機器學習
K-Means

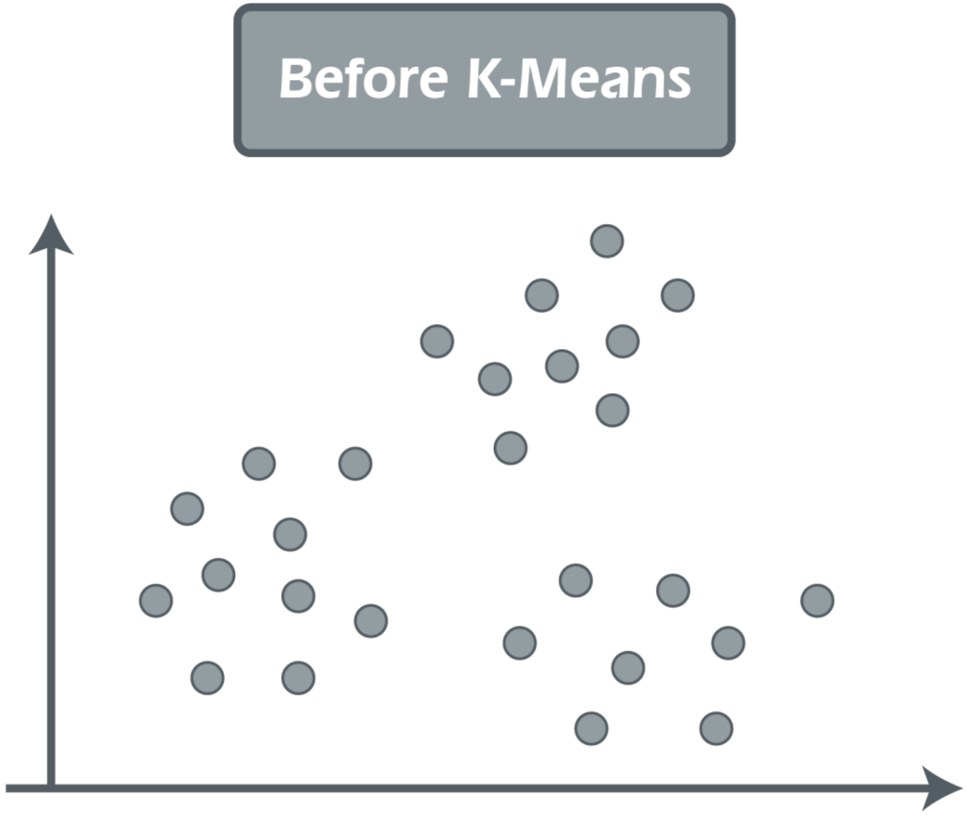
授課老師：林彥廷

K-Means

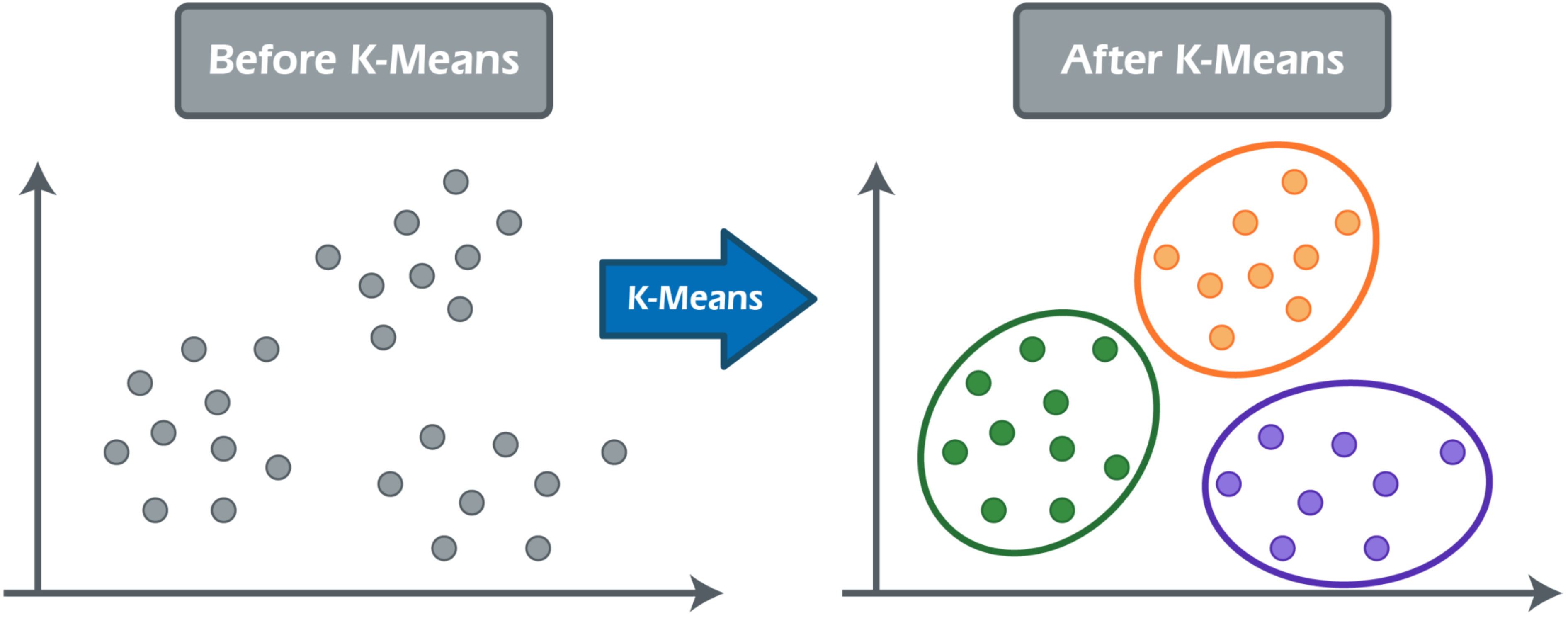
- K-Means是常見的分群(Clustering)演算法之一
- K-Means屬於非監督式學習演算法
- 非監督式學習是資料並沒有標籤，讓機器直接從資料中學習出規則

K-Means Intuition: Understanding K-Means

What K-Means does for you



What K-Means does for you



How did it do that?

STEP 1: Choose the number K of clusters

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)

STEP 3: Assign each data point to the closest centroid → That forms K clusters

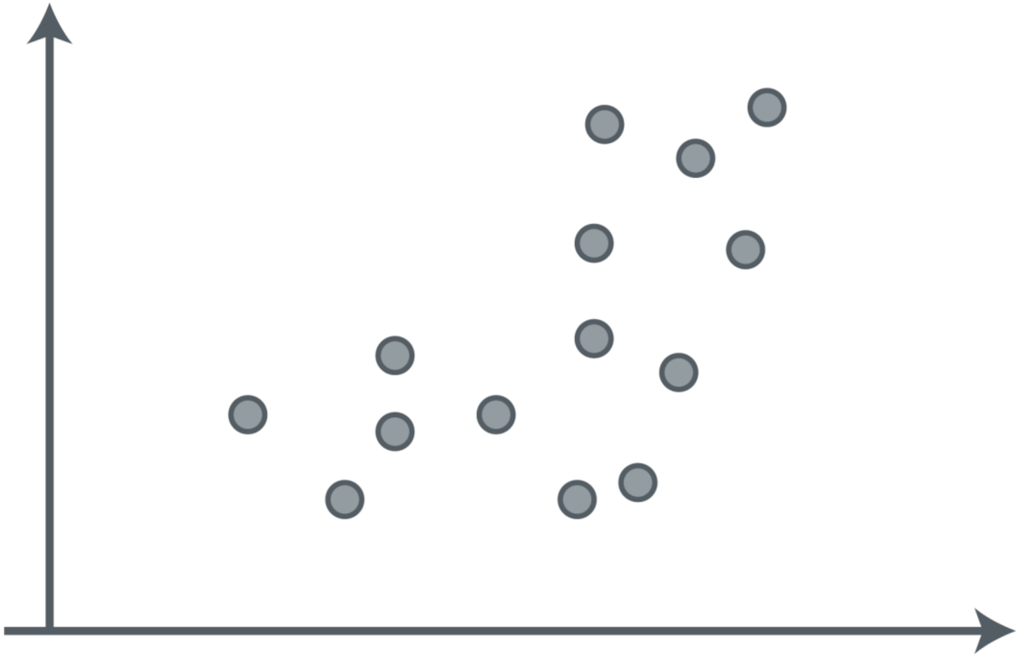
STEP 4: Compute and place the new centroid of each cluster

STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.

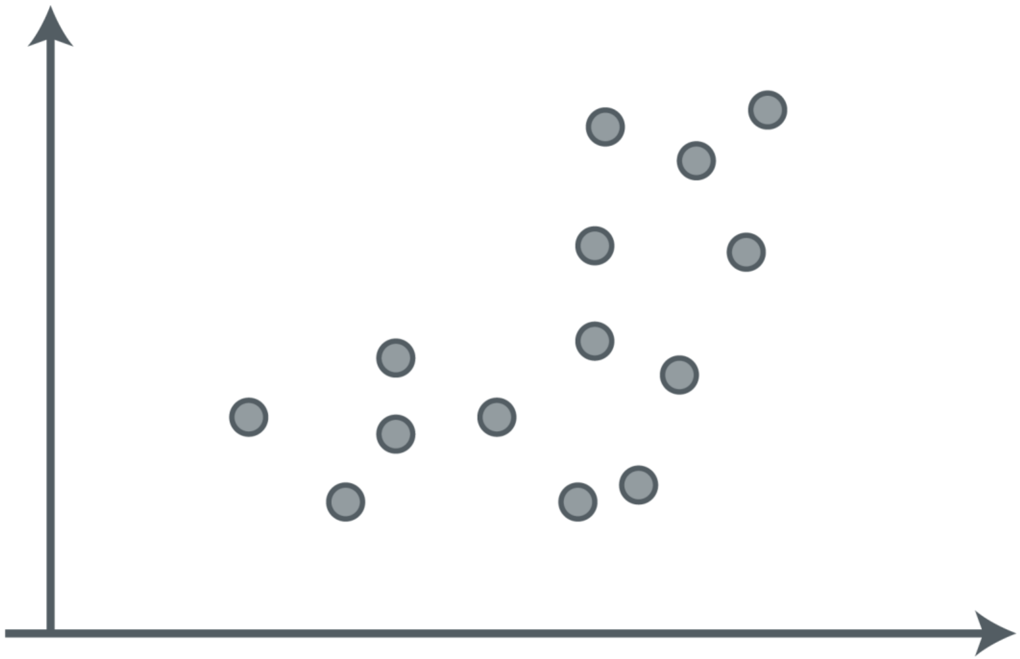
K-Means algorithm

STEP 1: Choose the number K of clusters: K=2



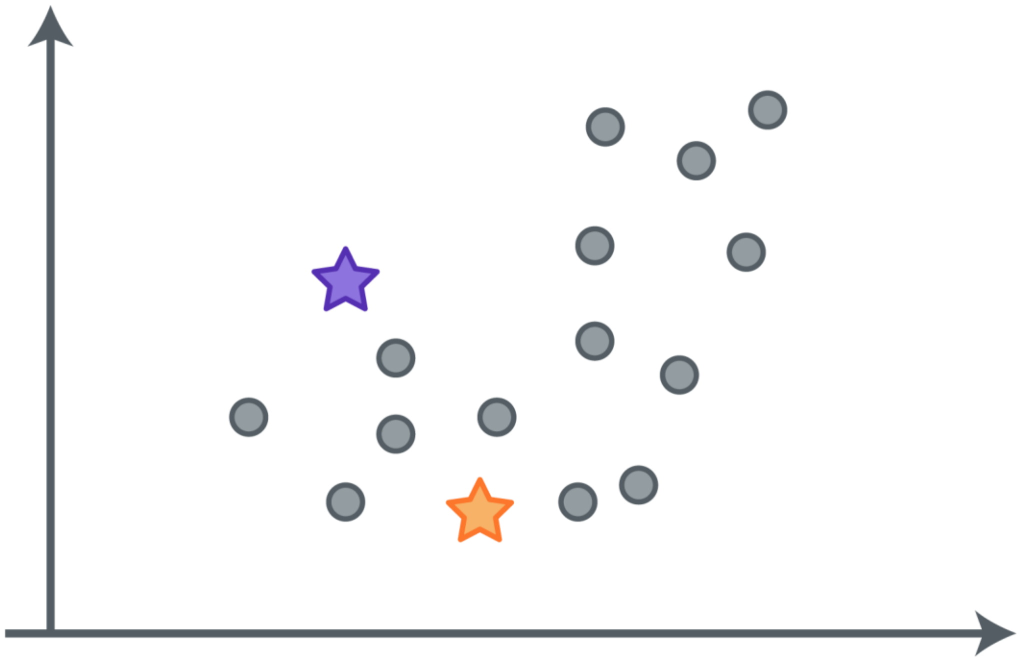
K-Means algorithm

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)



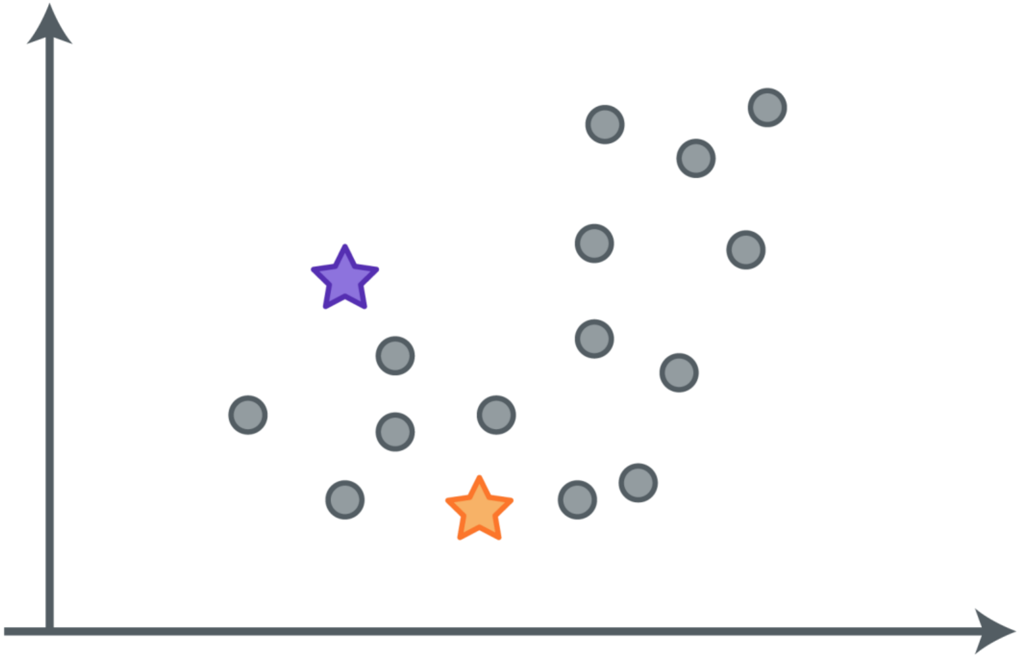
K-Means algorithm

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)



