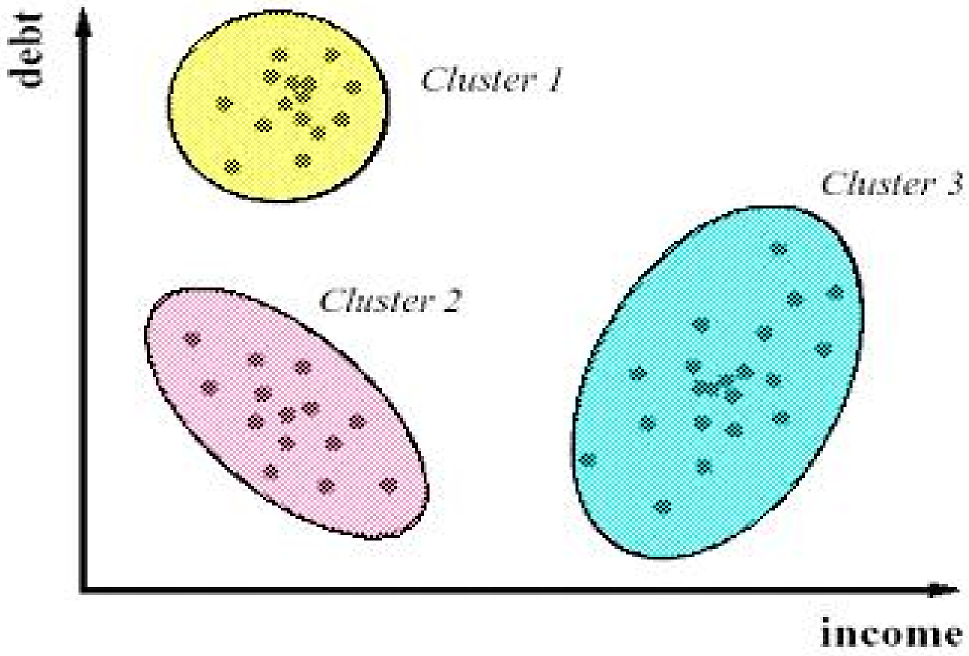
CLUSTERING

What is Cluster

- A cluster is a subset of objects which are “similar”
- A subset of objects such that the distance between any two objects in the cluster is less than the distance between any object in the cluster and any object not located inside it.

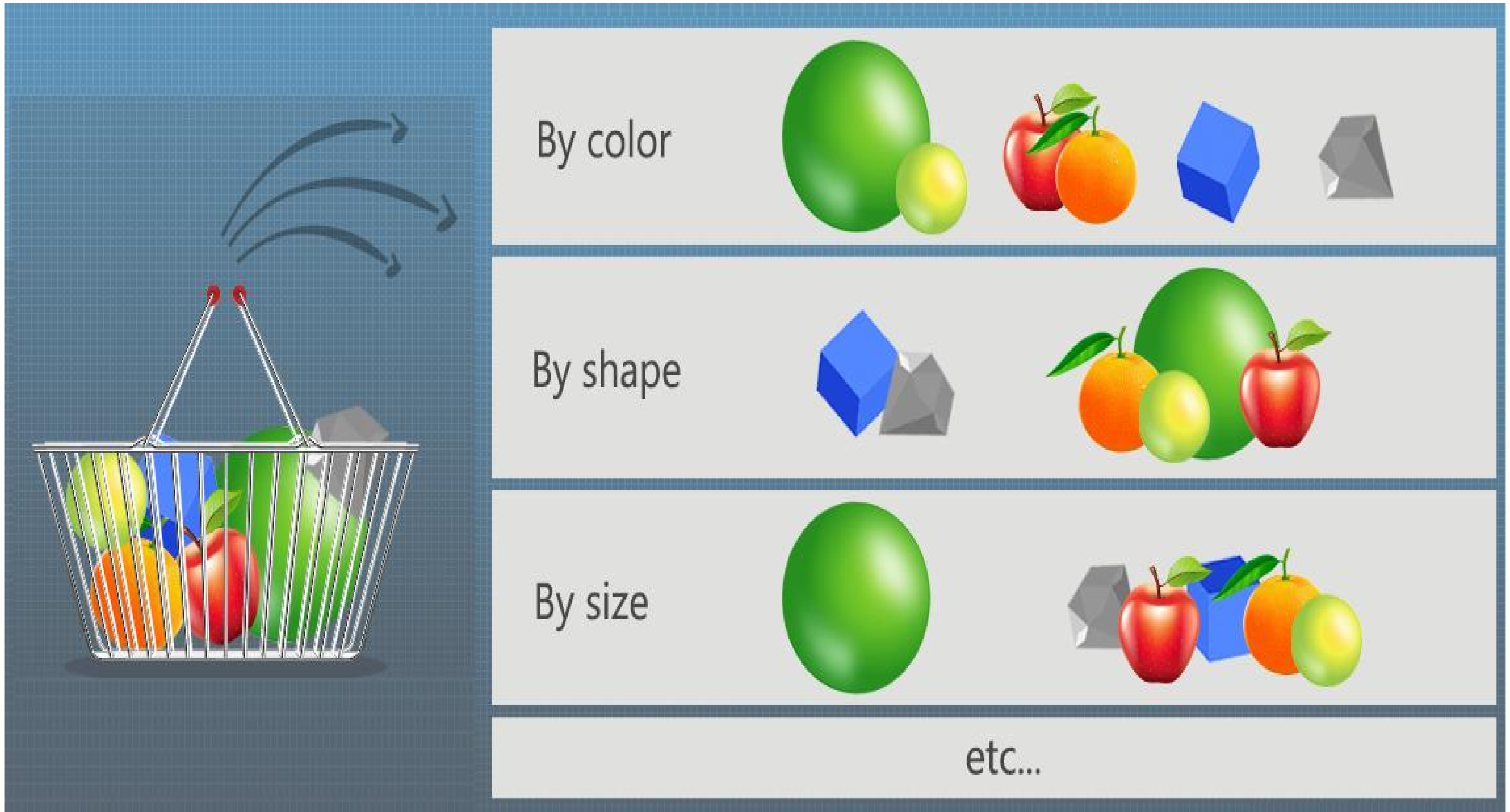
What is Clustering

- Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups.
- Clustering is the unsupervised learning where we don’t know the class labels or number of labels beforehand.
- Clustering algorithm forms the groups of objects on the basis of similarity and dissimilarity between them.



Clustering Example

Imagine of a number of objects in a basket. Each item has a distinct set of features (size, shape, color, etc.). Now the task is to group each of the objects in the basket. A natural first question to ask is, ‘on what basis these objects should be grouped?’ Perhaps size, shape, or color.



Clustering Application

- **Marketing**: It can be used to characterize & discover customer segments for marketing purposes.
- **Biology:** It can be used for classification among different species of plants and animals.
- **Libraries:** It is used in clustering different books on the basis of topics and information.
- **Insurance:** It is used to acknowledge the customers, their policies and identifying the frauds.
- **City Planning:** It is used to make groups of houses and to study their values based on their geographical locations and other factors present.
- **Earthquake studies:** By learning the earthquake affected areas, the dangerous zones can be determined.
- **Healthcare:** It can be used in identifying and classifying the cancerous gene.
- **Search Engine:** It is the backbone behind the search engines. Search engines try to group similar objects in one cluster and the dissimilar objects far from each other.
- **Education:** It can be used to monitor the students' academic performance. Based on the students' score they are grouped into different-different clusters, where each clusters denoting the different level of performance.

Requirements of Clustering in Data Mining

- **Scalability** – Highly scalable clustering algorithms are needed to deal with large databases.
- **Ability to deal with different kinds of attributes -** Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.
- **Discovery of clusters with attribute shape -** The clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.
- **High dimensionality -** The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.
- **Ability to deal with noisy data** – Dataset contain noisy, missing or erroneous data. Some algorithms are sensitive to such data and may lead to poor quality clusters.
- **Interpretability** – The clustering results should be interpretable, comprehensible, and usable.

What is Good Clustering?

- A good clustering method will produce high quality clusters in which:
  - The intra-class (that is, intra-cluster) similarity is high.
  - The inter-class similarity is low.
- The quality of a clustering result also 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.
- However, objective evaluation is problematic: usually done by human / expert inspection.

Major Clustering Approaches

- **Partitioning algorithms:** Construct various partitions and then evaluate them by some criterion
- **Hierarchy algorithms:** Create a hierarchical decomposition of the set of data (or objects) using some criterion
- **Density-based:** based on connectivity and density functions
- **Grid-based:** based on a multiple-level granularity structure
- **Model-based:** A model is hypothesized for each of the clusters and the idea is to find the best fit of that model to each other

Distinction between **Partitional** and **Hierarchical** sets of clusters

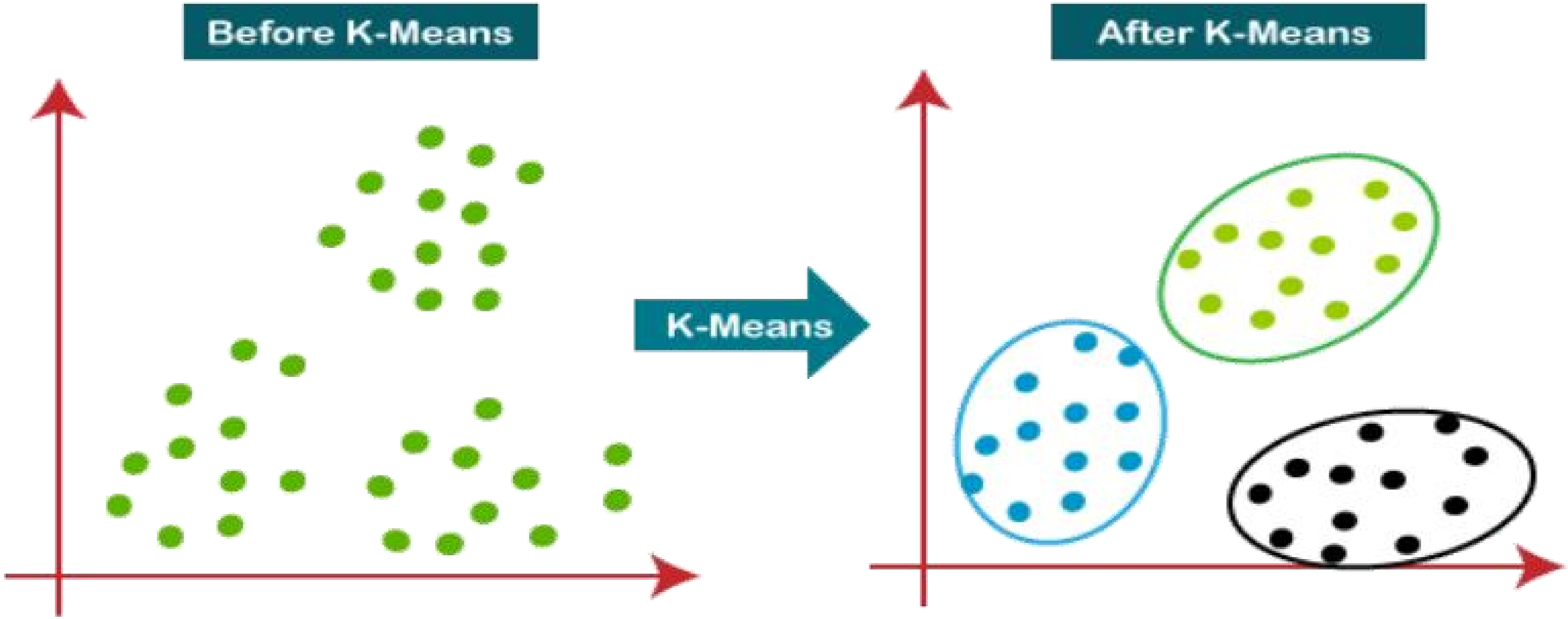
- **Partitional Clustering**
  - A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset. It construct a partition of a database ***D*** of ***n*** objects into a set of ***k*** (given) clusters
  - Heuristic methods: *k-means* and *k-medoids* algorithms
    - ➢ ***k-means*****:** Each cluster is represented by the center of the cluster
    - ➢ ***k-medoids*** or PAM (Partition around medoids): Each cluster is represented by one of the objects in the cluster
- **Hierarchical clustering**
  - A set of nested clusters organized as a hierarchical tree
  - There are mainly two types of hierarchical clustering:
    - ➢ **Agglomerative** hierarchical clustering
    - ➢ **Divisive** Hierarchical clustering

K- Means Clustering Algorithm

- It is an iterative algorithm that divides the unlabeled dataset into k different clusters in such a way that each dataset belongs only one group that has similar properties.
- It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters. ***A centroid is a data point that lies at the center of a cluster.***
- The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.
- The k-means clustering algorithm mainly performs two tasks:
  - Determines the best value for k center points or centroids by an iterative process.
  - Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.
- Hence each cluster has datapoints with some commonalities, and away from other clusters.

