Clustering

Lecture 06

Topics today…..

An Overview

- What is Clustering
- Types of Clustering
- Common distance Measure techniques
- K-means clustering
- How to implement
- Applications
- Limitations

What is Clustering?

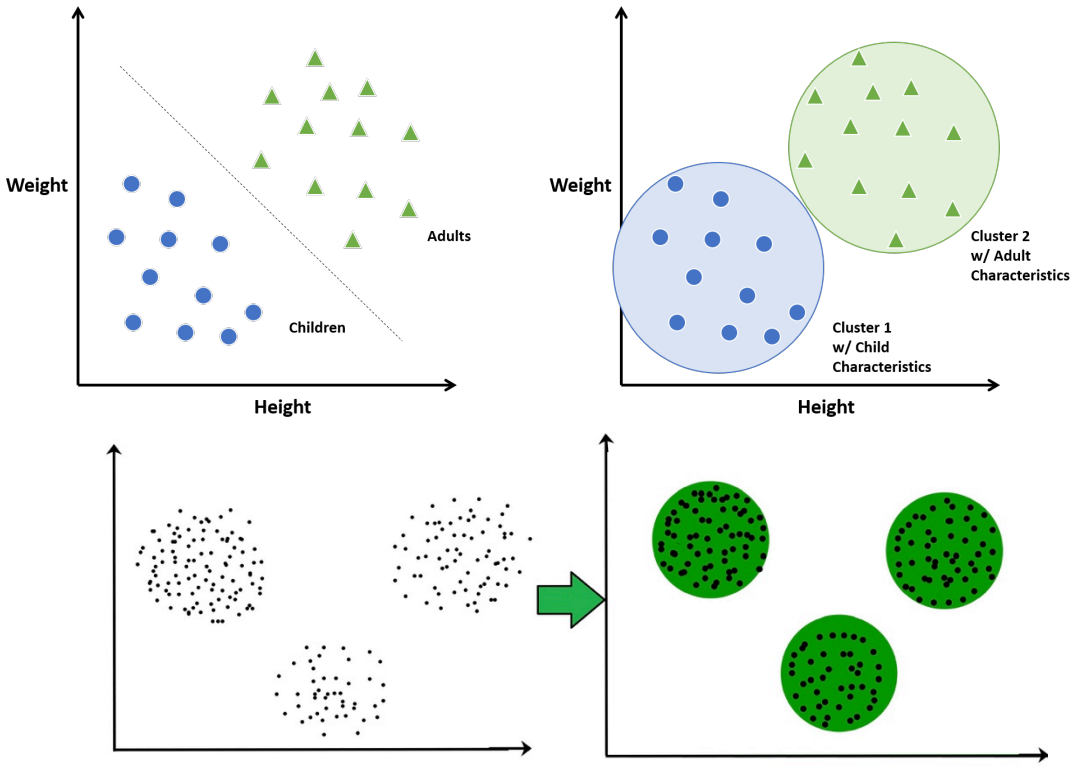
Definition

Clustering is the classification of objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often according to some defined distance measure.

Purpose

- Identify patterns,
- Simplify data analysis,
- Uncover insights within the data.

Classification vs Clustering



Types of Clustering

- Hierarchical algorithms: these find successive clusters using previously established clusters.

❖ Agglomerative ("bottom-up"): Agglomerative algorithms begin with each element as a separate cluster and merge them into successively larger clusters.

- Start with individual points and merge them into clusters

❖ Divisive ("top-down"): Divisive algorithms begin with the whole set and proceed to divide it into successively smaller clusters.

- Begin with a large cluster and split it into smaller clusters.

- Partitional clustering: Partitional algorithms determine all clusters at once.

They include:

- K-means and derivatives
- Fuzzy c-means clustering
- QT clustering algorithm

Heatmap Display

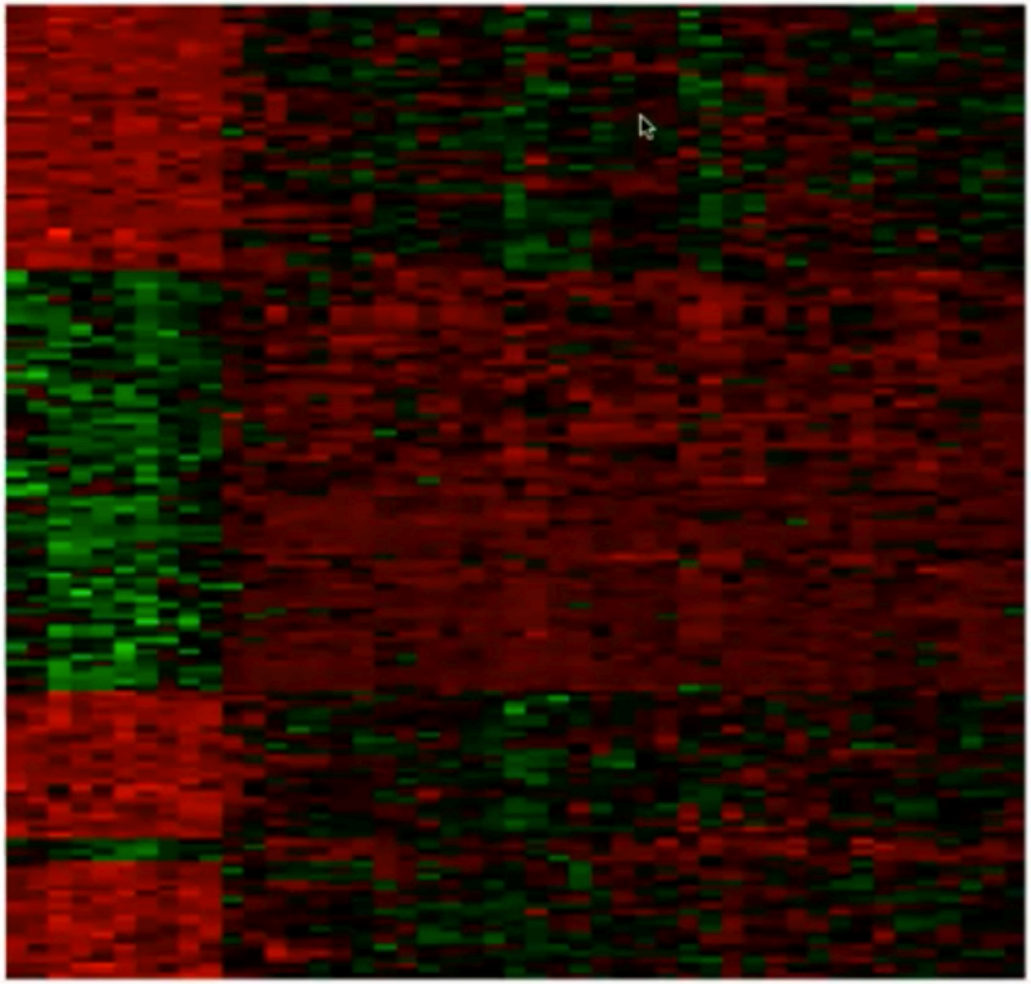
More intuitive than table of numbers

Row: gene

Column: sample

Order is important

How to Order?



Clustering

Can cluster genes or samples

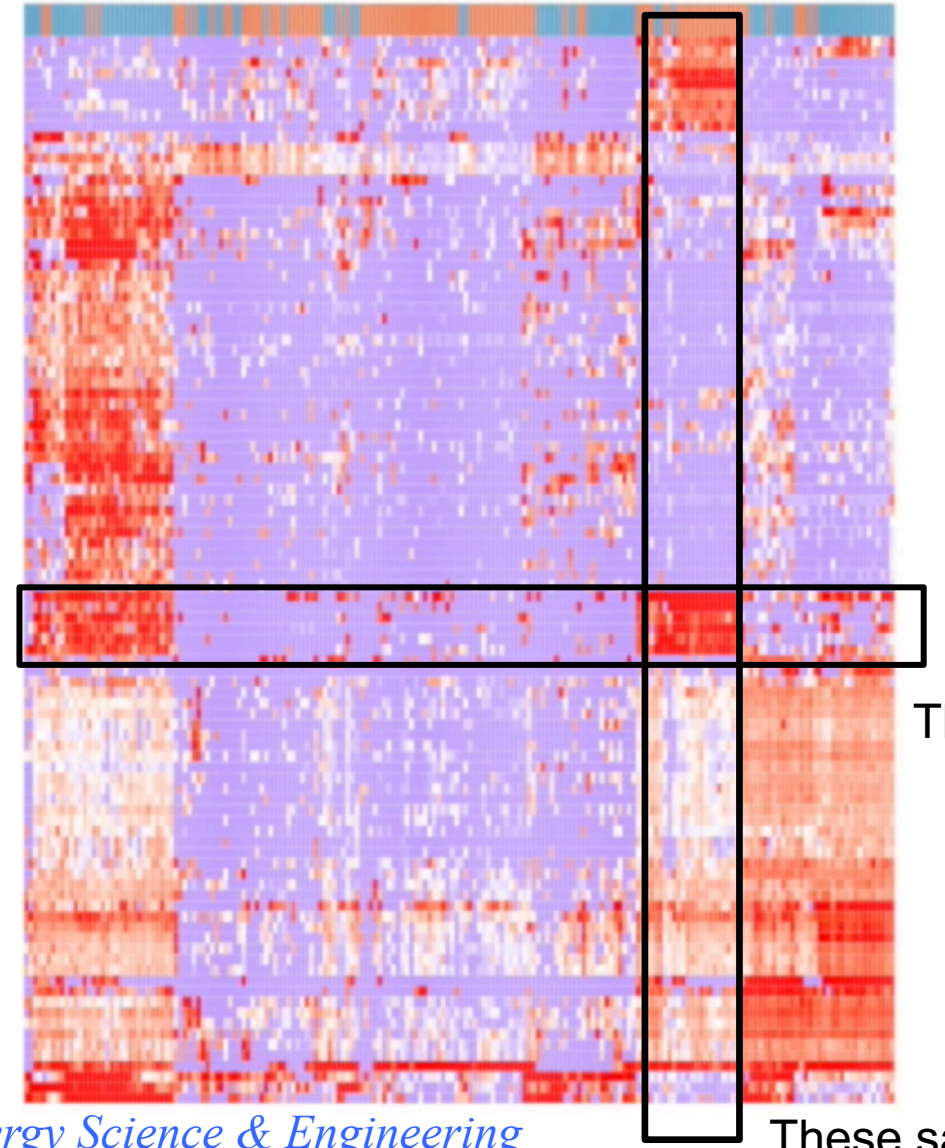
Genes: have similar expression profile over different sample or conditions

