# MACHINE LEARNING (ML-14)

Dr. NEERAJ GUPTA, Department of CEA, GLA University, Mathura

# AGENDA

- Machine Learning
- Types of Machine Learning
- Unsupervised Learning : Clustering
- Unsupervised Example
- Applications of Clustering
- K-Means Algorithm
- K-Means Examples

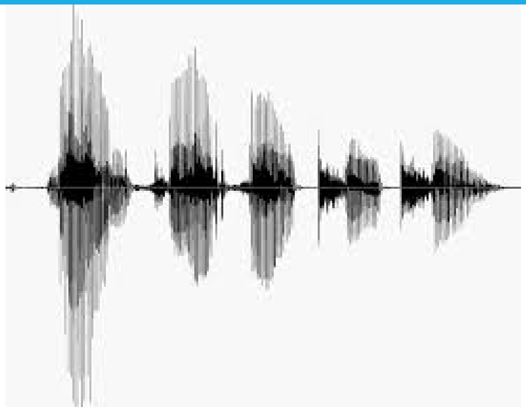
# WHEN DO WE USE MACHINE LEARNING?

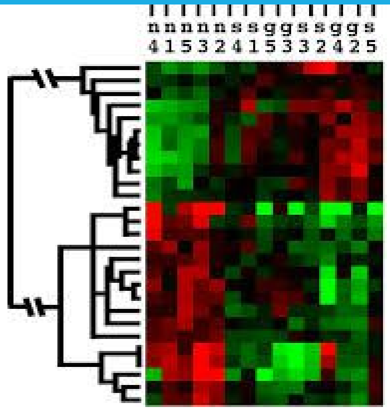
ML is used when:

- Human expertise does not exist (navigating on Mars)

**Learning isn't always useful:**

**There is no need to "learn" to calculate payroll**





SRC: Based on slide by E. Alpaydin

# TYPES OF LEARNING

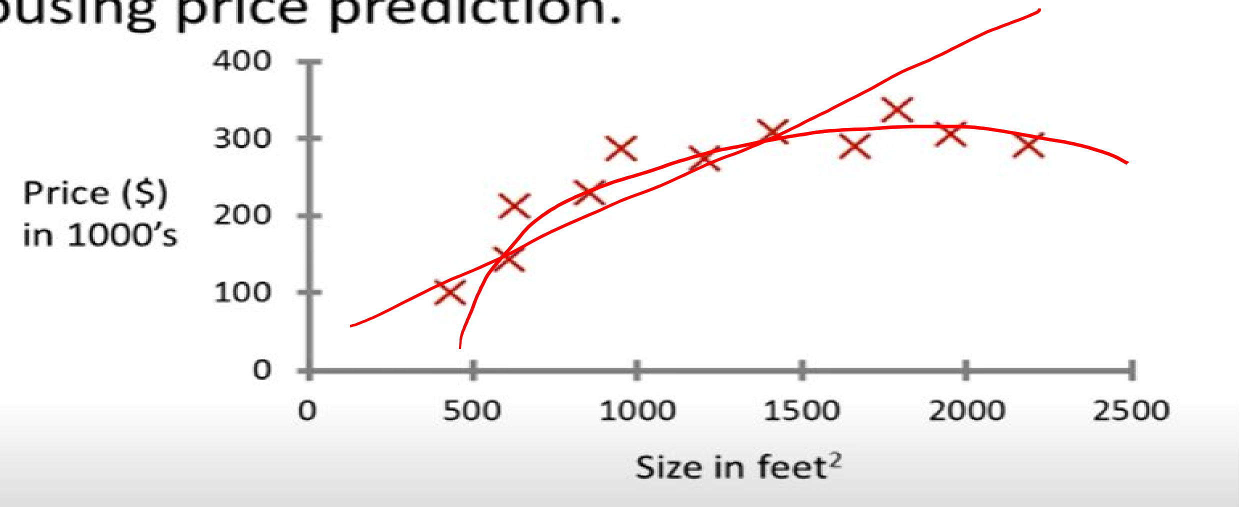
- **Supervised learning**
  - Given: training data+ desired outputs (labels)
- **Unsupervised learning**
  - Given: training data (without desired outputs)
- **Semi-supervised learning**
  - Given: training data + a few desired outputs
- **Reinforcement learning**
  - Rewards from sequence of actions

# SUPERVISED LEARNING: REGRESSION

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict $y$ given $x$
  - $y$ is real-valued == regression

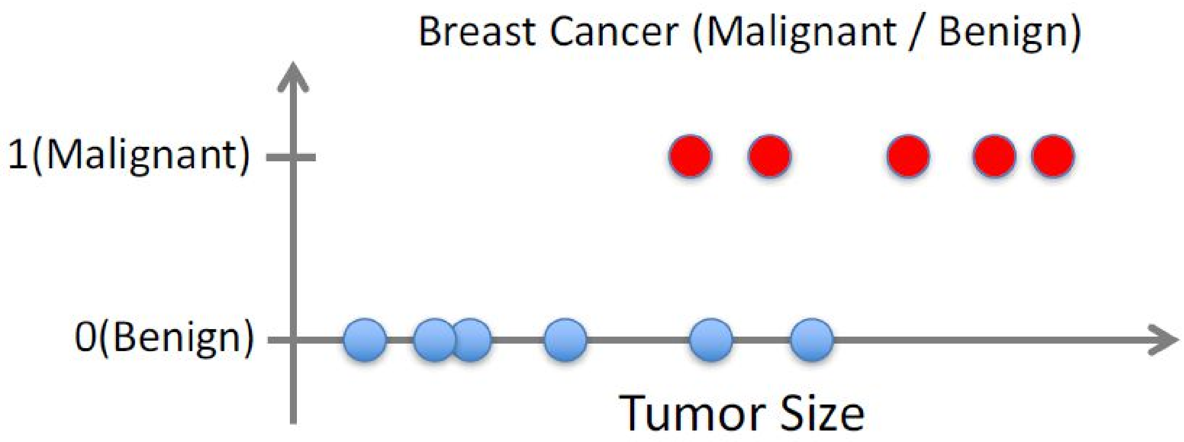
$$\boxed{h_\theta(x) = \theta x + \theta_0}$$

Housing price prediction.



# SUPERVISED LEARNING: CLASSIFICATION

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict $y$ given $x$
  - $y$ is categorical == classification



# UNSUPERVISED LEARNING

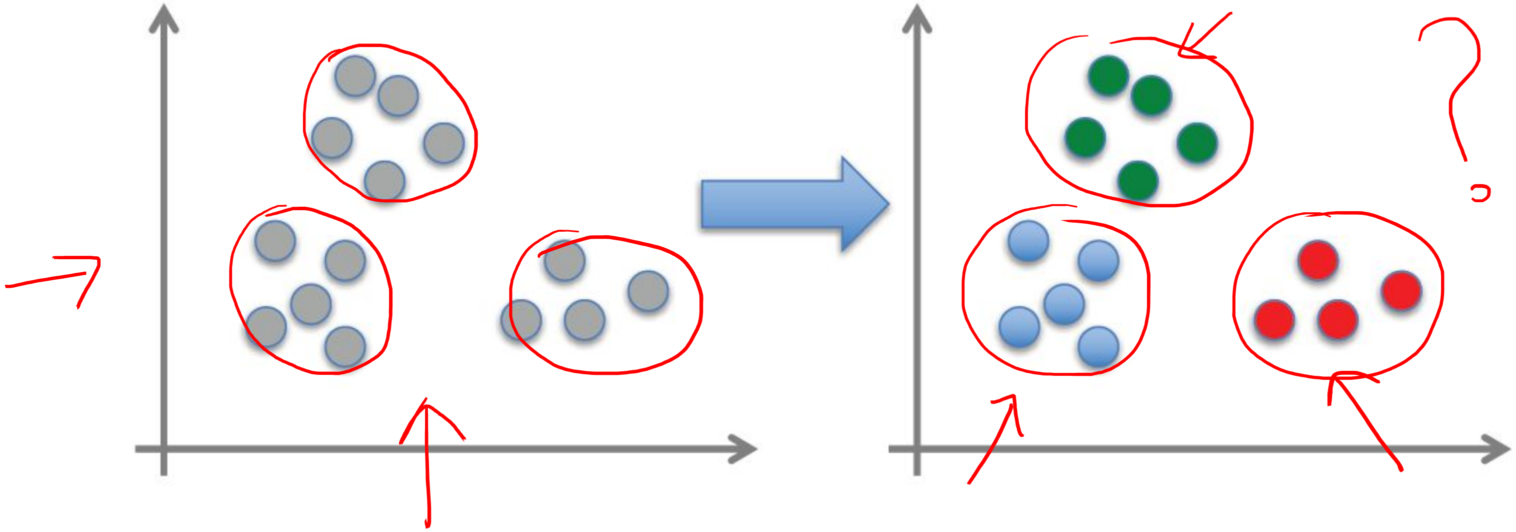
X, Y

target label

? ⇒ features

- Given $x_1, x_2, \ldots, x_n$ (without labels)
- Output hidden structure behind the $x$'s
  - E.g., clustering

| | W Kg | H cm |
|---|---|---|
| Small | 40 – 50 | 110 – 130 |
| Medium | 51 – 60 | 130 – 145 |
| High | 61 – 70 | 146 – 170 |



?

How

# SUPERVISED VS UNSUPERVISED LEARNING

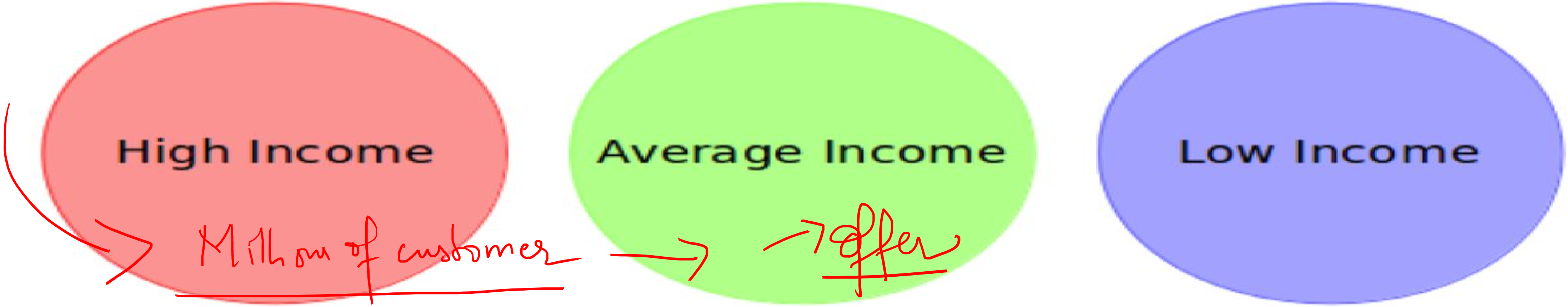
Supervised learning: discover patterns in the data that relate data attributes with a target (class) attribute.

- Patterns are then utilized to predict the values of the target attribute in future data instances.

Unsupervised learning: The data have no target attribute.

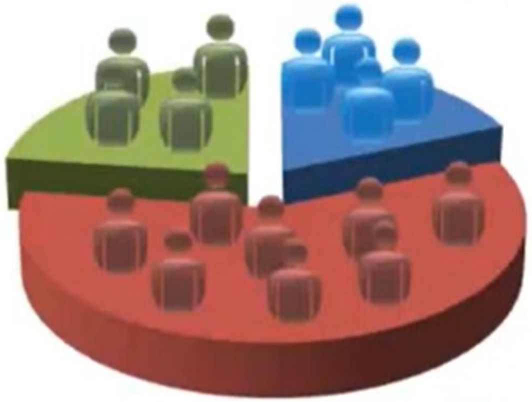
- To explore the data to find some intrinsic structures in them.

