LAB Manual

PART A

(PART A : TO BE REFFERED BY STUDENTS)

**Experiment No.10**

**Aim:** Implementation of K-means Algorithm for Clustering on Loan data set.

**Prerequisites:**

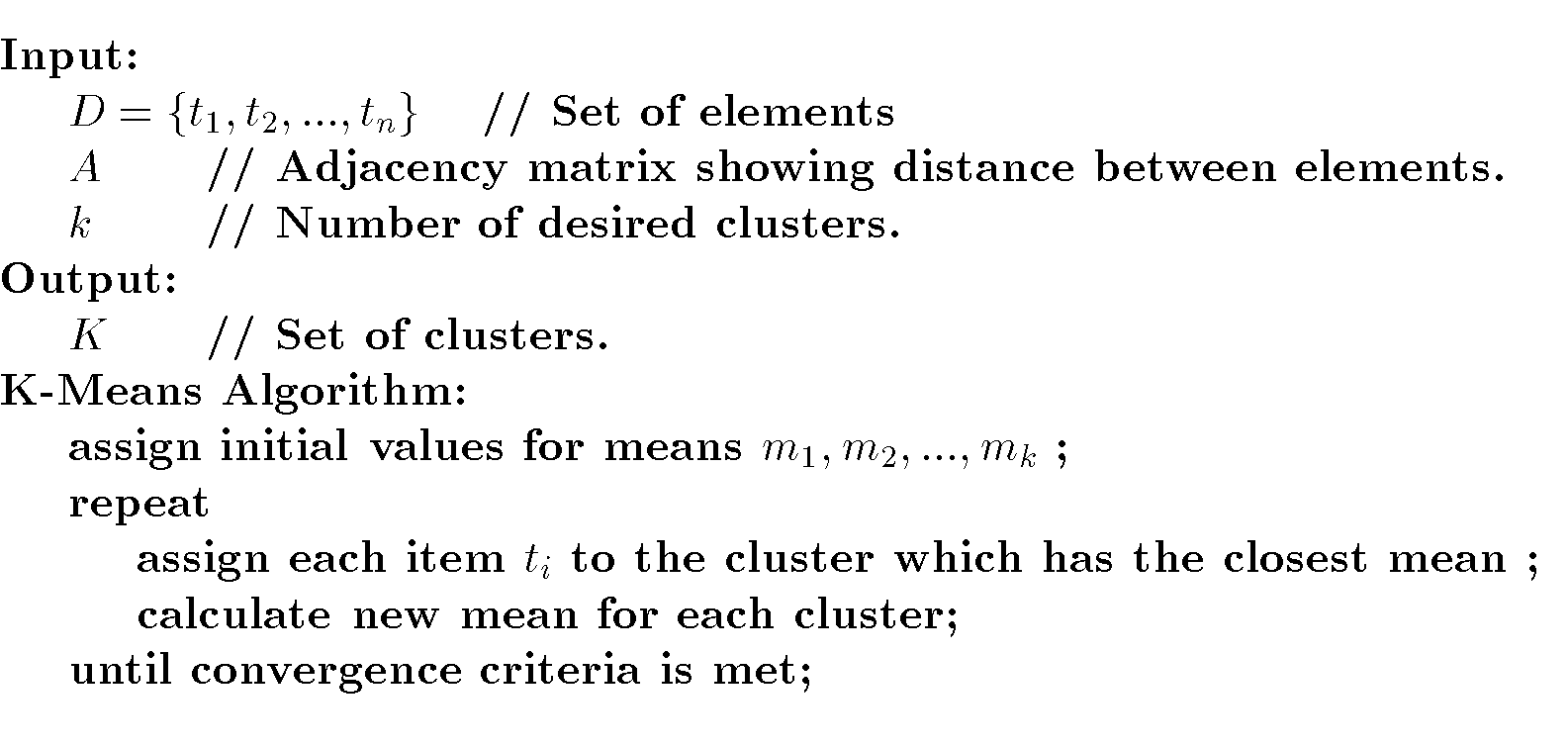
Python Programming

**Learning Outcomes:**

Concepts of K-means Algorithm and Clustering.

