Capstone Project

The Battle of Neighborhoods in Beijing: Restaurants

1. **Introduction**

   As one of the most popular countries to travel, China has a long history that creates multiple cultures in architecture, languages, and food. Beijing, as the capital city of China, has 16 districts with more than 20 million population. For such a huge city, it has always been difficult for travelers with limited time to choose better places to visit and try some local restaurants than just for tourism. Therefore, in this capstone project, I will use the Foursquares Location data and data science to analyze restaurants in the main districts which have more than 1,000,000 population so that it could help visitors to make better decision.

2. **Data Resource**

   The data using in this project:

   - **All the counties in Beijing from Wikipedia.**

   Link:

   https://en.wikipedia.org/wiki/List_of_counties_in_China

   - **Restaurants in each neighborhood of Beijing in the main districts.**

   Source:

   Foursquare APIs

3. **Methodology**

   3.1 Data Preparation

The first step in data preparation is to scrap all the counties in China from a Wikipedia Website and make it as a data frame in python by using the Pandas.

```
source = requests.get("https://en.wikipedia.org/wiki/List_of_counties_in_China")
soup = BeautifulSoup(source, "html.parser")

table = soup.find("table",{'class':"wikitable"})
# Dataframe with 3 columns

df = pd.read_html(str(table))
df = pd.DataFrame(df[0])
df.head()
```

	Name	Prefecture	Province	Type	Population Census 2010
0	Yaohai	Hefei	Anhui	District	902830
1	Luyang	Hefei	Anhui	District	609239
2	Shushan	Hefei	Anhui	District	1022321
3	Baohe	Hefei	Anhui	District	817686
4	Changfeng	Hefei	Anhui	County	629535

In this data frame, it contains the Name, Prefecture, Province, Type, and the Population Census 2010 columns for all the counties in China. For this analysis, I only concentrate on the main districts in Beijing. Therefore, I add some filter to get the only data needed, as say that I select the districts in Beijing which has a population larger than 1,000,000 in 2010 Census.

The data selected is showed below:

As it shows, there are 7 main districts in Beijing selected.

```
df = df[df['Province'] == 'Beijing']
df = df.drop(df.index[[-1,-2]])
df['Population Census 2010']=pd.to_numeric(df['Population Census 2010'] )
df = df[df['Population Census 2010'] > 1000000]
df
```

	Name	Prefecture	Province	Type	Population Census 2010
107	Xicheng	Directly administered	Beijing	District	1243000
108	Chaoyang	Directly administered	Beijing	District	3545000
109	Haidian	Directly administered	Beijing	District	3281000
110	Fengtai	Directly administered	Beijing	District	2112000
112	Tongzhou	Directly administered	Beijing	District	1184000
114	Changping	Directly administered	Beijing	District	1661000
115	Daxing	Directly administered	Beijing	District	1365000

In the next step, the main goal is to add the coordinates for these 7 main districts. An CSV file contains the latitudes and longitudes of all the 7 main districts is upload as a data frame in python.

```
df_geo_coor = pd.read_csv('Coor.csv')
df_geo_coor
```

	District	Latitude	Longitude
0	Xicheng	39.9123	116.3659
1	Chaoyang	39.9215	116.4431
2	Haidian	39.9600	116.2983
3	Fengtai	39.8584	116.2871
4	Tongzhou	39.9099	116.6564
5	Changping	40.2207	116.2312
6	Daxing	39.7269	116.3414

Then, add the coordinates to the original data frame and drop the columns that will not be used. As shown below:

```
df = pd.merge(df, df_geo_coor, how='left', left_on = 'Name', right_on = 'District')
# remove the "District" column
df.drop("District", axis=1, inplace=True)
df
```

	Name	Prefecture	Province	Type	Population Census 2010	Latitude	Longitude
0	Xicheng	Directly administered	Beijing	District	1243000	39.9123	116.3659
1	Chaoyang	Directly administered	Beijing	District	3545000	39.9215	116.4431
2	Haidian	Directly administered	Beijing	District	3281000	39.9600	116.2983
3	Fengtai	Directly administered	Beijing	District	2112000	39.8584	116.2871
4	Tongzhou	Directly administered	Beijing	District	1184000	39.9099	116.6564
5	Changping	Directly administered	Beijing	District	1661000	40.2207	116.2312
6	Daxing	Directly administered	Beijing	District	1365000	39.7269	116.3414

To visualize the geographic details of the 7 main districts of Beijing, I use the Folium Package in python to create a map by using the latitudes and longitudes in the data frame.