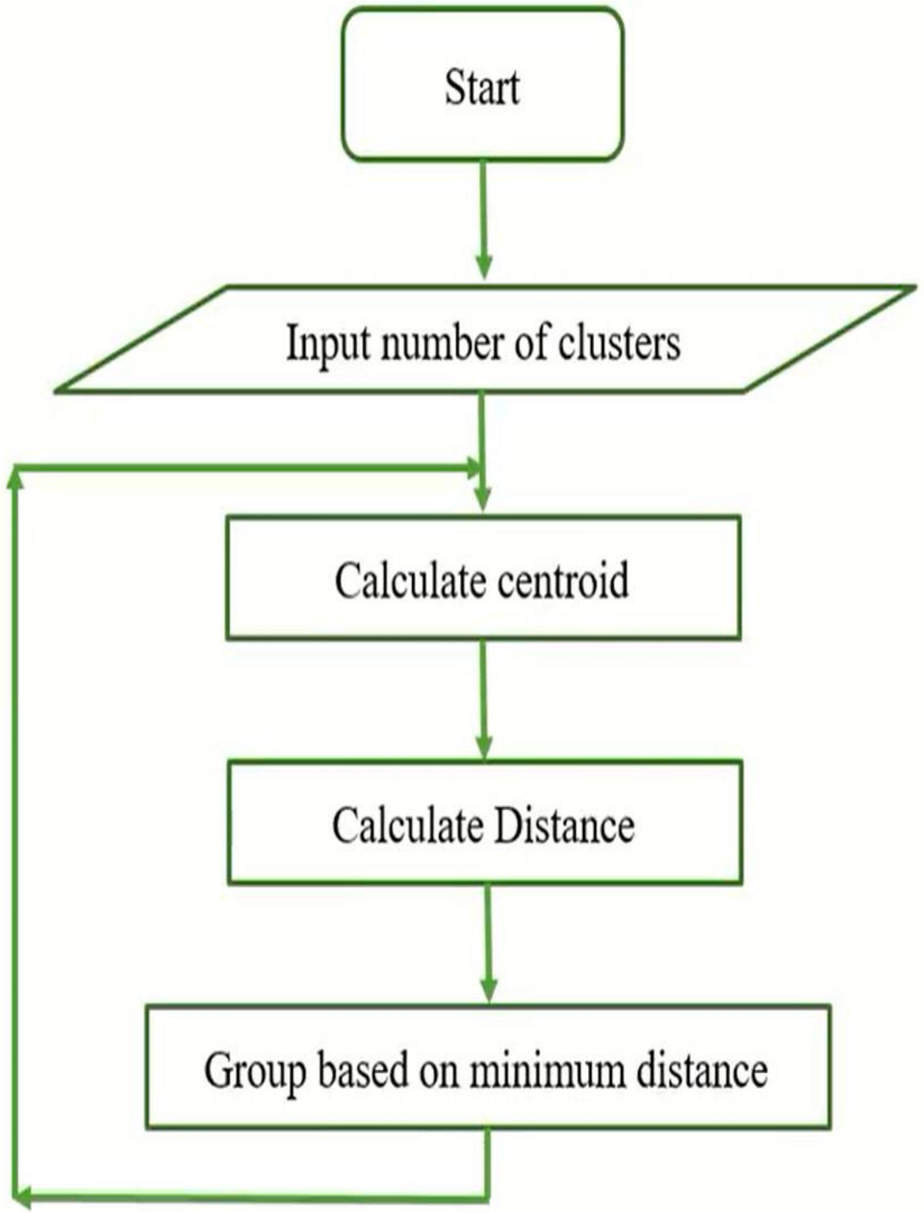
K- Means Algorithm cont...

The below diagram explains the working of the K-means Clustering Algorithm:



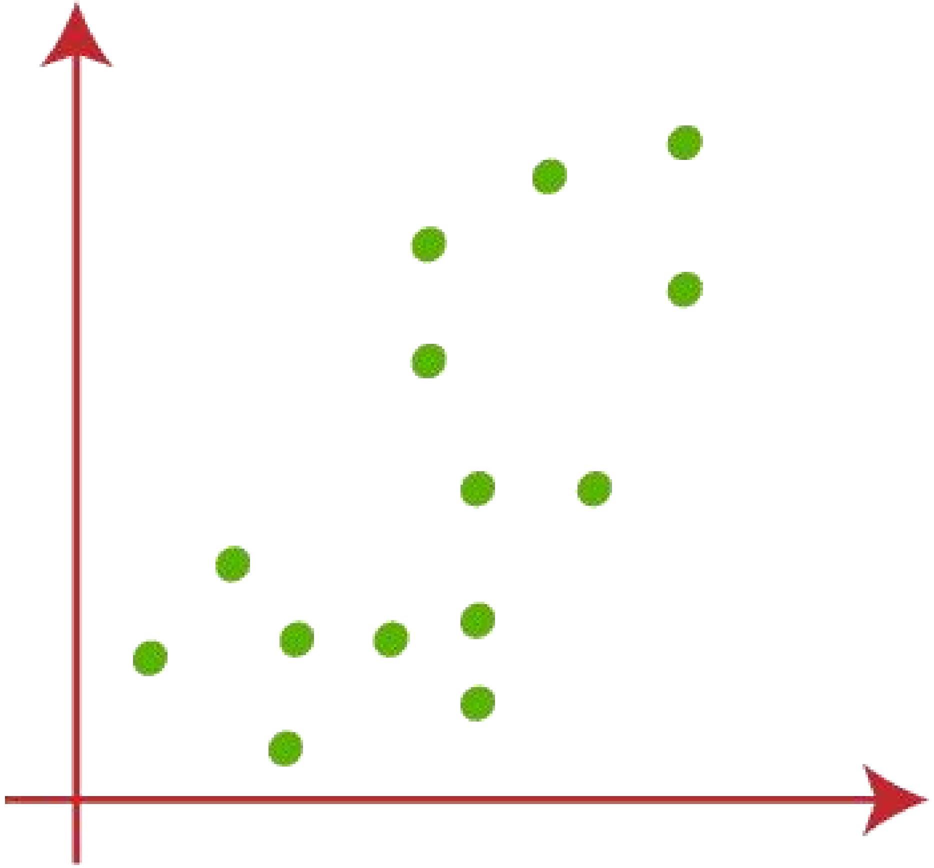
How does the K-Means Algorithm Work?

1. Step-1: Select the number K to decide the number of clusters.
2. Step-2: Select random K points or centroids. (It can be other from the input dataset).
3. Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.
4. Step-4: Calculate the variance and place a new centroid of each cluster.
5. Step-5: Repeat the third steps, i.e. reassign each datapoint to the new closest centroid of each cluster.
6. Step-6: If any reassignment occurs, then go to step-4 else go to step-7.
7. Step-7: The model is ready.
8. End



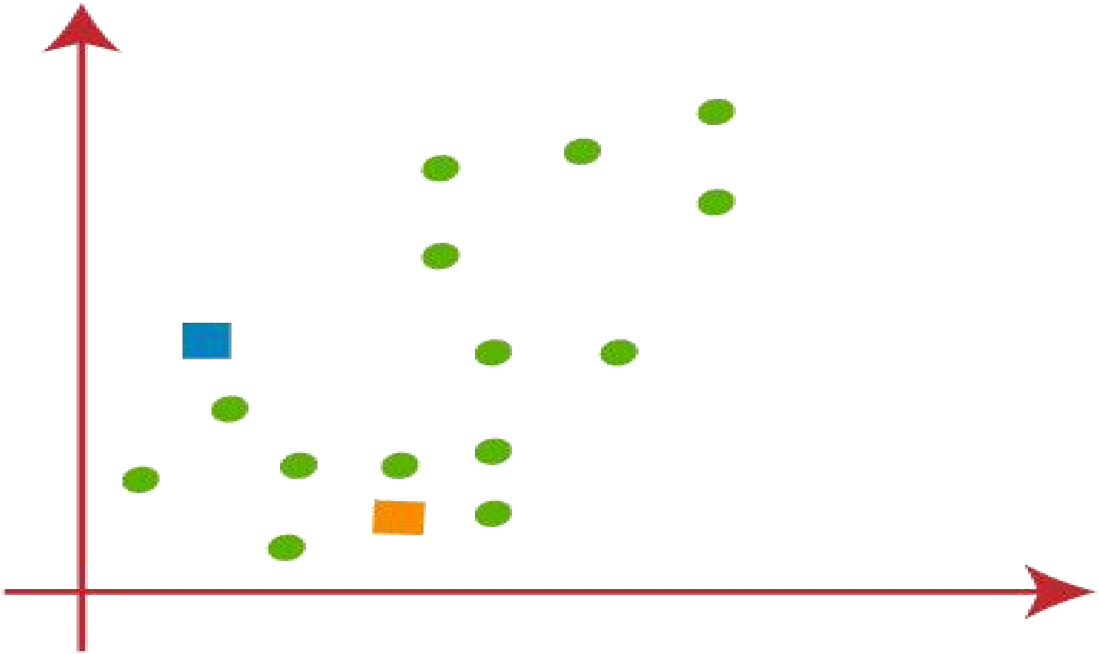
Working of K-Means Algorithm

Suppose we have two variables x and y. The x-y axis scatter plot of these two variables is given below:



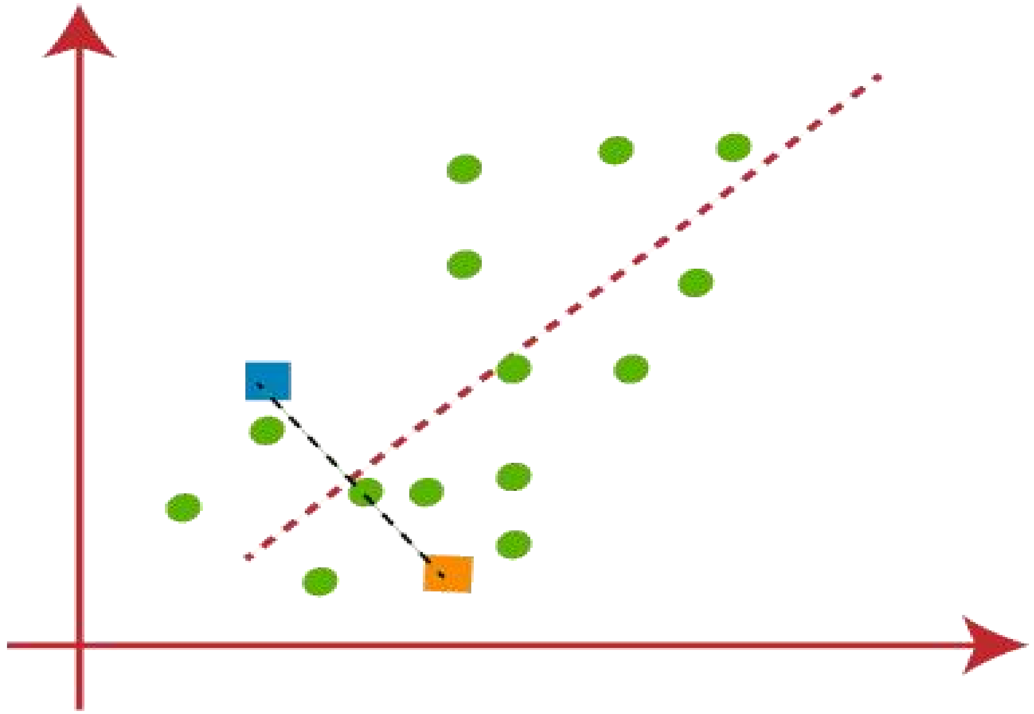
Working of K-Means Algorithm cont…

- Let's take number k of clusters, i.e., K=2, to identify the dataset and to put them into different clusters. It means here we will try to group these datasets into two different clusters.
- We need to choose some random K points or centroid to form the cluster. These points can be either the points from the dataset or any other point. So, here we are selecting the below two points as K points, which are not the part of dataset. Consider the below image:



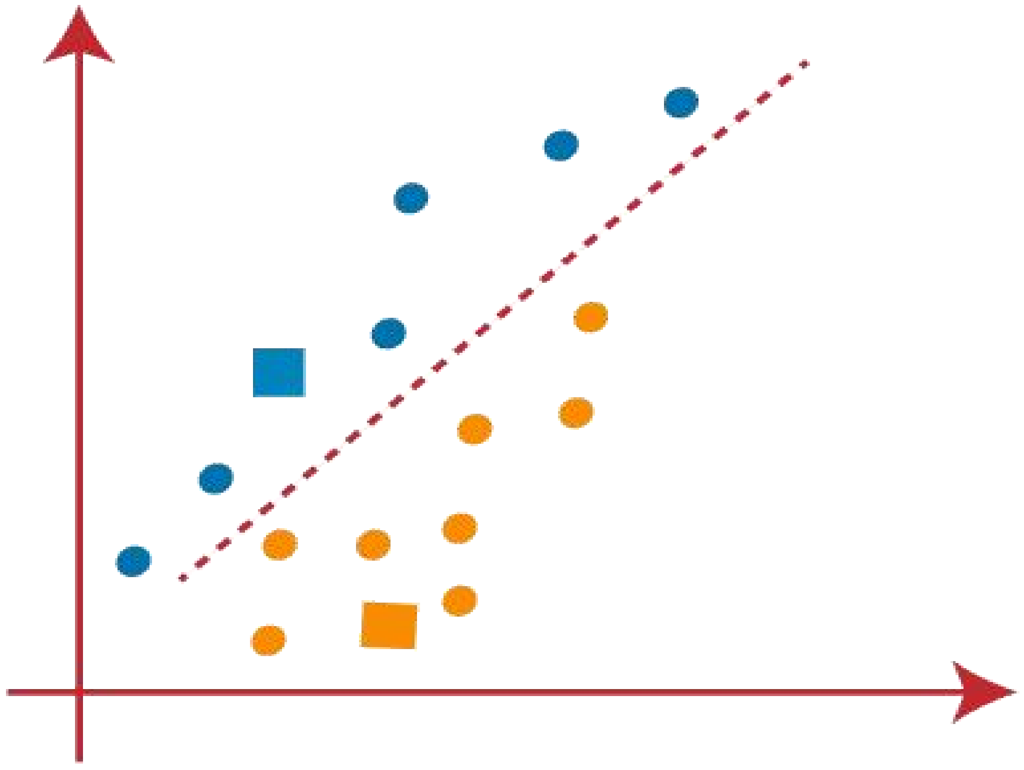
Working of K-Means Algorithm cont…

- Now we will assign each data point of the scatter plot to its closest K-point or centroid. We will compute it by calculating the distance between two points. So, we will draw a median between both the centroids. Consider the below image:



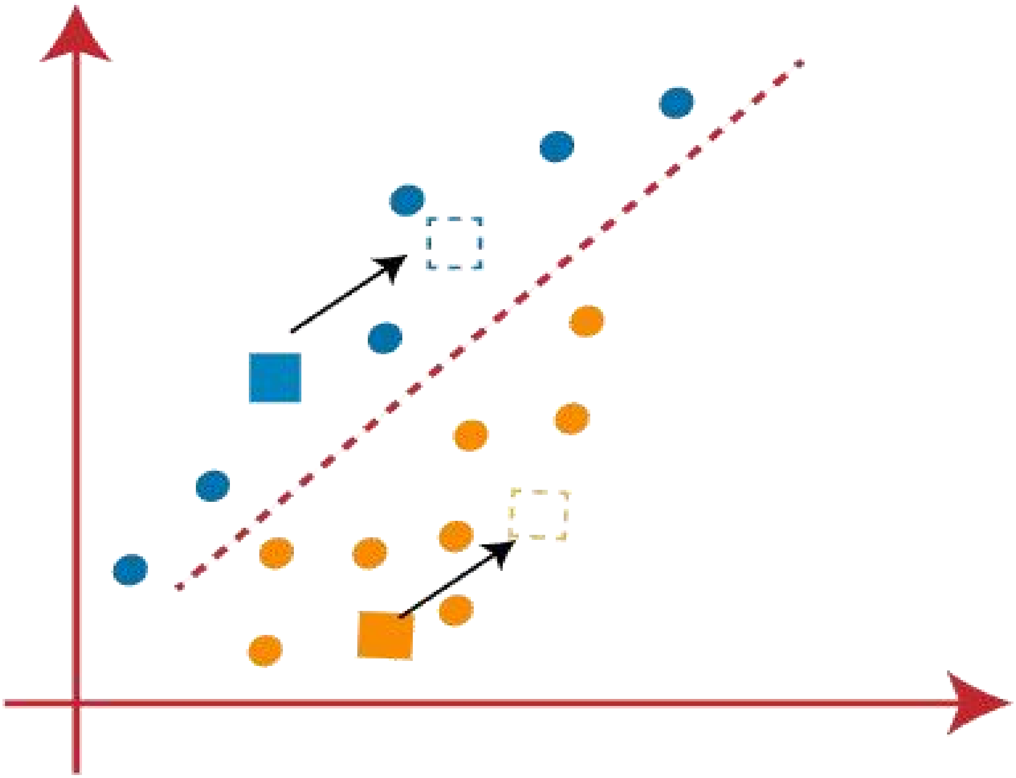
Working of K-Means Algorithm cont...

- From the image, it is clear that points left side of the line is near to the K1 or blue centroid, and points to the right of the line are close to the yellow centroid i.e. K2. Let's color them as blue and yellow for clear visualization.



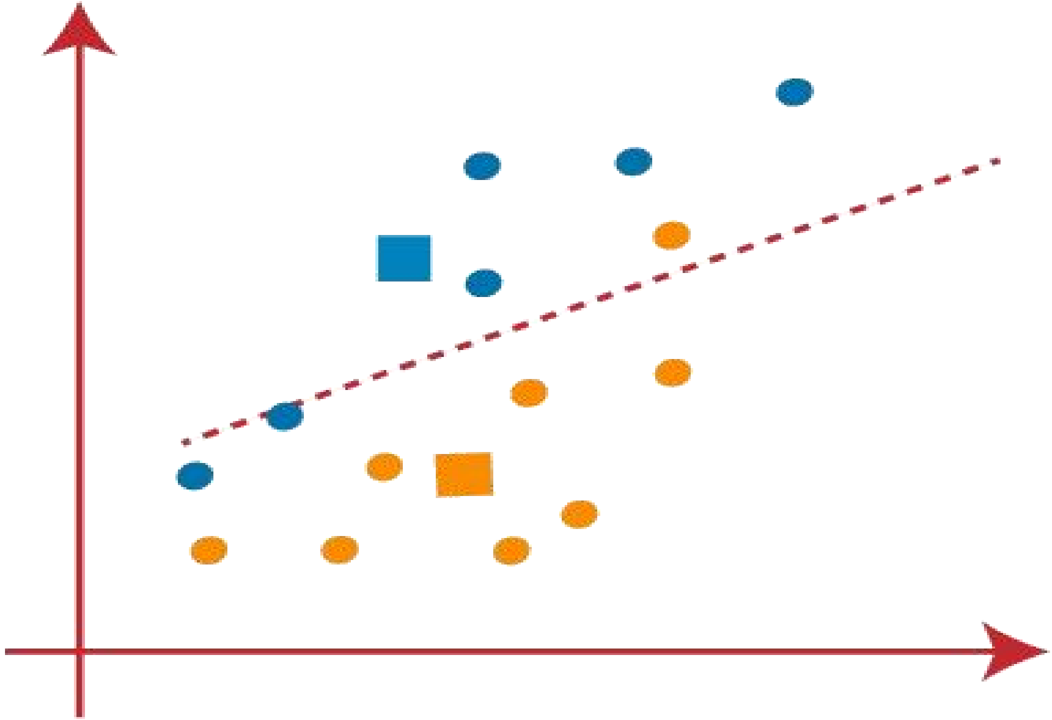
Working of K-Means Algorithm cont…

- As we need to find the closest cluster, so we will repeat the process by choosing a new centroid. To choose the new centroids, we will compute the center of gravity of these centroids, and will find new centroids as below:



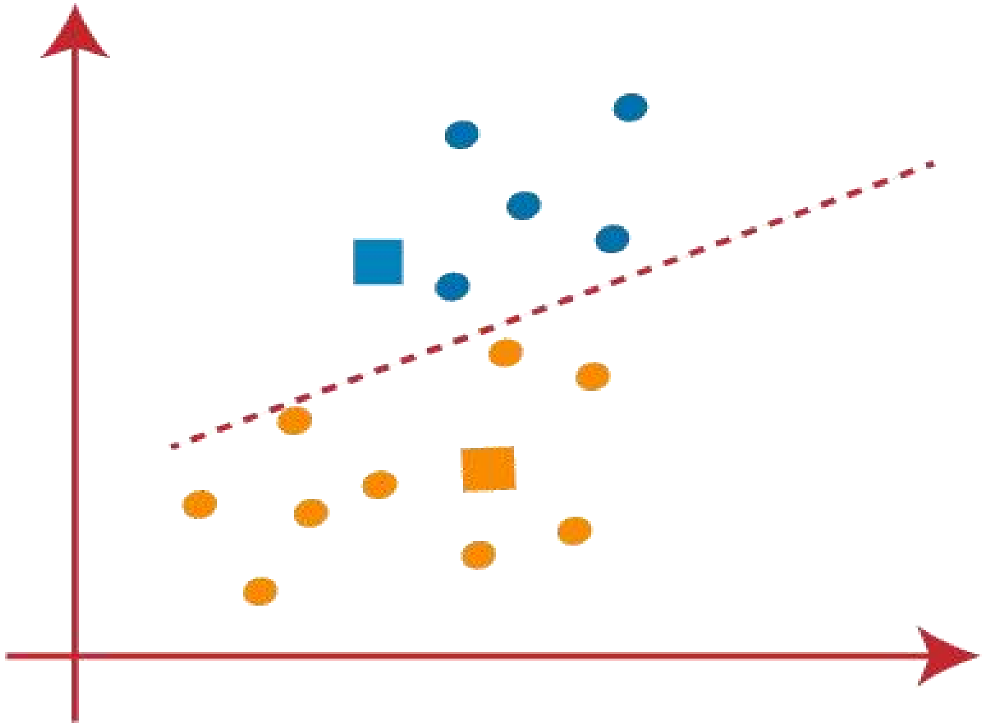
Working of K-Means Algorithm cont…

- Next, we will reassign each datapoint to the new centroid. For this, we will repeat the same process of finding a median line. The median will be like below image:



From the above image, we can see, one yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new centroids.

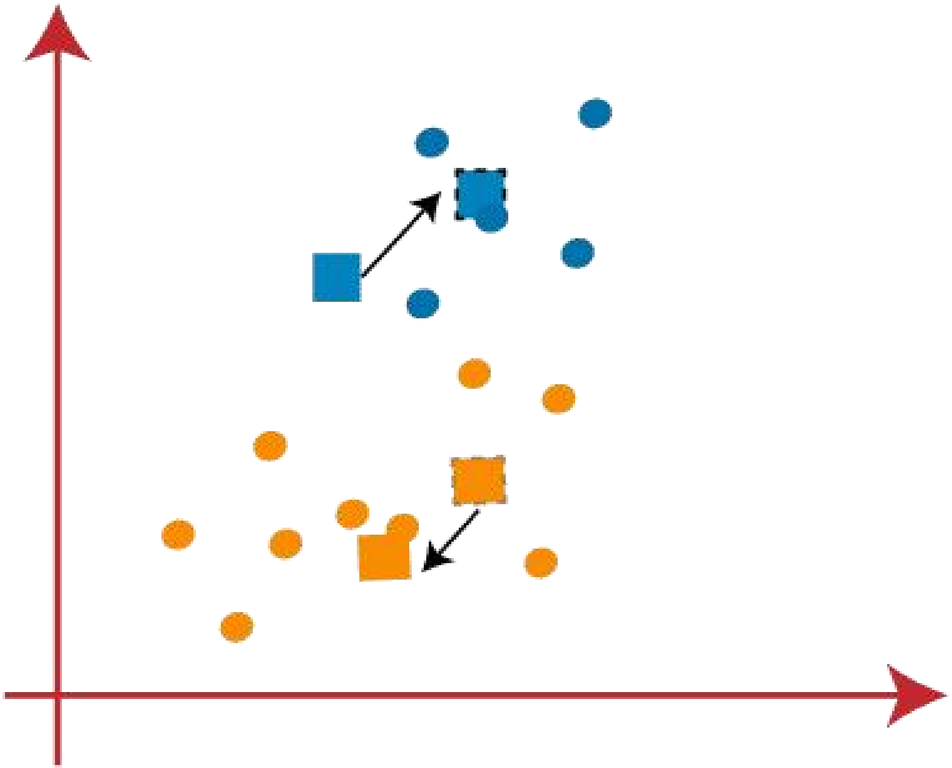
Working of K-Means Algorithm cont…



- As reassignment has taken place, so we will again go to the step-4, which is finding new centroids or K-points.

