CAPSTONE PROJECT:
BATTLE OF THE NEIGHBORHOODS

Recommendation for Gym / Fitness Center installation

Final Assignment: Capstone Project: Battle of Neighborhoods

NAME: AKSHAY SINGH
EMAIL: mailforakshaysingh@gmail.com

Contents

INTRODUCTION

Taking care of the body has become one of the biggest concerns of our century. Due the increased demand, the number of gym centers is increasing, which is leading to a tough competition.

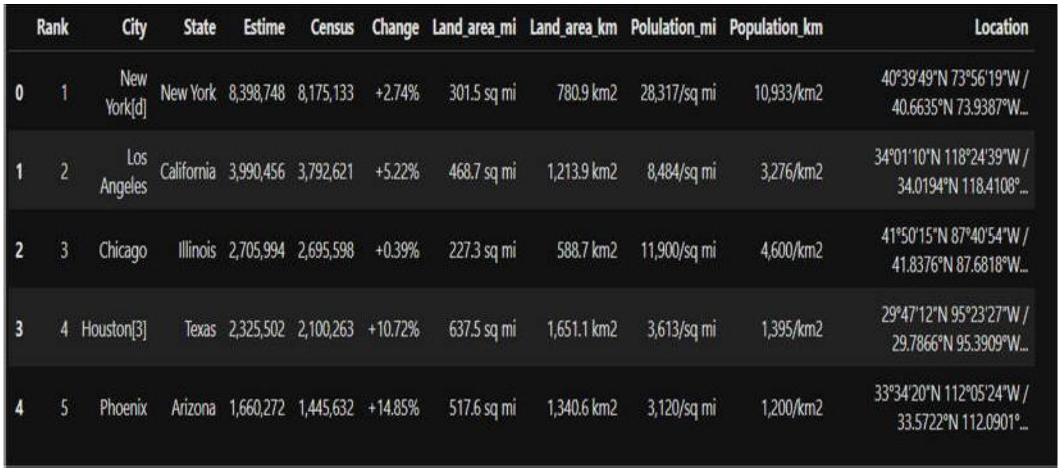
The owner of one Gym / Fitness center in France named Armando, wants to settle in the United States. Due to USA high diversity and very large size, he asked for help in order to find the best place for the installation of his Gym / Fitness center. Building a system for the best location recommendation would help Armando to minimize the competition and have a high income. The best locality is defined bases on the following criteria:

- Location with high average income.
- Location with high population rate.
- Near activity area such as park, playground etc.
- Near residential district, university/school and offices.
- low amount of competition (less or no gym / fitness centers around)

DATA

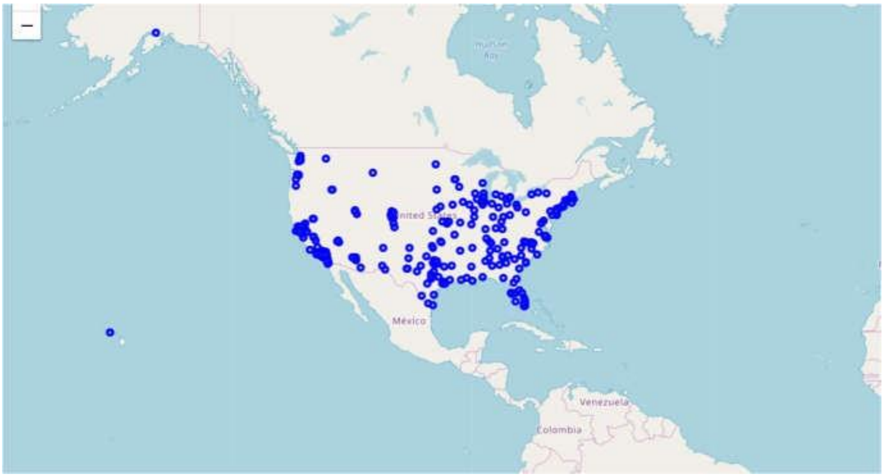
In order to help Armando to find the best location for his Gym / Fitness center, we will consider to access the following data:

- List of all the cities in United States with population density and coordinates
  https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population
  It is the list of the 314 incorporated places in the United States with a population of at least 100,000 on July 1, 2018, as estimated by the United States Census Bureau
  - ✓ Rank: The city rank by population as of July 1, 2018, as estimated by the United States Census Bureau
  - ✓ City: The city's name
  - ✓ State: The name of the state in which the city lies
  - ✓ 2018 Estime: The city population as of July 1, 2018, as estimated by the United States Census Bureau
  - ✓ 2010 Census: The city population as of April 1, 2010, as enumerated by the 2010 United States Census
  - ✓ Change: The city percent population change from April 1, 2010, to July 1, 2018
  - ✓ 2016 Land area: The city land area as of January 1, 2016
  - ✓ 2016 Population Density: The city population density as of July 1, 2016 (residents per unit of land area)
  - ✓ The city latitude and longitude coordinates



	Rank	City	State	Estime	Census	Change	Land_area_mi	Land_area_km	Polulation_mi	Population_km	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°...

