Data Mining & Data Warehouse

Dr. Raed Ibraheem Hamed

University of Human Development,
College of Science and Technology
Department of Information Technology

2016 – 2017

Road map

- Common Distance measures
- The Euclidean Distance between 2 variables
- K-means Clustering
- How the K-Mean Clustering algorithm works?
- Step 1:
- Step 2:
- Step 3:
- Step 4:
- More examples of K-Mean Clustering
- Demonstration of PAM
- Steps of PAM

Common Distance measures:

- **Distance measure** will determine how the *similarity* of two elements is calculated and it will influence the shape of the clusters.

  They include:

1. The **Euclidean distance** (also called 2-norm distance) is given by:

$$d = \sqrt{\sum_{i=1}^{p} (v_{1i} - v_{2i})^2}$$

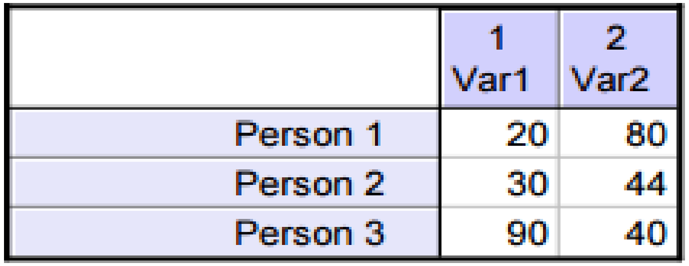
2. The **Manhattan distance** .

The Euclidean Distance between 2 variables

The formula for calculating the distance between the two variables, given three persons scoring on each as shown below is:

$$d = \sqrt{\sum_{i=1}^{p} (v_{1i} - v_{2i})^2}$$

Table 1



	1 Var1	2 Var2
Person 1	20	80
Person 2	30	44
Person 3	90	40

For the distance between person 1 and 2, the calculation is:

$$d = \sqrt{(20-30)^2 + (80-44)^2} = 37.36$$

For the distance between person 1 and 3, the calculation is:

$$d = \sqrt{(20-90)^2 + (80-40)^2} = 80.62$$

For the distance between person 2 and 3, the calculation is:

$$d = \sqrt{(30-90)^2 + (44-40)^2} = 60.13$$

K-means Clustering

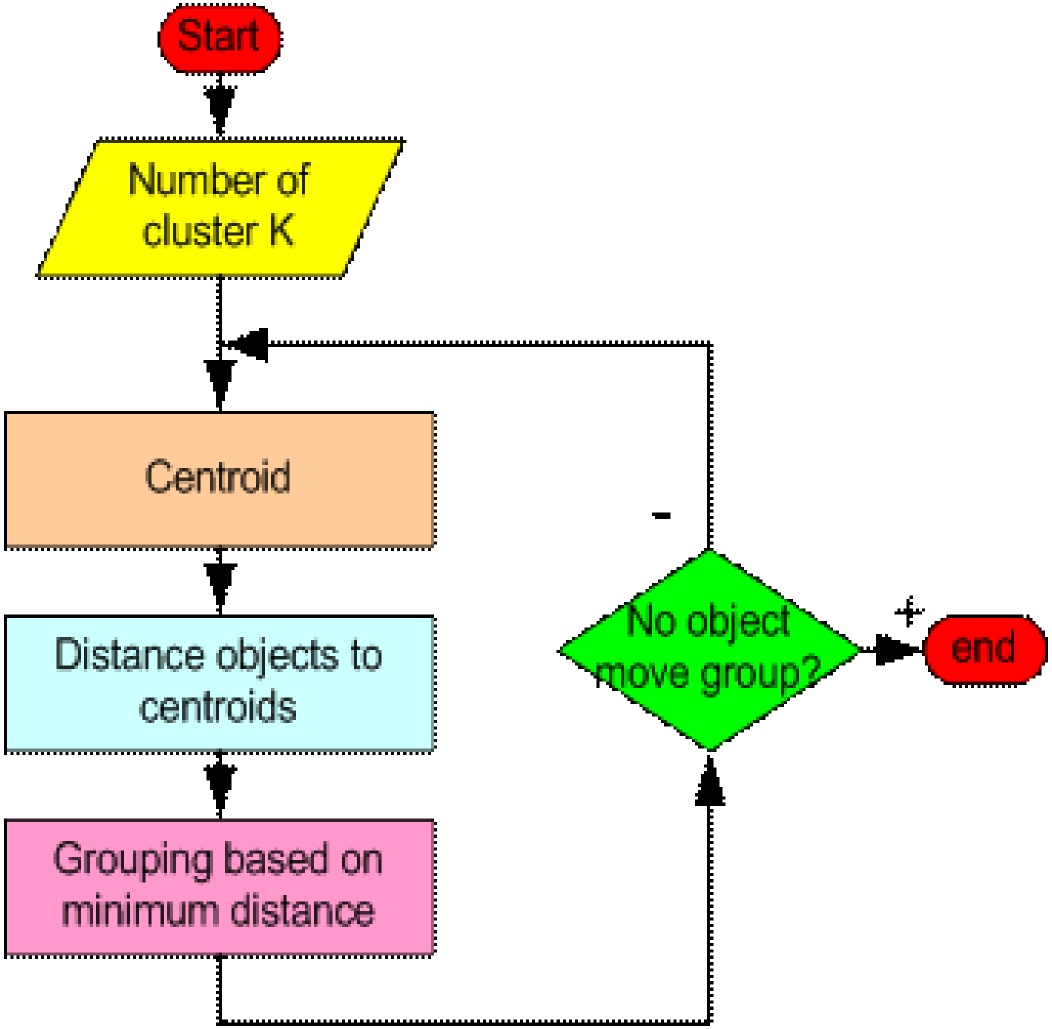
Basic Algorithm:

- **Step 0**: select K
- **Step 1**: randomly select initial cluster seeds
- **Step 2**: calculate distance from each object to each cluster seed.
- What type of distance should we use?
  - **Squared Euclidean distance**
- **Step 3**: Assign each object to the closest cluster

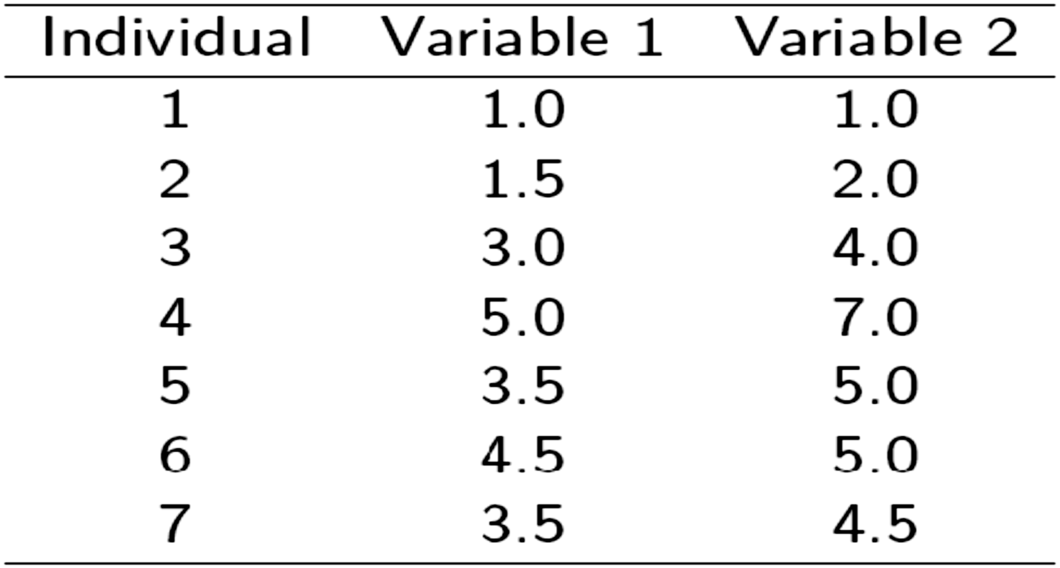
K-means Clustering

- **Step 4**: Compute the new centroid for each cluster
- **Iterate:**
  - Calculate distance from objects to cluster centroids.
  - Assign objects to closest cluster
  - Recalculate new centroids
- **Stop based on convergence criteria**
  - No change in clusters
  - Max iterations

How the K-Mean Clustering algorithm works?


