Battle of the Neighborhoods – IBM Capstone Project

Where to develop a Karaoke Bar

Introduction

A small investment company inquires where they should best allocate their resources to develop a Karaoke-themed bar. They are looking for an area where starting a new business will attract the right customers and generate the highest revenue. This problem is useful for any business looking to see which markets to develop a business in for various avenues.

Background

They've been racking their brains trying to pinpoint a city, but the group just can't come to a consensus. Luckily, they heard about you through a mutual colleague that you specialize in data analytics and could come up with best possible solutions based on data. What they're looking for. Generally speaking, they're looking to define customers spending patterns in various localities to determine the best U.S. based location to set up their karaoke business.

Business Problem

In order to help the investment company from maximizing profits and minimizing loss, I will definitely be incorporating data from websites to define which areas are the most densely populated, have higher than average incomes, use the Foursquare API to obtain venues in various U.S. cities, then categorize the venues based on proportional weights to designate value. Once the city(s) is determined, we can narrow in on distinct neighborhoods and obtain visual representation of the coordinates. In particular, we want to focus on the venues listed for each locality (using Foursquare API) and determine why a certain location could be considered more appealing.

Data

Data will be scraped from (https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population) and (https://en.wikipedia.org/wiki/List_of_United_States_counties_by_per_capita_income) to obtain a pandas dataframe using the 'BeautifulSoup' library discussed in the modules. Clean the data and pre-process for exploratory data analysis. The two Wikipedia websites should provide enough information to determine which location would provide the best potential return on investment for the company. Once the data from the Wikipedia pages are clean and populated into dataframes, I'll be implementing the Foursquare API to determine relevant venues near areas.

Methodology(s)

Briefly discussed in the business problem section, we will need to analyst and suggest the best possible location to develop the Karaoke business (based on population and income).

- List_of_United_States_cities_by_population was scraped using the BeautifulSoup library to build a pandas dataframe. The result data frame lists the cities, states, coordinates, area and population density

[4]:

	Rank	City	State	del1	del2	del3	Sq.Area	del5	population density in Sq Mi	Population density in Km2	Location
0	1	New York[d]	New York	8,398,748	8,175,133	+2.74%	301.5 sq mi	780.9 km2	28,317/sq mi	10,933/km2	40°39′49″N 73°56′19″W / 40.6635°N 73.9387°W...
1	2	Los Angeles	California	3,990,456	3,792,621	+5.22%	468.7 sq mi	1,213.9 km2	8,484/sq mi	3,276/km2	34°01′10″N 118°24′39″W / 34.0194°N 118.4108°...
2	3	Chicago	Illinois	2,705,994	2,695,598	+0.39%	227.3 sq mi	588.7 km2	11,900/sq mi	4,600/km2	41°50′15″N 87°40′54″W / 41.8376°N 87.6818°W...
3	4	Houston[3]	Texas	2,325,502	2,100,263	+10.72%	637.5 sq mi	1,651.1 km2	3,613/sq mi	1,395/km2	29°47′12″N 95°23′27″W / 29.7866°N 95.3909°W...
4	5	Phoenix	Arizona	1,660,272	1,445,632	+14.85%	517.6 sq mi	1,340.6 km2	3,120/sq mi	1,200/km2	33°34′20″N 112°05′24″W / 33.5722°N 112.0901°...

