Reinventing Restaurants: A Data Based Approach

How to survive in a COVID-19 pandemic economy?

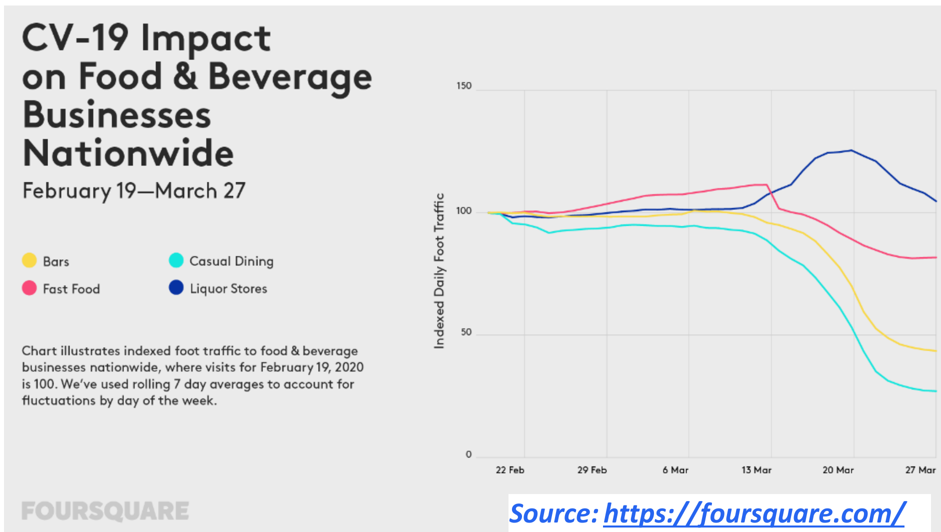
Capstone Project: The Battle of Neighborhoods

Authored by: Ayan Guha

Introduction

This project is a part of the IBM Data Science Specialization course where we learned what data scientists go through in real life when working with data. In this project we shall define a business problem, scrape data from the web and with the skills of a data scientist, create a compelling narrative from the data. Our tools for this project will be density-based clustering using K-means algorithm and other machine learning techniques. In this project, we shall provide a step-by-step description of the processes involved starting from defining the business problem, data preparation and the techniques that forge the data to create a final analysis. The project ends with a conclusion section which can be leveraged by business stakeholders to make decisions.

Business Problem:



Source: https://foursquare.com/

In January 2020, a new decade started with a hopeful bang - while a coronavirus (SARS-CoV-2), born from a wet market, started gaining a foothold in China. In next two months, coronavirus pandemic (COVID-19) stretched around the world forcing countries to enforce lockdown measures like stopping air travel and closing borders. In US, a nationwide emergency was declared on March 13 and most states started enforcing stringent lockdown measures. Apart from a tragically large number of human lives being lost, millions of Americans lost their livelihood resulting in the worst economic downturn since the Great Depression. The coronavirus recession - also known as ***Great Lockdown*** - shuttered schools, businesses, and restaurants, delivering the fastest and deepest economic shock in history. Consumer behavior changed drastically since the COVID-19 outbreak began. After an initial rush for groceries and stock piling of necessary supplies, families cloistered themselves in the safety of their homes - online shopping surged as replacement for old fashioned trips to grocery stores. The graph above (compiled by Foursquare) shows the impact of COVID-19 on Food & Beverage Businesses in US. As we can see, Casual Dining industry suffered the most. Nationally, foot traffic to sit-down casual dining restaurants decreased more than 73%, while visits to Quick Service Restaurants (QSR) were only down by 18%. Since QSRs offer drive through and take out, consumers felt more comfortable visiting those establishments. Interestingly, people visiting QSRs are spending less time in those locations. The week ending March 27, had 10% more customers spending less than 15 mins as compared to a month earlier. Majority of the customers (91%) preferred drive through or take-out as compared to dining in. Also, they preferred visiting QSRs closer to their homes than a QSR further away.

Reading this analysis, reminded me of my friend Mr. Lohanathan (LN), who owns an Indian Restaurant in Austin. Here is how our conversation went:

Restaurant during COVID-19 times: Waiting for customers

Me: *How is your restaurant doing in these troubling times?*

LN: *I am devastated. Our revenues have dropped more than 90%, we are open only for take-out. But very few customers are entering through the door.*

Me: *Sorry to hear that, maybe you should change your strategy. Remember the phrase "If the mountain will not come to Muhammad, then Muhammad must go to the mountain"*

LN: *I am now confused. What do you mean?*

Me: *If customers will not come to your shop, then the shop must go to the customers. It's time to take your restaurant mobile. Why not start a food truck?*

LN: *Good idea, but people are off the streets, there is no reason for food trucks to go to familiar lunch spots.*

Me: *What if we go directly to the neighborhoods? Being mobile has another advantage, you can source your supplies further away from the city. With food supply chain severely strained, you can now procure fresh vegetables, poultry, and meat from farms in Austin suburbs.*

LN: *Well, that is an advantage for sure.*

Me: *By traveling to residential areas, you have access to those customers who are self-quarantined but would like Indian food. You can also serve those spots that otherwise encounter difficulty with normal food delivery service.*

LN: *I am sold to this idea, but which neighborhoods should I visit?*

Me: *Let me put on my data scientist hat and help you find out the neighborhoods where your customers live.*

With this background, I was motivated to solve the following business problem statement.

Which neighborhoods in Austin are profitable for a mobile Indian restaurant to visit?

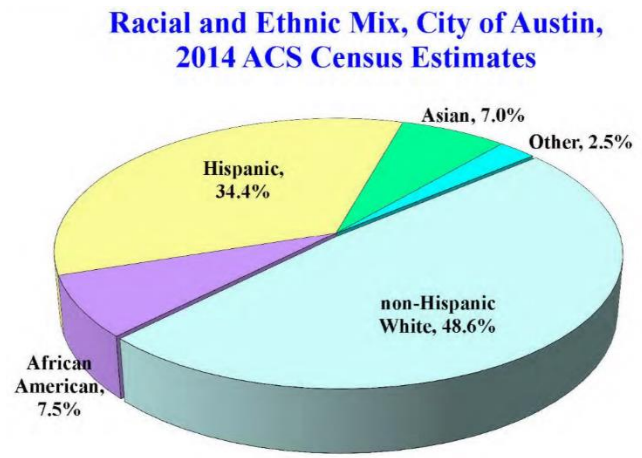
Apart from Mr. Lohanathan - a business owner who is looking for creative solutions to save his restaurant, this analysis will also be valuable to the following people:

- ***Business investors***, who are looking for opportunities to invest in mobile food delivery business.
- ***Event organizers***, who can organize block parties in open venues (like a school parking lot) where the invitees can enjoy casual dining while maintaining social distancing guidelines.
- ***City planners***, who can look at possibility of outdoor seating for mobile food vendors in shuttered streets so that citizens can safely practice social distancing during COVID-19 pandemic.
- ***Consumers***, who will be interested in finding out a new feature (namely, access to mobile food delivery) in their neighborhood.

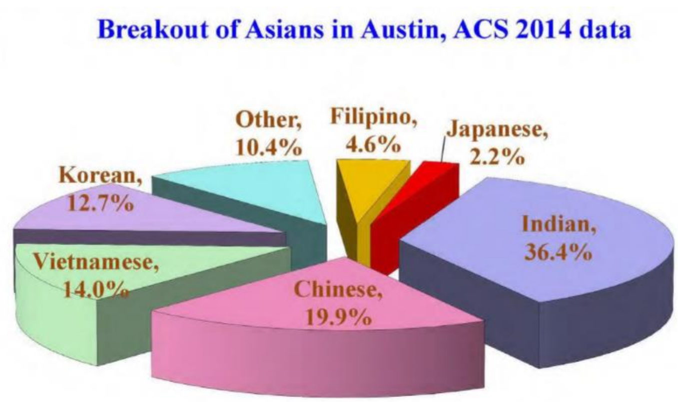
Before starting my data collection, I decided to research on the demographic characteristic of Indian population in Austin.

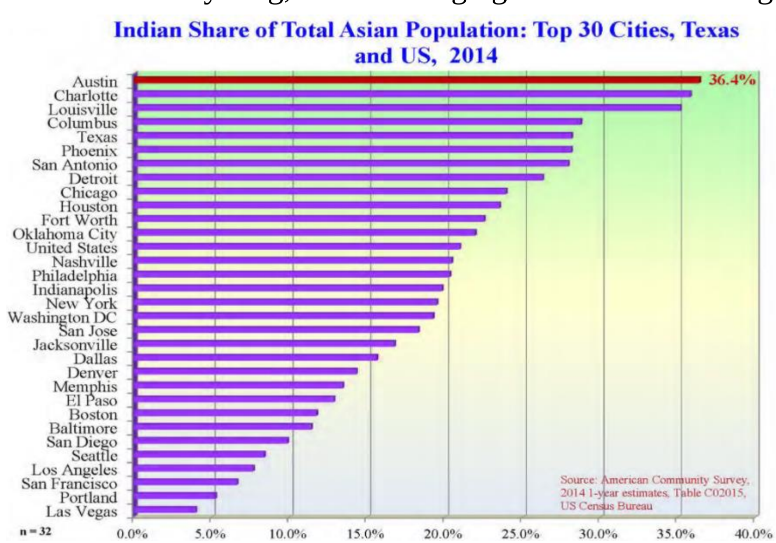
Background research:

http://www.austintexas.gov/department/about-asian-american-quality-life



In recent years, Austin has firmly established itself as a destination city for Asian families from a wide range of national backgrounds and ethnicities. Located in the heart of Texas, Austin has recently transformed from a college town to one of the largest, most vibrant cities in the U.S. Asians in Austin are the fastest growing demographic in terms of percentage year-to-year gain and are doubling in total size roughly every 12 years. Austin's burgeoning knowledge economy is a magnet for highly educated Asians whose Median Family Income (MFI) is the highest among any demographic group in the city. In terms of Asian MFI, Austin ranks the 3rd highest among nation's 30 largest cities. With many tech companies calling Austin their second home, the city is being called mini "Bay Area" with people moving in from western and eastern states of the U.S. Austin is currently the fastest growing city in the United States with young, recent college graduates and retiring baby boomers constituting majority of the new arrivals. The Asian community in Austin is highly diverse with significant subgroups formed by Indians, Chinese, Vietnamese, Koreans and Filipinos. ***Indians represent roughly 1/3rd of all Asians in Austin and is the fastest growing Asian subgroup***. With 36.4% share of the total Asian population, ***the Indian subgroup ranks first in the nation*** – no other cities among the top 30 cities in the country has a larger Indian share of Total Asian Population.





With such an encouraging demographic trend, Austin is the perfect city for an Indian mobile restaurant that would cater to its rapidly growing population.

Data Section:

The data from this project will be sourced from the following places:

- **data.austintexas.gov (https://data.austintexas.gov/) :**

A data portal that provides easy access to open data and information about the city government.

