IBM – Applied Data Science Capstone

CAPSTONE PROJECT – FINAL REPORT

The Battle of Neighbourhoods

By Lakshmi Sajja

Contents

1. Introduction 3

Business Problem 3

Target Audience 3

2. Data 3

3. Solution Design Approach 3

4. Methodology 3

Exploratory Data Analysis 3

Data Modelling 5

5. Results 7

6. Discussion 7

7. Conclusion 7

1. Introduction

Business Problem

One of the question many budding entrepreneurs would have is what is the right place to start a business for ex: Grocery Store. Budding entrepreneurs would need to travel or enquire people who living in every community to find out an answer. Thanks to data socialization, we already have data available for every city now. So, Data science will help us in providing inputs required for entrepreneurs to find out suitable neighbourhood to start or expand their business.

Target Audience

Business Entrepreneurs or Companies who would like to start a Grocery store business **Vancouver City**, Canada.. The objective is to choose safest borough by analysing crime data and short list Neighbourhood where grocery stores are not amongst them.

2. Data

Steps to address business problem defined:

- Find safest borough based on crime statistics
- Find most common venues
- Choose right neighbourhood within the borough

Following data sources are needed to extract/generate required information:

- A dataset consisting of the crime statistics of each Neighbourhood in Vancouver along with type of crime, recorded year, month and hour.
- Borough information from Wikipedia to map with existing neighborhood data
- Use Open Cage Geocoder to find safest borough and explore neighbourhood by plotting it on maps using Folium and perform exploratory data analysis.
- Fetch data using Four Square API to explore neighbourhood venues and to apply machine learning algorithm to cluster neighbourhoods and present findings by plotting it on maps using Folium.

3. Solution Design Approach

Solution is approached in seven steps as listed below

1 Read data from crime report of Vancouver in 2018 - https://raw.githubusercontent.com/LakshmiSajja/Courseera_Capstone/master/vancouver_crime_records_2018_v1.csv

2 Gather additional details of Neighbourhoods from Wikipedia and Merge table to include crime data to include Borough.

3 Visualise crime repots in boroughs to identify safest borough and normalise the neighbourhoods of that borough. We will Use the resulting data and find 10 most common venues in each neighbourhood

4 Explore common venues of neighbourhoods in safest Borough using Foursquare API

5 One hot encoding to analyse each Neighbourhood

6 Cluster neighbourhoods using a unsupervised machine learning algorithm that clusters data based on predefined cluster size. Use K-Means clustering to address this problem so as to group data based on existing venues which will help in the decision making process.

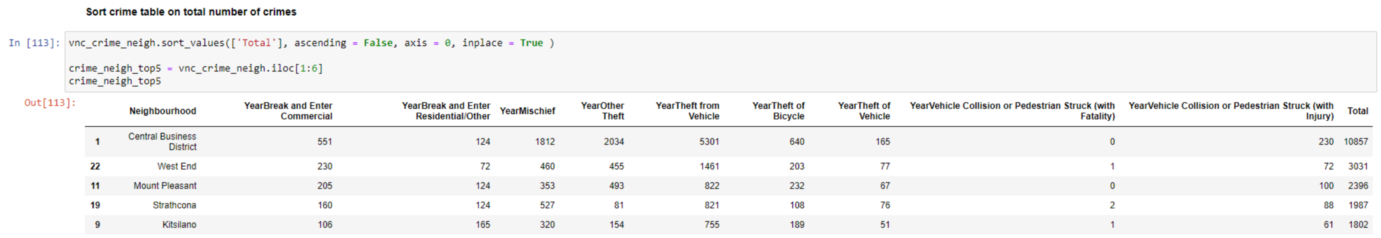
7 Concluding the Choices of Restaurants & Locations basis of the data analysis in Step

4. Methodology

Exploratory Data Analysis

Visualise the crime repots in different Vancouver boroughs to identity the safest borough and normalise the neighbourhoods of that borough. We will Use the resulting data and find 10 most common venues in each neighbourhood.