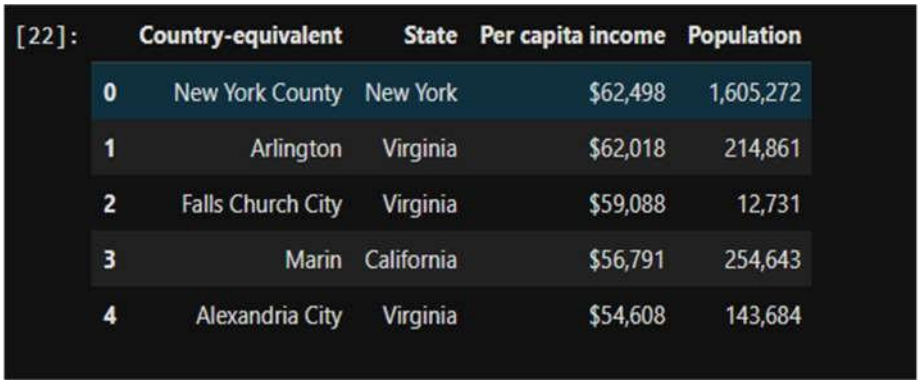
Plotting of all the cities of USA that we have extracted



- List of all the cities in United States with Per Capita Income_ https://en.wikipedia.org/wiki/List_of_United_States_counties_by_per_capita_income

This is a list of United States counties by per capita income. Data for the 50 states and the District of Columbia is from the 2009-2013 American Community Survey 5-Year Estimates; data for Puerto Rico is from the 2013-2017 American Community Survey 5-Year estimates, and data for the other U.S. territories is from the 2010 U.S.

✓ Country
✓ State federal district
✓ Per capita income
✓ Median household income
✓ Median family income
✓ Population
✓ Number of households



[22]:

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

Furthermore, we will use Foursquare (http://www.foursquare.com/) to get the venues in each city of United State:

- The high school venues of the Localities.
- The universities venues of the Localities.
- The offices venues of the Localities.
- The Gym / fitness of the localities.
- The park, playground of the localities.

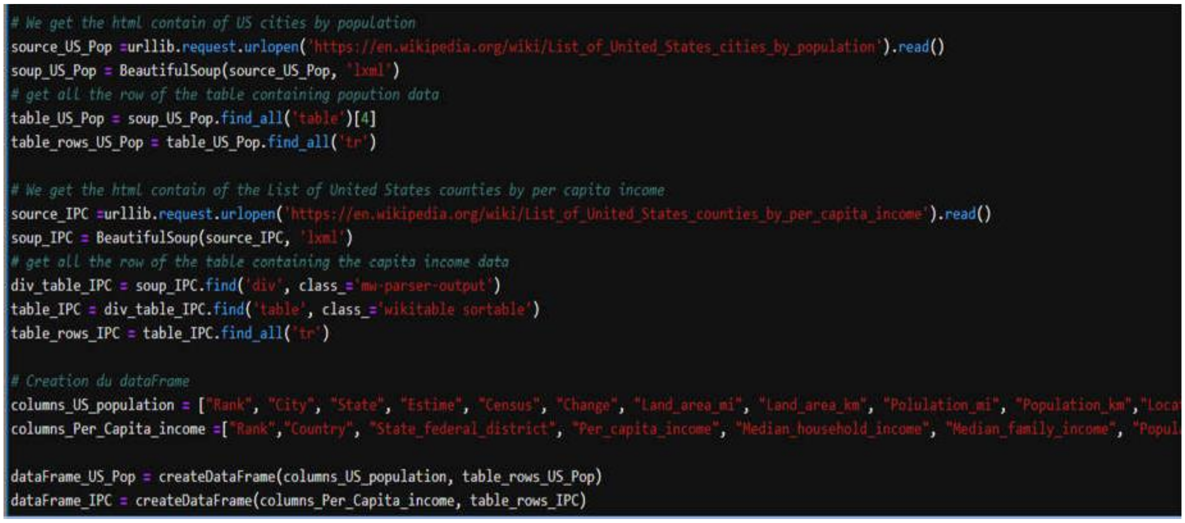
The following information are retrieved:

- City
- City coordinate: Latitude and Longitude
- Venue name
- Venue coordinate: Latitude and Longitude
- Venue category name

