The Battle of Neighborhoods | Final Report

Introduction:

This project has the purpose of helping people explore facilities around their neighbourhood, or possible neighbourhoods they would consider living in. Lots of people are migrating to São Paulo, Brazil, and needed lots of research for good housing prices and reputated schools for their children. This project aims ease of access to Cafe, School, Super market, medical shops, grocery shops, mall, theatre, hospital, like minded people, etc.

It was built upon data colected from São Paulo, Brazil, but I could easily be reused for any city around the world, just changing parameters in the code. One could easily compare locations in order to choose a place that better suits them.

The Location:

We require geographical location data for São Paulo and its neighbourhoods. Postal codes are useful as a starting point. Using Postal codes we can find out the neighborhoods, boroughs, venues and their most popular venue categories.

Foursquare API:

This project would use Four-square API as its prime data gathering source as it has a database of millions of places, especially their API places which provides the ability to perform location search, location sharing and details about a business.

Due to http request limitations the number of places per neighborhood parameter would reasonably be set to 300 and the radius parameter would be set to 1000.

Clustering Approach:

To compare the similarities of two neighbourhoods, we decided to segment them and group them into clusters, definined by the most commom places found near each location. To be able to do that, we need to cluster data which is a form of unsupervised machine learning: k-means clustering algorithm

Libraries Which are Used to Develope the Project:

Pandas: For creating and manipulating dataframes. Folium: Python visualization library would be used to visualize the neighborhoods cluster distribution using interactive leaflet map. Scikit Learn: For importing k-means clustering. JSON: Library to handle JSON files. XML: To separate data from presentation and XML stores data in plain text format. Geocoder: To retrieve Location Data. Beautiful Soup and Requests: To scrap and library to handle http requests. Matplotlib: Python Plotting Module.

Data Description:

Data Link: https://pt.wikipedia.org/wiki/Lista_dos_distritos_de_S%C3%A3o_Paulo_por_popula%C3%A7%C3%A3o

Which contains the dataset of Postal Codes in São Paulo, as we scrapped from wikipedia on Week 3. Dataset consisting of latitude and longitude, zip codes.

Foursquare API Data:

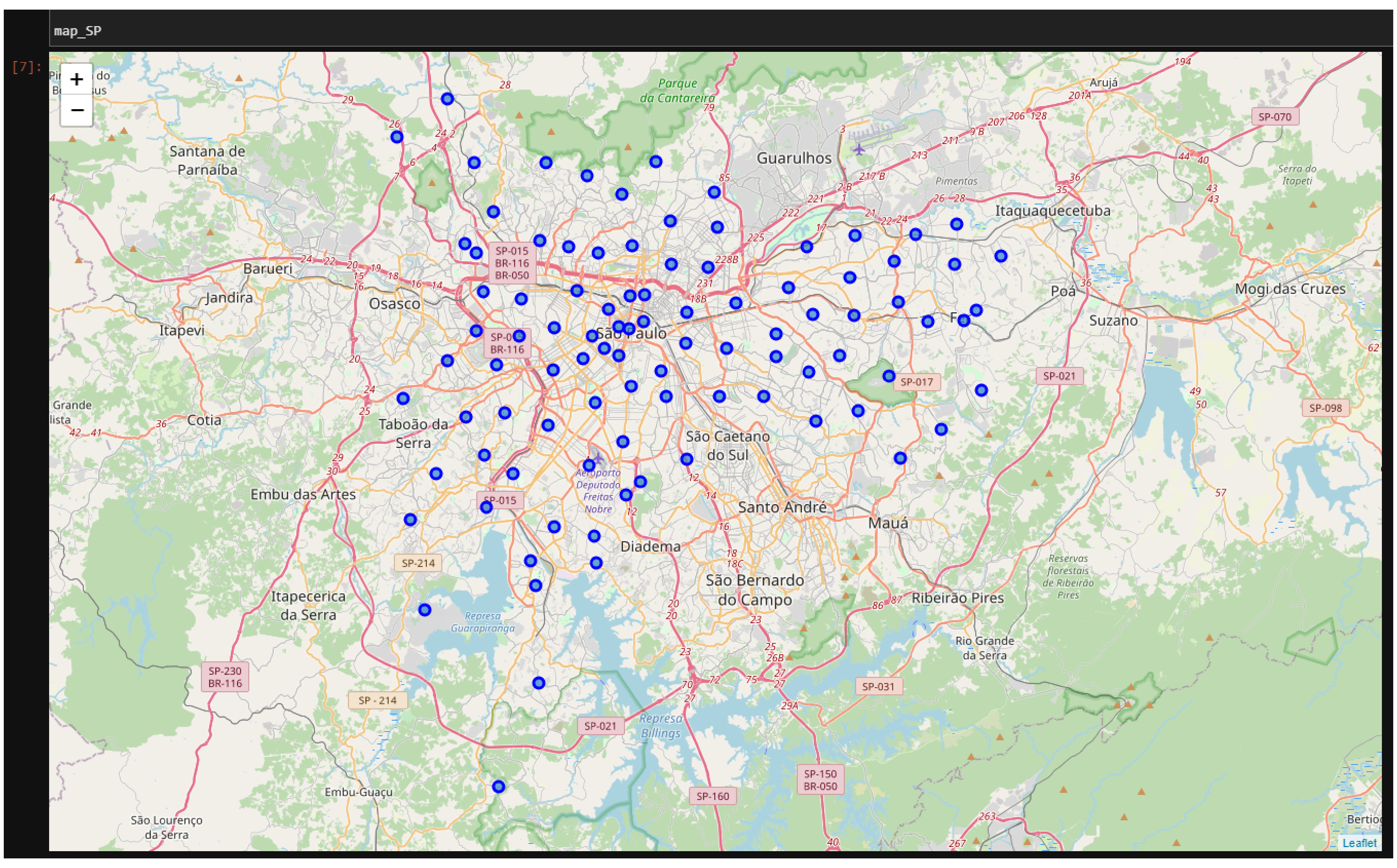
We will need data about different venues in different neighborhoods of that specific borough. In order to gain that information we will use Foursquare's location information. Foursquare is a location data provider with information about all sorts of venues and events within an area of interest. Such information includes venue names, locations, menus and even photos. As such, the foursquare location platform will be used as the sole data source since all the stated required information can be obtained through the API.

