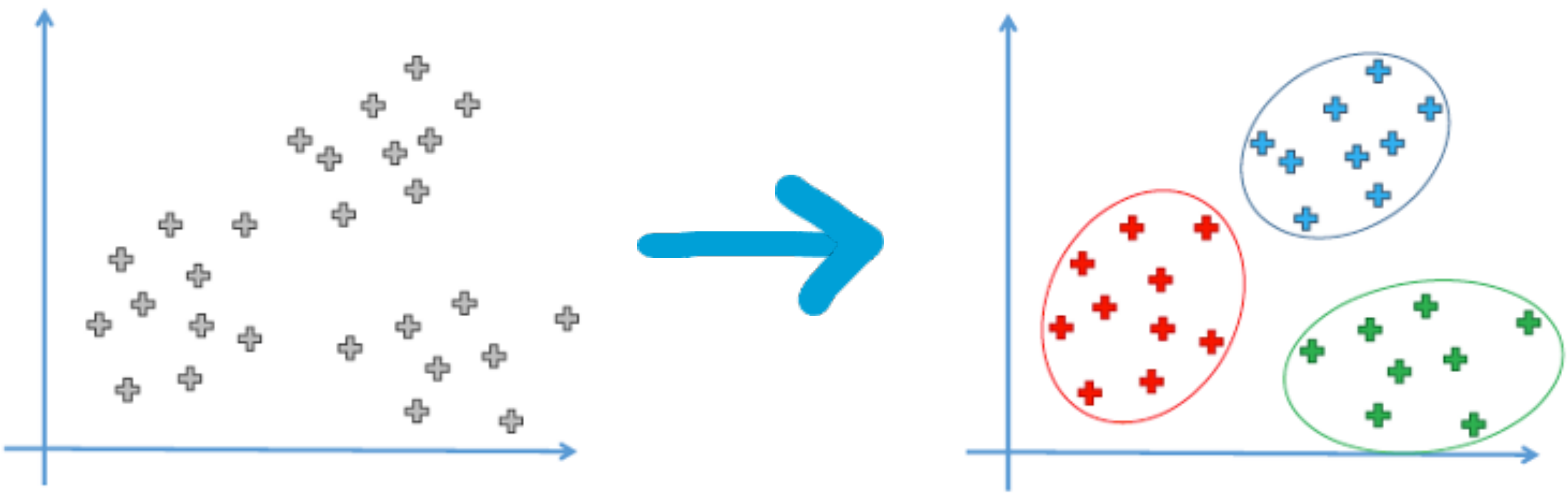
Hierarchical Clustering

Hierarchical Clustering



Same as k-mean clustering but apply different process.

Two methods: Agglomerative & Divisive

Agglomerative HC

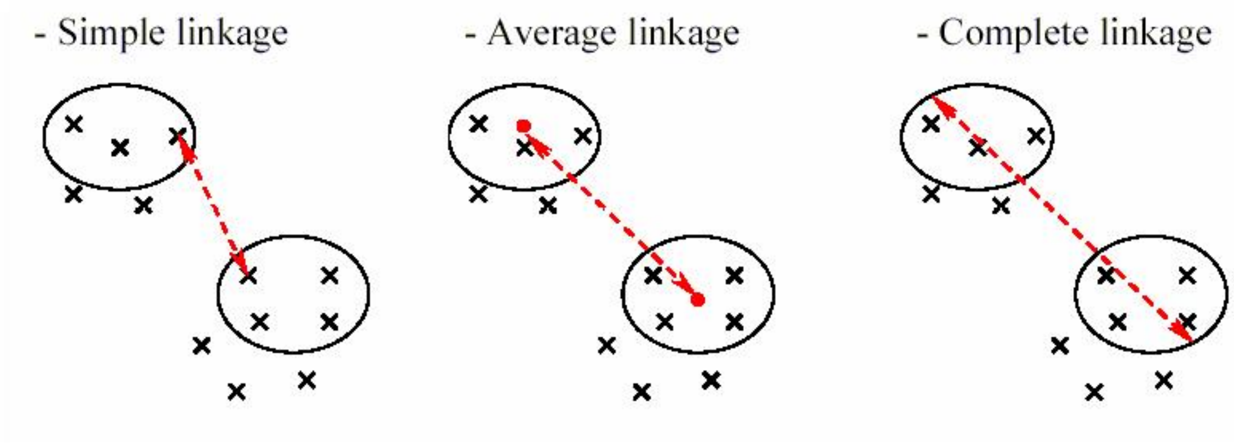
- Step 1: make each data point a single point cluster (which forms N clusters)
- Step 2: take the two closest data points and make them into one cluster (which forms N-1 clusters)
- Step 3: take the two closest clusters and make them into one cluster (which forms N-2 clusters)
- Step 4: repeat step 3 until there on only one cluster

Done!

Distance between clusters

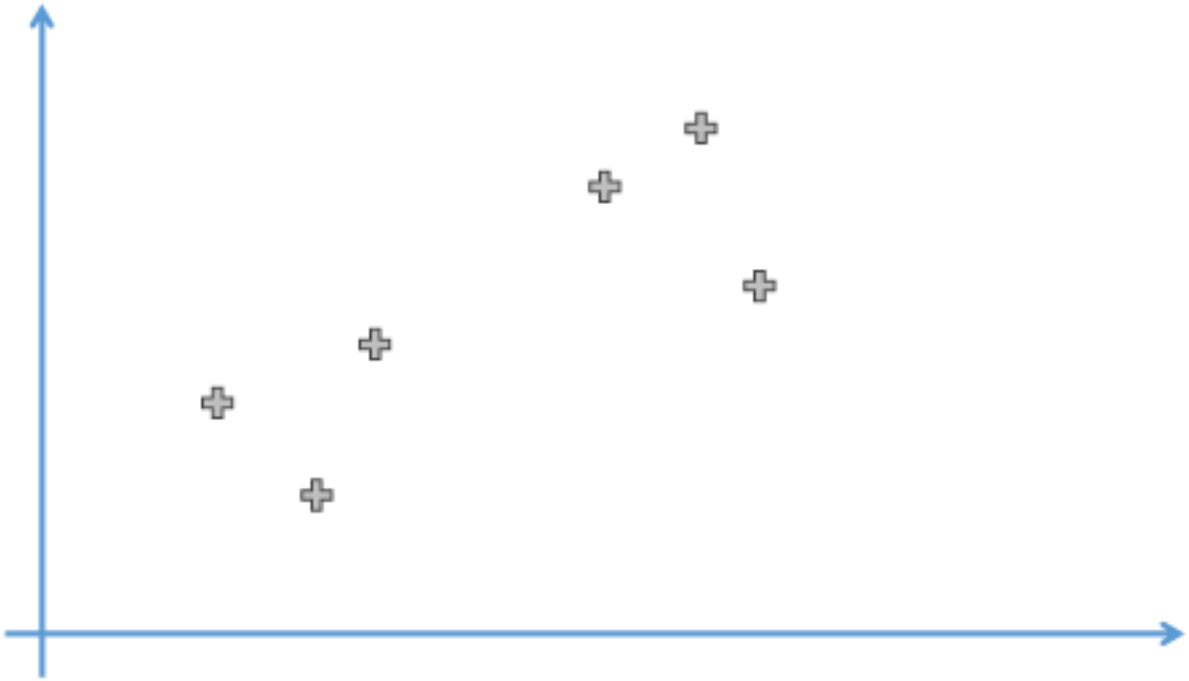
How to find distance between two clusters:

- Option 1: closest points - **Simple (single) Linkage**
- Option 2: furthest points - **Complete Linkage**
- Option 3: distance between centroids - **Average Linkage**



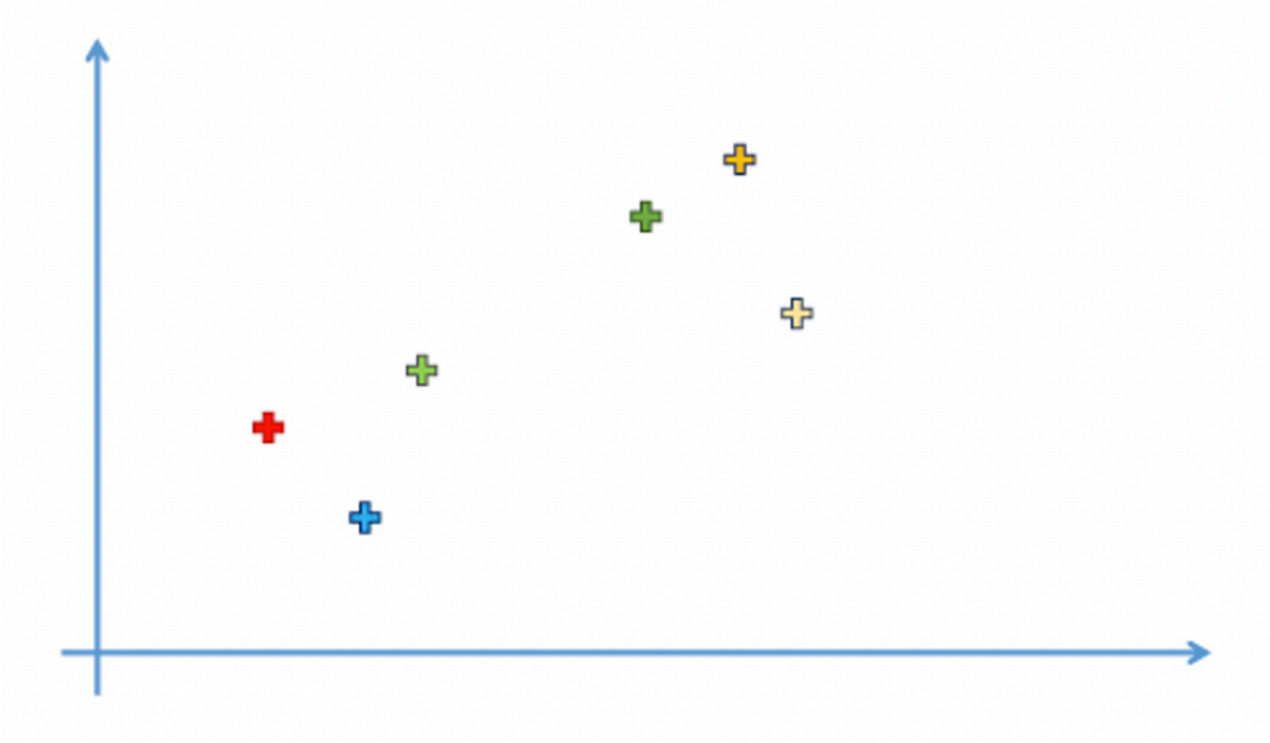
Agglomerative HC

Consider the following data points (N=6)



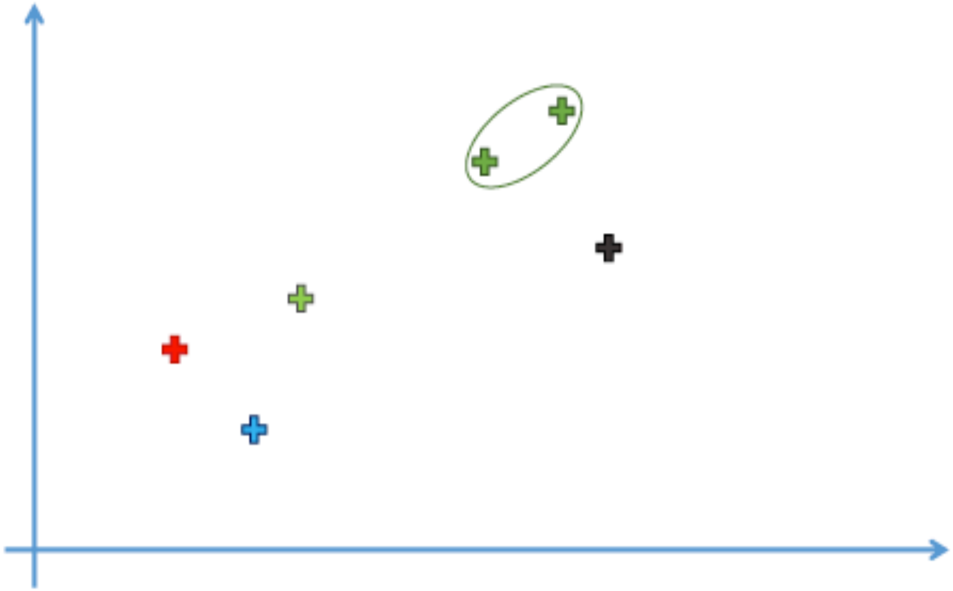
Agglomerative HC

Step 1: make each data point a single point cluster (6 clusters will be formed)



