Capstone Project - The Battle of Neighbourhoods

1. BUSINESS PROBLEM

One of the client/owner wants to open a branch of his restaurant in a new venue Madrid. Madrid is the capital and largest city of Spain, one of the busiest cities in Europe. More than 3 million people live in the city.

Statistic shows that annual number of international tourists visiting the Community of Madrid are increasing every year, on an average 800,000 visiting each month.

Client has some branches in European countries, after having successfully opened a branch in Valencia city in El Carme neighbourhood, he is planning to restaurant in Madrid.

Problem Statement :

Analyzing data by applying tools and methodologies using data science need to find out which is the neighbourhood from Madrid will be having similar characteristics like El Carme in Valencia city to open a new restaurant.

2. DATA

The data to be used for this project comes from different sites / locations:

- **Foursquare** is a social location service that allows users to explore the world around them, which provides information on different types of entertainment, drinking and dining venues.

  The Foursquare API allows application developers to interact with the Foursquare platform, The API itself is a RESTful set of addresses to which you can send requests and find information related to the venues, such as location, overall category, reviews and tips.

- Madrid Neighborhood Names and geographic coordinates. Available on https://datos.madrid.es/, this is used to obtain the neighborhood location information from the city.

- Valencia City Neighborhood Names and geographic coordinates. Data available on http://mapas.valencia.es/lanzadera/opendata/Barrios/SHAPE

- Madrid census data, were we can get the population and income statistics, available in http://www-2.munimadrid.es/CSE6/jsps/menuBancoDatos.jsp

Below the details of how we will use each data source during this project.

2.1. Foursquare API data

For this project we will use the Foursquare Places API. One of the features of this API is to provide a list of venues within a specific location, based on the Lat/Lon coordinates and a radius.
In order to obtain a list of venues within a specified area, we use the "explore" endpoint from the API. By passing the proper parameters via an HTTP request to the *explore* endpoint, we get a JSON object.

The *location* object contains the coordinates of each venue, which will be used to associate it with its respective neighborhood.
The *categories* array will be used to categorize the neighborhood. Basically, we will count how many venues from all available categories are found on each neighborhood, and then use that information to compare neighborhoods from Madrid with El Carme in Valencia.

2.2. Madrid Neighborhoods
The Madrid city government has made available to the public a series of datasets with information of interest. We will be using the "Divisiones administrativas: distritos, barrios y divisiones históricas" dataset, available in the following URL: https://datos.madrid.es/egob/catalogo/200078-10-distritos-barrios.zip.
The data insinde the .zip file is in ESRI format. To convert this to a dataframe that we can use, *geopandas* python library.

2.3. Valencia City Neighborhoods
Valencia City Neighborhood Names and geographic coordinates. Data available on http://mapas.valencia.es/lanzadera/opendata/Barrios/SHAPE.
This data is also available in ESRI format.

2.4. Madrid Census data
To complement our analysis we will be using the statistics of the population and average income per neighborhood in madrid. This data is available in the municipality data bank, http://www-2.munimadrid.es/CSE6/jsps/menuBancoDatos.jsp

3. METHODOLOGY

- ### Neighborhood basic information and census data

During the data preprocessing stage, we prepare the data to be used during the machine learning process. The data structure for the neighborhood information is different between Madrid and Valencia, so we need to adapt both of them.

Madrid Neighborhood Data

```
#Import Neighborhoods geodata
madrid_neighborhoods = gpd.read_file("D:/python_examples/code/Data Science/Coursera_Capstone/data/neighborhoods/BARRIOS.
```

Lets get some basic information on the imported data

```
madrid_neighborhoods.head(3)
```

	OBJECTID	geodb_oid	CODDIS	NOMDIS	CODBAR	CODDISTRIT	CODBARRIO	NOMBRE	ORIG_FID	geometry
0	108	108	17	Villaverde	172	17	17-2	San Cristobal	107	POLYGON ((441930.868000005 4466853.1887, 4419...
1	109	109	17	Villaverde	173	17	17-3	Butarque	108	POLYGON ((444144.8566044134 4464473.210504748,...
2	111	111	17	Villaverde	175	17	17-5	Los Angeles	110	POLYGON ((441147.7280000008 4466374.483400001,...

