k-medoids or PAM (Partition Around Medoids)

Mean Vs Median

Salary
10000
20000
15000
12000
18000
16000

Mean

$$\frac{(10000+20000+15000+12000+18000+16000)}{6} = 15166.67$$

Median

- Arrange in ascending order, take the middle element
- = (15000+16000)/2
- = 15500

Mean Vs Median

Salary
10000
20000
15000
12000
18000
16000
100000

Mean

$$\frac{(10000+20000+15000+12000+18000+16000+100000)}{7} = 27285.71$$

Median

- Arrange in ascending order, take the middle element
- = 16000

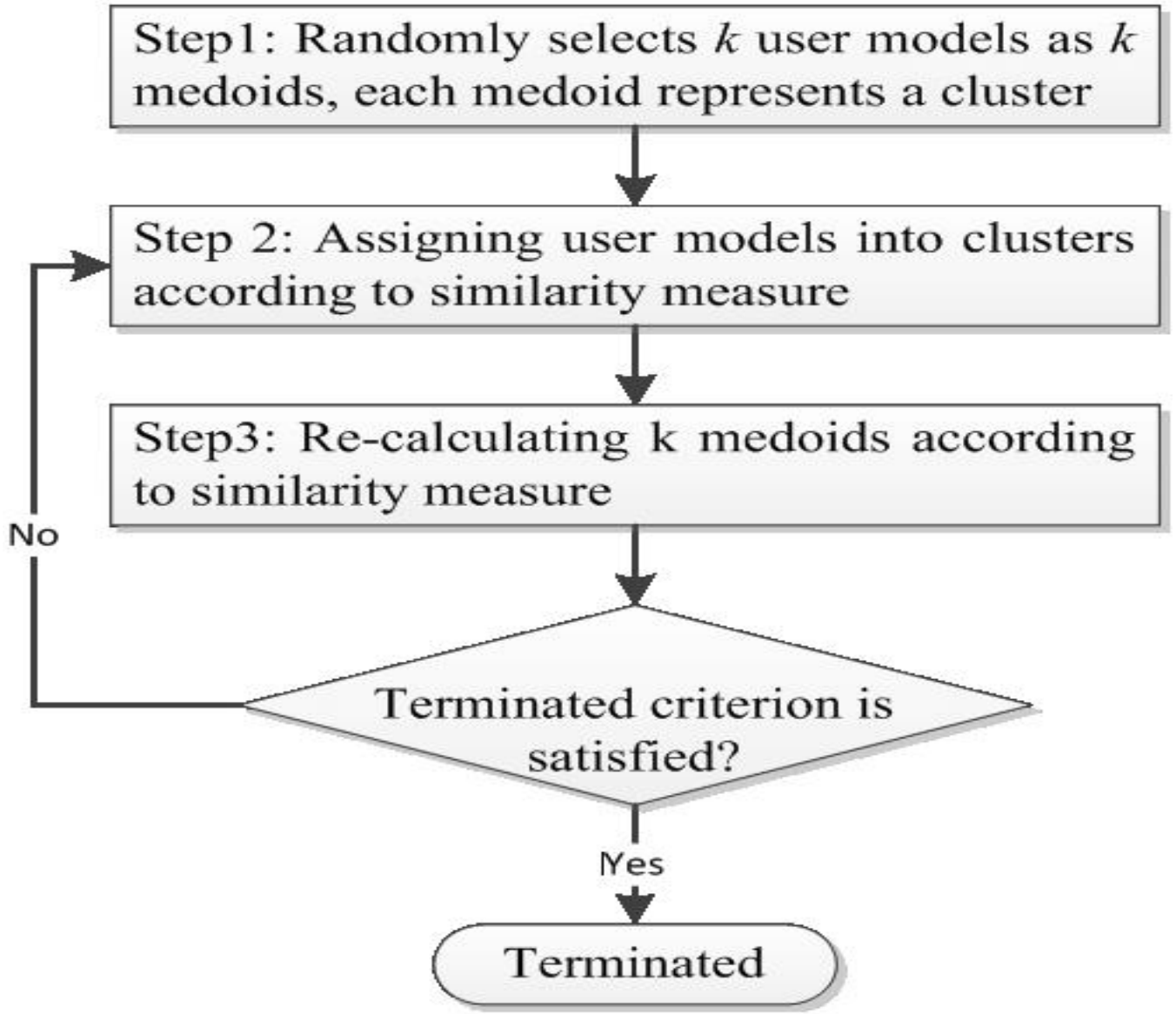
k-medoids or PAM (Partition around medoids)

