# Applied Data Science Capstone

IBM: (Use this one because you can see and interact with the maps)
https://dataplatform.cloud.ibm.com/analytics/notebooks/v2/192d2983-e7c0-43df-8895-b1d6672e4df6/view?access_token=756c9afb215e04c3439f6482018d4e9f94892f619be0e42ad10b82eb2ff2949d

GitHub:
https://github.com/coursera-coder/Coursera_Capstone/blob/master/Applied%20Data%20Science%20Capstone.ipynb

## Introduction

**TL;DR;** *Going to find out what restaurant to open that best serves police stations, and which station(s) will benefit the most.*

**Problem Description**

The purpose of this project is to find the best place for an associated business. For example, around a hospital you will most likely find pharmacies, or a Pawn shop and a Casino. Now we may think that police officers love donuts, so donut shops may be clustered around police stations. Is this true or just a stereotype? This project we will identify the most common restaurants around police stations in a particular city, then we will determine which type of restaurant around a particular police station that someone should open.

**Why do I want to do this?**

Intellectual curiosity on my part, but it does have some important impacts for others. First, it gives someone an opportunity to choose the best business to support an organization that they love. For me that would be to support local law enforcement.

**Why is this important, and who would be interested?**

Someone who wants to support an organization, but still open a successful business would be interested in the results of this project. For example, you might be a retired police officer in a new city. You may have your own opinions on what is best, but this city is different. The data provided by this project would give you a greater chance of success while still supporting the law enforcement community that you love.

As a side benefit of interest to anyone, this can support or discredit stereotypes.

# Data

**TL;DR;** *Get locations of police stations, Find closest businesses, K-Means cluster, find most common, apply suggestion to least common.*

I am anticipating that most if not all data will be coming from Foursquare. This section is broken down into two parts, Gathering Data, Working with Data.

**Gathering Data**

We need to start somewhere, so seeing that our target audience is someone retiring from the industry, and Florida is a great place to retire, we will be using Orlando, Florida as our test market. However, we can use the data from any city and apply the same analysis. If this was a web-based tool, we would let the user select the location, and category that they would like to support, however this is a class assignment so I am choosing Police and Orlando Florida.

**Get the locations of Police Stations and Filter as needed**

Lucky for us, Foursquare has categorized venues, and Police Stations is one of the categories! (https://developer.foursquare.com/docs/build-with-foursquare/categories/)
**Police Station**
**4bf58dd8d48988d12e941735**

To get the police stations in Orlando Florida, we will use the venue search API call with the appropriate category ID (https://developer.foursquare.com/docs/api-reference/venues/search/)

The call is a nested structure and we do not know if they want a category or end node, therefore we will have to flatten the table.

```
In [8]: print('We are searching for the {} category'.format(which_category_do_you_want_to_support))

        df_categories.loc[df_categories['name'] == which_category_do_you_want_to_support].dropna(how='all')
```

```
We are searching for the Police Station category
```

Out[8]:

| | id | name | pluralName | shortName |
|---|---|---|---|---|
| 839 | 4bf58dd8d48988d12e941735 | Police Station | Police Stations | Police Station |

**Get nearby Venue Data**

https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}

We will then gather and filter nearby venues. This includes location data so we will be using folium to map the data.

**Analyze the areas surrounding the user defined category**

Get the most common businesses for each area surrounding the user defined category, in our case police stations.

**Cluster the data**

use k-means to cluster by popularity. This lets us know what the categories are for popularity.

**Find the most underserved area to recommend to our end user**

Which police station ranks last? that is most likely the best location for opening up the top item(s) from the ranking.

We can also answer the stereotype problem. Let us see if donut shops rank #1.

# Methodology

**TL;DR;** K Means used for clustering, Elbow curve and Silhouette Method used for automatic identification of clustering.

## Setup:

We should be able to run this for any city. For the sake of demonstration, we are using Orlando, FL and Police stations as our inputs. The notebook was written to take in any city or venue.

**What we are seraching for. This can be convereted to user input for future versions**

```
In [3]: where_do_you_want_to_be = 'Orlando, Florida'
        which_category_do_you_want_to_support = "Police Station"

        print('Our end user wants to be in {}, and support {}.'.format(where_do_you_want_to_be, which_category_do_you_want_to_support))

        Our end user wants to be in Orlando, Florida, and support Police Station.
```

## Get 1

We need to get all the police stations. In order to do this, we need to find out what category police stations are. Getting the categories is easy. Just use this call:

https://developer.foursquare.com/docs/api-reference/venues/categories/

The issue is that the return is a nested structure. We need to flatten it. Check the notebook for the formula we used.

