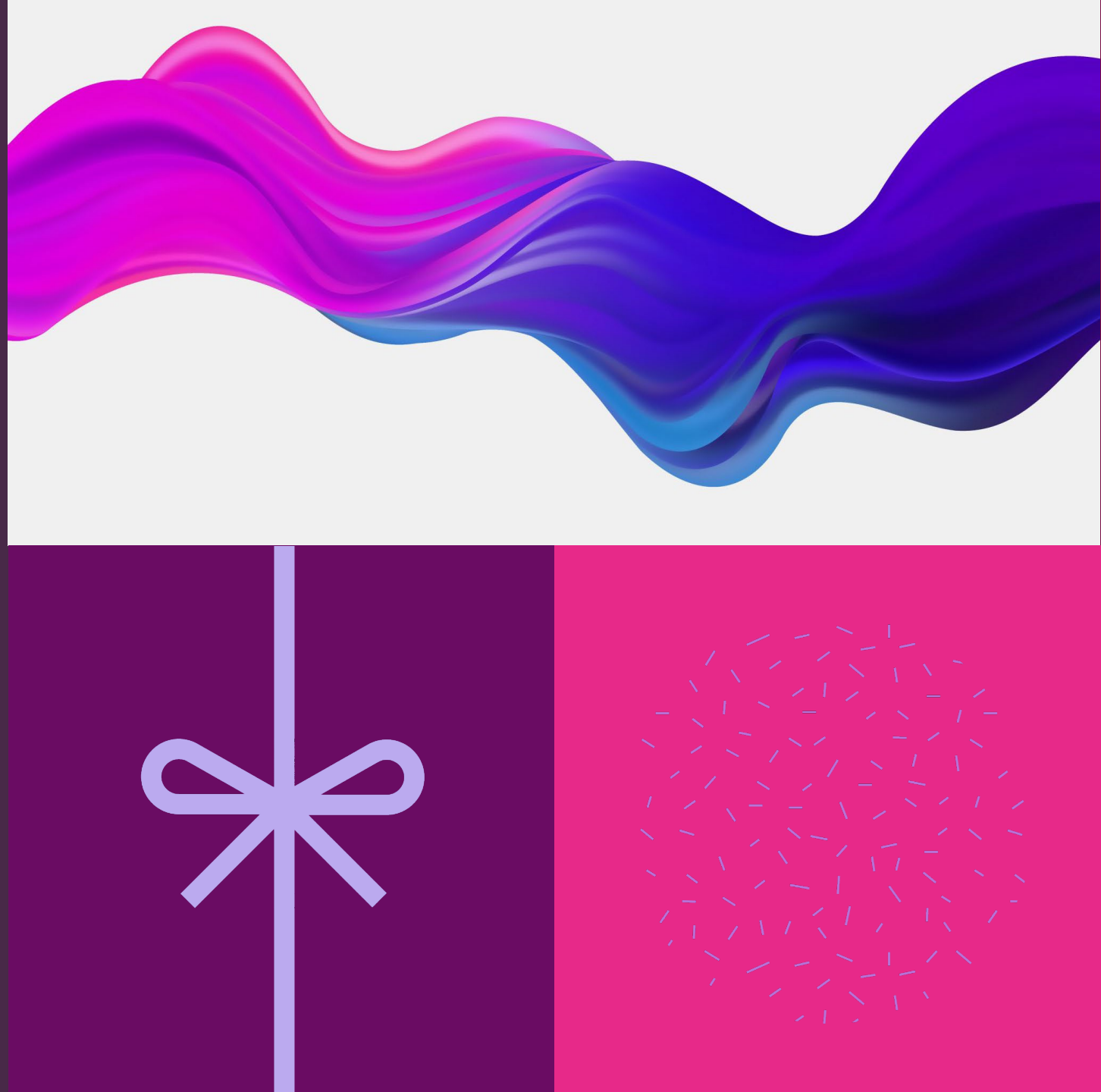
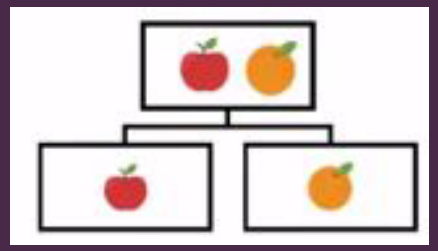
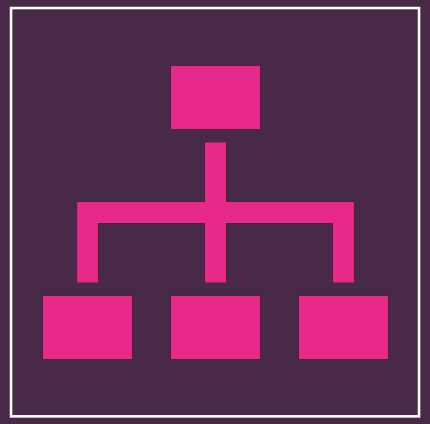
clustering

Clustering is a type of unsupervised learning wherein data points are grouped into different sets based on their degree of similarity.



