Best Neighborhood in Denver, Colorado based on Crime and Venues data

By Deryan Thimoti Yudianto

1. Introduction

Denver is one of the most well-known city in Colorado, United States, and it also known as the 'Mile High City' because of the city's altitude which is 1 mile above sea level. There are more than 200 parks in Denver. Denver also had four major professional sports teams: Denver Nuggets (Basketball), Denver Broncos (American Football), Denver Avalanche (Ice Hockey), and Colorado Rockies (Baseball). This city also considered as one of the top city for job prospects and business growth.

Because of those reasons, Denver became one of the most important city in United States. People who want to settle a move to Denver must consider a lot of things, such as tourist attractions, restaurants, public transportations, crime data, and much more. In this project, I'll use the data to cluster the best Denver neighborhood based on public venues and crime data. The stakeholders in this project are people who want to move to Denver, either they are university students, young executives, or families.

2. Data

2.1. Data Collection

There are several data that I'll be using in this project. First, I'll be using Denver Crime Data that I downloaded from Kaggle, then I'm also going to use Denver Neighborhoods that I took from Wikipedia. Overall, these are some of the data that I'm going to use in my project:

1. Denver Crime Data (taken from Denver Open Data Catalog)
2. List of neighborhoods in Denver (web scrapped from Wikipedia)
3. List of venues in Denver (taken with Foursquare API)

2.2.Data Cleaning and Preprocessing

2.2.1. Denver Neighborhood Data

In my project, I used Beautiful Soup to scrap the Neighborhoods in Denver from Wikipedia (https://en.wikipedia.org/wiki/List_of_neighborhoods_in_Denver). The next step is to collect all of the latitudes and longitudes of the neighborhoods with Geopy. In this step, I also changed some of the coordinates of the neighborhoods because some of those locations are wrong / inaccurate. The following step is putting the neighborhoods, latitude, and longitudes in a Data Frame called **df_denver_neigh**. I decided to remove Park Hill from my Data Frame because it didn't have any relevant data, and the location was quite close to both North Park Hill and South Park Hill. In the end, I map all the neighborhoods in a folium map to show the exact location of each neighborhoods.

2.2.2. Denver Crime Data

The csv file itself was quite large, and there were too many rows, because apparently there were traffic collision data in the csv file. I made a new Data Frame called **df_denver** and put the table in that Data Frame. In Figure 1 we can see the number of rows in the Data Frame.

```
df_denver = pd.read_csv('denver_crime.csv')
df_denver.shape
```

```
1]: (470859, 19)
```

Figure 1, the amount of rows in the denver_crime csv file

There are more than 470000 rows of data, and it was way too much. I decided to remove any traffic collision data, because they were shown to be irrelevant with the my project, as shown in Figure 2.

```
df_denver = df_denver[df_denver.IS_CRIME != 0]
df_denver.shape
```

```
2]: (348056, 19)
```

Figure 2, df_denver after traffic collision data was excluded

I managed to cut almost 25% of the whole data, however it turned out that I still have too many data. To increase the effectiveness, I decided to filter the data once again, this time I filtered the crimes that happened between January 1st 2018 and February 29th 2020 as shown in Figure 3.

```
start = "1/1/2018  12:00:01 AM"
end = "2/29/2020  23:59:59 AM"
mask = (df_denver['FIRST_OCCURRENCE_DATE'] > start) & (df_denver['FIRST_OCCURRENCE_DATE'] < end)
```

```
df_denver = df_denver.loc[mask]
df_denver.shape
```

```
(134913, 19)
```

Figure 3, df_denver after further reduction

Finally, I got the most optimal value. The table below shows 19 columns in the data frame, the status of each column, and the reason why I'm keeping it or dropping it.

Table 1, df_denver columns

Column Name	**Status**	**Notes**
INCIDENT_ID	Dropped	Dropped because no longer used
OFFENSE_ID	Dropped	Dropped because no longer used
OFFENSE_CODE	Active	
OFFENSE_CODE_EXTENSION	Dropped	Dropped because no longer used
OFFENSE_TYPE_ID	Active	
OFFENSE_CATEGORY_ID	Active	Used for grouping
FIRST_OCCURENCE_DATE	Active	The date of occurrence between 2018 and 2020
LAST_OCCURENCE_DATE	Dropped	Dropped because no longer used
REPORTED_DATE	Dropped	Dropped because no longer used
INCIDENT_ADDRESS	Dropped	
GEO_X	Dropped	Dropped because no longer used
GEO_Y	Dropped	Dropped because no longer used
GEO_LON	Active	Longitude of the place where the crime took place

GEO_LAT	Active	Latitude of the place where the crime took place
DISTRICT_ID	Active	
PRECINCT_ID	Active	
NEIGHBORHOOD_ID	Active	Neighborhood names in Denver
IS_CRIME	Dropped	Dropped because no longer used
IS_TRAFFIC	Dropped	Dropped because no longer used

After removing some unnecessary columns, I changed the format of NEIGHBORHOOD_ID, OFFENSE_TYPE_ID, and OFFENSE_CATEGORY_ID to a more readable format. I used OFFENSE_CATEGORY_ID as the main column for clustering later. Because one of the category is too crowded with data, I decided to add some changes to OFFENSE_CATEGORY_ID based on OFFENSE_TYPE_ID. In the table 2 below are the changes of category that I made in this project.

