

LEVEL 2 - TASK 3: GEOGRAPHIC ANALYSIS

--3:1 Plot the locations of restaurants on a map using longitude and latitude coordinates.

-3:2 Identify any patterns or clusters of restaurants in specific areas.

3:1 PLOT THE LOCATIONS OF RESTAURANTS ON A MAP USING LONGITUDE AND LATITUDE COORDINATES.

```
#import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

```
pip install folium pandas
```

Collecting folium

Downloading folium-0.19.4-py2.py3-none-any.whl.metadata (3.8 kB)

Requirement already satisfied: pandas in c:\users\dimpi\anaconda3\lib\site-packages (2.2.2)

Collecting branca>=0.6.0 (from folium)

Downloading branca-0.8.1-py3-none-any.whl.metadata (1.5 kB)

Requirement already satisfied: jinja2>=2.9 in c:\users\dimpi\anaconda3\lib\site-packages (from folium) (3.1.4)

Requirement already satisfied: numpy in c:\users\dimpi\anaconda3\lib\site-packages (from folium) (1.26.4)

Requirement already satisfied: requests in c:\users\dimpi\anaconda3\lib\site-packages (from folium) (2.32.3)

Requirement already satisfied: xyzservices in c:\users\dimpi\anaconda3\lib\site-packages (from folium) (2022.9.0)

Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\dimpi\anaconda3\lib\site-packages (from pandas) (2.9.0.post0)

Requirement already satisfied: pytz>=2020.1 in c:\users\dimpi\anaconda3\lib\site-packages (from pandas) (2024.1)

Requirement already satisfied: tzdata>=2022.7 in c:\users\dimpi\anaconda3\lib\site-packages (from pandas) (2023.3)

Requirement already satisfied: MarkupSafe>=2.0 in c:\users\dimpi\anaconda3\lib\site-packages (from jinja2->folium) (2.1.3)

Requirement already satisfied: six>=1.5 in c:\users\dimpi\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)

Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\dimpi\anaconda3\lib\site-packages (from requests->folium) (3.3.2)

Requirement already satisfied: idna<4,>=2.5 in c:\users\dimpi\anaconda3\lib\site-packages (from requests->folium) (3.7)

Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\dimpi\anaconda3\lib\site-packages (from requests->folium) (2.2.3)

Requirement already satisfied: certifi>=2017.4.17 in c:\users\dimpi\anaconda3\lib\site-packages (from requests->folium) (2024.8.30)

Downloading folium-0.19.4-py2.py3-none-any.whl (110 kB)
 Downloading branca-0.8.1-py3-none-any.whl (26 kB)
 Installing collected packages: branca, folium
 Successfully installed branca-0.8.1 folium-0.19.4
 Note: you may need to restart the kernel to use updated packages.

