Capstone Project

Battle of Seoul

1. Introduction

Now that you have been equipped with the skills and the tools to use location data to explore a geographical location, over the course of two weeks, you will have the opportunity to be as creative as you want and come up with an idea to leverage the Foursquare location data to explore or compare neighborhoods or cities of your choice or to come up with a problem that you can use the Foursquare location data to solve. If you cannot think of an idea or a problem, here are some ideas to get you started:

1. In Module 3, we explored New York City and the city of Toronto and segmented and clustered their neighborhoods. Both cities are very diverse and are the financial capitals of their respective countries. One interesting idea would be to compare the neighborhoods of the two cities and determine how similar or dissimilar they are. Is New York City more like Toronto or Paris or some other multicultural city? I will leave it to you to refine this idea.

2. In a city of your choice, if someone is looking to open a restaurant, where would you recommend that they open it? Similarly, if a contractor is trying to start their own business, where would you recommend that they set up their office?

These are just a couple of many ideas and problems that can be solved using location data in addition to other datasets. No matter what you decide to do, make sure to provide sufficient justification of why you think what you want to do or solve is important and why would a client or a group of people be interested in your project. So I decided to try to answer this simple question: where would you recommend to open a new restaurant?

1.1. Business problem

The city chosen to answer the initial question is Seoul a capital and the most populous city in South Korea. Its continuously built-up urban area, that stretches well beyond the boundaries of the administrative metropolitan city with over 9.7 million inhabitants.

Seoul is considered a leading alpha global city, with strengths in the field of the art, commerce, design, education, entertainment, fashion, finance, healthcare, media, services, research and tourism. Its business district hosts Korea's stock exchange, and the headquarters of national and international banks and companies.

1.2. Target audience

- A business entrepreneur that wants open a new restaurant in Seoul.

- Business Analyst or Data Scientists, who wish to analyze the districts of Seoul using python, jupyter notebook and some machine learning techniques.

- Someone curious about data that want to have an idea, how beneficial it is to open a restaurant and what are the pros and cons of this business.

2. Data Section

First we need some information about the area of Seoul such as borough, districts, population etc... I think a good place to take a look is wikipedia.

The districts are 24 with these coordinates:

	District	Population	Area(km2)	Population_Density(km2)	Latitude	Longitude
0	Dobong	355712	20.70	17184	37.695000	127.046940
1	Dongdaemun	376319	14.21	26483	37.571000	127.009700
2	Dongjak	419261	16.35	25643	37.512403	126.939253
3	Eunpyeong	503243	29.70	16944	37.602697	126.929111
4	Gangbuk	338410	23.60	14339	37.639611	127.025656
5	Gangdong	481332	24.59	19574	37.530000	127.123890
6	Gangnam	583446	39.50	14771	37.496670	127.027500
7	Gangseo	591653	41.43	14281	37.548610	126.850830
8	Geumcheon	258030	13.02	19818	37.451853	126.902036
9	Guro	457131	20.12	22720	37.495000	126.887000
10	Gwanak	531960	29.57	17990	37.478400	126.951600
11	Gwangjin	377375	17.06	22120	37.537900	127.082100
12	Jongno	165344	23.91	6915	37.599440	126.974720
13	Jung	136227	9.96	13677	37.556000	126.970000
14	Jungnang	423411	18.50	22887	37.606400	127.092600
15	Mapo	395830	23.84	16604	37.563800	126.908400
16	Nowon	586056	35.44	16536	37.654192	127.056794
17	Seocho	454288	47.00	9666	37.483610	127.032500
18	Seodaemun	320861	17.61	18220	37.579170	126.936670
19	Seongbuk	475961	24.58	19364	37.589170	127.018330
20	Seongdong	303891	16.86	19364	37.563330	127.036940
21	Songpa	671794	33.88	19829	37.514170	127.106670
22	Yangcheon	490708	17.40	28202	37.516872	126.866397
23	Yeongdeungpo	421436	24.53	17180	37.526390	126.896390
24	Yongsan	249914	21.87	11427	37.538330	126.965560

3. Methodology

3.1. Business Understanding

The aim of this project is to find the best district of Seoul to open a new restaurant.

