IBM Data Science Capstone Project

The Battle of Neighbourhoods

Final Report

Introduction

Boise has been one of the fastest growing cities in the United States for the past few years according to multiple sources. Main drivers for the population growth include families or retirees moving from other cities/states in search of lower home prices and crime rates, and increasingly more new hires recruited by high-tech companies such as Micron Technology and ON semiconductor. In addition, with the COVID19 outbreak and therefore more companies allowing employees to work remotely from home, Boise has also attracted a lot of out-of-the-states employees that are based in other cities/states. With a growing population comes with a growing demand in restaurants, thus the aim of this project is to study the neighbourhoods in Boise to determine possible locations for opening new restaurants. This project can benefit business owners and entrepreneurs who want to expand their business and invest in Boise.

Data section

Data used in this project are collected from multiple sources, a summary of which is provided below.

The neighbourhood list is scrapped from the city of Boise development website: https://www.cityofboise.org/departments/planning-and-development-services/planning-and-zoning/comprehensive-planning/neighborhood-planning/neighborhood-almanac/ using BeautifulSoup. Data cleaning is performed to remove some inactive neighbourhoods as they are certainly not the ideal locations for any business.

Geographical coordinates are obtained from the GeoPy library in python for the neighbourhood list obtained above.

Venue data are extracted using the Foursquare API and then KMeans clustering is performed to find out the ideal locations for opening new restaurants.

Methodology

1. Import Python libraries to Jupiter Notebook.

```
!pip install geopy
!pip install geocoder
!pip install folium

import numpy as np
import pandas as pd
import seaborn as sns
from geopy.geocoders import Nominatim
import geocoder
import requests
import folium
import matplotlib.cm as cm
import matplotlib.colors as colors
import matplotlib.pyplot as plt

from sklearn.cluster import KMeans

from pandas.io.json import json_normalize
from sklearn.metrics import silhouette_score

from bs4 import BeautifulSoup
```

2. Neighbourhood data collection and preprocessing

Extract table elements by using BeautifulSoup:

```
url = 'https://www.cityofboise.org/departments/planning-and-development-services/planning-and-zoning/comprehensive-planning/neighborhood-planning/neighb
html = requests.get(url)
soup = BeautifulSoup(html.content, "html5lib")
table=soup.find('table')
td = table.findAll('td')

td
```

By checking the table on the website and comparing it with the td elements, we know that the useful neighbourhood data is from 3,5,... in list td. Extract it with a for loop:

```
neigh = []

for i in np.arange(3,len(td),2):
    neigh_add = str(td[i]).strip('<td style="width:50%">').strip('</p>').split('<br/>')
    neigh.extend(neigh_add)

neigh
```

We notice that some neighbourhoods are labeled as "inactive". Remove those from the list:

```
neigh_active = []
neigh_active.extend(x for x in neigh if 'inactive' not in x)

neigh_active
```

Converting this list to panda dataframe:

```
df = pd.DataFrame(neigh_active)
df.columns = ['Neighbourhoods']

df
```

Now the extraction and cleaning of neighbourhoods data has been completed. Next step is to obtain the geographical coordinates from geocoder:

```
Lat = []
Lon = []

for neigh in df['Neighbourhoods']:
    g = geocoder.arcgis('{}, Boise, Idaho'.format(neigh))
    Lat.append(g.latlng[0])
    Lon.append(g.latlng[1])
```

