

Clustering- Unit4

Dr. H C Vijayalakshmi

References

1. <https://www.datavedas.com/hierarchical-clustering/>
2. <https://researchhubs.com/post/ai/fundamentals/agglomerative-hierarchical-clustering-algorithm-numerical-example.html>
3. IITKharagpur CS40003: Data Analytics

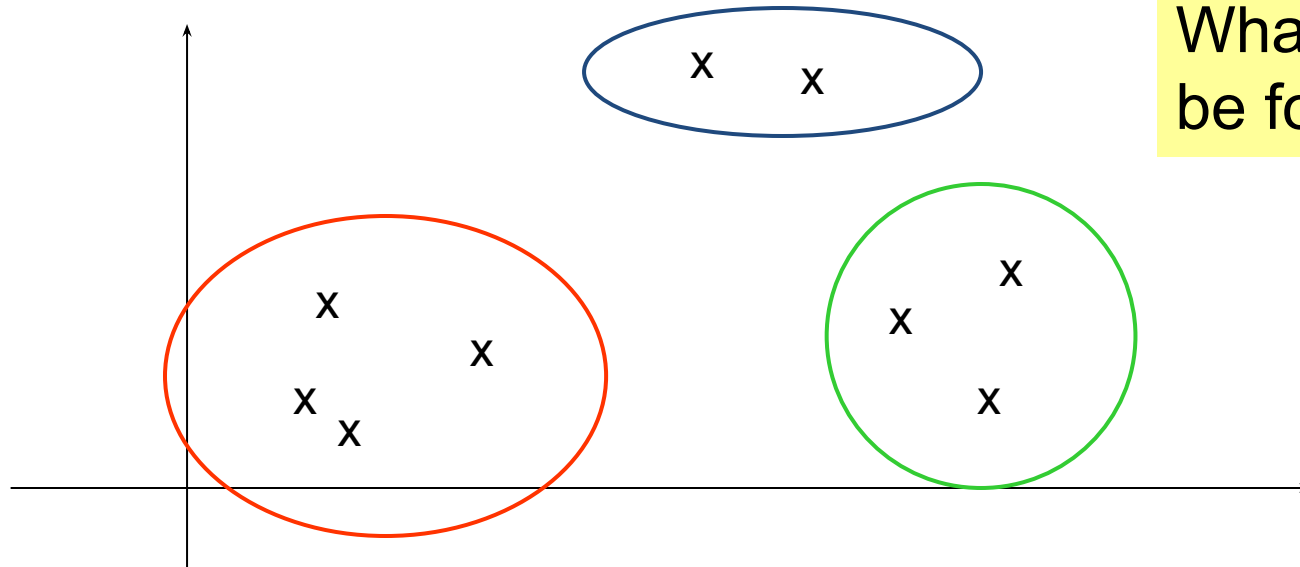
Cluster: A collection of data objects

Similar (or related) to one another within the same group

Dissimilar (or unrelated) to the objects in other groups

- The objective of cluster analysis is to assign observations to groups (clusters) so that observations within each group are similar to one another
- with respect to variables or attributes of interest, and the groups themselves stand apart from one another.
- **In other words, the objective is to divide the observations into homogeneous and distinct groups**

- The goal of clustering is to
 - group data points that are close (or **similar**) to each other
 - identify such groupings (or clusters) in an **unsupervised** manner
 - Unsupervised: no information is provided to the algorithm on which data points belong to which clusters
- Example



What should the clusters be for these data points?

Cluster analysis (or clustering, data segmentation, ...)

Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters

Unsupervised learning: no predefined classes (i.e., learning by observations)

- **Typical applications**

- As a stand-alone tool to get insight into data distribution

- As a preprocessing step for other algorithms

Some Applications of Clustering

- Pattern Recognition

- Image Processing : cluster images based on their visual content

- Bio-informatics

- WWW and IR** : Document classification

- cluster Weblog data to discover groups of similar access patterns

Example 1: Groups people of similar race and built,
Clothes of similar sizes together to make “small”, “medium”
and “large” category.

Example 2: In marketing, segment customers according to their similarities
To do targeted marketing.

Example 3: Given a collection of text documents, we want to organize them
according to their content similarities : To produce a topic hierarchy

Clustering: Task of grouping a set of data points such that data points in the same group are more similar to each other than data points in another group (group is known as cluster)

It groups data instances that are similar to (near) each other in one cluster and data instances that are very different (far away) from each other into different clusters.

Types of clustering:

1. Hierarchical Clustering:

- Agglomerative clustering,
divisive clustering

2. Partitional Clustering

- Forgy's Algorithm

- Exclusive Clustering: K-means

- Isodata (Combination of Forgy's and k Means)

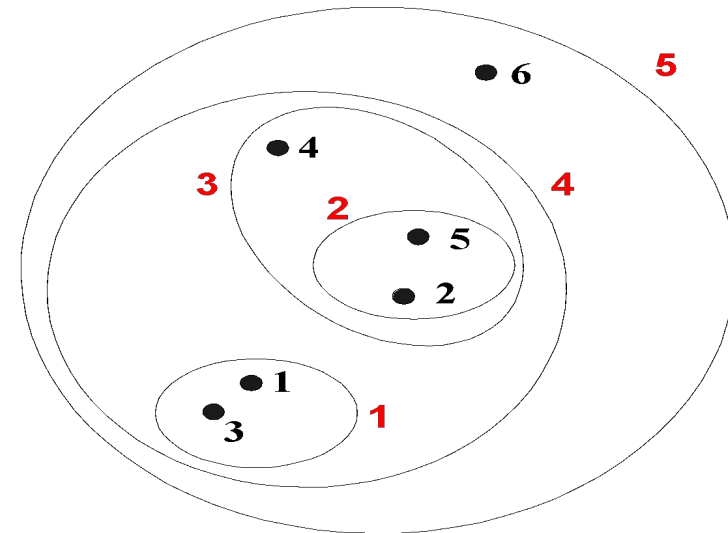
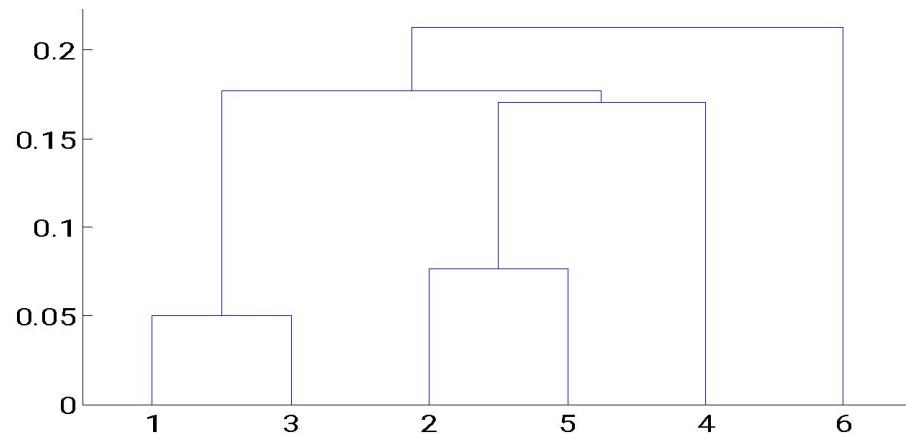
3. Probabilistic Clustering: Mixture of Gaussian models

What Is Good Clustering?

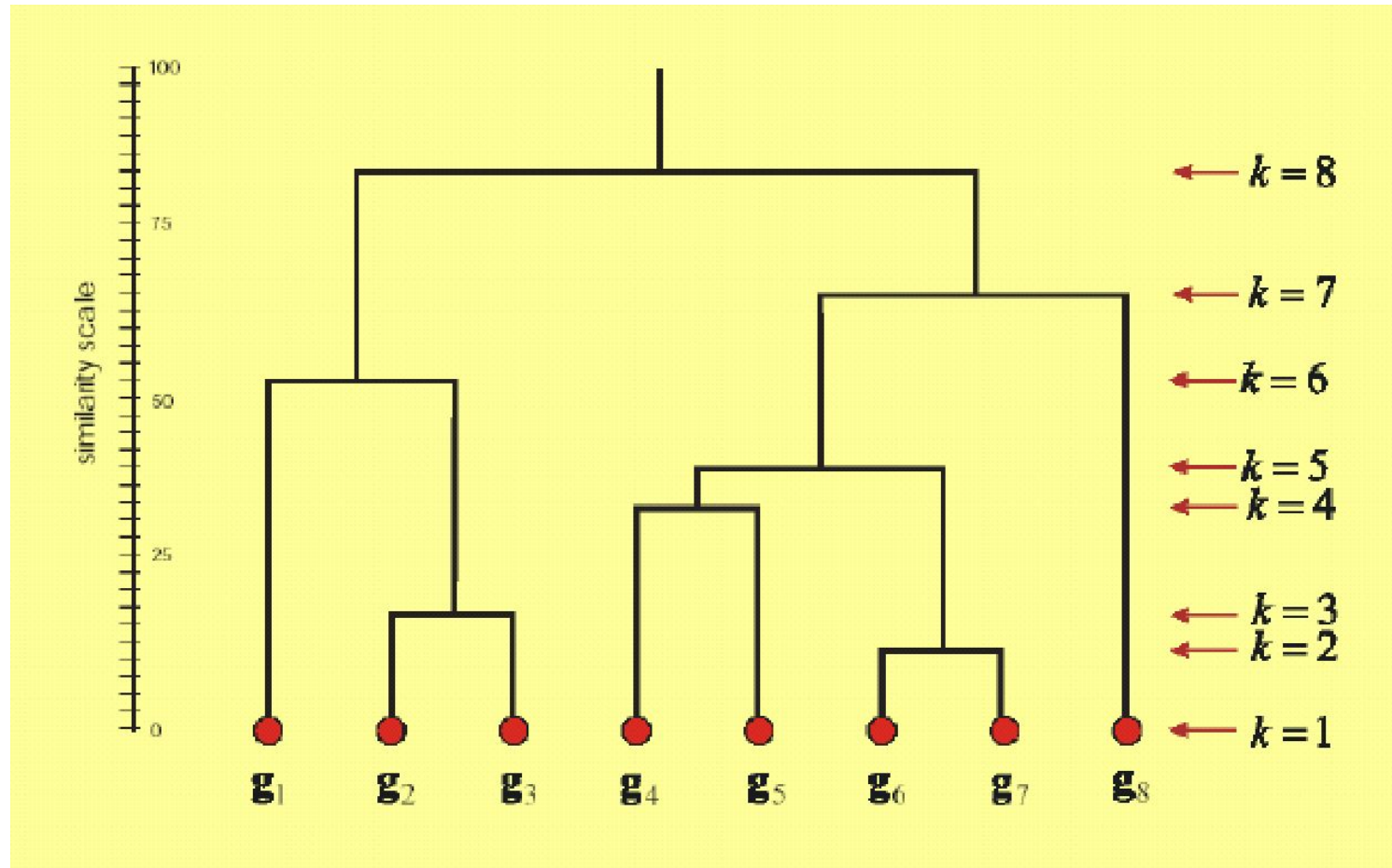
- A good clustering method will produce high quality clusters with
 - high intra-class similarity
 - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation.
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns.

Hierarchical Clustering : Hierarchy represented by a tree structure

- This refers to a clustering process that organizes the data into large groups, which contain smaller groups and so on in turn.
- Produces a set of nested clusters organized as a hierarchical tree
- **Can be visualized as a dendrogram**
 - Dendrogram is a visual representation of the compound correlation data.
 - The individual compounds (the finest group) are arranged along the bottom of the dendrogram
 - A tree like diagram that records the sequences of merges or splits



Hierarchical clustering



Strengths of Hierarchical Clustering

- Do not have to assume any particular number of clusters
 - Any desired number of clusters can be obtained by ‘cutting’ the dendrogram at the proper level
- They may correspond to meaningful taxonomies
 - Example in biological sciences (e.g., animal kingdom, phylogeny reconstruction, ...)

Pros

Dendrograms are great for visualization

Provides hierarchical relations between clusters

Shown to be able to capture concentric clusters

Cons

Not easy to define levels for clusters

Experiments showed that other clustering techniques outperform hierarchical clustering

Hierarchical Clustering Continued

- Two main types of hierarchical clustering
 - Agglomerative:
 - Start with the points as individual clusters
 - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
 - Divisive:
 - Start with one, all-inclusive cluster
 - At each step, split a cluster until each cluster contains a point (or there are k clusters)
- **Traditional hierarchical algorithms use a similarity or distance matrix**
 - Merge or split one cluster at a time

Agglomerative clustering :

More popular hierarchical clustering technique
Basic algorithm is straightforward

This bottom-up strategy starts by placing each object in its own cluster .

Then merges these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster **or**
Until certain termination conditions are satisfied.

It is more popular than divisive methods.

Each node/object is a cluster initially

Merge clusters that have the least dissimilarity

Ex: single-linkage, complete-linkage, Average linking algorithm etc.

Go on in a non-descending fashion

Eventually, all nodes belong to the same cluster

This method Uses Linkage Criteria

Determines the distance between sets of observations as a function of the pairwise distances between observations.

Some commonly used criteria:

Single Linkage:

Distance between two clusters is the smallest pairwise distance between two observations/nodes, each belonging to different clusters.

Complete Linkage:

Distance between two clusters is the smallest of the largest pairwise distance between two observations/nodes, each belonging to different clusters.