METHODOLOGY

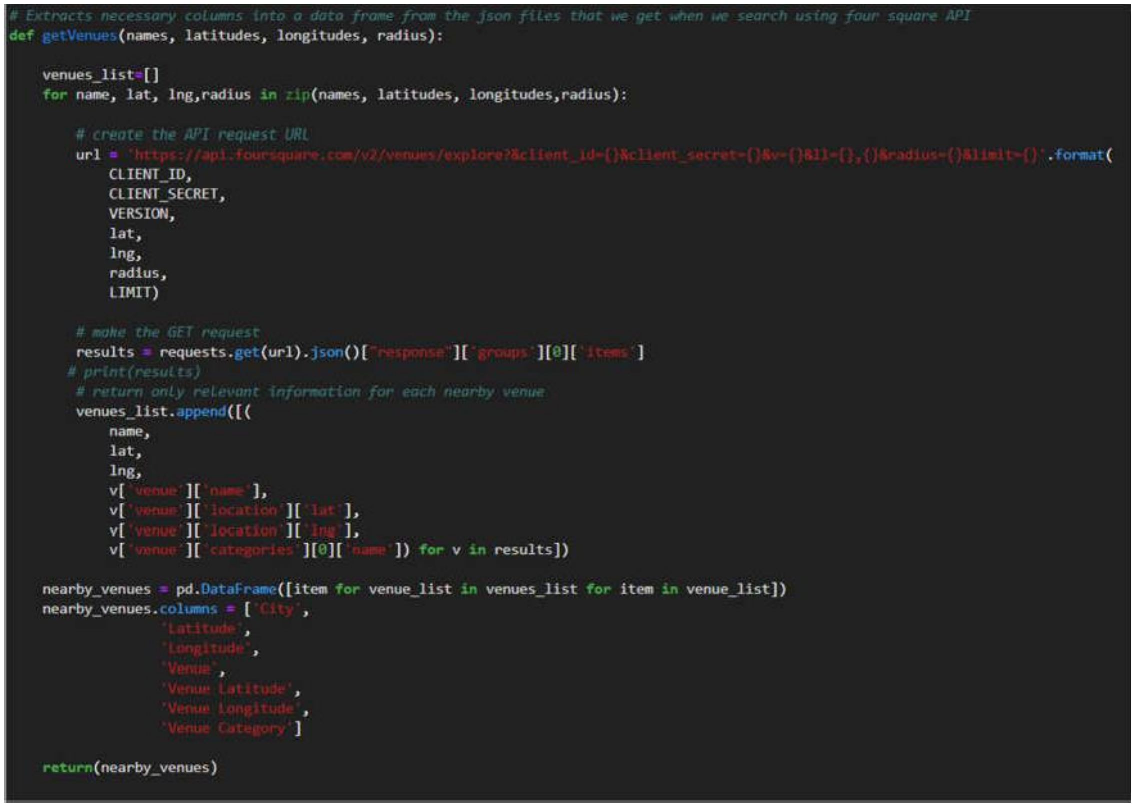
1. **Data collection**:

- BeautifulSoup, a Python library for pulling data out of HTML and XML files is used to get United States cities by population data from Wikipedia as well as United States counties by per capita income data.



```
# We get the html contain of US cities by population
source_US_Pop =urllib.request.urlopen('https://en.wikipedia.org/wiki/List_of_United_States_cities_by_population').read()
soup_US_Pop = BeautifulSoup(source_US_Pop, 'lxml')
# get all the row of the table containing popution data
table_US_Pop = soup_US_Pop.find_all('table')[4]
table_rows_US_Pop = table_US_Pop.find_all('tr')

# We get the html contain of the List of United States counties by per capita income
source_IPC =urllib.request.urlopen('https://en.wikipedia.org/wiki/List_of_United_States_counties_by_per_capita_income').read()
soup_IPC = BeautifulSoup(source_IPC, 'lxml')
# get all the row of the table containing the capita income data
div_table_IPC = soup_IPC.find('div', class_='mw-parser-output')
table_IPC = div_table_IPC.find('table', class_='wikitable sortable')
table_rows_IPC = table_IPC.find_all('tr')

# Creation du dataFrame
columns_US_population = ["Rank", "City", "State", "Estime", "Census", "Change", "Land_area_mi", "Land_area_km", "Polulation_mi", "Population_km","Loca
columns_Per_Capita_income =["Rank","Country", "State_federal_district", "Per_capita_income", "Median_household_income", "Median_family_income", "Popul

dataFrame_US_Pop = createDataFrame(columns_US_population, table_rows_US_Pop)
dataFrame_IPC = createDataFrame(columns_Per_Capita_income, table_rows_IPC)
```

- The Foursquare API is also used to get the venues in each city of United State, based on the categories of each venue as decided by the Armando, we have assigned weights to each of them



```
# Extracts necessary columns into a data frame from the json files that we get when we search using four square API
def getVenues(names, latitudes, longitudes, radius):

    venues_list=[]
    for name, lat, lng,radius in zip(names, latitudes, longitudes,radius):

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

        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']
       # print(results)
        # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

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

    return(nearby_venues)
```

2. **Data Processing:**

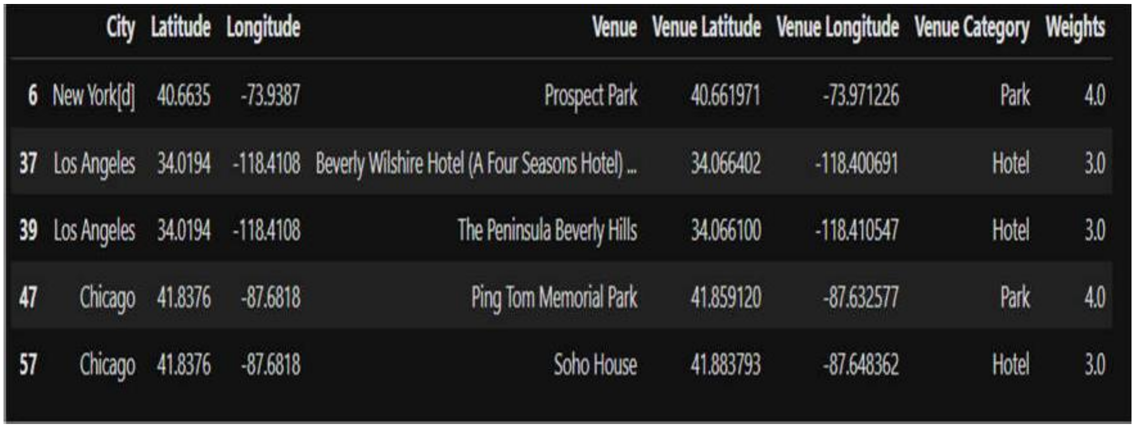
   a. **Data cleaning, formatting and regrouping**:

   The retrieved data contains some un-wanted entries and needs some cleanup. Therefore:

   - We drop unnecessary columns: information like Rank, Estime, Census, Change, Land Area, are useless for this study, so we remove them from the United State population table. From the United States counties by per capita income table, we removed Rank, Median household income, Median family income and Number household.
   - Adding geographical coordinates of each town location: the latitude and the longitude of every city were extracted and assigned
   - Fix data types: Latitude and longitude were converted into float, as they are giving in string in the retrieved data form Wikipedia.

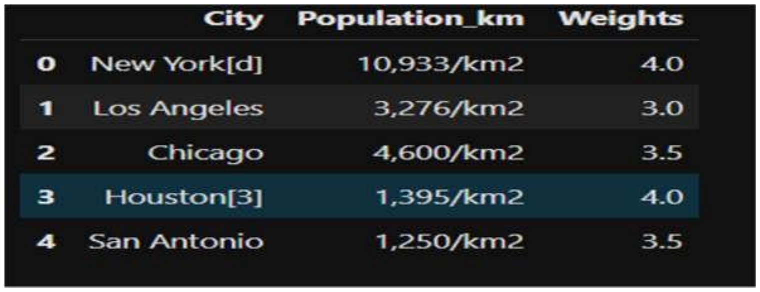
METHODOLOGY

- Preprocessing the population density in Km2 column as we have to normalize these values.
- For each category of venue obtained with the Foursquare API, a weight (or penalty) has been defined according to what Armando considers the most important.
  - ✓ Gym, Gym / Fitness Center, Climbing Gym, Gymnastics Gym and Spa have been weighted with -1, since Armando wants to avoid concurrence
  - ✓ Neighborhood is weighted to 6 as Armando priority is to be closed to neighborhood since it represented the main source of revenue for his center
  - ✓ Playground and park are weighted to 4 respectively as the owner wants to be closed to activities areas
  - ✓ Hotel is weighted to 3



	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Weights
6	New York[d]	40.6635	-73.9387	Prospect Park	40.661971	-73.971226	Park	4.0
37	Los Angeles	34.0194	-118.4108	Beverly Wilshire Hotel (A Four Seasons Hotel) ...	34.066402	-118.400691	Hotel	3.0
39	Los Angeles	34.0194	-118.4108	The Peninsula Beverly Hills	34.066100	-118.410547	Hotel	3.0
47	Chicago	41.8376	-87.6818	Ping Tom Memorial Park	41.859120	-87.632577	Park	4.0
57	Chicago	41.8376	-87.6818	Soho House	41.883793	-87.648362	Hotel	3.0

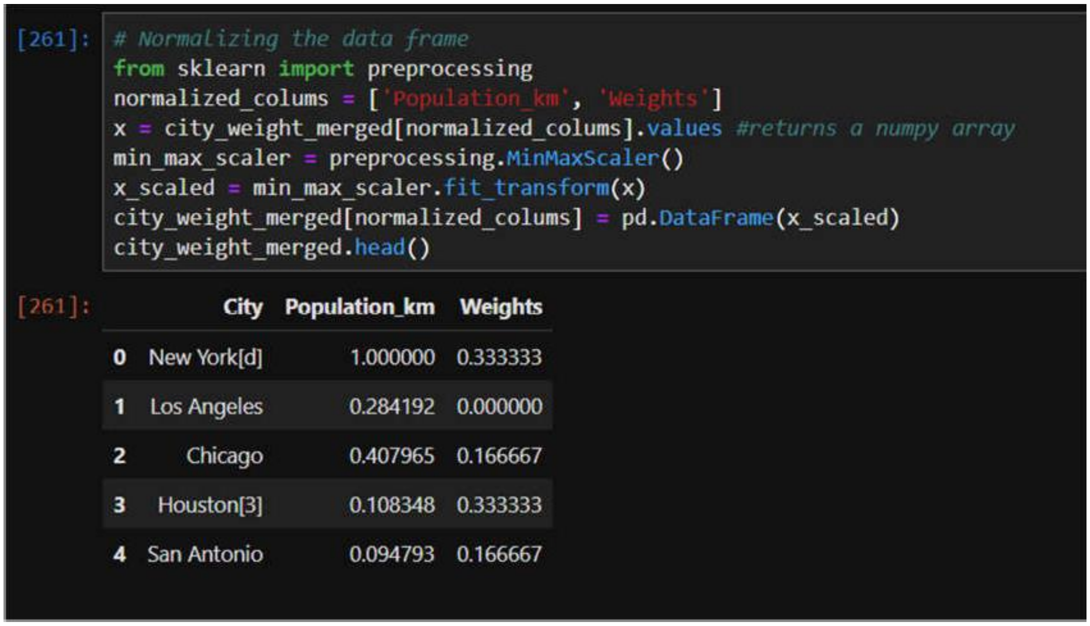
- Dropping the rows that we are not giving any weight
- The venues are grouped by city and then weight mean of each city were calculates. The population data collected from Wikipedia is then merged to the weight's cities



