Hierarchical clustering approaches for high-throughput data

Colin Dewey
BMI/CS 576
www.biostat.wisc.edu/bmi576
colin.dewey@wisc.edu
Fall 2016
The clustering landscape

• There are many different clustering algorithms

• They differ with respect to several properties
  – hierarchical vs. flat
  – hard (no uncertainty about which profiles belong to a cluster) vs. soft clusters
  – overlapping (a profile can belong to multiple clusters) vs. non-overlapping
  – deterministic (same clusters produced every time for a given data set) vs. stochastic
  – distance (similarity) measure used
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

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

height of bar indicates degree of distance within cluster
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 $X=\{x_1, \ldots, x_n\}$ 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 **iteratively** build a tree by splitting (top-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:
    - $X_1, X_2 = flat_cluster(X, K=2)$
    - $T_1 = topdown_cluster(X_1)$
    - $T_2 = topdown_cluster(X_2)$
    - return a tree with children $T_1$ and $T_2
Bottom-up hierarchical clustering

given: a set \( X = \{x_1...x_n\} \) of instances

for \( i := 1 \) to \( n \) do

\[ c_i := \{x_i\} \quad \text{// each instance is initially its own cluster, and a leaf in tree} \]

\[ C := \{c_1...c_n\} \]

\[ j := n \]

while \( |C| > 1 \)

\[ j := j + 1 \]

\[ (c_a, c_b) := \arg \min_{(c_u, c_v)} \text{dist}(c_u, c_v) \quad \text{// find least distant pair in } C \]

\[ c_j = c_a \cup c_b \quad \text{// create a new cluster for pair} \]

\[ C := C - \{c_a, c_b\} \cup \{c_j\} \quad \text{// Add new cluster to list of clusters to be joined in the tree} \]

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
    \[
    \text{dist}(c_u, c_v) = \min \{\text{dist}(a, b) | a \in c_u, b \in c_v\}
    \]
  - *complete link*: distance of two least similar profiles
    \[
    \text{dist}(c_u, c_v) = \max \{\text{dist}(a, b) | a \in c_u, b \in c_v\}
    \]
  - *average link*: average distance between profiles
    \[
    \text{dist}(c_u, c_v) = \text{avg} \{\text{dist}(a, b) | a \in c_u, b \in c_v\}
    \]
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:
    \[
    \text{dist}(c_j, c_k) = \min\{\text{dist}(c_u, c_k), \text{dist}(c_v, c_k)\}
    \]
  - 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

Single linkage might result in a “chaining” effect

Average linkage
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
Summary of clustering

• Many different methods for clustering
  – Flat clustering
  – Hierarchical clustering
  – Dissimilarity metrics among objects can influence clustering results a lot

• Picking the number of clusters is difficult but there are some ways to do this

• Evaluation of clusters is often difficult
  – Comparison with other sources of information can help assess cluster quality