# UNSUPERVISED EXAMPLE

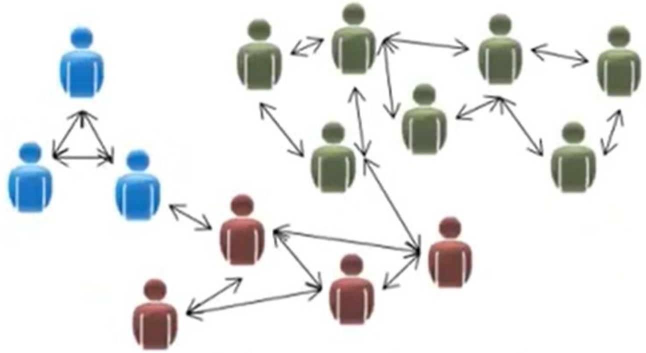



Clustering is the process of dividing the entire data into groups (also known as clusters) based on the patterns in the data.

# APPLICATIONS OF CLUSTERING IN REAL-WORLD


Market segmentation


Organize computing clusters


Social network analysis

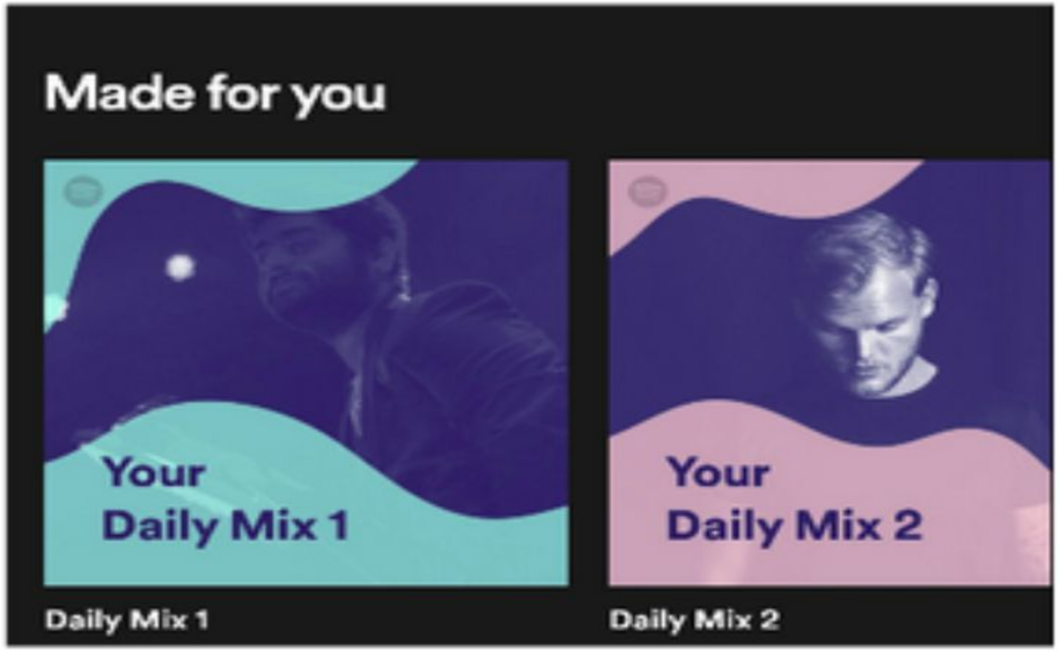

Astronomical data analysis

# APPLICATIONS OF CLUSTERING IN REAL-WORLD

**Image Segmentation**

Clustering is used to collect similar pixels in the same group.

# APPLICATIONS OF CLUSTERING IN REAL-WORLD




**Recommendation Engines**

GLA UNIVERSITY MATHURA
Accredited with A Grade by NAAC

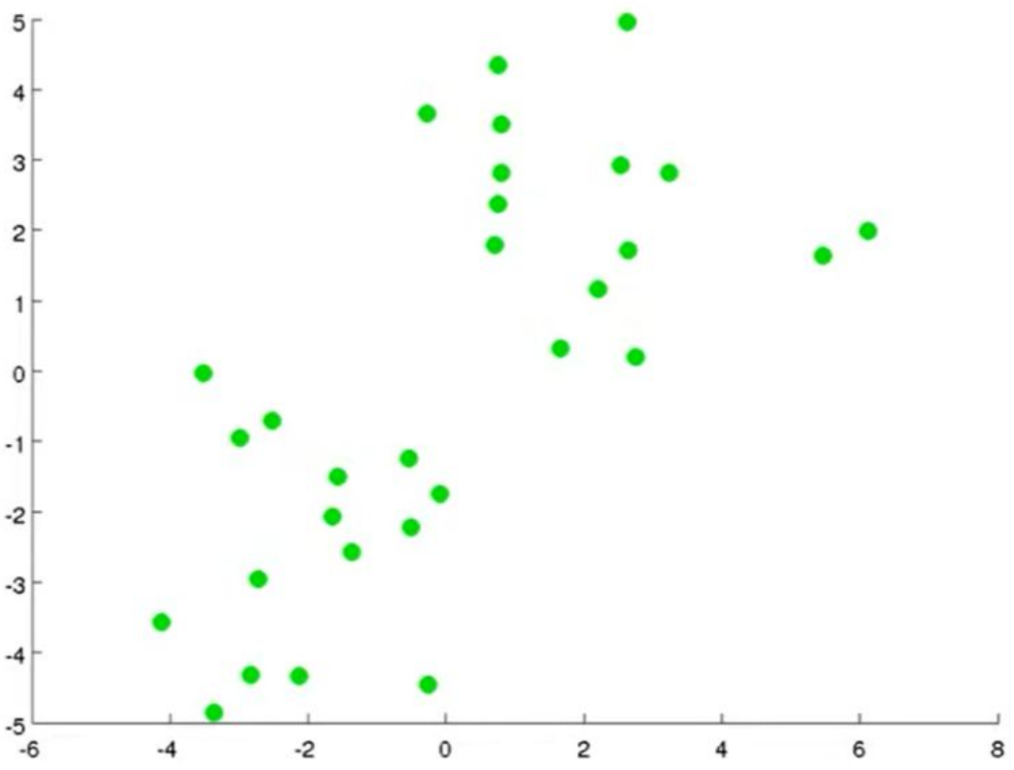
# K-MEANS CLUSTERING

Centroid

$K = 2, 3, 4, \ldots n$

$\boxed{K = 2}$ $\boxed{K=2}$

$x_2$




$x_1$

The main objective of the K-Means algorithm is to minimize the sum of distances between the points and their respective cluster centroid.

Distance ?

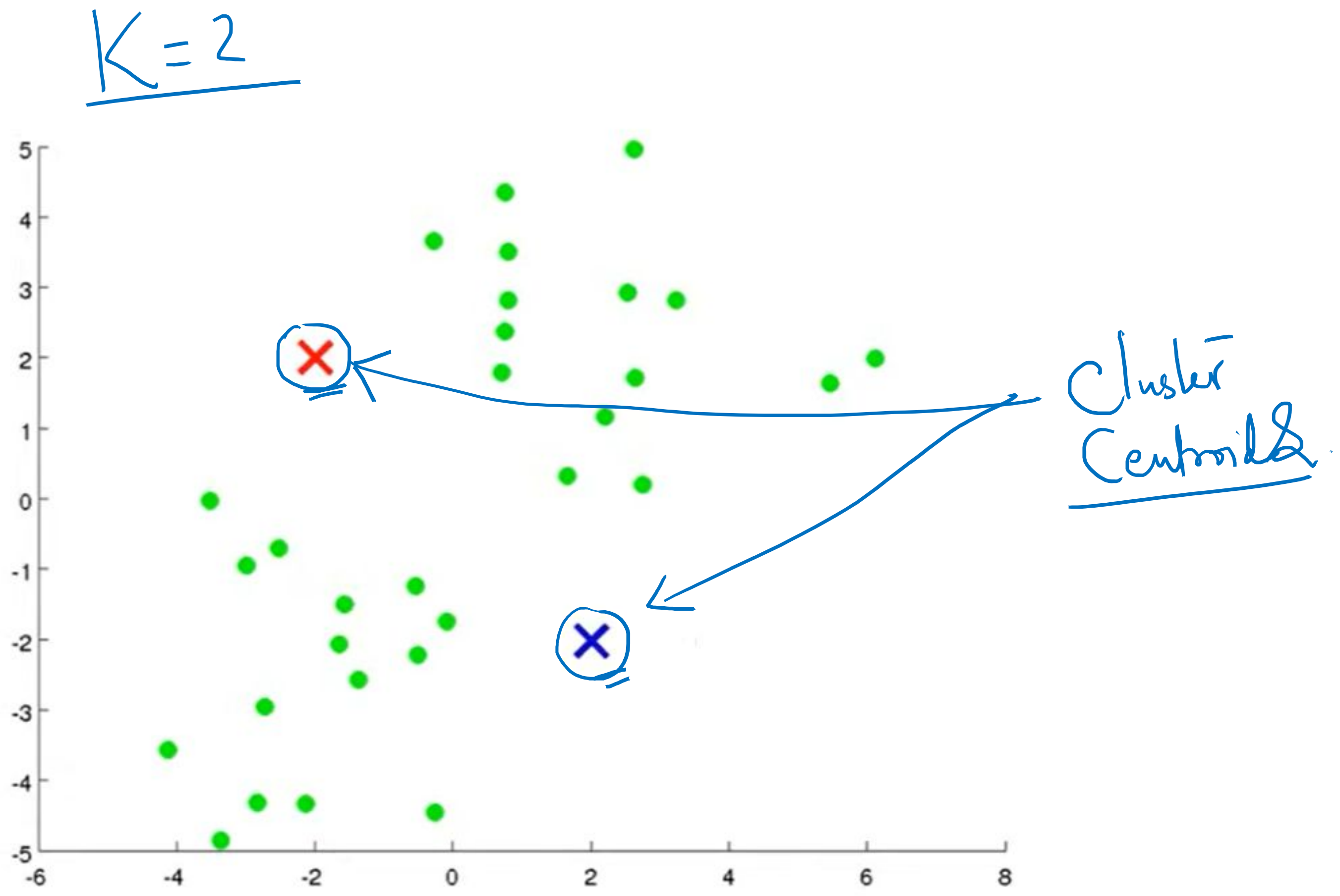
$P_2$ $(x_2, y_2)$

$P_1$ $(x_1, y_1)$

$$E.D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

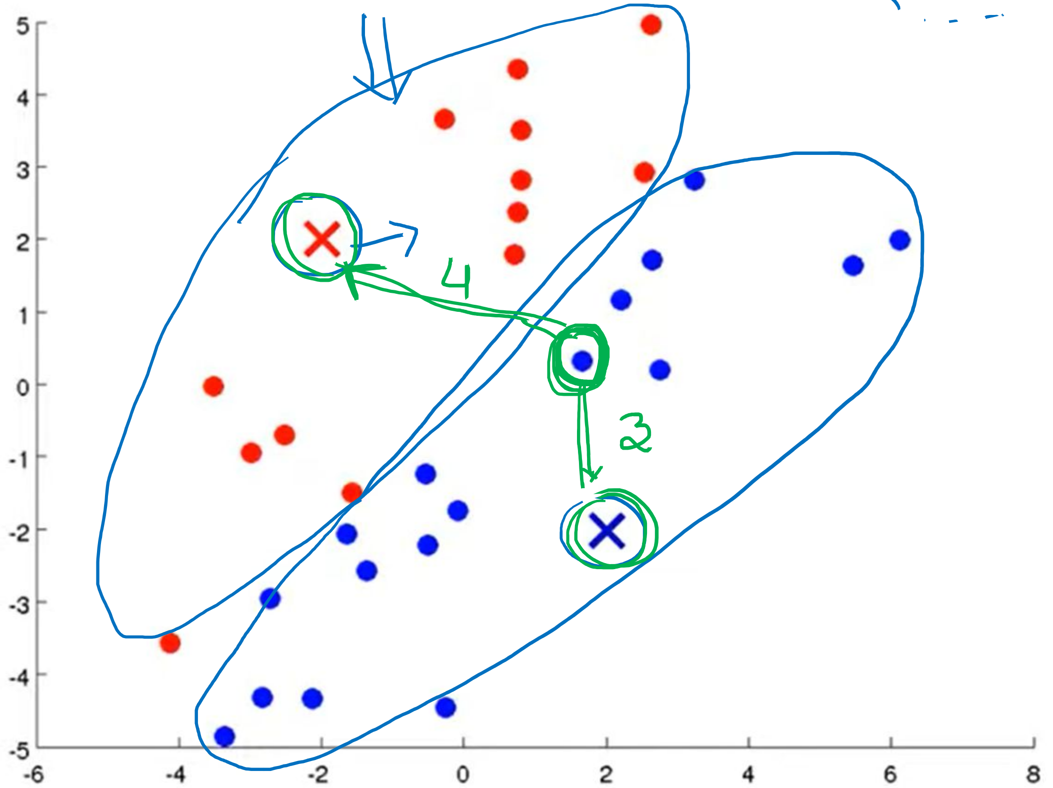
$$\text{Rectilinear} = |x_1 - x_2| + |y_1 - y_2|$$