	City	Population_km	Weights
0	New York[d]	10,933/km2	4.0
1	Los Angeles	3,276/km2	3.0
2	Chicago	4,600/km2	3.5
3	Houston[3]	1,395/km2	4.0
4	San Antonio	1,250/km2	3.5

b. Data normalization:

Population and Weights were normalized, using MinMaxScaler() function of sklearn. The normalized data will be the one we will used for the segmentation and clustering.



```
[261]: # Normalizing the data frame
       from sklearn import preprocessing
       normalized_colums = ['Population_km', 'Weights']
       x = city_weight_merged[normalized_colums].values #returns a numpy array
       min_max_scaler = preprocessing.MinMaxScaler()
       x_scaled = min_max_scaler.fit_transform(x)
       city_weight_merged[normalized_colums] = pd.DataFrame(x_scaled)
       city_weight_merged.head()
```

[261]:

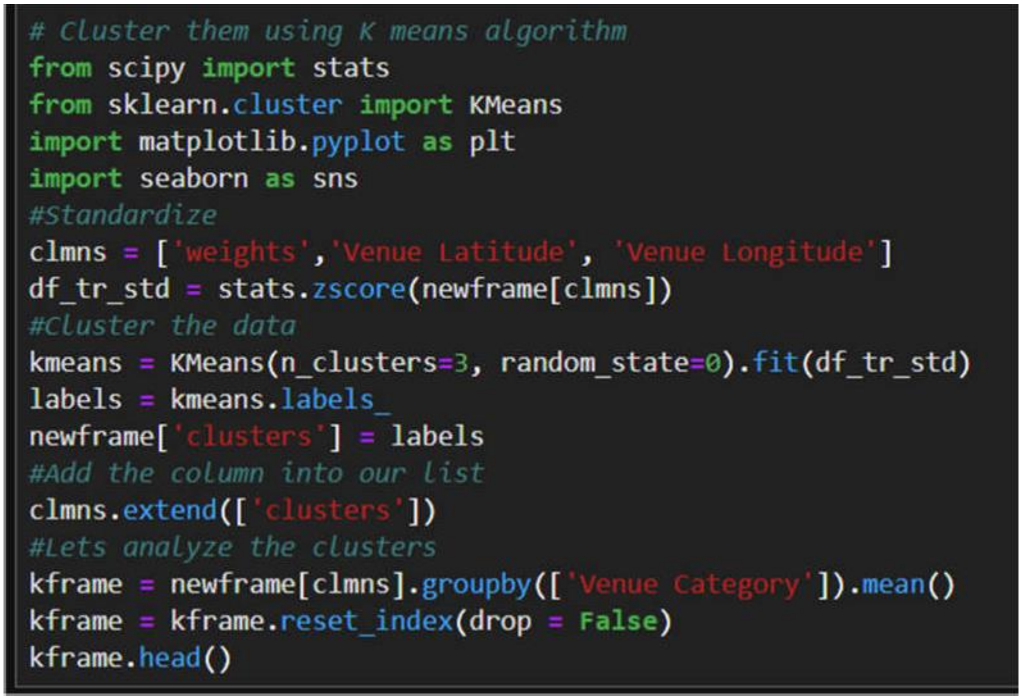
	City	Population_km	Weights
0	New York[d]	1.000000	0.333333
1	Los Angeles	0.284192	0.000000
2	Chicago	0.407965	0.166667
3	Houston[3]	0.108348	0.333333
4	San Antonio	0.094793	0.166667

After the normalization, we calculate the sum of the weight of every city. The city with the high weight value will be selected. The capita income of this city has to be >= 30.000$ to be considered as the best city for the gym installation

3. Segmentation and clustering:

After selecting the best location with the high weight and the income >=30.000$, we used again the Foursquare API to get the avenues of that location and assign a weight to them.

We cluster every venue using K means algorithm and calculate the weights of each of them.