Samples: have similar expression profile over all the genes

probe set	gene	Normal m	Normal m4	Normal m4	Normal m	Normal m4	MM m282	MM m331	MM m332	MM m333	MM m334	MM m353
31307_at	pre-T/NK c	28.53	32.61	29.56	36.55	33.19	25.1	32.79	34.3	35.44	28.48	29.55
31308_at	pre-T/NK c	69.14	53.69	52.78	62.07	58.74	67.88	85.82	83.54	85.91	60.93	62.82
31309_r_a	Human bre	16.9	67.7	27.61	46.16	51.46	45.62	35.57	32.62	35.14	96.18	45.94
31310_at	glycine rec	67.42	49.55	55.51	59.57	68.42	91.06	91.23	83.66	76.37	71.23	74.95
31311_at	Homo sapi	78.73	62.91	60.84	72.98	72.9	79.39	85.52	82.57	69.69	63.72	64.29
31312_at	potassium	66.68	59.46	55.47	61.75	69.92	75.28	85.53	97.91	69.92	74.77	71.83
31313_at	mannosyl (	115.33	95.51	84.48	94.99	109.04	105.05	118.68	106.76	142.88	103.72	106.19
31314_at	bone morp	71.89	36.24	41.86	46.99	45.94	46.67	67.56	66.14	53.95	40.97	47.96
31315_at	immunoglc	103.99	88.27	83.81	81.81	254.63	87.12	99.11	109.56	86.37	75.03	74.97
31316_at	Human vac	16.79	10.08	9.53	16.48	11.98	12.8	16.7	18.76	11.25	12.09	18.89
31317_r_a	Human unp	316.79	269.61	254.92	352.61	342.4	327.12	366.39	346	308.43	279.81	312.4
31318_at	Stem cell f	32.66	19.79	27.45	29.56	28.34	26.55	38.04	41.05	31.91	22.76	23.58

Hierarchical clustering is often associated with heatmap

The columns represent different samples



The rows represent measurements from different genes

Hierarchical clustering orders the rows and/or the columns based on similarity

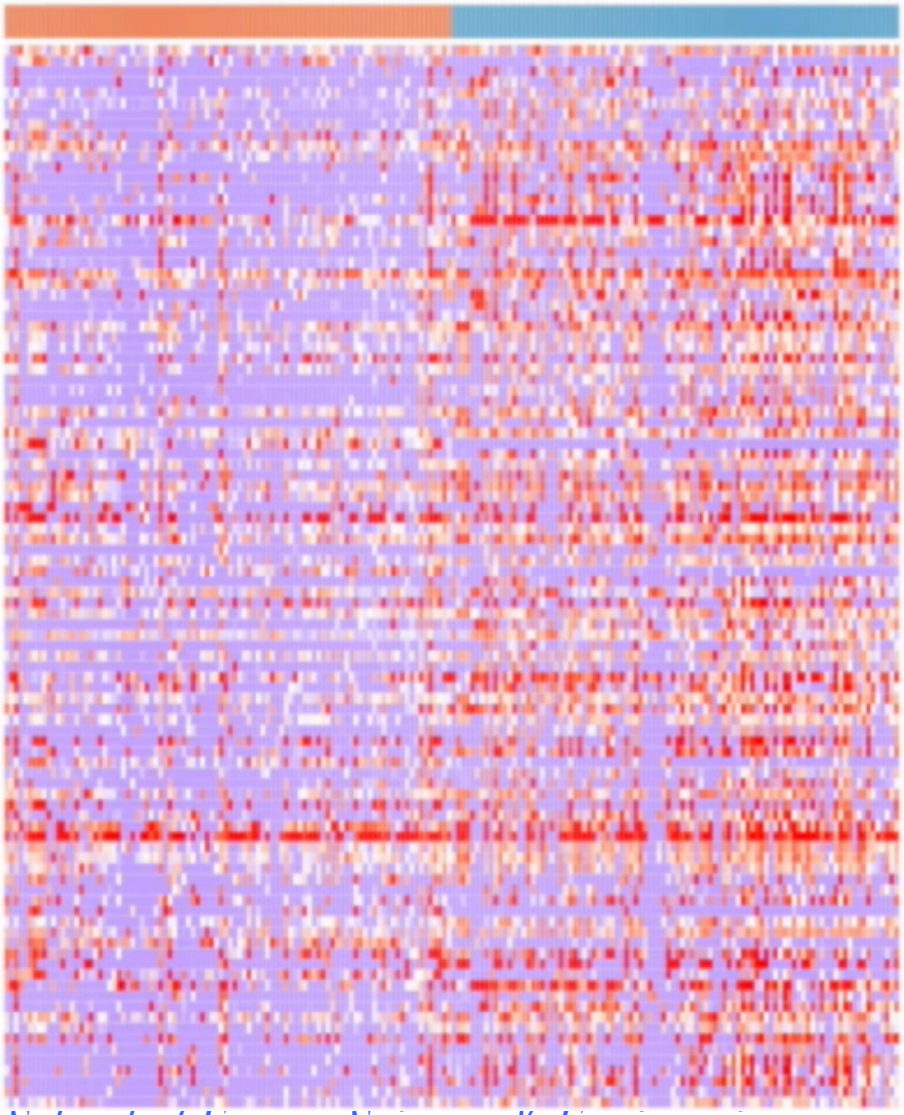
This makes it easy to see correlations in the data