Mean or average linkage clustering:

Distance between two clusters is the average of all the pairwise distances, each node/observation belonging to different clusters.

Centroid linkage clustering:

Distance between two clusters is the distance between their centroids.

Agglomerative Clustering Algorithm

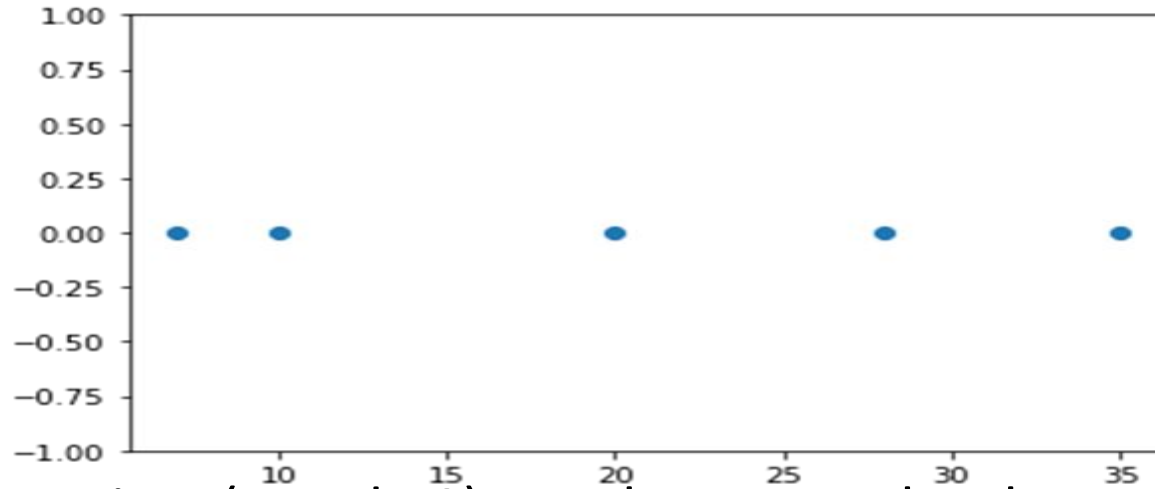
1. Compute the proximity matrix
2. Let each data point be a cluster
3. **Repeat**
4. Merge the two closest clusters
5. Update the proximity matrix
6. **Until** only a single cluster remains

Key operation is the computation of the proximity of two clusters

- Different approaches to defining the distance between clusters distinguish the different algorithms

Note : A square matrix in which the entry in cell (j, k) is **some measure of the** similarity (or distance) between the items to which row j and column k correspond.

A simple example would be a standard mileage chart—the smaller the entry, the closer together are the two items.

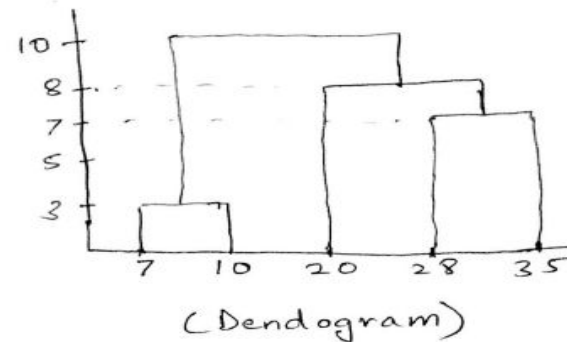
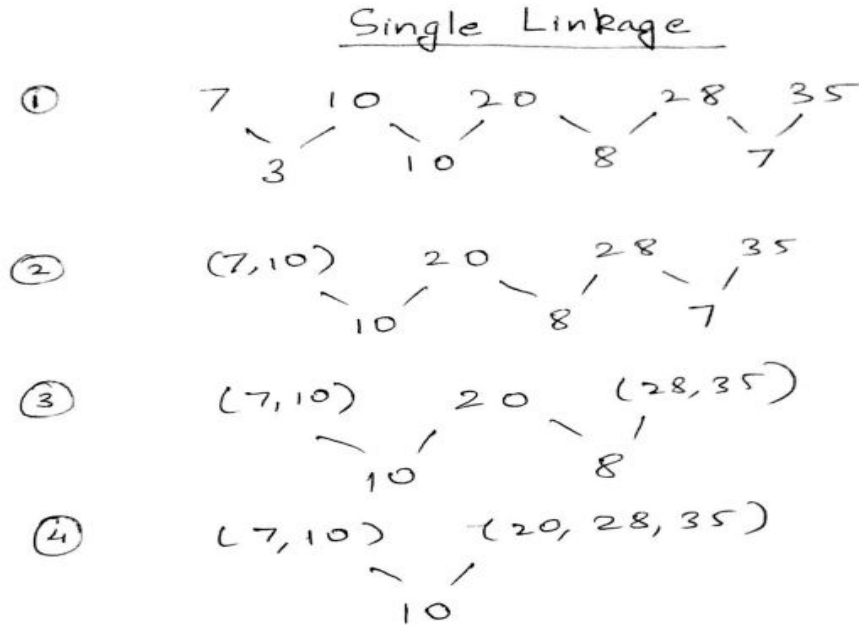


- The first two points (5 and 10) are close to each other and should be in the same cluster
- Also, the last two points (28 and 35) are close to each other and should be in the same cluster
- Cluster of the center point (20) is not easy to conclude
- Let's solve the problem by hand using both the types of agglomerative hierarchical clustering :
- **Single Linkage** : In single link hierarchical clustering, we merge in each step the two clusters, whose two closest members have the smallest distance.

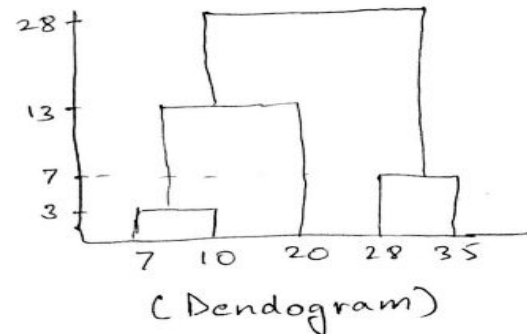
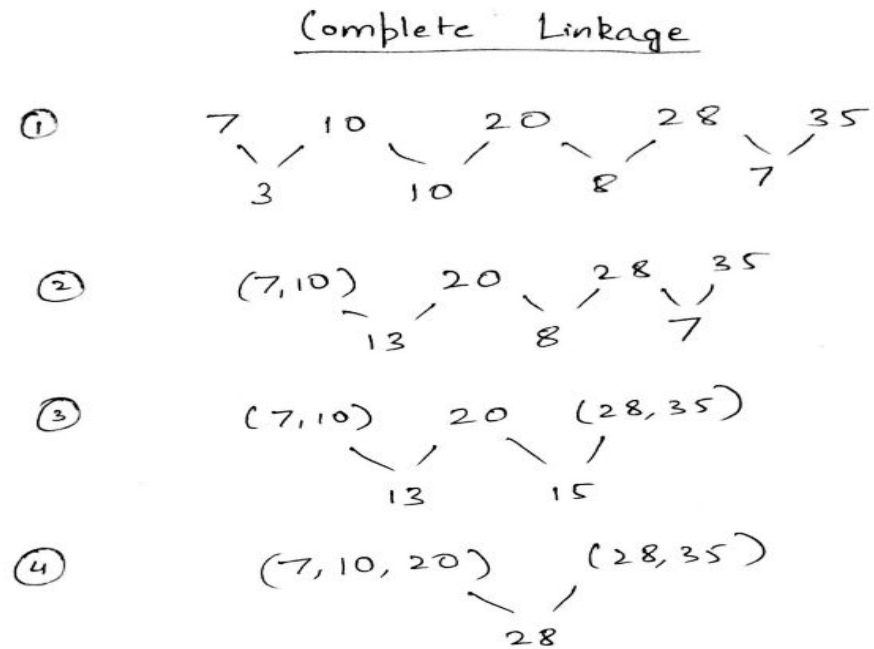
Using single linkage two clusters are formed :

Cluster 1 : (7,10)

Cluster 2 : (20,28,35)

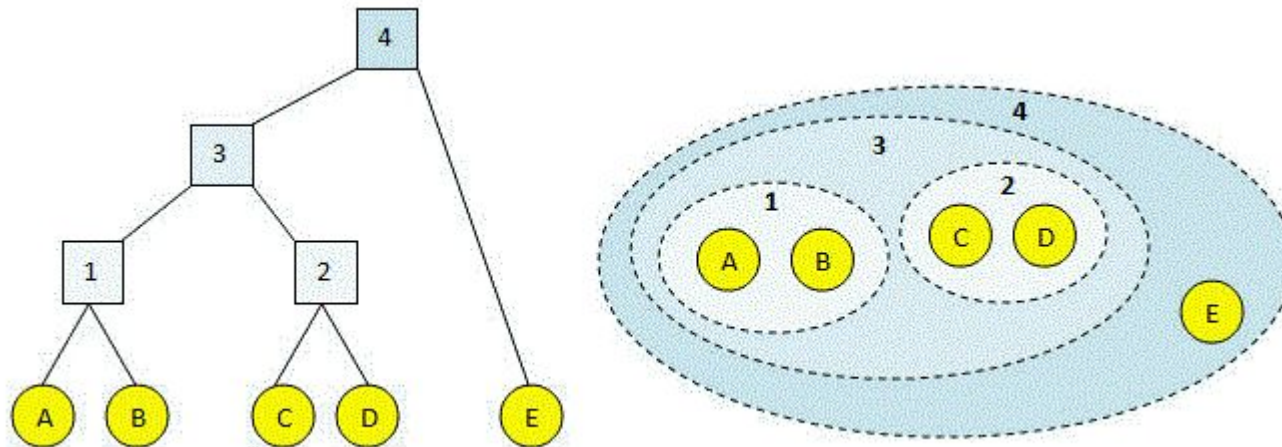


- **Complete Linkage** : In complete link hierarchical clustering, we merge in the members of the clusters in each step, which provide the smallest maximum pairwise distance.



Single linkage... Continued

- The single linkage algorithm is also known as the minimum method and the nearest neighbor method.
- Consider C_i and C_j as two clusters.
- 'a' and 'b' are samples from cluster C_i and C_j respectively.
$$D(C_i, C_j) = \min d(a, b) \text{ here 'a' } \in C_i \text{ and 'b' } \in C_j$$
- Where $d(a, b)$ represents the distance between 'a' and 'b'



First level of distance computation D1 (Euclidean distance used)

	x	y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12

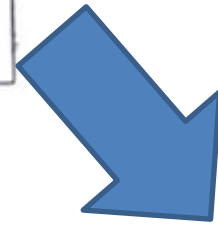


	1	2	3	4	5
1	—	4.0	11.7	20.0	21.5
2	4.0	—	8.1	16.0	17.9
3	11.7	8.1	—	9.8	9.8
4	20.0	16.0	9.8	—	8.0
5	21.5	17.9	9.8	8.0	—

- Use Euclidean distance for distance between samples.
- The table shown in the previous slide gives feature values for each sample and the distance d between each pair of samples.
- The algorithm begins with five clusters, each consisting of one sample.
- The two nearest clusters are then merged.
- The smallest number is 4 which is the distance between (1 and 2), so they are merged. Merged matrix is as shown in next slide.

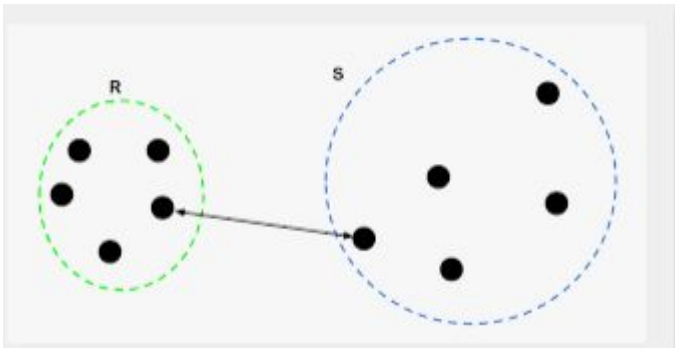
D2 matrix

	1	2	3	4	5
1	—	4.0	11.7	20.0	21.5
2	4.0	—	8.1	16.0	17.9
3	11.7	8.1	—	9.8	9.8
4	20.0	16.0	9.8	—	8.0
5	21.5	17.9	9.8	8.0	—



$\{1, 2\}, \{3\}, \{4\}, \{5\}.$

Next obtain the matrix that gives the distances between these clusters:



	$\{1,2\}$	3	4	5
$\{1,2\}$	—	8.1	16.0	17.9
3	8.1	—	9.8	9.8
4	16.0	9.8	—	8.0
5	17.9	9.8	8.0	—

- In the next level, the smallest number in the matrix is 8
- It is between 4 and 5.
- Now the cluster 4 and 5 are merged.
- With this we will have 3 clusters: $\{1,2\}$, $\{3\}$, $\{4,5\}$
- The matrix is as shown in the next slide.

