IBM Coursera Applied Data Science Capstone

Tom Thomas

The Battle of the Neighborhoods

Introduction

The US is notorious for being one among the highest ranked countries in terms of obesity rates. The highly commercialized fast-food industry makes it difficult for the average American to stay healthy, especially in large cities, where advertisements are more prominent. This report explores how one could tackle the temptations of living in a large metropolitan city and remain healthy.

There are over 196,000 Fast Food restaurants in the US alone. Despite the pandemic in 2020, the industry grew by 1.1% since then. That's over 2000 new restaurants in the past year alone. In 2012 alone, the industry spent $4.6 Billion dollars on advertising their products. All this leads to show how hard it is to remain healthy in the U.S.

For this analysis, the aim is to focus on the two main factors at hand when it comes to an individual who may be attempting to stay healthy - fast-food restaurants and fitness centers.

Why these two factors? That’s a good question. Consider this scenario - you walk out the door and there are billboards and temptations everywhere. They are all trying to draw you in with well-crafted words by the marketing teams of these giant corporations. Sure, you can resist for a while but imagine waking up everyday to this. The matter is made even worse when one tries to attempt to exercise. Travelling to/from Fitness Centers, the path is filled with the latest offers by Fast Food restaurants. This makes it harder for the average individual to remain healthy. Hence, we will analyze the data to observe any patterns between these two types. Whether there are

more fitness centers where there are fast-food restaurants or the opposite? And if there can be anything that can be done to reduce the desire to visit a fast-food chain restaurant.

The intended audience of this report is anyone who wishes to seek a healthier lifestyle while living in a city, and in our case L.A. So for this particular approach, we will focus on those interested in moving to L.A. or currently living in L.A.

Data

For a thorough analysis, we require the following data:

1. **List of Neighborhoods in a Metropolitan City** - We have chosen Los Angeles for this project, which is a large city with a lot of fast food options and fitness options allowing for a more concrete observation. We will collect this information from a project published by the LA times. The L.A. Times have compiled an accurate list of all the Neighborhoods and Regions of Los Angeles.

   Mapping L.A. - Los Angeles Times (latimes.com)

2. **Department of Public Health Data for LA County** - This dataset will be useful for evaluating correlation between obesity rates and the two main factors. This data will be sourced from the LA County Public Health Records.

   Department of Public Health (lacounty.gov)

3. **Geospatial Data using Geocoder and Geopy Package** - We will use these packages to capture the required Latitudes and Longitudes for LA County.

   Welcome to GeoPy's documentation! — GeoPy 2.2.0 documentation

   OpenCage Geocoder - Easy, Open, Worldwide, Affordable Geocoding (opencagedata.com)

4. **GeoJSON Data for LA County** - This will be the data we use to create choropleth maps for Los Angeles.

https://apps.gis.ucla.edu/geodata/dataset/93d71e41-6196-4ecb-9ddd-15f1a4a7630c/resource/6cde4e9e-307c-477d-9089-cae9484c8bc1/download/la-county-neighborhoods-v6.geojson

5. **FourSquare Venue Data** - We will utilize the FourSquare API to collect venue information for the different neighborhoods.

The Foursquare API is an independent and global, location based platform that collects user-generated information via their app and other sources for public points of interest into a streamlined database. Developers can access this data by creating an account on their platform. I used the API to return public venue information based on geospatial data (latitudes, longitudes).

Methodology

THE LIBRARIES

The main packages and libraries that we implemented to allow for data manipulation, visualization, API calls, data extraction and cleaning were these.

```
# Importing requried libraries
# Install necessary dependencies if you haven't already
!pip install opencage

import pandas as pd
import json
import requests
import numpy as np
import folium

from folium import plugins
from bs4 import BeautifulSoup
from geopy.geocoders import Nominatim
from opencage.geocoder import OpenCageGeocode
from pandas.io.json import json_normalize
```

Fig 1: Required Libraries

Pandas - Pandas is a data manipulation library that helps us clean messy data and convert it into a clean tabular format called DataFrame.

Numpy - Numpy helps us apply mathematical operations to data in an array format.

JSON - JSON helps us import the JSON format into Python into a dict format, that's easily modifiable for manipulation.

Requests - Requests help us make API calls. This has been useful while implementing the Foursquare API calls.

Folium - Folium is a visualization tool that works well with geospatial data representing tabular data in a visual format that makes sense.

BS4 - bs4 or BeautifulSoup helps us work with HTML or XML files to extract text data.

Geopy/Opencage - These packages help us convert text string into lat-long values or vice-versa.