This code snippet collects the neighborhood boundaries from the City of Austin website.

```
# Importing data from the Austin Government website
from sodapy import Socrata
# First 2000 results, returned as JSON from API / converted to Python
client = Socrata("data.austintexas.gov",None)
results = client.get("nz5f-3t2e",limit=2000)
# Convert to pandas DataFrame
df_neighbors = pd.DataFrame.from_records(results)
df_neighbors.head()
```

	the_geom	_feature_i	planning_a	shape_area	shape_leng	label
0	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	1.0	ALLANDALE	65792689.5531	42253.1072105	Allandale
1	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	2.0	BARTON HILLS	88901714.7947	48353.9339254	Barton Hills
2	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	3.0	BOULDIN CREEK	33258999.9494	25667.3403756	Bouldin Creek
3	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	4.0	BRENTWOOD	44207756.068	29612.4036976	Brentwood
4	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	5.0	CENTRAL EAST AUSTIN	26970983.8606	22198.5298969	Central East Austin

The same portal also provides census tracts along with the demographic data each tract.

```
# Importing data from the Austin Government website
from sodapy import Socrata
# First 2000 results, returned as JSON from API
client = Socrata("austin-energy.data.socrata.com",None)
results = client.get("vi83-tbx3",limit=2000)
# Convert to pandas DataFrame
df_results = pd.DataFrame.from_records(results)
df_results.head()
```

	the_geom	tractce10	sum_totpop	sum_white	sum_black	sum_hispan	sum_asian	sum_other	sum_multi	sum_hu_tot	...	sum_age62_	sum_age651
0	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	000101	3611.0	3130.0	39.0	276.0	90.0	9.0	67.0	2076.0	...	81.0	38.0
1	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	000102	2552.0	2311.0	23.0	150.0	42.0	3.0	23.0	1153.0	...	126.0	70.0
2	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	000203	1546.0	1073.0	87.0	213.0	143.0	5.0	25.0	817.0	...	17.0	8.0
3	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	000204	3009.0	2396.0	43.0	323.0	161.0	8.0	78.0	1750.0	...	77.0	21.0
4	{'type': 'MultiPolygon', 'coordinates': [[[[-9...	000205	3394.0	2333.0	110.0	676.0	162.0	24.0	89.0	2153.0	...	50.0	35.0

❖ **United States Census Bureau (https://www.census.gov/data.html/) :**
The United States Census Bureau is a principal agency of the U.S. Federal Statistical System, responsible for producing data about the American people and economy. A snippet of the data obtained from the Census Bureau providing the population of Indian living in each Census Tract of Travis County.

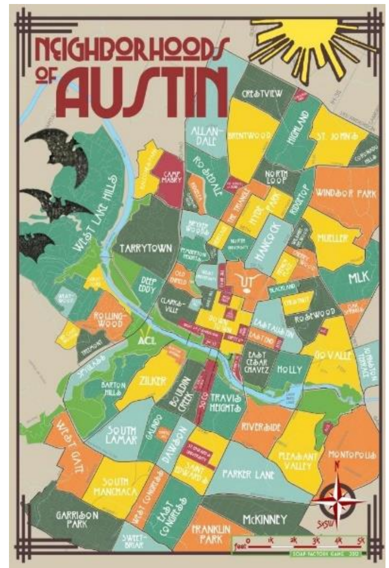
GEO_ID	POPGROUP_TTL	NAME	HCT001001
id	Popgroup Label	Geographic Area Name	Total
1400000US48453000302	Asian Indian alone (400-401)	Census Tract 3.02, Travis County, Texas	73
1400000US48453000401	Asian Indian alone (400-401)	Census Tract 4.01, Travis County, Texas	81
1400000US48453000500	Asian Indian alone (400-401)	Census Tract 5, Travis County, Texas	73
1400000US48453000601	Asian Indian alone (400-401)	Census Tract 6.01, Travis County, Texas	15
1400000US48453000603	Asian Indian alone (400-401)	Census Tract 6.03, Travis County, Texas	149
1400000US48453000604	Asian Indian alone (400-401)	Census Tract 6.04, Travis County, Texas	181
1400000US48453001100	Asian Indian alone (400-401)	Census Tract 11, Travis County, Texas	71
1400000US48453001714	Asian Indian alone (400-401)	Census Tract 17.14, Travis County, Texas	260

From Wikipedia: Travis County is a county in south central Texas. As of the 2010 census, the population was 1,024,266; the estimated population in 2019 was 1,273,954.[1] It is the fifth-most populous county in Texas. Its county seat is Austin,[2] the capital of Texas. Travis County is part of the Austin-Round Rock Metropolitan Statistical Area.

❖ **FOURSQUARE (https://foursquare.com/) :**
An American technology company that provides location platform for several businesses like UBER, TARGET, Twitter, SUBWAY, and many others. In this project, we shall make calls to the Foursquare API to search for venues as well as specific type of venues or restaurants around a given location. With the data available from FOURSQUARE, we shall explore neighborhoods, segment and group them into clusters to find similar neighborhoods in Austin. A density-based clustering algorithm, called K-means shall be used in this project to model the neighborhoods.

Data Cleaning:

The census data obtained from the City of Austin data portal identifies the census tracts (column "tractce10") as "object" datatype with 6 digits. The census data obtained from the United States Census Bureau identifies the census tracts (column "Geographic Area Name") as "Census Tract *. **, Travis County, Texas". To clean up the dataframe from Census Bureau, the extra strings were removed, and the data type was converted to integer. Similarly, the annoying "0"s in front of census tracts were removed by converting the "tractce10" column (from City of Austin data portal) to integer datatype. Finally, we have the Census Tract column of same type (namely, integer) for both data sets, which is essential for searching, filtering and merging the information from both datasets. Since the datasets didn't have any "NaN" or "non-standard missing values", no further cleaning was necessary, apart from dropping columns not needed for our analysis.



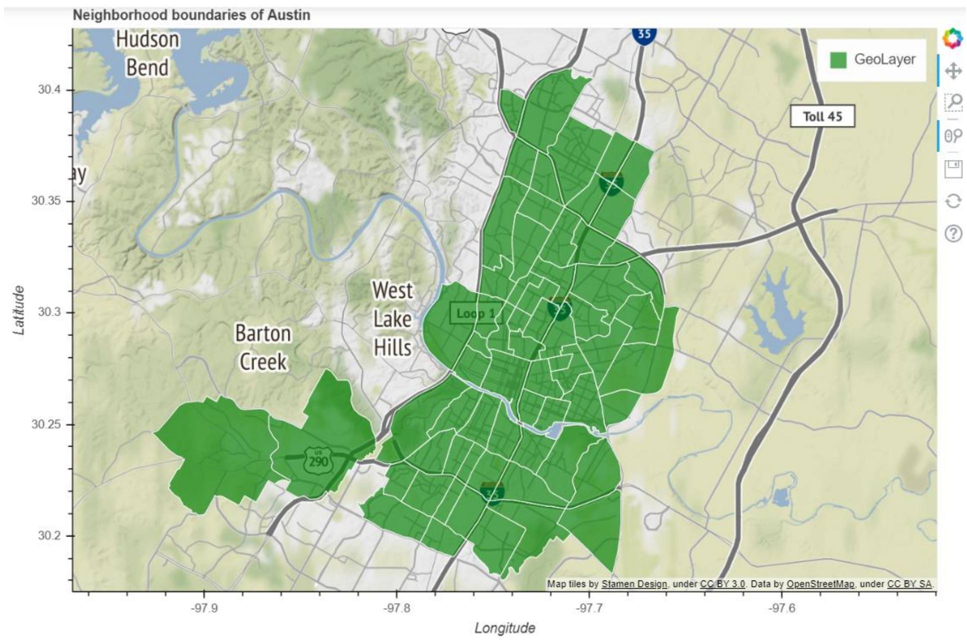
Visualizing Austin Neighborhoods:

Our analysis starts with creating a dataframe (***df_boundaries***) of the boundaries of Austin Neighborhoods. The dataframe was created by reading a GeoJson file obtained from the City of Austin data portal.

```
import pandas as pd
import geopandas as gpd
# Read the GeoJson file
df_boundaries = gpd.read_file("data/Boundaries_ Austin Neighborhood Planning Areas.geojson")
df_boundaries.head()
```

	planning_a	shape_area	label	shape_leng	_feature_i	geometry
0	ALLANDALE	65792689.5531	Allandale	42253.1072105	1.0	MULTIPOLYGON (((-97.73974 30.32808, -97.73962 ...
1	BARTON HILLS	88901714.7947	Barton Hills	48353.9339254	2.0	MULTIPOLYGON (((-97.79627 30.23398, -97.79767 ...
2	BOULDIN CREEK	33258999.9494	Bouldin Creek	25667.3403756	3.0	MULTIPOLYGON (((-97.75962 30.24211, -97.76031 ...
3	BRENTWOOD	44207756.068	Brentwood	29612.4036976	4.0	MULTIPOLYGON (((-97.73692 30.31449, -97.73757 ...
4	CENTRAL EAST AUSTIN	26970983.8606	Central East Austin	22198.5298969	5.0	MULTIPOLYGON (((-97.71925 30.27073, -97.71903 ...

A visualization of the neighborhood boundaries using Bokeh library:



Our next steps are: **(a)** calculate the centroid of each boundary
(b) extract Latitude & Longitude co-ordinates
(c) drop unnecessary columns to create a concise dataframe ***df_austin*** which has 65 rows and 4 columns.

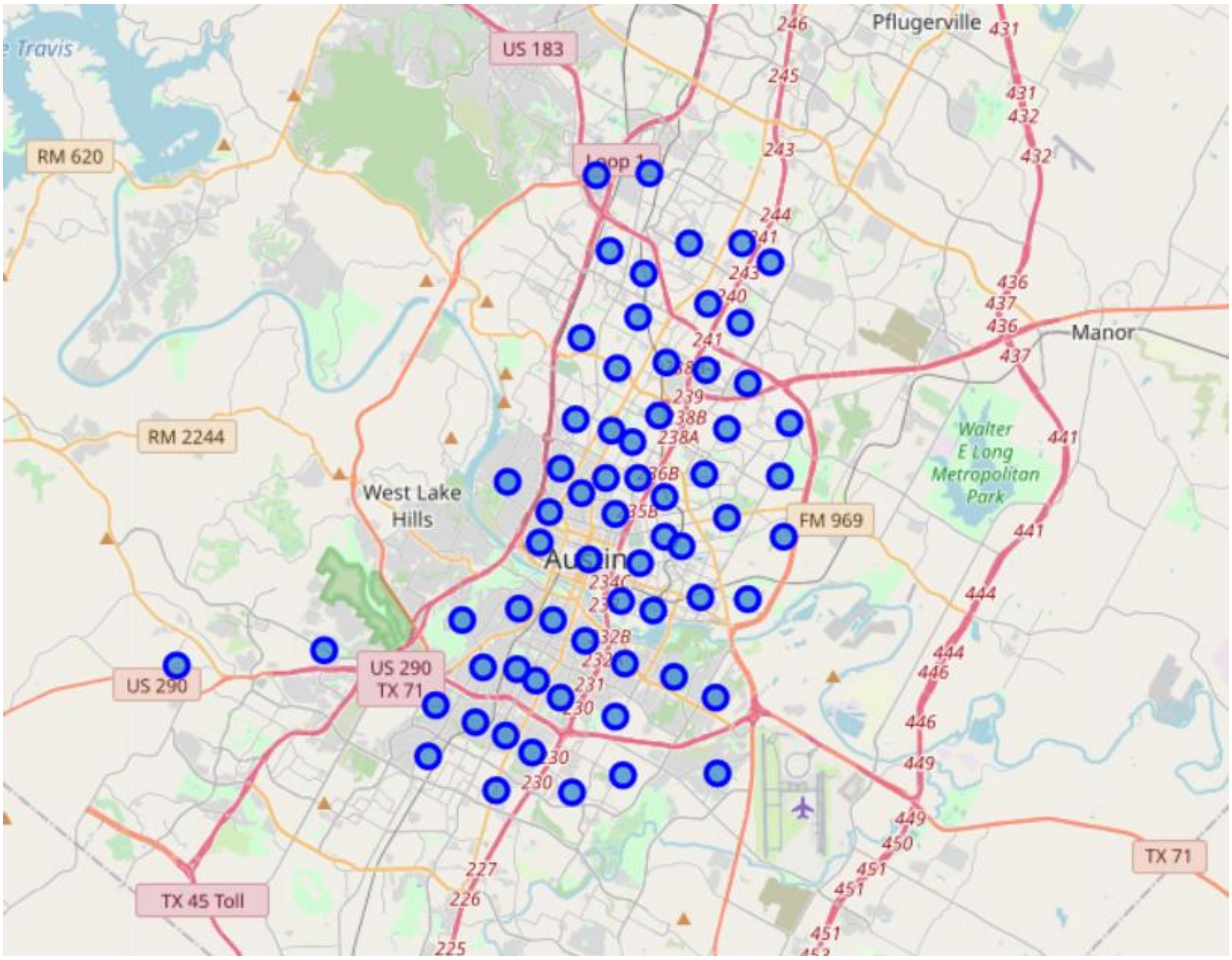
```
In [6]: # Let's create a new dataframe
df_austin = df_boundaries.copy()
# Let's drop columns not needed
df_austin.drop(['planning_a','shape_area','shape_leng','_feature_i','borders'], axis=1, inplace=True)
# Let's extract longitude & latitude values from centroid_column
df_austin['Latitude']=df_austin['centroid_column'].apply(lambda p: p.y)
df_austin['Longitude']=df_austin['centroid_column'].apply(lambda p: p.x)
# Let's rename two columns of the final DataFrame
df_austin.rename(columns={'label':'Neighborhood','centroid_column':'Centroid'},inplace=True)
print("Shape of dataframe of Austin Neighborhoods:",df_austin.shape)
df_austin.head()
```

```
Shape of dataframe of Austin Neighborhoods: (65, 4)
```