Types of clustering





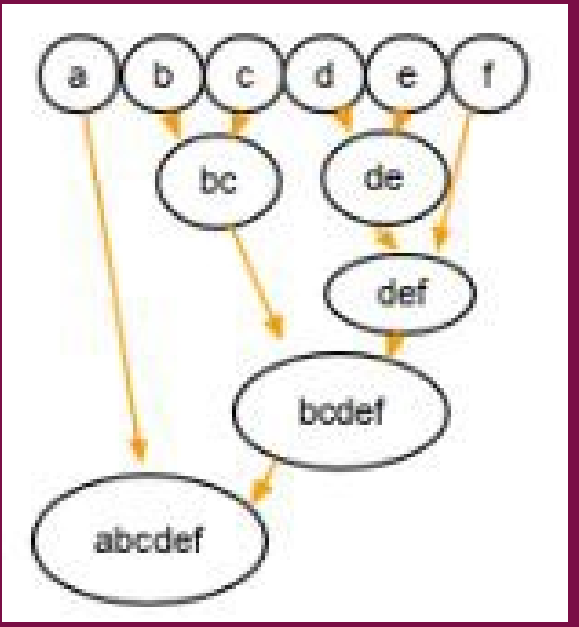
Hierarchical clustering

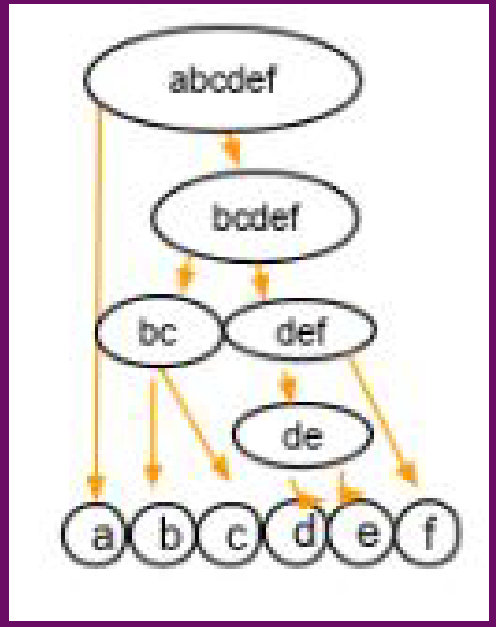
Partitioning clustering

Hierarchical clustering

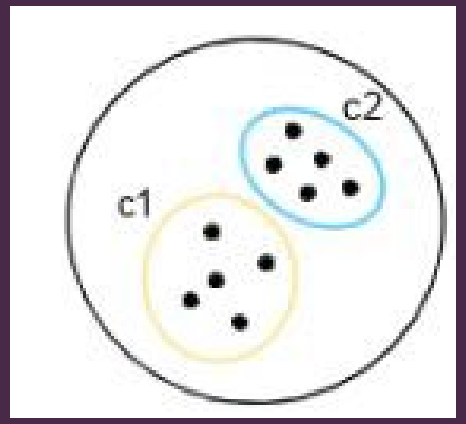
Agglomerative clustering

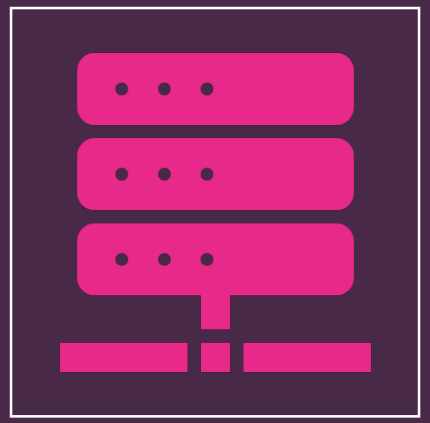
Divisive clustering



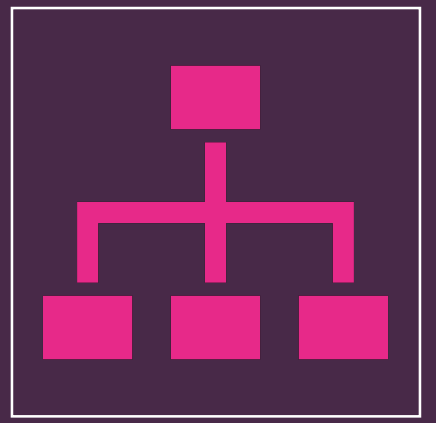


Partitioning clustering

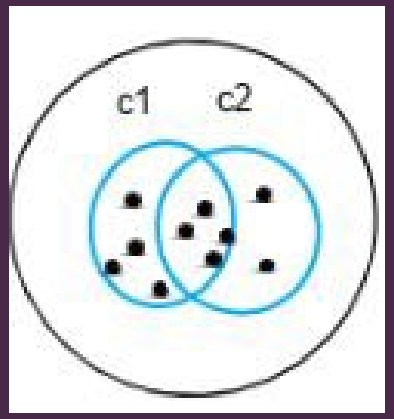




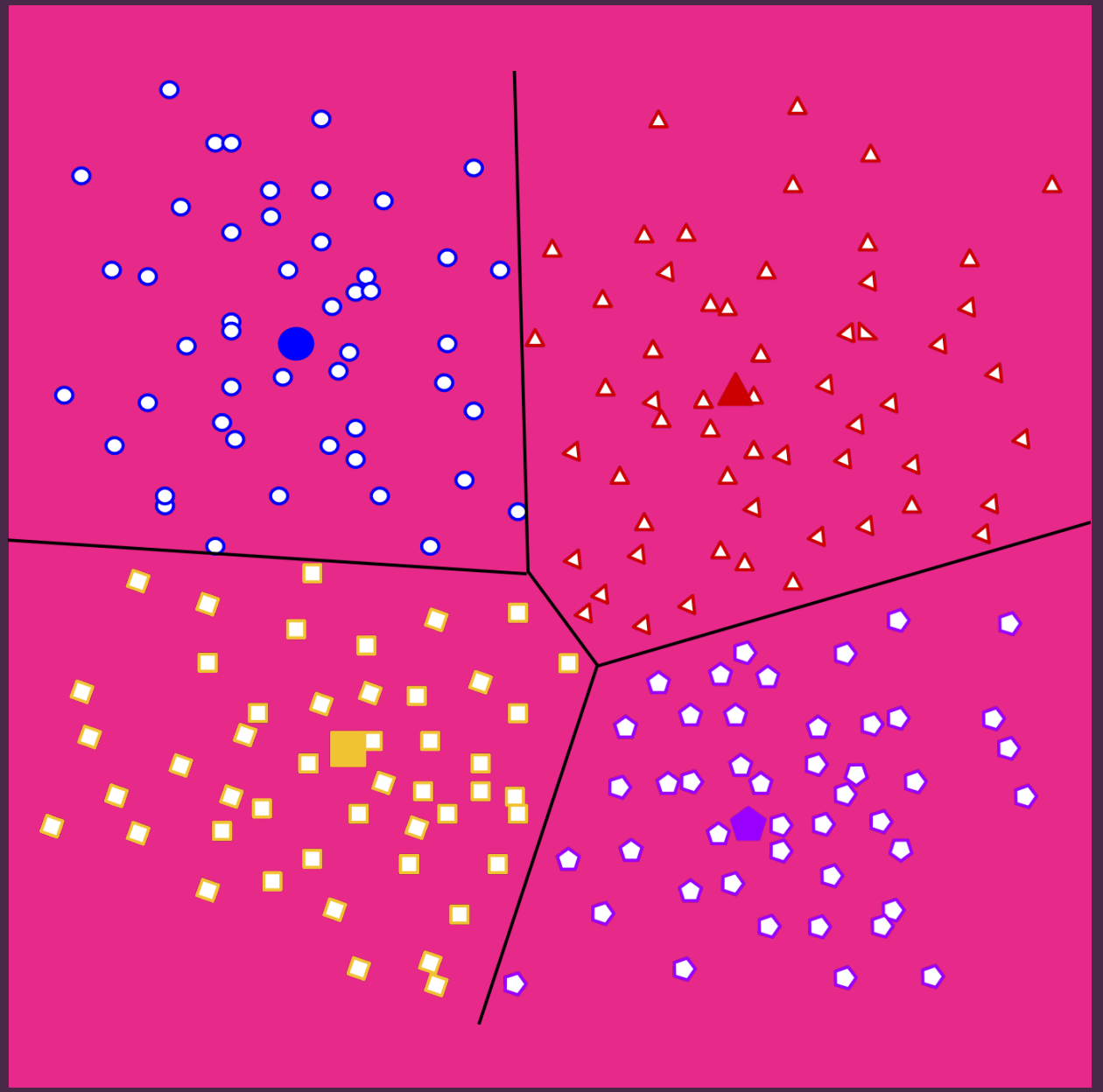
K-Means clustering



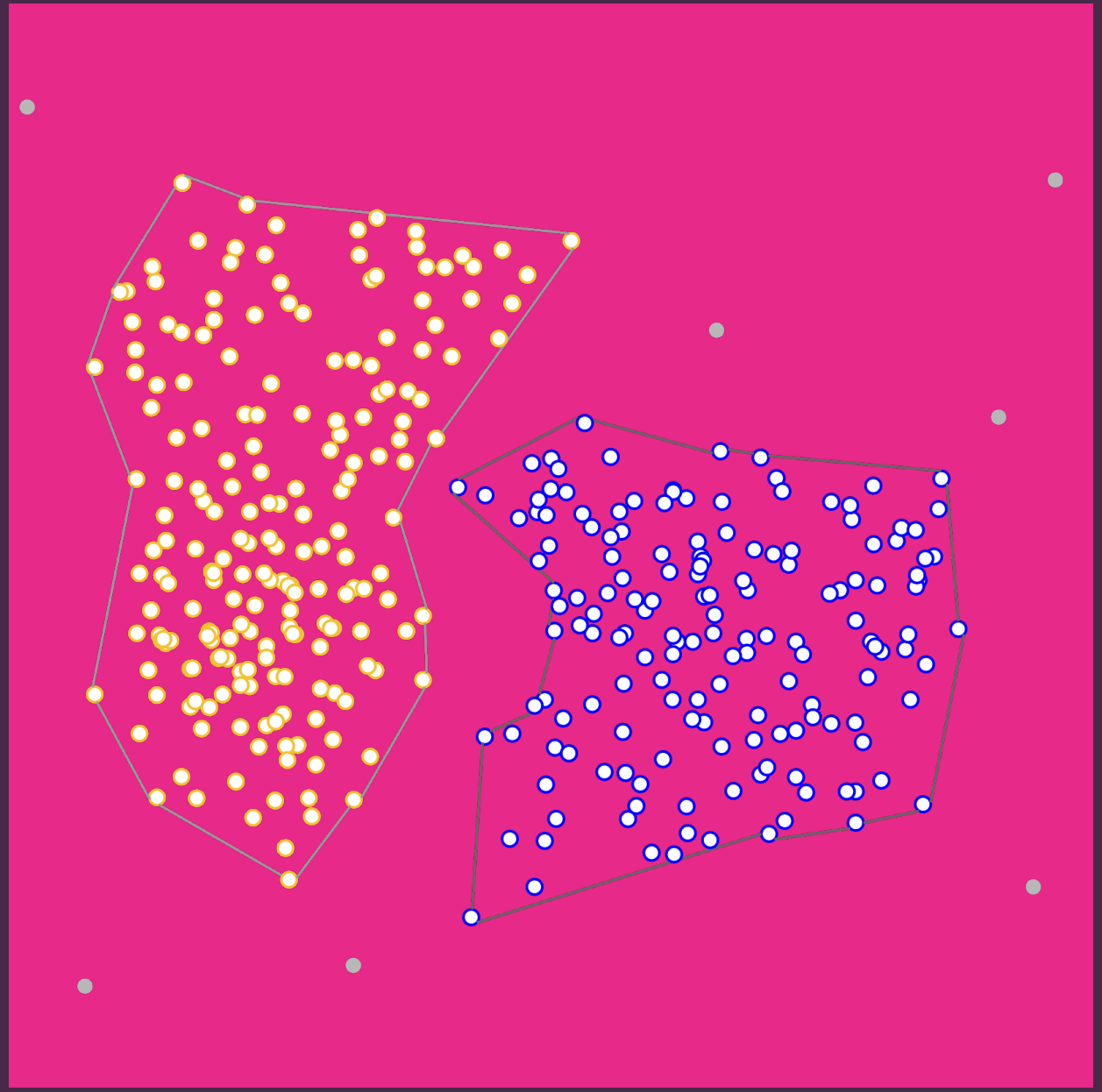
Fuzzy C-Means clustering



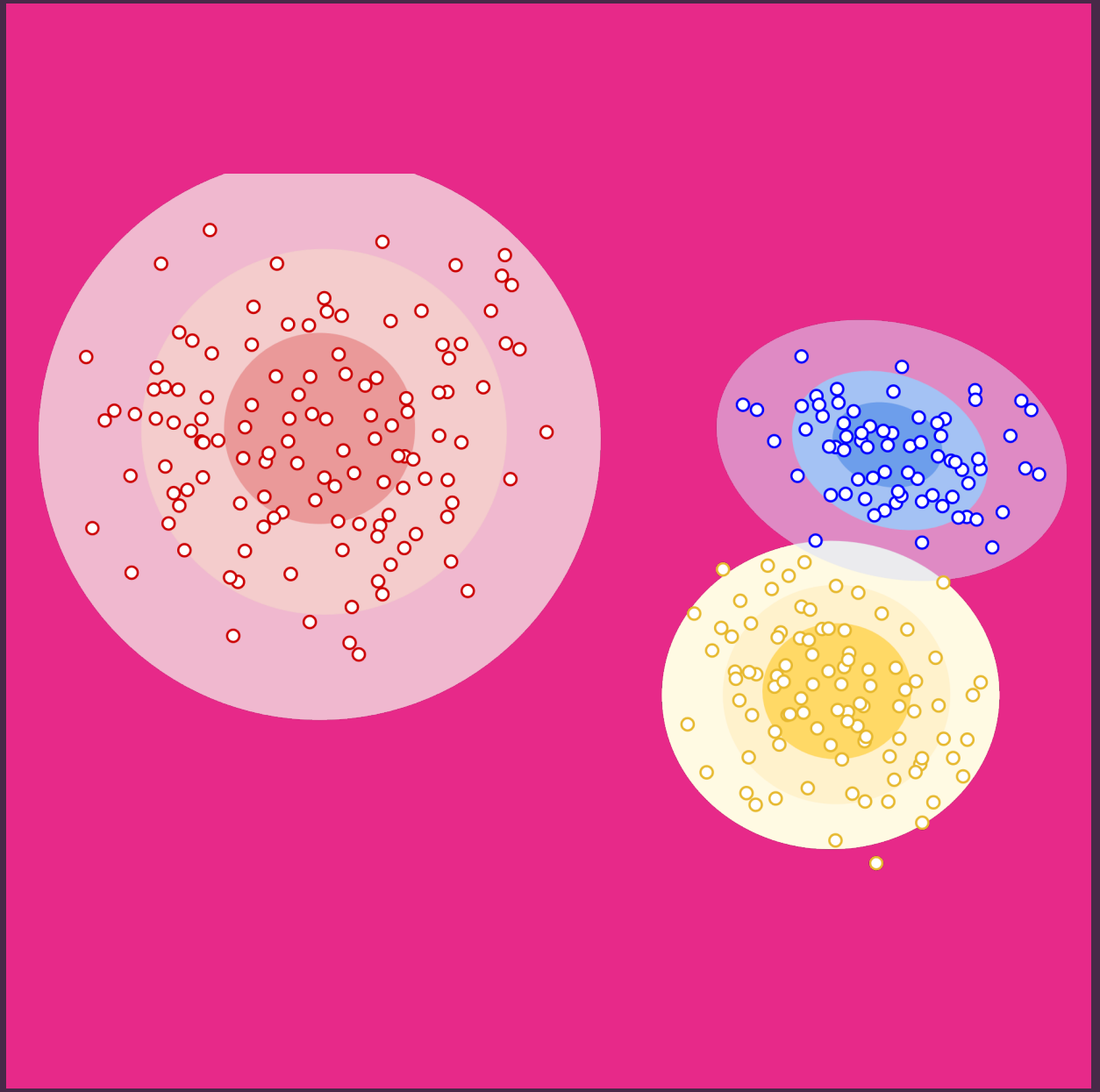
Centroid based clustering



Density based clustering



Distribution-based Clustering



K MEANS CLUSTERING

It is a centroid-based algorithm, where each cluster is associated with a centroid.

The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters.

K means clustering

K-Means performs the division of objects into clusters that share similarities and are dissimilar to the objects belonging to another cluster.

The term ‘K’ is a number. You need to tell the system how many clusters you need to create.

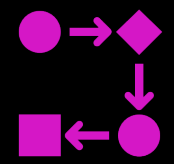
For example, K = 2 refers to two clusters.

K MEANS CLUSTERING

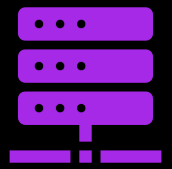
The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters.

The value of k should be predetermined in this algorithm.