These genes behave the same

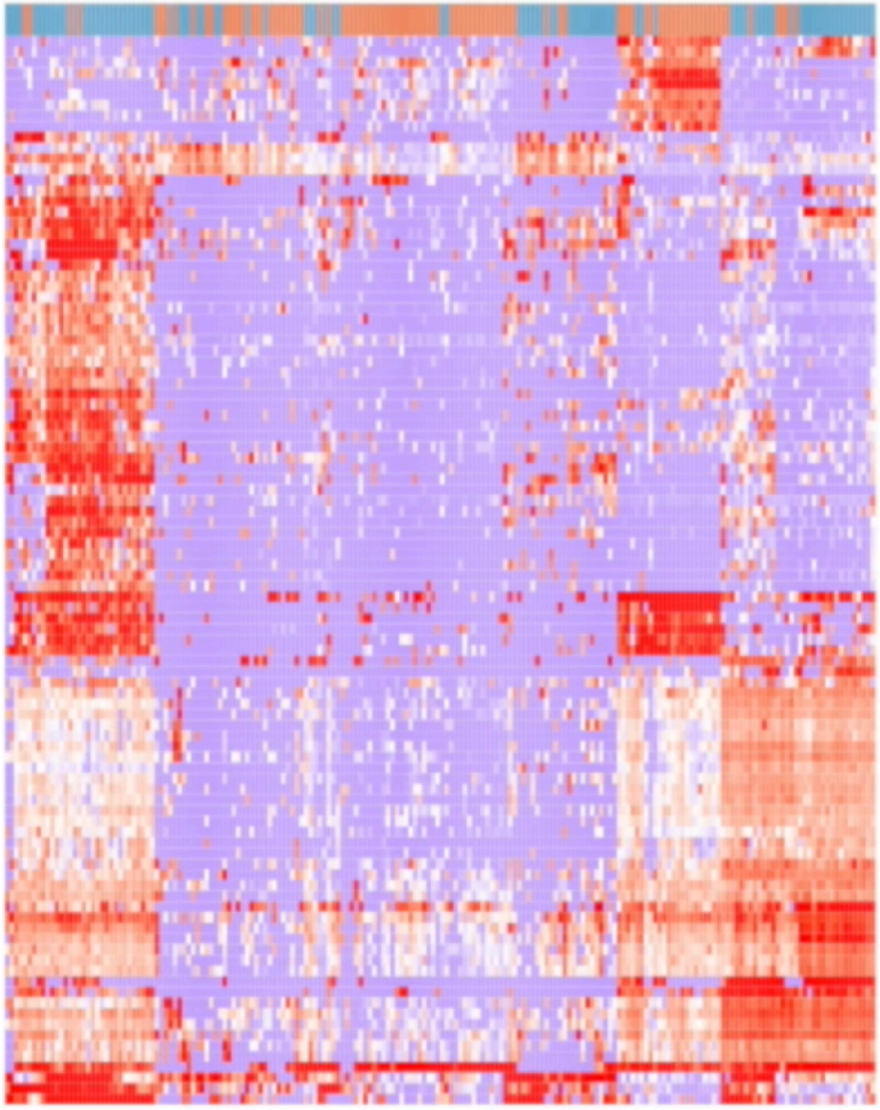
These samples express the same genes

Hierarchical clustering is often associated with heatmaps

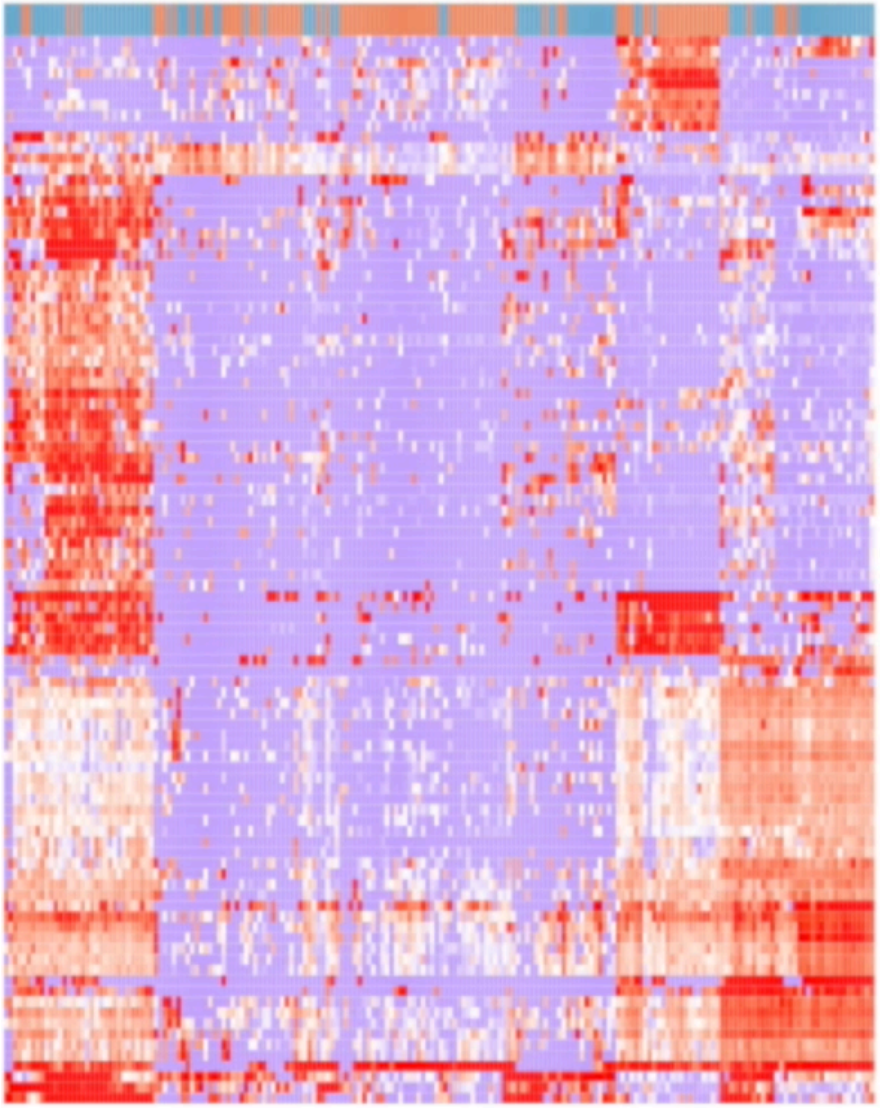
Without hierarchical clustering

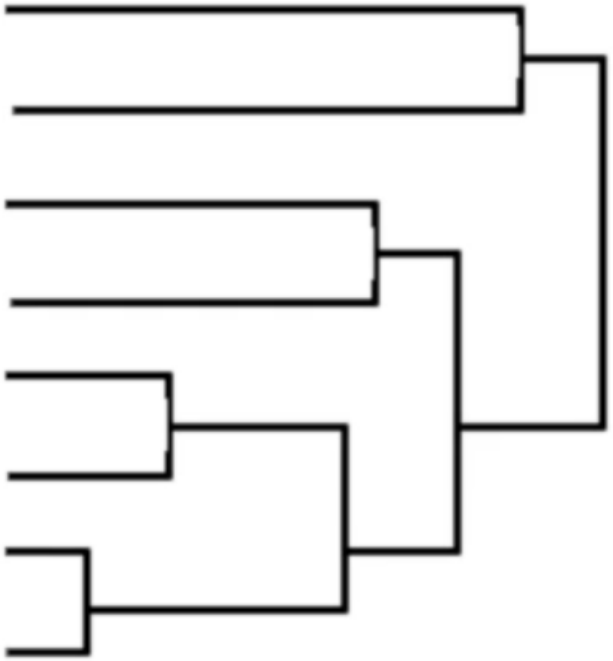


With hierarchical clustering

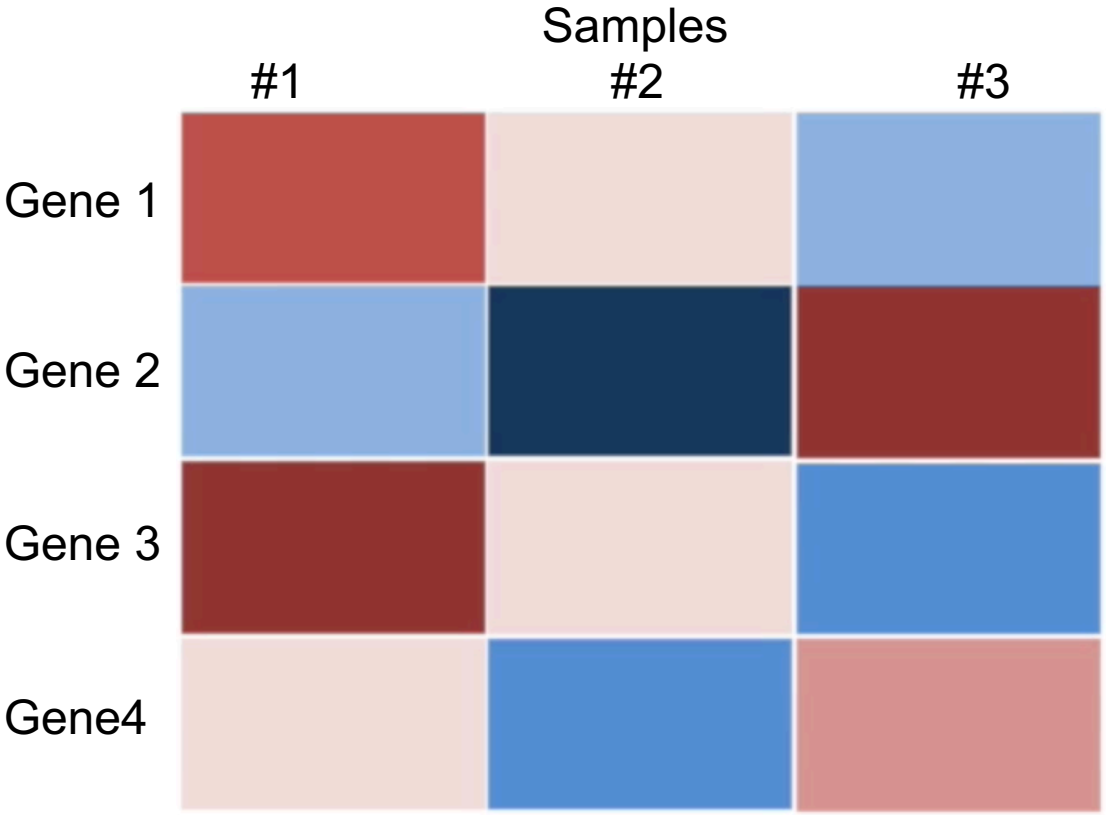


Heatmaps often come with dendrogams



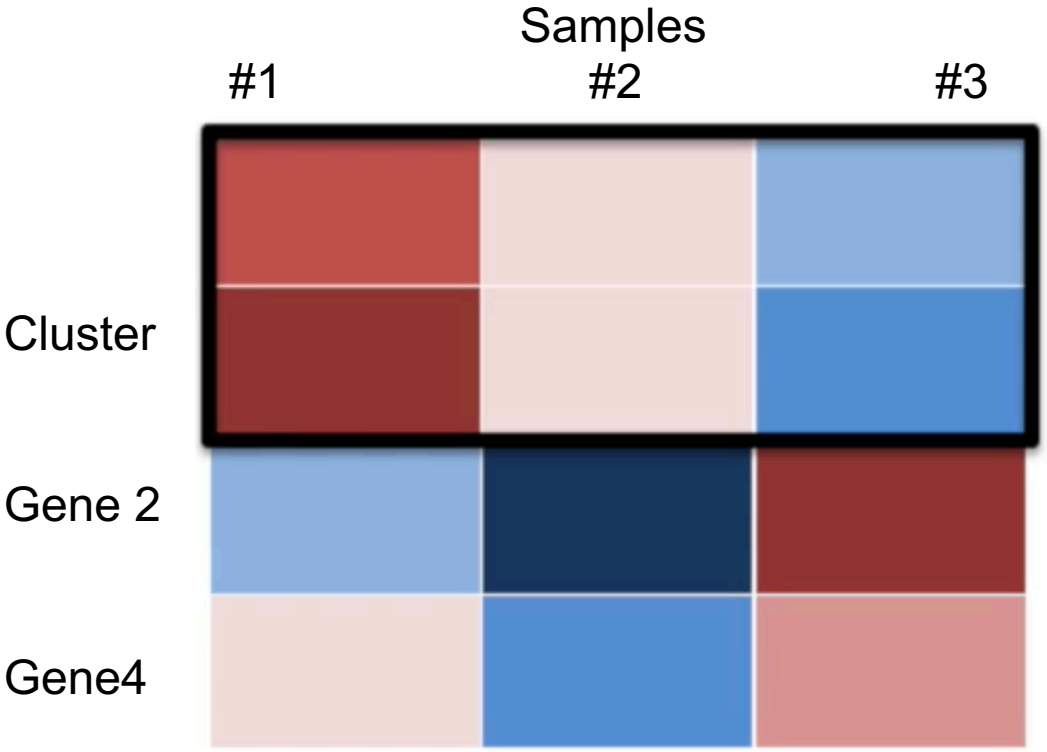


Hierarchical clustering



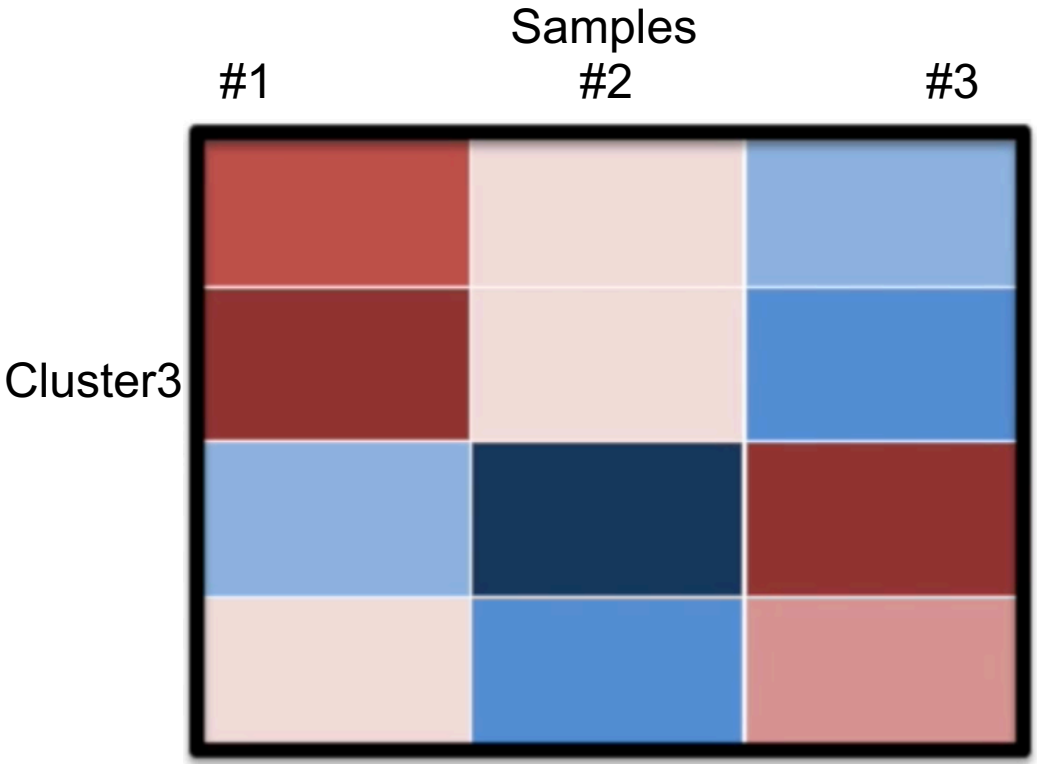
1. Figure out which gene is most similar to gene #1.
2. Figure out which gene is most similar to gene #2 (and then #3 and then #4).
3. Of the different combinations, figure out which two genes are the most similar. Merge them into cluster.
4. Go back to step 1, but now treat the new cluster like it's a single gene.