Table 2, Changes in some categories

Old Category	**Type of Crime**	**New Category**
All Other Crimes	Gambling Device	Gambling
	Gambling Gaming Operation	
All Other Crimes	Money Laundering	White Collar Crime
All Other Crimes	Wiretapping	Privacy Violation
	Eavesdropping	
	Criminal Trespassing	
All Other Crimes	Kidnap Adult Victim	Kidnapping
	Kidnap Dv	
All Other Crimes	Explosive Incendiary Dev Pos	Illegal Item Possession
	Explosive Incendiary Dev Use	
	Explosives Posses	
	Contraband Possession	
	Fireworks Possession	
All Other Crimes	Extortion	Bribery and Extortion
	Bribery	
All Other Crimes	Animal Cruelty	Animal Cruelty

	Other Environment Animal Violation	
All Other Crimes	Police Disobey Lawful Order	Police Related Events
	Police False Information	
	Police Making A False Rpt	
	Police Resisting Arrest	
	Police Interference	
	Police Obstruct Investigation	
	Disarming A Peace Officer	
All Other Crimes	Weapon Altering Serial Number	Weapon Related Violation
	Weapon By Prev Offender Powpo	
	Weapon Carrying Concealed	
	Weapon Carrying Prohibited	
	Weapon Flourishing	
	Weapon Other Violation	
	Weapon Poss Illegal Dangerous	
	Weapon Unlawful Discharge Of	
	Weapon Unlawful Sale	
All Other Crimes	Littering	Public Disorder
	Public Fighting	
	Illegal Dumping	
	Escape	
	Escape Aiding	
	Bomb Threat	
	Intimidation Of A Witness	
	Reckless Endangerment	
All Other Crimes	Traf Vehicular Assault	Aggravated Assault

2.2.3. Denver Venues Data

In this project, I used Foursquare API to get all the venues near a specified neighborhood. I created a new Data Frame called **denver_venues** which consisted of Neighborhood name with its latitude and longitude, name of venue along with latitude and longitude, and also the venue's category. The venue's category column will play the key role in this project. I limited the number of venues fetch for each neighborhood to 50, and the radius of the neighborhood to 650. After all the venues data have been fetched, I looked at the number of different venue categories and I was completely astonished. There were more than 270 categories of places. I decided to combine some of the nearly identical category in order to get more relevant results. Table 3 in the next page are the changes in some categories in venues data.

Table 3, Changes in Venue Categories

Venues Category	**New Venues Category**
Whisky Bar	Bar
Bar	
Dive Bar	
Beach Bar	
Cocktail Bar	
Juice Bar	
Wine Bar	
Sports Bar	
Hotel Bar	
Piano Bar	
Beer Bar	
Greek Restaurant	European Restaurant
German Restaurant	
Pizza Place	
Mediterranean Restaurant	
French Restaurant	
Tapas Restaurant	
Modern European Restaurant	
Italian Restaurant	
Eastern European Restaurant	
Japanese Restaurant	Japanese Restaurant
Ramen Restaurant	
Sushi Restaurant	
Middle Eastern Restaurant	Asian Restaurant
Cantonese Restaurant	
Malay Restaurant	
Vietnamese Restaurant	

Mongolian Restaurant	
Dim Sum Restaurant	
Israeli Restaurant	
Thai Restaurant	
Chinese Restaurant	
Asian Restaurant	
Indian Restaurant	
Hawaiian Restaurant	North American Restaurant
Carribean Restaurant	
Cuban Restaurant	
Burrito Palace	Mexican Restaurant
Tex-Mex Restaurant	
Basketball Court	Sports Venue
Tennis Court	
Volleyball Court	
Baseball Field	
Stadium	
Paintball Field	
Athletic & Sports	
Football Stadium	
Gay Bar	Adult Venue
Strip Club	
Hookah Bar	
Gym / Fitness Center	Health & Fitness Center
Weight Loss Center	
Pilates Studio	
Martial Arts Dojo	
History Museum	Museum
Art Museum	
Science Museum	
Donut Shop	Snack / Dessert
Ice Cream Shop	
Dessert	
Bagel Shop	
Frozen Yogurt Place	
Snack Place	

Cupcake Shop	
Candy Store	

3. Methodology

3.1. Geopy

In this project I use Geopy to get latitudes and longitudes of Denver neighborhoods. However, because there are some neighborhood with incorrect / inaccurate locations, I changed some of the location data by searching it manually in Google, then I inserted it into the Data Frame.

Code Segment 1 : Fetch location data using Geopy and then save it in a list

```
denver_neighborhood = []
for x in table:
    locator = Nominatim(user_agent="myGeocoder", timeout=10)
    location = locator.geocode(x.get_text() + ", Denver, Colorado")
    if(hasattr(location, 'latitude')):
        if(x.get_text()=="Indian Creek"):
            denver_neighborhood.append([x.get_text(), 39.6863898, -104.
9051744])
        elif(x.get_text()=="Country Club"):
            denver_neighborhood.append([x.get_text(), 39.7201184, -104.
9749908])
        elif(x.get_text()=="Rosedale"):
            denver_neighborhood.append([x.get_text(), 39.6731541, -104.
9849813])
        elif(x.get_text()=="Park Hill"):
            denver_neighborhood.append([x.get_text(), 39.7655473, -104.
9572032])
        elif(x.get_text()=="University Park"):
            denver_neighborhood.append([x.get_text(), 39.6758905, -104.
9587653])
        else:
            denver_neighborhood.append([x.get_text(), location.latitude
, location.longitude])

    else:
        location2 = locator.geocode(x.get_text() + ", Denver")
        if(hasattr(location2, 'latitude')):
            if(x.get_text()=="Bear Valley"):
                denver_neighborhood.append([x.get_text(), 39.6601587, -
105.0848119])
            elif(x.get_text()=="Virginia Village"):
                denver_neighborhood.append([x.get_text(), 39.6893186, -
104.939641])
            elif(x.get_text()=="Washington Virginia Vale"):
                denver_neighborhood.append([x.get_text(), 39.6893186, -
104.939641])
            else:
                denver_neighborhood.append([x.get_text(), location.lati
tude, location.longitude])

        else:
            if(x.get_text() == "Cory-Merrill"):
                denver_neighborhood.append([x.get_text(), 39.6898585, -
104.9588103])
            elif(x.get_text() == "Gateway / Green Valley Ranch"):
                denver_neighborhood.append([x.get_text(), 39.7867524, -
104.7901241])
            elif(x.get_text() == "College View / South Platte"):
                denver_neighborhood.append([x.get_text(), 39.6707655, -
105.0247368])
                            else:
                print("Neighborhood = {}, Latitude = Unknown, Longitude
= Unknown".format(x.get_text()))
```

In the Code Segment on the, I used Geopy to fetch the location of each neighborhood. However after further inspection of each neighborhood, it turns out that some of the location data are inaccurate. In order to compensate for the error of the data, I looked through each and every single neighborhood, and then I searched the true latitudes and longitudes of the inaccurate neighborhood data. Finally, I inserted it to the same list.