```

#import data
dataset= pd.read_csv("dataset.csv")
  
```

```

#check data
dataset.head(10)
  
```

	Restaurant ID	Restaurant Name	Country
Code \			
0	6317637	Le Petit Souffle	
162			
1	6304287	Izakaya Kikufuji	
162			
2	6300002	Heat - Edsa Shangri-La	
162			
3	6318506	Ooma	
162			
4	6314302	Sambo Kojin	
162			
5	18189371	Din Tai Fung	
162			
6	6300781	Buffet 101	
162			
7	6301290	Vikings	
162			
8	6300010	Spiral - Sofitel Philippine Plaza Manila	
162			
9	6314987	Locavore	
162			

	City	Address
\		
0	Makati City	Third Floor, Century City Mall, Kalayaan Avenu...
1	Makati City	Little Tokyo, 2277 Chino Roces Avenue, Legaspi...
2	Mandaluyong City	Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal...
3	Mandaluyong City	Third Floor, Mega Fashion Hall, SM Megamall, O...
4	Mandaluyong City	Third Floor, Mega Atrium, SM Megamall, Ortigas...
5	Mandaluyong City	Ground Floor, Mega Fashion Hall, SM Megamall, ...
6	Pasay City	Building K, SM By The Bay, Sunset Boulevard, M...

7	Pasay City	Building B, By The Bay, Seaside Boulevard, Mal...
8	Pasay City	Plaza Level, Sofitel Philippine Plaza Manila, ...
9	Pasig City	Brixton Technology Center, 10 Brixton Street, ...

	Locality \
0	Century City Mall, Poblacion, Makati City
1	Little Tokyo, Legaspi Village, Makati City
2	Edsa Shangri-La, Ortigas, Mandaluyong City
3	SM Megamall, Ortigas, Mandaluyong City
4	SM Megamall, Ortigas, Mandaluyong City
5	SM Megamall, Ortigas, Mandaluyong City
6	SM by the Bay, Mall of Asia Complex, Pasay City
7	SM by the Bay, Mall of Asia Complex, Pasay City
8	Sofitel Philippine Plaza Manila, Pasay City
9	Kapitolyo

	Locality Verbose	Longitude
Latitude \		
0	Century City Mall, Poblacion, Makati City, Mak...	121.027535
14.565443		
1	Little Tokyo, Legaspi Village, Makati City, Ma...	121.014101
14.553708		
2	Edsa Shangri-La, Ortigas, Mandaluyong City, Ma...	121.056831
14.581404		
3	SM Megamall, Ortigas, Mandaluyong City, Mandal...	121.056475
14.585318		
4	SM Megamall, Ortigas, Mandaluyong City, Mandal...	121.057508
14.584450		
5	SM Megamall, Ortigas, Mandaluyong City, Mandal...	121.056314
14.583764		
6	SM by the Bay, Mall of Asia Complex, Pasay Cit...	120.979667
14.531333		
7	SM by the Bay, Mall of Asia Complex, Pasay Cit...	120.979333
14.540000		
8	Sofitel Philippine Plaza Manila, Pasay City, P...	120.980090
14.552990		
9	Kapitolyo, Pasig City	121.056532
14.572041		

	Cuisines ...	Currency \
0	French, Japanese, Desserts ...	Botswana Pula(P)
1	Japanese ...	Botswana Pula(P)
2	Seafood, Asian, Filipino, Indian ...	Botswana Pula(P)
3	Japanese, Sushi ...	Botswana Pula(P)
4	Japanese, Korean ...	Botswana Pula(P)
5	Chinese ...	Botswana Pula(P)
6	Asian, European ...	Botswana Pula(P)

```

7 Seafood, Filipino, Asian, European ... Botswana Pula(P)
8           European, Asian, Indian ... Botswana Pula(P)
9           Filipino ... Botswana Pula(P)

```

```

Has Table booking Has Online delivery Is delivering now \
0           Yes           No           No
1           Yes           No           No
2           Yes           No           No
3           No            No           No
4           Yes           No           No
5           No            No           No
6           Yes           No           No
7           Yes           No           No
8           Yes           No           No
9           Yes           No           No

```

```

Switch to order menu Price range Aggregate rating Rating color \
0           No            3           4.8       Dark Green
1           No            3           4.5       Dark Green
2           No            4           4.4         Green
3           No            4           4.9       Dark Green
4           No            4           4.8       Dark Green
5           No            3           4.4         Green
6           No            4           4.0         Green
7           No            4           4.2         Green
8           No            4           4.9       Dark Green
9           No            3           4.8       Dark Green

```

```

Rating text Votes
0   Excellent   314
1   Excellent   591
2   Very Good   270
3   Excellent   365
4   Excellent   229
5   Very Good   336
6   Very Good   520
7   Very Good   677
8   Excellent   621
9   Excellent   532

```

```
[10 rows x 21 columns]
```

```

#check database shape
dataset.shape

```

```
(9551, 21)
```

```

#check dataset information
dataset.info()

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9551 entries, 0 to 9550
Data columns (total 21 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Restaurant ID                        9551 non-null   int64
1   Restaurant Name                      9551 non-null   object
2   Country Code                        9551 non-null   int64
3   City                                9551 non-null   object
4   Address                             9551 non-null   object
5   Locality                            9551 non-null   object
6   Locality Verbose                    9551 non-null   object
7   Longitude                           9551 non-null   float64
8   Latitude                            9551 non-null   float64
9   Cuisines                            9542 non-null   object
10  Average Cost for two                 9551 non-null   int64
11  Currency                            9551 non-null   object
12  Has Table booking                   9551 non-null   object
13  Has Online delivery                 9551 non-null   object
14  Is delivering now                   9551 non-null   object
15  Switch to order menu                9551 non-null   object
16  Price range                         9551 non-null   int64
17  Aggregate rating                    9551 non-null   float64
18  Rating color                        9551 non-null   object
19  Rating text                         9551 non-null   object
20  Votes                              9551 non-null   int64
dtypes: float64(3), int64(5), object(13)
memory usage: 1.5+ MB

```