DATA IMPORTS


MAPPING L.A. LIST

Neighborhoods

Select one of L.A.'s 272 neighborhoods

NAME	REGION
Acton	Antelope Valley
Adams-Normandie	South L.A.
Agoura Hills	Santa Monica Mountains
Agua Dulce	Northwest County
Alhambra	San Gabriel Valley
Alondra Park	South Bay
Altadena	Verdugos

Fig 2: Neighborhood Data Website

The first data we needed to import was from the L.A. Times. They have been compiling an accurate list of each of L.A.'s Neighborhoods for several years. We utilized BeautifulSoup to extract the table on the webpage.

```
# Create BeautifulSoup object
page = requests.get('http://maps.latimes.com/neighborhoods/neighborhood/list/
soup = BeautifulSoup(page,'html5lib')
tbl = str(soup.table)

# Convert to table
list1 = pd.read_html(tbl)
la_list = list1[0]
la_list.rename(columns={'Name':'Neighborhood'}, inplace =True)

la_list.head()
```

Fig 3: Extract text from HTML

This was a fairly straightforward process as the Table was easily identifiable in the webpage. After extracting the webpage and using pandas, we were able to convert it into a DataFrame. We also renamed the columns for streamlined naming going forward.

	Neighborhood	Region
0	Acton	Antelope Valley
1	Adams-Normandie	South L.A.
2	Agoura Hills	Santa Monica Mountains
3	Agua Dulce	Northwest County
4	Alhambra	San Gabriel Valley

Fig 4: Neighborhood Data Table

Secondly, we needed the Public Health Data. The Public Health Department of L.A. County held these .xlsx files which we then imported in using the .read_excel function that pandas provides.

We are going to get two types of indicators for our analysis:

1. The Percentage of Adults in LA County Meeting Recommended Guidelines for Physical Activity
2. The Percentage of Adults in LA County Who are Obese

```
# Read Excel data
fit = pd.read_excel("https://github.com/tomthomas/Coursera_Capstone/blob/main/
obs = pd.read_excel("https://github.com/tomthomas/Coursera_Capstone/blob/main/

# Drop unnecessary columns
fit.drop(fit.columns[[-1,-2]], axis = 1,inplace=True)
obs.drop(obs.columns[[-1,-2]], axis = 1,inplace=True)
fit.rename(columns={'Percentage':'Healthy Adult Percentage'}, inplace =True)
obs.rename(columns={'Percent':'Obesity Rate'}, inplace =True)
health_data = pd.merge(fit, obs, on = 'City/Community')

health_data.head()
```

Fig 5: Health Data Import

We feel that these two indicators could be key to our analysis. Our goal is to find Neighborhoods that would reduce the Fast Food footprint and these would include those. Data import was fairly smooth. After the Excel files were imported, we merged them together to form a singular DataFrame.

	City/Community	Healthy Adult Percentage	Obesity Rate
0	Alhambra	0.273130	0.135779
1	Altadena	0.348242	0.244131
2	Arcadia	0.266197	0.057046
3	Azusa	0.375943	0.260744
4	Baldwin Park	0.316393	0.265167

Fig 6: Health Data

Thirdly, we needed to acquire the GeoJSON file which would contain the boundaries for each of the polygons that would cover L.A. This would be useful when it comes to displaying Choropleth Maps in Folium. Getting the JSON file was fairly simple. The file is stored into a dictionary which can then be accessed by it's keys.

One of the issues we ran into was with the Neighborhood column of the Neighborhood DataFrame. Los Angeles seems to be a bit of an anomaly when it comes to the total number of officially designated communities.

https://en.wikipedia.org/wiki/List_of_cities_in_Los_Angeles_County,_California states that there are 88 cities which seems more in alignment with our Public Health dataframe.

https://en.wikipedia.org/wiki/List_of_unincorporated_communities_in_Los_Angeles_County,_California states that there are 76 unincorporated communities with a small population residing in each of them. We dropped these rows since these are not just part of the city of LA but rather of the greater county of LA and hence, may stick out as outliers in our analysis.

To keep things simple, we used the 87 Cities found in the Public Health dataset and designated Regions for each City/Community.

We also noticed that it conflicted with the geopy/opencage's string to coordinates service since there were several Council Districts in L.A. The packages were unable to identify the correct coordinates for these locations and kept returning the same coordinates for all of them (There were 15 rows). To fix this, luckily, the Geojson file also had Neighborhood and Region metadata. We appended these values into a DataFrame.

```
# Find all the Neighborhoods and Regions in the data
la_geo_name = []
la_geo_region = []

for feature in geo['features']:
    print (feature['properties']['name'])
    la_geo_name.append(feature['properties']['name'])
    la_geo_region.append(feature['properties']['metadata']['region'])

# Create a DataFrame
la_geo = pd.DataFrame({'Neighborhood':la_geo_name,
                       'Region':la_geo_region})
```