Hierarchical clustering

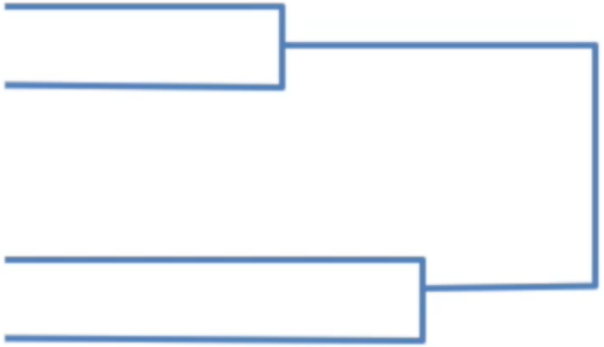


1. Figure out which gene is most similar to cluster #1.
2. Figure out which gene is most similar to gene #2 (and then #4).
3. Of the different combinations, figure out which two genes are the most similar. Merge them into cluster.
4. Go back to step 1, but now treat the new cluster like it's a single gene.

Hierarchical clustering



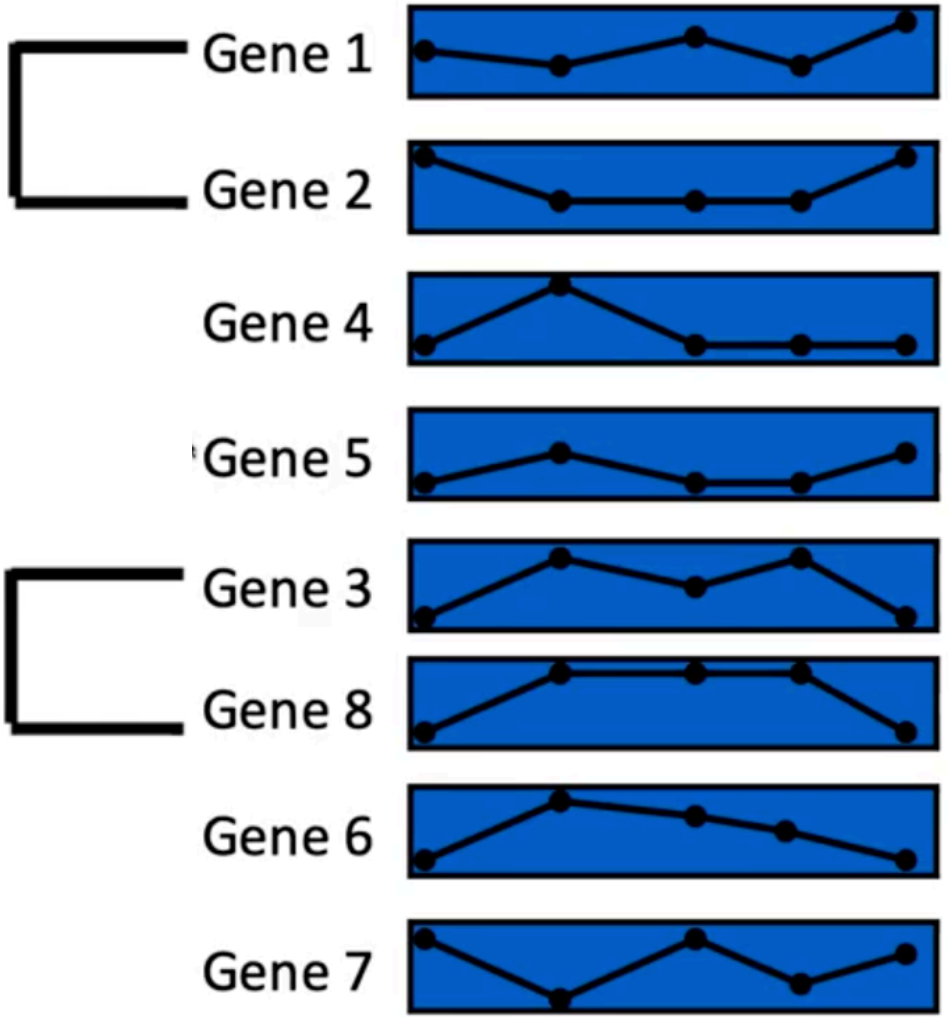
Cluster #1 was formed first and is most similar. It has the shortest branch



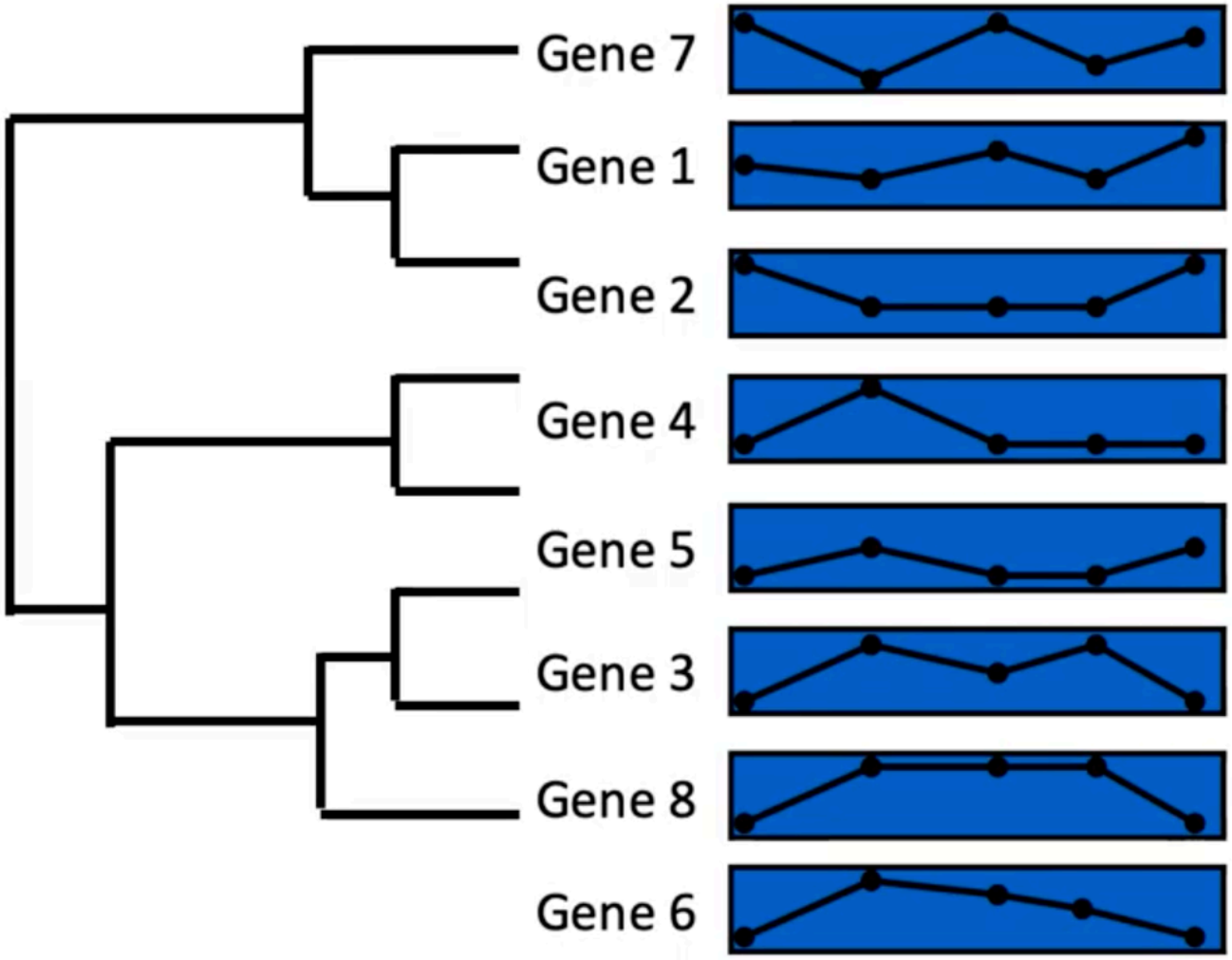
Cluster #2 was second and is second most similar. It has the second shortest branch

Since all we have left are 2 clusters, we merge them.