3.2. Analytical Approach

The total number of districts in Seoul are 24 so we need to find a way to cluster them based on their similarities, that are the number and the kind of restaurant. Briefly, after some steps of Data Cleaning and Data Exploration, I will use a K-Means algorithm to extract the clusters, produce a map and make an argument on the final result.

3.3. Data Exploration

To explore the data, I will use “Folium” a python library that can create interactive leaflet map using coordinate data.

Create map of Seoul using latitude and longitude values

3.3. Data Exploration

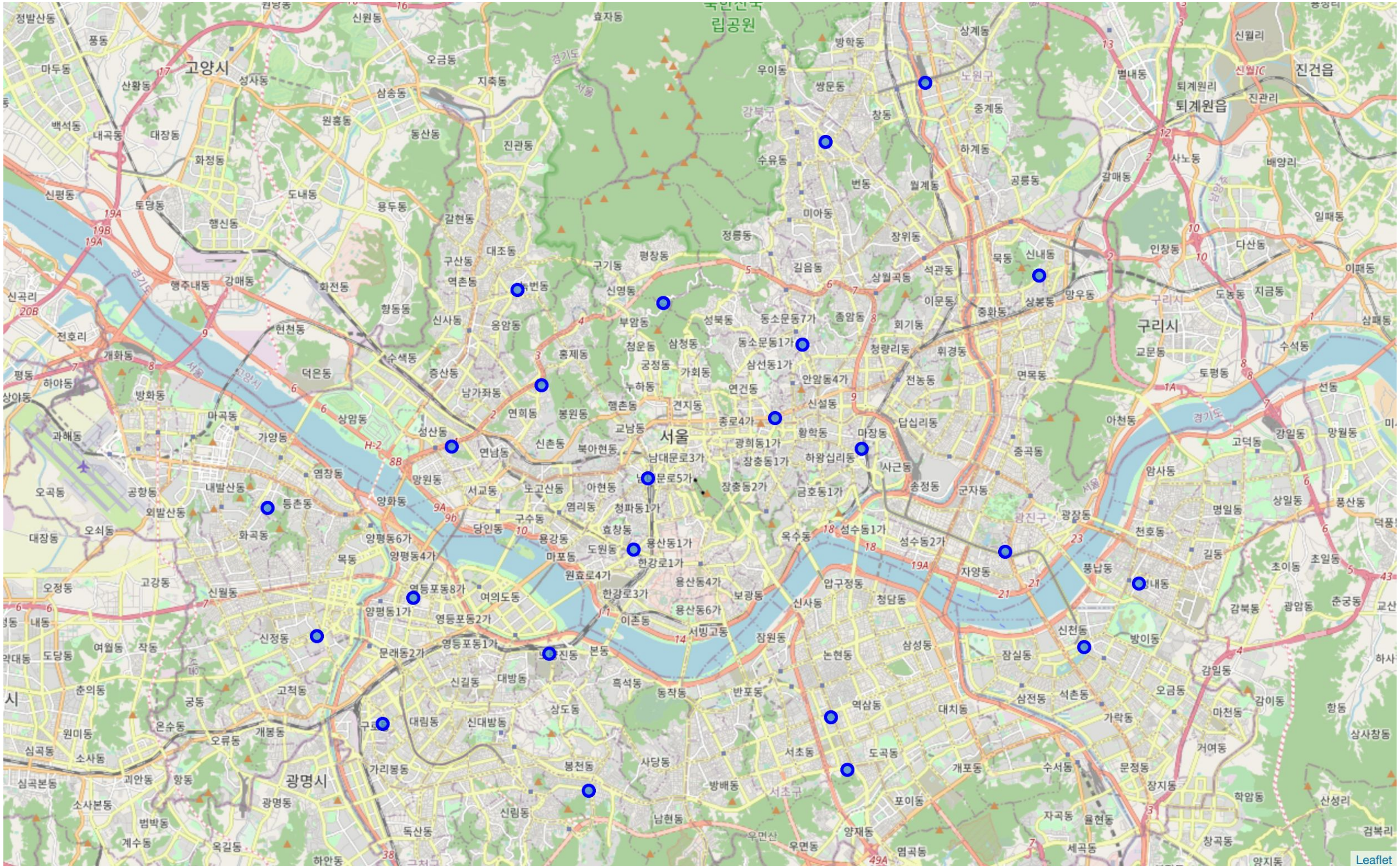
It's pretty important to use some good visualization to understand better the area