Fig 7: Retrieve Neighborhood Data from GeoJSON

	Neighborhood	Healthy Adult Percentage	Obesity Rate	Region
0	Florence-Graham	0.298663	0.312639	NaN
1	Los Angeles Council District 1	0.314527	0.21641	NaN
2	Los Angeles Council District 2	0.386215	0.207762	NaN
3	Los Angeles Council District 3	0.365171	0.20308	NaN
4	Los Angeles Council District 4	0.417725	0.15148	NaN
5	Los Angeles Council District 5	0.423018	0.102214	NaN
6	Los Angeles Council District 6	0.329751	0.2296	NaN
7	Los Angeles Council District 7	0.356249	0.286106	NaN
8	Los Angeles Council District 8	0.356369	0.318689	NaN
9	Los Angeles Council District 9	0.322849	0.325907	NaN
10	Los Angeles Council District 10	0.330935	0.227564	NaN
11	Los Angeles Council District 11	0.415252	0.103938	NaN
12	Los Angeles Council District 12	0.342960	0.215699	NaN
13	Los Angeles Council District 13	0.355653	0.186854	NaN
14	Los Angeles Council District 14	0.368921	0.284092	NaN
15	Los Angeles Council District 15	0.349099	0.273553	NaN
16	Los Angeles, City of	0.363092	0.219263	NaN
17	Los Angeles County	0.341288	0.23517	NaN

Fig 8: Messy Data issues with Neighborhood Data

We renamed the Council Districts with more appropriate Neighborhood names from various sources. We also replaced the Region data with what we found in the GeoJSON. Once it was replaced, it was formatted to appear more visually pleasing.

```
la_merged1 = pd.merge(la_merged, la_geo, on = 'Neighborhood')

# Drop old Region Column
la_merged1.drop(['Region_x'], axis= 1, inplace=True)
la_merged1.rename(columns={"Region_y":"Region"},inplace=True)

#Make it aesthetic
la_merged1['Region'] = la_merged1['Region'].str.replace('-',' ')
la_merged1['Region'] = la_merged1['Region'].str.title()
la_merged1['Region'] = la_merged1['Region'].replace("La", "LA", regex=True)
```

Fig 9: Fixing Regional Data

Fourthly, we needed to acquire the Geospatial data or the latitude and longitude values for each of our Neighborhoods. We used Nominatim to translate the coordinates of LA which we could then use as a base map for Folium and for the Foursquare API calls. We then used geocoder to get the values for each Neighborhood. The values were then appended into the DataFrame.

	Neighborhood	Obesity Rate	Healthy Adult Percentage	Region	Latitude	Longitude
0	Alhambra	0.135779	0.273130	San Gabriel Valley	34.093042	-118.127060
1	Altadena	0.244131	0.348242	Verdugos	34.186316	-118.135233
2	Arcadia	0.057046	0.266197	San Gabriel Valley	34.136207	-118.040150
3	Azusa	0.260744	0.375943	San Gabriel Valley	34.133875	-117.905605
4	Baldwin Park	0.265167	0.316393	San Gabriel Valley	34.085474	-117.961176

Fig 10: LA Data with Coordinates

We used Folium to plot an initial map of LA to show all our Neighborhoods.