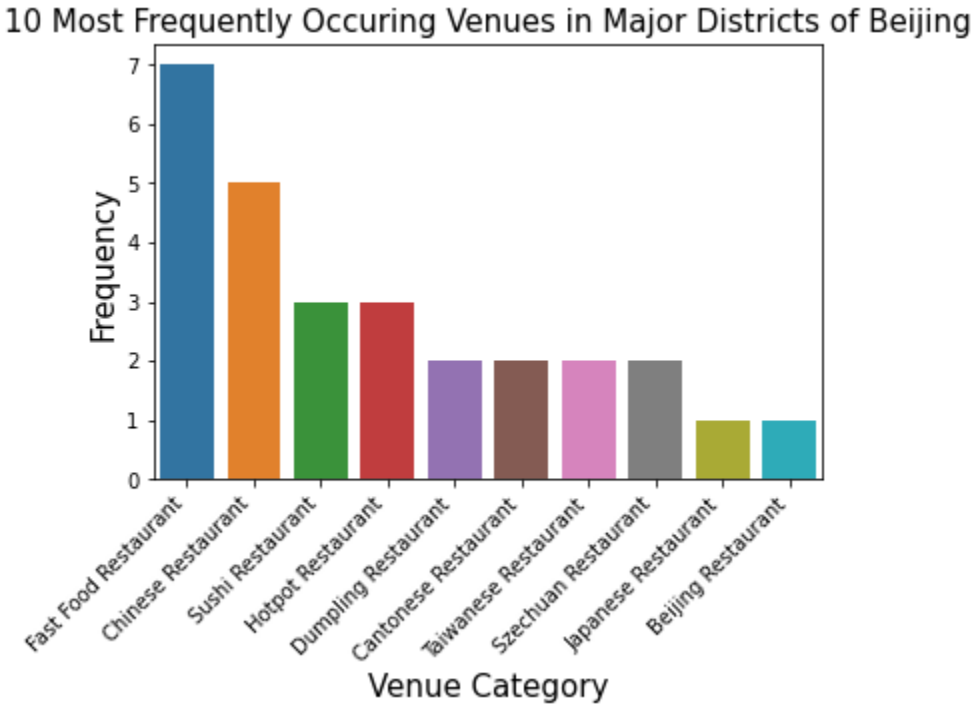
```
# create map of Beijing using latitude and longitude values
map_Beijing = folium.Map(location=[latitude, longitude], zoom_start=10)

 # add markers to map
for lat, lng, label in zip(df['Latitude'], df['Longitude'], df['Name']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_Beijing)

map_Beijing
```

3.2 Exploratory Data Analysis (EDA)

In this section, I use the Foursquare Location data to explore the neighborhoods of the 7 main districts in Beijing. Especially, it will be focus on the restaurant's aspects.

```
Neighborhood
Changping     1
Chaoyang     16
Fengtai       3
Tongzhou      2
Xicheng      10
Name: Venue Category, dtype: int64
```

	Venue_Category	Frequency
0	Fast Food Restaurant	7
1	Chinese Restaurant	5
2	Sushi Restaurant	3
3	Hotpot Restaurant	3
4	Dumpling Restaurant	2
5	Cantonese Restaurant	2
6	Taiwanese Restaurant	2
7	Szechuan Restaurant	2
8	Japanese Restaurant	1
9	Beijing Restaurant	1



As shown above, the fast food restaurants is on the top of the list.

Next main section is to analyze each districts with top 5 types of restaruants in Beijing.