A map of Seoul with centroids of every district:

```
map_s = folium.Map(location=[latitude, longitude], zoom_start=12)

# add markers to map
for lat, lng, district in zip(swiki_df['Latitude'],
                              swiki_df['Longitude'],
                              swiki_df['District']):
    label = '{}'.format(district)
    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_s)

map_s
```



서울
고양시
구리시
광명시
진건읍
퇴계원읍
Leaflet

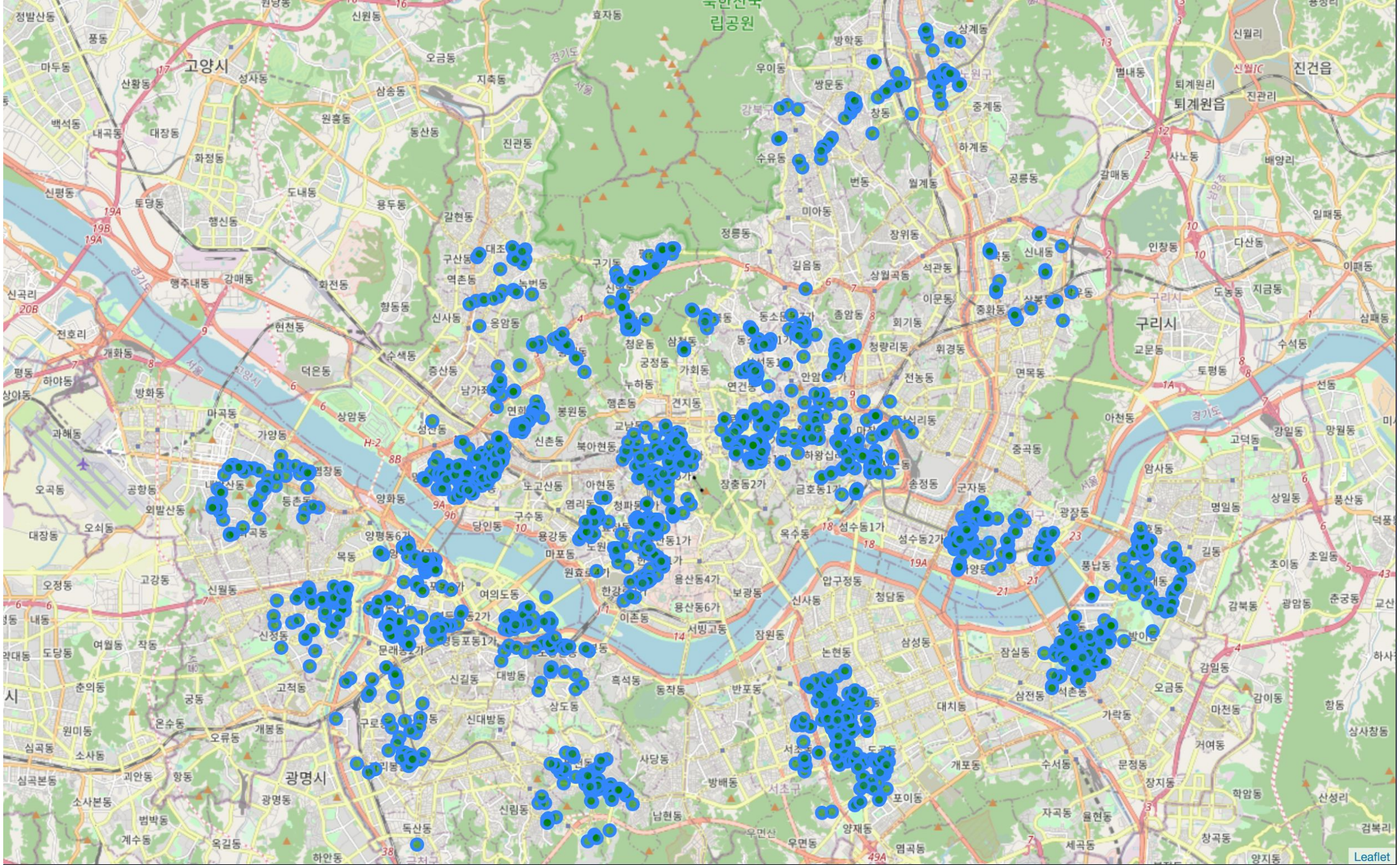
3.3. Data Exploration

Extract venues for each district in Seoul

```
# create the API request URL
url = 'https://api.foursquare.com/v2/venues/explore?&section=food&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    lat,
    lng,
    radius,
    LIMIT)
```

```
nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in venue_list])
nearby_venues.columns = ['District',
                'District Latitude',
                'District Longitude',
                'Venue',
                'Venue Latitude',
                'Venue Longitude',
                'Venue Category']
```

	District	District Latitude	District Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Dobong	37.695	127.04694	PARIS BAGUETTE	37.683948	127.045930	Bakery
1	Dobong	37.695	127.04694	우리나라	37.701103	127.054487	BBQ Joint
2	Dobong	37.695	127.04694	산넘어남촌 의정부점	37.704067	127.047874	Korean Restaurant