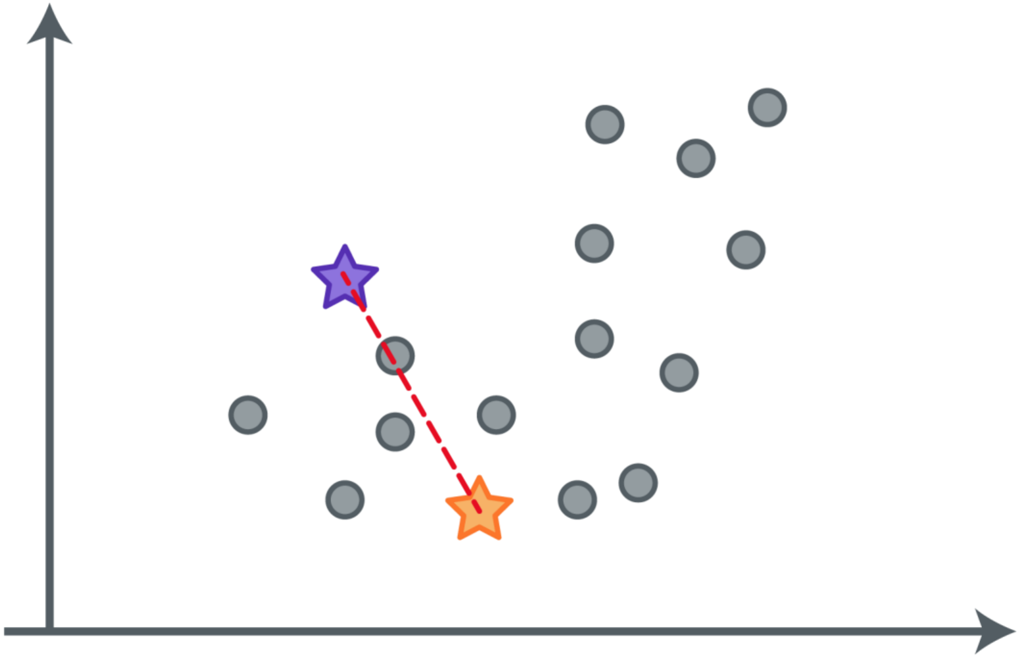
K-Means algorithm

STEP 3: Assign each data point to the closest centroid → That forms K clusters



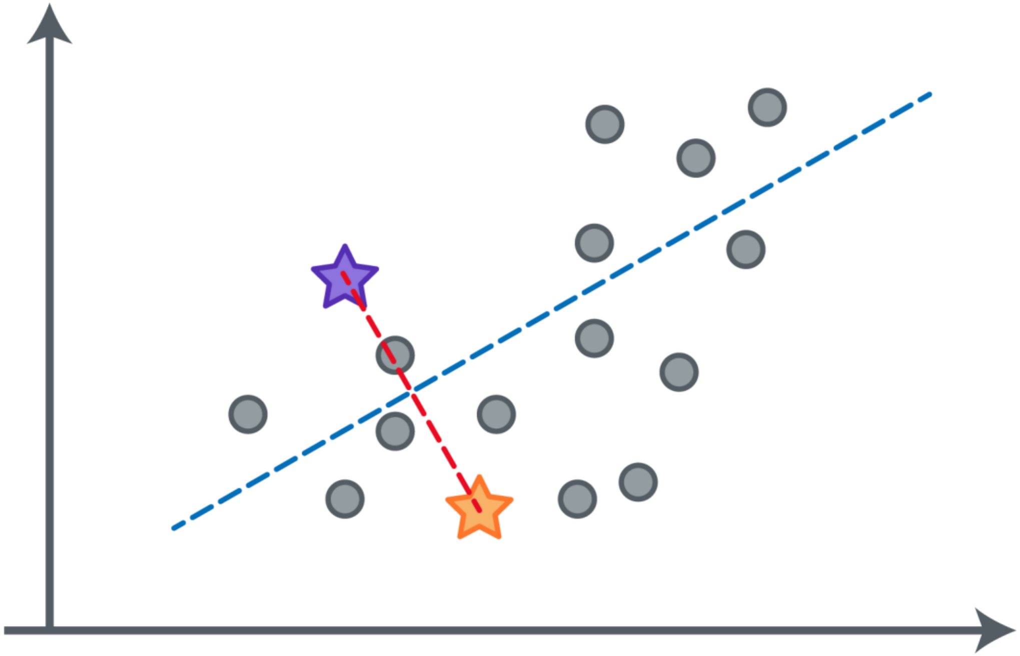
K-Means algorithm

STEP 3: Assign each data point to the closest centroid → That forms K clusters



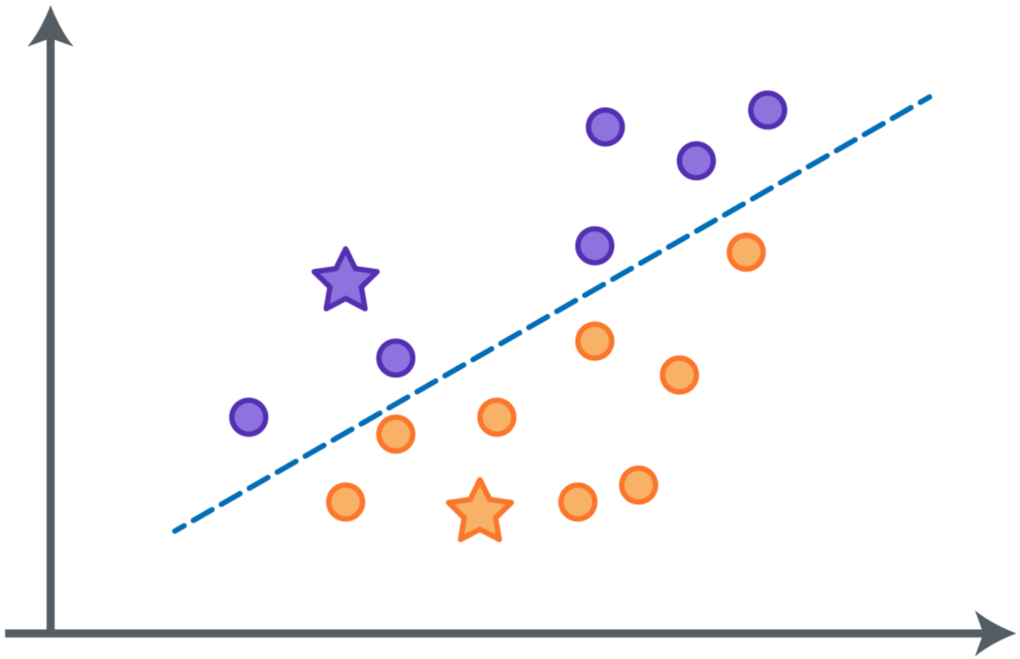
K-Means algorithm

STEP 3: Assign each data point to the closest centroid → That forms K clusters



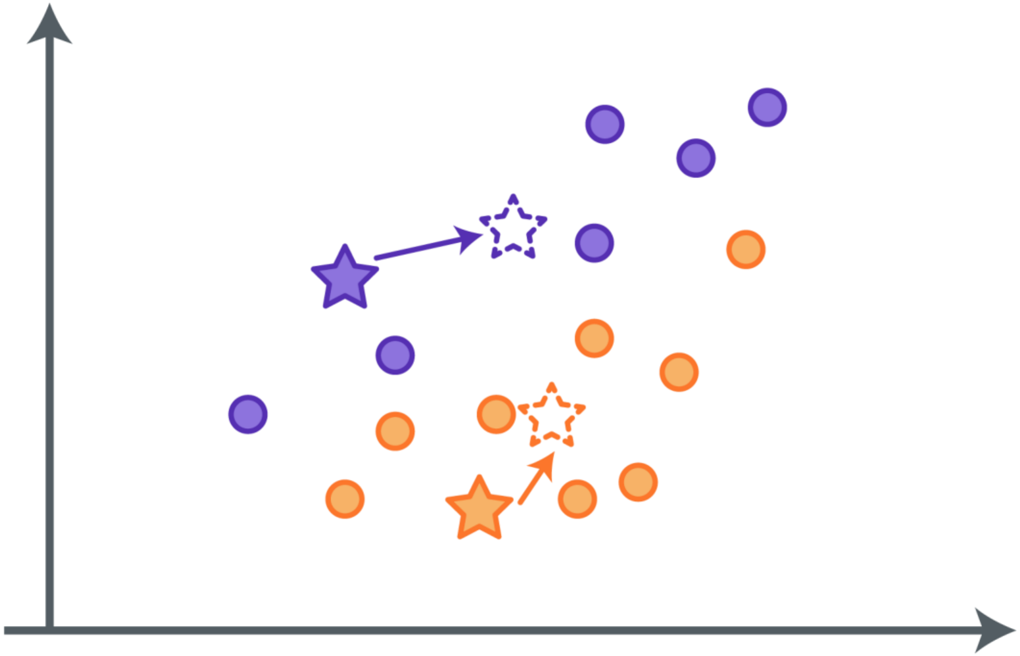
K-Means algorithm

STEP 3: Assign each data point to the closest centroid → That forms K clusters



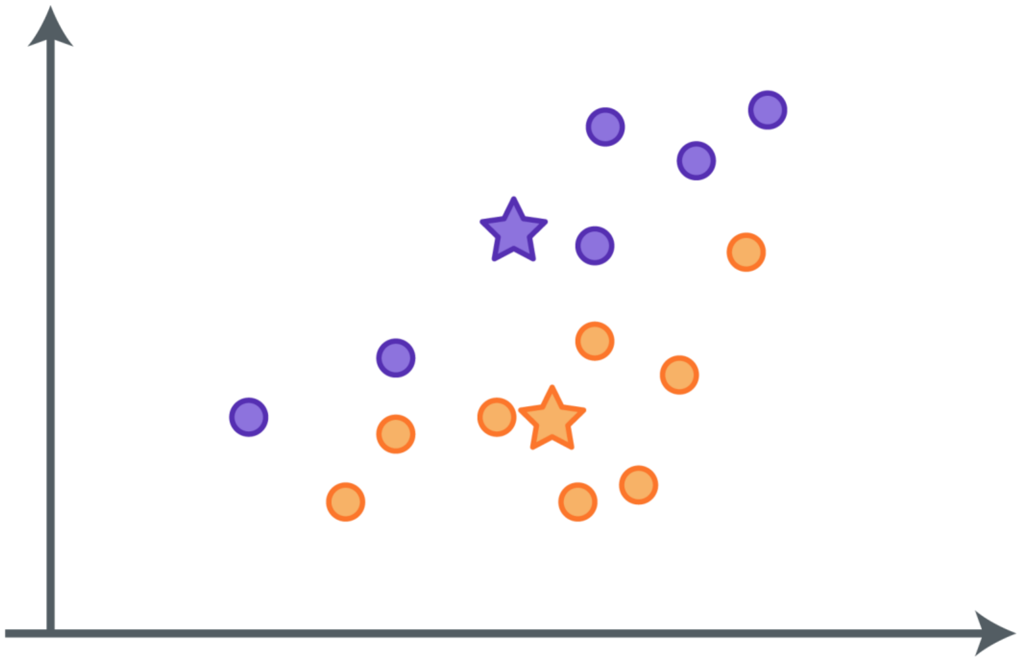
K-Means algorithm

STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

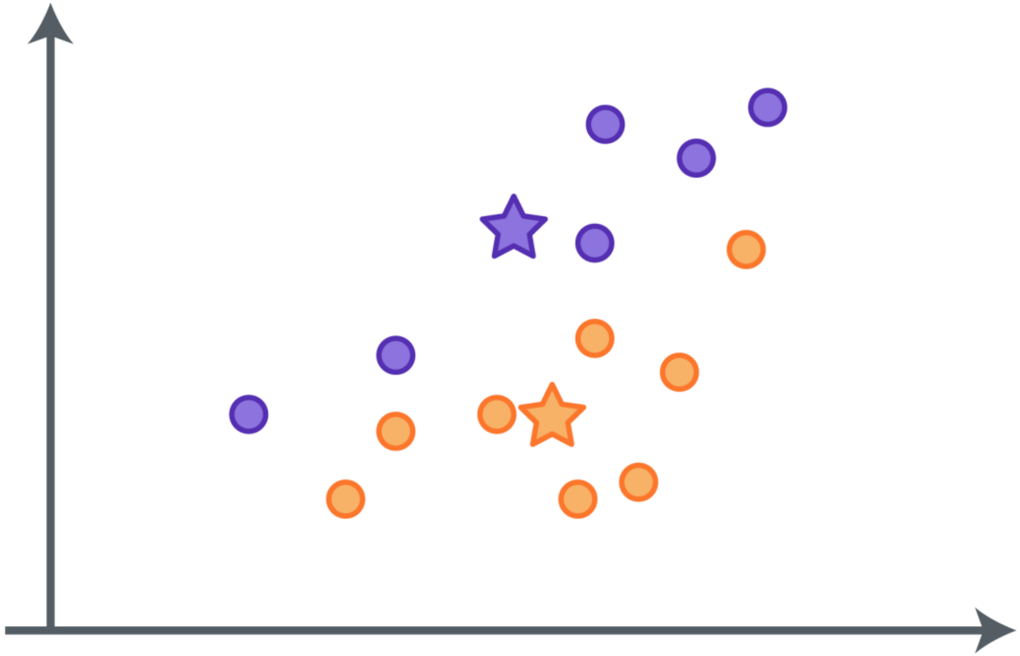
STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

STEP 5: Reassign each data point to the new closest centroid.

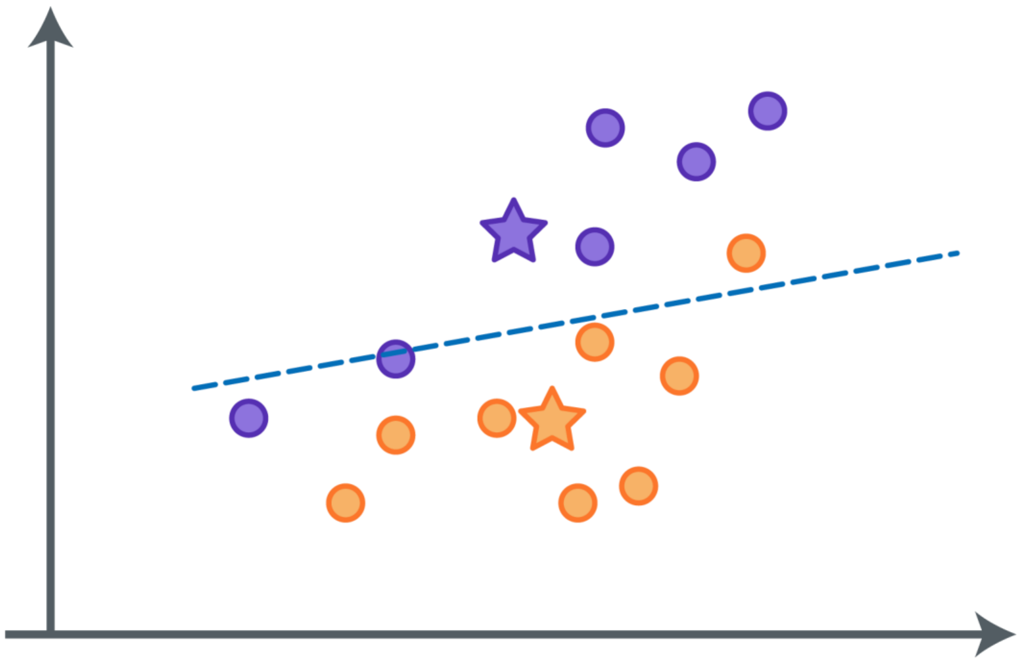
If any reassignment took place, go to STEP 4, otherwise go to FIN.



K-Means algorithm

STEP 5: Reassign each data point to the new closest centroid.

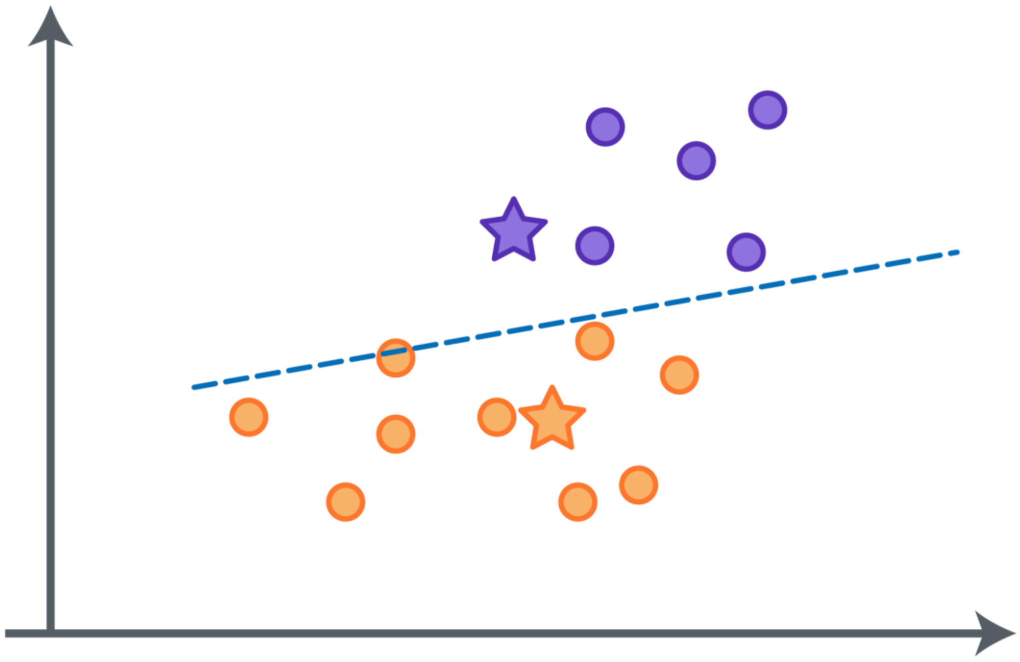
If any reassignment took place, go to STEP 4, otherwise go to FIN.