3.2. Folium

I used Folium to visualize the location of each neighborhoods in Boston and the result of clustering. Code Segment 2 below shows the visualization of Denver Neighborhoods and Figure 4 shows the result of visualization using Folium.

Code Segment 2 : Denver neighborhoods visualization using Folium

```
map_denver = folium.Map(location=[39.7490235, -105.0103623], zo
om_start=11)

for lt, lng, neighborhood, in zip(df_denver_neigh['LATITUDE'],
df_denver_neigh['LONGITUDE'], df_denver_neigh['NEIGHBORHOOD']):
    label = '{}'.format(neighborhood)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lt, lng],
        radius=6,
        popup=label,
        color='purple',
        fill=True,
        fill_color='#f5f12c',
        fill_opacity=0.7,
        parse_html=False).add_to(map_denver)

map_denver
```

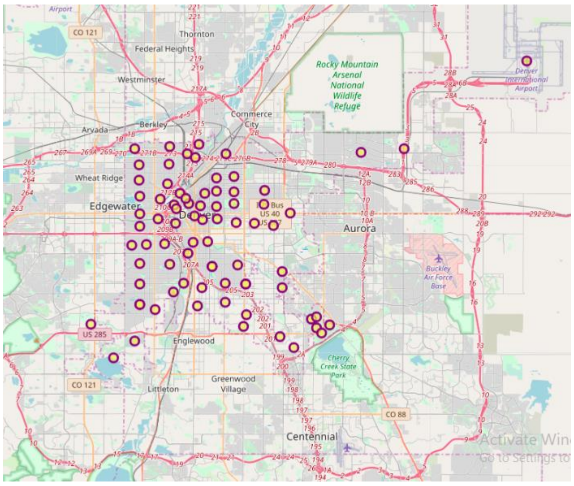

Figure 4, Result of Denver Neighborhood visualization

3.3. One Hot Encoding

According to Wikipedia, One Hot is a group of bits among which the legal combinations of values are only those with 1 and 0. In One Hot Encoding, I transformed the Data Frame to a categorical value. In short, I made a new Data Frame consisted of the neighborhoods along with the ‘dummies’, which are the transformed categorical data into 1 and 0. 1 indicates that the crime belong to that particular neighborhood, while 0 indicates otherwise. 1 row will always consist at least a value of 1. Code segment 3 shows the process of making a new Data Frame with One Hot Encoding.

Code Segment 3 : One Hot Encoding in Denver’s Crime Data Frame

```
denver_onehot = pd.get_dummies(df_denver[['OFFENSE_CATEGORY_ID'
]], prefix="", prefix_sep="")

denver_onehot['NEIGHBORHOOD_ID'] = df_denver['NEIGHBORHOOD_ID']

fixed_columns = [denver_onehot.columns[-1]] + list(denver_oneho
t.columns[:-1])
denver_onehot = denver_onehot[fixed_columns]
```

	NEIGHBORHOOD_ID	Aggravated Assault	All Other Crimes	Arson	Auto Theft	Bribery and Extortion	Burglary	Drug Alcohol	Gambling	Illegal Items Possession	...	Larceny	Murder	Other Crimes Against Persons	Police Related Events	Privacy Violation	Public Disorder	Robbery	Theft From Motor Vehicle	Weapon Related Violation	White Collar Crime
0	Stapleton	0	0	0	0	0	0	0	0	0	...	1	0	0	0	0	0	0	0	0	0
3	West Colfax	0	0	0	0	0	0	0	0	0	...	1	0	0	0	0	0	0	0	0	0
4	Montbello	0	1	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
11	College View / South Platte	0	0	0	0	0	0	0	0	0	...	1	0	0	0	0	0	0	0	0	0
12	Five Points	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	1	0	0

Figure 5, One Hot Encoding of Denver’s Crime Data

3.4. Top 5 most occurred crimes in Denver Neighborhoods

I decided to pick only top 5 of the most occurred crimes in Denver Neighborhoods because it allowed me to analyze the result much easier. Besides, there were only 21 categories of Crime. Code segment 4 in the next page shows how I sorted the crime by occurrence and Figure 6 shows the result in a table.

	NEIGHBORHOOD	1st Most Occurred Crime	2nd Most Occurred Crime	3rd Most Occurred Crime	4th Most Occurred Crime	5th Most Occurred Crime
0	Athmar Park	All Other Crimes	Public Disorder	Drug Alcohol	Theft From Motor Vehicle	Auto Theft
1	Auraria	Privacy Violation	Larceny	Public Disorder	All Other Crimes	Drug Alcohol
2	Baker	Larceny	All Other Crimes	Public Disorder	Theft From Motor Vehicle	Drug Alcohol
3	Barnum	All Other Crimes	Public Disorder	Auto Theft	Drug Alcohol	Theft From Motor Vehicle
4	Barnum West	All Other Crimes	Auto Theft	Public Disorder	Theft From Motor Vehicle	Drug Alcohol

Figure 6, the result of top 5 most occurred crimes in Denver Neighborhoods

Code Segment 4 : Getting top 5 most occurred crimes in each of Denver Neighborhoods

```
num_top_crime = 5

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

# create columns according to number of crimes
columns = ['NEIGHBORHOOD']
for ind in np.arange(num_top_crime):
    try:
        columns.append('{}{} Most Occured Crime'.format(ind+1, indicat
ors[ind]))
    except:
        columns.append('{}th Most Occured Crime'.format(ind+1))

# create a new dataframe
denver_neigh_crime_sorted = pd.DataFrame(columns=columns)
denver_neigh_crime_sorted['NEIGHBORHOOD'] = denver_grouped['NEIGHBORHO
OD_ID']

for ind in np.arange(denver_grouped.shape[0]):
    denver_neigh_crime_sorted.iloc[ind, 1:] = return_most_common_crime
(denver_grouped.iloc[ind, :], num_top_crime)

denver_neigh_crime_sorted.head()
```

3.5. Top 5 most common venues in Denver Neighborhoods

For Denver venues', I also decided to pick only top 5 of the most common venues in Denver for the same reason. However, there were more than 200 venue categories, so I reduced the number of categories by putting some of the venues in a similar categories as explained in Denver Venues Data. Code segment 5 shows how I sorted the top venues and Figure 7 shows the result in a table.

