Clustering

Introduction

- ❑ There are many cases in which we do not have labelled data and need to find the hidden patterns from the given dataset.
- ❑ To solve such problems in machine learning, we use unsupervised learning techniques.

Unsupervised Learning:

- ❑ Is a type of machine learning in which models are trained using unlabelled dataset and are allowed to act on that data without any supervision.
- ❑ It cannot be applied directly to regression or classification problem, because we have the input data but no corresponding output data.
- ❑Goal is to find the underlying structure of dataset, group that data according to similarities and represent the data in a compressed format.

Why we use unsupervised learning?

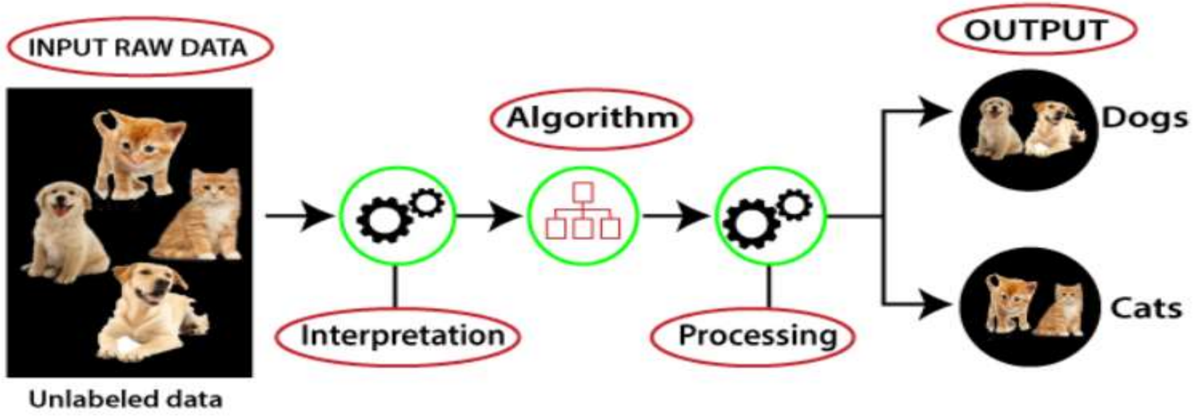
❑ Helpful to find the useful insights from the data.

❑ Much similar as human learns to think by their own experience, which makes it closer to AI.

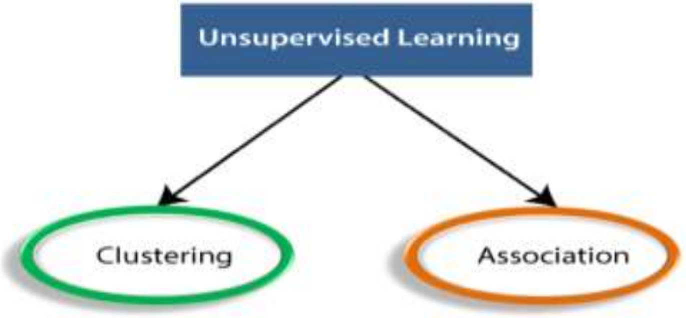
❑ In real world we do not always have input data with the corresponding output so to solve such cases we need unsupervised learning.

Working of unsupervised learning

- We take an unlabeled input data (not categorized and corresponding outputs are also not given).
- Now, this unlabeled input data is fed to the machine learning model in order to train it. Firstly, it will interpret the raw data to find the hidden patterns from the data and then will apply suitable algorithms.
- Once it applies the suitable algorithm, the algorithm divides the data objects into groups according to the similarities and difference between the objects.



Types of Unsupervised Learning Algorithm:



- **Clustering:** Clustering is a method of grouping the objects into clusters such that objects with most similarities remains into a group and has less or no similarities with the objects of another group.

- **Association**: An association rule is an unsupervised learning method which is used for finding the relationships between variables in the large database. It determines the set of items that occurs together in the dataset. Association rule makes marketing strategy more effective.

Clustering

❑ A way of grouping the data point into different clusters, consisting of similar data points.

❑ The objects with the possible similarities remain in a group, that has less or no similarities with another group.

❑ It finds some similar patterns in the unlabeled dataset such as shape, size, color etc and divides them as per the presence and absence of those similar patterns.