K-Means algorithm

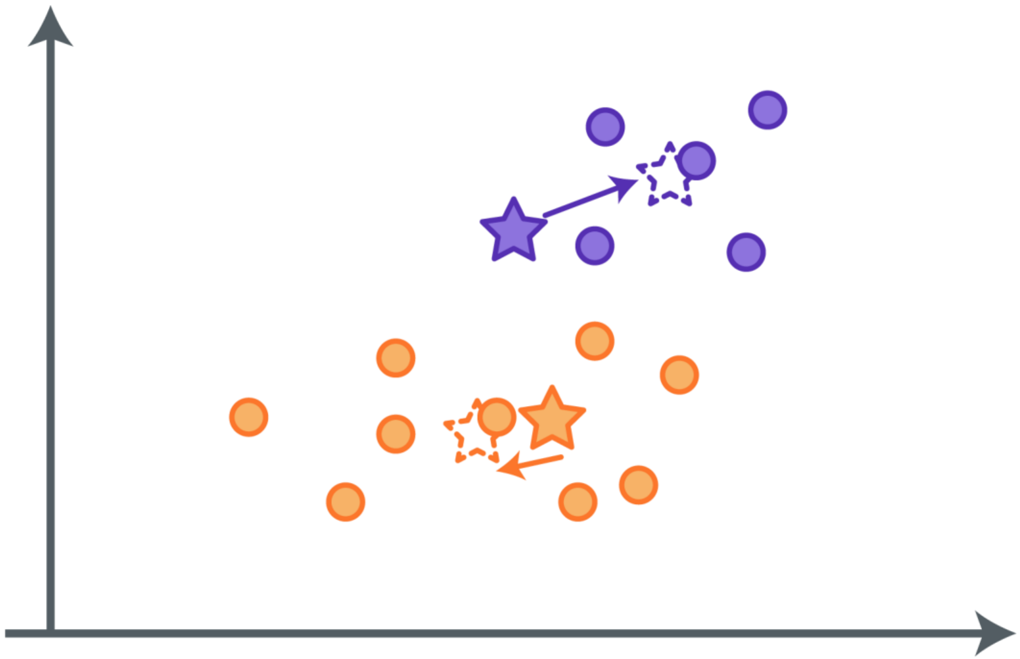
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.



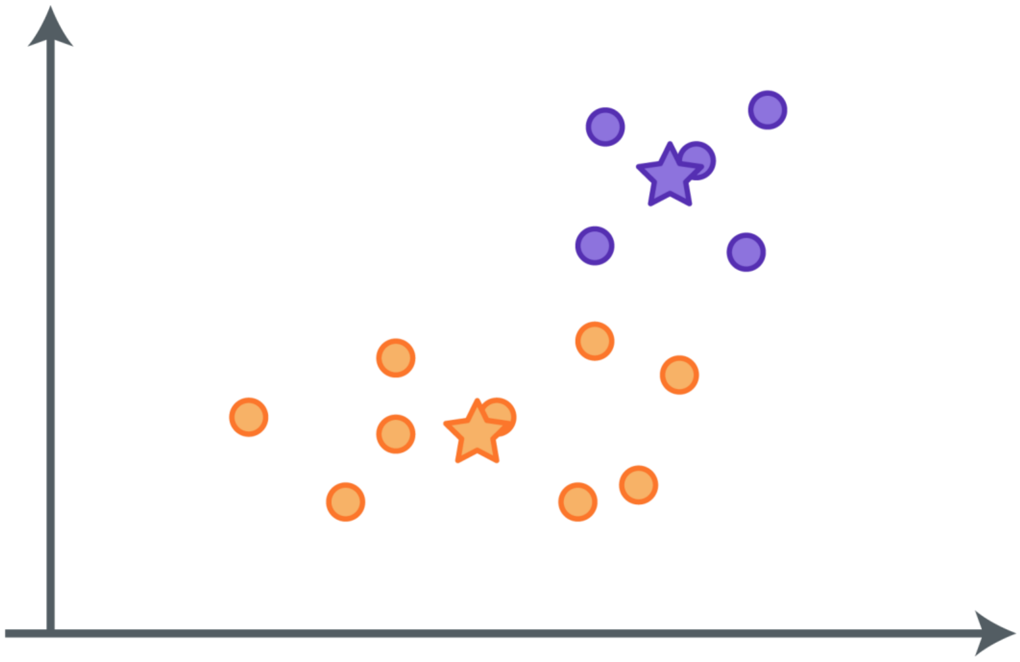
K-Means algorithm

STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

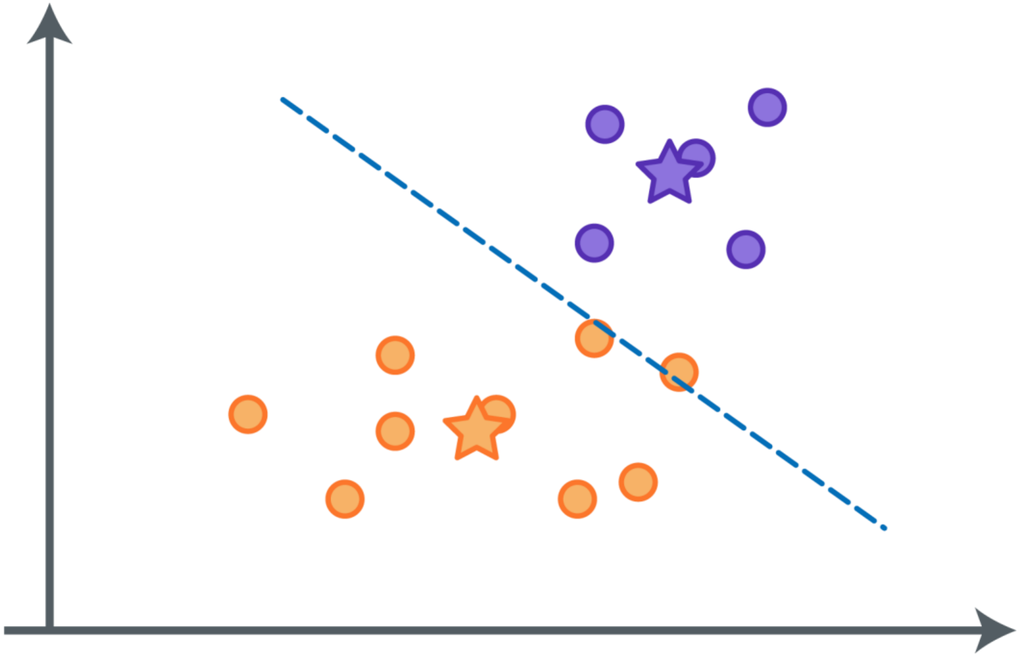
STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

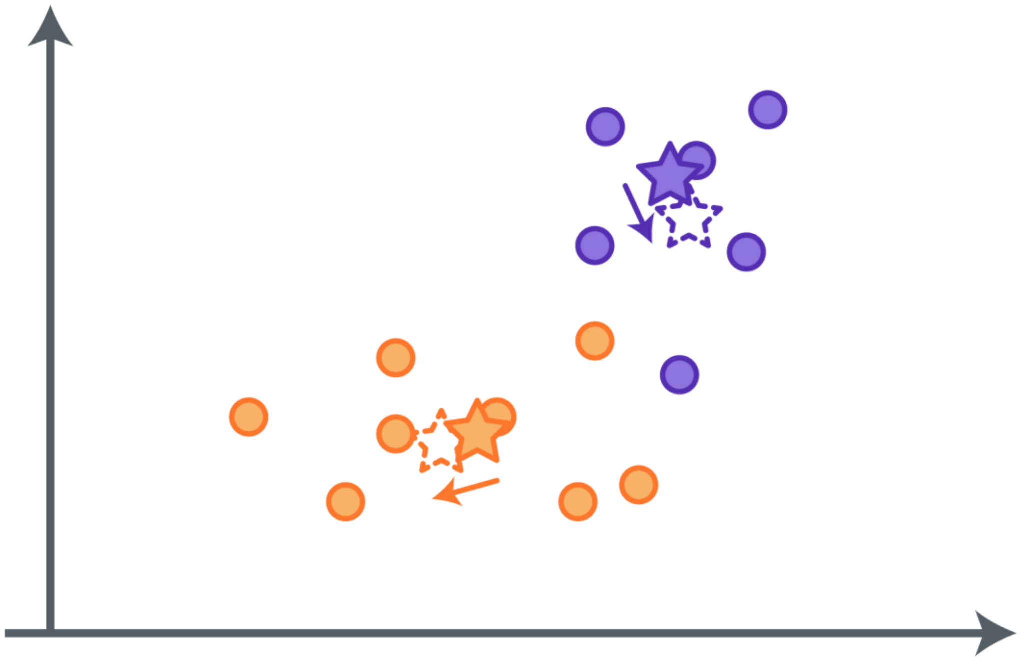
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.



K-Means algorithm

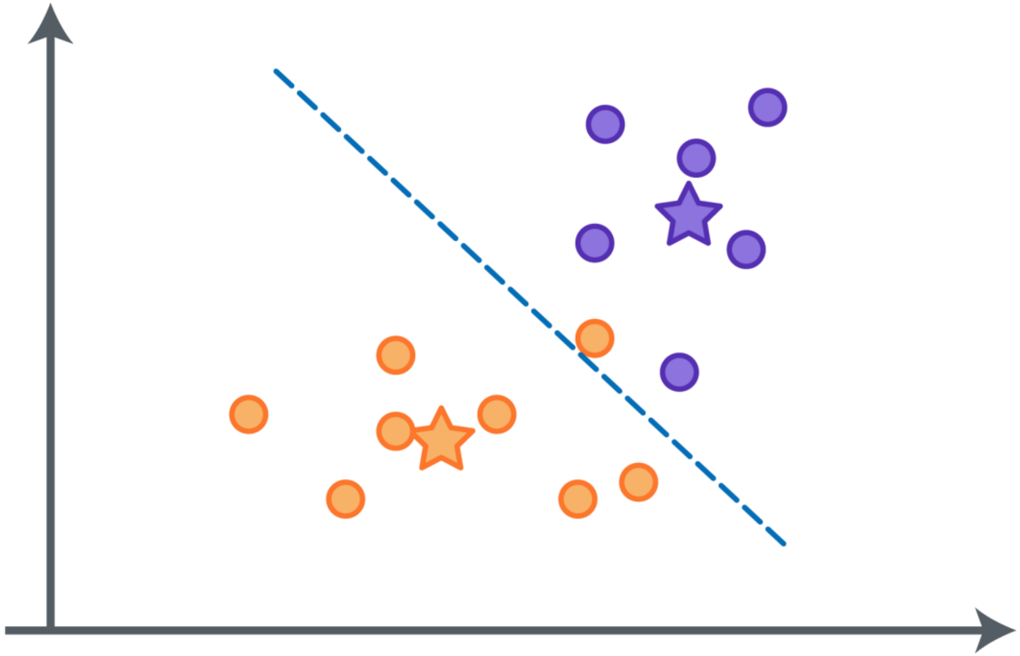
STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

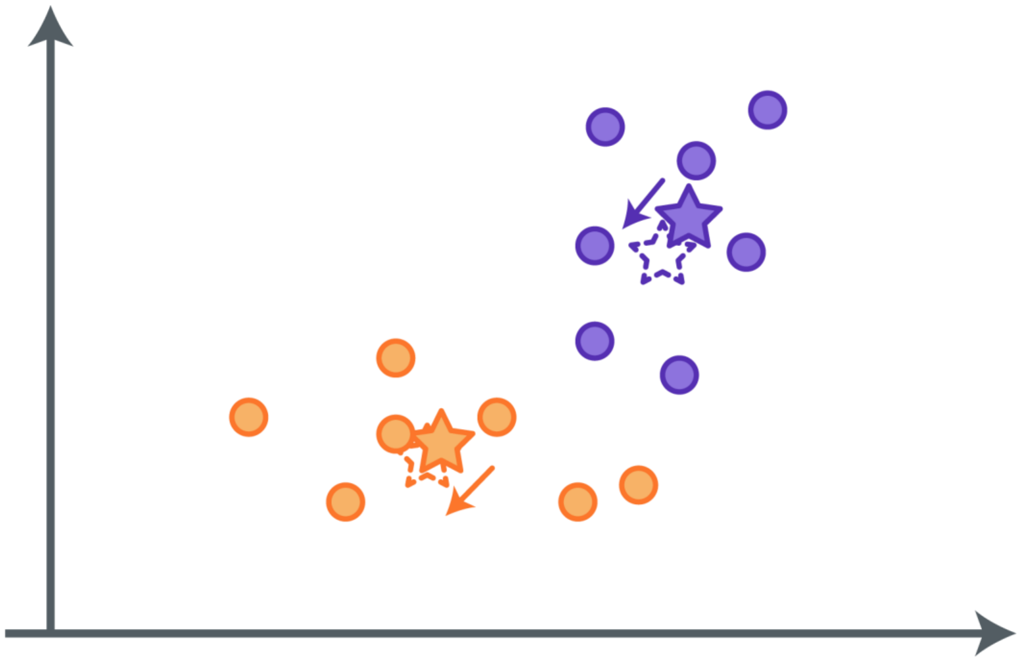
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.



