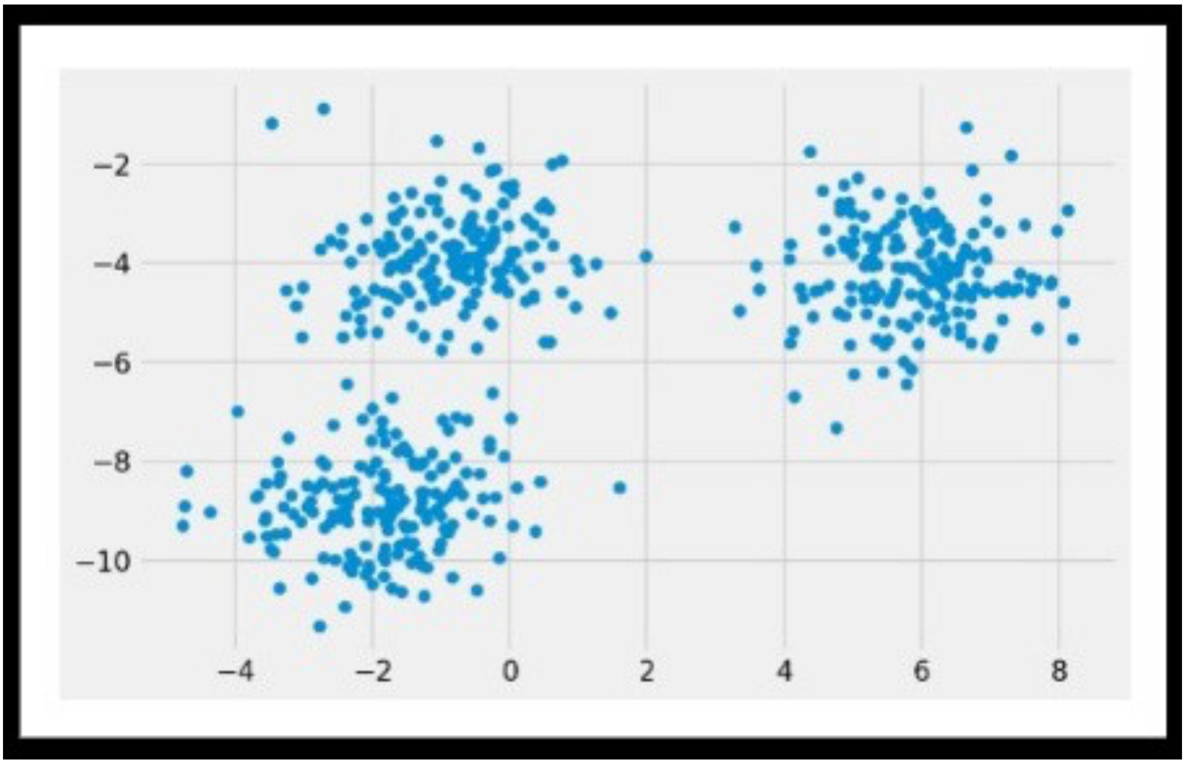
# k means algorithm

***Lloyd’s Algorithm:***
Lloyd’s algorithm is an approximation iterative algorithm used to cluster points. The steps of the algorithm are as follows:

- Initialization
- Assignment
- Update Centroid
- Repeat Step 2 and 3 until convergence.