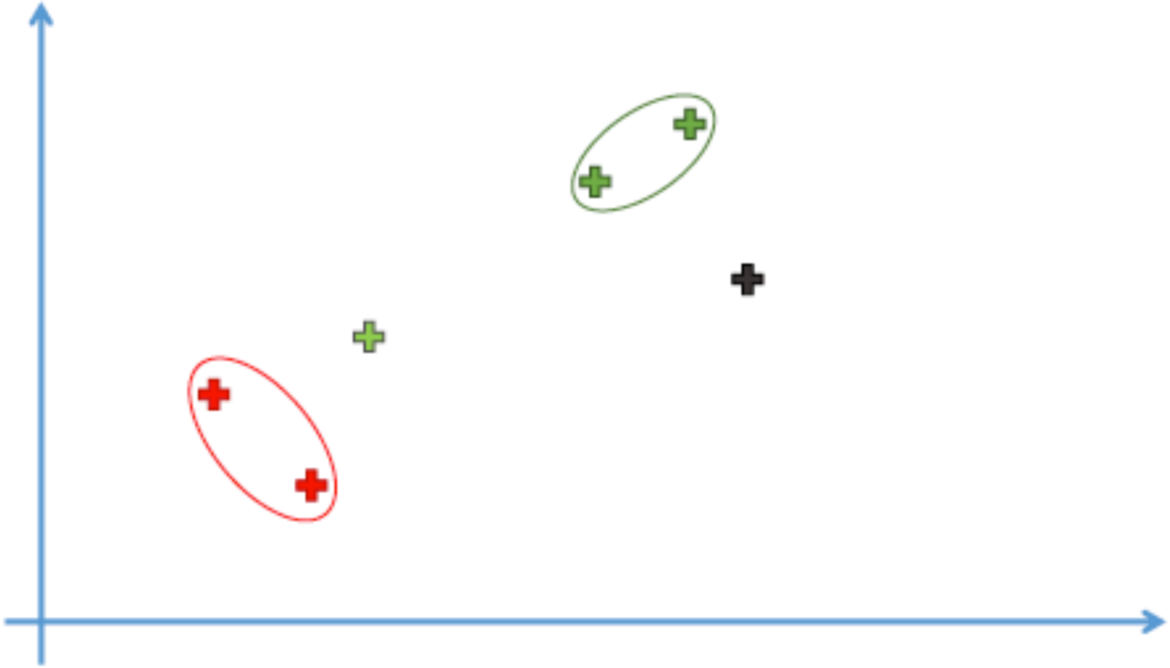
Agglomerative HC

Step 2: take the two closest data points and make them into one cluster (5 clusters will be formed)



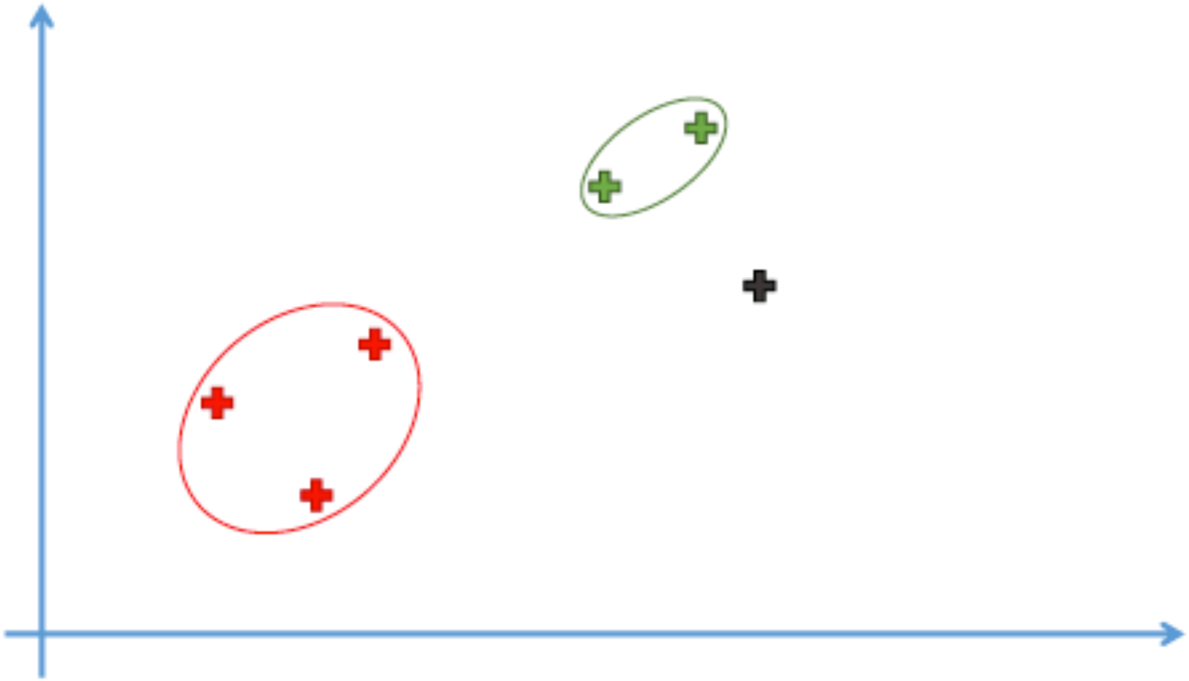
Agglomerative HC

Step 3: take the two closest clusters and make them into one cluster (4 clusters will be formed)



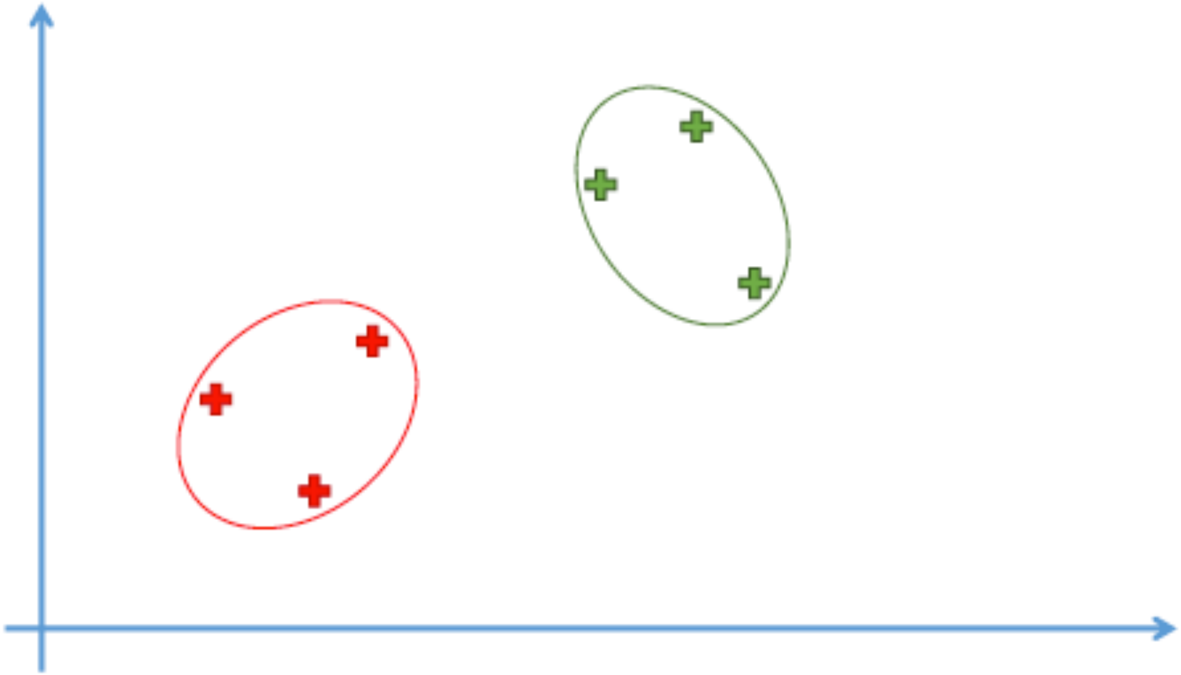
Agglomerative HC

Step 4: repeat step 3 until there is only one cluster



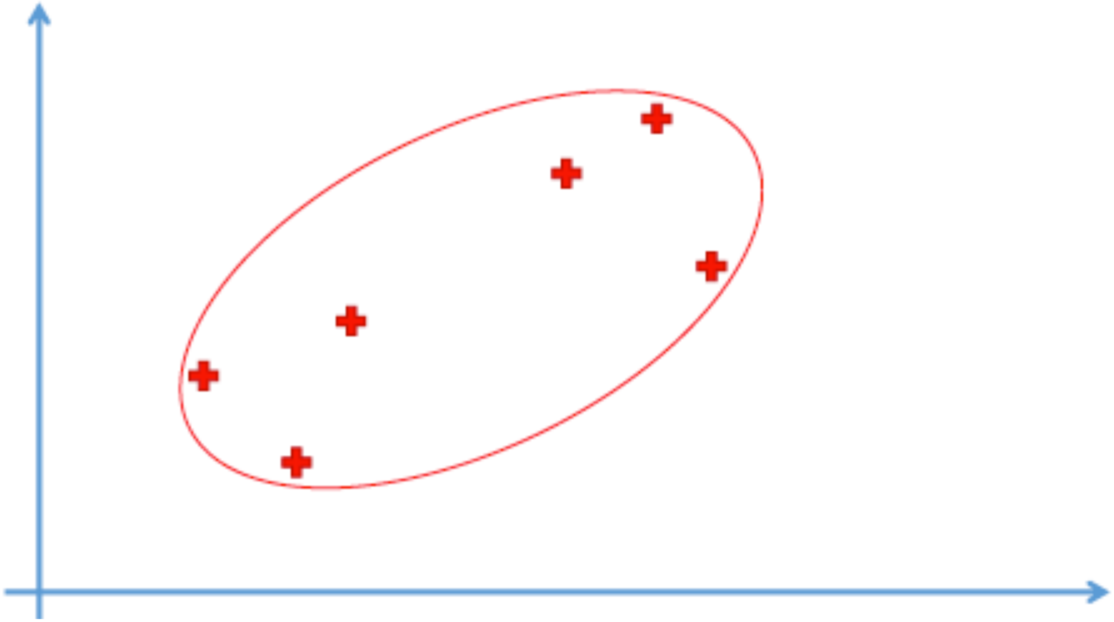
Agglomerative HC

Step 4: repeat step 3 until there is only one cluster.



Agglomerative HC

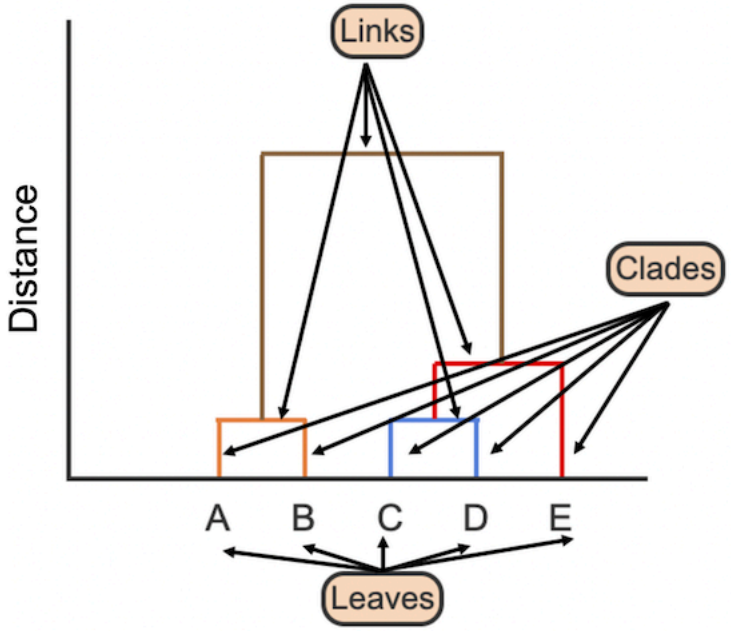
Step 4: keep repeating step 3 until there is only one cluster