K-Means algorithm

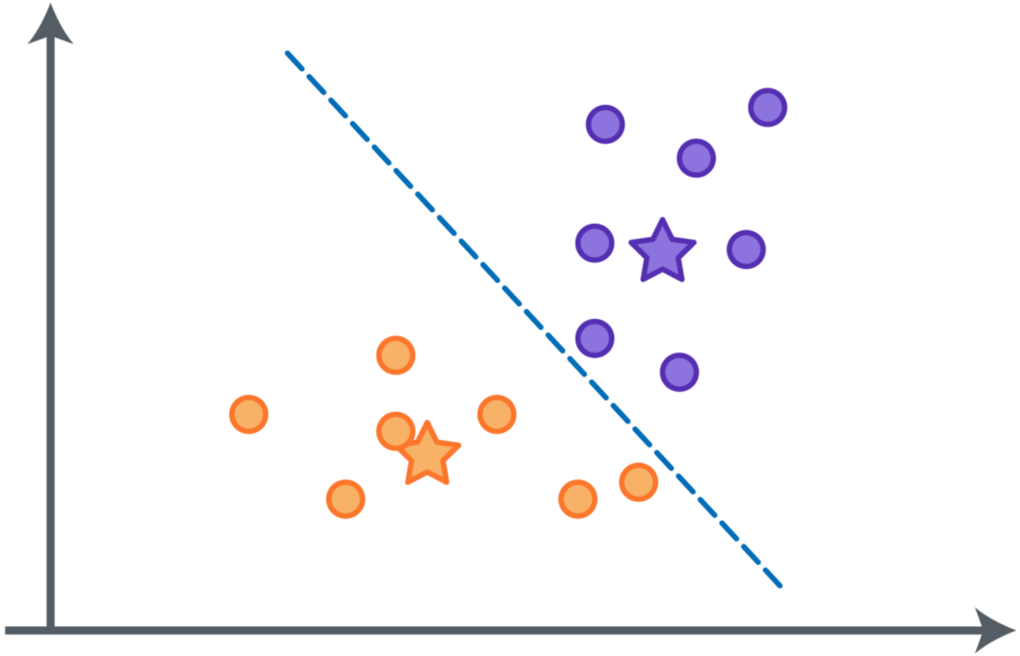
STEP 4: Compute and place the new centroid of each cluster



K-Means algorithm

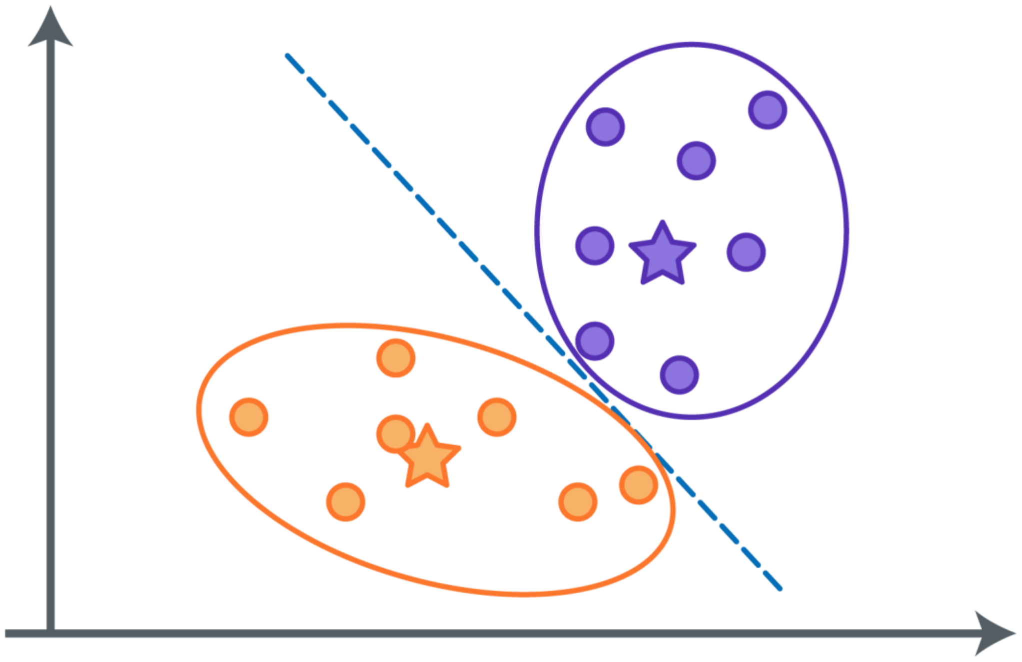
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.



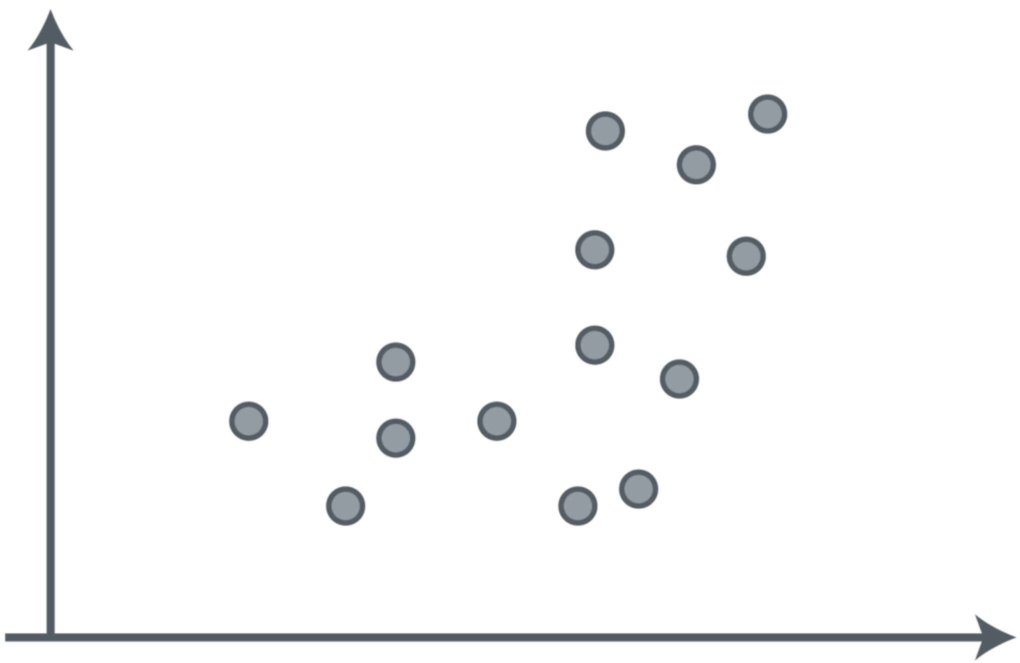
K-Means algorithm

FIN: Your Model Is Ready



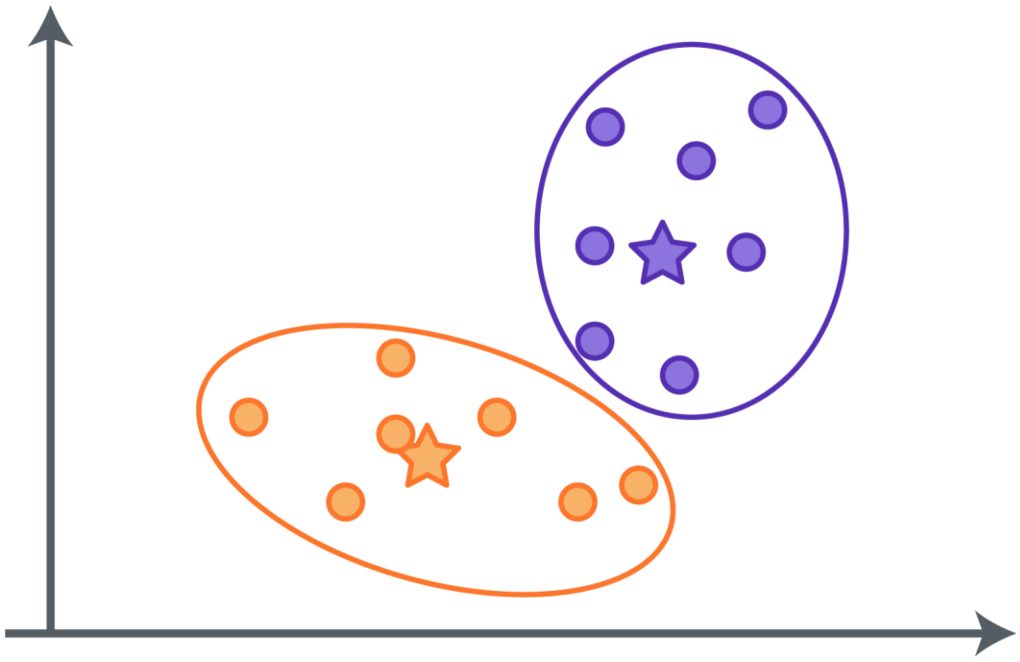
K-Means algorithm

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)



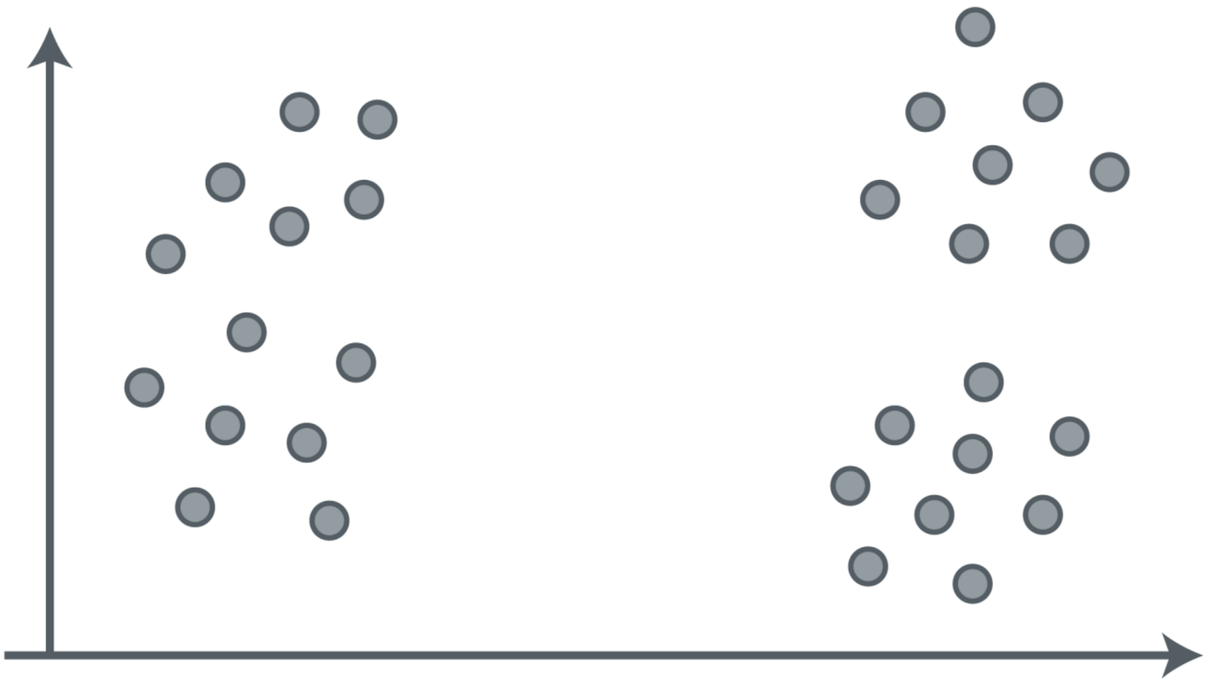
K-Means algorithm

FIN: Your Model Is Ready



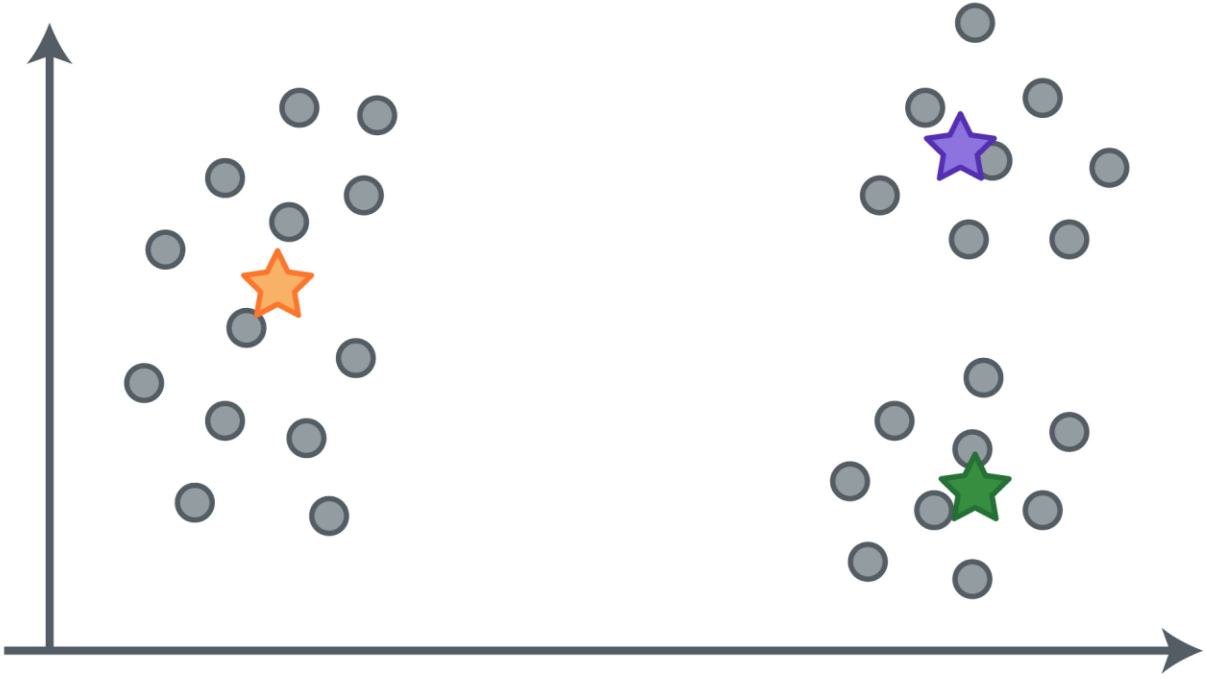
K-Means Intuition:
Random Initialization Trap

Random Initialization Trap



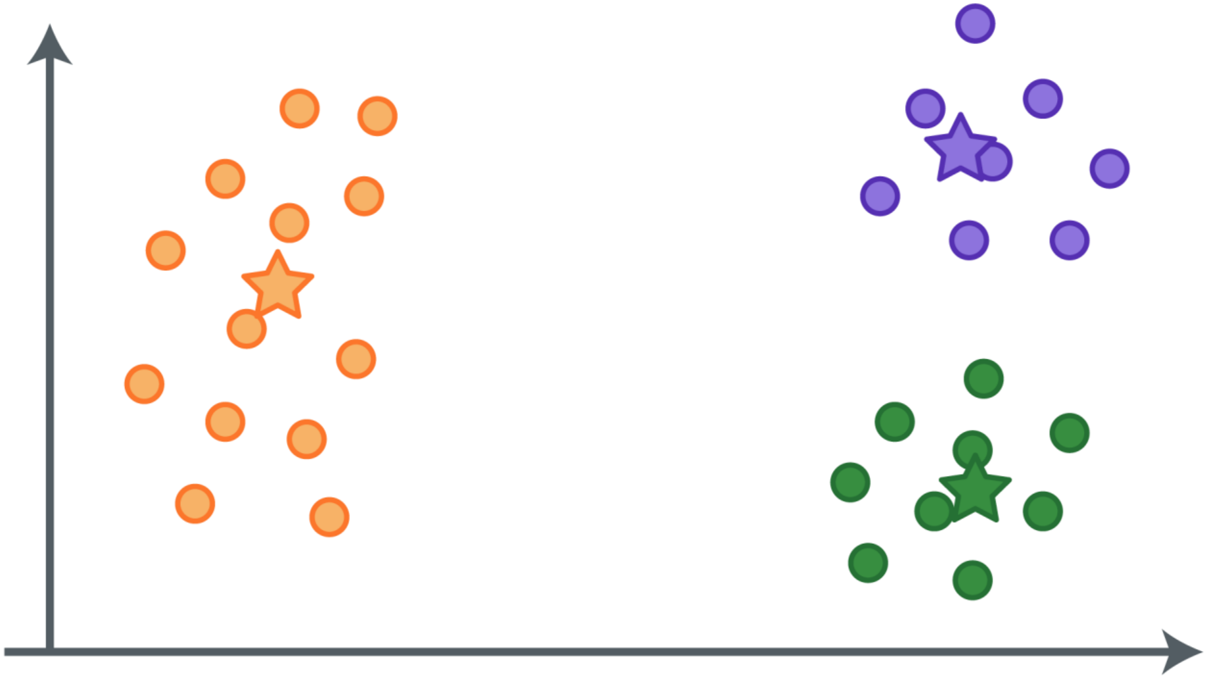
If we choose K = 3 clusters…

Random Initialization Trap



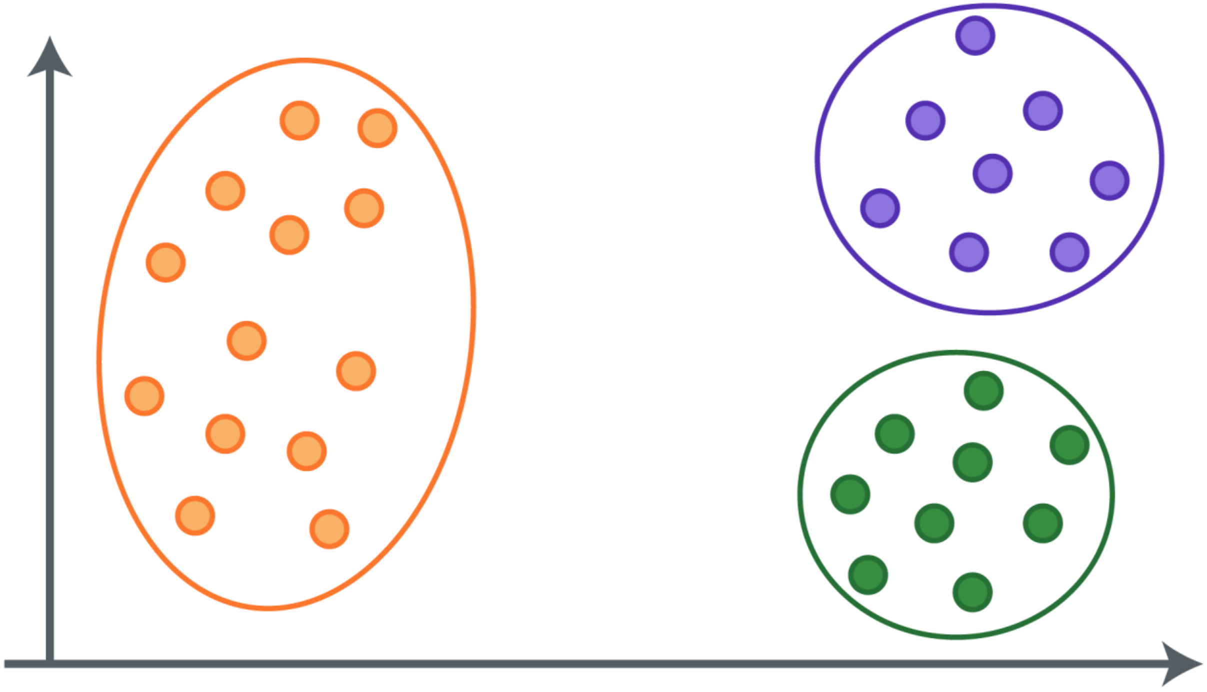
…this correct random initialization would lead us to…

Random Initialization Trap



…the following three clusters

Random Initialization Trap



…the following three clusters

Random Initialization Trap

But what would happen if we had a bad random initialization?

