# Supervised and Unsupervised learning

Source: http://dataaspirant.com/2014/09/19/supervised-and-unsupervised-learning/

# Supervised and Unsupervised learning

Supervised learning is a data mining task of inferring a function from labeled training data. The training data consist of a set of training examples. Supervised learning is a data mining task of inferring a function from labeled training data.
The training data consist of a set of training examples.

**Supervised learning**: Learning from the know label data to create a model then predicting target class for the given input data.

**Unsupervised learning**: Learning from the unlabeled data to differentiating the given input data

**Real-Life EXAMPLE**
Task to arrange collection of fruits
**Supervised Learning:**

- From previous work we know the physical characters of fruits
- In data mining terminology the earlier work is called as training the data. You already learn the things
- from your train data. This is because of response variable which is a decision variable

| No. | SIZE | COLOR | SHAPE | FRUIT NAME (response variable) |
|---|---|---|---|---|
| 1 | Big | Red | Rounded shape with depression at the top | Apple |
| 2 | Small | Red | Heart-shaped to nearly globular | Cherry |
| 3 | Big | Green | Long curving cylinder | Banana |
| 4 | Small | Green | Round to oval,Bunch shape Cylindrical | Grape |

# Supervised Learning Algorithms:

All classification and regression algorithms come under supervised learning.

Logistic Regression

Decision trees

Support vector machine (SVM)

k-Nearest Neighbours

Naive Bayes

Random forest

Linear regression

polynomial regression

SVM for regression

# Unsupervised Learning:

This time, you don’t know anything about the fruits, this is the first time you have seen them. as You have no clue about those.
You will take a fruit and you will arrange them by considering the physical character of that particular fruit.

Suppose you have considered color.
Then the groups will be something like this.

**RED COLOR GROUP: apples & cherry fruits.**
**GREEN COLOR GROUP: bananas & grapes.**

So now you will take another physical character such as size.

**RED COLOR AND BIG SIZE: apple.**
**RED COLOR AND SMALL SIZE: cherry fruits.**
**GREEN COLOR AND BIG SIZE: bananas.**
**GREEN COLOR AND SMALL SIZE: grapes.**

The job has done,.
Here you did not learn anything before ,means no train data and no response variable.
In data mining or machine learning, this kind of learning is known as unsupervised learning.

# Unsupervised learning algorithms:

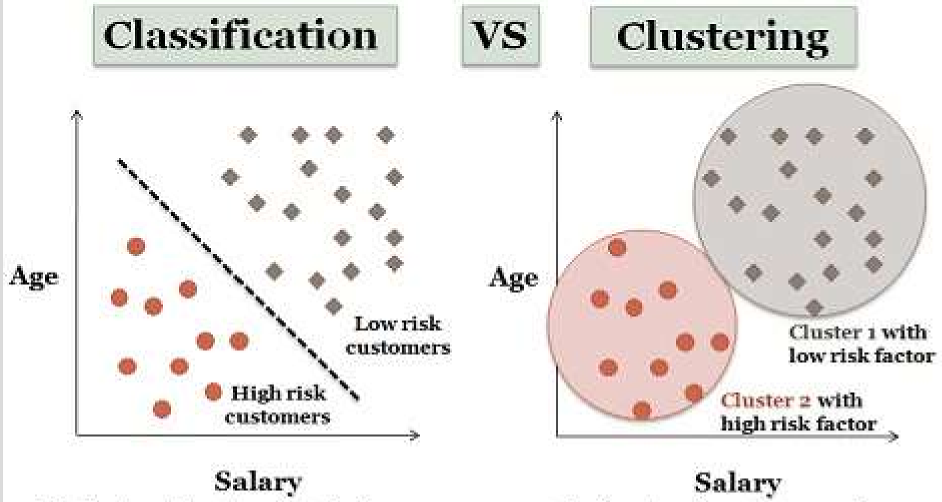
All clustering algorithms come under unsupervised learning algorithms.

- K – means clustering
- Hierarchical clustering
- Hidden Markov models

# Classification (वर्गीकरण) vs clustering (समूहीकरण)

- Classification and Clustering are the two types of learning methods which characterize objects into groups by one or more features.

| BASIS FOR COMPARISON | CLASSIFICATION | CLUSTERING |
|---|---|---|
| Basic | This model function classifies the data into one of numerous already defined definite classes. | This function maps the data into one of the multiple clusters where the arrangement of data items is relies on the similarities between them. |
| Involved in | Supervised learning | Unsupervised learning |
| Training sample | Labeled data is provided. | Unlabelled data provided. |



Risk classification for the loan payees on the basis of customer salary

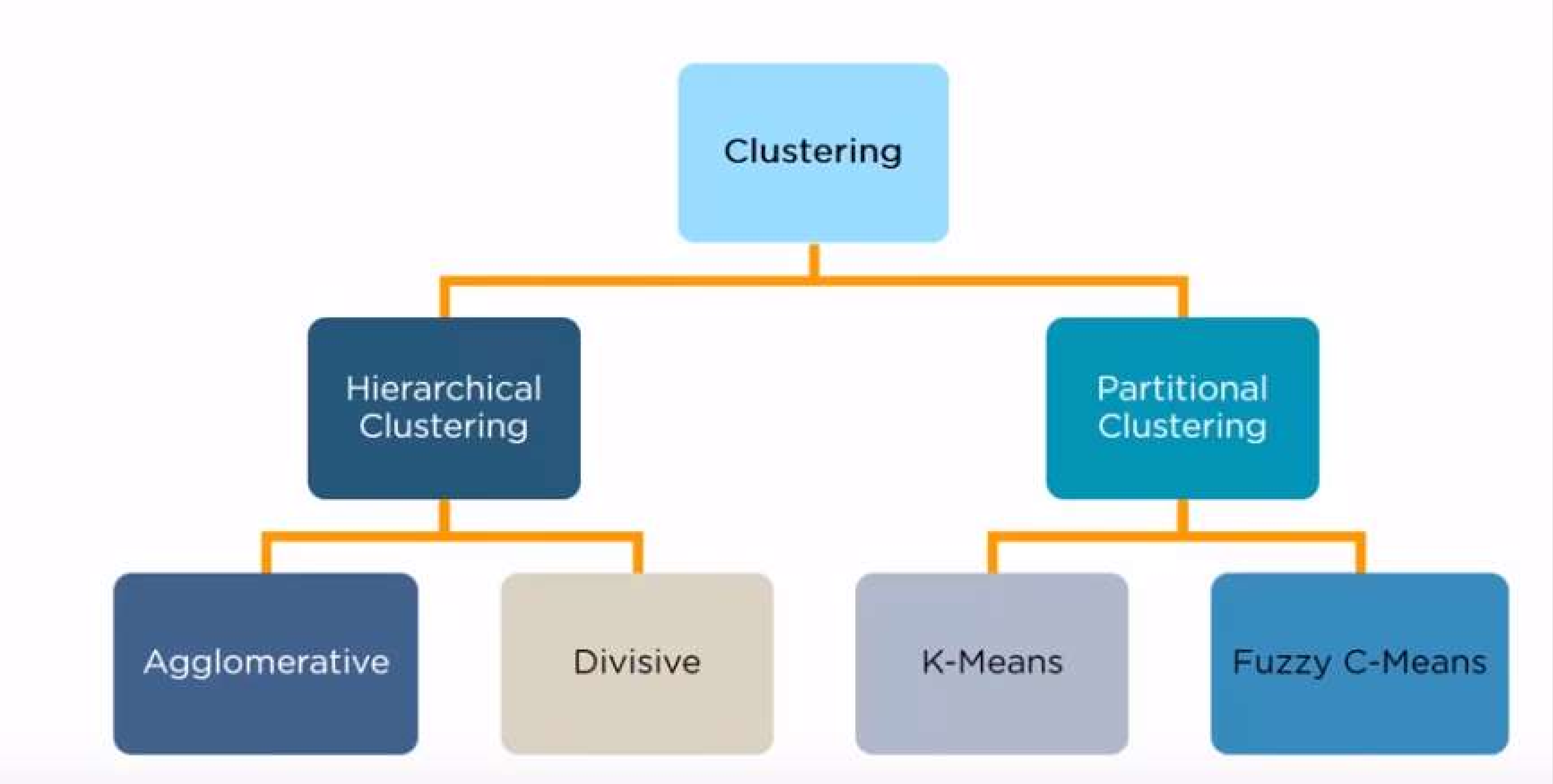
## Key Differences Between Classification and Clustering

