UNIT - III

ENSEMBLE TECHNIQUES & UNSUPERVISED LEARNING

UNIT III ENSEMBLE TECHNIQUES & UNSUPERVISED LEARNING 9

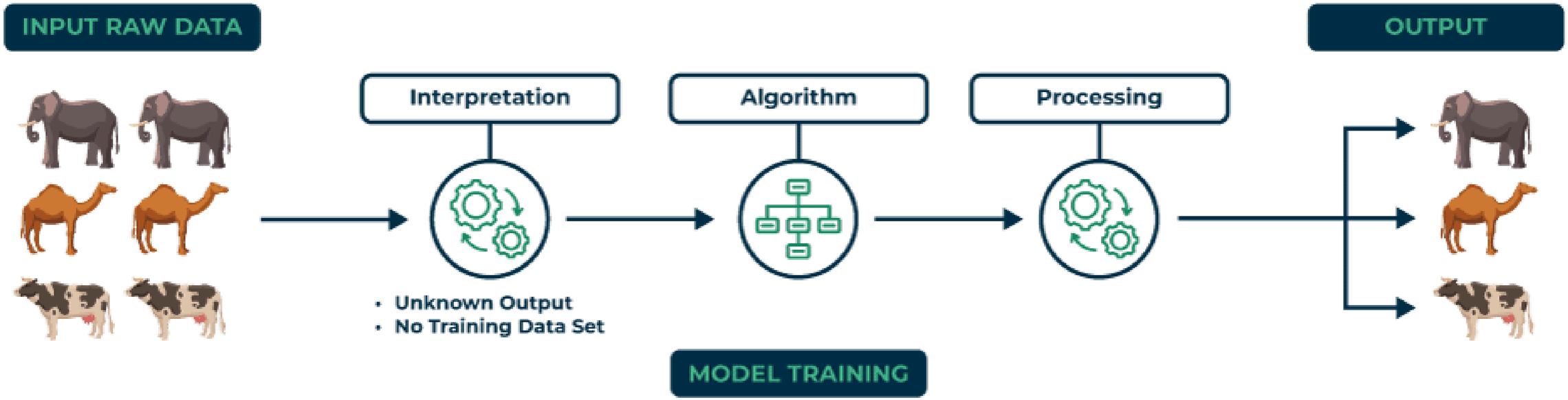
Combining multiple learners: Model combination schemes, Voting, Ensemble Learning - bagging, boosting, stacking, Unsupervised learning: K-means, Instance-Based Learning: KNN, Gaussian mixture models and Expectation maximization

UNSUPERVISED LEARNING

UNSUPERVISED LEARNING

- Unsupervised learning in artificial intelligence is a type of machine learning that learns from data without human supervision.
- Unlike supervised learning, unsupervised machine learning models are given unlabeled data and allowed to discover patterns and insights without any explicit guidance or instruction.

Unsupervised Learning



Unsupervised Learning Approaches

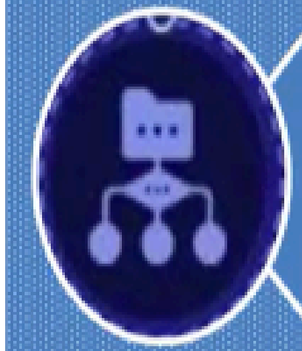
- ✓ Clustering
- ✓ Association
- ✓ Dimensionality Reduction

Clustering

- Clustering is a method of grouping the objects into clusters such that objects with most similarities remains into a group and has less or no similarities with the objects of another group.
- Cluster analysis finds the commonalities between the data objects and categorizes them as per the presence and absence of those commonalities.

Association

- An association rule is an unsupervised learning method which is used for finding the relationships between variables in the large database.
- It determines the set of items that occurs together in the dataset.
- Association rule makes marketing strategy more effective.
- Such as people who buy X item (suppose a bread) are also tend to purchase Y (Butter/Jam) item. A typical example of Association rule is Market Basket Analysis.