Out[6]:

	Neighborhood	Centroid	Latitude	Longitude
0	Allandale	POINT (-97.74517 30.34030)	30.340301	-97.745169
1	Barton Hills	POINT (-97.78837 30.25202)	30.252021	-97.788367
2	Bouldin Creek	POINT (-97.75563 30.25170)	30.251705	-97.755626
3	Brentwood	POINT (-97.73245 30.33062)	30.330625	-97.732451
4	Central East Austin	POINT (-97.72415 30.26974)	30.269742	-97.724153

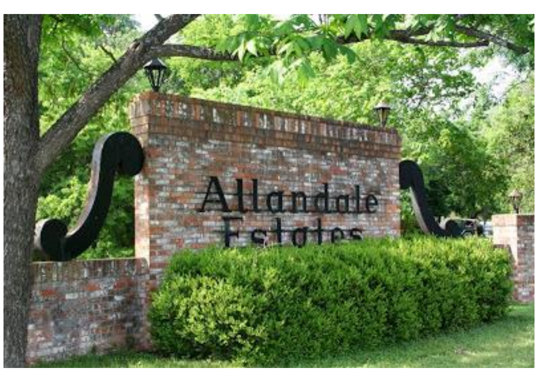
Here is a Leaflet map using the Folium library of the centroid positions of Austin neighborhoods.



Exploring the first neighborhood:

The first neighborhood in our neighborhood dataframe is ***"Allandale"***. By using the explore endpoint, latitude and longitude coordinates of the centroid along with my credentials, a regular API call was made to Foursquare to collect a list of maximum 100 popular spots around 1000m of Allandale. The call returned 38 locations which was put in the following dataframe ***(nearby_venues).***

The top 10 rows of the dataframe is shown here:



	name	categories	lat	lng
0	Barley Swine	New American Restaurant	30.341256	-97.738458
1	Yard Bar	Bar	30.342881	-97.738871
2	Bufalina Due	Pizza Place	30.341030	-97.738422
3	Three Little Pigs	Food Truck	30.340192	-97.738426
4	Lick Ice Creams Burnet Road	Ice Cream Shop	30.341143	-97.738408
5	Taqueria Arandas No. 3	Mexican Restaurant	30.341555	-97.739067
6	Growler Room	Food & Drink Shop	30.344500	-97.738420
7	Asahi Imports	Supermarket	30.336347	-97.739225
8	Hot Rod Coffee Trailer	Food Truck	30.341761	-97.738946
9	The Aristocrat Lounge	Lounge	30.340241	-97.738617

	Categories	Count
0	Food Truck	3
1	Mexican Restaurant	2
2	Pizza Place	2
3	Cosmetics Shop	1
4	Grocery Store	1
5	Supermarket	1
6	Thai Restaurant	1
7	Video Store	1
8	Food & Drink Shop	1
9	New American Restaurant	1

In Allandale, the most popular categories are "Food Trucks" followed by "Mexican Restaurants" and "Pizza parlors".

Exploring all the neighborhoods:

Following the above procedure, we can now explore all the neighborhoods in Austin. The resulting dataframe (***austin_venues***) has 3234 rows and 5 columns. The first 10 rows of ***austin_venues*** are as follows:

	Neighborhood	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Allandale	Barley Swine	30.341256	-97.738458	New American Restaurant
1	Allandale	Yard Bar	30.342881	-97.738871	Bar
2	Allandale	Bufalina Due	30.341030	-97.738422	Pizza Place
3	Allandale	Three Little Pigs	30.340192	-97.738426	Food Truck
4	Allandale	Lick Ice Creams Burnet Road	30.341143	-97.738408	Ice Cream Shop
5	Allandale	Taqueria Arandas No. 3	30.341555	-97.739067	Mexican Restaurant
6	Allandale	Growler Room	30.344500	-97.738420	Food & Drink Shop
7	Allandale	Asahi Imports	30.336347	-97.739225	Supermarket
8	Allandale	Hot Rod Coffee Trailer	30.341761	-97.738946	Food Truck
9	Allandale	The Aristocrat Lounge	30.340241	-97.738617	Lounge

	Categories	Venue Category
0	Mexican Restaurant	143
1	Food Truck	131
2	Coffee Shop	114
3	Park	73
4	Sandwich Place	71
5	Bar	71
6	Convenience Store	69
7	Hotel	69
8	Pizza Place	68
9	Taco Place	59

By counting how many times a particular category appears in Austin neighborhood, we can see that the most popular categories are "Mexican Restaurants" followed by "Food Trucks".

Austin has the fastest-growing food truck industry in the United States, and the second most food trucks per capita in the nation

Let's see which neighborhoods returned largest number of venues:

	Neighborhood	Venues
0	Bouldin Creek	100
1	Central East Austin	100
2	Downtown	100
3	East Cesar Chavez	100
4	Gateway	100
5	Riverside	100
6	UT	100
7	West University	100
8	Zilker	100
9	North University	89

Austin's second downtown is taking shape in the Gateway Neighborhood, which returned large number of venues along with other neighborhoods in Central Austin

```
In [22]: print('There are {} uniques categories.'.format(len(austin_venues['Venue Category'].unique())))

         There are 319 uniques categories.
```

There are 319 unique categories of venues in the ***austin_venues*** dataframe.

One hot encoding is a representation of categorical variables as binary vectors. The following code splits the column containing categorical features to 319 columns – each column contains "0" or "1".

```
# one hot encoding to create dummy columns
austin_onehot = pd.get_dummies(austin_venues[['Venue Category']],prefix="",prefix_sep="")
# add neighborhood column back to dataframe
austin_onehot['Neighborhood'] = austin_venues['Neighborhood']
# move neighborhood column to the first column
fixed_columns = [austin_onehot.columns[-1]] + list(austin_onehot.columns[:-1])
# create the DataFrame
austin_onehot = austin_onehot[fixed_columns]
# Display austin_onehot
print("Shape of one hot encoded dataframe of venues in Austin neighborhood:",austin_onehot.shape)
austin_onehot.head()
```

```
Shape of one hot encoded dataframe of venues in Austin neighborhood: (3234, 319)
```

	Yoga Studio	ATM	Adult Boutique	African Restaurant	American Restaurant	Antique Shop	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	...	Video Store	Vietnamese Restaurant
0	0	0	0	0	0	0	0	0	0	0	...	0	0
1	0	0	0	0	0	0	0	0	0	0	...	0	0
2	0	0	0	0	0	0	0	0	0	0	...	0	0
3	0	0	0	0	0	0	0	0	0	0	...	0	0
4	0	0	0	0	0	0	0	0	0	0	...	0	0

Let's group the one hot encoded dataframe by neighborhood and by the mean of the frequency of occurrence of each category.

```
austin_grouped = austin_onehot.groupby('Neighborhood').mean().reset_index()
print("Shape of grouped neighborhood dataframe:",austin_grouped.shape)
austin_grouped.head()
```

```
Shape of grouped neighborhood dataframe: (65, 319)
```

	Neighborhood	Yoga Studio	ATM	Adult Boutique	African Restaurant	American Restaurant	Antique Shop	Arcade	Argentinian Restaurant	Art Gallery	...	Video Store
0	Allandale	0.000000	0.0	0.0	0.0	0.000000	0.000000	0.0	0.00	0.0	...	0.026316
1	Barton Hills	0.000000	0.0	0.0	0.0	0.000000	0.000000	0.0	0.00	0.0	...	0.000000
2	Bouldin Creek	0.010000	0.0	0.0	0.0	0.010000	0.000000	0.0	0.00	0.0	...	0.000000
3	Brentwood	0.014085	0.0	0.0	0.0	0.028169	0.028169	0.0	0.00	0.0	...	0.014085
4	Central East Austin	0.030000	0.0	0.0	0.0	0.010000	0.000000	0.0	0.01	0.0	...	0.000000

5 rows × 319 columns

We can also check how many food trucks are in each neighborhood:

	Neighborhood	Food Truck
0	Central East Austin	0.160000
1	Southeast	0.125000
2	MLK-183	0.111111
3	East Cesar Chavez	0.090000
4	Georgian Acres	0.086957
5	Montopolis	0.083333
6	West Congress	0.083333
7	Franklin Park	0.083333
8	South River City	0.082192
9	Old Enfield	0.080000

Let's create a new dataframe with Food trucks and all Restaurants

```
# Create a new DataFrame for all restaurants
m = np.core.defchararray.find(austin_grouped.columns.values.astype(str),'Restaurant') >=0
austin_restaurant = pd.DataFrame(austin_grouped.values[:,m],austin_grouped.index,austin_grouped.columns[m])
# Another DataFrame for Food Truck
austin_truck2 = pd.DataFrame(austin_grouped[['Neighborhood', 'Food Truck']])
# Combine both the dataframes to create a
austin_res = pd.merge(austin_truck2,austin_restaurant,left_on=austin_truck2.index,right_on=austin_truck2.index,how='left')
austin_res.drop(['key_0'], axis = 1, inplace = True)
print("Shape of resturants dataframe:",austin_res.shape)
austin_res.head()
```

```
Shape of resturants dataframe: (65, 47)
```

	Neighborhood	Food Truck	African Restaurant	American Restaurant	Argentinian Restaurant	Asian Restaurant	Cajun / Creole Restaurant	Cantonese Restaurant	Caribbean Restaurant	Chinese Restaurant	...	Southern / Soul Food Restaurant	Sushi Restaurant	Sze Rest
0	Allandale	0.078947	0	0	0	0	0	0	0	0	...	0	0	
1	Barton Hills	0.181818	0	0	0	0	0	0	0	0	...	0	0	
2	Bouldin Creek	0.050000	0	0.01	0	0	0	0	0	0	...	0	0.01	
3	Brentwood	0.014085	0	0.028169	0	0.0140845	0	0	0	0	...	0	0	
4	Central East Austin	0.160000	0	0.01	0.01	0.01	0.01	0	0	0.01	...	0	0	