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Athmar Park	Warehouse Store	Home Service	Park	Bakery	Discount Store
1	Auraria	Theme Park	Theme Park Ride / Attraction	Bar	European Restaurant	Asian Restaurant
2	Baker	Bar	Asian Restaurant	Mexican Restaurant	Marijuana Dispensary	Hardware Store
3	Barnum	Home Service	Convenience Store	Marijuana Dispensary	Market	Gym
4	Barnum West	Mexican Restaurant	Discount Store	Snack / Dessert	Convenience Store	American Restaurant

Figure 7, the result of top 5 most common venues in Denver Neighborhoods

Code Segment 5 : Getting top 5 venues in each of Denver Neighborhoods

```
num_top_venues = 5

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

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

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = denver_grouped_venue['Ne
ighborhood']

for ind in np.arange(denver_grouped_venue.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_ven
ues(denver_grouped_venue.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()
```

3.6. K-Means Clustering

In this project I'm using K-Means Clustering to segment neighborhoods based on Crime data and Venues data. I'm running K-Means twice in this project, once for each data. I also set a different number of k so that the result match what I expected. Code Segment 6 and 7 show the k-Means Clustering on both Denver Crime Data and Denver Venues Data.

Code Segment 6 : k-Means Clustering on Denver Neighborhoods based on Denver Crime Data

```
kclusters = 4

denver_grouped_clustering = denver_grouped.drop('NEIGHBORHOOD_ID', 1)

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

Code Segment 7 : k-Means Clustering on Denver Neighborhoods based on Denver Venues Data

```
kclusters = 3

denver_grouped_clustering_venues = denver_grouped_venue.drop('N
eighborhood', 1)

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

4. Results

The first thing I'm going to discuss is the result of k-Means clustering based on Denver's crime data.

	NEIGHBORHOOD	LATITUDE	LONGITUDE	CLUSTER_LABELS	1st Most Occurred Crime	2nd Most Occurred Crime	3rd Most Occurred Crime	4th Most Occurred Crime	5th Most Occurred Crime
0	Baker	39.711595	-104.993750	2	Larceny	All Other Crimes	Public Disorder	Theft From Motor Vehicle	Drug Alcohol
1	Capitol Hill	39.735875	-104.979921	3	Drug Alcohol	Public Disorder	Larceny	All Other Crimes	Theft From Motor Vehicle
2	Central Business District	39.747378	-104.992737	3	Larceny	Privacy Violation	Public Disorder	Drug Alcohol	All Other Crimes
3	Cherry Creek	39.663610	-104.877444	2	Larceny	Theft From Motor Vehicle	Burglary	Public Disorder	All Other Crimes
4	Cheesman Park	39.736027	-104.966402	3	Drug Alcohol	Public Disorder	Larceny	All Other Crimes	Theft From Motor Vehicle

Figure 8, the cluster result of each neighborhood based on crime data

Code Segment 8 : Mapping the crime data cluster on Denver's map

```
map_clusters = folium.Map(location=[39.7490235, -105.0103623], zoom_st
art=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(denver_merged['LATITUDE'], denver_me
rged['LONGITUDE'], denver_merged['NEIGHBORHOOD'], denver_merged['CLUST
ER_LABELS']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_
html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

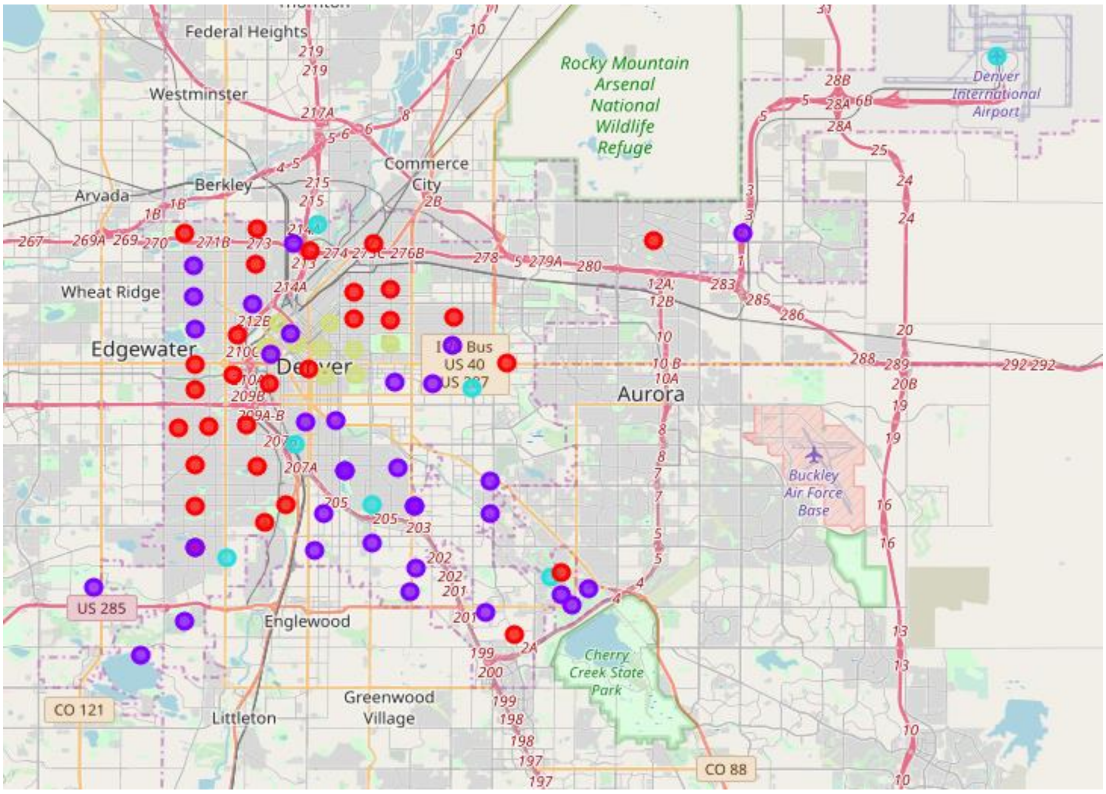

Figure 8, Cluster mapping result on Denver map based on crime data

Next is Denver venues' data. Quite similar to the crime data, however venues data have more categories.

Code segment 9 : Mapping the venues data cluster on Denver's map

