a set $X=\{x_1,.. x_n\}$ of data points, where each x_i is a *p*-dimensional vector
- **Output**: a rooted tree with the data points at the leaves of the tree
- Two major strategies
  - **top-down** (divisive)
  - **bottom-up** (agglomerative)
- Both strategies **recursively** build a tree by splitting (to-down) or merging (bottom-up) subsets of data points
- We will focus on bottom-up clustering

Top-down clustering

- **Basic idea**: use a flat clustering method to recursively split a set, X, of data points into *K (*usually *K=2*) disjoint subsets
- topdown_cluster(X):

```
if X has only one element x:
    return a tree with a single leaf node labeled by x
else:
    X1, X2 = flat_cluster(X, K=2)
    T1 = topdown_cluster(X1)
    T2 = topdown_cluster(X2)
    return a tree with children T1 and T2
```

Bottom-up hierarchical clustering

```
given: a set $X = \{x_1 \dots x_n\}$ of instances
for $i := 1$ to $n$ do
    $c_i := \{x_i\}$        // each instance is initially its own cluster, and a leaf in tree
$C := \{c_1 \dots c_n\}$
$j := n$
while $|C| > 1$
    $j := j + 1$
    $(c_a, c_b) := \underset{(c_u, c_v)}{\text{argmin}} \; \text{dist}(c_u, c_v)$    // find least distant pair in $C$
    $c_j = c_a \cup c_b$        // create a new cluster for pair
    $C := C - \{c_a, c_b\} \cup \{c_j\}$    // Add new cluster to list of clusters to be
                                           joined in the tree
return tree with root node $j$
```

Distance between two clusters

- The distance between two clusters c_u and c_v can be determined in several ways
  - *single link*: distance of two most similar profiles

$$\text{dist}(c_u, c_v) = \min\left\{\text{dist}(a,b) \mid a \in c_u, b \in c_v\right\}$$

  - *complete link*: distance of two least similar profiles

$$\text{dist}(c_u, c_v) = \max\left\{\text{dist}(a,b) \mid a \in c_u, b \in c_v\right\}$$

  - *average link*: average distance between profiles

$$\text{dist}(c_u, c_v) = \text{avg}\left\{\text{dist}(a,b) \mid a \in c_u, b \in c_v\right\}$$

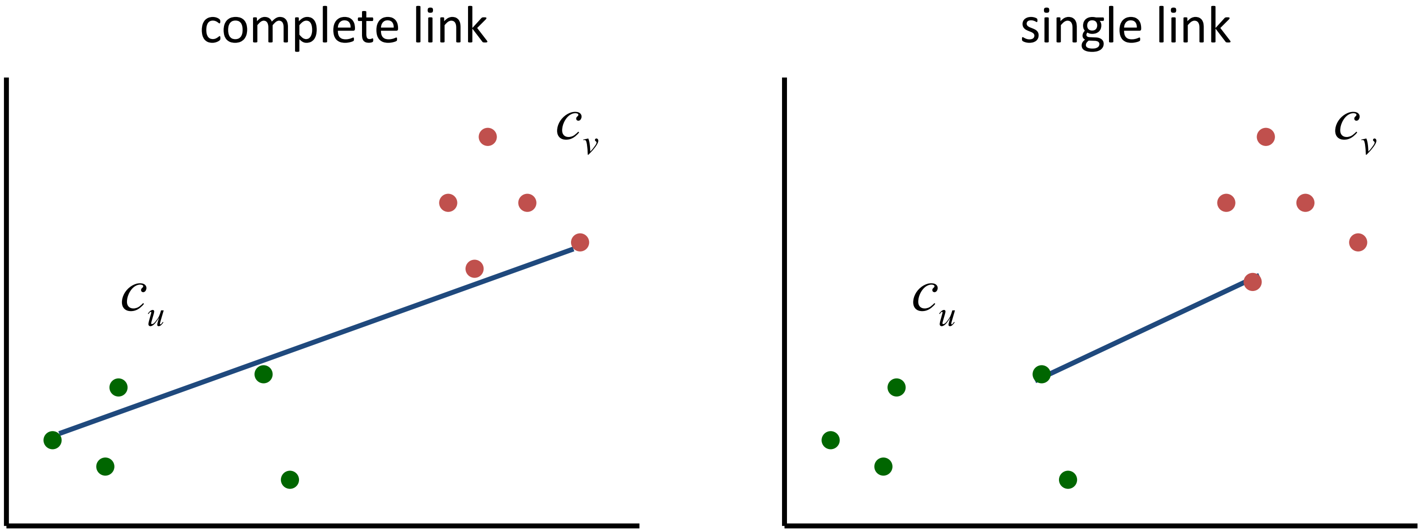
Haven't We Already Seen This?