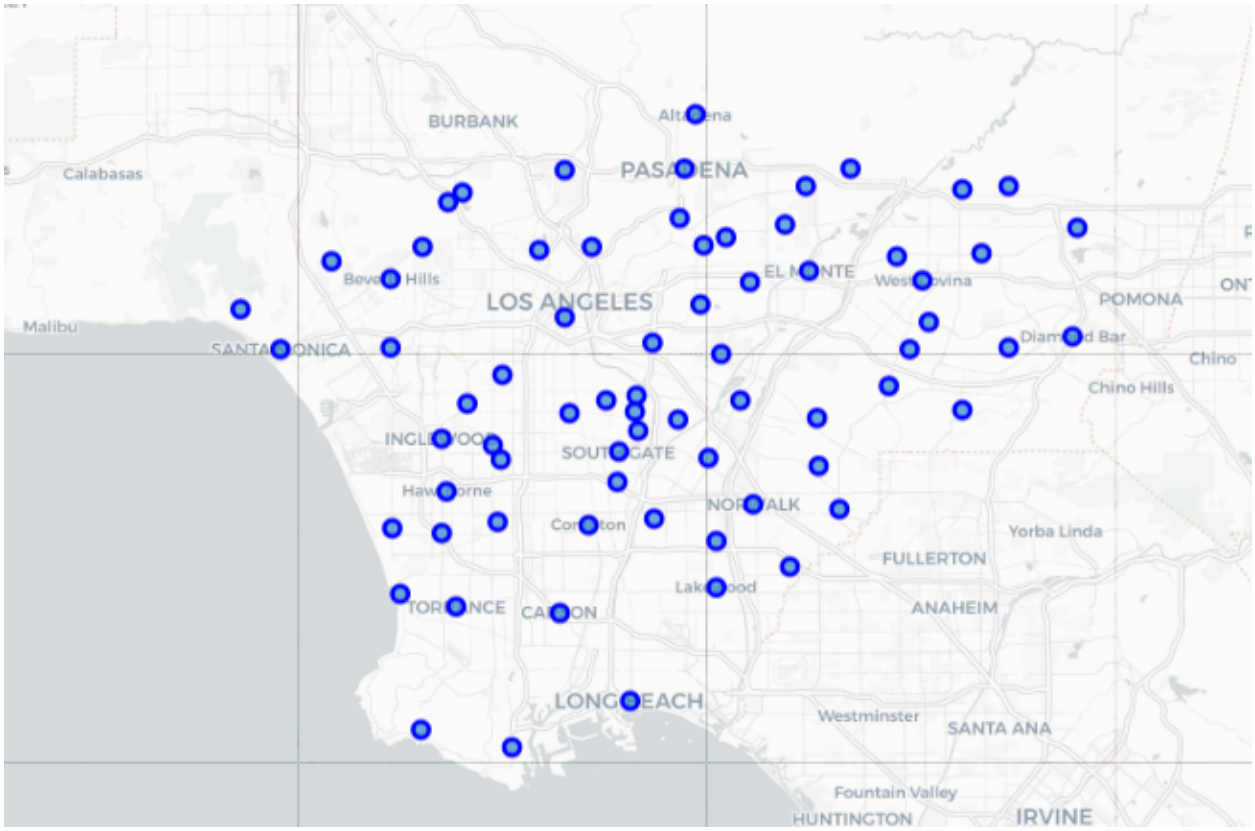

Fig 11: LA with reduced Neighborhood markers

We noticed that there were a lot more Neighborhoods than we needed as this data is not just for Los Angeles but for the greater LA County. Hence we sliced the data to remove some Regions, so that we could focus on the City of Los Angeles itself.

Fifthly, we needed to call the Foursquare API to get the Venue information. Foursquare operates on collecting all the public points of interest within a coordinate and returning them to us. We identified two methods of collecting the venues. The goal of attempting two methods was to simply work with a sample size for the initial analysis and then an extensive list for more detailed analysis. This was done to reduce resource usage.

Method 1 allows you to import a certain number of venues of all types from the API. It then extracts just the venues we need (Fast Food and Fitness). This is more resource friendly. However, we will acquire less specific venues.

The change lies in the url we specify to Foursquare to inspect, and the limits we set.

url_method1 = '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)

Method 2 purposely seeks out just those venues for Fast Food and Fitness Centers using their specific Category ID's that the API has declared. Using this method, we are able to get more specific venues and a larger amount. This is more resource heavy and hence, will take a longer time. This is useful when exploring in-detail for each area of Los Angeles.

Url_method2= 'https://api.foursquare.com/v2/venues/explore?categoryId={}&intent=browse&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(cat_id,CLIENT_ID, CLIENT_SECRET, VERSION, lat, lng, radius, LIMIT)

Notice that the second url uses the category ID’s as well which we can specify to single out and request only the venues for those categories. Once the calls were made, we stored them in DataFrames. The second method only increased the LIMIT to 200 and the Radius to 5000.

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Alhambra	34.093042	-118.127060	In-N-Out Burger	34.106211	-118.134465	Fast Food Restaurant
1	Alhambra	34.093042	-118.127060	Planet Fitness	34.077420	-118.116117	Gym
2	Altadena	34.186316	-118.135233	24 Hour Fitness	34.183503	-118.159222	Gym
3	Altadena	34.186316	-118.135233	Equinox Pasadena	34.145032	-118.145536	Gym
4	Altadena	34.186316	-118.135233	iLoveKickboxing - Pasadena, CA	34.145764	-118.114105	Gym

Fig 12: Foursquare API Venues

Doing a count on the DataFrame we created using Method 1 shows us that we imported over 271 Fast Food Restaurants and 111 Fitness Centers.

```
[32] health_stat = la_health.groupby(by=['Venue Category'])
     health_stat.count()
```

Venue Category	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude
Fast Food Restaurant	271	271	271	271	271	271
Gym	111	111	111	111	111	111

Fig 13: Number of Fast Food and Fitness locations

Using Method 2, we brought in over 5000+ Fast Food Restaurants and 4000+ Fitness Centers.

```
print(la_fatness.shape)
la_fatness.head()
```