Unsupervised Learning

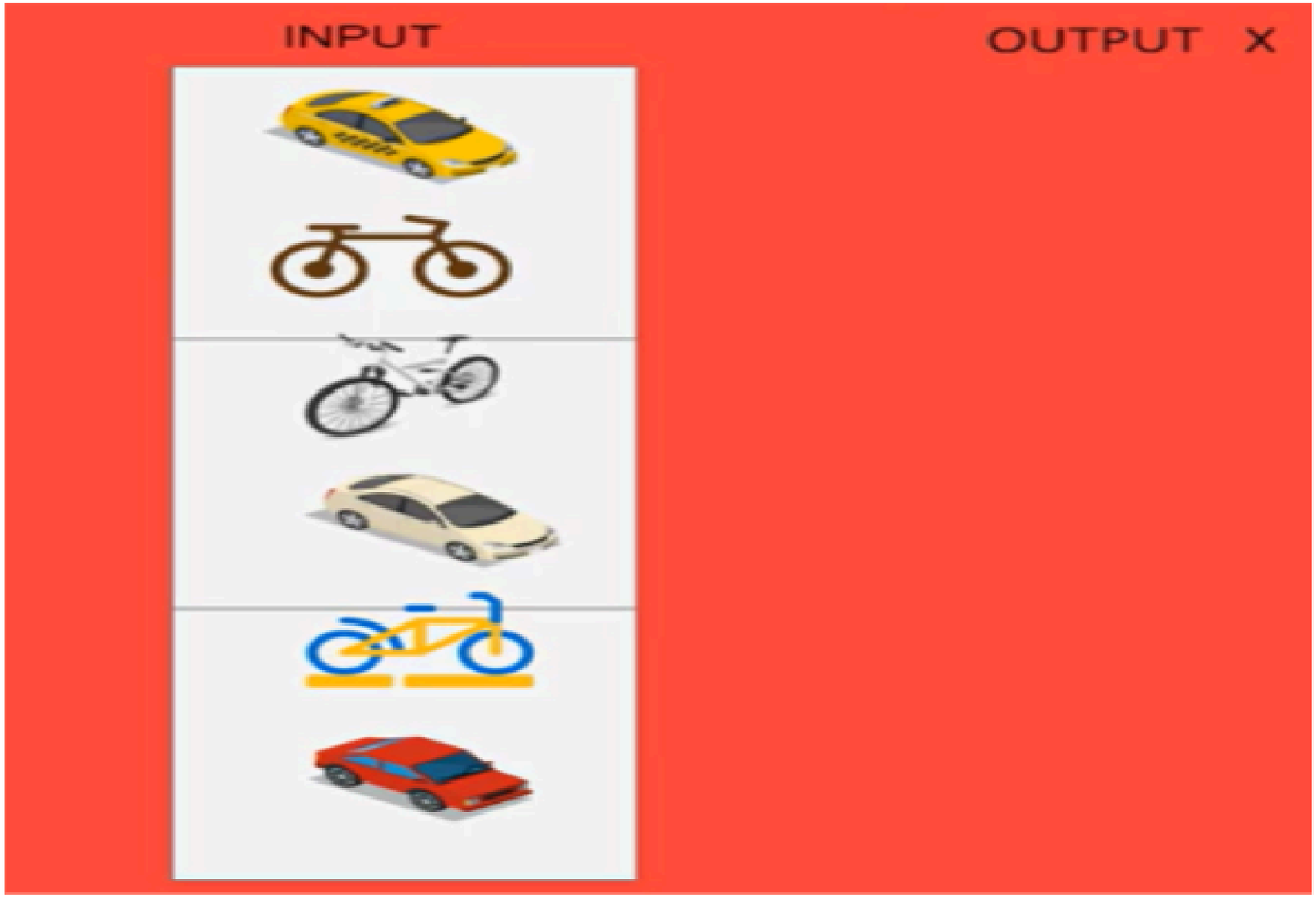
Clustering

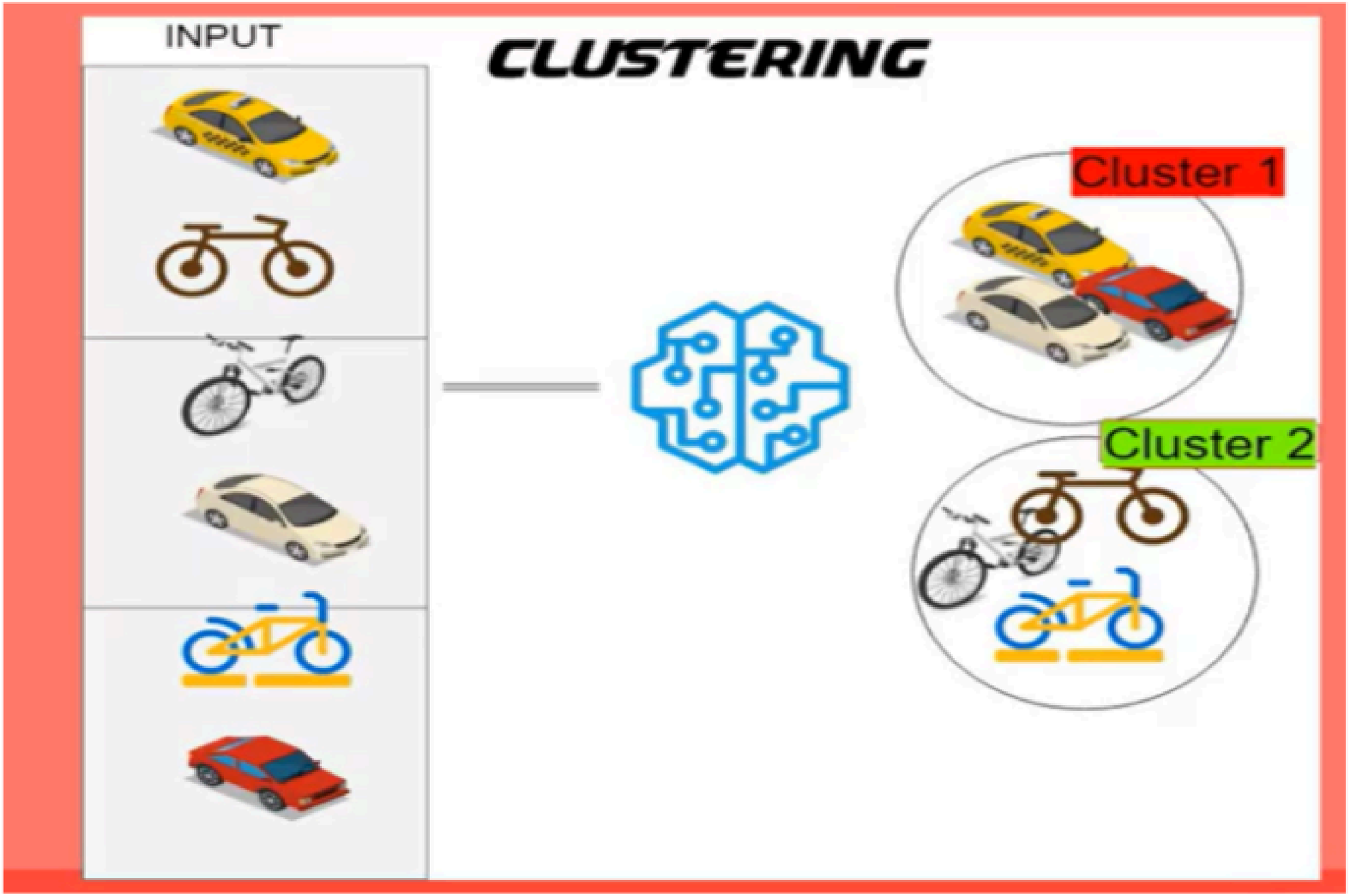
It is a method of grouping the objects into clusters such that objects with most similarities remains into a group and has less or no similarities with the objects of another group.

