The Battle of the Neighborhoods: Final Report

Introduction/Business Problem

The leadership group of a Gaming group wants to set up a gaming arcade in the United States. They want to figure out where the best place would be to install an arcade. The group wants to maximize the number of customers interested in coming to the arcade, and find possible cities that demonstrate high values (weights) for an arcade.

Some things we will need to consider:

- Firstly, we should know why to start off with a place in particular?
- If a place is chosen, we must have a very clear understanding of who the potential customers be?
- We need to have a clear understanding of the customers spending patterns in that locality etc.
- Last but not the least, we need to take care of the legal aspects involved in that locality.

Data

Tools needed for data:

- Foursquare API
- IBM Watson Account
- List of United States cities by population
- Jupyter Notebooks

The data will be extracted from multiple sources:

- https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population
- Foursquare API data on venues.

A brief on the data

Here's some brief about the locations in particular(source: Wikipedia):

As defined by the United States Census Bureau, an "incorporated place" includes a variety of designations, including city, town, village, borough, and municipality. A few exceptional census-designated places (CDPs) are also included in the Census Bureau's listing of incorporated places. Consolidated city-counties represent a distinct type of government that includes the entire population of a county, or county equivalent. Some consolidated city-counties, however, include multiple incorporated places.

About the State income levels: State income levels and income data for the United States as a whole are included for comparison. Note that county-equivalents in Louisiana are called "parishes" and in Alaska are called in "boroughs," and also that in Alaska census areas in the Unorganized Borough are county-equivalents. For states where independent cities are county-equivalents, the word "city" is included to identify the independent cities and to differentiate them from counties with identical names; the counties with the identical names have the word "county" following them. The word "county" is included in the names of counties that have names identical to the names of U.S. states or cities to differentiate them.

Methodology

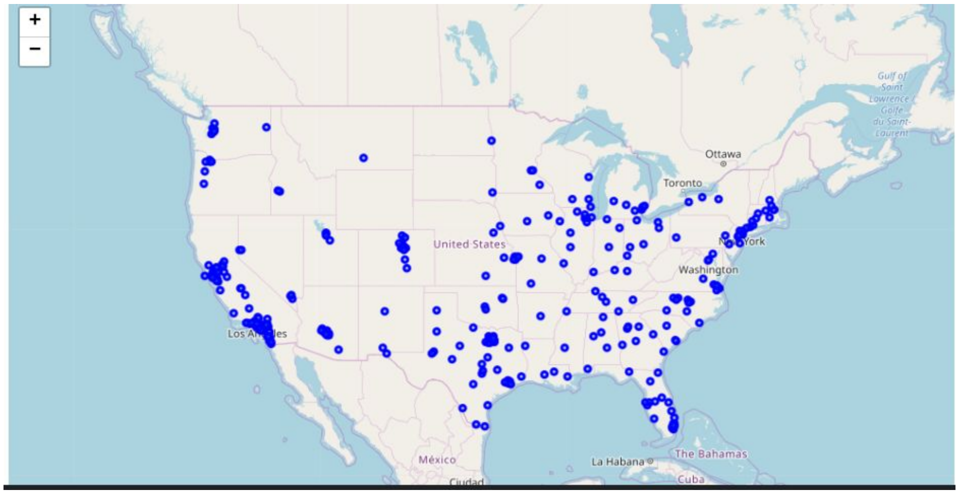
In order to do the analysis and suggest the best location, the following are the steps needed:

Importing all the necessary libraries needing to do the analysis:

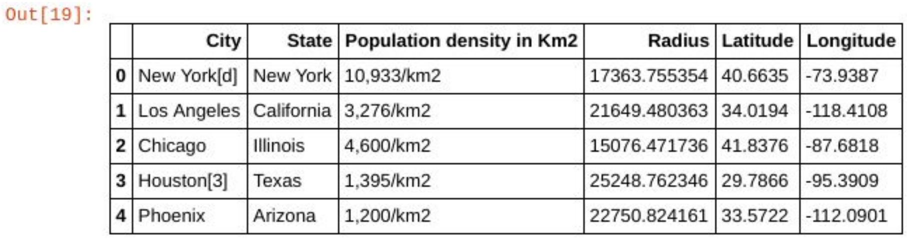
```
import numpy as np # library to handle data in a vectorized manner
import pandas as pd # library for data analysis pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
import json # library to handle JSON files
!conda install -c conda-forge geopy --yes # uncomment this line if you haven't completed the Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude and longitude values
import requests # library to handle requests from pandas.io.json import json_normalize # transform JSON file into a
pandas dataframe
Matplotlib and associated plotting modules import matplotlib.cm as cm import matplotlib.colors as colors
import k-means from clustering stage from sklearn.cluster import KMeans
for webscraping import Beautiful Soup from bs4 import BeautifulSoup
import xml
!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't completed the Foursquare API
lab import folium # map rendering library
print('Libraries imported.')
```

Approach

- The Wikipedia page (https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population) was scraped using the BeautifulSoup library to build a pandas dataframe listing the cities, states, coordinates, area and population density. The dataframe was cleaned and processed appropriately. A map was created to show the cities.



Here is an example of some cities and their population densities:



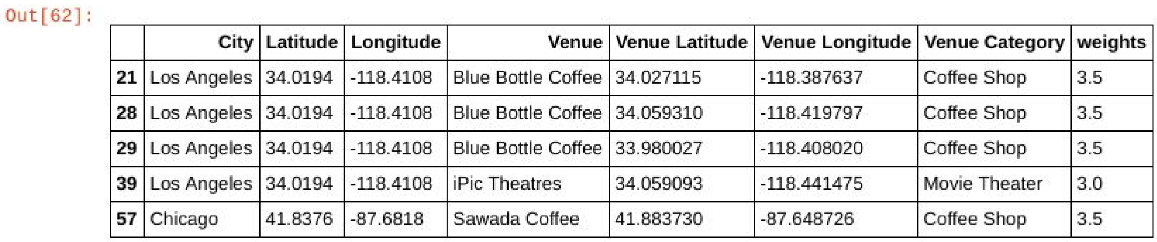
Out[19]:

	City	State	Population density in Km2	Radius	Latitude	Longitude
0	New York[d]	New York	10,933/km2	17363.755354	40.6635	-73.9387
1	Los Angeles	California	3,276/km2	21649.480363	34.0194	-118.4108
2	Chicago	Illinois	4,600/km2	15076.471736	41.8376	-87.6818
3	Houston[3]	Texas	1,395/km2	25248.762346	29.7866	-95.3909
4	Phoenix	Arizona	1,200/km2	22750.824161	33.5722	-112.0901

- The Foursquare API is then used to get the venues in each city of United States

- Based on the categories of each venue as decided by the leadership group, we have assigned weights to each of them and got the city that has the maximum weight.
- The weights are determined by the population density and the number of venues in that area. For venues example: ‘Movie Theater':3,'Beach':3,'Concert Hall':2.5,'Playground':3,'Coffee Shop':3.5,'Food Court':4,'Nightclub':4,'Toy / Game Store':4.5,'Theme Park Ride / Attraction':4,'Pub':4.

Here is an example of some of the cities:

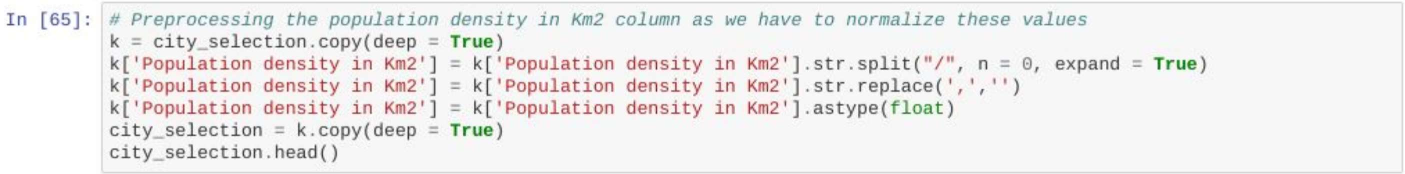


Out[62]:

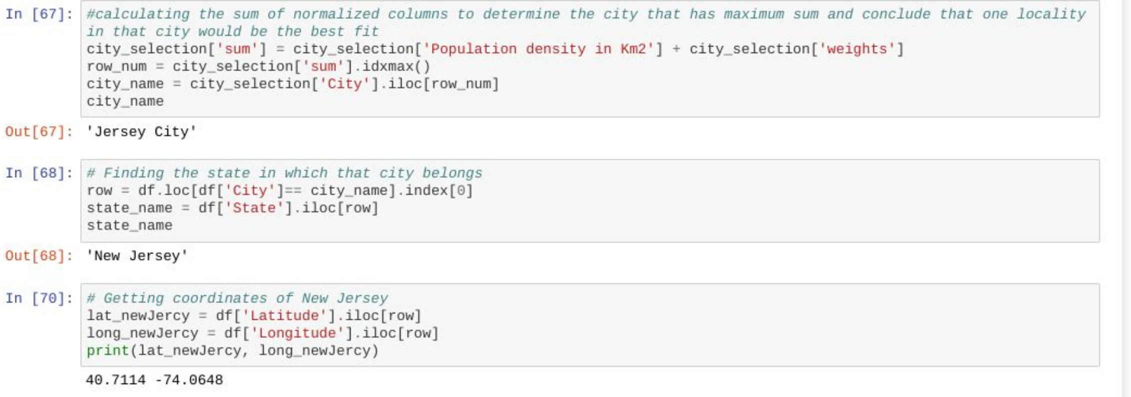
	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	weights
21	Los Angeles	34.0194	-118.4108	Blue Bottle Coffee	34.027115	-118.387637	Coffee Shop	3.5
28	Los Angeles	34.0194	-118.4108	Blue Bottle Coffee	34.059310	-118.419797	Coffee Shop	3.5
29	Los Angeles	34.0194	-118.4108	Blue Bottle Coffee	33.980027	-118.408020	Coffee Shop	3.5
39	Los Angeles	34.0194	-118.4108	iPic Theatres	34.059093	-118.441475	Movie Theater	3.0
57	Chicago	41.8376	-87.6818	Sawada Coffee	41.883730	-87.648726	Coffee Shop	3.5

- Once the city is finalized, we again use FourSquare API to get the venues within that city and assign weights to each category.

- We then use K means to first find the city. We cluster the venues based on the category and get the coordinates of the cluster that has maximum weight which is also our preferred location to setup a gaming arcade.



```
In [65]: # Preprocessing the population density in Km2 column as we have to normalize these values
         k = city_selection.copy(deep = True)
         k['Population density in Km2'] = k['Population density in Km2'].str.split("/", n = 0, expand = True)
         k['Population density in Km2'] = k['Population density in Km2'].str.replace(',','')
         k['Population density in Km2'] = k['Population density in Km2'].astype(float)
         city_selection = k.copy(deep = True)
         city_selection.head()
```

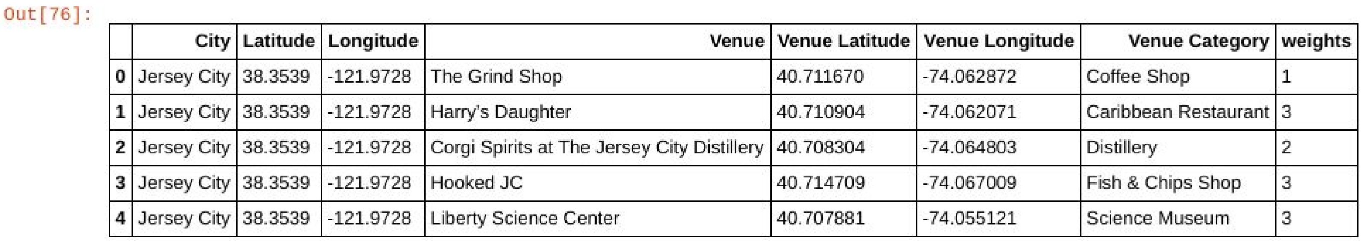


```
In [67]: #calculating the sum of normalized columns to determine the city that has maximum sum and conclude that one locality
         in that city would be the best fit
         city_selection['sum'] = city_selection['Population density in Km2'] + city_selection['weights']
         row_num = city_selection['sum'].idxmax()
         city_name = city_selection['City'].iloc[row_num]
         city_name

Out[67]: 'Jersey City'

In [68]: # Finding the state in which that city belongs
         row = df.loc[df['City']== city_name].index[0]
         state_name = df['State'].iloc[row]
         state_name

Out[68]: 'New Jersey'

In [70]: # Getting coordinates of New Jersey
         lat_newJercy = df['Latitude'].iloc[row]
         long_newJercy = df['Longitude'].iloc[row]
         print(lat_newJercy, long_newJercy)

         40.7114 -74.0648
```