5 rows × 47 columns

Segmenting neighborhoods based on Restaurants & Food Trucks:

The following code creates a list of top 5 restaurants from each neighborhood.

```
# Number of top 5 most common venues
num_top_venues = 5
# Creating the list of top 5 most common venues
for hood in austin_res['Neighborhood']:
    # print the neighborhood in first line
    print("----" + hood + "----")
    temp = austin_res[austin_res['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq'] # Column headings
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float) # Freq is cast as float
    temp = temp.round({'freq': 2}) # Rounded by two decimal points
    # See the top 5 venues from all neighborhoods
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top_venues))
    print('\n')
```

```
----Allandale----
                  venue  freq
0            Food Truck  0.08
1    Mexican Restaurant  0.05
2       Thai Restaurant  0.03
3  New American Restaurant  0.03
4  Fast Food Restaurant  0.03
```

Top venue in Allandale neighborhood is "Food Truck" followed by Mexican, Thai, New American and Fast Food restaurants

The top 10 restaurants from each neighborhood is added to a dataframe (***neighborhood_res_sorted***). Dataframe ***neighborhood_res_sorted*** has 65 rows and 11 columns. Here are the 1st five rows:

Shape of grouped & sorted neighborhood restaurant dataframe: (65, 11)

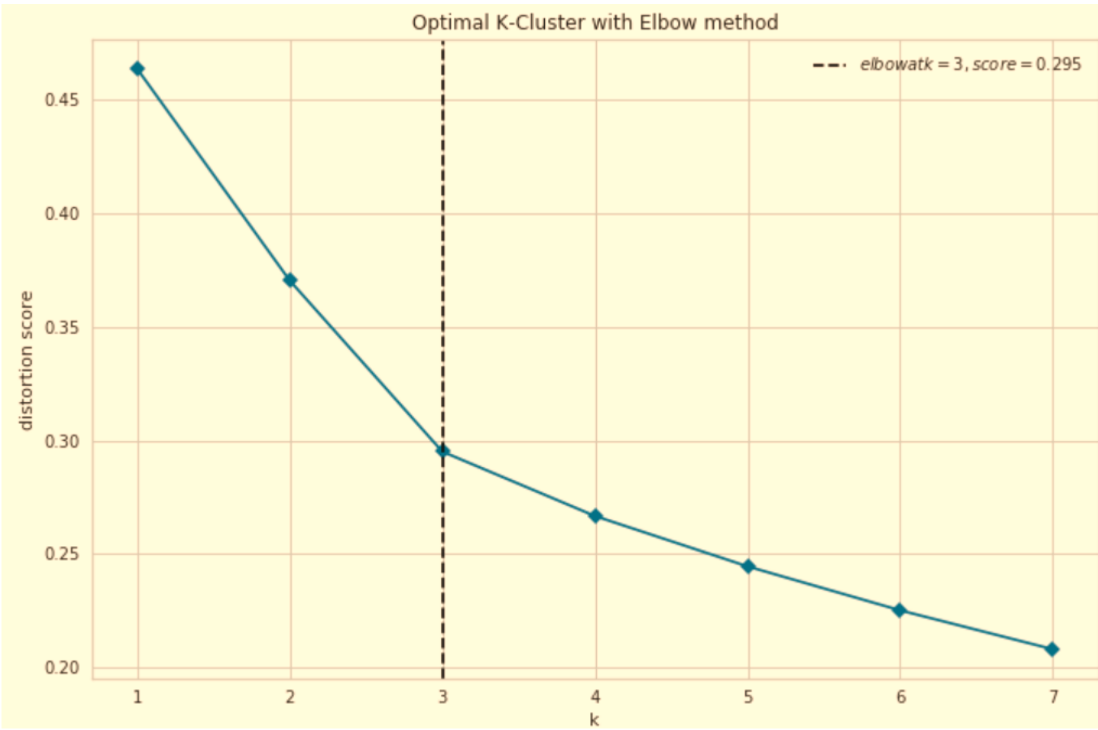
	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue
0	Allandale	Food Truck	Mexican Restaurant	Thai Restaurant	New American Restaurant	Fast Food Restaurant	Ethiopian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
1	Barton Hills	Food Truck	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
2	Bouldin Creek	Food Truck	New American Restaurant	Italian Restaurant	Thai Restaurant	Mexican Restaurant	Restaurant	Vegetarian / Vegan Restaurant	Japanese Restaurant	Sushi Restaurant
3	Brentwood	Mexican Restaurant	American Restaurant	Thai Restaurant	Vietnamese Restaurant	Asian Restaurant	Falafel Restaurant	Korean Restaurant	Mediterranean Restaurant	Japanese Restaurant
4	Central East Austin	Food Truck	Mexican Restaurant	Restaurant	Italian Restaurant	American Restaurant	Argentinian Restaurant	Asian Restaurant	Cajun / Creole Restaurant	Chinese Restaurant

Now, we have two data frames related to food service businesses in Austin:

- ***austin_res,*** hot-encoded dataframe which shall be used for predictive analysis
- ***neighborhood_res_sorted,*** dataframe segmented by 10 most common restaurants in each neighborhood.

k-Means Clustering of neighborhoods based on Restaurants & Food Trucks:

In cluster analysis, the elbow method is used to determine the number of clusters in a data set. The ***kElbowVisualizer*** implements the ***"elbow"*** method to select the optimal number of clusters by fitting the model with a range of values for K (number of clusters). ***"elbow",*** which is the point of inflection on the relationship between distortion score and number of clusters, is a good indication that the underlying model fits best at that point.




```
from sklearn.cluster import KMeans
from yellowbrick.cluster import KElbowVisualizer
# Instantiate the clustering model and visualizer
model = KMeans()
titleKElbow = "Optimal K-Cluster with Elbow method"
visualizer = KElbowVisualizer(model, k=(1,8),timings=False,title=titleKElbow,size=(800,520))
visualizer.fit(austin_res_clustering)        # Fit the data to the visualizer
visualizer.show()        # Finalize and render the figure
```

"elbow" analysis shows that there are 3 clusters in this dataset

```
# set number of clusters
kclusters = 3
# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(austin_res_clustering)
# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:30]
```

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

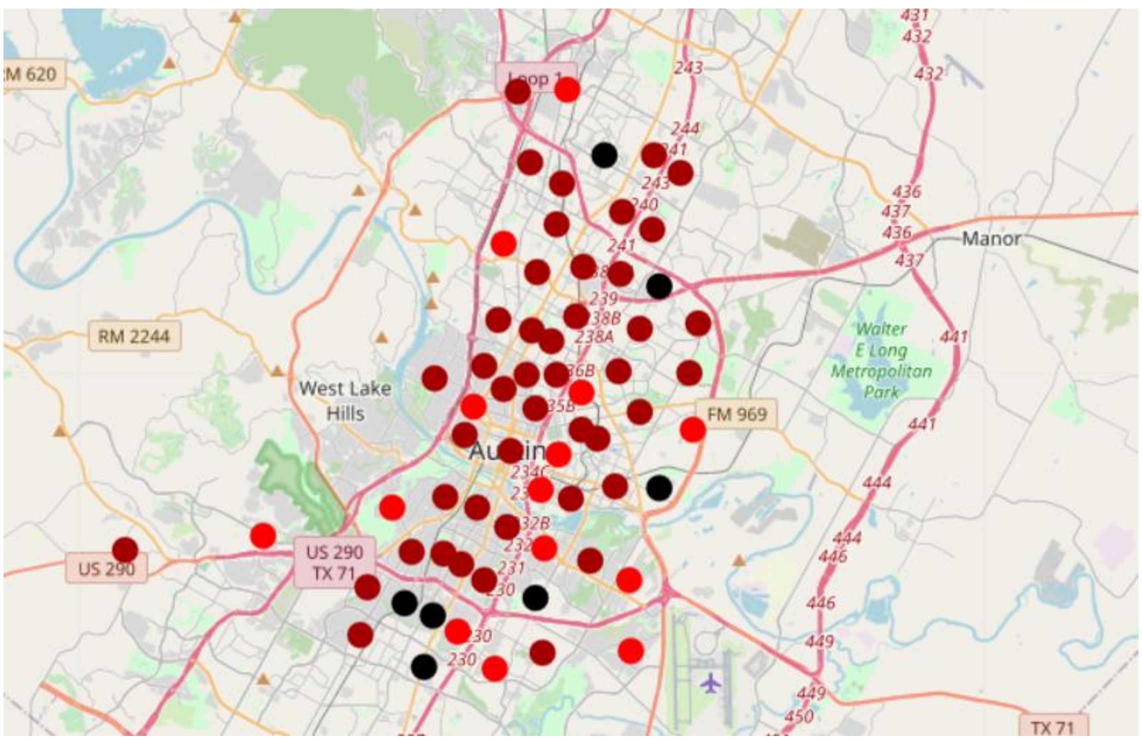
Using the hot-encoded dataframe, we have cluster labels from each neighborhood. Let's merge the labels with the segmented dataframe (***neighborhood_res_sorted***) to find the ten top restaurants for each cluster.

```
# Let's add clustering labels to the restaurant dataframe
neighborhoods_res_sorted.insert(0, 'Cluster Labels', kmeans.labels_)
# New Dataframe copied from austin neighborhood dataframe
restaurant_merged = df_austin.copy()
# Merge the new dataframe with neighborhoods_venues_sorted to add latitude/longitude for each neighborhood
restaurant_merged = restaurant_merged.join(neighborhoods_res_sorted.set_index('Neighborhood'), on='Neighborhood')
# Let's drop the Centroid column
restaurant_merged.drop(["Centroid"],axis=1,inplace=True)
# Display the sorted dataframe
print("Shape of the final popular restaurant dataframe:",restaurant_merged.shape)
restaurant_merged.head()
```

```
Shape of the final popular restaurant dataframe: (65, 14)
```

	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue
0	Allandale	30.340301	-97.745169	0	Food Truck	Mexican Restaurant	Thai Restaurant	New American Restaurant	Fast Food Restaurant	Ethiopian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
1	Barton Hills	30.252021	-97.788367	0	Food Truck	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
2	Bouldin Creek	30.251705	-97.755626	1	Food Truck	New American Restaurant	Italian Restaurant	Thai Restaurant	Mexican Restaurant	Restaurant	Vegetarian / Vegan Restaurant	Japanese Restaurant	Sushi Restaurant
3	Brentwood	30.330625	-97.732451	1	Mexican Restaurant	American Restaurant	Thai Restaurant	Vietnamese Restaurant	Asian Restaurant	Falafel Restaurant	Korean Restaurant	Mediterranean Restaurant	Japanese Restaurant
4	Central East Austin	30.269742	-97.724153	0	Food Truck	Mexican Restaurant	Restaurant	Italian Restaurant	American Restaurant	Argentinian Restaurant	Asian Restaurant	Cajun / Creole Restaurant	Chinese Restaurant

Visualizing the clusters:



Markers: (a) Black: Cluster #2; (b) Magenta: Cluster #1; (c) Red: Cluster #0

Characteristic of each cluster:

Let's inspect the most common venue from the 1st cluster:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
0	Allandale	Food Truck	Mexican Restaurant	Thai Restaurant	New American Restaurant	Fast Food Restaurant	Ethiopian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant
1	Barton Hills	Food Truck	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
4	Central East Austin	Food Truck	Mexican Restaurant	Restaurant	Italian Restaurant	American Restaurant	Argentinian Restaurant	Asian Restaurant	Cajun / Creole Restaurant	Chinese Restaurant	Indian Restaurant
11	East Cesar Chavez	Food Truck	Mexican Restaurant	Restaurant	American Restaurant	Argentinian Restaurant	Asian Restaurant	Fast Food Restaurant	Indian Restaurant	Korean Restaurant	Mediterranean Restaurant
12	East Congress	Food Truck	Restaurant	Asian Restaurant	Mexican Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
13	East Oak Hill	Food Truck	Restaurant	Italian Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
14	Franklin Park	Food Truck	Mexican Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
28	MLK-183	Food Truck	American Restaurant	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
29	Montopolis	Food Truck	American Restaurant	Latin American Restaurant	Mexican Restaurant	Fast Food Restaurant	Falafel Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
31	North Burnet	Food Truck	Fast Food Restaurant	Mexican Restaurant	American Restaurant	Asian Restaurant	Mediterranean Restaurant	Italian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant
35	Old Enfield	Food Truck	American Restaurant	Mexican Restaurant	Fast Food Restaurant	Italian Restaurant	Falafel Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
41	Riverside	Food Truck	Mexican Restaurant	Vietnamese Restaurant	Fast Food Restaurant	New American Restaurant	Vegetarian / Vegan Restaurant	Asian Restaurant	Korean Restaurant	Mediterranean Restaurant	Chinese Restaurant
48	Southeast	Food Truck	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
55	Upper Boggy Creek	Food Truck	American Restaurant	Mexican Restaurant	Fast Food Restaurant	Ethiopian Restaurant	Italian Restaurant	Mediterranean Restaurant	Peruvian Restaurant	Ramen Restaurant	Restaurant

Most common venue from the 2nd cluster is as follows:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
2	Bouldin Creek	Food Truck	New American Restaurant	Italian Restaurant	Thai Restaurant	Mexican Restaurant	Restaurant	Vegetarian / Vegan Restaurant	Japanese Restaurant	Sushi Restaurant	Ramen Restaurant
3	Brentwood	Mexican Restaurant	American Restaurant	Thai Restaurant	Vietnamese Restaurant	Asian Restaurant	Falafel Restaurant	Korean Restaurant	Mediterranean Restaurant	Japanese Restaurant	Food Truck
5	Chestnut	American Restaurant	Mexican Restaurant	Food Truck	Southern / Soul Food Restaurant	Italian Restaurant	Mediterranean Restaurant	Peruvian Restaurant	Restaurant	Cajun / Creole Restaurant	Caribbean Restaurant
7	Crestview	Mexican Restaurant	Japanese Restaurant	Food Truck	Korean Restaurant	Asian Restaurant	Thai Restaurant	Ethiopian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
8	Dawson	Mexican Restaurant	Food Truck	Hawaiian Restaurant	Cuban Restaurant	African Restaurant	Argentinian Restaurant	Fast Food Restaurant	Indian Restaurant	Indian Chinese Restaurant	American Restaurant
9	Heritage Hills	Chinese Restaurant	Vietnamese Restaurant	Ethiopian Restaurant	Israeli Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
10	Downtown	Food Truck	Italian Restaurant	Sushi Restaurant	New American Restaurant	Restaurant	Seafood Restaurant	French Restaurant	American Restaurant	Cajun / Creole Restaurant	Middle Eastern Restaurant
15	Galindo	Mexican Restaurant	Food Truck	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
16	Highland	Korean Restaurant	Chinese Restaurant	Mexican Restaurant	Vietnamese Restaurant	Peruvian Restaurant	American Restaurant	Caribbean Restaurant	Indian Restaurant	Middle Eastern Restaurant	Japanese Restaurant
17	Garrison Park	Mexican Restaurant	Food Truck	Chinese Restaurant	Fast Food Restaurant	Seafood Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
18	Gateway	American Restaurant	Asian Restaurant	Mexican Restaurant	Seafood Restaurant	Italian Restaurant	Vegetarian / Vegan Restaurant	Greek Restaurant	French Restaurant	New American Restaurant	Fast Food Restaurant
19	Georgian Acres	Fast Food Restaurant	Vietnamese Restaurant	Mexican Restaurant	Food Truck	Tex-Mex Restaurant	Cuban Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant

Here are the most common venues from the 3rd cluster:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th Most Common Venue
6	Coronado Hills	Mexican Restaurant	Fast Food Restaurant	Asian Restaurant	South American Restaurant	Comfort Food Restaurant	Vietnamese Restaurant	Falafel Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant
24	Johnston Terrace	Mexican Restaurant	Vietnamese Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
30	North Austin Civic Association	Mexican Restaurant	Vietnamese Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
37	Parker Lane	Vietnamese Restaurant	Mexican Restaurant	Seafood Restaurant	Asian Restaurant	Fast Food Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant
38	South Manchaca	Mexican Restaurant	Fast Food Restaurant	Vietnamese Restaurant	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant
46	West Congress	Mexican Restaurant	Food Truck	Ethiopian Restaurant	Indian Restaurant	Indian Chinese Restaurant	Hawaiian Restaurant	Greek Restaurant	Gluten-free Restaurant	French Restaurant	Fast Food Restaurant
51	Sweetbriar	Mexican Restaurant	Fast Food Restaurant	Food Truck	Sushi Restaurant	Chinese Restaurant	Indian Restaurant	Tex-Mex Restaurant	Asian Restaurant	American Restaurant	Indian Chinese Restaurant

Conclusion of Neighborhood clustering analysis:

The three unique clusters of food establishments in Austin neighborhoods are as follows:

- Cluster #1 comprises of 14 neighborhoods where 1st most common venue is Food Truck. From the leaflet map, we see that this cluster comprises of neighborhoods from East and South-East.
- Cluster #2 comprises of 44 neighborhoods where multicuisine restaurants are 1st most common venues. This cluster comprises of most of the neighborhoods along the North-South corridor.
- Cluster #3 comprises of 7 neighborhoods where 1st most common venue is Mexican Restaurant. 5 members of this cluster are neighborhoods in South Austin.

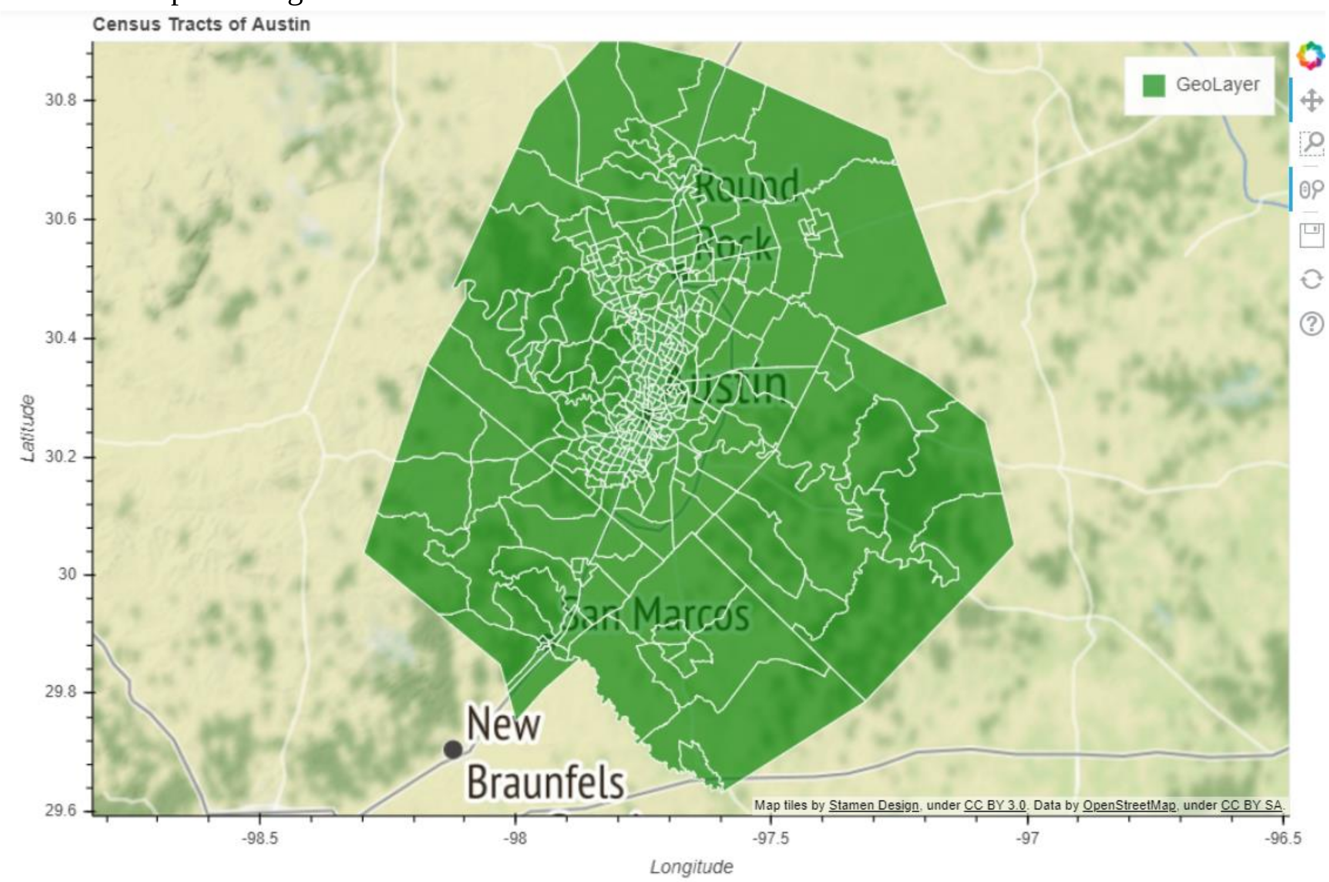
Now that we know where most of the food establishments are clustered, we would like to narrow our search for Indian restaurants with respect to Indian population in Austin. Since the customer base for Mr. Lohanathan's new venture will be primarily Indians, he would like to know which neighborhoods have a larger share of Asian Indians and if that neighborhood is under-served as far as Indian Restaurants are concerned. For the next part of the project, we shall analyze the demographic data from the U.S. Census Bureau (which provides population count of Asian Indians) along with the demographic data available in City of Austin data portal. We shall use Foursquare API to find the number of Indian Restaurants in each census tract and explore if there is any correlation between the Asian Indian population and Indian Restaurant in each census tract.

Visualizing the Asian-Indian Population in Austin:

A dataframe (***df_census_tracts***) was created by reading a GeoJson file from the City of Austin data portal followed by data cleanup (dropping un-necessary columns and renaming columns). Here is the head of the dataframe:

	tractce10	sum_totpop	sum_white	sum_black	sum_hispan	sum_asian	geometry
0	000101	3611.0	3130.0	39.0	276.0	90.0	MULTIPOLYGON (((-97.75153 30.33456, -97.75037 ...
1	000102	2552.0	2311.0	23.0	150.0	42.0	MULTIPOLYGON (((-97.76028 30.34086, -97.75999 ...
2	000203	1546.0	1073.0	87.0	213.0	143.0	MULTIPOLYGON (((-97.73144 30.31778, -97.73174 ...
3	000204	3009.0	2396.0	43.0	323.0	161.0	MULTIPOLYGON (((-97.74056 30.29890, -97.74094 ...
4	000205	3394.0	2333.0	110.0	676.0	162.0	MULTIPOLYGON (((-97.72808 30.32322, -97.72825 ...

