Machine learning

Prepared by : Dr. Hanaa Bayomi
Updated By: Prof Abeer ElKorany

Lecture 10: Clustering Part1 (K means)

CLUSTERING

- Cluster Analysis is like Classification, but the class label of each object is not known.

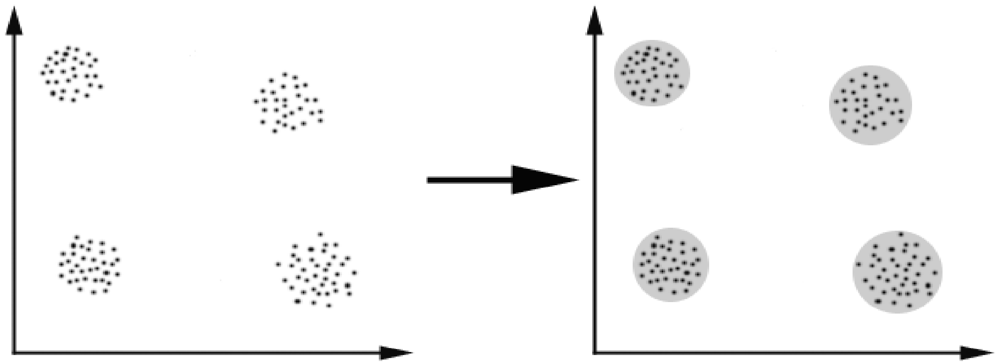
- Clustering can be considered the most *important unsupervised learning problem*; so, as every other problem of this kind, it deals with *finding a structure in a collection of unlabeled data.*

- **Cluster is** a subset of data which are similar

- **Clustering** is the *process of grouping the data into classes or clusters* so that objects within a cluster have high similarity in comparison to one another, but are very dissimilar to objects in other clusters.

SIMPLE GRAPHICAL EXAMPLE:

- In this case we easily identify the 4 clusters into which the data can be divided; the similarity criterion is *distance*: two or more objects belong to the same cluster if they are "close" according to a given distance. This is called *distance-based clustering*.