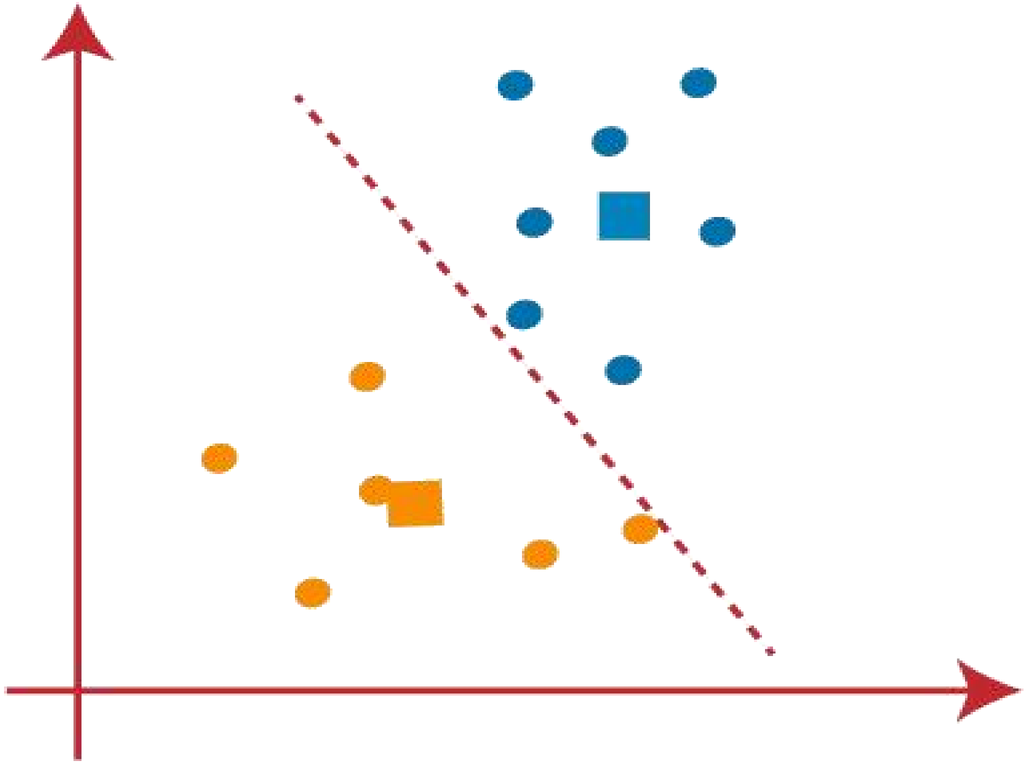
Working of K-Means Algorithm cont…

- We will repeat the process by finding the center of gravity of centroids, so the new centroids will be as shown in the below image:



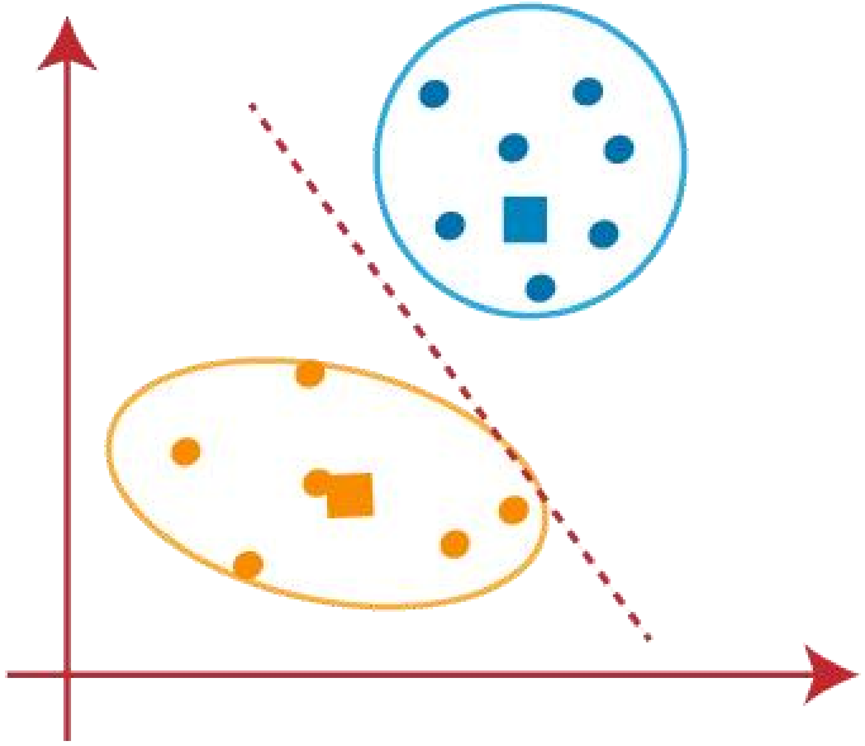
Working of K-Means Algorithm cont…

- As we got the new centroids so again will draw the median line and reassign the data points. So, the image will be:



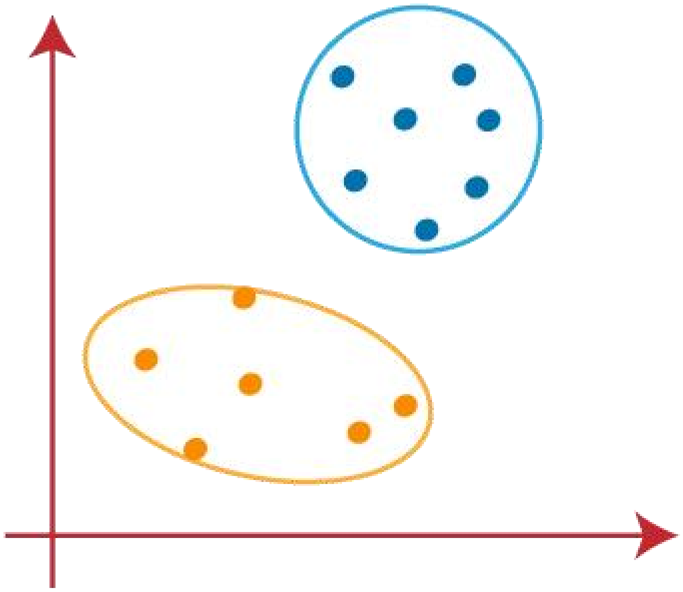
Working of K-Means Algorithm cont…

- We can see in the previous image; there are no dissimilar data points on either side of the line, which means our model is formed. Consider the below image:



Working of K-Means Algorithm cont…

- As our model is ready, so we can now remove the assumed centroids, and the two final clusters will be as shown in the below image:



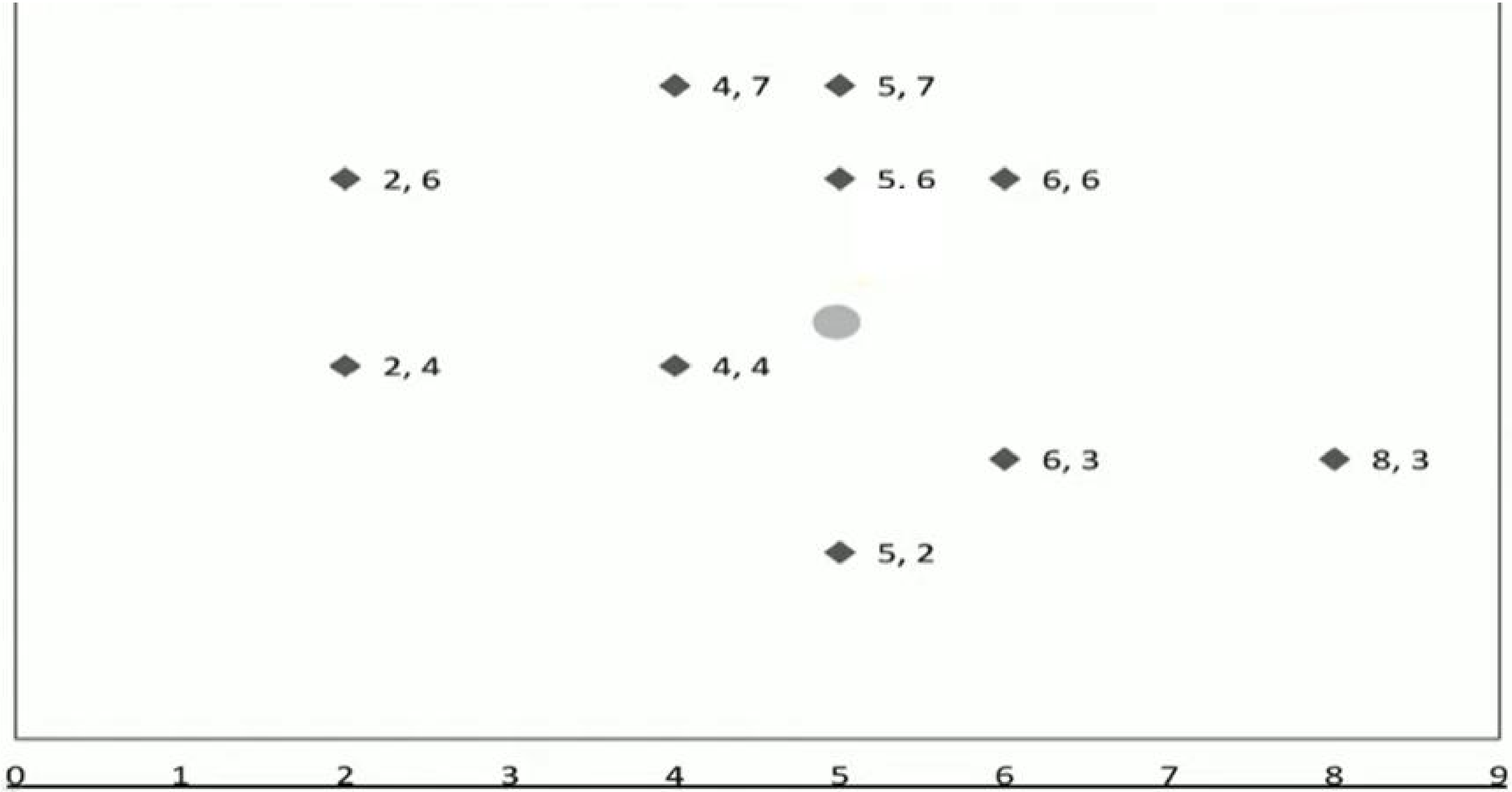
K-Means Clustering Example

10 data points with 2 attributes are given. Question is to divide the data points into K number (let k=3) of clusters.

X	Y
2	4
2	6
5	6
4	7
8	3
6	6
5	2
5	7
6	3
4	4

K-Means Clustering Example cont..

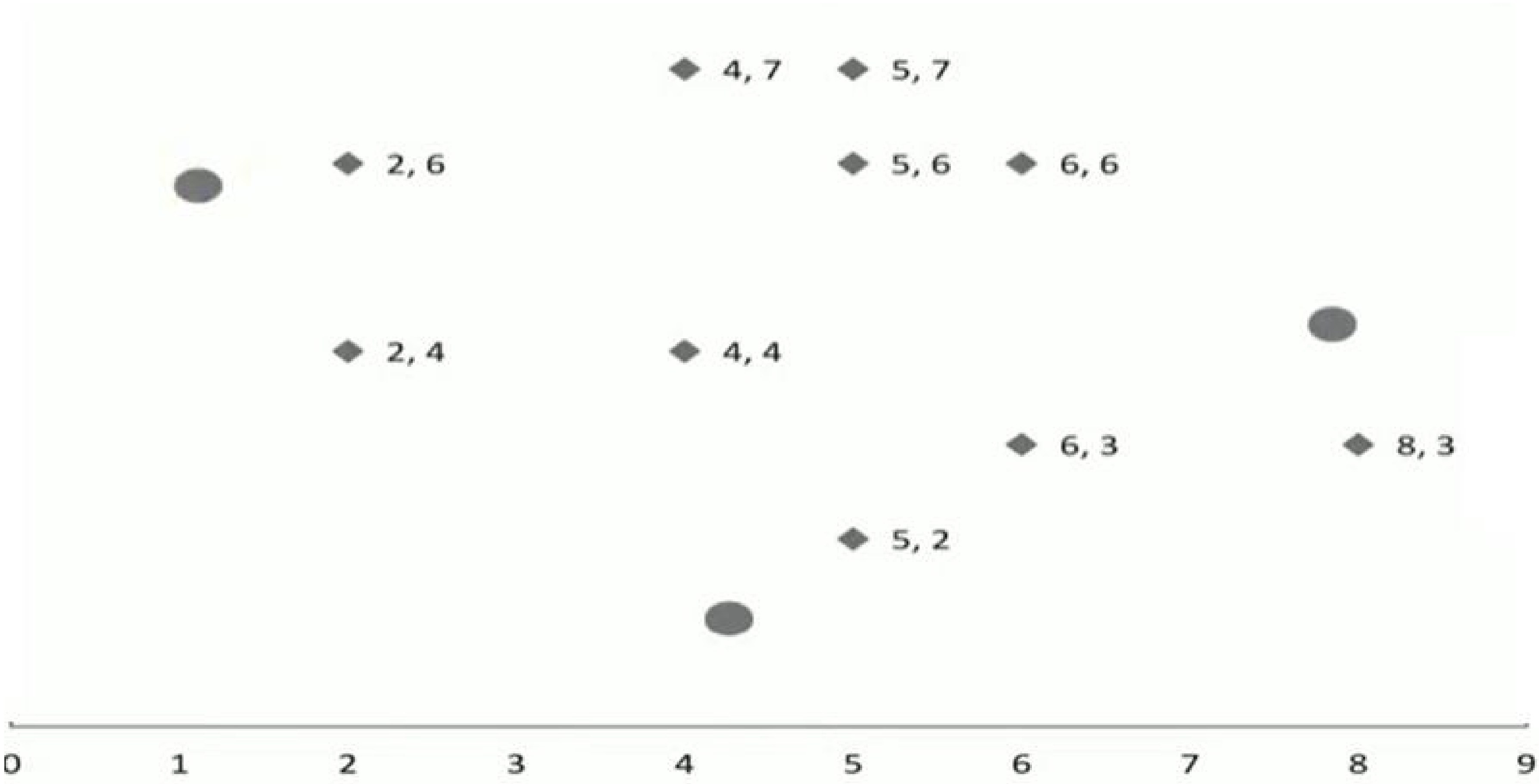
First plot the data point and find the centoid of the data points. Assume initially all the data points belongs to single/same cluser.



K-Means Clustering Example cont..