3	Dobong	37.695	127.04694	도봉산갈비	37.685881	127.046143	Korean Restaurant
4	Dobong	37.695	127.04694	도봉산 산두부	37.686635	127.037702	Korean Restaurant



고양시
구리시
광명시
퇴계원읍
진건읍
북한산국립공원
정발산동
풍동
마두동
백석동
대장동
화정동
신평동
토당동
행신동
성사동
삼송동
원흥동
동산동
오금동
지축동
진관동
효자동
방학동
상계동
신월리
별내동
퇴계원리
진관리
우이동
쌍문동
창동
중계동
수유동
하계동
사노동
배양리
번동
월계동
공릉동
갈매동
미아동
장위동
일패동
정릉동
길음동
석관동
신내동
인창동
다산동
상월곡동
이문동
망우동
도농동
지금동
이패동
회기동
중화동
종암동
청량리동
휘경동
교문동
수석동
전농동
면목동
토평동
선동
행주내동
강매동
화전동
향동동
구산동
역촌동
신사동
응암동
구기동
청운동
궁정동
삼청동
가회동
누하동
연건동
행촌동
견지동
현천동
수색동
증산동
남가좌동
연희동
봉원동
신촌동
북아현동
교남동
개화동
방화동
마곡동
상암동
가양동
성산동
과해동
공항동
외발산동
오곡동
오쇠동
대장동
등촌동
염창동
양화동
노고산동
아현동
염리동
구수동
당인동
용강동
마포동
원효로4가
한강로3가
이촌동
용산동4가
용산동6가
보광동
서빙고동
신사동
청담동
압구정동
잠원동
논현동
삼성동
잠실동
장충동2가
금호동1가
하왕십리동
옥수동
성수동1가
성수동2가
송정동
군자동
광장동
중곡동
아차산동
암사동
명일동
상일동
풍산동
길동
초이동
초일동
강일동
망월동
고덕동
풍납동
방이동
감북동
광암동
춘궁동
강일동
감일동
오금동
감이동
가락동
마천동
항동
삼사창동
거여동
문정동
장지동
학암동
신성리
창곡동
검복리
자곡동
율현동
세곡동
수서동
삼전동
석촌동
대치동
개포동
포이동
양재동
염곡동
우면동
서초구
방배동
남현동
사당동
반포동
동작동
흑석동
상도동
신대방동
대방동
신길동
영등포동1가
문래동2가
여의도동
양평동6가
목동
신월동
신정동
오정동
고강동
작동
여월동
도당동
내동
춘의동
고척동
궁동
개봉동
오류동
온수동
구로동
원미동
심곡동
소사동
괴안동
항동
심곡본동
소사본동
범박동
광명동
신림동
독산동
계수동
옥길동
하안동
신곡리
전호리
평동
하야동
상야동
더은동
Leaflet

3.4. Clustering

To analyze which district of Seoul is good to open a new restaurant, I will use a K-means clustering: a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity.