- k-medoids or PAM (Partition around medoids) Each cluster is represented by one of the objects in the cluster
- The mean in k-means clustering is sensitive to outliers. Since an object with an extremely high value may substantially distort the distribution of data.
- Hence we move to k-medoids.
- Instead of taking mean of cluster we take the most centrally located point in cluster as it’s center.
- These are called medoids.

Contd…

- K-Medoids (also called as Partitioning Around Medoid) algorithm was proposed in 1987 by Kaufman and Rousseeuw.
- A medoid can be defined as the point in the cluster, whose dissimilarities with all the other points in the cluster is minimum.
- The dissimilarity of the medoid(Ci) and object(Pi) is calculated by using E = |Pi - Ci|
- The cost in K-Medoids algorithm is given as $$c = \sum_{Ci} \sum_{Pi \in Ci} |Pi - Ci|$$

K-medoids


Step1: Randomly selects k user models as k medoids, each medoid represents a cluster
Step 2: Assigning user models into clusters according to similarity measure
Step3: Re-calculating k medoids according to similarity measure
No
Terminated criterion is satisfied?
Yes
Terminated

K-medoids - Basic Algorithm

1. Initialize: select k random points out of the n data points as the medoids.

2. Associate each data point to the closest medoid by using any common distance metric methods.

While the cost decreases:

For each medoid m, for each data o point which is not a medoid:

a) Swap m and o, associate each data point to the closest medoid, recompute the cost.

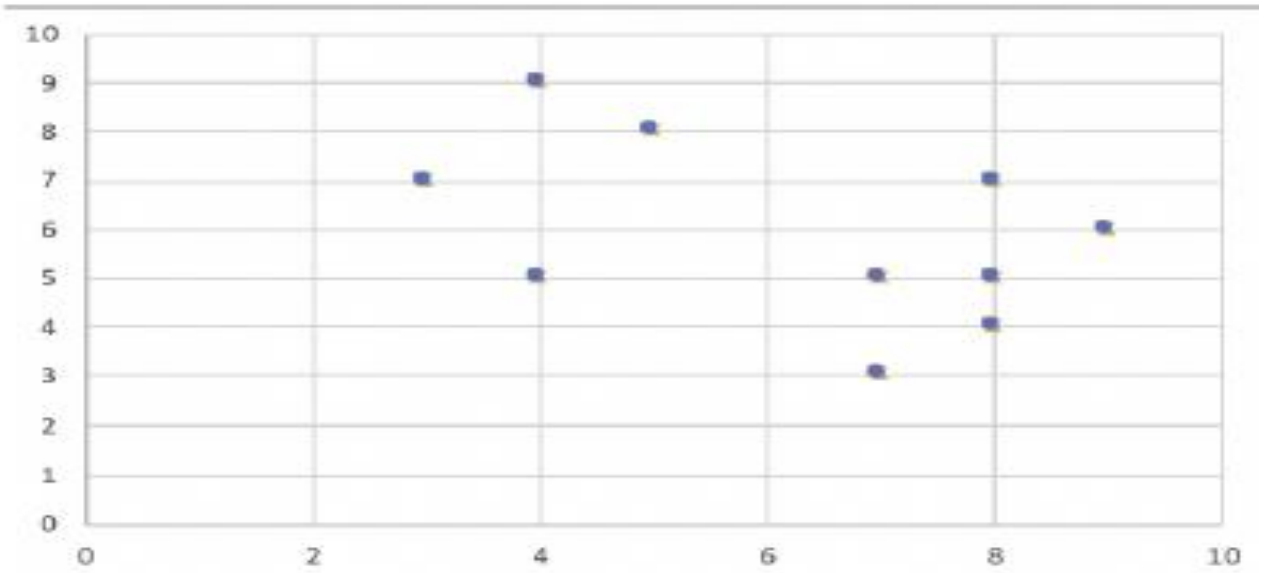
b) If the total cost is more than that in the previous step, undo the swap.