We get a list of categories:

| | id | name | pluralName | shortName |
|---|---|---|---|---|
| 0 | 52f2ab2ebcbc57f1066b8b4a | Tunnel | Tunnels | Tunnel |
| 1 | 4f04b25d2fb6e1c99f3db0c0 | Travel Lounge | Travel Lounges | Lounge |
| 2 | 54541b70498ea6ccd0204bff | Transportation Service | Transportation Services | Transportation Services |
| 3 | 52f2ab2ebcbc57f1066b8b51 | Tram Station | Tram Stations | Tram |
| 4 | 4bf58dd8d48988d129951735 | Train Station | Train Stations | Train Station |

From there it is easy to get our category:

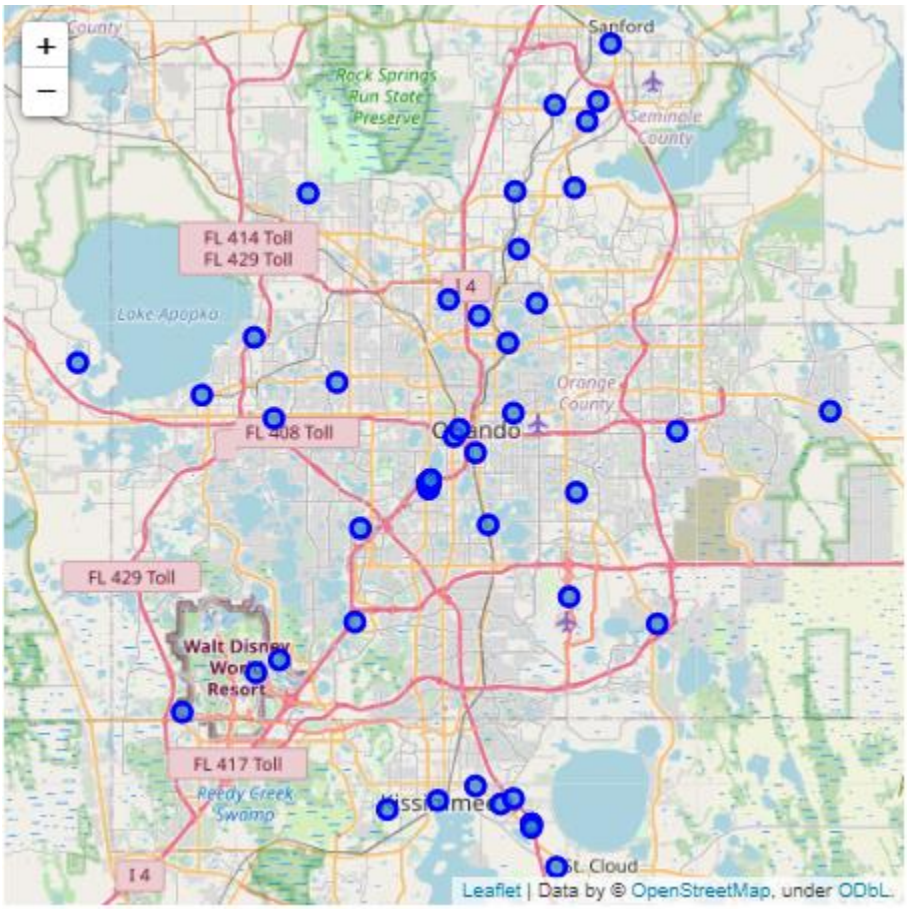
| | id | name | pluralName | shortName |
|---|---|---|---|---|
| 839 | 4bf58dd8d48988d12e941735 | Police Station | Police Stations | Police Station |