Start
Number of cluster K
Centroid
Distance objects to centroids
Grouping based on minimum distance
No object move group?
-
+
end

A Simple example showing the implementation of k-means algorithm



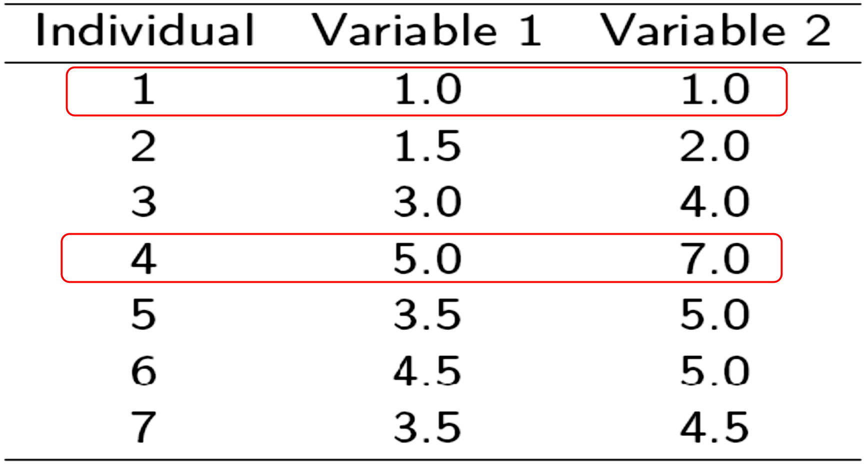
Individual	Variable 1	Variable 2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5

(Using K=2)

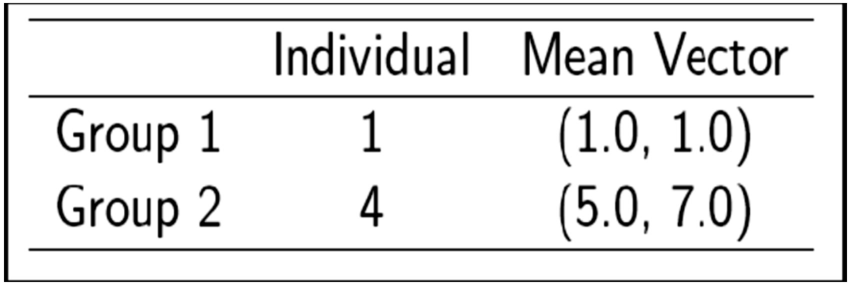
Step 1:

Initialization: Randomly we choose following two centroids (k=2) for two clusters.

In this case the 2 centroid are: **m1**=(1.0,1.0) and **m2**=(5.0,7.0).



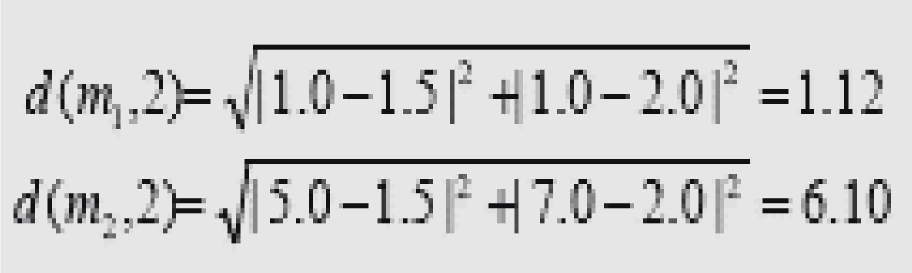
Individual	Variable 1	Variable 2
1	1.0	1.0
2	1.5	2.0
3	3.0	4.0
4	5.0	7.0
5	3.5	5.0
6	4.5	5.0
7	3.5	4.5



	Individual	Mean Vector
Group 1	1	(1.0, 1.0)
Group 2	4	(5.0, 7.0)

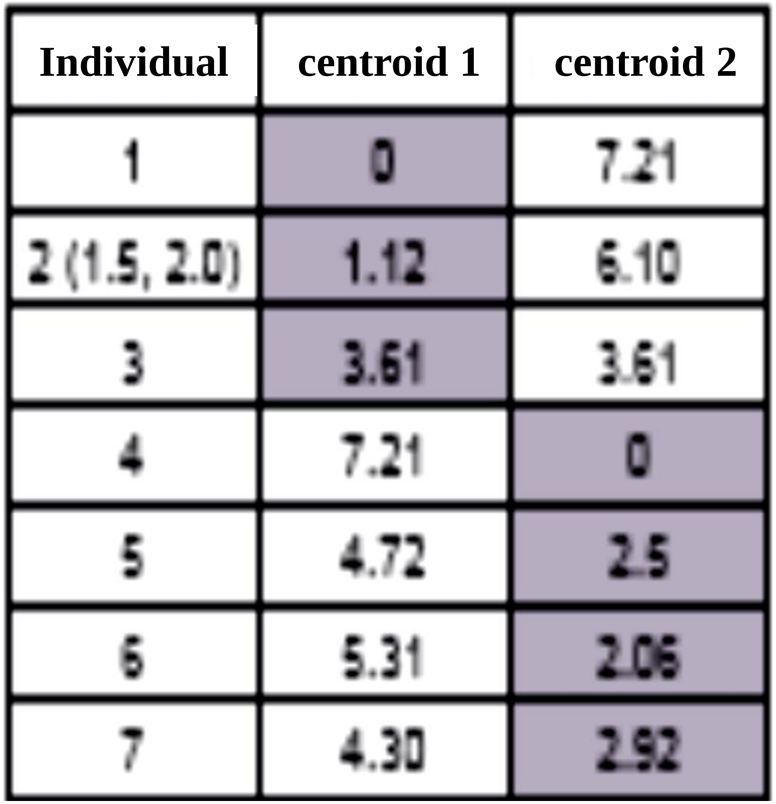
Step 2:

- Now using these centroids (i.e. m1(1.0, 1.0), and m2(5.0, 7.0)) we compute the **Euclidean distance of each object**, as shown in table.



$$d(m_1,2)=\sqrt{|1.0-1.5|^2+|1.0-2.0|^2}=1.12$$

$$d(m_2,2)=\sqrt{|5.0-1.5|^2+|7.0-2.0|^2}=6.10$$



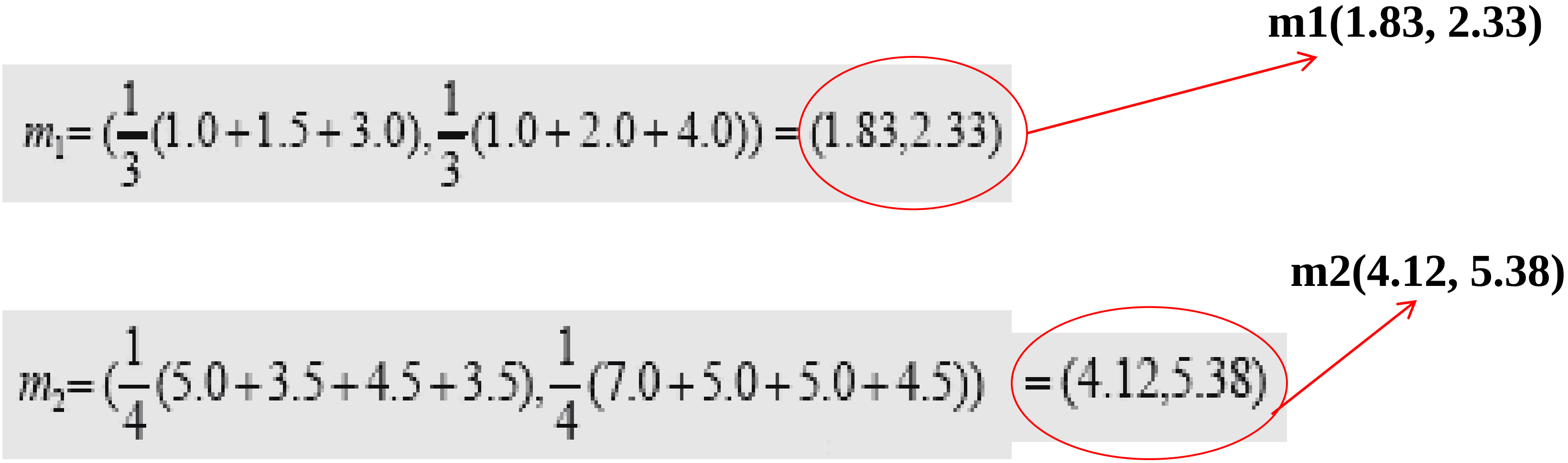
Individual	centroid 1	centroid 2
1	0	7.21
2 (1.5, 2.0)	1.12	6.10
3	3.61	3.61
4	7.21	0
5	4.72	2.5
6	5.31	2.06
7	4.30	2.92

- Thus, we obtain two clusters containing:

{1,2,3} and {4,5,6,7}.