1.Classification is the process of classifyin the data with the help of class labels. On the other hand, Clustering is similar to classification but there are no predefined class labels.

2.Classification is geared with supervised learning. As against, clustering is also known as unsupervised learning.

3.Training sample is provided in classification method while in case of clustering training data is not provided.

# Types of clustering algorithms: A simple classification



A cluster refers to a collection of data points aggregated together because of certain similarities.

K-means - division of objects into clusters such that each object is in exactly one cluster, not several

Fuzzy means clustering - division of objects into cluster such that each object belong to multiple cluster

# Clustering: Types

- Clustering can be broadly divided into two subgroups:

- **Hard clustering**: in hard clustering, each data object or point either belongs to a cluster
- **Soft clustering**: in soft clustering, a data point can belong to more than one cluster with some probability or likelihood value.

# Type of Clustering Algorithms: a rigorous classification

- Clustering algorithms can be categorized into 4 classes -

1) **Connectivity-based clustering**: the main idea behind this clustering is that data points that are closer in the data space are more related (similar) than to data points farther away. The clusters are formed by connecting data points according to their distance. At different distances, different clusters will form and can be represented using a dendrogram, which gives away why they are also commonly called "hierarchical clustering". These methods do not produce a unique partitioning of the dataset, rather a **hierarchy from which the user still needs to choose appropriate clusters by choosing the level where they want to cluster**. They are also not very robust towards outliers, which might show up as additional clusters or even cause other clusters to merge.

2) **Centroid-based clustering/Partition clustering**: in this type of clustering, clusters are represented by a central vector or a centroid. This centroid might not necessarily be a member of the dataset. This is an iterative clustering algorithms in which the notion of similarity is derived by how close a data point is to the centroid of the cluster. **k-means** is a centroid based clustering,

3) **Distribution-based clustering**: this clustering is very closely related to statistics: distributional modeling. Clustering is based on the notion of how probable is it for a data point to belong to a certain distribution, such as the Gaussian distribution, for example. Data points in a cluster belong to the same distribution. These models have a strong theoritical foundation, however they often suffer from overfitting. Gaussian mixture models, using the **Fuzzy C-means , expectation-maximization** algorithm is a famous distribution based clustering method.

4) **Density-based methods** search the data space for areas of varied density of data points. Clusters are defined as areas of higher density within the data space compared to other regions. Data points in the sparse areas are usually considered to be noise and/or border points. The drawback with these methods is that they expect some kind of density guide or parameters to detect cluster borders. DBSCAN and OPTICS are some prominent density based clustering. [Source of this slide – Datacamp]

- Source of this slide: https://www.datacamp.com/community/tutorials/k-means-clustering-r

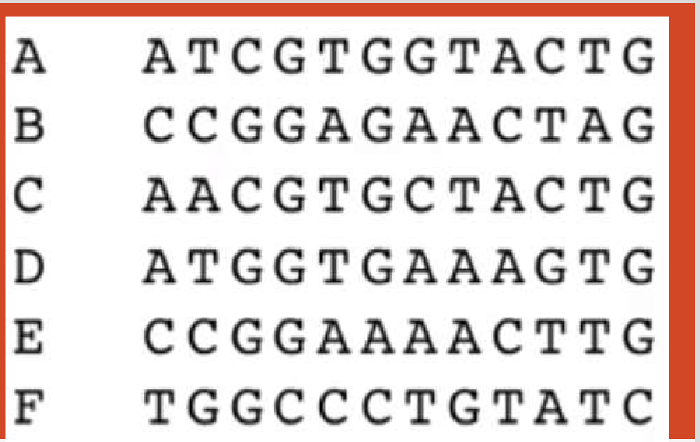
# Hierarchical (श्रेणीबद्ध) Clustering

- There are two type of hierarchical clustering approaches –
- **Agglomerative** - "Bottom up approach" assume all members as a separate cluster, then merge them in to larger and larger groups for form a tree shaped structure (dendrogram)
- **Divisive** - "Top down approach begin with the whole set and proceed to divide

# Hierarchical Clustering: Steps

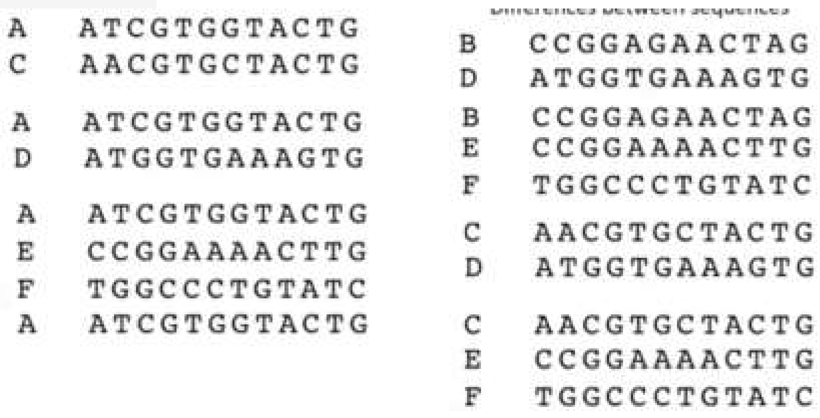
1. It starts by calculating the distance between every pair of observation points and store it in a distance matrix.
2. It then puts every point in its own cluster.
3. Then it starts merging the closest pairs of points based on the distances from the distance matrix and as a result the amount of clusters goes down by 1.
4. Then it recomputes the distance between the new cluster and the old ones and stores them in a new distance matrix.
5. Lastly it repeats steps 2 and 3 until all the clusters are merged into one single cluster.

