Desirable Neighborhoods in San Francisco

Vien Nguyen

March 6, 2019

1. Introduction

A couple is moving into San Francisco, and they would like to pick their neighborhood based on several factors. They want to be close to nearby parks, restaurants, and public transit. They would also like the adjacent neighborhoods to be desirable as well. The target audience for this analysis are couples that value living near parks, food, and transit in the city of San Francisco.

1.1 Background & Interest

As a current resident within San Francisco, this analysis has some personal interest for me. There are many parks and outdoor recreation opportunities in this city, and getting around with public transit is important due to the lack of parking in the busier neighborhoods. Also, San Francisco is well known for having many good restaurants in all price ranges. Thus, this analysis focuses on outdoors & recreation, food, and transit available within each neighborhood.

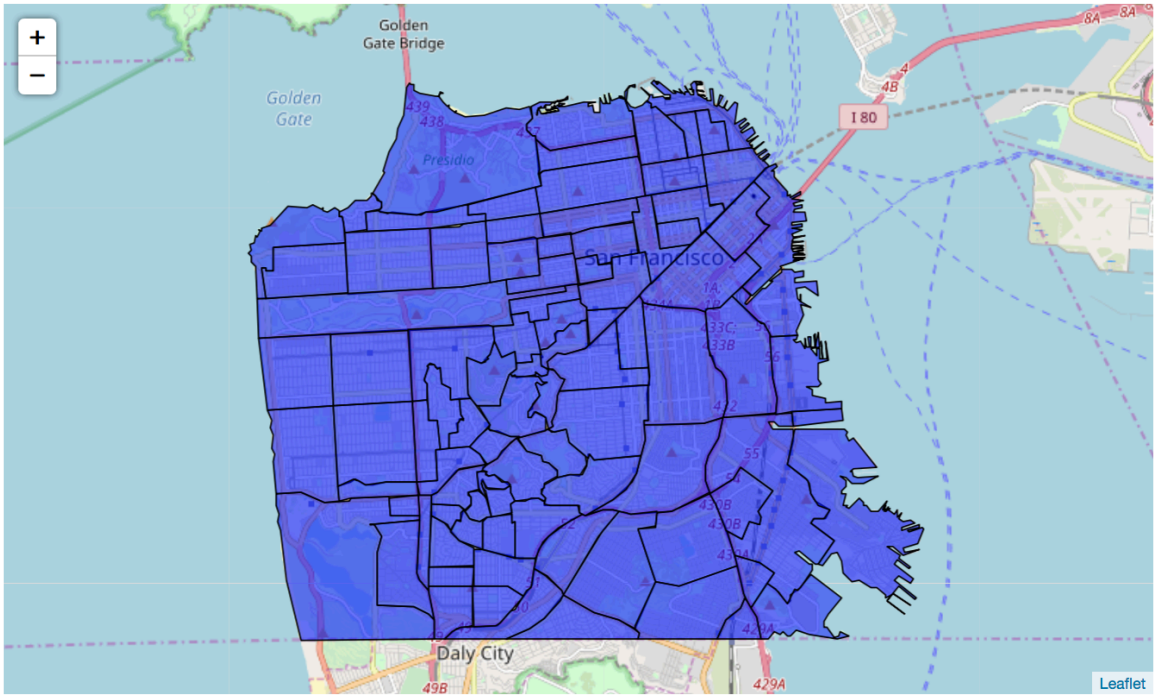
1.2 Problem

Can we build a model to recommend a neighborhood for the couple to live in, based on their specified criteria?

2. Data Acquisition and Cleaning

2.1 Data Sources

The data draws upon neighborhood data from a GeoJSON file available on DataSF, a publicly available source of municipal datasets. This data details San Francisco Neighborhoods as designated by the San Francisco Association of Realtors (SFAR) in August 2010. This data will define the boundaries of each of the 92 neighborhoods.

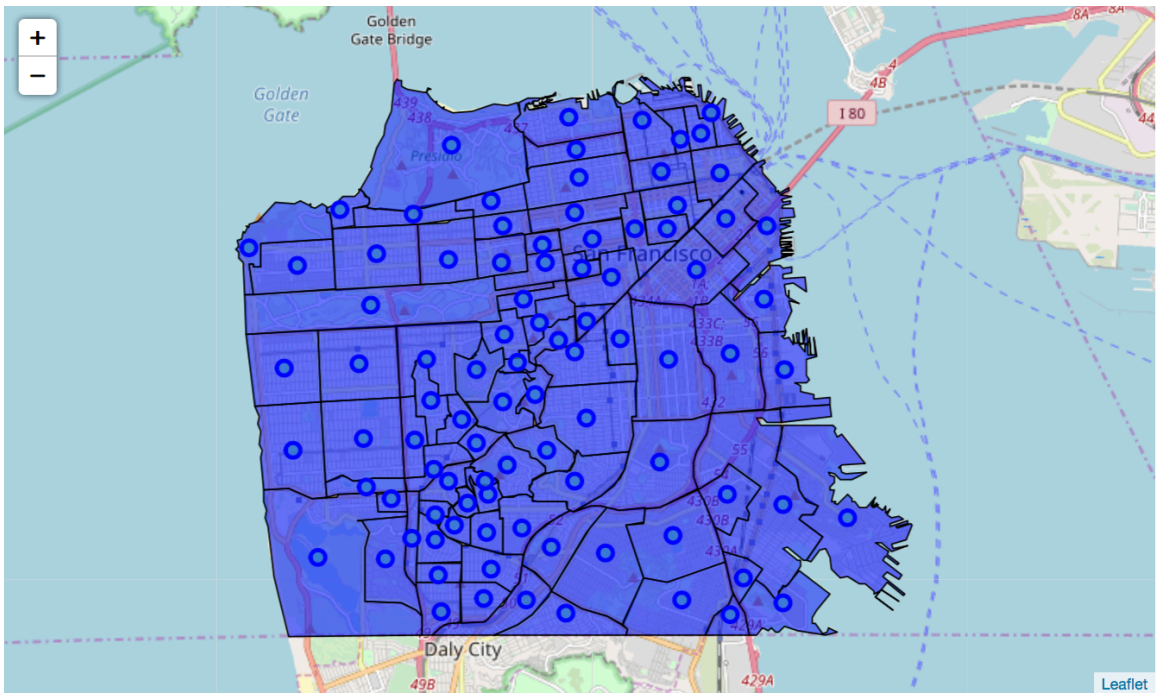


The data for this analysis also draws upon the venues given by the FourSquare API. The venue data includes the transit, park, and restaurant data that is important to the addressing the problem. Since there are many venue types, they will need to be consolidated into meaningful categories. For example, bus stops, train stations, and light rails all fall under transit.

