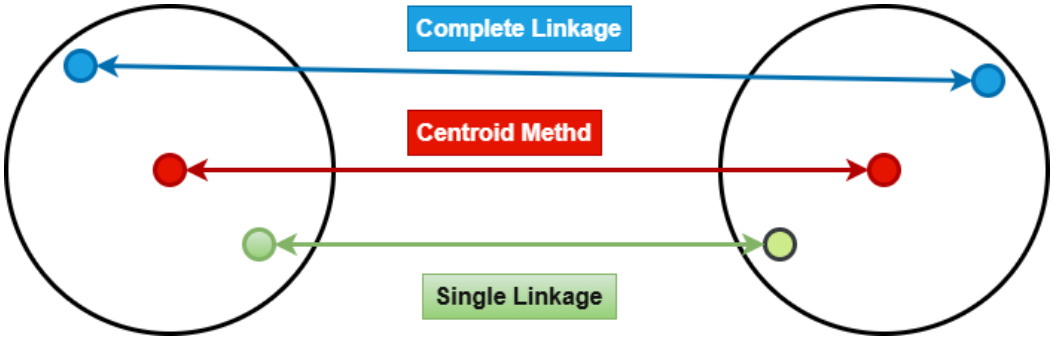
Hierarchical Agglomerative Clustering

- **Introduction:**
  - Use distance matrix as clustering criteria.
  - This method does not require the number of clusters k as an input but needs a termination condition.
  - Need to define a distance d(Ci,Cj) between two clusters Ci and Cj.
  - Commonly used distance measures:
    - Single Linkage
    - Complete Linkage
    - Average Linkage
    - Centroid Method
    - Ward distance



- **Single Linkage:**
  - In single linkage, the distance between two clusters is defined as the minimum distance between any point in one cluster and any point in the other cluster.
  - This results in long, chain-like clusters, as it allows clusters to be joined based on the nearest points.
  - The **distance** d(Ci,Cj) between two clusters Ci and Cj is defined as:

$$d(Ci,Cj)=\min\{d(x,y) \mid x\in Ci, y\in Cj\}$$

where:

x and y are points in clusters Ci and C_j respectively.

d(x,y) is the Euclidean distance (or another metric) between the points.

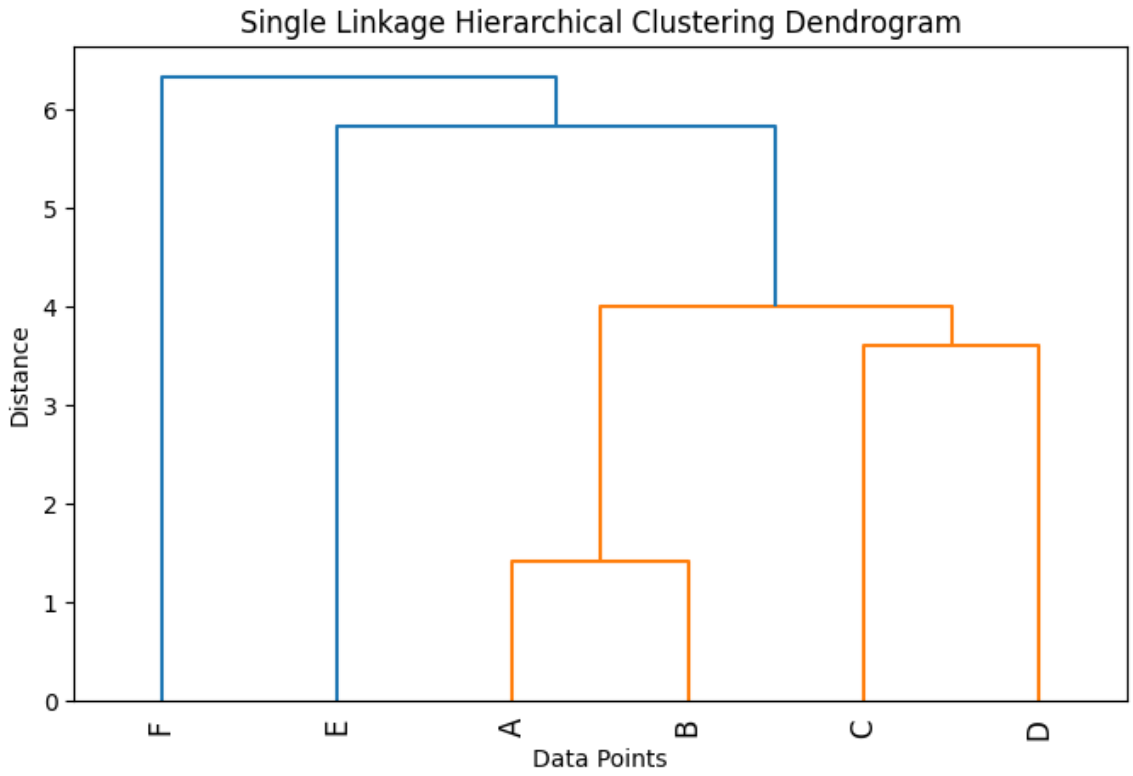
- Algorithm:
  - Initialize: Treat each data point as its own cluster.
  - Compute Distances: Calculate the distance between all pairs of clusters using the shortest distance between any two points (one from each cluster).
  - Merge Clusters: Merge the two clusters that have the smallest minimum distance.
  - Update Distances: Recompute distances between the new cluster and remaining clusters.
  - Repeat until all data points belong to a single cluster or the desired number of clusters is reached.
- Example:

A	1	3
B	2	4
C	5	10
D	2	8
E	8	5
F	11	12

  - Distance matrix:

	A(1,3)	**B(2,4)**	**C(5,10)**	**D(2,8)**	**E(8,5)**	**F(11,12)**
A(1,3)	0	1.41	7.62	5.00	7.07	13.60
B(2,4)	1.41	0	6.40	4.00	6.08	12.04
C(5,10)	7.62	6.40	0	3.61	5.83	6.40
D(2,8)	5.00	4.00	3.61	0	6.32	9.43
E(8,5)	7.07	6.08	5.83	6.32	0	7.62

	A(1,3)	B(2,4)	C(5,10)	D(2,8)	E(8,5)	F(11,12)
F(11,12)	13.60	12.04	6.40	9.43	7.62	0



- **Complete Linkage:**
  - The distance between two clusters is defined as the maximum distance between any pair of points—one from each cluster.
  - Join the two clusters with the nearest farthest neighbors.
  - Makes "tighter," spherical clusters that are typically preferable.
  - The **distance** d(Ci,Cj) between two clusters Ci and Cj is defined as:

$$\mathrm{d(Ci, Cj)} = \max\{(d(x, y)\ where\ x \in \mathrm{Ci, and\ y\ Cj}\}$$

  - Algorithm:
    - Start with each point as its own cluster.
    - Find the two closest clusters based on the maximum pairwise distance.