D3 distance

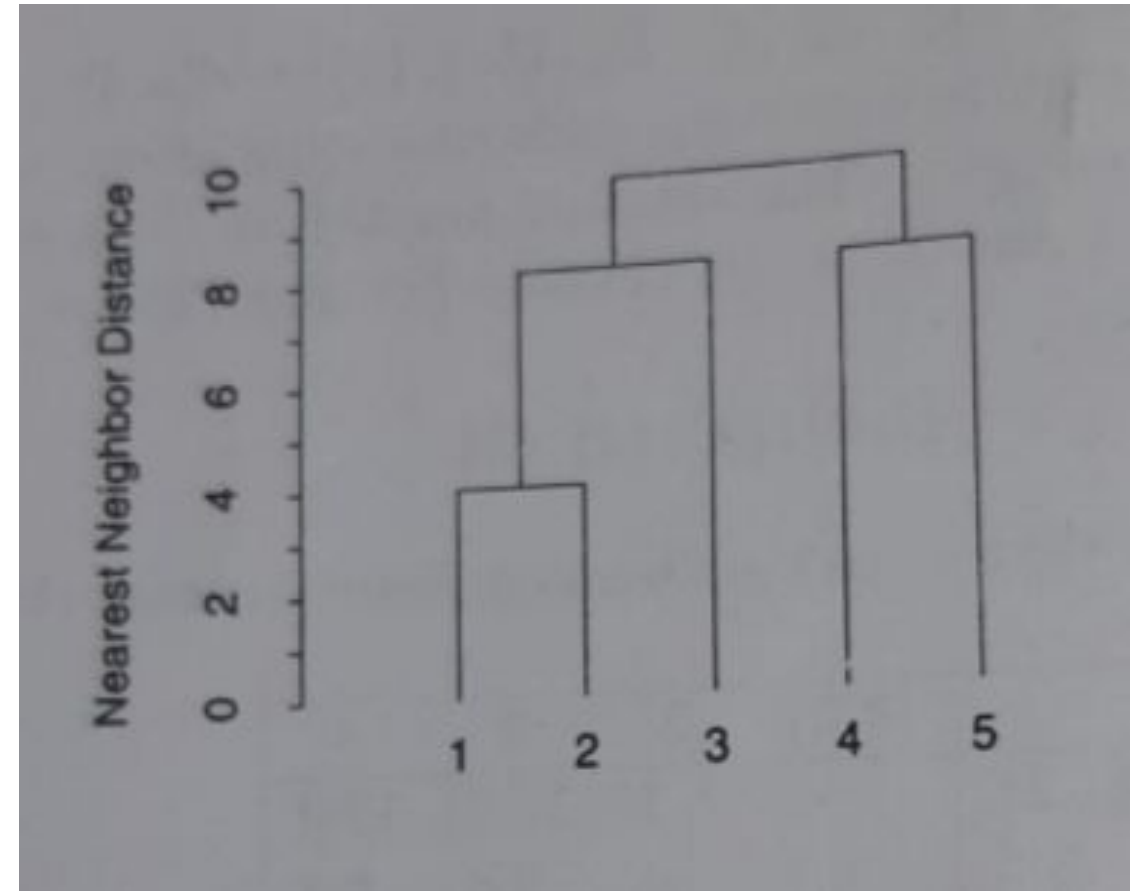
	$\{1,2\}$	3	4	5
$\{1,2\}$	—	8.1	16.0	17.9
3	8.1	—	9.8	9.8
4	16.0	9.8	—	8.0
5	17.9	9.8	8.0	—



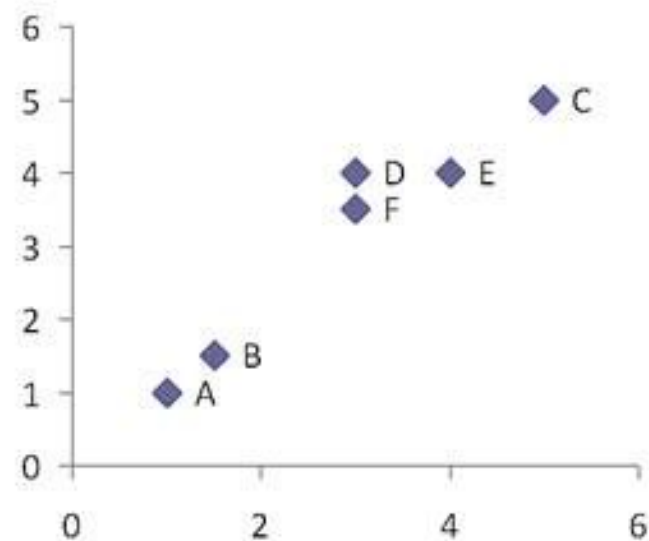
	$\{1,2\}$	3	$\{4,5\}$
$\{1,2\}$	—	8.1	16.0
3	8.1	—	9.8
$\{4,5\}$	16.0	9.8	—

- In the next step $\{1,2\}$ will be merged with $\{3\}$.
- Now we will have two cluster $\{1,2,3\}$ and $\{4,5\}$
- In the next step.. these two are merged to have single cluster.
- Dendrogram is as shown here.
- Height of the dendrogram is decided based on the merger distance.

For example: 1 and 2 are merged at the least distance 4. hence the height is 4.



	X1	X2
A	1	1
B	1.5	1.5
C	5	5
D	3	4
E	4	4
F	3	3.5



Dist	A	B	C	D	E	F
A	0.00	0.71	5.66	3.61	4.24	3.20
B	0.71	0.00	4.95	2.92	3.54	2.50
C	5.66	4.95	0.00	2.24	1.41	2.50
D	3.61	2.92	2.24	0.00	1.00	0.50
E	4.24	3.54	1.41	1.00	0.00	1.12
F	3.20	2.50	2.50	0.50	1.12	0.00

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	?	4.24
B	0.71	0.00	4.95	?	3.54
C	5.66	4.95	0.00	?	1.41
D, F	?	?	?	0.00	?
E	4.24	3.54	1.41	?	0.00

- Using the input distance matrix, distance between cluster (D, F) and cluster A is computed as

$$d_{(D,F) \rightarrow A} = \min(d_{DA}, d_{FA}) = \min(3.61, 3.20) = 3.20$$

- Distance between cluster (D, F) and cluster B is

$$d_{(D,F) \rightarrow B} = \min(d_{DB}, d_{FB}) = \min(2.92, 2.50) = 2.50$$

- Similarly, distance between cluster (D, F) and cluster C is

$$d_{(D,F) \rightarrow C} = \min(d_{DC}, d_{FC}) = \min(2.24, 2.50) = 2.24$$

- Finally, distance between cluster E and cluster (D, F) is calculated as

$$d_{E \rightarrow (D,F)} = \min(d_{ED}, d_{EF}) = \min(1.00, 1.12) = 1.00$$

Then, the updated distance matrix becomes

Min Distance (Single Linkage)

Dist	A	B	C	D, F	E
A	0.00	0.71	5.66	3.20	4.24
B	0.71	0.00	4.95	2.50	3.54
C	5.66	4.95	0.00	2.24	1.41
D, F	3.20	2.50	2.24	0.00	1.00
E	4.24	3.54	1.41	1.00	0.00

Using the input distance matrix (size 6 by 6), distance between cluster C and cluster (D, F) is computed as

$$d_{C \rightarrow (A,B)} = \min(d_{CA}, d_{CB}) = \min(5.66, 4.95) = 4.95$$

Dist	A,B	C	(D, F)	E
A,B	0	?	?	?
C	?	0	2.24	1.41
(D, F)	?	2.24	0	1.00
E	?	1.41	1.00	0

Distance between cluster (D, F) and cluster (A, B) is the minimum distance between all objects involves in the two clusters

$$d_{(D,F) \rightarrow (A,B)} = \min(d_{DA}, d_{DB}, d_{FA}, d_{FB}) = \min(3.61, 2.92, 3.20, 2.50) = 2.50$$

Similarly, distance between cluster E and (A, B) is

$$d_{E \rightarrow \{A, B\}} = \min \{d_{EA}, d_{EB}\} = \min \{4.24, 3.54\} = 3.54$$

Then the updated distance matrix is

Min Distance (Single Linkage)

Dist	A,B	C	(D, F)	E
A,B	0	4.95	2.50	3.54
C	4.95	0	2.24	1.41
(D, F)	2.50	2.24	0	1.00
E	3.54	1.41	1.00	0

Distance between cluster ((D, F), E) and cluster (A, B) is calculated as

Min Distance (Single Linkage)

Dist	(A,B)	C	(D, F), E
(A,B)	0.00	4.95	2.50
C	4.95	0.00	1.41
(D, F), E	2.50	1.41	0.00

$$d_{((D,F),E) \rightarrow (A,B)} = \min(d_{DA}, d_{DB}, d_{FA}, d_{FB}, d_{EA}, d_{EB}) = \min(3.61, 2.92, 3.20, 2.50, 4.24, 3.54) = 2.50$$

Distance between cluster ((D, F), E) and cluster C yields the minimum distance of 1.41. This distance is computed as

$$d_{((D,F),E) \rightarrow C} = \min(d_{DC}, d_{FC}, d_{EC}) = \min(2.24, 2.50, 1.41) = 1.41$$

Min Distance (Single Linkage)

Dist	(A,B)	((D, F), E), C
(A,B)	0.00	2.50
((D, F), E), C	2.50	0.00

Consider the points

	a	b
Point		
P1	0.07	0.83
P2	0.85	0.14
P3	0.66	0.89
P4	0.49	0.64
P5	0.80	0.46

Step1

	P1	P2	P3	P4	P5
P1	0				
P2	1.04139	0			
P3	0.59304	0.77369	0		
P4	0.46098	0.61612	0.30232	0	
P5	0.81841	0.32388	0.45222	0.35847	0

Step2

	P1	P2	✓P3	P4	P5
P1	0				
P2	1.04139	0			
✓P3	0.59304	0.77369	0		
✓P4	0.46098	0.61612	0.30232	0	
P5	0.81841	0.32388	0.45222	0.35847	0

Step3

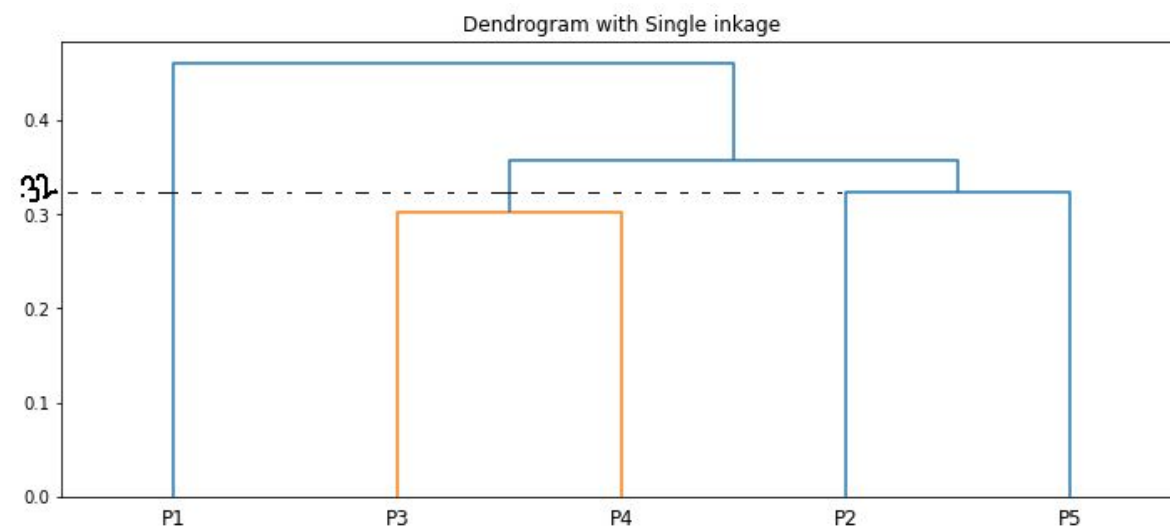
	P1	P2	P3,P4	P5
P1	0			
P2	1.04139	0		
P3,P4	0.46098	0.61612	0	
P5	0.81841	0.32388	0.35847	0

	P1	✓P2	P3,P4	P5
P1	0			
P2✓	1.04139	0		
P3,P4	0.46098	0.61612	0	
P5✓	0.81841	0.32388	0.35847	0

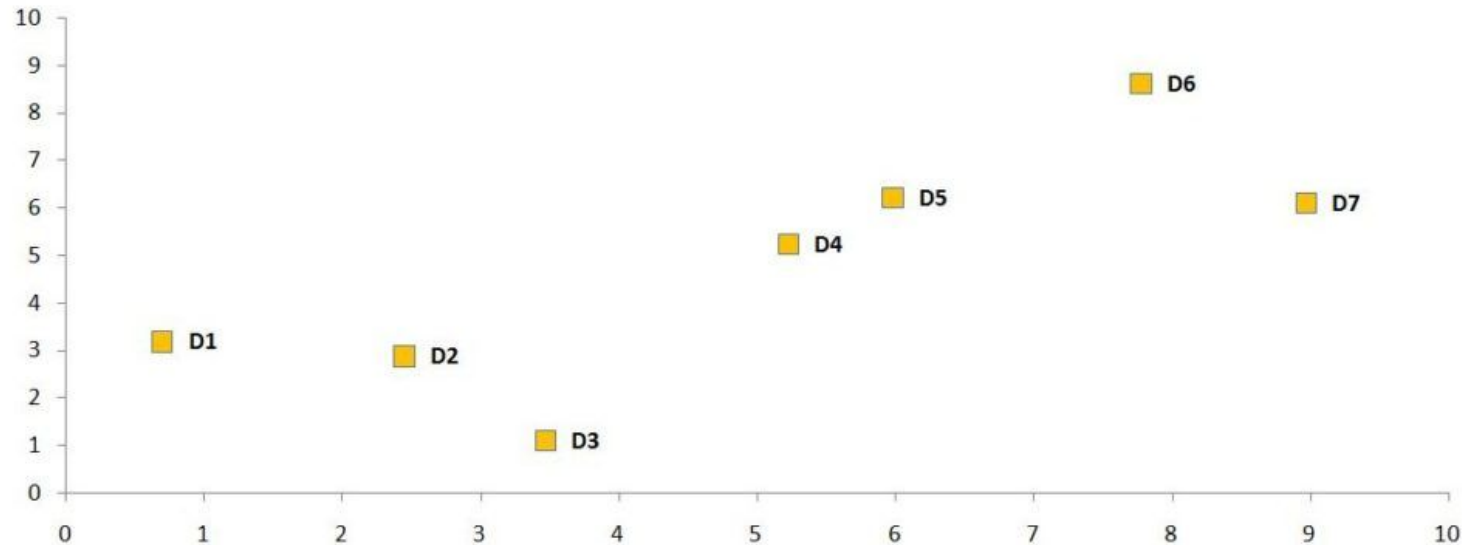
	P1	P2,P5	P3,P4
P1	0		
P2,P5	0.81841	0	
P3,P4	0.46098	0.35847	0