Step 2:

- Now we compute the new centroids as:



$$m_1 = (\frac{1}{3}(1.0+1.5+3.0), \frac{1}{3}(1.0+2.0+4.0)) = (1.83, 2.33)$$

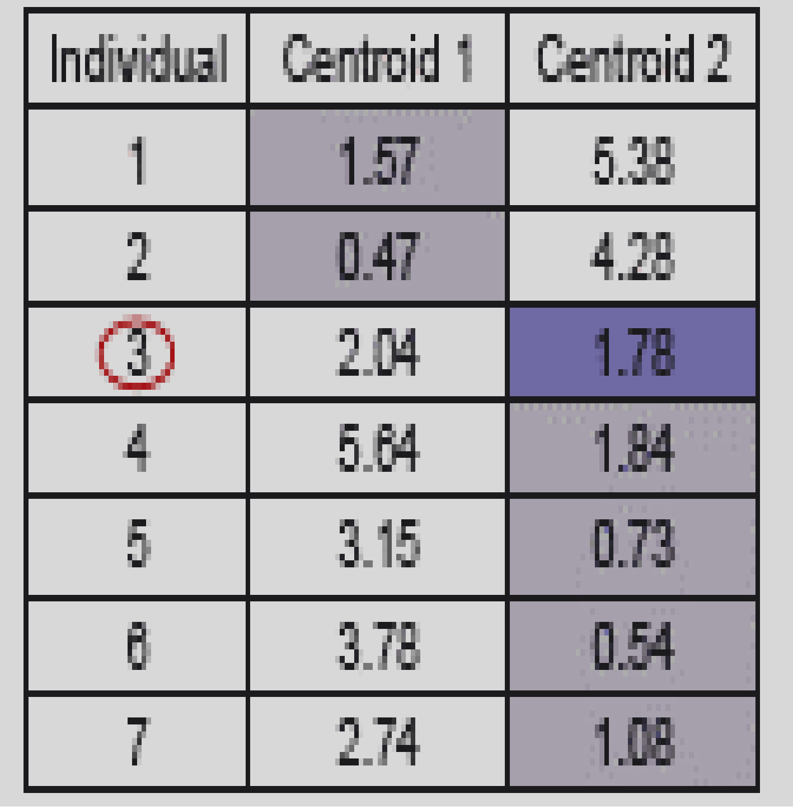
m1(1.83, 2.33)

$$m_2 = (\frac{1}{4}(5.0+3.5+4.5+3.5), \frac{1}{4}(7.0+5.0+5.0+4.5)) = (4.12, 5.38)$$

m2(4.12, 5.38)

Step 3:

- Now using these centroids (i.e. m1(1.83, 2.33), and m2(4.12, 5.38)) **to compute the Euclidean distance** of each object, as shown in table.



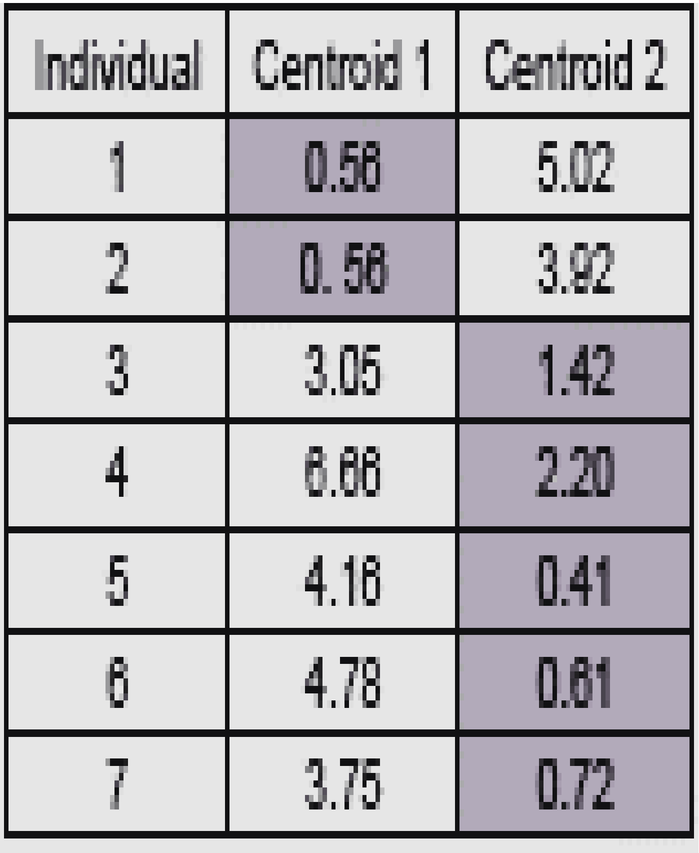
Individual	Centroid 1	Centroid 2
1	1.57	5.38
2	0.47	4.28
3	2.04	1.78
4	5.64	1.84
5	3.15	0.73
6	3.78	0.54
7	2.74	1.08

Step 3:

- Therefore, the new clusters are: {1,2} and {3,4,5,6,7}
- Next centroids are: m1=(1.25,1.5) and m2 = (3.9,5.1)
- **Note: every time we need to compute the new centroids depending on the original table.**

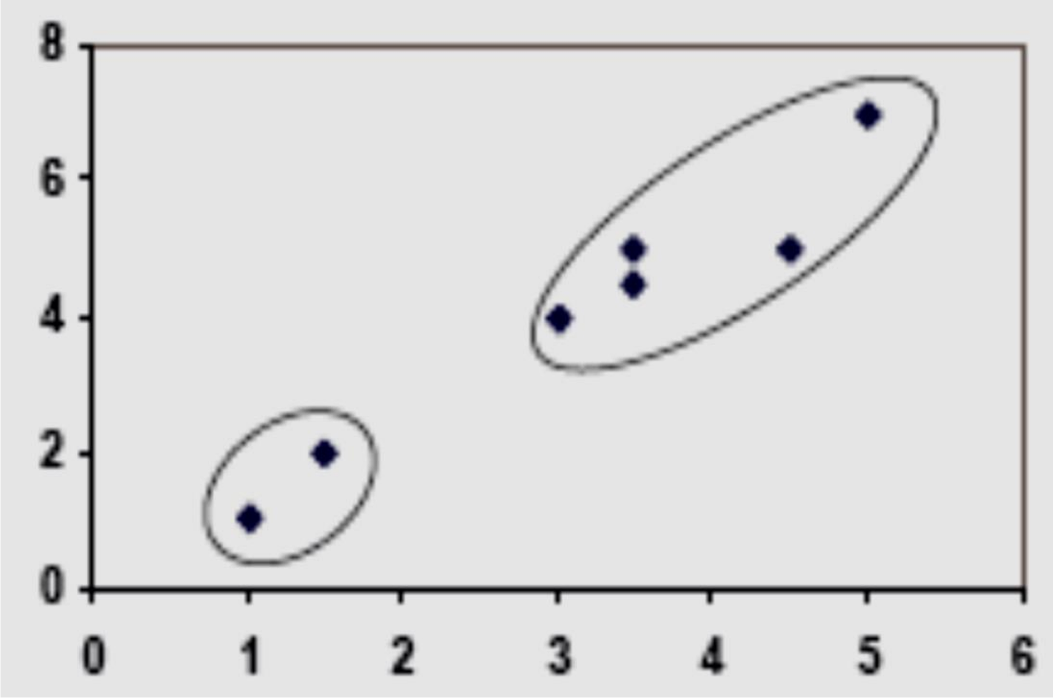
Step 4 :

- **We compute the Euclidean distance**
- The clusters obtained are: {1,2} and {3,4,5,6,7}
- Therefore, there is no change in the cluster.
- **Thus, the algorithm comes to a halt here and final result consist of 2 clusters {1,2} and {3,4,5,6,7}.**