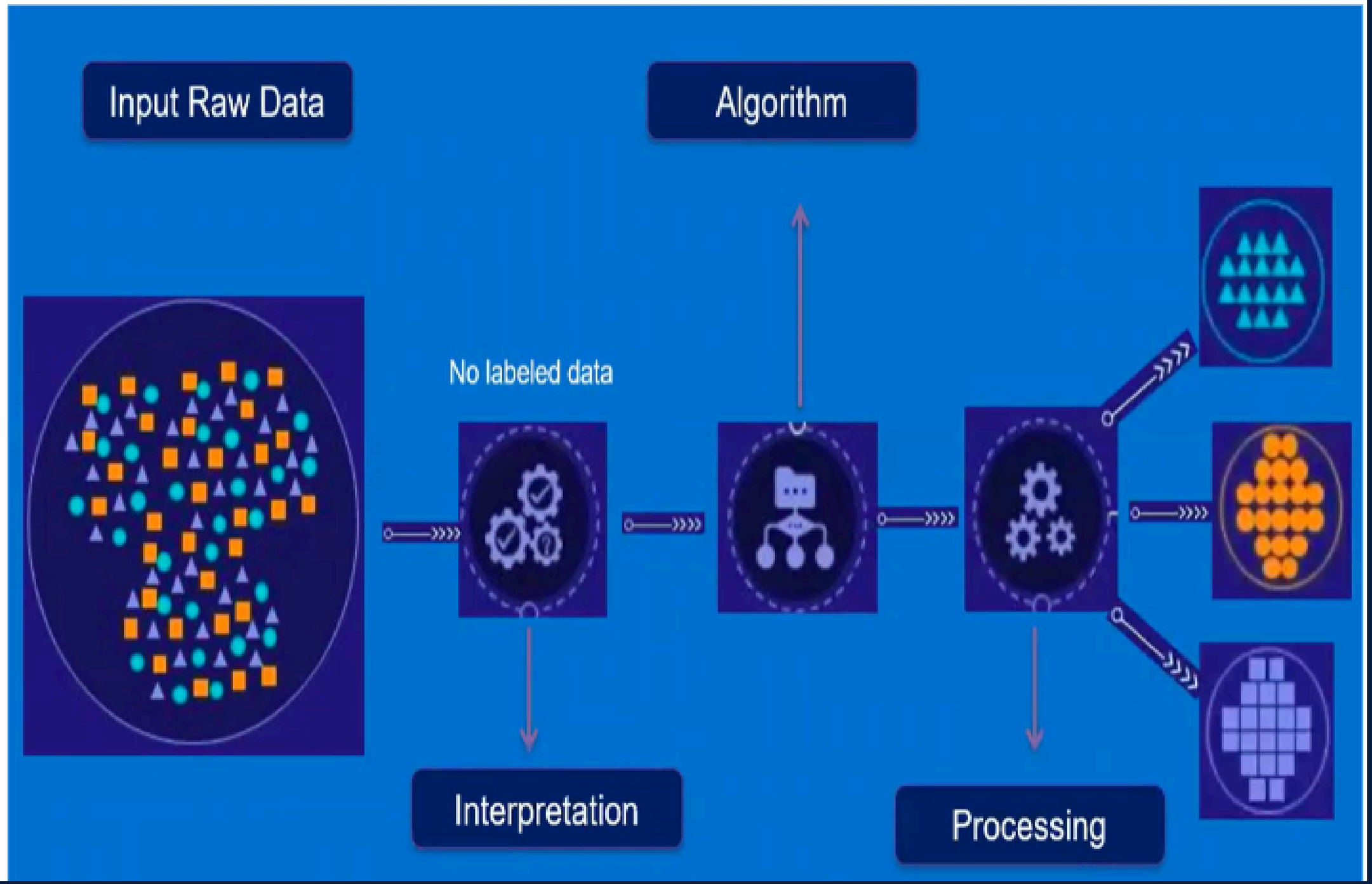
Association

Discovering the probability of the co-occurrence of items in a collection

Working of Unsupervised Learning




INPUT
CLUSTERING
Cluster 1
Cluster 2


Input Raw Data
Algorithm
No labeled data
Interpretation
Processing

- If input is fed into the clustering algorithm, it identifies the clusters in the input data and returns it.
- Issue – Number of clusters to be given manually
- Once the number of clusters given, similar feature samples are assigned to their respective clusters
- In real time if a dataset is given, number of clusters cannot be identified properly.
- To identify the cluster count, we can utilize Domain expert or trial and error basis - elbow method

Association

Customer1

- Bread
- Milk
- Fruits
- Wheat

Customer2

- Bread
- Milk
- Rice
- Butter

Customer3

If a new customer purchases bread, he is likely to purchase milk too

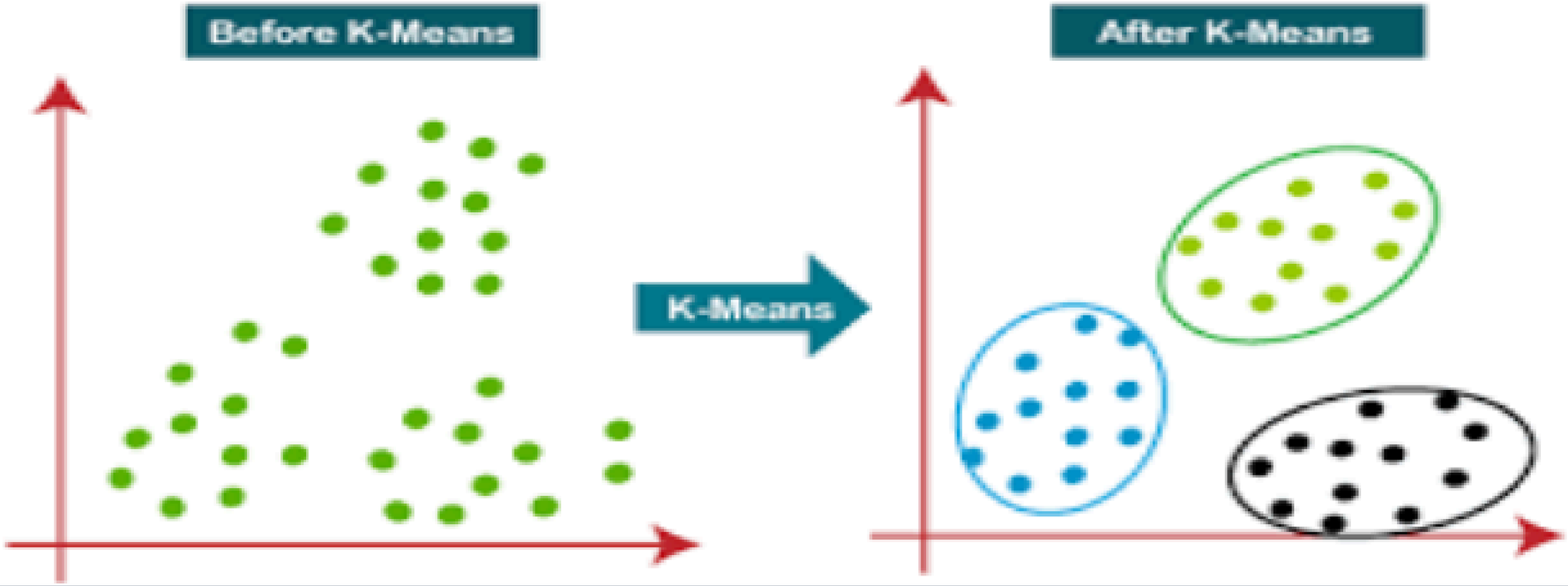
Unsupervised Learning Algorithms

- K Means Clustering
- KNN
- Anomaly Detection
- Principal Component Analysis
- Hierarchical Clustering
- Neural Networks
- Independent Component Analysis

K-Means Clustering Algorithm

- **K-Means Clustering** is an unsupervised learning algorithm that is used to **solve the clustering problems** in machine learning or data science.
- K-Means Clustering - **groups** the **unlabeled dataset** into **different clusters**.
- Here **K** defines the number of **pre-defined clusters** that need to be created in the process, as if K=2, there will be two clusters, and for K=3, there will be three clusters, and so on.

- It is an iterative algorithm that divides the unlabeled dataset into k different clusters.
- The main aim of K means algorithm is to minimize the sum of distances between the data point and their corresponding clusters.
- It is a centroid-based algorithm, where each cluster is associated with a centroid.