Random Initialization Trap

STEP 1: Choose the number K of clusters

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)

STEP 3: Assign each data point to the closest centroid → That forms K clusters

STEP 4: Compute and place the new centroid of each cluster

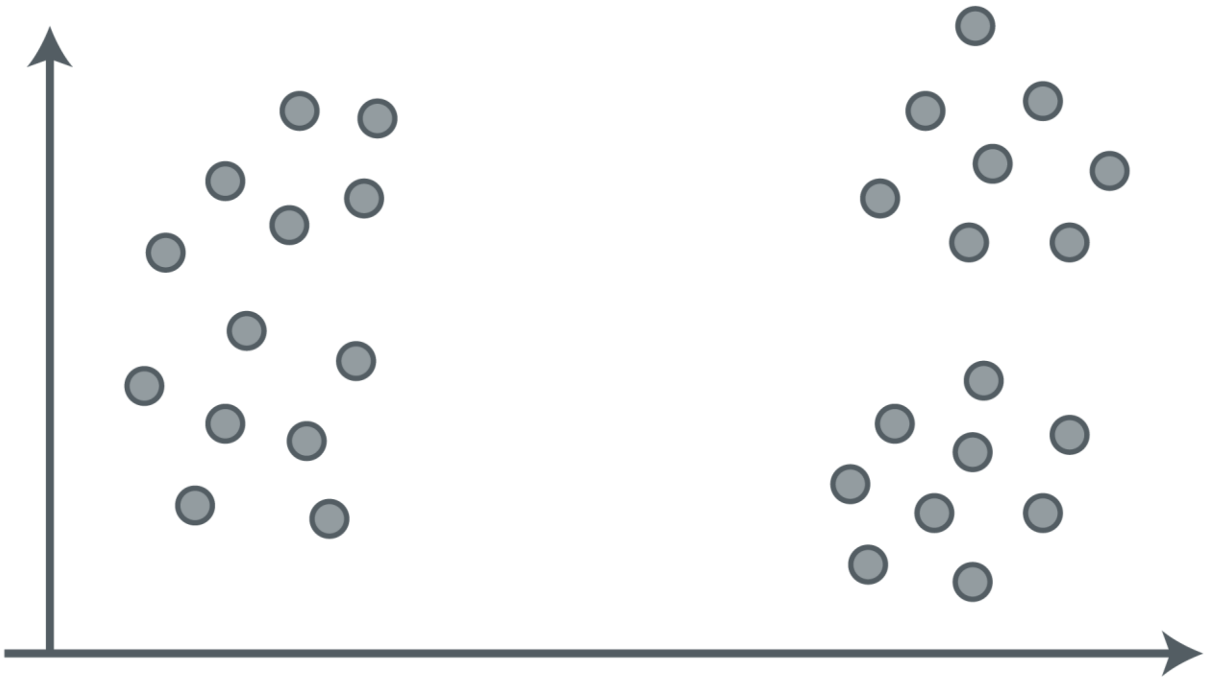
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.

Your Model is Ready

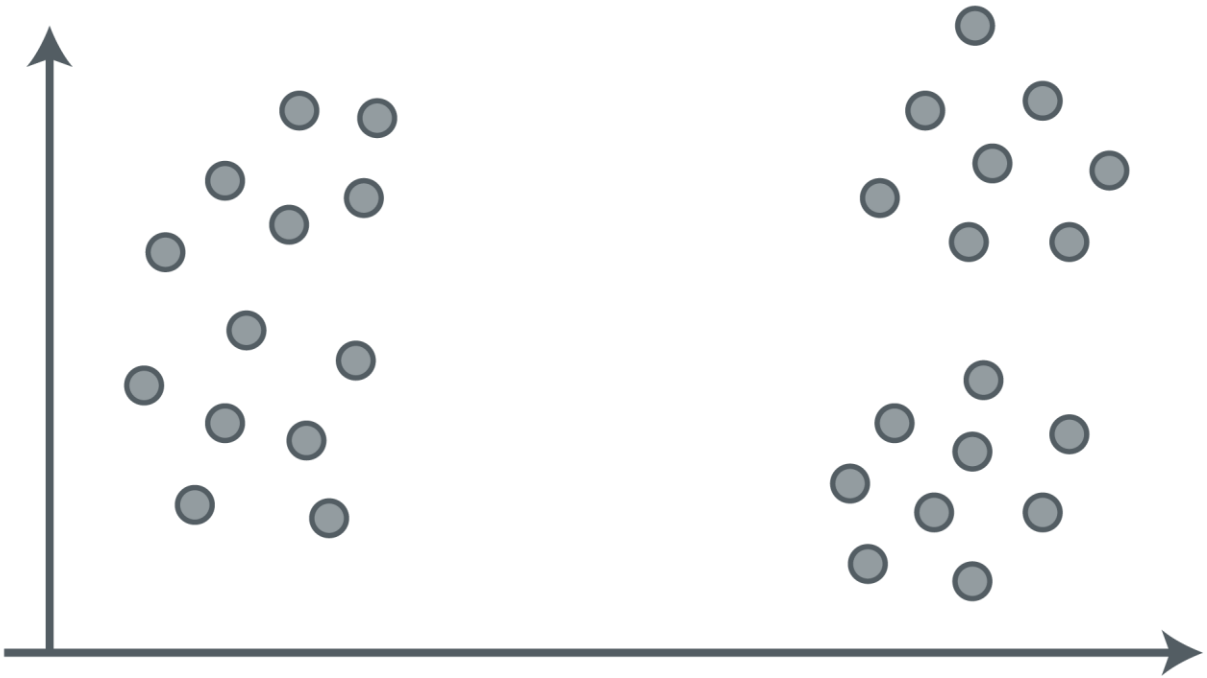
Random Initialization Trap

STEP 1: Choose the number K of clusters: K = 3



Random Initialization Trap

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)

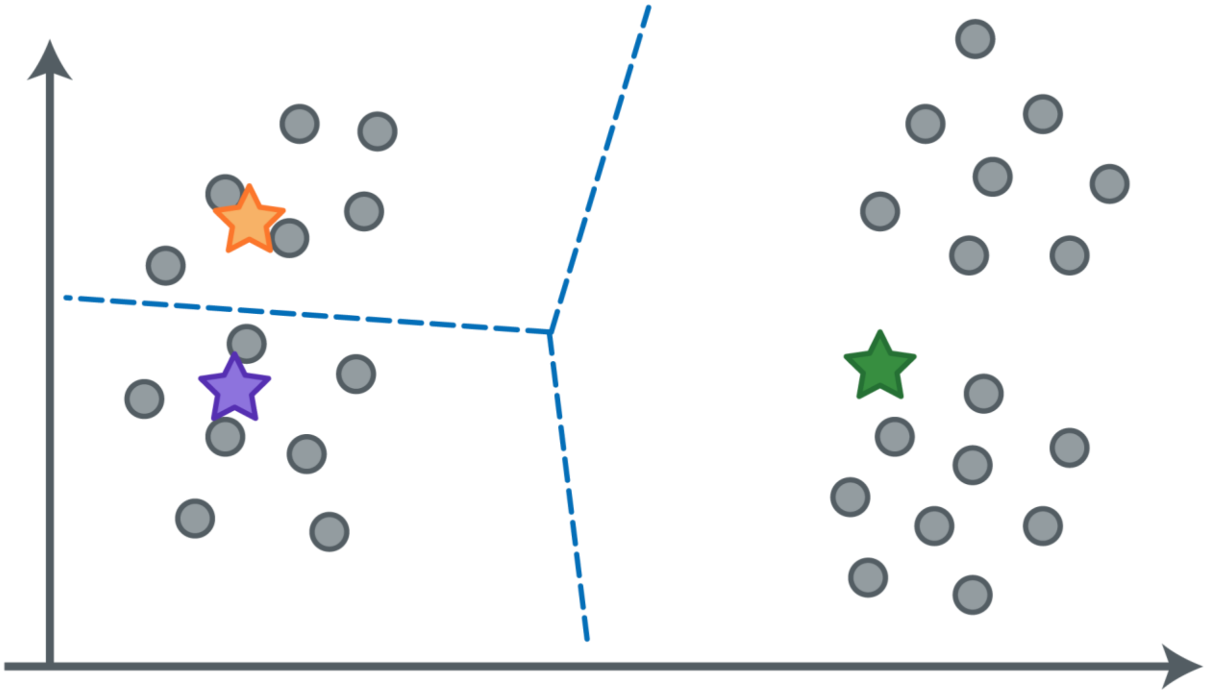


Random Initialization Trap

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)

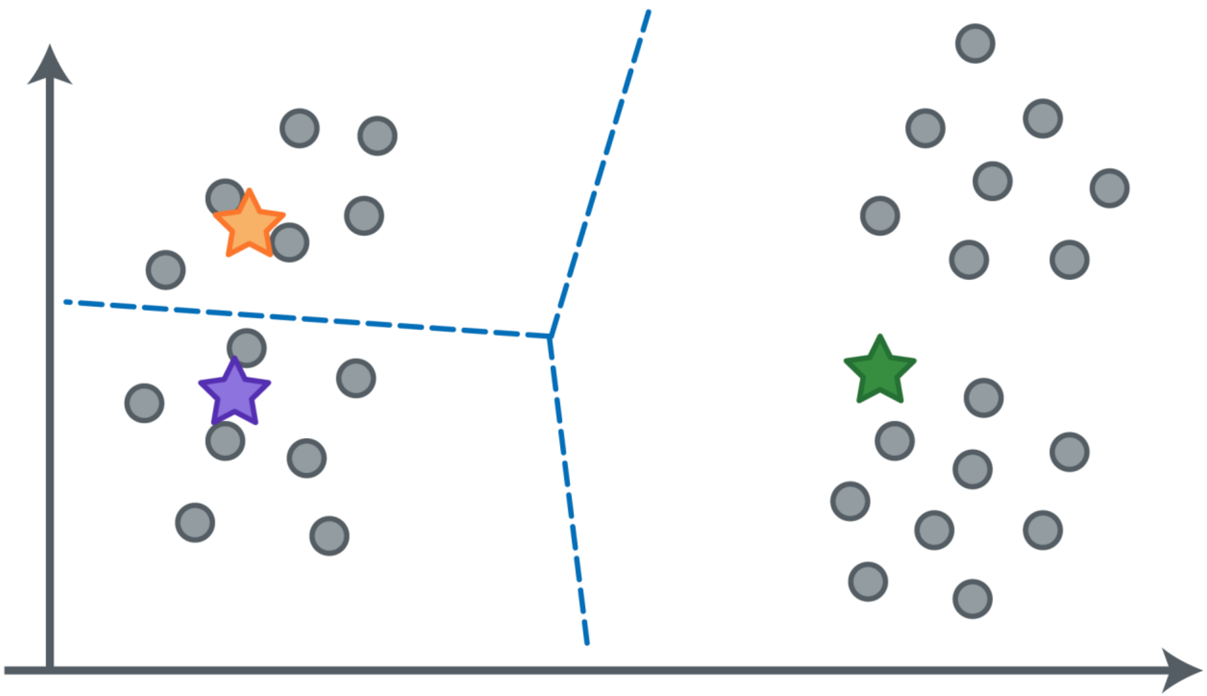
Random Initialization Trap

STEP 2: Select at random K points, the centroids (not necessarily from your dataset)