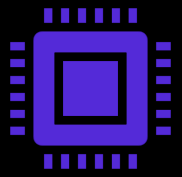
STEPS



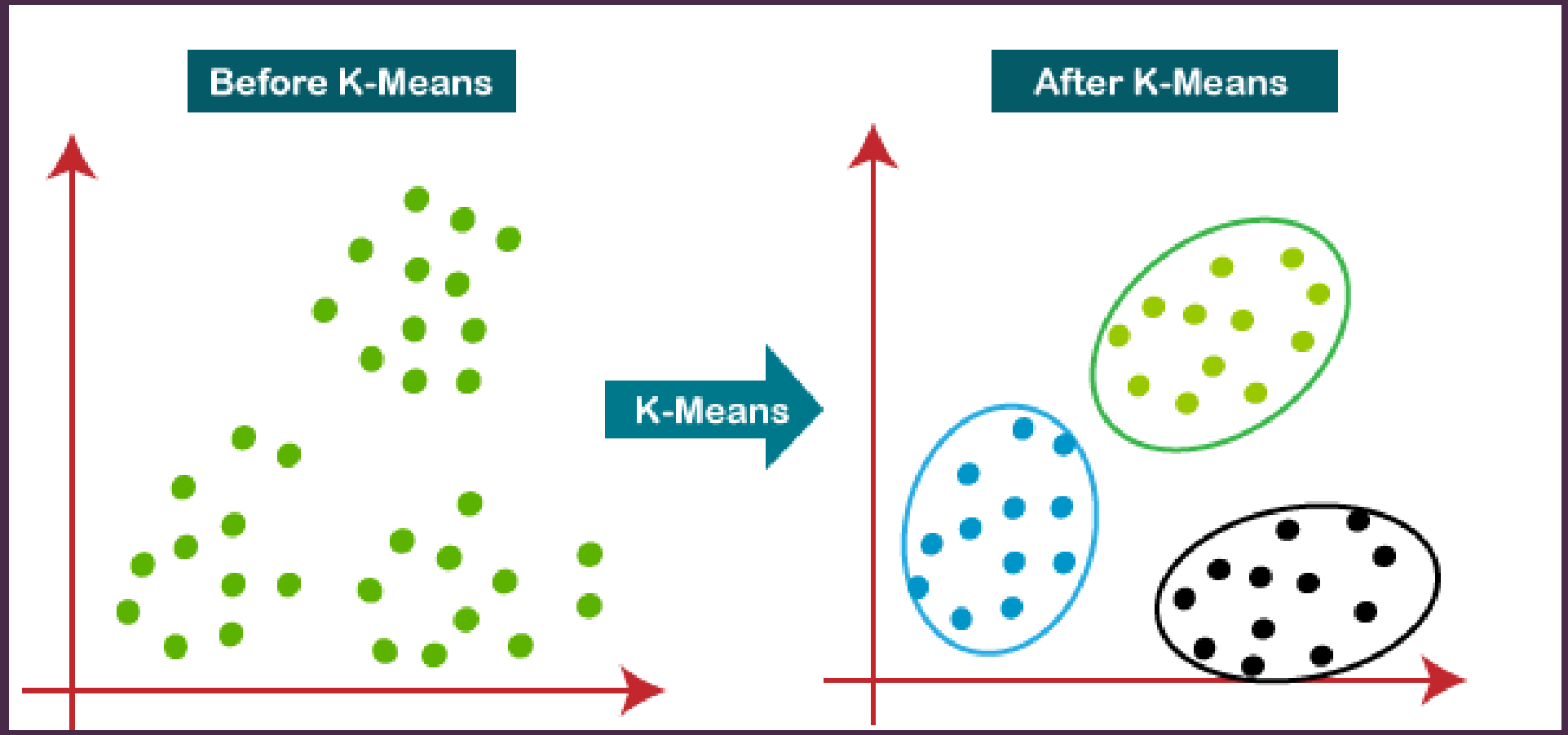
Determines the best value for K center points or centroids by an iterative process.



Assigns each data point to its closest k-center.



Those data points which are near to the particular k-center, create a cluster.


Before K-Means
K-Means
After K-Means

ALGORITHM

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be other from the input dataset).

Step-3: Assign each data point to their closest centroid, forming the predefined K clusters.

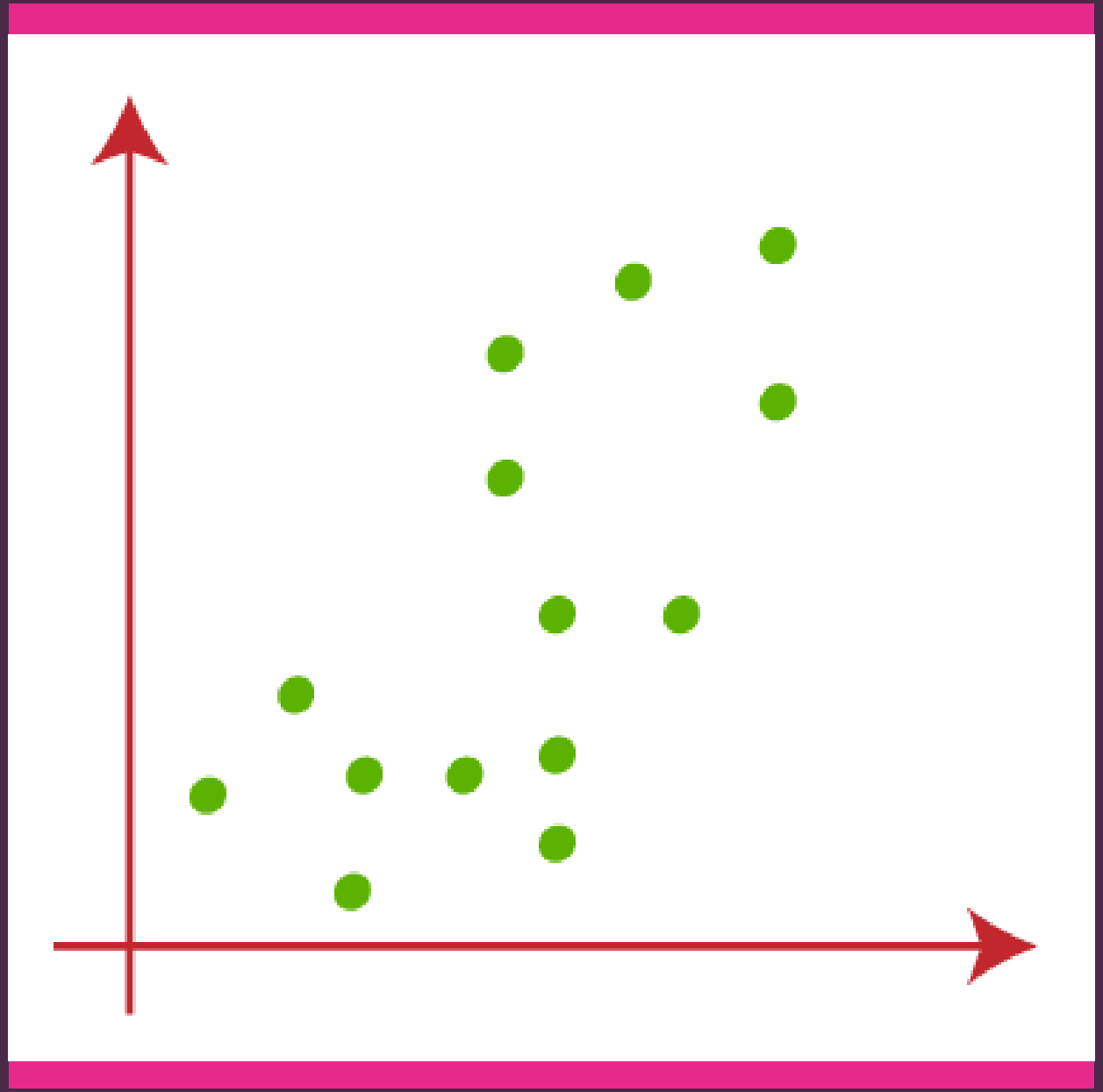
Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means reassign each datapoint to the new closest centroid of each cluster.

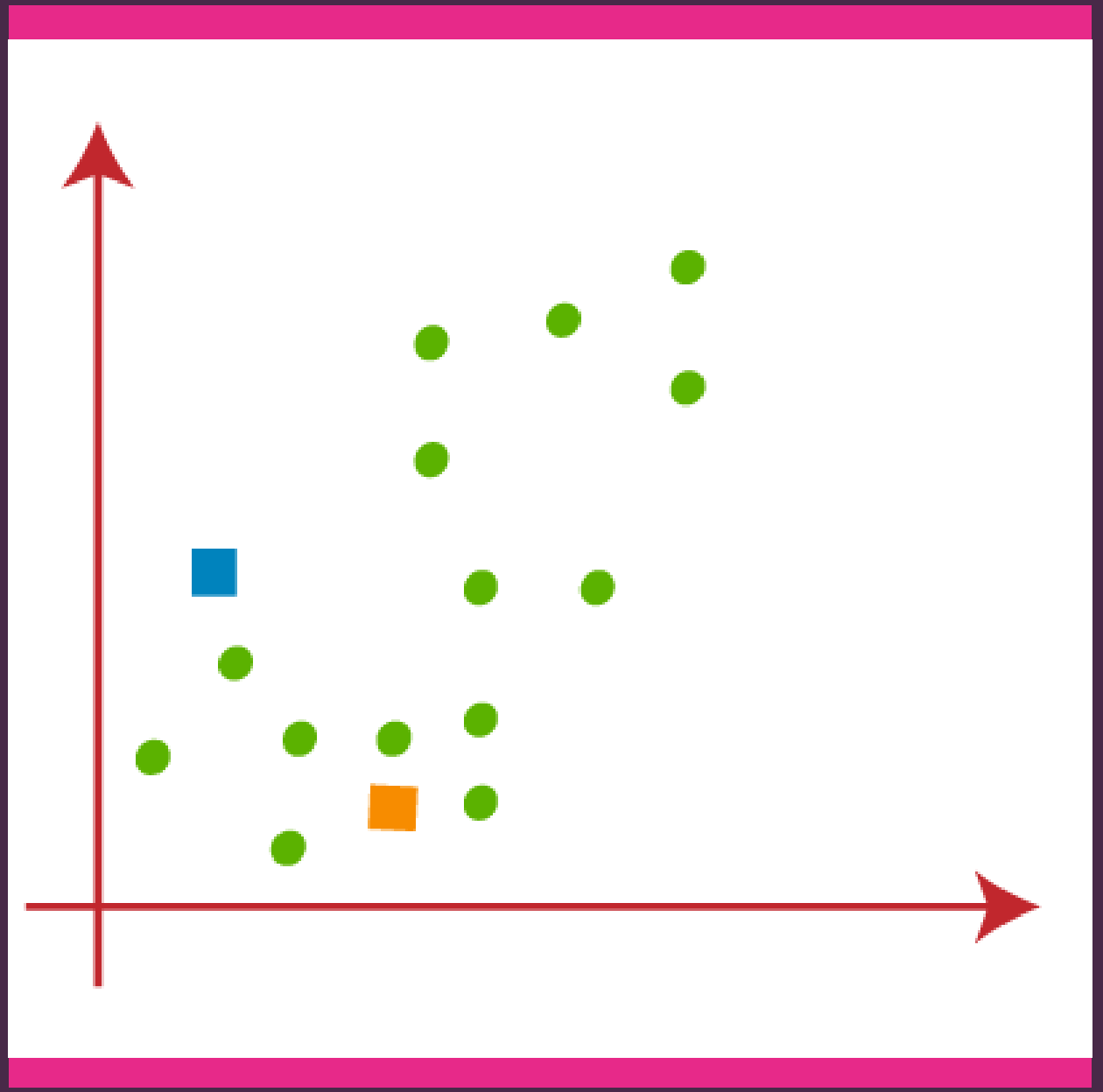
Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready

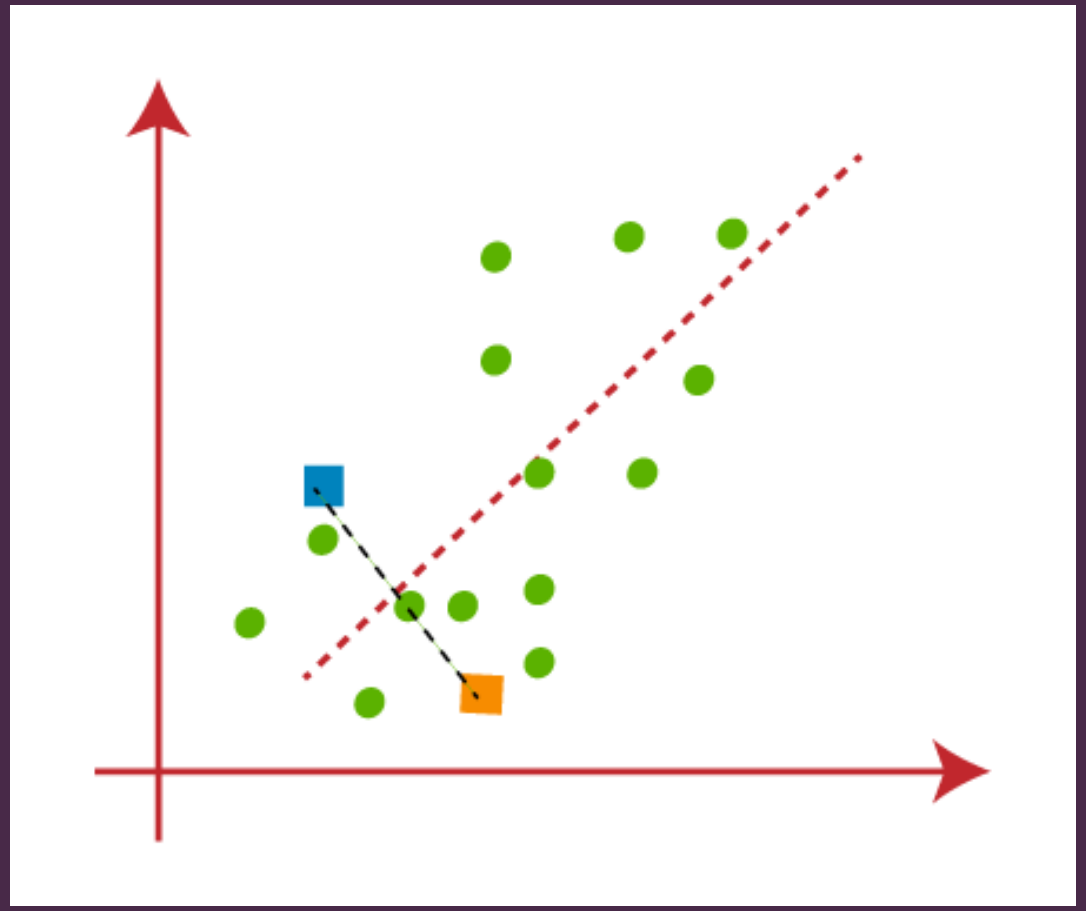
Let's take number k of clusters, i.e., K=2, to identify the dataset and to put them into different clusters. It means here we will try to group these datasets into two different clusters.