❑ After clustering technique, each cluster is provided with a cluster-ID

Application

1. Market segmentation
2. Image segmentation
3. Social Network Analysis
4. Amazon: Used in the recommendation system to provide the recommendation as per the search of products.
5. Netflix: recommend movies and webseries as per watch history

Example

Suppose you are the head of a rental store and wish to understand the preferences of your customers to scale up your business. Is it possible for you to look at the details of each customer and devise a unique business strategy for each one of them?

What you can do is cluster all of your customers into, say 10 groups based on their purchasing habits and use a separate strategy for customers in each of these 10 groups. And this is what we call clustering.

Types of clustering methods

Broadly speaking, clustering can be divided into two subgroups:

- ❑ **Hard Clustering:** In this, each input data point either belongs to a cluster completely or not. In the above example, each customer is put into one group out of the 10 groups.

- ❑**Soft Clustering**: In this, instead of putting each input data point into a separate cluster, a probability or likelihood of that data point being in those clusters is assigned. Soft clustering is method of grouping the data items such that an item can exist in multiple clusters. Each customer is assigned a probability to be in either of the 10 clusters of the retail store.

Different types of clustering algorithms

1. Partitioning Clustering
2. Density Based Clustering
3. Distribution Model-Based clustering
4. Hierarchical Clustering
5. Fuzzy clustering

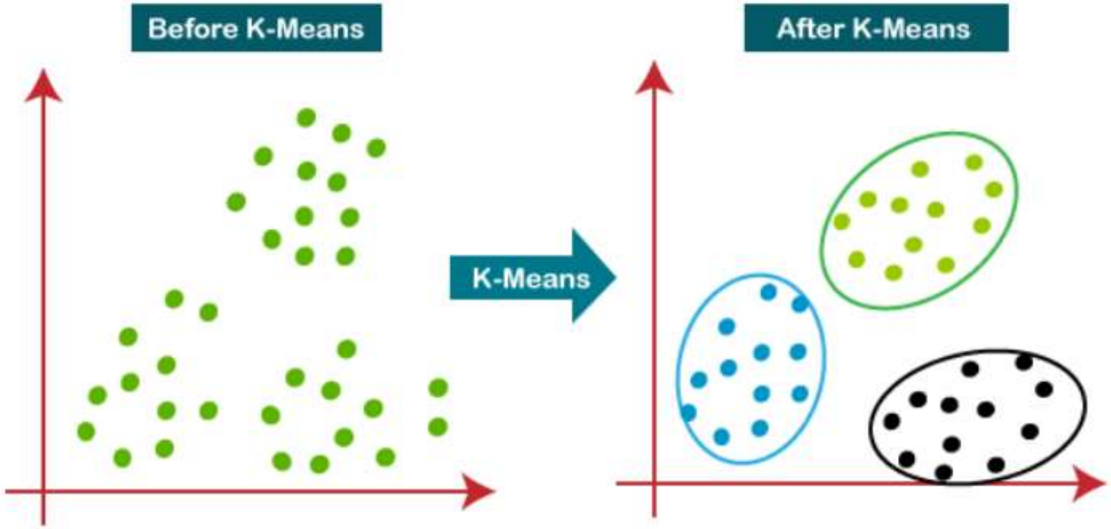
Partitioning Clustering

- ❑ Divides data into non-hierarchical groups
- ❑ Also known as centroid-based method
- ❑ These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid or cluster center of the clusters.
- ❑ The most common example of partitioning clustering is the K-Means Clustering algorithm.

K-means clustering

- ❑ K-Means Clustering is an unsupervised learning algorithm, which groups the unlabeled dataset into different clusters.
- ❑Here K defines the number of pre-defined clusters that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.
- ❑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 classify dataset by dividing the samples into different clusters of equal variances.
- ❑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.



Algorithm

Step 1: Randomly select K clusters, v1, v2….vk

Step 2: Calculate the distance between each data points aj and each cluster centers vi

Step 3: Assign each data point a_j to the cluster center vi for which the distance $\|a_j – v_i\|$ is minimum.

Step 4: Recalculate each cluster center by taking the average of clusters data points.

Step 5: Repeat from step 2 to step 5 until the recalculated cluster centers are some as previous or no reassignment of data points happened.

