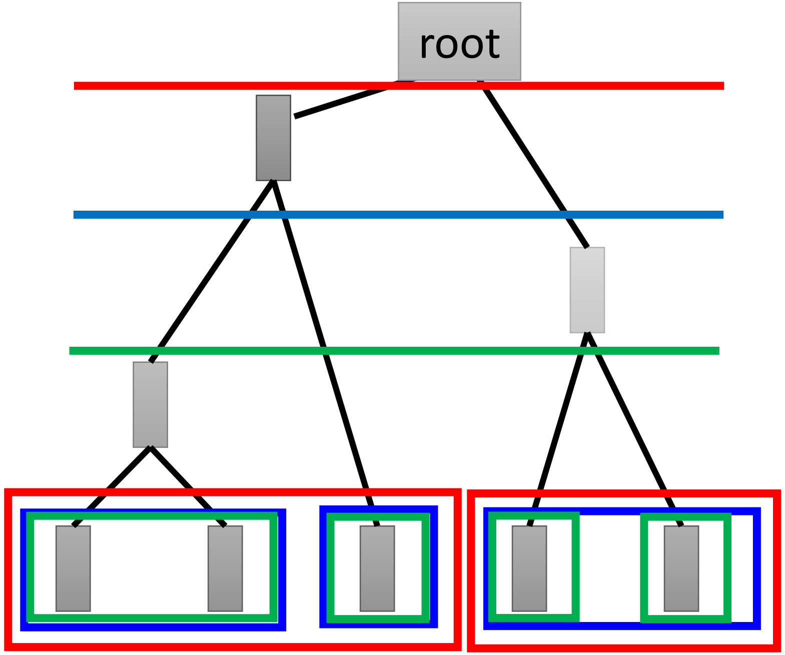
Clustering

Part III

Hierarchical Agglomerative Clustering (HAC)

Step 1: build a tree

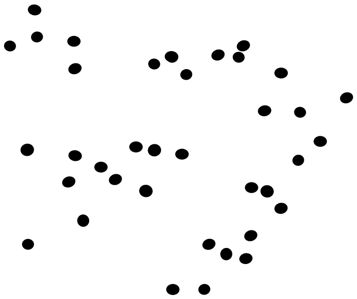
Step 2: pick a threshold



Agglomerative Clustering: Part I

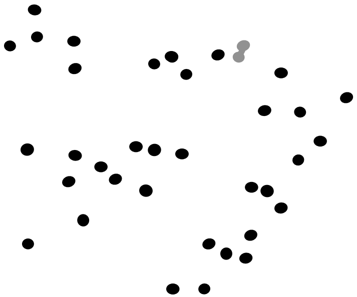
1. Start with *n* clusters (each record is its own cluster)
2. Merge two closest records into one cluster
3. At each successive step, the two clusters closest to each other are merged
4. Finish when the desired number of clusters is reached

Agglomerative Clustering: Part II



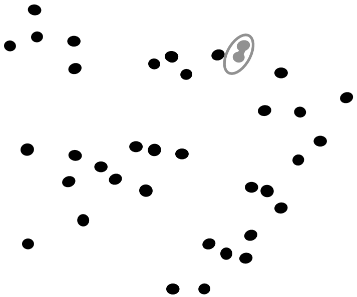
1. Say “Every point is its own cluster”

Agglomerative Clustering: Part III

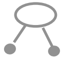


1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters

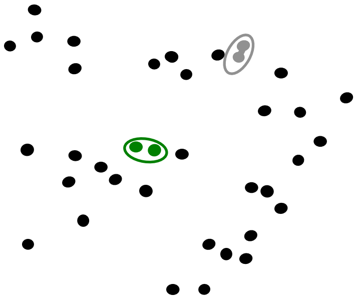
Agglomerative Clustering: Part IV



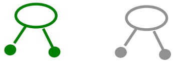
1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster



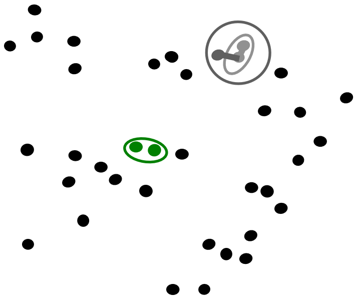
Agglomerative Clustering: Part V



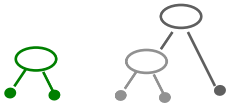
1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster
4. Repeat