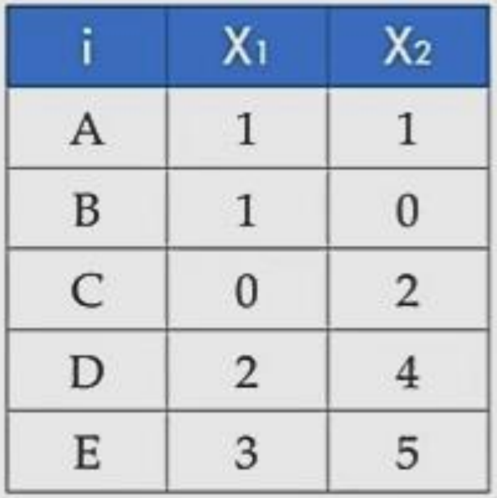
Individual	Centroid 1	Centroid 2
1	0.56	5.02
2	0. 56	3.92
3	3.05	1.42
4	6.66	2.20
5	4.16	0.41
6	4.78	0.61
7	3.75	0.72

PLOT

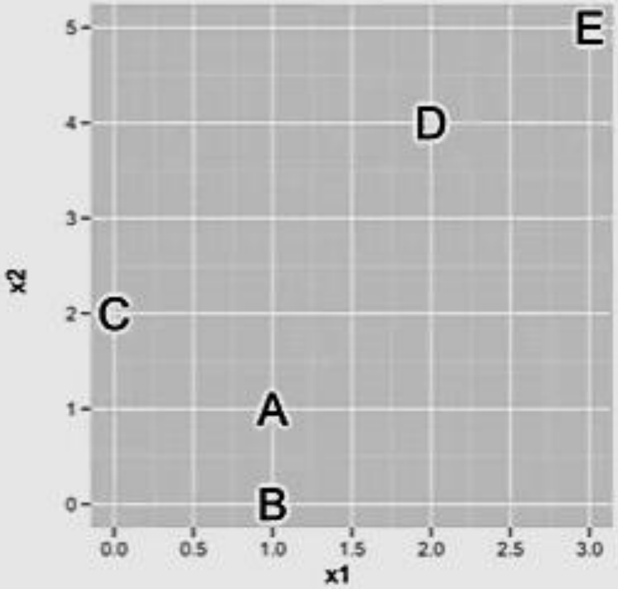


Step - 0

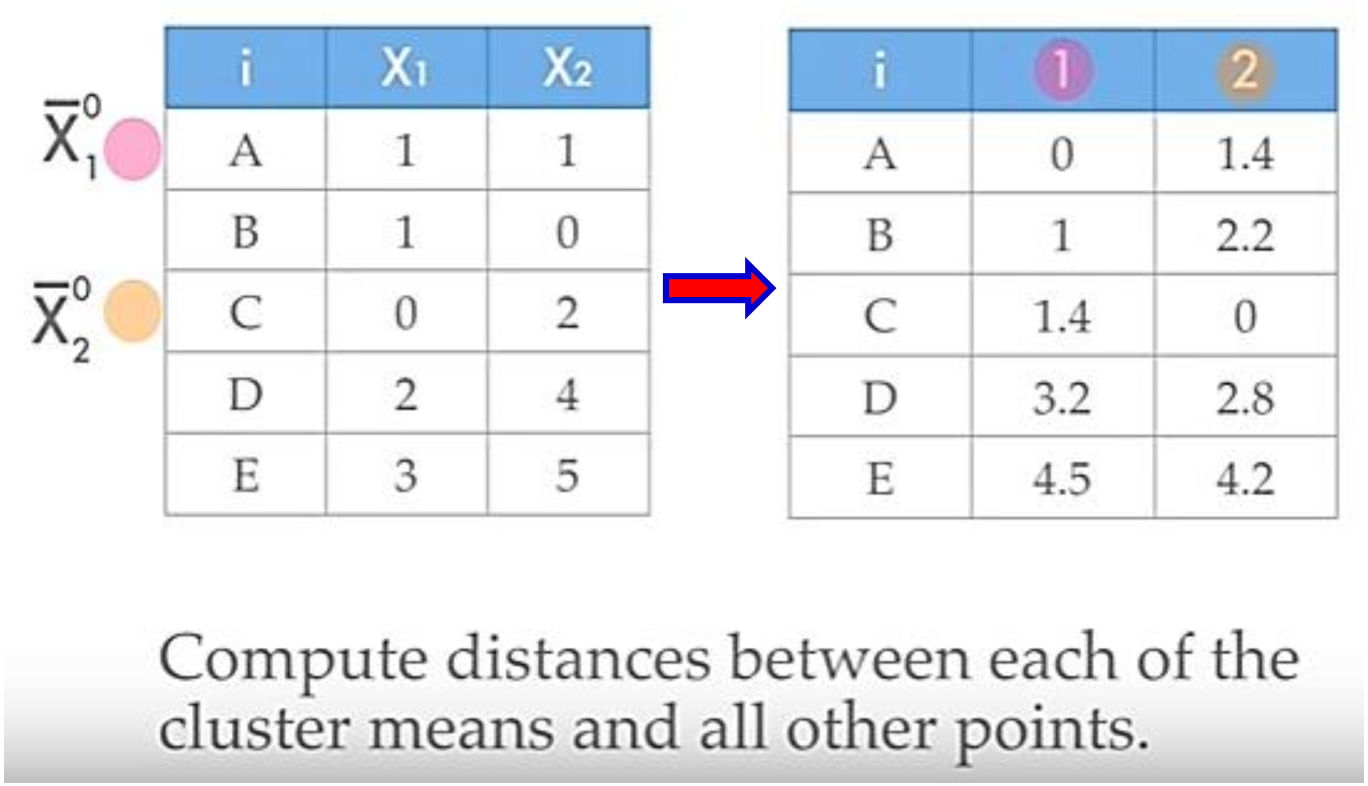
- Use K=2 Suppose **A** and **C** are Randomly selected as the initial means.



i	X_1	X_2
A	1	1
B	1	0
C	0	2
D	2	4
E	3	5



Step – 1.1



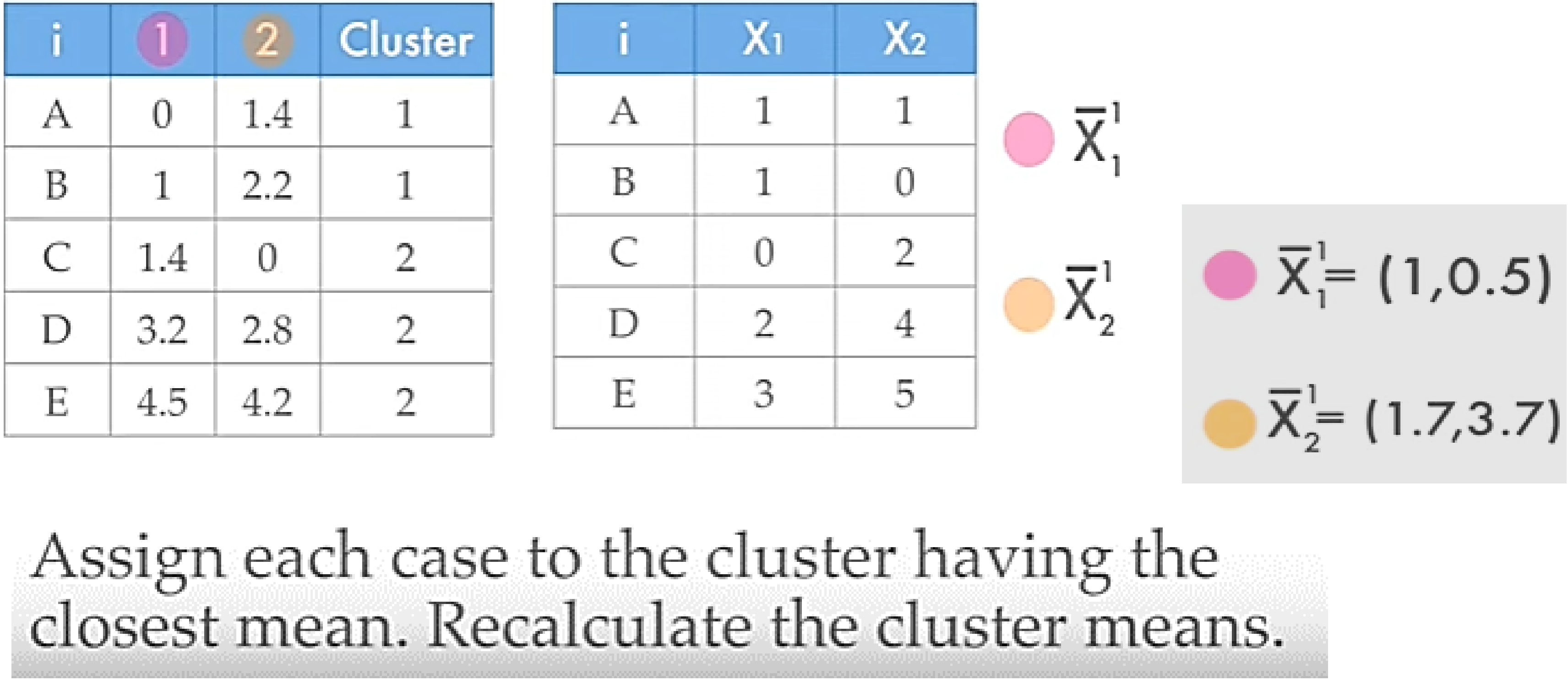
	i	X_1	X_2
$\bar{X}_1^0$ ●	A	1	1
	B	1	0
$\bar{X}_2^0$ ●	C	0	2
	D	2	4
	E	3	5