```
map_venue_clusters = folium.Map(location=[39.7490235, -105.0103623], z
oom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(denver_venues_merged['LATITUDE'], de
nver_venues_merged['LONGITUDE'], denver_venues_merged['NEIGHBORHOOD'],
denver_venues_merged['CLUSTER_LABELS']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_
html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_venue_clusters)

map_venue_clusters
```

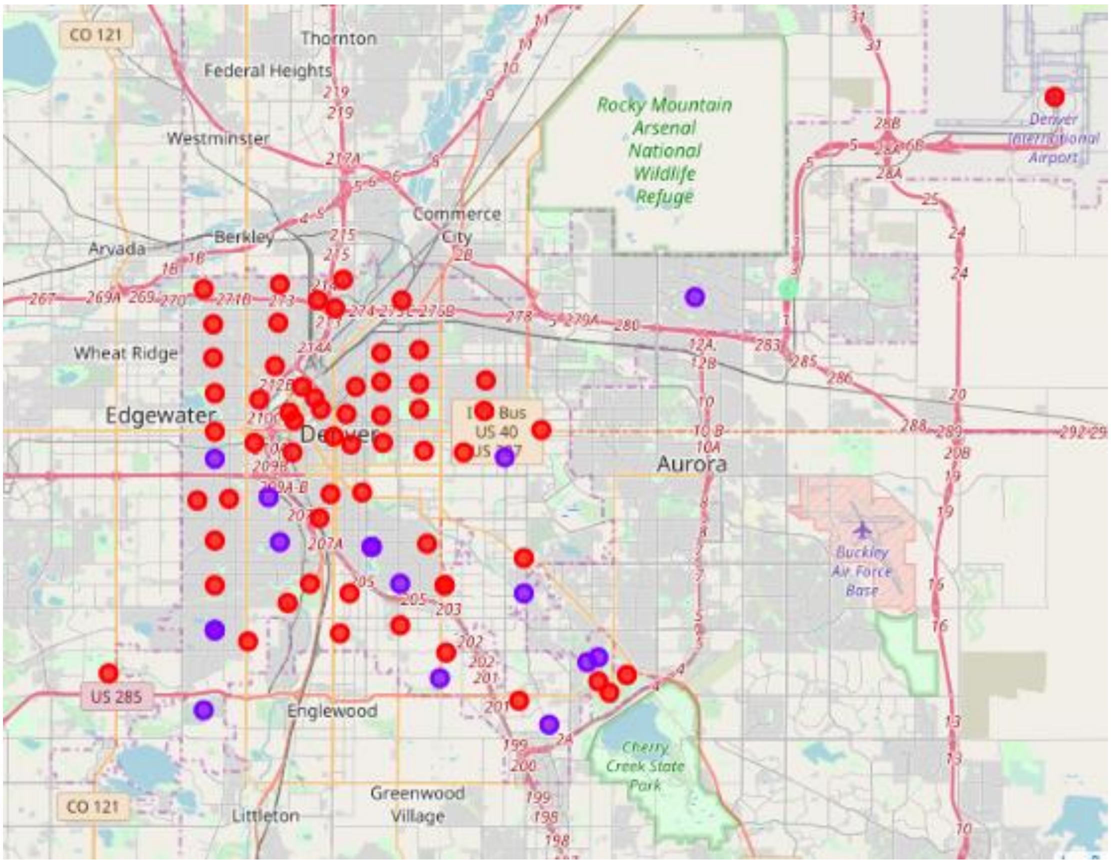

Figure 9, Cluster mapping result on Denver map based on venues data

5. Discussion

First, I changed the number of cluster in Denver's crime data to level of safety.

- Cluster 0 : Safest
- Cluster 1 : The most unsafe
- Cluster 2 : Unsafe
- Cluster 3 : Neutral

	NEIGHBORHOOD	LATITUDE	LONGITUDE	Safety Level	1st Most Occurred Crime	2nd Most Occurred Crime	3rd Most Occurred Crime	4th Most Occurred Crime	5th Most Occurred Crime
0	Baker	39.711595	-104.993750	Unsafe	Larceny	All Other Crimes	Public Disorder	Theft From Motor Vehicle	Drug Alcohol
1	Capitol Hill	39.735875	-104.979921	Neutral	Drug Alcohol	Public Disorder	Larceny	All Other Crimes	Theft From Motor Vehicle
2	Central Business District	39.747378	-104.992737	Neutral	Larceny	Privacy Violation	Public Disorder	Drug Alcohol	All Other Crimes
3	Cherry Creek	39.663610	-104.877444	Unsafe	Larceny	Theft From Motor Vehicle	Burglary	Public Disorder	All Other Crimes
4	Cheesman Park	39.736027	-104.966402	Neutral	Drug Alcohol	Public Disorder	Larceny	All Other Crimes	Theft From Motor Vehicle

Figure 10, The Dataframe after the clusters changed into safety level

Before I go to the venues data, let's discuss why I decided to change the cluster to level of safety. Cluster 0 contains the safest neighborhoods because most of the crimes that happened there were mainly minor crimes. I already changed some categories from All Other Crimes to some other categories to reduce the number of major crimes in All Other Crimes category. So, neighborhoods in cluster 0 were relatively safer than the neighborhoods in other clusters.

On the other hand, neighborhoods in cluster 1 and 2 considered unsafe thanks to the numbers of major crimes such as larceny, theft, burglary, and robbery. Those crimes are very threatening for people who lived in those neighborhoods. Cluster 1 was the most dangerous, because most of the crimes that happened there were extremely treacherous. For those reasons, I put 'Unsafe' label on cluster 2 and 'The most unsafe' label on cluster 1. Figure 12, 13, and 14 below show the neighborhoods with each respective cluster labels.

	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
0	Baker	39.711595	-104.993750	0	Bar	Marijuana Dispensary	Asian Restaurant	Mexican Restaurant	Intersection
1	Capitol Hill	39.735875	-104.979921	0	European Restaurant	Bar	Sandwich Place	Coffee Shop	Museum
2	Central Business District	39.747378	-104.992737	0	Hotel	American Restaurant	European Restaurant	Steakhouse	Sandwich Place
4	Cheesman Park	39.736027	-104.966402	0	Bar	Snack / Dessert	Café	Nightclub / Pub	Mexican Restaurant
5	City Park	39.747340	-104.950285	0	Zoo Exhibit	Museum	American Restaurant	Park	Track
6	Congress Park	39.733720	-104.948367	0	European Restaurant	Coffee Shop	Snack / Dessert	New American Restaurant	Burger Joint
7	City Park West	39.745376	-104.966577	0	European Restaurant	Pharmacy	Mexican Restaurant	Bar	Brewery
8	Civic Center	39.738181	-104.987744	0	Yoga Studio	Nightclub / Pub	Museum	Marijuana Dispensary	Breakfast Spot
9	Country Club	39.720118	-104.974991	0	Bus Stop	Park	Coffee Shop	Assisted Living	Furniture / Home Store