So the first step is identify the best “K” using a famous analytical approach: the elbow method.

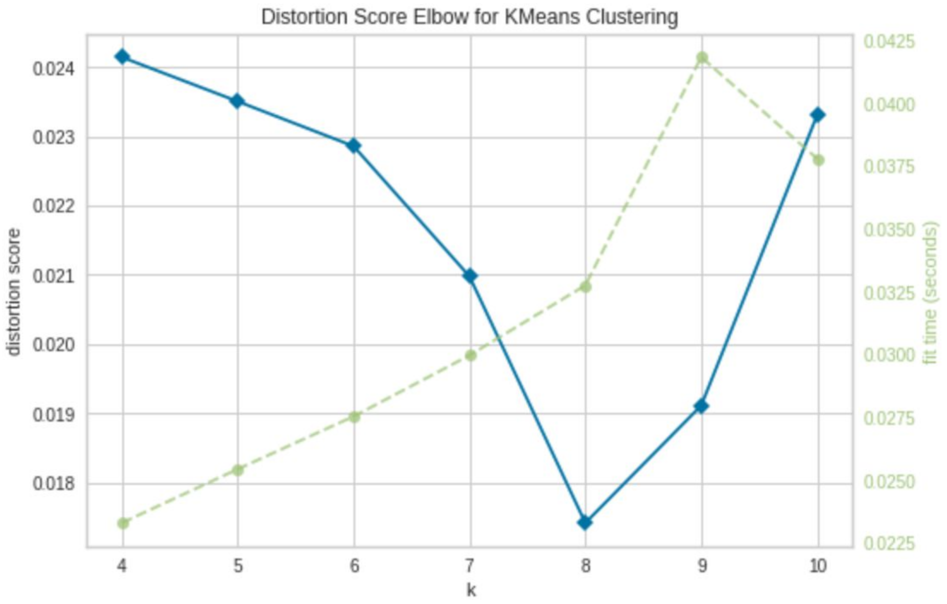
3.4. Clustering

```python
from sklearn.cluster import KMeans
from yellowbrick.cluster import KElbowVisualizer

s_part_clustering = s_grouped.drop('District', 1)

# Instantiate the clustering model and visualizer
model = KMeans()
visualizer = KElbowVisualizer(model, k=(4,11))

visualizer.fit(s_part_clustering)        # Fit the data to the visualizer
visualizer.poof()    # Draw/show/poof the data
```



From the plot up here, I can easily say that the best K is 7.

3.4. Clustering

Finally, we can try to cluster the neighborhood based on the venue categories and use K-Means clustering. The 7 clusters are partitioned based on similar type of restaurants that belong to neighborhoods.

To run the cluster, I have used the code snippet below.

```
# set number of clusters
kclusters = 7

s_grouped_clustering = s_grouped.drop('District', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(s_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]
```

3.4. Clustering

And merge to obtain the final dataset:

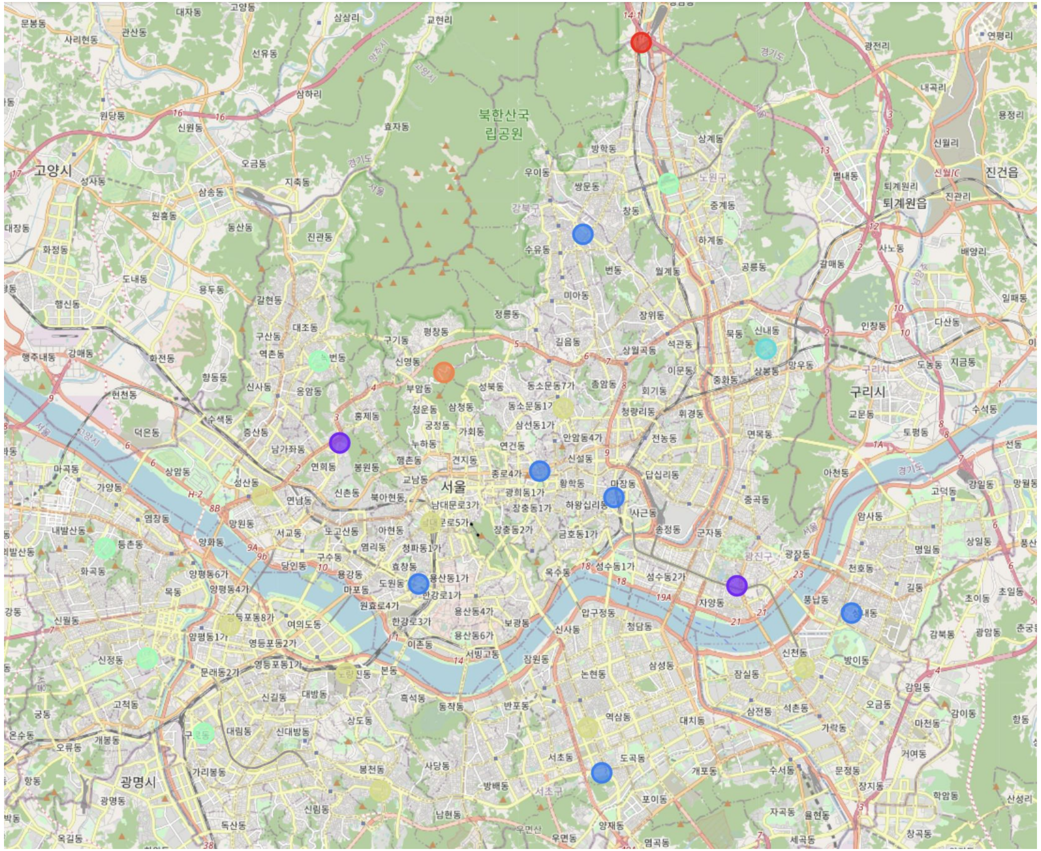
```python
s_complete = swiki_df.join(s_sorted.set_index('District'), on='District')
s_complete['Cluster Labels'] = s_complete['Cluster Labels'].fillna(0)
s_complete['Cluster Labels'] = s_complete['Cluster Labels'].astype(int)

s_complete.head()
```

	District	Population	Area(km2)	Population_Density(km2)	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Dobong	355712	20.70	17184	37.695000	127.046940	0	Korean Restaurant	Pizza Place	Spanish Restaurant	Vegetarian / Vegan Restaurant	Restaurant	Japanese Restaurant	Seafood Restaurant	Sandwich Place	Sushi Restaurant	Chinese Restaurant
1	Dongdaemun	376319	14.21	26483	37.571000	127.009700	2	Korean Restaurant	Chinese Restaurant	Noodle House	Japanese Restaurant	Indian Restaurant	Seafood Restaurant	Pizza Place	Fried Chicken Joint	Fast Food Restaurant	Burger Joint
2	Dongjak	419261	16.35	25643	37.512403	126.939253	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Seafood Restaurant	Fast Food Restaurant	Noodle House	Fried Chicken Joint	Italian Restaurant	Food Court	Steakhouse
3	Eunpyeong	503243	29.70	16944	37.602697	126.929111	4	Korean Restaurant	Fast Food Restaurant	Japanese Restaurant	Chinese Restaurant	Fried Chicken Joint	Seafood Restaurant	Sushi Restaurant	Steakhouse	Spanish Restaurant	Diner
4	Gangbuk	338410	23.60	14339	37.639611	127.025656	2	Korean Restaurant	Fast Food Restaurant	Fried Chicken Joint	Japanese Restaurant	Noodle House	Sushi Restaurant	Spanish Restaurant	Diner	Pizza Place	Restaurant