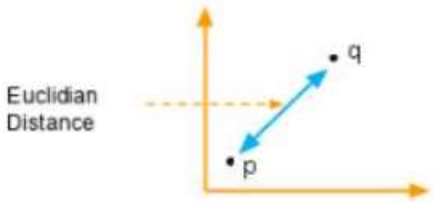
Distance between datapoints

- ❑ We assume that each data point is a n-dimensional vector
- ❑K-Means clustering supports various kinds of distance measures, such as:
    1. Euclidean distance measure
    2. Manhattan distance measure

Euclidean Distance Measure

❑**If we have a point P and point Q, the Euclidean distance is an ordinary straight line.**

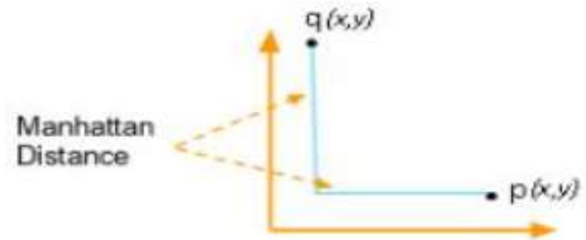
$$\mathbf{d}=\sqrt{}\textstyle\sum_{i=1}^{n}(\ q_{i}-p_{i})^{2}$$



Manhattan Distance Measure

❑ **The Manhattan distance is the simple sum of the horizontal and vertical components or the distance between two points measured along axes at right angles.**

$$\boxed{\mathbf{d}= \sum_{i=1}^{n} | q_x - p_x \; | + | q_x - p_y |}$$



Question

Use K-means clustering algorithm to decide the following data into two clusters.

x1	1	2	2	3	4	5
x2	1	1	3	2	3	5

Step 1: Choose randomly two clusters:

  - V1=(2,1) and v2=(2,3)

Step 2: Find the distance between the cluster center's and each data points.

Datapoint	Distance from v1	Distance from v2	Assigned center
a1(1,1)	1	2.24	V1
a2(2,1)	0	2	V1
a3(2,3)	2	0	V2
a4(3,2)	1.4	1.4	V1
a5(4,3)	2.83	2	V2
a6(5,5)	5	3.61	v2

Step 3: Assign each datapoint to clusters

cluster of v1 ={a1,a2,a4}

cluster of v2 = {a3,a5,a6}

Step 4: Recalculate the cluster center’s

v1=[a1+a2+a4]/3

= (2,1.33)

v2=[a3+a5+a6]/3

= (3.67,3.67)

Step 5: Repeat from step 2 until we get same cluster center or same cluster elements as in the previous iteration

datapoint	Distance from v1	Distance from v2	Assigned center
a1(1,1)	1.05	3.78	v1
a2(2,1)	0.33	3.15	v1
a3(2,3)	1.67	1.8	v1
a4(3,2)	1.204	1.8	v1
a5(4,3)	2.605	0.75	v2
a6(5,5)	4.74	1.88	v2

Cluster 1 of v1 ={a1,a2,a3,a4}

Cluster 2 of v2={a5,a6}

Recalculate the cluster centers:

v1= (2,1.75)

v2=(4.5,4)

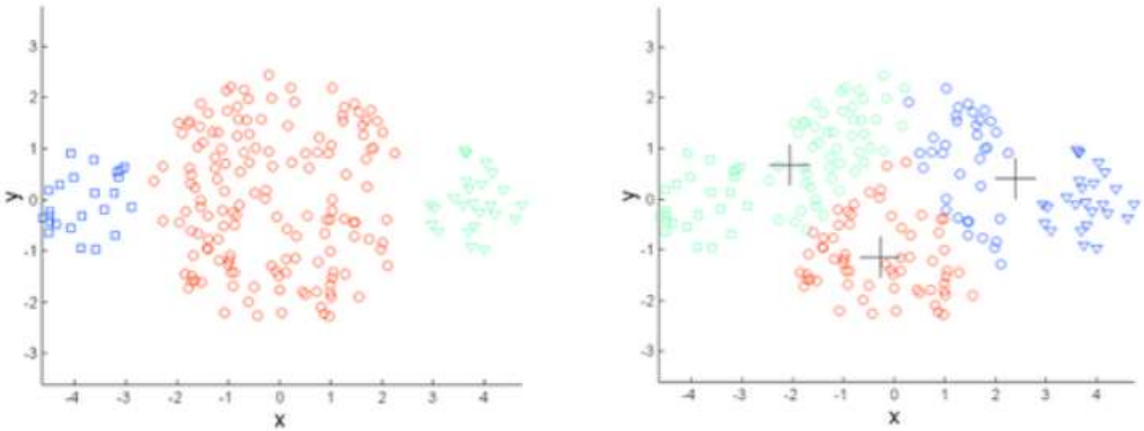
Final clusters:

datapoint	Distance from v1	Distance from v2	Assigned center
a1(1,1)	1.25	4.61	v1
a2(2,1)	0.75	3.9	v1
a3(2,3)	1.25	2.69	v1
a4(3,2)	1.03	2.5	v1
a5(4,3)	2.36	1.12	v2
a6(5,5)	4.42	1.12	v2

Challenges with the k-means clustering

- **Finding the optimal k value,** especially for noisy data. The appropriate value of k depends on the data structure and the problem being solved. It is important to choose the right value of k, as a small value can result in under-clustered data, and a large value can cause over-clustering.

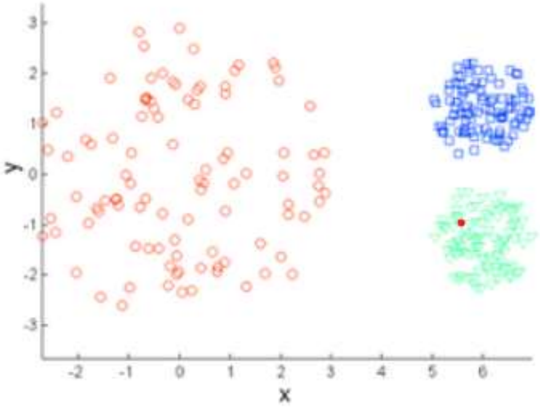
- **The size of clusters is different.**



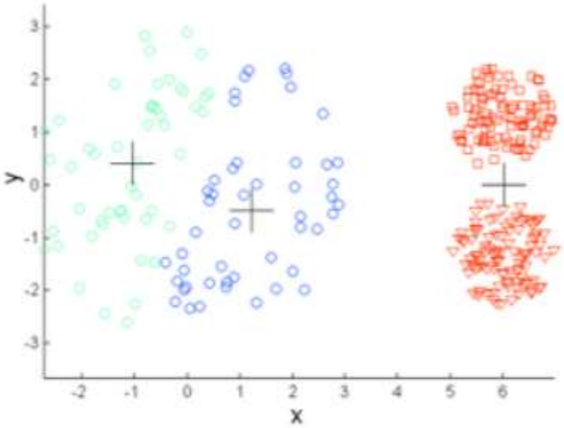
Original Points

K-means (k = 3)

- The densities of the original points are different.



Original Points



K-means (k = 3)

Solutions

- Use a higher number of clusters

- Determining the optimal number of clusters: One commonly used method to find the optimal number of clusters is the **elbow method**, which plots the sum of squared Euclidean distances between data points and their cluster center and chooses the number of clusters where the change in the sum of squared distances begins to level off.