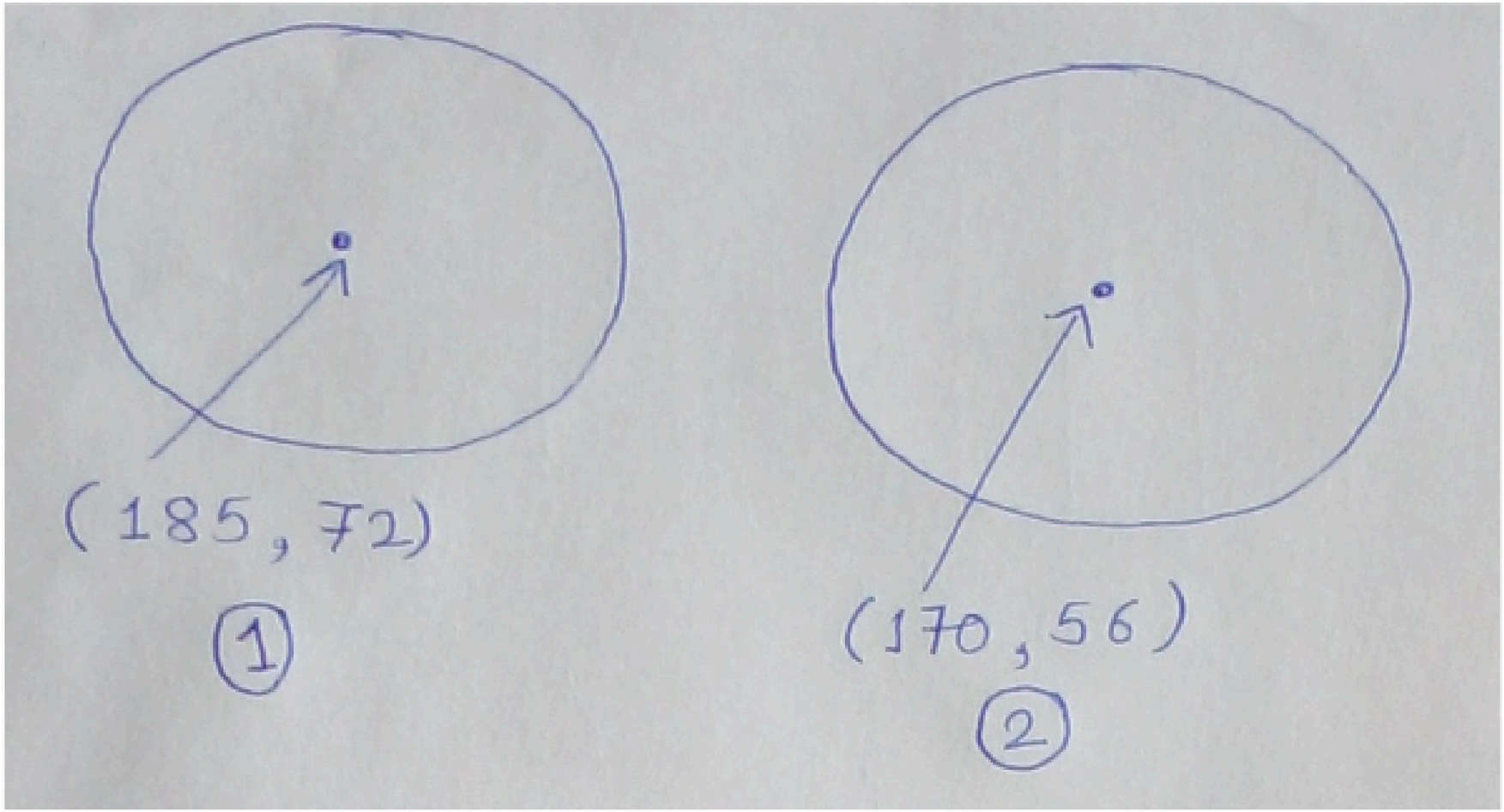
Working of K Means Algorithm

- The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters.
- The value of k should be predetermined in this algorithm.
- The k-means clustering algorithm mainly performs two tasks:
  - ❑ Determines the best value for K center points or centroids by an iterative process.
  - ❑ Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.
- Hence each cluster has data points with some commonalities, and it is away from other clusters.

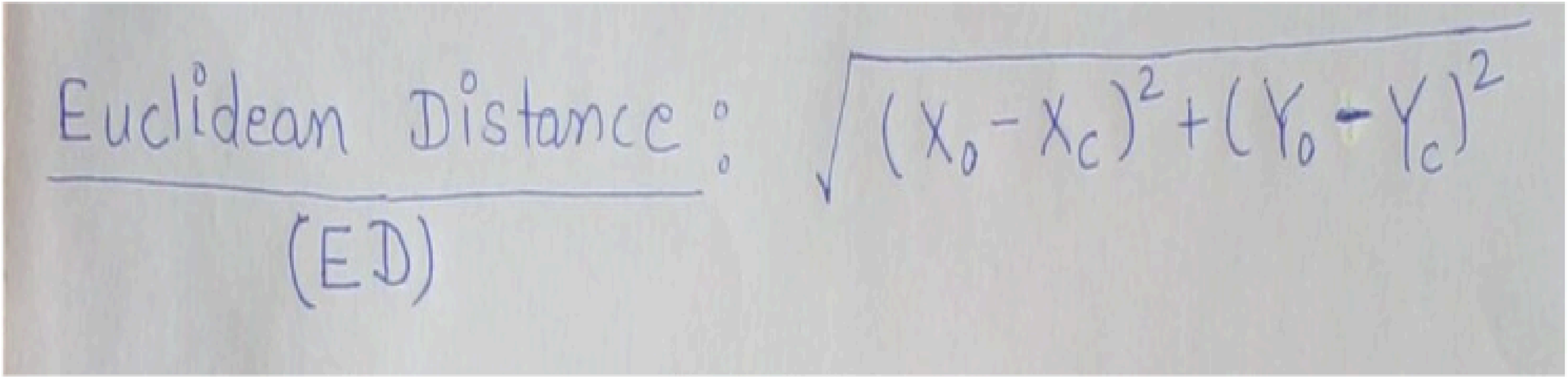
- **Step-1:** Select the number K to decide the number of clusters.
- **Step-2:** Select random K points or centroids.
- **Step-3:** Assign each data point to their closest centroid, which will form the predefined K clusters.
- **Step-4:** Calculate the variance and place a new centroid of each cluster.
- **Step-5:** Repeat the third steps, which means reassign each data point to the new closest centroid of each cluster.
- **Step-6:** If any reassignment occurs, then go to step-4 else go to FINISH.
- **Step-7**: The model is ready.