2.2 Data Cleaning

The GeoJSON file is easily usable with the Folium library for visualization of the neighborhood borders. Its JSON structure is also simple to parse into a Pandas dataframe.I use the Shapely library to find the center of each neighborhood, as each neighborhood's coordinates can be used to create a Polygon object, which has a representative point function.

	Neighborhood	Latitude	Longitude
0	Alamo Square	37.776076	-122.433919
1	Anza Vista	37.780611	-122.443255
2	Balboa Terrace	37.730649	-122.468267
3	Bayview	37.732391	-122.387170
4	Bernal Heights	37.740230	-122.415885



The Foursquare API is called to get venues within each neighborhood. Calling the API results in a JSON file, which we can parse for venue data.


```
{'meta': {'code': 200, 'requestId': '5c81f1b51ed2196e48bf5806'},
 'response': {'venues': [{'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categ
ories_v2/travel/busstation_',
       'suffix': '.png'},
      'id': '52f2ab2ebcbc57f1066b8b4f',
      'name': 'Bus Stop',
      'pluralName': 'Bus Stops',
      'primary': True,
      'shortName': 'Bus Stop'}],
    'hasPerk': False,
    'id': '4f6d0ddae4b0725b60d290e3',
    'location': {'address': 'Van Ness Ave',
     'cc': 'US',
     'city': 'San Francisco',
     'country': 'United States',
     'crossStreet': 'at McAllister St',
     'distance': 103,
     'formattedAddress': ['Van Ness Ave (at McAllister St)',
      'San Francisco, CA 94102',
      'United States'],
     'labeledLatLngs': [{'label': 'display',
       'lat': 37.779925625368705,
       'lng': -122.4200782149065}],
     'lat': 37.779925625368705,
     'lng': -122.4200782149065,
     'postalCode': '94102',
     'state': 'CA'},
    'name': 'MUNI Bus Stop - Van Ness & McAllister',
    'referralId': 'v-1552019893'},
   {'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/travel/bussta
tion_',
       'suffix': '.png'},
      'id': '52f2ab2ebcbc57f1066b8b4f',
      'name': 'Bus Stop'
```


It has a result limit of 50, so in order to get around this limitation, I call the API for various specific category IDs. These category IDs include the following:

- **Arts & Entertainment** 4d4b7104d754a06370d81259
- **Food** 4d4b7105d754a06374d81259
- **Outdoors & Recreation** 4d4b7105d754a06377d81259
- **Shop & Service** 4d4b7105d754a06378d81259
- **Bus Station** 4bf58dd8d48988d1fe931735
- **Bus Stop** 52f2ab2ebcbc57f1066b8b4f
- **Cable Car** 52f2ab2ebcbc57f1066b8b50
- **Light Rail Station** 4bf58dd8d48988d1fc931735
- **Metro Station** 4bf58dd8d48988d1fd931735
- **Train Station** 4bf58dd8d48988d129951735
- **Tram Station** 52f2ab2ebcbc57f1066b8b51

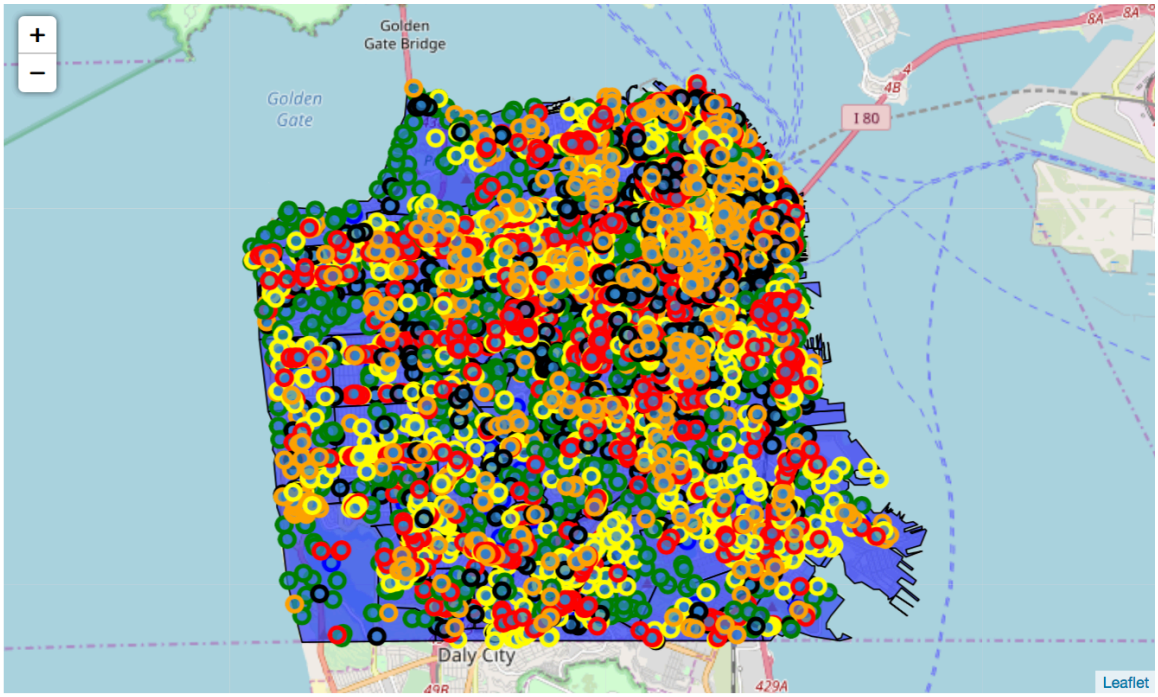
The first 4 categories will be treated as Master Categories in our later analysis. The Bus Station, Bus Stop, Cable Car, Light Rail Station, Metro Station, Train Station, and Tram Station categories will fall under the Transit Master Category. With 11 categories and 92 neighborhoods, we expect 1,012 API calls each time I call the getNearbyVenues function, which I adapted from the Week 3 assignment to target specific categories. In the initial stages of this analysis, I found that the 500 meter radius did not adequately cover enough of the larger neighborhoods. Subsequently, a 3000 meter radius tended to get results outside of the smaller neighborhoods, such that some of the neighborhoods would not have any results. Ultimately I found that combining the results for radii of 500 meters and 2000 meters made sure that I had results for every neighborhood, and there were no excessive gaps in geographic coverage. This gives me a dataframe with 36,379 rows with the neighborhood name, the neighborhood coordinates, the venue name, the venue coordinates, the Foursquare venue category, and the Master Category.