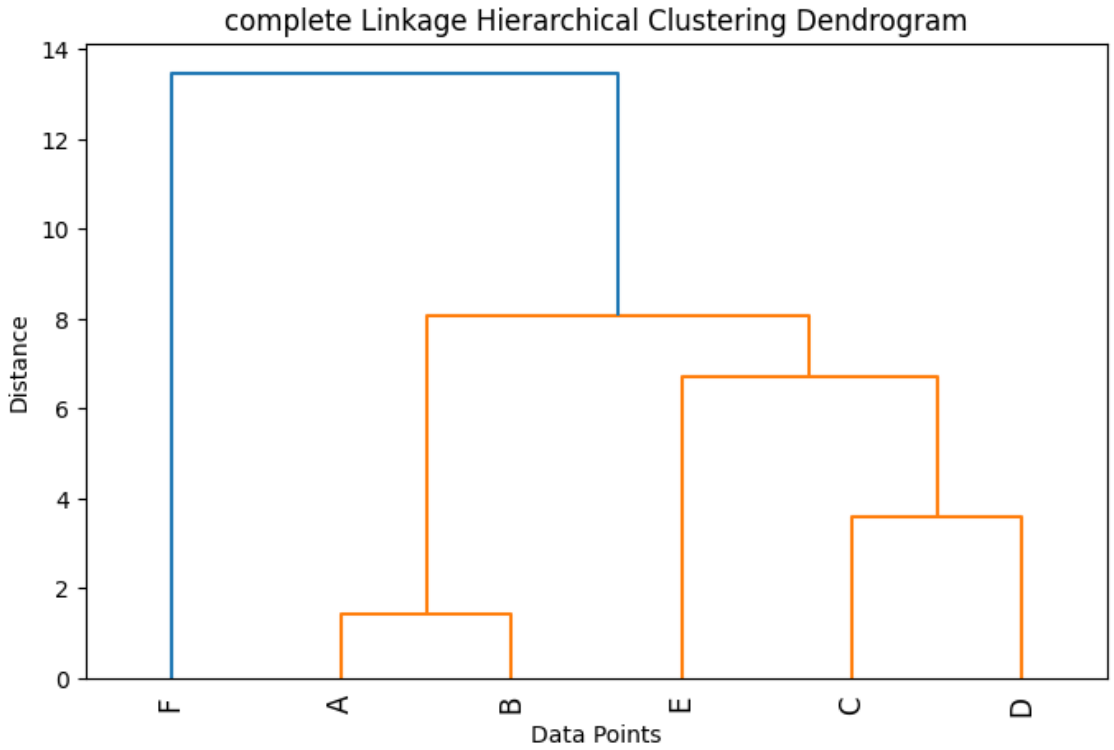
- Merge them into a single cluster.
- Repeat steps 2-3 until only one cluster remains.

- Drawback: Sensitive to outliers
  - A single far-out point can influence cluster merging since it relies on the maximum distance.
- Example:

A	1	3
B	2	4
C	5	10
D	2	8
E	8	5
F	11	12

- Distance matrix:

	A(1,3)	**B(2,4)**	**C(5,10)**	**D(2,8)**	**E(8,5)**	**F(11,12)**
A(1,3)	0	1.41	7.62	5.00	7.07	13.60
B(2,4)	1.41	0	6.40	4.00	6.08	12.04
C(5,10)	7.62	6.40	0	3.61	5.83	6.40
D(2,8)	5.00	4.00	3.61	0	6.32	9.43
E(8,5)	7.07	6.08	5.83	6.32	0	7.62
F(11,12)	13.60	12.04	6.40	9.43	7.62	0



- **Average Linkage:**
  - The distance between two clusters is defined as the average distance between all pairs of points, where one point belongs to one cluster and the other belongs to the second cluster.
  - Reduces extreme cases of single linkage (long chains) and complete linkage (tight clusters).
  - Produces moderately compact and well-separated clusters.
  - The **distance** d(Ci,Cj) between two clusters Ci and Cj is defined as:

$$\mathrm{d(Ci, Cj)} = \frac{1}{|C_i||C_j|} \sum_{x \in, C_i, y \in C_j} d(x, y)$$

  - Algorithm:
    - Initialize: Each data point starts as a cluster.
    - Compute Distances: The distance between two clusters is calculated as the average of all pairwise distances between points in the two clusters.

