★★★★★★★★★★

IBM Data Science

Analyzing Neighborhoods of Warsaw For Starting New Restaurant

Capstone Project

★★★★★★★★★★

Krzysztof Kamiński

02/20/2021

Contents

Introduction

Important problem to be solve when opening a restaurant is location. Warsaw is the capital and the largest city in Poland. It is the most popular city for earning money, due to the highest average salary from all of the polish cities. Warsaw and Krakow are most of popular places visited by tourists from all over the world.

Business Problem

Client want to open polish restaurant and he is wondering where should he should placed it. For last several years in Warsaw has been opened a variety of restaurants, from different sides of world. Location is very important for success of this kind of business, so it could gain interested tourist and local people.

Data

In order to achieve our final goal we will need to following data:

- Neighbourhoods of Warsaw
- Geographical coordinates of the neighbourhoods
- Venue data from FourSquare

Neighbourhoods Data

This data was extracted from Neighbourhoods of Warsaw Wikipedia page (https://en.wikipedia.org/wiki/Category:Neighbourhoods_of_Warsaw) using web scraping with BeautifulSoup library in Python. This will give us a detailed list of neighbourhoods present in Warsaw.

Geographical Coordinates

After list of neighbourhoods will be ready, geographical; coordinates of various neighbourhoods were extracted using GeoPy library in Python. Geographical coordinates are necessary for ploting maps during the project for visualizing data. After using GeoPy I will add two columns to dataframe with latitude and longitude of each neighbourhood as shown below:

	Neighbourhood	Latitude	Longitude
0	Boernerowo	52.26343	20.90139
1	Bródno	52.28612	21.03228
2	Czyste	52.22867	20.97193
3	Falenica	52.15590	21.20303
4	Frascati, Warsaw	52.22655	21.02636

Venue Data from FourSquare

Later there will be time for extracted venue data using FourSquare API. This venue data was used to study the venues in various neighbourhoods in Warsaw. This data provided important details of various restaurants in the area and helped us understand the competition. This data was very important because it helped us draw the main conclusion of the project.

Methodology

Feature Extraction

Feature Extraction was carried out through One Hot Encoding. In this method, each feature is a category that belongs to a venue which is then converted into binary, this means that 1 means this category is found in the venue and 0 means the opposite. Then, all the venues are grouped by the neighbourhoods, computing at the same time the mean. This will give us a venue for each row and each column will contain the frequency of occurrence of that particular category.

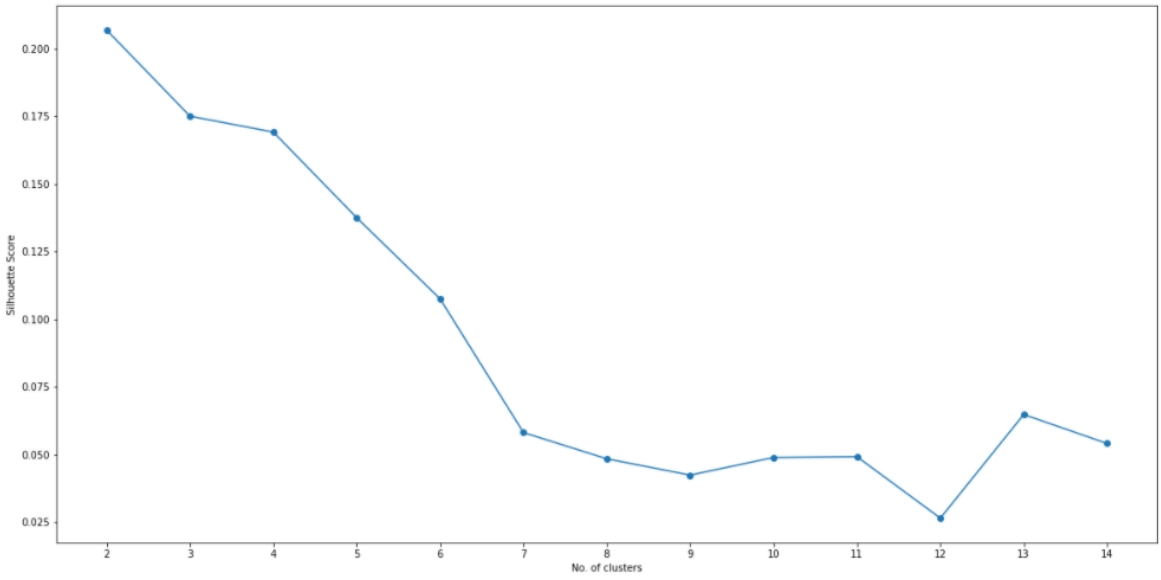
```
waw_1hot = pd.get_dummies(explore_waw[['Venue Category']], prefix="", prefix_sep="")


# Add neighbourhood column back to dataframe
waw_1hot['Neighbourhood'] = explore_waw['Neighbourhood']


# Move neighbourhood column to the first column
fixed_columns = [waw_1hot.columns[-1]] + waw_1hot.columns[:-1].values.tolist()
waw_1hot = waw_1hot[fixed_columns]

waw_1hot.head()
```