	P1	✓P2,P5	P3,P4
P1	0		
P2,P5	0.81841	0	
P3,P4✓	0.46098	0.35847	0

	P1	P2,P5,P3,P4
P1	0	
P2,P5,P3,P4	0.46098	0



Data Points	X	Y
D1	0.7	3.2
D2	2.45	2.89
D3	3.47	1.12
D4	5.23	5.24
D5	5.98	6.23
D6	7.778	8.63
D7	8.97	6.12



Step 1:

We first create what is known as a distance matrix. Here we compute the distance from each observation to all the observations of the dataset. The distance metric that we are using in this example is Euclidean distance.

The formula for Euclidean distance is:

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

	D1	D2	D3	D4	D5	D6	D7
D1	0	1.78	3.46	4.97	6.09	8.92	8.77
D2	1.78	0	2.04	3.64	4.86	7.83	7.28
D3	3.46	2.04	0	4.48	5.69	8.66	7.43
D4	4.97	3.64	4.48	0	1.24	4.24	3.84
D5	6.09	4.86	5.69	1.24	0	3.00	2.99
D6	8.92	7.83	8.66	4.24	3.00	0	2.78
D7	8.77	7.28	7.43	3.84	2.99	2.78	0

If we look closely, the distances in the upper and lower fold of the distance matrix are same. Also if we look diagonally, we notice that there are 0s. For example, when we calculate the distance from D1 to D1, we have 0 distance, similarly from D2 to D2 and so forth, thus the distance is nil.

	D1	D2	D3	D4	D5	D6	D7
D1	0	1.78	3.46	4.97	6.09	8.92	8.77
D2	1.78	0	2.04	3.64	4.86	7.83	7.28
D3	3.46	2.04	0	4.48	5.69	8.66	7.43
D4	4.97	3.64	4.48	0	1.24	4.24	3.84
D5	6.09	4.86	5.69	1.24	0	3.00	2.99
D6	8.92	7.83	8.66	4.24	3.00	0	2.78
D7	8.77	7.28	7.43	3.84	2.99	2.78	0

Therefore, for the sake of simplicity, we remove the duplicates and 0 values and come up with a simplified distance table.

	D1	D2	D3	D4	D5	D6	D7
D1							
D2	1.78						
D3	3.46	2.04					
D4	4.97	3.64	4.48				
D5	6.09	4.86	5.69	1.24			
D6	8.92	7.83	8.66	4.24	3.00		
D7	8.77	7.28	7.43	3.84	2.99	2.78	

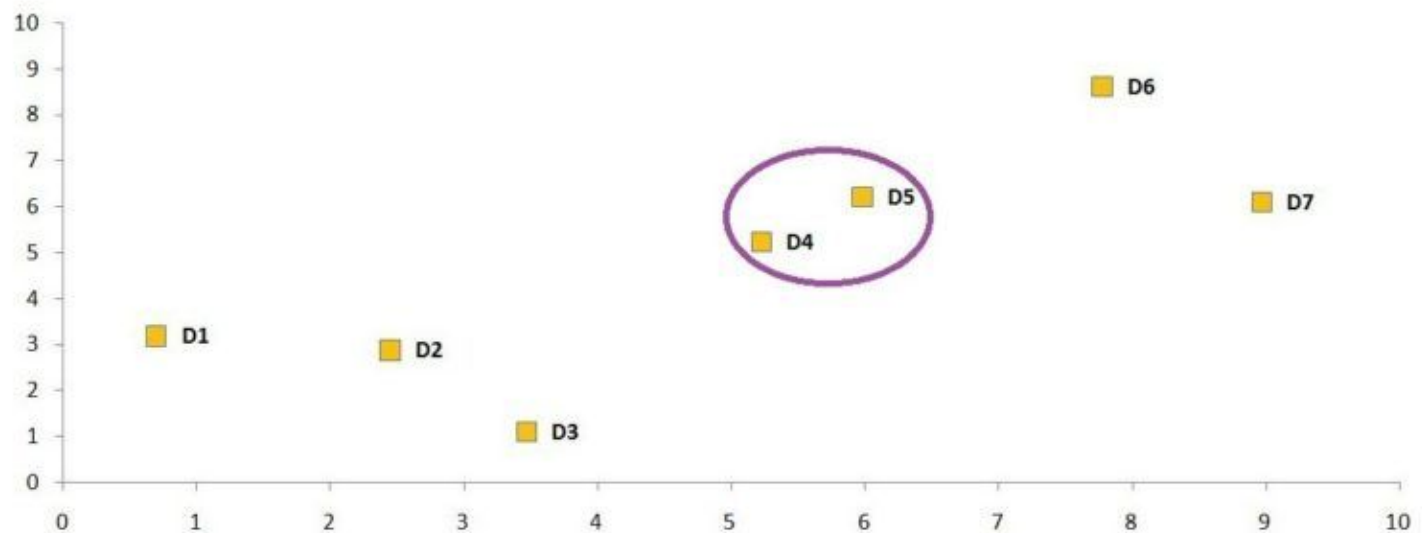
As mentioned at the beginning, Agglomerative clustering is a bottom-up approach, therefore, we merge observations together based on their similarity (minimum distance).

Therefore, we look at the table to find those data points that are closest to each other.

	D1	D2	D3	D4	D5	D6	D7
D1							
D2	1.78						
D3	3.46	2.04					
D4	4.97	3.64	4.48				
D5	6.09	4.86	5.69	1.24			
D6	8.92	7.83	8.66	4.24	3.00		
D7	8.77	7.28	7.43	3.84	2.99	2.78	

We look for the minimum value in our table and find that the minimum value is 1.24 which is the distance between D4 and D5. Therefore, the two closest data points in the dataset are D4 and D5.

Thus, we merge these two data points to form a cluster. We can now visualise this cluster on the graph and create a dendrogram for it.

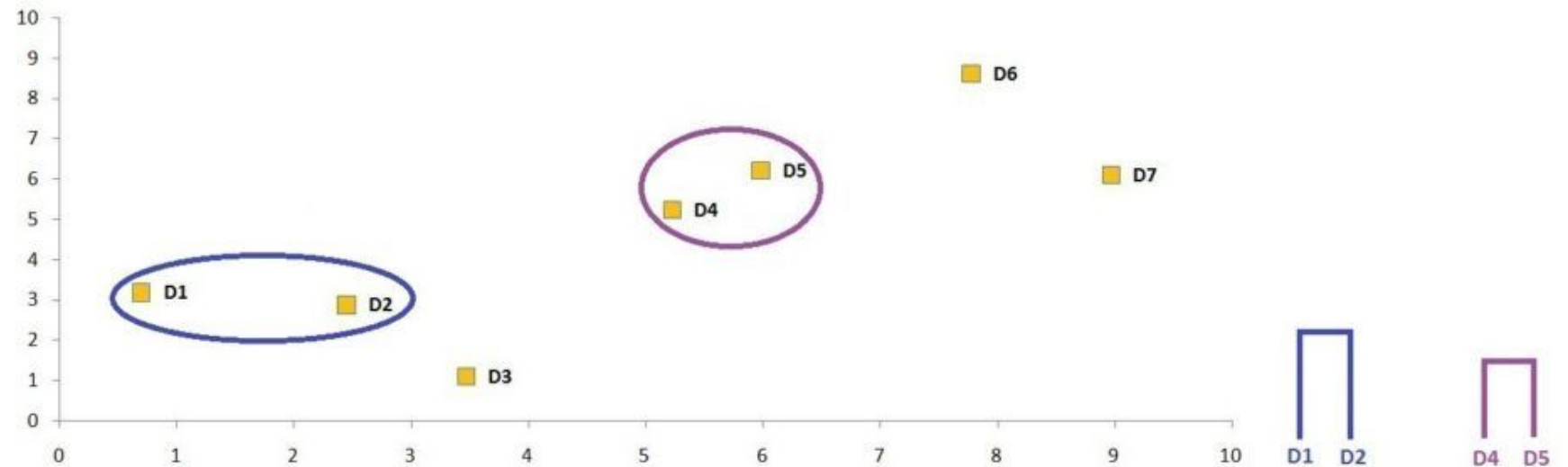


As we are using single linkage, we choose the minimum distance, therefore, we choose 4.97 and consider it as the distance between the D1 and D4, D5. If we were using complete linkage then the maximum value would have been selected as the distance between D1 and D4, D5 which would have been 6.09. If we were to use Average Linkage then the average of these two distances would have been taken. Thus, here the distance between D1 and D4, D5 would have come out to be 5.53 $(4.97 + 6.09 / 2)$.

	D1	D2	D3	D4,D5	D6	D7
D1						
D2	1.78					
D3	3.46	2.04				
D4,D5	4.97	3.64	4.48			
D6	8.92	7.83	8.66	3.00		
D7	8.77	7.28	7.43	2.99	2.78	

	D1	D2	D3	D4,D5	D6	D7
D1						
D2	1.78					
D3	3.46	2.04				
D4,D5	4.97	3.64	4.48			
D6	8.92	7.83	8.66	3.00		
D7	8.77	7.28	7.43	2.99	2.78	

From now on we will simply repeat Step 2 and Step 3 until we are left with one cluster. We again look for the minimum value which comes out to be 1.78 indicating that the new cluster which can be formed is by merging the data points D1 and D2.

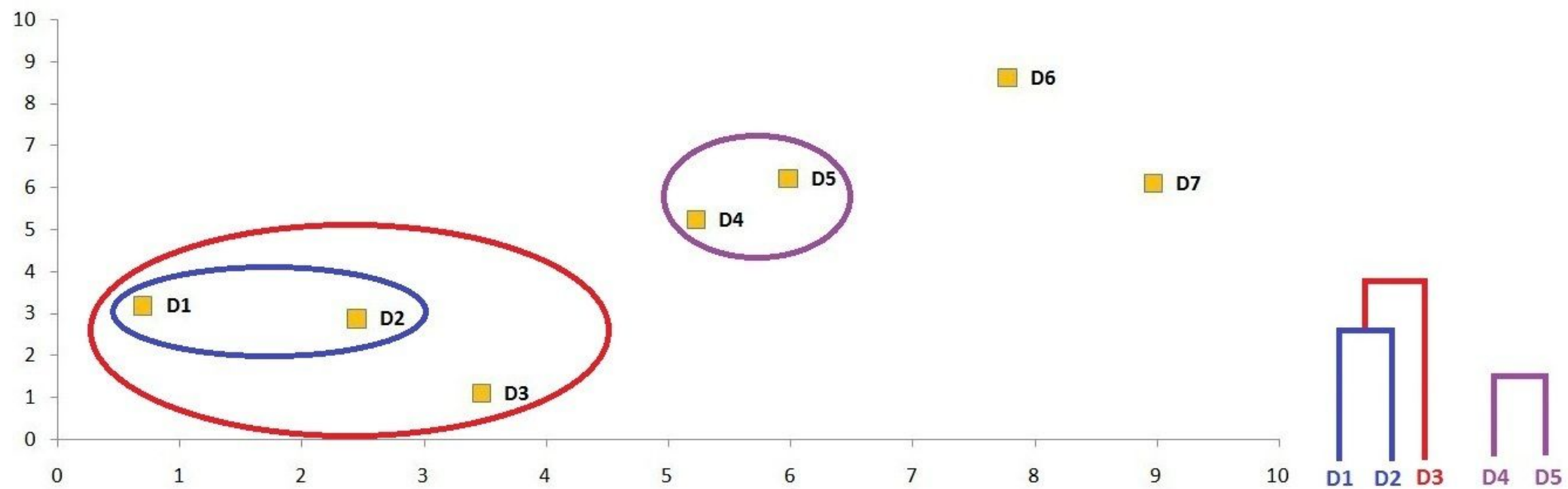


Similar to what we did in Step 3, we again recalculate the distance this time for cluster D1, D2 and come up with the following updated distance matrix.

