Cluster Analysis

- Partitioning Clustering
  - K-Means Clustering
  - K-Medoid Clustering

1

Partitioning Clustering

- Partitioning method: Construct a partition of *n* documents into a set of *K* clusters
- Given: a set of documents and the number *K*
- Find: a partition of *K* clusters that optimizes the chosen partitioning criterion
  - 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

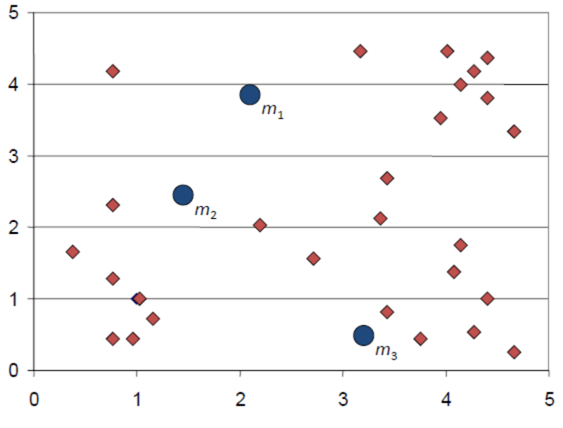
2

K-Means Algorithm

- The *k*-means algorithm partitions the given data into *k* clusters.
  - Each cluster has a cluster **center**, called **centroid**.
  - *k* is specified by the user
- Given *k*, the *k-means* algorithm consists of four steps:
  - Select initial centroids at random.
  - Assign each object to the cluster with the nearest centroid.
  - Compute each centroid as the mean of the objects assigned to it.
  - Repeat previous 2 steps until no change.

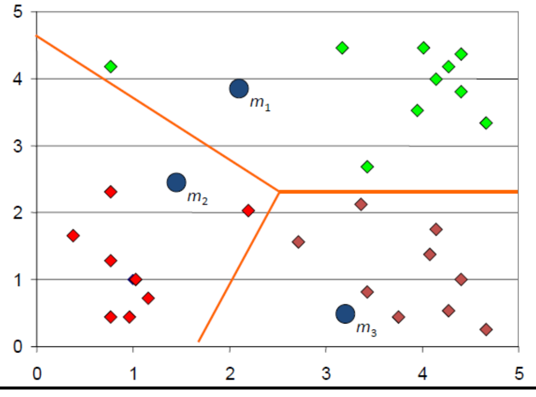
K-Means Algorithm

- **Initilization:** Determine the k cluster centers.



K-Means Algorithm

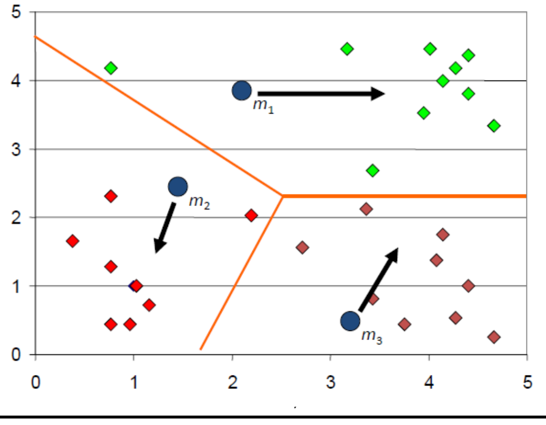
- **Cluster Assignment:** Assign each object to the cluster which has the closet distance from the centroid to the object.



5

K-Means Algorithm

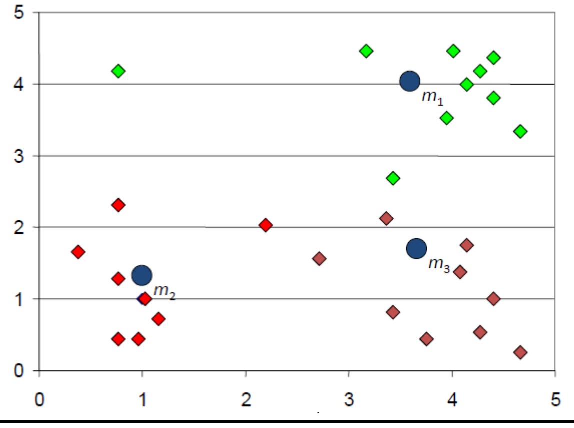
- **Update Cluster Centroid:** Compute cluster centroid as the center of the points in the cluster.



6

K-Means Algorithm

- **Update Cluster Centroid:** Compute cluster centroid as the center of the points in the cluster.