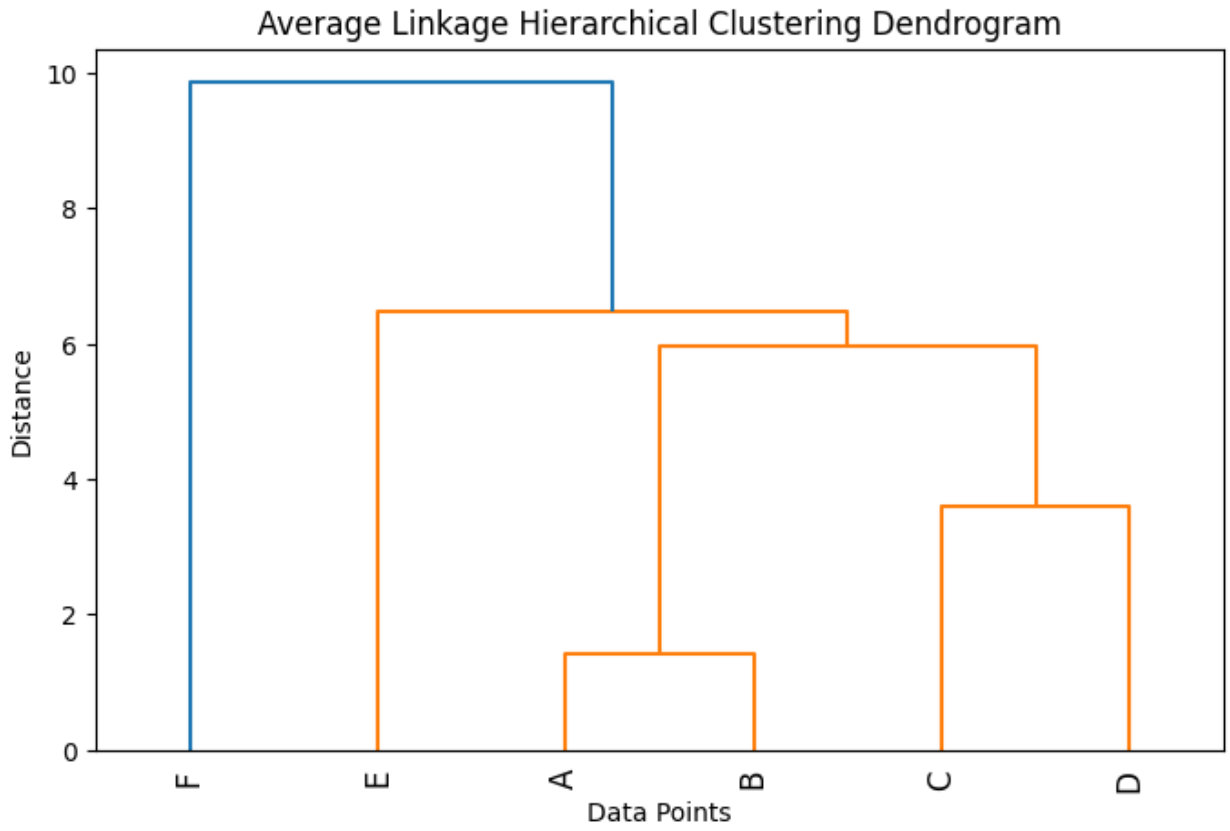
- Merge Clusters: The two clusters with the smallest average distance are merged.
- Update Distances: Recompute distances between the new cluster and remaining clusters.
- Repeat until all data points belong to a single cluster or the desired number of clusters is reached.

- Example:

A	1	3
B	2	4
C	5	10
D	2	8
E	8	5
F	11	12

- Distance matrix:

	A(1,3)	**B(2,4)**	**C(5,10)**	**D(2,8)**	**E(8,5)**	**F(11,12)**
A(1,3)	0	1.41	7.62	5.00	7.07	13.60
B(2,4)	1.41	0	6.40	4.00	6.08	12.04
C(5,10)	7.62	6.40	0	3.61	5.83	6.40
D(2,8)	5.00	4.00	3.61	0	6.32	9.43
E(8,5)	7.07	6.08	5.83	6.32	0	7.62
F(11,12)	13.60	12.04	6.40	9.43	7.62	0



- **Centroid Method:**
  - The Centroid Method, also known as Unweighted Pair Group Method with Centroid Mean (UPGMC), is a hierarchical clustering technique where clusters are merged based on the distance between their centroids (mean of all points in the cluster).
  - Creates balanced clusters (avoids chaining like single linkage).
  - It tends to create compact and well-separated clusters.
  - The **distance** $d(C_i,C_j)$ between two clusters C_i and C_j is defined as:

$$d(Ci, Cj) = \|centroid_i - centroid_j\|$$

Where

$centroid_i$ is the centroid of cluster C_i

and $centroid_j$ is the centroid of cluster C_j

- Algorithm:
  - Initialize: Start with each data point as a cluster.
  - Calculate Distances: Compute the distance between the centroids of all clusters.
  - Merge Clusters: At each step, merge the two clusters whose centroids are closest.
  - Update Centroid: Recalculate the centroid of the newly formed cluster.
  - Repeat until all points belong to a single cluster or the desired number of clusters is reached.