choose some random k points or centroid to form the cluster. These points can be either the points from the dataset or any other point

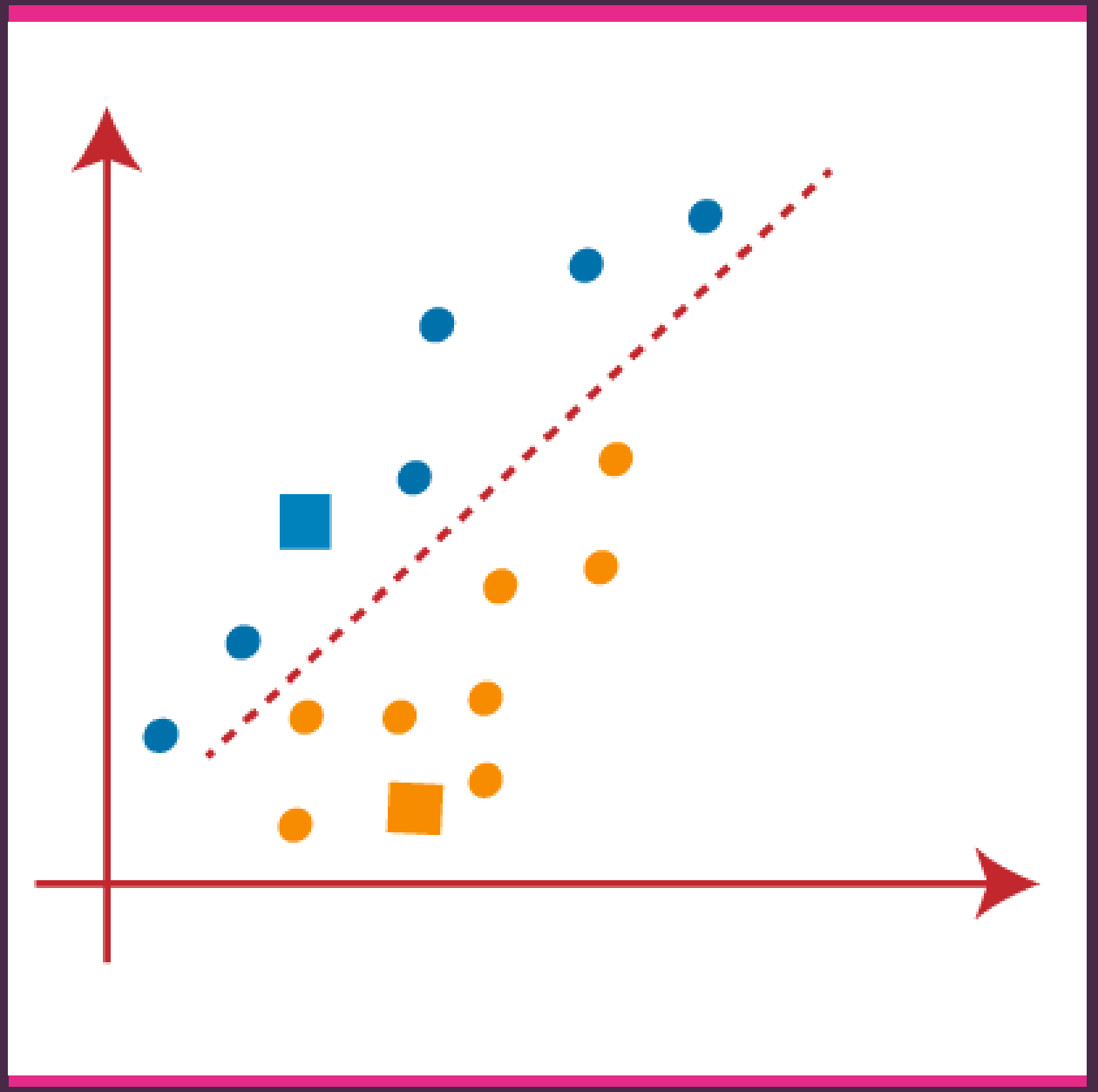


assign each data point of the scatter plot to its closest K-point or centroid



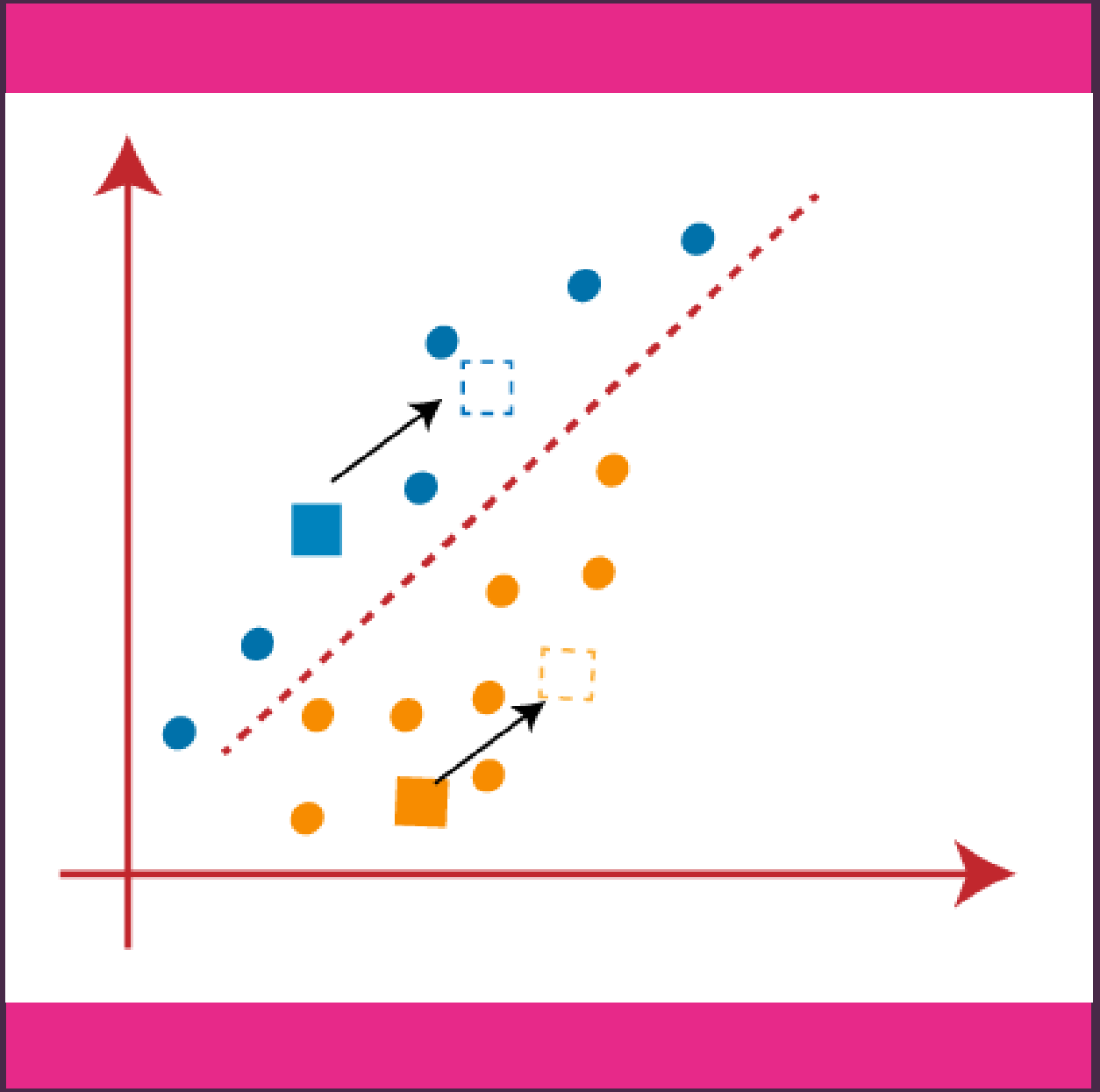
draw a median between both the centroids

- points left side of the line is near to the K1 or blue centroid
- points to the right of the line are close to the yellow centroid.

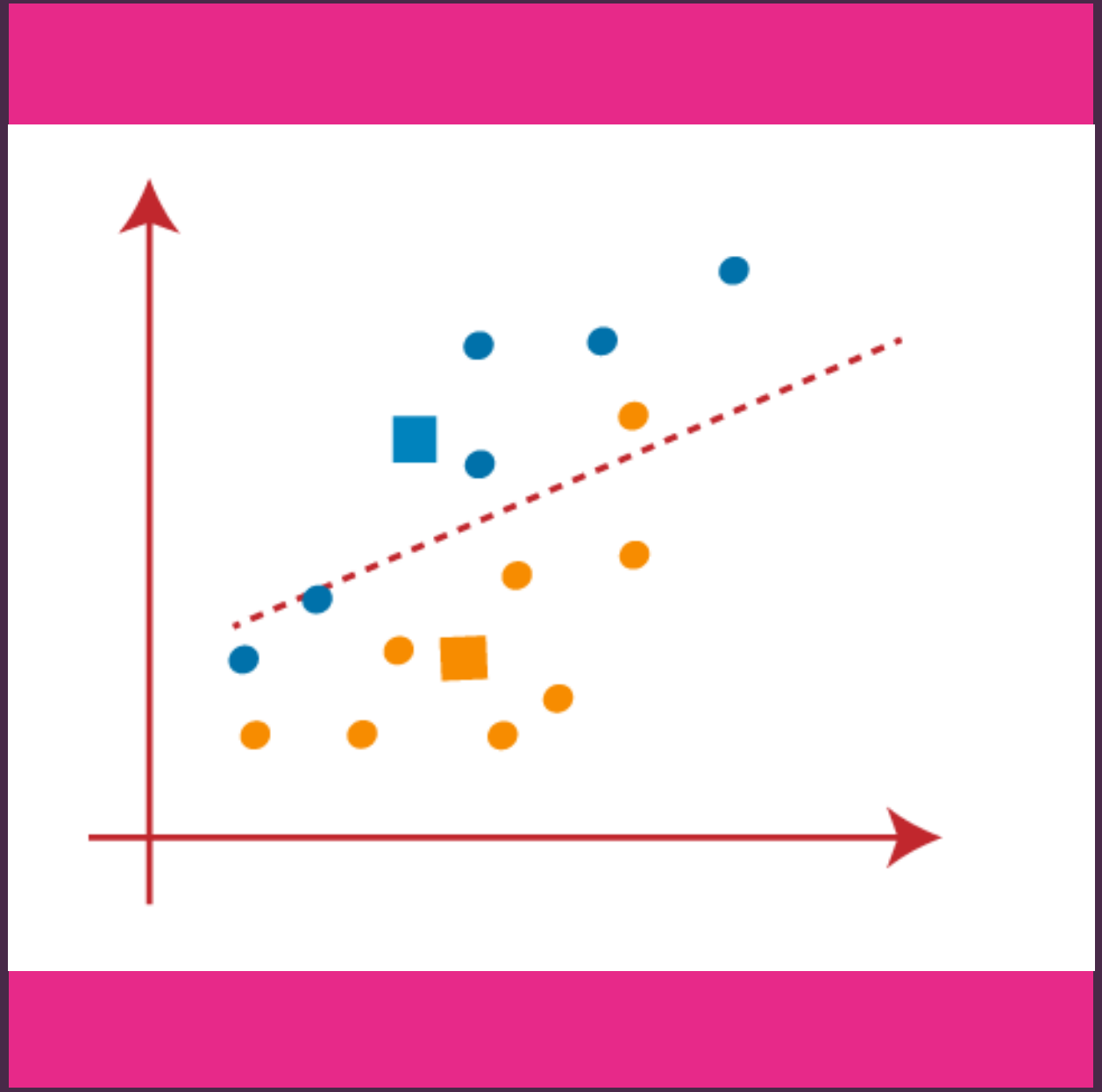


As we need to find the closest cluster, so we will repeat the process by choosing a new centroid.