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	Master Category
0	Alamo Square	37.776076	-122.433919	Alamo Square	37.776062	-122.433622	Park	Outdoors & Recreation
1	Alamo Square	37.776076	-122.433919	Alamo Square Dog Park	37.775878	-122.435740	Dog Run	Outdoors & Recreation
2	Hayes Valley	37.776076	-122.433919	Yoga Garden	37.771982	-122.437107	Yoga Studio	Outdoors & Recreation
3	Hayes Valley	37.776076	-122.433919	Duboce Park	37.769458	-122.433013	Park	Outdoors & Recreation
4	Hayes Valley	37.776076	-122.433919	Yoga Tree Hayes	37.776507	-122.425014	Yoga Studio	Outdoors & Recreation

I use my find_n function to determine which neighborhood each venue falls within. After removing duplicates, my dataframe has 8,923 unique venues. The venues within each category are as follows:

- Outdoors & Recreation: 1523
- Food: 2063
- Transit: 1823
- Arts & Entertainment: 940
- Shop & Service: 2574

To check the geographic distribution of these venues, I use Folium to add a marker for each venue:



There are some minor gaps, but these areas are generally residential, parkland, or industrial areas that are not relevant for this analysis.

2.3 Feature Selection

For feature selection, I use the one-hot encoding method to create dummy variables for each category, and assign it zero or one to each venue based on the category. A one represents a match and a zero represents no-match for the category.

	Neighborhood	Arts & Entertainment	Food	Outdoors & Recreation	Shop & Service	Transit
0	Alamo Square	0	0	1	0	0
1	Alamo Square	0	0	1	0	0
2	Hayes Valley	0	0	1	0	0
3	Hayes Valley	0	0	1	0	0
4	Hayes Valley	0	0	1	0	0

Then we can group the neighborhoods and find the proportion of each category in each neighborhood between zero and one. In this dataframe, a zero means that a neighborhood has no instances of such a category, and a one means that all the venues within the neighborhood fall under that category.

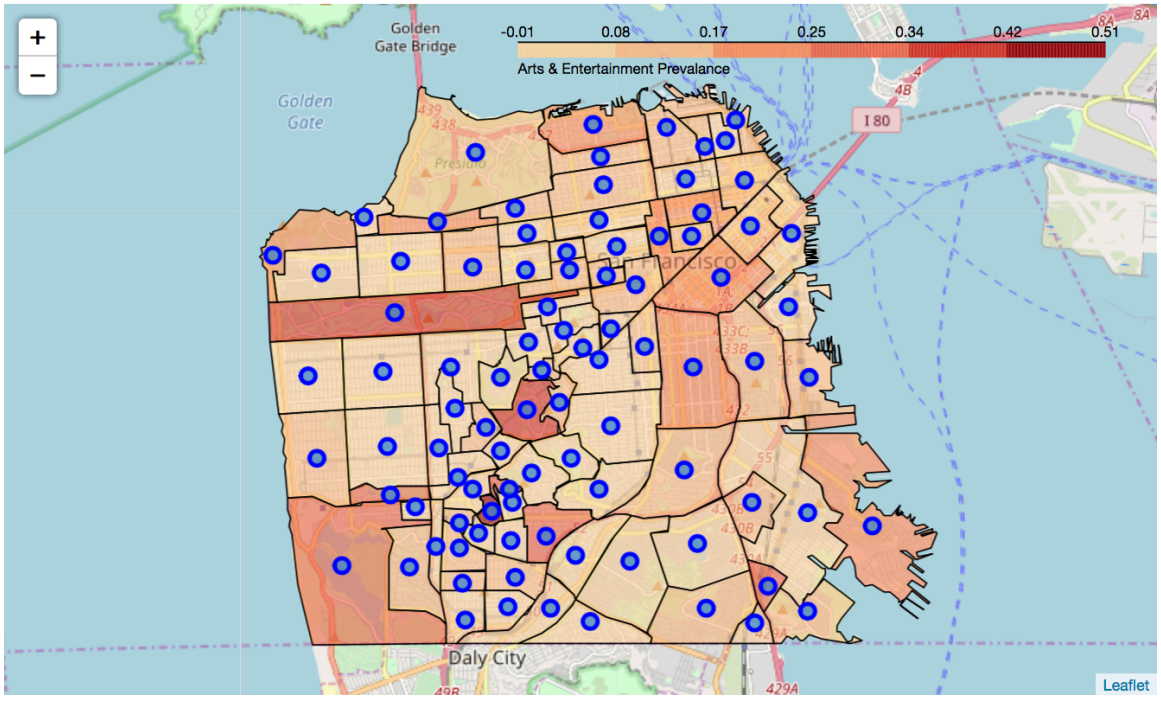
	Neighborhood	Arts & Entertainment	Food	Outdoors & Recreation	Shop & Service	Transit
0	Alamo Square	0.090000	0.260000	0.100000	0.360000	0.190000
1	Anza Vista	0.000000	0.107143	0.000000	0.714286	0.178571
2	Balboa Terrace	0.125000	0.125000	0.000000	0.625000	0.125000
3	Bayview	0.061111	0.244444	0.100000	0.472222	0.122222
4	Bayview Heights	0.272727	0.181818	0.272727	0.090909	0.181818

By itself, this data is certainly interesting in that we can see which neighborhoods have good proportions of each category, which is helpful in recommending which neighborhood to live in. However, we need to take into account the preferences of the target audience. We can later use these features to rank the neighborhoods based on the problem criteria.