**Theory:**

**Algorithm:**

****

**Introduction to K-Means Clustering**

K-means is a centroid-based algorithm, or a distance-based algorithm, where we calculate the distances to assign a point to a cluster. In K-Means, each cluster is associated with a centroid. The main objective of the K-Means algorithm is to minimize the sum of distances between the points and their respective cluster centroid.

Steps for K-means algorithm

1. Choose the number of clusters *k*

*2.* Select *k* random points from the data as centroids.

3. Calculate distance for each point from each of the cluster.

4. Assign all the points to the closest cluster centroid

5. Recompute the centroids of newly formed clusters

6. Repeat steps 3 to 5 till the selected stopping criteria is reached

**Stopping criteria for K-Means clustering**

There are essentially three stopping criteria that can be adopted to stop the K-means algorithm:

1. Centroids of newly formed clusters do not change
2. Points remain in the same cluster
3. Maximum number of iterations are reached

**Task to be performed:**

Task 1:

For the given data points identify clusters using K-Means (upto 2 iteration).

K=2 initial centers= 1 and 4

|  |  |  |
| --- | --- | --- |
|  | A | B |
| 1 | 1 | 1 |
| 2 | 1.5 | 2 |
| 3 | 3 | 4 |
| 4 | 5 | 7 |
| 5 | 3.5 | 5 |
| 6 | 4.5 | 5 |
| 7 | 3.5 | 4.5 |

Task 2:

Implementation of K-Means Clustering in Python using Loan data set.