Let K=3 or problem is to divide the dtata points into 3clusters.

For this, select 3 randome points/seed points/centroids of 3 clusters. Let (1, 5), (4. 1) & (8, 4)



K-Means Clustering Example cont..

(Calculate the distances of all data points from 3 centroids)

Iteration - 1

C1 - Seed Point1 – (1, 5)

C2 - Seed Point2 – (4, 1)

C3 - Seed Point3 – (8, 4)

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

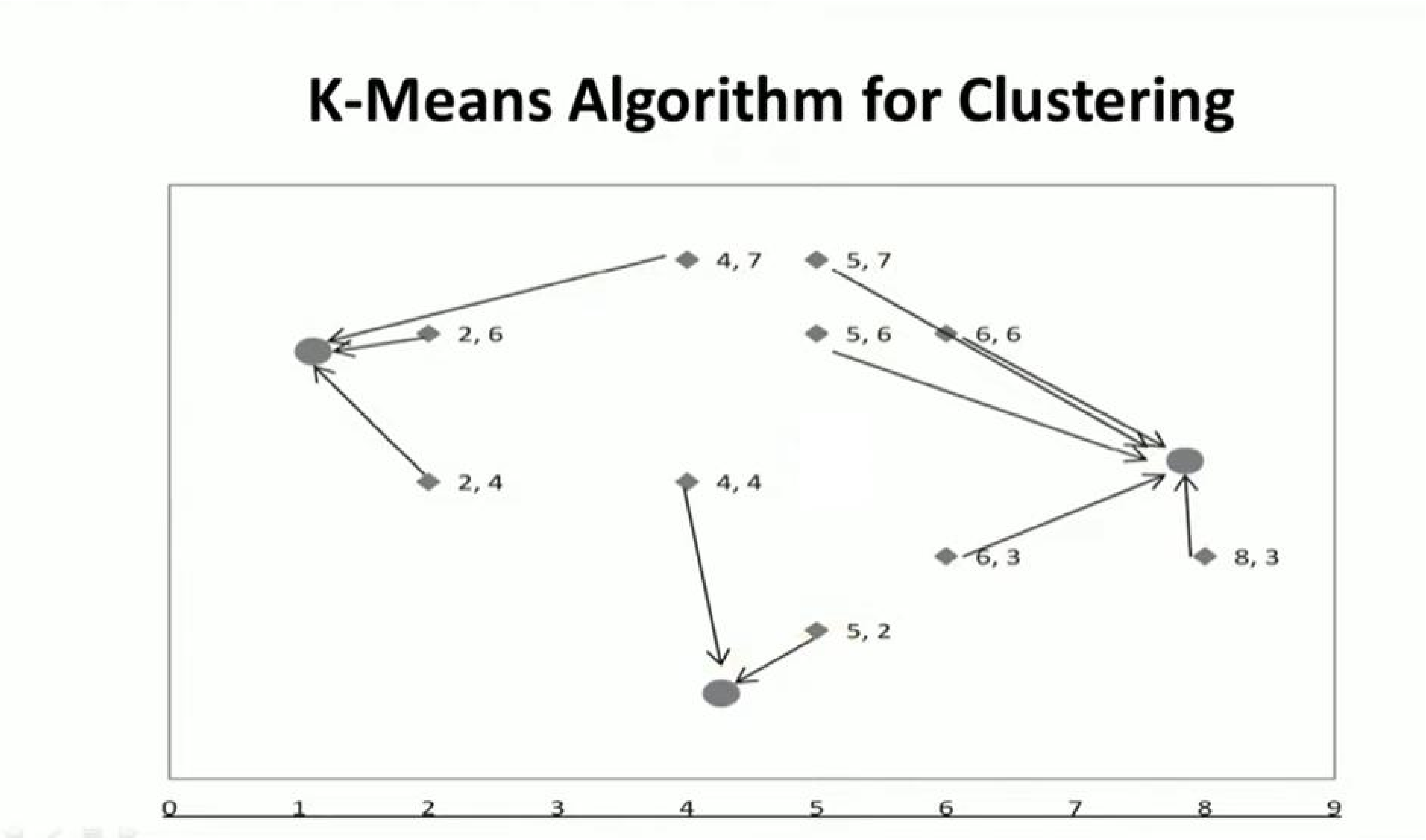
- Compare the 3 distances of individual data points from the corresponding centroids and find the minimumm distance.
- The data point will belong to the cluster to which it is closer

Sr. No.	X	Y	Distance to (1, 5)	Distance to (4, 1)	Distance to (8, 4)	Cluster Number
1	2	4	1.41	3.61	6.00	C1
2	2	6	1.41	5.39	6.32	C1
3	5	6	4.12	5.10	3.61	C3
4	4	7	3.61	6.00	5.00	C1
5	8	3	7.28	4.47	1.00	C3
6	6	6	5.10	5.39	2.83	C3
7	5	2	5.00	1.41	3.61	C2
8	5	7	4.47	6.08	4.24	C3
9	6	3	5.39	2.83	2.24	C3
10	4	4	3.16	3.00	4.00	C2

Clusters

G-1 = {1, 2, 4} G-2 = {7, 10} G-3 = {3, 5, 6, 8, 9}

K-Means Clustering Example cont..



K-Means Clustering Example cont..

Iteration-2

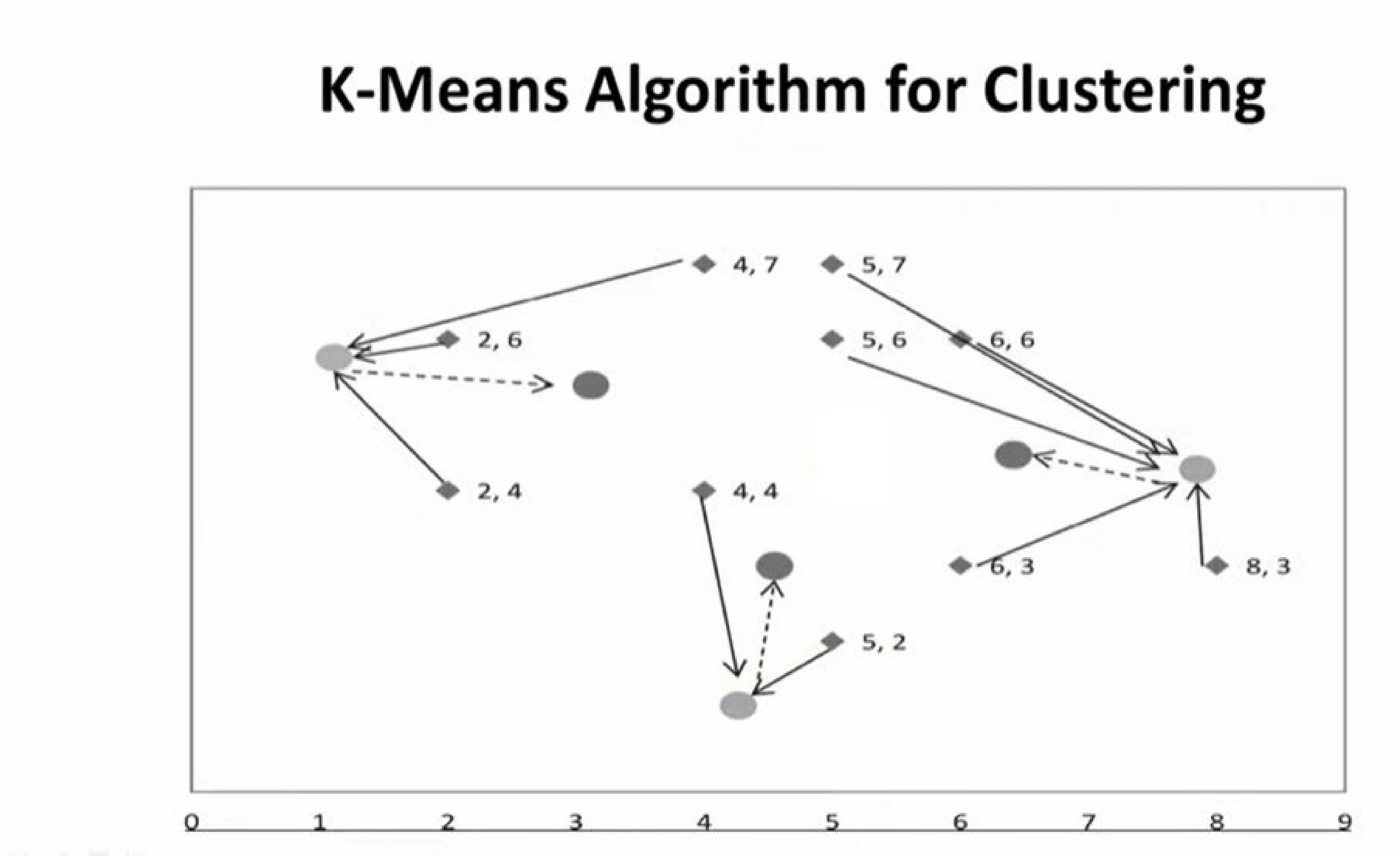
Centroid of Cluster 1: C1 =(2.66,5.66)
Centroid of Cluster 2: C2 = (4.5, 3)
Centroid of Cluster 3: C3 = (6, 5)

Sr. No.	X	Y	Distance to (2.66, 5.66)	Distance to (4.5, 3)	Distance to (6, 5)	Cluster Number
1	2	4	1.79	2.69	4.12	C1
2	2	6	0.74	3.91	4.12	C1
3	5	6	2.36	3.04	1.41	C3
4	4	7	1.90	4.03	2.83	C1
5	8	3	5.97	3.5	2.83	C3
6	6	6	3.36	3.35	1	C3
7	5	2	4.34	1.12	3.16	C2
8	5	7	2.70	4.03	2.24	C3
9	6	3	4.27	1.5	2	C2
10	4	4	2.13	1.12	2.24	C2

New Cluster

G-1 = {1, 2, 4} **G-2 = {7, 9, 10}** **G-3 = {3, 5, 6, 8}**

K-Means Clustering Example cont..



K-Means Clustering Example cont..

<u>Iteration-3</u>

Centroid of Cluster 1: C1 = (2.66, 5.66)
Centroid of Cluster 2: C2 = (5,3)
Centroid of Cluster 3: C3 = (6, 5.5)

Sr. No.	X	Y	Distance to (2.66, 5.66)	(5, 3)	(6, 5.5)	Cluster Number
1	2	4	1.79	3.16	4.27	C1
2	2	6	0.74	4.24	4.03	C1
3	5	6	2.36	3.00	1.12	C3
4	4	7	1.90	4.12	2.50	C1
5	8	3	5.97	3.00	3.20	C2
6	6	6	3.36	3.16	0.50	C3
7	5	2	4.34	1.00	3.64	C2
8	5	7	2.70	4.00	1.80	C3
9	6	3	4.27	1.00	2.50	C2
10	4	4	2.13	1.41	2.50	C2

New Cluster

G-1 = {1, 2, 4} **G-2 = {5, 7, 9, 10}** **G-3 = {3, 6, 8}**

K-Means Clustering Example cont..

<u>**Iteration-4**</u>

Centroid of Cluster 1: C1 = (2.66, 5.66)
Centroid of Cluster 2: C2 = (5.75,3)
Centroid of Cluster 3: C3 = (5.33, 6.33)

Sr. No.	X	Y	Distance to (2.66, 5.66)	Distance to (5.75, 3)	Distance to (5.33, 6.33)	Cluster Number
1	2	4	1.79	3.88	4.06	C1
2	2	6	0.74	4.80	3.35	C1
3	5	6	2.36	3.09	0.47	C3
4	4	7	1.90	4.37	1.49	C3
5	8	3	5.97	2.25	4.27	C2
6	6	6	3.36	3.01	0.75	C3
7	5	2	4.34	1.25	4.34	C2
8	5	7	2.70	4.07	0.75	C3
9	6	3	4.27	0.25	3.40	C2
10	4	4	2.13	2.02	2.68	C2

New Cluster

G-1 = {1, 2}　**G-2 = {5, 7, 9, 10}**　**G-3 = {3, 4, 6, 8}**

K-Means Clustering Example cont..

