Jaynab Khatun
January 30, 2019

Capstone Restaurant Project

Table of Contents :

Introduction:

There are many big city in the United States. Every city has developed. Most of cities are a center of attention for residential, job employment, tourism, education, shopping and sports activity. California , New York, Chicago are became the top choice for local and foreign communities. Restaurants are important for celebrities , foreigners. From this project we learn about culture, population, attraction of place.

Objective

In this interactive project, we will study in details the area classification using Foursquare data and machine learning segmentation and clustering. The goal of this project is find out most common cuisine, restaurant based on current rating. The rating, venue id, cuisine details will extract from Foursquare. Our objective is find out few answer which help us to open a new restaurants in particular city and also decide the cuisine.

The purpose of this project is to find answer of following questions :

- What is the expected rating and popularity of neighborhood at each city?
- What is the area of city with best cuisine and rating that meets criteria of customer?
- What is the distance from work place (Park Ave and 53 rd St) and the tentative future home?
- What are the venues of the two best places to live? How the prices compare?
- How venues distribute among cities neighborhoods and around metro stations?
- Are there tradeoffs between cuisine, Tips and location?
- Any other interesting statistical data findings of the real estate and overall data.
- Can I open a restaurant with which kind of cuisine will benefit the business.
- Which city is good for what kind of cuisine and its popularity also.
- How many cuisine and restaurant are there in a particular city.

Data

Description of data:

- The following data is required to answer the issues of the problem:
- List of neighborhoods restaurant of USA with their GeoData (latitude and longitude)
- List of City in USA where the restaurant situated.
- List of Cuisine of each neighborhood in metro city.
- Venue id for each Metro city neighborhood (than can be clustered)
- Rating of each restaurant in particular city.

How the data will be used to solve the problem. The data will be used as follows:

- Use Foursquare and geopy data to map top 10 venues for all Metro city neighborhoods and clustered in groups.
- Use Foursquare and geopy data to map the location of particular cuisine, in some form, linked to the subway locations.
- create a map that depicts, for instance, the average cuisine rating and feedback and tips.
- Addresses neighborhood from metro city will be converted to geodata(lat, long) using Geopy-
  distance and Nominatim.
- Data will be searched in open data sources if available, from food app.

Data Analysis

The Data analysis part has done with the help foursquare data. Raw Data downloaded from kaggle.com and put it in Github so that it is easy for reviewer to check the data. Visualize data and transform it into applicable form so that we can use machine learning algorithm on it.

```
df= pd.read_csv('https://raw.githubusercontent.com/jaanu/Capstone_neighboring/master/us_restaurant
.csv')
```

Here is data overview with header. Each attribute has their value and we remove Nan value.

	Restaurant Name	City	Cuisines	Latitude	Longitude
0	El Vaquero Mexican Restaurant	Albany	Mexican	40.094257	-83.085363
1	Chick-fil-A	Albany	Fast Food	38.573684	-75.286200
2	Guang Zhou Chinese Restaurant	Albany	Asian, Chinese, Vegetarian	14.068107	-60.955487
3	Harvest Moon	Albany	Pizza, Bar Food, Sandwich	54.312130	-1.690167
4	Hong Kong Cafe	Albany	Chinese, Seafood, Vegetarian	43.001832	-75.977458
5	Locos Grill & Pub	Albany	American, Burger, Sandwich	32.079725	-81.095565
6	Longhorn Steakhouse	Albany	American, Steak	40.540493	-105.076547
7	3 Squares Diner	Albany	American, Breakfast, Diner	31.039941	-84.876857
8	Mama's Boy Restaurant	Athens	Southern	41.890615	-87.629398
9	Sr. Sol 1	Athens	Mexican	42.425646	-2.078956
10	Big City Bread Cafe	Athens	Breakfast, Sandwich	33.959298	-83.384128
11	Taqueria Del Sol	Athens	Mexican, Spanish	36.126600	-86.789405
12	The National	Athens	International, Southern	47.611228	-122.339494
13	The Royal Peasant	Athens	Bar Food	33.938038	-83.387066
14	Transmetropolitan	Athens	Italian, Pizza, Sandwich	33.958487	-83.376544
15	Trappeze Pub	Athens	Burger, Bar Food	33.958494	-83.379016
16	The Bee's Knees	Augusta	International, Tapas, Vegetarian	41.001192	-111.910303
17	Boll Weevil Cafe	Augusta	Desserts, Sandwich, Southern	31.314453	-85.854016
18	Mellow Mushroom	Augusta	Italian, Pizza, Sandwich	33.921787	-84.379529
19	Rhinehart's Oyster Bar	Augusta	Bar Food, Sandwich, Seafood	33.513429	-82.050222