Distance functions

Euclidean $$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

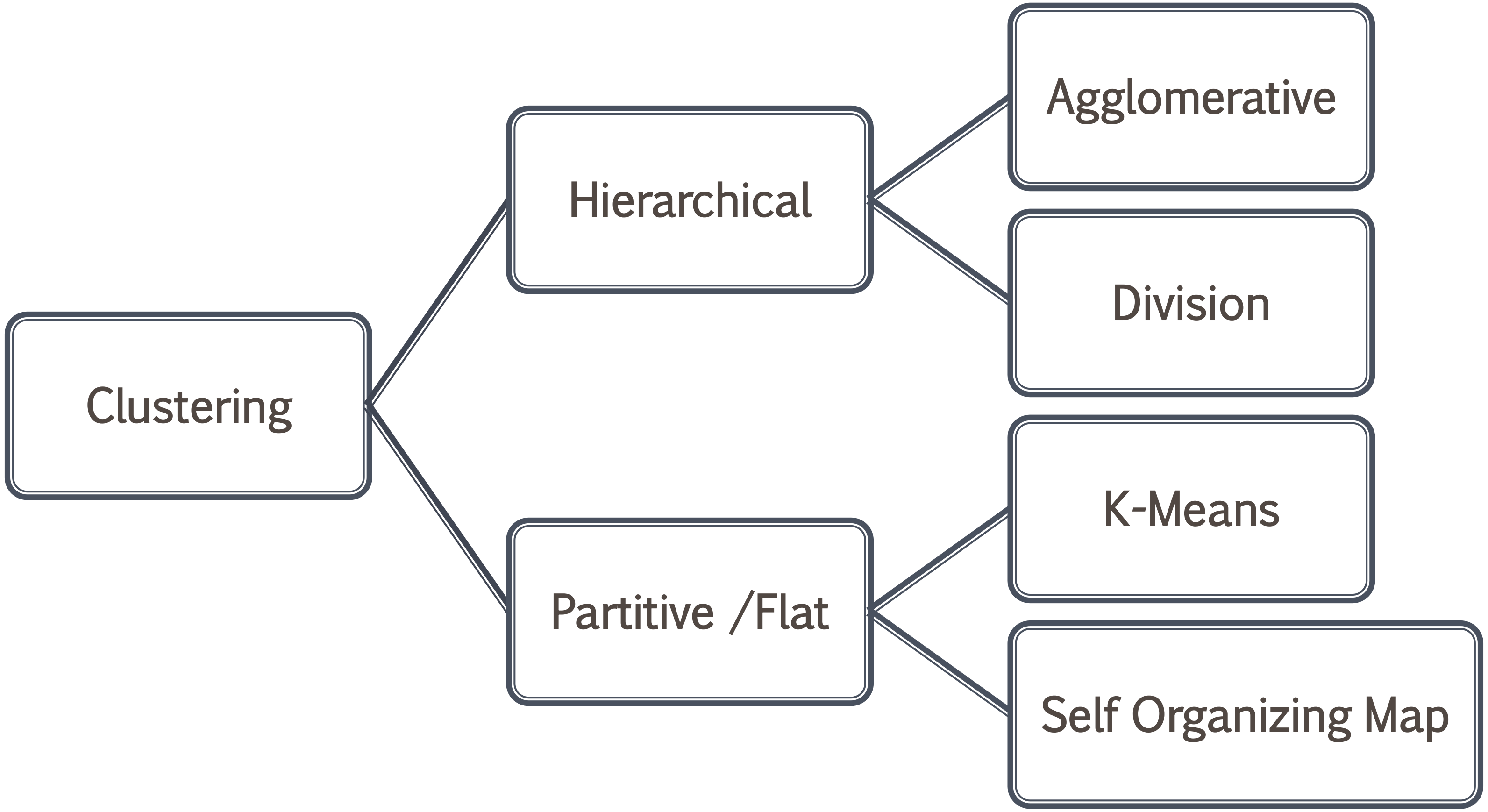
Manhattan $$\sum_{i=1}^{k}|x_i - y_i|$$

Minkowski $$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$$

APPLICATIONS OF CLUSTERING

- Marketing: finding groups of customers with similar behavior given a large database of customer data containing their properties and past buying records;
- Biology: classification of plants and animals given their features;
- Libraries: book ordering;
- Insurance: identifying groups of motor insurance policy holders with a high average claim cost; identifying frauds;
- City-planning: identifying groups of houses according to their house type, value and geographical location;
- Earthquake studies: clustering observed earthquake epicenters to identify dangerous zones;
- WWW: document classification; clustering weblog data to discover groups of similar access patterns.

Two main groups of clustering algorithms



Clustering Algorithms

- Partition/Flat algorithms
  - Usually start with a random (partial) partitioning
  - Refine it iteratively
    - *K* means clustering
    - (Model based clustering)
- Hierarchical algorithms
  - Bottom-up, agglomerative
  - Top-down, divisive

Hierarchical methods

Hierarchical methods again come in two varieties, **agglomerative** and **divisive.**

Agglomerative methods:

• Start with partition P_n, where each object forms its own cluster.

• Merge the two closest clusters, obtaining P_{n-1}.

• Repeat merge until only one cluster is left.

Divisive methods

• Start with P_1.

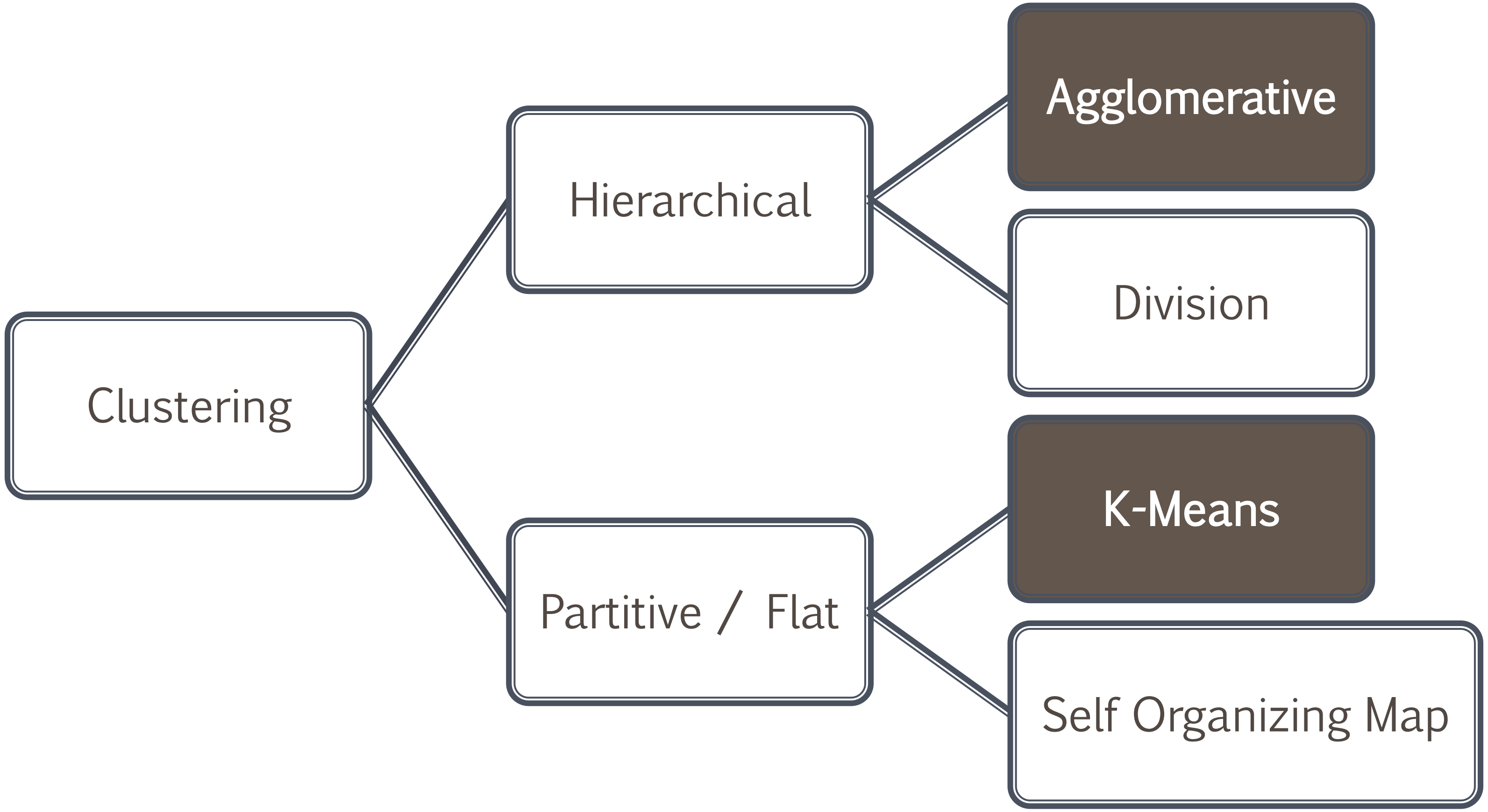
• Split the collection into two clusters that are as homogenous (and as different from each other) as possible.