	D1,D2	D3	D4,D5	D6	D7
D1,D2					
D3	1.78				
D4,D5	3.64	4.48			
D6	7.83	8.66	3.00		
D7	7.28	7.43	2.99	2.78	

	D1,D2	D3	D4,D5	D6	D7
D1,D2					
D3	1.78				
D4,D5	3.64	4.48			
D6	7.83	8.66	3.00		
D7	7.28	7.43	2.99	2.78	

We repeat what we did in step 2 and find the minimum value available in our distance matrix. The minimum value comes out to be 1.78 which indicates that we have to merge D3 to the cluster D1, D2.

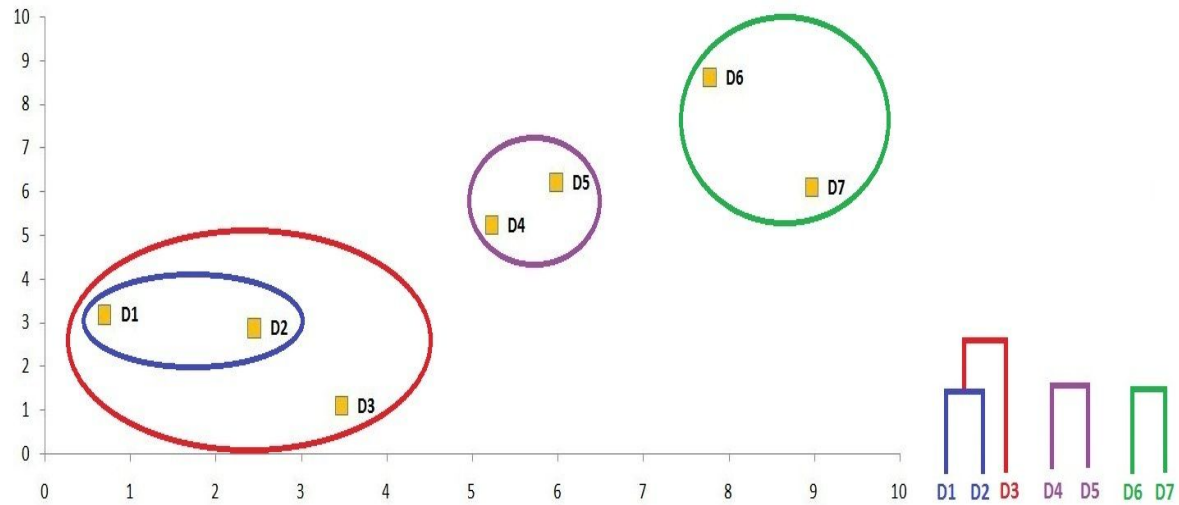


	D1,D2,D3	D4,D5	D6	D7
D1,D2,D3				
D4,D5	3.64			
D6	7.83	3.00		
D7	7.28	2.99	2.78	

Update the distance matrix using Single Link method.

	D1,D2,D3	D4,D5	D6	D7
D1,D2,D3				
D4,D5	3.64			
D6	7.83	3.00		
D7	7.28	2.99	2.78	

Find the minimum distance in the matrix.



Merge the data points accordingly and form another cluster.

	D1,D2,D3	D4,D5	D6,D7
D1,D2,D3			
D4,D5	3.64		
D6,D7	7.28	2.99	

Update the distance matrix using Single Link method.

	D1,D2,D3	D4,D5	D6,D7
D1,D2,D3			
D4,D5	3.64		
D6,D7	7.28	2.99	

The complete linkage Algorithm

- It is also called the **maximum method or the farthest neighbor method**.
- It is obtained by defining the distance between two clusters to be largest distance between a sample in one cluster and a sample in the other cluster.
- If C_i and C_j are clusters, we define:

$$D_{CL}(C_i, C_j) = \max_{a \in C_i, b \in C_j} d(a, b).$$

Example : Complete linkage algorithm

- Consider the same samples used in single linkage:
- Apply Euclidean distance and compute the distance.
- Algorithm starts with 5 clusters.
- As earlier samples 1 and 2 are the closest, they are merged first.

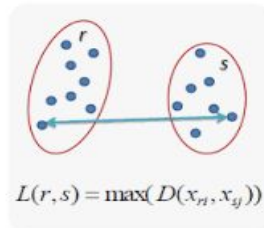
	x	y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12



	1	2	3	4	5
1	—	4.0	11.7	20.0	21.5
2	4.0	—	8.1	16.0	17.9
3	11.7	8.1	—	9.8	9.8
4	20.0	16.0	9.8	—	8.0
5	21.5	17.9	9.8	8.0	—

- While merging the maximum distance will be used to replace the distance/ cost value.
- For example, the distance between 1&3 = 11.7 and 2&3=8.1. This algorithm selects 11.7 as the distance.
- In complete linkage hierarchical clustering, the distance between two clusters is defined as **the longest distance between two points in each cluster.**

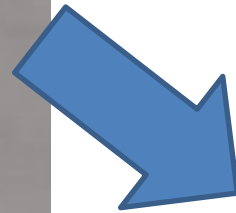
	1	2	3	4	5
1	—	4.0	11.7	20.0	21.5
2	4.0	—	8.1	16.0	17.9
3	11.7	8.1	—	9.8	9.8
4	20.0	16.0	9.8	—	8.0
5	21.5	17.9	9.8	8.0	—



	{1,2}	3	4	5
{1,2}	—	11.7	20.0	21.5
3	11.7	—	9.8	9.8
4	20.0	9.8	—	8.0
5	21.5	9.8	8.0	—

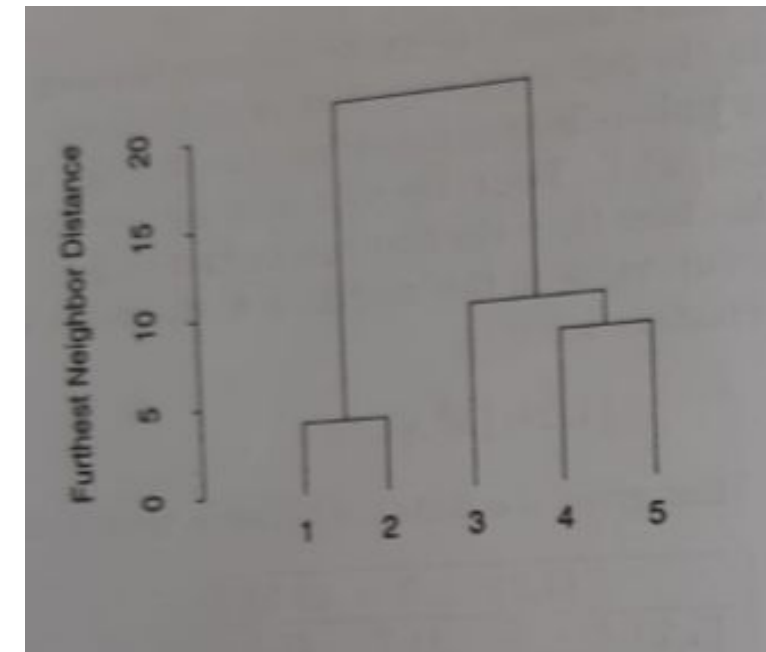
- In the next level, the smallest distance in the matrix is 8.0 between 4 and 5. Now merge 4 and 5.

	{1,2}	3	4	5
{1,2}	—	11.7	20.0	21.5
3	11.7	—	9.8	9.8
4	20.0	9.8	—	8.0
5	21.5	9.8	8.0	—



	{1,2}	3	{4,5}
{1,2}	—	11.7	21.5
3	11.7	—	9.8
{4,5}	21.5	9.8	—

- In the next step, the smallest distance is 9.8 between 3 and {4,5}, they are merged.
- At this stage we will have two clusters {1,2} and {3,4,5}.
- Notice that these clusters are different from those obtained from single linkage algorithm.
- At the next step, the two remaining clusters will be merged.
- The hierarchical clustering will be complete.
- The dendrogram is as shown in the figure.



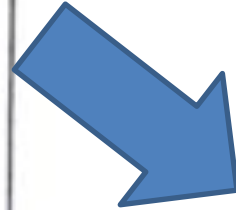
The Average Linkage Algorithm

- The average linkage algorithm, is an attempt to compromise between the extremes of the single and complete linkage algorithm.
- It is also known as the **unweighted pair group method using arithmetic averages**.

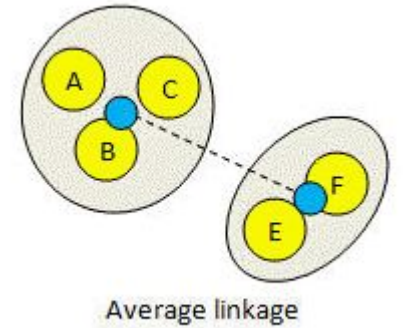
Example: Average linkage clustering algorithm

- Consider the same samples: compute the Euclidian distance between the samples

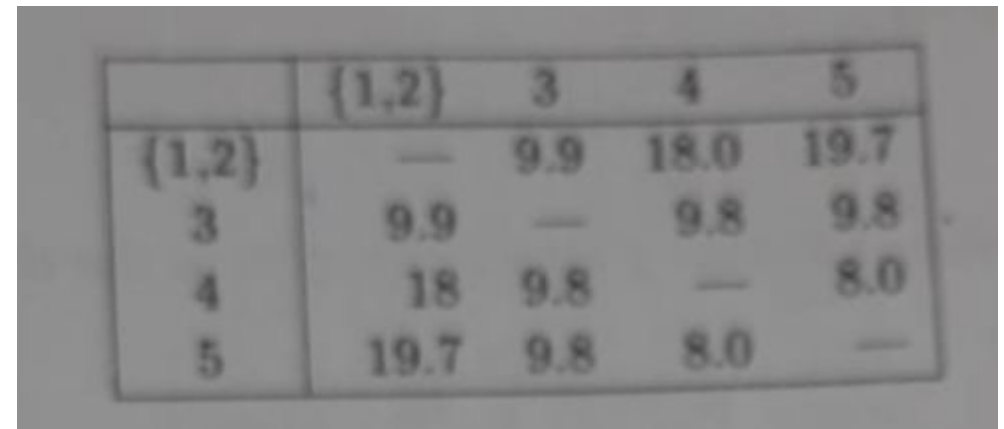
	x	y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12



	1	2	3	4	5
1	—	4.0	11.7	20.0	21.5
2	4.0	—	8.1	16.0	17.9
3	11.7	8.1	—	9.8	9.8
4	20.0	16.0	9.8	—	8.0
5	21.5	17.9	9.8	8.0	—



- In the next step, cluster 1 and 2 are merged, as the distance between them is the least.
- The distance values are computed based on the average values.
- For example distance between 1 & 3 =11.7 and 2&3=8.1 and the average is 9.9. This value is replaced in the matrix between {1,2} and 3.



A handwritten distance matrix table with 5 rows and 5 columns. The columns are labeled {1,2}, 3, 4, and 5. The rows are labeled {1,2}, 3, 4, and 5. The diagonal elements are dashes (—). The values in the matrix are: Row {1,2}: {1,2}=—, 3=9.9, 4=18.0, 5=19.7; Row 3: {1,2}=9.9, 3=—, 4=9.8, 5=9.8; Row 4: {1,2}=18, 3=9.8, 4=—, 5=8.0; Row 5: {1,2}=19.7, 3=9.8, 4=8.0, 5=—.

	{1,2}	3	4	5
{1,2}	—	9.9	18.0	19.7
3	9.9	—	9.8	9.8
4	18	9.8	—	8.0
5	19.7	9.8	8.0	—

- In the next stage 4 and 5 are merged:

	$\{1,2\}$	3	$\{4,5\}$
$\{1,2\}$	—	9.9	18.9
3	9.9	—	9.8
$\{4,5\}$	18.9	9.8	—

● **Ward's Algorithm:** This is also called minimum variance method.

- Begins with one cluster for each individual sample point.
- At each iteration, among all pairs of clusters, it merges pairs with least squared error
- The squared error for each cluster is defined as follows

If a cluster contains 'm' samples $x_1, x_2, x_3, \dots, x_m$ and

where x_i is the feature vector $(x_{i1}, x_{i2}, \dots, x_{id})$, the squared error for sample x_i ,

which is the squared Euclidean distance from the mean----- $\sum_{j=1}^d (x_{ij} - \mu_j)^2$

- Where μ_j is the mean of the feature j for the values in the cluster

$$\mu_j = \frac{1}{m} \sum_{i=1}^m (x_{ij})$$

- **The squared error E for the entire cluster is the sum of the squared errors for the samples**

$$E = \sum_{i=1}^m \sum_{j=1}^d (x_{ij} - \mu_j)^2 = m \sigma^2$$

- The vector composed of the means of each feature, $(\mu_1, \dots, \mu_d) = \mu$, is called the mean of the vector or centroid of the cluster.
- The squared error is thus the total variance of the cluster σ^2 times the number of samples m .

Clusters	Squared Error, E
$\{1,2\},\{3\},\{4\},\{5\}$	8.0
$\{1,3\},\{2\},\{4\},\{5\}$	68.5
$\{1,4\},\{2\},\{3\},\{5\}$	200.0
$\{1,5\},\{2\},\{3\},\{4\}$	232.0
$\{2,3\},\{1\},\{4\},\{5\}$	32.5
$\{2,4\},\{1\},\{3\},\{5\}$	128.0
$\{2,5\},\{1\},\{3\},\{4\}$	160.0
$\{3,4\},\{1\},\{2\},\{5\}$	48.5
$\{3,5\},\{1\},\{2\},\{4\}$	48.5
$\{4,5\},\{1\},\{2\},\{3\}$	32.0

	x	y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12

5.5: Squared errors for each way of creating four clusters.