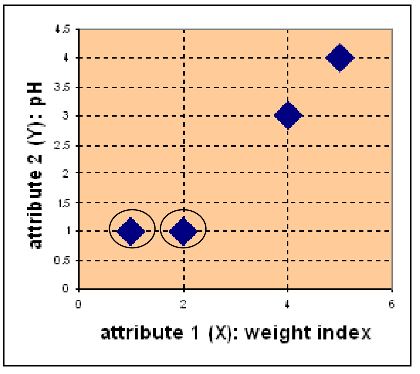
K-Means Algorithm

- **Example:** Assume that there are 4 objects in the data set and each object has 2 features.

Object	Feature 1 (X)	Feature 2 (Y)
Medicine A	1	1
Medicine B	2	1
Medicine C	4	3
Medicine D	5	4

K-Means Algorithm

- **Example:** Assume that there are 4 objects in the data set and each object has 2 features.



K-Means Algorithm

- Centroid coordinates are C1 = (1,1) and C2 = (2,1).

Medicine A (1,1)

C_1: $\sqrt{(1-1)^2+(1-1)^2} = 0$

C_2: $\sqrt{(2-1)^2+(1-1)^2} = 1$

Medicine B (2,1)

C_1: $\sqrt{(1-2)^2+(1-1)^2} = 1$

C_2: $\sqrt{(2-2)^2+(1-1)^2} = 0$

Medicine C (4,3)

C_1: $\sqrt{(1-4)^2+(1-3)^2} = 3.6$

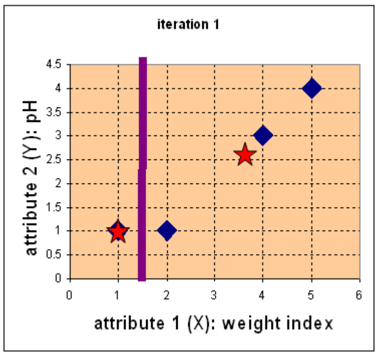
C_2: $\sqrt{(2-4)^2+(1-3)^2} = 2.8$

Medicine D (5,4)

C_1: $\sqrt{(1-5)^2+(1-4)^2} = 5$

C_2: $\sqrt{(2-5)^2+(1-4)^2} = 4.24$

K-Means Algorithm



$$C_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3}\right) = (3.67, 2.67)$$

K-Means Algorithm

- Centroid coordinates are C1 = (1,1) and C2 = (3.67,2.67).

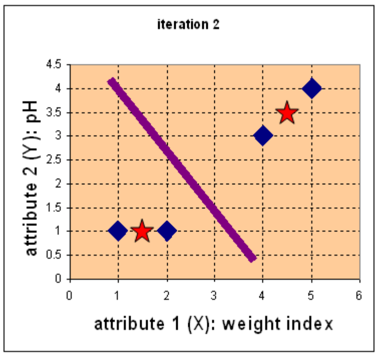
Medicine A (1,1)
- C_1: $\sqrt{(1-1)^2+(1-1)^2} = 0$
- C_2: $\sqrt{(3.67-1)^2+(2.67-1)^2} = 3.14$

Medicine B (2,1)
- C_1: $\sqrt{(1-2)^2+(1-1)^2} = 1$
- C_2: $\sqrt{(3.67-2)^2+(2.67-1)^2} = 2.36$

Medicine C (4,3)
- C_1: $\sqrt{(1-4)^2+(1-3)^2} = 3.6$
- C_2: $\sqrt{(3.67-4)^2+(2.67-3)^2} = 0.47$

Medicine D (5,4)
- C_1: $\sqrt{(1-5)^2+(1-4)^2} = 5$
- C_2: $\sqrt{(3.67-5)^2+(2.67-4)^2} = 1.89$

K-Means Algorithm



$$C_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = (1.5,1)$$

$$C_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = (4.5,3.5)$$

13

K-Means Algorithm

- Centroid coordinates are C1 = (1.5,1) and C2 = (4.5,3.5).

Medicine A (1,1)
- C_1: $\sqrt{(1.5-1)^2+(1-1)^2} = 0.5$
- C_2: $\sqrt{(4.5-1)^2+(3.5-1)^2} = 4.3$

Medicine B (2,1)
- C_1: $\sqrt{(1.5-2)^2+(1-1)^2} = 0.5$
- C_2: $\sqrt{(4.5-2)^2+(3.5-1)^2} = 3.54$

Medicine C (4,3)
- C_1: $\sqrt{(1.5-4)^2+(1-3)^2} = 3.20$
- C_2: $\sqrt{(4.5-4)^2+(3.5-3)^2} = 0.71$

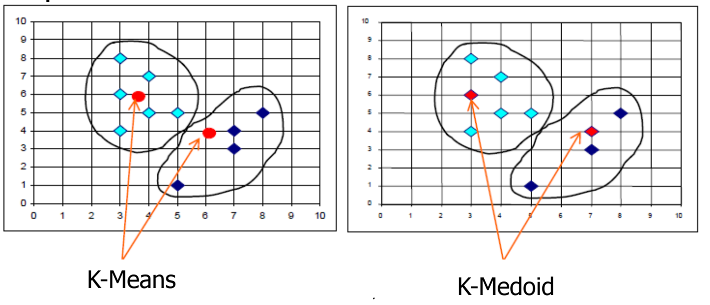
Medicine D (5,4)
- C_1: $\sqrt{(1.5-5)^2+(1-4)^2} = 4.61$
- C_2: $\sqrt{(4.5-5)^2+(3.5-4)^2} = 0.71$