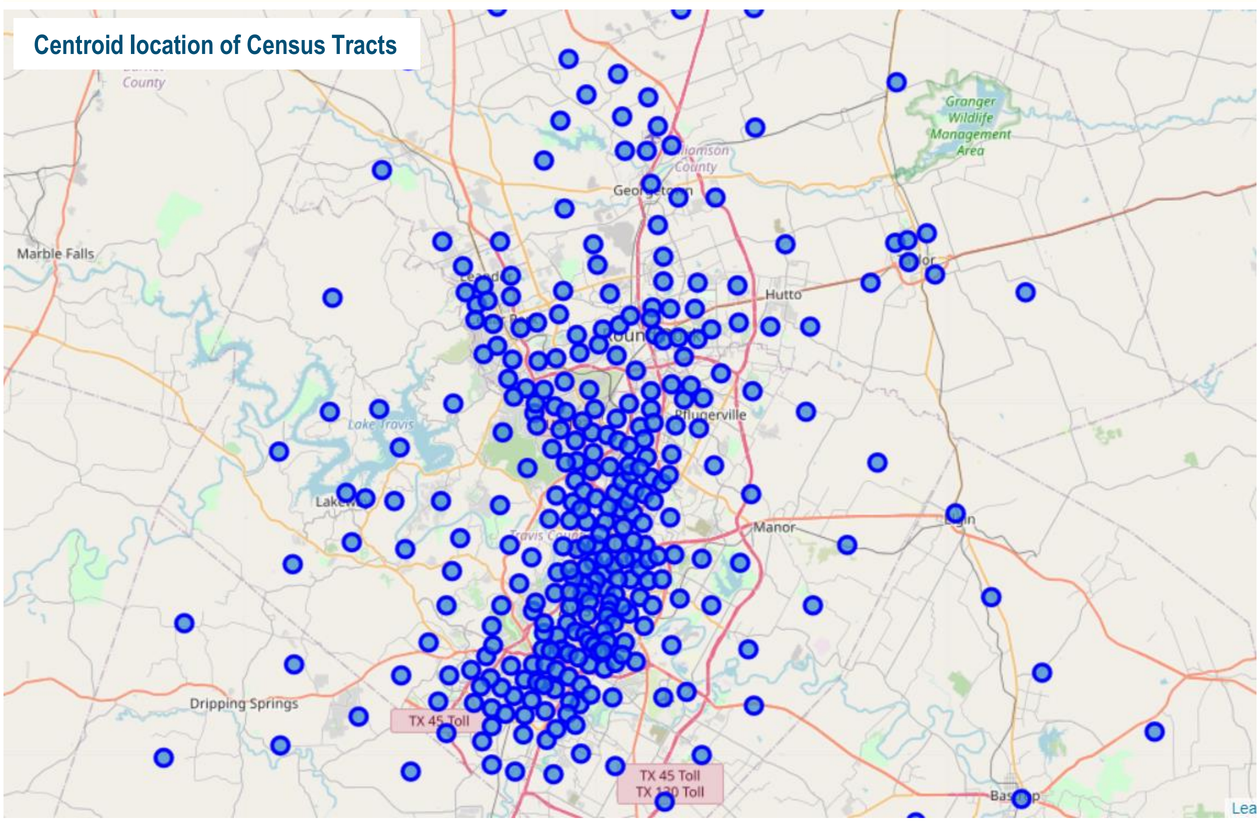
A Bokeh map showing the boundaries for 350 census tracts in the dataframe:



Our next steps would be to find the centroids of these census tracts, extract Latitude and Longitude co-ordinates so that we can visualize the centroids in a leaflet map built with Folium library.

```python
# Renaming the geometry column to borders
df_census_tracts = df_census_tracts.rename(columns={'geometry':'borders'}).set_geometry('borders')
# Now, we create centroids and make it the geometry
df_census_tracts['centroid'] = df_census_tracts.centroid
df_census_tracts = df_census_tracts.set_geometry('centroid')
# Let's extract longitude & latitude values from centroid_column
df_census_tracts['Latitude'] = df_census_tracts['centroid'].apply(lambda p: p.y)
df_census_tracts['Longitude'] = df_census_tracts['centroid'].apply(lambda p: p.x)
df_census_tracts.drop(['borders','centroid'],axis = 1,inplace = True)
df_census_tracts.head(10)
```

	tractce10	sum_totpop	sum_white	sum_black	sum_hispan	sum_asian	Latitude	Longitude
0	000101	3611.0	3130.0	39.0	276.0	90.0	30.323164	-97.753269
1	000102	2552.0	2311.0	23.0	150.0	42.0	30.325969	-97.767229
2	000203	1546.0	1073.0	87.0	213.0	143.0	30.312628	-97.736659
3	000204	3009.0	2396.0	43.0	323.0	161.0	30.307167	-97.744464
4	000205	3394.0	2333.0	110.0	676.0	162.0	30.326162	-97.734317
5	000206	2687.0	2183.0	79.0	289.0	66.0	30.326154	-97.743515
6	000302	4939.0	3661.0	88.0	620.0	420.0	30.302560	-97.726503
7	000304	3045.0	2065.0	103.0	630.0	177.0	30.309376	-97.716906
8	000305	3223.0	2524.0	49.0	426.0	128.0	30.314872	-97.726302
9	000306	2697.0	1409.0	199.0	847.0	190.0	30.297156	-97.700273



Centroid location of Census Tracts

U.S Census Data for Travis County:

For our demography analysis, we selected Travis County whose county seat is Austin. With population more than 1.3 million, Travis County is the fifth-most populous county in Texas. Travis county census tract data for Asian Indians, is downloaded from U.S. Census Bureau and loaded in dataframe (***df_asian_indian***). Here is the head of the dataframe:

```
Shape of Travis County census_tracts for Asian Indians (42, 4)
```

	geo_id	tract_name	tractce10	sum_indian
0	1400000US48453000302	Census Tract 3.02, Travis County, Texas	302	73
1	1400000US48453000401	Census Tract 4.01, Travis County, Texas	401	81
2	1400000US48453000500	Census Tract 5, Travis County, Texas	500	73
3	1400000US48453000601	Census Tract 6.01, Travis County, Texas	601	15
4	1400000US48453000603	Census Tract 6.03, Travis County, Texas	603	149

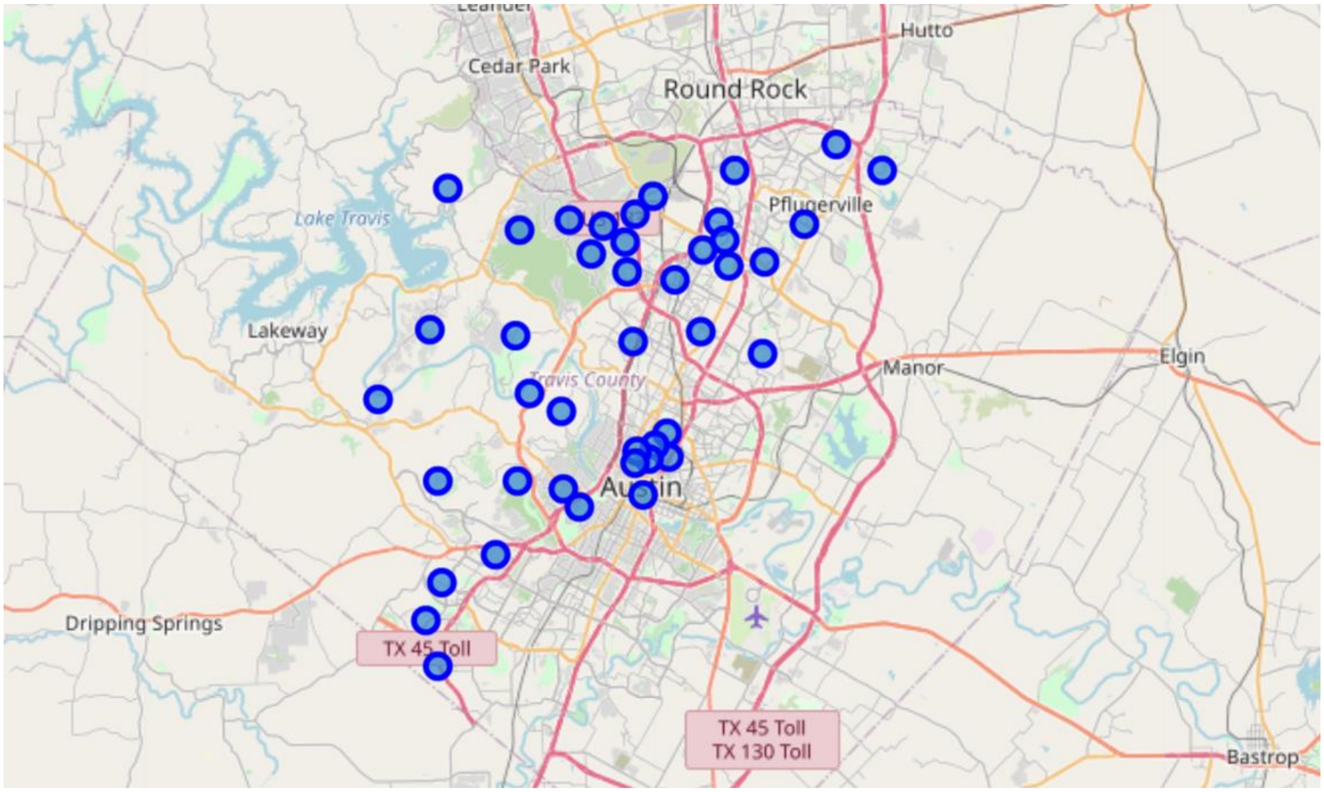
Since the City of Austin demographic dataframe (***df_census_tract***) did not have the population count of Asian Indians, we merged the two data-frames (***df_census_tract*** and ***df_asian_indian***) to create Travis County population dataframe (***df_travis***)

```
# Let's create a new dataframe from the census_tracts
df_census_copy = df_census_tracts.copy()
# Let's create a list of census tracts representing Travis County
tract_list = df_asian_indian['tractce10'].tolist()
# Select rows in the census dataframe using tract_list
df_new = df_census_copy[df_census_copy.tractce10.isin(tract_list)]
# Reset index
df_new.reset_index(drop=True,inplace=True)
# Merge two dataframes to create our final dataframe for Travis County
df_new2 = pd.merge(df_new,df_asian_indian,left_on='tractce10',right_on='tractce10',how='left')
# Rearrange the columns in the DataFrame
df_travis = df_new2[['tractce10','sum_totpop','sum_white','sum_black','sum_hispan','sum_indian','Latitude','Longitude']]
# Rename the columns in the DataFrame for clarity
df_travis = df_travis.rename(columns={'tractce10':'Census_Tract','sum_totpop':'Total_population','sum_white':'White','sum_black'
df_travis.head()
```

	Census_Tract	Total_population	White	African_Americans	Hispanic	Indian	Latitude	Longitude
0	302	4939.0	3661.0	88.0	620.0	73.0	30.302560	-97.726503
1	401	3574.0	2446.0	128.0	445.0	81.0	30.288248	-97.725056
2	500	4518.0	3305.0	67.0	545.0	73.0	30.296063	-97.734173
3	601	9207.0	4491.0	494.0	1972.0	15.0	30.287412	-97.738044
4	603	7793.0	5121.0	149.0	972.0	149.0	30.291992	-97.746270

Centroids of Census tracts in Travis County:

Let us visualize the centroids of the census tracts in Travis County in a Leaflet map built using the Folium library.