Now we find out all the coordinate of all restaurant using geopy.geocoder. Lets find out all the venue and extract the venue id . Here we use foursquare data.

```
        location = geolocator.geocode(address)
        latitude = location.latitude
        longitude =location.longitude
    except:
        latitude=0
        longitude=0
    coordinates = coordinates.append({'Latitude':latitude,'Longitude':longitude},ignore_index=True
)
```

```
df= df.join(coordinates, how ='outer')
df
```

```
import requests
# tranforming json file into a pandas dataframe library
from pandas.io.json import json_normalize



venueid=pd.DataFrame(columns=['venue_id'])
for i,r in df.iterrows():
    search_query=r['Restaurant Name']
    latitude = r['Latitude']
    longitude = r['Longitude']
    try:
        url = 'http://cladiusfernando-eval-test.apigee.net/foursquare/v2/venues/search?client_id={
}&client_secret={}&ll={},{}&v={}&query={}&radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET, lat
itude, longitude, VERSION, search_query, radius, LIMIT)
        venues = requests.get(url).json()['response']['venues']
        dataframe = json_normalize(venues)
        venue_id=dataframe['id'][0]
    except:
        print(search_query)
        venue_id=np.nan

    venueid =venueid.append({'venue_id': venue_id}, ignore_index = True)
    """url2 = 'https://api.foursquare.com/v2/venues/{}?client_id={}&client_secret={}&v={}'.format(
venue_id, CLIENT_ID, CLIENT_SECRET, VERSION)
    try:
        rating = requests.get(url2).json()['response']['venue']['rating']
    except:
        rating=0
    print(rating)
    Rating =Rating.append({'rating': rating}, ignore_index = True)"""
df= df.join(venueid, how ='outer')
```

	Restaurant Name	City	Cuisines	Latitude	Longitude	venue_id
0	Chick-fil-A	Albany	Fast Food	38.573684	-75.286200	5ab51bd3446ea6289e2bf8c6
1	Hong Kong Cafe	Albany	Chinese, Seafood, Vegetarian	43.001832	-75.977458	4ea4a5a5be7ba4918f303261
2	Locos Grill & Pub	Albany	American, Burger, Sandwich	32.079725	-81.095565	50c9385ffe1e45fe50f0e06f
3	Longhorn Steakhouse	Albany	American, Steak	40.540493	-105.076547	56d42849cd10d6b76f71a6d1
4	3 Squares Diner	Albany	American, Breakfast, Diner	31.039941	-84.876857	4c3f06100596c928dc0a8578
5	Mama's Boy Restaurant	Athens	Southern	41.890615	-87.629398	57cf6782498e577643efc1bd
6	Big City Bread Cafe	Athens	Breakfast, Sandwich	33.959298	-83.384128	4af18c02f964a5204de121e3
7	Taqueria Del Sol	Athens	Mexican, Spanish	36.126600	-86.789405	50f8a693e4b05404c154501e
8	The National	Athens	International, Southern	47.611228	-122.339494	4dc6d5311f6ef43b8a382bec
9	The Royal Peasant	Athens	Bar Food	33.938038	-83.387066	4b5b9ec3f964a520800b29e3
10	Transmetropolitan	Athens	Italian, Pizza, Sandwich	33.958487	-83.376544	4b0752a1f964a520fffb22e3
11	Trappeze Pub	Athens	Burger, Bar Food	33.958494	-83.379016	4e766e10ae60c3285192db22
12	Boll Weevil Cafe	Augusta	Desserts, Sandwich, Southern	31.314453	-85.854016	4bc0aaa62a89ef3b46def088

