Robberies in correlation with average income 2015

Group 11

Ashad Ahmed (100745913)
Milan Saju Samuel (100757350)
Shahroze Butt (100701891)

Background

- Goal: Crime rate and revenue correlation
- Anticipate likelihood of future crime
- Examine districts in Toronto
- Crime as a result of poverty
- Other factors

Data Sources

- Panda libraries
- Neighborhood profile Toronto
- Neighborhood crime rates
- Wikipedia statistics about Toronto

Methodology

- Python used as a programming language
- Pandas library to make data frame more readable
- BeautifulSoup to extract data from Wikipedia source HTML website
- Folium library to produce colours blue, green, yellow, orange and red which indicated likelihood of theft respectively

Code - Installing and importing libraries

Libraries that were installed to run the code

- It gives the system the ability to read longitudinal and latitude coordinates as well as the postal code.

```
!pip install beautifulsoup4
!pip install lxml
!pip install folium

#it is a library for data analysis
import pandas as pd
#it tranforms the json file into a panda dataframe library
from pandas.io.json import json_normalize
#library that handles all the data within vectorized manner
import numpy as np
import json
#libraray that handkes requests

import requests
#it helps plot libararies
import folium
import requests
#it is a library that displays images
from IPython.display import display_html
from IPython.core.display import HTML
from bs4 import BeautifulSoup
```

```
Requirement already satisfied: beautifulsoup4 in c:\programdata\anaconda3\lib\site-packages (4.10.0)
Requirement already satisfied: soupsieve>1.2 in c:\programdata\anaconda3\lib\site-packages (from beautifulsoup4) (2.2.1)
Requirement already satisfied: lxml in c:\programdata\anaconda3\lib\site-packages (4.6.3)
Requirement already satisfied: folium in c:\programdata\anaconda3\lib\site-packages (0.12.1.post1)
Requirement already satisfied: jinja2>=2.9 in c:\programdata\anaconda3\lib\site-packages (from folium) (2.11.3)
Requirement already satisfied: branca>=0.3.0 in c:\programdata\anaconda3\lib\site-packages (from folium) (0.4.2)
Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-packages (from folium) (2.26.0)
Requirement already satisfied: numpy in c:\programdata\anaconda3\lib\site-packages (from folium) (1.20.3)
Requirement already satisfied: MarkupSafe>=0.23 in c:\programdata\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (1.
1.1)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\programdata\anaconda3\lib\site-packages (from requests->foliu
m) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (3.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\programdata\anaconda3\lib\site-packages (from requests->folium) (202
1.10.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\programdata\anaconda3\lib\site-packages (from requests->folium)
(1.26.7)
```

Code - Reading Geospatial Coordinates CSV file data into dataframe

Postcode, Latitude, and Longitude

-It reads the Geospatial_Coordinates and outputs the postal code, along with the longitude and the latitutide.

In [5]:
```
df1 = pd.read_csv('Geospatial_Coordinates.csv')
df1
```

Out[5]:

	Postcode	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476
...	...	...	...
98	M9N	43.706876	-79.518188
99	M9P	43.696319	-79.532242
100	M9R	43.688905	-79.554724
101	M9V	43.739416	-79.588437
102	M9W	43.706748	-79.594054

103 rows × 3 columns

Code - Scraping Wikipedia page

Using the Wikipedia Page to create Data

-Using the Wikipedia page provided to create a dataset while assuring that values aren't being duplicated but rather removed if there is a duplicate available.