1. The data set is uploaded in MS.Teams. Download it.

3. Apply K-Means algorithm on the data set for K=3 clusters.

4. Use Elbow method to identify numbers of cluster.

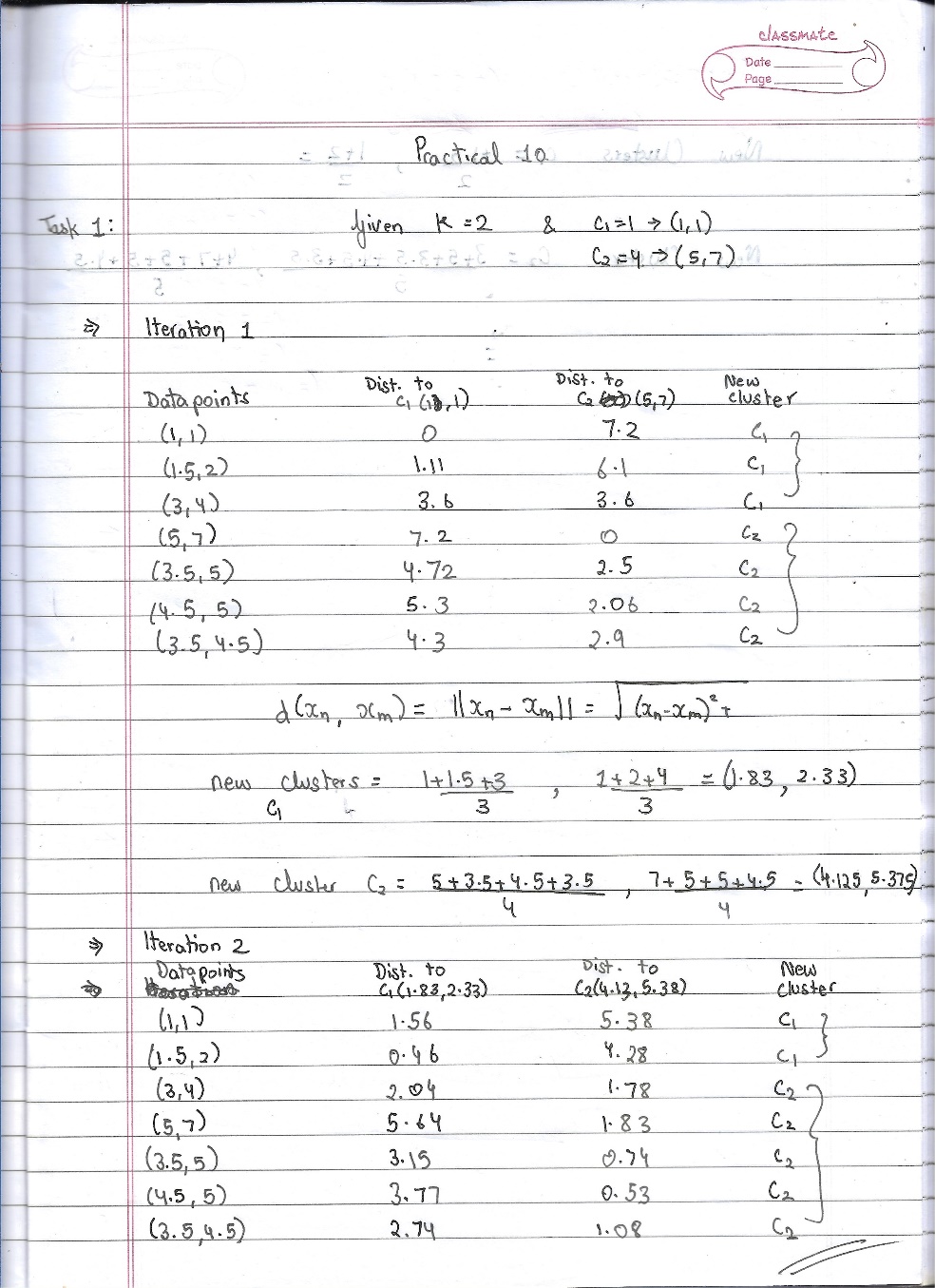
PART B

(PART B : TO BE COMPLETED BY STUDENTS)

***(Students must submit the soft copy as per following segments within two hours of the practical slot. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)***

|  |  |
| --- | --- |
| Roll No. N052 | Name: Pratyush Kumar |
| Class : MBA Tech CE | Batch : B2 |
| Date of Experiment: 16-03-2024 | Date of Submission : 16-03-2024 |
| Grade : | Time of Submission: |
| Date of Grading: | |

**B.1 Solved solution for task 1:**



**B.2 Software Code written by student:**

***(Paste your c/c++/java code completed during the 2 hours of practical in the lab here)***

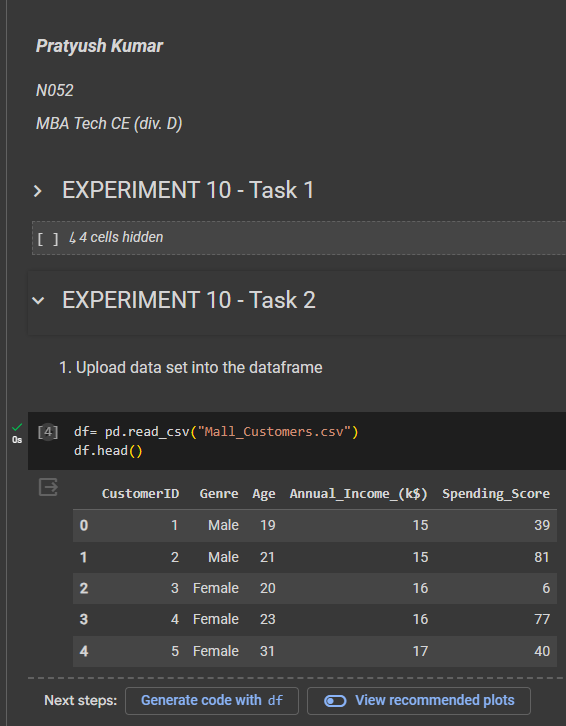
**Colab link:** https://colab.research.google.com/drive/1J4lUIt6zZrHyyunUcwTTCtKaXQ1bB1YE?usp=sharing