Valencia Neighborhood Data

```
# Import neighborhoods
val_neighborhoods = gpd.read_file("http://mapas.valencia.es/lanzadera/opendata/Barrios/SHAPE")
```

Lets get some basic information on the imported data

```
val_neighborhoods.head(3)
```

	codbarrio	nombre	coddistbar	coddistrit	geometry
0	1	BENIFARAIG	171	17	POLYGON ((725499.03 4378693.39, 725477.797 437...
1	1	BENICALAP	161	16	POLYGON ((725164.733 4375392.58, 725187.044 43...
2	2	TORREFIEL	152	15	POLYGON ((726040.348 4375385.446, 725995.041 4...

Information about both cities after concatenation is :

```
#Concatenate two dataframes
neighborhoods = pd.concat([madrid_neighborhoods,val_neighborhoods])
```

```
#Check random neighborhoods Madrid
neighborhoods[neighborhoods['City']=='Madrid'].head(3)
```

	District	Neighborhood	Longitude	Latitude	City
0	VILLAVERDE	SAN CRISTOBAL	-3.688372	40.340888	Madrid
1	VILLAVERDE	BUTARQUE	-3.676254	40.337115	Madrid
2	VILLAVERDE	LOS ANGELES	-3.699137	40.355790	Madrid

```
#Check random neighborhoods Valencia
neighborhoods[neighborhoods['City']=='Valencia'].head(3)
```

	District	Neighborhood	Longitude	Latitude	City
0	17	BENIFARAIG	-0.384621	39.525644	Valencia
1	16	BENICALAP	-0.391002	39.493006	Valencia
2	15	TORREFIEL	-0.376932	39.495198	Valencia

Creating dataset that contains the census data per neighborhood.

```
madrid_income = pd.read_excel('data/income-madrid/income-madrid.xls')
madrid_income.head()
```

	District	Neighborhood	Average Income
0	CENTRO	PALACIO	34675.85
1	CENTRO	EMBAJADORES	25999.83
2	CENTRO	CORTES	34952.68
3	CENTRO	JUSTICIA	40314.88
4	CENTRO	UNIVERSIDAD	30701.65

Population per neighborhood for madrid city

z.munimadrid.es/CSEv/jsps/menuBancoDatos.jsp

```
madrid_population = pd.read_excel('data/population-madrid/population-madrid.xls', skipfooter=4, skiprows=4)
madrid_population.head()
```

	Distrito	Barrio	Edad	Total
0	CENTRO	PALACIO	Total	22984
1	CENTRO	EMBAJADORES	Total	45433
2	CENTRO	CORTES	Total	10525
3	CENTRO	JUSTICIA	Total	17205
4	CENTRO	UNIVERSIDAD	Total	31809

- **Using Foursquare API data**

  using explore endpoint to get the dataset for top 100 venues within 500 mts from the center of each neighbourhood.

	Neighborhood	District	City	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	SAN CRISTOBAL	VILLAVERDE	Madrid	40.340888	-3.688372	Cercanías San Cristóbal de Los Ángeles	40.341710	-3.683878	Train Station
1	SAN CRISTOBAL	VILLAVERDE	Madrid	40.340888	-3.688372	Igreen Aire Acondicionado y Climatización	40.341581	-3.686213	Furniture / Home Store
2	SAN CRISTOBAL	VILLAVERDE	Madrid	40.340888	-3.688372	Bar Vietnam	40.341090	-3.686568	Snack Place
3	SAN CRISTOBAL	VILLAVERDE	Madrid	40.340888	-3.688372	El Rincón de Peri	40.342427	-3.691998	Breakfast Spot
4	BUTARQUE	VILLAVERDE	Madrid	40.337115	-3.676254	Mercadona	40.340165	-3.675179	Grocery Store

- **Exploratory Data Analysis**

  Below displays the number of neighbourhoods we are working with on each city.
  We can see number if neighborhoods in Madrid is more compared with Valencia

```
print('The number of neighborhoods in Madrid is: {}'.format(madrid_neighborhoods['Neighborhood'].nunique()))
print('The number of districts in Madrid is: {}'.format(madrid_neighborhoods['District'].nunique()))
```

```
The number of neighborhoods in Madrid is: 131
The number of districts in Madrid is: 21
```

```
print('The number of neighborhoods in Valencia is: {}'.format(val_neighborhoods['Neighborhood'].nunique()))
print('The number of districts in Valencia is: {}'.format(val_neighborhoods['District'].nunique()))
```

```
The number of neighborhoods in Valencia is: 88
The number of districts in Valencia is: 19
```

Comparing the distribution of types of venues found on each city:
We can see that Spanish Restaurants are more in both the cities.