In [6]:

```
source = requests.get('https://en.wikipedia.org/w/index.php?title=List_of_postal_codes_of_Canada:_M&oldid=945633050').text
soup = BeautifulSoup(source,'lxml')
table = str(soup.table)
dfs = pd.read_html(table)
df = dfs[0]

# Removing the rows where Neighbourhood is 'Not assigned'
df0 = df[df.Neighbourhood != 'Not assigned']

#Drop neighbourhood names with same Postcode
df2 = df0.drop_duplicates(subset=['Postcode'])
df2.reset_index(inplace=True)
df2 = df2.drop('index', 1)
df2
```

Out[6]:

	Postcode	Borough	Neighbourhood
0	M3A	North York	Parkwoods
1	M4A	North York	Victoria Village
2	M5A	Downtown Toronto	Harbourfront
3	M6A	North York	Lawrence Heights
4	M7A	Downtown Toronto	Queen's Park
...	...	...	...
98	M8X	Etobicoke	The Kingsway
99	M4Y	Downtown Toronto	Church and Wellesley
100	M7Y	East Toronto	Business Reply Mail Processing Centre 969 Eastern
101	M8Y	Etobicoke	Humber Bay
102	M8Z	Etobicoke	Kingsway Park South West

Code - Merging geospatial coordinates CSV file data and Wikipedia source data

Datasets being Merged

-After having the Geospatial_coordinates we combine both the tables that we had created along with the postal codes and created the table below.

In [7]:
```
df3 = pd.merge(df1,df2,on = 'Postcode')
df3
```

Out[7]:

	Postcode	Latitude	Longitude	Borough	Neighbourhood
0	M1B	43.806686	-79.194353	Scarborough	Rouge
1	M1C	43.784535	-79.160497	Scarborough	Highland Creek
2	M1E	43.763573	-79.188711	Scarborough	Guildwood
3	M1G	43.770992	-79.216917	Scarborough	Woburn
4	M1H	43.773136	-79.239476	Scarborough	Cedarbrae
...	...	...	...	...	...
98	M9N	43.706876	-79.518188	York	Weston
99	M9P	43.696319	-79.532242	Etobicoke	Westmount
100	M9R	43.688905	-79.554724	Etobicoke	Kingsview Village
101	M9V	43.739416	-79.588437	Etobicoke	Albion Gardens
102	M9W	43.706748	-79.594054	Etobicoke	Northwest

103 rows × 5 columns

Code - Reading Neighborhood Crime Rates CSV data and merging with dataset

Merging the Neighbourhood Crime Rate Dataset

-They called for the csv link for the Neighbourhood Crime Rate and merging it with the table.

In [5]:
```
#calling for the file neighbourhood crime rate to then analyze and read the dataset
df4 = pd.read_csv('Neighbourhood_Crime_Rates.csv')
#we had to merge the dataset after the continuation with the neighbourhood
df5 = pd.merge(df3,df4,on = 'Neighbourhood')
df5
```

Out[5]:

	Postcode	Latitude	Longitude	Borough	Neighbourhood	OBJECTID	Hood_ID	Population	Assault_2014	Assault_2015	...	TheftOver_2015	Theft
0	M1B	43.806686	-79.194353	Scarborough	Rouge	98	131	46496	177	167	...	8	
1	M1C	43.784535	-79.160497	Scarborough	Highland Creek	72	134	12494	49	50	...	5	
2	M1E	43.763573	-79.188711	Scarborough	Guildwood	37	140	9917	52	38	...	2	
3	M1G	43.770992	-79.216917	Scarborough	Woburn	112	137	53485	352	395	...	13	
4	M1J	43.744734	-79.239476	Scarborough	Scarborough Village	15	139	16724	161	153	...	2	
5	M1M	43.716316	-79.239476	Scarborough	Cliffcrest	25	123	15935	79	97	...	5	
6	M1P	43.757410	-79.273304	Scarborough	Dorset Park	64	126	25003	135	168	...	10	
7	M1V	43.815252	-79.284577	Scarborough	Agincourt North	80	129	29113	67	76	...	5	
8	M2H	43.803762	-79.363452	North York	Hillcrest Village	62	48	16934	63	59	...	5	
9	M2K	43.786947	-79.385975	North York	Bayview Village	114	52	21396	79	104	...	8	
10	M2R	43.782736	-79.442259	North York	Willowdale West	97	37	16936	86	116	...	7	
11	M3C	43.725900	-79.340923	North York	Flemingdon Park	26	44	21933	128	147	...	3	
12	M3H	43.754328	-79.442259	North York	Bathurst Manor	86	34	15873	42	38	...	2	
13	M4A	43.725882	-79.315572	North York	Victoria Village	39	43	17510	118	138	...	6	
14	M4E	43.676357	-79.293031	East Toronto	The Beaches	7	63	21567	83	108	...	5	
15	M4H	43.705369	-79.349372	East York	Thorncliffe Park	9	55	21108	86	105	...	6	
16	M5P	43.696948	-79.411307	Central Toronto	Forest Hill North	34	102	12806	24	28	...	4	
17	M6C	43.693781	-79.428191	York	Humewood-Cedarvale	11	106	14365	43	52	...	3	
18	M6J	43.647927	-79.419750	West Toronto	Little Portugal	133	84	15559	85	84	...	1	