10	Lincoln Park	39.733149	-105.005190	0	Café	Coffee Shop	Bar	Art Gallery	Arts & Entertainment
11	North Capitol Hill	39.745624	-104.981598	0	American Restaurant	Coffee Shop	Hotel	European Restaurant	Mexican Restaurant
12	Speer	39.719187	-104.989091	0	Bar	Nightclub / Pub	American Restaurant	Coffee Shop	Brewery
13	Union Station	39.754891	-105.001352	0	Bar	Hotel	Restaurant	American Restaurant	Japanese Restaurant
14	Belcaro	39.703094	-104.946730	0	Gym / Fitness Center	Bakery	Mexican Restaurant	Paper / Office Supplies Store	Snack / Dessert
16	East Colfax	39.740629	-104.897748	0	Bar	South American Restaurant	Theme Park Ride / Attraction	Snack / Dessert	Sandwich Place
17	Hale	39.733021	-104.931128	0	Pharmacy	Coffee Shop	Park	Bar	Snack / Dessert
18	Hilltop	39.783079	-104.993961	0	Steakhouse	Sporting Goods Shop	Restaurant	Rental Car Location	Shipping Store
20	Lowry	39.698220	-104.905175	0	Liquor Store	Snack / Dessert	Asian Restaurant	European Restaurant	Grocery Store
23	Virginia Village	39.689319	-104.939641	0	Asian Restaurant	Mexican Restaurant	Coffee Shop	Clothing Store	Bakery
24	Washington Virginia Vale	39.689319	-104.939641	0	Asian Restaurant	Mexican Restaurant	Coffee Shop	Clothing Store	Bakery
25	Windsor	39.750865	-104.995794	0	Bar	European Restaurant	American Restaurant	Hotel	Restaurant
27	Cole	39.765630	-104.966557	0	Beer Garden	Mexican Restaurant	Convenience Store	European Restaurant	Rental Car Location
28	Elyria-Swansea	39.782958	-104.958113	0	Farm	Intersection	Mexican Restaurant	Sports Venue	Food Truck
29	Five Points	39.754658	-104.977986	0	Mexican Restaurant	Brewery	Convenience Store	Coffee Shop	Bar
30	Globeville	39.780732	-104.986972	0	Bar	Sporting Goods Shop	Intersection	Restaurant	Business Service
31	North Park Hill	39.756826	-104.921732	0	Brewery	Snack / Dessert	Bistro	Gym / Fitness Center	Liquor Store
32	South Park Hill	39.746650	-104.922043	0	European Restaurant	Gym / Fitness Center	Gym	Wine Shop	Auto Garage

Figure 12, Denver neighborhood in cluster 0 based on nearby venues

	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
3	Cherry Creek	39.663610	-104.877444	1	Construction & Landscaping	Park	Trail	Farmers Market	Zoo
15	Cory-Merrill	39.689858	-104.958810	1	Sports Venue	Park	Big Box Store	Breakfast Spot	Fabric Shop
19	Indian Creek	39.686390	-104.905174	1	Park	Gym / Fitness Center	Dance Studio	Lawyer	Zoo
21	Montclair	39.731735	-104.912948	1	Mexican Restaurant	Park	Asian Restaurant	Snack / Dessert	European Restaurant
26	Clayton	39.766760	-104.950199	1	Park	Coffee Shop	Market	Marijuana Dispensary	Mexican Restaurant
34	Whittier	39.756363	-104.966558	1	Park	Bar	Café	Grocery Store	Dog Run
37	Montbello	39.784223	-104.831154	1	Park	Playground	Sports Venue	Liquor Store	Zoo
38	Northeast Park Hill	39.665239	-104.872474	1	Park	Construction & Landscaping	Gym	Auto Dealership	Deli / Bodega
56	Washington Park	39.702081	-104.971034	1	Park	Sports Venue	Botanical Garden	Bike Rental / Bike Share	Track
57	Washington Park West	39.702081	-104.971034	1	Park	Sports Venue	Botanical Garden	Bike Rental / Bike Share	Track
58	Wellshire	39.658484	-104.941353	1	Pharmacy	Park	Golf Course	Playground	Sports Venue
59	Goldsmith	39.643414	-104.894206	1	Asian Restaurant	Mexican Restaurant	Park	Liquor Store	Discount Store
65	Fort Logan	39.647734	-105.043810	1	Park	Zoo	Dessert Shop	Farm	Fabric Shop
66	Harvey Park	39.674664	-105.039167	1	Historic Site	Park	Lake	Recreation Center	Mexican Restaurant
67	Harvey Park South	39.674664	-105.039167	1	Historic Site	Park	Lake	Recreation Center	Mexican Restaurant
69	Athmar Park	39.703682	-105.010741	1	Warehouse Store	Home Service	Park	Bakery	Construction & Landscaping