# Hands-on Exercise: Cluster the following genes using Hierarchical clustering



| | |
|---|---|
| A | ATCGTGGTACTG |
| B | CCGGAGAACTAG |
| C | AACGTGCTACTG |
| D | ATGGTGAAAGTG |
| E | CCGGAAAACTTG |
| F | TGGCCCTGTATC |

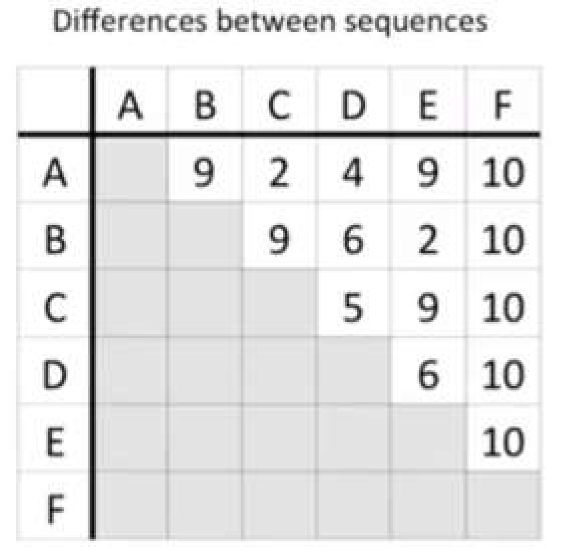
**Step-1 calculating the distance between every pair of data points**



Differences between sequences

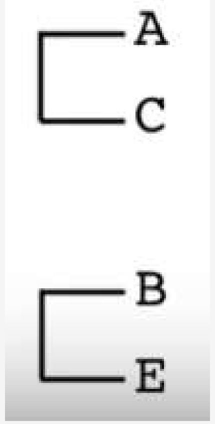
```
A   ATCGTGGTACTG          B   CCGGAGAACTAG
C   AACGTGCTACTG          D   ATGGTGAAAGTG

A   ATCGTGGTACTG          B   CCGGAGAACTAG
D   ATGGTGAAAGTG          E   CCGGAAAACTTG
                          F   TGGCCCTGTATC
A   ATCGTGGTACTG
E   CCGGAAAACTTG          C   AACGTGCTACTG
F   TGGCCCTGTATC          D   ATGGTGAAAGTG
A   ATCGTGGTACTG
                          C   AACGTGCTACTG
                          E   CCGGAAAACTTG
                          F   TGGCCCTGTATC
```

**Nxt-Step: Prepare distance matrix table of differences & identify the sequences with fewest difference between them**

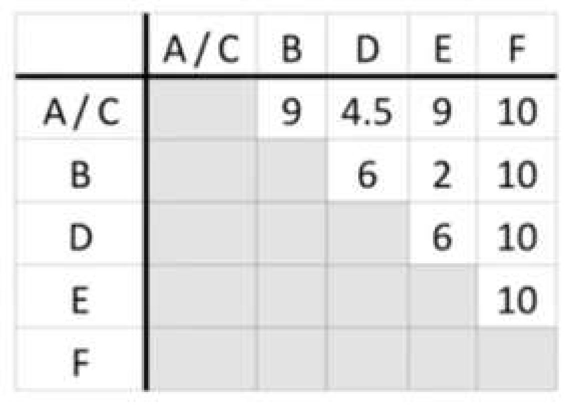


Differences between sequences

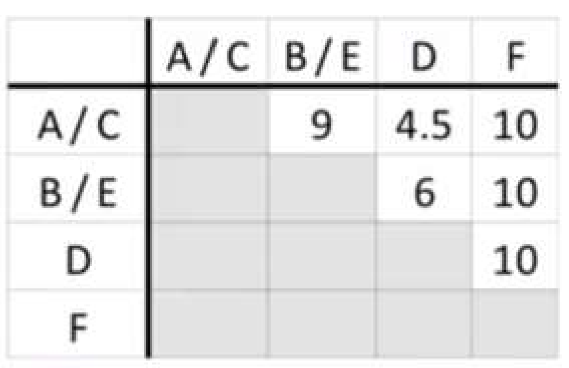
| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| A | | 9 | 2 | 4 | 9 | 10 |
| B | | | 9 | 6 | 2 | 10 |
| C | | | | 5 | 9 | 10 |
| D | | | | | 6 | 10 |
| E | | | | | | 10 |
| F | | | | | | |



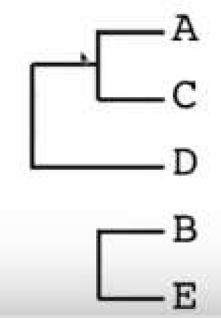
**Nxt-Step: Now regroup them and prepare new distance matrix table**



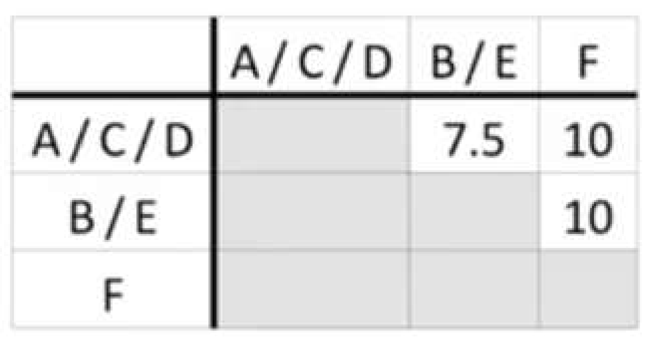
| | A/C | B | D | E | F |
|---|---|---|---|---|---|
| A/C | | 9 | 4.5 | 9 | 10 |
| B | | | 6 | 2 | 10 |
| D | | | | 6 | 10 |
| E | | | | | 10 |
| F | | | | | |



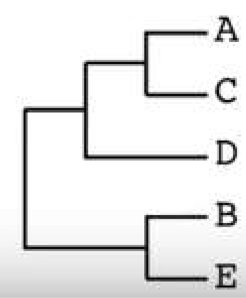
| | A/C | B/E | D | F |
|---|---|---|---|---|
| A/C | | 9 | 4.5 | 10 |
| B/E | | | 6 | 10 |
| D | | | | 10 |
| F | | | | |