Using foursquare data, find out all rating for corresponding venue id. We need this rating to analyze the new restaurant in any area

```
Rating =pd.DataFrame(columns=['rating'])
for i,r in df.iterrows():

    url2 = 'https://api.foursquare.com/v2/venues/{}?client_id={}&client_secret={}&v={}'.format(r['
venue_id'], CLIENT_ID, CLIENT_SECRET, VERSION)
    try:
        rating = requests.get(url2).json()['response']['venue']['rating']
    except:
        rating=np.nan
    print(rating)
    Rating =Rating.append({'rating': rating}, ignore_index = True)
```

	Restaurant Name	City	Cuisines	Latitude	Longitude	venue_id	rating
0	Chick-fil-A	Albany	Fast Food	38.573684	-75.286200	5ab51bd3446ea6289e2bf8c6	8.1
1	Longhorn Steakhouse	Albany	American, Steak	40.540493	-105.076547	56d42849cd10d6b76f71a6d1	6.9
2	Mama's Boy Restaurant	Athens	Southern	41.890615	-87.629398	57cf6782498e577643efc1bd	6.3
3	Big City Bread Cafe	Athens	Breakfast, Sandwich	33.959298	-83.384128	4af18c02f964a5204de121e3	8.1
4	Taqueria Del Sol	Athens	Mexican, Spanish	36.126600	-86.789405	50f8a693e4b05404c154501e	8.2
5	The National	Athens	International, Southern	47.611228	-122.339494	4dc6d5311f6ef43b8a382bec	6.5
6	The Royal Peasant	Athens	Bar Food	33.938038	-83.387066	4b5b9ec3f964a520800b29e3	9.2
7	Transmetropolitan	Athens	Italian, Pizza, Sandwich	33.958487	-83.376544	4b0752a1f964a520fffb22e3	8.4
8	Trappeze Pub	Athens	Burger, Bar Food	33.958494	-83.379016	4e766e10ae60c3285192db22	9.1
9	Boll Weevil Cafe	Augusta	Desserts, Sandwich, Southern	31.314453	-85.854016	4bc0aaa62a89ef3b46def088	7.3
10	Mellow Mushroom	Augusta	Italian, Pizza, Sandwich	33.921787	-84.379529	4a4797edf964a520dca91fe3	8.6
11	Rhinehart's Oyster Bar	Augusta	Bar Food, Sandwich, Seafood	33.513429	-82.050222	4bc25dbc2a89ef3b7fbcf388	8.3
12	Takosushi	Augusta	Mexican, Southwestern, Sushi	34.849926	-82.399637	4b44135ff964a52023f125e3	7.6

Now we have concise data which can be used for machine learning algorithm. The venue id help us to collect rating for each restaurant. Now we focus on cuisine to decide which restaurant is famous for what type of food.

Methodology

In this project, we will use the basic methodology. As of now, we have done convert addresses into their equivalent latitude and longitude values. Then we will use the Foursquare API to explore neighborhoods in multiple cities like Athens, Augusta, Albany. After that, explore function to get the most common venue categories in each neighborhood, and then use this feature to group the neighborhoods into clusters K-means clustering algorithm will be use to complete this task. And also, the Folium library to visualize the neighborhoods in USA and their emerging clusters.
Based on data frame analysis above, we found out that American food has highest popularity in many city.

Lets Install required packages like folium, geopy.geocoder.

Now We have the restaurant data with cuisine, rating and city.