Getting the police stations is easy:

https://developer.foursquare.com/docs/api-reference/venues/search/

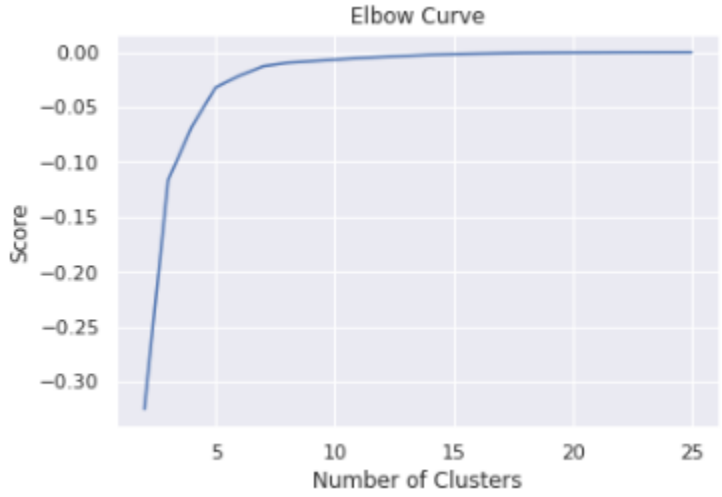
We place the results on a map. The map is auto sized based on the returned results. See notebook for code. This eliminates the need for human input allowing returns for multiple locations without massaging.

Here is a map of the police stations that were found:

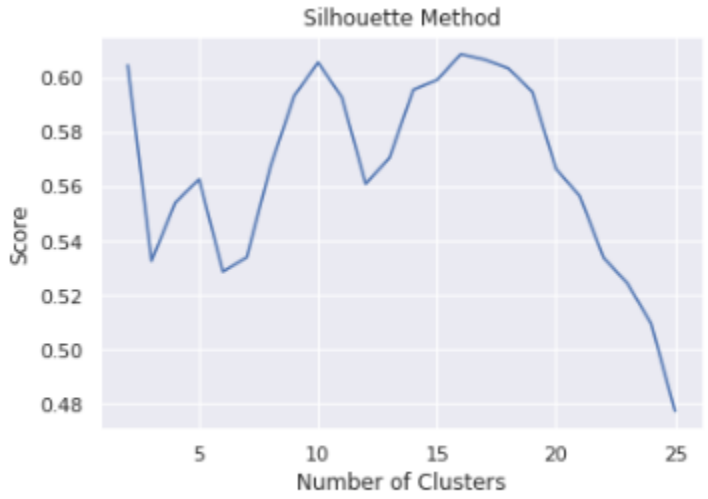



## Cluster K-Means

We use K-Means to cluster the data. But we need to find the ideal number of clusters we use. Using the elbow method, we can visually see a good answer.

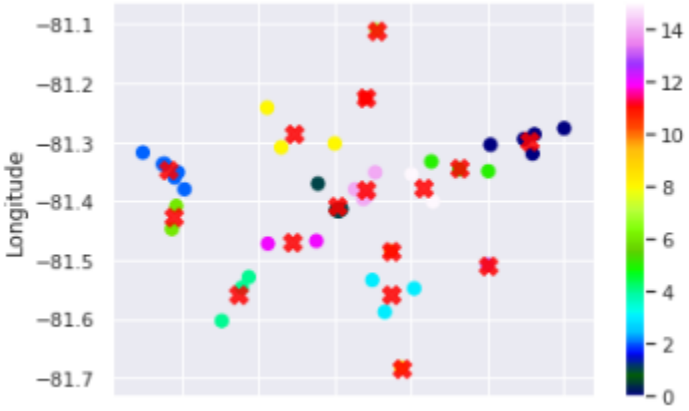


Elbow curves are good for visual, but we found that auto-selection of clusters was hit or miss. Using the silhouette method was much better and gave better results in our testing.