# K-MEANS CLUSTERING

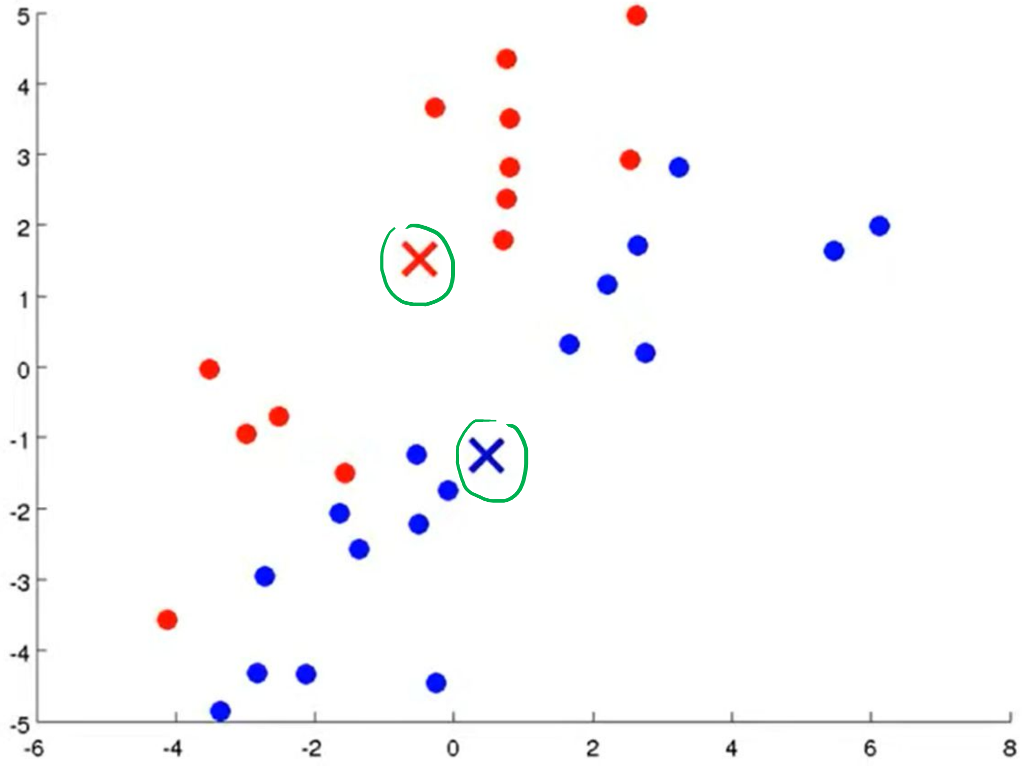


# K-MEANS CLUSTERING

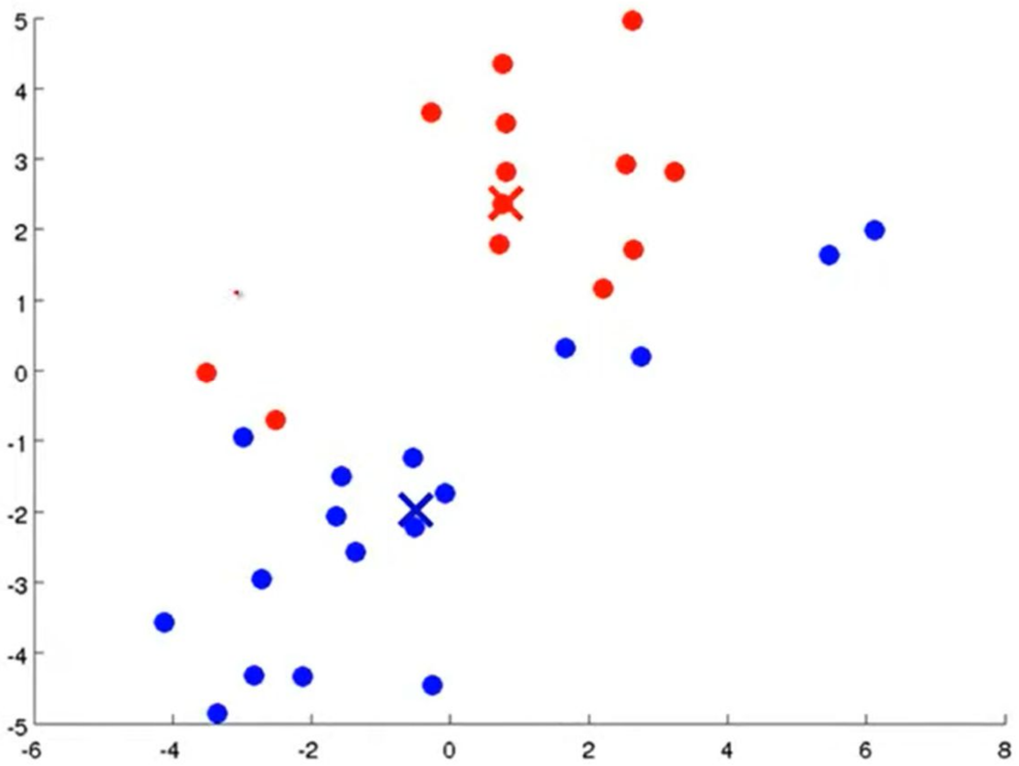
On basis of distance — ED
R.D



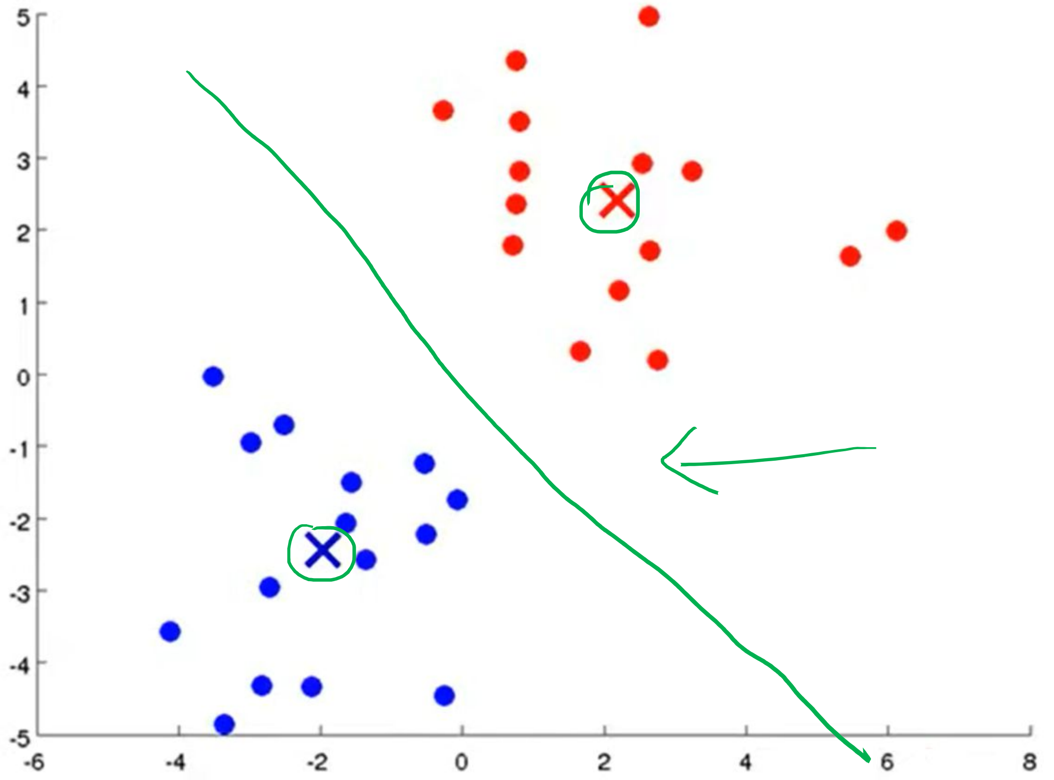
# K-MEANS CLUSTERING



# K-MEANS CLUSTERING



# K-MEANS CLUSTERING



# K-MEANS ALGORITHM

Input:

?

$\boxed{K=2}$

1 - $K$ (number of clusters)

2 - Training set $\{x^{(1)}, x^{(2)}, \ldots, x^{(m)}\}$

$\boxed{\text{data points}}$

$$x^{(i)} \in \mathbb{R}^n$$

# K-MEANS ALGORITHM

$\mu_1$ $\mu_2$

Randomly initialize $K$ cluster centroids $\mu_1, \mu_2, \ldots, \mu_K \in \mathbb{R}^n$

Repeat {

cluster assignment step

for $i$ = 1 to $m$

$c^{(i)}$ := index (from 1 to $K$) of cluster centroid closest to $x^{(i)}$

$\boxed{\min_k \|x^{(i)} - \mu_k\|^2}$

$c^{(i)}$

Move centroid

for $k$ = 1 to $K$

$\mu_k$ := average (mean) of points assigned to cluster $k$

$x^{(1)}, x^{(2)}, x^{(6)}, x^{(7)}$

$$\mu_2 = \frac{1}{4}\left[x^{(1)} + x^{(2)} + x^{(6)} + x^{(7)}\right] \in \mathbb{R}^n$$

}

# EXAMPLE

Divide the given sample data in two (2) clusters using K-Means algorithm using Euclidean Distance.

| Sno. | Height(H) | Weight (W) |
|---|---|---|
| C1 → 1 ✓ | 185 | 72 |
| C2 → 2 ✓ | 170 | 56 |
| → 3 | 168 | 60 |
| → 4 | 179 | 68 |
| 5 | 182 | 72 |
| 6 | 188 | 77 |
| 7 | 180 | 71 |
| 8 | 180 | 70 |
| 9 | 183 | 84 |
| 10 | 180 | 88 |
| 11 | 180 | 67 |
| 12 | 177 | 76 |

Initialize two clusters (C1, C2)

| | H | W | Centroid |
|---|---|---|---|
| C1 | 185 | 72 | (185, 72) |
| C2 | 170 | 56 | (170, 56) |

$$ED = \sqrt{(X_H - H_1)^2 + (X_W - W_1)^2}$$

Observed value

Centroid value

ED of Row 3 (C1) $= \sqrt{(168-185)^2 + (60-72)^2}$

$= \sqrt{289 + 144} = 20.80$

ED of Row 3 (C2) $= \sqrt{(168-170)^2 + (60-56)^2}$

$= \sqrt{4+16} = \sqrt{20} = 4.48$

min (C1, C2)

min (20.80, 4.48) $= \boxed{4.48}$ C2

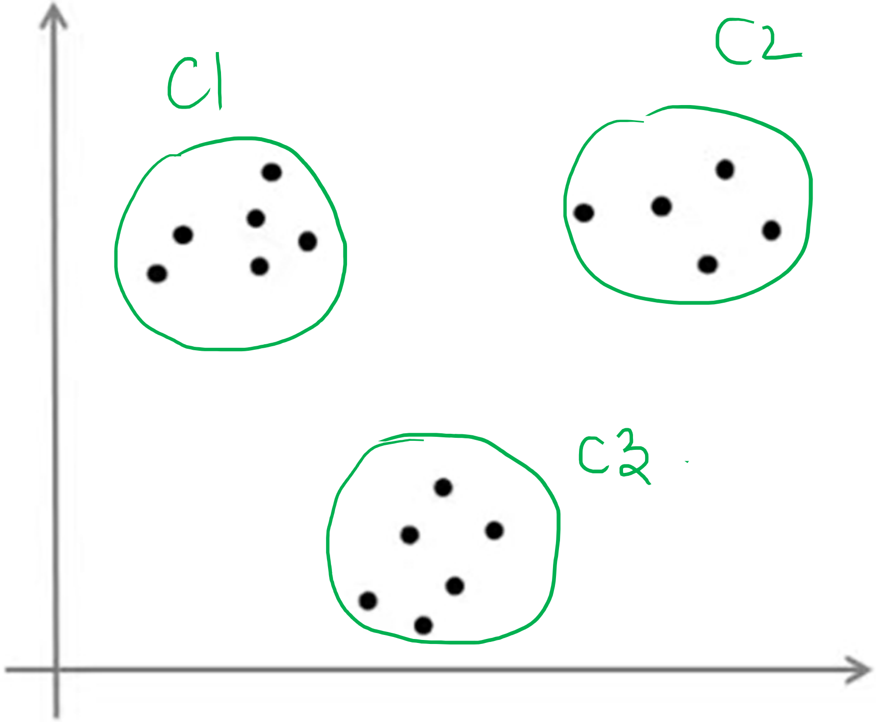
ED of Row 4 (C1) $= \sqrt{(179-185)^2 + (68-72)^2}$