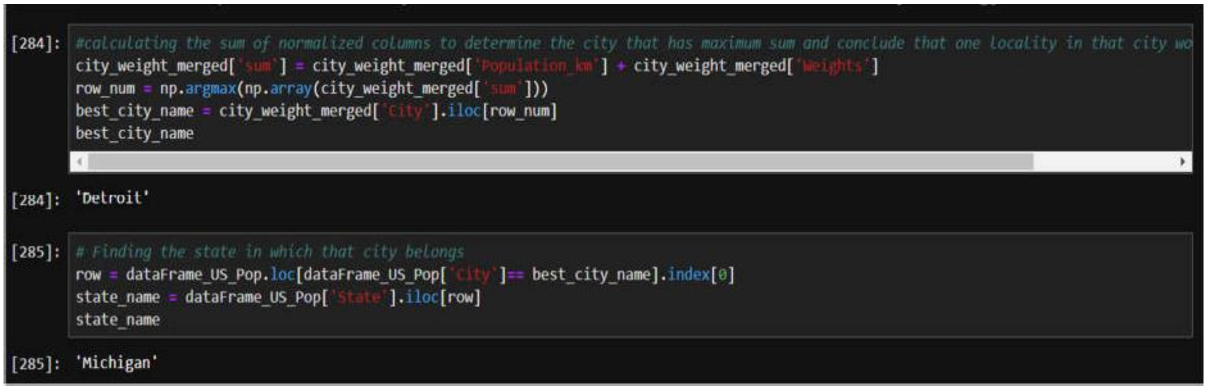
```
# Cluster them using K means algorithm
from scipy import stats
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import seaborn as sns
#Standardize
clmns = ['weights','Venue Latitude', 'Venue Longitude']
df_tr_std = stats.zscore(newframe[clmns])
#Cluster the data
kmeans = KMeans(n_clusters=3, random_state=0).fit(df_tr_std)
labels = kmeans.labels_
newframe['clusters'] = labels
#Add the column into our list
clmns.extend(['clusters'])
#Lets analyze the clusters
kframe = newframe[clmns].groupby(['Venue Category']).mean()
kframe = kframe.reset_index(drop = False)
kframe.head()
```

We group venues by clusters and add mean weights to each cluster.

The location with the maximum weight is chosen and then ploted with a circle of 50M. The Gym / fitness center of Armando can be installed within this circle.

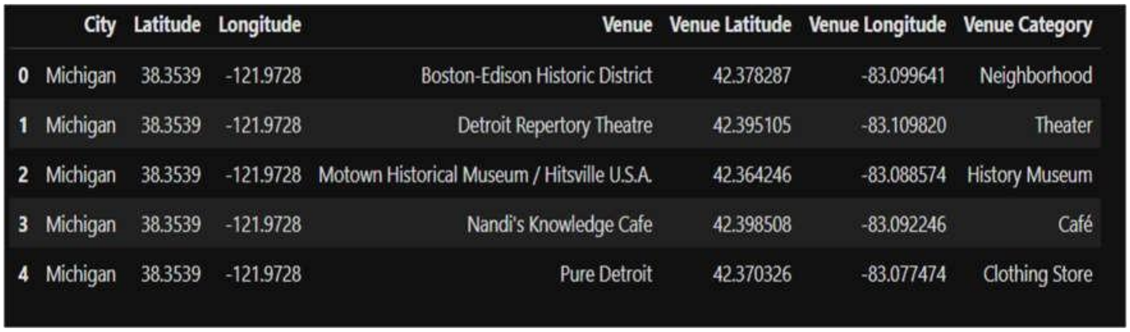
RESULTS

1. **Best city and state: Detroit - Michigan**



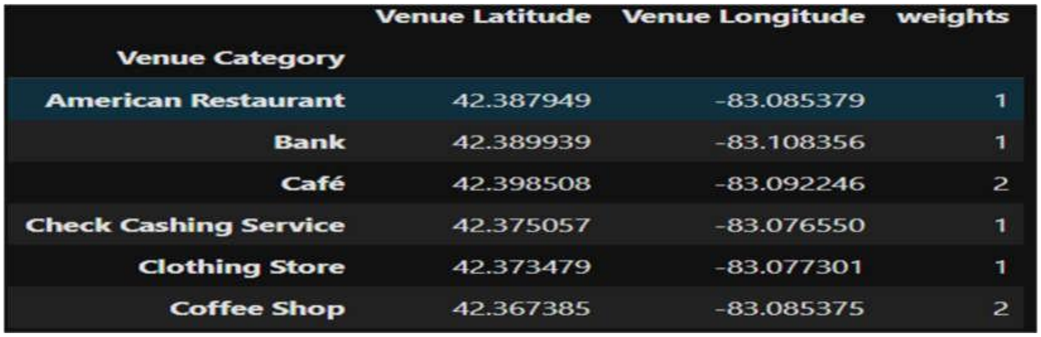
```
[284]: #calculating the sum of normalized columns to determine the city that has maximum sum and conclude that one locality in that city wo
       city_weight_merged['sum'] = city_weight_merged['Population_km'] + city_weight_merged['Weights']
       row_num = np.argmax(np.array(city_weight_merged['sum']))
       best_city_name = city_weight_merged['City'].iloc[row_num]
       best_city_name

[284]: 'Detroit'

[285]: # Finding the state in which that city belongs
       row = dataFrame_US_Pop.loc[dataFrame_US_Pop['City']== best_city_name].index[0]
       state_name = dataFrame_US_Pop['State'].iloc[row]
       state_name

[285]: 'Michigan'
```