i	1	2
A	0	1.4
B	1	2.2
C	1.4	0
D	3.2	2.8
E	4.5	4.2

Compute distances between each of the cluster means and all other points.

Step – 1.1

❖ The clusters obtained are: {A,B} and {C,D,E}



i	1	2	Cluster
A	0	1.4	1
B	1	2.2	1
C	1.4	0	2
D	3.2	2.8	2
E	4.5	4.2	2

i	X_1	X_2
A	1	1
B	1	0
C	0	2
D	2	4
E	3	5

● $\bar{X}_1^1$

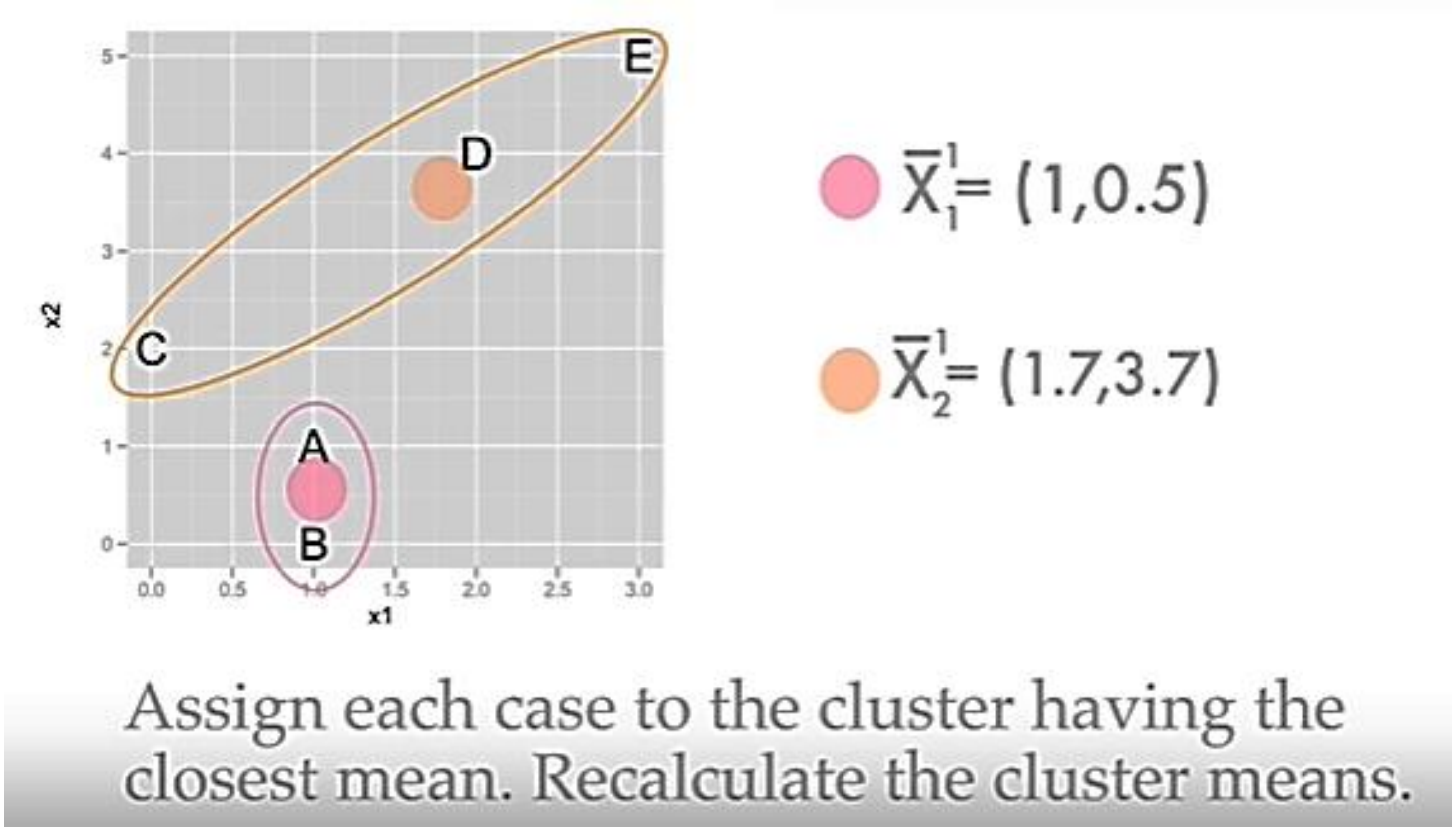
● $\bar{X}_2^1$

● $\bar{X}_1^1 = (1,0.5)$

● $\bar{X}_2^1 = (1.7,3.7)$

Assign each case to the cluster having the closest mean. Recalculate the cluster means.

Step – 1.1 PLOTS



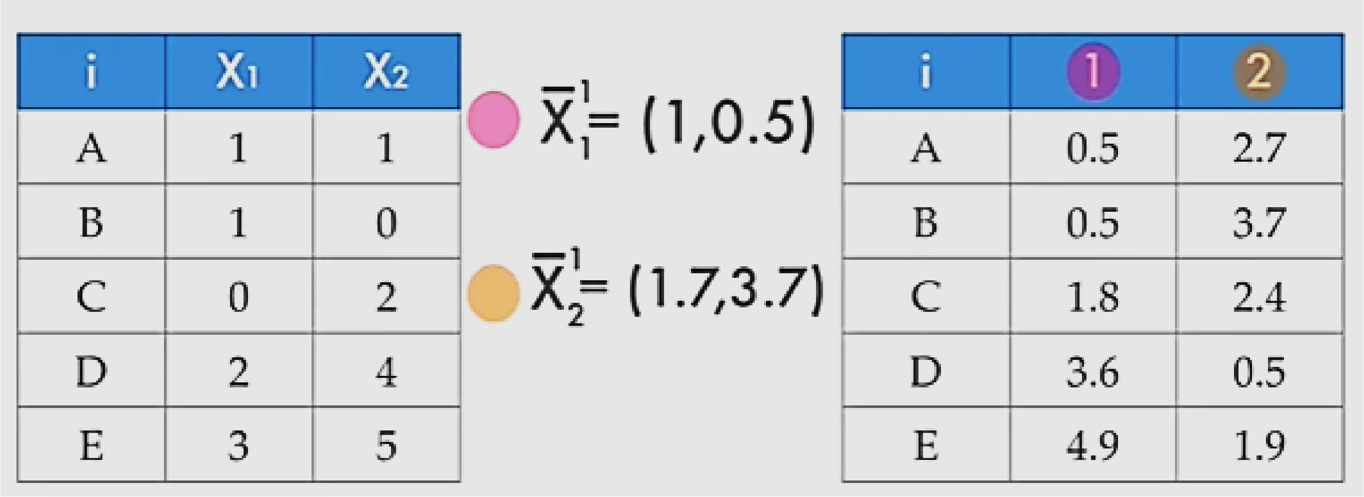
● $\bar{X}_1^1 = (1,0.5)$

● $\bar{X}_2^1 = (1.7,3.7)$

Assign each case to the cluster having the closest mean. Recalculate the cluster means.