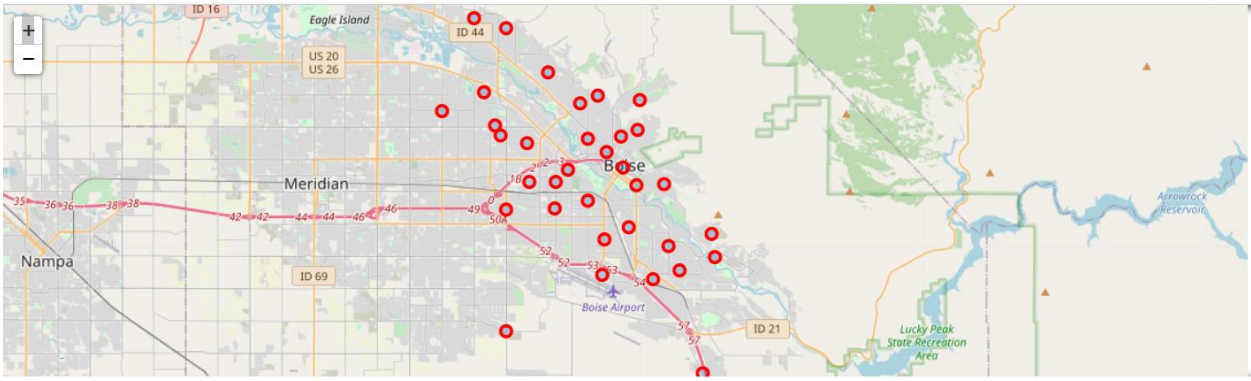
Add the geographical coordinates to the dataframe df. Also make a copy of Lat and Lon because geocoder sometimes will time out.

```
Lat_copy = Lat.copy()
Lon_copy = Lon.copy()

df['Latitude'] = Lat
df['Longitude'] = Lon

df
```

3. Exploratory Data visualization

```
address = "Boise, ID, USA"
geolocator = Nominatim(user_agent = "Boise_explorer")
location = geolocator.geocode(address)
Boise_lat = location.latitude
Boise_lon = location.longitude
```

```
map_Boise = folium.Map(location=[Boise_lat, Boise_lon], zoom_start=12)

for lat, lng, neigh in zip(df['Latitude'], df['Longitude'], df['Neighbourhoods']):
    label = '{}'.format(neigh)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='red',
        fill=True,
        fill_color='#3183cc',
        fill_opacity=0.3,
        parse_html=False).add_to(map_Boise)

map_Boise
```



4. Obtaining venue info from FSQ API

Taking the first neighbourhood as an example and explore how the data is structured from FSQ API:

```
neighbourhood_name = df.loc[0, 'Neighbourhoods']
neighbourhood_lat = df.loc[0, 'Latitude']
neighbourhood_lon = df.loc[0, 'Longitude']
neighbourhood_name
```

Find 100 venues in 1km range:

```
LIMIT = 100
radius = 1000

url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighbourhood_lat,
    neighbourhood_lon,
    radius,
    LIMIT)

results = requests.get(url).json()
results
```

Define the get_category_type function from the course:

```
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']
```

Then clean the JSON object (pick the relavant data) and store data in a dataframe:

```
venues = results['response']['groups'][0]['items']

venues_df = pd.json_normalize(venues)

needed_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.location.lng']
venues_df = venues_df.loc[:, needed_columns]

venues_df['venue.categories'] = venues_df.apply(get_category_type, axis=1)

venues_df.columns = [col.split(".")[-1] for col in venues_df.columns]

venues_df
```

	name	categories	lat	lng
0	Peak Thermo King – Boise	Home Service	43.526572	-116.149991
1	Cummins Sales and Service	Automotive Shop	43.520874	-116.147346
2	Eurest dining	Café	43.525227	-116.145209
3	Micron 17C Cafeteria	Cafeteria	43.530095	-116.149415

We can see this is not a very good location as it only has 4 venues in total, 2 out of which are cafeteria, but we got an idea of how the data is structured overall and verified how to obtain relavant data. Now we can repeat it for every neighbourhood in the list:

```
def get_venues(names, lats, lons, radius=1000):

    venues_list = []

    for name, lat, lon in zip(names, lats, lons):

        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            LIMIT)

        results = requests.get(url).json()['response']['groups'][0]['items']

        venues_list.append([(
            name,
            lat,
            lng,
            item['venue']['name'],
            item['venue']['categories'][0]['name'],
            item['venue']['location']['lat'],
            item['venue']['location']['lng']) for item in results])

    nearby_venues = pd.DataFrame([item for venue in venues_list for item in venue])
    nearby_venues.columns = ['Neighbourhood',
                  'Neighbourhood Lat',
                  'Neighbourhood Lon',
                  'Venue',
                  'Venue Category',
                  'Venue Lat',
                  'Venue Lon']

    return(nearby_venues)
```

```
Boise_venues = get_venues(df['Neighbourhoods'],df['Latitude'],df['Longitude'],radius = 1000)
```

```
Boise_venues
```