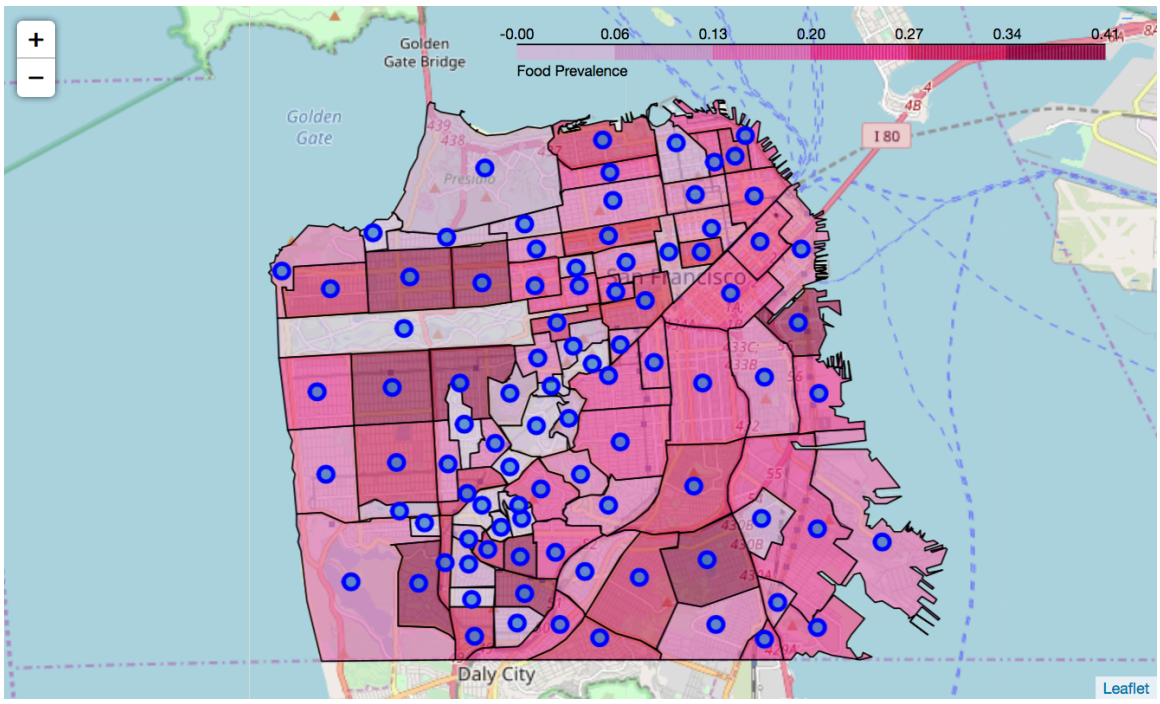
3. Exploratory Data Analysis (Methodology)

First, I generate choropleths to get an idea of the prevalence of the venues in each Category.

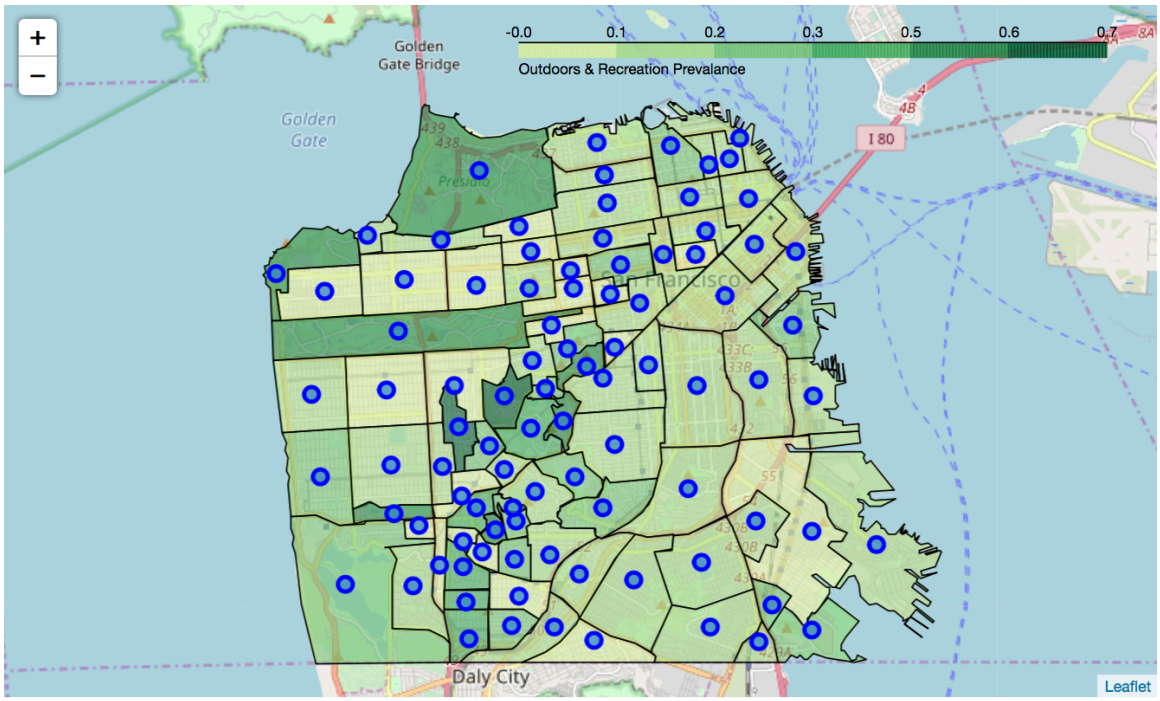
In the Arts & Entertainment category, the neighborhoods of Monterey Heights, Midtown Terrace, and Golden Gate Park have a high prevalence.



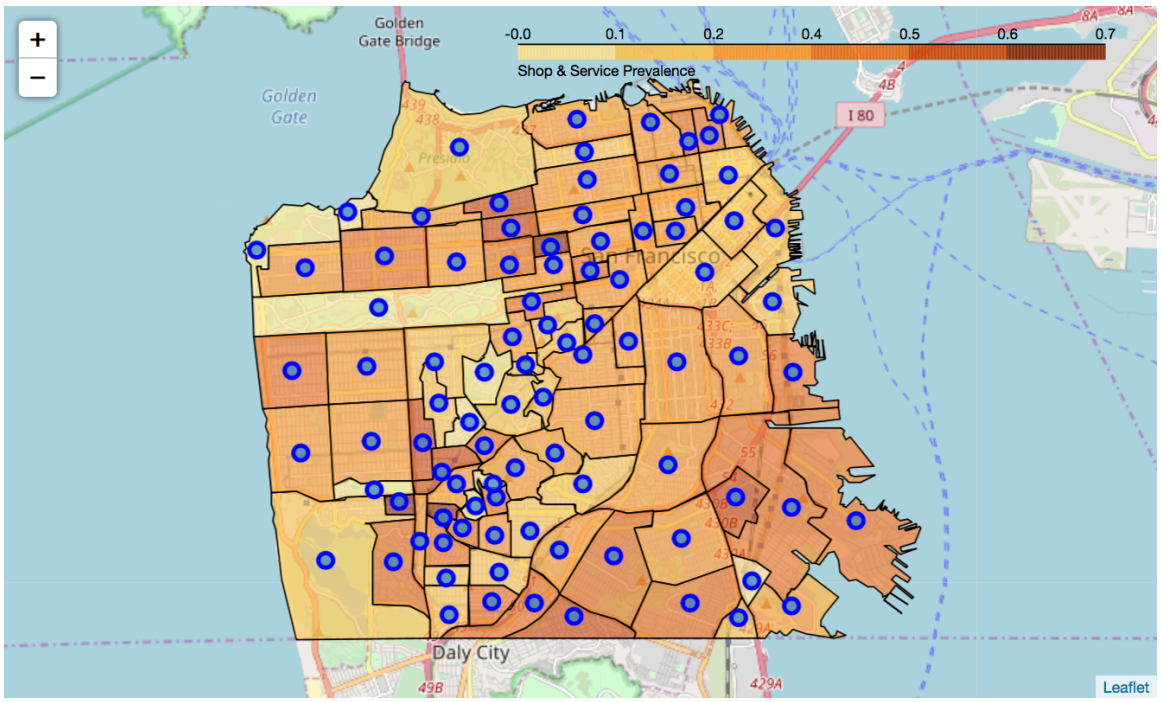
For the Food category, there are many neighborhoods with high prevalence, including Central and Inner Richmond, Central and Inner Sunset, Portola, Mission Bay, Stonestown, Westwood Park, and Ingleside.