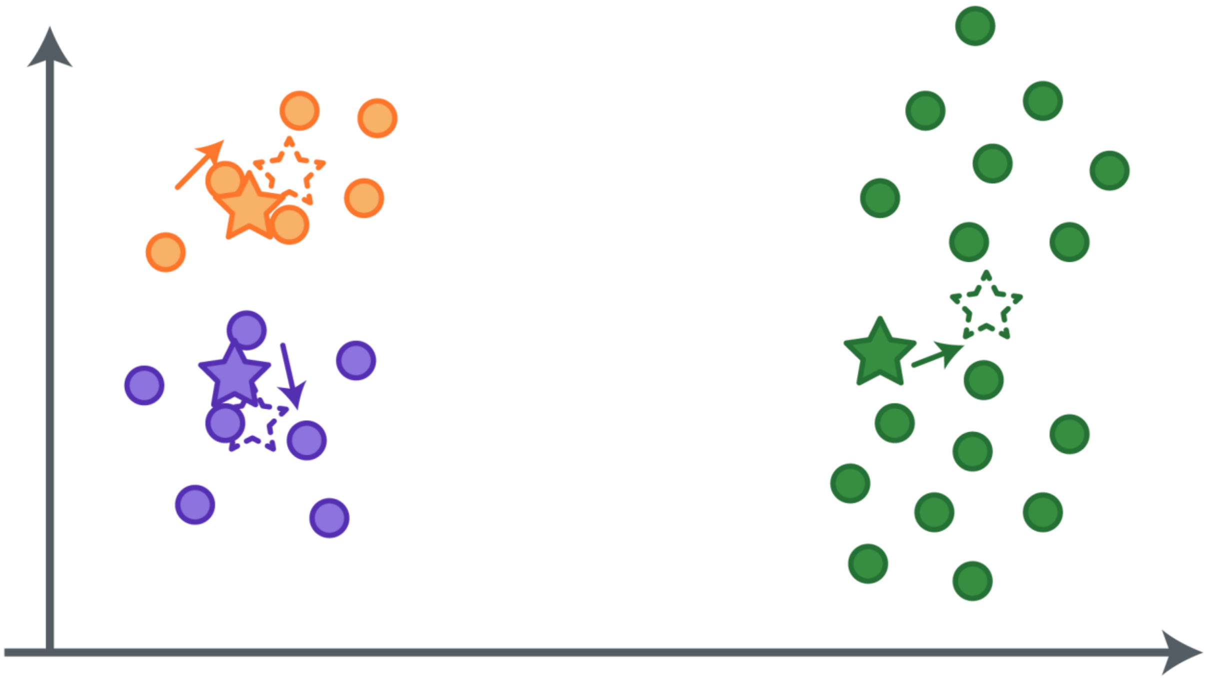
Random Initialization Trap

STEP 3: Assign each data point to the closest centroid → That forms K clusters



Random Initialization Trap

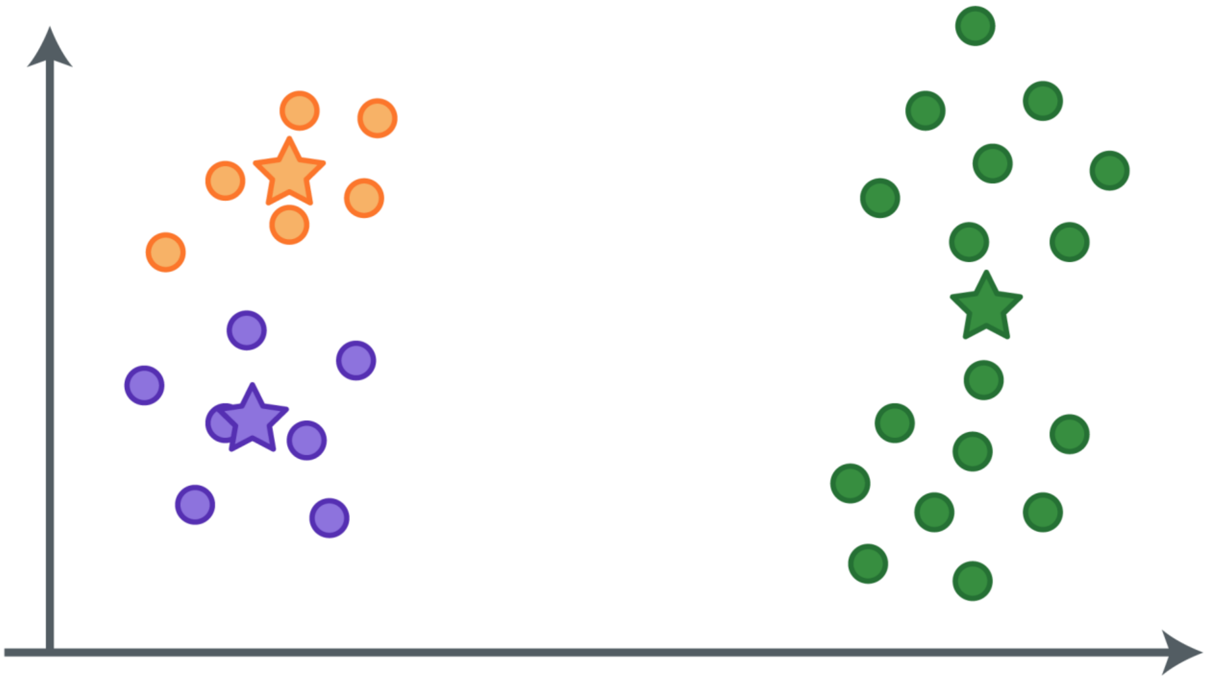
STEP 3: Assign each data point to the closest centroid → That forms K clusters



Random Initialization Trap

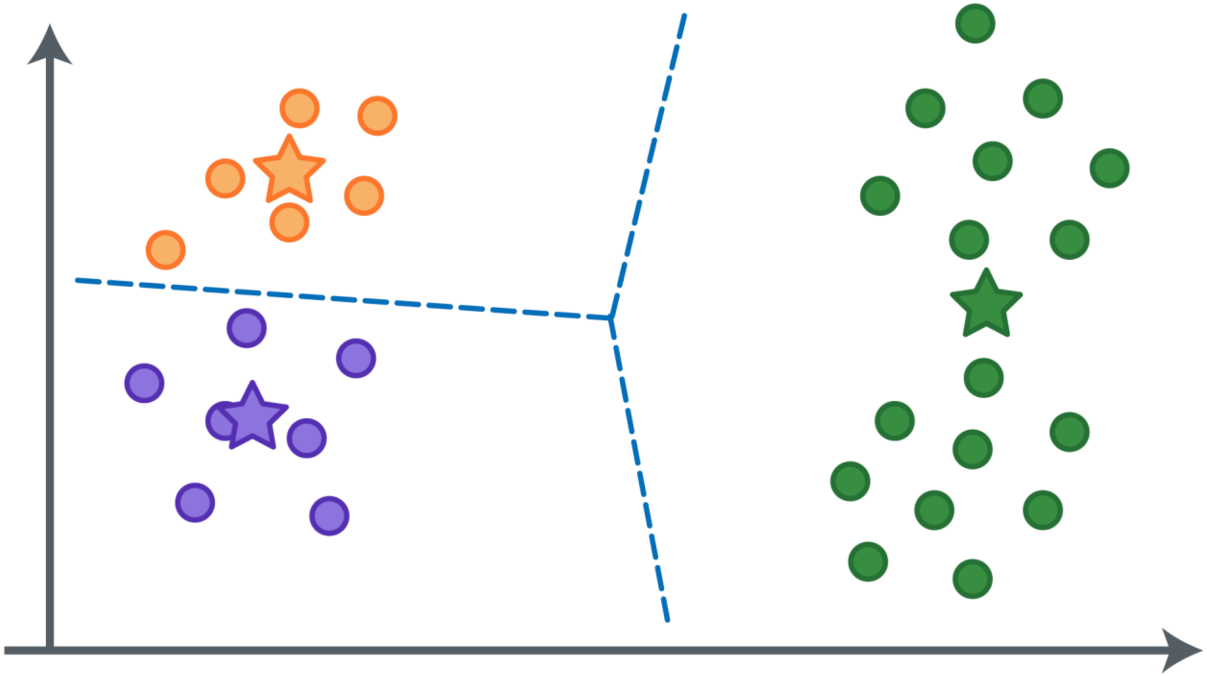
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.



Random Initialization Trap

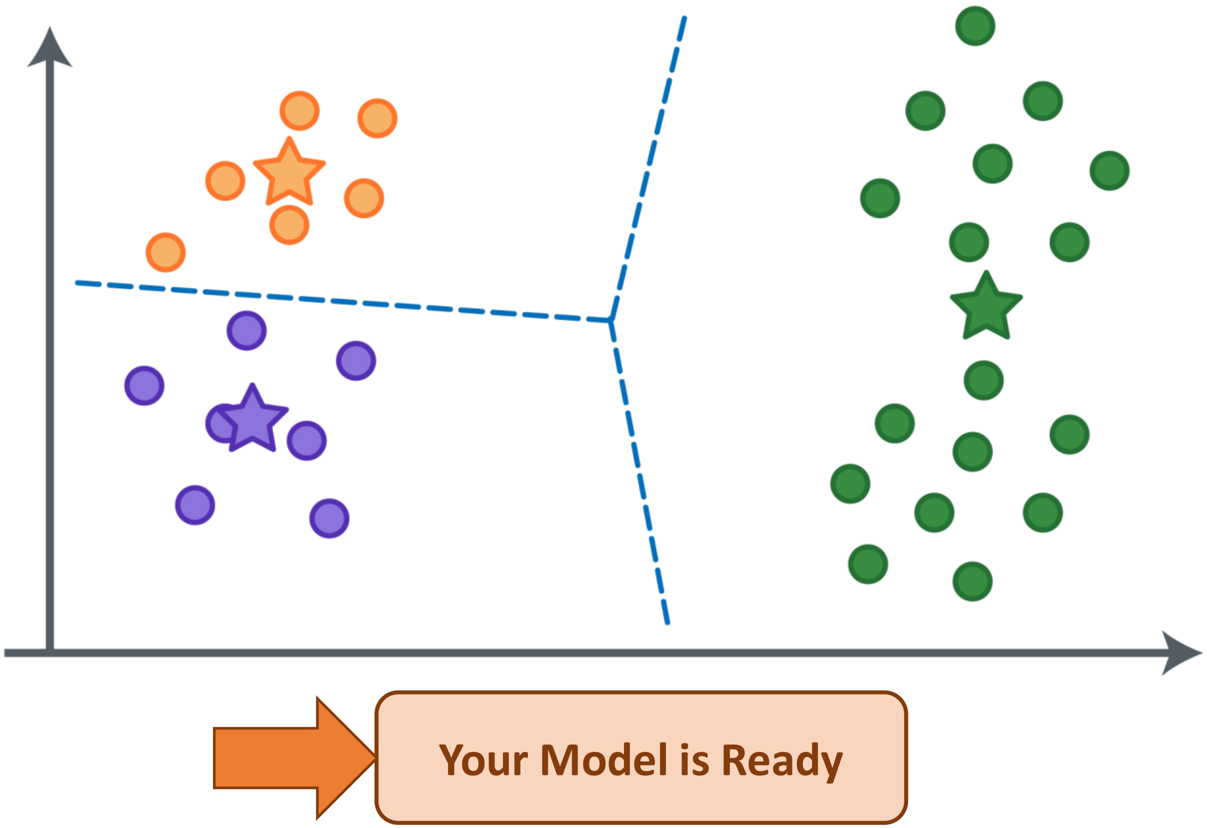
STEP 5: Reassign each data point to the new closest centroid.
If any reassignment took place, go to STEP 4, otherwise go to FIN.



Random Initialization Trap

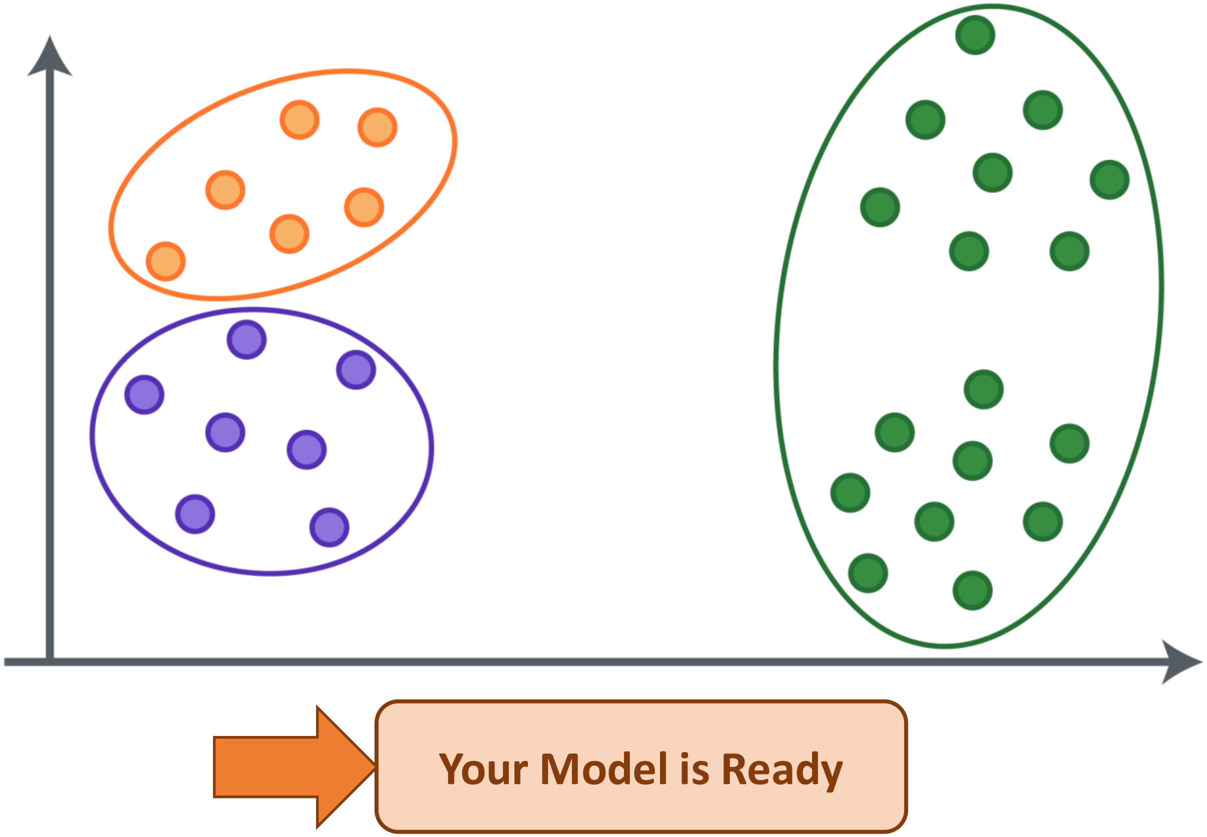
STEP 5: Reassign each data point to the new closest centroid.

If any reassignment took place, go to STEP 4, otherwise go to FIN.

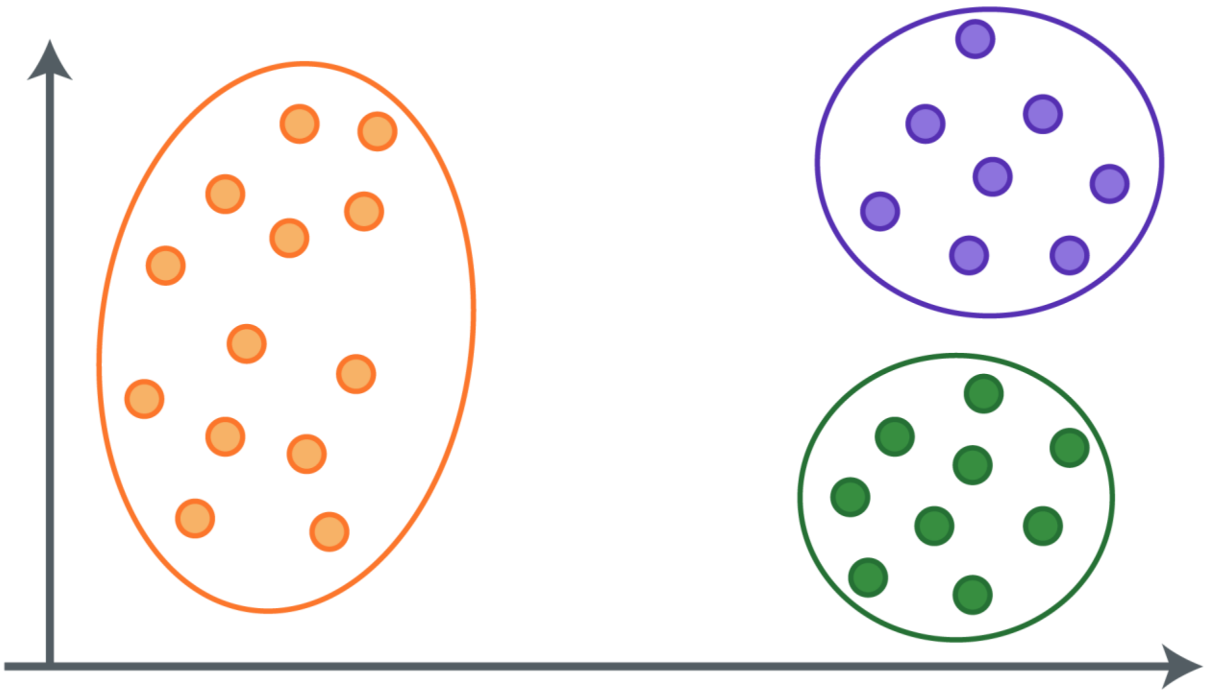


Random Initialization Trap

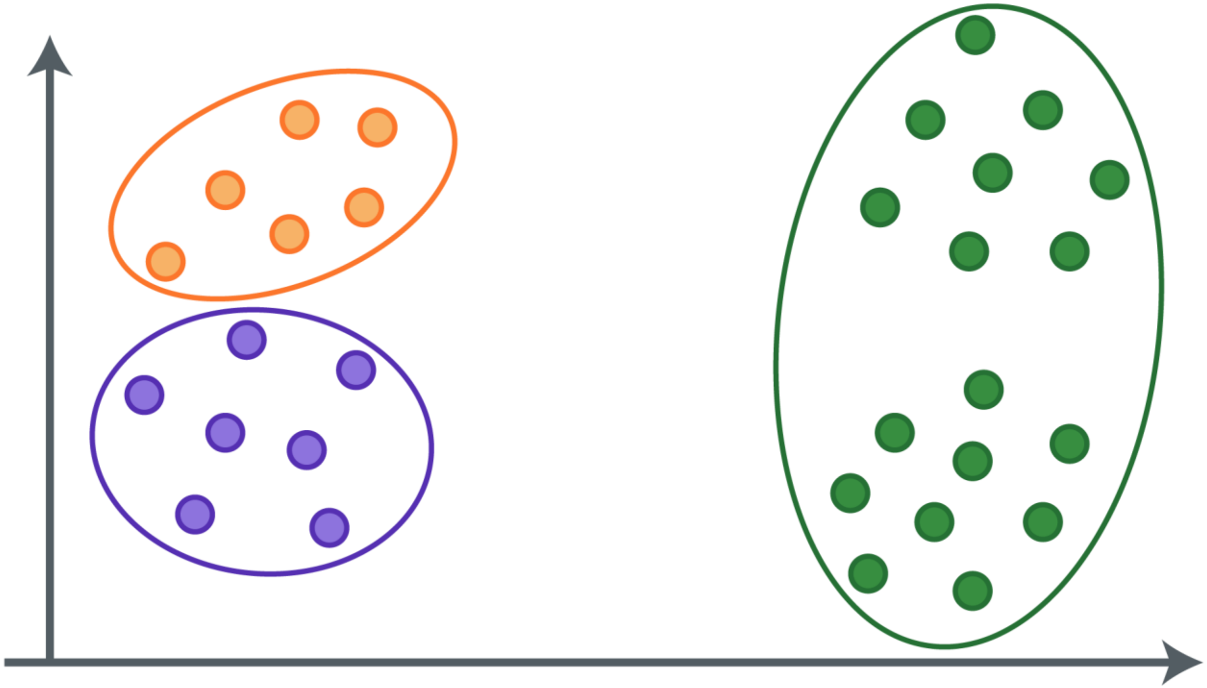
STEP 5: Reassign each data point to the new closest centroid.
If any reassignment took place, go to STEP 4, otherwise go to FIN.