4. Result and Discussion

Before to start to analyze all the clusters, let’s take a look on a folium map:



4. Result and Discussion

As we can see, each cluster belong to a color with different characteristics. You can read the complete list above:

Cluster 1:

```
[49] s_complete.loc[s_complete['Cluster Labels'] == 0, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	355712	37.695	127.04694	0	Korean Restaurant	Pizza Place	Spanish Restaurant	Vegetarian / Vegan Restaurant	Restaurant	Japanese Restaurant	Seafood Restaurant	Sandwich Place	Sushi Restaurant	Chinese Restaurant

Cluster 2:

```
[50] s_complete.loc[s_complete['Cluster Labels'] == 1, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
11	377375	37.53790	127.08210	1	Chinese Restaurant	Korean Restaurant	Italian Restaurant	Fast Food Restaurant	Japanese Restaurant	Sushi Restaurant	Asian Restaurant	Pizza Place	Restaurant	Noodle House
18	320861	37.57917	126.93667	1	Korean Restaurant	Chinese Restaurant	Fast Food Restaurant	Italian Restaurant	Japanese Restaurant	Noodle House	Pizza Place	Restaurant	American Restaurant	Sandwich Place

4. Result and Discussion

Cluster 3:

```
[48] s_complete.loc[s_complete['Cluster Labels'] == 2, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
1	376319	37.571000	127.009700	2	Korean Restaurant	Chinese Restaurant	Noodle House	Japanese Restaurant	Indian Restaurant	Seafood Restaurant	Pizza Place	Fried Chicken Joint	Fast Food Restaurant	Burger Joint
4	338410	37.639611	127.025656	2	Korean Restaurant	Fast Food Restaurant	Fried Chicken Joint	Japanese Restaurant	Noodle House	Sushi Restaurant	Spanish Restaurant	Diner	Pizza Place	Restaurant
5	481332	37.530000	127.123890	2	Korean Restaurant	Fast Food Restaurant	Japanese Restaurant	Seafood Restaurant	Chinese Restaurant	Noodle House	Restaurant	Steakhouse	Sandwich Place	Food Court
17	454288	37.483610	127.032500	2	Korean Restaurant	Seafood Restaurant	Noodle House	Chinese Restaurant	Burger Joint	Sushi Restaurant	Pizza Place	Restaurant	Japanese Restaurant	Asian Restaurant
20	303891	37.563330	127.036940	2	Korean Restaurant	Fast Food Restaurant	Seafood Restaurant	Fried Chicken Joint	Food Truck	Burger Joint	Chinese Restaurant	Steakhouse	Sandwich Place	Asian Restaurant
24	249914	37.538330	126.965560	2	Korean Restaurant	Chinese Restaurant	Noodle House	Japanese Restaurant	Seafood Restaurant	Fast Food Restaurant	Ramen Restaurant	Dim Sum Restaurant	Fried Chicken Joint	Sushi Restaurant

Cluster 4:

```
[51] s_complete.loc[s_complete['Cluster Labels'] == 3, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
8	258030	37.451853	126.902036	3	Fast Food Restaurant	Korean Restaurant	Chinese Restaurant	Spanish Restaurant	Vegetarian / Vegan Restaurant	Pizza Place	Restaurant	Japanese Restaurant	Seafood Restaurant	Sandwich Place
14	423411	37.606400	127.092600	3	Fast Food Restaurant	Pizza Place	Korean Restaurant	Steakhouse	Spanish Restaurant	Diner	Restaurant	Japanese Restaurant	Seafood Restaurant	Sandwich Place