(5322, 7)

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Alhambra	34.093042	-118.12706	Grill 'Em All	34.095595	-118.126725	Fast Food Restaurant
1	Alhambra	34.093042	-118.12706	In-N-Out Burger	34.106211	-118.134465	Fast Food Restaurant
2	Alhambra	34.093042	-118.12706	Bon Appétea	34.093905	-118.128832	Café
3	Alhambra	34.093042	-118.12706	Newport Tan Cang Seafood Restaurant	34.102310	-118.106975	Chinese Restaurant
4	Alhambra	34.093042	-118.12706	Huge Tree Pastry	34.067634	-118.134897	Taiwanese Restaurant

Fig 14: Fast Food Restaurants - Method 2

```
print(la_fitness.shape)
la_fitness.head()
```

(4086, 7)

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Alhambra	34.093042	-118.12706	Planet Fitness	34.077420	-118.116117	Gym / Fitness Center
1	Alhambra	34.093042	-118.12706	Yoga House	34.125574	-118.150645	Yoga Studio
2	Alhambra	34.093042	-118.12706	LA Fitness	34.083529	-118.151576	Gym / Fitness Center
3	Alhambra	34.093042	-118.12706	Snap Fitness	34.116371	-118.140191	Gym
4	Alhambra	34.093042	-118.12706	YMCA of South Pasadena & San Marino	34.107716	-118.136688	Gym / Fitness Center

Fig 15: Fitness Centers - Method 2

The aim of this project was to identify the ideal fitness center locations that would aid in staying healthy for an adult in LA County.

This goal is based on 2 main assumptions:

1. Visiting and Leaving Fitness Centers in the vicinity of a greater number of fast-food restaurants reduces the motivation of the participant to return to the fitness center OR increases the probability of visiting a fast-food restaurant after a workout.
2. Neighborhoods with a larger number of fast-food restaurants have a higher obesity rate.

To analyze these assumptions and to identify the ideal fitness center location, we used the following methods.

- Based on the neighborhood data acquired and cleaned, import geospatial data to pinpoint lat,long values for each Neighborhood.
- Acquire Venue information for each Neighborhood from Foursquare API.
- Analyze Venue Information using Visualization to find reasons to support our assumptions.
- Identify which Fitness Centers best qualify as a candidate for training to maintain a healthy diet.

Results

Based on our Analysis, we found several things that seemed to support our assumptions. We used the ability of Folium to show us our results.

We combined the Fast Food and Fitness data acquired and separated them into two DataFrames. Based on this separation, we were able to apply them as separate markers in Folium allowing for a different color. The visual separation of the data allowed for greater clarity on where the key areas would be for a healthy adult to reside in LA so that they can gain the benefits of a healthy living while eliminating the temptations of having to visit Fast Food restaurants.

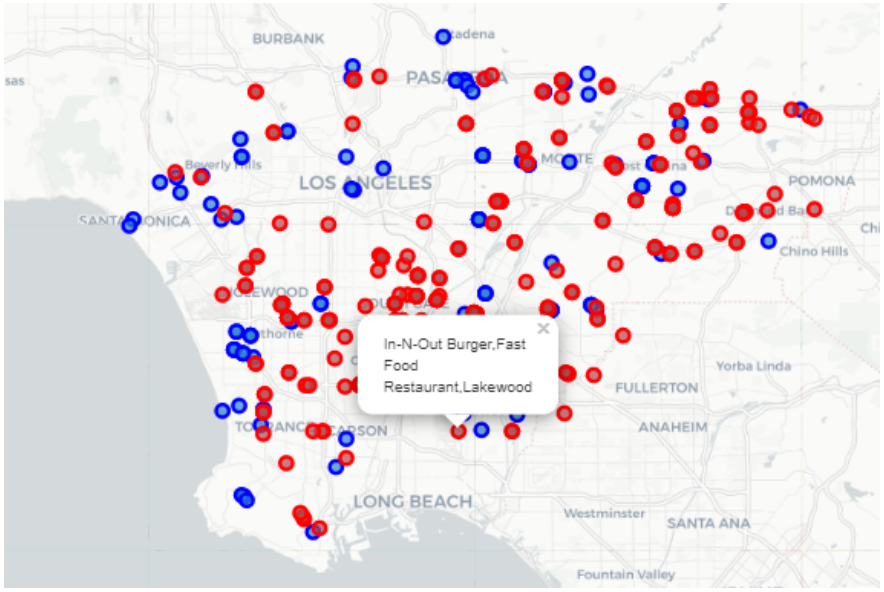

Fig 16: Red points represent Fast Food, Blue points for Fitness Centers.