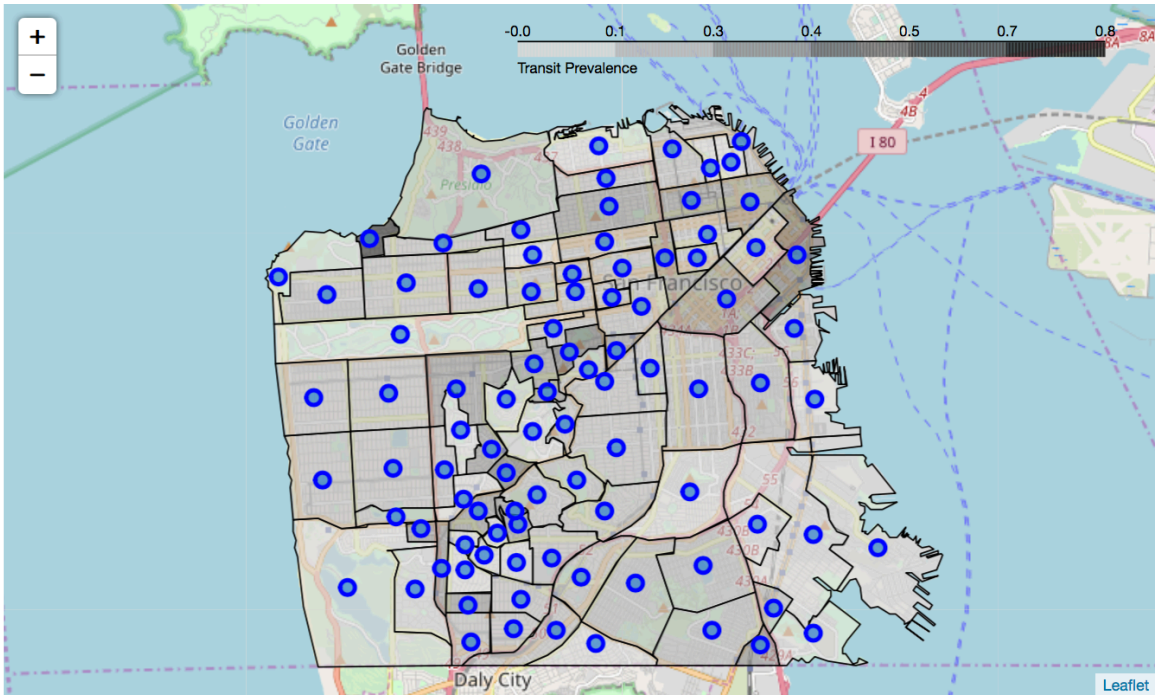
For the Outdoors & Recreation category, Forest Knolls and Golden Gate Heights have the highest prevalence. The Presidio, Golden Gate Park, Pine Lake Park, and Lincoln Park neighborhoods also have a high proportion of parks.



For the Shop & Service category, Anza Vista, Balboa Terrace, and Merced Manor have a high proportion of this category.



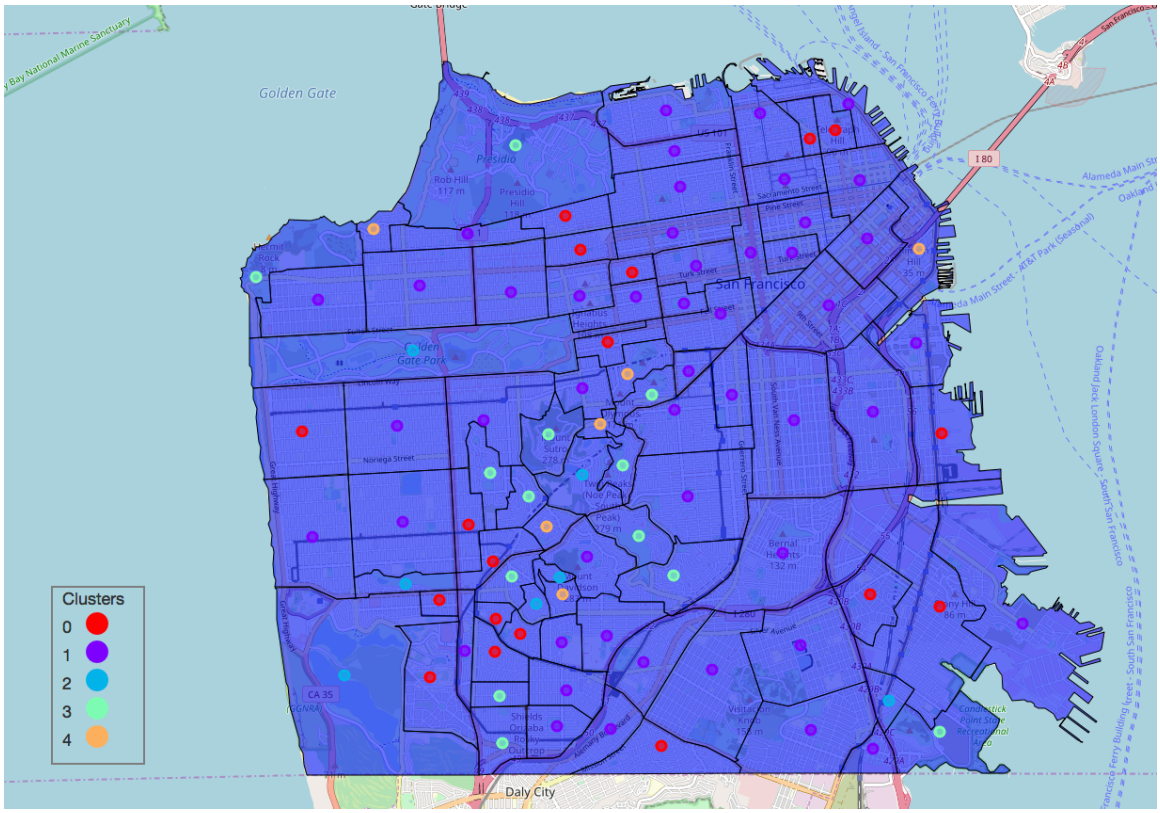
For the Transit category, Sea Cliff stands out as having a high proportion of this category.