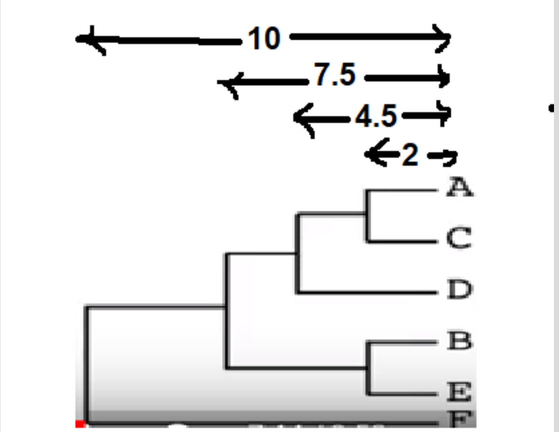


**Nxt-Step: Again, regroup and prepare new distance matrix table**



| | A/C/D | B/E | F |
|---|---|---|---|
| A/C/D | | 7.5 | 10 |
| B/E | | | 10 |
| F | | | |

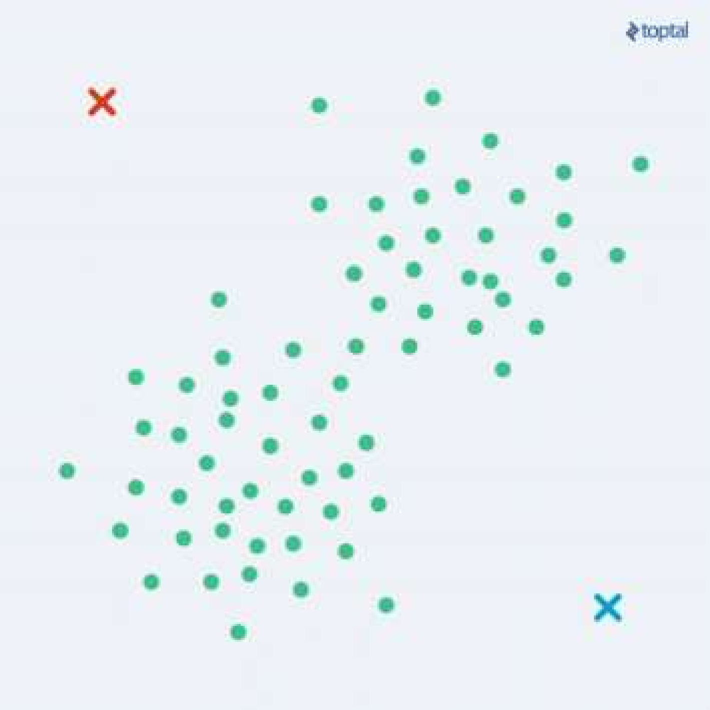




**At distance 2 to 4.5, there're 4-clusters**
**distance 4.5 to 7.5, 3-clusters**
**distance 7.5 to 10, 2-clusters**

Redundance cases: beyond 10, 1 cluster; at distance 0 to 2, 6-clusters

# Centroid-based clustering/partition clustering



## K-means clustering

- Step-1 Initially two centroids are assigned randomly
- Step-2: the Euclidean distance is used to find out which centroid is closest to each data point and the data points are assigned to the corresponding centroids.
- Step-3 Reposition the two centroids for optimization
- Step-4 The process is iteratively repeated  till the repositioning of the centroids stops . Same means in two successive steps
- cluster these data points – {2, 3,4, 10 11, 12, 20, 25, 30}

- Example

K1={2,3,4,10,11,12} mean 7 and k2={20,25,30} mean 25

**Application:**

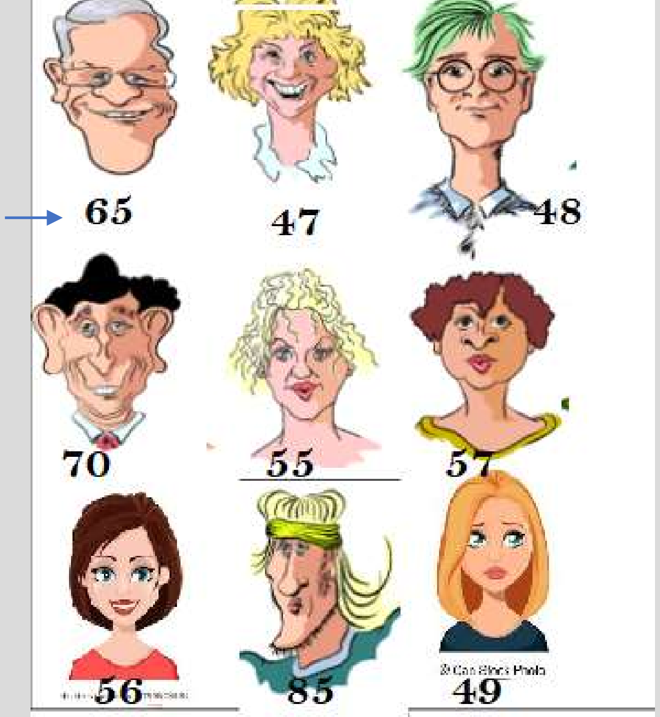
**academic performance, search engine, diagnostic, wireless sensor network**

# Hands-on exercise for Centroid-based clustering

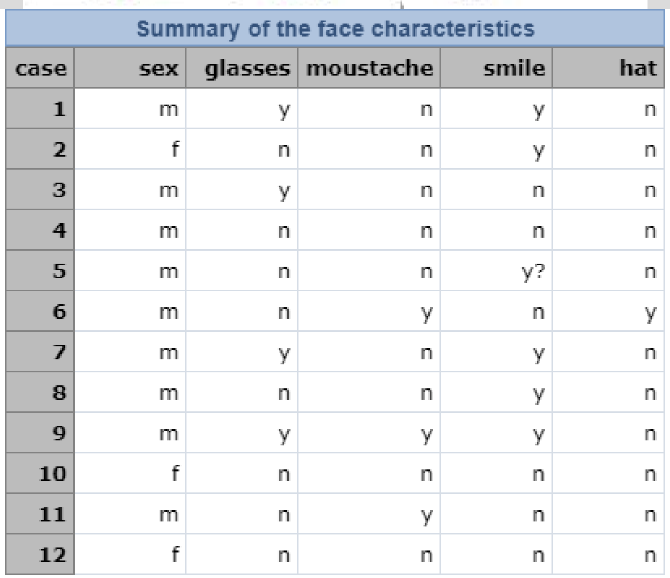
Problem - Cluster these data points – {2, 3,4, 10 11, 12, 20, 25, 30} using K-means clustering.

**Description:** In this exercise, of Gender Taxonomy, we want to group Homo sapiens species into – Male and Female categories based on characters as shown in the opposite figure. Since it is not possible to do hands-on calculations in the classroom with several characters, we have taken just one character – weight. We have following individual species (OTUs – operational Taxonomic Units) having character (weight in Kg) – {47, 48, 49, 55, 56, 57, 65, 70, 85}. Since hand calculations of such large numbers may take time, to facilitate we have subtracted 45 from each data and our data of OTUs is as follows – {2, 3,4, 10 11, 12, 20, 25, 30} so that calculations can be performed in short time. Now Cluster and classify these gender based OTUs