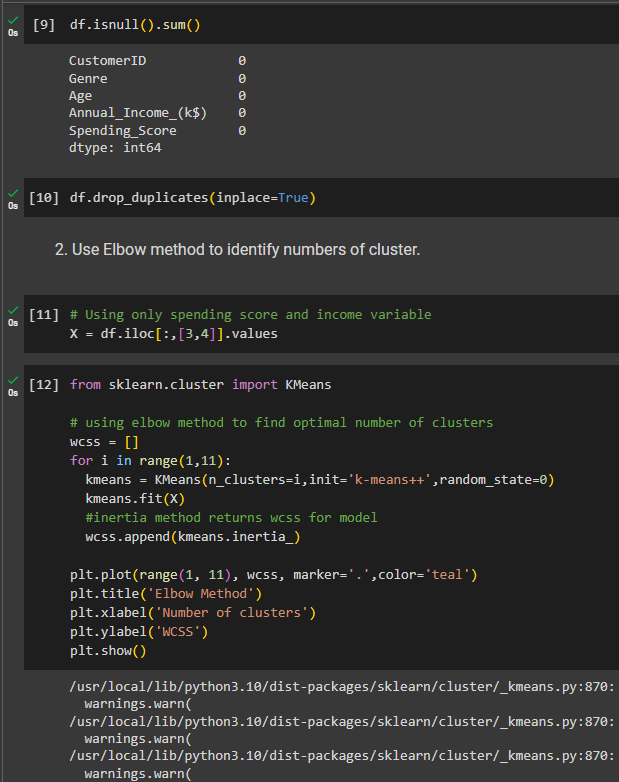
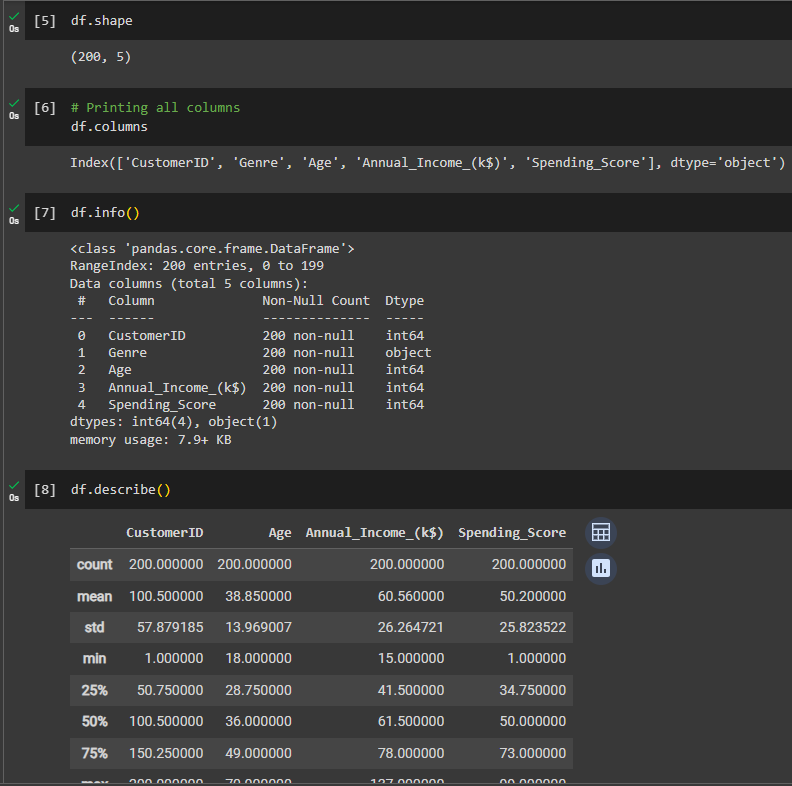
* **Source Code**

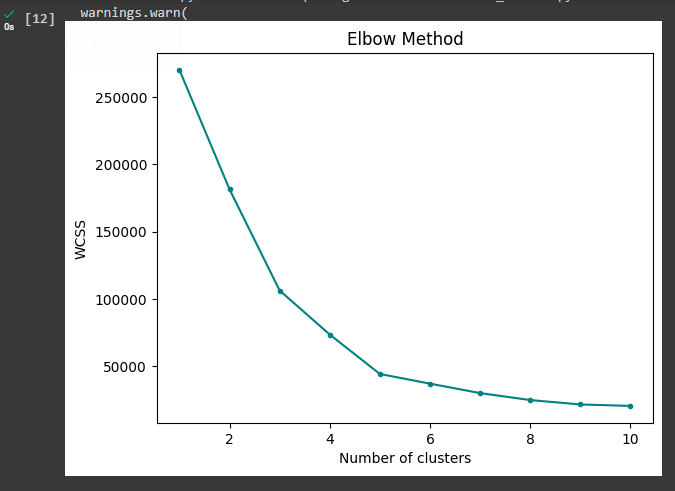
*"""  
 \* This file contains code snippets to implement K-means Algorithm for Clustering on Loan data set.  
 \* ML-E10-Task2  
 \*  
 \* Original file is located at: https://colab.research.google.com/drive/1J4lUIt6zZrHyyunUcwTTCtKaXQ1bB1YE  
 \* @author Pratyush Kumar (github.com/pratyushgta)  
"""*"""  
## EXPERIMENT 10 - Task 2  
"""  
  