• Apply splitting procedure recursively to the clusters.

Partitioning Algorithms

- Flat methods generate a single partition into k clusters. The number k of clusters has to be determined by the user ahead of time.
- Partitioning method: Construct a partition of n instances 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

Two main groups of clustering algorithms



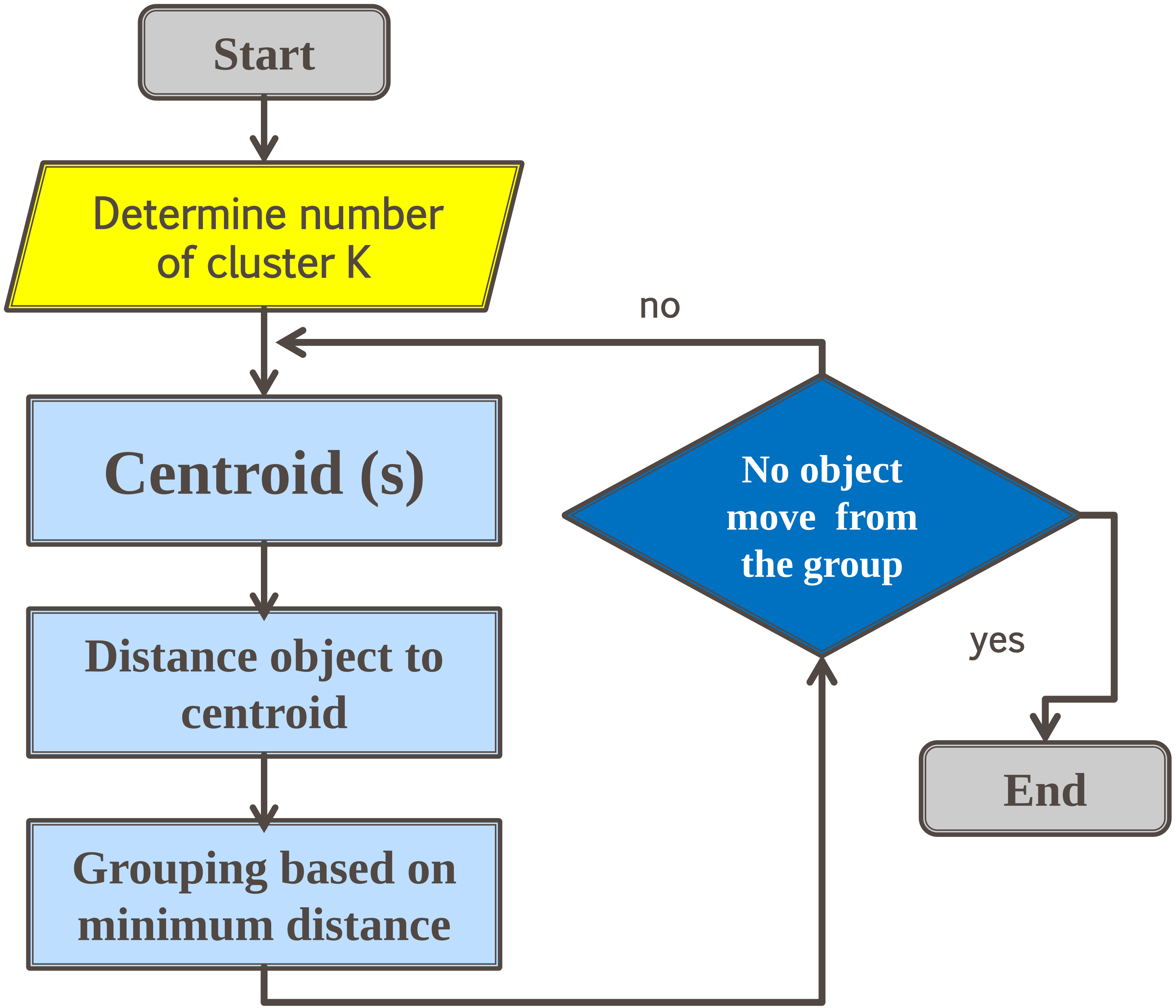
K-MEANS CLUSTERING

- Intends to partition n objects into k clusters in which *each object belongs to the cluster with the nearest mean*
- This method produces exactly k different clusters of greatest possible distinction
- The best number of clusters k leading to the greatest separation (distance) is not known as a priori and must be computed from the data