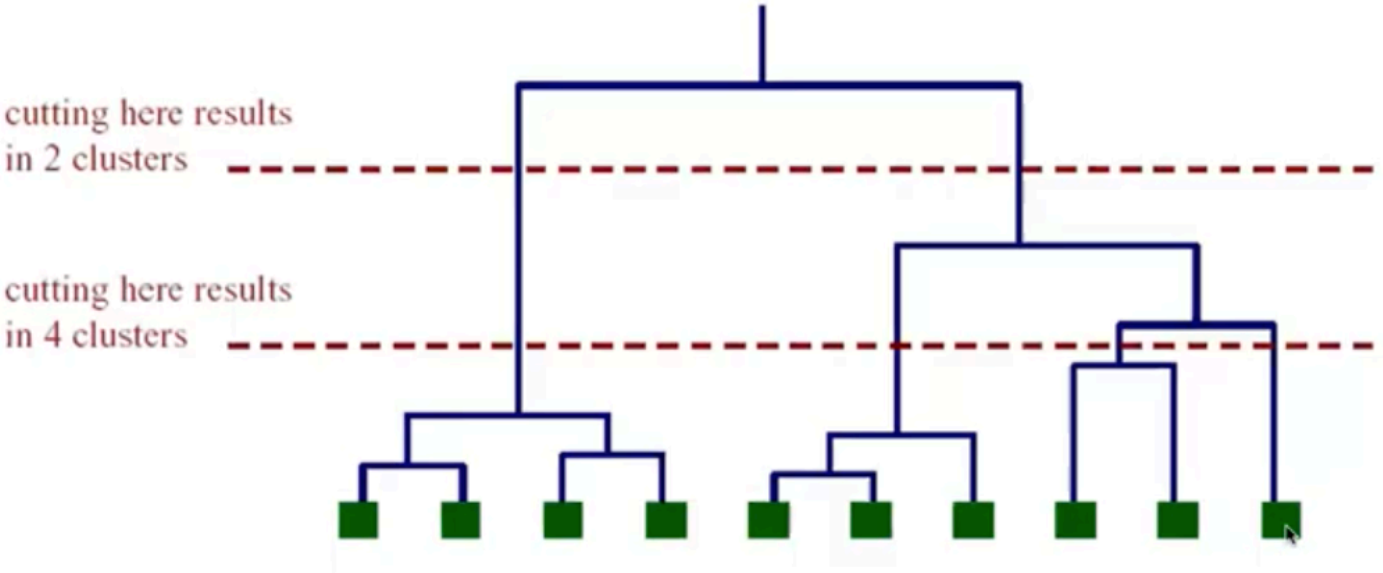
Hierarchical clustering



Hierarchical clustering



Hierarchical clustering

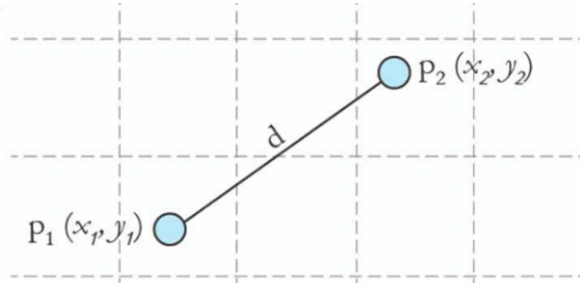


Common Distance Measures

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

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

$$d(x,y) = \sqrt{\sum_{i=1}^{P} \left(\left| x_i - y_i \right| \right)^2}$$



The Manhattan distance (also called taxicab norm or 1-norm) is given by:

$$d(x,y) = \sum_{i=1}^{P} \left| x_i - y_i \right|$$

3. The maximum norm is given by

$$d(x,y) = \max_{1 \le i \le P} \left| x_i - y_i \right|$$

4. The Mahalanobis distance corrects data for different scales and corrections in the variables.

5. Inner product space: The angle between two vectors can be used as a distance measures when clustering high dimensional data

6. Hamming distance (sometimes edit distance) measures the minimum number of substitutions required to change one members to another.

Example

- Let take two points:
- A(1,2)
- B(4,6)
- Calculate Euclidean, Manhattan & maximum norm distance.

$$d(x,y) = \sqrt{\sum_{i=1}^{P}\left(\left|x_i - y_i\right|\right)^2}$$

$$\text{Euclidean Distance} = \sqrt{\left(4-1\right)^2 + \left(6-2\right)^2} = 5$$

$$d(x,y) = \sum_{i=1}^{P}\left|x_i - y_i\right|$$

$$\text{Manhattan Distance} = \left|4-1\right| + \left|6-2\right| = 3+4 = 7$$

$$d(x,y) = \max_{1\le i\le P}\left|x_i - y_i\right|$$

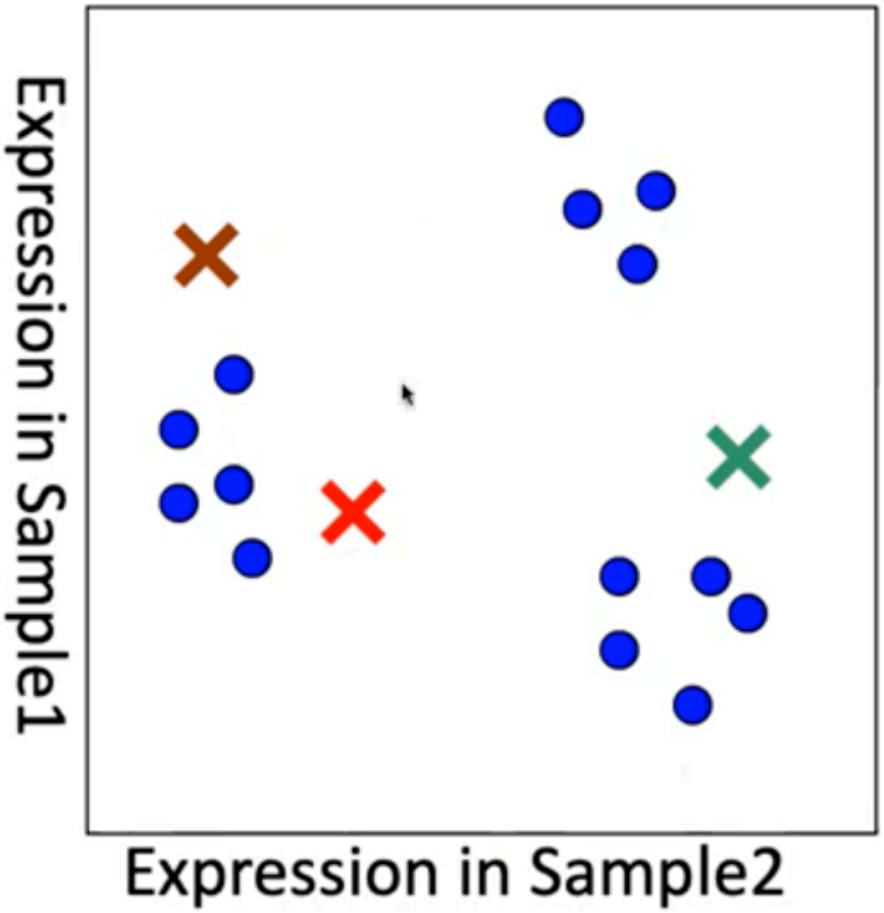
$$\text{Max Norm Distance} = \max\left(\left|4-1\right|,\left|6-2\right|\right) = \max\left(3,4\right) = 4$$

Summary

- **Euclidean Distance** is useful for measuring straight-line distances.
  - ➢ Measuring the straight-line distance between two points
- **Manhattan Distance** is appropriate for grid-based pathfinding scenarios.
  - ➢ Calculating the distance a taxi would drive in a city with a grid layout (like New York City).
- **Max Norm Distance** is suitable in cases where only the largest difference in any dimension matters.
  - ➢ Evaluating the distance a king would move in chess, where the king can move one square in any direction.