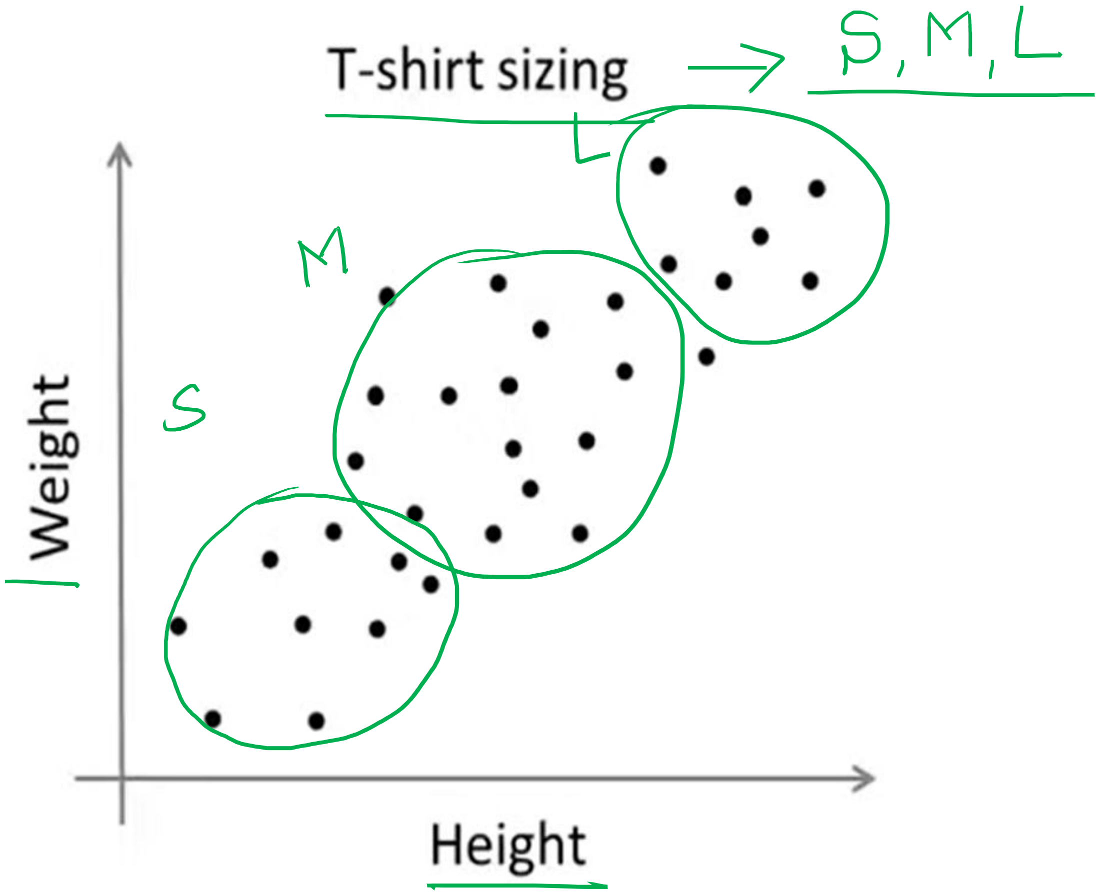
$= \sqrt{52} = 7.2$

ED of Row (4) C2 $= \sqrt{(179-170)^2 + (68-56)^2}$

$= \sqrt{15} = 3.87$

# K-MEANS FOR NON-SEPARATED CLUSTERS



# EXAMPLE

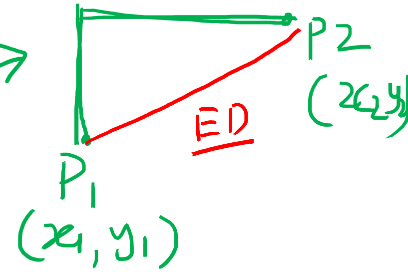
$\boxed{K=3}$

Cluster the following eight points (with (x, y) representing locations) into three clusters A1(2, 10) A2(2, 5) A3(8, 4) A4(5, 8) A5(7, 5) A6(6, 4) A7(1, 2) A8(4, 9).

C1 C2 C3 (Initial)

Initial cluster centers are: A1(2, 10), A4(5, 8) and A7(1, 2).

The distance function between two points $a=(x1, y1)$ and $b=(x2, y2)$ is defined as: $\rho(a, b) = \boxed{|x2 - x1|} + \boxed{|y2 - y1|}$.




Use k-means algorithm to find the three cluster centers after the second iteration.

# SOLUTION

$f(a,b) = |x_1 - x_2| + |y_1 - y_2|$
$= |2-2| + |10-10|$
$= 0$

C2 $|2-5| + |10-8| = 3 + 2 = 5$

C3 $|2-1| + |10-2|$
$= 1 + 8 = 9$

Iteration 1

| | | C1<br>A1 (2, 10) | C2<br>A4 (5, 8) | C3<br>A7 (1, 2) | |
|---|---|---|---|---|---|
| | Point | Dist Mean 1 | Dist Mean 2 | Dist Mean 3 | Cluster |
| 1 A1 | (2, 10) min (0, | 5, | 9) | | C1 |
| 2 A2 | (2, 5) | | | | |
| 3 A3 | (8, 4) | | | | |
| 4 A4 | (5, 8) | | | | |
| 5 A5 | (7, 5) | | | | |
| 6 A6 | (6, 4) | | | | |
| 7 A7 | (1, 2) | | | | |
| 8 A8 | (4, 9) | | | | |

# SOLUTION

| | | (2, 10) | (5, 8) | (1, 2) | |
|---|---|---|---|---|---|
| | **Point** | **Dist Mean 1** | **Dist Mean 2** | **Dist Mean 3** | **Cluster** |
| A1 | (2, 10) | 0 | 5 | 9 | 1 C1 |
| A2 | (2, 5) | | | | |
| A3 | (8, 4) | | | | |
| A4 | (5, 8) | | | | |
| A5 | (7, 5) | | | | |
| A6 | (6, 4) | | | | |
| A7 | (1, 2) | | | | |
| A8 | (4, 9) | | | | |

# SOLUTION

| | Point | (2, 10) Dist Mean 1 | (5, 8) Dist Mean 2 | (1, 2) Dist Mean 3 | **Cluster** |
|---|---|---|---|---|---|
| A1 | (2, 10) | 0 | 5 | 9 | C1 |
| A2 | (2, 5) | 5 | 6 | 4 | C3 |
| A3 | (8, 4) | 12 | 7 | 9 | C2 |
| A4 | (5, 8) | 5 | 0 | 10 | C2 |
| A5 | (7, 5) | 10 | 5 | 9 | C2 |
| A6 | (6, 4) | 10 | 5 | 7 | C2 |
| A7 | (1, 2) | 9 | 10 | 0 | C3 |
| A8 | (4, 9) | 3 | 2 | 10 | C2 |

# SOLUTION

Iteration 1

| | | C1 (2, 10) | C2 (5, 8) | C3 (1, 2) | |
|---|---|---|---|---|---|
| | **Point** | **Dist Mean 1** | **Dist Mean 2** | **Dist Mean 3** | **Cluster** |
| A1 | (2, 10) | 0 | 5 | 9 | 1 |
| A2 | (2, 5) | 5 | 6 | 4 | 3 |
| A3 | (8, 4) | 12 | 7 | 9 | 2 |
| A4 | (5, 8) | 5 | 0 | 10 | 2 |
| A5 | (7, 5) | 10 | 5 | 9 | 2 |
| A6 | (6, 4) | 10 | 5 | 7 | 2 |
| A7 | (1, 2) | 9 | 10 | 0 | 3 |
| A8 | (4, 9) | 3 | 2 | 10 | 2 |

| Cluster 1 | Cluster 2 | Cluster 3 |
|---|---|---|
| (2, 10) | (8, 4) | (2, 5) |
| | (5, 8) | (1, 2) |
| | (7, 5) | |
| | (6, 4) | |
| | (4, 9) | |

# SOLUTION

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.
- For Cluster 1, we only have one point A1(2, 10), which was the old mean, so the cluster center remains the same.
- For Cluster 2, we have ( (8+5+7+6+4)/5, (4+8+5+4+9)/5 ) = (6, 6)
- For Cluster 3, we have ( (2+1)/2, (5+2)/2 ) = (1.5, 3.5)

# SOLUTION

| | | (2, 10) | (6, 6) | (1.5, 3.5) | |
|---|---|---|---|---|---|
| | Point | Dist Mean 1 | Dist Mean 2 | Dist Mean 3 | **Cluster** |
| A1 | (2, 10) | 0 | 8 | 7 | 1 |
| A2 | (2, 5) | 5 | 5 | 2 | 3 |
| A3 | (8, 4) | 12 | 4 | 7 | 2 |
| A4 | (5, 8) | 5 | 3 | 8 | 2 |
| A5 | (7, 5) | 10 | 2 | 7 | 2 |
| A6 | (6, 4) | 10 | 2 | 5 | 2 |
| A7 | (1, 2) | 9 | 9 | 2 | 3 |
| A8 | (4, 9) | 3 | 5 | 8 | 1 |

# SOLUTION

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.

- In Cluster 1, we have points 1 and 8. Therefore the centroid is: $((2+4)/2,(10+9)/2)=(3,9.5)$
- In Cluster 2, we have points 3,4,5 and 6. Therefore, the centroid is: $((8+5+7+6)/4,(4+8+5+4)/4)=(6.5,5.25)$
- For Cluster 3, we have points 2 and 7. Therefore, the centroid is: $((2+1)/2,(5+2)/2)=(1.5,3.5)$

# SOLUTION

| | | (3, 9.5) | (6.5 ,5.25) | (1.5, 3.5) | |
|---|---|---|---|---|---|
| | Point | Dist Mean 1 | Dist Mean 2 | Dist Mean 3 | **Cluster** |
| A1 | (2, 10) | 1.5 | 9.25 | 7 | **1** |
| A2 | (2, 5) | 5.5 | 4.75 | 2 | **3** |
| A3 | (8, 4) | 10.5 | 2.75 | 7 | **2** |
| A4 | (5, 8) | 3.5 | 4.25 | 8 | **1** |
| A5 | (7, 5) | 8.5 | 0.75 | 7 | **2** |
| A6 | (6, 4) | 8.5 | 1.75 | 5 | **2** |
| A7 | (1, 2) | 9.5 | 8.75 | 2 | **3** |
| A8 | (4, 9) | 1.5 | 6.25 | 8 | **1** |

# SOLUTION

- Next, we need to re-compute the new cluster centers (means). We do so, by taking the mean of all points in each cluster.

