1. Discussion and Background of the Business Problem:

Rio de Janeiro is one of the most visited cities in the Southern Hemisphere and is known for natural settings, Carnival, samba, bossa nova, and balneario beaches such as Copacabana, Ipanema, and Leblon. In addition to the beaches, some of the most famous landmarks include the giant statue of Christ the Redeemer atop Corcovado mountain, named one of the New Seven Wonders of the World. Rio was the host of the 2016 Summer Olympics and the 2016 Summer Paralympics, making the city the first South American and Portuguese-speaking city to ever host the events, and the third time the Olympics were held in a Southern Hemisphere city.

Rio de Janeiro receives more than 1 million tourists every year. Like so many tourist delays and various touristic districts it is difficult for the visitor to decide which neighborhood to stay in Rio. The purpose of this paper is to analyze, only touristic neighborhood, identify the differences, so that tourists can stay in the neighborhood that fits their profile.

So, Where should I stay in Rio? Let's get started !

Target Audience

1. Tourists, who wants to visit Rio and want to stay in the best spot that fits his profile
2. Travel agency, that would like to assist your client to try the best experience possible
3. Municipal government that can do better marketing for different people profiles

2. Data acquisition and sources

2.1 Data sources

The following data was used for this project:

- The best neighborhoods to enjoy Rio de Janeiro
  https://www.feriasbrasil.com.br/rj/riodejaneiro/bairros.cfm

- Rio de Janeiro geo data, shape file with limits of each neighborhoods(geoJson)
  http://www.data.rio/datasets/limite-bairro/data?geometry=-44.313%2C-23.138%2C-42.579%2C-22.695

- Foursquare venue data based on each neighborhoods

2.2 Getting Coordinates of Major Districts

Searching the coordinates of the tourist districts. Using the code snippet as below.

[50]:
```
latitude = []
longitude = []
for ng in df.index:
    address = df['Neighborhood'][ng] + ' , RJ'
    geolocator = Nominatim(user_agent="rj_explorer")
    location = geolocator.geocode(address)
    latitude.append(location.latitude)
    longitude.append(location.longitude)
df['Latitude'] = latitude
df['Longitude'] = longitude
```

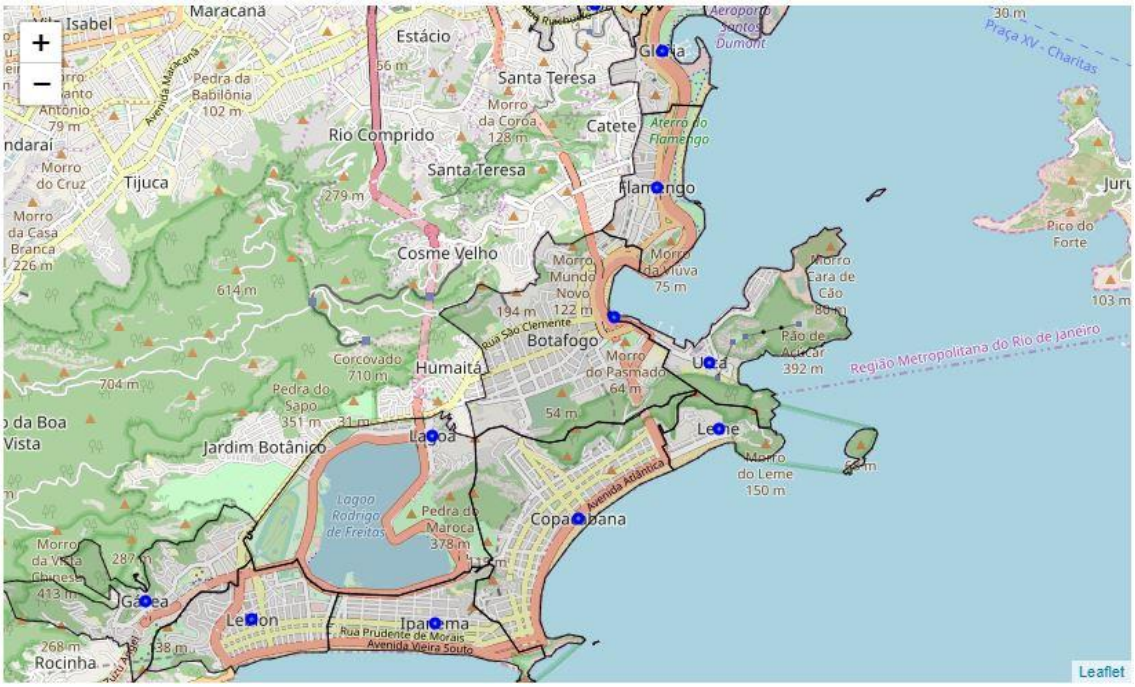
[51]:
```
df.to_csv('rio_tur_geo.csv', index = False)
```