K-Means Clustering

Choose K centroids at random

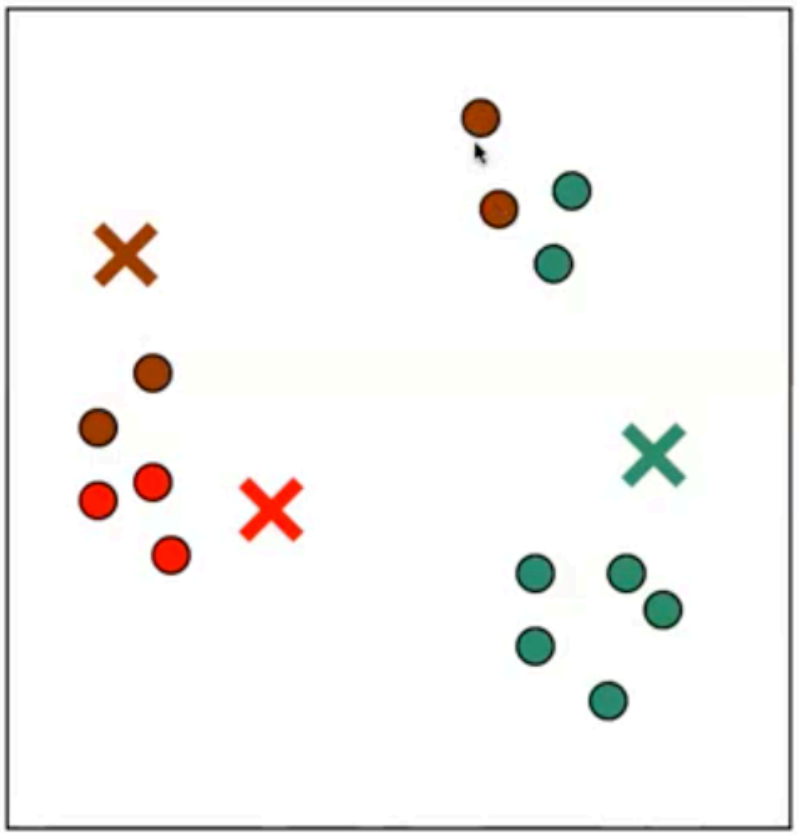


Iteration = 0

K-Means Clustering

Choose K centroids at random

Assign object i to closest centroid



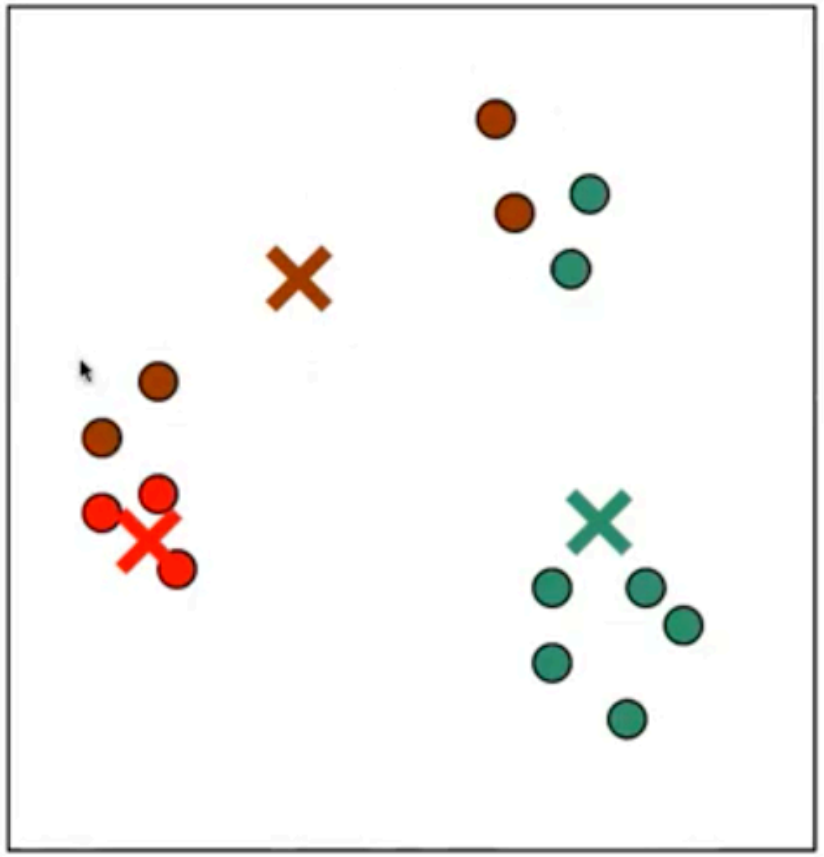
Iteration = 1

K-Means Clustering

Choose K centroids at random

Assign object i to closest centroid

Recalculate centroid based on current cluster assignment



Iteration = 2

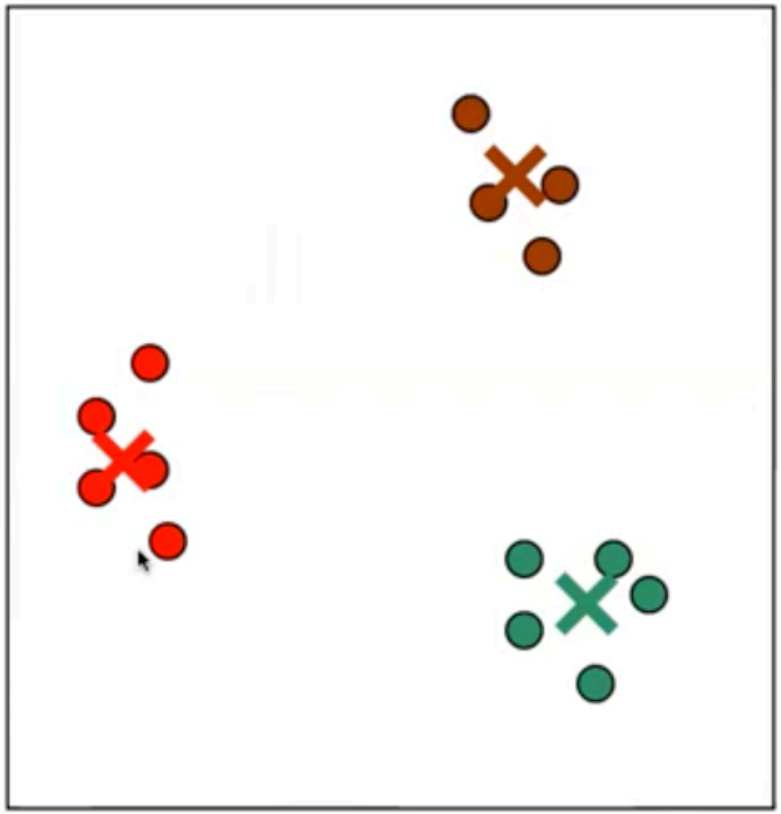
K-Means Clustering

Choose K centroids at random

Assign object i to closest centroid

Recalculate centroid based on current cluster assignment

Repeat until assignment stabilize



Iteration = 3

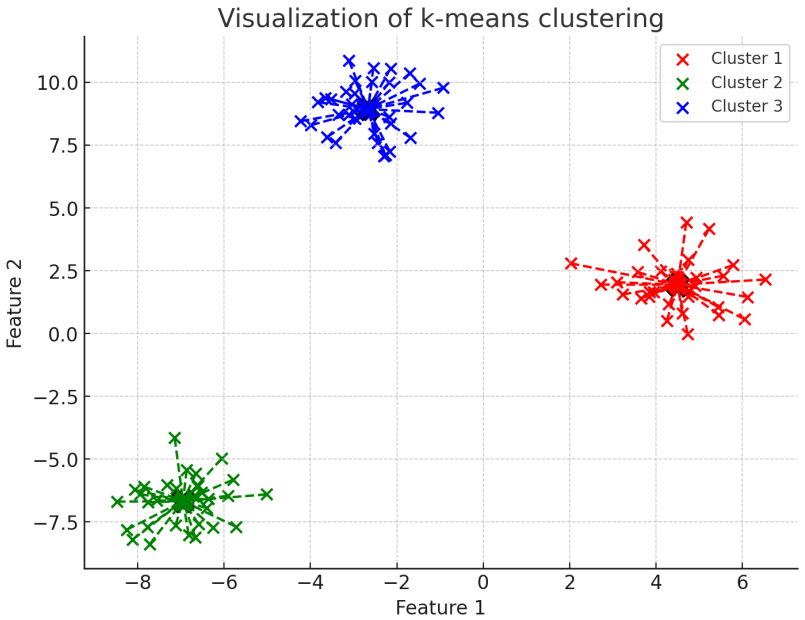
K-Means Clustering

- The k-means algorithm is an algorithm to cluster n objects based on attributes into k partitions, where k < n.
- It is similar to the expectation-maximization algorithm for mixtures of Gaussians in that they both attempt to find the centers of natural clusters in the data.
- It assumes that the object attributes form a vector space.
- An algorithm for partitioning (or clustering) N data points into K disjoint subsets Sj containing data points so as to minimize the sum-of-squares criterion

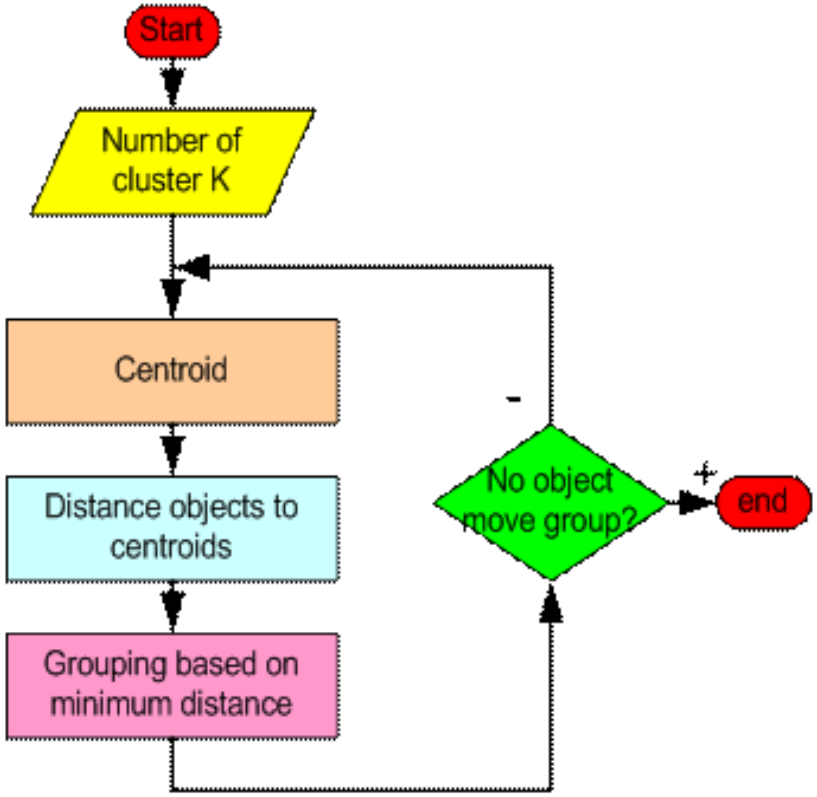
$$J = \sum_{j-1}^{K} \sum_{n \in S_j} \left(\left| x_i - y_i \right| \right)^2$$

where x_n is a vector representing the the nth data point and μ_j is the geometric centroid of the data points in Sj.

- Simply speaking k-means clustering is an algorithm to classify or to group the objects based on attributes/features into K number of group.
- K is positive integer number.
- The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid.



How K-Means Clustering works?



How to do K-Means Clustering

- Step 1: Begin with a decision on the value of k = number of clusters .
- Step 2: Put any initial partition that classifies the data into k clusters. You may assign the training samples randomly, or systematically as the following:

1. Take the first k training sample as single-element clusters
2. Assign each of the remaining (N-k) training sample to the cluster with the nearest centroid. After each assignment, recompute the centroid of the gaining cluster.

How to do K-Means Clustering

- Step 3: Take each sample in sequence and compute its distance from the centroid of each of the clusters. If a sample is not currently in the cluster with the closest centroid, switch this sample to that cluster and update the centroid of the cluster gaining the new sample and the cluster losing the sample.

- Step 4 . Repeat step 3 until convergence is achieved, that is until a pass through the training sample causes no new assignments.

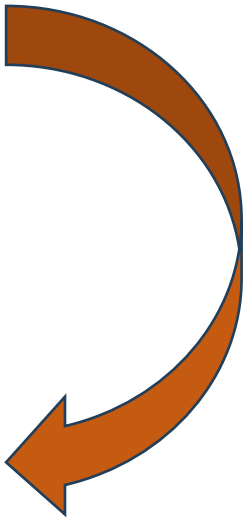
A Simple example showing the implementation of k-means algorithm (using K=2)

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

A Simple example showing the implementation of k-means algorithm (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:

Thus, we obtain two clusters containing:
{1,2,3} and {4,5,6,7}.
Their new centroids are:

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

$$m_2 = \left(\frac{1}{4}(5.0+3.5+4.5+3.5), \frac{1}{3}(7.0+5.0+5.0+4.5)\right) = (4.12, 5.38)$$

Individual	Variables	Centroid 1	Centroid 2
1	(1.0, 1.0)	0	7.21
2	(1.5, 2.0)	1.12	6.10
3	(3.0, 4.0)	3.61	3.61
4	(5.0, 7.0)	7.21	0
5	(3.5, 5.0)	4.72	2.5
6	(4.5, 5.0)	5.32	2.06
7	(3.5, 4.5)	4.30	2.92

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

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

Step 3:

Now using these centroids we compute the Euclidean distance of each object, as shown in table.