```
#check dataset column names
```

```
dataset.columns
```

```

Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City',
      'Address',
      'Locality', 'Locality Verbose', 'Longitude', 'Latitude',
      'Cuisines',
      'Average Cost for two', 'Currency', 'Has Table booking',
      'Has Online delivery', 'Is delivering now', 'Switch to order
menu',
      'Price range', 'Aggregate rating', 'Rating color', 'Rating
text',
      'Votes'],
      dtype='object')

```

Data Preprocessing

```
#check for null values
```

```
pd.isnull(dataset).sum()
```

Restaurant ID	0
Restaurant Name	0
Country Code	0
City	0
Address	0
Locality	0
Locality Verbose	0
Longitude	0
Latitude	0
Cuisines	9
Average Cost for two	0
Currency	0
Has Table booking	0
Has Online delivery	0
Is delivering now	0
Switch to order menu	0
Price range	0
Aggregate rating	0
Rating color	0
Rating text	0
Votes	0

dtype: int64

#drop all null values

dataset.dropna(inplace=True)

#check database

dataset.shape

(9542, 21)

dataset.info()

<class 'pandas.core.frame.DataFrame'>

Index: 9542 entries, 0 to 9550

Data columns (total 21 columns):

#	Column	Non-Null Count	Dtype
0	Restaurant ID	9542 non-null	int64
1	Restaurant Name	9542 non-null	object
2	Country Code	9542 non-null	int64
3	City	9542 non-null	object
4	Address	9542 non-null	object
5	Locality	9542 non-null	object
6	Locality Verbose	9542 non-null	object
7	Longitude	9542 non-null	float64
8	Latitude	9542 non-null	float64
9	Cuisines	9542 non-null	object
10	Average Cost for two	9542 non-null	int64
11	Currency	9542 non-null	object

```

12 Has Table booking      9542 non-null object
13 Has Online delivery    9542 non-null object
14 Is delivering now      9542 non-null object
15 Switch to order menu   9542 non-null object
16 Price range            9542 non-null int64
17 Aggregate rating       9542 non-null float64
18 Rating color           9542 non-null object
19 Rating text            9542 non-null object
20 Votes                  9542 non-null int64

```

```

dtypes: float64(3), int64(5), object(13)
memory usage: 1.6+ MB

```

#check description of data

```
dataset[['Average Cost for two', 'Price range', 'Aggregate rating',
'Votes']].describe()
```

	Average Cost for two	Price range	Aggregate rating
Votes			
count	9542.000000	9542.000000	9542.000000
mean	1200.326137	1.804968	2.665238
std	156.772060	0.905563	1.516588
min	430.203324	0.000000	1.000000
25%	0.000000	1.000000	2.500000
50%	250.000000	1.000000	2.500000
75%	400.000000	2.000000	3.200000
max	31.000000	2.000000	3.700000
	700.000000	2.000000	3.700000
	130.000000	4.000000	4.900000
	800000.000000	4.000000	4.900000
	10934.000000		

```

import folium
from folium.plugins import MarkerCluster

```

#create a base map centered around the average coordinates

```

map_center= [dataset["Latitude"].mean(), dataset["Longitude"].mean()]
restaurant_map = folium.Map(location=map_center, zoom_start=12)

```

#add a marker cluster for better visualization

```
marker_cluster= MarkerCluster().add_to(restaurant_map)
```

#add restaurant location to the map