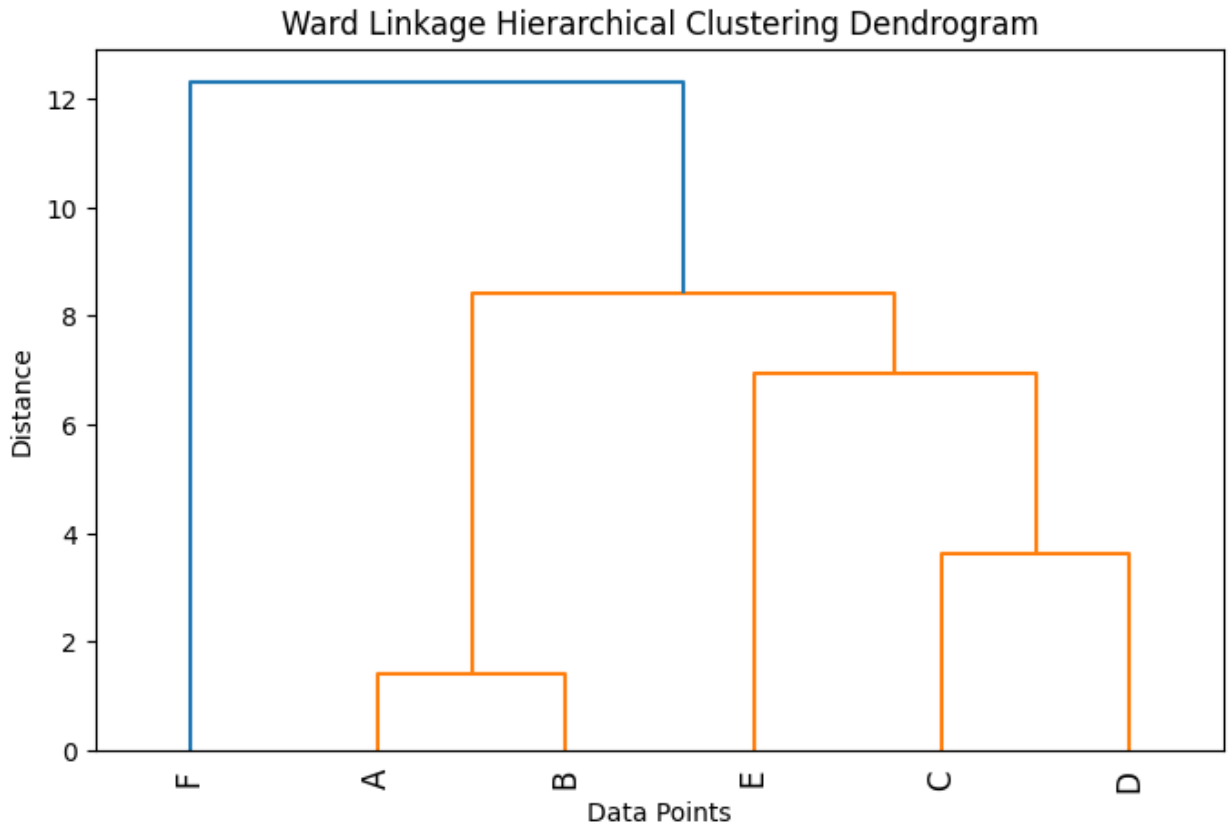
- **Ward's method:**
  - Another method for merging clusters.
  - It takes into account the number of data points in the cluster.
  - The **distance** d(Ci,Cj) between two clusters Ci and Cj is defined as:

$$\mathrm{d(Ci, Cj)} = 2\frac{|C_i||C_j|}{|C_i| + |C_j|}\left\|centroid_i - centroid_j\right\|^2$$

Where

$centroid_i$ is the centroid of cluster C_i

and $centroid_j$ is the centroid of cluster C_j


Ward Linkage Hierarchical Clustering Dendrogram
Distance
0
2
4
6
8
10
12
F
A
B
E
C
D
Data Points