K-means Clustering algorithm

- Partitional clustering approach
- Each cluster is associated with a centroid (center point)
- Each point is assigned to the cluster with the closest centroid
- Number of clusters, K, must be specified
- The basic algorithm is very simple

```
Select K points as the initial centroids.
repeat
    Form K clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```


Start
Determine number of cluster K
Centroid (s)
Distance object to centroid
Grouping based on minimum distance
No object move from the group
no
yes
End

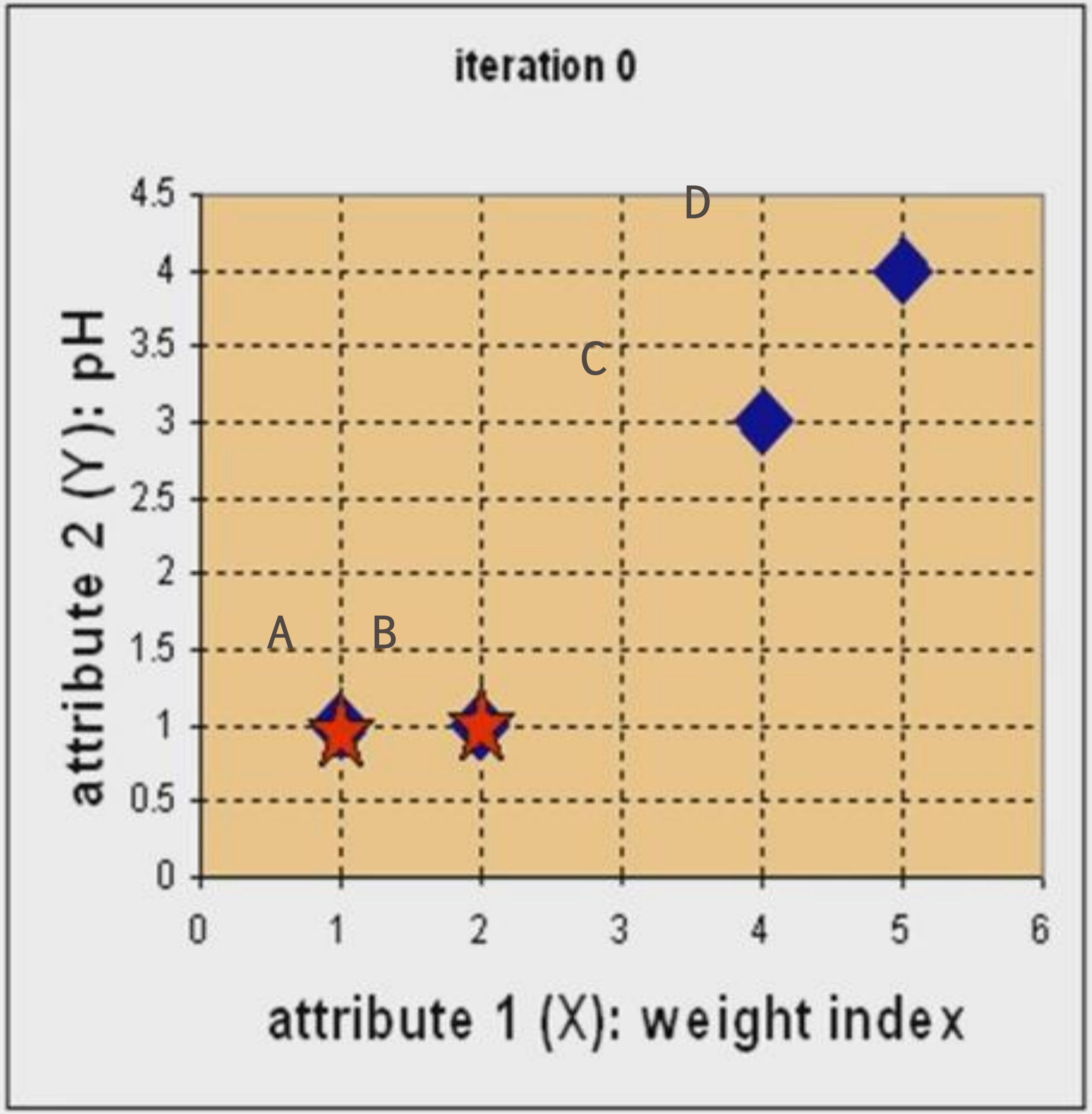
Real-Life Numerical Example of K-Means Clustering

We have 4 medicines as our training data points object and each medicine has 2 attributes. Each attribute represents coordinate of the object. We have to determine which medicines belong to cluster 1 and which medicines belong to the other cluster.

Object	Attribute1 (X): weight index	Attribute 2 (Y): pH
Medicine A	1	1
Medicine B	2	1
Medicine C	4	3
Medicine D	5	4

Step 1:

- **Initial value of centroids** : Suppose we use medicine A and medicine B as the first centroids.
- Let and c_1 and c_2 denote the coordinate of the centroids, then $c_1=(1,1)$ and $c_2=(2,1)$



- **Object Centroid distance:** calculate the distance between each cluster centroid and each point using Euclidean distance

$$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$$

$$\begin{array}{c} \\ x \\ y \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \end{array}$$

$$D^0 = \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix} \end{array} \begin{array}{l} \\ C1=(1,1) \\ C2=(2,1) \end{array}$$

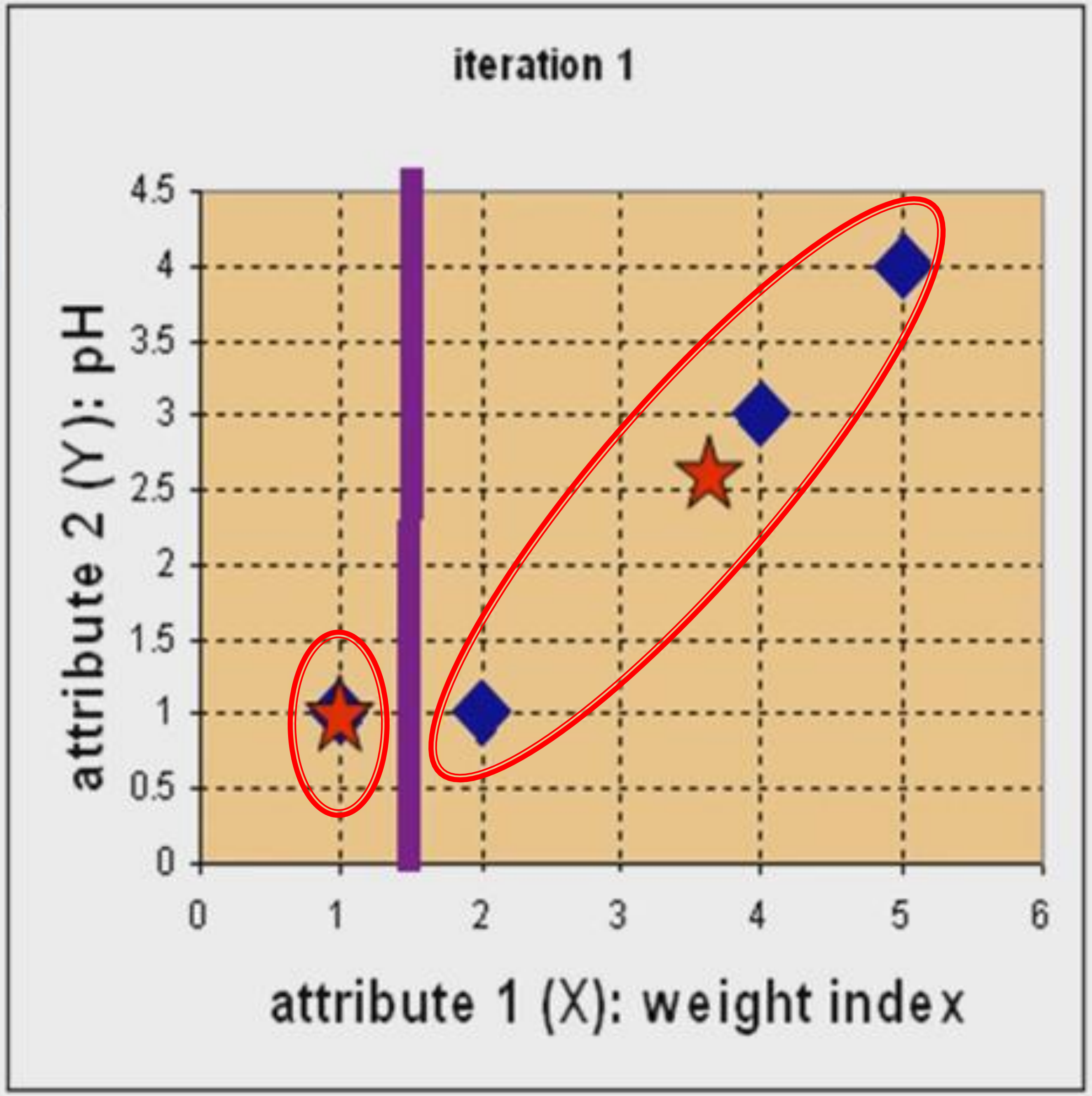
Minimum distance matrix

For example, distance from medicine C = (4, 3) to the first centroid $c_1=(1,1)$ is , $\sqrt{(4-1)^2+(3-1)^2}=3.61$ and its distance to the second centroid is , $c_2=(2,1)$ is $\sqrt{(4-2)^2+(3-1)^2}=2.83$ etc.

Step 2:

- **Objects clustering** : We assign each object based on the minimum distance.
- Medicine A is assigned to group 1, medicine B to group 2, medicine C to group 2 and medicine D to group 2.
- The elements of Group matrix below is 1 if and only if the object is assigned to that group.

$$\mathbf{G}^0 = \begin{array}{cc} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} & \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} & \end{array}$$



- **<u>Iteration-1, Objects-Centroids distances</u>** : The next step is to compute the distance of all objects to the new centroids.
- Similar to step 2, we have distance matrix at iteration 1 is

$$\mathbf{D}^1 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \end{bmatrix} \quad \begin{matrix} \mathbf{c}_1 = (1,1) & group-1 \\ \mathbf{c}_2 = (\frac{11}{3}, \frac{8}{3}) & group-2 \end{matrix}$$

$$\begin{array}{cc} & \begin{matrix} A & B & C & D \end{matrix} \\ \begin{matrix} x \\ y \end{matrix} & \begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \end{array}$$

$$c_2\ x = \frac{2+4+5}{3} = \frac{11}{3}$$

$$c_2\ y = \frac{1+3+4}{3} = \frac{8}{3}$$

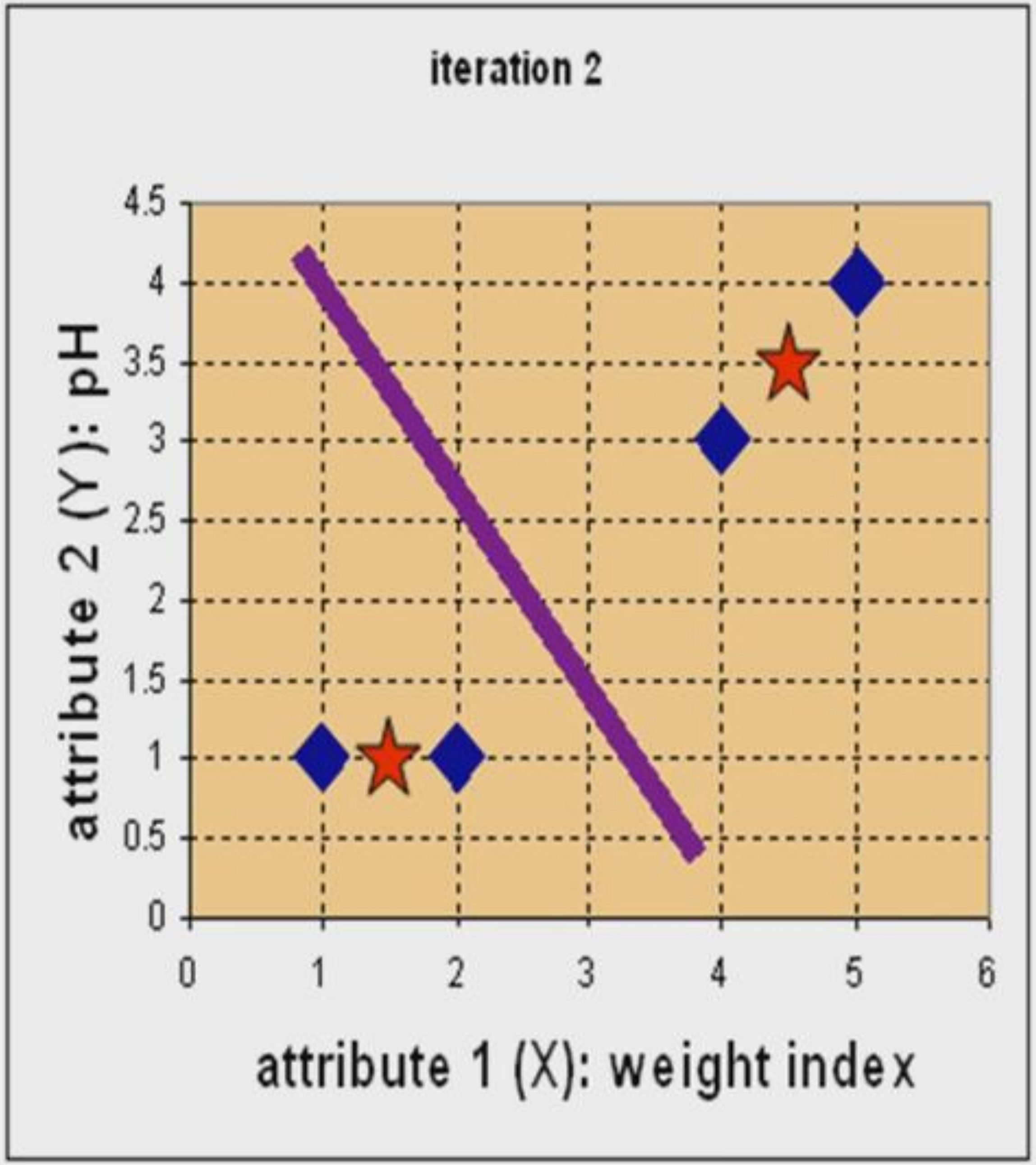
- **Iteration-1, Objects clustering:** Based on the new distance matrix, we move the medicine B to Group 1 while all the other objects remain. The Group matrix is shown below

$$\mathbf{G}^1 = \begin{array}{c} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} \end{array}$$

Compare

$$\mathbf{G}^0 = \begin{array}{c} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} \end{array}$$

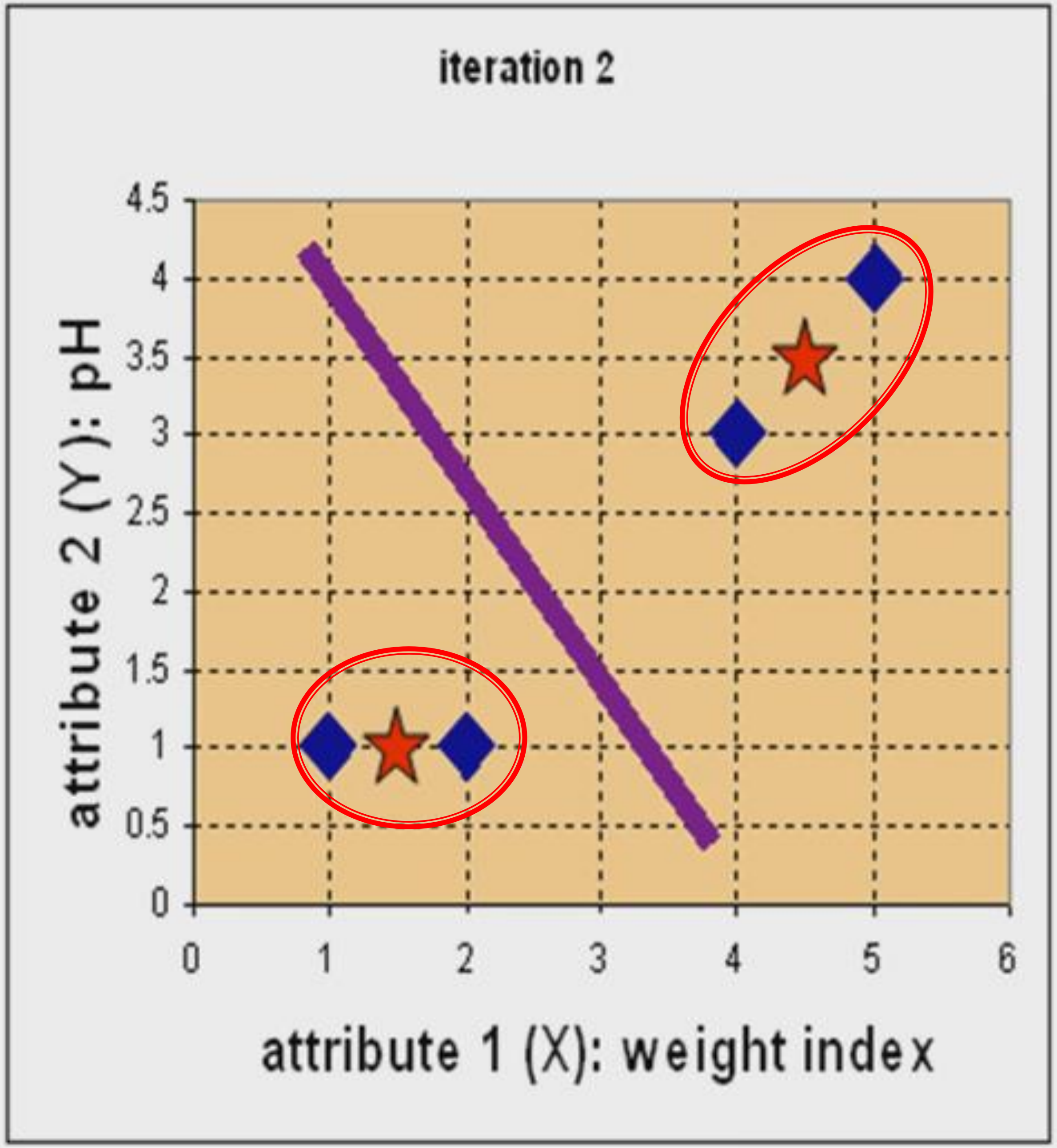
$$\mathbf{G}^1 = \begin{array}{c} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} \end{array}$$



- **Iteration-1, Objects clustering:** Based on the new distance matrix, we move the medicine B to Group 1 while all the other objects remain. The Group matrix is shown below

$$\mathbf{G}^1 = \begin{array}{c} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} \end{array}$$