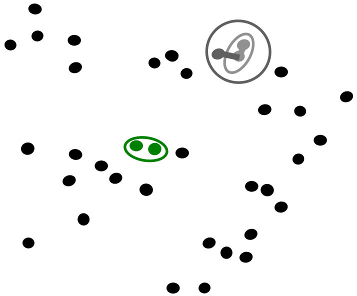
Agglomerative Clustering: Part VI



1. Say “Every point is its own cluster”
2. Find “most similar” pair of clusters
3. Merge it into a parent cluster
4. Repeat



Agglomerative Clustering: Part VII



1. Say "Every point is its own cluster"
2. Find "most similar" pair of clusters
3. Merge it into a parent cluster
4. Repeat…until you've merged the whole dataset into one cluster

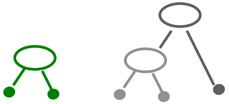
Based **on linkage**:

- Single linkage
- Complete linkage
- Average linkage
- Ward

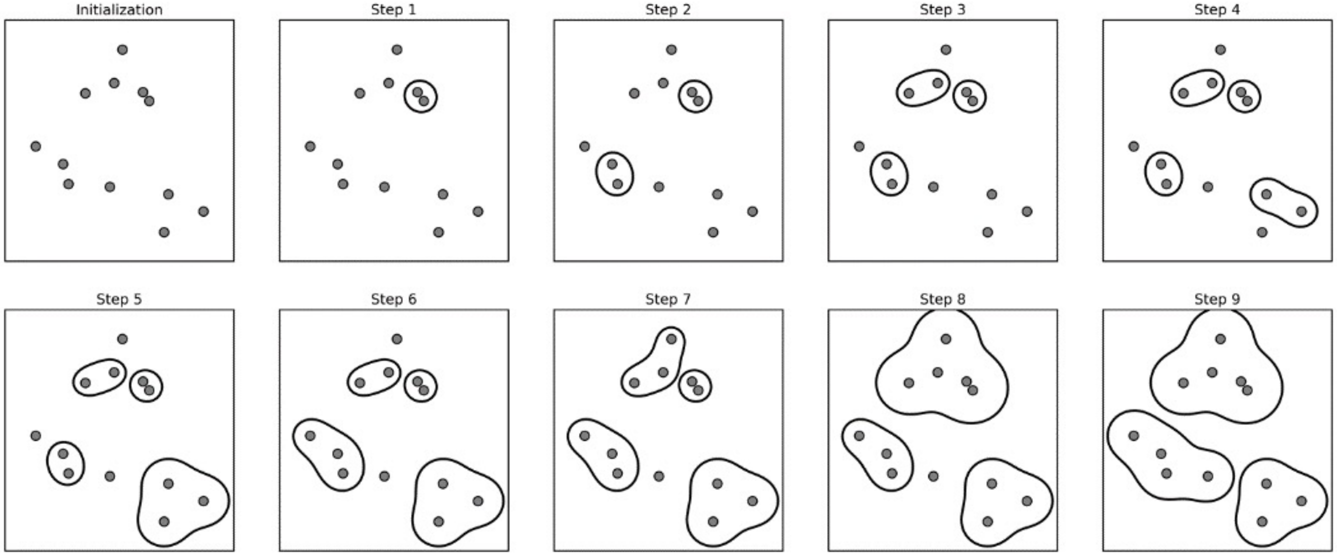
And **on distance:**

- Euclidean
- Manhattan
- Hamming
- …

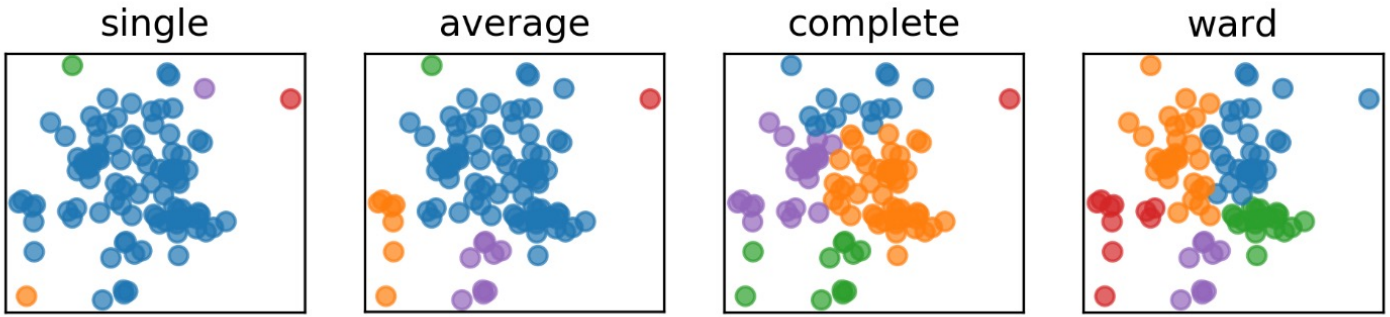
The algorithm will stop when it reaches the desired number of clusters