Iteration-5

Centroid of Cluster 1: C1 = (2, 5)
Centroid of Cluster 2: C2 = (5.75,3)
Centroid of Cluster 3: C3 = (5, 5.5)

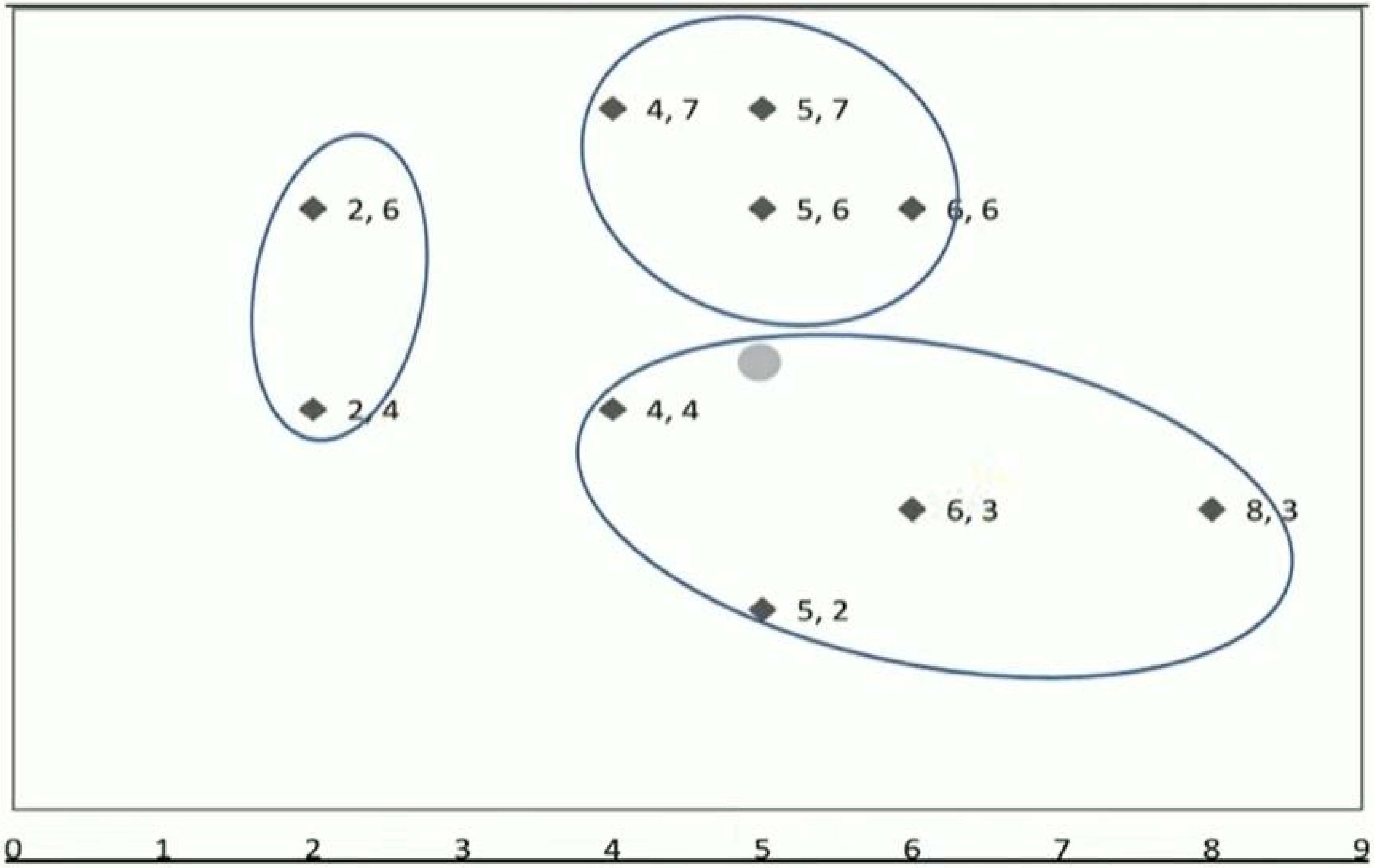
Sr. No.	X	Y	Distance to (2, 5)	Distance to (5.75, 3)	Distance to (5, 6.5)	Cluster Number
1	2	4	1.00	3.88	3.91	C1
2	2	6	1.00	4.80	3.04	C1
3	5	6	3.16	3.09	0.50	C3
4	4	7	2.83	4.37	1.12	C3
5	8	3	6.32	2.25	4.61	C2
6	6	6	4.12	3.01	1.12	C3
7	5	2	4.24	1.25	4.50	C2
8	5	7	3.61	4.07	0.50	C3
9	6	3	4.47	0.25	3.64	C2
10	4	4	2.24	2.02	2.69	C2

No movement of data Points
Hence these are the final
positions

New Cluster

G-1 = {1, 2} **G-2 = {5, 7, 9, 10}** **G-3 = {3, 4, 6, 8}**

K-Means Clustering Example cont..



Advantages

- Simple, understandable
- Items automatically assigned to clusters

Disadvantages

- Must pick number of clusters before hand
- Often terminates at a local optimum.
- All items forced into a cluster
- Too sensitive to outliers

K-Medoids (PAM) Clustering Method

- Find representative objects, called <u>medoids</u>, in clusters
  - To achieve this goal, only the definition of distance from any two objects is needed.
- PAM (Partitioning Around Medoids, 1987)
  - Starts from an initial set of medoids and iteratively replaces one of the medoids by one of the non-medoids if it improves the total distance of the resulting clustering.
  - PAM works effectively for small data sets, but does not scale well for large data sets.
- CLARA (Kaufmann & Rousseeuw, 1990)
- CLARANS (Ng & Han, 1994): Randomized sampling.
- Focusing + spatial data structure (Ester et al., 1995)

k-medoids algorithm

Use real object to represent the cluster

- Select k representative objects arbitrarily
- repeat
  - Assign each remaining object to the cluster of the nearest medoid
  - Randomly select a non-medoid object
  - Compute the total cost, S, of swapping o_j with o_{random}
  - If S < 0 then swap oj with orandom
- until there is no change

K-Medoids Example

- 1, 2, 6, 7, 8, 10, 15, 17, 20 – break into 3 clusters
  - ➢Cluster = 6 – 1, 2
  - ➢Cluster = 7
  - ➢Cluster = 8 – 10, 15, 17, 20
- Random non-medoid – 15 replace 7 (total cost=-13)
  - Cluster = 6 – 1 (cost 0), 2 (cost 0), 7(1-0=1)
  - Cluster = 8 – 10 (cost 0)
  - New Cluster = 15 – 17 (cost 2-9=-7), 20 (cost 5-12=-7)
- Replace medoid 7 with new medoid (15) and reassign
  - ➢Cluster = 6 – 1, 2, 7
  - ➢Cluster = 8 – 10
  - ➢Cluster = 15 – 17, 20

K-Medoids example (continued)

- Random non-medoid – 1 replaces 6 (total cost=2)
  - Cluster = 8 – 7 (cost 6-1=5)10 (cost 0)
  - Cluster = 15 – 17 (cost 0), 20 (cost 0)
  - New Cluster = 1 – 2 (cost 1-4=-3)
- 2 replaces 6 (total cost=1)
- Don't replace medoid 6
  - Cluster = 6 – 1, 2, 7
  - Cluster = 8 – 10
  - Cluster = 15 – 17, 20
- Random non-medoid – 7 replaces 6 (total cost=2)
  - Cluster = 8 – 10 (cost 0)
  - Cluster = 15 – 17(cost 0), 20(cost 0)
  - New Cluster = 7 – 6 (cost 1-0=1), 2 (cost 5-4=1)

K-Medoids example (continued)

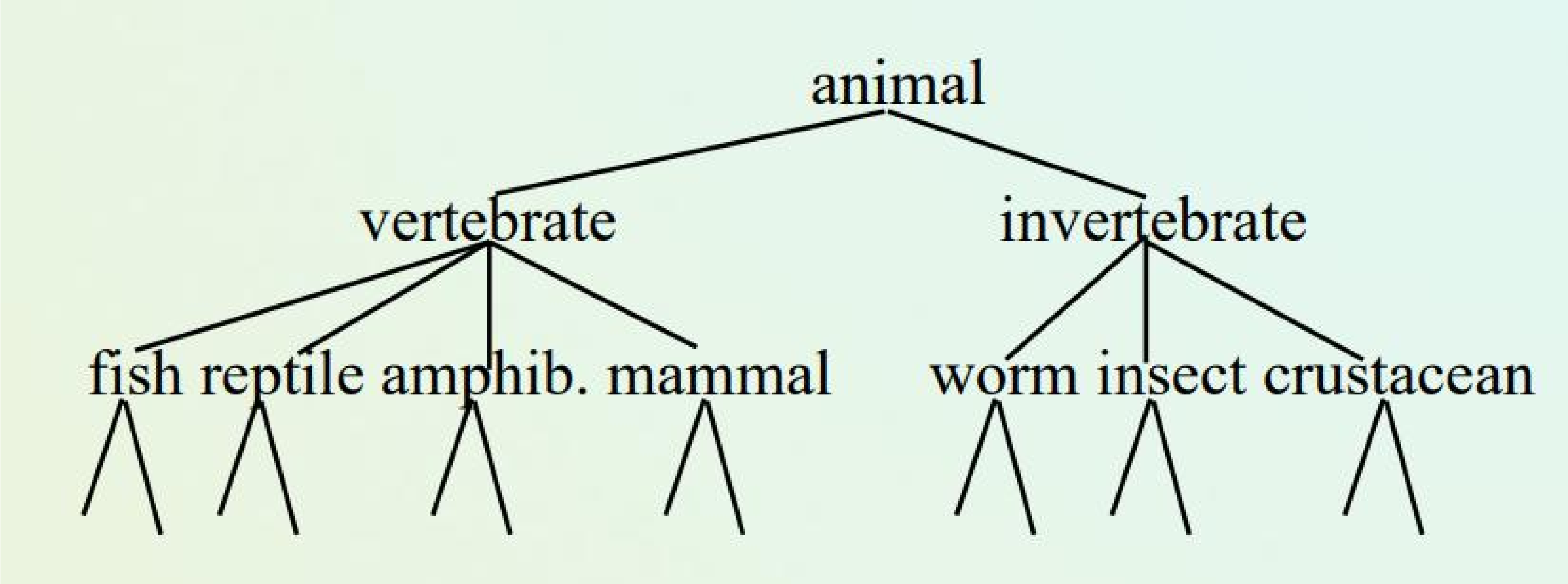
- Don't Replace medoid 6
  - Cluster = 6 – 1, 2, 7
  - Cluster = 8 – 10
  - Cluster = 15 – 17, 20
- Random non-medoid – 10 replaces 8 (total cost=2) don't replace
  - Cluster = 6 – 1(cost 0), 2(cost 0), 7(cost 0)
  - Cluster = 15 – 17 (cost 0), 20(cost 0)
  - New Cluster = 10 – 8 (cost 2-0=2)
- Random non-medoid – 17 replaces 15 (total cost=0) don't replace
  - Cluster = 6-1 (cost 0), 2 (cost 0), 7(cost 0)
  - Cluster = 8 – 10 (cost 0)
  - New Cluster = 17 – 15 (cost 2-0=2), 20(cost 3-5=-2)

K-Medoids example (continued)

- Random non-medoid – 20 replaces 15 (total cost=3) don't replace
  - Cluster = 6 – 1(cost 0), 2(cost 0), 7(cost 0)
  - Cluster = 8 – 10 (cost 0)
  - New Cluster = 20 – 15 (cost 5-0=2), 17(cost 3-2=1)
- Other possible changes all have high costs
  - 1 replaces 15, 2 replaces 15, 1 replaces 8, …
- No changes, F**inal clusters**
  - Cluster = 6 – 1, 2, 7
  - Cluster = 8 – 10
  - Cluster = 15 – 17, 20

Hierarchical Clustering

- Build a tree-based hierarchical taxonomy (dendrogram) from a set of unlabeled examples.
- Recursive application of a standard clustering algorithm can produce a hierarchical clustering.

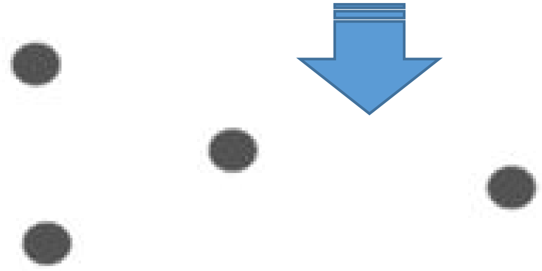


Hierarchical Clustering

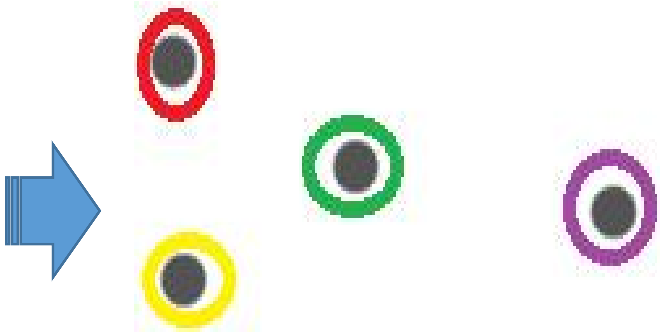
1. **Bottom up (aglomerative)**
   - Start with single-instance clusters
   - At each step, join the two closest clusters
   - Design decision: distance between clusters
     - e.g.two closest instances in clusters vs. distance between means
2. **Top down (divisive approach / deglomerative)**
   - Start with one universal cluster
   - Find two clusters
   - Proceed recursively on each subset
   - Can be very fast
3. **Both methods produce a dendrogram**

Hierarchical Clustering

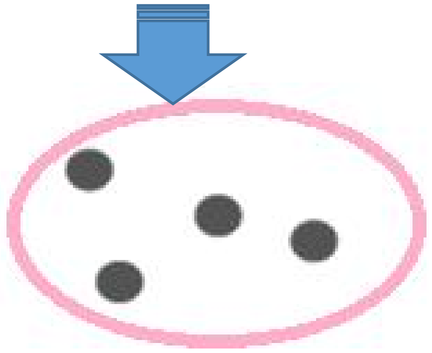
Let’s say we have the below points and we want to cluster them into groups.