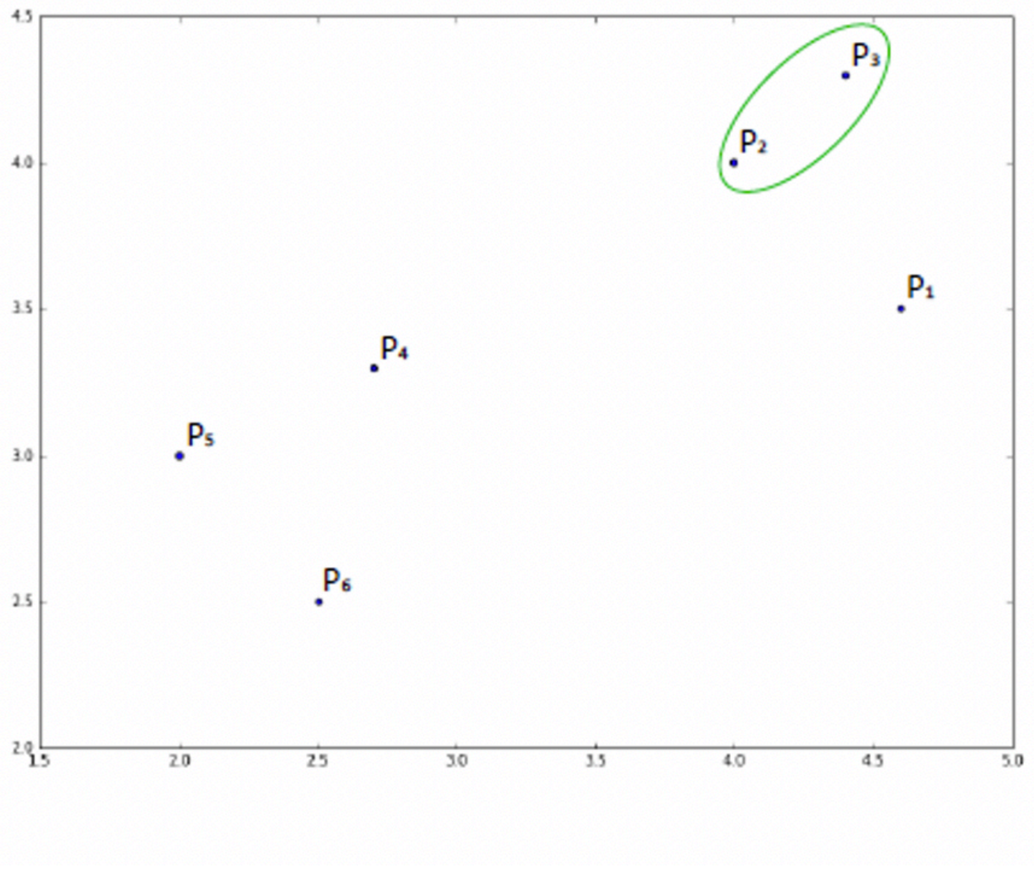
Dendograms

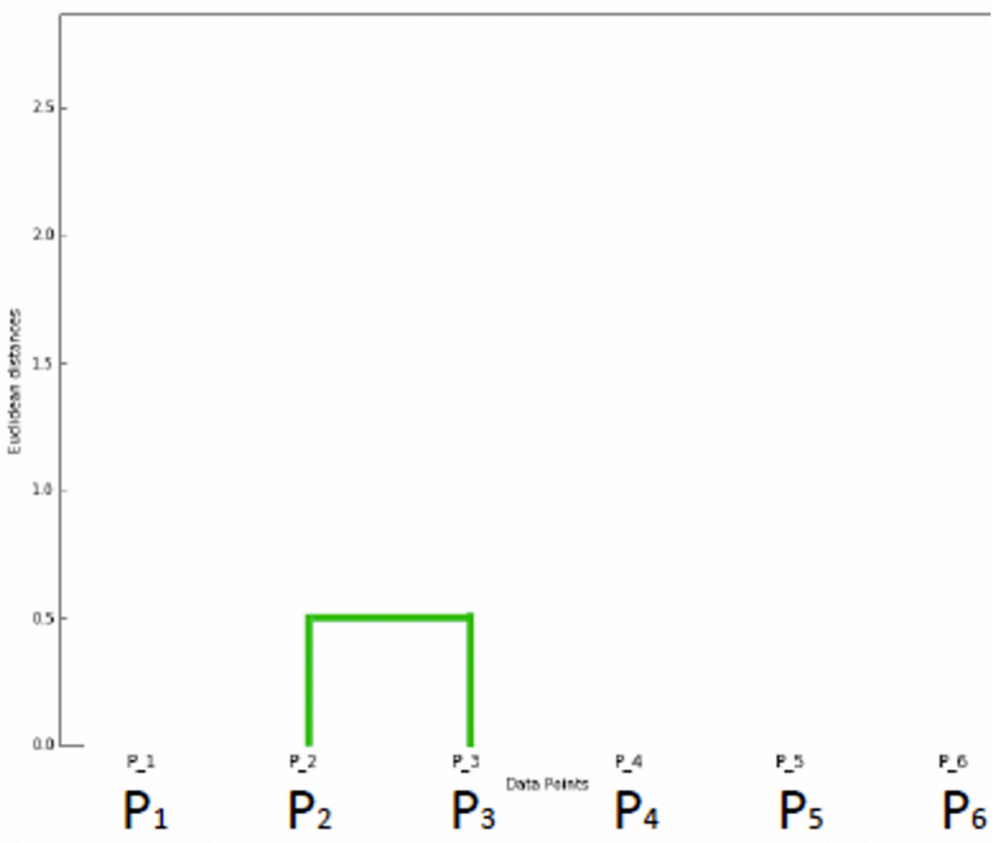
The output of a hierarchical clustering algorithm is a dendrogram. A dendrogram is a tree that shows the order in which clusters are grouped together and the distances between clusters. Parts of a dendrogram are listed below:

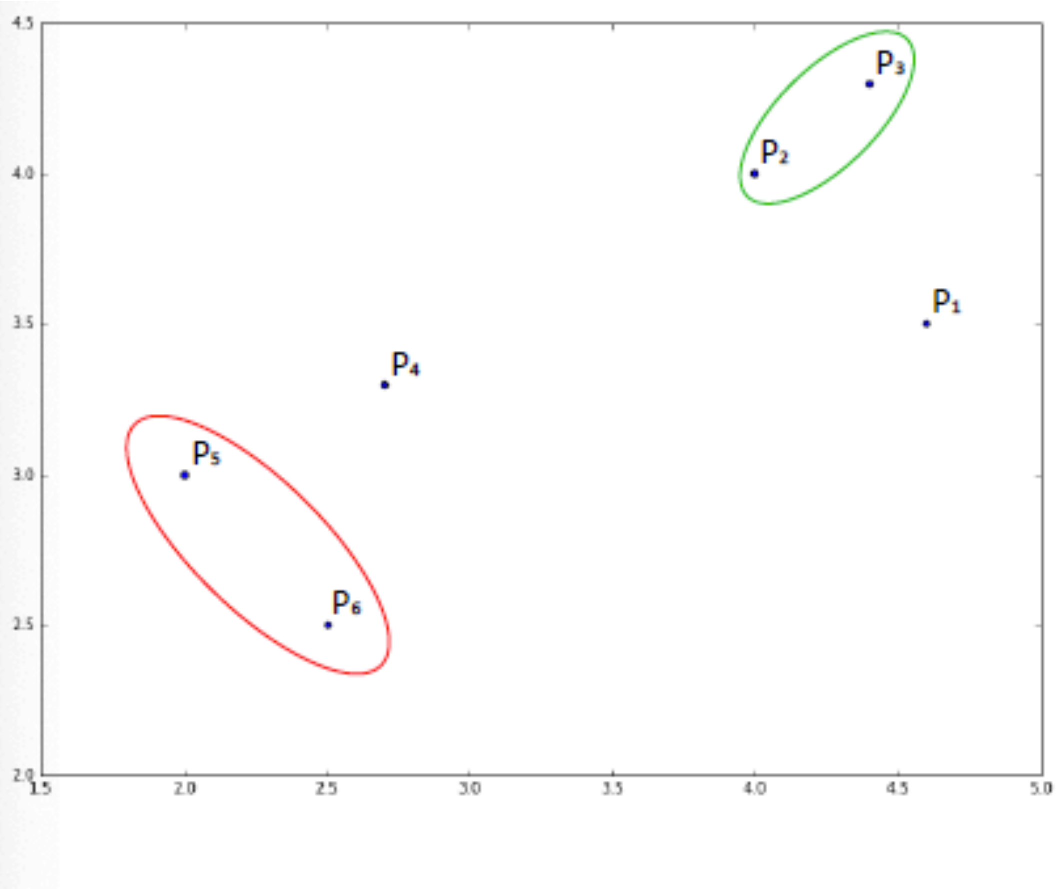
- A **clade** is a branch of a dendrogram or a vertical line.
- A **link** is a horizontal line that connects two clades, whose height gives the distance between clusters.
- A **leaf** is the terminal end of each clade in a dendrogram, which represents a single instance.



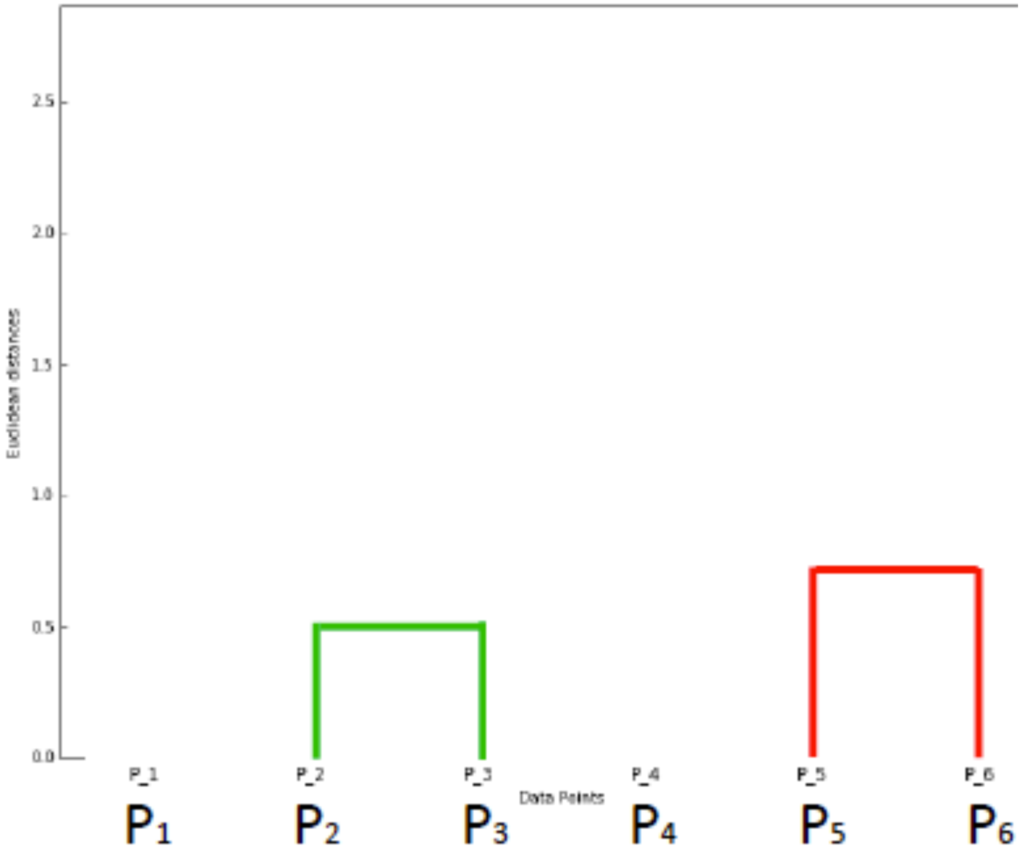
HC: How Do Dendrograms work?