First, create a new data frame by using the one-hot encoding for the rastaurant category.

```
# one hot encoding
Beijing_onehot = pd.get_dummies(Beijing_Venues_only_restaurant[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
Beijing_onehot['Neighborhood'] = Beijing_Venues_only_restaurant['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [Beijing_onehot.columns[-1]] + list(Beijing_onehot.columns[:-1])
Beijing_onehot = Beijing_onehot[fixed_columns]

Beijing_onehot.head()
```

	Neighborhood	Asian Restaurant	Beijing Restaurant	Cantonese Restaurant	Chinese Restaurant	Comfort Food Restaurant	Dumpling Restaurant	Fast Food Restaurant	Hotpot Restaurant	Italian Restaurant	Japanese Restaurant	Korean Restaurant	N Americ Restaur
1	Xicheng	0	0	0	0	0	1	0	0	0	0	0	
2	Xicheng	0	0	0	0	0	0	0	1	0	0	0	
3	Xicheng	0	0	0	0	0	0	0	0	0	0	0	
4	Xicheng	0	0	0	0	0	0	0	0	0	0	0	
5	Xicheng	0	0	0	0	0	0	0	0	0	0	0	

Secondly, group rows by neighborhood and taking the mean of frequency of occurrence in each category by using pandas.

```
# group rows by neighborhood and by taking the mean of the frequency of occurrence of each category
Beijing_grouped = Beijing_onehot.groupby('Neighborhood').mean().reset_index()
Beijing_grouped
```

	Neighborhood	Asian Restaurant	Beijing Restaurant	Cantonese Restaurant	Chinese Restaurant	Comfort Food Restaurant	Dumpling Restaurant	Fast Food Restaurant	Hotpot Restaurant	Italian Restaurant	Japanese Restaurant	Korean Restaurant	N Americ Restaur
0	Changping	0.0	0.0000	0.0000	0.000000	0.000000	0.0000	1.0000	0.0	0.0000	0.000	0.0000	0.0(
1	Chaoyang	0.0	0.0625	0.0625	0.125000	0.000000	0.0625	0.1875	0.0	0.0625	0.125	0.0625	0.0(
2	Fengtai	0.0	0.0000	0.0000	0.666667	0.333333	0.0000	0.0000	0.0	0.0000	0.000	0.0000	0.0(
3	Tongzhou	0.0	0.0000	0.0000	0.500000	0.000000	0.0000	0.5000	0.0	0.0000	0.000	0.0000	0.0(
4	Xicheng	0.1	0.0000	0.1000	0.000000	0.000000	0.1000	0.2000	0.1	0.0000	0.000	0.0000	0.0(

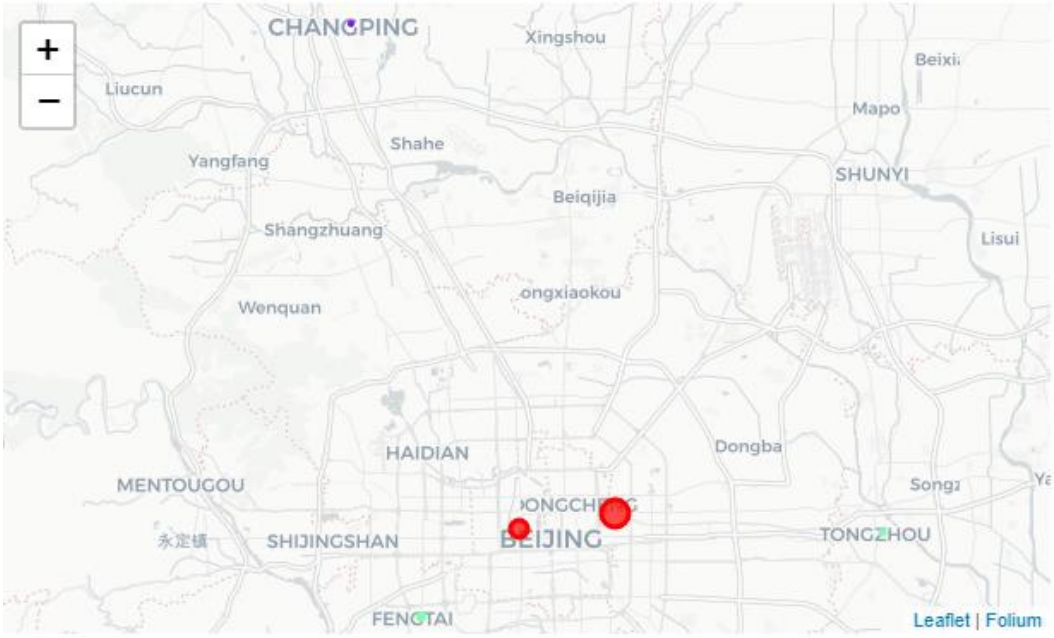
Thirdly, print the result for each district with top 5 common restaurants.

```
----Changping----
                venue  freq
0  Fast Food Restaurant   1.0
1      Asian Restaurant   0.0
2    Beijing Restaurant   0.0
3  Cantonese Restaurant   0.0
4    Chinese Restaurant   0.0

----Chaoyang----
                venue  freq
0  Fast Food Restaurant  0.19
1    Chinese Restaurant  0.12
2   Japanese Restaurant  0.12
3      Sushi Restaurant  0.12
4    Beijing Restaurant  0.06

----Fengtai----
                venue  freq
0    Chinese Restaurant  0.67
1  Comfort Food Restaurant  0.33
2      Asian Restaurant  0.00
3    Beijing Restaurant  0.00
4  Cantonese Restaurant  0.00

----Tongzhou----
                venue  freq
0    Chinese Restaurant   0.5
1  Fast Food Restaurant   0.5
2      Asian Restaurant   0.0
3    Beijing Restaurant   0.0
4  Cantonese Restaurant   0.0

----Xicheng----
                venue  freq
0  Fast Food Restaurant   0.2
1   Szechuan Restaurant   0.2
2      Asian Restaurant   0.1
3  Cantonese Restaurant   0.1
4   Dumpling Restaurant   0.1
```