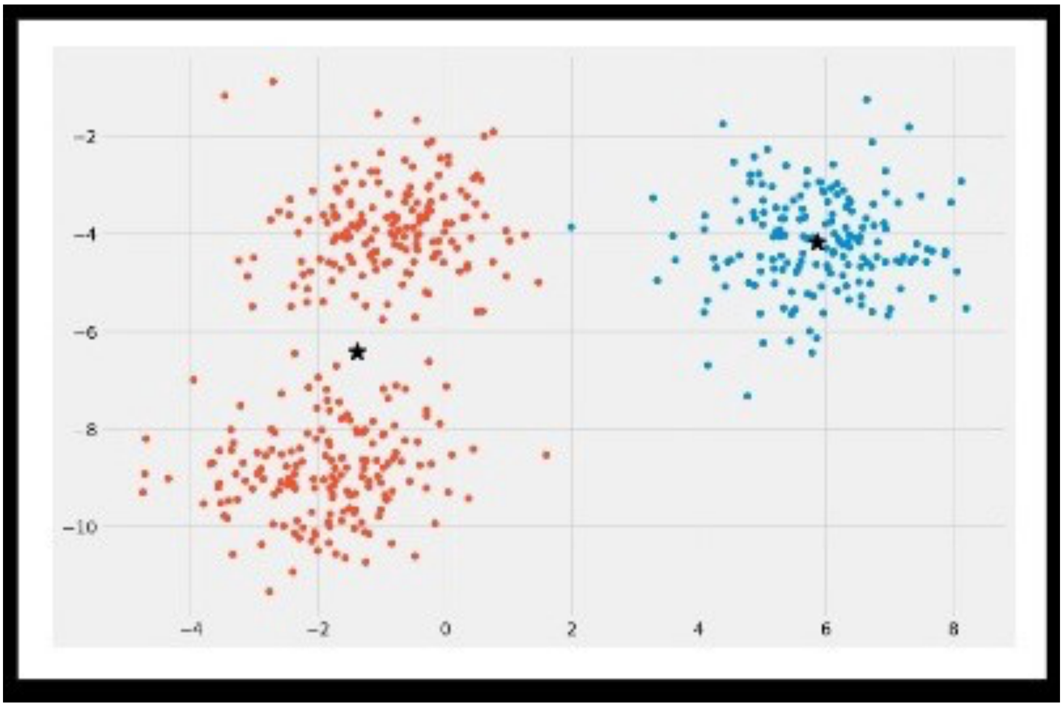
## *Iterative implementation of the K-Means algorithm:*

**Steps #1: Initialization:**
The initial k-centroids are randomly picked from the dataset of points (lines 27–28).
**Steps #2: Assignment:**
For each point in the dataset, find the euclidean distance between the point and all centroids (line 33). The point will be assigned to the cluster with the nearest centroid.
**Steps #3: Updation of Centroid:**
Update the value of the centroid with the new mean value (lines 39–40).
**Steps #4: Repeat:**
Repeat steps 2 and 3 unless convergence is achieved. If convergence is achieved then break the loop(line 43). Convergence refers to the condition where the previous value of centroids is equal to the updated value.
***Results:***
Plot for the initial dataset (Image 4)