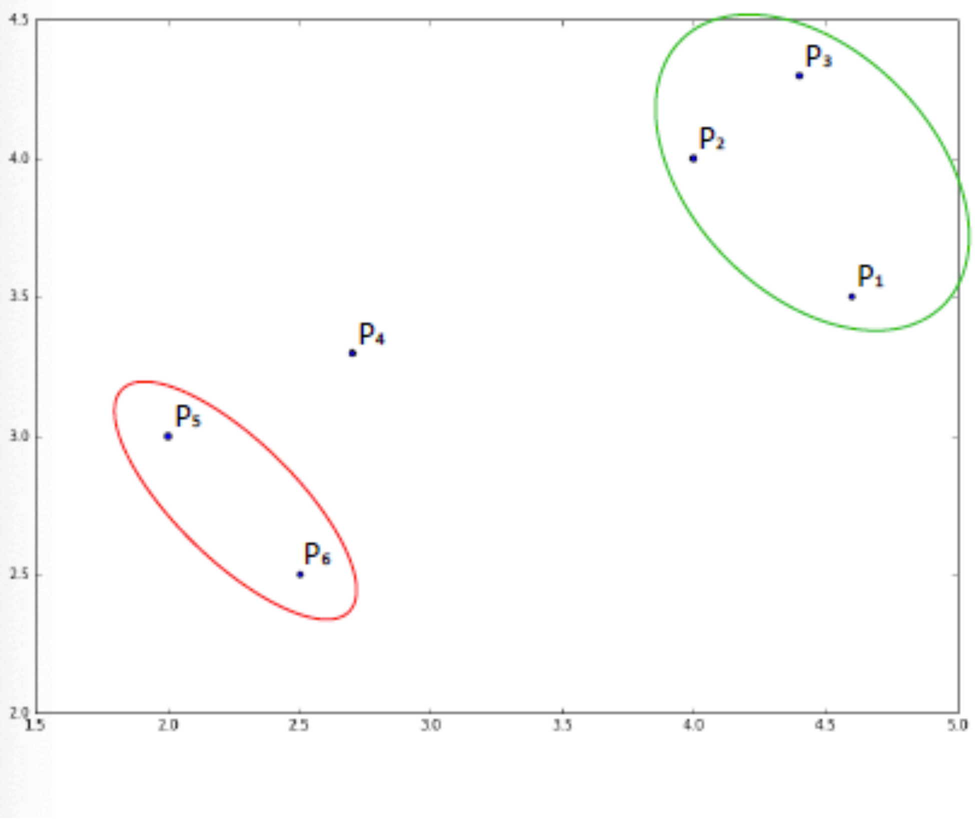


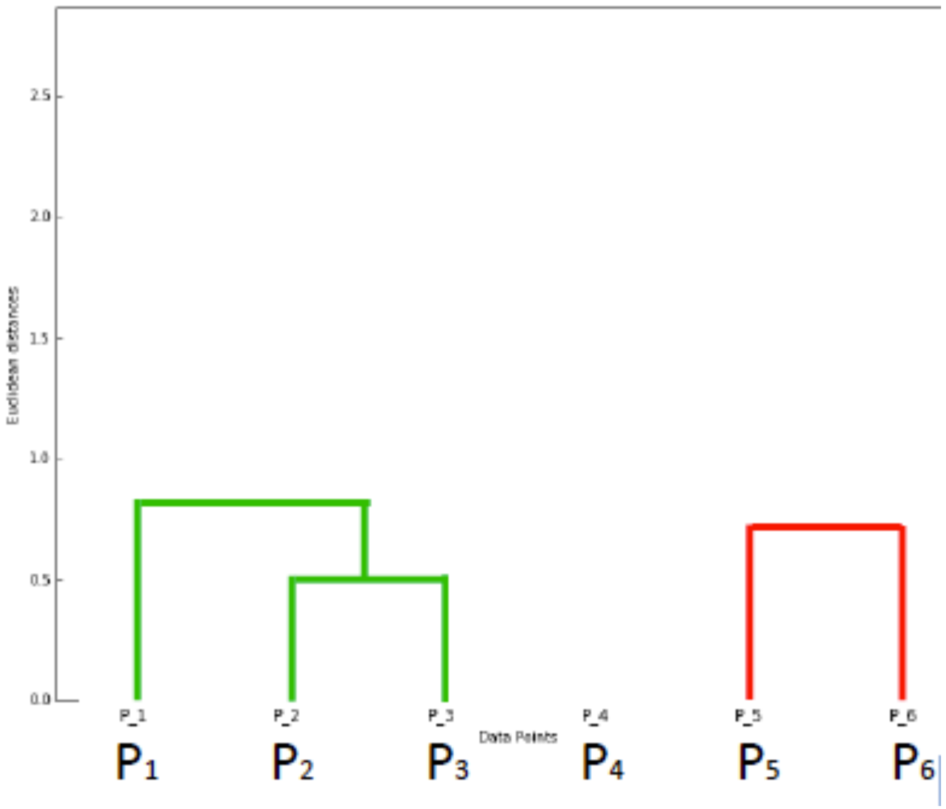


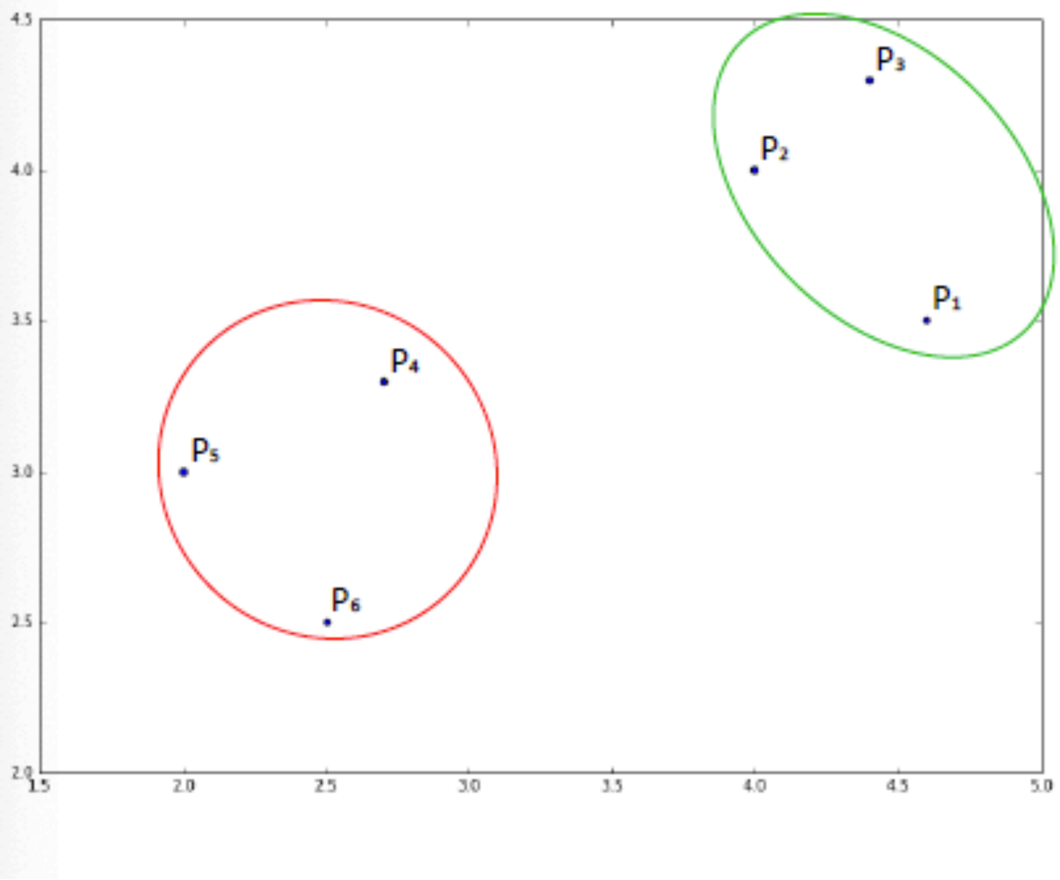
P₁
P₂
P₃
P₄
P₅
P₆
4.5
4.0
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

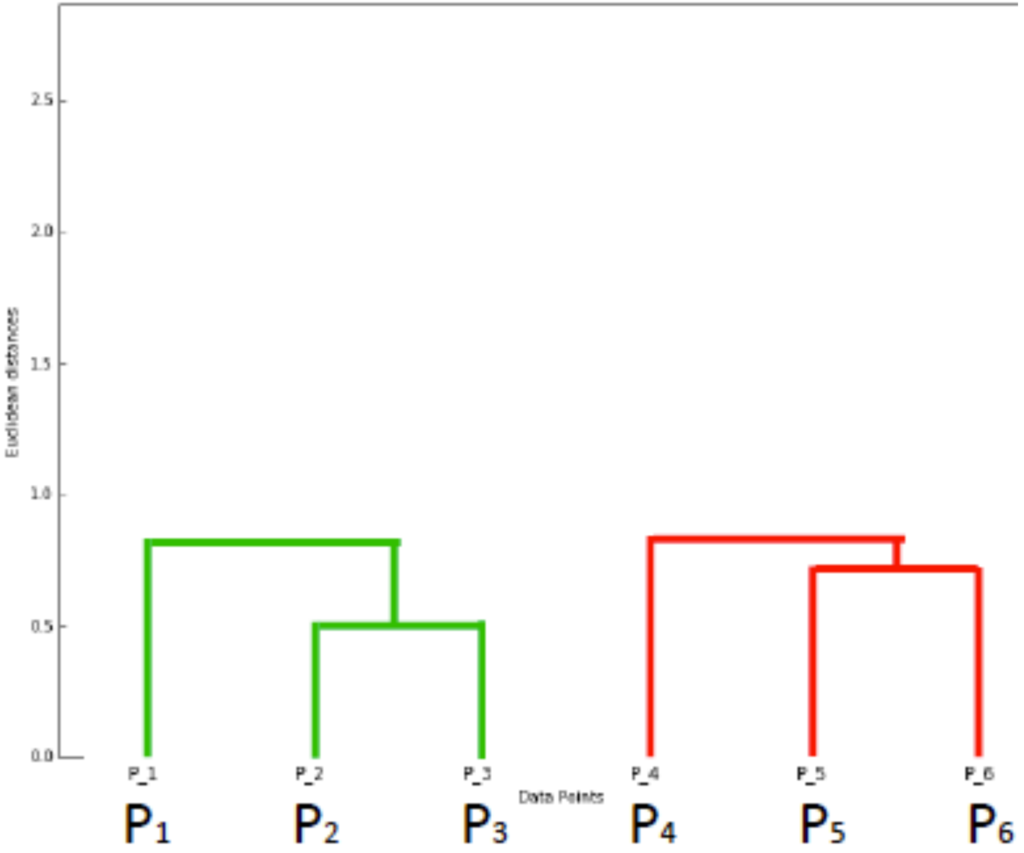


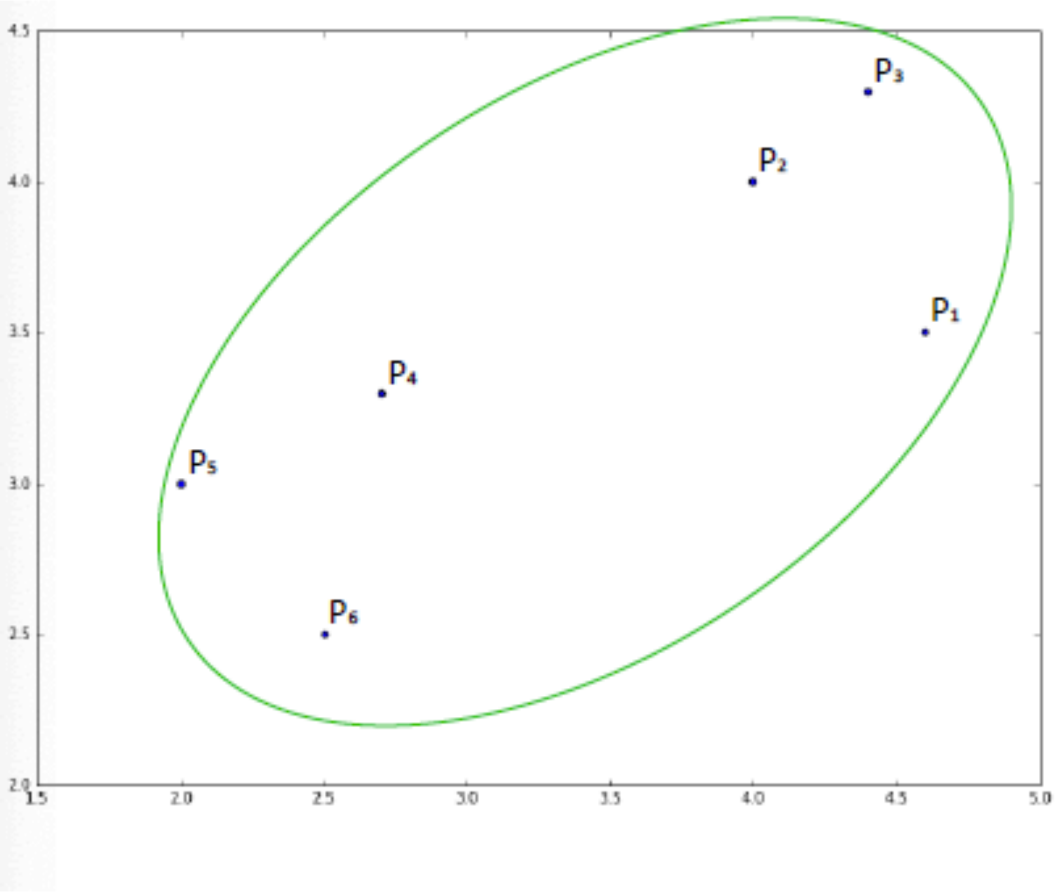
Euclidean distances
2.5
2.0
1.5
1.0
0.5
0.0
P_1
P_2
P_3
P_4
P_5
P_6
Data Points
P₁
P₂
P₃
P₄
P₅
P₆

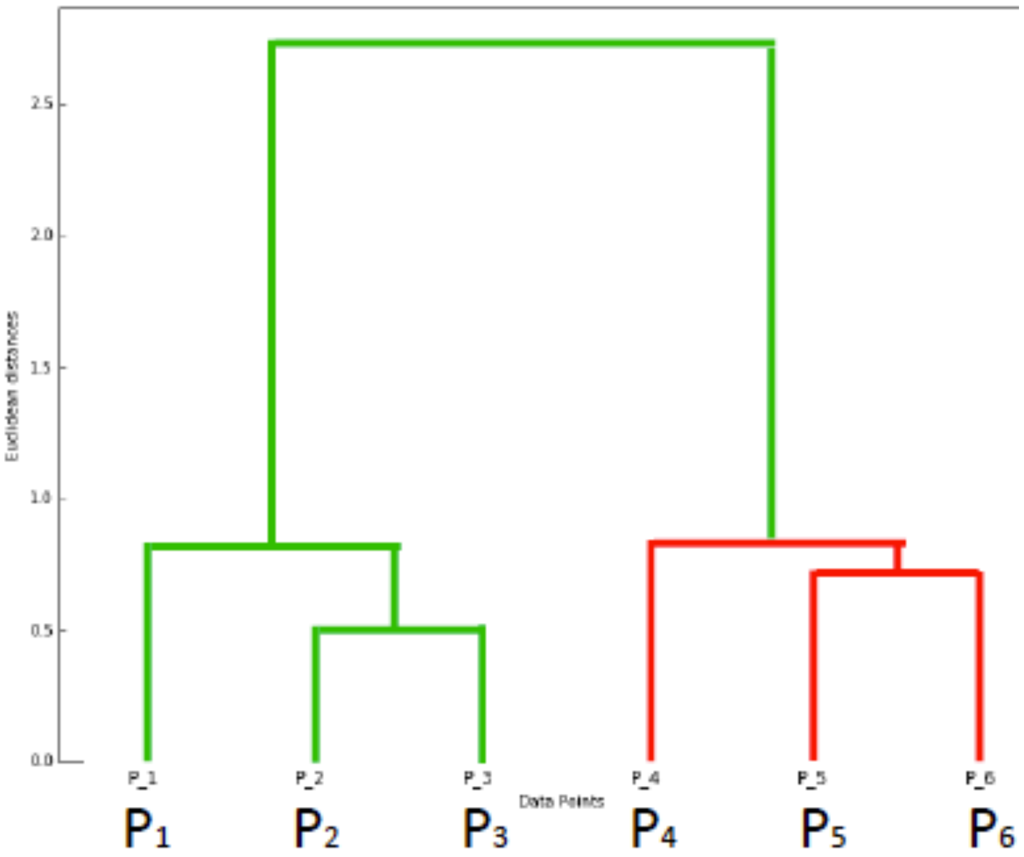

4.5
4.0
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
P_3
P_2
P_1
P_4
P_5
P_6