[52]:
```
df = pd.read_csv('rio_tur_geo.csv')
df.head()
```

[52]:

	Neighborhood	Latitude	Longitude
0	Copacabana	-22.971964	-43.184343
1	Leme	-22.961704	-43.166904
2	Lagoa	-22.962466	-43.202488
3	Gávea	-22.981424	-43.238324
4	Leblon	-22.983556	-43.224938

With neighborhood centroids and Rio shape file, we can see limits from each neighborhood as below.



2.3 Using Foursquare location data

For this business problem, I used Foursquare API to retrieve information about avenues around this Districts. From each neighborhood centroid with a limit of 50 venues and 1500 meters of radius we retrieve data in a JSON file an turn it into a data-frame. Below we can see data returned by Foursquare.

```
rio_venues.head()
```

```
(600, 7)
```

[7]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Copacabana	-22.971964	-43.184343	Praia de Copacabana	-22.972441	-43.183436	Beach
1	Copacabana	-22.971964	-43.184343	Windsor California Hotel	-22.972704	-43.185707	Hotel
2	Copacabana	-22.971964	-43.184343	JW Marriott Hotel Rio de Janeiro	-22.972259	-43.185825	Hotel
3	Copacabana	-22.971964	-43.184343	Kopenhagen	-22.970680	-43.185617	Chocolate Shop
4	Copacabana	-22.971964	-43.184343	Santa Satisfação	-22.972035	-43.186719	Bistro

3. Methodology

- Loaded the districts geo data
- Using Foursquare API, got all the needed venues information
- Explore data analysis
- Create clusters of districts based on venues category using K-Means
- Visualize cluster data pointing out their similarities and differences

4. Data exploration and visualization

With venue data and location let's explore this dataset. First let's see what kind of venues is more common in Rio.

```
[12]: rio_sort_df = rio_venues.groupby('Venue Category').count()
      rio_sort_df.sort_values('Neighborhood', ascending=False)
```

[12]:

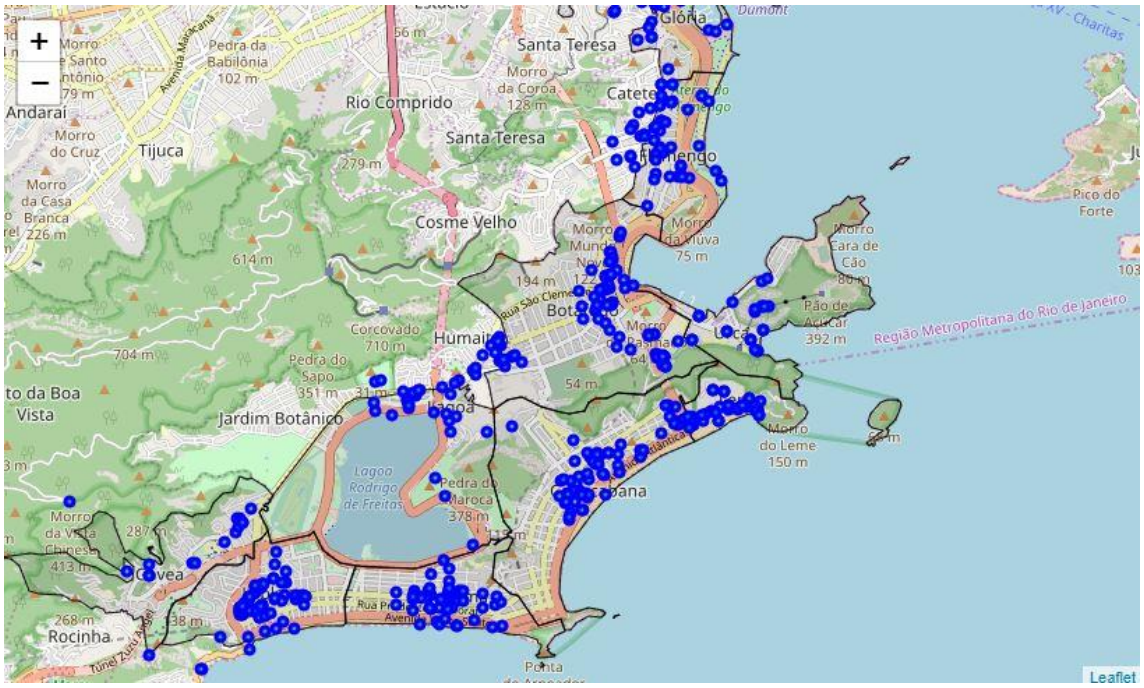
Venue Category	Neighborhood	Neighborhood Latitude	I
Brazilian Restaurant	41	41	
Bar	36	36	
Hotel	25	25	
Coffee Shop	17	17	
Beach	17	17	
Pizza Place	16	16	
Gym / Fitness Center	14	14	
Japanese Restaurant	14	14	
Bookstore	13	13	
Scenic Lookout	12	12	
Italian Restaurant	12	12	
Ice Cream Shop	10	10	
Salad Place	10	10	
Steakhouse	9	9	
Hostel	9	9	