"""  
1. Upload data set into the dataframe  
"""  
df= pd.read\_csv("Mall\_Customers.csv")  
df.head()  
  
df.shape  
  
# Printing all columns  
df.columns  
  
df.info()  
  
df.describe()  
  
df.isnull().sum()  
  
df.drop\_duplicates(inplace=True)  
  
"""2. Use Elbow method to identify numbers of cluster."""  
  
# Using only spending score and income variable  
X = df.iloc[:,[3,4]].values  
  
from sklearn.cluster import KMeans  
  
# using elbow method to find optimal number of clusters  
wcss = []  
for i in range(1,11):  
 kmeans = KMeans(n\_clusters=i,init='k-means++',random\_state=0)  
 kmeans.fit(X)  
 #inertia method returns wcss for model  
 wcss.append(kmeans.inertia\_)  
  
plt.plot(range(1, 11), wcss, marker='.',color='teal')  
plt.title('Elbow Method')  
plt.xlabel('Number of clusters')  
plt.ylabel('WCSS')  
plt.show()  
  
"""3. Apply K-Means algorithm on the data set for K=3 clusters."""  
  
# initializing KMeans model with 5 clusters  
kmeans = KMeans(n\_clusters = 5, init = "k-means++", random\_state = 42)  
y\_kmeans = kmeans.fit\_predict(X)  
  
df\_kmean=pd.DataFrame(y\_kmeans)  
print(y\_kmeans)  
  
#predict culster for x value  
#predict centers  
cent=kmeans.cluster\_centers\_  
print(cent)  
  
plt.scatter(X[:,0],X[:,1])  
plt.scatter(cent[:,0],cent[:,1],s=50,color='red',label='Centroids') #s = 100  
plt.scatter(X[y\_kmeans == 0, 0], X[y\_kmeans == 0, 1], s = 60, c = 'yellowgreen', label = 'Cluster1')  
plt.scatter(X[y\_kmeans == 1, 0], X[y\_kmeans == 1, 1], s = 60, c = 'slateblue', label = 'Cluster2')  
plt.scatter(X[y\_kmeans == 2, 0], X[y\_kmeans == 2, 1], s = 60, c = 'lightseagreen', label = 'Cluster3')  
plt.scatter(X[y\_kmeans == 3, 0], X[y\_kmeans == 3, 1], s = 60, c = 'plum', label = 'Cluster4')  
plt.scatter(X[y\_kmeans == 4, 0], X[y\_kmeans == 4, 1], s = 60, c = 'sandybrown', label = 'Cluster5')  
plt.xlabel('Annual Income ($)')  
plt.ylabel('Spending Score (1-100)')  
plt.legend()  
  
plt.show()