- Hierarchical clustering is very similar to distance-based phylogenetic methods
- Average link hierarchical clustering is equivalent to UPGMA for phylogenetics

Differences between general clustering and phylogenetic inference

- what a tree represents
  - phylogenetic inference: tree represents hypothesized sequence of evolutionary events; internal nodes represent hypothetical ancestors
  - clustering: inferred tree represents similarity of instances; internal nodes don’t represent ancestors
- form of tree
  - UPGMA: rooted tree
  - neighbor joining: unrooted
  - hierarchical clustering: rooted tree
- how distances among clusters are calculated
  - UPGMA: average link
  - neighbor joining: based on additivity
  - hierarchical clustering: various

Complete-link vs. single-link distances


complete link

single link

Updating distances efficiently

- If we just merged c_u and c_v into c_j, we can determine distance to each other cluster c_k as follows
  - single link:

$$\text{dist}(c_j, c_k) = \min\{\text{dist}(c_u, c_k), \text{dist}(c_v, c_k)\}$$

	AB	C	D
AB		8	12
C			7
D			

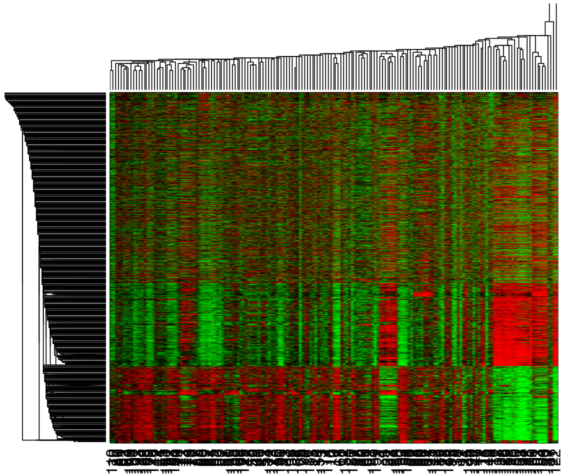
  - complete link:

$$\text{dist}(c_j, c_k) = \max\{\text{dist}(c_u, c_k), \text{dist}(c_v, c_k)\}$$

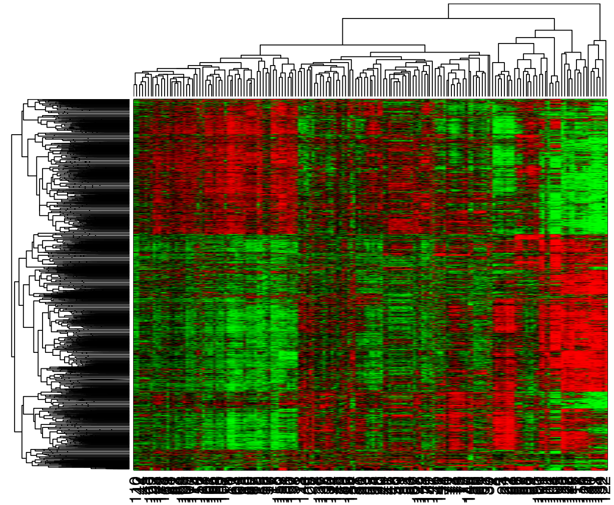
  - average link:

$$\text{dist}(c_j, c_k) = \frac{|c_u| \times \text{dist}(c_u, c_k) + |c_v| \times \text{dist}(c_v, c_k)}{|c_u| + |c_v|}$$