2. **List of Michigan venues**



	City	Latitude	Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Michigan	38.3539	-121.9728	Boston-Edison Historic District	42.378287	-83.099641	Neighborhood
1	Michigan	38.3539	-121.9728	Detroit Repertory Theatre	42.395105	-83.109820	Theater
2	Michigan	38.3539	-121.9728	Motown Historical Museum / Hitsville U.S.A.	42.364246	-83.088574	History Museum
3	Michigan	38.3539	-121.9728	Nandi's Knowledge Cafe	42.398508	-83.092246	Café
4	Michigan	38.3539	-121.9728	Pure Detroit	42.370326	-83.077474	Clothing Store

3. **Weight of the venues**



	Venue Latitude	Venue Longitude	weights
Venue Category			
American Restaurant	42.387949	-83.085379	1
Bank	42.389939	-83.108356	1
Café	42.398508	-83.092246	2
Check Cashing Service	42.375057	-83.076550	1
Clothing Store	42.373479	-83.077301	1
Coffee Shop	42.367385	-83.085375	2

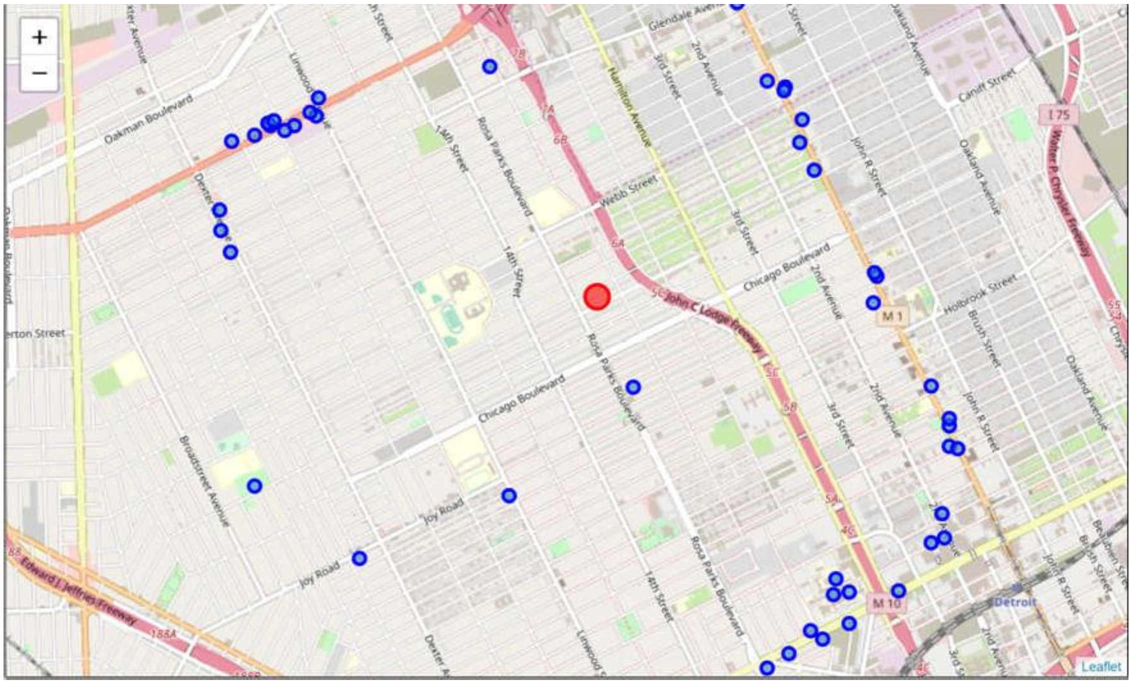
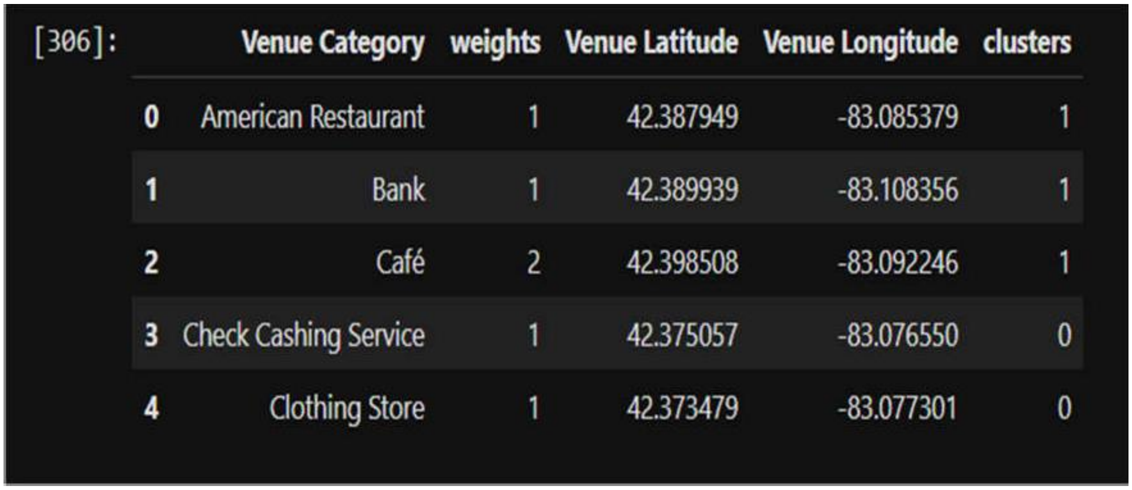


Figure 1: map of the venues of Michigan.