1. Sort data by Crimes per Neighbourhood



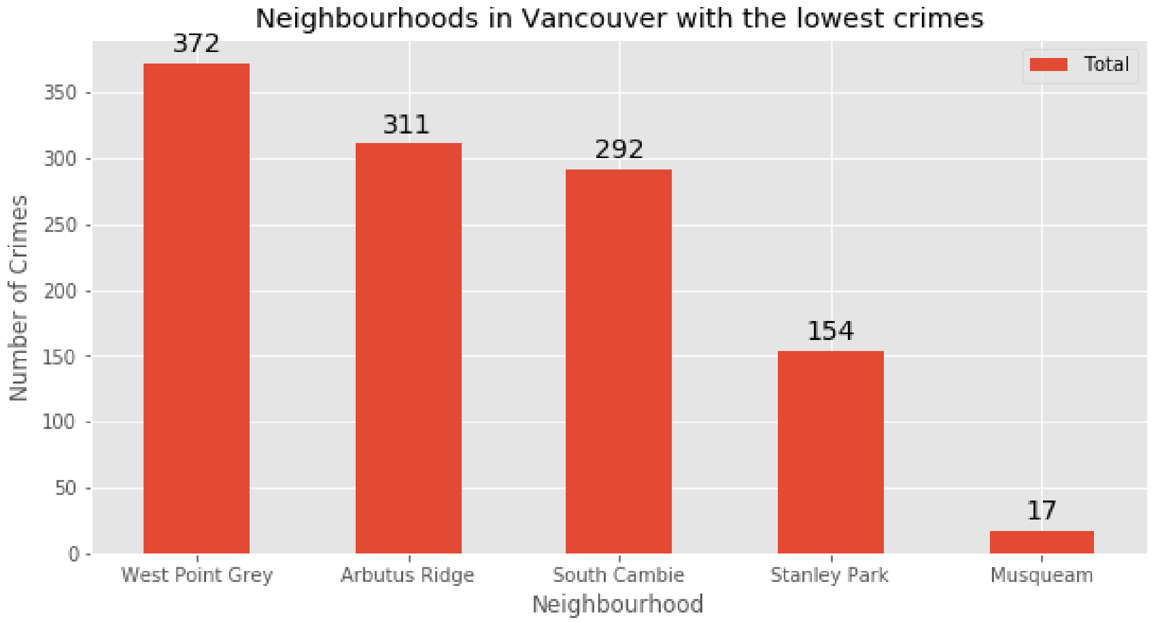
Sort crime table on total number of crimes

```
In [113]: vnc_crime_neigh.sort_values(['Total'], ascending = False, axis = 0, inplace = True )

          crime_neigh_top5 = vnc_crime_neigh.iloc[1:6]
          crime_neigh_top5
```

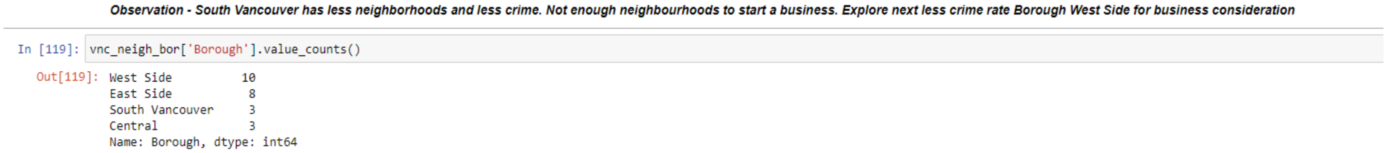
Out[113]:

	Neighbourhood	YearBreak and Enter Commercial	YearBreak and Enter Residential/Other	YearMischief	YearOther Theft	YearTheft from Vehicle	YearTheft of Bicycle	YearTheft of Vehicle	YearVehicle Collision or Pedestrian Struck (with Fatality)	YearVehicle Collision or Pedestrian Struck (with Injury)	Total
1	Central Business District	551	124	1812	2034	5301	640	165	0	230	10857
22	West End	230	72	460	455	1461	203	77	1	72	3031
11	Mount Pleasant	205	124	353	493	822	232	67	0	100	2396
19	Strathcona	160	124	527	81	821	108	76	2	88	1987
9	Kitsilano	106	165	320	154	755	189	51	1	61	1802

2. Neighbourhoods with Lowest crime



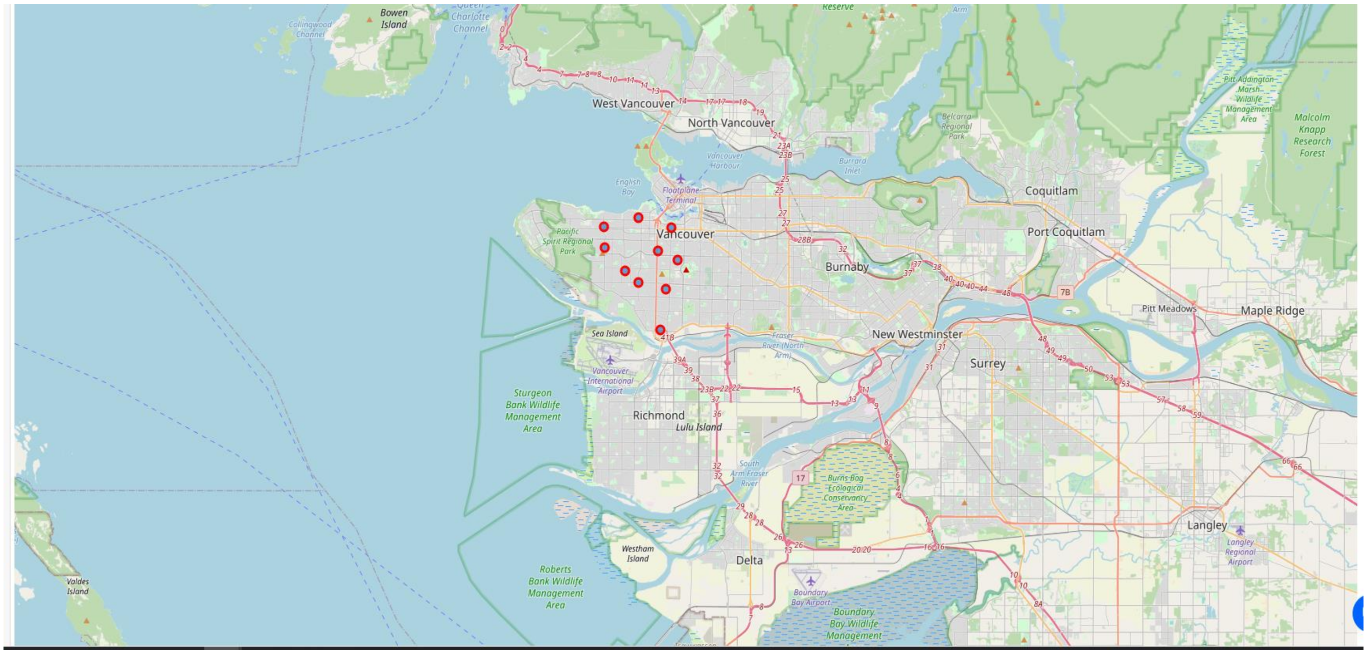
3. Number of Neighbourhoods in each Borough