[5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 314 entries, 0 to 313
Data columns (total 11 columns):
Rank                            314 non-null object
City                            314 non-null object
State                           314 non-null object
del1                            314 non-null object
del2                            314 non-null object
del3                            314 non-null object
Sq.Area                         314 non-null object
del5                            314 non-null object
population density in Sq Mi    314 non-null object
Population density in Km2       314 non-null object
Location                        314 non-null object
dtypes: object(11)
memory usage: 27.1+ KB
```

[ ]: Determine city radius using Sq.Area

[6]:
```
new = df["Sq.Area"].str.split("s", n=1, expand = True)
new = new[0].str.replace(u'\xa0',u'')
df["Sq.Area"] = new.str.replace(',','')
df["Sq.Area"] = df["Sq.Area"].astype(float)
df["Radius"] = np.sqrt(df["Sq.Area"])
```

- List_of_United_States_counties_by_per_capita_income was also scraped using the BeautifulSoup library to build a pandas dataframe listing the cities, states and capita income

Out[20]:

	Rank	Country-equivalent	State	Per capita income	del2	del3	Population	del5
0	1	New York County	New York	$62,498</td> <td>$69,659	$84,627</td> <td>1,605,272</td> <td>736,192</td> </tr> <tr> <th>1</th> <td>2</td> <td>Arlington</td> <td>Virginia</td> <td>$62,018	$103,208</td> <td>$139,244	214,861	94,454
2	3	Falls Church City	Virginia	$59,088</td> <td>$120,000	$152,857</td> <td>12,731</td> <td>5,020</td> </tr> <tr> <th>3</th> <td>4</td> <td>Marin</td> <td>California</td> <td>$56,791	$90,839</td> <td>$117,357	254,643	102,912
4	5	Alexandria City	Virginia	$54,608</td> <td>$85,706	$107,511	143,684	65,369	

Dropping unnecessary columns like Rank, del2, del3, del5 from the Table we extracted from the webpage that has population and percapita income

In [21]:
```
df_state.drop(columns = ['Rank','del2', 'del3', 'del5'], axis = 1, inplace = True)
```

In [22]:
```
df_state.head()
```

Out[22]:

	Country-equivalent	State	Per capita income	Population
0	New York County	New York	$62,498	1,605,272
1	Arlington	Virginia	$62,018</td> <td>214,861</td> </tr> <tr> <th>2</th> <td>Falls Church City</td> <td>Virginia</td> <td>$59,088	12,731
3	Marin	California	$56,791</td> <td>254,643</td> </tr> <tr> <th>4</th> <td>Alexandria City</td> <td>Virginia</td> <td>$54,608	143,684

- The Foursquare API is utilized to obtain venues for every U.S. city and based on the venue categories, we can process the data to give the venue weights for decision-making.

Out[199]:

	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	we
21	Los Angeles	34.0194	-118.4108	Blue Bottle Coffee	34.027115	-118.387637	Coffee Shop	3.5
39	Los Angeles	34.0194	-118.4108	iPic Theatres	34.059093	-118.441475	Movie Theater	3.0
58	Chicago	41.8376	-87.6818	Sawada Coffee	41.883730	-87.648726	Coffee Shop	3.5
77	Houston[7]	29.7866	-95.3909	Boomtown Coffee	29.802849	-95.400855	Coffee Shop	3.5
100	Philadelphia[8]	40.0094	-75.1333	Amalgam Comics & Coffeehouse	39.985120	-75.124364	Coffee Shop	3.5

- Based on a normalization using sklearn, we've determined the city with maximum weighted value was New Jersey (ehh…haven't heard great things about the Garden State, but maybe the Karaoke business could liven up the city).

```
city_selection['sum'] = city_selection['Population density in Km2'] + city_selection['weights']
row_num = city_selection['sum'].argmax()
city_name = city_selection['City'].iloc[row_num]
city_name
```

```
/opt/conda/envs/DSX-Python35/lib/python3.5/site-packages/ipykernel/__main__.py:3: FutureWarning: 'argmax'
is deprecated. Use 'idxmax' instead. The behavior of 'argmax' will be corrected to return the positional m
aximum in the future. Use 'series.values.argmax' to get the position of the maximum now.
  app.launch_new_instance()
```

34]: 'Jersey City'

- Lastly, obtain coordinates of the cluster (similar to earlier lesson) giving maximum weighted value for established the preferred location. This can be done through K-means clustering.

RESULTS

Using the Folium map output and the coordinates for the maximum weight value, we can plot the suggested location:

The location is suggested between Grove and Grand street, with maximum foot traffic and because we have determined the per capita income to be sufficient ($50349 based on python output), I can suggest

to the investment company the city of New Jersey may be the most suitable location for profitable business.

DISCUSSION

Given the influx of people in the area and the sufficient per capita income, we can reliably suggest to the investment company to develop their karaoke business in that area. If the Foursquare API allowed for more aggressive search results (> than 1000), we probably would not have choices limited to New Jersey. Perhaps a different machine learning method could be implemented to make more efficient use of time and reliability (SVMs?). The result is satisfactory given the limitations and it's definitely fascinating how machine learning and the use of open-source data can retrieve suggestive predictions.

CONCLUSION

I would have liked to done the project with more API calls and a larger database, but regardless of the constraints, I think the investment company will be thrilled to not have to argue which location is best suitable for their karaoke business. Perhaps L.A. or San Francisco would provide a better locale, but perhaps not, given it already has plenty of karaoke bars and the cost to rent a business there would be harder to maximize profits. N.J. it is!