Applying custom markers allowed us to identify these key areas that stood out on the map. The ideal locations were those that had a lower density of Fast Food Restaurants while maintaining a considerable amount of Fitness Centers.

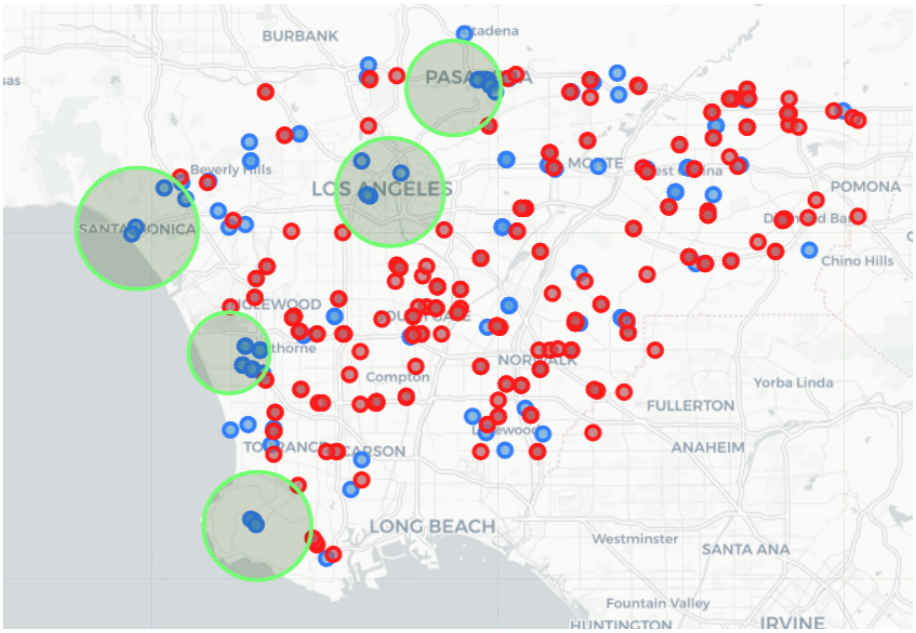

Fig 17: Custom markers representing possible ideal locations.

The custom markers were created by identifying the lat-long values in those Regions.

Our initial analysis showed 5 Neighborhoods in Los Angeles where we see a greater proportion of fitness options to fast food options:

1. Santa Monica
2. Manhattan Beach
3. Rancho Palos Verdes
4. Downtown LA
5. South Pasadena

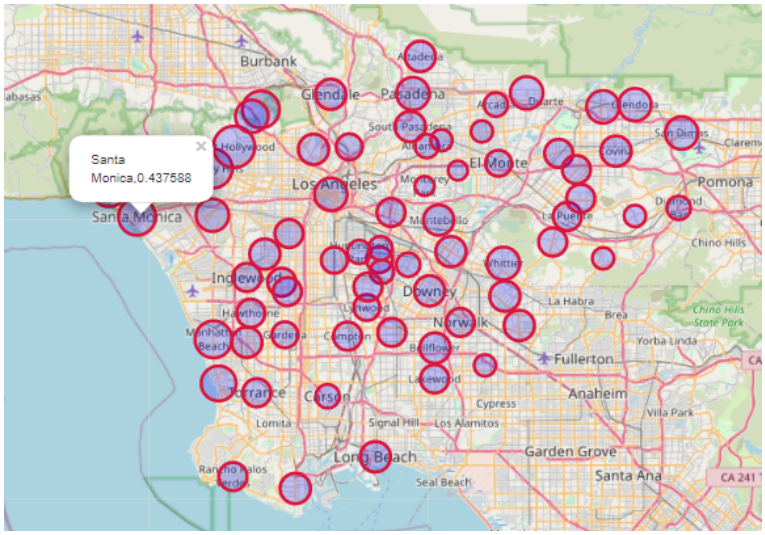

Fig 18: Custom Markers that varied in size to show Obesity Rates for LA

Folium also allowed us to visualize the Obesity Rate using custom markers that varied in radius. This helped us visualize Neighborhoods that had higher and lower Obesity Rates on an initial level. However, we found that Choropleth maps were far more effective visually when it came to discerning such data vs. points.