Unsupervised Learning

Unsupervised learning was carried out in order to find out the similarities between found similarities between neighbourhoods. K-Means, a clustering algorithm, was implemented. In this case K-Means is used due to its simplicity and its similarity approach to find patterns

- K-Means: K-Means is a clustering algorithm. This algorithm search clusters within the data and the main objective function is to minimize the data dispersion for each cluster. Thus, each group found represents a set of data with a pattern inside the multi-dimensional features. It is necessary for this algorithm to have a prior idea about the number of clusters since it is considered an input of this algorithm. For this reason, the elbow method is implemented. A chart that compares error vs number of cluster is done and the elbow is selected. Then, further analysis of each cluster is done.

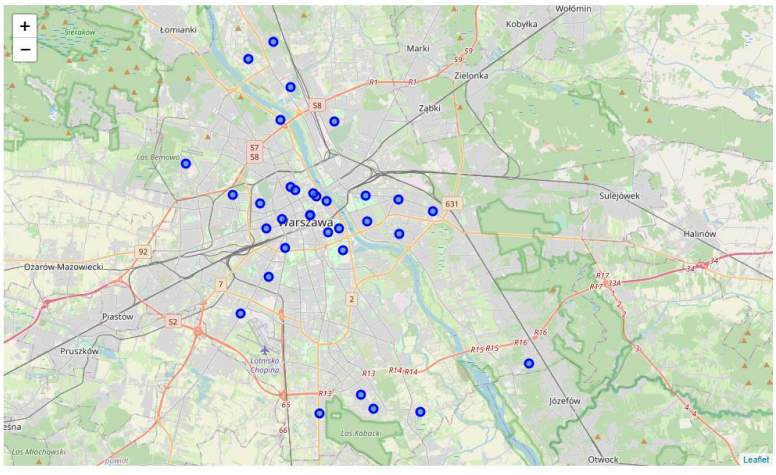
```
max_range = 15 #Max range 15 (number of clusters)

from sklearn.metrics import silhouette_samples, silhouette_score


indices = []
scores = []

for waw_clusters in range(2, max_range) :

    # Run k-means clustering
    waw_gc = waw_grouped_clustering
    kmeans = KMeans(n_clusters = waw_clusters, init = 'k-means++', random_state = 0).fit_predict(waw_gc)

    # Gets the score for the clustering operation performed
    score = silhouette_score(waw_gc, kmeans)

    # Appending the index and score to the respective lists
    indices.append(waw_clusters)
    scores.append(score)
```



Plotting

Various plotting techniques we used as well in order to visualize the data. Visualizing data often gives a clear understanding of the data as it is easier to spot patterns in a visualized data as compares to quantitative data.