For my exploratory data analysis, I will use the k-means clustering technique. This method works by initially randomly placing cluster centroids within the points of the feature set. It then calculates the distance between the centroid and all other points. After each iteration, it moves the centroid closer to the features and recalculates the distances. After enough iterations, the distances between the feature points and the centroids has been minimized enough to consider all the points within a cluster as related. By using k-means clustering, we group like neighborhoods together based on the values of the five categories.

	Arts & Entertainment	Food	Outdoors & Recreation	Shop & Service	Transit
0	0.090000	0.260000	0.100000	0.360000	0.190000
1	0.000000	0.137931	0.000000	0.689655	0.172414
2	0.125000	0.125000	0.000000	0.625000	0.125000
3	0.059783	0.244565	0.103261	0.483696	0.108696
4	0.272727	0.181818	0.272727	0.090909	0.181818

All of the neighborhoods within each cluster are similar in their proportion of categories. We can see some trends in this clustering. Clusters 1 and 2 have no Outdoors & Recreation, so those are not good neighborhoods to recommend for the target audience. This leaves Clusters 0, 3, and 4 as potential recommendations. The following map shows the clustering.



We can see that there are still many neighborhoods in the running, with no clear leader. To get a better recommendation, we will aggregate the category proportions into a single weighted Ranked Score.

3.1 Calculation of Primary Target Variable and its Relationship with the Feature Set

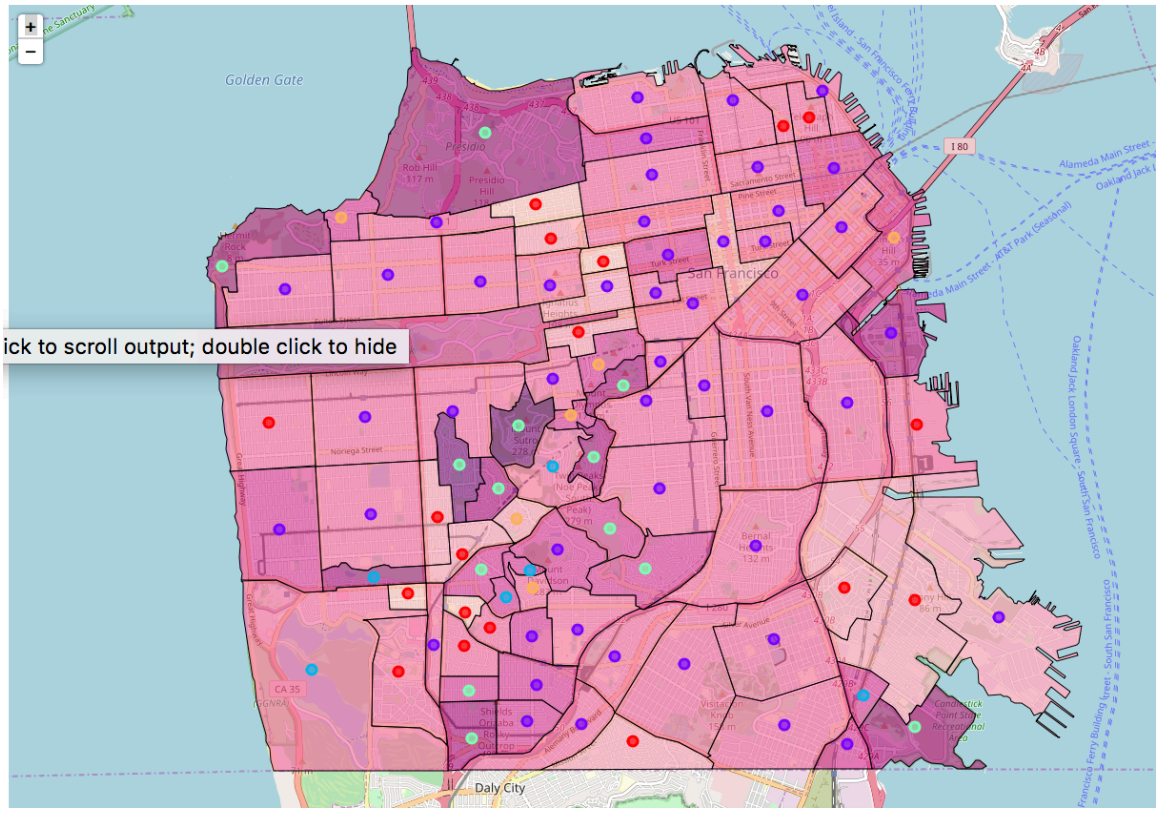
While this clustering is informative since it groups together similar neighborhoods, it does not give a clear indicator of which neighborhood excels in Outdoor & Recreation, Food, and Transit, as it treats all categories equally. We need to take into account the audience's preferences. To achieve this, I will weight each of the categories in order to match the preferences of the audience:

- 40% Outdoors & Recreation
- 30% Food
- 20% Transit
- 5% Arts & Entertainment
- 5% Shop & Service

This results in the primary target variable of Ranked Score. After calculating and sorting by Ranked Score, we get the following dataframe:

	Neighborhood	Arts & Entertainment	Food	Outdoors & Recreation	Shop & Service	Transit	Ranked Score
26	Golden Gate Heights	0.000000	0.117647	0.647059	0.117647	0.117647	0.323529
24	Forest Knolls	0.052632	0.105263	0.631579	0.105263	0.105263	0.313158
32	Ingleside Heights	0.051282	0.307692	0.384615	0.128205	0.128205	0.280769
42	Lincoln Park	0.183673	0.163265	0.489796	0.081633	0.081633	0.274490
84	Twin Peaks	0.086957	0.086957	0.478261	0.130435	0.217391	0.271739

Now, producing a choropleth of Ranked Score will let us visualize the neighborhoods with the highest scores.