After finding the list of neighborhoods, we then connect to the Foursquare API to gather information about venues inside each and every neighborhood. For each neighborhood, we have chosen the radius to be 100 meter.

The data retrieved from Foursquare contained information of venues within a specified distance of the longitude and latitude of the postcodes. The information obtained per venue as follows:

```
1. Neighborhood
2. Neighborhood Latitude
3. Neighborhood Longitude
4. Venue
5. Name of the venue e.g. the name of a store or restaurant
6. Venue Latitude
7. Venue Longitude
8. Venue Category
```

Map of São Paulo:

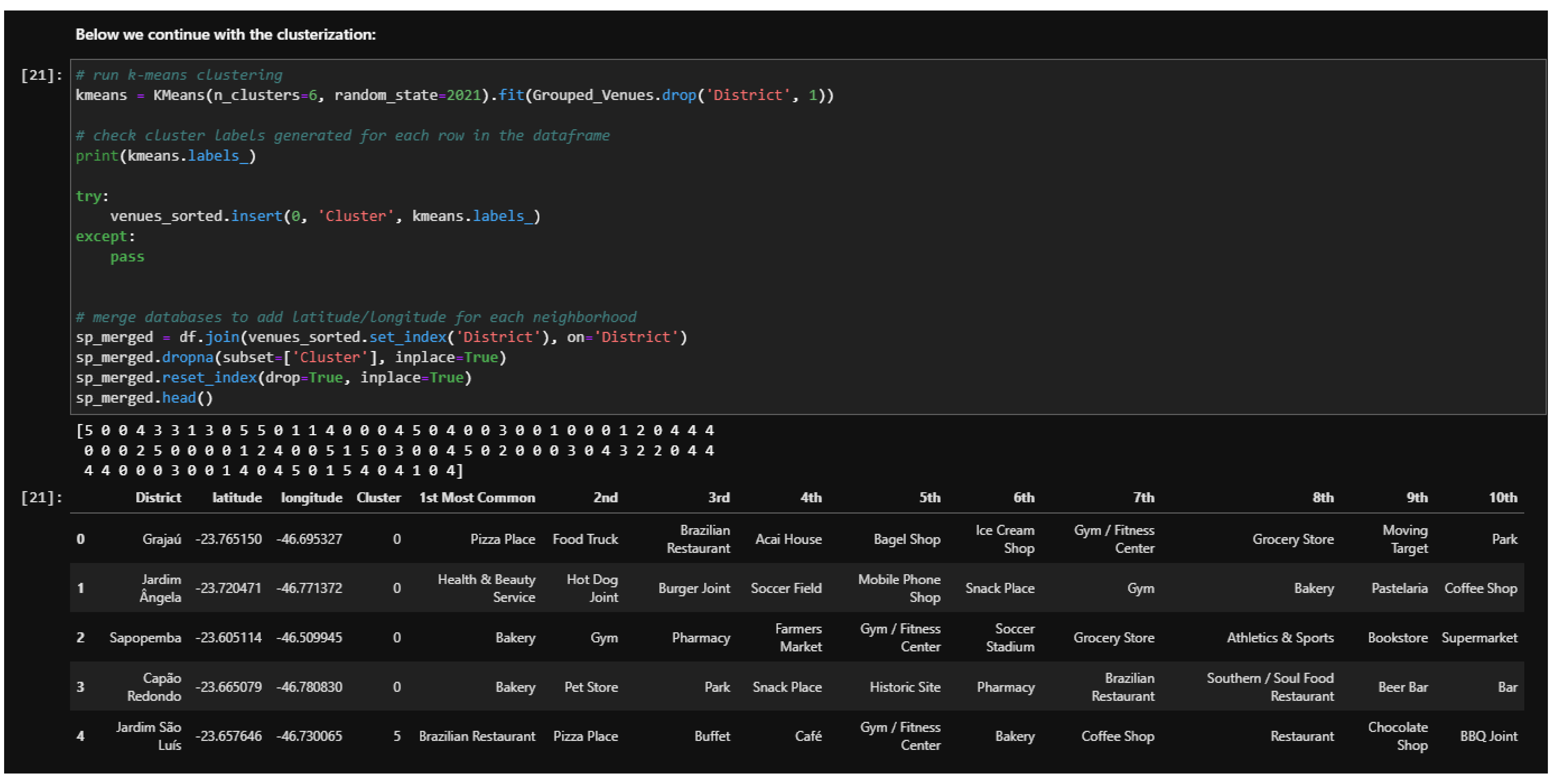


```
[7]: map_SP
```

Methodology Section

Clustering Approach: We decided to explore neighborhoods, segment them, and group them into clusters to find similar neighborhoods in a big city like São Paulo. To be able to do that, we need to cluster data which is a form of unsupervised machine learning: k-means clustering algorithm.

Using K-Means Clustering Approach:



Below we continue with the clusterization:

```
[21]: # run k-means clustering
kmeans = KMeans(n_clusters=6, random_state=2021).fit(Grouped_Venues.drop('District', 1))

# check cluster labels generated for each row in the dataframe
print(kmeans.labels_)

try:
    venues_sorted.insert(0, 'Cluster', kmeans.labels_)
except:
    pass


# merge databases to add latitude/longitude for each neighborhood
sp_merged = df.join(venues_sorted.set_index('District'), on='District')
sp_merged.dropna(subset=['Cluster'], inplace=True)
sp_merged.reset_index(drop=True, inplace=True)
sp_merged.head()
```

```
[5 0 0 4 3 3 1 3 0 5 5 0 1 1 4 0 0 0 4 5 0 4 0 0 3 0 0 1 0 0 0 1 2 0 4 4 4
 0 0 0 2 5 0 0 0 0 1 2 4 0 0 5 1 5 0 3 0 0 4 5 0 2 0 0 0 3 0 4 3 2 2 0 4 4
 4 4 0 0 0 3 0 0 1 4 0 4 5 0 1 5 4 0 4 1 0 4]
```