In the Finaly step, cluster the data by using K-Means and visilize the result by using Folium Package.

```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels2', kmeans.labels_)

Beijing_merged = df.drop([2,6])

Beijing_merged.rename(columns={'Name':'Neighborhood'}, inplace=True)

# merge Beijing_grouped with data to add latitude/longitude for each neighborhood
Beijing_merged = Beijing_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

Beijing_merged # check columns
```

	Neighborhood	Prefecture	Province	Type	Population Census 2010	Latitude	Longitude	Cluster Labels2	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue
0	Xicheng	Directly administered	Beijing	District	1243000	39.9123	116.3659	0	Szechuan Restaurant	Fast Food Restaurant	Taiwanese Restaurant	Ramen Restaurant	Hotpot Restaurant	Dumpling Restaurant
1	Chaoyang	Directly administered	Beijing	District	3545000	39.9215	116.4431	0	Fast Food Restaurant	Sushi Restaurant	Japanese Restaurant	Chinese Restaurant	Ramen Restaurant	New American Restaurant
3	Fengtai	Directly administered	Beijing	District	2112000	39.8584	116.2871	2	Chinese Restaurant	Comfort Food Restaurant	Taiwanese Restaurant	Szechuan Restaurant	Sushi Restaurant	Ramen Restaurant
4	Tongzhou	Directly administered	Beijing	District	1184000	39.9099	116.6564	2	Fast Food Restaurant	Chinese Restaurant	Taiwanese Restaurant	Szechuan Restaurant	Sushi Restaurant	Ramen Restaurant
5	Changping	Directly administered	Beijing	District	1661000	40.2207	116.2312	1	Fast Food Restaurant	Taiwanese Restaurant	Szechuan Restaurant	Sushi Restaurant	Ramen Restaurant	New American Restaurant



4. Results

- The most restaurant in the main districts of Beijing is the Fast-Food restaurants.
- Chaoyang and Xicheng Have the greatest number of restaurants.

- There is no efficient information of restaurants in Haidian and Daxing.

  This report analyzes the overview distribution of restaurants in the main districts in Beijing by using the Foursquare Location Data and K-Means algorithm.

5. Conclusion

This is a simple example that using the data to help people solve real-life problems and make better decisions. In this report, we use the Wikipedia data and Foursquare API to cluster the restaurants in the main districts of Beijing. As the result, Chaoyang and Xicheng have the greatest number of restaurants so that for traveler who are seeking for various food, they will be a good choice.

Moreover, this model could be improved by inputting more data like transportation, price, and scores in social media, or using different algorithms. To conclude, data science could help people to make better decision.