As shown by the dataframe, we have Golden Gate Heights and Forest Knolls in much darker purple compared to the rest of the neighborhoods. These neighborhoods also share the same cluster, so it appears that I need to further concentrate on these areas for the final recommendation.

3.2 Calculation of Secondary Target Variable and its Relationship with the Feature Set

To further narrow down the recommendation, I will calculate average of surrounding Rank Scores for each neighborhood, which will serve as the secondary target variable. This is done by iterating through the data in the GeoJSON file. First I create a one-hot table with neighborhoods in rows and columns. Then for each neighborhood, I check if its shape touches the shape of all the other neighborhoods in the GeoJSON file. Where it touches, I replace the value of where the row and column intersect with the Ranked Score for that neighborhood. If it does not touch, or if the neighborhood is the same, then I replace the value with NaN.

The first table shows when it is initialized.

Neighborhood	Alamo Square	Anza Vista	Balboa Terrace	Bayview	Bayview Heights	Bernal Heights	Buena Vista Park/Ashbury Heights	Candlesti Poi
Alamo Square	1	0	0	0	0	0	0	
Anza Vista	0	1	0	0	0	0	0	
Balboa Terrace	0	0	1	0	0	0	0	
Bayview	0	0	0	1	0	0	0	
Bayview Heights	0	0	0	0	1	0	0	

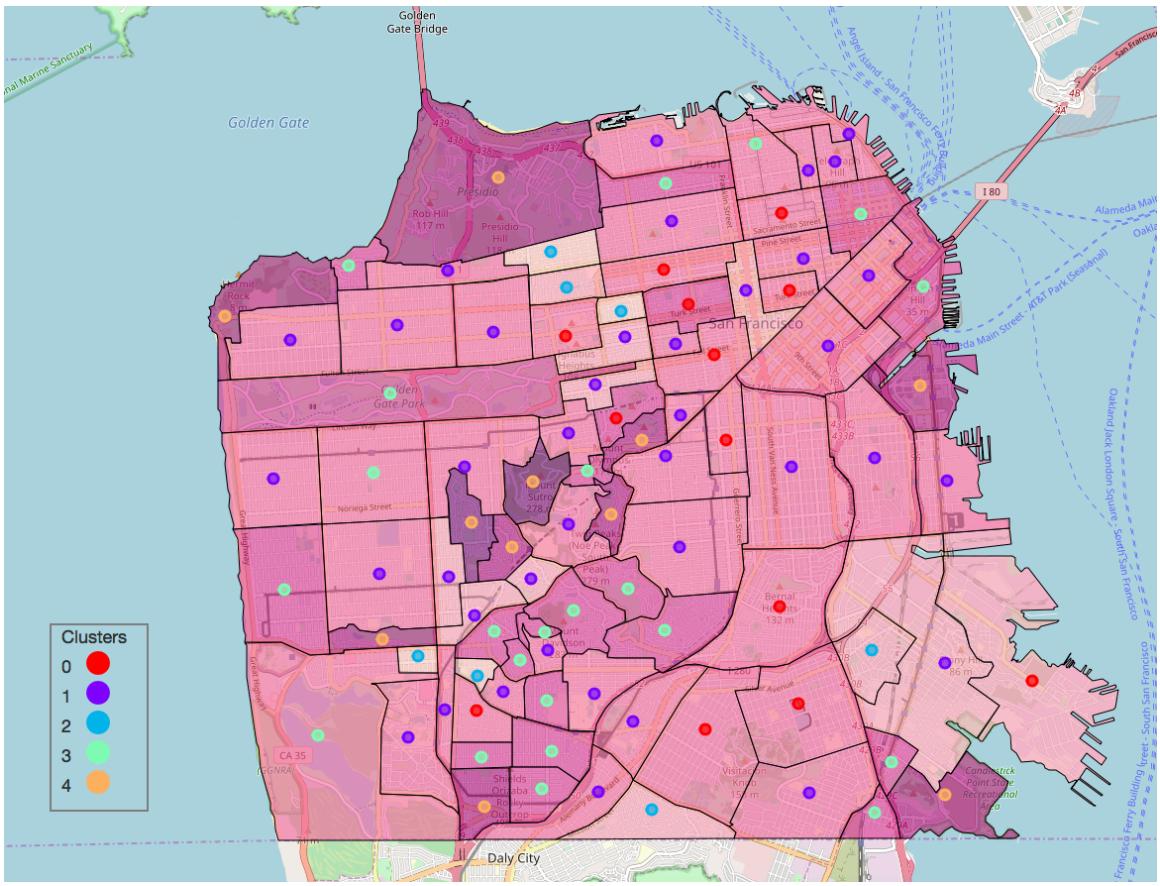
The second table shows after the GeoJSON has been processed.

Neighborhood	Alamo Square	Anza Vista	Balboa Terrace	Bayview	Bayview Heights	Bernal Heights	Buena Vista Park/Ashbury Heights	Candlestick Point
Alamo Square	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Anza Vista	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Balboa Terrace	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Bayview	NaN	NaN	NaN	NaN	0.218182	0.208681	NaN	0.256818
Bayview Heights	NaN	NaN	NaN	0.163587	NaN	NaN	NaN	0.256818

Now, I can get the average score for the adjacent neighborhoods. This average adjacent score serves as the secondary variable, as it is calculated from the features and the primary variable.