- In Cluster 1, we have points 1, 4, and 8. Therefore the centroid is: $((2+5+4)/2,(10+8+9)/2)=(3.67,9)$
- In Cluster 2, we have points 3,5 and 6. Therefore, the centroid is: $((8+7+6)/4,(4+5+4)/4)=(7,4.3)$
- For Cluster 3, we have points 2 and 7. Therefore, the centroid is: $((2+1)/2,(5+2)/2)=(1.5,3.5)$

# SOLUTION

| | | (3.67, 9) | (7 ,4.3) | (1.5, 3.5) | |
|---|---|---|---|---|---|
| | Point | Dist Mean 1 | Dist Mean 2 | Dist Mean 3 | **Cluster** |
| A1 | (2, 10) | 2.67 | 10.7 | 7 | **1** |
| A2 | (2, 5) | 5.67 | 5.7 | 2 | **3** |
| A3 | (8, 4) | 9.33 | 1.3 | 7 | **2** |
| A4 | (5, 8) | 2.33 | 5.7 | 8 | **1** |
| A5 | (7, 5) | 7.33 | 0.7 | 7 | **2** |
| A6 | (6, 4) | 7.33 | 1.3 | 5 | **2** |
| A7 | (1, 2) | 9.67 | 8.3 | 2 | **3** |
| A8 | (4, 9) | 0.33 | 7.7 | 8 | **1** |

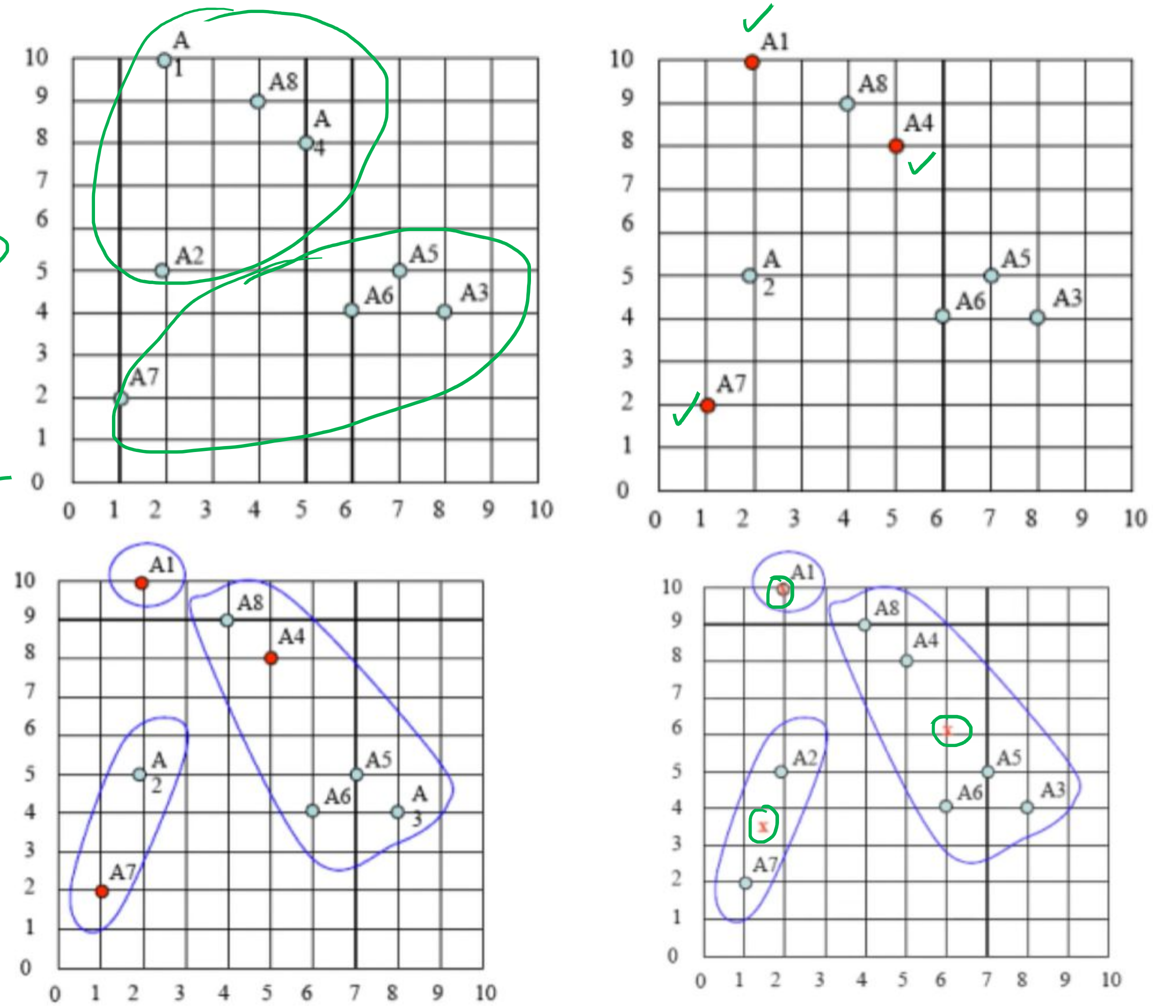
# VISUALIZATION OF SOLUTION

× K=1

× K=2

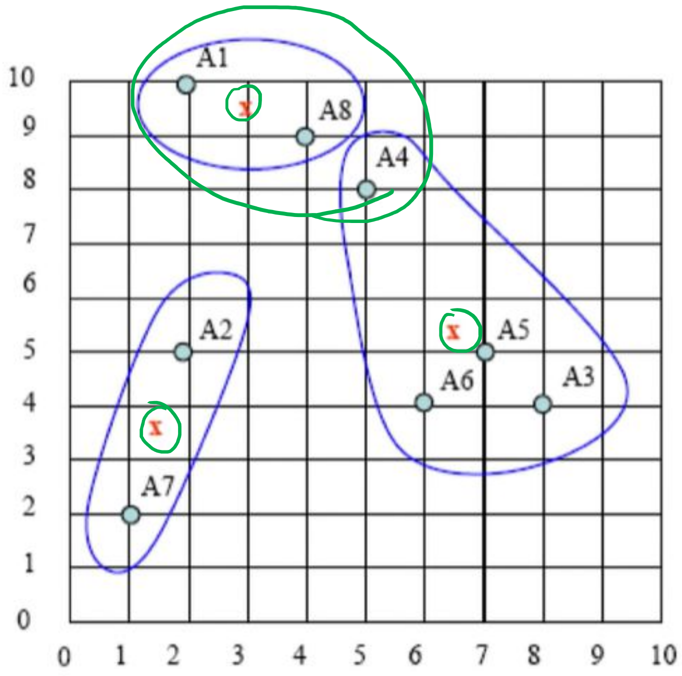
✓ K=3

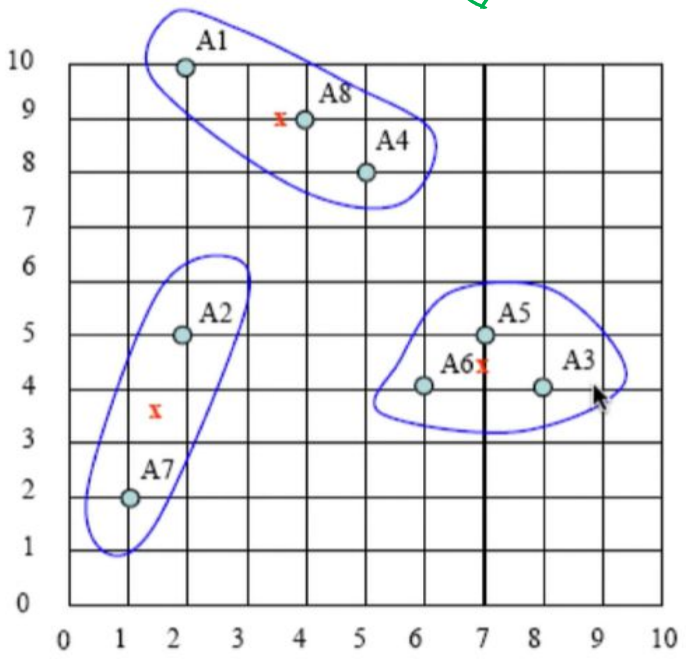
K=10



# VISUALIZATION OF SOLUTION

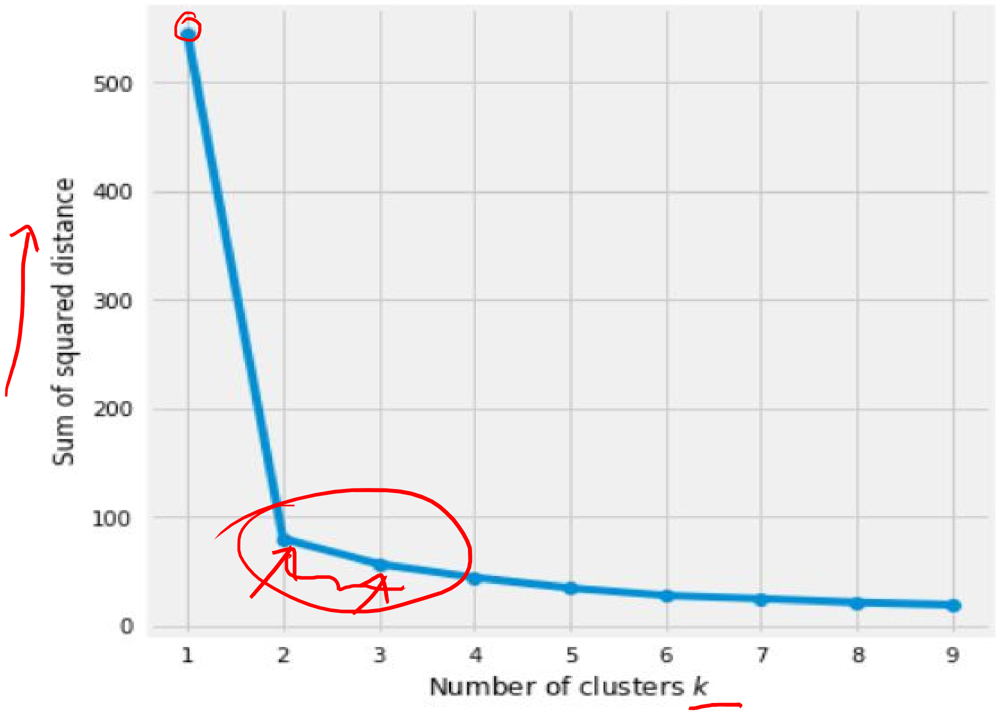
## Next two Iterations



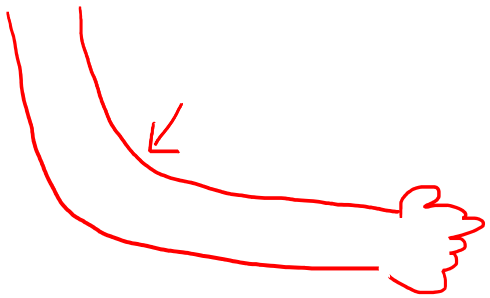
# HOW TO CHOOSE K: ELBOW METHOD

It is based on the sum of squared distance (SSE) between data points and their assigned clusters' centroids.




n data points

K = n



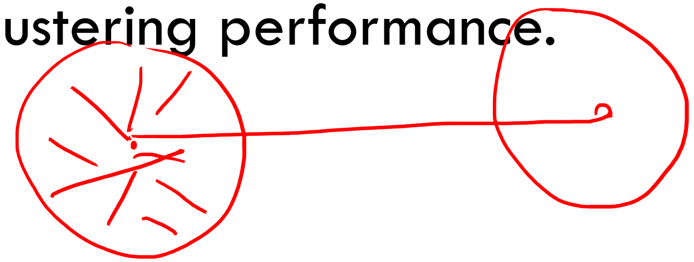
# DRAWBACKS

- K-means algorithm is good in capturing structure of the data if clusters have a spherical-like shape.
- It always try to construct a nice spherical shape around the centroid.
- The clusters which have a complicated geometric shapes, k-means does a poor job in clustering the data.

# CLUSTERING QUALITY

Ideal clustering is characterized by minimal intra cluster distance and maximal inter cluster distance.

There are majorly two types of measures to assess the clustering performance.



(i) *Extrinsic Measures* which require ground truth labels.