K = 2, Use Manhattan distance

$$Mdist = |x2 - x1| + |y2 - y1|$$

Points	X	Y
x1	8	7
x2	3	7
x3	4	9
x4	9	6
x5	8	5
x6	5	8
x7	7	3
x8	8	4
x9	7	5
x10	4	5

If a graph is drawn using the above data points, we obtain the following:



cluster 1= (x1, x4, x5, x7, x8, x9)
cluster 2= (x2, x3, x6, x10)

Step 1:

Let the randomly selected 2 medoids, so select k = 2 and let C2 -(4, 5) and C1 -(8, 5) are the two medoids.

Step 2: Calculating cost.

The dissimilarity of each non-medoid point with the medoids is calculated and tabulated:

Points	X	Y	C1 (8,5)	C2 (4,5)	Minimum Value	Clusters
x1	8	7	2	6	2	C1
x2	3	7	7	3	3	C2
x3	4	9	8	4	4	C2
x4	9	6	2	6	2	C1
x5	8	5	0	4	0	C1
x6	5	8	6	4	4	C2
x7	7	3	3	5	3	C1
x8	8	4	0	5	1	C1
x9	7	5	1	3	1	C1
x10	4	5	4	0	0	C2

Total cost = 20

cluster 1= (x1, x4, x5, x7, x8, x9)

cluster 2= (x2, x3, x6, x10)

Step 3: Randomly select one non-medoid point and recalculate the cost.

Let the randomly selected point be (8, 4).

The dissimilarity of each non-medoid point with the medoids – C1 (4, 5) and C2 (8, 4) is calculated and tabulated.

Points	X	Y	C1 (8,4)	C2 (4,5)	Minimum Value	Clusters
x1	8	7	3	6	3	C1
x2	3	7	8	3	3	C2
x3	4	9	9	4	4	C2
x4	9	6	3	6	3	C1
x5	8	5	1	4	1	C1
x6	5	8	7	4	4	C2
x7	7	3	2	5	2	C1
x8	8	4	0	5	0	C1
x9	7	5	2	3	2	C1
x10	4	5	5	0	0	C2