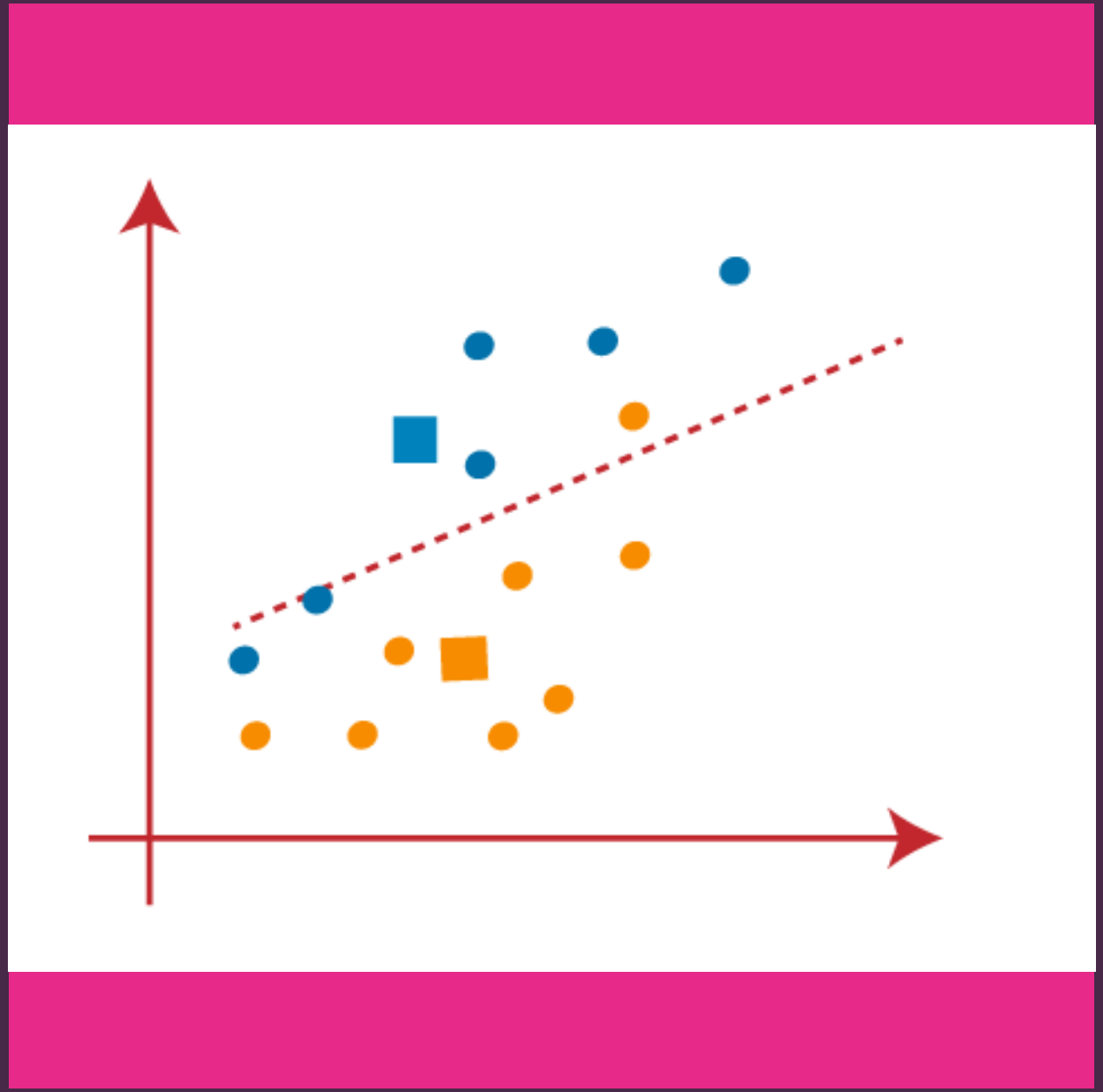
To choose the new centroids, we will compute the center of gravity of these centroids, and will find new centroids



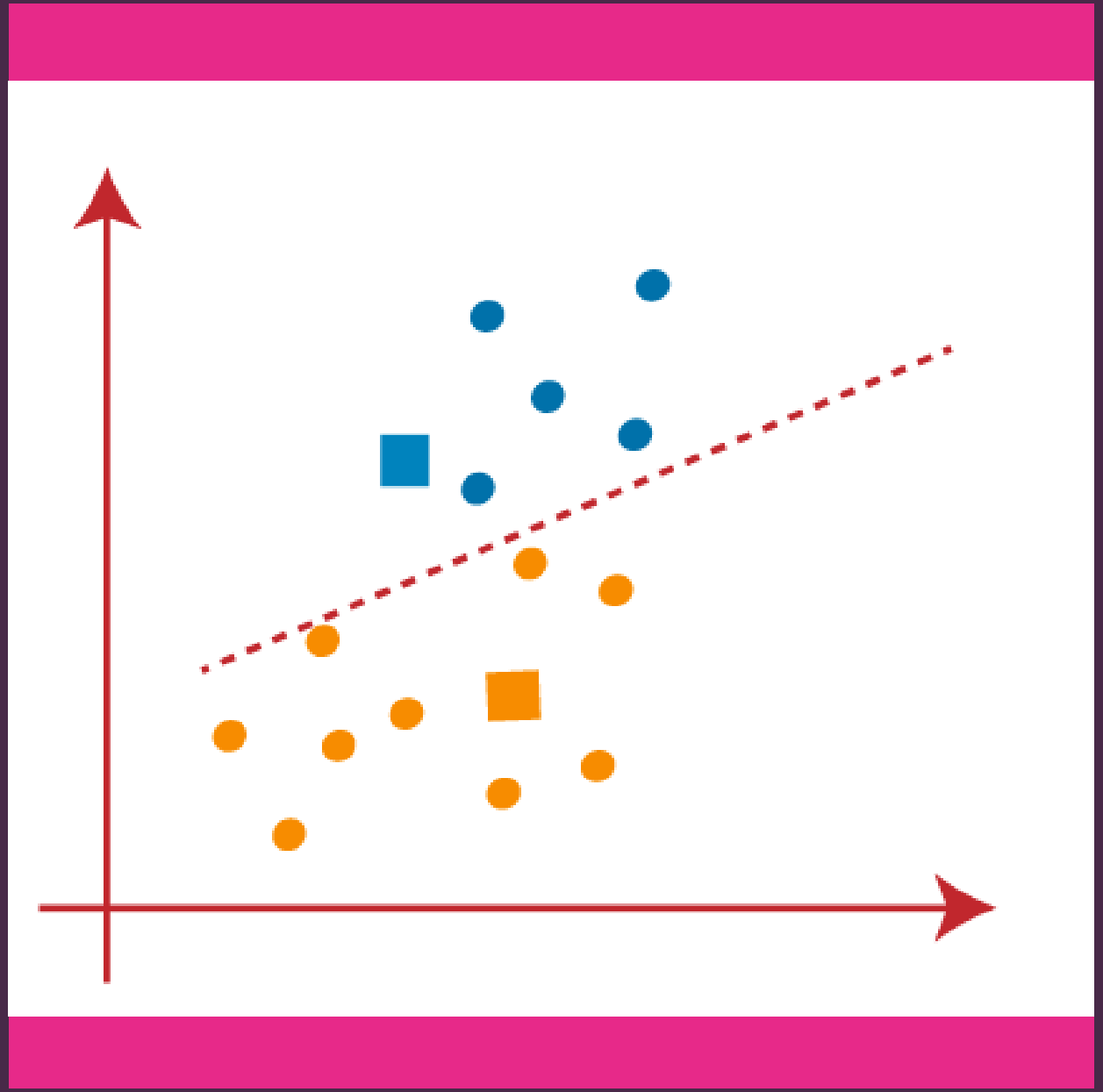
Next, we will reassign each datapoint to the new centroid. For this, we will repeat the same process of finding a median line.



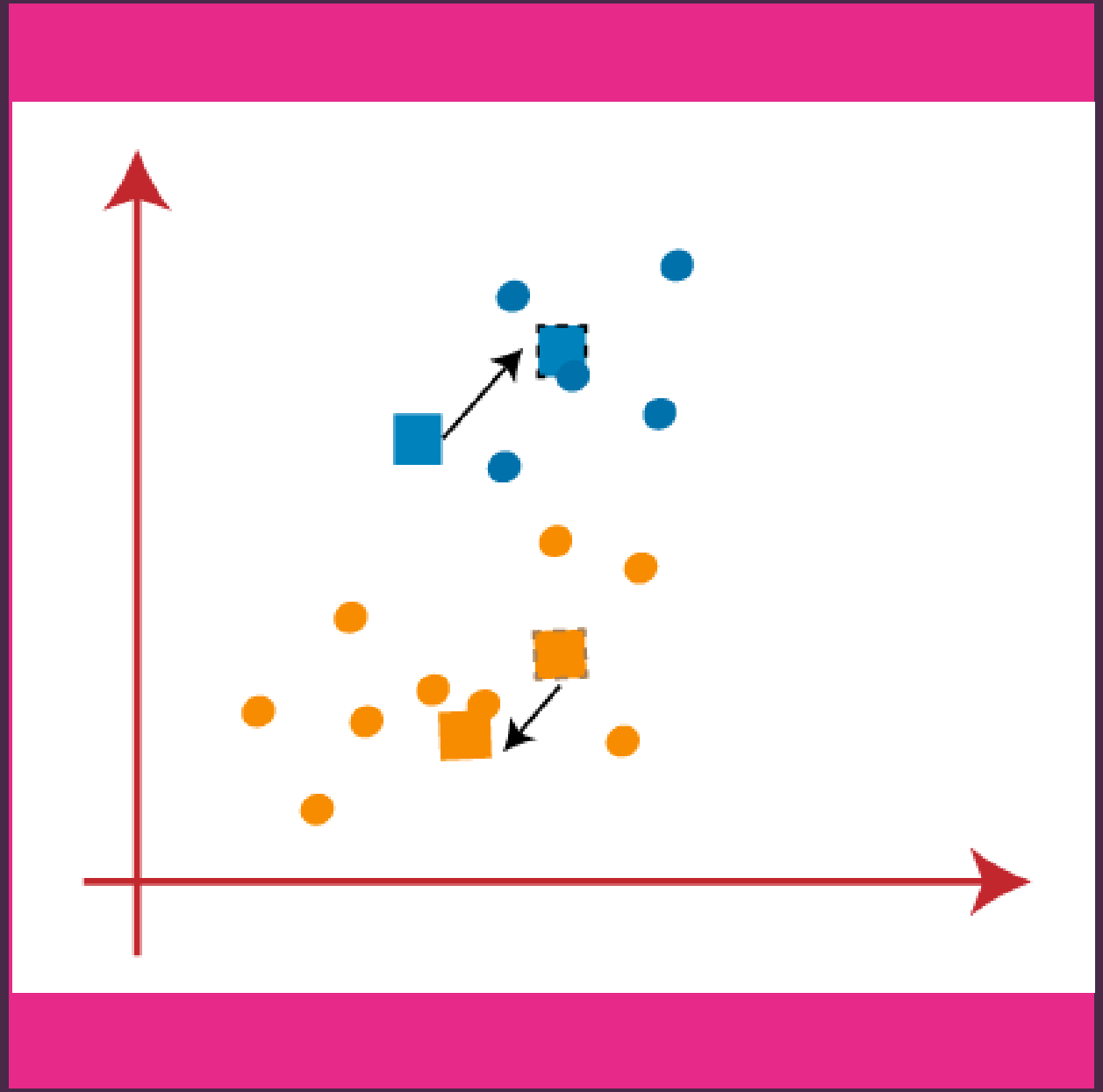
One yellow point is on the left side of the line, and two blue points are right to the line. So, these three points will be assigned to new centroids