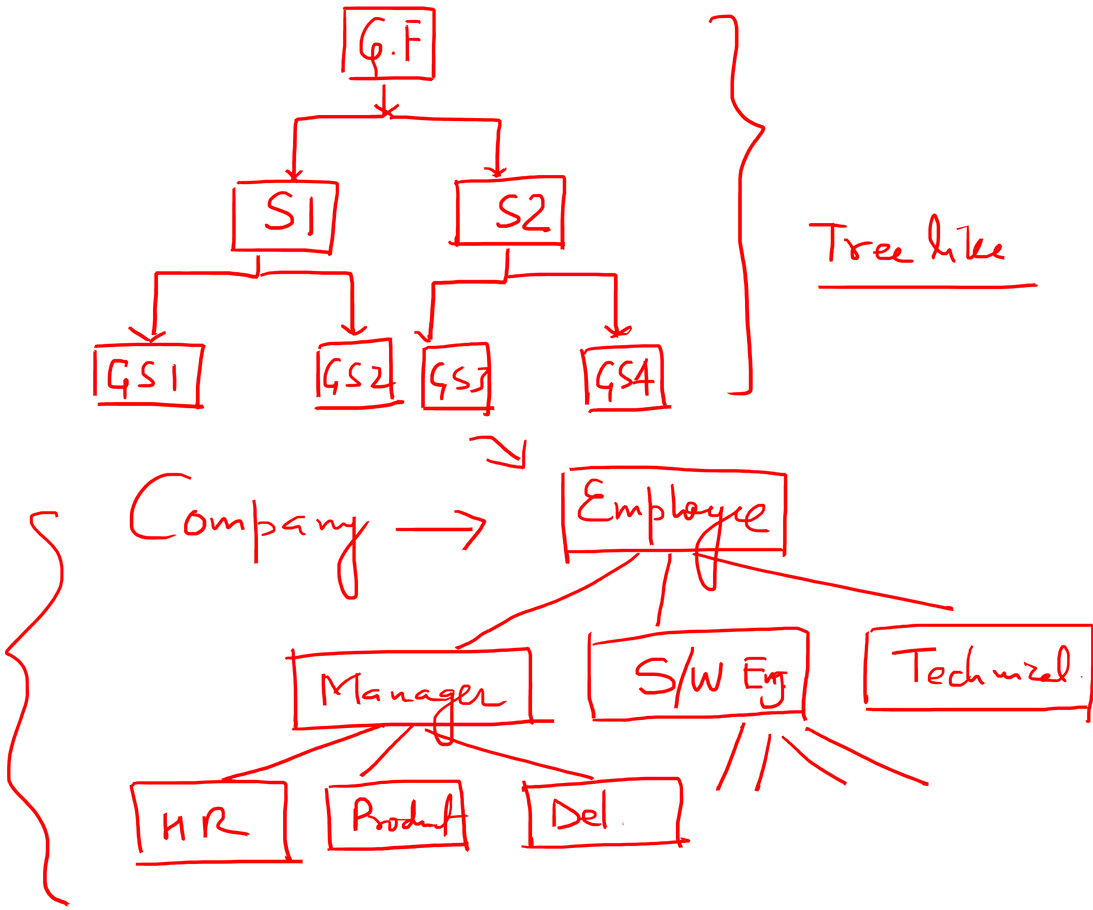
- Rand index

(ii) *Intrinsic Measures* that does not require ground truth labels.

- Silhouette Coefficient

# BASIC CLUSTERING METHODS

1. **Partitioning methods**
   - *k-means*
2. **Hierarchical methods**
   - *Agglomerative (bottom-up) or divisive (top-down)*
3. **Density-based methods**
4. **Grid-based methods**



# OVERVIEW OF CLUSTERING METHODS

| Method | General Characteristics |
|---|---|
| Partitioning methods<br>K-Means | – Find mutually exclusive clusters of spherical shape<br>– Distance-based<br>– May use mean or medoid (etc.) to represent cluster center<br>– Effective for small- to medium-size data sets |
| Hierarchical methods | – Clustering is a hierarchical decomposition (i.e., multiple levels)<br>– Cannot correct erroneous merges or splits<br>– May incorporate other techniques like microclustering or consider object "linkages" |
| Density-based methods | – Can find arbitrarily shaped clusters<br>– Clusters are dense regions of objects in space that are separated by low-density regions<br>– Cluster density: Each point must have a minimum number of points within its "neighborhood"<br>– May filter out outliers |
| Grid-based methods | – Use a multiresolution grid data structure<br>– Fast processing time (typically independent of the number of data objects, yet dependent on grid size) |

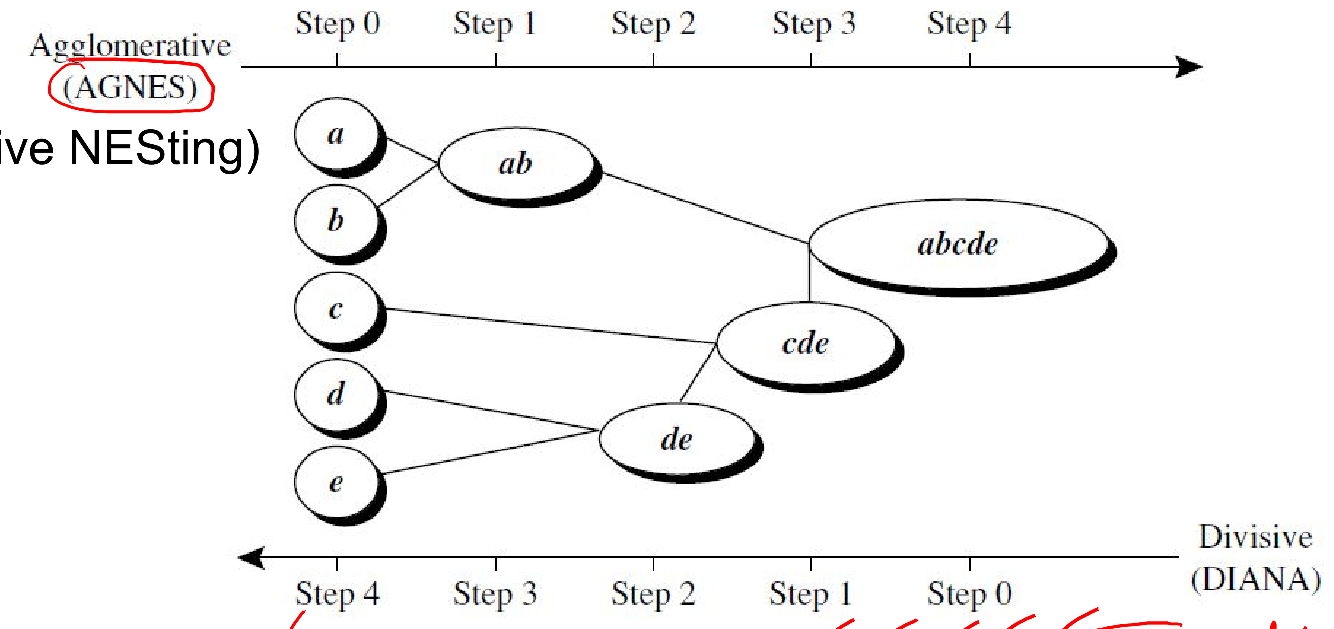
# HIERARCHICAL METHODS

- A **hierarchical clustering method** works by grouping data objects into a hierarchy or "tree" of clusters.
- Representing data objects in the form of a hierarchy is useful for data summarization and visualization.
- Agglomerative methods start with individual objects as clusters, which are iteratively merged to form larger clusters.
- Divisive methods initially let all the given objects form one cluster, which they iteratively split into smaller clusters.
- Hierarchical clustering methods can encounter difficulties regarding the selection of merge or split points.

# AGGLOMERATIVE VERSUS DIVISIVE HIERARCHICAL CLUSTERING

A hierarchical clustering method can be either *agglomerative* or *divisive*, depending on whether the hierarchical decomposition is formed in a bottom-up (merging) or top down (splitting) fashion.

**AGNES** (AGglomerative NESting)



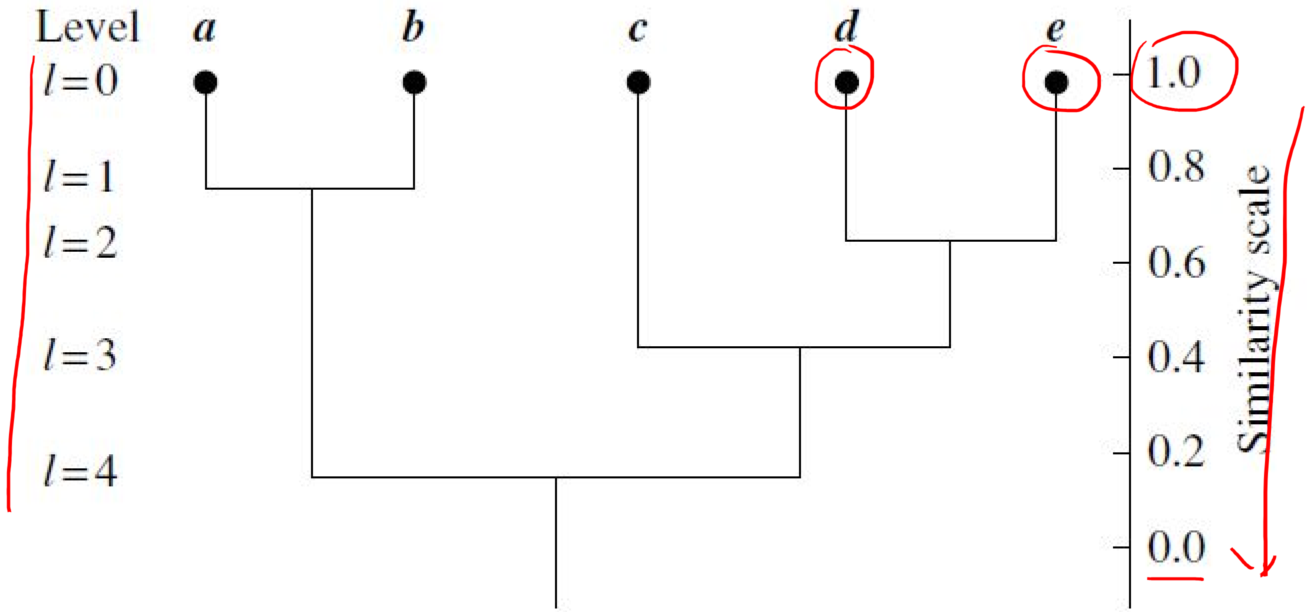

**DIANA** (DIvisive ANAlysis)

data objects {***a***, ***b***, ***c***, ***d***, ***e***}.

5 datapoint

# AGGLOMERATIVE VERSUS DIVISIVE HIERARCHICAL CLUSTERING

A tree structure called a **dendrogram** is commonly used to represent the process of hierarchical clustering.

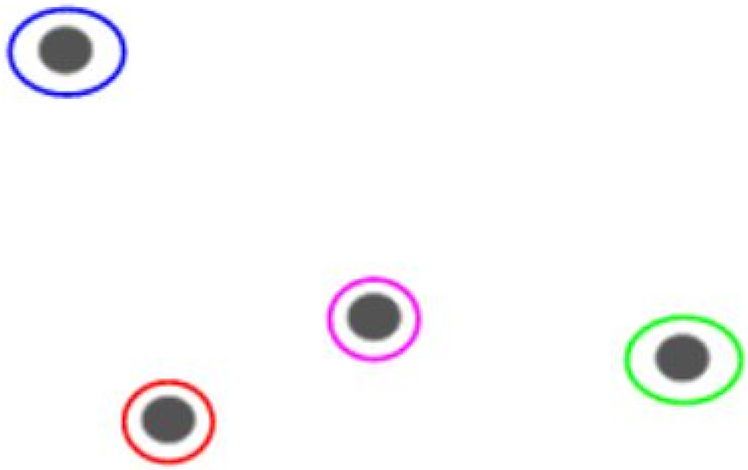



Dendrogram representation for hierarchical clustering of data objects {***a***, ***b***, ***c***, ***d***, ***e***}.