Here we are going to apply K-means to find some answer using this data

Install all required packages

```
import numpy as np
import pandas as pd
import csv
import requests
!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
# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors


!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
```

Here we have downloaded the data from Github repository.

```
df1= pd.read_csv('https://raw.githubusercontent.com/jaanu/Capstone_neighboring/master/Restaurant%20.csv')

df= pd.read_csv('https://raw.githubusercontent.com/jaanu/Capstone_neighboring/master/Restaurant%20.csv')
df
```

	Restaurant Name	City	Cuisines	Latitude	Longitude	rating
0	Chick-fil-A	Albany	American	40.142800	-82.990489	8.4
1	Longhorn Steakhouse	Albany	American	40.540493	-105.076547	7.1

[5]:
```
# one hot encoding
Athens_onehot = pd.get_dummies(df[['Cuisines']], prefix="", prefix_sep="")

# add city column back to dataframe
Athens_onehot['City'] = df['City']

# move city column to the first column
fixed_columns = [Athens_onehot.columns[-1]] + list(Athens_onehot.columns[:-1])
Athens_onehot = Athens_onehot[fixed_columns]

Athens_onehot.head()
```

[5]:

	City	American	Asian	Chinese	Indian	Italian	Japanese	Mexican	Seafood	Southern	Thai
0	Albany	1	0	0	0	0	0	0	0	0	0
1	Albany	1	0	0	0	0	0	0	0	0	0
2	Athens	0	0	0	0	0	0	0	0	1	0
3	Athens	1	0	0	0	0	0	0	0	0	0
4	Athens	0	0	0	0	0	0	1	0	0	0

find out all cuisine type and with mean value of each cuisine

[6]:
```
Athens_grouped = Athens_onehot.groupby('City').mean().reset_index()
Athens_grouped
```

[6]:

	City	American	Asian	Chinese	Indian	Italian	Japanese	Mexican	Seafood	Southern	Thai
0	Albany	1.000000	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	Athens	0.428571	0.0	0.0	0.000000	0.142857	0.000000	0.142857	0.000000	0.285714	0.000000
2	Augusta	0.000000	0.0	0.0	0.166667	0.166667	0.000000	0.000000	0.333333	0.166667	0.166667
3	Boise	0.500000	0.0	0.0	0.000000	0.300000	0.000000	0.100000	0.000000	0.000000	0.100000
4	Cedar Rapids/Iowa City	0.500000	0.1	0.1	0.000000	0.200000	0.000000	0.100000	0.000000	0.000000	0.000000
5	Clatskanie	1.000000	0.0	0.0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
6	Cochrane	0.000000	0.0	0.0	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000

Now lets group the city and calculate different cuisine, rating and coordinate.

```
from sklearn.cluster import KMeans
```

Group the city according to their cuisine

```
df2=df.groupby('City').count()
df2
```

City	Restaurant Name	Cuisines	Latitude	Longitude	rating
Albany	2	2	2	2	2
Athens	7	7	7	7	7
Augusta	6	6	6	6	6
Boise	10	10	10	10	10
Cedar Rapids/Iowa City	10	10	10	10	10
Clatskanie	1	1	1	1	1
Cochrane	1	1	1	1	1
Columbus	3	3	3	3	3

Now analyze cuisine category using onehot coding. Calculate the mean for every kind of cuisine in each area. So that we find out what category of food is popular in particular city.

```
num_top_venues = 5

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['City']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['City'] = Athens_grouped['City']

for ind in np.arange(Athens_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(Athens_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted
```

	City	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Albany	American	Thai	Southern	Seafood	Mexican
1	Athens	American	Southern	Mexican	Italian	Thai
2	Augusta	Seafood	Thai	Southern	Italian	Indian
3	Boise	American	Italian	Thai	Mexican	Southern
4	Cedar Rapids/Iowa City	American	Italian	Mexican	Chinese	Asian
5	Clatskanie	American	Thai	Southern	Seafood	Mexican
6	Cochrane	Japanese	Thai	Southern	Seafood	Mexican
7	Columbus	Seafood	Japanese	Italian	Thai	Southern

Now lets find popular cuisine in each city, so we will have an idea what kind of food can provide by restaurant .