**Answer:**
C1={2,3,4,10,11,12} mean 7 and
C2={20,25,30} mean 25



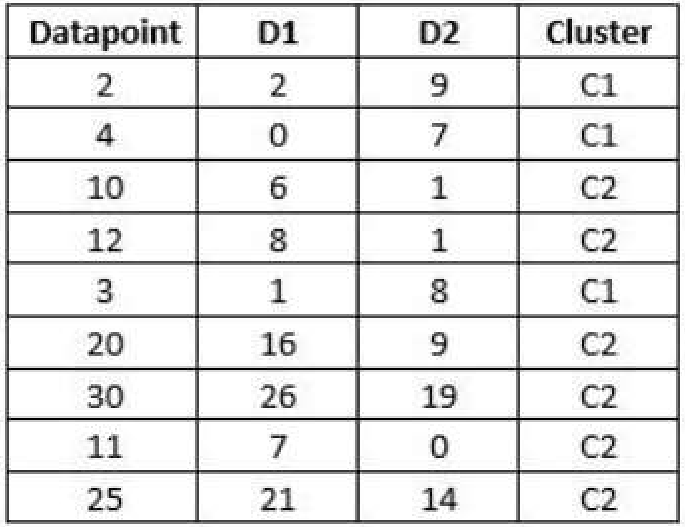
Characters
(Weight in Kg.)



| Summary of the face characteristics | | | | | |
|---|---|---|---|---|---|
| case | sex | glasses | moustache | smile | hat |
| 1 | m | y | n | y | n |
| 2 | f | n | n | y | n |
| 3 | m | y | n | n | n |
| 4 | m | n | n | n | n |
| 5 | m | n | n | y? | n |
| 6 | m | n | y | n | y |
| 7 | m | y | n | y | n |
| 8 | m | n | n | y | n |
| 9 | m | y | y | y | n |
| 10 | f | n | n | n | n |
| 11 | m | n | y | n | n |
| 12 | f | n | n | n | n |

In a typical example, to cluster the data into male and female categories there would be several features, for the present Hands on exercise, we have taken only one feature, i.e. weight

**Assume starting centroids are M1=4 and M2=11**

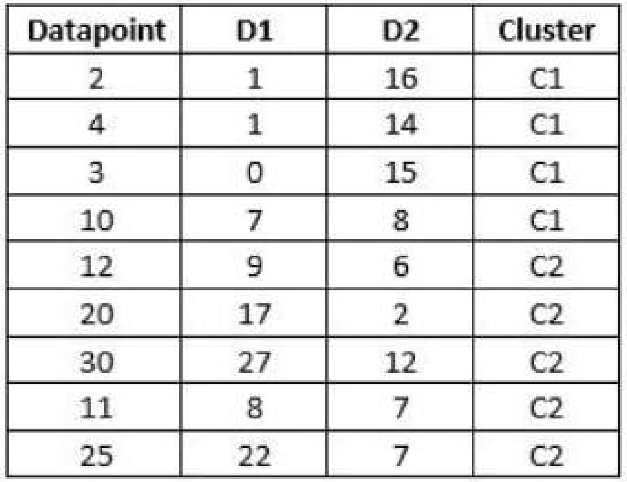


| Datapoint | D1 | D2 | Cluster |
|---|---|---|---|
| 2 | 2 | 9 | C1 |
| 4 | 0 | 7 | C1 |
| 10 | 6 | 1 | C2 |
| 12 | 8 | 1 | C2 |
| 3 | 1 | 8 | C1 |
| 20 | 16 | 9 | C2 |
| 30 | 26 | 19 | C2 |
| 11 | 7 | 0 | C2 |
| 25 | 21 | 14 | C2 |

Iteration 1

**<-- ITERATION-1**

**ITERATION-2 →**



| Datapoint | D1 | D2 | Cluster |
|---|---|---|---|
| 2 | 1 | 16 | C1 |
| 4 | 1 | 14 | C1 |
| 3 | 0 | 15 | C1 |
| 10 | 7 | 8 | C1 |
| 12 | 9 | 6 | C2 |
| 20 | 17 | 2 | C2 |
| 30 | 27 | 12 | C2 |
| 11 | 8 | 7 | C2 |
| 25 | 22 | 7 | C2 |

Iteration 2

C1= {2, 4, 3}
C2= {10, 12, 20, 30, 11, 25}

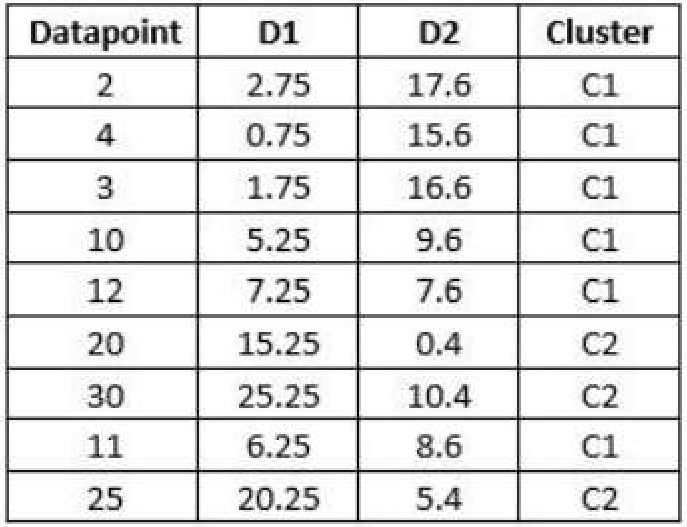
M1= (2+3+4)/3= 3
M2= (10+12+20+30+11+25)/6= 18

C1= {2, 3, 4, 10}
C2= {12, 20, 30, 11, 25}

M1= (2+3+4+10)/4= 4.75
M2= (12+20+30+11+25)/5= 19.6

**Symbol Meaning**: M1, M2 are centroids of clusters C1 and C2
D1, D2 are distances of data points from centroids M1 and M2

## -- ITERATION-3



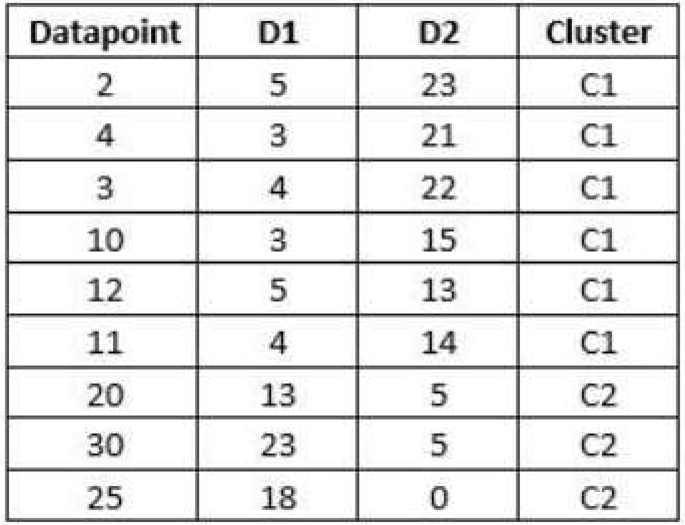
| Datapoint | D1 | D2 | Cluster |
|---|---|---|---|
| 2 | 2.75 | 17.6 | C1 |
| 4 | 0.75 | 15.6 | C1 |
| 3 | 1.75 | 16.6 | C1 |
| 10 | 5.25 | 9.6 | C1 |
| 12 | 7.25 | 7.6 | C1 |
| 20 | 15.25 | 0.4 | C2 |
| 30 | 25.25 | 10.4 | C2 |
| 11 | 6.25 | 8.6 | C1 |
| 25 | 20.25 | 5.4 | C2 |