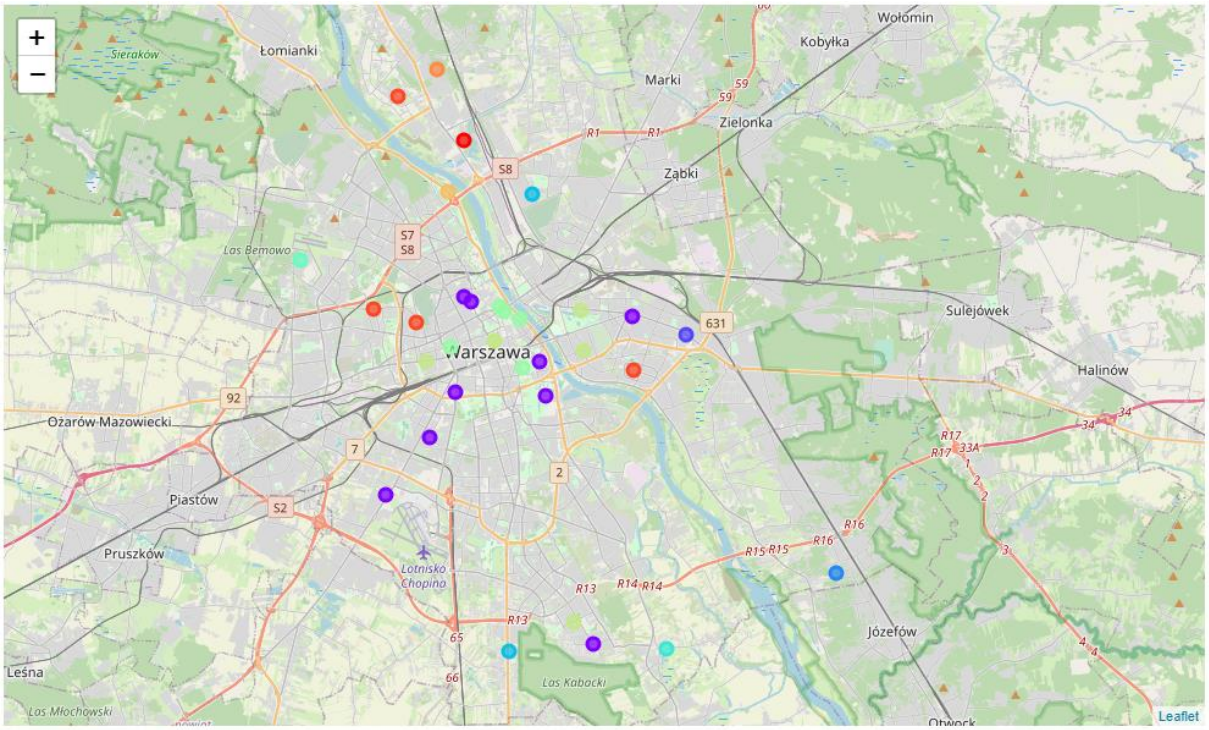
- Folium: Folium library was used to plot maps of Warsaw city as well as neighbourhoods. Folium was also used to visualize the cluster data.



Results

The above mentioned, K-Means clustering method was applied to the dataframe of neighbourhoods of Warsaw city. As mentioned earlier the number of clusters that was derived from elbow method was 12. The code as well as plotting of clusters can be seen below:

```
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# Setup color scheme for different clusters
x = np.arange(waw_clusters)
ys = [i + x + (i*x)**2 for i in range(waw_clusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

markers_colors = []
for lat, lon, poi, cluster in zip(waw_final['Latitude'], waw_final['Longitude'], waw_final['Neighbourhood'],
                                  waw_final['Cluster Labels']):
    label = folium.Popup(str(poi) + ' (Cluster ' + str(cluster + 1) + ')', parse_html=True)
    map_clusters.add_child(
        folium.features.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7))

map_clusters
```



After visualising the clusters, the individual clusters were studied and some important conclusions were derived. The neighbourhood that had the most number of restaurants was cluster number 9.

Discussion

As mentioned earlier the most suitable neighbourhoods for starting the restaurant business are present in the cluster number 9. Our K-Means model worked perfectly and successfully clustered similar neighbourhoods together. After studying all four clusters, it is recommended to the client that neighbourhoods such as Czyste, Kamionek, Natolin, Saska Kępa that fall in cluster 9 look like good locations for starting their restaurant business. The client can go ahead and make a decision depending on other factors like availability and legal requirements that are out of scope of this project.

Conlcusion

Data analysis and machine learning techniques used in this project can be very helpful in determining solutions of certain business problems. Python's inbuilt libraries such as GeoPy, Folium and BeautifulSoup make it very easy and effective for a data scientist to analyse a geographical location because these libraries make it very easy to extract data that is easily available online. In this project we studied the neighbourhoods of Warsaw city and came up with a recommendation of neighbourhoods where our client can start their restaurant business.