S.No	Height	Weight
1	185	72
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

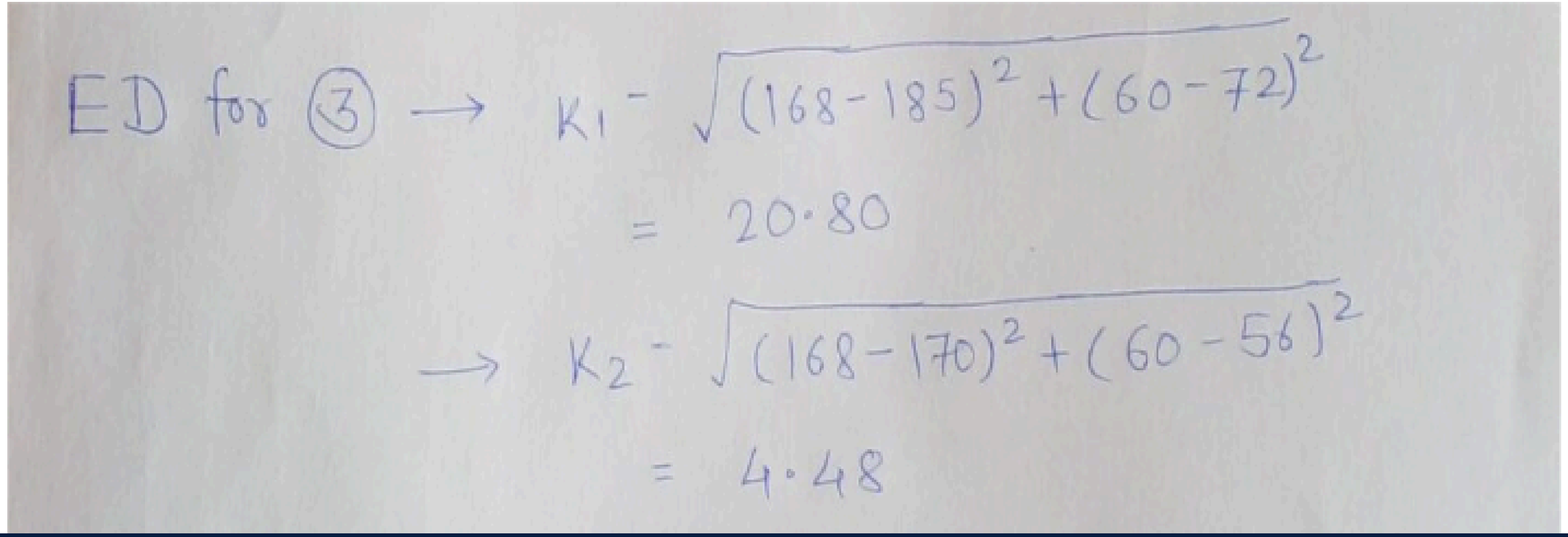
- we need to form 2 clusters.
- Now consider first two data points of our data and assign them as a centroid for each cluster.



- Now we need to assign each and every data point of our data to one of these clusters based on Euclidean distance calculation.



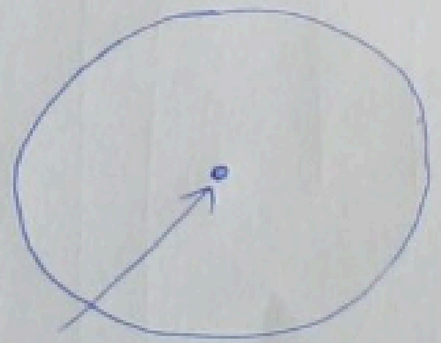
- Consider the next data point i.e. 3rd data point(168,60) and check its distance with the centroid of both clusters.

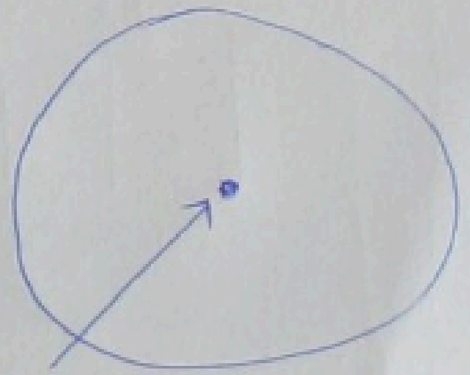


- 3rd data point(168,60) is more closer to k2(cluster 2), so we assign it to k2.
- After that we need to modify the centroid of k2 by using the old centroid values and new data point which we just assigned to k2.

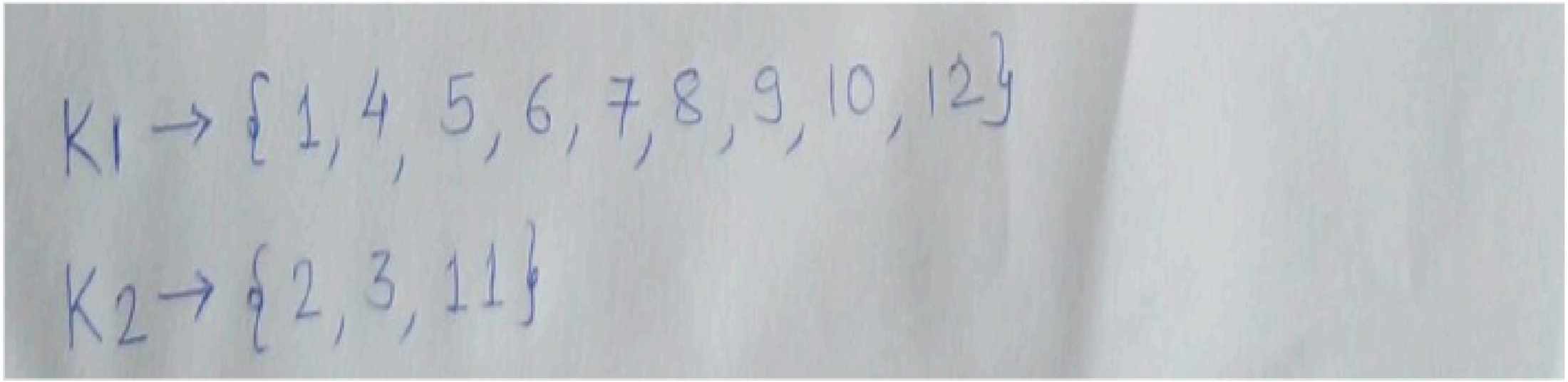
New Centroid Calculation -

for $K_2 = \left(\frac{170+168}{2}, \frac{60+56}{2}\right) = (169, 58)$





- Now after new centroid calculations we got new centroid value for k2 as (169,58) and k1 centroid value will remain the same as NO new data point is added to that cluster(k1).
- Repeat the above mentioned procedure until all data points are over.



How to choose the value of "K number of clusters" in K-means Clustering?

- The performance of the K-means clustering algorithm depends upon highly efficient clusters that it forms.
- But choosing the optimal number of clusters is a big task.
- There are some different ways to find the optimal number of clusters, but here we are discussing the most appropriate method to find the number of clusters or value of K.