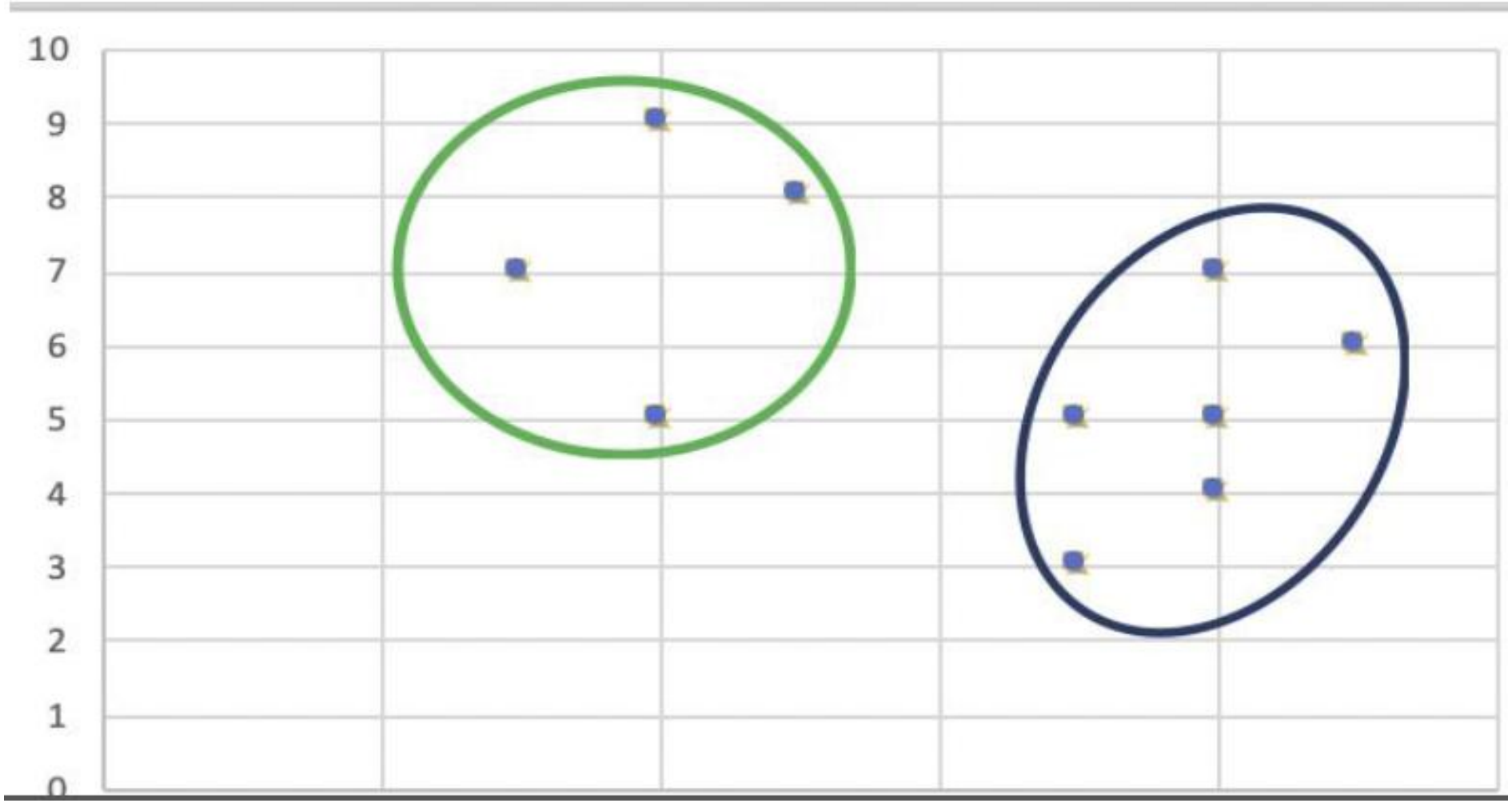
cluster 1= (x1, x4, x5, x7, x8, x9)

cluster 2= (x2, x3, x6, x10)

Total cost = 22

Swap Cost = New Cost – Previous Cost = **22 – 20** and 2 >0

As the swap cost is not less than zero, we undo the swap. Hence (3, 4) and (7, 4) are the final medoids. The clustering would be in the following way



Advantages

- It is simple to understand and easy to implement.
- K-Medoid Algorithm is fast and converges in a fixed number of steps.
- PAM is less sensitive to outliers than other partitioning algorithms

Disadvantages

- The main disadvantage of K-Medoid algorithms is that it is not suitable for clustering non-spherical (arbitrary shaped) groups of objects. This is because it relies on minimizing the distances between the non-medoid objects and the medoid (the cluster centre) – briefly, it uses compactness as clustering criteria instead of connectivity.
- 2.It may obtain different results for different runs on the same dataset because the first k medoids are chosen randomly.