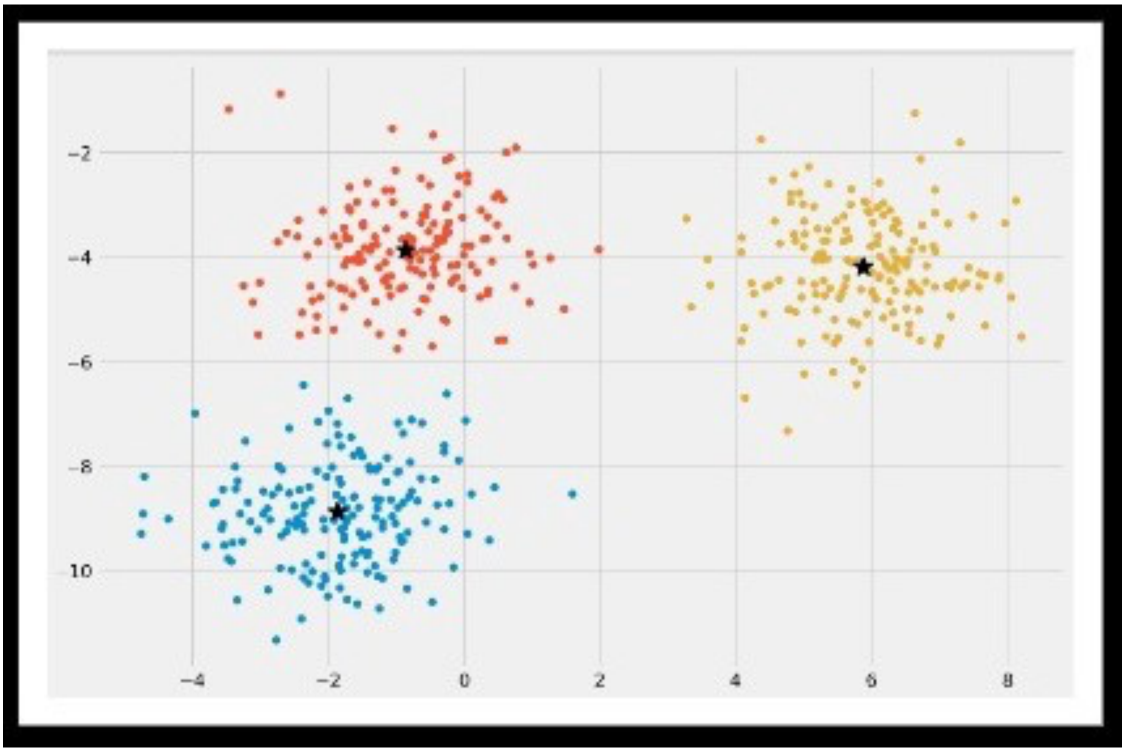
*The Plot of the dataset, (Image 4)*

Clustering result plot for k=2 (Image 5)



*The plot of clustering for k=2, (Image 5)*

Clustering result plot for k=3 (Image 6)



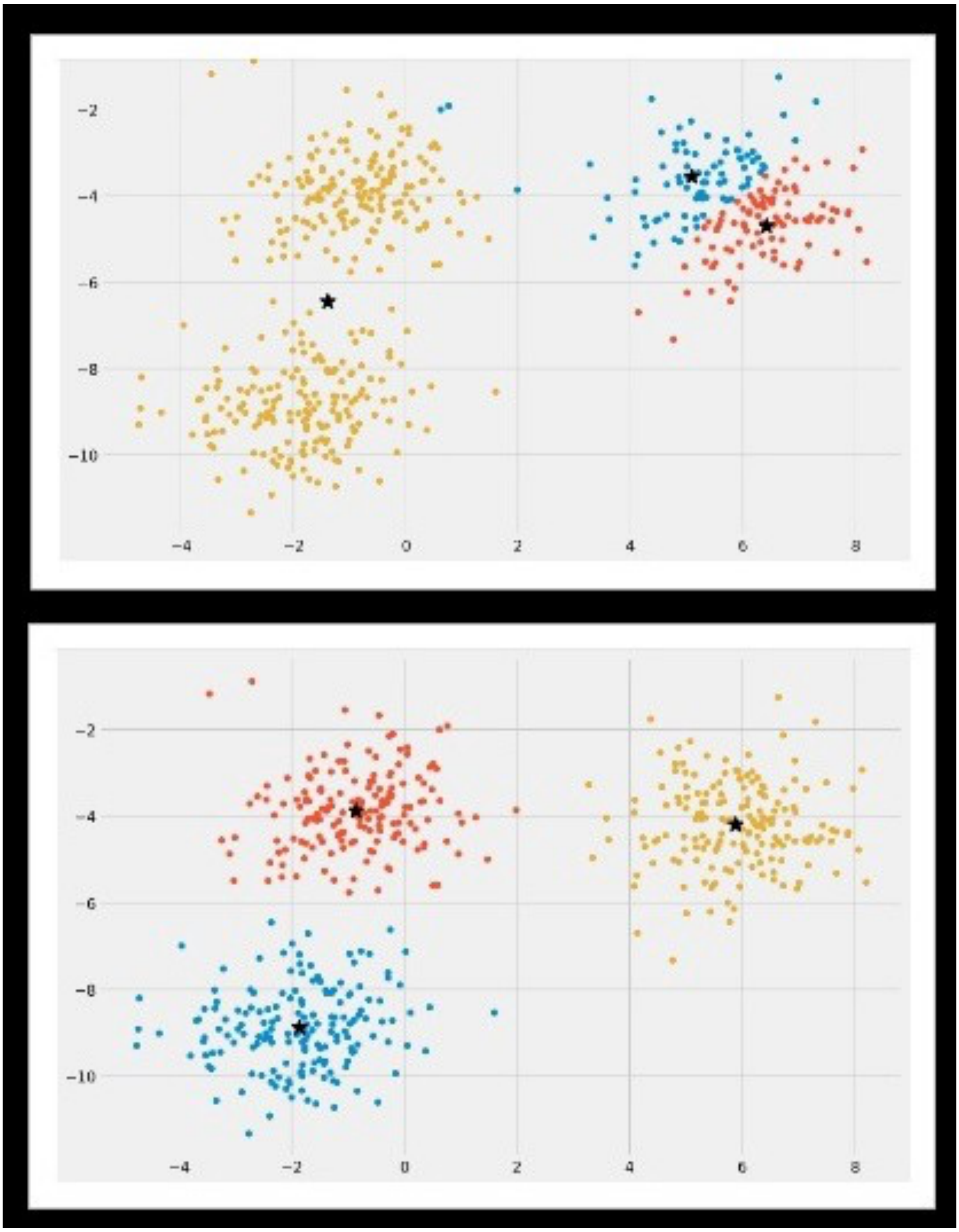
*The plot of clustering for k=3, (Image 6)*

## *K-Means++ Clustering:*

In the case of finding initial centroids using Lloyd's algorithm for K-Means clustering, we were using randomization. The initial k-centroids were picked randomly from the data points.