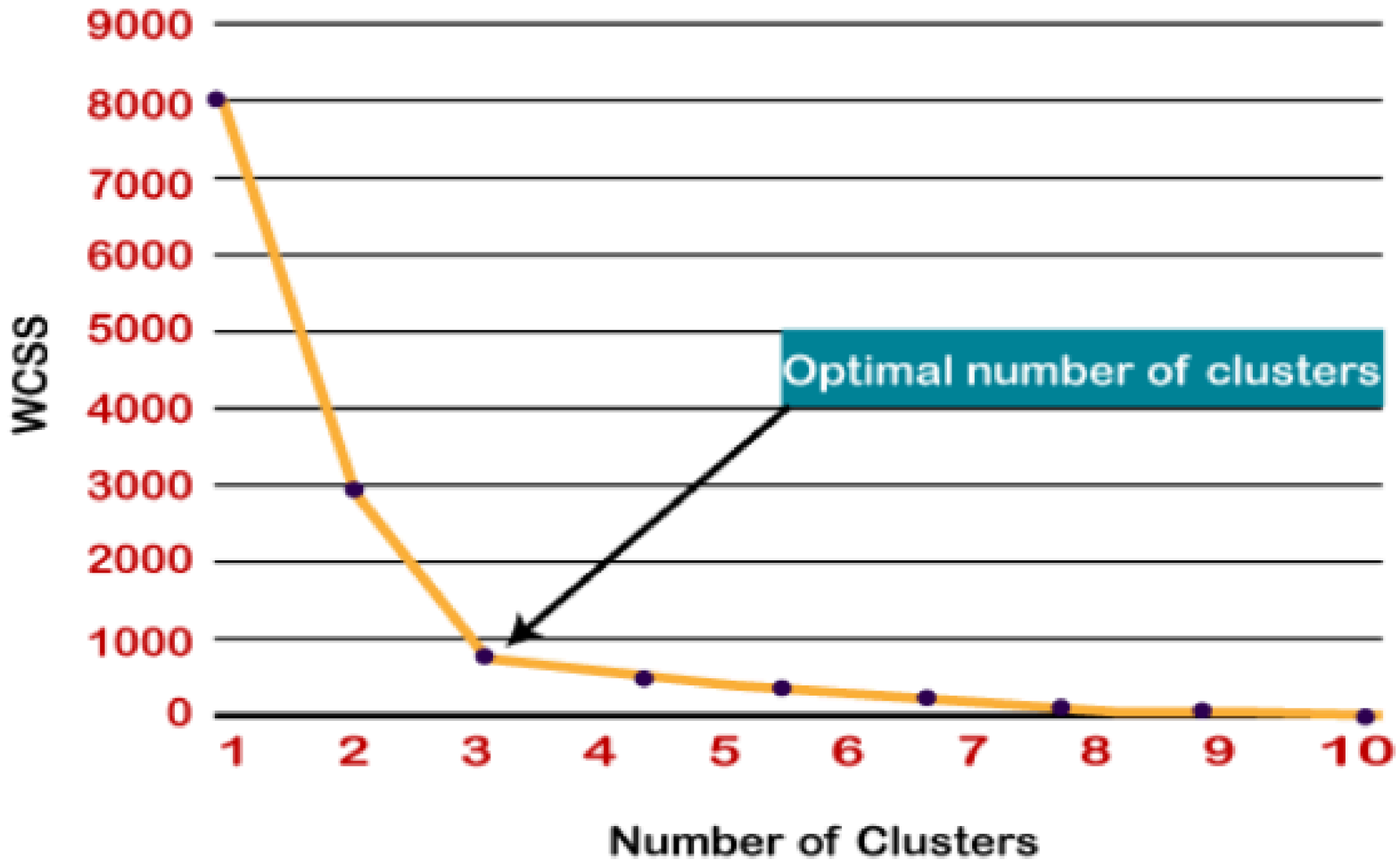
Elbow Method

- The Elbow method is a graphical representation to find the optimal number of clusters 'K' in a K-means clustering.
- This method uses the concept of WCSS value.
- **WCSS** stands for **Within Cluster Sum of Squares**, which defines the total variations within a cluster.
- The formula to calculate the value of WCSS (for 3 clusters) is given below:

- WCSS= $\sum_{Pi \text{ in } Cluster1} distance(P_i C_1)^2 + \sum_{Pi \text{ in } Cluster2} distance(P_i C_2)^2 + \sum_{Pi \text{ in } CLuster3} distance(P_i C_3)^2$
- $\sum_{Pi \text{ in } Cluster1} distance(P_i C_1)^2$: It is the sum of the square of the distances between each data point and its centroid within a cluster1.

- To find the optimal value of clusters, the elbow method follows the below steps:
- It executes the K-means clustering on a given dataset for different K values (ranges from 1-10).
- For each value of K, calculates the WCSS value.
- Plots a curve between calculated WCSS values and the number of clusters K.
- The sharp point of bend or a point of the plot looks like an arm, then that point is considered as the best value of K.

ELBOW CURVE


9000
8000
7000
6000
5000
4000
3000
2000
1000
0
WCSS
Optimal number of clusters
1
2
3
4
5
6
7
8
9
10
Number of Clusters

KNN CLASSIFIER

K-Nearest Neighbor (KNN)

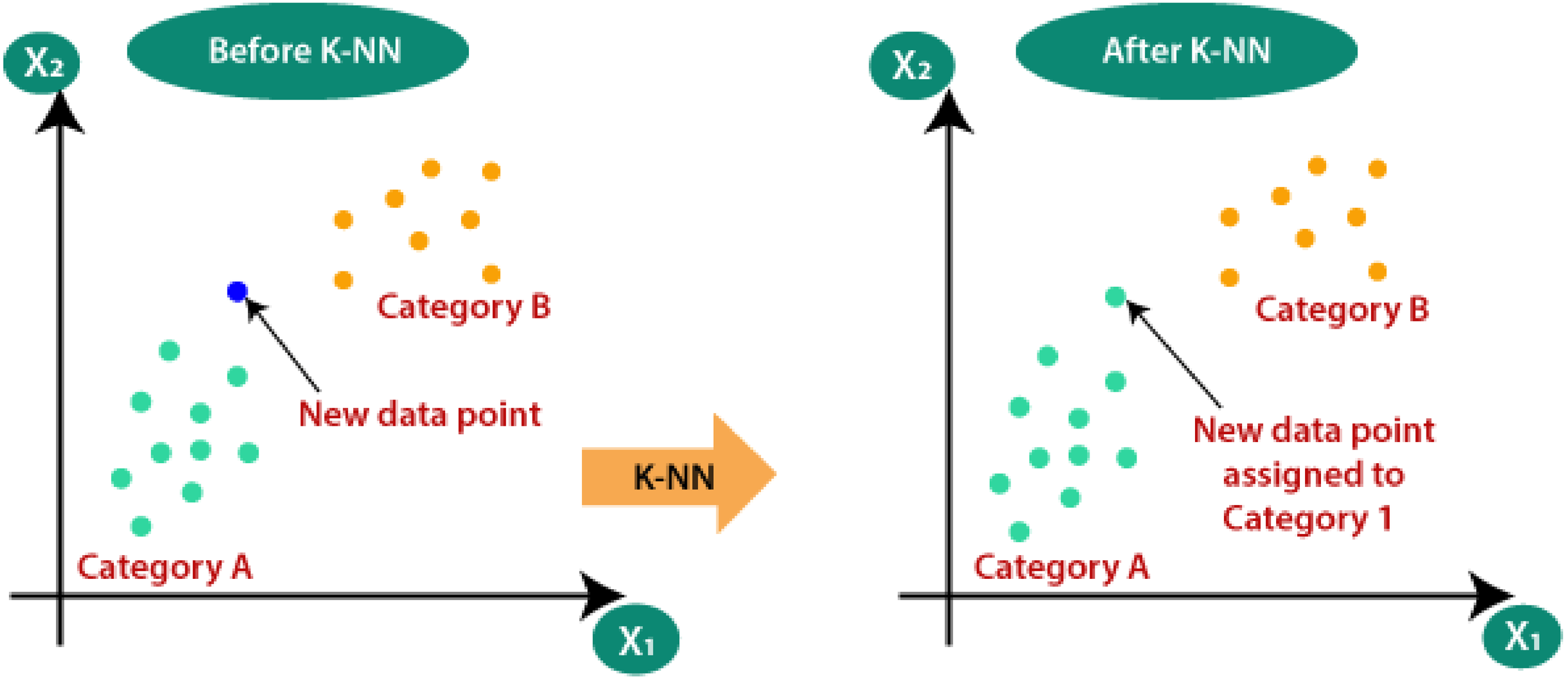
- K-Nearest Neighbour is the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-Nearest Neighbor (KNN)

- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

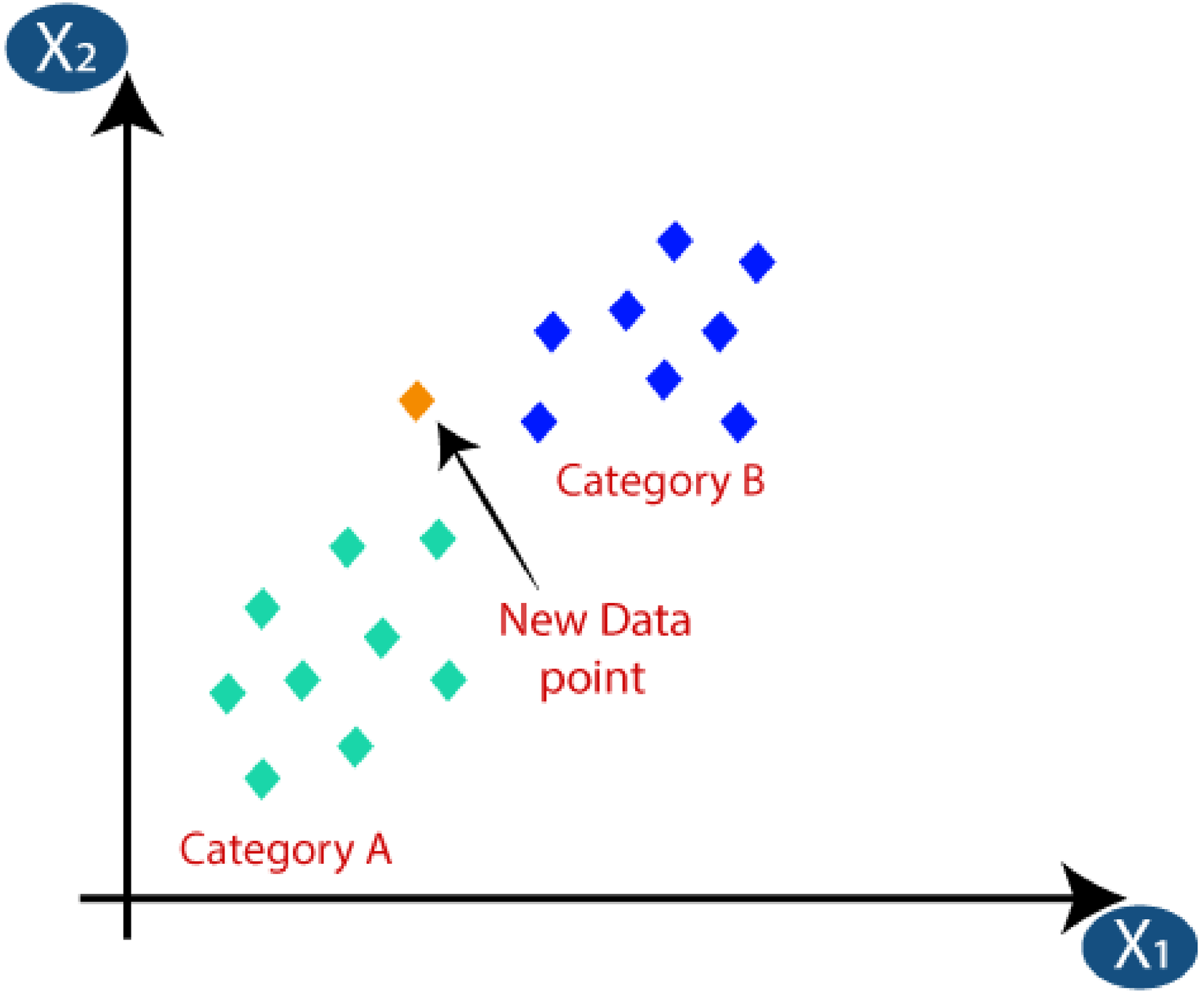
WHY DO WE NEED A K-NN ALGORITHM?

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset.