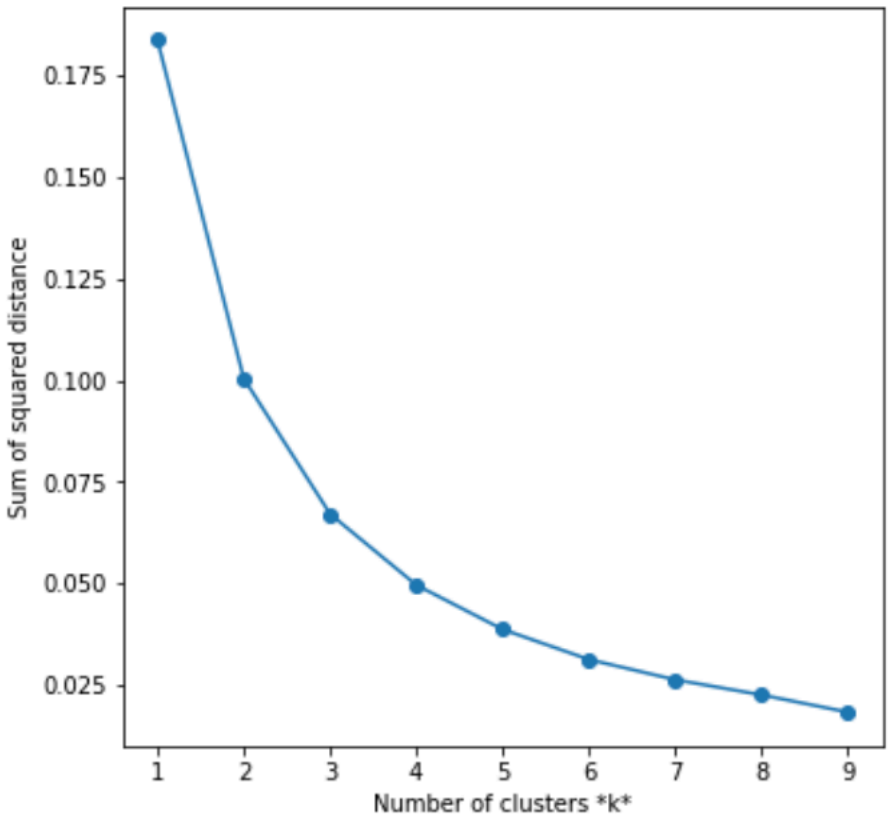
Neighborhood	Average Adjacent Score
Alamo Square	0.187764
Anza Vista	0.175835
Balboa Terrace	0.190930
Bayview	0.197360
Bayview Heights	0.210203

Finally, we can run k-means clustering on both the Ranked Score and the Average Adjacent Score.



3.3 Evaluation

I can see some trends in the k-means clustering analysis, as the similar neighborhoods are clustered together. However, how can I be sure that these results are meaningful? To reach a conclusion, I need to evaluate our results. I will do that by first determining the best number of clusters. The elbow method of evaluating k-means clustering involves iterating through different values of k and then seeing if there is a sharp angle in the line of sum of squared distance. This elbow represents the point where increasing k leads to greatly diminishing returns.

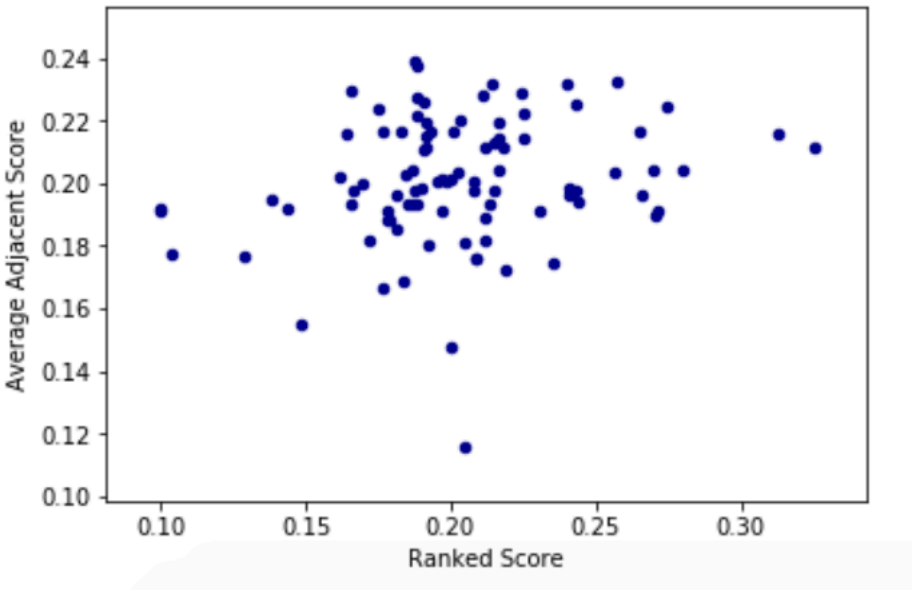


The plot above does not have have an obvious elbow in it, though we can see that as k increases, there are diminishing returns on the sum of squared distance between each cluster centroid and its members. While the decision to use k = 5 in the previous analyses was initially arbitrary, I can conclude that it is appropriate for this dataset.

4. Results

Neighborhood	Ranked Score	Average Adjacent Score
Golden Gate Heights	0.325000	0.211571
Forest Knolls	0.313158	0.216164
Ingleside Heights	0.279730	0.204443
Lincoln Park	0.274490	0.224153
Twin Peaks	0.271739	0.191271

When looking at the top 5 neighborhoods sorted by Ranked Score, we see that Golden Gate Heights is a clearly strong recommendation.



Looking at the point with the highest Ranked Score, we see that it is still higher than a majority of other points in terms of Average Adjacent Score. Looking at the point with the highest Average Adjacent Score, we can see that it falls well short in terms of Ranked Score.

5. Discussion

This analysis produced a clear recommendation based on the weights given by the target audience. By looking at the neighborhood features, I can group similar neighborhoods and see which ones can solve the business problem.

One of the limitations of this analysis is the lack of a complete dataset. For example, government buildings, airports, hotels, and cemeteries were the kinds of venues omitted as categories. While the Foursquare API can certainly be more thorough, I decided these other categories would not be relevant for this analysis. Also, while I am fairly certain that the Transit venues were well represented in this dataset, there are likely Restaurants that were missed due to the Foursquare result limit. Also, when calling the explore function in the API, there is an intent parameter which changes the way the API returns results. The checkin option returns for the most likely venues a user would check-in to, while the browse option returns all venues within a given area. Due to the lack of uniformity in neighborhood shapes, it may be better to split the city into a grid and run the API call against it. One other possibility is to use the Foursquare database, which would ensure a thorough dataset, but it would not be a free solution.

Another way to improve this analysis would be to include the number of venues as part of the analysis. Since I used the proportion of each category in each neighborhood, a neighborhood with a small number of venues is treated the same as a neighborhood with a larger number. However, this could possibly skew the analysis towards larger neighborhoods. This may require taking another look at the business problem; would quieter, more residential neighborhoods be considered desirable by the target audience? This may require looking at other data sources, as the Foursquare API does not include residences.

7. Conclusion

Using GIS data from DataSF and the venue data from Foursquare, I was able to determine which neighborhood would be desirable for a target audience that has a preference for Outdoors & Recreation, Food, and Transit. By using k-means clustering, I could group similar neighborhoods, which allows me to eliminate less desirable neighborhoods from the final recommendation. Looking at other data available in the Foursquare API can further expand this analysis. Finally, this analysis could be adapted to other audiences by changing the weights of the categories. In the city of San Francisco, there is sure to be a neighborhood for any audience's preferences.