Results

Based on the analysis we have done, the following is the result:

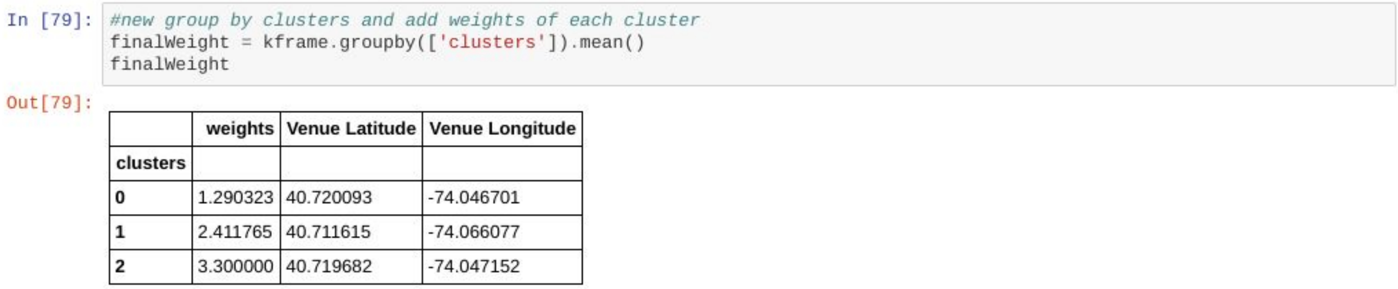
Jersey City was the best city for a gaming arcade to be installed. It had the highest sum of weights (population density, venues). Shown below are some areas within Jersey City that would be ideal.



Out[76]:

	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	weights
0	Jersey City	38.3539	-121.9728	The Grind Shop	40.711670	-74.062872	Coffee Shop	1
1	Jersey City	38.3539	-121.9728	Harry's Daughter	40.710904	-74.062071	Caribbean Restaurant	3
2	Jersey City	38.3539	-121.9728	Corgi Spirits at The Jersey City Distillery	40.708304	-74.064803	Distillery	2
3	Jersey City	38.3539	-121.9728	Hooked JC	40.714709	-74.067009	Fish & Chips Shop	3
4	Jersey City	38.3539	-121.9728	Liberty Science Center	40.707881	-74.055121	Science Museum	3

We use k means to find the best region (lat, long) within Jersey City for the gaming arcade. Shown below are the results