Step – 2.1

Compute distances between each of the cluster means and all other points.



i	X_1	X_2
A	1	1
B	1	0
C	0	2
D	2	4
E	3	5

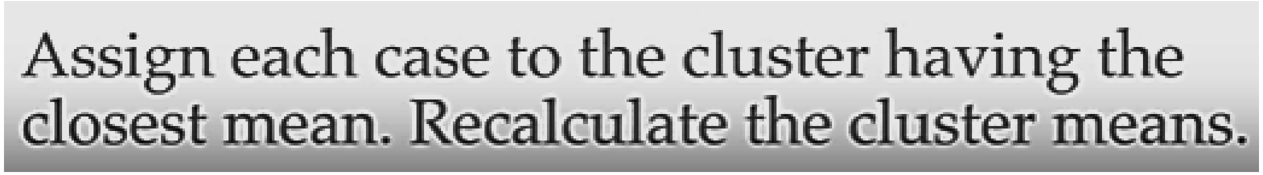
● $\bar{X}_1^1 = (1,0.5)$

● $\bar{X}_2^1 = (1.7,3.7)$

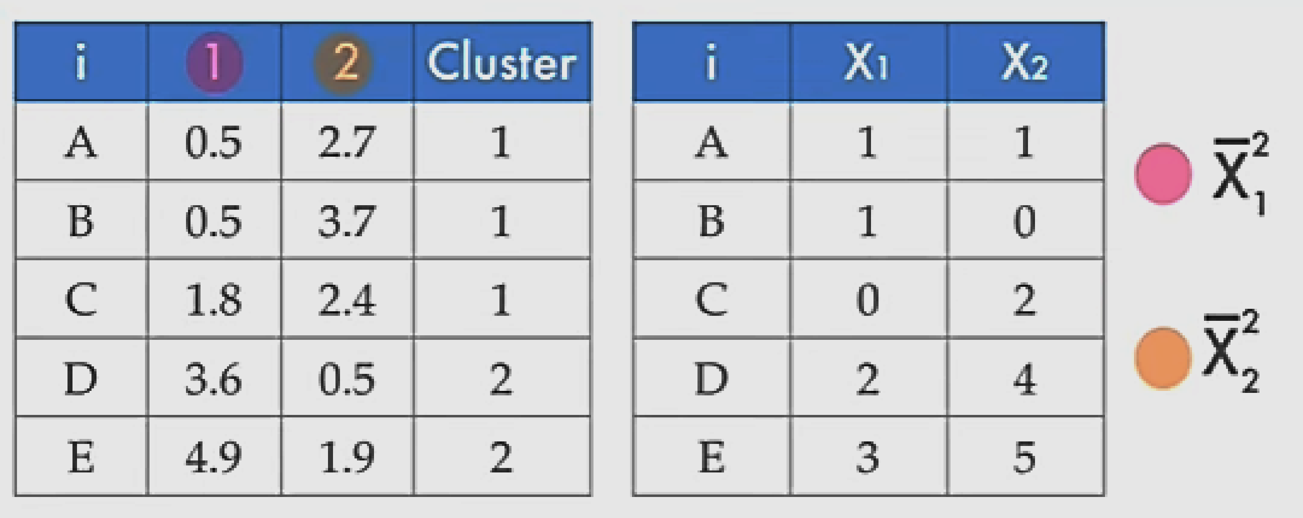
i	1	2
A	0.5	2.7
B	0.5	3.7
C	1.8	2.4
D	3.6	0.5
E	4.9	1.9

Step – 2.1

❖Therefore, the new clusters are: {A,B,C} and {D,E}



Assign each case to the cluster having the closest mean. Recalculate the cluster means.

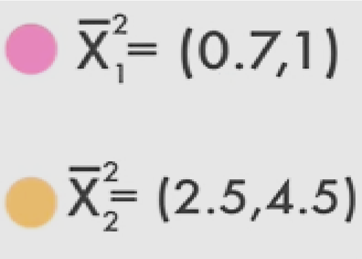


i	1	2	Cluster
A	0.5	2.7	1
B	0.5	3.7	1
C	1.8	2.4	1
D	3.6	0.5	2
E	4.9	1.9	2

i	X_1	X_2
A	1	1
B	1	0
C	0	2
D	2	4
E	3	5

● $\bar{X}_1^2$

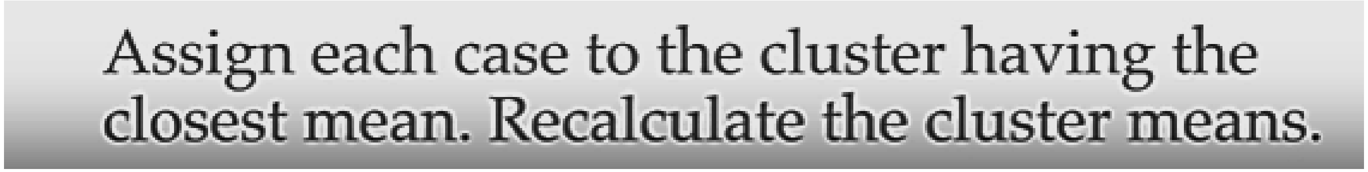
● $\bar{X}_2^2$



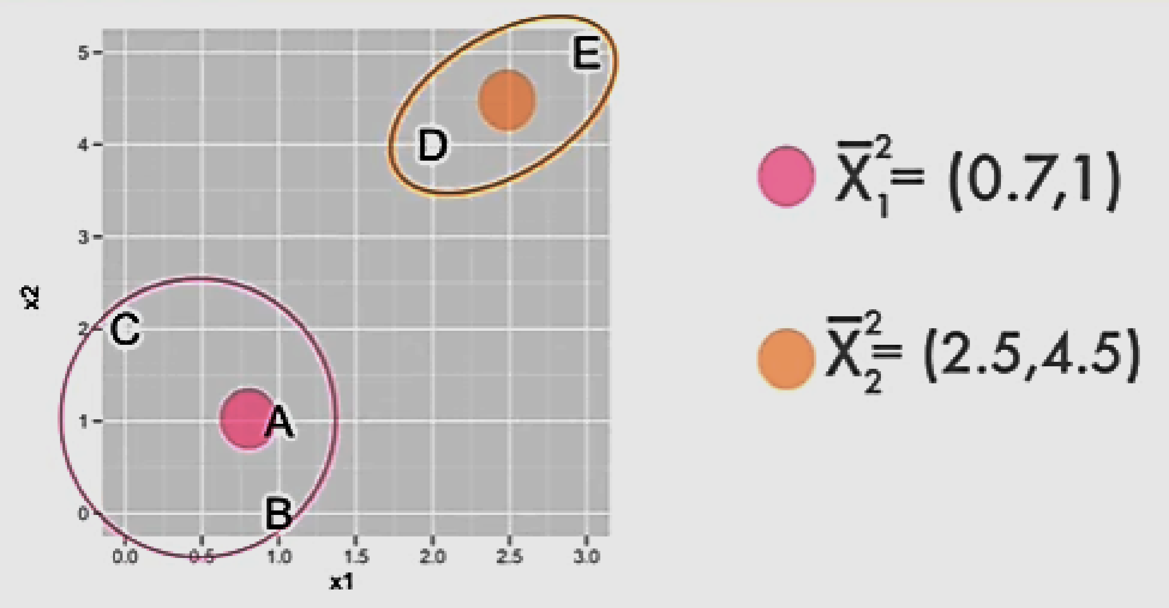
● $\bar{X}_1^2 = (0.7,1)$

● $\bar{X}_2^2 = (2.5,4.5)$

Step – 2.1 PLOTS



Assign each case to the cluster having the closest mean. Recalculate the cluster means.