Let's look inside each neighborhood and see what places are common, below we have a sample:

```
[109]: num_top_venues = 5

for hood in rio_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = rio_grouped[rio_grouped['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

```
----Botafogo----
          venue  freq
0         Hotel  0.08
1    Bookstore  0.08
2  Coffee Shop  0.06
3       Hostel  0.06
4  Beer Garden  0.04


----Centro----
                       venue  freq
0       Brazilian Restaurant  0.08
1                  Bookstore  0.08
2                     Church  0.06
3               Salad Place  0.06
4  Middle Eastern Restaurant  0.06


----Copacabana----
        venue  freq
0       Hotel  0.16
1      Bakery  0.06
2         Bar  0.06
3  Beach Bar  0.04
4         Gym  0.04
```

With the above data, we can see that each neighborhood has its own particularities that will be discussed later. Let's look below spatial distribution of venues.



5. Clustering the Districts

Finally, we cluster these neighborhoods based on the venue categories and use K-Means clustering. The expectation would be based on the similarities of venue categories.

[113]:
```
# set number of clusters
kclusters = 5

rio_grouped_clustering = rio_grouped.drop('Neighborhood', 1)

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

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

[113]:
```
array([0, 2, 3, 2, 2, 0, 0, 0, 4, 0], dtype=int32)
```

[114]:
```
# add clustering labels
rio_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

rio_merged = df

rio_merged = rio_merged.join(rio_venues_sorted.set_index('Neighborhood'), on='Neighborhood')

rio_merged.head() # check the last columns!
```

[114]:

	Neighborhood	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
0	Copacabana	-22.971964	-43.184343	3	Hotel	Bakery	Bar	Gym	Beach Bar	Lounge	Salad Place
1	Leme	-22.961704	-43.166904	1	Brazilian Restaurant	Beach	Hotel	Scenic Lookout	Bar	Seafood Restaurant	Pizza Place
2	Lagoa	-22.962466	-43.202488	0	Gym / Fitness Center	Bar	Pizza Place	Farmers Market	Park	Scenic Lookout	Bakery
3	Gávea	-22.981424	-43.238324	0	Brazilian Restaurant	Bar	Scenic Lookout	Pizza Place	Park	Bookstore	Dive Bar

Cluster Analyze:

[25]:
```
rio_merged.loc[rio_merged['Cluster Labels'] == 1, rio_merged.columns[[1] + list(range(4, rio_merged.shape[1]))]]
```

[25]:

	Latitude	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	-22.961704	Brazilian Restaurant	Beach	Hotel	Scenic Lookout	Bar	Seafood Restaurant	Pizza Place	Bookstore	Breakfast Spot	Deli / Bodega
8	-22.954074	Beach	Scenic Lookout	Brazilian Restaurant	Pizza Place	Mountain	Trail	Bar	Steakhouse	Hotel	Ice Cream Shop

[26]:
```
rio_merged.loc[rio_merged['Cluster Labels'] == 2, rio_merged.columns[[1] + list(range(4, rio_merged.shape[1]))]]
```

[26]:

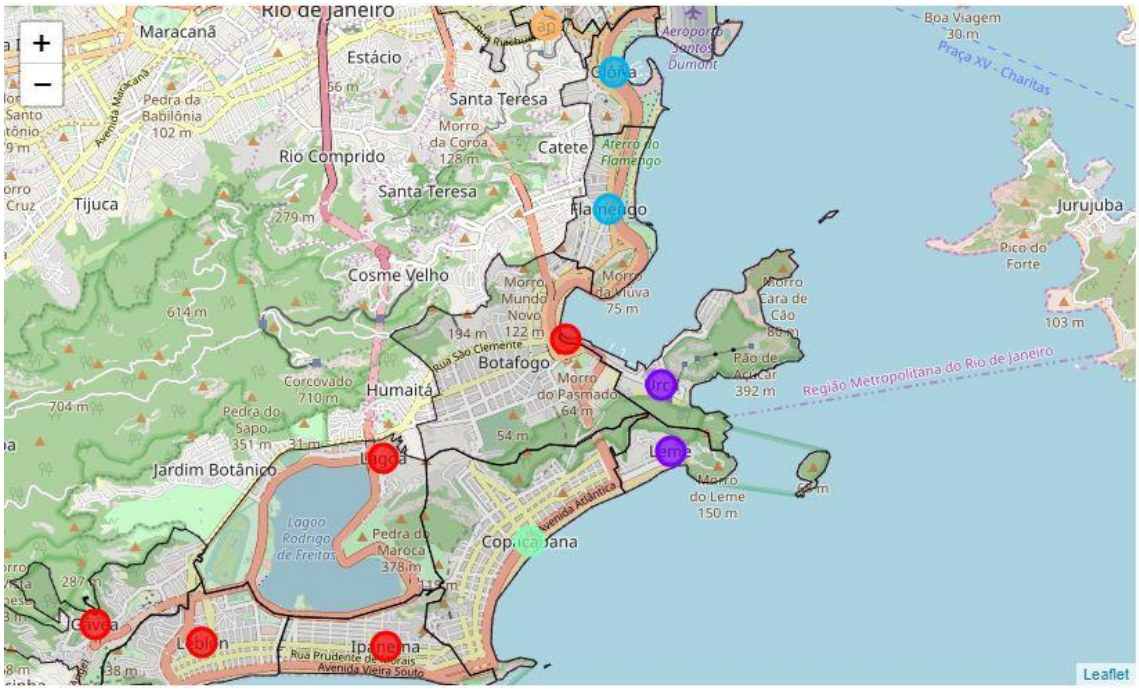
	Latitude	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
9	-22.918323	Historic Site	Music Venue	Coffee Shop	Theater	Garden	Movie Theater	Hostel	History Museum	Bar	Brazilian Restaurant
10	-22.904393	Brazilian Restaurant	Bookstore	Coffee Shop	Church	Middle Eastern Restaurant	Salad Place	Music Venue	Café	Italian Restaurant	Tram Station
11	-22.933984	Brazilian Restaurant	Japanese Restaurant	Coffee Shop	Churrascaria	Gym / Fitness Center	Track	Fruit & Vegetable Store	Park	Movie Theater	Cocktail Bar

[27]:
```
rio_merged.loc[rio_merged['Cluster Labels'] == 3, rio_merged.columns[[1] + list(range(4, rio_merged.shape[1]))]]
```

[27]:

	Latitude	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	-22.971964	Hotel	Bakery	Bar	Gym	Beach Bar	Lounge	Salad Place	Chocolate Shop	Churrascaria	Resort

We can represent these five clusters in a map using Folium library as below.



Analyzing clusters, let us focus only in differences between them:

Red: Park, gym, fitness center

Cyan: Beach bar, Gym, Resorts

Purple: Scenic Lookout, Mountain, Trail

Blue: Historic Site, Theater, History Museum

Brown: Music Venue, Nightclubs

Each Cluster has a lot of Bar, Hotel, Hostel, Café.

6. Results and Discussion

At the end of our analysis, we can see that each cluster has its similarities and peculiarities. Rio proves to be a city for all tastes, offering various attractions for different audiences.

Each cluster can meet a different tourist profile, and now you know which is the best place to stay, for so that you get off as little as possible, but not eliminating the possibility of visiting other regions of the city.

What are you most looking for in the city of Rio?

Adventure?

Beaches?

Natural beauties and stunning views?

History and theater?

Music and nightclubs?

Rio offers all this and much, but now, where to stay to make the most of your stay!

7. Conclusion

Finally to conclude this paper. We made use of some frequently uses python libraries to get data, analyze and view, we also use Foursquare API to explore avenues in a neighborhood for city of Rio.

There is a potential for this kind of analysis in real life business. One way to improve this report would be to include ratings of the venues and collect more data with a paid Foursquare account.