This randomization of picking k-centroids points results in the problem of initialization sensitivity. This problem tends to affect the final formed clusters. The final formed clusters depend on how initial centroids were picked.

Here are some outcomes of clustering where the initialization of centroids was different:



*Different Final Clusters are formed by different initialization, (Image 7)*

In the above image above (Image 7), the final formed clusters are different becomes the final formed clusters are dependent on the initialization of centroids. In the 1st part of the above image, it is observed that centroids (black *) and the clusters are not properly formed. In the 2nd part of the above image, it is observed that centroids (black *) and the clusters are properly formed.

There are two approaches to avoid this problem of initialization sensitivity:

- **Repeat K-means:** Repeat the algorithm and initialization of centroids several times and pick the clustering approach that has a small intracluster distance and large intercluster distance.
- **K-Means++:** K-Means++ is a smart centroid initialization technique.

The above two methods can be used to avoid the problem of initialization sensitivity but among the two K-Means++ is the best approach.

### *How do K-Means++ work?*

K-Means++ is a smart centroid initialization technique and the rest of the algorithm is the same as that of K-Means. The steps to follow for centroid initialization are:

- Pick the first centroid point (C_1) randomly.
- Compute the distance of all points in the dataset from the selected centroid. The distance of x_i point from the farthest centroid can be computed by

$$d_i = max_{(j:1 \mapsto m)} ||x_i - C_j||^2$$

d_i: Distance of x_i point from the farthest centroid
m: number of centroids already picked

- Make the point x_i as the new centroid that is having maximum probability proportional to d_i.
- Repeat the above two steps till you find k-centroids

k - numbers of cluster

4 steps:

1. Randomly choose k- centroids
2. assign each data set to cluster with the nearest centroids
3. compute each centroid as a mean of the object assigned to it
4. repeat step2 until no change happens

k = 2

| Data set | x1(feature) | y1 (feature) |
|---|---|---|
| 1 | 1.0 | 1.0 |
| 2 | 1.5 | 2.0 |
| 3 | 3.0 | 4.0 |
| 4 | 5.0 | 7.0 |
| 5 | 3.5 | 5.0 |
| 6 | 4.5 | 5.0 |
| 7 | 3.5 | 4.5 |

**Step1**: pick 2 centroids randomly

centroid 1 = (1.0, 1.0)
centroid 2 = (5.0, 7.0)

**step2:**

Euclidean distance : SQRT((x2—x1)^2 + (y2-y1)^2)

d(1, centroid1) = 0
d(1, centroid2) = sqrt( (1-5)^2 + (7-1)^2) = sqrt( 16+36)= sqrt(50) = 7.21

d(2, centriod1) =
d(2, centrod2) =

……

| Data set | Centroid1 | Centroid2 |
|---|---|---|
| 1 | 0 | 7.21 |
| 2 | 1.2 | 6.10 |
| 3 | 3.61 | 3.61 |
| 4 | 7.21 | 0 |
| 5 | 4.72 | 2.5 |
| 6 | 5.31 | 2.06 |
| 7 | 4.3 | 2.92 |

group 1: 1, 2, 3
group 2: 4, 5, 6, 7

**step3:**
new centroid (m1) = 1/3 (1+1.5+3), 1/3(1+2+4) = 1/3(5.5), 1/3(7) = 1.83, 2.33
new centroid (m2) = 1/4(5+3.5+4.5+3.5) + 1/4(7+5+5+4.5) = 4.12, 5.38

d(1, m1) = sqrt( (1-1.83)^2 + (2.33-1.0)^2) = 1.5677
d(1, m2) = 5.3776

……

| Data set | Centroid1 | Centroid2 |
|---|---|---|
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |

**step4:**Have any data sets switched groups?
If Yes, repeat step2.
if No, stop here.