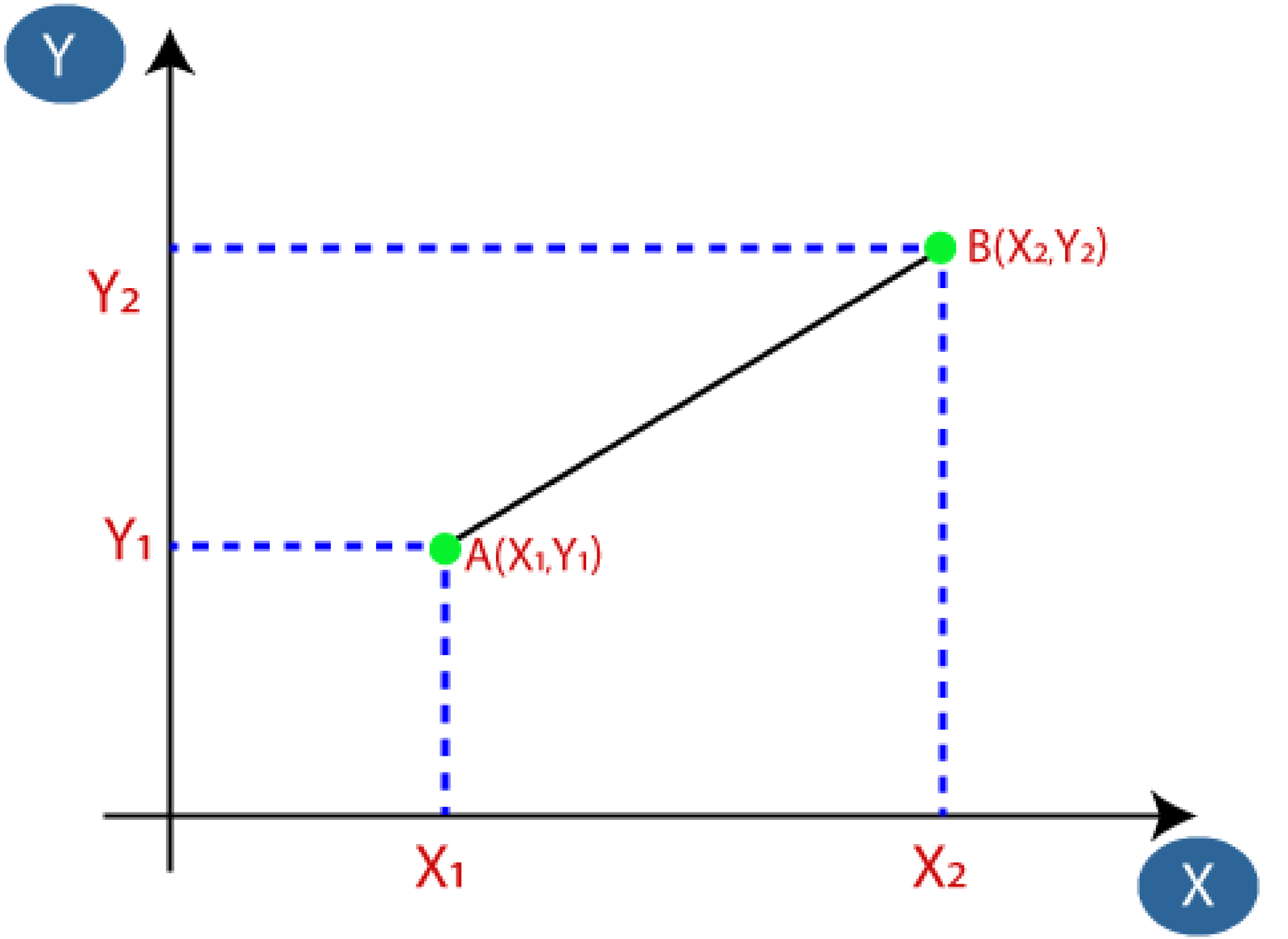

Before K-NN
X2
Category B
New data point
Category A
X1
K-NN
After K-NN
X2
Category B
New data point
assigned to
Category 1
Category A
X1

HOW DOES K-NN WORK?

Step-1: Select the number K of the neighbors
Step-2: Calculate the Euclidean distance of K number of neighbors
Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.
Step-4: Among these k neighbors, count the number of the data points in each category.
Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.
Step-6: Our model is ready.

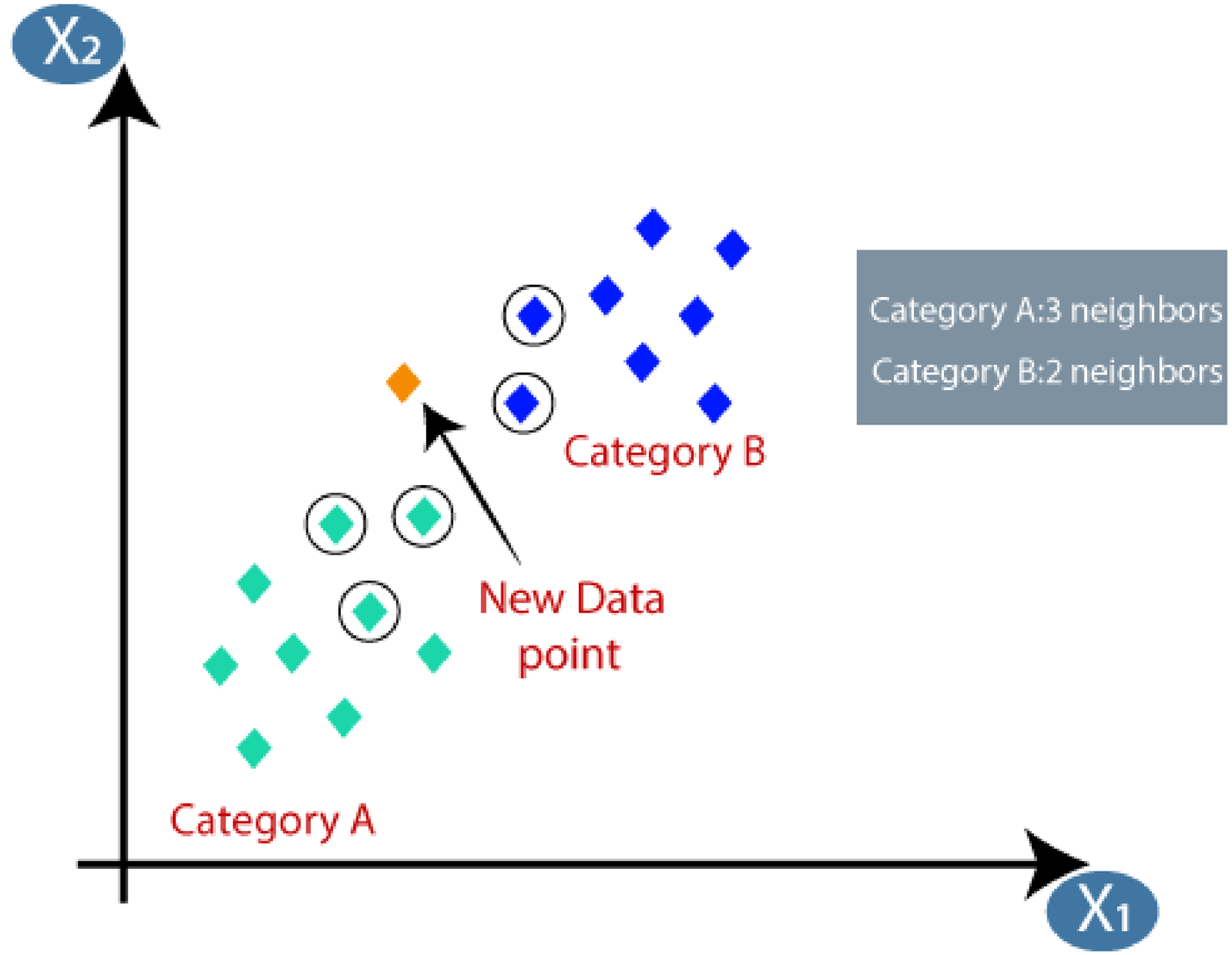

X2
Category B
New Data
point
Category A
X1

- Firstly, we will choose the number of neighbors, so we will choose the k=5.
- Next, we will calculate the Euclidean distance between the data points.
- The Euclidean distance is the distance between two points, which we have already studied in geometry. It can be calculated as:



Euclidean Distance between A_1 and B_2 = $\sqrt{(X_2-X_1)^2+(Y_2-Y_1)^2}$

- By calculating the Euclidean distance we got the nearest neighbors, as three nearest neighbors in category A and two nearest neighbors in category B.
- As we can see the 3 nearest neighbors are from category A, hence this new data point must belong to category A.



BRIGHTNESS	SATURATION	CLASS
40	20	Red
50	50	Blue
60	90	Blue
10	25	Red
70	70	Blue
60	10	Red
25	80	Blue

BRIGHTNESS	SATURATION	CLASS
20	35	?