- **Iteration 2, determine centroids:** Now we repeat step 4 to calculate the new centroids coordinate based on the clustering of previous iteration. Group1 and group 2 both has two members, thus the new centroids are $\mathbf{c}_1 = (\frac{1+2}{2}, \frac{1+1}{2}) = (1\frac{1}{2}, 1)$ and $\mathbf{c}_2 = (\frac{4+5}{2}, \frac{3+4}{2}) = (4\frac{1}{2}, 3\frac{1}{2})$



- **Iteration-2:Object Centroid distance:** calculate the distance between each cluster centroid and each point

$$\begin{array}{c c} & \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{array}{c} x \\ y \end{array} & \begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \end{array}$$

$$\begin{array}{c c} & \begin{array}{cccc} A & B & C & D \end{array} \\ D^2 = & \begin{bmatrix} 0.5 & 0.5 & 3.2 & 4.66 \\ 4.3 & 3.54 & 0.71 & 0.71 \end{bmatrix} \end{array} \quad \begin{array}{l} C1=(1.5,1) \\ C2=(4.5,3.5) \end{array}$$

Minimum distance matrix

- **Iteration-2, Objects clustering:** Again, we assign each object based on the minimum distance.

$$\mathbf{G}^2 = \begin{array}{c c} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} & \begin{array}{l} group-1 \\ group-2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} & \end{array}$$

Compare

$$\mathbf{G}^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array}$$

$$\begin{array}{cccc} A & B & C & D \end{array}$$

$$\mathbf{G}^2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group-1 \\ group-2 \end{array}$$

$$\begin{array}{cccc} A & B & C & D \end{array}$$

- We obtain result that $\mathbf{G}^2 = \mathbf{G}^1$ Comparing the grouping of last iteration and this iteration reveals that the objects does not move group anymore.
- Thus, the computation of the k-mean clustering has reached its stability and no more iteration is needed..

We get the final grouping as the results as:

Object	Feature1(X): weight index	Feature2 (Y): pH	Group (result)
Medicine A	1	1	1
Medicine B	2	1	1