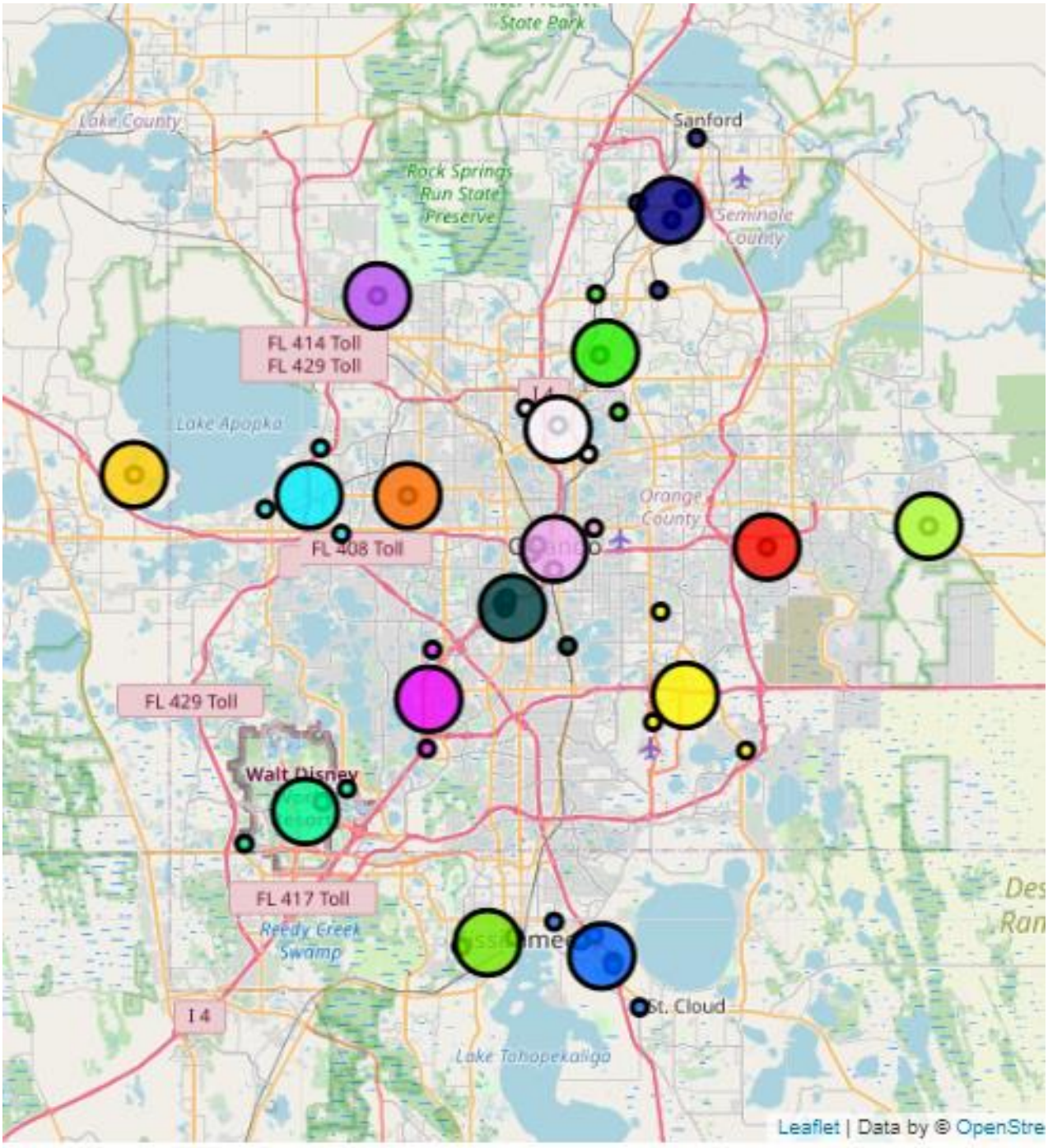

```
Our max is 16, We will use this number for KMeans for this run.
```

For this run, 16 clusters was the ideal number of clusters.




This chart shows the clustering. The additional benefit is that we can identify the center of each cluster. This will help us identify where to place a business in the future. For now we just store away all of the cluster centers because we don't know which cluster we should recommend.

Here are the clusters on the map. The large circles are the cluster centers, the smaller circles are the police stations. Each color is for a unique cluster assignment.



## Get 2

Now that we have our cluster locations, let us get some additional data for each cluster area.

Here is a sample of that collected data. As you can see, we have assigned it to the appropriate cluster and we know what type of venue it is.

| | Cluster | Cluster Latitude | Cluster Longitude | Venue | Venue Latitude | Venue Longitude | Venue Category |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 28.754578 | -81.296705 | 7-Eleven | 28.755337 | -81.297201 | Convenience Store |
| 1 | 0 | 28.754578 | -81.296705 | Bayhead Eye Centre | 28.756624 | -81.298470 | Office |
| 2 | 0 | 28.754578 | -81.296705 | Country Club Flower Shop | 28.756665 | -81.298470 | Flower Shop |
| 3 | 1 | 28.502287 | -81.409602 | Boating On Lake Holden | 28.501700 | -81.409418 | Lake |
| 4 | 1 | 28.502287 | -81.409602 | Aunt Lorry's House | 28.505838 | -81.412468 | Karaoke Bar |