Random Initialization Trap



Random Initialization Trap

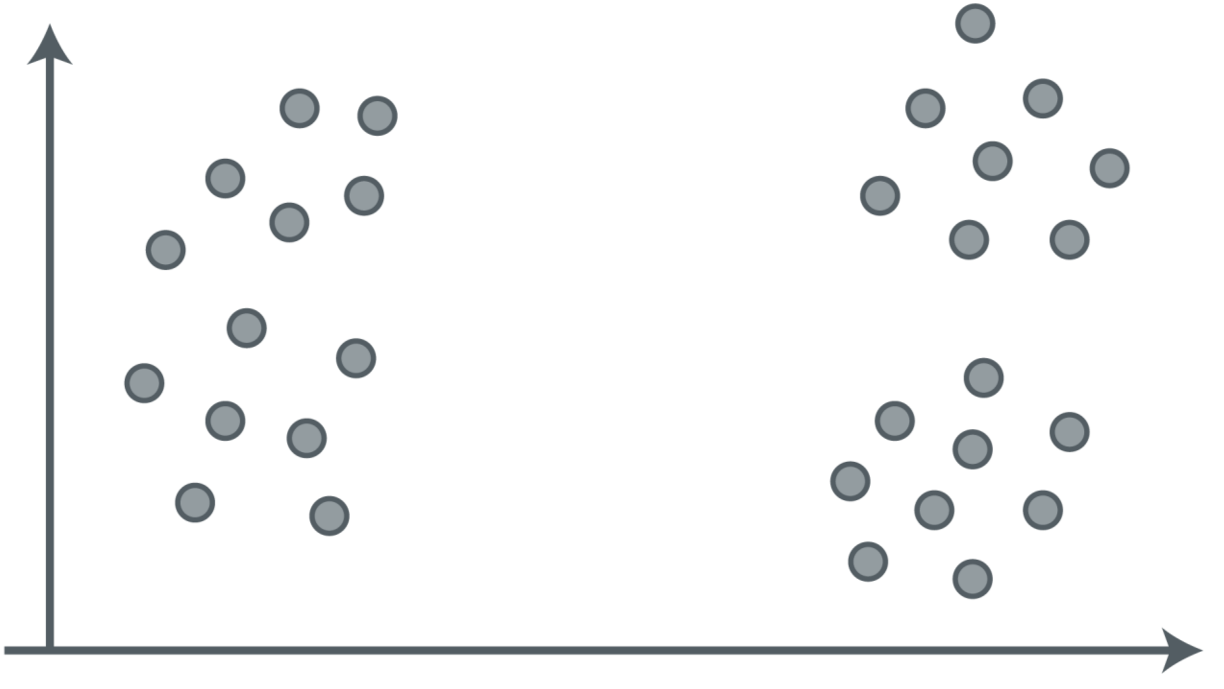


Random Initialization Trap

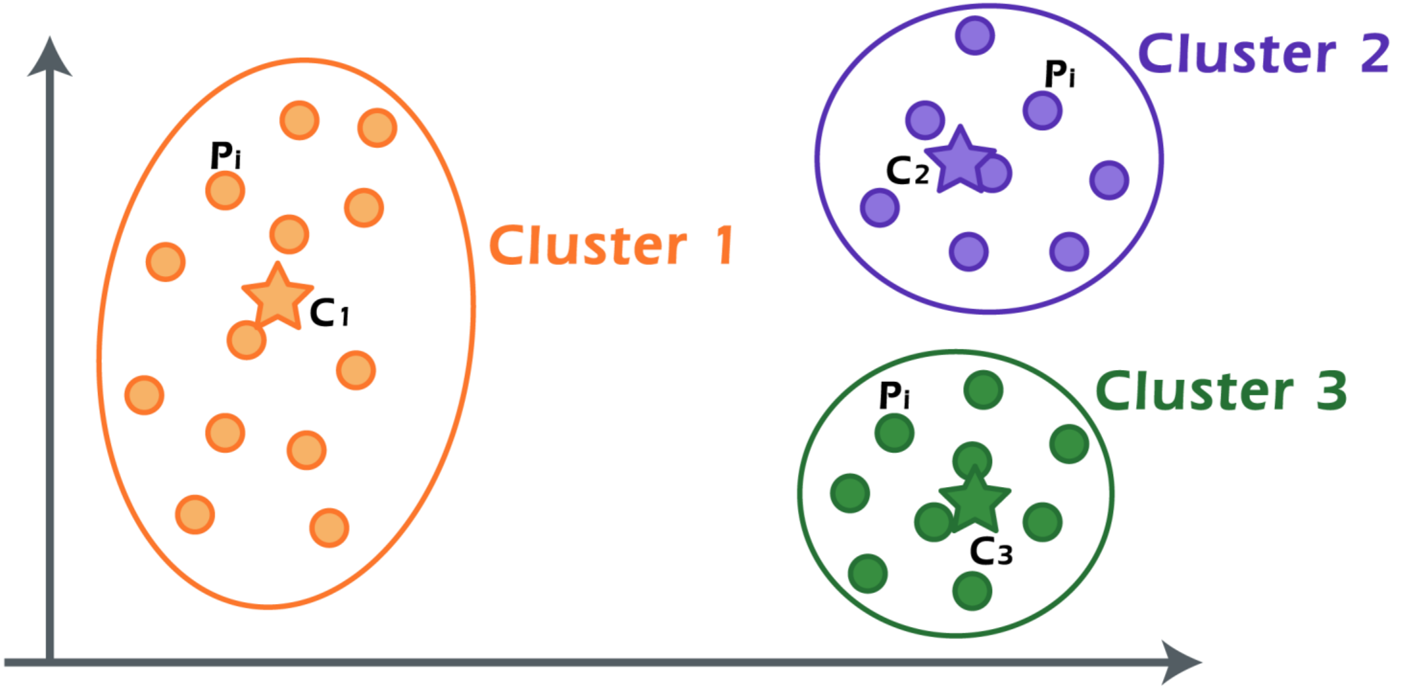
Solution → **K-Means++**

K-Means Intuition:
Choosing the right number of clusters

Choosing the right number of clusters



Choosing the right number of clusters

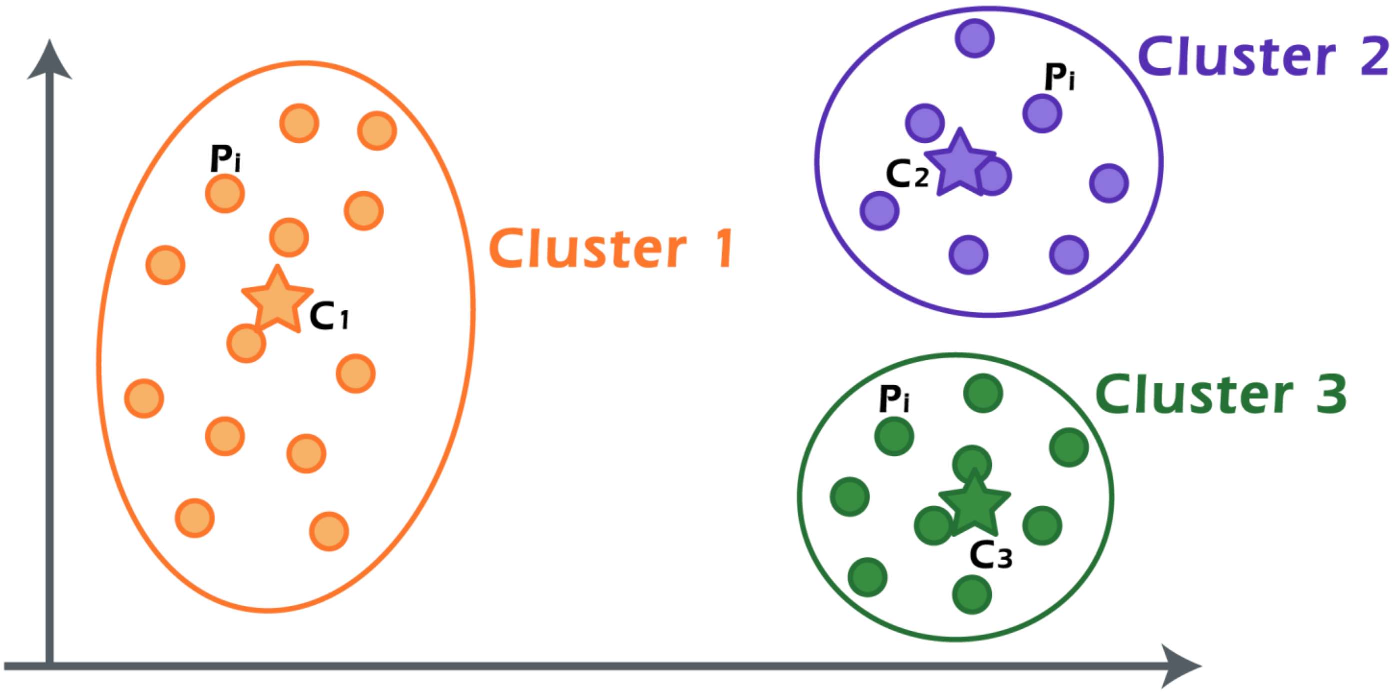


Choosing the right number of clusters

组内平方和

$$\mathrm{WCSS} = \sum_{\mathrm{P_i \ in \ Cluster \ 1}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_1})^2 + \sum_{\mathrm{P_i \ in \ Cluster \ 2}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_2})^2 + \sum_{\mathrm{P_i \ in \ Cluster \ 3}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_3})^2$$

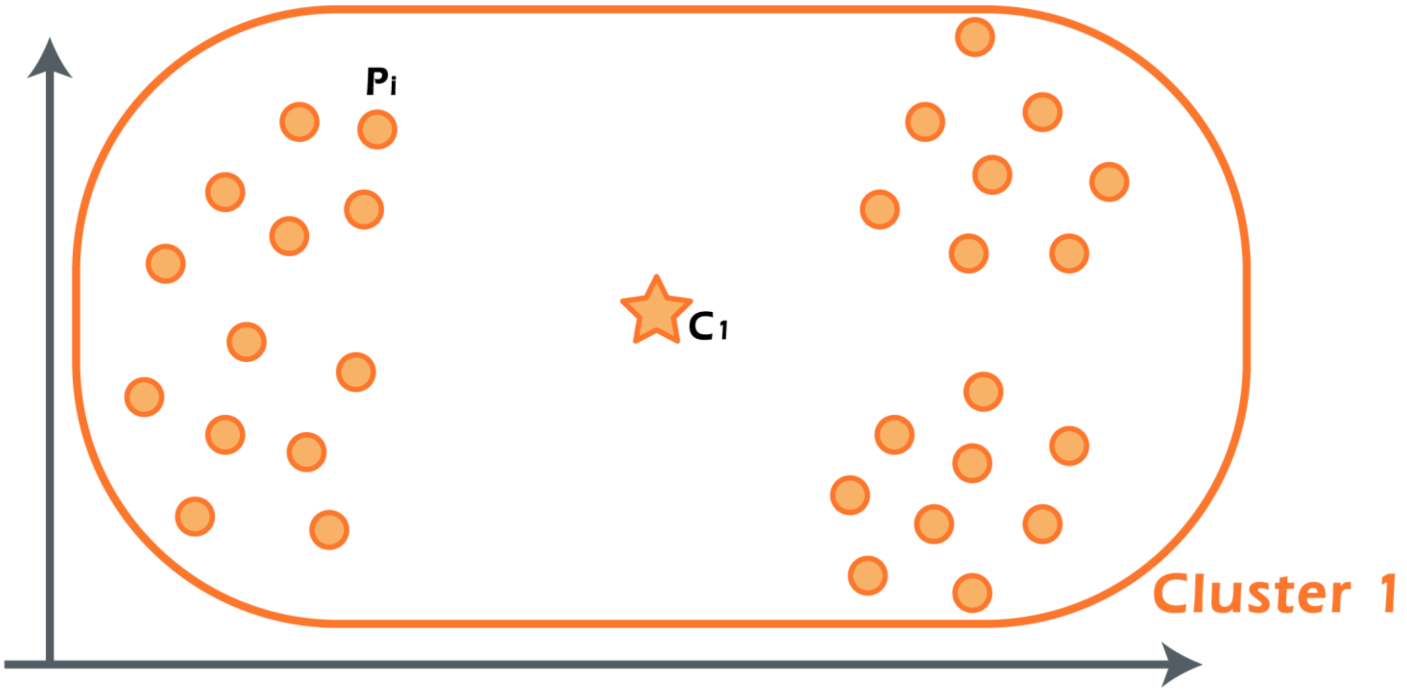
Choosing the right number of clusters



组内平方和

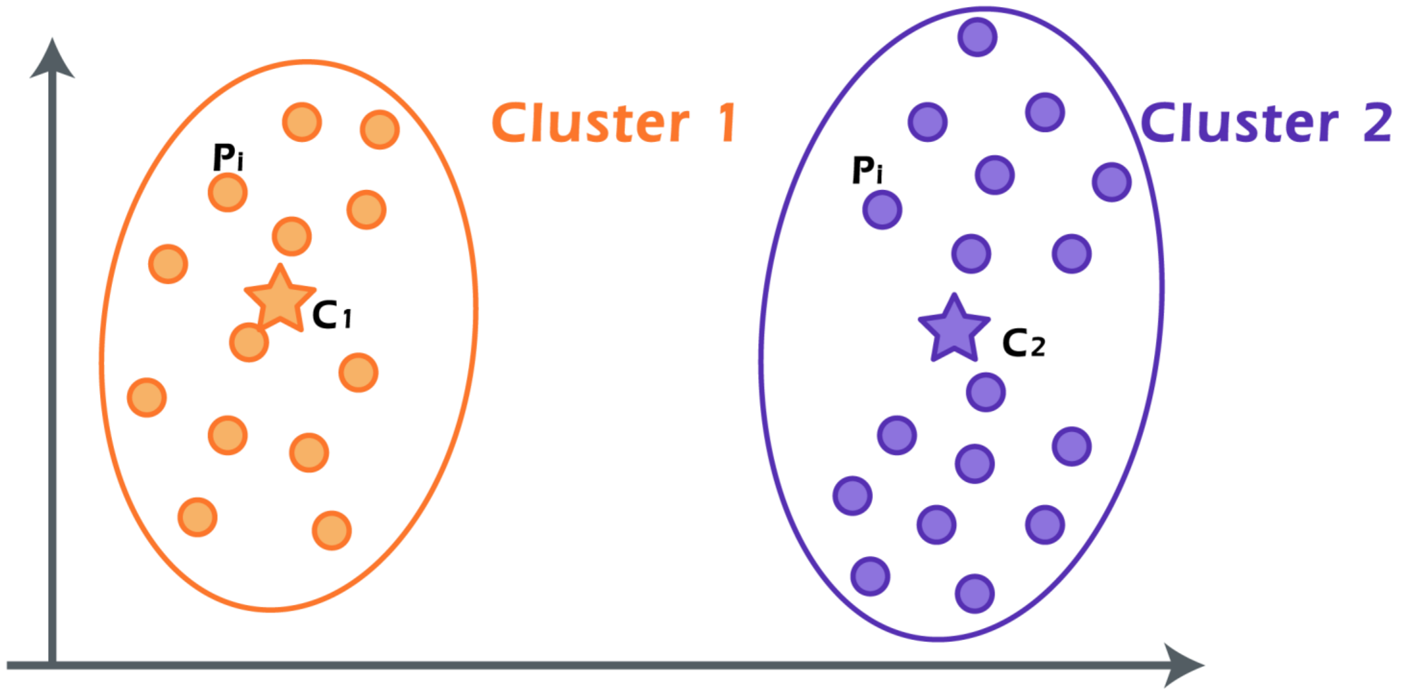
$$\mathrm{WCSS} = \sum_{P_i \text{ in Cluster } 1} \mathrm{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster } 2} \mathrm{distance}(P_i, C_2)^2 + \sum_{P_i \text{ in Cluster } 3} \mathrm{distance}(P_i, C_3)^2$$