```

for _,row in dataset.iterrows():          #'_' is used to ignore the
index                                     index
    folium.Marker(
        location=[row["Latitude"], row["Longitude"]],

```

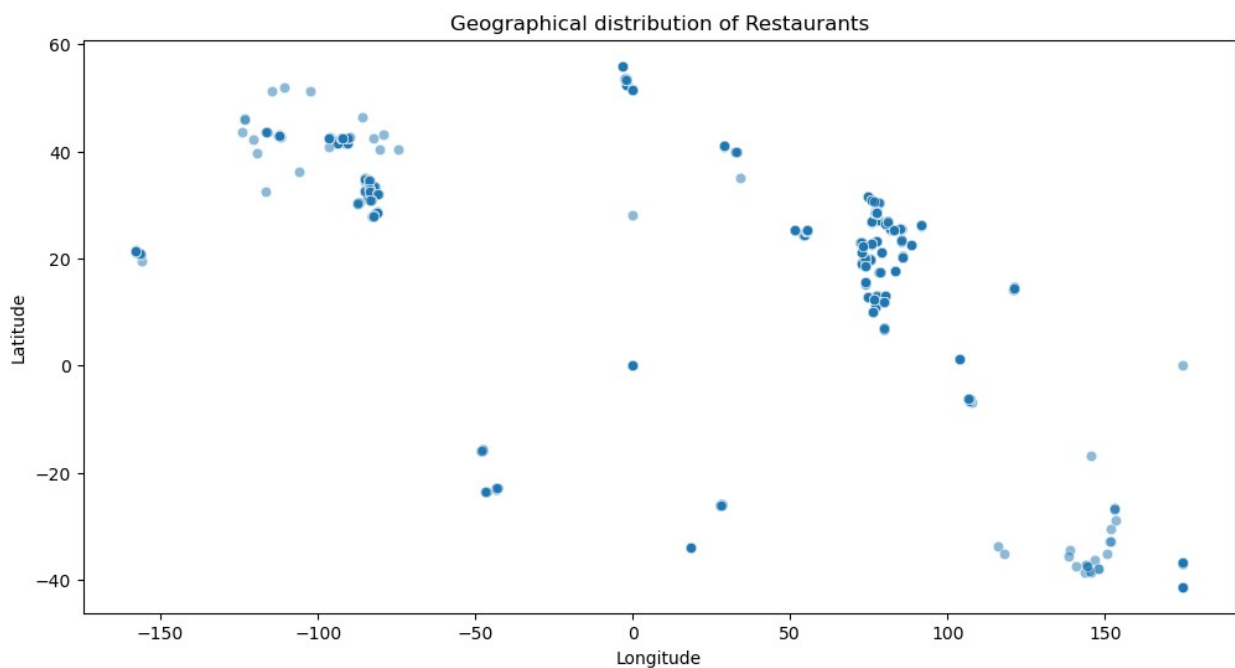
```

        popup= row["Restaurant Name"],
        icon=folium.Icon(color="blue", icon="cutlery",
prefix="fa"),
        ).add_to(marker_cluster)

#save the map to a file and display it
restaurant_map.save("restaurant_map.html")

#plot restaurant locations on a scatter ployt
plt.figure(figsize=(12,6))
sns.scatterplot(x=dataset["Longitude"],
y=dataset["Latitude"],alpha=0.5)
plt.xlabel("Longitude")
plt.ylabel("Latitude")
plt.title("Geographical distribution of Restaurants")
plt.show()

```



3:2 IDENTIFY ANY PATTERNS OR CLUSTERS OF RESTAURANTS IN SPECIFIC AREAS.

```

from sklearn.cluster import KMeans

#remove rows with missing latitude and longitude values
dataset_clean= dataset.dropna(subset=["Latitude","Longitude"])