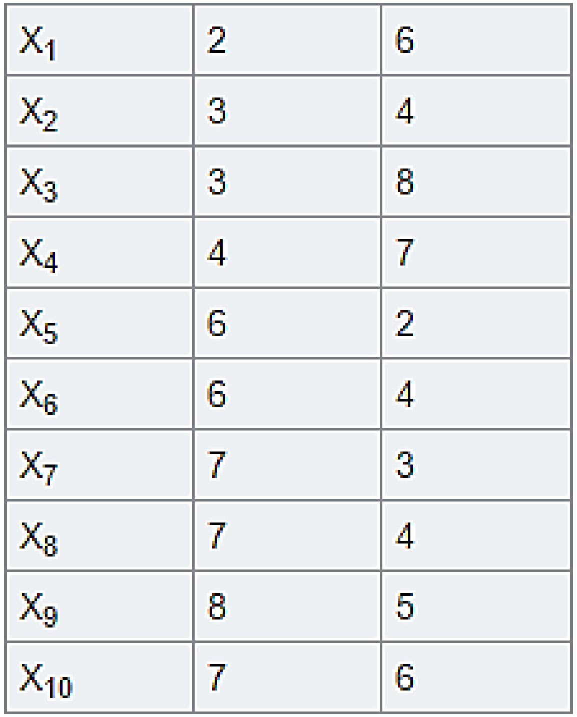


Step – 3

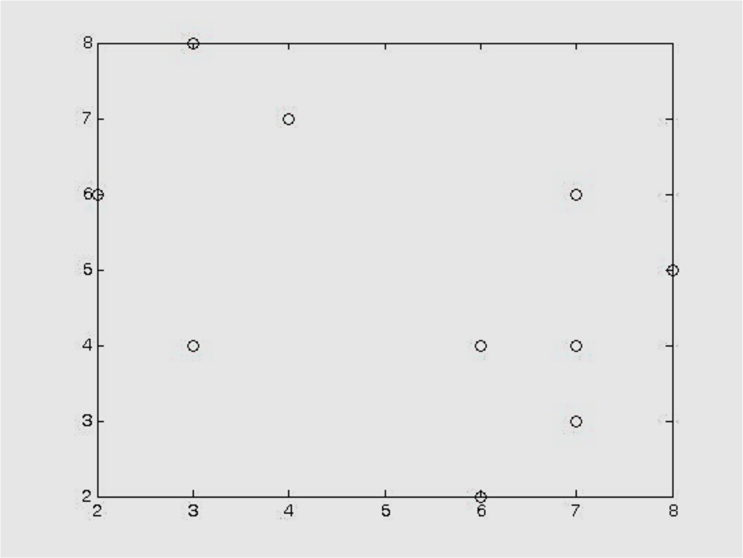
Algorithm has converged – recalculating distances, reassigning cases to clusters results in no change . This is the final solution .

Demonstration of PAM

- Cluster the following data set of ten objects into two clusters i.e. *k* = 2.
- Consider a data set of ten objects as follows :



X_1	2	6
X_2	3	4
X_3	3	8
X_4	4	7
X_5	6	2
X_6	6	4
X_7	7	3
X_8	7	4
X_9	8	5
X_{10}	7	6



Distribution of the data

Step 1:

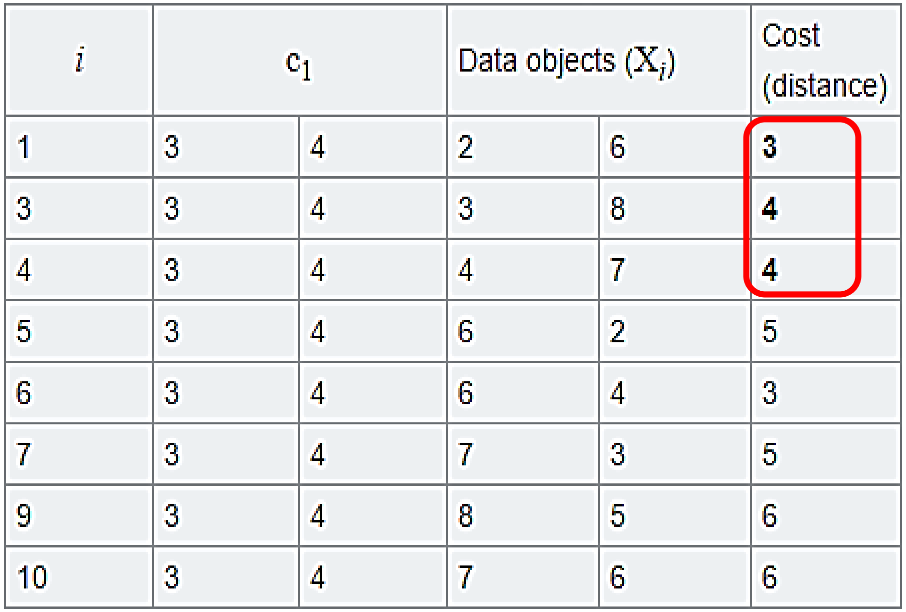
1. Initialize *k* centers.
2. Let us assume x_2 and x_8 are selected as **medoids**, so the centers are c_1 = (3,4) and c_2 = (7,4)
3. Calculate distances to each center so as to associate each data object to its nearest medoid.

- Cost is calculated using Manhattan distance (metric with *r* = 1).
- Costs to the nearest medoid are shown bold in the table.

$$\mathrm{cost}(x, c) = \sum_{i=1}^{d} |x_i - c_i|$$

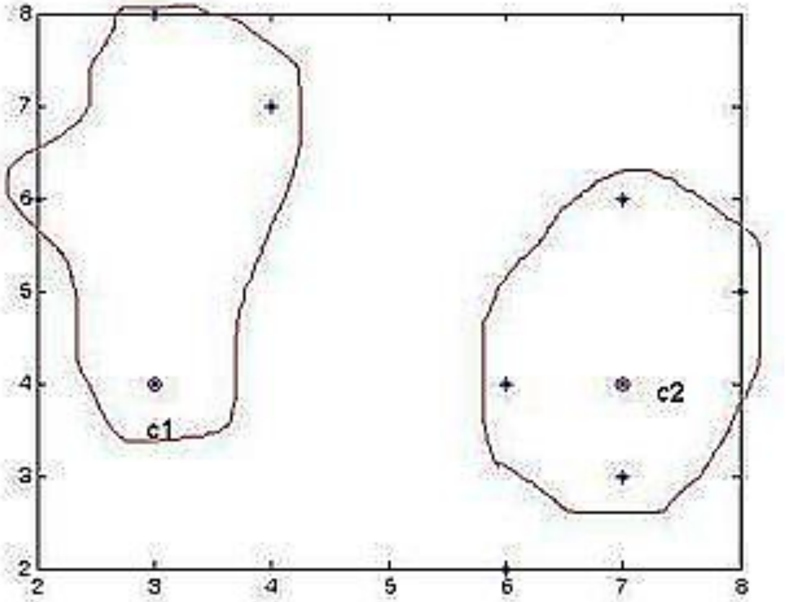
Step 1:

Cost (distance) to c_1



i	c_1		Data objects (X_i)		Cost (distance)
1	3	4	2	6	**3**
3	3	4	3	8	**4**
4	3	4	4	7	**4**
5	3	4	6	2	5
6	3	4	6	4	3
7	3	4	7	3	5
9	3	4	8	5	6
10	3	4	7	6	6

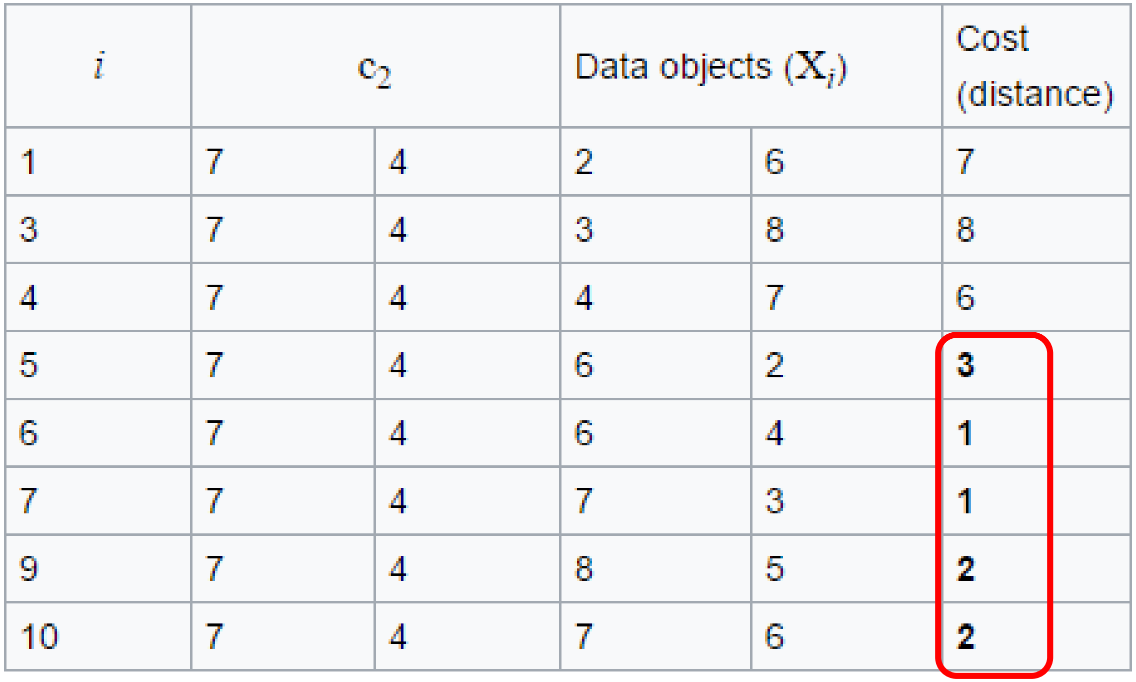
$$\text{cost}(x, c) = \sum_{i=1}^{d} |x_i - c_i|$$



clusters after step 1

Step 1:

Cost (distance) to c_2



i	c_2		Data objects (X_i)		Cost (distance)
1	7	4	2	6	7
3	7	4	3	8	8
4	7	4	4	7	6
5	7	4	6	2	**3**
6	7	4	6	4	**1**
7	7	4	7	3	**1**
9	7	4	8	5	**2**
10	7	4	7	6	**2**

Step 1:

Then the clusters become:

Cluster$_1$ = {(3,4)(2,6)(3,8)(4,7)}
Cluster$_2$ = {(7,4)(6,2)(6,4)(7,3)(8,5)(7,6)}

Since the points (2,6) (3,8) and (4,7) are closer to c$_1$ hence they form one cluster whilst remaining points form another cluster.

So the total cost involved is 20.

Step 1:

Where cost between any two points is found using formula

$$\text{cost}(x, c) = \sum_{i=1}^{d} |x_i - c_i|$$

where *x* is any data object, *c* is the medoid, and *d* is the dimension of the object which in this case is 2.

Total cost is the summation of the cost of data object from its medoid in its cluster so here:

$$\begin{aligned} \text{total cost} &= \{\text{cost}((3,4),(2,6)) + \text{cost}((3,4),(3,8)) + \text{cost}((3,4),(4,7))\} \\ &\quad + \{\text{cost}((7,4),(6,2)) + \text{cost}((7,4),(6,4)) + \text{cost}((7,4),(7,3)) \\ &\quad + \text{cost}((7,4),(8,5)) + \text{cost}((7,4),(7,6))\} \\ &= (3+4+4) + (3+1+1+2+2) \\ &= 20 \end{aligned}$$

Step 2:

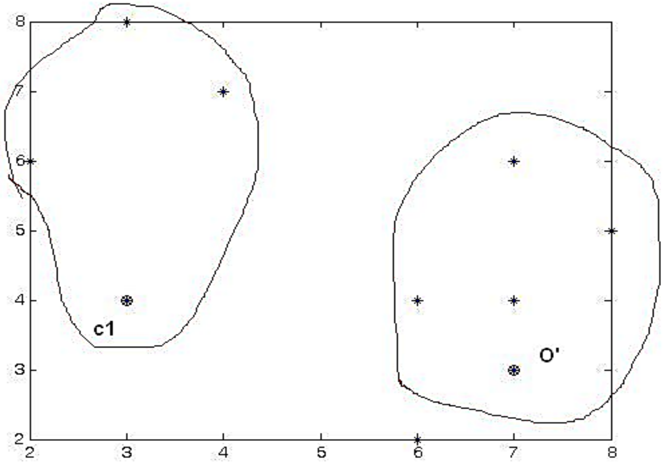
Select one of the nonmedoids O′

Let us assume O′ = (7,3), i.e. x_7.

So now the medoids are $c_1(3,4)$ and O′ (7,3)

If c1 and O′ are new medoids, calculate the total cost involved

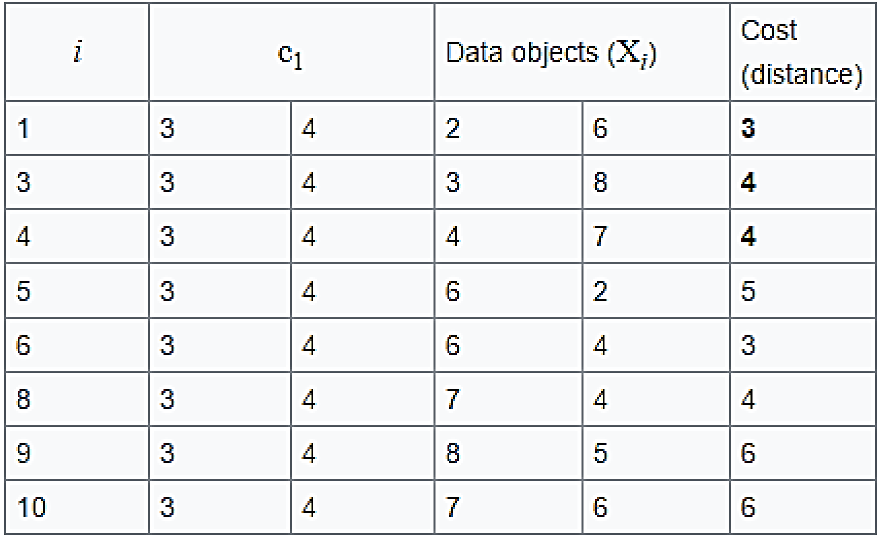
By using the formula in the step 1



clusters after step 2

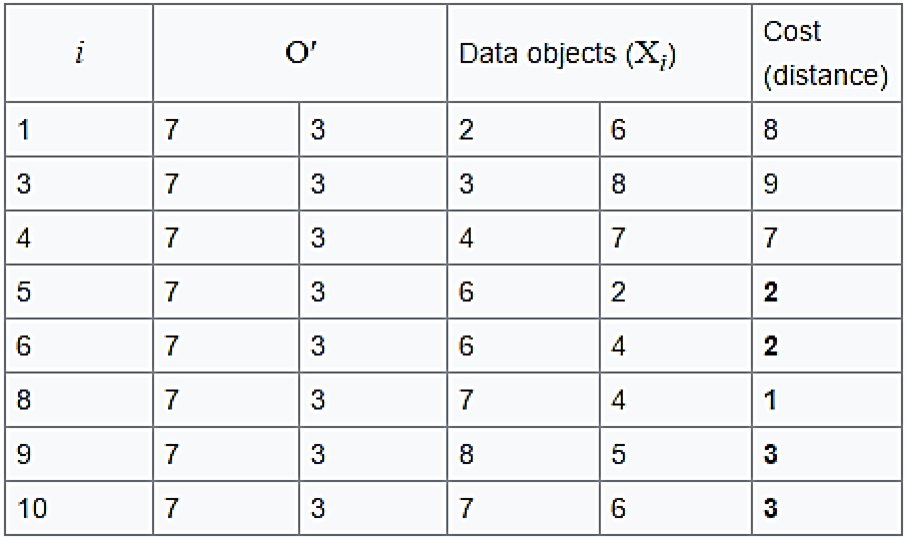
Step 2:

Cost (distance) to c_1



i	c_1		Data objects (X_i)		Cost (distance)
1	3	4	2	6	**3**
3	3	4	3	8	**4**
4	3	4	4	7	**4**
5	3	4	6	2	5
6	3	4	6	4	3
8	3	4	7	4	4
9	3	4	8	5	6
10	3	4	7	6	6

Cost (distance) to c_2



i	O'		Data objects (X_i)		Cost (distance)
1	7	3	2	6	8
3	7	3	3	8	9
4	7	3	4	7	7
5	7	3	6	2	**2**
6	7	3	6	4	**2**
8	7	3	7	4	**1**
9	7	3	8	5	**3**
10	7	3	7	6	**3**

Step 2:

$$\begin{aligned}\text{total cost} &= 3+4+4+2+2+1+3+3 \\ &= 22\end{aligned}$$

So cost of swapping medoid from c_2 to O' is

$$\begin{aligned}S &= \text{current total cost} - \text{past total cost} \\ &= 22 - 20 \\ &= 2 > 0.\end{aligned}$$

So moving to O′ would be a **bad idea**, so the **previous choice was good**. So we try other nonmedoids and found that our first choice was the best. So the configuration does not change and algorithm terminates here (i.e. there is no change in the medoids).

Thank you