]:

	Neighbourhood	Neighbourhood Lat	Neighbourhood Lon	Venue	Venue Category	Venue Lat	Venue Lon
0	South Eisenman	43.522998	-116.26159	World Center for Birds of Prey	Zoo	43.516729	-116.255983
1	South Eisenman	43.522998	-116.26159	El Rancho del Crabtreeio's	Scenic Lookout	43.522603	-116.251302
2	Barber Valley	43.575474	-116.26159	Mad Swede Brewing Company	Brewery	43.577719	-116.273188
3	Barber Valley	43.575474	-116.26159	Maverik Adventures First Stop	Gas Station	43.575330	-116.273377
4	Barber Valley	43.575474	-116.26159	Sherwin-Williams Floorcovering Store	Hardware Store	43.578249	-116.272041
...	...	...	...	...	...	...	...
586	Winstead Park	43.626460	-116.26159	The Original Sunrise Cafe	Breakfast Spot	43.617996	-116.264910
587	Winstead Park	43.626460	-116.26159	Over 19	Adult Boutique	43.630121	-116.250441
588	Winstead Park	43.626460	-116.26159	State Liquor Store Garden City-111	Liquor Store	43.632321	-116.252388
589	Winstead Park	43.626460	-116.26159	Loose Screw Beer Co.	Brewery	43.633308	-116.253609
590	Winstead Park	43.626460	-116.26159	Sturman's Wine & Cigar	Wine Bar	43.631646	-116.251452

591 rows × 7 columns

```
Boise_venues['Venue Category'].value_counts()
```

5]:

```
Gas Station             33
Brewery                 27
Sandwich Place          25
Coffee Shop             23
Fast Food Restaurant    22
                        ..
Locksmith                1
Bookstore                1
Cuban Restaurant         1
Scenic Lookout           1
Food & Drink Shop        1
Name: Venue Category, Length: 102, dtype: int64
```

Check how many venues were returned for each neighbourhood to get a general idea of which neighbourhoods are more developed:

```
Boise_venues.groupby('Neighbourhood').Venue.count()
```

]:

```
Neighbourhood
Barber Valley                  10
Boise Heights                  16
Borah                          29
Centennial                      1
Central Bench                  26
Central Foothills              26
Central Rim                    44
Collister                      15
Depot Bench                    13
Downtown Boise                 43
East End                       15
Glenwood Rim                   25
Highlands                      21
Hillcrest                       3
Liberty Park                   14
Lusk District                  10
Morris Hill                    14
North End                      15
North West                      1
Pierce Park                     5
South Boise Village            31
South Cole                      1
South Eisenman                  2
Southeast Boise                 8
Southwest Ada County Alliance   3
Sunset                         12
Veteran's Park                 16
Vista                          31
Warm Springs Mesa              36
West Bench                     16
West Downtown                  34
West End                       15
West Valley                     8
Winstead Park                  32
Name: Venue, dtype: int64
```

Filter neighbourhoods with cafe:

Filter neighbourhoods with cafe:

```
food = Boise_venues['Venue Category'].str.contains('Restaurant|Bar|Café|Cafe|Pub|Cuisine', case=False, regex=True)
Boise_foodvenues = Boise_venues[food].reset_index(drop=True)

Boise_foodvenues['Neighbourhood'].value_counts()
```

```
Warm Springs Mesa                9
South Boise Village              9
Downtown Boise                   7
Vista                            7
Central Rim                      7
Borah                            6
Central Bench                    6
Central Foothills                5
West Downtown                    5
Glenwood Rim                     5
Winstead Park                    5
Collister                        4
Morris Hill                      3
East End                         3
West Bench                       3
Boise Heights                    3
Liberty Park                     3
North End                        2
Highlands                        2
Depot Bench                      2
Veteran's Park                   2
West End                         2
Southeast Boise                  1
Southwest Ada County Alliance    1
Lusk District                    1
Hillcrest                        1
Name: Neighbourhood, dtype: int64
```

We can see that the top neighbourhoods with a lot of resturants are Warm Spring Mesa, South Boise Village, Downtown Boise etc.

5. KMeans clustering

First need to do one-hot encoding:

```
Boise_onehot = pd.get_dummies(Boise_venues['Venue Category'])
Boise_onehot['Neighbourhood'] = Boise_venues['Neighbourhood']

columns = ['Neighbourhood'] + list(Boise_onehot.columns[0:-1])
Boise_onehot = Boise_onehot[columns]

Boise_onehot
```

	Neighbourhood	ATM	Accessories Store	Adult Boutique	Airport	Alternative Healer	American Restaurant	Aquarium	Arcade	Arts & Crafts Store	...	Sushi Restaurant	Taco Place	Thai Restaurant	Thrift / Vintage Store	Toy / Game Store	Trail	Video Store	Waste Facility	W
0	South Eisenman	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
1	South Eisenman	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
2	Barber Valley	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
3	Barber Valley	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
4	Barber Valley	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
586	Winstead Park	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
587	Winstead Park	0	0	1	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
588	Winstead Park	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
589	Winstead Park	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
590	Winstead Park	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	

591 rows × 103 columns

Then group by neighbourhood and take its mean:

```
Boise_onehot_grouped = Boise_onehot.groupby('Neighbourhood').mean().reset_index()
Boise_onehot_grouped.head()
```

2]:

	Neighbourhood	ATM	Accessories Store	Adult Boutique	Airport	Alternative Healer	American Restaurant	Aquarium	Arcade	Arts & Crafts Store	...	Sushi Restaurant	Taco Place	Thai Restaurant	Thrift / Vintage Store	Toy / Game Store	Trail	Video Store
0	Barber Valley	0.000000	0.0	0.0000	0.1	0.000000	0.0	0.0	0.0	0.0	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0
1	Boise Heights	0.000000	0.0	0.0625	0.0	0.000000	0.0	0.0	0.0	0.0	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0
2	Borah	0.034483	0.0	0.0000	0.0	0.034483	0.0	0.0	0.0	0.0	...	0.034483	0.034483	0.034483	0.034483	0.034483	0.0	0.0
3	Centennial	0.000000	0.0	0.0000	0.0	0.000000	0.0	0.0	0.0	0.0	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.0
4	Central Bench	0.038462	0.0	0.0000	0.0	0.038462	0.0	0.0	0.0	0.0	...	0.038462	0.038462	0.038462	0.038462	0.038462	0.0	0.0

5 rows × 103 columns

Sort the venues in descending order:

```
# method to sort venues
def return_most_common_venues(row, num_top_venues=10):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

Create a new dataframe to display the top 10 venues for each neighbourhood

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

# create columns according to number of top venues
columns = ['Neighbourhood']
for i in np.arange(10):
    try:
        columns.append('{}{} Most Common Venue'.format(i+1, indicators[i]))
    except:
        columns.append('{}th Most Common Venue'.format(i+1))

# create a new dataframe
Boise_venues_sorted = pd.DataFrame(columns=columns)
Boise_venues_sorted['Neighbourhood'] = Boise_onehot_grouped['Neighbourhood']

for i in np.arange(Boise_onehot_grouped.shape[0]):
    Boise_venues_sorted.iloc[i, 1:] = return_most_common_venues(Boise_onehot_grouped.iloc[i, :], 10)

Boise_venues_sorted.head()
```

7]:

	Neighbourhood	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	Barber Valley	Construction & Landscaping	Hardware Store	Airport	Moving Target	Gas Station	Business Service	Automotive Shop	Brewery	Zoo	Food
1	Boise Heights	Smoke Shop	Gas Station	Lingerie Store	Pet Store	Fast Food Restaurant	Sandwich Place	Liquor Store	Dive Bar	Food Truck	Brewery
2	Borah	Fast Food Restaurant	Cosmetics Shop	Sandwich Place	ATM	Paintball Field	Gas Station	Clothing Store	Credit Union	Middle Eastern Restaurant	Ice Cream Shop
3	Centennial	Ice Cream Shop	Zoo	Food Truck	Department Store	Diner	Discount Store	Dive Bar	Eastern European Restaurant	Eye Doctor	Farmers Market
4	Central Bench	Fast Food Restaurant	Sandwich Place	ATM	Auto Garage	Gym	Gym / Fitness Center	Fruit & Vegetable Store	IT Services	Ice Cream Shop	Middle Eastern Restaurant

Now we are ready to run KMeans with 5 clusters:

```
kclusters = 5

Boise_onehot_clustering = Boise_onehot_grouped.drop('Neighbourhood',1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=42).fit(Boise_onehot_clustering)

# add clustering labels
Boise_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

Boise_venues_sorted
```

3]:

	Cluster Labels	Neighbourhood	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	4	Barber Valley	Construction & Landscaping	Hardware Store	Airport	Moving Target	Gas Station	Business Service	Automotive Shop	Brewery	Zoo	Food
1	0	Boise Heights	Smoke Shop	Gas Station	Lingerie Store	Pet Store	Fast Food Restaurant	Sandwich Place	Liquor Store	Dive Bar	Food Truck	Brewery
2	0	Borah	Fast Food Restaurant	Cosmetics Shop	Sandwich Place	ATM	Paintball Field	Gas Station	Clothing Store	Credit Union	Middle Eastern Restaurant	Ice Cream Shop
3	3	Centennial	Ice Cream Shop	Zoo	Food Truck	Department Store	Diner	Discount Store	Dive Bar	Eastern European Restaurant	Eye Doctor	Farmers Market
4	0	Central Bench	Fast Food Restaurant	Sandwich Place	ATM	Auto Garage	Gym	Gym / Fitness Center	Fruit & Vegetable Store	IT Services	Ice Cream Shop	Middle Eastern Restaurant
5	4	Central Foothills	Coffee Shop	Brewery	Gym	Bowling Alley	Harbor / Marina	Home Service	Juice Bar	Motorsports Shop	Nightclub	Greek Restaurant
6	0	Central Rim	Gas Station	Pizza Place	Fast Food Restaurant	Pool Hall	Sandwich Place	Automotive Shop	Burger Joint	Mexican Restaurant	Furniture / Home Store	Burrito Place
7	4	Collister	Gym	Coffee Shop	Harbor / Marina	Cosmetics Shop	Nightclub	Juice Bar	Automotive Shop	Motorsports Shop	Pet Store	American Restaurant
8	0	Depot Bench	Cosmetics Shop	Spa	Gas Station	IT Services	Ice Cream Shop	Fruit & Vegetable Store	Middle Eastern Restaurant	Diner	Park	Clothing Store
9	0	Downtown Boise	Pizza Place	Gas Station	Automotive Shop	Burger Joint	Pool Hall	Sandwich Place	Fast Food Restaurant	Hobby Shop	Furniture / Home Store	Mexican Restaurant
10	0	East End	Gas Station	Cosmetics Shop	Fruit & Vegetable Store	Clothing Store	Eye Doctor	Middle Eastern Restaurant	Fast Food Restaurant	Sandwich Place	Spa	Bar

Results and Discussion

Examine each cluster to see if it's suitable for restaurants:

```
Boise_venues_sorted[Boise_venues_sorted['Cluster Labels'] == 0]
```

9]:

	Cluster Labels	Neighbourhood	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
1	0	Boise Heights	Smoke Shop	Gas Station	Lingerie Store	Pet Store	Fast Food Restaurant	Sandwich Place	Liquor Store	Dive Bar	Food Truck	Brewery
2	0	Borah	Fast Food Restaurant	Cosmetics Shop	Sandwich Place	ATM	Paintball Field	Gas Station	Clothing Store	Credit Union	Middle Eastern Restaurant	Ice Cream Shop
4	0	Central Bench	Fast Food Restaurant	Sandwich Place	ATM	Auto Garage	Gym	Gym / Fitness Center	Fruit & Vegetable Store	IT Services	Ice Cream Shop	Middle Eastern Restaurant
6	0	Central Rim	Gas Station	Pizza Place	Fast Food Restaurant	Pool Hall	Sandwich Place	Automotive Shop	Burger Joint	Mexican Restaurant	Furniture / Home Store	Burrito Place
8	0	Depot Bench	Cosmetics Shop	Spa	Gas Station	IT Services	Ice Cream Shop	Fruit & Vegetable Store	Middle Eastern Restaurant	Diner	Park	Clothing Store
9	0	Downtown Boise	Pizza Place	Gas Station	Automotive Shop	Burger Joint	Pool Hall	Sandwich Place	Fast Food Restaurant	Hobby Shop	Furniture / Home Store	Mexican Restaurant
10	0	East End	Gas Station	Cosmetics Shop	Fruit & Vegetable Store	Clothing Store	Eye Doctor	Middle Eastern Restaurant	Fast Food Restaurant	Sandwich Place	Spa	Bar

```
Boise_venues_sorted[Boise_venues_sorted['Cluster Labels'] == 1]
```

9]:

	Cluster Labels	Neighbourhood	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
18	1	North West	Trail	Zoo	Food & Drink Shop	Deli / Bodega	Department Store	Diner	Discount Store	Dive Bar	Eastern European Restaurant	Eye Doctor

```
Boise_venues_sorted[Boise_venues_sorted['Cluster Labels'] == 2]
```

1]:

	Cluster Labels	Neighbourhood	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
21	2	South Cole	River	Zoo	Credit Union	Deli / Bodega	Department Store	Diner	Discount Store	Dive Bar	Eastern European Restaurant	Eye Doctor

```
Boise_venues_sorted[Boise_venues_sorted['Cluster Labels'] == 3]
```

2]:

	Cluster Labels	Neighbourhood	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
3	3	Centennial	Ice Cream Shop	Zoo	Food Truck	Department Store	Diner	Discount Store	Dive Bar	Eastern European Restaurant	Eye Doctor	Farmers Market

```
Boise_venues_sorted[Boise_venues_sorted['Cluster Labels'] == 4]
```