Medicine C	4	3	2
Medicine D	5	4	2

K-means Clustering – Details

- Initial centroids are often chosen randomly.
  - Clusters produced vary from one run to another.
- The centroid is (typically) the mean of the points in the cluster.
- ‘Closeness’ is measured by Euclidean distance, cosine similarity, correlation, etc.
- K-means will converge for common similarity measures mentioned above.

Seed Choice

- Results can vary based on random seed selection.
- Some seeds can result in poor convergence rate, or convergence to sub-optimal clusterings.
  - Select good seeds using a heuristic (e.g., doc least similar to any existing mean)
  - Try out multiple starting points
  - Initialize with the results of another method.

Example showing sensitivity to seeds

In the above, if you start with B and E as centroids you converge to {A,B,C} and {D,E,F}
If you start with D and F you converge to {A,B,D,E} {C,F}

K-means Clustering – Details

- Most of the convergence happens in the first few iterations.
  - Often the stopping condition is changed to ‘Until relatively few points change clusters’
- Complexity is O(n * K * I * d)
  - n = number of points, K = number of clusters, I = number of iterations, d = number of attributes

Termination conditions

- Several possibilities, e.g.,
  - A fixed number of iterations.
  - Cluster partition unchanged.
  - Centroid positions don’t change.

Does this mean that the samples in a cluster are unchanged?

Convergence

- Why should the *K*-means algorithm ever reach a *fixed point*?
  - A state in which clusters don't change.
- *K*-means is a special case of a general procedure known as the *Expectation Maximization (EM) algorithm*.
  - EM is known to converge.
  - Number of iterations could be large.
    - But in practice usually isn't

K-means issues, variations, etc.

- Recomputing the centroid after every assignment (rather than after all points are re-assigned) can improve speed of convergence of *K*-means