Clusters	Squared Error, E
$\{1,2,3\},\{4\},\{5\}$	72.7
$\{1,2,4\},\{3\},\{5\}$	224.0
$\{1,2,5\},\{3\},\{4\}$	266.7
$\{1,2\},\{3,4\},\{5\}$	56.5
$\{1,2\},\{3,5\},\{4\}$	56.5
$\{1,2\},\{4,5\},\{3\}$	40.0

Figure 5.6: Squared errors for three clusters.

5.2. HIERARCHICAL CLUSTERING

The algorithm begins with 5 clusters with a sample each

The squared error at this point is zero

10 Possible ways to merge a pair of clusters: Merge {1} and {2} or {1} and {3} so on.

Since the sample 1 has feature vector (4,4) and sample 2 has feature vector (8,4), the feature mean is (6,4)

The squared error of cluster {1,2} is

$$(4 - 6)^2 + (8 - 6)^2 + (4 - 4)^2 + (4 - 4)^2 = 8$$

Thus the total squared error for the clusters {1,2},{3},{4},{5} is $8+0+0+0 = 8$

Similarly compute the squared error for other merged cluster combinations.

From the figure 5.5, it is evident that 8 is the least squared error and hence choose it and repeat the procedure as shown in 5.6

Clusters	Squared Error, E
{1,2},{3},{4},{5}	8.0
{1,3},{2},{4},{5}	68.5
{1,4},{2},{3},{5}	200.0
{1,5},{2},{3},{4}	232.0
{2,3},{1},{4},{5}	32.5
{2,4},{1},{3},{5}	128.0
{2,5},{1},{3},{4}	160.0
{3,4},{1},{2},{5}	48.5
{3,5},{1},{2},{4}	48.5
{4,5},{1},{2},{3}	32.0

Figure 5.5: Squared errors for each way of creating four clusters.

Clusters	Squared Error, E
{1,2,3},{4},{5}	72.7
{1,2,4},{3},{5}	224.0
{1,2,5},{3},{4}	266.7
{1,2},{3,4},{5}	56.5
{1,2},{3,5},{4}	56.5
{1,2},{4,5},{3}	40.0

Figure 5.6: Squared errors for three clusters.

Clusters	Squared Error, E
$\{1,2,3\}, \{4,5\}$	104.7
$\{1,2,4,5\}, \{3\}$	380.0
$\{1,2\}, \{3,4,5\}$	94.0

Figure 5.7: Squared errors for two clusters.



Figure 5.8: Dendrogram for Ward's method.

DIVISIVE CLUSTERING

All the points in the dataset belong to one cluster and split is performed recursively as one moves down the hierarchy.

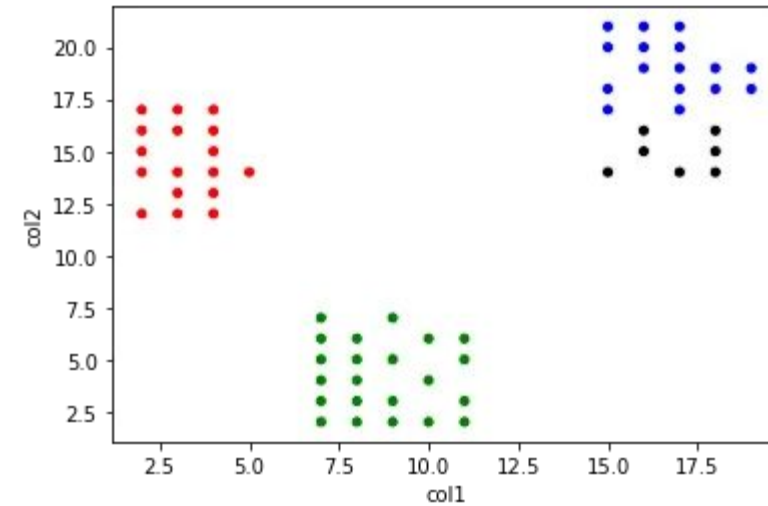
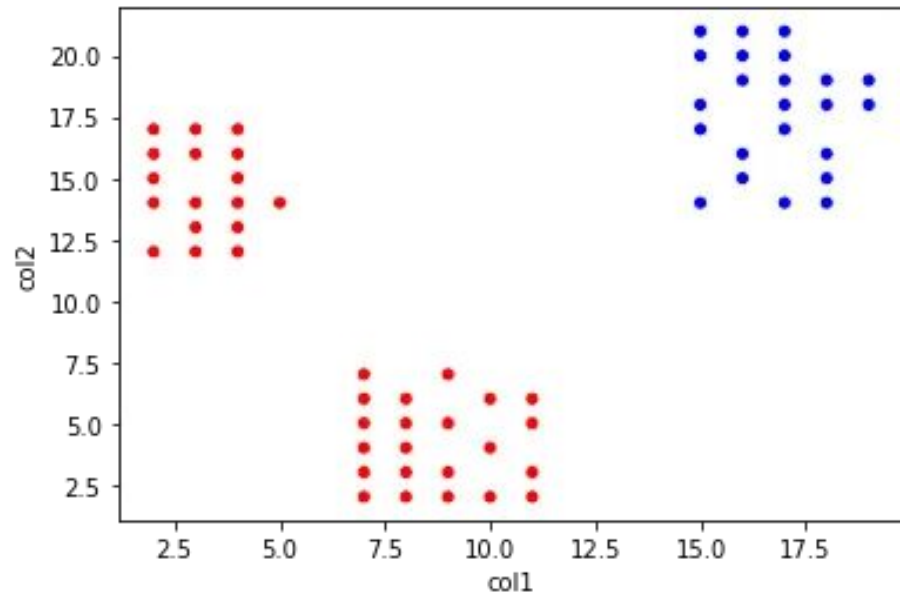
Steps of Divisive Clustering:

1. Initially, all points in the dataset belong to one single cluster.
2. Partition the cluster into two least similar cluster
3. Proceed recursively to form new clusters until the desired number of clusters is obtained.

How to choose which cluster to split?

Check the sum of squared errors of each cluster and choose the one with the largest value.

Once we have decided to split which cluster, then the question arises on how to split the chosen cluster into 2 clusters. One way is to use [Ward's criterion](#) to chase for the largest reduction in the difference in the SSE criterion as a result of the split.



Partitional Clustering:

Agglomerative clustering creates a series of Nested clusters.

In partitional clustering the goal is to usually create one set of clusters that partitions the data into similar groups.

Samples close to one another are assumed to be in one cluster.

This is the goal of partitional clustering.

Partitional clustering algorithm is used to divide the data set into two groups.

Then each group is divided into two parts and so on, a hierarchical dendrogram could be produced from the top down.

The hierarchy is produced using divisive technique is more general than the bottom-up technique used by agglomerative technique because the groups can be divided into more than two subgroups in one step.

One of the simplest partitioning algorithm is the Forgy's algorithm.

Apart from the data, the input to the algorithm is 'k' , the number of clusters to be constructed 'k' samples are called seed points.

The seed points could be chosen randomly, or some knowledge of the desired could be used to guide their selection.

Forgy's Algorithm

- 1. Initialize the cluster centroid to the seed points.**
- 2. For each sample, find the cluster centroid nearest to it. Put the sample in the nearest cluster identified with the cluster centroid.**
- 3. Compute the centroids of the resulting clusters.**
- 4. Repeat the steps 2 to 3 until sample points do not change their clusters.**
- 5. stop**

Algorithm

Input: D is a dataset containing n objects, k is the number of cluster

Output: A set of k clusters

Steps:

1. Randomly choose k objects from D as the initial cluster centroids.
2. **For** each of the objects in D **do**
 - Compute distance between the current objects and k cluster centroids
 - Assign the current object to that cluster to which it is closest.
3. Compute the “cluster centers” of each cluster. These become the new cluster centroids.
4. Repeat step 2-3 until the convergence criterion is satisfied
5. Stop

Consider the Data points listed in the table and set $k = 2$ to produce two clusters
Use the first two samples (4,4) and (8,4) as the seed points.
Now applying the algorithm by computing the distance from each cluster centroid and assigning them to the clusters:

Data Points	X	Y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12

Sample	Nearest Cluster Centroid
(4,4)	(4,4)
(8,4)	(8,4)
(15,8)	(8,4)
(24,4)	(8,4)
(24,12)	(8,4)

The clusters $\{(4,4)\}$ and $\{(8,4),(15,8),(24,4),(24,12)\}$ are formed.

Now re-compute the cluster centroids

The first cluster $(4,4)$ and

The second cluster centroid is $x = (8+15+24+24)/4 = 17.75$

$$y = (4+8+4+12)/4 = 7$$

Sample	Nearest Cluster Centroid
(4,4)	(4,4)
(8,4)	(4,4)
(15,8)	(17.75, 7)
(24,4)	(17.75, 7)
(24,12)	(17.75, 7)

The clusters $\{(4,4),(8,4)\}$ and $\{(15,8),(24,4),(24,12)\}$ are formed.

Now re-compute the cluster centroids

The first cluster centroid $x = (4+8)/2 = 6$ and $y = (4+4)/2 = 4$

The second cluster centroid is $x = (15+24+24)/3 = 21$

$$y = (8+4+12)/4 = 12$$

In the next step notice that the cluster centroid does not change

And samples also do not change the clusters.

Algorithm terminates.

Sample	Nearest Cluster Centroid
(4,4)	(6,4)
(8,4)	(6,4)
(15,8)	(21,12)
(24,4)	(21,12)
(24,12)	(21,12)

Example-2 Illustration Forgy's clustering algorithms

Plotting data of Table

A_1	A_2
6.8	12.6
0.8	9.8
1.2	11.6
2.8	9.6
3.8	9.9
4.4	6.5
4.8	1.1
6.0	19.9
6.2	18.5
7.6	17.4
7.8	12.2
6.6	7.7
8.2	4.5
8.4	6.9
9.0	3.4
9.6	11.1

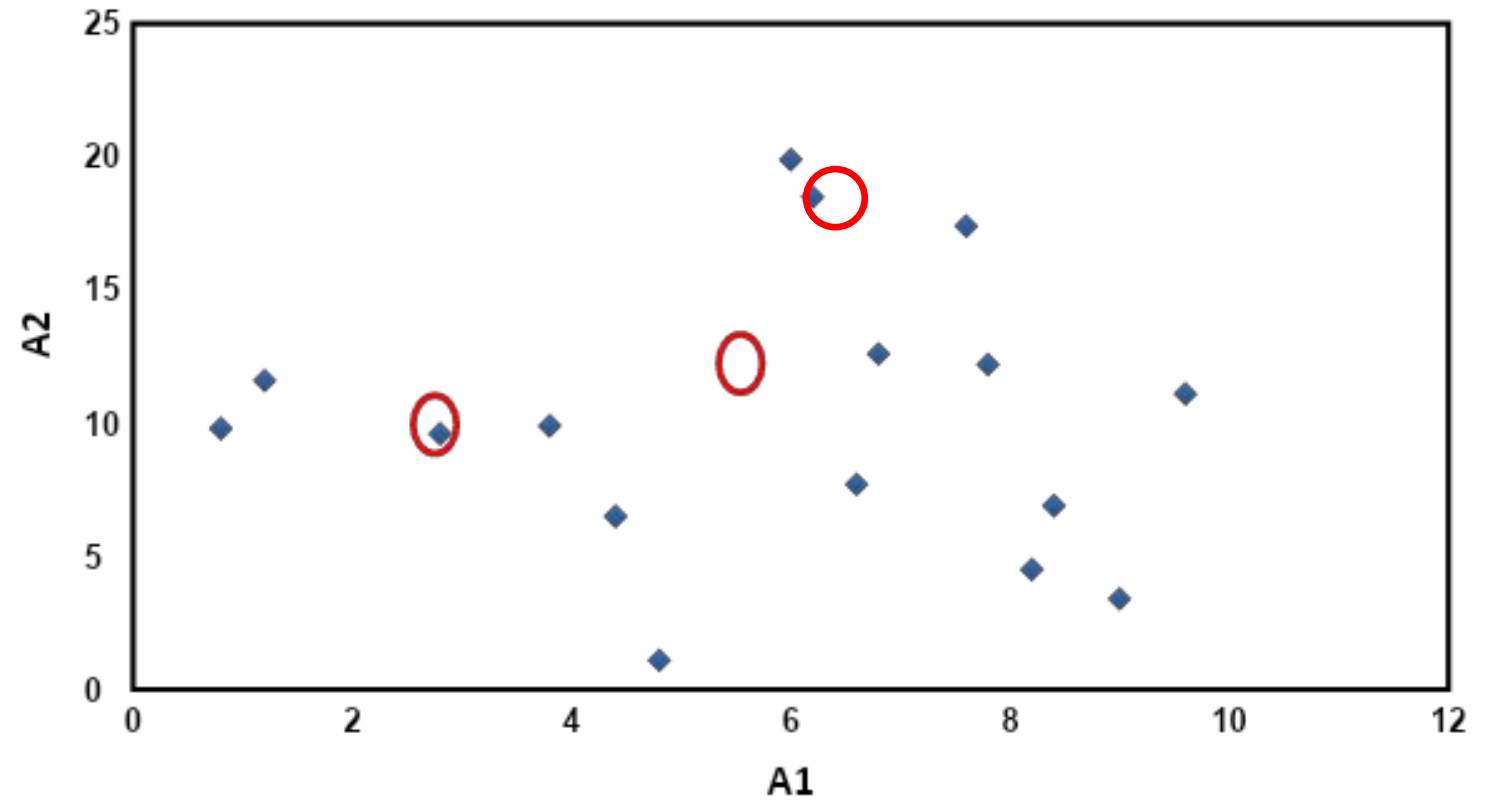


Illustration Forgy's clustering algorithms

- Suppose, $k=3$. Three objects are chosen at random shown as circled. These three centroids are shown below.