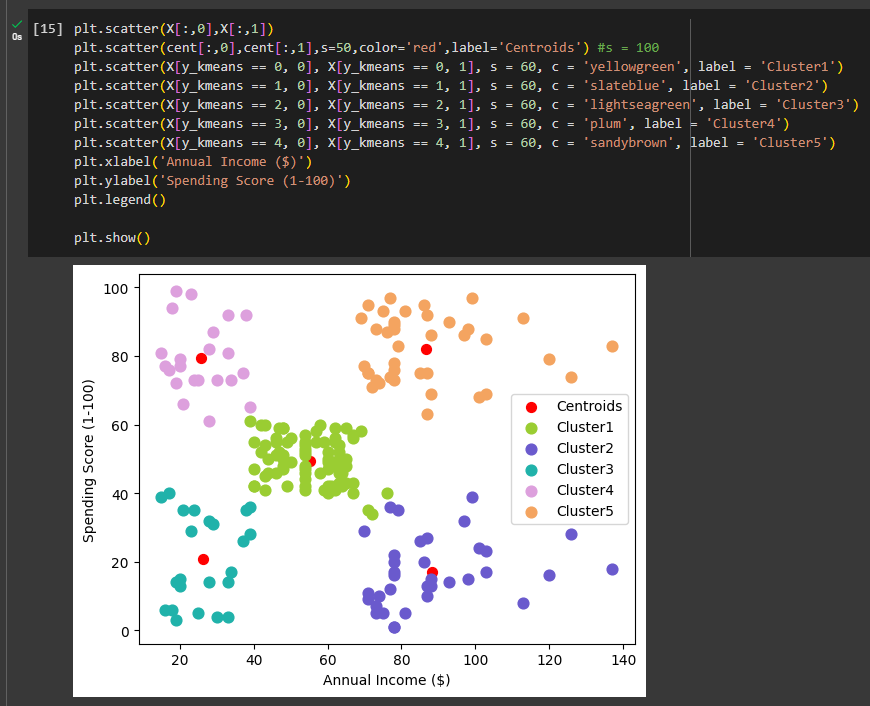
**B.3 Input and Output:**

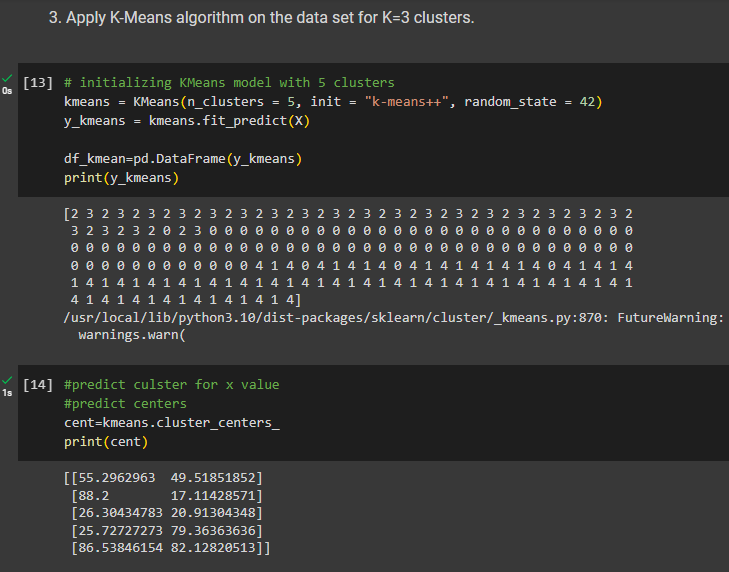
***(Paste your program input and output in following format, If there is error then paste the specific error in the output part. In case of error with due permission of the faculty extension can be given to submit the error free code with output in due course of time. Students will be graded accordingly.)***

****

****

****

****

****

**B.4 Conclusion:**

*(****Students must write the conclusions based on their learning)***

Performed K-means Algorithm for Clustering on Loan data set to group similar loan profiles into distinct clusters based on their features. It has been observed that the K-means algorithm, with its iterative approach to partitions data into K clusters based on similarity. By iteratively optimizing cluster centroids to minimize variance, it is able to efficiently assign data points to the most appropriate cluster.

**B.5 Question of Curiosity**

**Q.1 State any disadvantage\s or short comings of K-Means algorithm. Identify if any solution is available.**

**Disadvantages:**

1. **Sensitivity to Initial Centroid Selection:** The algorithm is sensitive to the initial placement of centroids. Different initializations can lead to different local optima, resulting in varying clustering outcomes.
2. **Choosing K manually:** The number of clusters needs to be set manually, which can be challenging as an incorrect choice can lead to inaccurate results.
3. **Inability to handle categorical data:** It works with numerical data where distances between data points can be calculated. However, it struggles with categorical data
4. **Outliers:** Centroids can be influenced by outliers, or outliers might get their own cluster instead of being ignored.

**Solutions to these disadvantages:**

1. **Elbow Method for Determining K:** Utilizing methods like the elbow method, silhouette score, or cross-validation can aid in selecting an appropriate number of clusters based on the intrinsic structure of the data.
2. **Outlier Handling:** Employing robust techniques for outlier detection and removal before clustering can help mitigate the impact of outliers on the clustering process.
3. **Initialization Techniques:** Using more sophisticated initialization techniques such as K-means++ can help mitigate the sensitivity to initial centroid selection, leading to more stable and accurate clustering results.