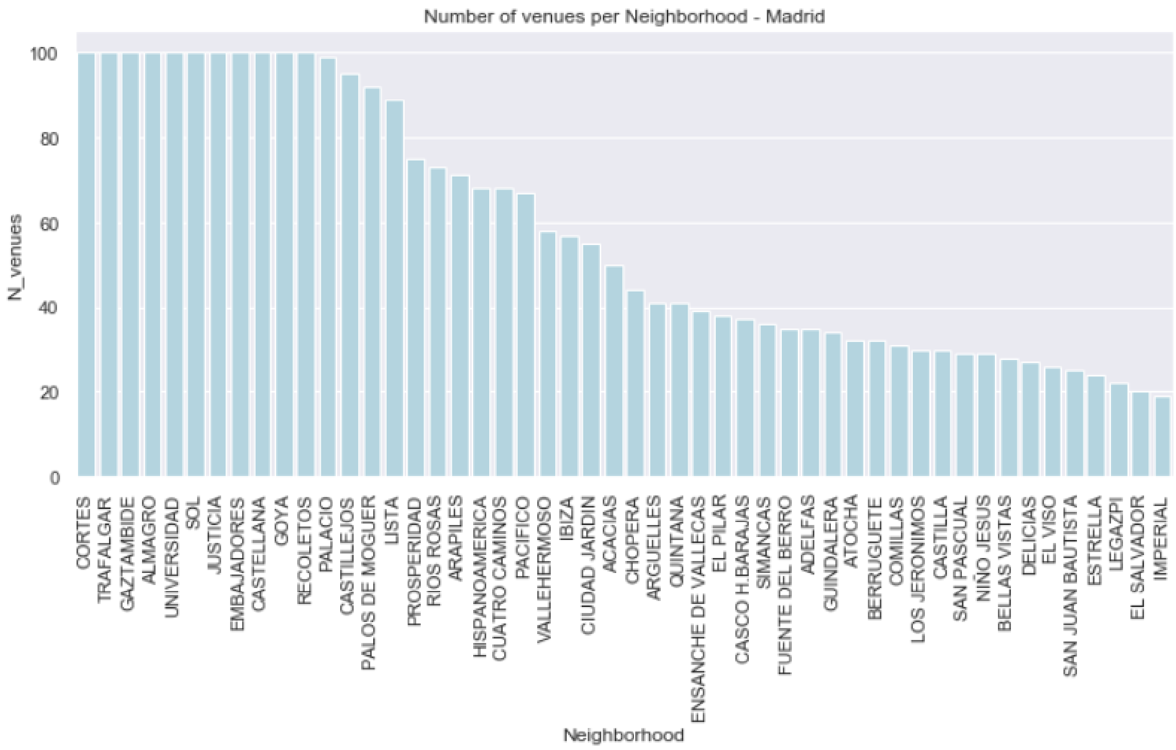
```python
# Count the number of locations per Venue Category in Madrid
venues[venues['City']=='Madrid'].groupby('Venue Category').count()['Neighborhood'].sort_values(ascending=False).head(10)
```