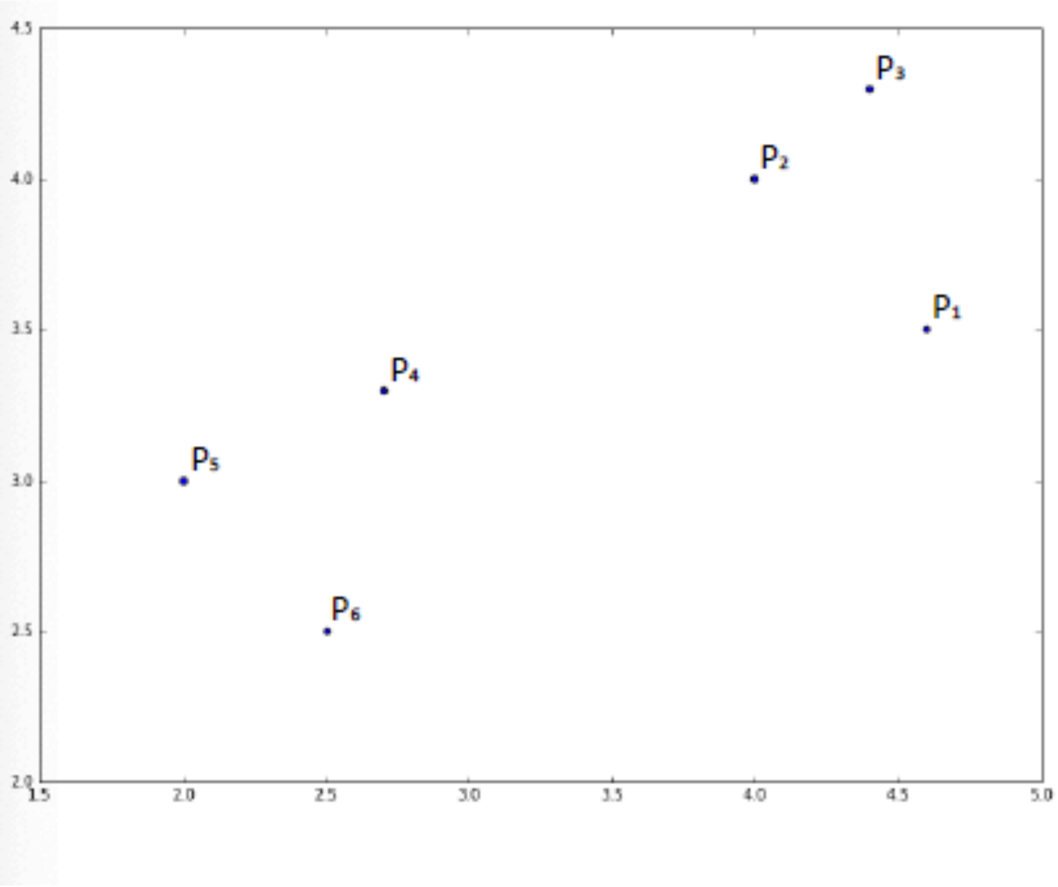

Euclidean distances
2.5
2.0
1.5
1.0
0.5
0.0
P_1
P_2
P_3
P_4
P_5
P_6
Data Points
P_1
P_2
P_3
P_4
P_5
P_6

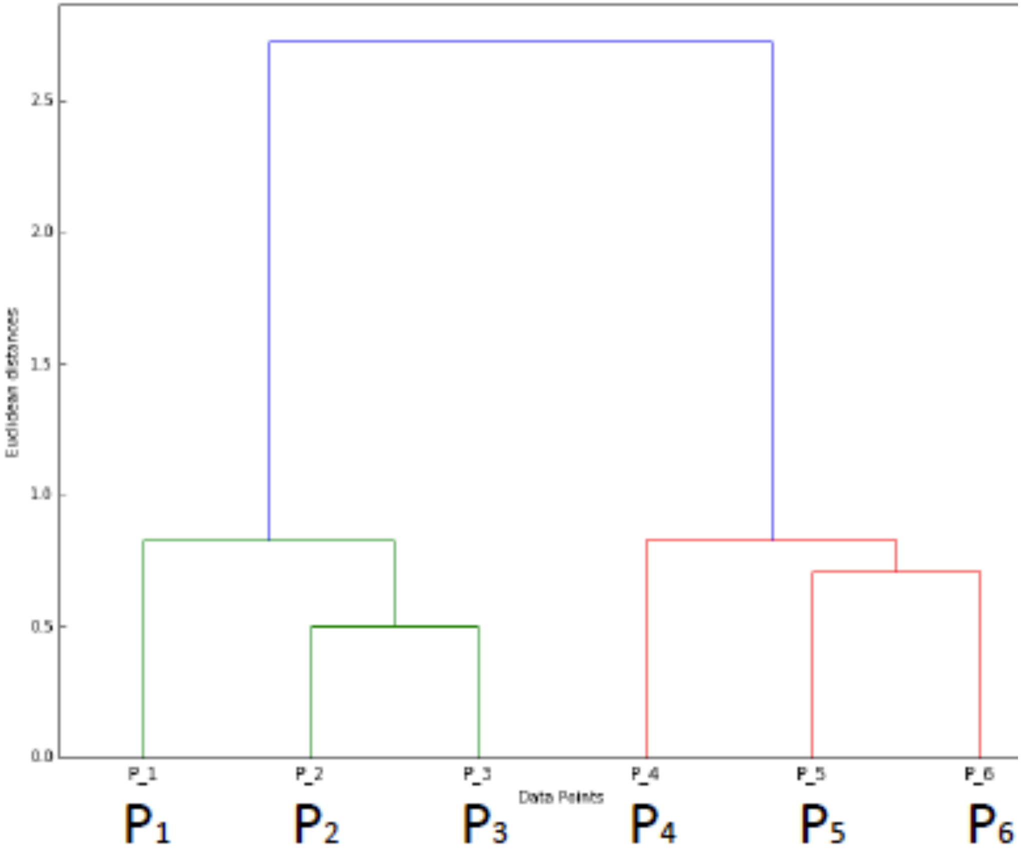

4.5
4.0
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
P_3
P_2
P_1
P_4
P_5
P_6


2.5
2.0
1.5
1.0
0.5
0.0
Euclidean distances
P_1
P_2
P_3
Data Points
P_4
P_5
P_6
P_1
P_2
P_3
P_4
P_5
P_6

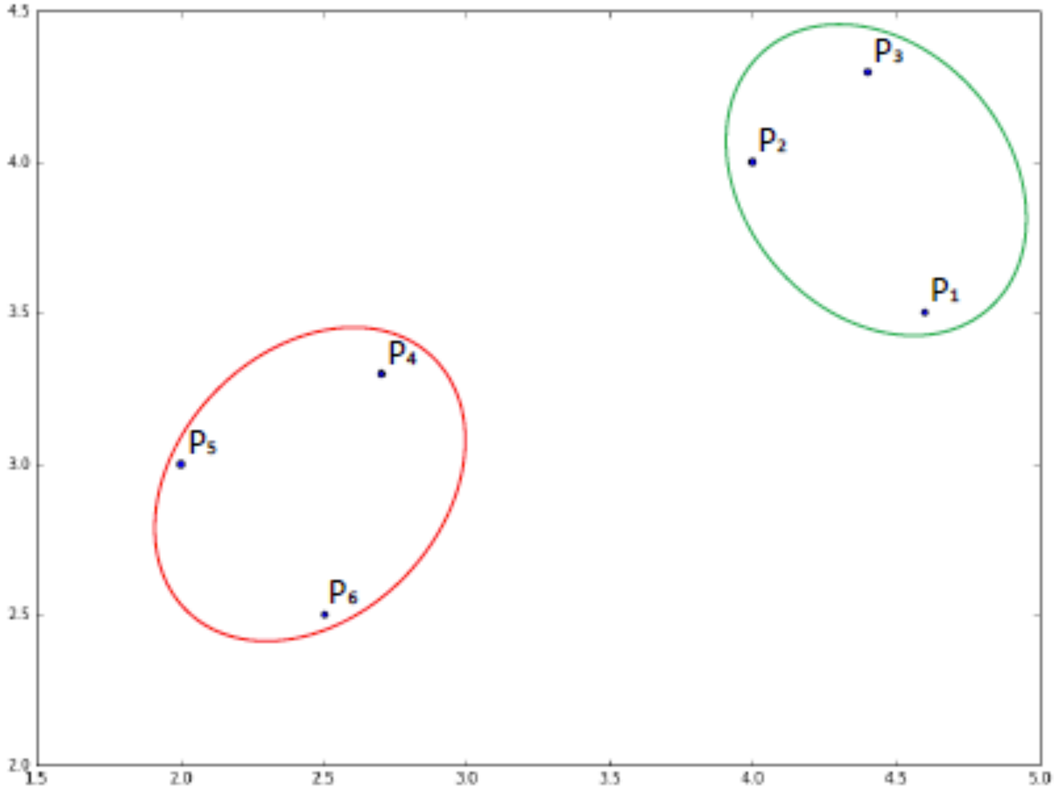

P₃
P₂
P₁
P₄
P₅
P₆
4.5
4.0
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0

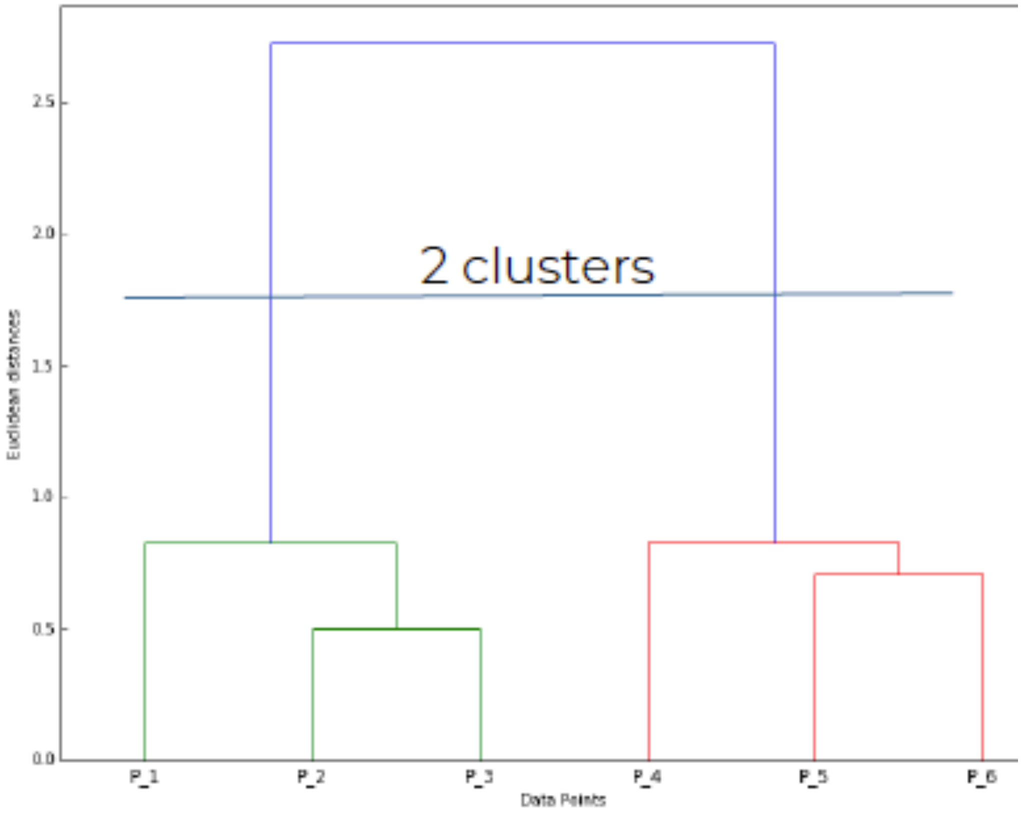

Euclidean distances
2.5
2.0
1.5
1.0
0.5
0.0
P_1
P_2
P_3
P_4
P_5
P_6
Data Points
P₁
P₂
P₃
P₄
P₅
P₆


4.5
4.0
3.5
3.0
2.5
2.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
P₃
P₂
P₁
P₄
P₅
P₆