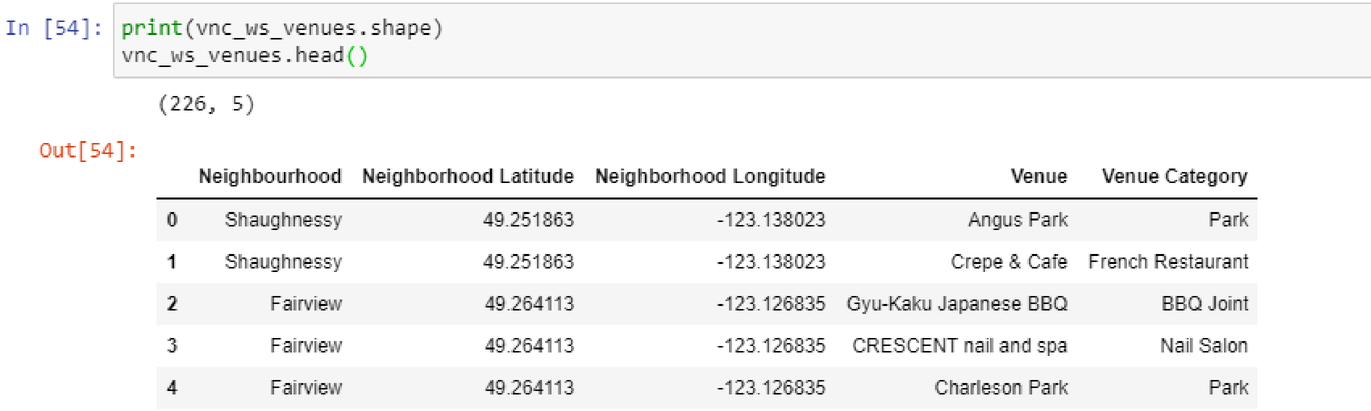
Observation - South Vancouver has less neighborhoods and less crime. Not enough neighbourhoods to start a business. Explore next less crime rate Borough West Side for business consideration

```
In [119]: vnc_neigh_bor['Borough'].value_counts()

Out[119]: West Side          10
          East Side           8
          South Vancouver     3
          Central             3
          Name: Borough, dtype: int64
```

4. Plot Westside Neighbourhood



5. Data frame of Neighbourhoods with Venues and count of Venues for each Neighbourhood



```
In [54]: print(vnc_ws_venues.shape)
         vnc_ws_venues.head()
```

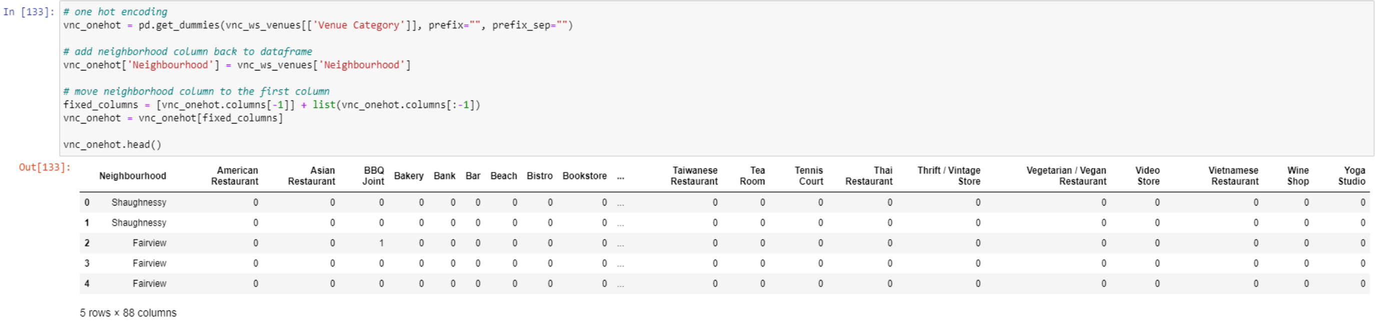
```
(226, 5)
```

Out[54]:

	Neighbourhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Category
0	Shaughnessy	49.251863	-123.138023	Angus Park	Park
1	Shaughnessy	49.251863	-123.138023	Crepe & Cafe	French Restaurant
2	Fairview	49.264113	-123.126835	Gyu-Kaku Japanese BBQ	BBQ Joint
3	Fairview	49.264113	-123.126835	CRESCENT nail and spa	Nail Salon
4	Fairview	49.264113	-123.126835	Charleson Park	Park