K-Medoids Clustering

K-Medoids Clustering: Find representative objects (medoids) in clusters

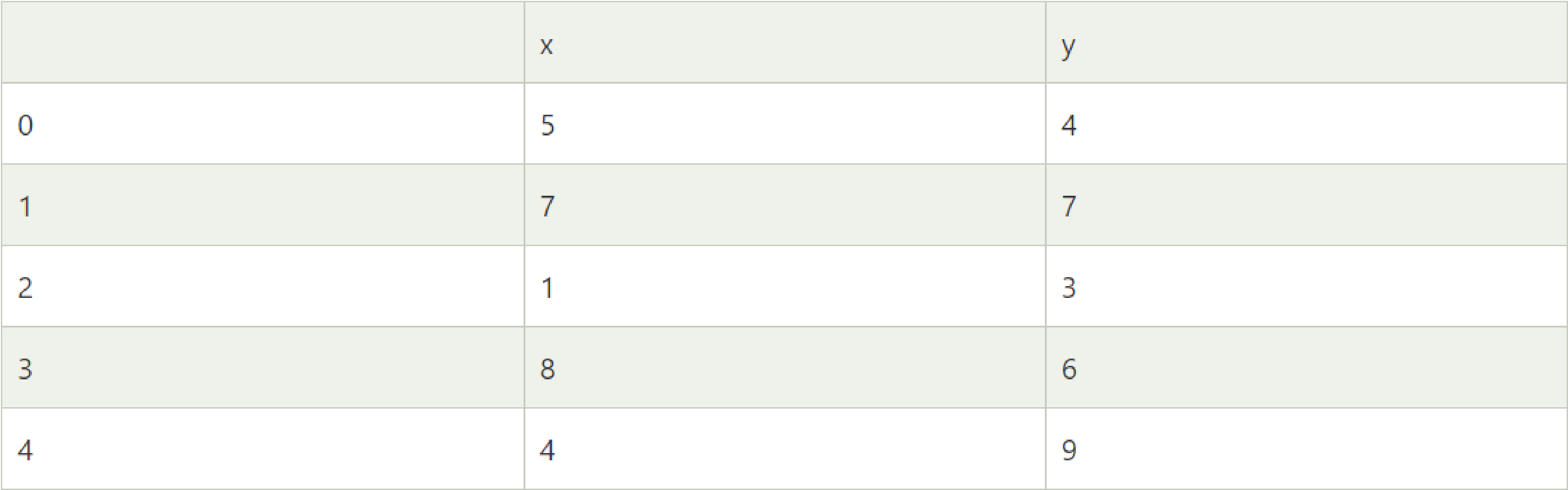
PAM (Partitioning Around Medoids, Kaufmann & Rousseeuw 1987)

- 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 (due to the computational complexity)

K-Medoids:

1. Choose k number of random points from the data and assign these k points to k number of clusters. These are the initial medoids.
2. For all the remaining data points, calculate the distance from each medoid and assign it to the cluster with the nearest medoid.
3. Calculate the total cost (Sum of all the distances from all the data points to the medoids)
4. Select a random point as the new medoid and swap it with the previous medoid. Repeat 2 and 3 steps.
5. If the total cost of the new medoid is less than that of the previous medoid, make the new medoid permanent and repeat step 4.
6. If the total cost of the new medoid is greater than the cost of the previous medoid, undo the swap and repeat step 4.
7. The Repetitions have to continue until no change is encountered with new medoids to classify data points.

example



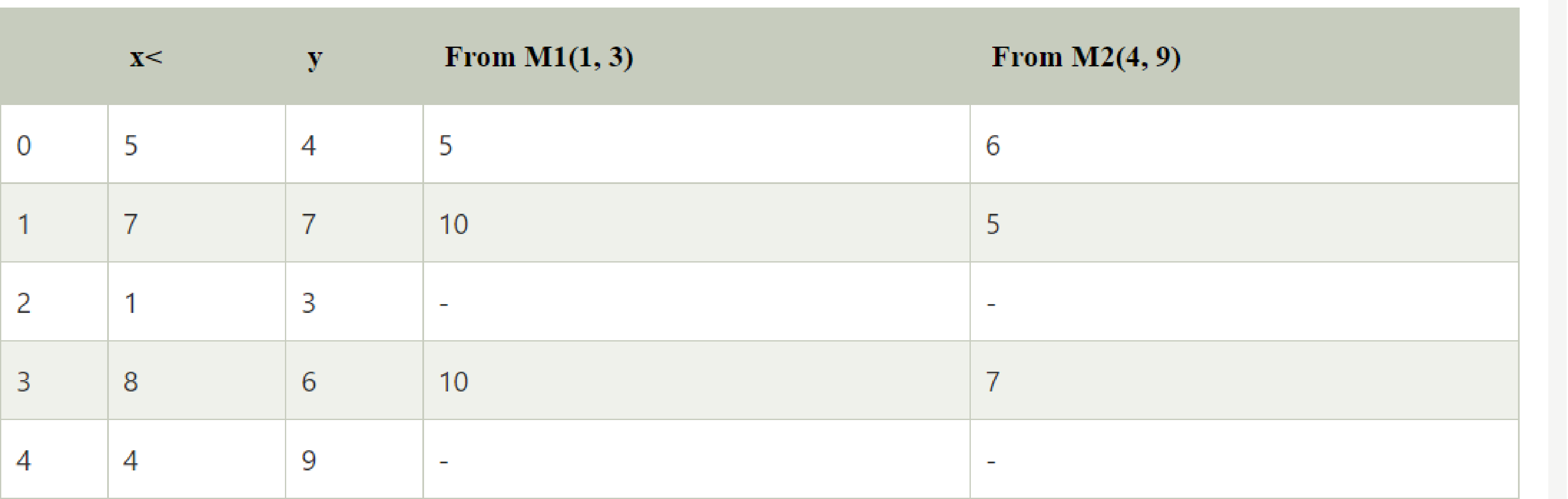
	x	y
0	5	4
1	7	7
2	1	3
3	8	6
4	4	9

K=2

Initial medoids: M1(1, 3) and M2(4, 9)