Another Example



Linkage Criteria



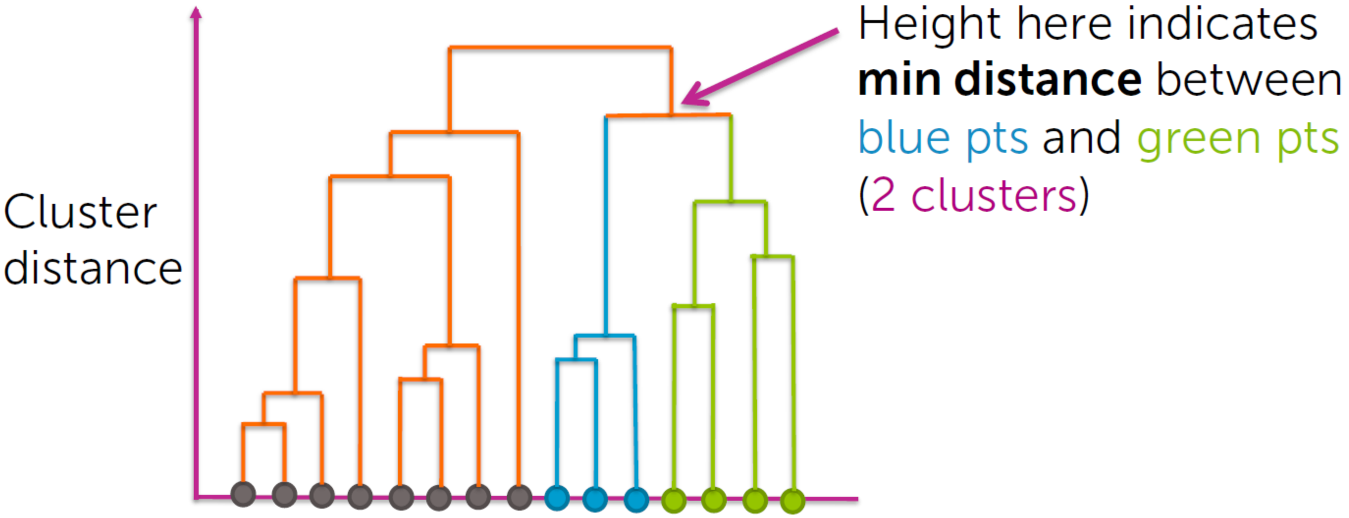
Cluster sizes:

```
single :    [96  1  1  1  1]
average :   [82  9  7  1  1]
complete : [50 24 14 11  1]
ward :      [31 30 20 10  9]
```

- Single Linkage
  - Smallest minimum distance
- Average Linkage
  - Smallest average distance between all pairs in the clusters
- Complete Linkage
  - Smallest maximum distance
- Ward (default in sklearn)
  - Smallest increase in within-cluster variance
  - Leads to more equally sized clusters.

The Dendrogram: Part I

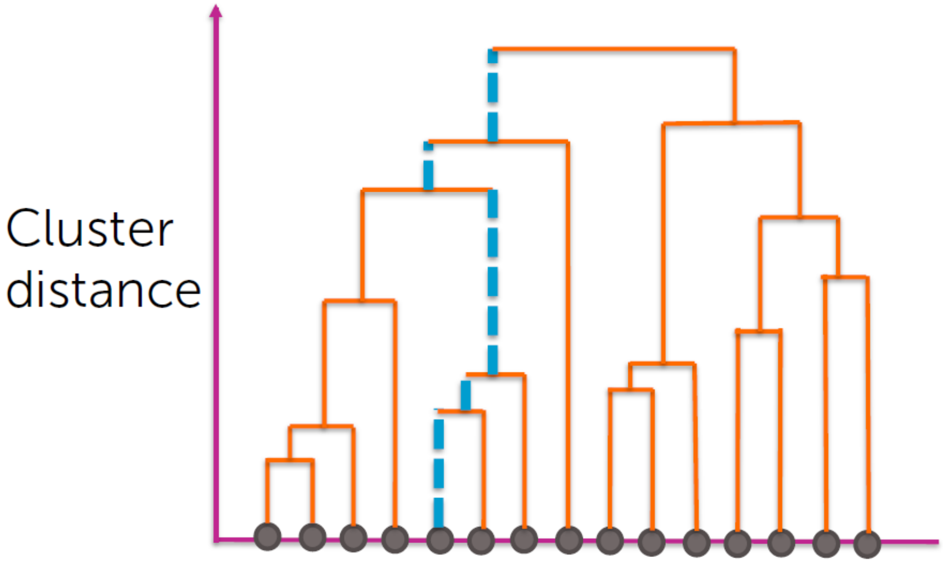
- x axis shows data points (carefully ordered)
- y-axis shows distance between pair of clusters