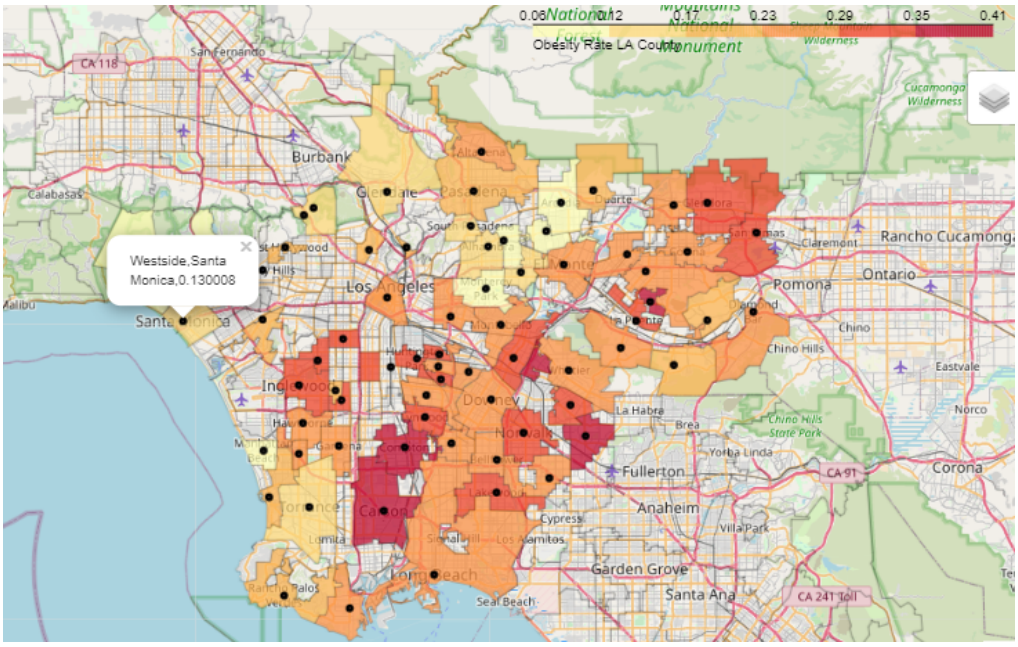

Fig 19: Choropleth Map displaying the Obesity Rate of L.A.

From the data, we can clearly see that:

The highest Obesity Rates were found in these Neighborhoods

1. Carson, Harbor
2. Compton, Southeast
3. La Mirada, Southeast
4. Valinda, San Gabriel Valley

The lowest Obesity Rates were found in these Neighborhoods

1. Arcadia, San Gabriel Valley
2. San Gabriel, San Gabriel Valley
3. Manhattan Beach, South Bay
4. Pacific Palisades, Westside

Choropleth Maps allow for easy visual perception when it comes to identifying patterns. We noticed that the Southern and Eastern areas of LA are areas we may want to avoid when it comes to choosing an ideal place to reside. Conversely, the West Coast seems to be an ideal

location which equates well with the initial observations we saw before with the regular marker maps.

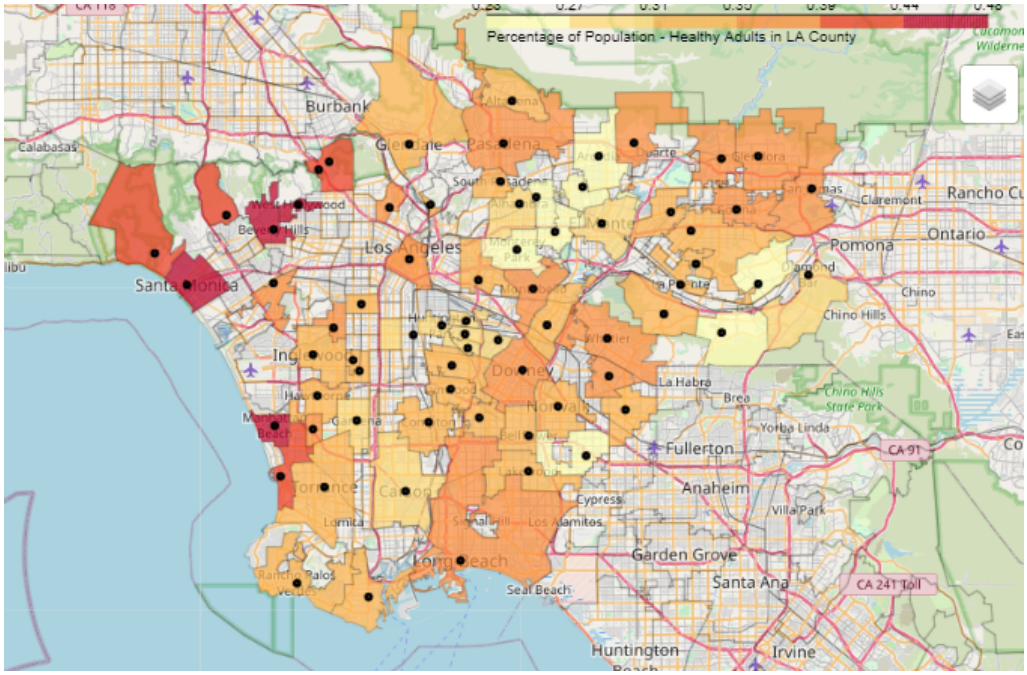

Fig 20: Choropleth Map displaying the Healthy Adult Rate of L.A.