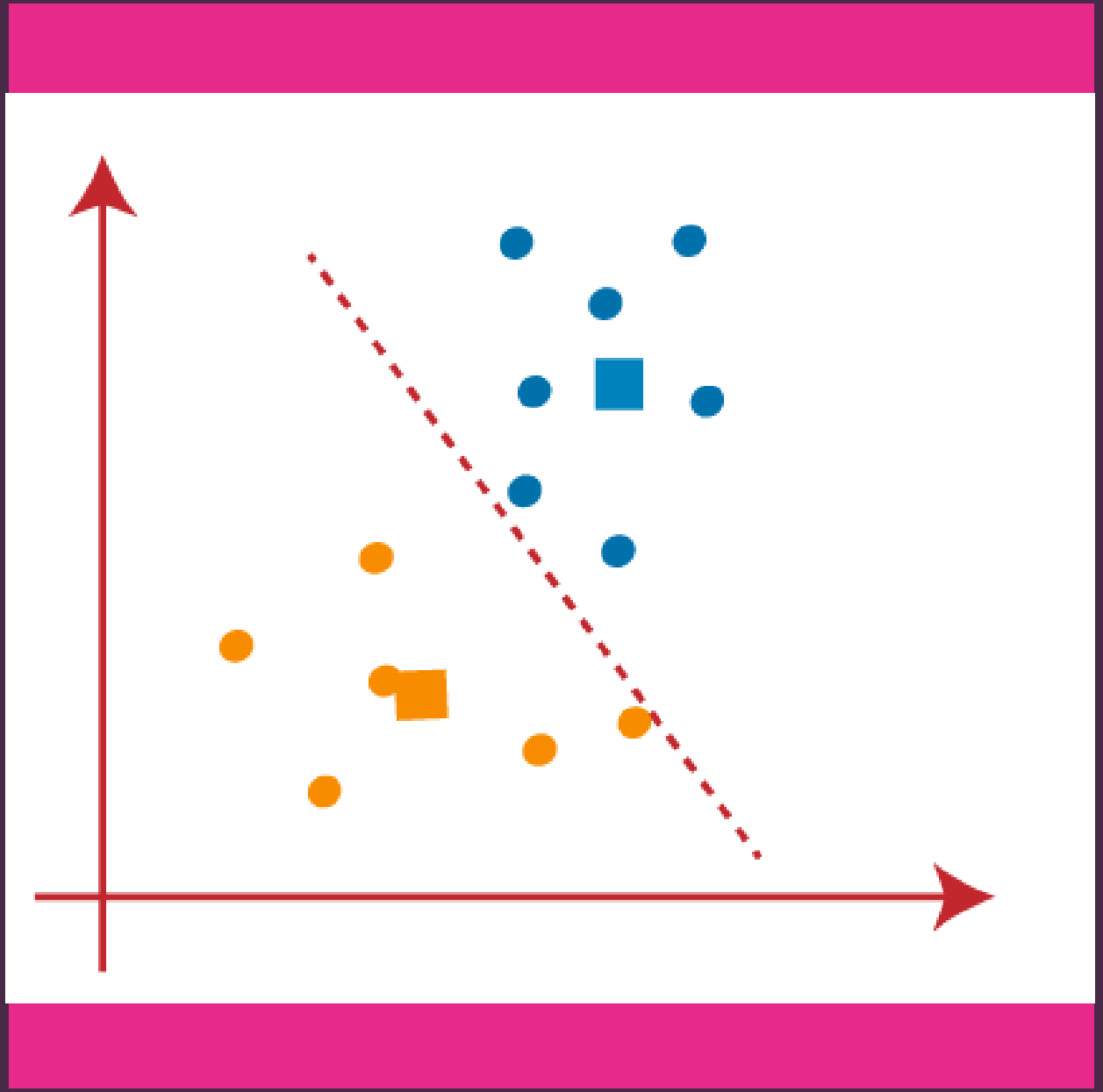
As reassignment has taken place, so we will again go to the step-4, which is finding new centroids or K-points.



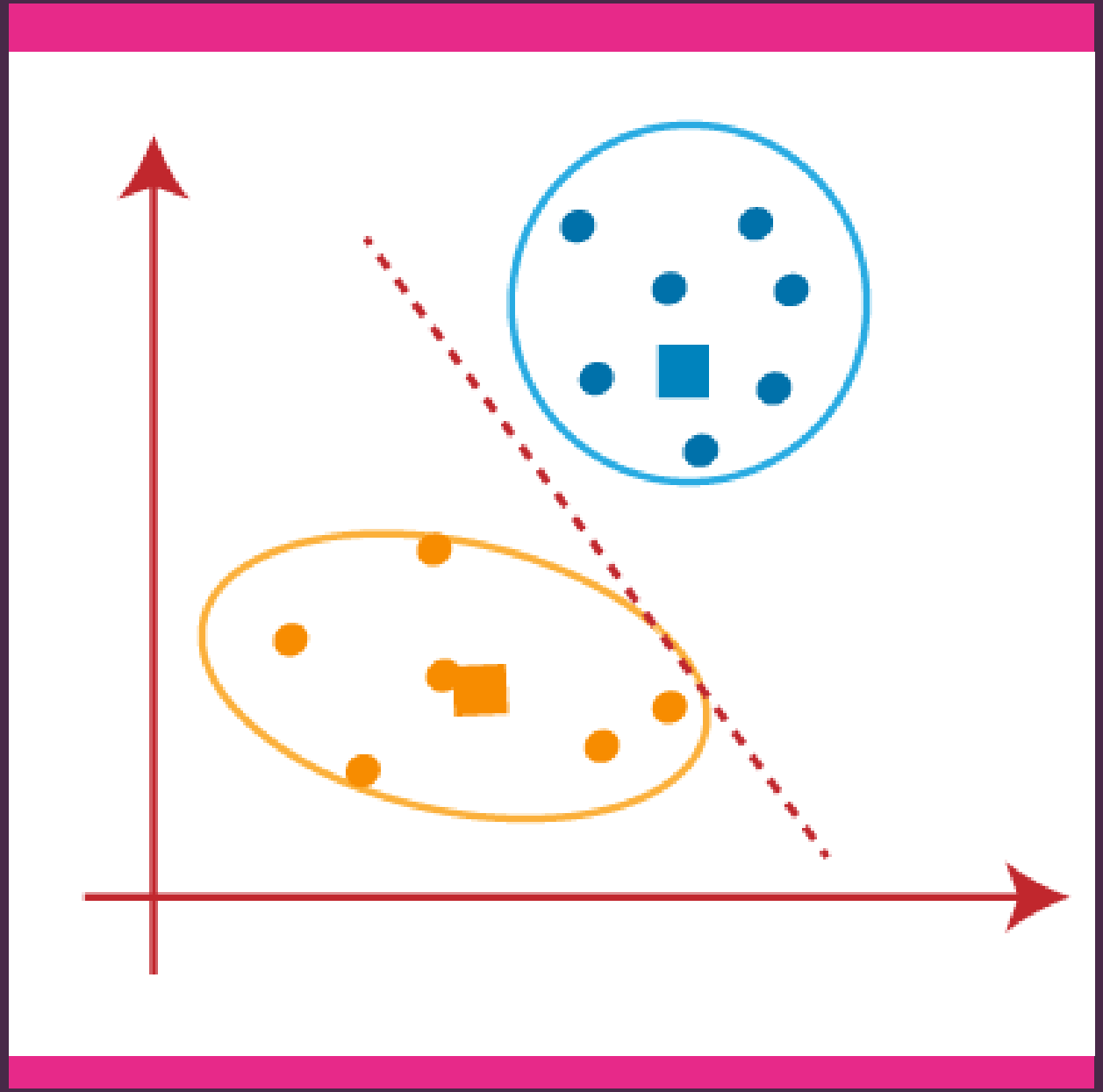
repeat the process by finding the center of gravity of centroids, so the new centroids →



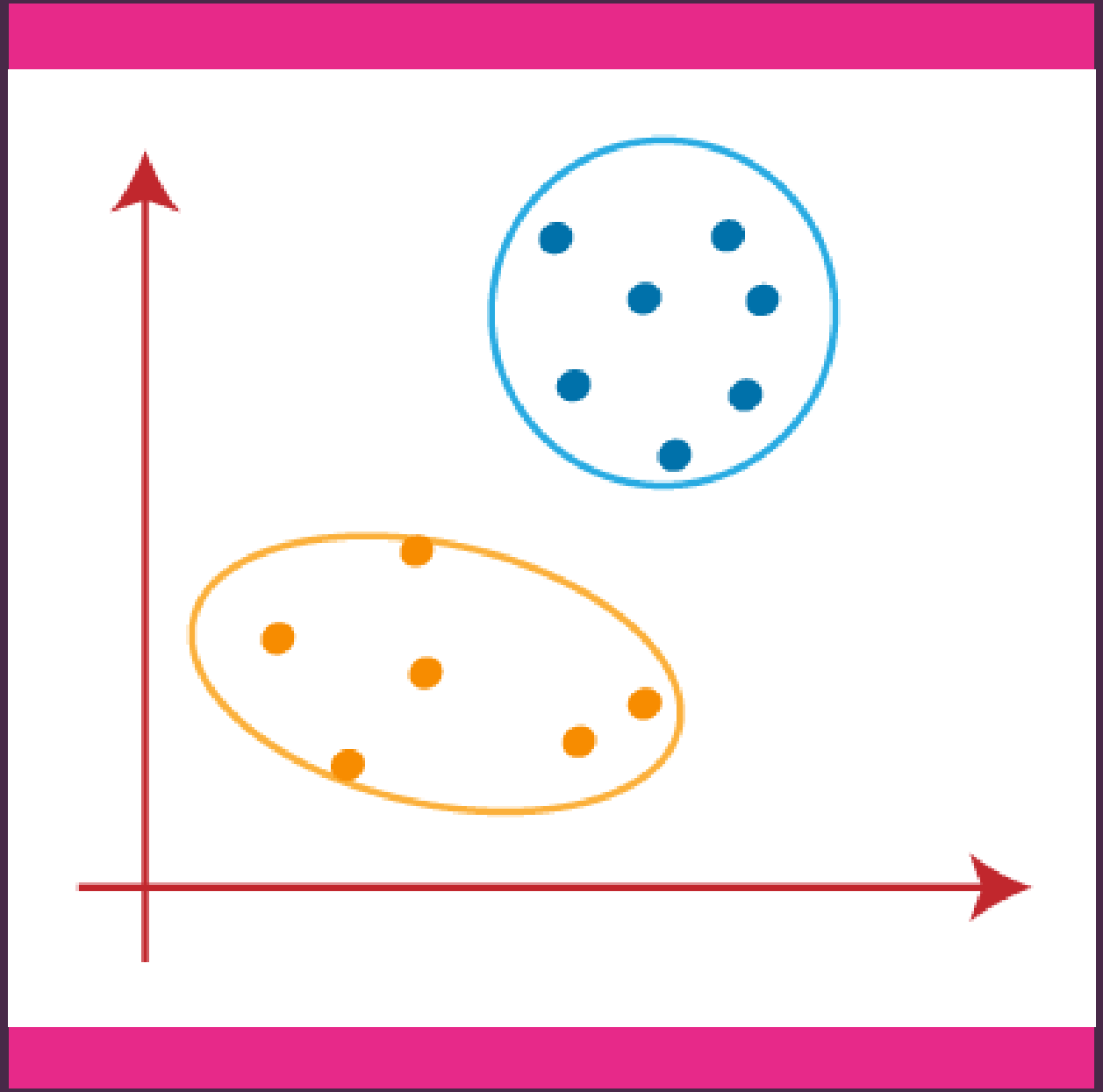
new → so again will draw the median line and reassign the data points



there are no dissimilar data points on either side of the line, which means our model is formed



now remove the assumed centroids



Euclidean Distance between two points in space

$$d(\mathbf{p}, \mathbf{q}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2}.$$

If each cluster centroid is denoted by ci, then each data point x is assigned to a cluster based on

$$\arg\min_{c_i \in C} dist(c_i, x)^2$$

Finding the new centroid from the clustered group of points

$$c_i = \frac{1}{|S_i|} \sum_{x_i \in S_i} x_i$$

Si is the set of all points assigned to the ith cluster.