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)

Individual	Variables	Centroid 1	Centroid 2
1	(1.0, 1.0)	1.57	5.38
2	(1.5, 2.0)	0.47	4.28
3	(3.0, 4.0)	2.04	1.78
4	(5.0, 7.0)	5.64	1.84
5	(3.5, 5.0)	3.15	0.73
6	(4.5, 5.0)	3.78	0.54
7	(3.5, 4.5)	2.74	1.08

Step 4 :

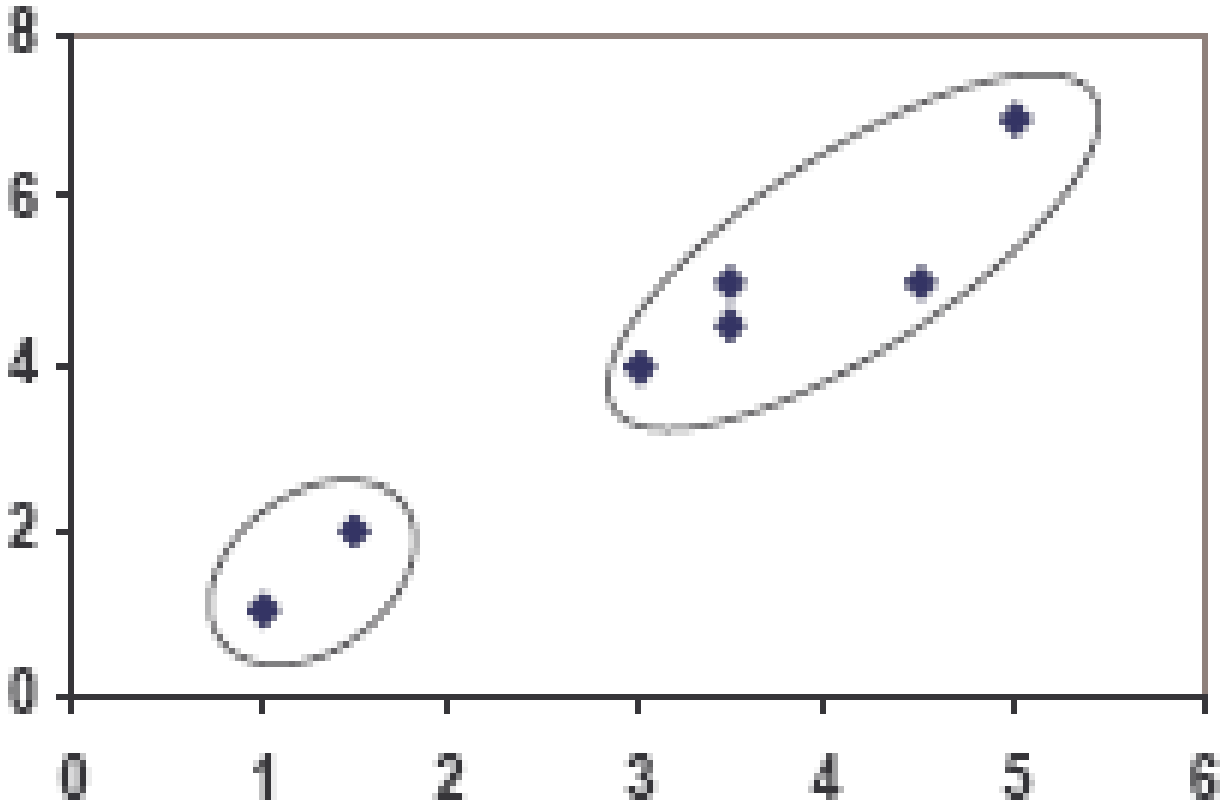
The clusters obtained are:
{1,2} and {3,4,5,6,7}

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



(using K=3)

Individual	m1=1	m2=2	m3=3	cluster
1	0	1.11	3.61	1
2	1.12	0	2.5	2
3	3.61	2.5	0	3
4	7.21	6.10	3.61	3
5	4.72	3.61	1.12	3
6	5.31	4.24	1.80	3
7	4.30	3.20	0.71	3

C_3

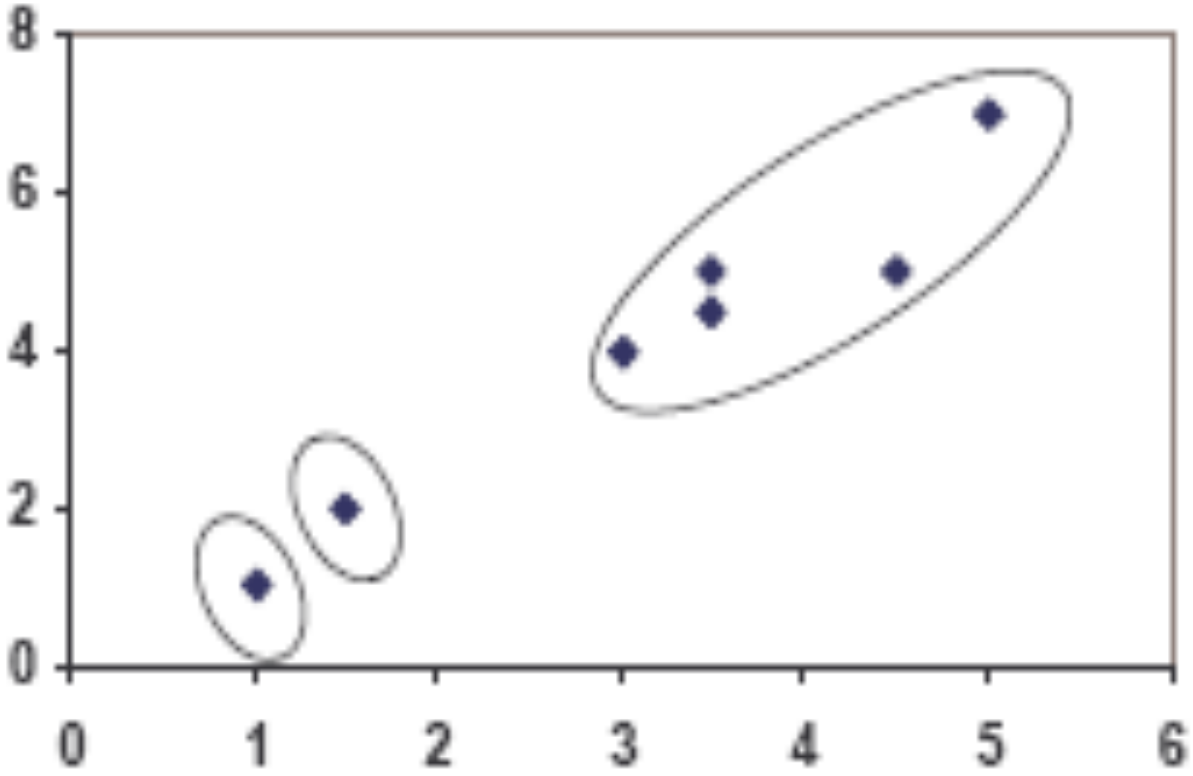
Clustering with initial centroids(1,2,3)

Step 1

Individual	m1 (1.0,1.0)	m2 (1.5,2.0)	m3 (3.9,5.1)	cluster
1	0	1.11	5.02	1
2	1.12	0	3.92	2
3	3.61	2.5	1.42	3
4	7.21	6.10	2.20	3
5	4.72	3.61	0.41	3
6	5.31	4.24	0.61	3
7	4.30	3.20	0.72	3

Step 2

Plot



Real-Life Numerical Example of K-Means Clustering

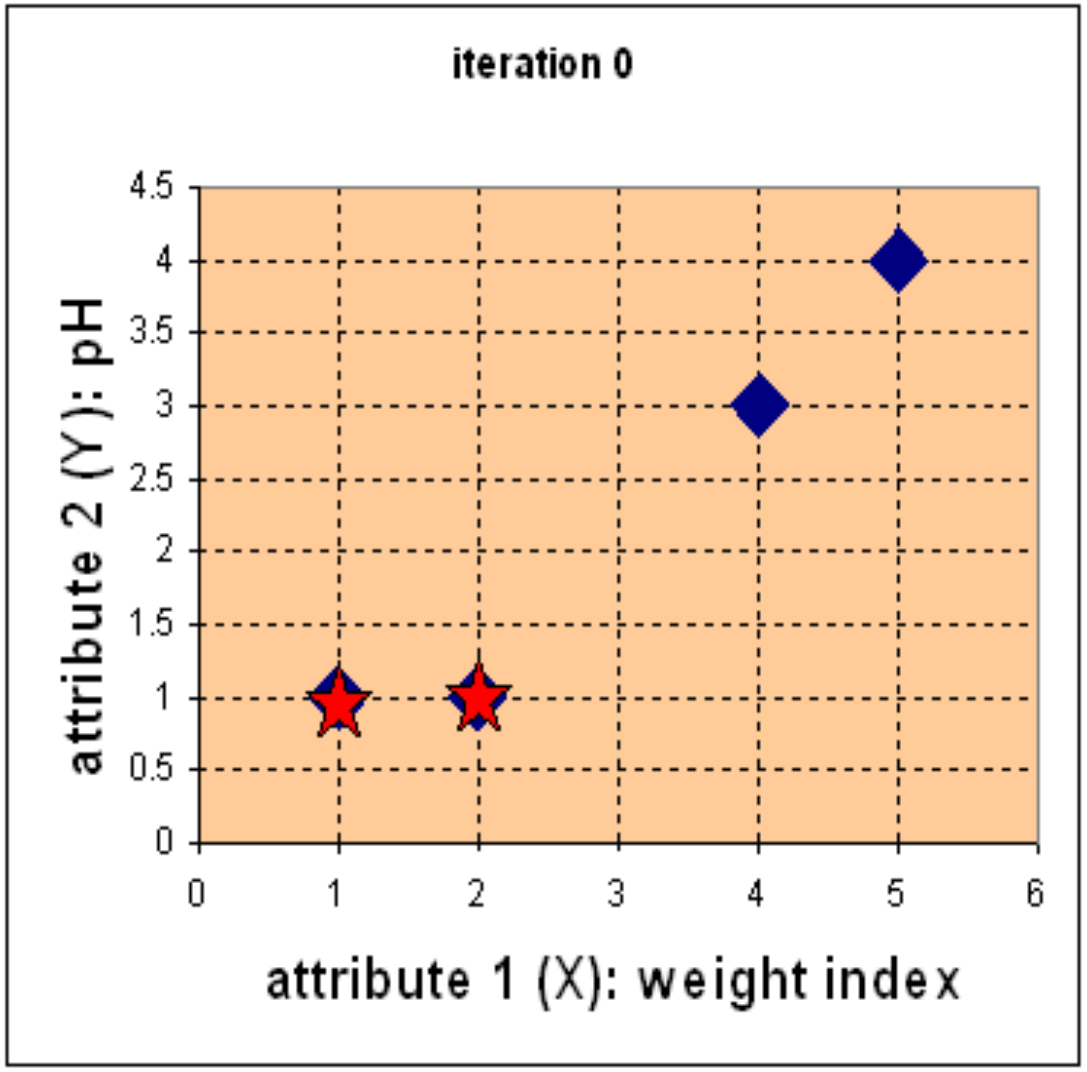
- We have 4 medicines as our training data points object and each medicine has 2 attributes. Each attribute represents coordinate of the object. We have to determine which medicines belong to cluster 1 and which medicines belong to the other cluster.