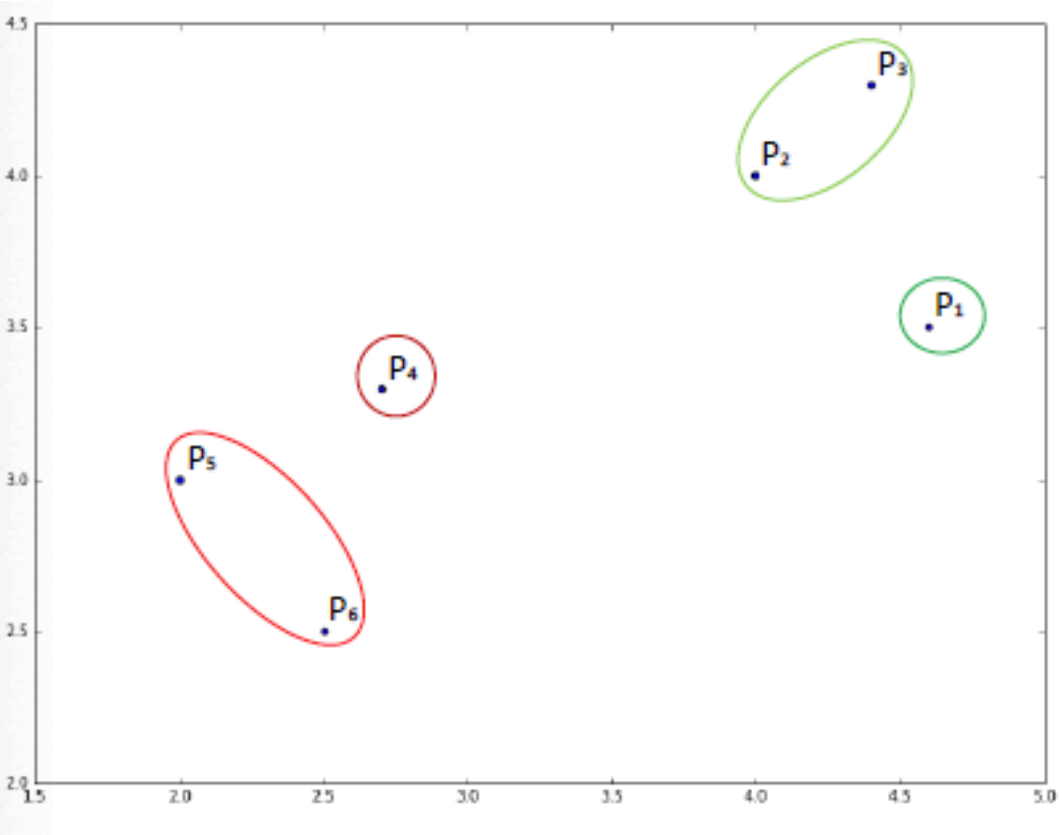

2.5
2.0
1.5
1.0
0.5
0.0
Euclidean distances
P_1
P_2
P_3
P_4
P_5
P_6
Data Points
P₁
P₂
P₃
P₄
P₅
P₆