## Rank

For each area we rank the 10 most common venues

| | Cluster | 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 | 0 | Flower Shop | Convenience Store | Office | Video Store | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop |
| 1 | 1 | Karaoke Bar | Lake | Video Store | Food & Drink Shop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop |
| 2 | 2 | Convenience Store | Video Store | Grocery Store | Chinese Restaurant | Sandwich Place | Donut Shop | Fast Food Restaurant | Pharmacy | Intersection | Ice Cream Shop |
| 3 | 3 | Carpet Store | Video Store | Food & Drink Shop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | Event Space |
| 4 | 4 | Hotel Bar | Lounge | Hotel Pool | American Restaurant | Bus Stop | Resort | Buffet | Ice Cream Shop | Jewelry Store | Laundromat |

## Find

For the entire city for areas around our clusters, the ranking of most popular venues are:

| | Venue | Commonality |
|---|---|---|
| 0 | Bar | 0.080321 |
| 1 | American Restaurant | 0.056225 |
| 2 | Lounge | 0.048193 |
| 3 | Hotel | 0.044177 |
| 4 | Convenience Store | 0.040161 |
| 5 | Steakhouse | 0.028112 |
| 6 | Pizza Place | 0.028112 |
| 7 | Cocktail Bar | 0.024096 |
| 8 | Mexican Restaurant | 0.024096 |
| 9 | Burger Joint | 0.024096 |

At the top of the list is a bar. So that is most likely what we should recommend that the user opens.

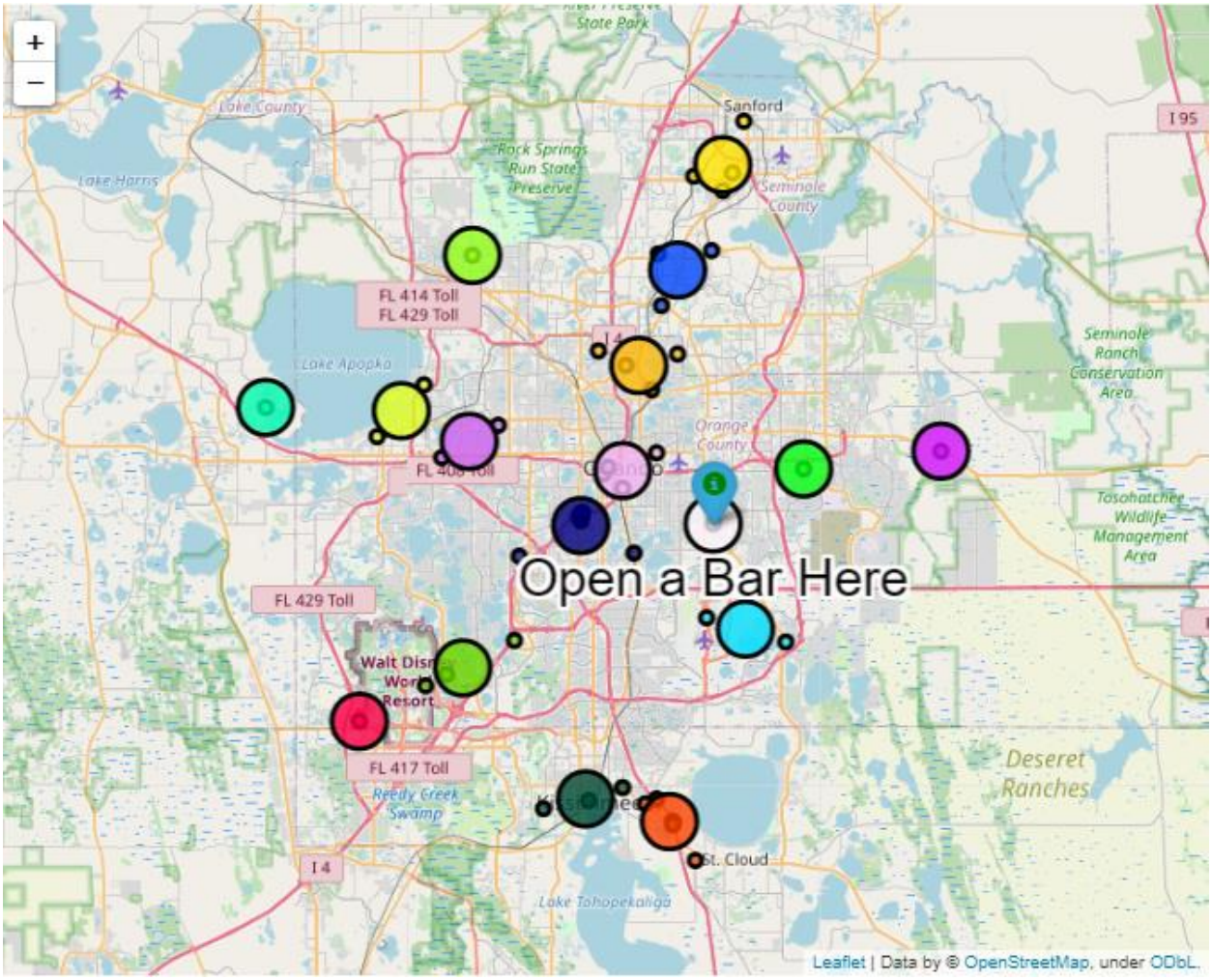
To find the most underserved area, we need to rank each category according to its commonality score with the top 10 venues.