14

K-Medoid Algorithm

Difference between K-means and K-medoids:

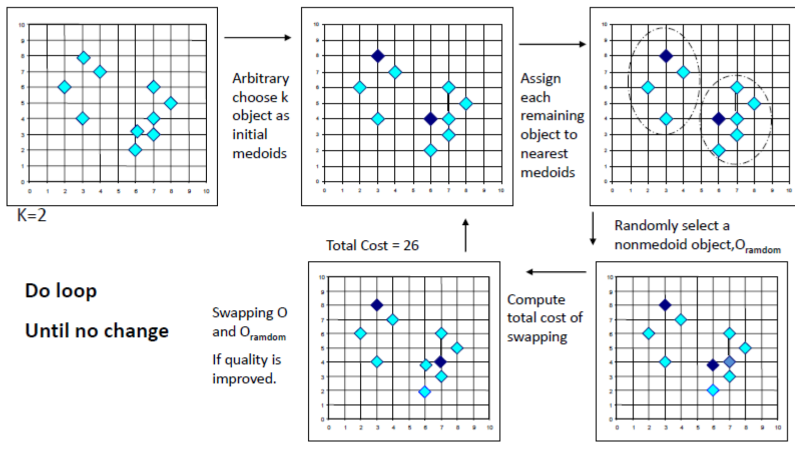
- K-means: Cluster centers may not be the original data point.
- K-medoids: Each cluster's centroid is represented by a point called **medoid** in the cluster.



K-Medoid Algorithm

- Arbitrarily choose k objects as the initial medoids.
- Associate each data point to the closest medoid.
- While the cost of the configuration decreases:
  - For each medoid *m*, for each non-medoid data point *o*:
    - Swap *m* and *o*, recompute the cost (sum of distances of points to their medoid)
    - If the total cost of the configuration increased in the previous step, undo the swap

K-Medoid Algorithm



17

K-Medoid Algorithm

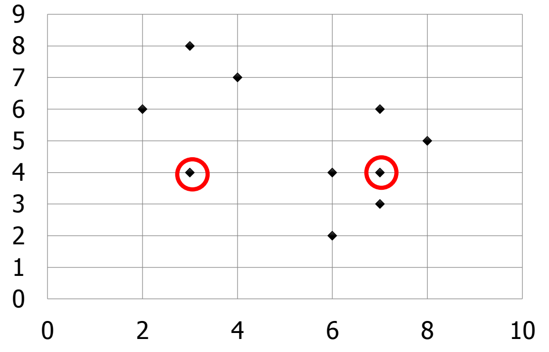
- **Example:** Assume that there are 10 objects in the data set and each object has 2 features.

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

18

K-Medoid Algorithm

- **Example:** Assume that there are 10 objects in the data set and each object has 2 features.



K-Medoid Algorithm

Data object		Distance to	
i	X_i	$c_1 = (3,4)$	$c_2 = (7,4)$
1	(2, 6)	**3**	7
2	(3, 4)	**0**	4
3	(3, 8)	**4**	8
4	(4, 7)	**4**	6
5	(6, 2)	5	**3**
6	(6, 4)	3	**1**
7	(7, 3)	5	**1**
8	(7, 4)	4	**0**
9	(8, 5)	6	**2**
10	(7, 6)	6	**2**
Cost		**11**	**9**

- Cluster_1 : {(3,4)(2,6)(3,8)(4,7)}

- Cluster_2 : {(7,4)(6,2)(6,4)(7,3)(8,5)(7,6)}

K-Medoid Algorithm

- The total cost of this clustering is the sum of the distance between a data point and its cluster center:

$$\underbrace{3+0+4+4}_{\text{Cluster 1}} + \underbrace{3+1+1+0+2+2}_{\text{Cluster 2}} = 20$$

K-Medoid Algorithm

- Select one of the nonmedoids O′
- Let us assume O′ = (7,3).
- 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

K-Medoid Algorithm

Data Object		Distance To	
i	Xi	c1	Q
1	(2,6)	**3**	8
2	(3,4)	**0**	5
3	(3,8)	**4**	9
4	(4,7)	**4**	7
5	(6,2)	5	**2**
6	(6,4)	3	**2**
7	(7,3)	5	**0**
8	(7,4)	4	**1**
9	(8,5)	6	**3**
10	(7,6)	6	**3**
Total Cost		**11**	**11**

Total Cost = 22 > 20

23