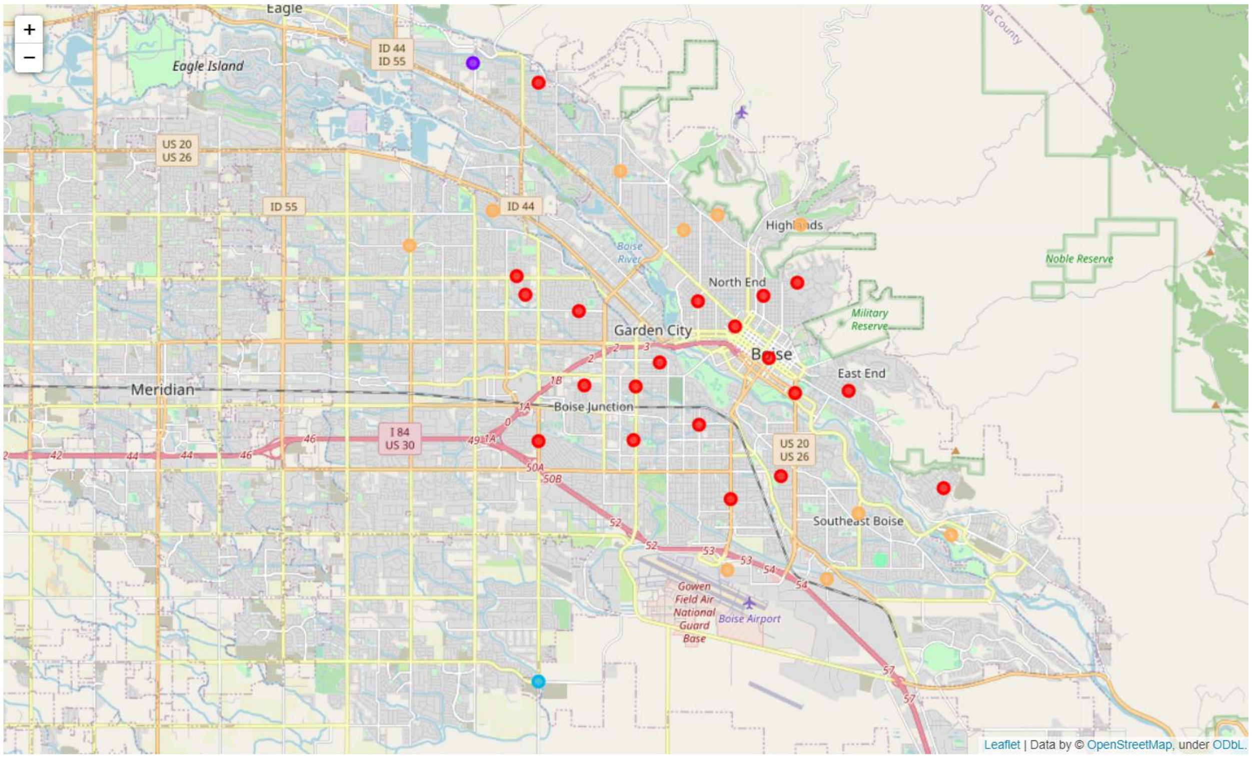
3]:

	Cluster Labels	Neighbourhood	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	4	Barber Valley	Construction & Landscaping	Hardware Store	Airport	Moving Target	Gas Station	Business Service	Automotive Shop	Brewery	Zoo	Food
5	4	Central Foothills	Coffee Shop	Brewery	Gym	Bowling Alley	Harbor / Marina	Home Service	Juice Bar	Motorsports Shop	Nightclub	Greek Restaurant
7	4	Collister	Gym	Coffee Shop	Harbor / Marina	Cosmetics Shop	Nightclub	Juice Bar	Automotive Shop	Motorsports Shop	Pet Store	American Restaurant
11	4	Glenwood Rim	Coffee Shop	Brewery	Gym	Sports Bar	Nightclub	Greek Restaurant	Paper / Office Supplies Store	Pizza Place	Rental Car Location	Salon / Barbershop

By checking the five clusters we obtained above, we notice that some of the clusters have more food related venues in their top 5 most common venues, while other clusters have a mixture of different venues. Neighbourhoods in cluster 0 are the most suitable for opening new restaurants. Finally we merge the coordinates back for plotting map.

```
Boise_venues_merged = df.copy()
Boise_venues_merged.rename(columns={'Neighbourhoods':'Neighbourhood'}, inplace=True)
Boise_venues_merged = Boise_venues_merged.join(Boise_venues_sorted.set_index('Neighbourhood'), on = 'Neighbourhood')

Boise_venues_merged.head()
```

```
# create map
map_clusters = folium.Map(location=[df["Latitude"][0], df["Longitude"][0]], zoom_start=12)

# 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(Boise_venues_merged['Latitude'], Boise_venues_merged['Longitude'], Boise_venues_merged['Neighbourhood'], Boise_venues_
    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
```



Conclusion

This project demonstrated basic data science skills in the following aspects: business problem formulation, data collection, preprocessing, exploratory visualization, machine learning and data analysis. The neighbourhoods in Boise are analyzed to provide recommendations for potential business owners to open new restaurants.