We can assign each of these points to a separate cluster.



Now, based on the similarity of these clusters, the most similar clusters combined together and this process is repeated until only a single cluster is left.



We are essentially building a hierarchy of clusters. That’s why this algorithm is called hierarchical clustering.

Hierarchical Clustering Example

Suppose a faculty wants to divide the students into different groups. The faculty has the marks scored by each student in an assignment and based on these marks, he/she wants to segment them into groups. There's no fixed target as to how many groups to have. Since the faculty does not know what type of students should be assigned to which group, it cannot be solved as a supervised learning problem. So, hierarchical clustering is applied to segment the students into different groups. Let's take a sample of 5 students.

Creating a Proximity Matrix

Roll No	Mark
1	10
2	7
3	28
4	20
5	35

First, a proximity matrix to be created which tell the distance between each of these points (marks). Since the distance is calculated of each point from each of the other points, a square matrix of shape n X n (where n is the number of observations) is obtained. Let's make the 5 x 5 proximity matrix for the example.

Hierarchical Clustering Example cont…

Proximity Matrix

Roll No	1	2	3	4	5
1	0	3	18	10	25
2	3	0	21	13	28
3	18	21	0	8	7
4	10	13	8	0	15
5	25	28	7	15	0

The diagonal elements of this matrix is always 0 as the distance of a point with itself is always 0. The Euclidean distance formula is used to calculate the rest of the distances. So, to calculate the distance between
Point 1 and 2: $\sqrt{(10-7)^2} = \sqrt{9} = 3$
Point 1 and 3: $\sqrt{(10-28)^2} = \sqrt{324} = 18$ and so on…
Similarly, all the distances are calculated and the proximity matrix is filled.

Steps to Perform Hierarchical Clustering

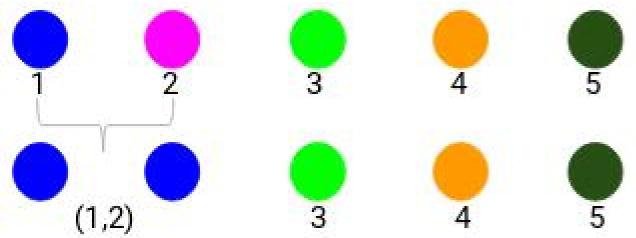
Step 1: First, all the points to an individual cluster is assigned. Different colors here represent different clusters. Hence, 5 different clusters for the 5 points in the data.

Step 2: Next, look at the smallest distance in the proximity matrix and merge the points with the smallest distance. Then the proximity matrix is updated.

Roll No	1	2	3	4	5
1	0	3	18	10	25
2	3	0	21	13	28
3	18	21	0	8	7
4	10	13	8	0	15
5	25	28	7	15	0

Here, the smallest distance is 3 and hence point 1 and 2 is merged.

Steps to Perform Hierarchical Clustering cont…



Let's look at the updated clusters and accordingly update the proximity matrix. Here, we have taken the maximum of the two marks (7, 10) to replace the marks for this cluster. Instead of the maximum, the minimum value or the average values can also be considered.

Roll No	Mark
(1, 2)	10
3	28
4	20
5	35

Now, the proximity matrix for these clusters is calculated again.

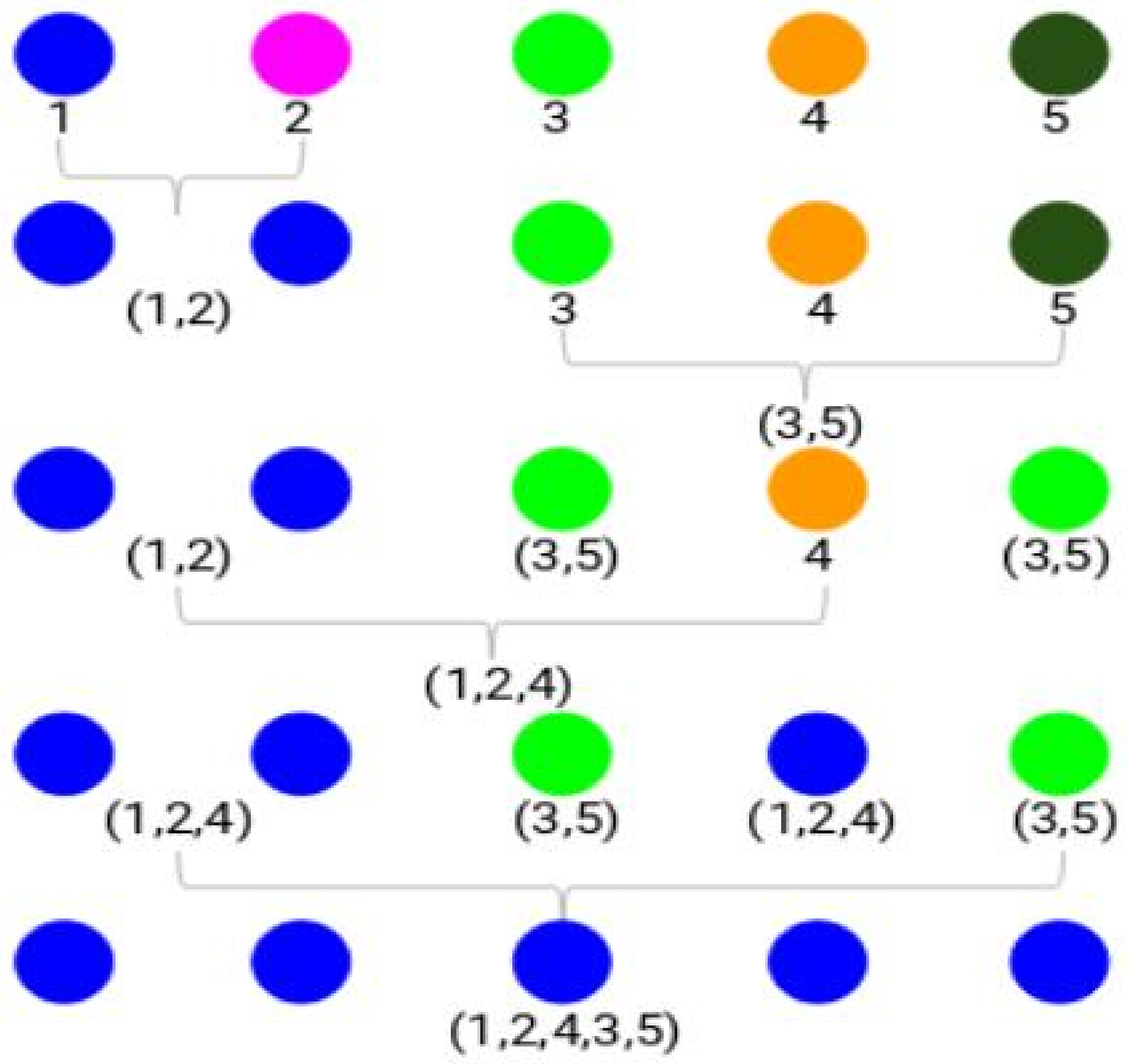
Steps to Perform Hierarchical Clustering cont…

Revised Proximity Matrix

Roll No	1, 2	3	4	5
1, 2	0	18	10	25
3	18	0	8	7
4	10	8	0	15
5	25	7	15	0

Step 3: Step 2 is repeated until only a single cluster is left. So, look at the minimum distance in the proximity matrix and then merge the closest pair of clusters. We will get the merged clusters after repeating these steps:

Steps to Perform Hierarchical Clustering cont…



We started with 5 clusters and finally have a single cluster.

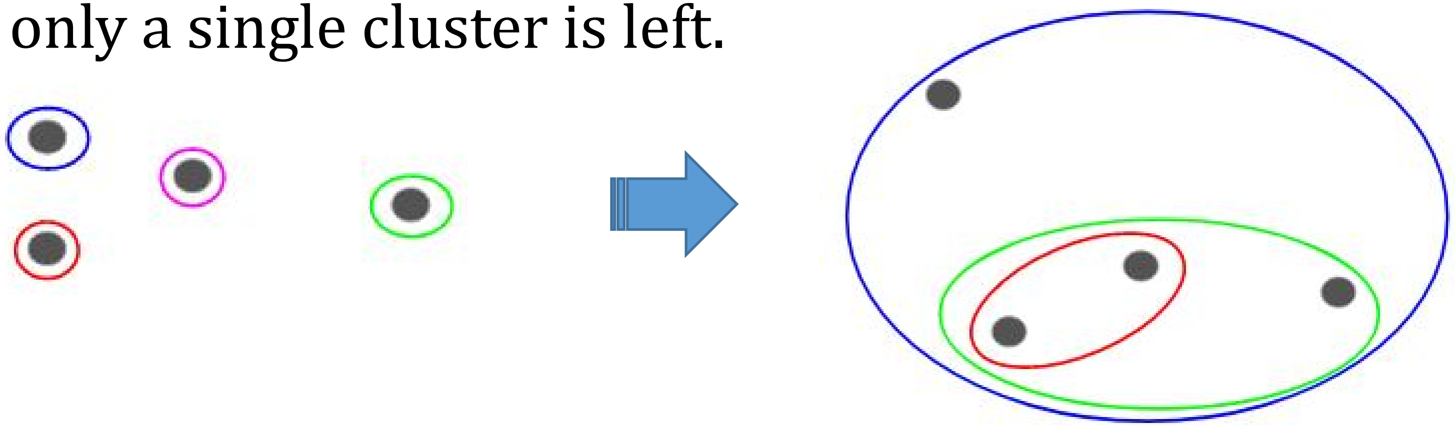
Types of Hierarchical Clustering

There are mainly two types of hierarchical clustering:

- Agglomerative hierarchical clustering
- Divisive Hierarchical clustering

Agglomerative Hierarchical Clustering

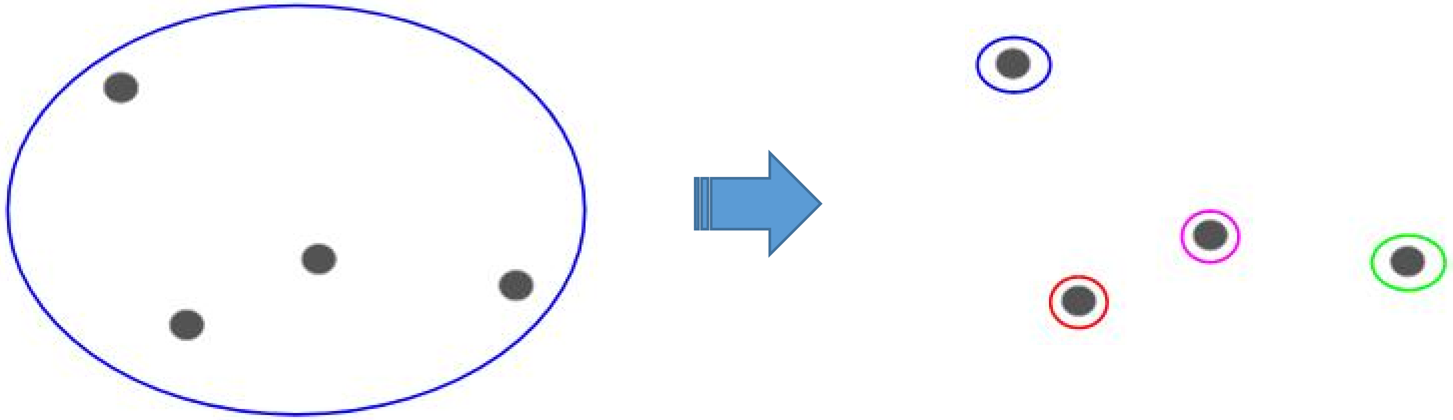
- Each point is assigned to an individual cluster in this technique. Suppose there are 4 data points, so each of these points would be assigned to a cluster and hence there would be 4 clusters in the beginning.
- Then, at each iteration, closest pair of clusters are merged and this step is repeated until only a single cluster is left.