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 c1 and c2 denote the coordinate of the centroids, then c1=(1,1) and c2=(2,1)



Objects-Centroids distance: We calculate the distance between cluster centroid to each object. Let us use Euclidean distance, then we have distance matrix at iteration 0 is

$$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix} \quad \begin{matrix} c_1 = (1,1) & group-1 \\ c_2 = (2,1) & group-2 \end{matrix}$$

$$\begin{matrix} A & B & C & D \\ \begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} & & & \begin{matrix} X \\ Y \end{matrix} \end{matrix}$$

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

Each column in the distance matrix symbolizes the object. The first row of the distance matrix corresponds to the distance of each object to the first centroid and the second row is the distance of each object to the second centroid. For example, distance from medicine C=(4,3) to the first centroid C1=(1,1) is

$$\sqrt{(4-1)^2 + (3-1)^2} = 3.61$$

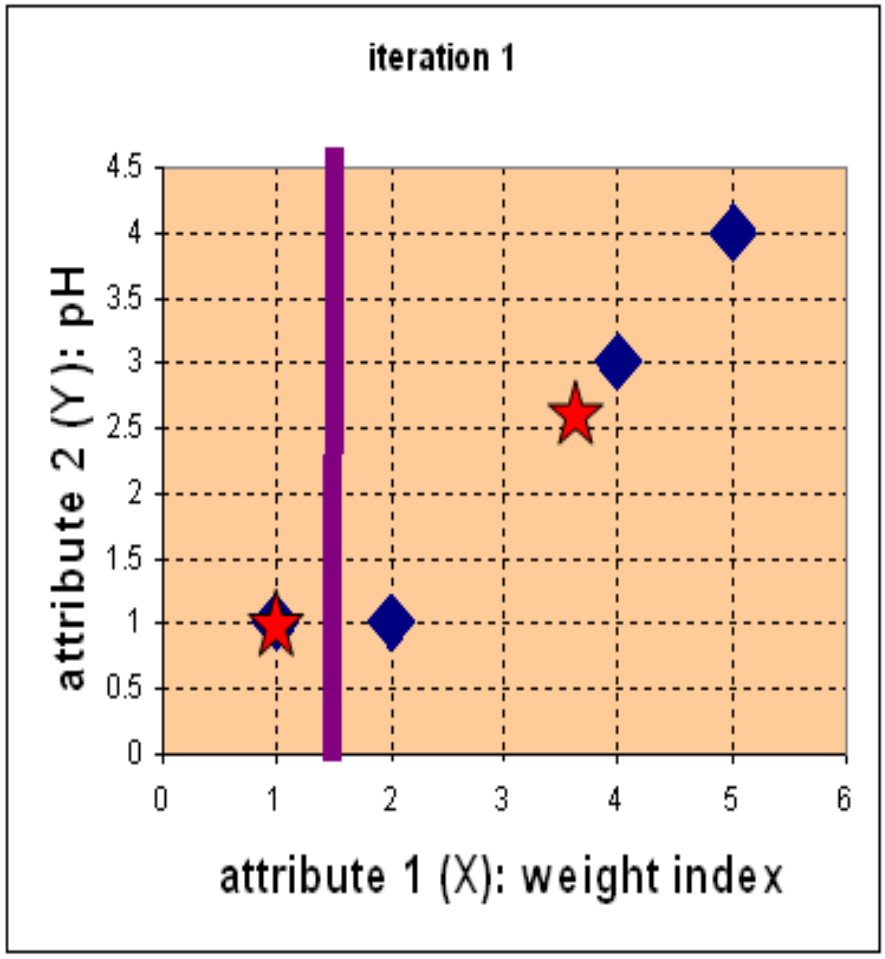
Its distance to the second centroid C1=(2,1) is $\sqrt{(4-2)^2 + (3-1)^2} = 2.83$

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.

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

$$A \quad B \quad C \quad D$$



Iteration-1, Objects-Centroids distances :

The next step is to compute the distance of all objects to the new centroids.
Group1 has only one member.
So that, the centroid will be that value only i.e, C1=(1,1)
Group 2 has 3 members, thus the new centroids are $C2 = \left(\frac{2+4+5}{2}, \frac{1+3+4}{2}\right) = \left(\frac{11}{3}, \frac{8}{3}\right) = (3.67, 2.67)$

Similar to step 2, we have distance matrix at iteration 1 is

$$\mathbf{D}^1 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \end{bmatrix} \quad \begin{matrix} \mathbf{c}_1 = (1,1) & group-1 \\ \mathbf{c}_2 = (3.67, 2.67) & group-2 \end{matrix}$$

$$\begin{matrix} A & B & C & D \end{matrix}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

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

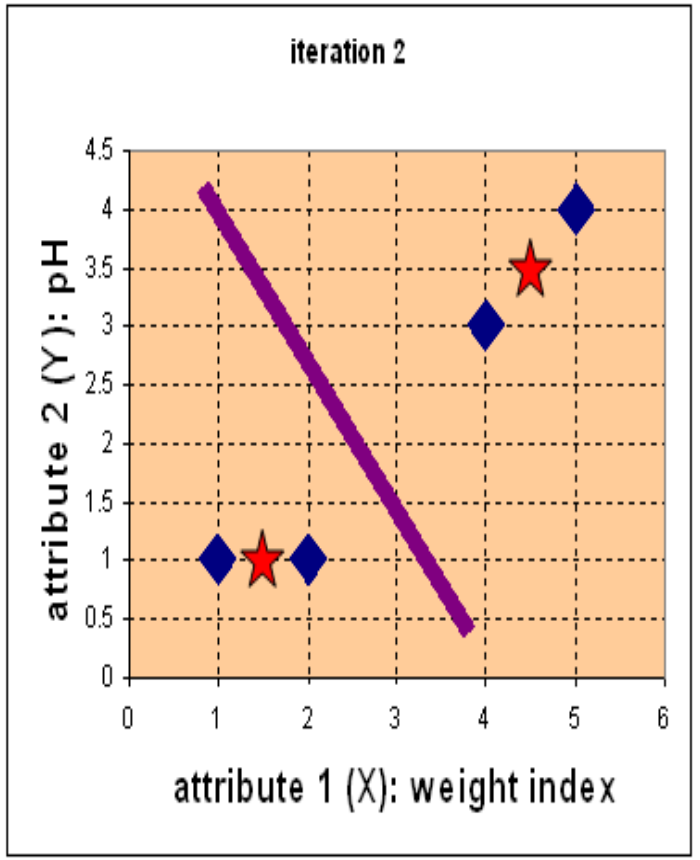
$$\mathbf{G}^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \quad \begin{matrix} group-1 \\ group-2 \end{matrix}$$

$$A \quad B \quad C \quad D$$

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 $C1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = (1.5,1)$ and $C2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = (4.5,3.5)$



Iteration-2,

Objects-Centroids distances :

Repeat step 2 again, we have new distance matrix at iteration 2 as

$$\mathbf{D}^2 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.30 & 3.54 & 0.71 & 0.71 \end{bmatrix} \quad \begin{matrix} \mathbf{c}_1 = (1.5,1) \quad group-1 \\ \mathbf{c}_2 = (4.5,3.5) \quad group-2 \end{matrix}$$

$$\begin{matrix} A & B & C & D \end{matrix}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$$

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

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

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

$$A \quad B \quad C \quad D$$

- We obtain result that $G^1=G^2$. 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

Weaknesses of K-Mean Clustering

- When the numbers of data are not so many, initial grouping will determine the cluster significantly.
- The number of cluster, K, must be determined before hand. Its disadvantage is that it does not yield the same result with each run, since the resulting clusters depend on the initial random assignments.
- We never know the real cluster, using the same data, because if it is inputted in a different order it may produce different cluster if the number of data is few.
- It is sensitive to initial condition. Different initial condition may produce different result of cluster. The algorithm may be trapped in the local optimum.

Applications of K-Mean Clustering

- It is relatively efficient and fast. It computes result at **O(tkn),** where n is number of objects or points, k is number of clusters and t is number of iterations.
- k-means clustering can be applied to machine learning or data mining
- Used on acoustic data in speech understanding to convert waveforms into one of k categories (known as Vector Quantization or Image Segmentation).
- Also used for choosing color palettes on old fashioned graphical display devices and Image Quantization.
- K-means has found wide spread usage in lot of fields, ranging from unsupervised learning of neural network, Pattern recognitions, Classification analysis, Artificial intelligence, image processing, machine vision, and many others.

Numerical

Determine which state belong to which cluster after 3 iterations in K-means clustering.

	GPD (in Lakh crores)	Illiteracy Rate (%)
State 1	10.5	12
State 2	21.4	3.2
State 3	14.6	3.7
State 4	39.5	2.1
State 5	19.2	4.5