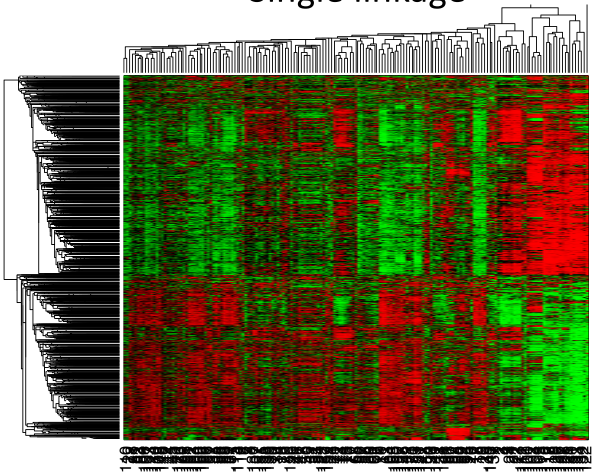
Effect of different linkage methods



Single linkage



Complete linkage

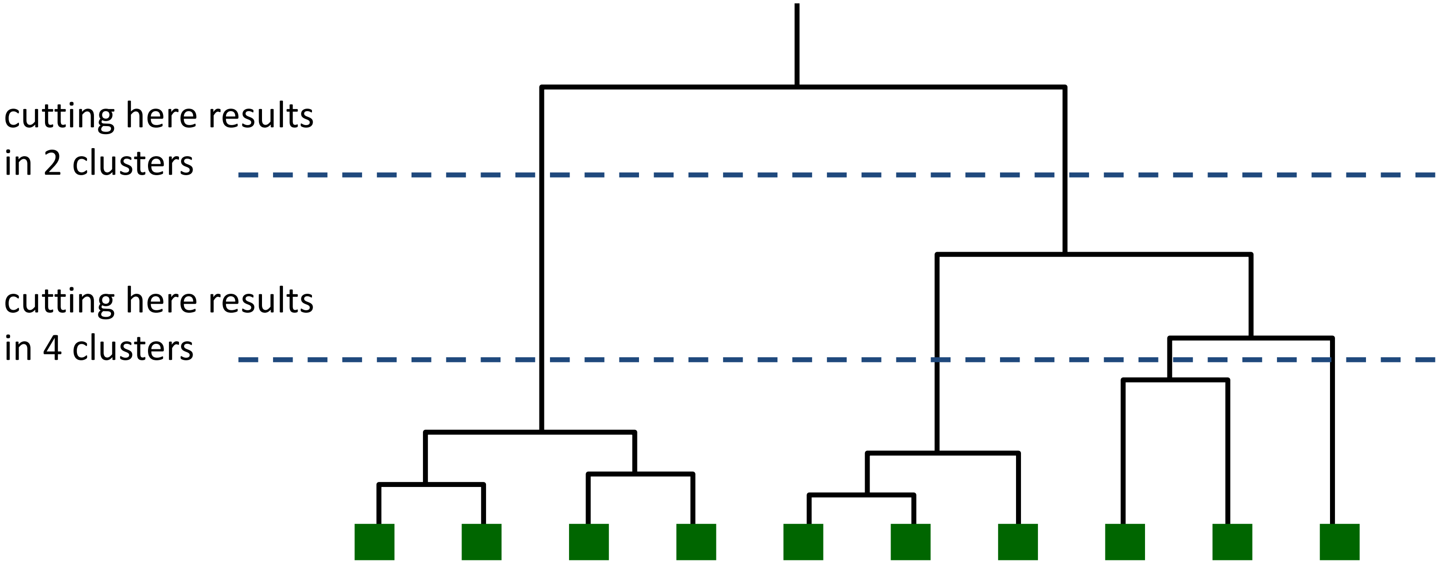


Average linkage

Single linkage might result in a “chaining” effect

Flat clustering from a hierarchical clustering

- We can always generate a flat clustering from a hierarchical clustering by "cutting" the tree at some distance threshold



cutting here results in 2 clusters

cutting here results in 4 clusters

Naïve computational complexity

- The naïve implementation of hierarchical clustering has $O(n^3)$time complexity, where n is the number of objects
  - computing the initial distance matrix takes $O(n^2)$time
  - there are $O(n)$merging steps
  - on each step, we have to update the distance matrix $O(n)$and select the next pair of clusters to merge $O(n^2)$

Computational Complexity

- for single-link clustering, we can update and pick the next pair in $O(n)$ time, resulting in an $O(n^2)$ algorithm
- for complete-link and average-link we can do these steps in $O(n \log n)$ time resulting in an $O(n^2 \log n)$ method

How to pick the right clustering algorithm?

- If you have a sense of what the right number of clusters are, K-means or Gaussian mixture models might be good
- If you want to control for the extent of dissimilarity you should use hierarchical
- Hierarchical clustering is deterministic
  - Always gives the same solution with the same distance metric
- K-means and Gaussian mixture model are non-deterministic
- We have talked about clustering of gene expression profiles
  - However clustering could be used to find groupings among more complex objects
  - *All* we need is to define the right distance metric