- Assumes clusters are spherical in vector space
    - Sensitive to coordinate changes, weighting etc.
- Disjoint and exhaustive
    - Doesn't have a notion of "outliers" by default
    - But can add outlier filtering

How Many Clusters?

- Number of clusters K is given
  - Partition n docs into predetermined number of clusters
- Finding the "right" number of clusters is part of the problem
  - Given docs, partition into an "appropriate" number of subsets.
  - E.g., for query results - ideal value of K not known up front - though UI may impose limits.
- Can usually take an algorithm for one flavor and convert to the other.

K not specified in advance

- Given a clustering, define the Benefit for a doc to be the cosine similarity to its centroid
- Define the Total Benefit to be the sum of the individual doc Benefits.

Why is there always a clustering of Total Benefit *n*?

Penalize lots of clusters

- For each cluster, we have a Cost C.
- Thus for a clustering with K clusters, the Total Cost is KC.
- Define the Value of a clustering to be =

  Total Benefit - Total Cost.
- Find the clustering of highest value, over all choices of K.
  - Total benefit increases with increasing K. But can stop when it doesn't increase by "much". The Cost term enforces this.

COMPLEXITY

- In each round, we have to examine each input point exactly once to find closest centroid
- Each round is $O(kN)$ for N points, k clusters
- But the number of rounds to convergence can be very large!

The *K-Means* Clustering Method

Strength

- *Relatively efficient*: $O(tkn)$,
  - n is # objects,
  - k is # clusters
  - t is # iterations. Normally, $k, t \ll n$.

Weakness

- Applicable only when *mean* is defined (e.g., a vector space)
- Need to specify *k,* the *number* of clusters, in advance.
- It is sensitive to noisy data and *outliers* since a small number of such data can substantially influence the mean value.