From the data, we can see that:

The highest Healthy Adult Percentage Rates were found in these Neighborhoods

1. Beverly Hills, Westside
2. West Hollywood, Central LA
3. Manhattan Beach , South Bay
4. Santa Monica, Westside

The lowest Healthy Adult Percentage Rates were found in these Neighborhoods

1. Rosemead, San Gabriel Valley
2. Monterey Park, San Gabriel Valley
3. Cerritos, Southeast
4. Rowland Heights, San Gabriel Valley

Continuing the pattern, we are seeing the West Coast to be an ideal location as here, we choose the higher end of the range for Healthy Adults. These turn out to be places that also have a lesser density of Fast Food restaurants.

Additionally, we also used Heat Maps to explore insights. We used the API Call Method 2 data with over 9000+ points of interest to get an accurate Heat Map. Based on the Heat Maps, we found these areas to be potential ideal locations.

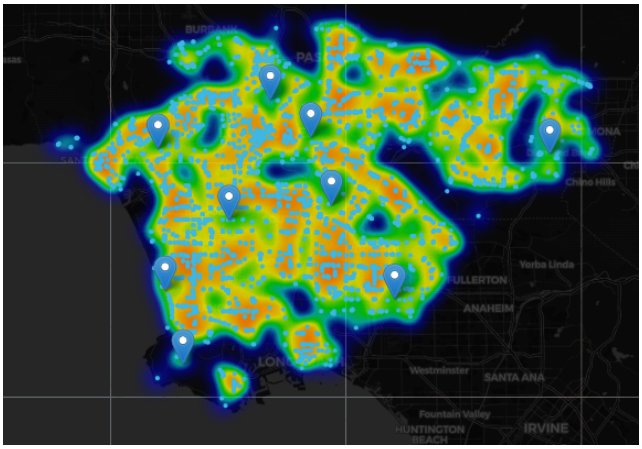

Fig 21: Heat Map of LA for Fast Food Restaurants.

We can see that the locations are a bit more spread out now with little spots here and there that could also be considered.

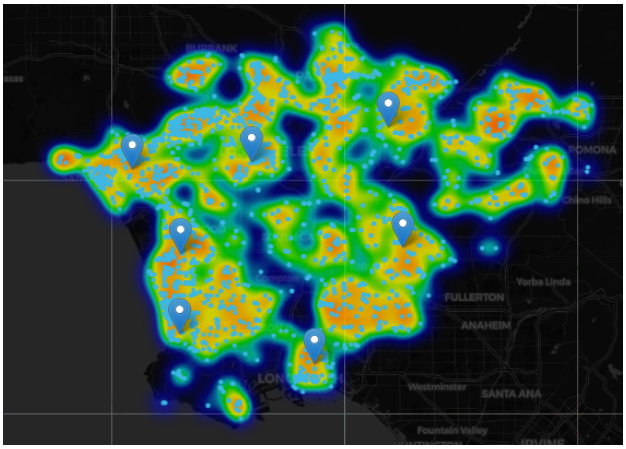

Fig 22: Heat Map of LA for Fitness Centers

Yet, one thing to note is that we want to choose points that intersect both these maps. Since we want both high density Fitness Centers and Low density Fast Food restaurants.

When we consider that, we see that the West Coast seems to stand out again in it's favor.

Discussion

The approach we took can definitely be taken further or reconsidered. Some additional things to consider:

1. The number of Venues that you can collect from Foursquare increases your accuracy for better results. There could be locations outside the U.S. that may not cover venues as extensively as L.A.
2. We covered Los Angeles on a high-level. One could take it even further by exploring each Region of L.A.
3. Another approach could be that one could look into the route that Anna may take to/from Fitness Centers. This route can be optimized to minimize Fast Food restaurants. We have lat-long values for each establishment. These could be brought into use in this scenario.
4. There may be other factors that correlate better than Obesity rate such as access to healthier food options for each Neighborhood.
5. The venue data we pulled from Foursquare consisted of one category type for each: Fast Food Restaurants and Gyms. There are other categories that we haven't considered such as other forms of food establishments that may be unhealthy such as Donut shops or Pizza places, or other Fitness options besides a gym such as outdoor areas, pools, Tennis/Basketball courts and so on. Including these would result in a more accurate outcome.

Conclusion

Therefore, the ideal locations for someone to pursue a healthy lifestyle in LA include the Neighborhoods of Santa Monica Area, The Hollywood Hills, Ranchos Palos Verdes, and Manhattan Beach.