72	Mar Lee	39.689192	-105.039141	1	Fried Chicken Joint	Park	Grocery Store	Taco Place	Liquor Store
75	Valverde	39.718301	-105.015823	1	Park	Brewery	Gym / Fitness Center	Indie Theater	Dance Studio
76	Villa Park	39.730727	-105.039212	1	Park	Liquor Store	European Restaurant	Food & Drink Shop	Zoo

Figure 13, Denver neighborhood in cluster 1 based on nearby venues

	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
36	Gateway / Green Valley Ranch	39.786752	-104.790124	2	Bus Station	Zoo	Dessert Shop	Farm	Fabric Shop

Figure 14, Denver neighborhood in cluster 2 based on nearby venues

As you see from those clustering results, Cluster 0 is more suitable for workers and students because the amount of hangout places in the neighborhoods. Venues such as Coffee shops, bars, and restaurants are just the perfect spot for hangout. Meanwhile, venues in Cluster 1 are mostly parks, zoos, and historic sites which are perfect for families. Cluster 2 only consists of 1 neighborhood, and it can be suitable for everyone. For those reasons, I changed the number of cluster in venues data to the targeted groups of people. Here are the changes I made in the dataframe :

- Cluster 0 : Students / Workers
- Cluster 1 : Families
- Cluster 2 : Everybody

Figure 15 shows the dataframe after the change.

	NEIGHBORHOOD	LATITUDE	LONGITUDE	Targeted Groups	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Baker	39.711595	-104.993750	Students / Workers	Bar	Asian Restaurant	Mexican Restaurant	Marijuana Dispensary	Hardware Store
1	Capitol Hill	39.735875	-104.979921	Students / Workers	European Restaurant	Bar	Sandwich Place	Breakfast Spot	Bookstore
2	Central Business District	39.747378	-104.992737	Students / Workers	Hotel	American Restaurant	Steakhouse	Coffee Shop	Sandwich Place
3	Cherry Creek	39.663610	-104.877444	Families	Construction & Landscaping	Park	Trail	Farmers Market	Zoo
4	Cheesman Park	39.736027	-104.966402	Students / Workers	Bar	Nightclub / Pub	Café	Snack / Dessert	Mexican Restaurant

Figure 15, The Dataframe after the clusters were changed into targeted groups

6. Conclusion

The aim of this project is to show which neighborhood in Denver is the safest and the most suitable for targeted groups of people. Because of that, I grouped the result into a new dataframe called **result** which held the safest neighborhoods in Denver along with the targeted groups. The result dataframe consisted of 29 neighborhoods, 7 were more suitable for families while the rest 22 were more suitable for workers / students. Figure 16 shows the result dataframe and figure 17 shows the safest Denver neighborhood in the map.

	NEIGHBORHOOD	LATITUDE_x	LONGITUDE_x	Targeted Groups	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	LATITUDE_y	LONGITUDE_y	Safety Level
8	Civic Center	39.738181	-104.987744	Students / Workers	Nightclub / Pub	Museum	Marijuana Dispensary	Yoga Studio	Breakfast Spot	39.738181	-104.987744	Safest
10	Lincoln Park	39.733149	-105.005190	Students / Workers	Café	Bar	Coffee Shop	Art Gallery	Steakhouse	39.733149	-105.005190	Safest
16	East Colfax	39.740629	-104.897748	Students / Workers	Bar	South American Restaurant	Tattoo Parlor	Latin American Restaurant	Fast Food Restaurant	39.740629	-104.897748	Safest
25	Clayton	39.766760	-104.950199	Students / Workers	Mexican Restaurant	Coffee Shop	Market	Marijuana Dispensary	Liquor Store	39.766760	-104.950199	Safest
26	Cole	39.765630	-104.966557	Students / Workers	Park	European Restaurant	Perfume Shop	Nightclub / Pub	Convenience Store	39.765630	-104.966557	Safest