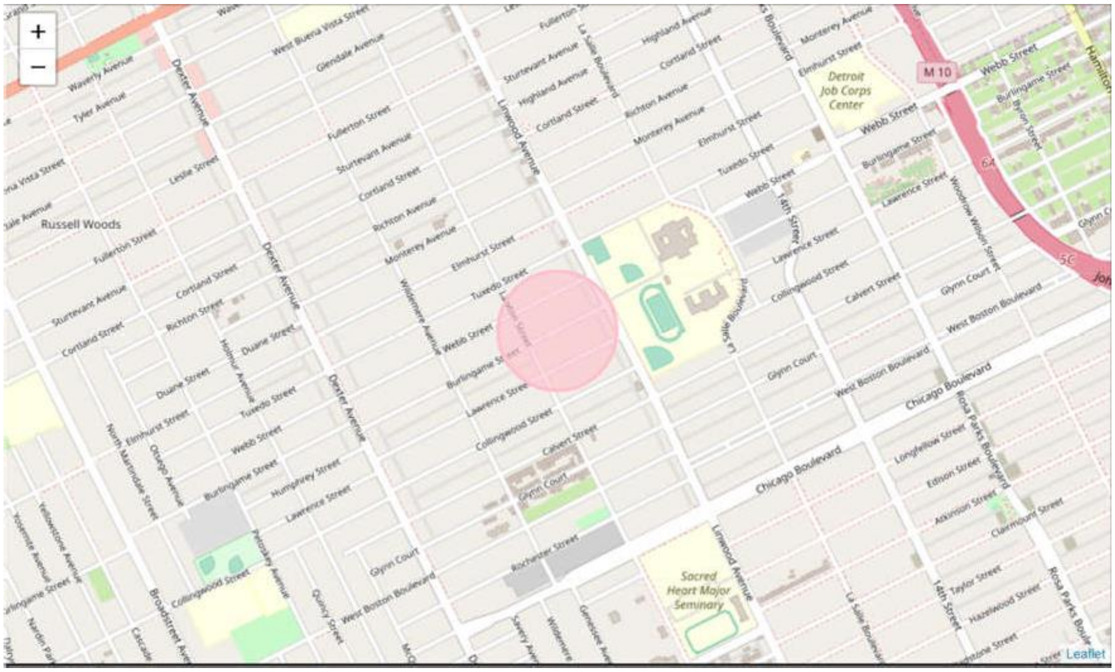
4. **K means algorithm results**



[306]:

	Venue Category	weights	Venue Latitude	Venue Longitude	clusters
0	American Restaurant	1	42.387949	-83.085379	1
1	Bank	1	42.389939	-83.108356	1
2	Café	2	42.398508	-83.092246	1
3	Check Cashing Service	1	42.375057	-83.076550	0
4	Clothing Store	1	42.373479	-83.077301	0

5. **Map of the best place to start the Gym / Fitness center in Michigan**
The pink circle on the map represent the best place to start.


+
−
Russell Woods
Detroit Job Corps Center
Sacred Heart Major Seminary
M 10
Webb Street
Chicago Boulevard
West Boston Boulevard
Dexter Avenue
Linwood Avenue
La Salle Boulevard
14th Street
Rosa Parks Boulevard
Tuxedo Street
Burlingame Street
Lawrence Street
Collingwood Street
Calvert Street
Glynn Court
Elmhurst Street
Monterey Avenue
Richton Avenue
Cortland Street
Sturtevant Avenue
Fullerton Street
Highland Avenue
West Buena Vista Street
Glendale Avenue
Waverly Avenue
Tyler Avenue
Leslie Street
Duane Street
Humphrey Street
Wildemere Avenue
Holmur Avenue
Chicago Avenue
Rochester Street
Longfellow Street
Edison Street
Atkinson Street
Clairmount Street
Taylor Street
Hazelwood Street
Woodrow Wilson Street
Leaflet

DISCUSSIONS AND CONCLUSION

In this work, analysis of best venue recommendations based on venue category, for the installation of Gym /Fitness center has been presented. We found that the city of Detroit in Michigan is the best place to start. The information extracted in this study present on the town areas, will be a good supplement to web-based recommendations for those who would like to implement a gym center in USA and avoid competition. Using Foursquare API, we have collected a good amount of venue in Michigan and a cluster were built.

The generated clusters from our results shows that there are very good and interesting places located in the area of Michigan where the installation of Gym center will be the best. This kind of results may be very interesting for those who want to start this kind of busyness in USA.

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
- ✓ We can increase the Radius for more venue results from each city
- ✓ Consider more categories. For example, like "Universities, Schools, Offices, subway station, bus station" which are also a good source.
- ✓ In the Locality itself, it can also be computed the distance between all the venues in order to find a place with the greatest number of potential customers.