Clustering
Hierarchical clustering, k-mean clustering

Genome 559: Introduction to Statistical and Computational Genomics
Elhanan Borenstein
A quick review

- The clustering problem:
  - partition genes into distinct sets with high homogeneity and high separation
  - Different representations

- Homogeneity vs Separation

- Many possible distance metrics

- Method matters; metric matters; definitions matter;

- One problem, numerous solutions
Hierarchical clustering
Hierarchical clustering

- **Hierarchical** clustering is an **agglomerative** clustering method
  - Takes as input a distance matrix
  - Progressively regroups the closest objects/groups

**Distance matrix**

<table>
<thead>
<tr>
<th></th>
<th>object 1</th>
<th>object 2</th>
<th>object 3</th>
<th>object 4</th>
<th>object 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>object 1</td>
<td>0.00</td>
<td>4.00</td>
<td>6.00</td>
<td>3.50</td>
<td>1.00</td>
</tr>
<tr>
<td>object 2</td>
<td>4.00</td>
<td>0.00</td>
<td>6.00</td>
<td>2.00</td>
<td>4.50</td>
</tr>
<tr>
<td>object 3</td>
<td>6.00</td>
<td>6.00</td>
<td>0.00</td>
<td>5.50</td>
<td>6.50</td>
</tr>
<tr>
<td>object 4</td>
<td>3.50</td>
<td>2.00</td>
<td>5.50</td>
<td>0.00</td>
<td>4.00</td>
</tr>
<tr>
<td>object 5</td>
<td>1.00</td>
<td>4.50</td>
<td>6.50</td>
<td>4.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

1. Assign each object to a separate cluster.
2. Find the pair of clusters with the shortest distance, and regroup them into a single cluster.
3. Repeat 2 until there is a single cluster.
Hierarchical clustering algorithm

1. Assign each object to a separate cluster.
2. Find the pair of clusters with the shortest distance, and regroup them into a single cluster.
3. Repeat 2 until there is a single cluster.

- The result is a tree, whose intermediate nodes represent clusters.
- Branch lengths represent distances between clusters.
mmm...
Déjà vu anyone?
Hierarchical clustering

1. Assign each object to a separate cluster.
2. Find the pair of clusters with the shortest distance, and regroup them into a single cluster.
3. Repeat 2 until there is a single cluster.

- One needs to define a (dis)similarity metric between two groups. There are several possibilities
  - **Average linkage:** the average distance between objects from groups A and B
  - **Single linkage:** the distance between the closest objects from groups A and B
  - **Complete linkage:** the distance between the most distant objects from groups A and B
Impact of the agglomeration rule

- These four trees were built from the same distance matrix, using 4 different agglomeration rules.

**Note:** these trees were computed from a matrix of random numbers. The impression of structure is thus a complete artifact.
Hierarchical clustering result

Five clusters
K-mean clustering

Divisive (vs. Agglomerative)
K-mean clustering

- An algorithm for partitioning $n$ observations/points into $k$ clusters such that each observation belongs to the cluster with the nearest mean/center.
K-mean clustering: Chicken and egg

- An algorithm for partitioning $n$ observations/points into $k$ clusters such that each observation belongs to the cluster with the nearest mean/center.

- The chicken and egg problem:
  I do not know the means before I determine the partitioning into clusters.
  I do not know the partitioning into clusters before I determine the means.

- Key principle - cluster around mobile centers:
  - Start with some random locations of means/centers, partition into clusters according to these centers, and then correct the centers according to the clusters.
  [similar to EM (expectation-maximization) algorithms]
K-mean clustering algorithm

- The number of centers, $k$, has to be specified a-priori

**Algorithm:**

1. Arbitrarily select $k$ initial centers
2. Assign each element to the closest center
3. Re-calculate centers (mean position of the assigned elements)
4. Repeat 2 and 3 until ...
The number of centers, $k$, has to be specified a-priori.