Iteration 3

C1= {2, 3, 4, 10, 12, 11}
C2= {20, 30, 25}

M1= (2+3+4+10+12+11)/6=7
M2= (20+30+25)/3= 25

## ITERATION-4 →



| Datapoint | D1 | D2 | Cluster |
|---|---|---|---|
| 2 | 5 | 23 | C1 |
| 4 | 3 | 21 | C1 |
| 3 | 4 | 22 | C1 |
| 10 | 3 | 15 | C1 |
| 12 | 5 | 13 | C1 |
| 11 | 4 | 14 | C1 |
| 20 | 13 | 5 | C2 |
| 30 | 23 | 5 | C2 |
| 25 | 18 | 0 | C2 |

Iteration 4

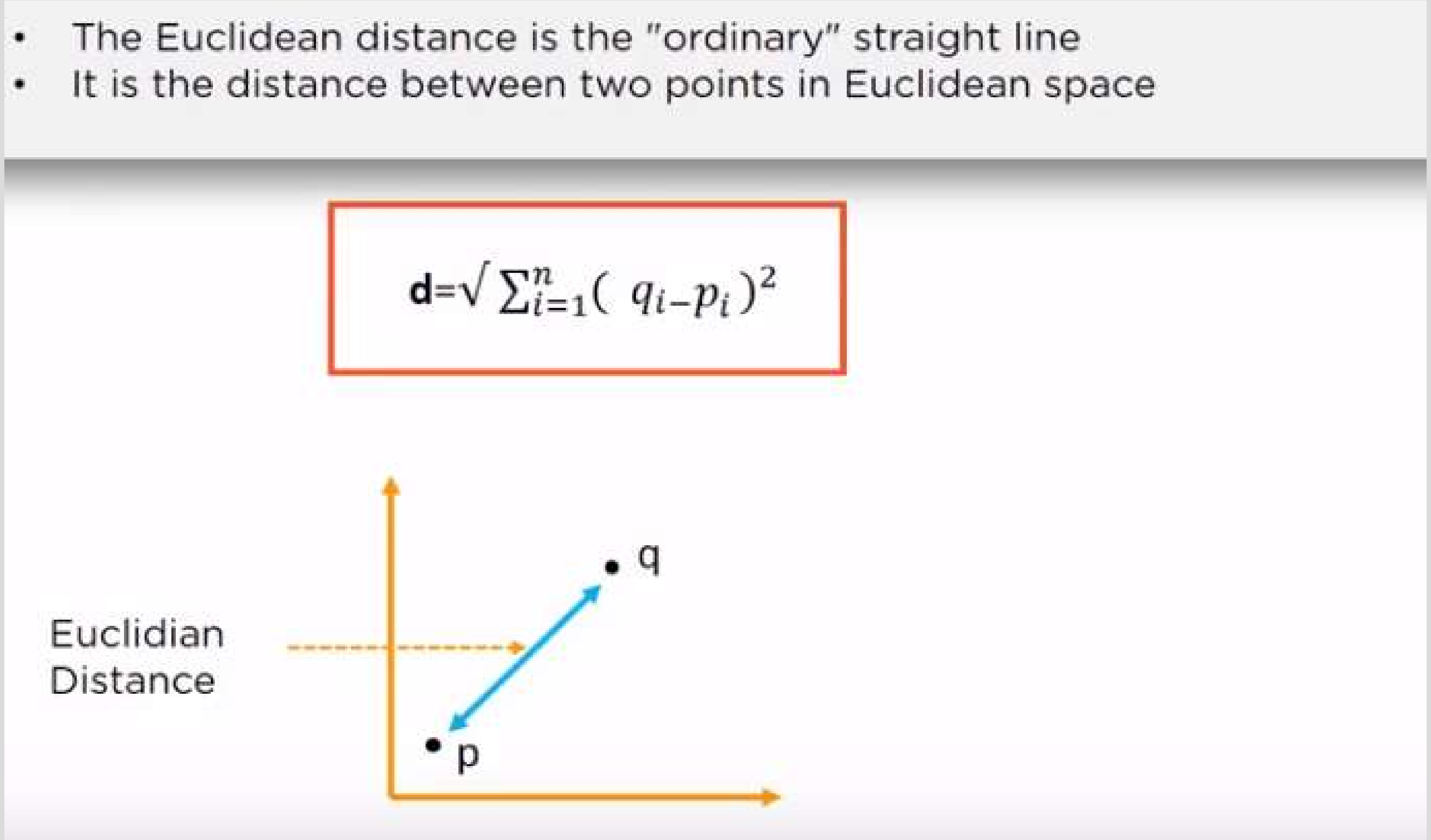
C1= {2, 3, 4, 10, 12, 11}
C2= {20, 30, 25}

# Various types of Distances

**Distance measure**

will determine the similarity between two elements and it will influence the shape of cluster

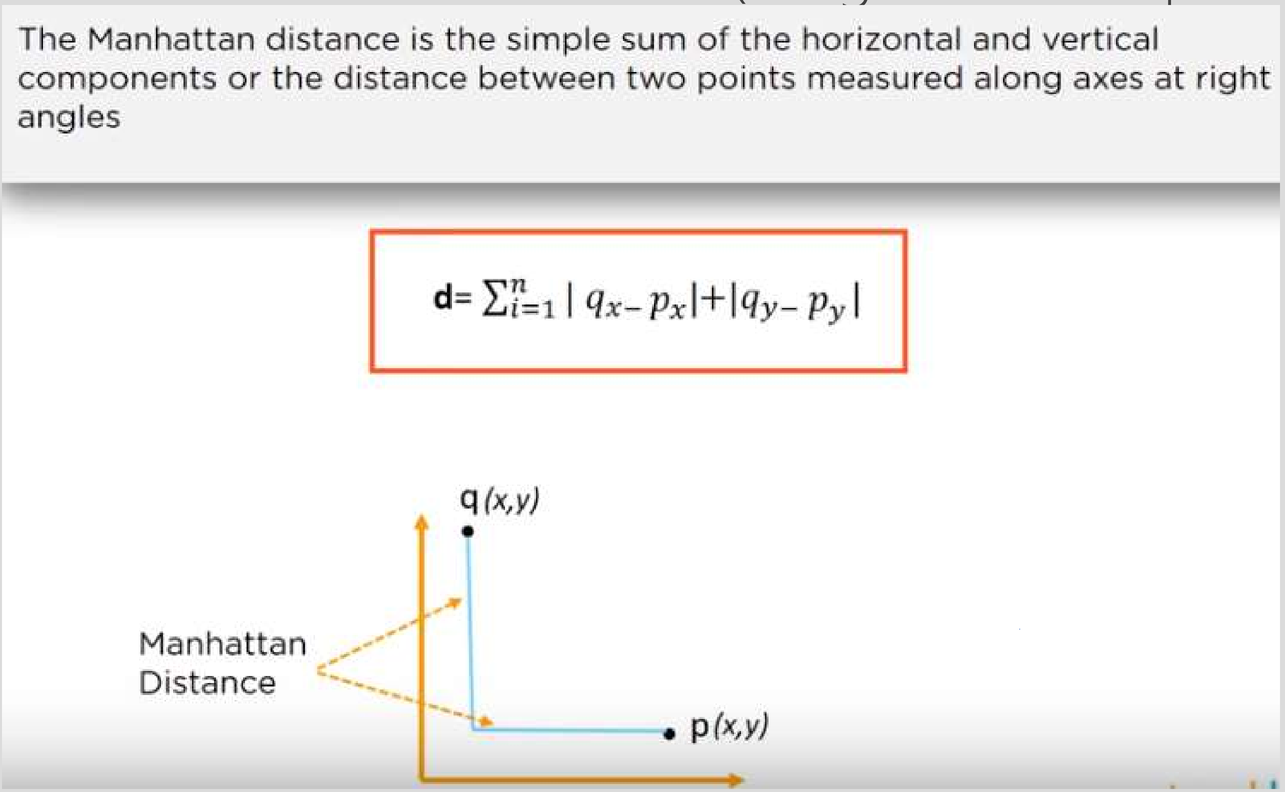
- There are multiple metrics for deciding the closeness of two clusters
- (1) Euclidean distance (d) and
- (2) Euclidean square distance ($d^2$)