Code - Reading Neighborhood Profiles CSV data and merging with dataset

Neighbourhood Data

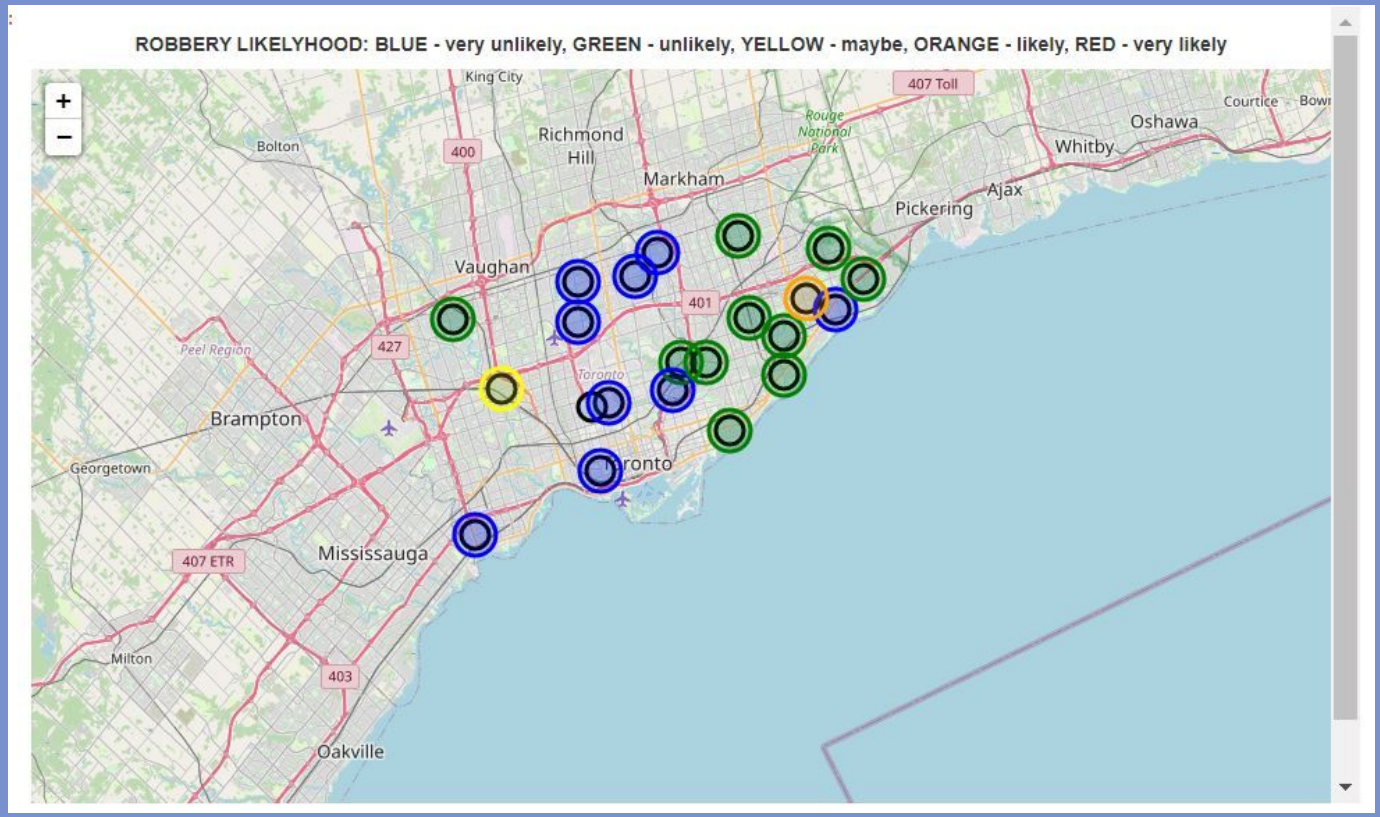
-It provides detailed information about each neighborhood within Toronto.