Instance based learning

- The Machine Learning systems which are categorized as instance-based learning are the systems that learn the training examples by heart and then generalizes to new instances based on some similarity measure.
- It is called instance-based because it builds the hypotheses from the training instances.
- It is also known as memory-based learning or lazy-learning

Instance based learning

- Each time whenever a new query is encountered, its previously stores data is examined
- If we were to create a spam filter with an instance-based learning algorithm, instead of just flagging emails that are already marked as spam emails, our spam filter would be programmed to also flag emails that are very similar to them.

Instance based learning

- K Nearest Neighbor (KNN)
- Self-Organizing Map (SOM)
- Learning Vector Quantization (LVQ)
- Locally Weighted Learning (LWL)
- Case-Based Reasoning

KNN

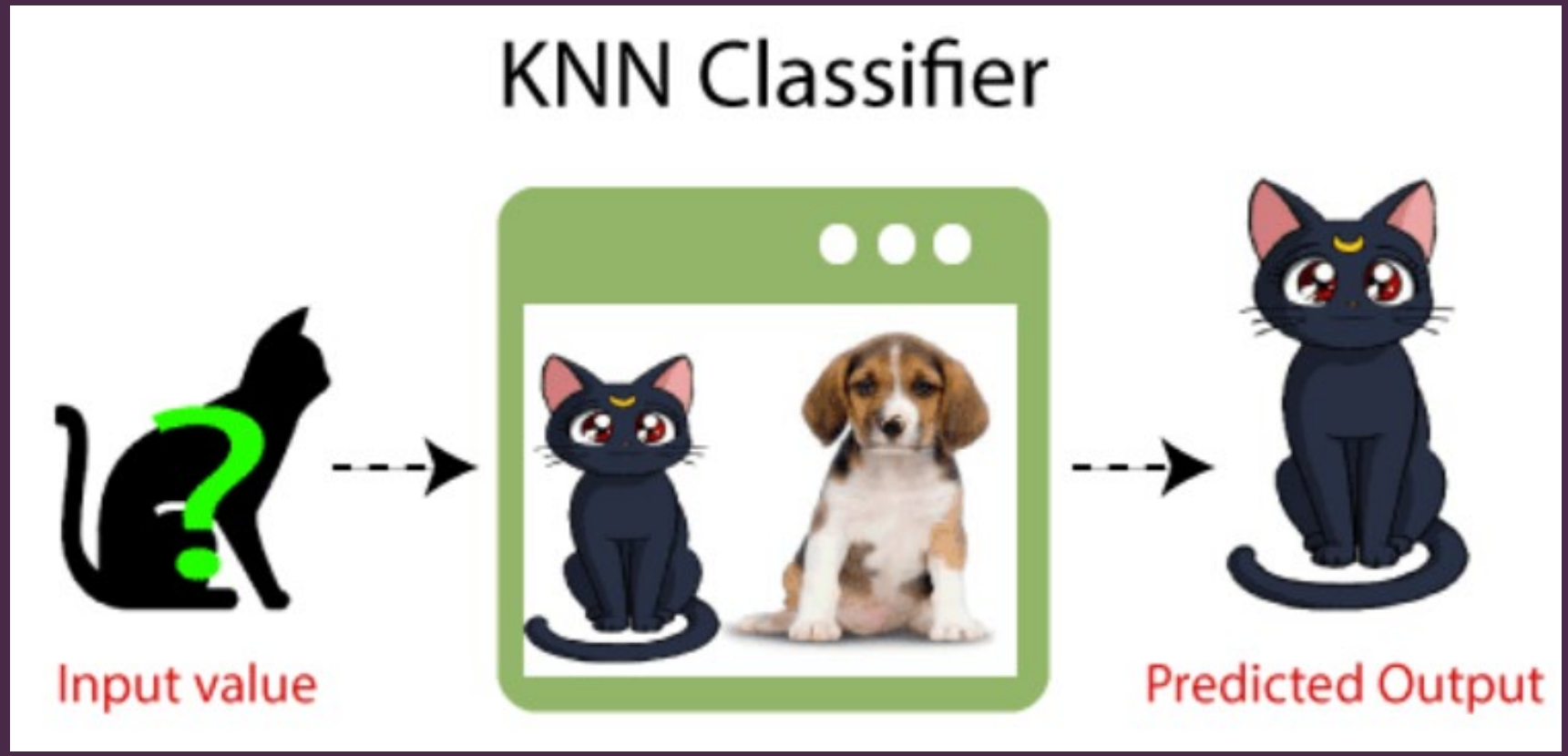
- Based on supervised learning
- Assume similarity of new case with available case and put in a suitable category
- Uses for regression and classification mostly classification

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.

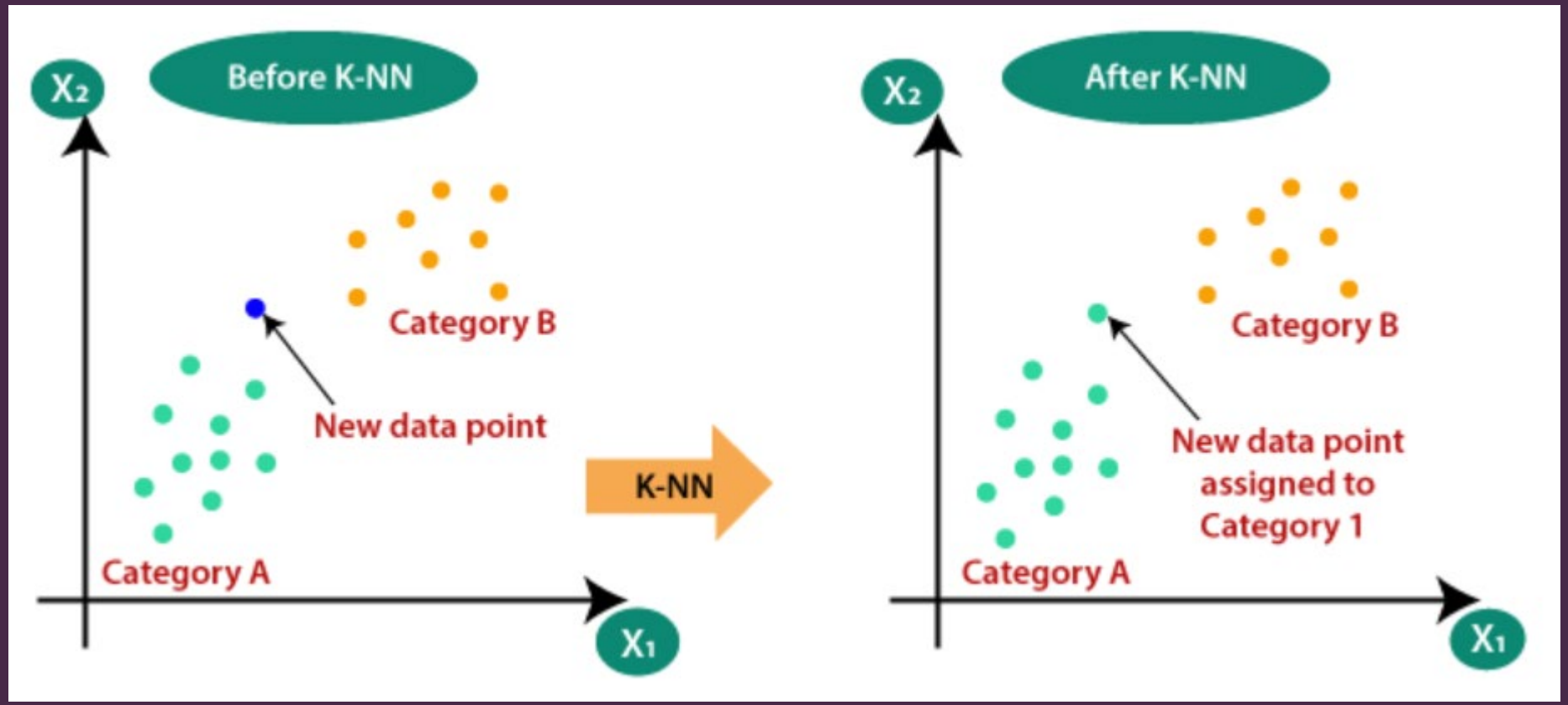
KNN

- Works on similarity measure



KNN why ?

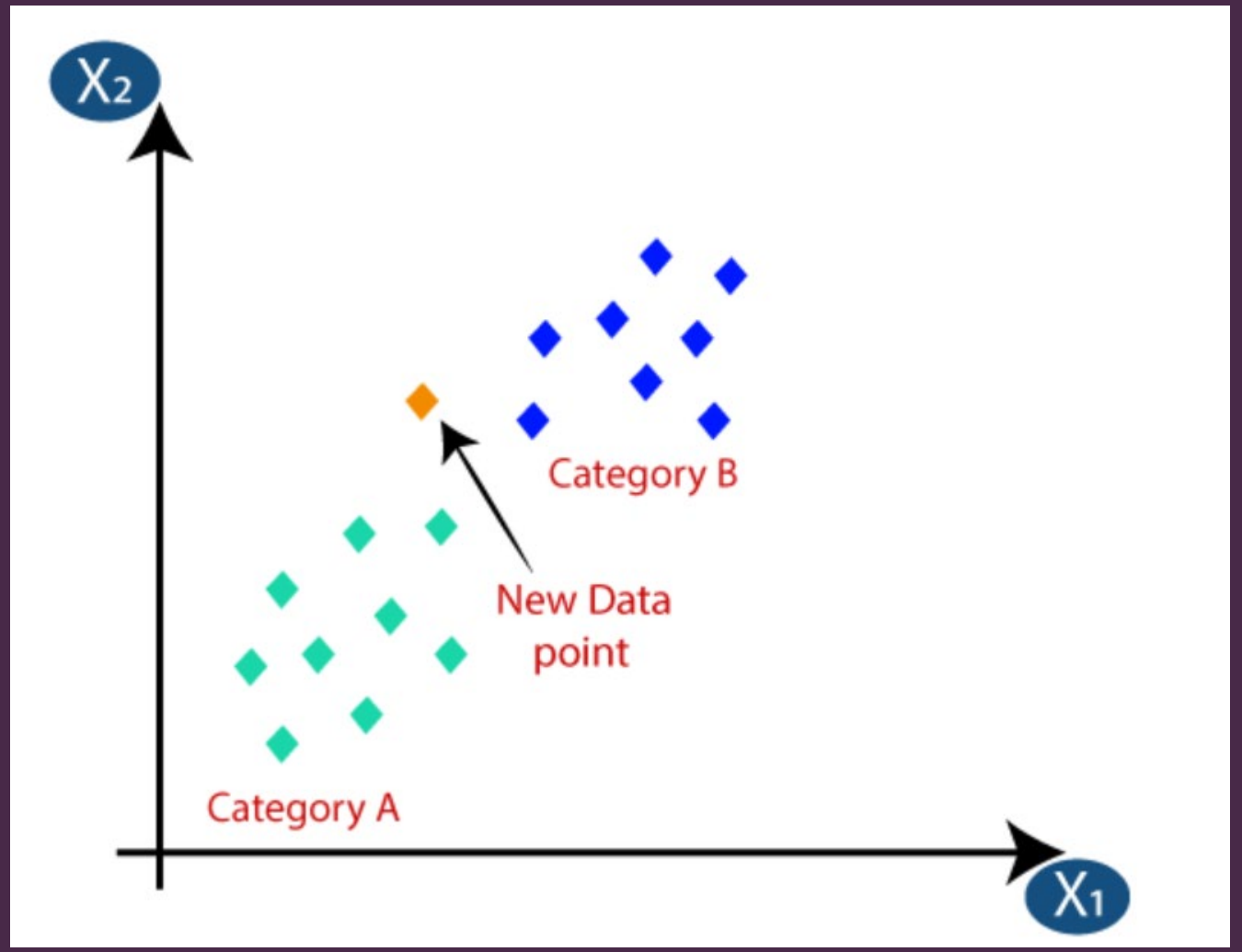
- To which category this data point belongs to?



