 Problem Statement

Program to implement k-means clustering technique using any standard dataset available in the public domain

 Dataset Description

In this project, we will be using the dataset holding the information of carbon dioxide emission from different car models. The dataset includes 36 instances with 5 columns which can be briefed as:

**Column Description**

Car Brand of the car

Model Model of the car

Volume Total space available inside the car (in )

*litres*

Weight Total weightof the car (in )

*kg*

*CO*2

Total emission of carbon dioxide from the car

**Note:** (This is a manually created custom dataset for this project.)

 List of Activities

**Activity 1:** Import Modules and Read Data

**Activity 2:** Data Cleaning

**Activity 3:** Find Optimal Value of K

**Activity 4:** Plot Silhouette Scores

 Activity 1: Import Modules and Read Data

Import the necessary Python modules along with the following modules:

KMeans - For clustering using K-means.

re - To remove unwanted rows using regex.

Read the data from a CSV file to create a Pandas DataFrame and go through the necessary data-cleaning process (if required). **Dataset link:** https://raw.githubusercontent.com/jiss-sngce/CO\_3/main/jkcars.csv

1

# Import the modules and Read the data.

2

3

# Print the first five records

4

1

# Get the total number of rows and columns, data types of columns and missing values (if exist) in the dataset. 2

 Activity 3: Find Optimal value of K

In this activity, you need to find the optimal value of K using the silhouette score. 1. Create a subset of the dataset consisting of three columns i.e Volume , Weight , and CO2 .

1

# Create a new DataFrame consisting of three columns 'Volume', 'Weight', 'CO2'.

2

3

# Print the first 5 rows of this new DataFrame.

4

2. Compute K-Means clustering for the 3D dataset data\_3d by varying K from 2 to 10 clusters. Also, for each K , calculate silhouette score using silhouette\_score function.

**Steps to Follow**

Create an empty list to store silhouette scores obtained for each K (let's say sil\_scores ).

Initiate a for loop that ranges from 2 to 10.

Perform K-means clustering for the current value of K inside for loop.

Use fit() and predict() to create clusters.

Calculate silhouette score for current K value using silhouette\_score() function and append it to the empty list sil\_scores .

Create a DataFrame with two columns. The first column must contain K values from 2 to 10 and the second column must contain silhouette values obtained after the for loop.

1

# Calculate inertia for different values of 'K'.

2

3

# Create an empty list to store silhouette scores obtained for each 'K'

4

**Q**: What are the maximum silhouette score and the corresponding cluster value?

**A**:

Maximum silhouette score=

Corresponding cluster value=

 Activity 4: Plot silhouette Scores find optimal value for K

*x y*

Create a line plot with K ranging from 2 to 10 on the -axis and the silhouette scores stored in sil\_scores list on the -axis.

1 2

# Plot silhouette scores vs number of clusters.

**Q:** Write your observations of the graph.

**A:** From the graph, we can conclude that the optimal value of K is 3.

1 2 3 4 5 6 7 8 9 10

1 2

# Clustering the dataset for K = 3

# Perform K-Means clustering with n\_clusters = 3 and random\_state = 10

# Fit the model to the scaled\_df

# Make a series using predictions by K-Means

# Create a DataFrame with cluster labels for cluster visualisation