Initial Centroids chosen randomly

Centroid	Objects	
c_1	3.8	9.9
c_2	7.8	12.2
c_3	6.2	18.5

- Let us consider the Euclidean distance measure (L_2 Norm) as the distance measurement in our illustration.
- Let d_1 , d_2 and d_3 denote the distance from an object to c_1 , c_2 and c_3 respectively. The distance calculations are shown in Table 16.2.
- Assignment of each object to the respective centroid is shown in the right-most column and the clustering so obtained is shown in Fig 16.2.

Illustration Forgy's clustering algorithms

A_1	A_2	d_1	d_2	d_3	cluster
6.8	12.6	4.0	1.1	5.9	2
0.8	9.8	3.0	7.4	10.2	1
1.2	11.6	3.1	6.6	8.5	1
2.8	9.6	1.0	5.6	9.5	1
3.8	9.9	0.0	4.6	8.9	1
4.4	6.5	3.5	6.6	12.1	1
4.8	1.1	8.9	11.5	17.5	1
6.0	19.9	10.2	7.9	1.4	3
6.2	18.5	8.9	6.5	0.0	3
7.6	17.4	8.4	5.2	1.8	3
7.8	12.2	4.6	0.0	6.5	2
6.6	7.7	3.6	4.7	10.8	1
8.2	4.5	7.0	7.7	14.1	1
8.4	6.9	5.5	5.3	11.8	2
9.0	3.4	8.3	8.9	15.4	1
9.6	11.1	5.9	2.1	8.1	2

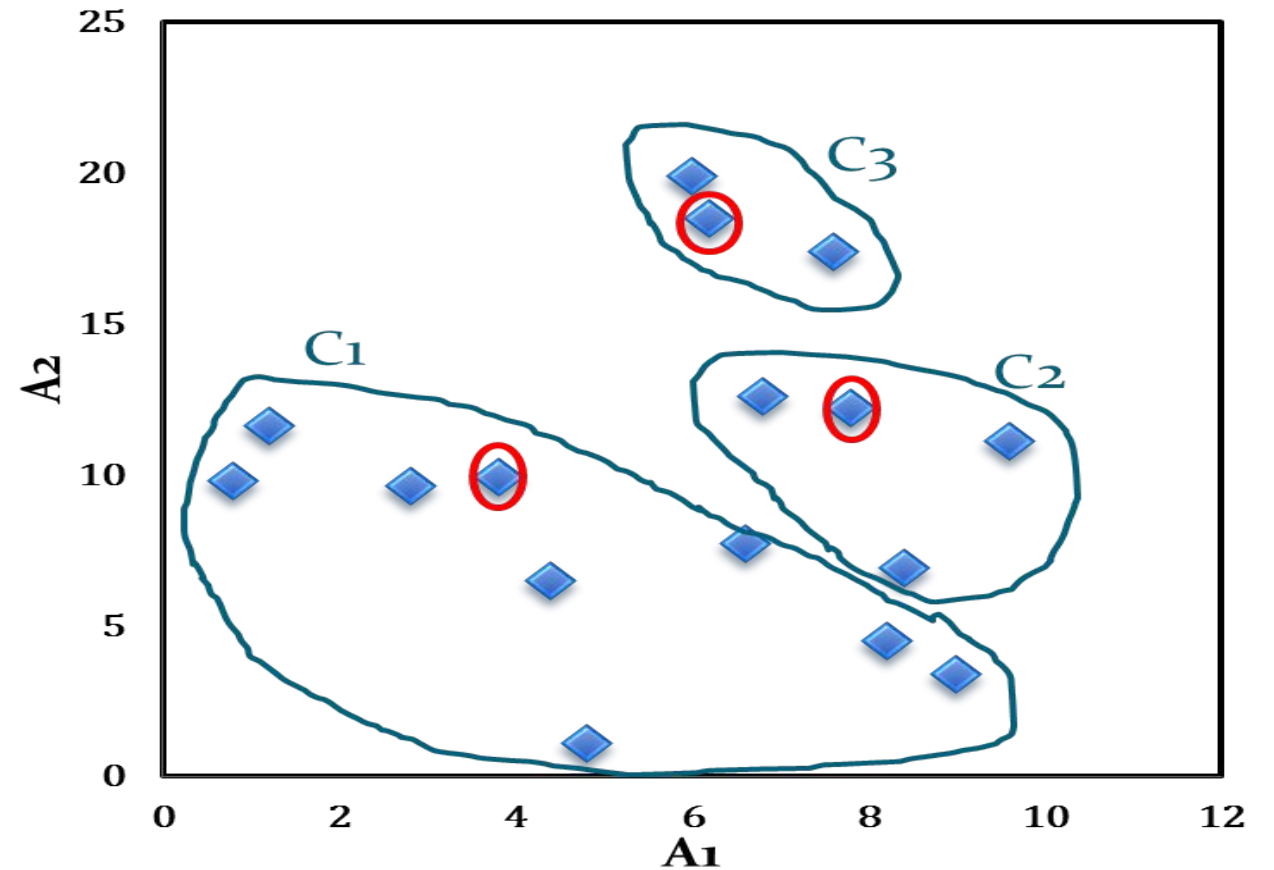
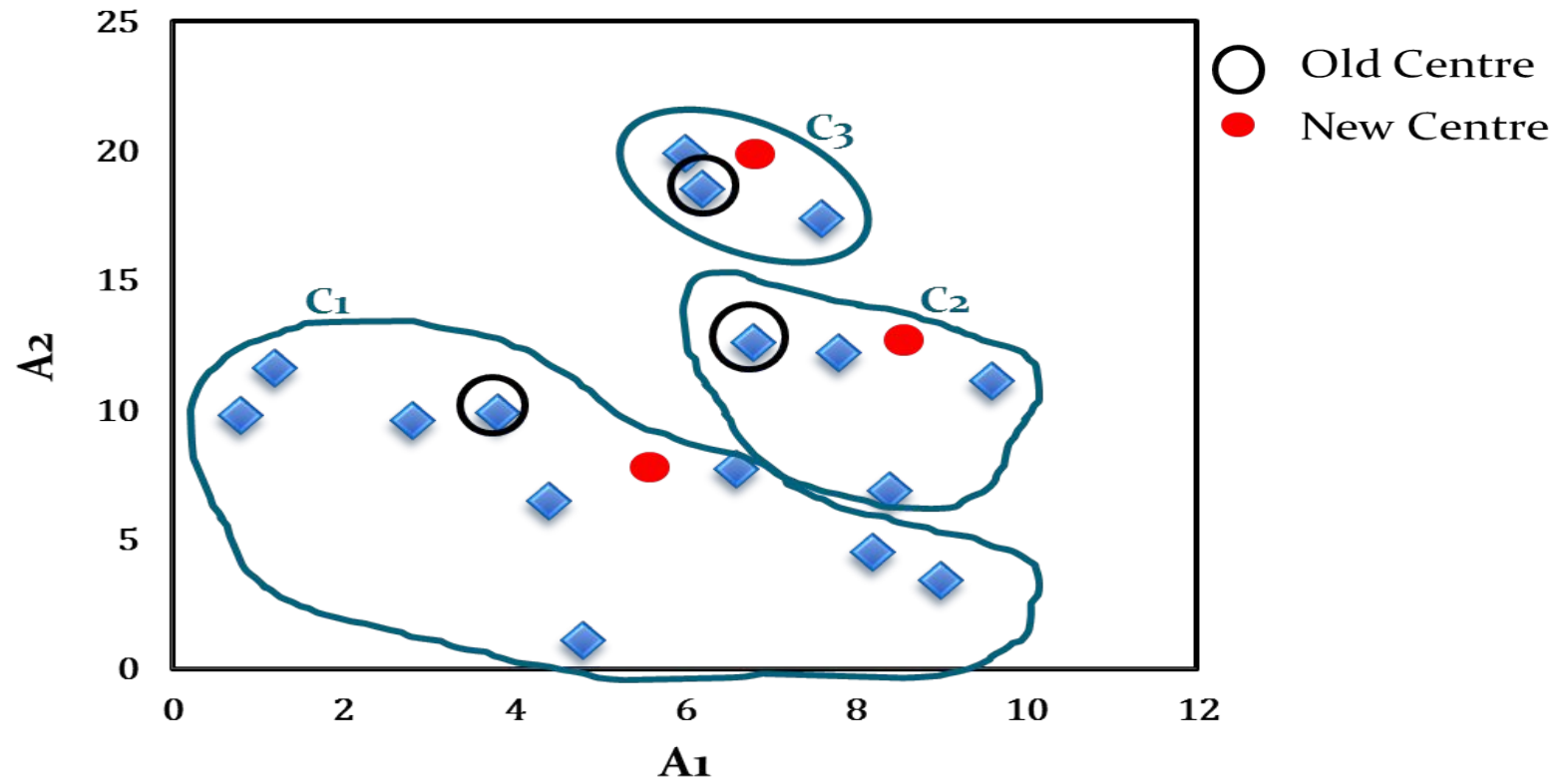


Illustration Forgý's clustering algorithms

The calculation new centroids of the three cluster using the mean of attribute values of A_1 and A_2 is shown in the Table below. The cluster with new centroids are shown in Fig 16.3.

Calculation of new centroids

New Centroid	Objects	
c_1	4.6	7.1
c_2	8.2	10.7
c_3	6.6	18.6

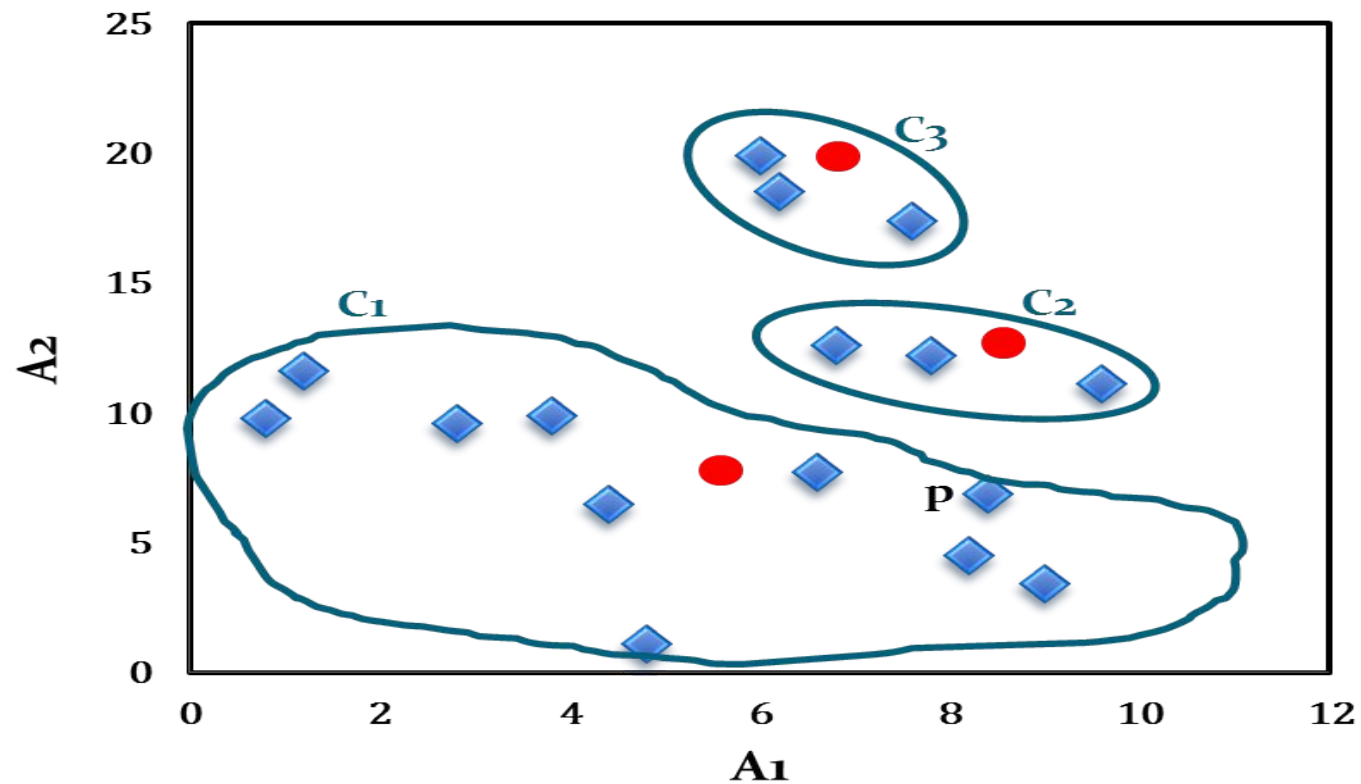


Next cluster with new centroids

Illustration of Forgy's clustering algorithms

We next reassign the 16 objects to three clusters by determining which centroid is closest to each one. This gives the revised set of clusters shown in.