Types of Hierarchical Clustering cont…

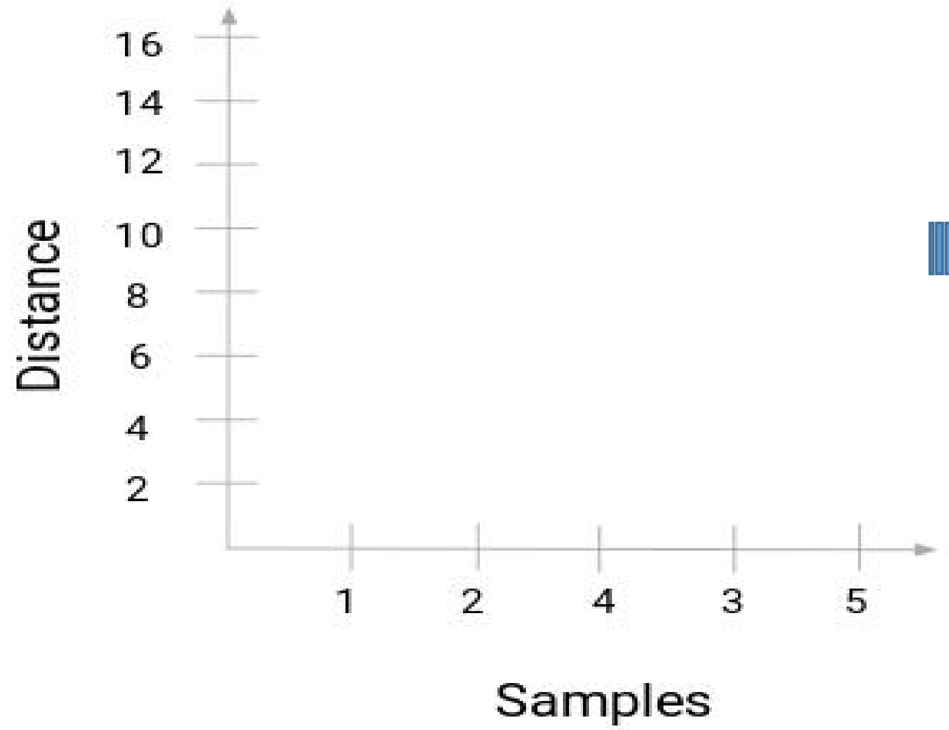
Divisive Hierarchical clustering

- Divisive hierarchical clustering works in the opposite way. Instead of starting with n clusters (in case of n observations), we start with a single cluster and assign all the points to that cluster. So, it doesn't matter if we have 10 or 1000 data points. All these points will belong to the same cluster at the beginning.
- Now, at each iteration, farthest point in the cluster is split and this process is repeated until each cluster only contains a single point.



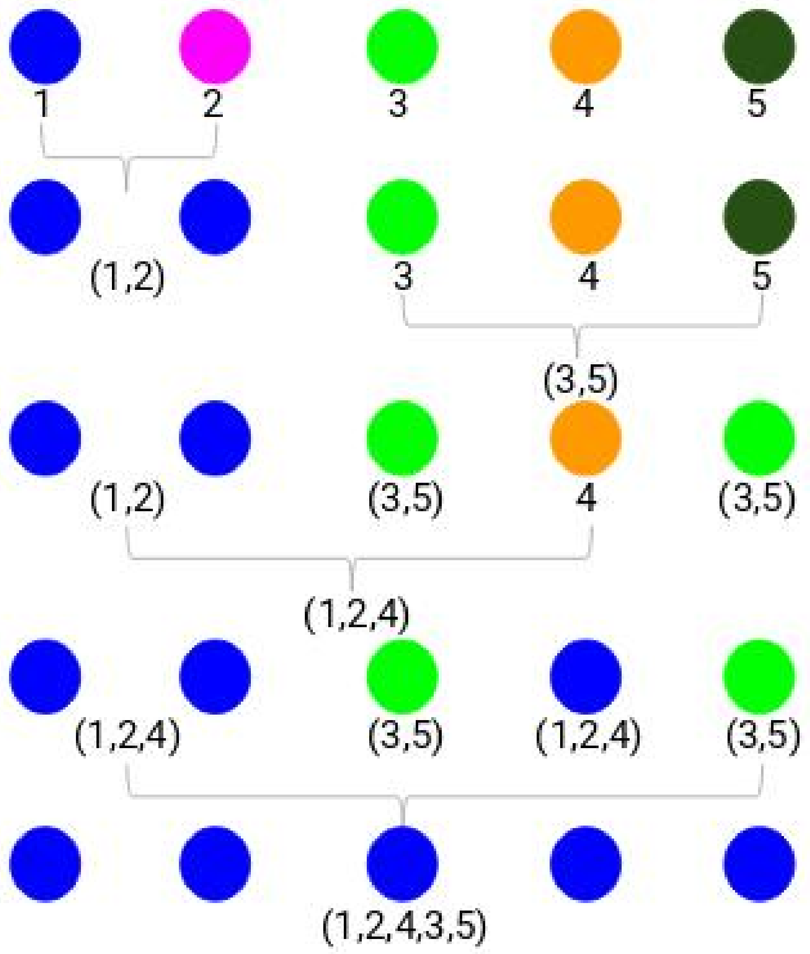
Dendrogram

- To get the number of clusters for hierarchical clustering, we make use of the concept called a **Dendrogram**.
- A dendrogram is a tree-like diagram that records the sequences of merges or splits.
- Let's get back to faculty-student example. Whenever we merge two clusters, a dendrogram record the distance between these clusters and represent it in graph f

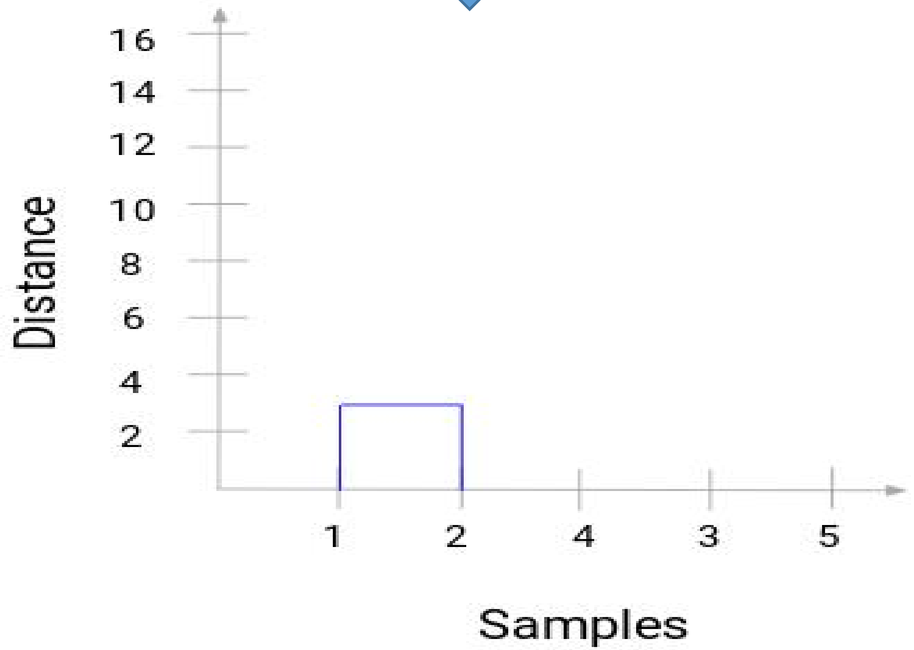


We have the samples of the dataset on the x-axis and the distance on the y-axis. Whenever two clusters are merged, we will join them in this dendrogram and the height of the join will be the distance between these points.

Dendrogram cont...



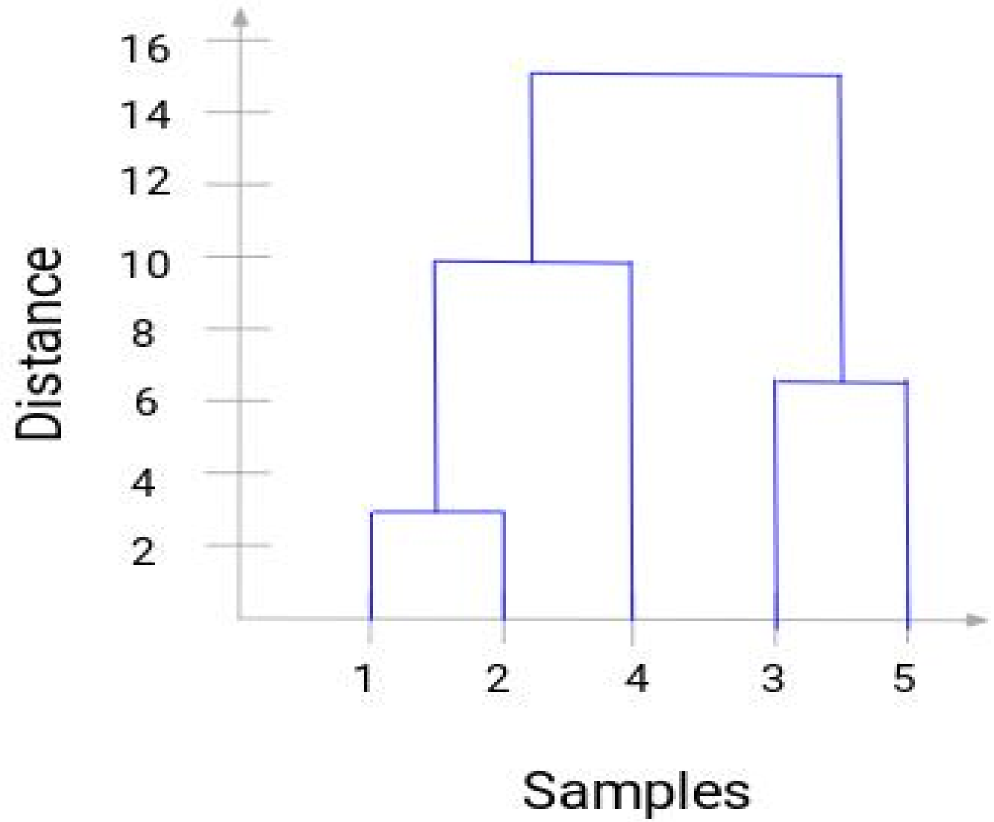
We started by merging sample 1 and 2 and the distance between these two samples was 3. Let's plot this in the dendrogram.



Here, we can see that we have merged sample 1 and 2. The vertical line represents the distance between these samples.

Dendrogram cont…

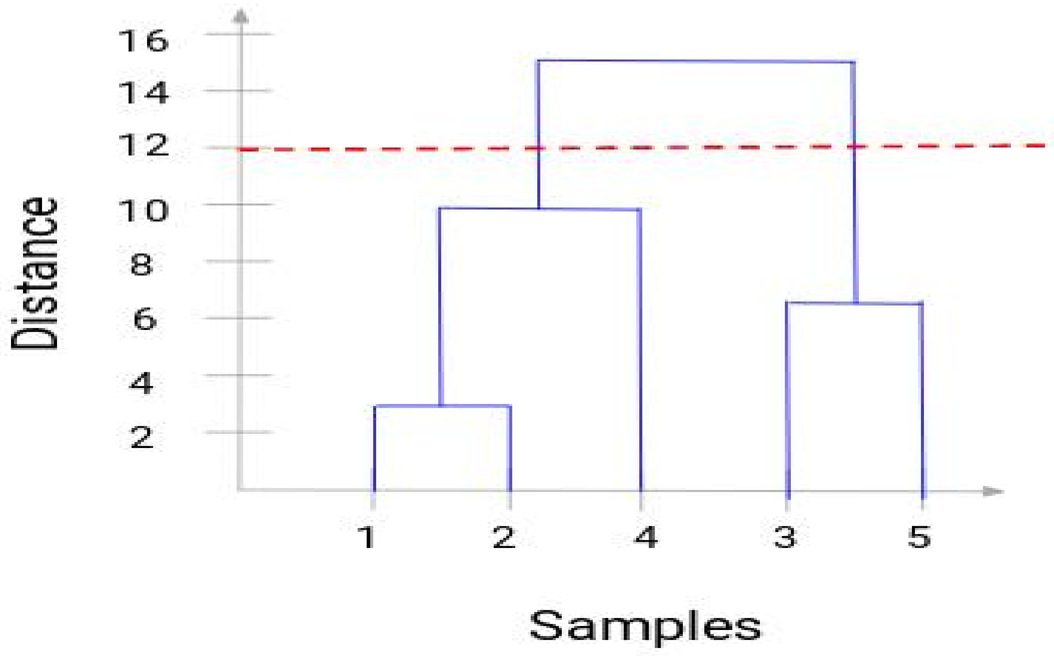
Similarly, we plot all the steps where we merged the clusters and finally, we get a dendrogram like this:



We can clearly visualize the steps of hierarchical clustering. **More the distance of the vertical lines in the dendrogram, more the distance between those clusters.**

Dendrogram cont…

Now, we can set a threshold distance and draw a horizontal line (Generally, the threshold is set in such a way that it cuts the tallest vertical line). Let's set this threshold as 12 and draw a horizontal line:



The number of clusters will be the number of vertical lines which are being intersected by the line drawn using the threshold. In the above example, since the red line intersects 2 vertical lines, we will have 2 clusters. One cluster will have a sample (1,2,4) and the other will have a sample (3,5).

Hierarchical Clustering closeness of two clusters

The decision of merging two clusters is taken on the basis of closeness of these clusters. There are multiple metrics for deciding the closeness of two clusters and primarily are:

- Euclidean distance
- Squared Euclidean distance
- Manhattan distance
- Maximum distance
- Mahalanobis distance

Names	Formula
Euclidean distance	$\|a-b\|_2 = \sqrt{\sum_i (a_i - b_i)^2}$
Squared Euclidean distance	$\|a-b\|_2^2 = \sum_i (a_i - b_i)^2$
Manhattan distance	$\|a-b\|_1 = \sum_i \lvert a_i - b_i \rvert$
maximum distance	$\|a-b\|_\infty = \max_i \lvert a_i - b_i \rvert$
Mahalanobis distance	$\sqrt{(a-b)^\top S^{-1} (a-b)}$ where S is the Covariance matrix