In [8]:
```
# it calls and reads for the neighbourhood profile
df6 = pd.read_csv('neighbourhood-profiles.csv')
#we had to merge the dataset after the continuation with the neighbourhood
df7 = pd.merge(df6,df5,on = 'Neighbourhood')
df7['Total - Income statistics in 2015 for the population aged 15 years and over in private households'] = pd.to_numeric(df7[
df7
```

Out[8]:

	Neighbourhood	Neighbourhood Number	TSNS2020 Designation	Population, 2016	Population, 2011	Population Change 2011-2016	Total private dwellings	Private dwellings occupied by usual residents	Population density per square kilometre	Land area in square kilometres	...	TheftOver_2015
0	Agincourt North	129	No Designation	29,113	30,279	-3.90%	9,371	9,120	3,929	7.41	...	5
1	Alderwood	20	No Designation	12,054	11,904	1.30%	4,732	4,616	2,435	4.95	...	3
2	Bathurst Manor	34	No Designation	15,873	15,434	2.80%	6,418	6,089	3,377	4.7	...	2
3	Bayview Village	52	No Designation	21,396	17,671	21.10%	10,111	9,532	4,195	5.1	...	8
4	Cliffcrest	123	No Designation	15,935	15,703	1.50%	6,094	5,902	2,273	7.01	...	5
5	Dorset Park	126	Emerging Neighbourhood	25,003	24,363	2.60%	8,995	8,777	4,146	6.03	...	10
6	Flemingdon Park	44	NIA	21,933	22,168	-1.10%	7,964	7,830	9,026	2.43	...	3
7	Forest Hill North	102	No Designation	12,806	12,474	2.70%	5,784	5,446	8,054	1.59	...	4
8	Guildwood	140	No Designation	9,917	9,816	1.00%	4,044	3,991	2,673	3.71	...	2
9	Highland Creek	134	No Designation	12,494	13,097	-4.60%	3,907	3,700	2,403	5.2	...	5
10	Hillcrest Village	48	No Designation	16,934	17,656	-4.10%	6,642	6,398	3,148	5.38	...	5
11	Humber Summit	21	NIA	12,416	12,525	-0.90%	4,288	3,897	1,570	7.91	...	16
12	Humewood-Cedarvale	106	No Designation	14,365	14,108	1.80%	6,865	6,566	7,682	1.87	...	3
13	Little Portugal	84	No Designation	15,559	12,050	29.10%	8,095	7,427	12,859	1.21	...	1
14	Rouge	131	No Designation	46,496	45,912	1.30%	13,730	13,389	1,260	36.89	...	8
15	Scarborough	139	NIA	16,724	16,609	0.70%	6,133	5,923	5,395	3.1		2

Clustered dataset visualised with a Folium Map



Results

- Areas with very low total income had higher levels of crime rate
- Areas near downtown Toronto which has the highest level of income, showed the lowest crime levels
- Areas near the suburbs like near Vaughan and Pickering showed less security than Toronto

Thank You