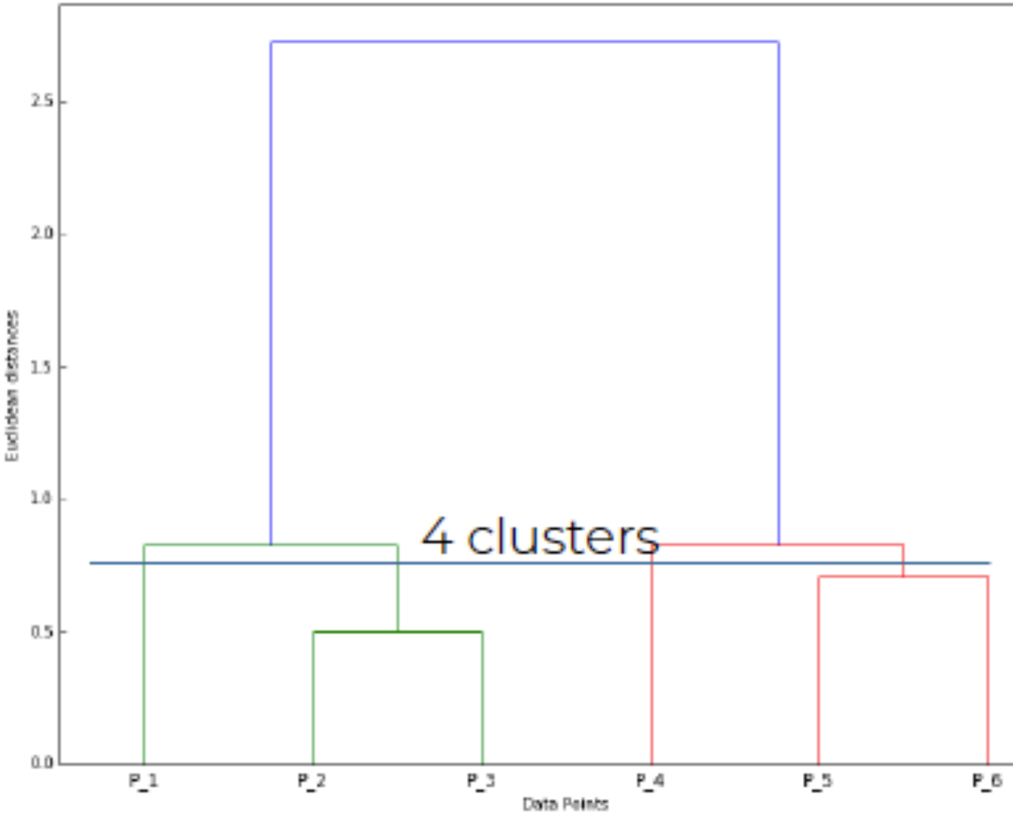
Two clusters



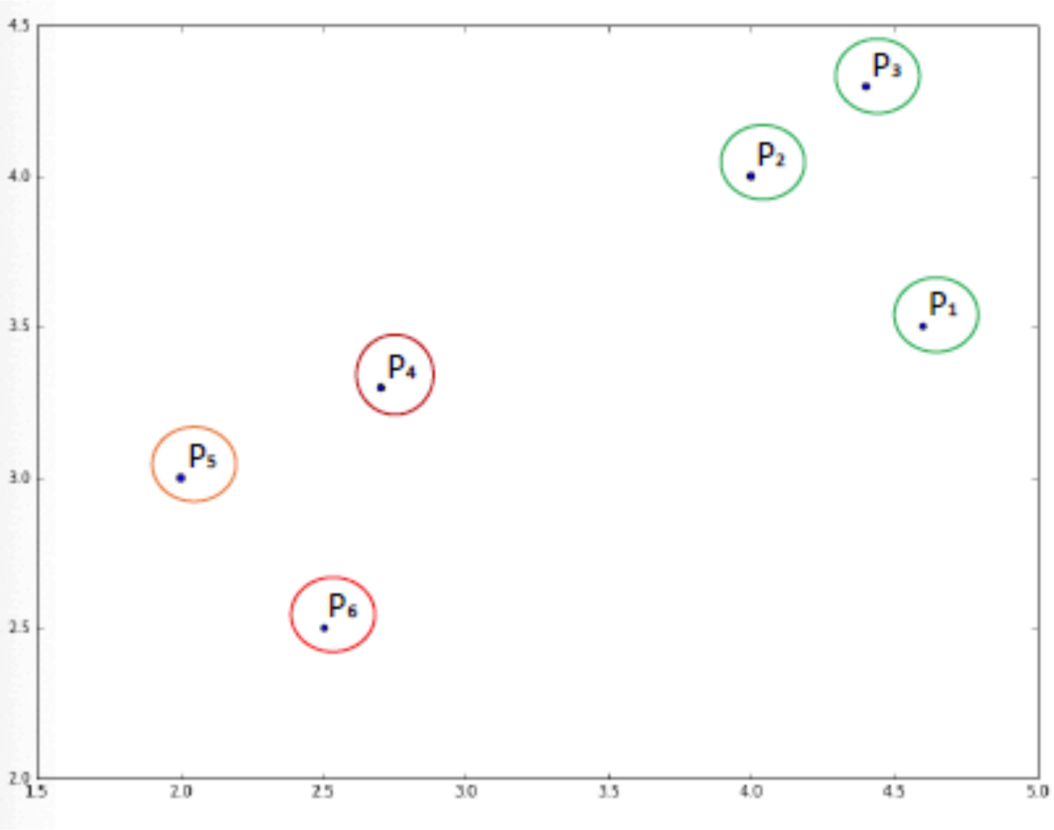


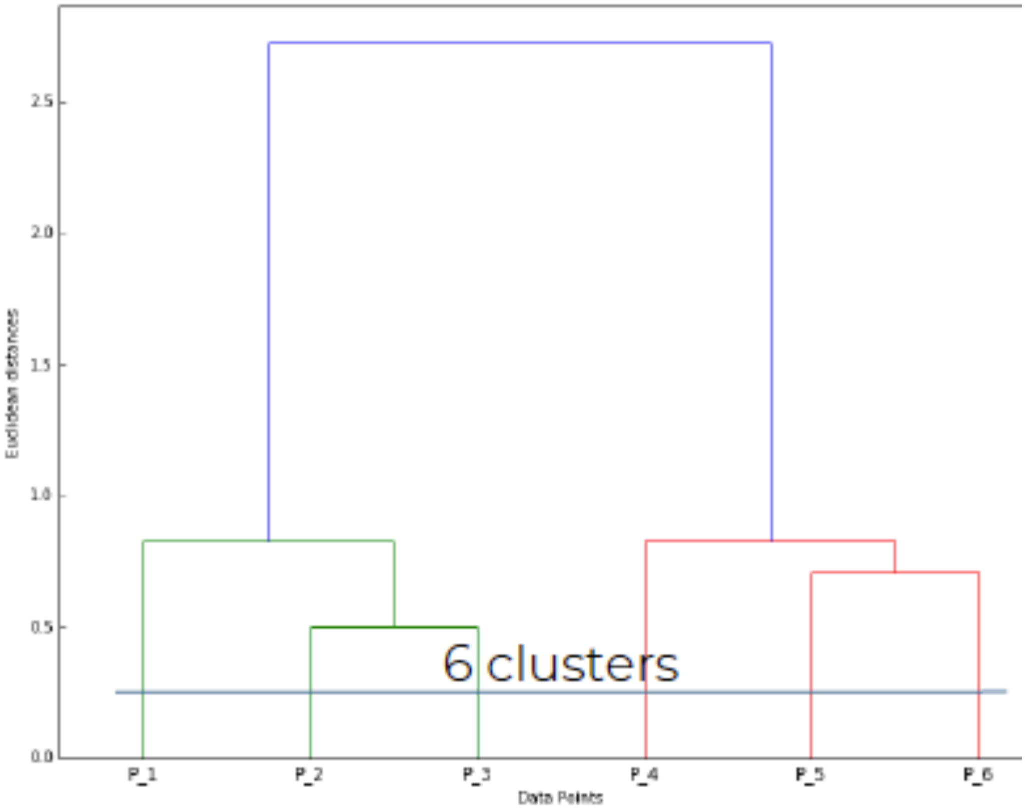
Four clusters



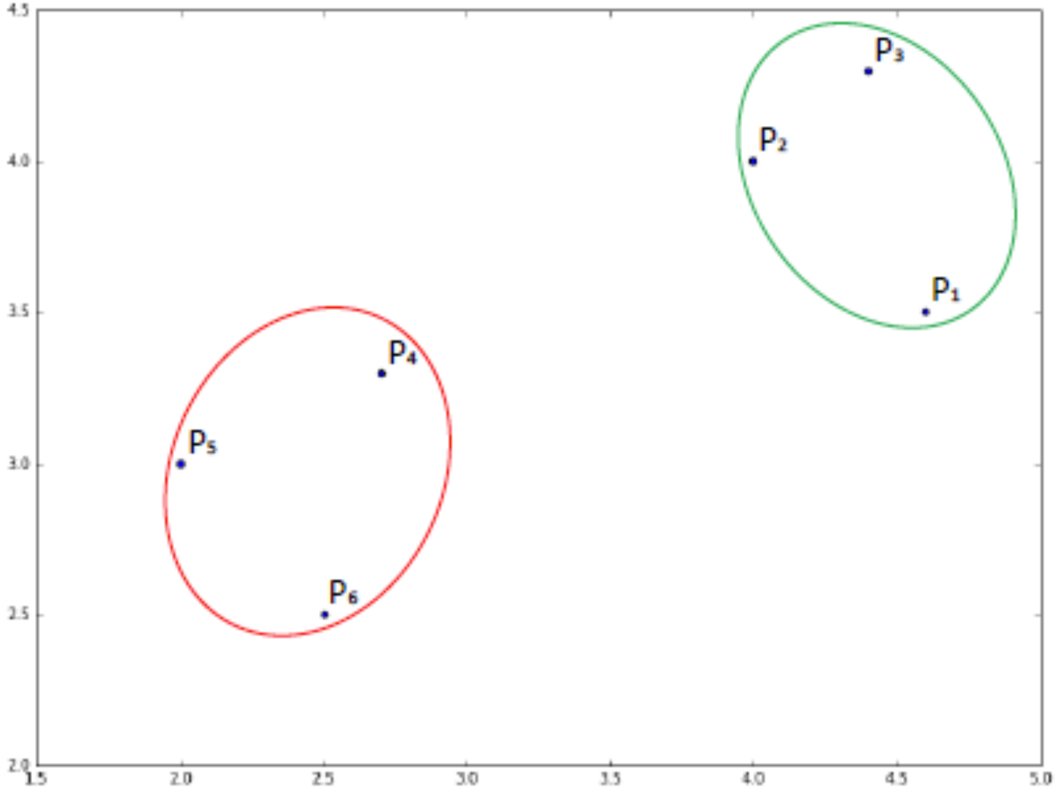


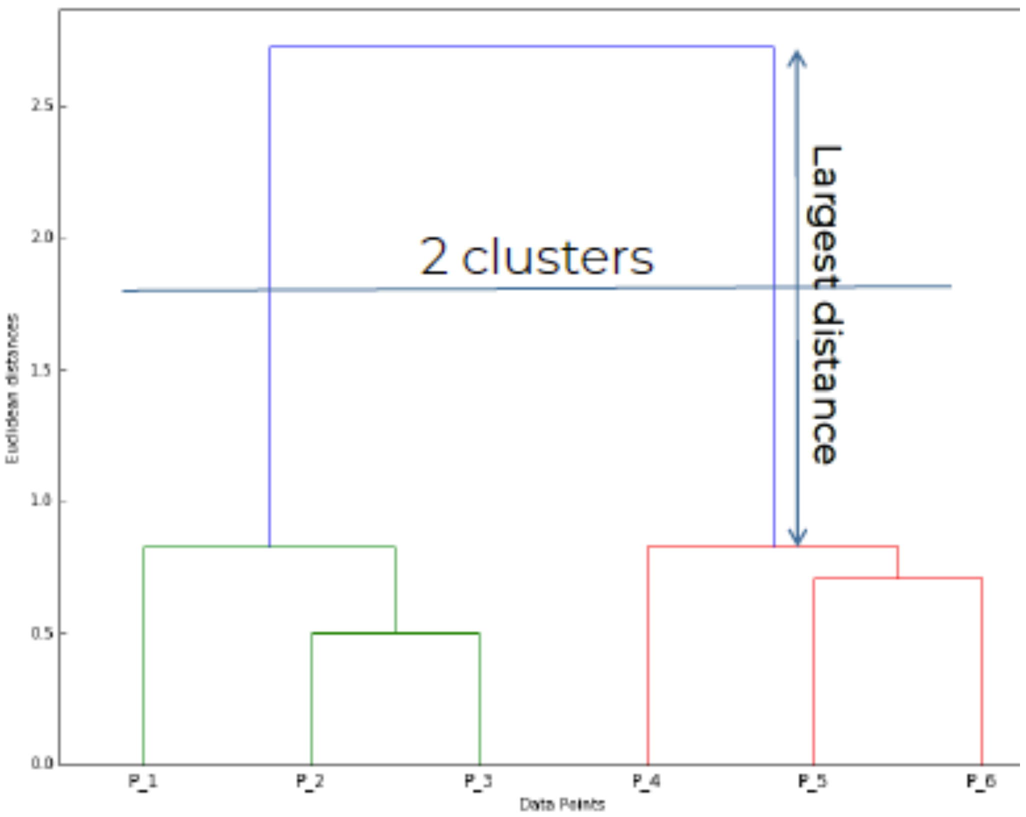
Six clusters



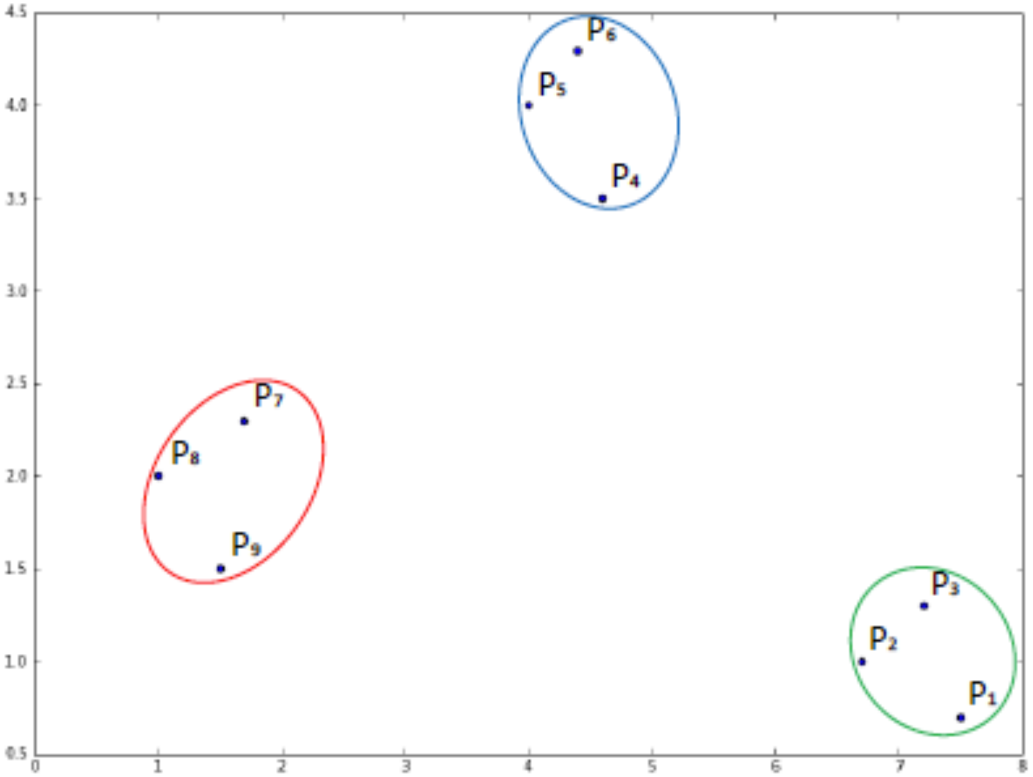


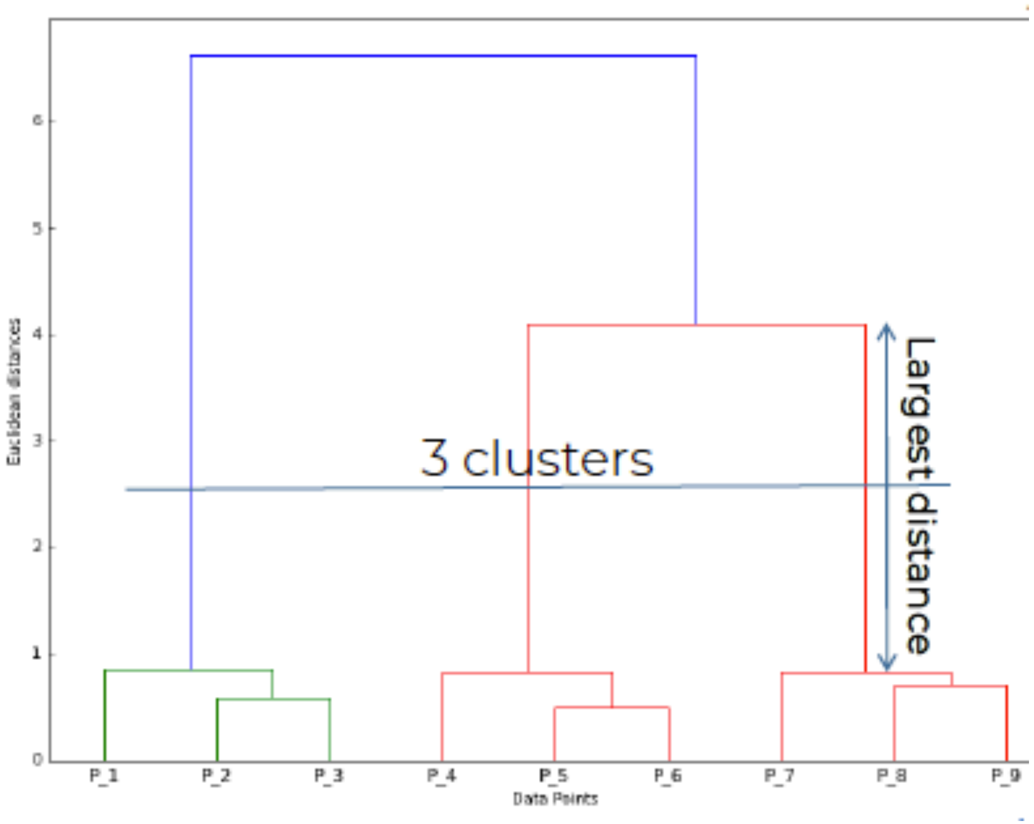
How to choose the optimal number of clusters:





Another example of optimal # of clusters:

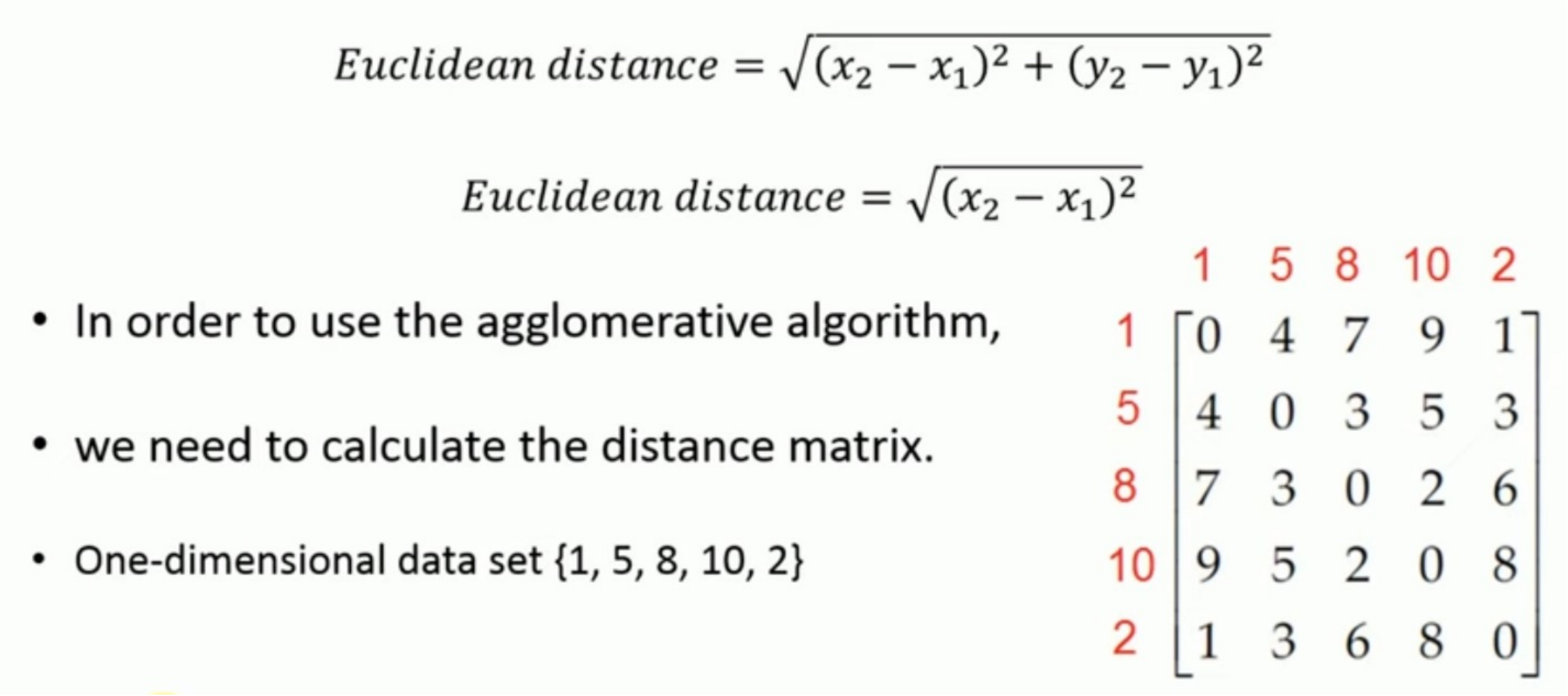




Practical Example: Complete Linkage-Agglomerative clustering

- Given a one dimensional data set {1,5,8,10,2}, use the agglomerative clustering algorithm with the complete link with Euclidean distance to establish a hierarchical grouping relationship.
- Assume we will use a threshold of 6, how many clusters are there?
- What are the data points in each clusters?