Calculation of distances

Calculation of distance:



	x<	y	From M1(1, 3)	From M2(4, 9)
0	5	4	5	6
1	7	7	10	5
2	1	3	-	-
3	8	6	10	7
4	4	9	-	-

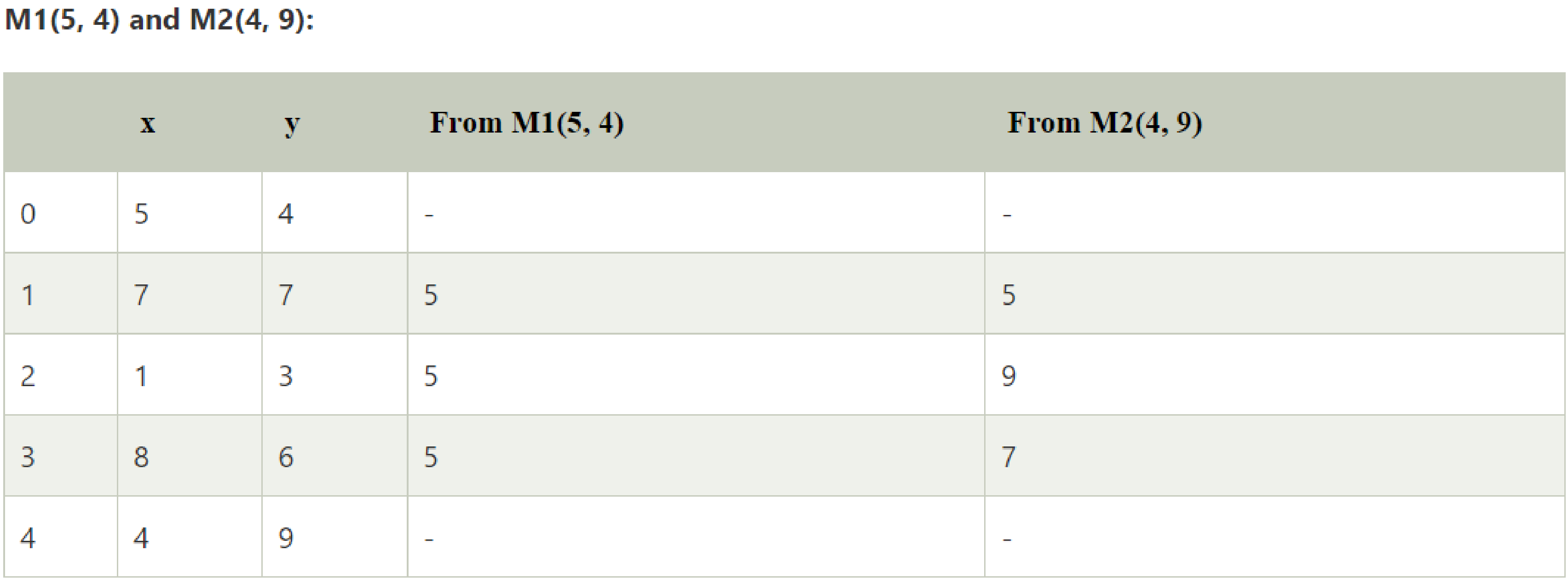
Cluster 1: 0

Cluster 2: 1, 3

Calculation of total cost:

(5) + (5 + 7) = 17

Random medoid: (5, 4)



M1(5, 4) and M2(4, 9):

	x	y	From M1(5, 4)	From M2(4, 9)
0	5	4	-	-
1	7	7	5	5
2	1	3	5	9
3	8	6	5	7
4	4	9	-	-