Cluster 5:

```
[52] s_complete.loc[s_complete['Cluster Labels'] == 4, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
3	503243	37.602697	126.929111	4	Korean Restaurant	Fast Food Restaurant	Japanese Restaurant	Chinese Restaurant	Fried Chicken Joint	Seafood Restaurant	Sushi Restaurant	Steakhouse	Spanish Restaurant	Diner
7	591653	37.548610	126.850830	4	Korean Restaurant	Fast Food Restaurant	Italian Restaurant	Steakhouse	Restaurant	Japanese Restaurant	Sushi Restaurant	Chinese Restaurant	Noodle House	Asian Restaurant
9	457131	37.495000	126.887000	4	Korean Restaurant	Fast Food Restaurant	Chinese Restaurant	Japanese Restaurant	Fried Chicken Joint	Sushi Restaurant	Steakhouse	Noodle House	Restaurant	Food Truck
16	586056	37.654192	127.056794	4	Korean Restaurant	Fast Food Restaurant	Japanese Restaurant	Italian Restaurant	Fried Chicken Joint	Steakhouse	Seafood Restaurant	Noodle House	American Restaurant	Pizza Place
22	490708	37.516872	126.866397	4	Korean Restaurant	Fast Food Restaurant	Chinese Restaurant	Sushi Restaurant	Fried Chicken Joint	Salad Place	Italian Restaurant	Food Court	Asian Restaurant	Steakhouse

4. Result and Discussion

Cluster 6:

```
[53] s_complete.loc[s_complete['Cluster Labels'] == 5, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
2	419261	37.512403	126.939253	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Seafood Restaurant	Fast Food Restaurant	Noodle House	Fried Chicken Joint	Italian Restaurant	Food Court	Steakhouse
6	583446	37.496670	127.027500	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Sushi Restaurant	Fried Chicken Joint	Noodle House	Seafood Restaurant	Burger Joint	Italian Restaurant	Pizza Place
10	531960	37.478400	126.951600	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Fast Food Restaurant	Burger Joint	Spanish Restaurant	Sushi Restaurant	Italian Restaurant	Noodle House	Asian Restaurant
13	136227	37.556000	126.970000	5	Korean Restaurant	Chinese Restaurant	Seafood Restaurant	Japanese Restaurant	Italian Restaurant	Bistro	Noodle House	Ramen Restaurant	Sandwich Place	Sushi Restaurant
15	395830	37.563800	126.908400	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Italian Restaurant	Sandwich Place	Sushi Restaurant	Ramen Restaurant	Noodle House	Vegetarian / Vegan Restaurant	Breakfast Spot
19	475961	37.589170	127.018330	5	Korean Restaurant	Noodle House	Chinese Restaurant	Fast Food Restaurant	Indian Restaurant	Japanese Restaurant	Fried Chicken Joint	Sandwich Place	Burger Joint	Sushi Restaurant
21	671794	37.514170	127.106670	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Italian Restaurant	Seafood Restaurant	Sandwich Place	Indian Restaurant	Restaurant	Burger Joint	Fast Food Restaurant
23	421436	37.526390	126.896390	5	Korean Restaurant	Japanese Restaurant	Chinese Restaurant	Asian Restaurant	Restaurant	Seafood Restaurant	Noodle House	Italian Restaurant	Pizza Place	Fried Chicken Joint

Cluster 7:

```
[54] s_complete.loc[s_complete['Cluster Labels'] == 6, s_complete.columns[[1] + list(range(4, s_complete.shape[1]))]]
```

	Population	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
12	165344	37.59944	126.97472	6	Korean Restaurant	Italian Restaurant	Chinese Restaurant	Japanese Restaurant	Restaurant	Fried Chicken Joint	Noodle House	Burger Joint	Pizza Place	Seafood Restaurant

5. Conclusion

As the analysis is performed on small set of data, we can achieve better results by increasing the district information (see the next chapter). Anyway Seoul is an international city with many different types of new restaurant business to offer and I think we have gone through the process of identifying the business problem, specifying the data required, clean the datasets, performing a machine learning algorithm using k-means clustering and providing some useful tips to our stakeholder.

6. Next Developments

Next steps I recommend would be:

- Use a different Venue API with more data. Unfortunately foursquare isn’t pretty famous in Korea.
- Mostly users prefer Google Maps or Facebook.
- Find and use updated demographics data about Milan’s Neighborhood.
- Try a Neighborhood-Based Clustering.