Note that point p moves from cluster C_2 to cluster C_1 .



Cluster after first iteration

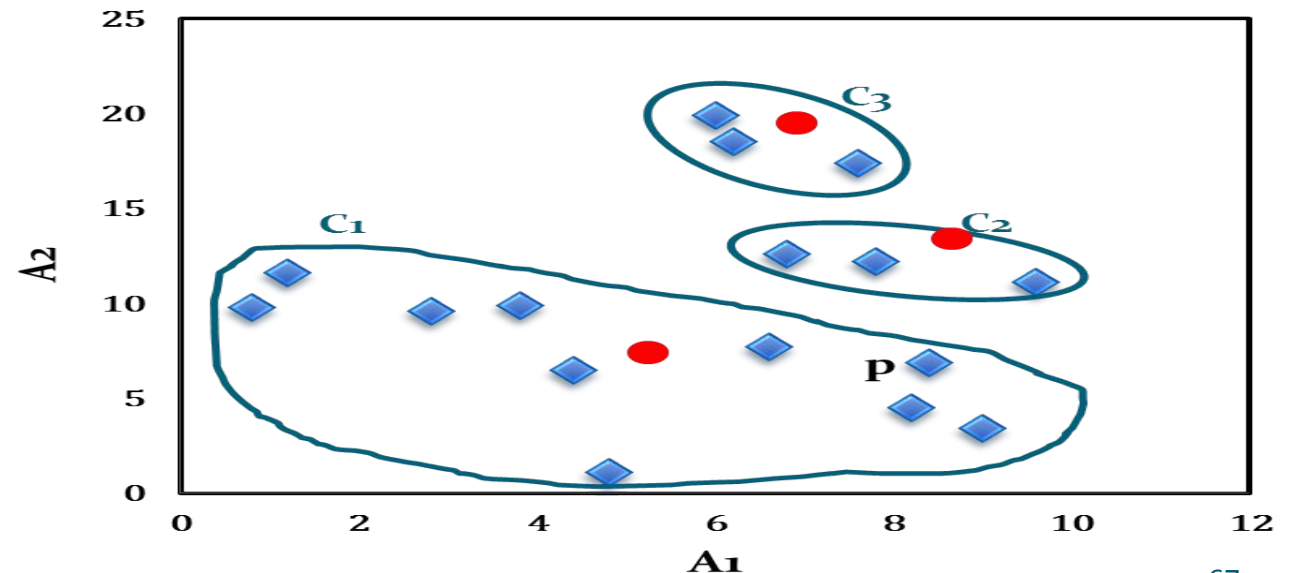
Illustration of Forgy's clustering algorithms

- The newly obtained centroids after second iteration are given in the table below. Note that the centroid c_3 remains unchanged, where c_2 and c_1 changed a little.
- With respect to newly obtained cluster centres, 16 points are reassigned again. These are the same clusters as before. Hence, their centroids also remain unchanged.
- Considering this as the termination criteria, the k-means algorithm stops here. Hence, the final cluster in Figure is same as Fig.

Cluster centres after second iteration

Centroid	Revised Centroids	
c_1	5.0	7.1
c_2	8.1	12.0
c_3	6.6	18.6

Fig 16.5: Cluster after Second iteration



Apply Forgy's algorithm for the following dataset with $K = 2$;

Sample	X	Y
1	0.0	0.5
2	0.5	0.0
3	1.0	0.5
4	2.0	2.0
5	3.5	8.0
6	5.0	3.0
7	7.0	3.0

Pros

- Simple, fast to compute

- Converges to local minimum of within-cluster squared error

Cons

- Setting k

- Sensitive to initial centres

- Sensitive to outliers

- Detects spherical clusters

K-Means Algorithm: It is similar to Forgy's algorithm. The k-means algorithm differs from Forgy's algorithm in that the centroids of the clusters are recomputed as soon as sample joins a cluster. Also unlike Forgy's algorithm which is iterative in nature, the k-means only two passes through the data set.

The k-means Algorithm

1. Begin with k clusters, each consisting of one of the first k samples. For each remaining $n-k$ samples, find the centroid nearest it. Put the sample in the cluster identified with this nearest centroid. After each sample is assigned, re-compute the centroid of the altered cluster.
2. Go through the data a second time. For each sample, find the centroid nearest it. Put the sample in the cluster identified with the nearest cluster. (**During this step do not recompute the centroid**)

Apply k-means Algorithm on the following sample points

Data Points	X	Y
1	4	4
2	8	4
3	15	8
4	24	4
5	24	12

Begin with two clusters $\{(8,4)\}$ and $\{(24,4)\}$ with the centroids

$(8,4)$ and $(24,4)$

For each remaining samples, find the nearest centroid and put it in that Cluster.

Then re-compute the centroid of the cluster.

The next sample $(15,8)$ is closer to $(8,4)$ so it joins the cluster $\{(8,4)\}$.

The centroid of the first cluster is updated to $(11.5,6)$.

$(8+15)/2 = 11.5$ and $(4+8)/2 = 6$.

The next sample is $(4,4)$ is nearest to the centroid $(11.5,6)$ so it joins the cluster $\{(8,4),(15,8),(4,4)\}$.

Now the new centroid of the cluster is $(9,5.3)$

The next sample $(24,12)$ is closer to centroid $(24,4)$ and joins the cluster $\{(24,4),(24,12)\}$.

Now the new centroid of the second cluster is updated to (24,8).

At this point step1 is completed.

For step2 examine the samples one by one and put each sample in the identified with the nearest cluster centroid.

The resulting clusters are

$\{(8,4),(15,8),4,4)\}$ and $\{(24,12),(24,4)\}$

Sample	Distance to centroid (9.5,3)	Distance to centroid (24,8)
(8,4)	1.6	16.5
(24,4)	15.1	4.0
(15.8)	6.6	9.0
(4,4)	6.6	40.0
(24,12)	16.4	4.0

Note: The goal of Forgy's and k-Means algorithm is to minimize the squared error for a fixed number of clusters.

These algorithms assign samples to clusters so as to reduce the squared error and in the iterative versions, they stop when there is no further reduction..

Comments on Partitional algorithm

Value of k :

- The k-means algorithm produces only one set of clusters, for which, user must specify the desired number, k of clusters.
- In fact, k should be the **best guess** on the number of clusters present in the given data. Choosing the best value of k for a given dataset is, therefore, an issue.
- We may not have an idea about the possible number of clusters for high dimensional data, and for data that are not scatter-plotted.
- Further, possible number of clusters is hidden or ambiguous in image, audio, video and multimedia clustering applications etc.
- There is no principled way to know what the value of k ought to be. We may try with successive value of k starting with 2.
- The process is stopped when two consecutive k values produce more-or-less identical results (with respect to some cluster quality estimation).
- Normally $k \ll n$ and there is heuristic to follow $k \approx \sqrt{n}$.

Choosing initial centroids:

- Another requirement in the k-Means algorithm to choose initial cluster centroid for each k would be clusters.
- It is observed that the k-Means algorithm terminate whatever be the initial choice of the cluster centroids.
- It is also observed that initial choice influences the ultimate cluster quality. In other words, the result may be trapped into local optima, if initial centroids are chosen properly.
- One technique that is usually followed to avoid the above problem is to choose initial centroids in multiple runs, each with a different set of randomly chosen initial centroids, and then select the best cluster (with respect to some quality measurement criterion, e.g. SSE).
- However, this strategy suffers from the combinational explosion problem due to the number of all possible solutions.

Original image



10 colors



8 colors



6 colors



4 colors

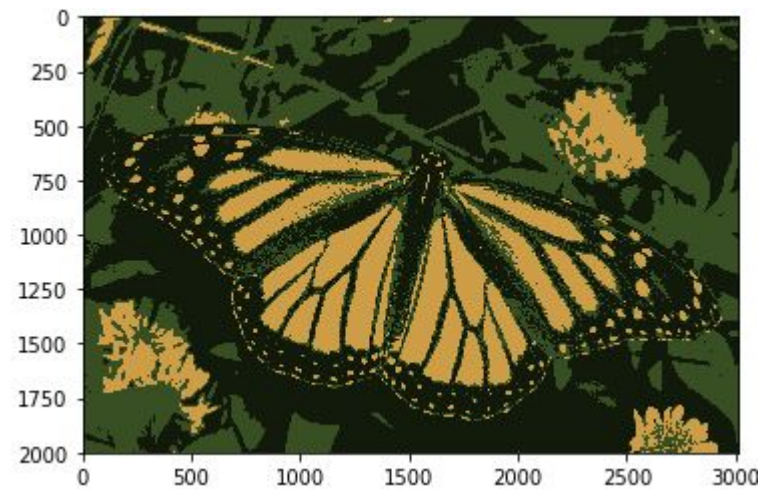


2 colors

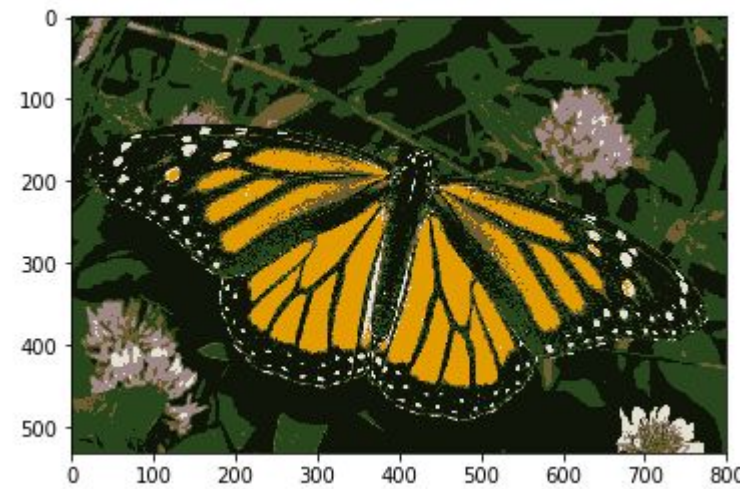




Original Image



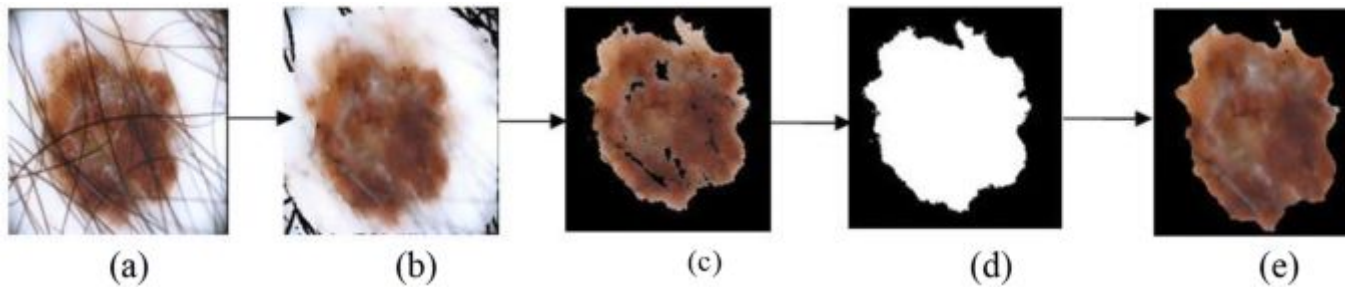
Segmentation into 3 regions
using $K = 3$



Segmentation into 6 regions
using $k = 6$



Face Region Detection Using Skin Segmentation



SKIN LESION SEGMENTATION

2-Using K-Means clustering algorithm, create two clusters for the following objects. Consider Object#1 and Object#4 as the initial centers. (You should not use R for this question-Please show your works)

Object#	A	B
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

PART A

Point	X	Y
P1	0.35	0.53
P2	0.65	0.70
P3	0.35	0.50
P4	0.45	0.65
P5	0.50	0.60
P6	0.45	0.85

Figure 1.

1. Based on the point given in Figure 1, calculate the dissimilarity matrix by using the Euclidean Distance formula.

(3 marks)

2. Find the two shortest distance from the dissimilarity matrix table for the first two cluster. After that, find the point with the minimum distance in the existing clusters by using:

Dataset1
Complete Linkage

Dataset2
Single Linkage

Dataset3
Ward's Algorithm

Data #	x	y
1	1.90	0.97
2	1.76	0.84
3	2.32	1.63
4	2.31	2.09
5	1.14	2.11
6	5.02	3.02
7	5.74	3.84
8	2.25	3.47
9	4.71	3.60
10	3.17	4.96

Find the group of the following dataset using k-means, when $k=2$, $k=3$, $k=4$.

	X	Y
P1	0.40	0.53
P2	0.22	0.38
P3	0.35	0.32
P4	0.26	0.19
P5	0.08	0.41
P6	0.45	0.30

Dataset4
Forgy's Algorithm

Figure 2: K-means dataset co-ordinates

Suppose you want to cluster the 8 points shown below:



	x	y
P1	2	10
P2	2	5
P3	8	4
P4	5	8
P5	7	5
P6	6	4
P7	1	2
P8	4	9



Dataset5
K-Means Algorithm