27	Elyria-Swansea	39.782958	-104.958113	Students / Workers	Farm	Intersection	Sports Venue	Boxing Gym	Mexican Restaurant	39.782958	-104.958113	Safest
29	Globeville	39.780732	-104.986972	Students / Workers	Bar	Park	Metro Station	Sporting Goods Shop	Hotel	39.780732	-104.986972	Safest
30	North Park Hill	39.756826	-104.921732	Students / Workers	Clothing Store	Arts & Crafts Store	Snack / Dessert	Food	Brewery	39.756826	-104.921732	Safest
32	Skyland	39.755713	-104.950224	Students / Workers	Zoo Exhibit	Gift Shop	Snack / Dessert	American Restaurant	Exhibit	39.755713	-104.950224	Safest
33	Whittier	39.756363	-104.966558	Students / Workers	Sports Venue	Park	Grocery Store	Café	Bar	39.756363	-104.966558	Safest
36	Montbello	39.784223	-104.831154	Families	Park	Sports Venue	Business Service	Food	Spa	39.784223	-104.831154	Safest
37	Northeast Park Hill	39.665239	-104.872474	Families	Steakhouse	Gym	Park	Deli / Bodega	Auto Dealership	39.665239	-104.872474	Safest
41	Chaffee Park	39.788442	-105.010995	Students / Workers	Grocery Store	Clothing Store	Thrift / Vintage Store	Liquor Store	Mexican Restaurant	39.788442	-105.010995	Safest
43	Jefferson Park	39.750621	-105.019779	Students / Workers	Mexican Restaurant	Theme Park Ride / Attraction	Bar	Brewery	Asian Restaurant	39.750621	-105.019779	Safest
44	Regis	39.786779	-105.043968	Students / Workers	Convenience Store	Golf Course	Burrito Place	Liquor Store	Zoo	39.786779	-105.043968	Safest
46	Sunnyside	39.775680	-105.011696	Students / Workers	Coffee Shop	Mexican Restaurant	Bar	Park	Japanese Restaurant	39.775680	-105.011696	Safest
49	Overland	39.689988	-104.997874	Students / Workers	Coffee Shop	Furniture / Home Store	Miscellaneous Shop	Mexican Restaurant	Pharmacy	39.689988	-104.997874	Safest
58	Goldsmith	39.643414	-104.894206	Families	Asian Restaurant	Mexican Restaurant	Park	Liquor Store	Distillery	39.643414	-104.894206	Safest
65	Harvey Park	39.674664	-105.039167	Families	Mexican Restaurant	Lake	Construction & Landscaping	Recreation Center	Park	39.674664	-105.039167	Safest
67	Athmar Park	39.703682	-105.010741	Families	Warehouse Store	Home Service	Park	Bakery	Discount Store	39.703682	-105.010741	Safest
68	Barnum	39.717683	-105.032455	Students / Workers	Home Service	Convenience Store	Marijuana Dispensary	Market	Gym	39.717683	-105.032455	Safest
69	Barnum West	39.717155	-105.046497	Students / Workers	Mexican Restaurant	Discount Store	Snack / Dessert	Convenience Store	American Restaurant	39.717155	-105.046497	Safest
70	Mar Lee	39.689192	-105.039141	Students / Workers	Fried Chicken Joint	Business Service	Discount Store	Check Cashing Service	Park	39.689192	-105.039141	Safest
71	Ruby Hill	39.683598	-105.007205	Students / Workers	Ski Area	Park	Music Venue	Liquor Store	Marijuana Dispensary	39.683598	-105.007205	Safest
72	Sun Valley	39.736379	-105.021459	Students / Workers	Brewery	Gym / Fitness Center	Bar	Hot Dog Joint	Taco Place	39.736379	-105.021459	Safest
73	Valverde	39.718301	-105.015823	Families	Park	Brewery	Indie Theater	Gym / Fitness Center	Business Service	39.718301	-105.015823	Safest
74	Villa Park	39.730727	-105.039212	Families	Park	Liquor Store	Home Service	Food & Drink Shop	European Restaurant	39.730727	-105.039212	Safest
75	West Colfax	39.740093	-105.039202	Students / Workers	Coffee Shop	Convenience Store	Mexican Restaurant	Snack / Dessert	Bar	39.740093	-105.039202	Safest
76	Westwood	39.704193	-105.039174	Students / Workers	Mexican Restaurant	Museum	Liquor Store	Distillery	Farm	39.704193	-105.039174	Safest

Figure 16, Result dataframe

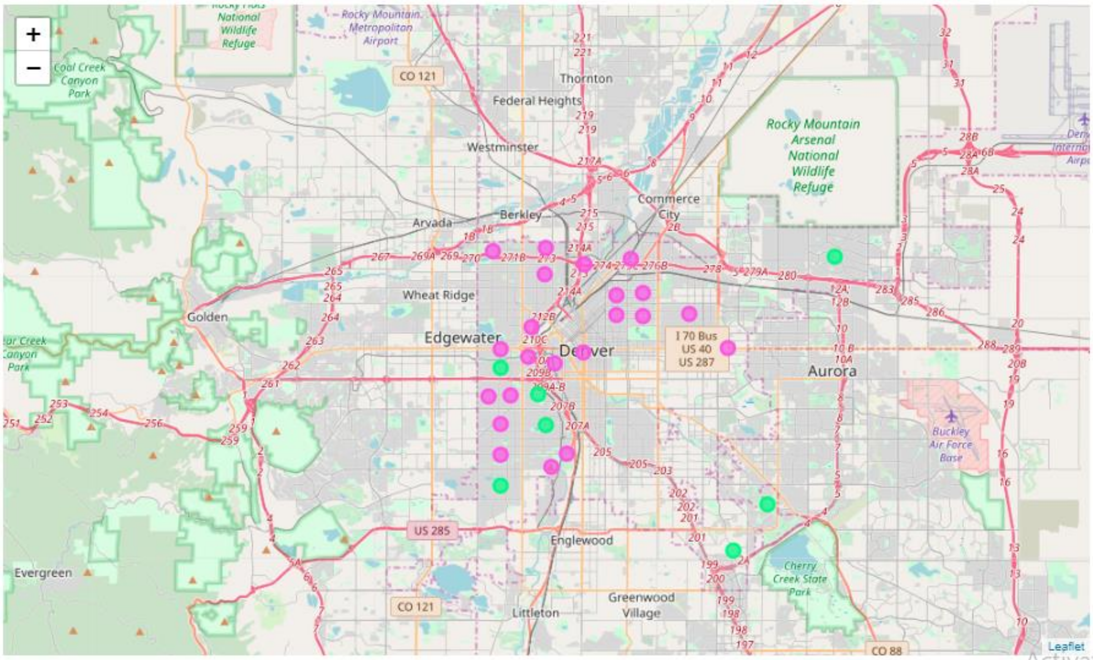

Figure 17, Denver safest neighborhood. Pink means suitable for workers / students while green means suitable for families