# AGGLOMERATIVE HIERARCHICAL CLUSTERING

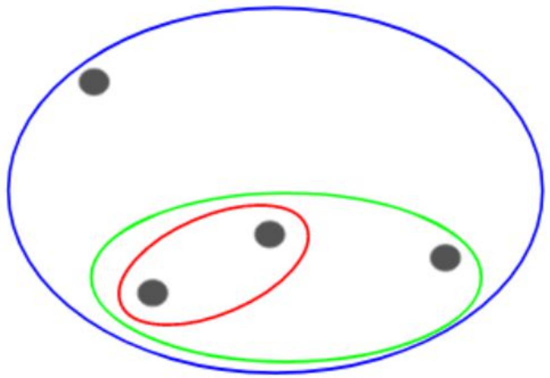
We assign each point to an individual cluster in this technique.

Suppose there are 4 data points. We will assign each of these points to a cluster and hence will have 4 clusters in the beginning:



# AGGLOMERATIVE HIERARCHICAL CLUSTERING

Then, at each iteration, we merge the closest pair of clusters and repeat this step until only a single cluster is left:



We are merging (or adding) the clusters at each step.

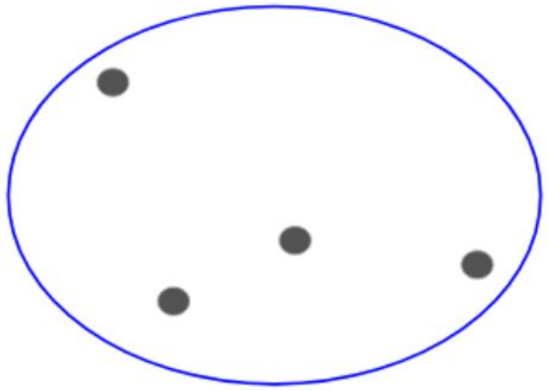
Hence, this type of clustering is also known as **additive hierarchical clustering.**

# DIVISIVE HIERARCHICAL CLUSTERING

Divisive hierarchical clustering works in the opposite way.

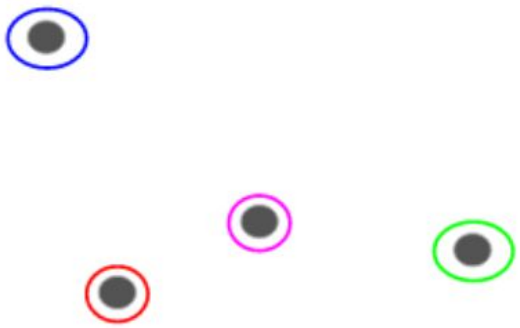
Instead of starting with n clusters (in case of n observations), we start with a single cluster and assign all the points to that cluster.

So, it doesn't matter if we have 10 or 1000 data points. All these points will belong to the same cluster at the beginning:



# DIVISIVE HIERARCHICAL CLUSTERING

Now, at each iteration, we split the farthest point in the cluster and repeat this process until each cluster only contains a single point:



We are splitting (or dividing) the clusters at each step, hence the name divisive hierarchical clustering.

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

We merge the most similar points or clusters in hierarchical clustering – we know this. Now the question is –

How do we decide which points are similar and which are not

?

**Distance-based algorithm**

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

In hierarchical clustering, we have a concept called a **proximity matrix**. This stores the distances between each point.

Suppose a teacher wants to divide her students into different groups. She has the marks scored by each student in an assignment and based on these marks, she wants to segment them into groups. There's no fixed target here as to how many groups to have. Since the teacher does not know what type of students should be assigned to which group, it cannot be solved as a supervised learning problem. So, we will try to apply hierarchical clustering here and segment the students into different groups.

Let's take a sample of 5 students:

| Student_ID | Marks |
|---|---|
| 1 | 10 |
| 2 | 7 |
| 3 | 28 |
| 4 | 20 |
| 5 | 35 |

# CREATING A PROXIMITY MATRIX

Let's make the 5 x 5 proximity matrix for our example:

| Student_ID | Marks |
|---|---|
| 1 | 10 |
| 2 | 7 |
| 3 | 28 |
| 4 | 20 |
| 5 | 35 |

$d_{(1,1)} = \sqrt{(10-10)^2}$
$= 0$

$d_{(1,2)} = \sqrt{(10-7)^2}$
$= \sqrt{3^2}$
$= 3$

$d(2,1) = 3$

$d(1,3) = \sqrt{(10-28)^2}$
$= 18$

| ID | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 3 | 18 | 10 | 25 |
| 2 | 3 | 0 | 21 | 13 | 28 |
| 3 | 18 | 21 | 0 | 8 | 7 |
| 4 | 10 | 13 | 8 | 0 | 15 |
| 5 | 25 | 28 | 7 | 15 | 0 |

Min – 3

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

**Step 1:** First, we assign all the points to an individual cluster


**Step 2:** Next, we will look at the smallest distance in the proximity matrix and merge the points with the smallest distance. We then update the proximity matrix:

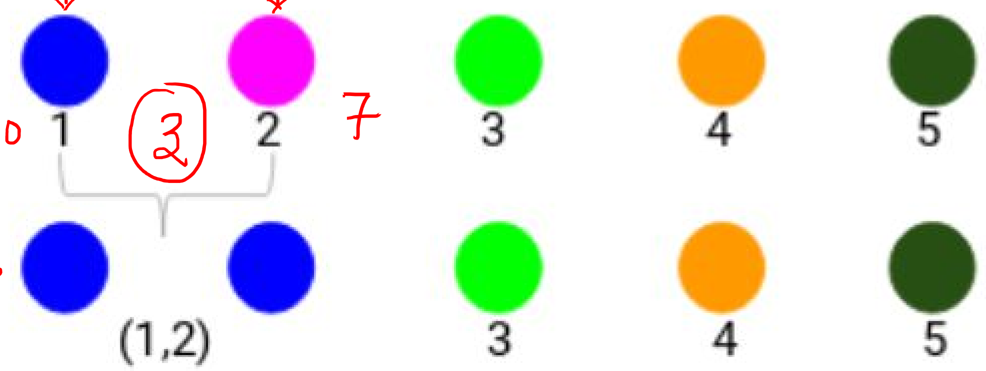
| ID | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| 1 | 0 | 3 | 18 | 10 | 25 |
| 2 | 3 | 0 | 21 | 13 | 28 |
| 3 | 18 | 21 | 0 | 8 | 7 |
| 4 | 10 | 13 | 8 | 0 | 15 |
| 5 | 25 | 28 | 7 | 15 | 0 |

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

Here, the smallest distance is 3 and hence we will merge point 1 and 2.

Max(10,7)
Min
Avg.

Marks 10




3

7

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

Let's look at the updated clusters and accordingly update the proximity matrix:

| Student_ID | Marks |
|---|---|
| 1. (1,2) | 10 |
| 2. 3 | 28 |
| 3. 4 | 20 |
| 4. 5 | 35 |

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

Now, we will again calculate the proximity matrix for these clusters:

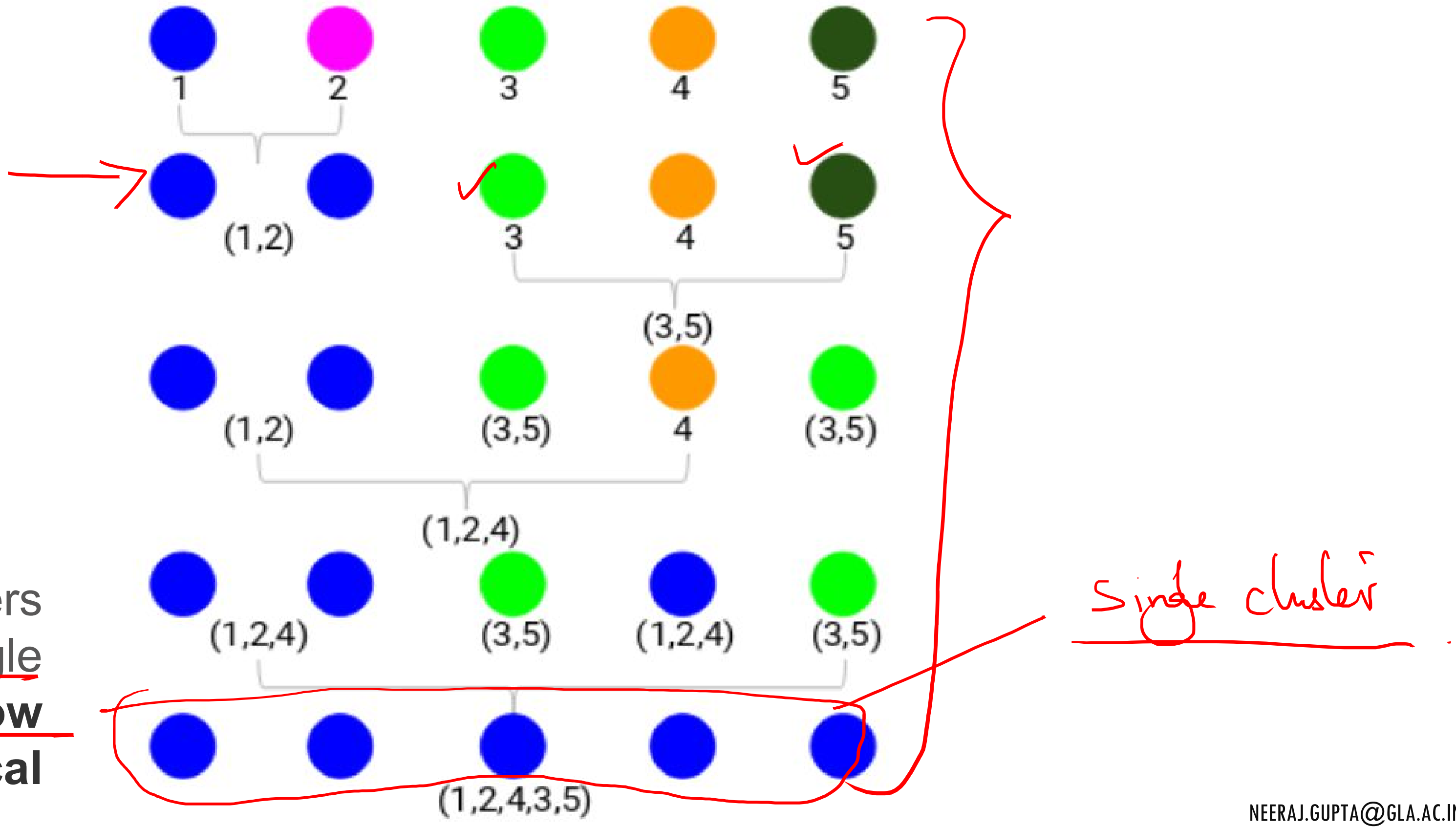
(1,2) — 10
3 — 28

$\sqrt{(10-28)^2}$
$= 18$

| ID | (1,2) | 3 | 4 | 5 |
|---|---|---|---|---|
| (1,2) | 0 | 18 | 10 | 25 |
| 3 | 18 | 0 | 8 | 7 |
| 4 | 10 | 8 | 0 | 15 |
| 5 | 25 | 7 | 15 | 0 |

# STEPS TO PERFORM HIERARCHICAL CLUSTERING

**Step 3:** We will repeat step 2 until only a single cluster is left.

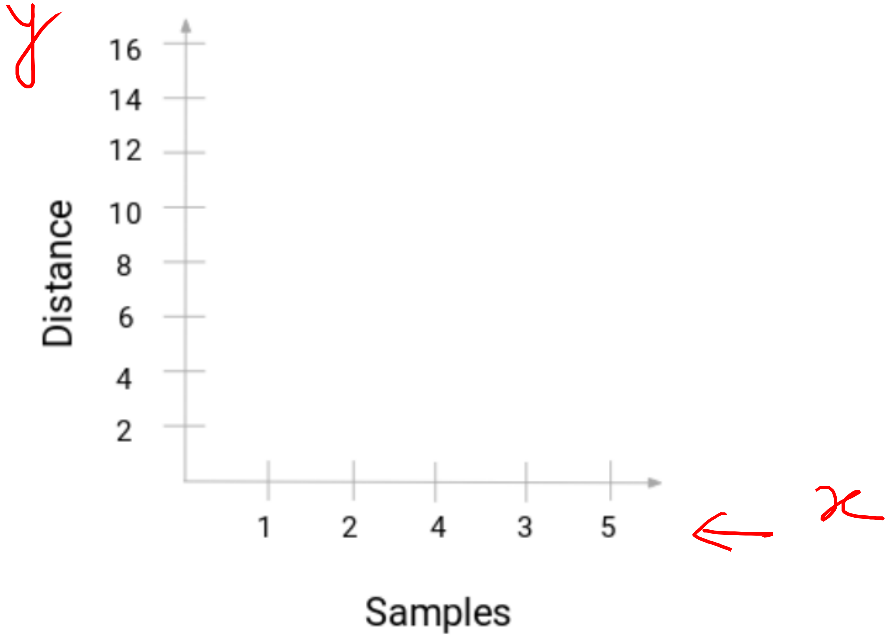



We started with 5 clusters and finally have a single cluster. **This is how agglomerative hierarchical clustering works**.

