EXPLORING THE CAR DATASET Guided by: MADHAVI NIMKAR MAM (MITAOE)

Name of group members:

1.Omkar Abhale	(601)
2. Aryaman Deshmukh	(606)
3. Sachin Chandrukar	(609)
4. Swapnil Ahire	(603)
5. Rajesh Gaikwad	(617)

#Code of Linear Regression:-

```
#Linear Regression

import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
data = pd.read_csv('/content/AQI and Lat Long of Countries2.csv')
X = data[['lat', 'lng']]
y = data['AQI Value']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
rmse = mean_squared_error(y_test, y_pred, squared=False)

print(f"RMSE: {rmse}")
r2 = r2_score(y_test, y_pred)

print(f"R-squared: {r2}")
example_data = [[44.7444, 44.2031]]
predicted_aqi = model.predict(example_data)

print(f"Predicted AQI Value: {predicted_aqi}")
```

Output :-

```
RMSE:45.3923032194421
R-squared: 0.02458718467384724
Predicted AQI Value: [65.34166434]
```

#Code of KNN:-

```
#KNN
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score
data = pd.read_csv('/content/AQI and Lat Long of Countries2.csv')
X = data[['AQI Value', 'CO AQI Value', 'Ozone AQI Value', 'NO2 AQI
Value', 'PM2.5 AQI Value']]
y = data['AQI Category']
le = LabelEncoder()
y = le.fit_transform(y)
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
k = 3
knn = KNeighborsClassifier(n_neighbors=k)
knn.fit(X_train, y_train)
y_pred = knn.predict(X_test)
y_pred = le.inverse_transform(y_pred)
accuracy = accuracy_score(le.inverse_transform(y_test), y_pred)
print(f"Accuracy: {accuracy}")
example_data = [[60, 2, 35, 1, 40]]
predicted_category = le.inverse_transform(knn.predict(example_data))
print(f"Predicted AQI Category: {predicted_category}")
```

Output:-

```
Accuracy: 0.9850374064837906
Predicted AQI Category: ['Good']
```

#Code of K-means

```python
#K-means
import numpy as np
from sklearn.cluster import KMeans
data = np.array([
[44.7444, 44.2031],
[-5.29, -44.49],
[-11.2958, -41.9869],
[37.1667, 15.1833],
[53.0167, 20.8833],
[16.1005, -88.8074],
[26.8941, -82.0513],
[51.0761, 4.2803],
[44.05, 43.0667],
[21.95, 108.6167],
[52.3833, 6.2667],
[50.3892, 3.4858],
[37.3833, 14.7],
[33.5722, -112.0892],
[52.0667, 21.0167],
[-22.5128, -44.0008],
[9.7919, -74.7872],
[43.9333, 23.0833],
[56.45, 60.1833]
])
k = 3

kmeans = KMeans(n_clusters=k, random_state=0).fit(data)
labels = kmeans.labels_
centers = kmeans.cluster_centers_
```

```
for i in range(k):
cluster_points = data[labels == i]

print(f"Cluster {i+1}:")
for point in cluster_points:

print(f"Latitude: {point[0]}, Longitude: {point[1]}")
print()

print("Cluster Centers:")
for center in centers:

print(f"Latitude: {center[0]}, Longitude: {center[1]}")
```

```
Output:-
Cluster 1:
Latitude: 44.7444, Longitude: 44.2031
Latitude: 37.1667, Longitude: 15.1833
Latitude: 53.0167, Longitude: 20.8833
Latitude: 51.0761, Longitude: 4.2803
Latitude: 44.05, Longitude: 43.0667
Latitude: 52.3833, Longitude: 6.2667
Latitude: 50.3892, Longitude: 3.4858
Latitude: 37.3833, Longitude: 14.7
Latitude: 52.0667, Longitude: 21.0167
Latitude: 43.9333, Longitude: 23.0833
Latitude: 56.45, Longitude: 60.1833
Cluster 2:
Latitude: -5.29, Longitude: -44.49
Latitude: -11.2958, Longitude: -41.9869
Latitude: 16.1005, Longitude: -88.8074
Latitude: 26.8941, Longitude: -82.0513
Latitude: 33.5722, Longitude: -112.0892
Latitude: -22.5128, Longitude: -44.0008
Latitude: 9.7919, Longitude: -74.7872
Cluster 3:
Latitude: 21.95, Longitude: 108.6167
Cluster Centers:
Latitude: 47.514518181818175, Longitude: 23.304772727272727
Latitude: 6.751442857142859, Longitude: -69.74468571428571
Latitude: 21.95, Longitude: 108.6167
```