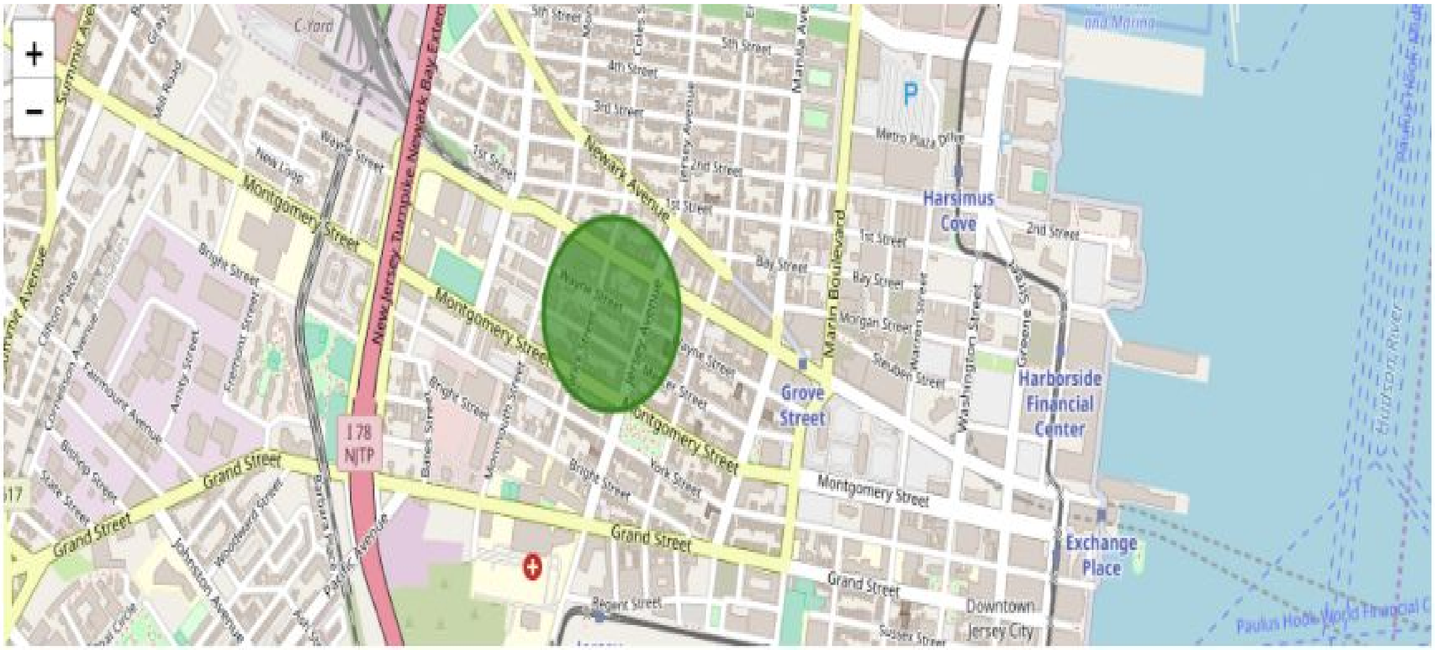
In [79]:

```
#new group by clusters and add weights of each cluster
finalWeight = kframe.groupby(['clusters']).mean()
finalWeight
```

Out[79]:

	weights	Venue Latitude	Venue Longitude
clusters			
0	1.290323	40.720093	-74.046701
1	2.411765	40.711615	-74.066077
2	3.300000	40.719682	-74.047152

Lastly, we create a map based on the best region to install a gaming arcade. The map is shown below.



Discussion

This project can be enhanced by considering many more attributes to define the weights and do the analysis and also by extending the LIMIT and Radius of the search that we are giving to extract the number of venues. As we have an API limit in the free trial of foursquare API we had to limit our search within a small Radius. In the FourSquare API, we have queried the Venues of a locality by specifying the LIMIT and Radius of our choice. We have chosen less LIMIT as the number of API calls that can be done using a free account in Four Square are less. We can increase the limit for more accurate results.

In the venue categories we are choosing only few out of 2000 that are available to give weights and identify the best cluster. Hence, assigning weights must be done relatively for each category and then considering more number of venue categories would actually yield better output.

This was a challenging problem to answer. There are plenty of other variables to consider for this project. Even though population density and venue options were used to decide the location, people might not want to spend money at a gaming arcade. Overall, these two weighted variables at least give the leadership group a better idea of where to install a new gaming arcade, and a more predictive outcome for success.

Conclusion

The purpose of this project was to figure out the best city in the United States for a leadership group of a gaming company to install a new gaming arcade. The leadership group also wanted to know exactly where within that city would be ideal for business. We extracted data from wikipedia on cities in the US, created a map, utilized venue data on FourSquare, and used K means clustering to identify the city and the area in the city.

The results show that Jersey City, New Jersey had the highest sum of weighted values. Those weighted values were population density and the number of venues that were similar to a gaming arcade. Even more specifically, the best area in Jersey City to install a gaming arcade is between Groove Street and Grand Street. The amount of people in that area along with how many venues similar to gaming would bring a high potential for foot-traffic and possible business.