Exploring Travis County census tracts using Foursquare API: By using the explore endpoint, centroid latitude and longitude coordinates along with my credentials, a regular API call was made to Foursquare to explore popular spots around 1000m of each census tract. The call returned 1315 venues which was put in the dataframe ***(travis_venues)*** along with Census Tract number and GPS coordinates**.**

The top 10 rows of ***travis_venues*** are shown below:

	Census_Tract	Venue	Venue Latitude	Venue Longitude	Venue Category
0	302	Antonelli's Cheese Shop	30.304378	-97.726498	Cheese Shop
1	302	Quack's 43rd St Bakery	30.304731	-97.726511	Bakery
2	302	Hyde Park Bar & Grill	30.304222	-97.726705	American Restaurant
3	302	Antonelli's Cheese House	30.304281	-97.726466	Cheese Shop
4	302	Asti Trattoria	30.304794	-97.726176	Italian Restaurant
5	302	Fresh Plus Grocery	30.304841	-97.726369	Grocery Store
6	302	Mother's Cafe & Garden	30.304420	-97.726027	Vegetarian / Vegan Restaurant
7	302	Julio's	30.304624	-97.726398	Mexican Restaurant
8	302	Uncle Nicky's	30.304278	-97.726667	Italian Restaurant
9	302	Juiceland	30.307210	-97.724873	Juice Bar

Which venue categories are most popular in Travis County?

As shown in the adjacent table, ***sandwich places are most popular in Travis County*** followed by coffee shops. Fast food restaurants, Pizza place and Food trucks are more popular than any dine-in restaurants

	Categories	Count
0	Sandwich Place	45
1	Coffee Shop	39
2	Fast Food Restaurant	36
3	Pizza Place	34
4	Food Truck	32
5	Mexican Restaurant	29
6	Park	26
7	Hotel	25
8	Gym	25
9	Grocery Store	24

Which Census Tracts returned the largest number of venues?

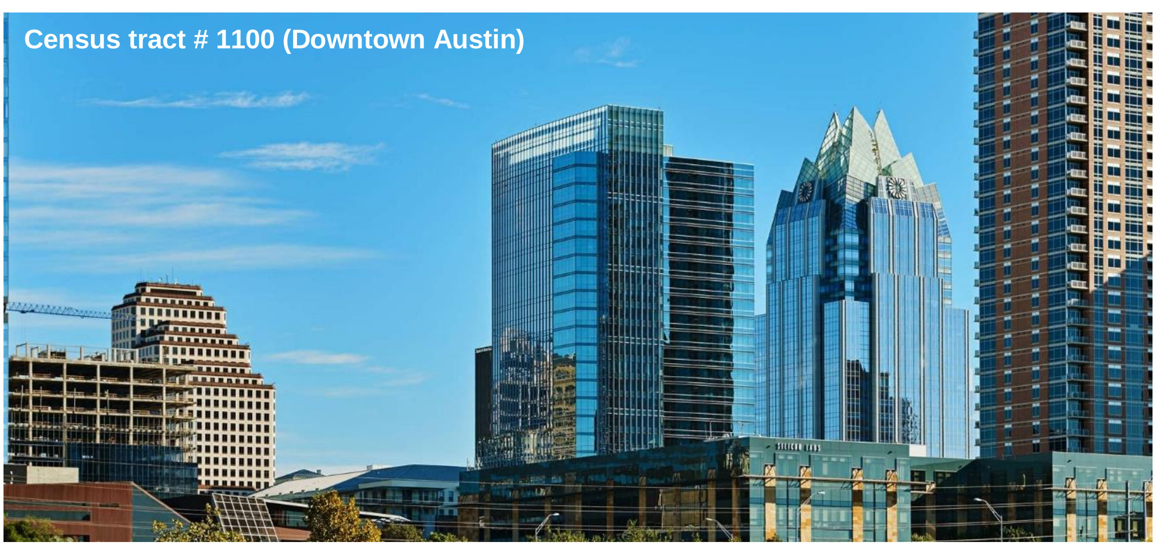
As shown in the table below, ***Census Tract #s 1100, 601, 604 and 603*** returned the highest number of venues

	Census Tract	Count
0	1100	100
1	601	100
2	604	100
3	603	100
4	500	92
5	1849	78
6	401	71
7	302	63
8	1722	52
9	1910	52

Census tract # 1100 represents downtown Austin and returned one of the largest number of venues

```
# Let's check the Latitude and Longitude of the Census Tract # 1100
df_travis.loc[df_travis['Census_Tract']==1100]
```

	Census_Tract	Total_population	White	African_Americans	Hispanic	Indian	Latitude	Longitude
6	1100	5512.0	3871.0	445.0	806.0	71.0	30.266295	-97.742436



How many unique venue categories in Travis County?

```
print('There are {} uniques categories.'.format(len(travis_venues['Venue Category'].unique())))
```

```
There are 243 uniques categories.
```

Indian Restaurant is one among 243 unique categories.

One Hot Coding to create dummy variables:

```
# one hot encoding to create dummy columns
travis_onehot = pd.get_dummies(travis_venues[['Venue Category']],prefix="",prefix_sep="")
# add Census_Tract column back to dataframe
travis_onehot['Census_Tract'] = travis_venues['Census_Tract']
# move neighborhood column to the first column
fixed_columns = [travis_onehot.columns[-1]] + list(travis_onehot.columns[:-1])
# create the DataFrame
travis_onehot = travis_onehot[fixed_columns]
# Display travis_onehot
print("Shape of one hot encoded dataframe of venues in Travis Census Tract:",travis_onehot.shape)
travis_onehot.head()
```

```
Shape of one hot encoded dataframe of venues in Travis Census Tract: (1315, 244)
```

Now we group by Census Tracts the total occurrence of each category:

```
travis_grouped = travis_onehot.groupby('Census_Tract').sum().reset_index()
print("Shape of Travis Census dataframe grouped by Census_Tract:",travis_grouped.shape)
travis_grouped.head()
```

```
Shape of Travis Census dataframe grouped by Census_Tract: (40, 244)
```

Let's check how many Indian Restaurants are in each Census tract:

```
# Create a new DataFrame for Indian Restaurants arranged in descending order
travis_new = pd.DataFrame(travis_grouped[['Census_Tract', 'Indian Restaurant']].sort_values(by=['Indian Restaurant'],
# Reset index of the new dataframe
travis_new.reset_index(drop=True,inplace=True)
travis_new.head(10)
```

	Census_Tract	Indian Restaurant
0	604	2
1	1850	2
2	1826	1
3	601	1
4	603	1
5	1819	1
6	1785	1
7	1910	1
8	1752	1
9	1756	1

The dataframe ***travis_new*** contains the number of Indian Restaurants for each Census Tract arranged in descending order. Here are some observations:

a) *There are **12 Indian restaurants** in Travis County*

b) *These restaurants are spread over **9 Census tracts***

c) *Census Tract **# 1850** and # 604 returned two restaurants each*

Let's check the Asian Indian population in these Census Tracts:

This will help us answer the following questions:

a) ***Is there is a correlation between Asian Indian population and the number of Indian Restaurants?***

b) ***Is there a census tract where a new Indian Restaurant can cater to an underserved community of Asian Indians ?***

Let's create a dataframe of Asian Indian population that has atleast one Indian Restaurant in the Census Tract:

```
# Let's create a new dataframe from the census_tracts
df_travis_copy = df_travis.copy()
# Let's create a list of census tracts representing Travis County
tract_list = travis_indian['Census_Tract'].tolist()
# Select rows in the census dataframe using tract_list
df_travis_indian = df_travis_copy[df_travis_copy.Census_Tract.isin(tract_list)]
# Reset index
df_travis_indian.reset_index(drop=True,inplace=True)
df_travis_indian.head(10)
```

	Census_Tract	Total_population	White	African_Americans	Hispanic	Indian	Latitude	Longitude
0	601	9207.0	4491.0	494.0	1972.0	15.0	30.287412	-97.738044
1	603	7793.0	5121.0	149.0	972.0	149.0	30.291992	-97.746270
2	604	6496.0	3842.0	138.0	852.0	181.0	30.285534	-97.747727
3	1752	3583.0	1934.0	91.0	622.0	99.0	30.357620	-97.749829
4	1756	3997.0	2959.0	104.0	402.0	46.0	30.426141	-97.769773
5	1785	4025.0	2249.0	281.0	828.0	92.0	30.444346	-97.736238
6	1819	4265.0	696.0	310.0	2820.0	43.0	30.363431	-97.702651
7	1826	2276.0	1033.0	294.0	543.0	104.0	30.402243	-97.683026
8	1850	3890.0	1594.0	522.0	1159.0	81.0	30.411905	-97.701786
9	1910	4210.0	3424.0	37.0	398.0	33.0	30.269903	-97.798175

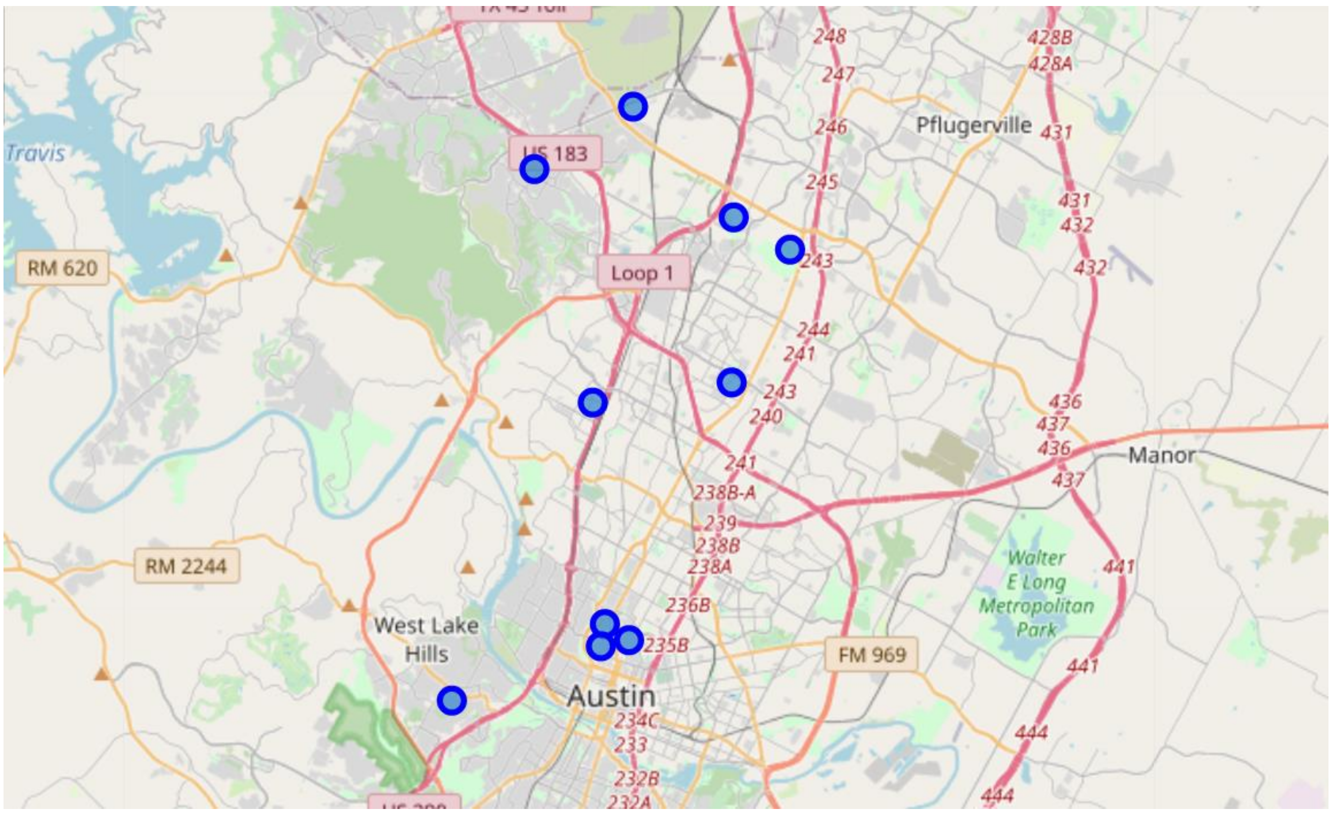
Merging this dataframe with Indian Restaurant dataframe:

```
# Merge two dataframes to create our final dataframe for Asian Indian population & restaurant for Travis County
df_combined = pd.merge(df_travis_indian,travis_indian,left_on='Census_Tract',right_on='Census_Tract',how='left')
# Rearrange the columns
df_asian_indian = df_combined[["Census_Tract","Total_population","White","African_Americans","Hispanic","Indian"
# Rename Indian Restaurant column name
df_asian_indian = df_asian_indian.rename(columns={'Indian Restaurant':'Indian_Restaurant'})
df_asian_indian.head(10)
```