Choosing the right number of clusters



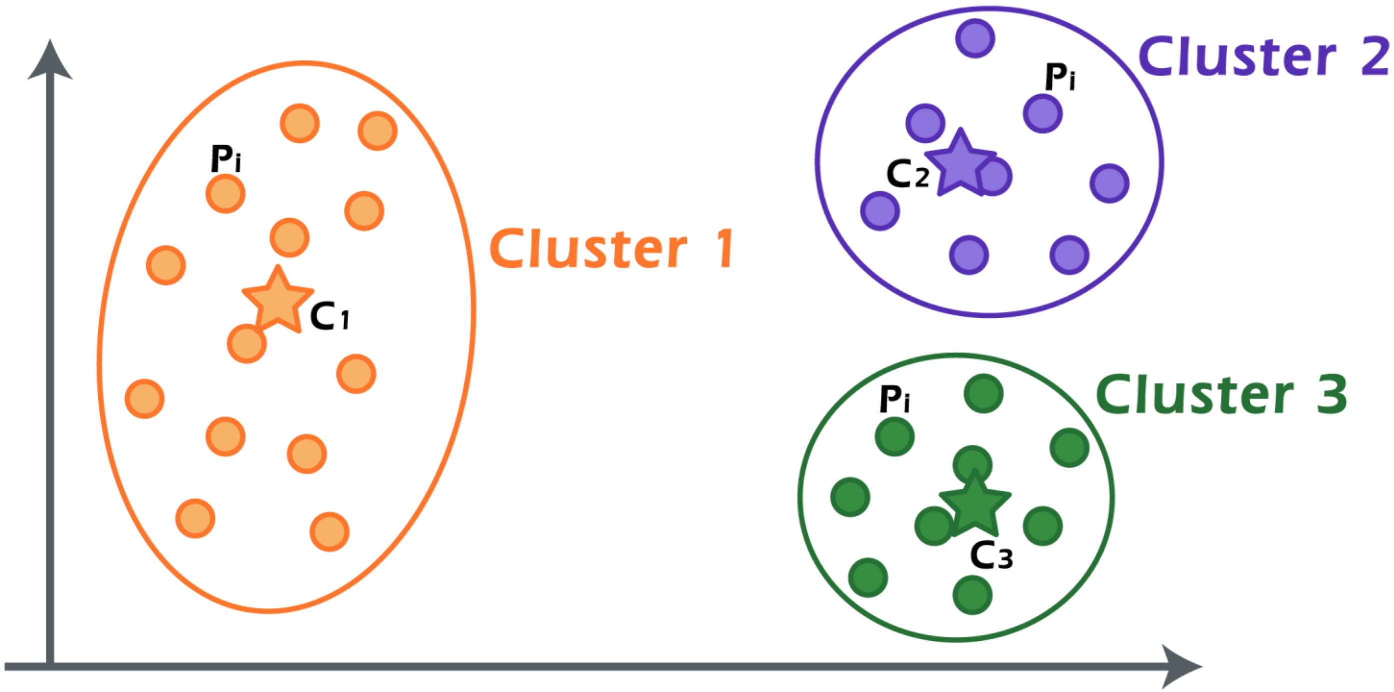
$$\mathrm{WCSS} = \sum_{\mathrm{P_i\ in\ Cluster\ 1}} \mathrm{distance}(\mathrm{P_i}, \mathrm{C_1})^2$$

Choosing the right number of clusters



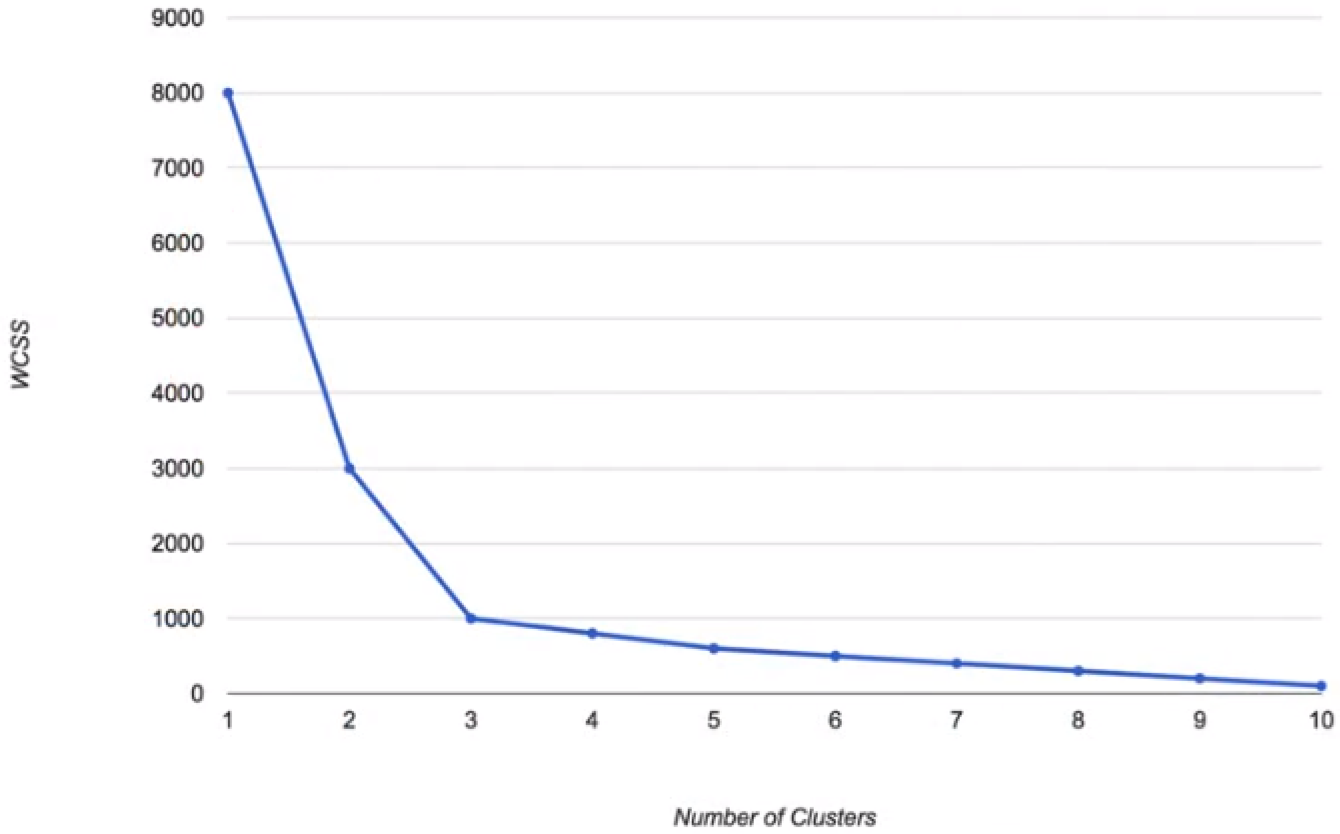
$$\mathrm{WCSS} = \sum_{P_i \text{ in Cluster 1}} \mathrm{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \mathrm{distance}(P_i, C_2)^2$$

Choosing the right number of clusters



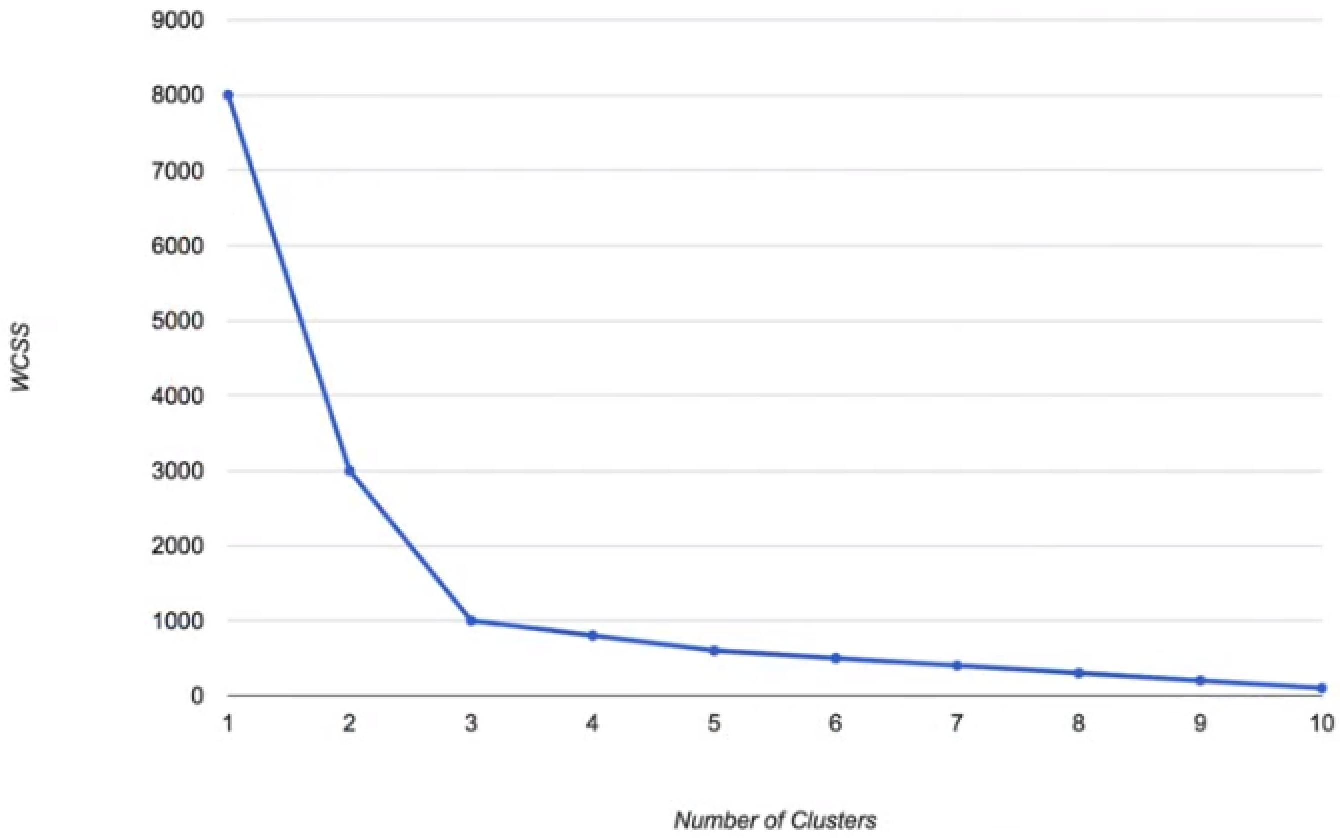
$$\text{WCSS} = \sum_{P_i \text{ in Cluster 1}} \text{distance}(P_i, C_1)^2 + \sum_{P_i \text{ in Cluster 2}} \text{distance}(P_i, C_2)^2 + \sum_{P_i \text{ in Cluster 3}} \text{distance}(P_i, C_3)^2$$

Choosing the right number of clusters



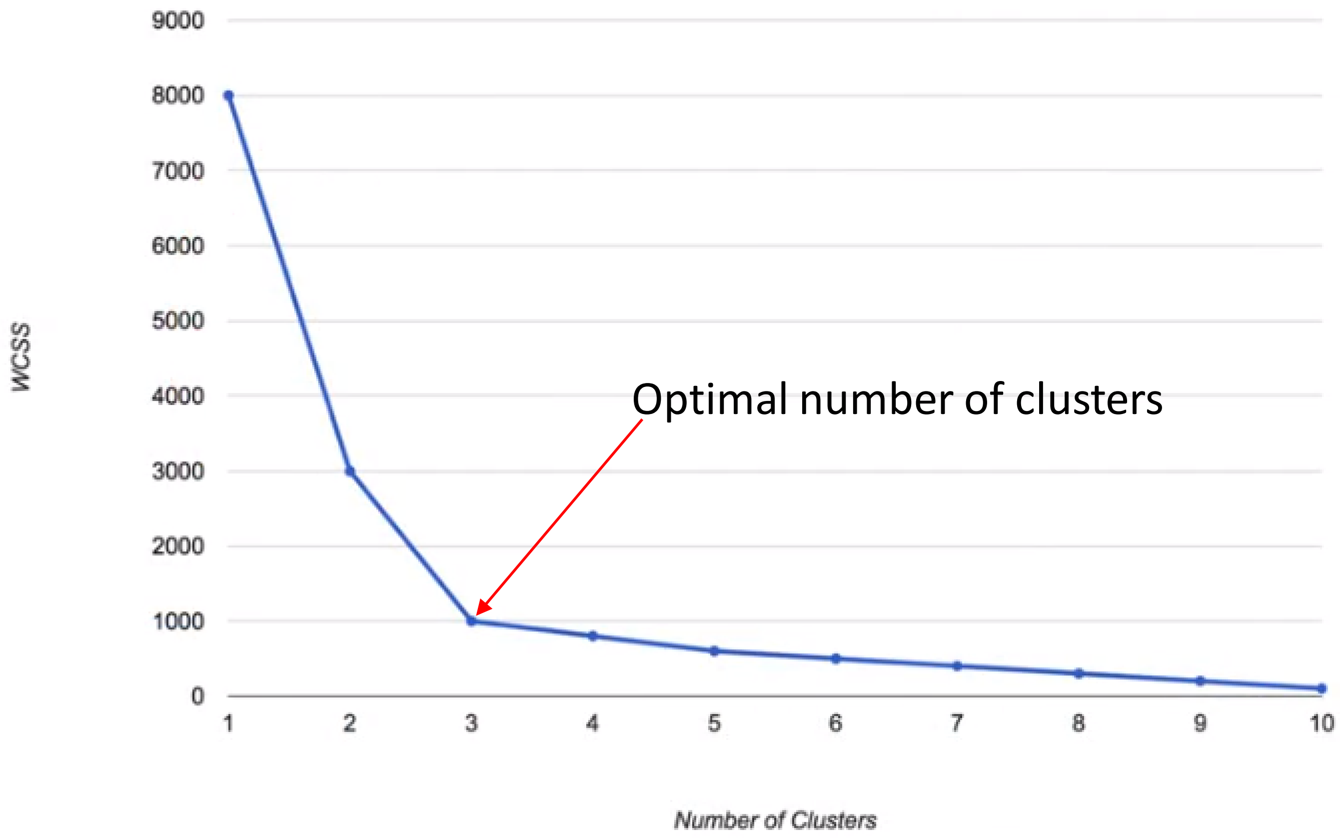
Choosing the right number of clusters

- The Elbow Method 手肘法則



Choosing the right number of clusters

- The Elbow Method 手肘法則



THE END

ytlin@mail.nptu.edu.tw