Example1:

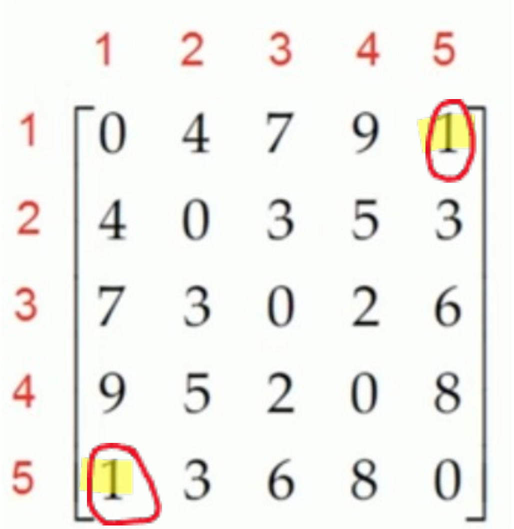


$$Euclidean\ distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$Euclidean\ distance = \sqrt{(x_2 - x_1)^2}$$

- In order to use the agglomerative algorithm,
- we need to calculate the distance matrix.
- One-dimensional data set {1, 5, 8, 10, 2}

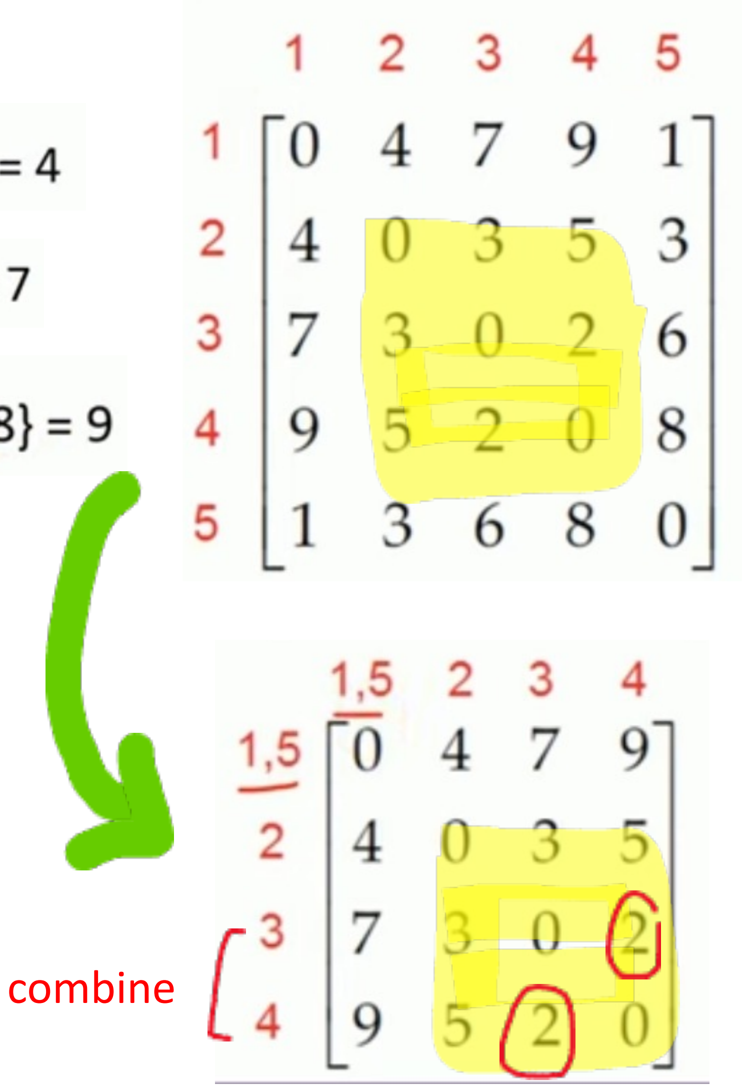
$$\begin{array}{c|ccccc} & 1 & 5 & 8 & 10 & 2 \\ \hline 1 & 0 & 4 & 7 & 9 & 1 \\ 5 & 4 & 0 & 3 & 5 & 3 \\ 8 & 7 & 3 & 0 & 2 & 6 \\ 10 & 9 & 5 & 2 & 0 & 8 \\ 2 & 1 & 3 & 6 & 8 & 0 \end{array}$$



$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 4 & 7 & 9 & 1 \\ 2 & 4 & 0 & 3 & 5 & 3 \\ 3 & 7 & 3 & 0 & 2 & 6 \\ 4 & 9 & 5 & 2 & 0 & 8 \\ 5 & 1 & 3 & 6 & 8 & 0 \end{array}$$

Example1 continue:

- d(2, {1,5}) = max{ d(2,1), d(2,5) } = max {4, 3} = 4
- d(3, {1,5}) = max{ d(3,1), d(3,5) } = max {7, 6} = 7
- d(4, {1,5}) = max{ d(4,1), d(4,5) } = max {9, 8} = 9

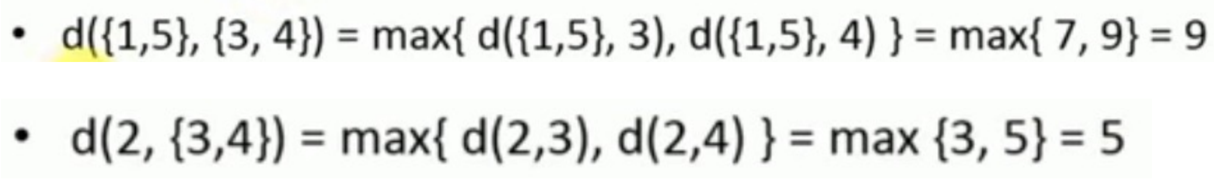


$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 4 & 7 & 9 & 1 \\ 2 & 4 & 0 & 3 & 5 & 3 \\ 3 & 7 & 3 & 0 & 2 & 6 \\ 4 & 9 & 5 & 2 & 0 & 8 \\ 5 & 1 & 3 & 6 & 8 & 0 \end{array}$$

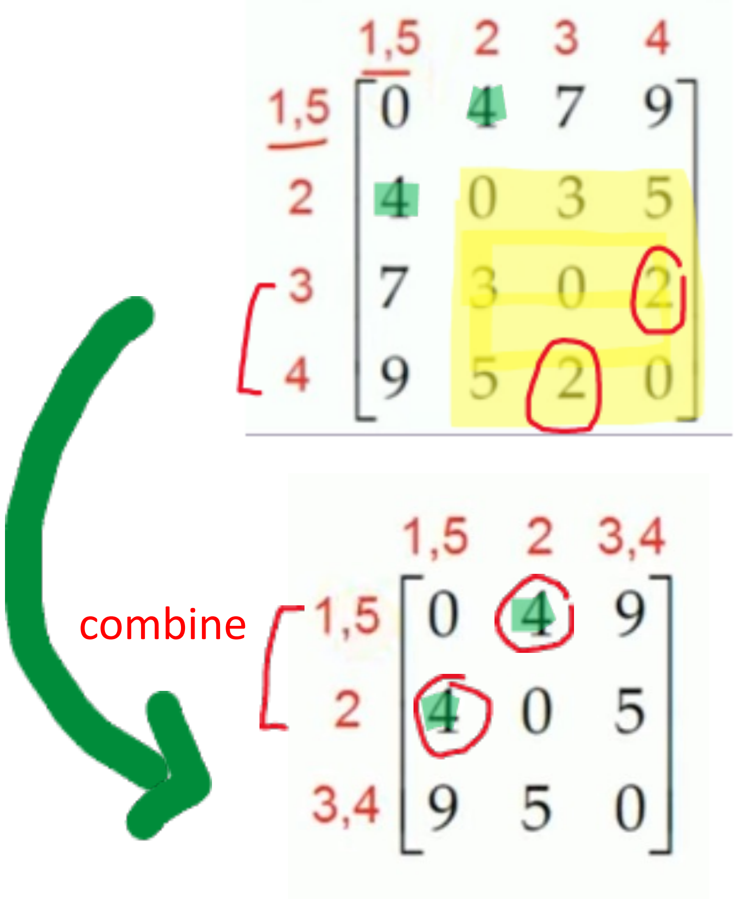
$$\begin{array}{c|cccc} & 1,5 & 2 & 3 & 4 \\ \hline 1,5 & 0 & 4 & 7 & 9 \\ 2 & 4 & 0 & 3 & 5 \\ 3 & 7 & 3 & 0 & 2 \\ 4 & 9 & 5 & 2 & 0 \end{array}$$

combine

Example1 continue:



- d({1,5}, {3, 4}) = max{ d({1,5}, 3), d({1,5}, 4) } = max{ 7, 9} = 9
- d(2, {3,4}) = max{ d(2,3), d(2,4) } = max {3, 5} = 5

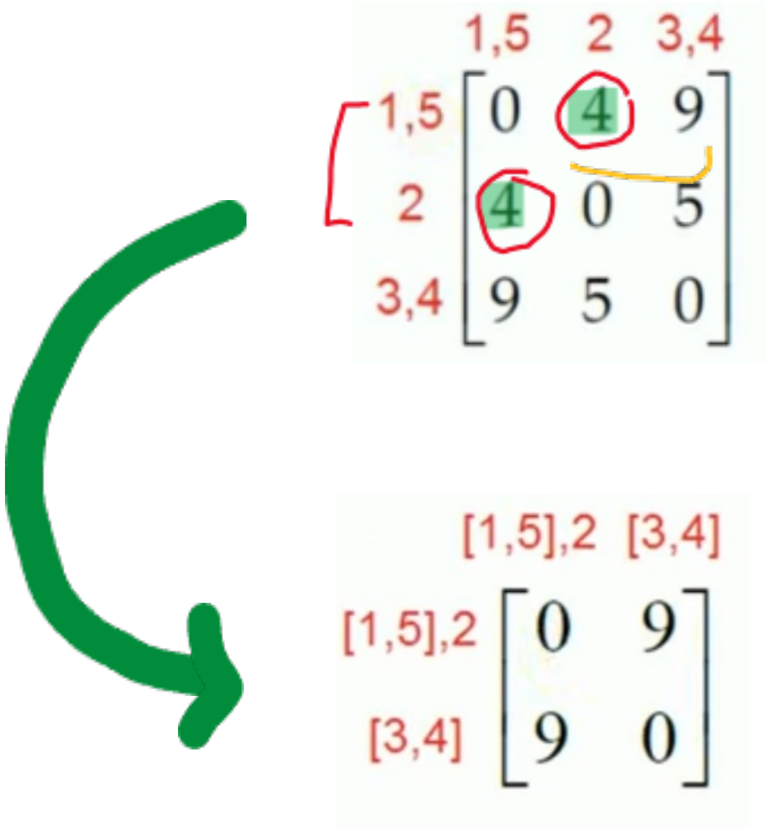


$$\begin{array}{c|cccc} & 1,5 & 2 & 3 & 4 \\ \hline 1,5 & 0 & 4 & 7 & 9 \\ 2 & 4 & 0 & 3 & 5 \\ 3 & 7 & 3 & 0 & 2 \\ 4 & 9 & 5 & 2 & 0 \end{array}$$

combine

$$\begin{array}{c|ccc} & 1,5 & 2 & 3,4 \\ \hline 1,5 & 0 & 4 & 9 \\ 2 & 4 & 0 & 5 \\ 3,4 & 9 & 5 & 0 \end{array}$$

Example1 continue:



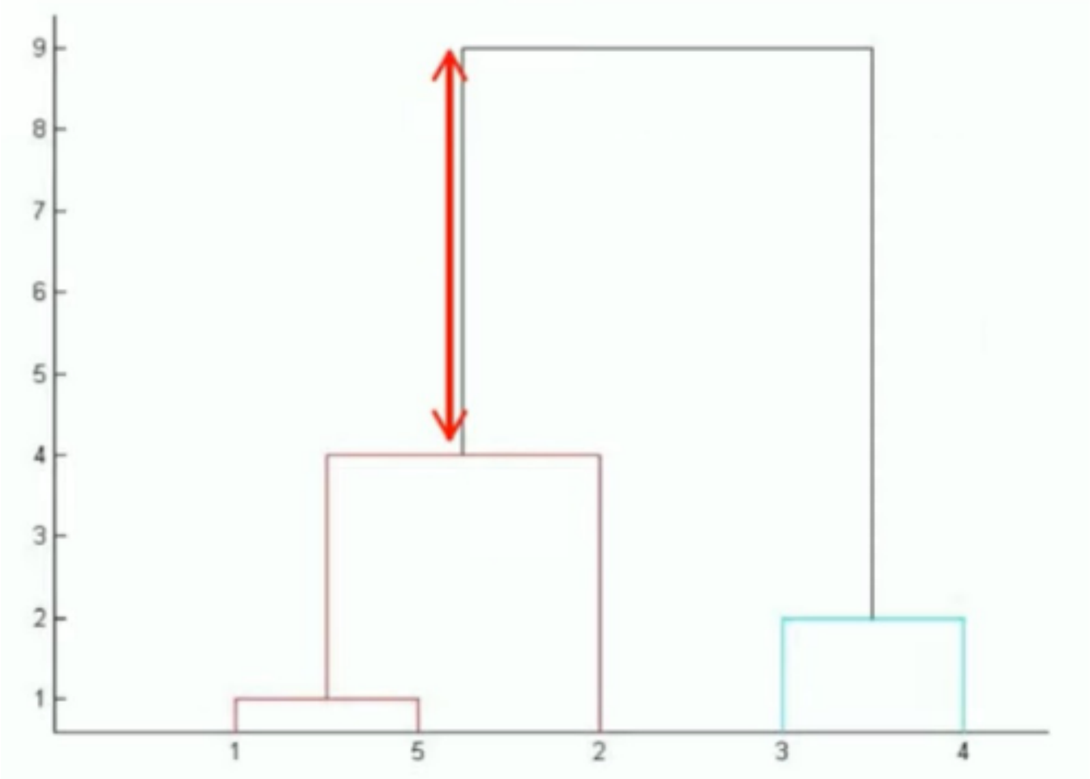
$$\begin{array}{c|ccc} & 1,5 & 2 & 3,4 \\ \hline 1,5 & 0 & \boxed{4} & 9 \\ 2 & \boxed{4} & 0 & 5 \\ 3,4 & 9 & 5 & 0 \end{array}$$

Pick max distance

$$\begin{array}{c|cc} & [1,5],2 & [3,4] \\ \hline [1,5],2 & 0 & 9 \\ [3,4] & 9 & 0 \end{array}$$

Example1 continue:

- After increasing the distance threshold to 9, all clusters would merge.
- Based on all the distance matrices we calculated, we draw the dendrogram tree as follows:


9
8
7
6
5
4
3
2
1
1
5
2
3
4