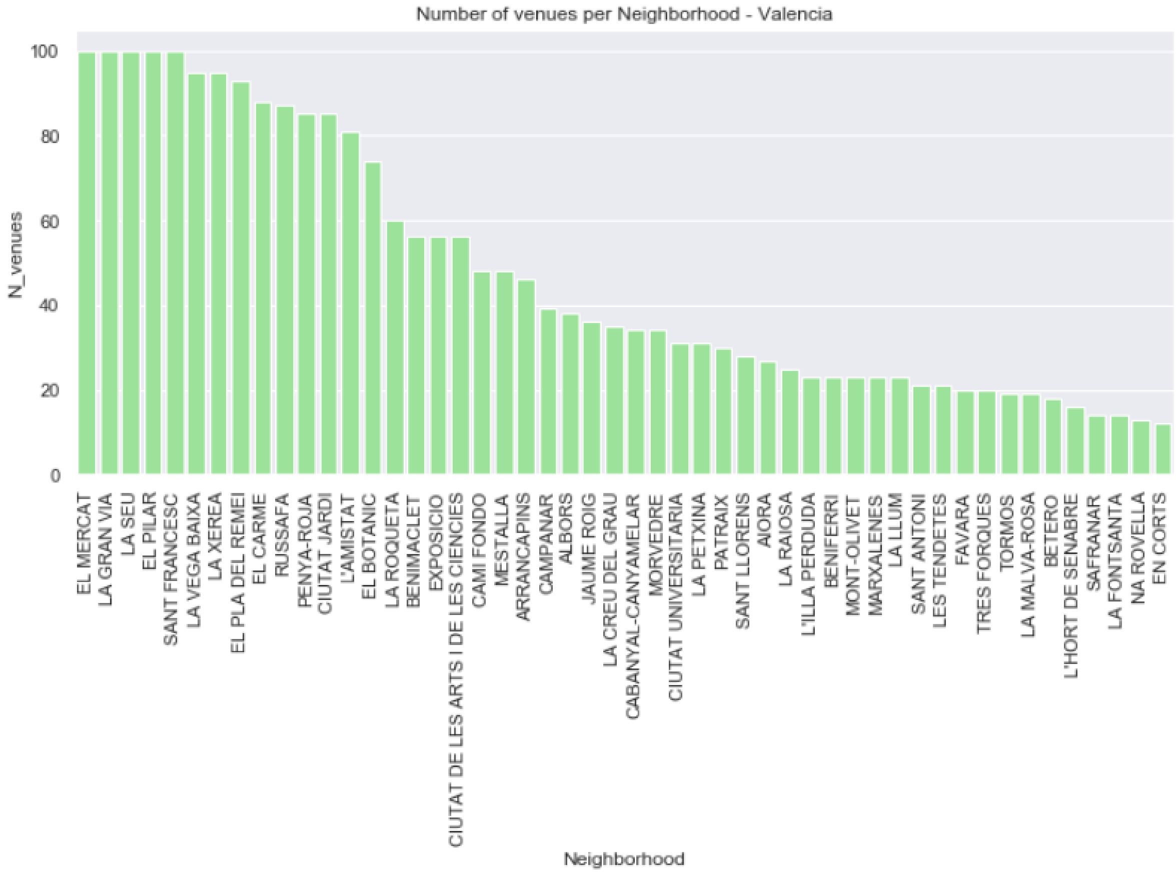
```
Venue Category
Spanish Restaurant    381
Restaurant            193
Bar                   166
Tapas Restaurant      154
Café                  109
Hotel                 100
Coffee Shop            91
Bakery                 84
Pizza Place            74
Italian Restaurant     73
Name: Neighborhood, dtype: int64
```

```python
# Count the number of locations per Venue Category in Valencia
venues[venues['City']=='Valencia'].groupby('Venue Category').count()['Neighborhood'].sort_values(ascending=False).head(10)
```

```
Venue Category
Spanish Restaurant         178
Tapas Restaurant           153
Restaurant                 113
Mediterranean Restaurant   103
Café                        87
Hotel                       85
Grocery Store               83
Italian Restaurant          80
Bakery                      65
Pub                         59
Name: Neighborhood, dtype: int64
```

The graphs below show the top neighborhoods by venues:





- **Clustering Model :**

Group the neighborhoods into clusters using the KMeans Clustering method.

Now lets initialize the k-means model using K=20

```
]: k_means = KMeans(init = "k-means++", n_clusters = 20, n_init = 15)
```

```
]: # Fit the model
   k_means.fit(venues_grouped.drop('Neighborhood',axis=1))
```

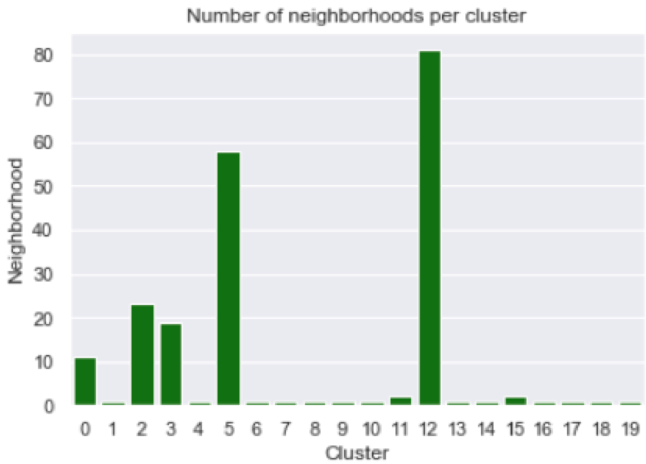
```
]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
       n_clusters=20, n_init=15, n_jobs=None, precompute_distances='auto',
       random_state=None, tol=0.0001, verbose=0)
```

```
]: #Add the labels to the venues_grouped dataset
   venues_grouped['Cluster']=k_means.labels_
```

```
]: #Obtain the number of neighborhoods per cluster
   venues_grouped.groupby('Cluster')['Neighborhood'].count()
```