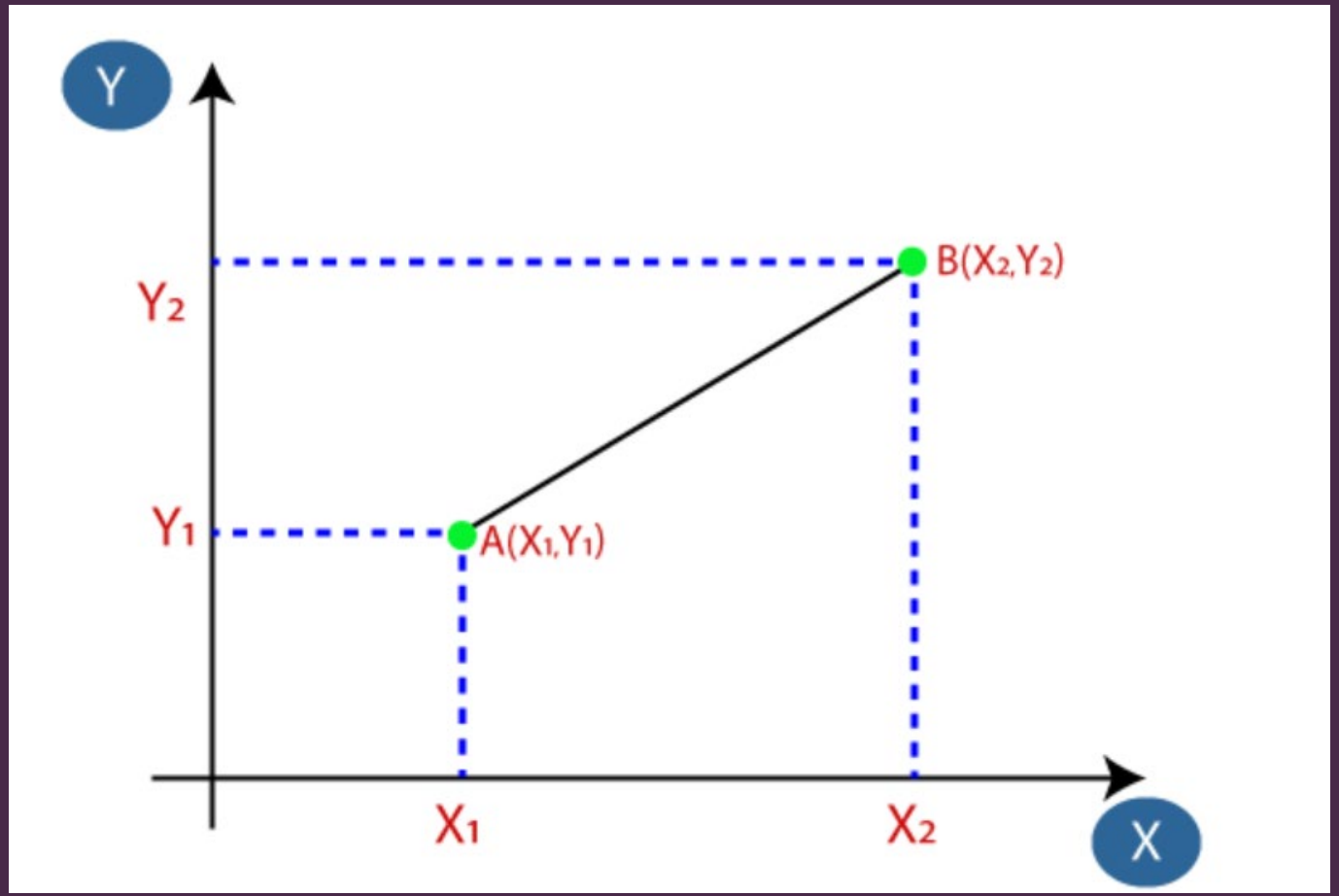
Steps

- **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.

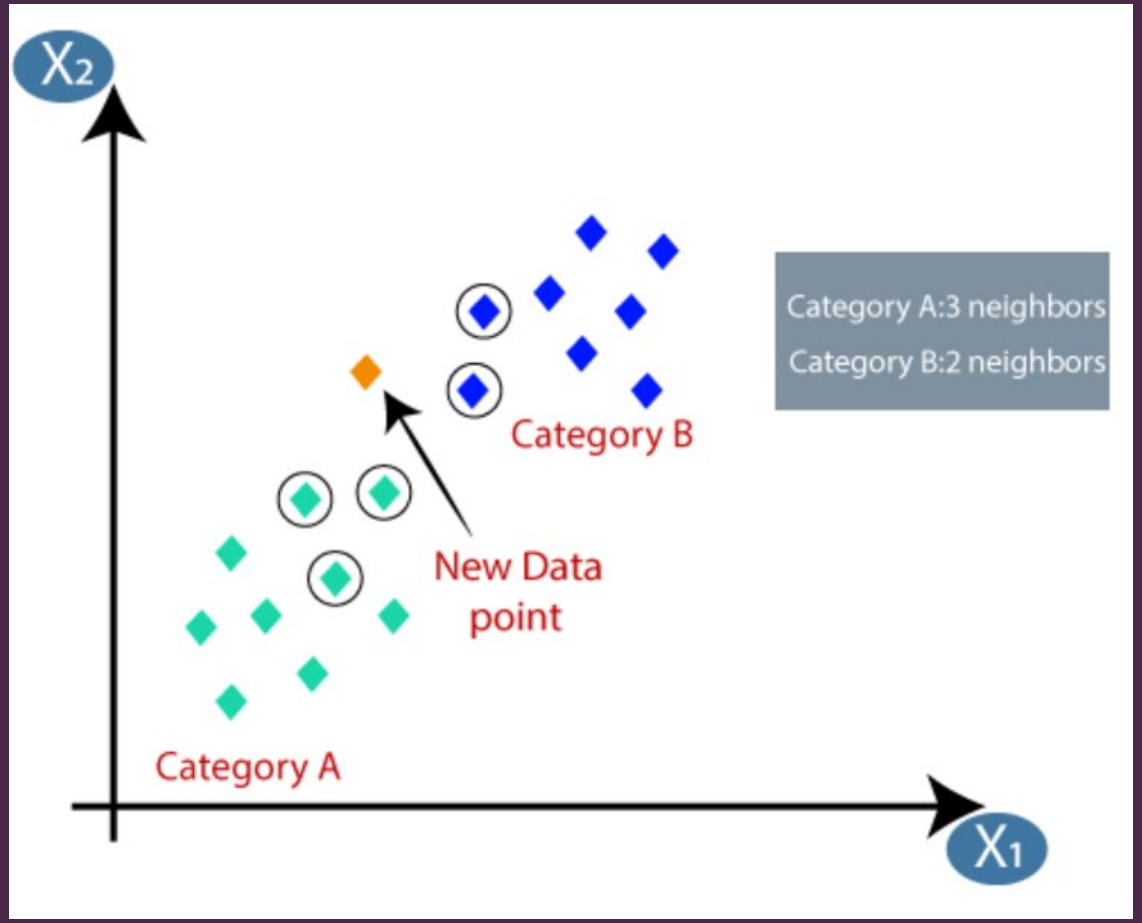
Steps



Choose number of neighbor k=5



Calculate nearest neighbors



How to determine K

- There is no particular way to determine the best value for "K", so we need to try some values to find the best out of them. The most preferred value for K is 5.
- A very low value for K such as K=1 or K=2, can be noisy and lead to the effects of outliers in the model.
- Large values for K are good, but it may find some difficulties.

Advantages

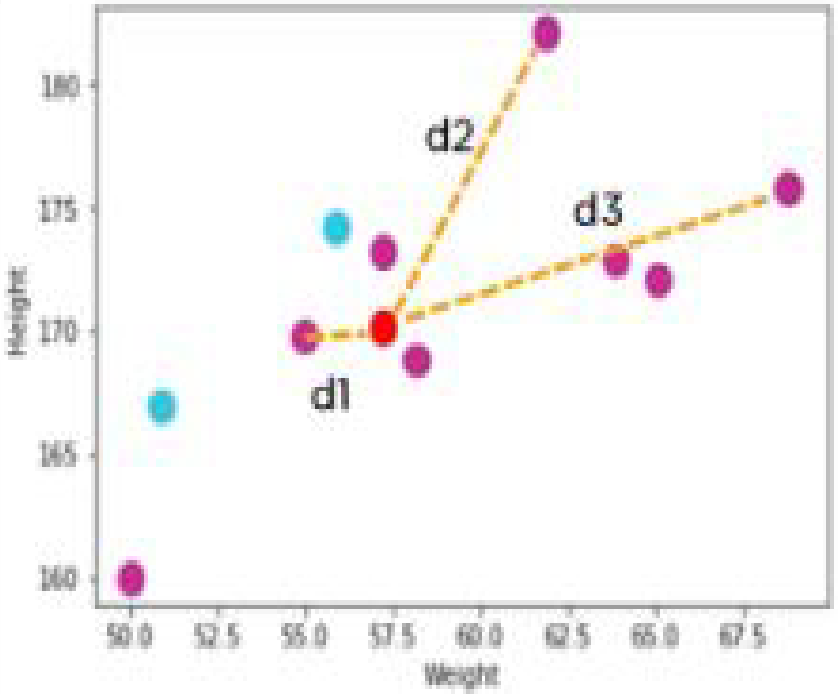
- It is simple to implement.
- It is robust to the noisy training data
- It can be more effective if the training data is large.

Disadvantages

- Always needs to determine the value of K which may be complex some time.
- The computation cost is high because of calculating the distance between the data points for all the training samples.

Weight(x2)	Height(y2)	Class
51	167	Underweight
62	182	Normal
69	176	Normal
64	173	Normal
65	172	Normal
56	174	Underweight
58	169	Normal
57	173	Normal
55	170	Normal

57 kg	170 cm	?



Unknown data point

dist(d1)= √(170-167)² + (57-51)² -= 6.7

dist(d2)= √(170-182)² + (57-62)² -= 13

dist(d3)= √(170-176)² + (57-69)² -= 13.4

Similarly, we will calculate Euclidean distance of unknown data point from all the points in the dataset

Distance between unknown to other points

Weight(x2)	Height(y2)	Class	Euclidean Distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2

K=3

Weight(x2)	Height(y2)	Class	Euclidean Distance
51	167	Underweight	6.7
62	182	Normal	13
69	176	Normal	13.4
64	173	Normal	7.6
65	172	Normal	8.2
56	174	Underweight	4.1
58	169	Normal	1.4
57	173	Normal	3
55	170	Normal	2