Now we have our final dataframe (***df_asian_indian***) which contains the ***Census_Tract***, ***Total population, White population, African American population, Hispanic population, Indian population***, ***Indian restaurants*** and ***GPS co-ordinates***. Here are the 1st ten rows of the dataframe:

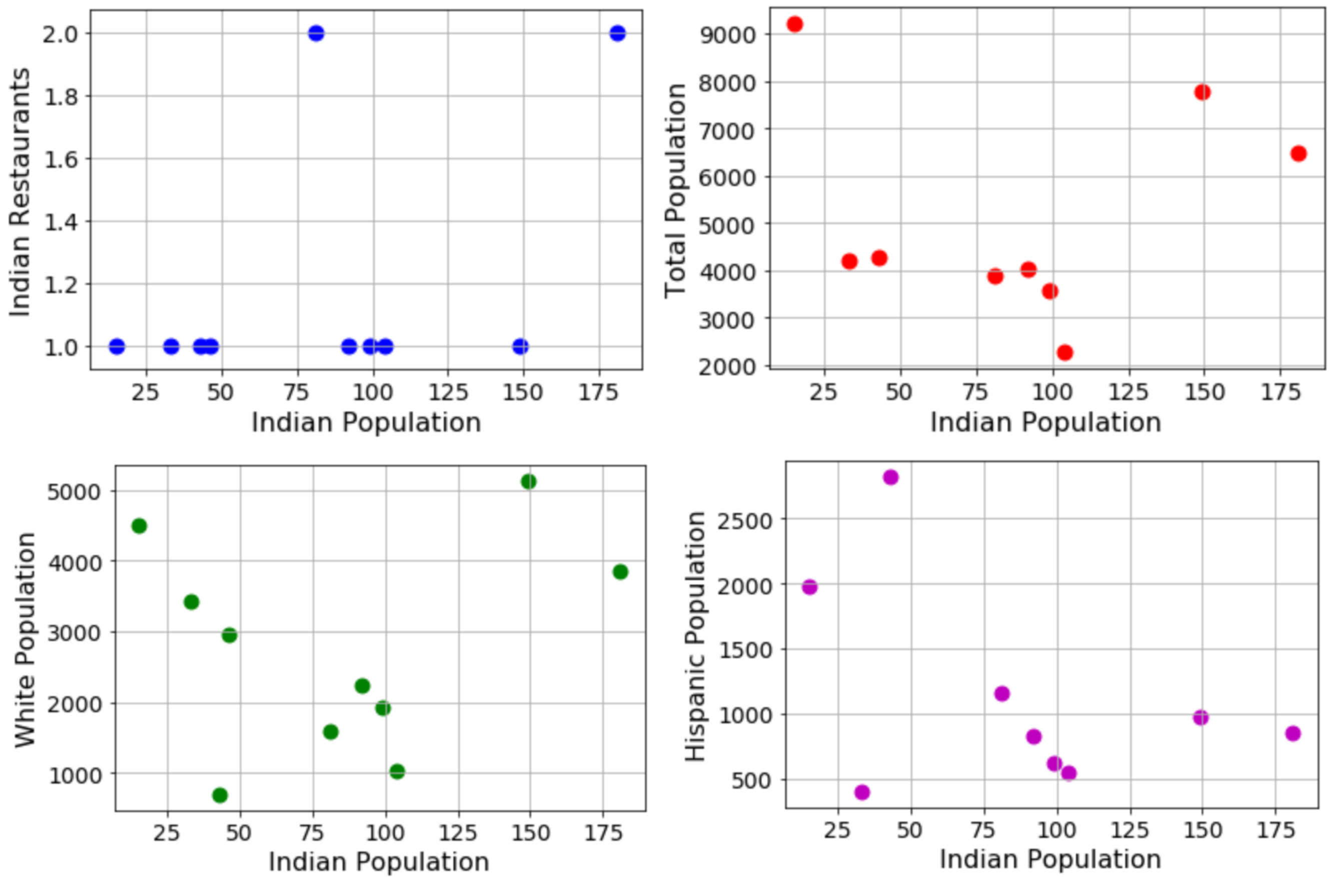
	Census_Tract	Total_population	White	African_Americans	Hispanic	Indian	Indian_Restaurant	Latitude	Longitude
0	601	9207.0	4491.0	494.0	1972.0	15.0	1	30.287412	-97.738044
1	603	7793.0	5121.0	149.0	972.0	149.0	1	30.291992	-97.746270
2	604	6496.0	3842.0	138.0	852.0	181.0	2	30.285534	-97.747727
3	1752	3583.0	1934.0	91.0	622.0	99.0	1	30.357620	-97.749829
4	1756	3997.0	2959.0	104.0	402.0	46.0	1	30.426141	-97.769773
5	1785	4025.0	2249.0	281.0	828.0	92.0	1	30.444346	-97.736238
6	1819	4265.0	696.0	310.0	2820.0	43.0	1	30.363431	-97.702651
7	1826	2276.0	1033.0	294.0	543.0	104.0	1	30.402243	-97.683026
8	1850	3890.0	1594.0	522.0	1159.0	81.0	2	30.411905	-97.701786
9	1910	4210.0	3424.0	37.0	398.0	33.0	1	30.269903	-97.798175

Centroids of Census tracts with atleast one Indian Restaurant:



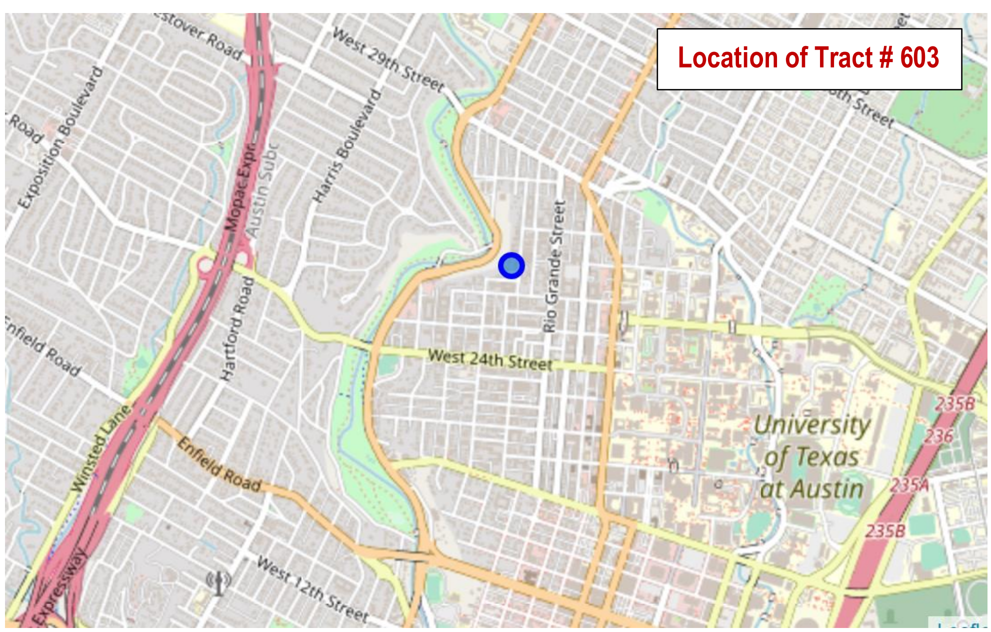
Studying the above dataframe (***df_asian_indian***), we see that Census_Tract # 604 has the largest Asian Indian population and has access to two Indian Restaurants. But the Census_Tract # 603 with second largest Indian population has only one Indian Restaurant (one less restaurant than Census_Tract # 1850, which has 46% less population). ***Census_Tract # 603*** is an underserved area – where an Indian mobile restaurant would find a large number of customers.

Looking for correlation between Indian Population & the other columns in the dataframe:



From the scatter plots shown above, we see that there ***doesn't exist any correlation*** between the Asian Indian population and the other columns in our final dataframe.

We have identified ***Census_Tract # 603*** as an underserved market for restaurant business. The location of this tract is shown in the adjacent Leaflet map. Interestingly, this tract falls in ***Cluster #1*** ***(magenta marker)*** of our neighborhood clustering. This cluster is far away from ***"food truck"*** cluster and ***"mexican restaurant"***



cluster. Thus, we are confident that this location would provide a unique business opportunity for Mr. Lohanathan to start his mobile restaurant business.

Conclusion:

We started this project with a mission to find a profitable location to start a mobile food business catering to the Asian Indian population in Austin, Texas. Using k-means clustering method we created three unique clusters of restaurants and food trucks in our Austin neighborhood. Analyzing the Census data from City of Austin data portal and U.S. Census Bureau we pin-pointed a tract where a mobile restaurant can serve an underserved market. Thus, we are confident that ***Census Tract # 603 within Cluster # 1*** is the optimum location for starting a mobile food business catering to the underserved Asian Indian population.

PostScript: *At the end of the project, I decided to call Mr. Lohanathan to convey my findings.*

Me: *I know the right place where you should move your mobile business. In normal times, I would have suggested Cluster # 0 where all the food trucks are located. But during lockdown, you don't expect your customers to visit those venues. Cluster # 2 is not suitable either because it is a cluster of Mexican Restaurants. So,* ***Cluster #1 is the best option****. But I have found a* ***particular tract (tract # 603)*** *in this cluster where there is a large number of Asian Indian population in an underserved market.*

LN: *Thank you so much for your data driven analysis. Is there anything more you would like to do in this project?*

Me: *Now that I have the analysis done for Travis county****, I would very much like to see the neighboring counties like Williamson and Round Rock*** *where companies like Apple, Amazon, Facebook are setting up offices. This would result in large increase in Asian Indians in those communities.*

*Also****, I would like to get feedback from you****, so that I can refine my model and maybe find another opportune cluster.*

LN: *Of course, my friend. By targeting locations based on data, we shall bring a paradigm shift in the mobile restaurant business in Austin.*