Below shows the amount of neighborhoods per each cluster. As we can see, there are 5 “dominant” clusters, out of which Cluster 12 has the highest amount of neighborhoods (81). Remember this analysis includes both Madrid and Valencia, now we have to separate only the Madrid results.

```
venues_grouped_count = venues_grouped.groupby('Cluster')['Neighborhood'].count().to_frame()
venues_grouped_count.reset_index(inplace=True)
ax = sns.barplot(x='Cluster', y='Neighborhood', data=venues_grouped_count, color='green')
ax.set_title('Number of neighborhoods per cluster');
```



We identify that the target neighborhood “El Carme” is located in cluster 12:

```
target_cluster_df = neighborhoods_venues_sorted.loc[neighborhoods_venues_sorted['Neighborhood']=='EL CARME']
target_cluster_df.reset_index(inplace=True)
target_cluster=target_cluster_df.loc[0].at['Cluster']
print('The target cluster is: {} '.format(target_cluster))
```

```
The target cluster is: 12
```

And finally we can determine the neighborhoods from Madrid that belong to this cluster:

```
#Filter neighborhoods from Madrid that belong to the target cluster
possible_neighborhoods = neighborhoods_venues_sorted[
    (neighborhoods_venues_sorted['Cluster']==target_cluster) &
    (neighborhoods_venues_sorted['City']=='Madrid')]

print('There are {} neighborhoods in Madrid with similar characteristics than El Carme'
      .format(possible_neighborhoods.shape[0]))
```

```
There are 48 neighborhoods in Madrid with similar characteristics than El Carme
```

4. Results Summary

```
After performing a clustering analysis a group of 48 possible neighborhoods was identified with similar characteristics
to the target neighborhood from Valencia.
```

```
]: possible_neighborhoods
```

District	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	Population	Population_Normalized	Income_Normalized	Non_Italian_Restaurants	Ranking
ARGANZUELA	ACACIAS	Spanish Restaurant	Tapas Restaurant	Bar	36907	0.554642	0.397693	0.888889	0.505403
CHAMBERÍ	ALMAGRO	Spanish Restaurant	Restaurant	Bar	19858	0.298428	0.612413	0.555556	0.419114
CHAMBERÍ	ARAPILES	Spanish Restaurant	Bar	Bakery	24518	0.368459	0.375800	0.888889	0.404648
MONCLOA - ARAVACA	ARGUELLES	Spanish Restaurant	Tapas Restaurant	Hotel	24191	0.363545	0.475964	1.000000	0.448360
ARGANZUELA	ATOCHA	Restaurant	Spanish Restaurant	Grocery Store	1176	0.017673	0.337776	0.888889	0.215947
TETUÁN	BELLAS VISTAS	Spanish Restaurant	Bar	Pizza Place	29245	0.439497	0.277110	1.000000	0.416737
TETUÁN	BERRUGUETE	Tapas	Bar	Spanish	25089	0.377040	0.257704	1.000000	0.378717

RESULTS DISCUSSION

After clustering the Madrid and Valencia neighborhoods based on the results from the Foursquare API data, we were able to separate our dataset into 5 distinct clusters, and then from our target cluster pick the best candidates for our customer to open their new Italian restaurant.

restaurant in Madrid The selected neighborhoods have similar characteristics. Most of them are dominated by Spanish Restaurants and Bars, are densely populated neighborhoods and have few or no Italian restaurants. These constitute good candidates for opening a restaurant.

One issue I noted during the clustering analysis was that, even though we set the KMeans Clustering method with K=20 (aiming to segregate the neighborhoods as much as possible) we found several “one neighborhood” clusters.

CONCLUSION:

We were able to determine a good set of ten options to propose to our customer to open a new restaurant, considering the variables described in the previous sections.