Create data-frame with top cuisine in each city

```
# set number of clusters
kclusters = 5

Athens_grouped_clustering = Athens_grouped.drop('City', 1)

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

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

```
array([3, 0, 4, 0, 0, 3, 2, 1], dtype=int32)
```

Merge dataframe by kmean label

```
Athens_merged = df2

# add clustering labels
Athens_merged['Cluster Labels'] = kmeans.labels_

# merge toronto_grouped with toronto_data to add latitude/longitude for each neighborhood
Athens_merged = Athens_merged.join(neighborhoods_venues_sorted.set_index('City'), on='City')

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

City	Restaurant Name	Cuisines	Latitude	Longitude	rating	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Albany	2	2	2	2	2	3	American	Thai	Southern	Seafood	Mexican
Athens	7	7	7	7	7	0	American	Southern	Mexican	Italian	Thai
Augusta	6	6	6	6	6	4	Seafood	Thai	Southern	Italian	Indian
Boise	10	10	10	10	10	0	American	Italian	Thai	Mexican	Southern
Cedar Rapids/Iowa City	10	10	10	10	10	0	American	Italian	Mexican	Chinese	Asian

here is map indicating all restaurants

```
# create map of Manhattan using latitude and longitude values
map_Restaurant  = folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(df['Latitude'], df['Longitude'], df['City']):
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_Restaurant)

map_Restaurant
```

Kmeans cluster for all cities

top cuisine in each city

[7]:
```
num_top_venues = 2

for hood in Athens_grouped['City']:
    print("----"+hood+"----")
    temp = Athens_grouped[Athens_grouped['City'] == hood].T.reset_index()
    temp.columns = ['Cuisine','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')
```

```
----Albany----
    Cuisine  freq
0  American   1.0
1     Asian   0.0


----Athens----
    Cuisine  freq
0  American  0.43
1  Southern  0.29


----Augusta----
   Cuisine  freq
0  Seafood  0.33
1   Indian  0.17


----Boise----
    Cuisine  freq
0  American   0.5
1   Italian   0.3


----Cedar Rapids/Iowa City----
    Cuisine  freq
0  American   0.5
1   Italian   0.2
```

Result

```
Athens_merged.loc[Athens_merged['Cluster Labels'] == 0, Athens_merged.columns[[1] + list(range(5, Athens_merged.shape[1]))]]
```

City	Cuisines	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Athens	7	0	American	Southern	Mexican	Italian	Thai
Boise	10	0	American	Italian	Thai	Mexican	Southern
Cedar Rapids/Iowa City	10	0	American	Italian	Mexican	Chinese	Asian

```
Athens_merged.loc[Athens_merged['Cluster Labels'] == 1, Athens_merged.columns[[1] + list(range(5, Athens_merged.shape[1]))]]
```

City	Cuisines	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Columbus	3	1	Seafood	Japanese	Italian	Thai	Southern

```
Athens_merged.loc[Athens_merged['Cluster Labels'] == 2, Athens_merged.columns[[1] + list(range(5, Athens_merged.shape[1]))]]
```

City	Cuisines	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Cochrane	1	2	Japanese	Thai	Southern	Seafood	Mexican

```
Athens_merged.loc[Athens_merged['Cluster Labels'] == 4, Athens_merged.columns[[1] + list(range(5, Athens_merged.shape[1]))]]
```

City	Cuisines	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Augusta	6	4	Seafood	Thai	Southern	Italian	Indian

```
Athens_merged.loc[Athens_merged['Cluster Labels'] == 5, Athens_merged.columns[[1] + list(range(5, Athens_merged.shape[1]))]]
```

City	Cuisines	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue

Discussion

Based on cluster for each cities above, we believe that classification for each cluster can be done better with calculation of Cuisine categories (most common) in each cities. From this project we can say each cluster as follow:

Cluster 0: Athens, Bois, Iwoa cityr: American food

Cluster 1: Columbus : seafood

Cluster 2: Cochrane: Japanese

Cluster 4: Augusta : Seafood

Cluster 5: no category

So far we have calculated cuisine in each city with popularity.

Now we assume that American food is Athens, Bois, Iwoa City. So if we open any restaurant , it is highly possible for profit of the company. In Augusta, seafood are popular.

Conclusions

We used Foursquare API to captured data of common places all around USA. We have found restaurant with popular cuisine and rating. In Conclusion, we are able to find out our answer based on description. The data is small. In future we can use a large number of data. This method also can use for Hotels which is big part of industries. Overall this method able to provide some data so it can use to take decision.