Data Modelling

1. One hot encoding to analyse each neighbourhood



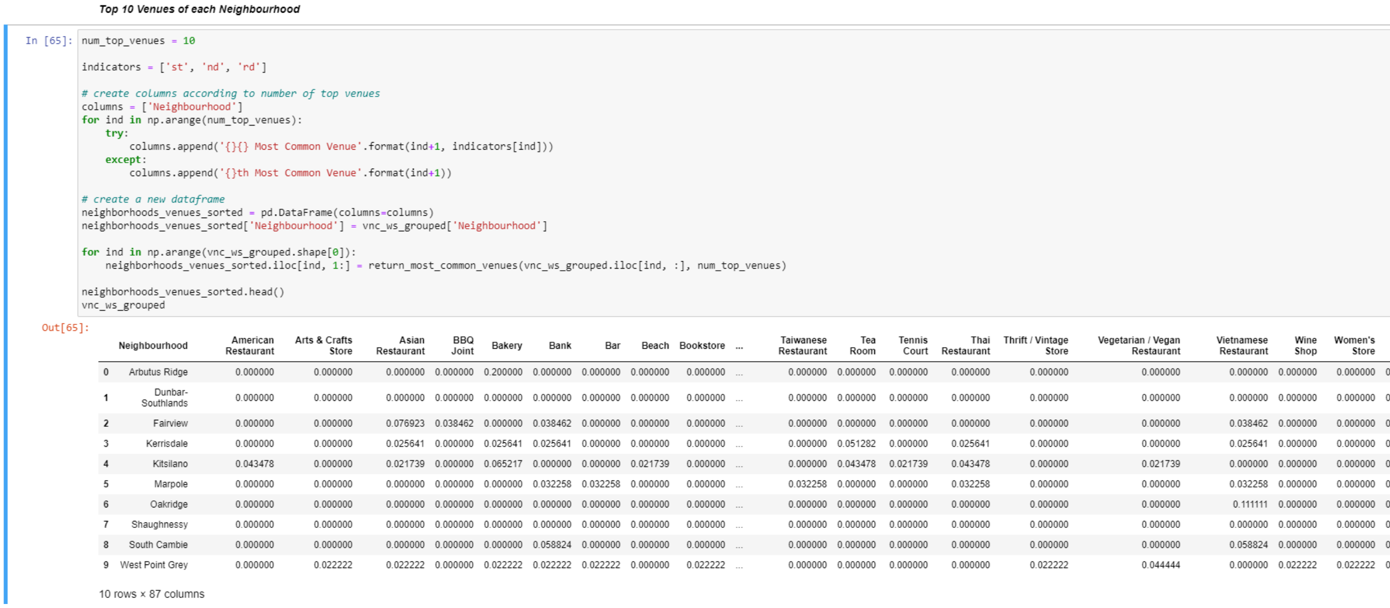
```
In [133]: # one hot encoding
          vnc_onehot = pd.get_dummies(vnc_ws_venues[['Venue Category']], prefix="", prefix_sep="")

          # add neighborhood column back to dataframe
          vnc_onehot['Neighbourhood'] = vnc_ws_venues['Neighbourhood']

          # move neighborhood column to the first column
          fixed_columns = [vnc_onehot.columns[-1]] + list(vnc_onehot.columns[:-1])
          vnc_onehot = vnc_onehot[fixed_columns]

          vnc_onehot.head()
```

Out[133]:

	Neighbourhood	American Restaurant	Asian Restaurant	BBQ Joint	Bakery	Bank	Bar	Beach	Bistro	Bookstore	...	Taiwanese Restaurant	Tea Room	Tennis Court	Thai Restaurant	Thrift / Vintage Store	Vegetarian / Vegan Restaurant	Video Store	Vietnamese Restaurant	Wine Shop	Yoga Studio
0	Shaughnessy	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
1	Shaughnessy	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
2	Fairview	0	0	1	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
3	Fairview	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4	Fairview	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