- The Euclidean distance is the "ordinary" straight line
- It is the distance between two points in Euclidean space

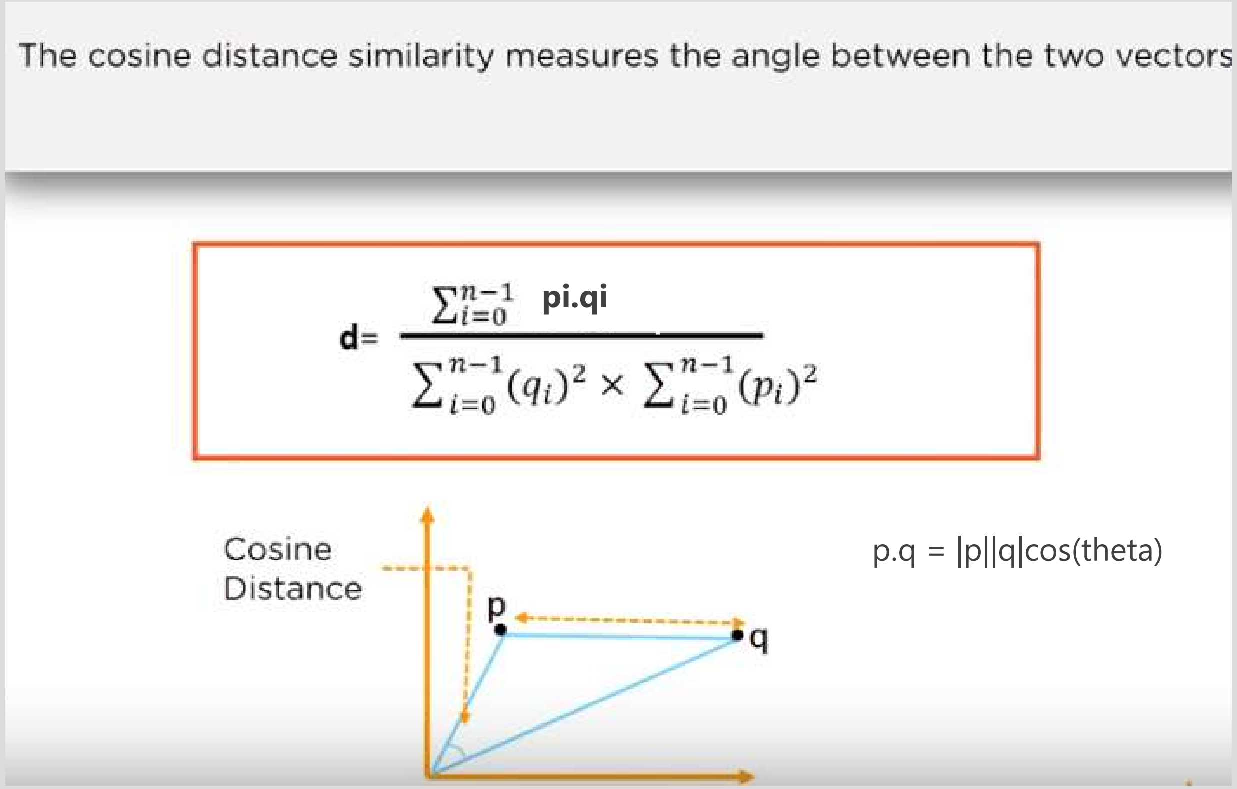
$$\mathbf{d}=\sqrt{\textstyle\sum_{i=1}^{n}}(\ q_i - p_i\ )^2$$

- (3) Manhattan distance: $||a-b||_1 = \Sigma|a_i-b_i|$
- (4) Maximum distance:$||a-b||_{INFINITY} = \max_i|a_i-b_i|$
- (5) Mahalanobis distance: $\sqrt{((a-b)^T S^{-1} (-b))}$ {where, s : covariance matrix}
- (6) Cosine distance, 7) In bioinformatics, % identity or similar measures can be distance between data (i.e. genomic or proteomic sequences)



The Manhattan distance is the simple sum of the horizontal and vertical components or the distance between two points measured along axes at right angles