**Algorithm:**

1. Arbitrarily select $k$ initial centers
2. Assign each element to the closest center
3. Re-calculate centers (mean position of the assigned elements)
4. Repeat 2 and 3 until one of the following termination conditions is reached:
   i. The clusters are the same as in the previous iteration
   ii. The difference between two iterations is smaller than a specified threshold
   iii. The maximum number of iterations has been reached

How can we do this efficiently?
Partitioning the space

- Assigning elements to the closest center
Partitioning the space

- Assigning elements to the closest center

Diagram showing points A and B.
Partitioning the space

- Assigning elements to the closest center

![Diagram showing partitioning of space with points A, B, and C and lines indicating which elements are closer to which centers.](image)
Partitioning the space

- Assigning elements to the closest center

Diagram showing three points A, B, and C, with lines partitioning the space into regions closest to A, B, and C.
Partitioning the space

- Assigning elements to the closest center
Voronoi diagram

- Decomposition of a metric space determined by distances to a specified discrete set of “centers” in the space.
- Each colored cell represents the collection of all points in this space that are closer to a specific center $s$ than to any other center.
- Several algorithms exist to find the Voronoi diagram.
K-mean clustering algorithm

- The number of centers, \( k \), has to be specified a priori

**Algorithm:**

1. Arbitrarily select \( k \) initial centers
2. Assign each element to the closest center *(Voronoi)*
3. Re-calculate centers (mean position of the assigned elements)
4. Repeat 2 and 3 until one of the following termination conditions is reached:
   - i. The clusters are the same as in the previous iteration
   - ii. The difference between two iterations is smaller than a specified threshold
   - iii. The maximum number of iterations has been reached
K-mean clustering example

- Two sets of points randomly generated
  - 200 centered on (0,0)
  - 50 centered on (1,1)
K-mean clustering example

- Two points are randomly chosen as centers (stars)
K-mean clustering example

- Each dot can now be assigned to the cluster with the closest center
K-mean clustering example

- First partition into clusters

```
iter.max = 1 ; iterations = 1
```
Centers are re-calculated.

K-mean clustering example

iter.max = 1 ; iterations = 1
K-mean clustering example

- And are again used to partition the points.
K-mean clustering example

- Second partition into clusters

iter.max = 2 ; iterations = 2
K-mean clustering example

- Re-calculating centers again

iter.max = 2; iterations = 2
K-mean clustering example

- And we can again partition the points

iter.max = 2 ; iterations = 2
K-mean clustering example

- Third partition into clusters
K-mean clustering example

- After 6 iterations:
- The calculated centers remains stable
K-mean clustering: Summary

- The convergence of k-mean is usually quite fast (sometimes 1 iteration results in a stable solution)

- K-means is time- and memory-efficient

**Strengths:**
- Simple to use
- Fast
- Can be used with very large data sets

**Weaknesses:**
- The number of clusters has to be predetermined
- The results may vary depending on the initial choice of centers
K-mean clustering: Variations

- Expectation-maximization (EM): maintains probabilistic assignments to clusters, instead of deterministic assignments, and multivariate Gaussian distributions instead of means.

- k-means++: attempts to choose better starting points.

- Some variations attempt to escape local optima by swapping points between clusters.
The take-home message

Hierarchical clustering

K-mean clustering

D’haeseleer, 2005
What else are we missing?
What else are we missing?

- What if the clusters are not “linearly separable”?
Cell cycle
Clustering methods

- We can distinguish between two types of clustering methods:
  1. **Agglomerative**: These methods build the clusters by examining small groups of elements and merging them in order to construct larger groups.
  2. **Divisive**: A different approach which analyzes large groups of elements in order to divide the data into smaller groups and eventually reach the desired clusters.

- There is another way to distinguish between clustering methods:
  1. **Hierarchical**: Here we construct a hierarchy or tree-like structure to examine the relationship between entities.
  2. **Non-Hierarchical**: In non-hierarchical methods, the elements are partitioned into non-overlapping groups.