# HOW SHOULD WE CHOOSE THE NUMBER OF CLUSTERS IN HIERARCHICAL CLUSTERING?

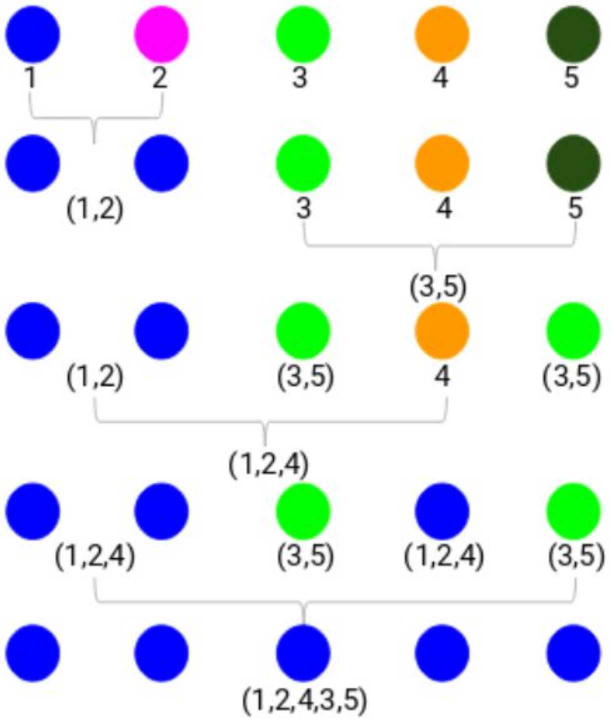
A dendrogram is a tree-like diagram that records the sequences of merges or splits.

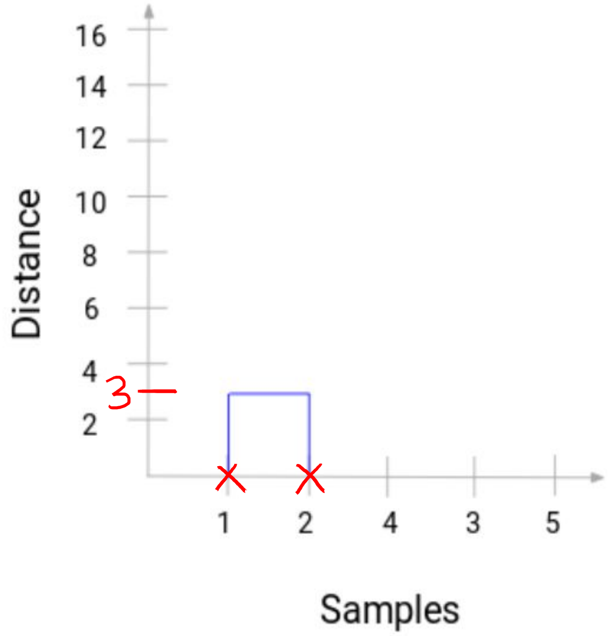
Let's see how a dendrogram looks like:



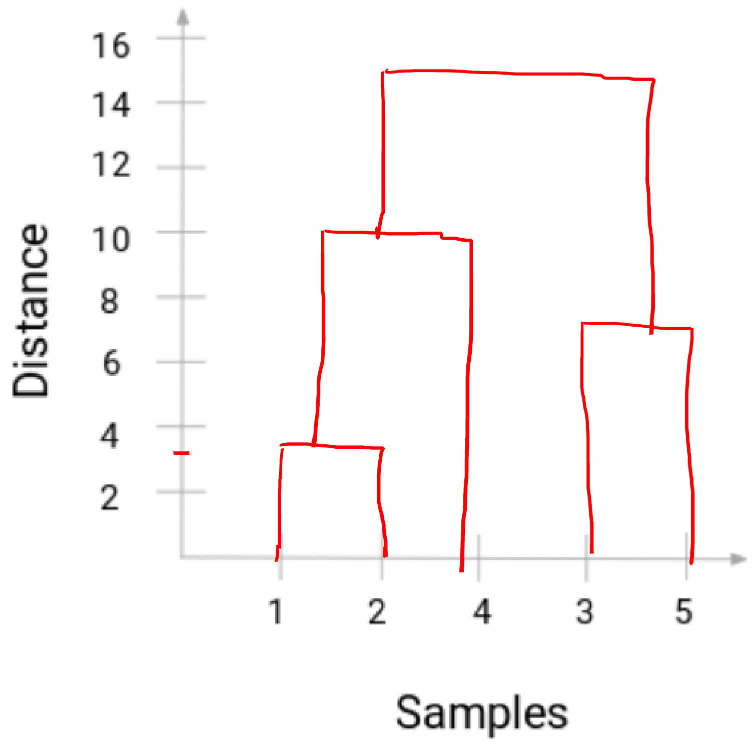
# HOW SHOULD WE CHOOSE THE NUMBER OF CLUSTERS IN HIERARCHICAL CLUSTERING?

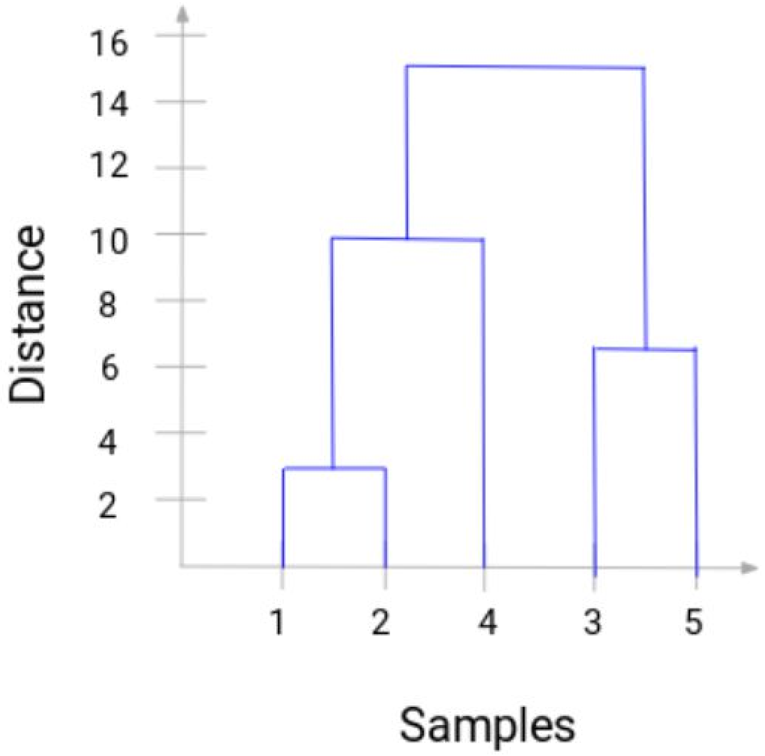
**Whenever two clusters are merged, we will join them in this dendrogram and the height of the join will be the distance between these points.**



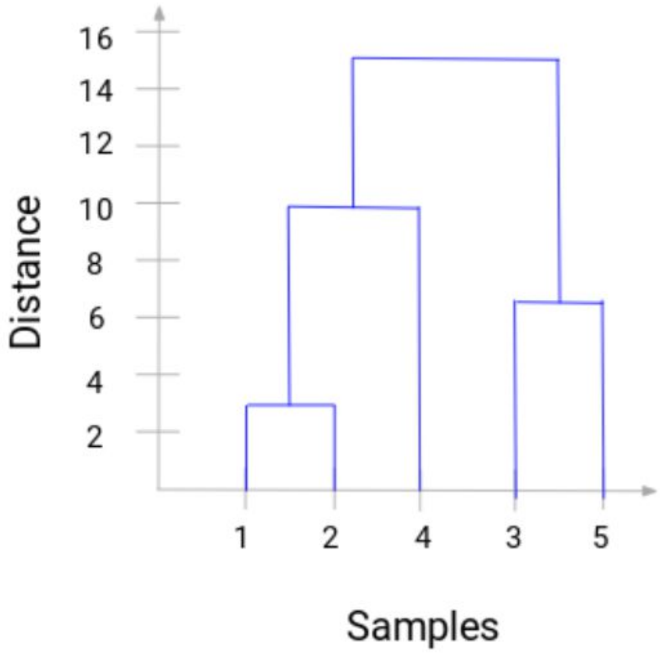
# HOW SHOULD WE CHOOSE THE NUMBER OF CLUSTERS IN HIERARCHICAL CLUSTERING?

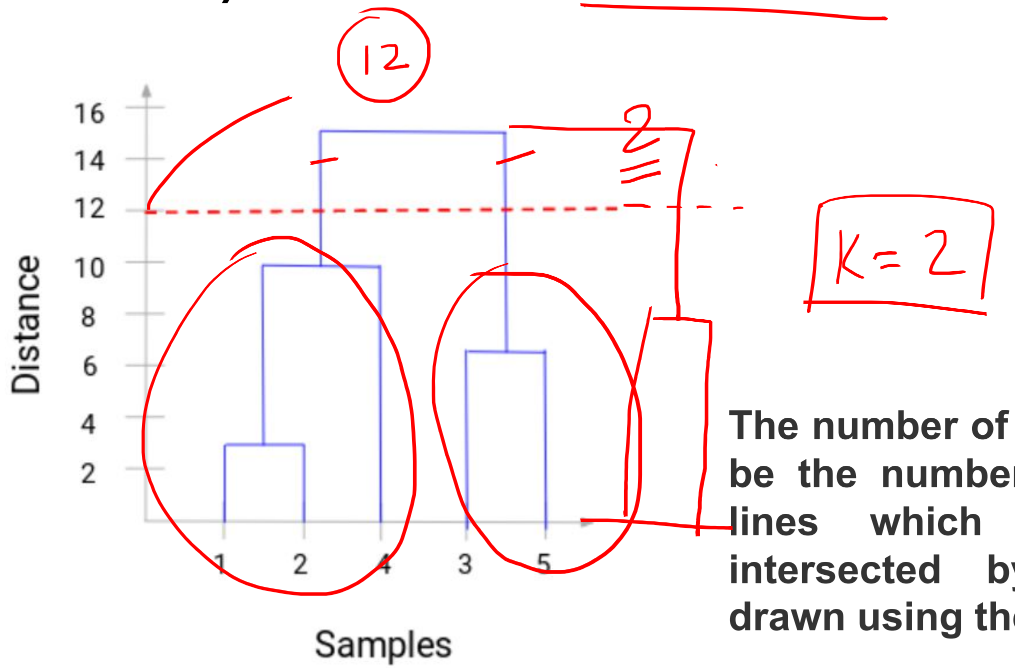


# HOW SHOULD WE CHOOSE THE NUMBER OF CLUSTERS IN HIERARCHICAL CLUSTERING?

**More the distance of the vertical lines in the dendrogram, more the distance between those clusters.** Now, we can set a threshold distance and draw a horizontal line (*Generally, we try to set the threshold in such a way that it cuts the tallest vertical line*)




**The number of clusters will be the number of vertical lines which are being intersected by the line drawn using the threshold.**

# THANKS


Keep Learning
Keep Growing


Dr. Neeraj Gupta
Assistant Professor, Dept. of CEA
neeraj.gupta@gla.ac.in