5 rows × 88 columns

2. Top 10 venues for each Neighbourhood



Top 10 Venues of each Neighbourhood

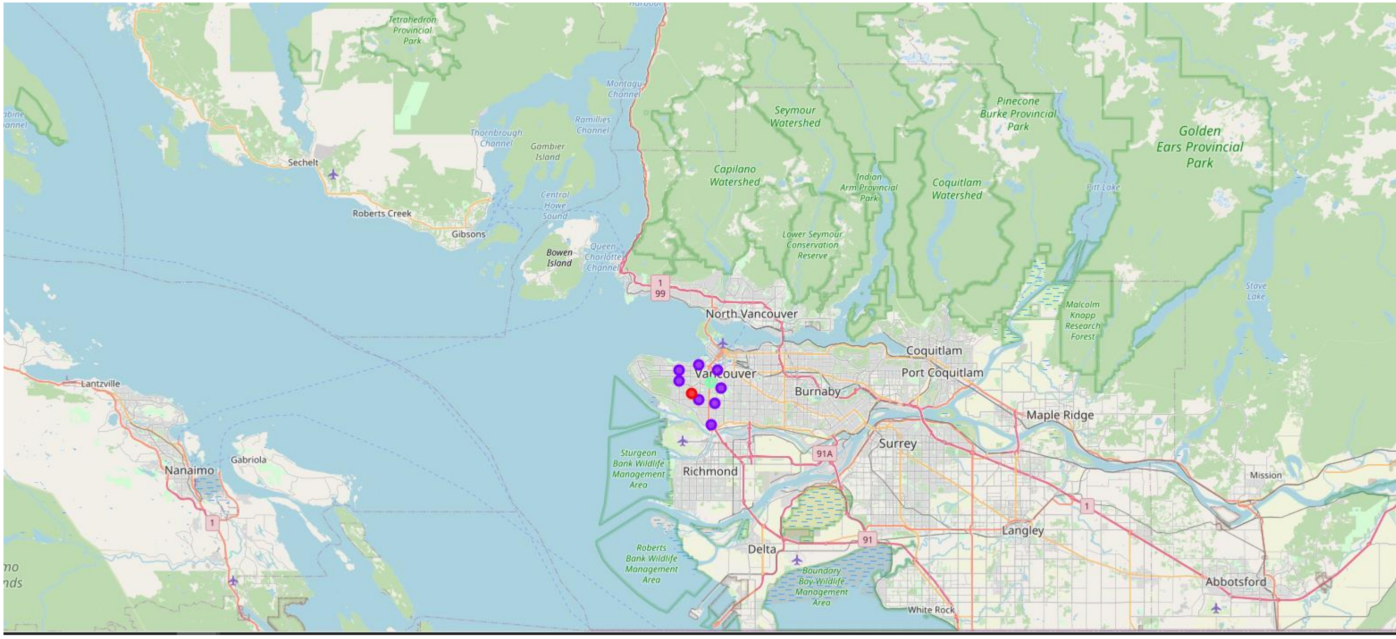
```
In [65]: num_top_venues = 10

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

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

         for ind in np.arange(vnc_ws_grouped.shape[0]):
             neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(vnc_ws_grouped.iloc[ind, :], num_top_venues)

         neighborhoods_venues_sorted.head()
         vnc_ws_grouped
```

Out[65]:

	Neighbourhood	American Restaurant	Arts & Crafts Store	Asian Restaurant	BBQ Joint	Bakery	Bank	Bar	Beach	Bookstore	...	Taiwanese Restaurant	Tea Room	Tennis Court	Thai Restaurant	Thrift / Vintage Store	Vegetarian / Vegan Restaurant	Vietnamese Restaurant	Wine Shop	Women's Store
0	Arbutus Ridge	0.000000	0.000000	0.000000	0.000000	0.200000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
1	Dunbar-Southlands	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	Fairview	0.000000	0.000000	0.076923	0.038462	0.000000	0.038462	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.038462	0.000000	0.000000
3	Kerrisdale	0.000000	0.000000	0.025641	0.000000	0.025641	0.025641	0.000000	0.000000	0.000000	...	0.000000	0.051282	0.000000	0.025641	0.000000	0.000000	0.025641	0.000000	0.000000
4	Kitsilano	0.043478	0.000000	0.021739	0.000000	0.065217	0.000000	0.000000	0.021739	0.000000	...	0.000000	0.043478	0.021739	0.043478	0.000000	0.021739	0.000000	0.000000	0.000000
5	Marpole	0.000000	0.000000	0.000000	0.000000	0.000000	0.032258	0.032258	0.000000	0.000000	...	0.032258	0.000000	0.000000	0.032258	0.000000	0.000000	0.032258	0.000000	0.000000
6	Oakridge	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.111111	0.000000	0.000000
7	Shaughnessy	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
8	South Cambie	0.000000	0.000000	0.000000	0.000000	0.000000	0.058824	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.058824	0.000000	0.000000
9	West Point Grey	0.000000	0.022222	0.022222	0.000000	0.022222	0.022222	0.022222	0.000000	0.022222	...	0.000000	0.000000	0.000000	0.000000	0.022222	0.044444	0.000000	0.022222	0.022222

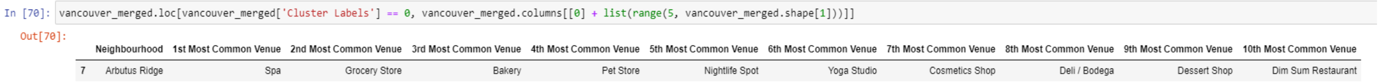
10 rows × 87 columns

3. Cluster Neighbourhoods – cluster Map



4. Examining Clusters

Cluster 1 –

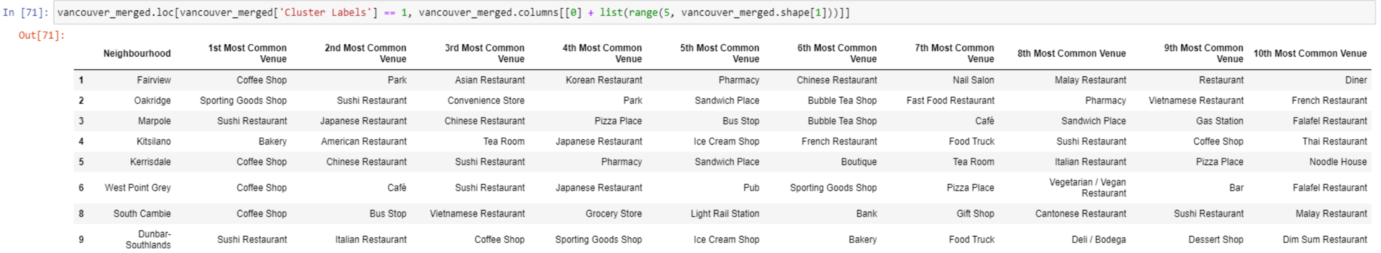


```
In [70]: vancouver_merged.loc[vancouver_merged['Cluster Labels'] == 0, vancouver_merged.columns[[0] + list(range(5, vancouver_merged.shape[1]))]]
```

Out[70]:

	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
7	Arbutus Ridge	Spa	Grocery Store	Bakery	Pet Store	Nightlife Spot	Yoga Studio	Cosmetics Shop	Deli / Bodega	Dessert Shop	Dim Sum Restaurant

Cluster 2 –

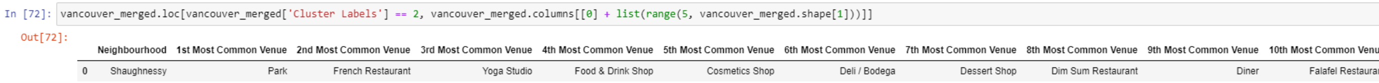


```
In [71]: vancouver_merged.loc[vancouver_merged['Cluster Labels'] == 1, vancouver_merged.columns[[0] + list(range(5, vancouver_merged.shape[1]))]]
```