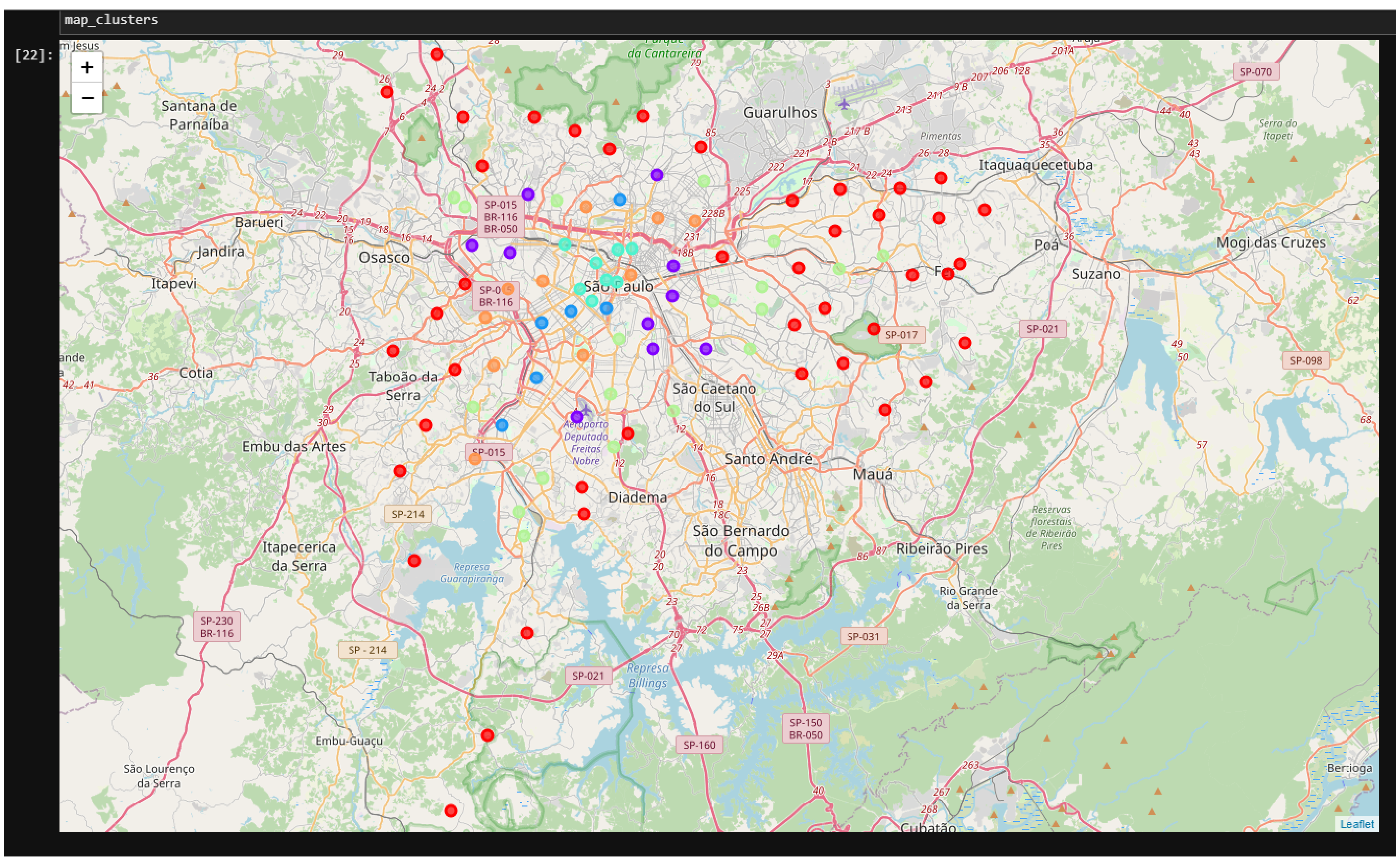
[21]:

	District	latitude	longitude	Cluster	1st Most Common	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
0	Grajaú	-23.765150	-46.695327	0	Pizza Place	Food Truck	Brazilian Restaurant	Acai House	Bagel Shop	Ice Cream Shop	Gym / Fitness Center	Grocery Store	Moving Target	Park
1	Jardim Ângela	-23.720471	-46.771372	0	Health & Beauty Service	Hot Dog Joint	Burger Joint	Soccer Field	Mobile Phone Shop	Snack Place	Gym	Bakery	Pastelaria	Coffee Shop
2	Sapopemba	-23.605114	-46.509945	0	Bakery	Gym	Pharmacy	Farmers Market	Gym / Fitness Center	Soccer Stadium	Grocery Store	Athletics & Sports	Bookstore	Supermarket
3	Capão Redondo	-23.665079	-46.780830	0	Bakery	Pet Store	Park	Snack Place	Historic Site	Pharmacy	Brazilian Restaurant	Southern / Soul Food Restaurant	Beer Bar	Bar
4	Jardim São Luís	-23.657646	-46.730065	5	Brazilian Restaurant	Pizza Place	Buffet	Café	Gym / Fitness Center	Bakery	Coffee Shop	Restaurant	Chocolate Shop	BBQ Joint

Work Flow:

Using credentials of Foursquare API features of near-by places of the neighborhoods would be mined. Due to http request limitations the number of places per neighborhood parameter would reasonably be set to 300 and the radius parameter would be set to 1000.

Results Section



```
[22]: map_clusters
```

Map of Clusters in São Paulo:

Conclusion Section

Using the k-means cluster algorithm, we separated the neighborhoods into 6 different clusters and for 96 different lattitude and logitude from dataset, which have very-similar neighborhoods around them.

I feel the results are very accurate with my experience living in São Paulo - Brazil, and they can benefit greatly someone new to the region in choosing a place to live.

Libraries used to Develop the Project:

> Pandas: For creating and manipulating dataframes.
> Numpy: To deal with numbers in general.
> Folium: Python visualization library would be used to visualize the neighborhoods cluster distribution of using interactive leaflet map.
> Scikit Learn: For importing k-means clustering.
> JSON: Library to handle JSON files.
> XML: To separate data from presentation and XML stores data in plain text format.
> Geocoder: To retrieve Location Data.
> Beautiful Soup and Requests: To scrap and library to handle http requests.
> Matplotlib: Python Plotting Module.
> sklearn: Imported k-means