Cluster 1: 2, 3
Cluster 2:1

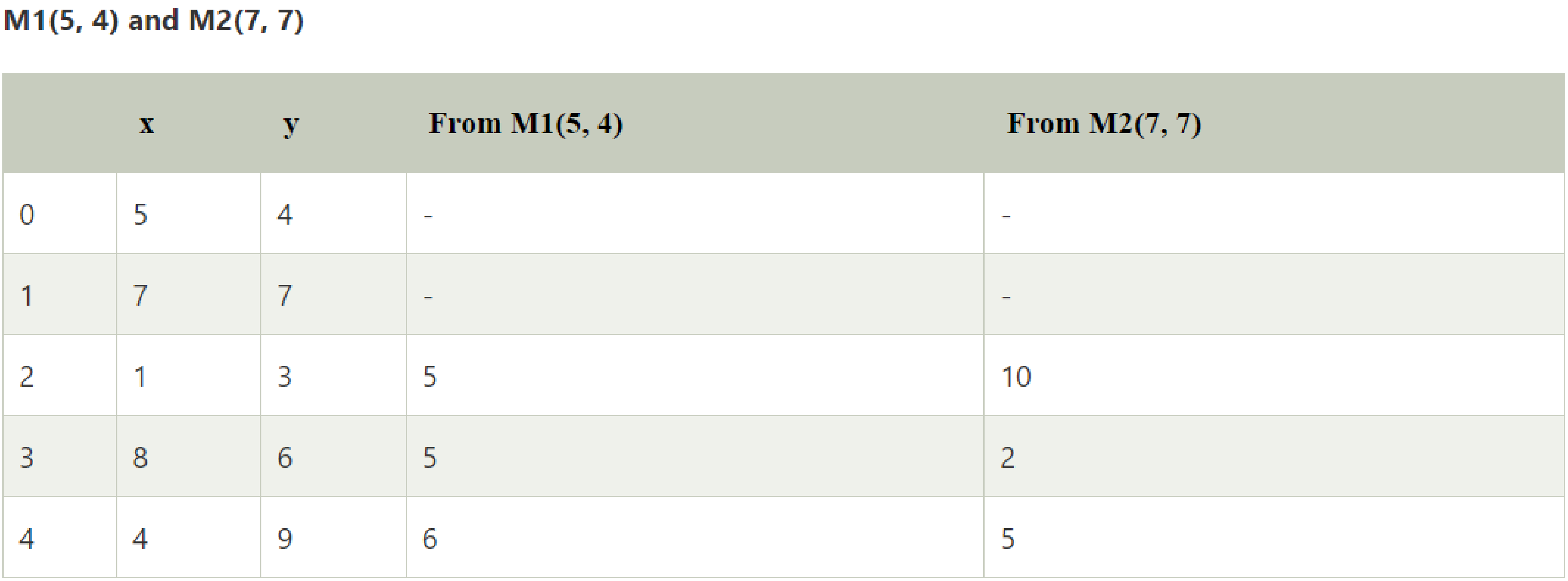
Calculation of total cost:

$$(5 + 5) + 5 = 15$$

Less than the previous cost

New medoid: (5, 4).

Random medoid: (7, 7)



M1(5, 4) and M2(7, 7)

	x	y	From M1(5, 4)	From M2(7, 7)
0	5	4	-	-
1	7	7	-	-
2	1	3	5	10
3	8	6	5	2
4	4	9	6	5