Height here indicates **min distance** between blue pts and green pts (2 clusters)

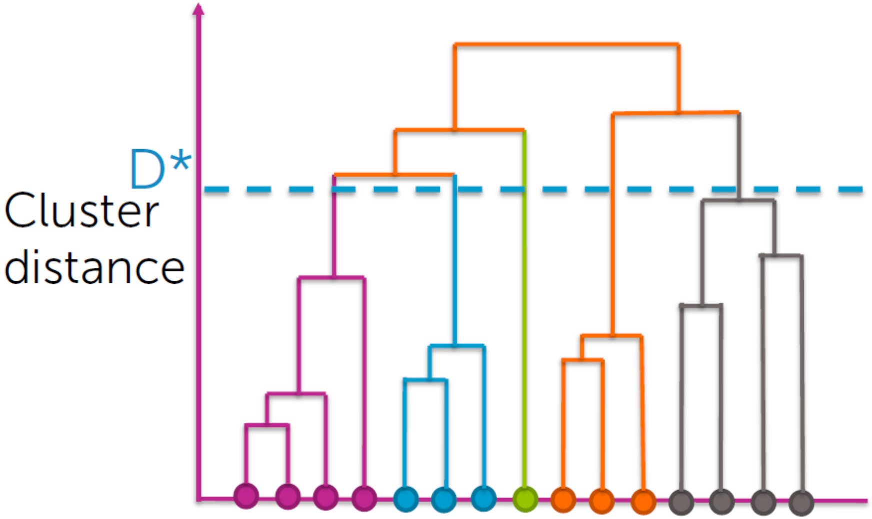
The Dendrogram: Part II

Path shows all clusters to which a point belongs and the order in which clusters merge

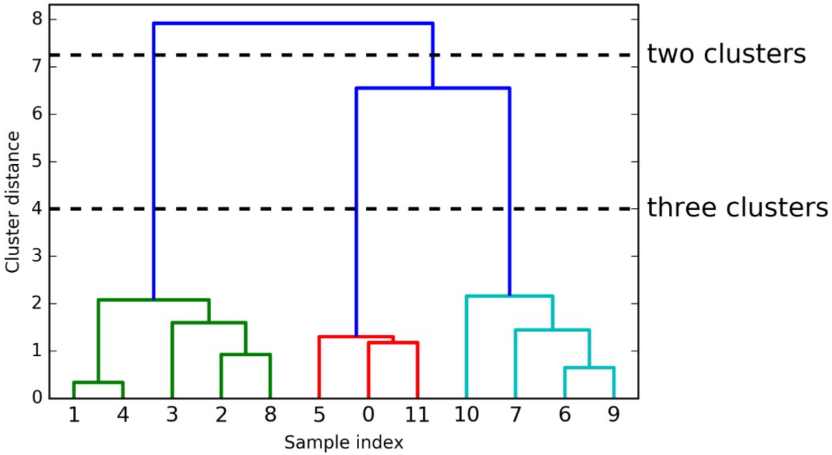


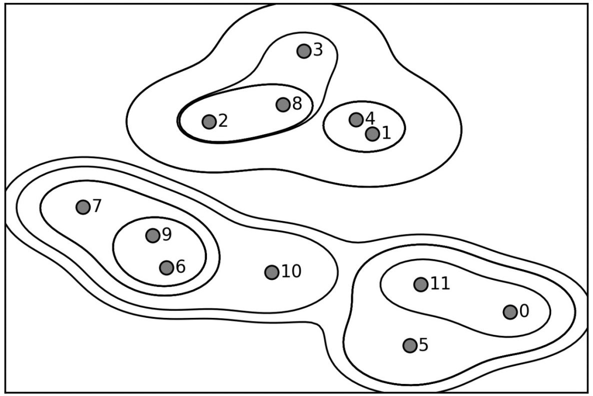
The Dendrogram: Part III

Every branch that crosses D* becomes a separate cluster



The Dendrogram: Part IV


two clusters
three clusters
Cluster distance
Sample index


3
8
2
4
1
7
9
6
10
11
0
5