#extract coordinates
coords= dataset_clean[["Latitude","Longitude"]].values

#determine the optimal number of clusters using the elbow method
inertia=[]

```



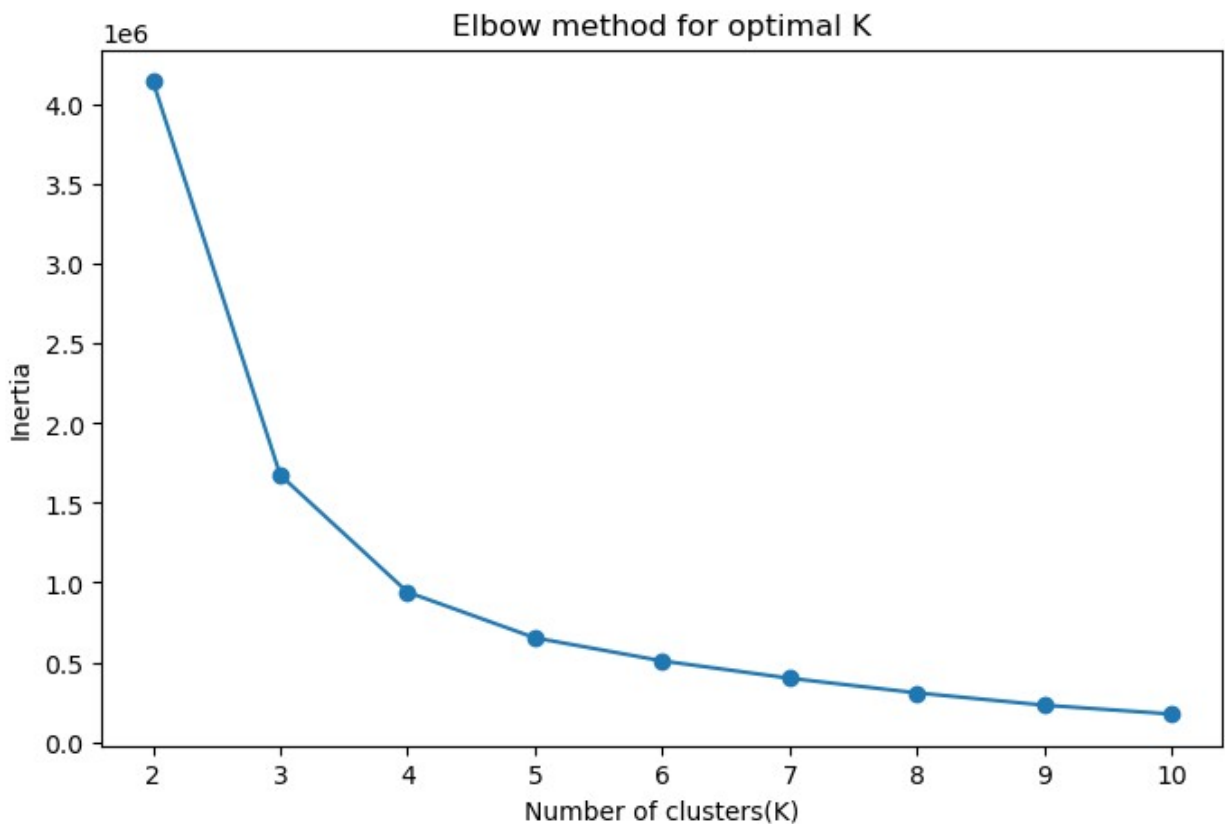
```

K_range= range(2,11)  #testing k from 2 to 10

for k in K_range:
    kmeans= KMeans(n_clusters=k, random_state=42, n_init=10)
    kmeans.fit(coords)
    inertia.append(kmeans.inertia_)

#plot the elbow curve
plt.figure(figsize=(8,5))
plt.plot(K_range,inertia,marker="o", linestyle="-")
plt.xlabel("Number of clusters(K)")
plt.ylabel("Inertia")
plt.title("Elbow method for optimal K")
plt.show()

```



```

# Based on the Elbow plot, selecting an optimal K (assuming the
"elbow" occurs around 4 or 5)
optimal_k = 5

# Apply K-Means clustering
kmeans = KMeans(n_clusters=optimal_k, random_state=42, n_init=10)
dataset_clean["Cluster"] = kmeans.fit_predict(coords)

# Plot clusters

```

```

plt.figure(figsize=(12, 6))
sns.scatterplot(
    x=dataset_clean["Longitude"],
    y=dataset_clean["Latitude"],
    hue=dataset_clean["Cluster"],
    palette="viridis",
    alpha=0.6
)

plt.scatter(
    kmeans.cluster_centers_[ :, 1],
    kmeans.cluster_centers_[ :, 0],
    c="red", marker="X", s=200, label="Centroids"
)

plt.xlabel("Longitude")
plt.ylabel("Latitude")
plt.title("Restaurant Clusters Based on Location")
plt.legend()
plt.show()

```