Cluster 1: 2

Cluster 2: 3, 4

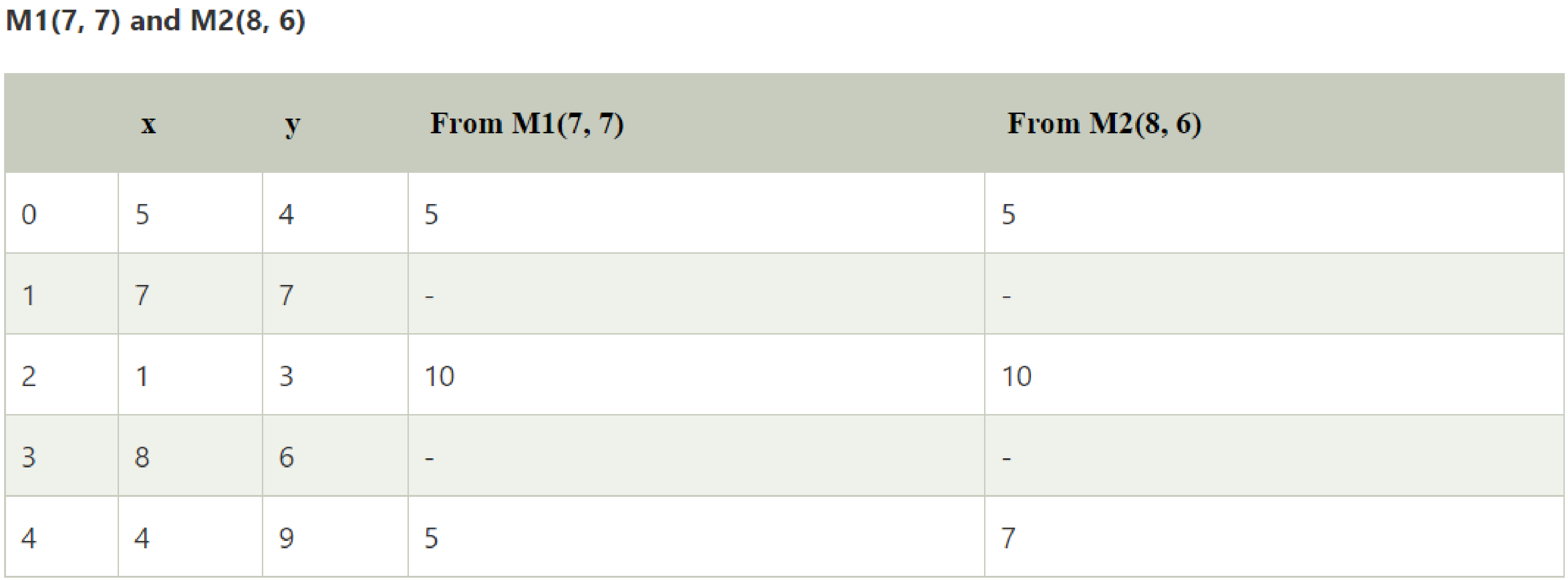
Calculation of total cost:

(5) + (2 + 5) = 12

Less than the previous cost

New medoid: (7, 7).

Random medoid: (8, 6)



M1(7, 7) and M2(8, 6)

	x	y	From M1(7, 7)	From M2(8, 6)
0	5	4	5	5
1	7	7	-	-
2	1	3	10	10
3	8	6	-	-
4	4	9	5	7

Cluster 1: 4

Cluster 2: 0, 2

Calculation of total cost: (5) + (5 + 10) = 20

Greater than the previous cost

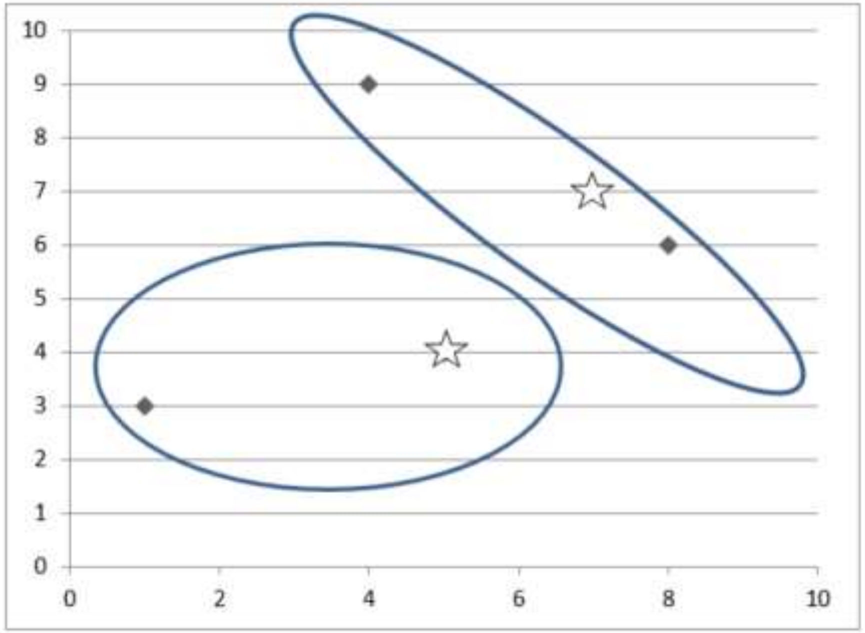
UNDO : Hence, the final medoids: M1(5, 4) and M2(7, 7)

Cluster 1: 2

Cluster 2: 3, 4

Total cost: 12

Final results


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