| | Cluster | 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 | Score |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | Karaoke Bar | Lake | Video Store | Food & Drink Shop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | 0.104418 |
| 1 | 6 | Airport Terminal | Gym | Baseball Field | Boat or Ferry | Video Store | French Restaurant | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | 0.104418 |
| 2 | 3 | Carpet Store | Video Store | Food & Drink Shop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | Event Space | 0.104418 |
| 3 | 11 | Liquor Store | Auto Workshop | Financial or Legal Service | French Restaurant | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | 0.112450 |
| 4 | 8 | Airport Service | Burger Joint | Food & Drink Shop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | Event Space | 0.144578 |
| 5 | 0 | Flower Shop | Convenience Store | Office | Video Store | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | 0.176707 |
| 6 | 15 | Cosmetics Shop | Southern / Soul Food Restaurant | Video Store | Flower Shop | Convenience Store | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | 0.176707 |
| 7 | 5 | Convenience Store | Mexican Restaurant | Intersection | Smoothie Shop | Garden Center | Construction & Landscaping | Mobile Phone Shop | Fast Food Restaurant | Bookstore | Furniture / Home Store | 0.232932 |
| 8 | 2 | Convenience Store | Video Store | Grocery Store | Chinese Restaurant | Sandwich Place | Donut Shop | Fast Food Restaurant | Pharmacy | Intersection | Ice Cream Shop | 0.246996 |
| 9 | 9 | Hotel | Video Store | Food & Drink Shop | Convenience Store | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | Donut Shop | 0.257028 |
| 10 | 7 | Convenience Store | Pizza Place | Food & Drink Shop | American Restaurant | Auto Workshop | Cosmetics Shop | Cuban Restaurant | Dessert Shop | Discount Store | Dive Bar | 0.329317 |
| 11 | 4 | Hotel Bar | Lounge | Hotel Pool | American Restaurant | Bus Stop | Resort | Buffet | Ice Cream Shop | Jewelry Store | Laundromat | 0.361446 |
| 12 | 13 | Pizza Place | American Restaurant | Discount Store | Steakhouse | Fast Food Restaurant | Food & Drink Shop | Convenience Store | Cosmetics Shop | Cuban Restaurant | Dessert Shop | 0.393574 |
| 13 | 14 | Bar | Lounge | Cocktail Bar | American Restaurant | Restaurant | Burger Joint | Art Gallery | Basketball Stadium | Cuban Restaurant | Pizza Place | 0.610442 |
| 14 | 12 | Hotel | American Restaurant | Steakhouse | Pizza Place | Bar | Mexican Restaurant | Restaurant | Theme Park Ride / Attraction | Pool | Gym | 0.618474 |

We sorted the above by rank (lowest to highest) and found that Cluster 1 was the most underserved.

# Results

**The analysis recommends that the end user should open a "Bar" in Area 1 centered at [ 28.500289 -81.302449]**




```
In [35]: printmd("# ",'User should open a "{}" in Area {} centerd at {}'.format(most_popular_venue, least_common_zone, centers[least_common_zone]), color="#006400")
```

**User should open a "Bar" in Area 16 centerd at [ 28.500289 -81.302449]**

# Discussion

There are a lot of improvements that can be made to this program, but it is a decent proof of concept. The biggest fault of the program is that there is more data collection that can be made and that Foursquare gives different results depending on the time of day. Data collection can easily be improved, but I am using the free addition and my calls (and number of results are limited). Time of day data would need to be added to the mix.

By the way, when looking at the data we did not find that Donut shops and Police Stations are usually close to each other. Guess we can throw that stereotype out.

# Conclusion

You can try it yourself!  All you need to do is copy the notebook and change the values in the beginning. (you will also need to add your Foursquare credentials.

I will leave you with the same data for Seattle Washington.