Out[71]:

	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	Fairview	Coffee Shop	Park	Asian Restaurant	Korean Restaurant	Pharmacy	Chinese Restaurant	Nail Salon	Malay Restaurant	Restaurant	Diner
2	Oakridge	Sporting Goods Shop	Sushi Restaurant	Convenience Store	Park	Sandwich Place	Bubble Tea Shop	Fast Food Restaurant	Pharmacy	Vietnamese Restaurant	French Restaurant
3	Marpole	Sushi Restaurant	Japanese Restaurant	Chinese Restaurant	Pizza Place	Bus Stop	Bubble Tea Shop	Café	Sandwich Place	Gas Station	Falafel Restaurant
4	Kitsilano	Bakery	American Restaurant	Tea Room	Japanese Restaurant	Ice Cream Shop	French Restaurant	Food Truck	Sushi Restaurant	Coffee Shop	Thai Restaurant
5	Kerrisdale	Coffee Shop	Chinese Restaurant	Sushi Restaurant	Pharmacy	Sandwich Place	Boutique	Tea Room	Italian Restaurant	Pizza Place	Noodle House
6	West Point Grey	Coffee Shop	Café	Sushi Restaurant	Japanese Restaurant	Pub	Sporting Goods Shop	Pizza Place	Vegetarian / Vegan Restaurant	Bar	Falafel Restaurant
8	South Cambie	Coffee Shop	Bus Stop	Vietnamese Restaurant	Grocery Store	Light Rail Station	Bank	Gift Shop	Cantonese Restaurant	Sushi Restaurant	Malay Restaurant
9	Dunbar-Southlands	Sushi Restaurant	Italian Restaurant	Coffee Shop	Sporting Goods Shop	Ice Cream Shop	Bakery	Food Truck	Deli / Bodega	Dessert Shop	Dim Sum Restaurant

Cluster 3 –



```
In [72]: vancouver_merged.loc[vancouver_merged['Cluster Labels'] == 2, vancouver_merged.columns[[0] + list(range(5, vancouver_merged.shape[1]))]]
Out[72]:
```

	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 Venu
0	Shaughnessy	Park	French Restaurant	Yoga Studio	Food & Drink Shop	Cosmetics Shop	Deli / Bodega	Dessert Shop	Dim Sum Restaurant	Diner	Falafel Restaura

5. Results

Out of the three clusters outlines, Arbutus Ridge Neighbourhood in Cluster1 has most common venue as Grocery Store. Cluster 2 and Cluster 3 has most common venues as Restaurants. So, Arbutus Ridge neighbourhood would be a fit to start Grocery Store while Cluster 2 and Cluster 3 would be fit to start a Restaurant.

6. Discussion

The objective of the business problem was to help stakeholders identify one of the safest borough in Vancouver, and an appropriate neighbourhood within the borough to start a Grocery store. This has been achieved using crime data, Neighbourhood info from Wikipedia, exploring and applying clustering algorithm to achieve solution needed. However, there is chance for further improvement considering below –

- When combining data from multiple sources, inconsistent can happen. And lots of efforts are required to check, research and change the data before merge.
- For data obtained through API calls, different results are returned with different set of parameters and different point of time. Multiple trial and error runs are required to get the optimal result.
- Even after the dataset has been constructed, lots of research and analysis are required to decide if the data should be kept as is or be transform by normalization or standardization.

7. Conclusion

It is an attempt to explore the different possible analysis we could do in the available data and rationalize the decision. Although all of the goals of this project were met there is definitely room for further improvement by analysing few more supplementary data points like demographic information, Average Spent of the population, Proximity of other crowd pulling venues like Malls, shopping complex, Cinema halls etc. However, this project could definitely be handy to narrow down a Neighbourhood to start a grocery store.