$$\mathbf{d}= \sum_{i=1}^{n} | q_{x-} p_x|+|q_{y-} p_y|$$



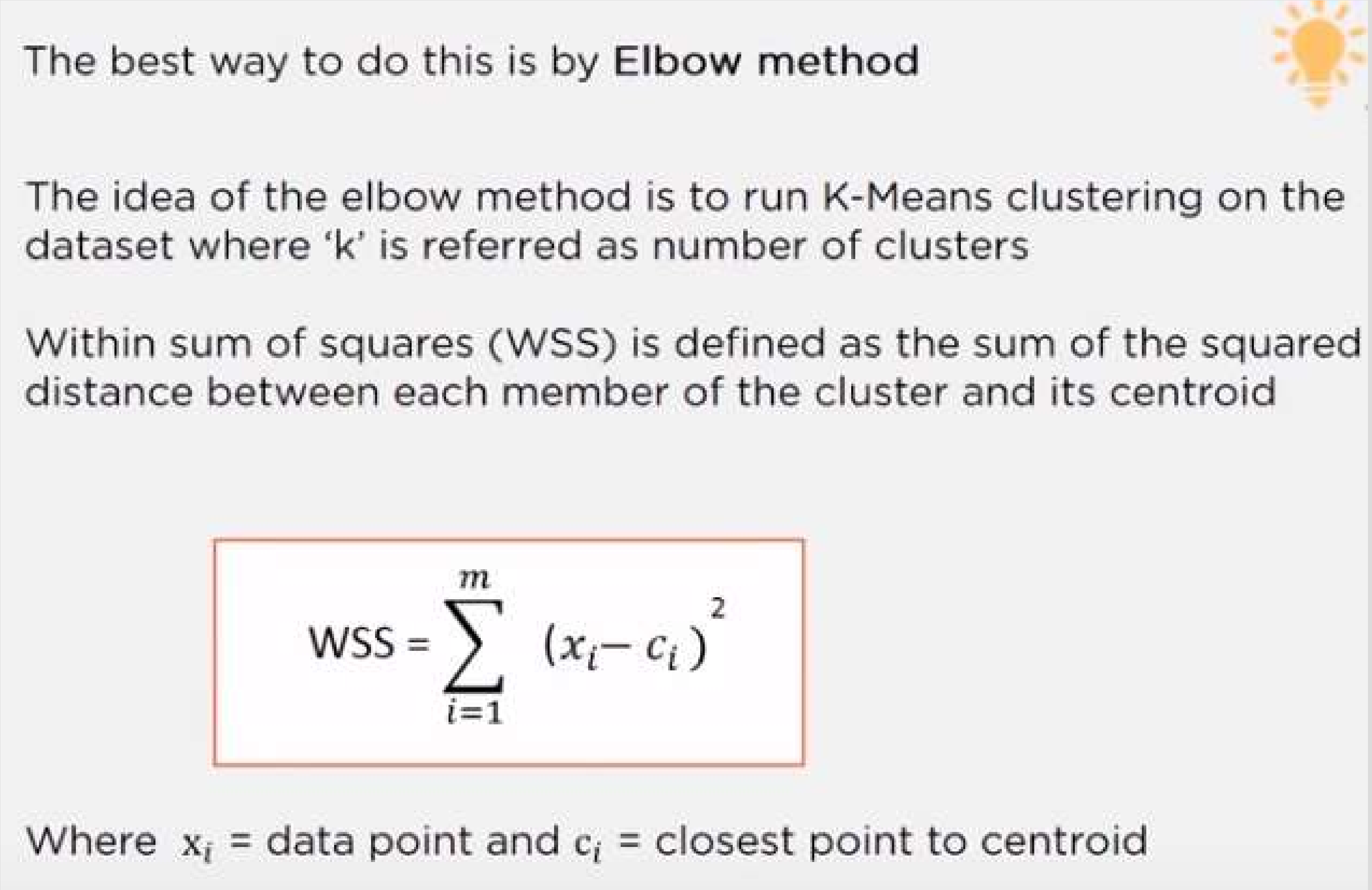
The cosine distance similarity measures the angle between the two vectors

$$\mathbf{d}= \frac{\sum_{i=0}^{n-1} \mathbf{pi.qi}}{\sum_{i=0}^{n-1}(q_i)^2 \times \sum_{i=0}^{n-1}(p_i)^2}$$

p.q = |p||q|cos(theta)

# How do we find the optimum number of clusters?

- One of the most common method is - Elbow method



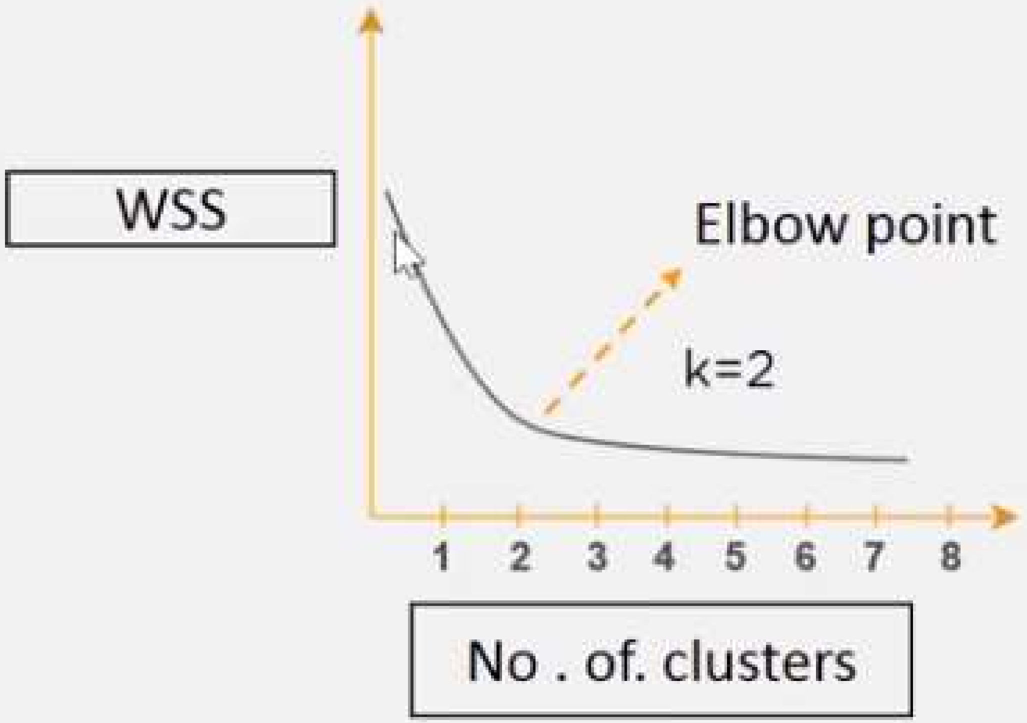
The best way to do this is by **Elbow method**

The idea of the elbow method is to run K-Means clustering on the dataset where ‘k’ is referred as number of clusters

Within sum of squares (WSS) is defined as the sum of the squared distance between each member of the cluster and its centroid

$$\boxed{\text{WSS} = \sum_{i=1}^{m} (x_i - c_i)^2}$$

Where $x_i$ = data point and $c_i$ = closest point to centroid



- Now, we draw a curve between *WSS* (within sum of squares) and the *number of clusters*

- Here, we can see a very slow change in the value of WSS after k=2, so you should take that elbow point value as the final number of clusters

# Leisure reading: Various approaches of finding optimum clusters

- **1. Cross Validation:** It’s a commonly used method for determining k value. It divides the data into X parts. Then, it trains the model on X-1 parts and validates (test) the model on the remaining part.
- The model is validated by checking the value of the sum of squared distance to the centroid. This final value is calculated by averaging over X clusters. Practically, for different values of k, we perform cross validation and then choose the value which returns the lowest error.
- **2. Elbow Method:** This method calculates the best k value by considering the percentage of variance explained by each cluster. It results in a plot similar to PCA’s scree plot. In fact, the logic behind selecting the best cluster value is the same as PCA.
- In PCA, we select the number of components such that they explain the maximum variance in the data. Similarly, in the plot generated by the elbow method, we select the value of k such that percentage of variance explained is maximum.
- **3. Silhouette Method:** It returns a value between -1 and 1 based on the similarity of an observation with its own cluster. Similarly, the observation is also compared with other clusters to derive at the similarity score. High value indicates high match, and vice versa. We can use any distance metric (explained above) to calculate the silhouette score.
- **4. X means Clustering:** This method is a modification of the k means technique. In simple words, it starts from k = 1 and continues to divide the set of observations into clusters until the best split is found or the stopping criterion is reached. But, how does it find the best split ? It uses the Bayesian information criterion to decide the best split.