

The Battle of the Neighborhoods - Week 4 Report

Data

In this project, I will be using the following datasets to help solve Mr Willis' problem.

The main criterion for the Willis Family in selecting a suitable borough will be based on:

- Crime and Safety
- Number of Restaurants
- Schools
- House within Budget

Key Libraries being utilised

Python packages and Dependencies:

- Pandas - Library for Data Analysis
- NumPy – Library to handle data in a vectorized manner
- JSON – Library to handle JSON files
- Geopy – To retrieve Location Data
- Requests – Library to handle http requests
- Matplotlib – Python Plotting Module
- Sklearn – Python machine learning Library
- Folium – Map rendering Library
- BeautifulSoup - Webscraping tool

General

Firstly we need to find out how many boroughs are in Greater Manchester and their locations. This information will be obtained by Webscraping Wikipedia and then converted into a dataframe using Pandas.

- List of Manchester Boroughs - https://en.wikipedia.org/wiki/List_of_places_in_Greater_Manchester ,and Manchester PostCodes - https://simple.wikipedia.org/wiki/M_postcode_area#:~:text=The%20M%20postcode

[%20area%2C%20also,Manchester%20boroughs%20except%20for%20Stockport.](#)

The Coordinates (Latitude and Longitude) of each Borough will be extracted from Google Maps manually as Wikipedia does not contain the coordinates of each of the boroughs.

We will also obtain the Demographics of each borough to gauge their size and extract any other information from this that may be of use to the Willis family.

- Demographics of Manchester -
<https://www.citypopulation.de/en/uk/greatermanchester/> or
https://en.wikipedia.org/wiki/List_of_Greater_Manchester_settlements_by_population

In order to extract the information on Wikipedia we will use BeautifulSoup to web scrape the information contained in the tables.

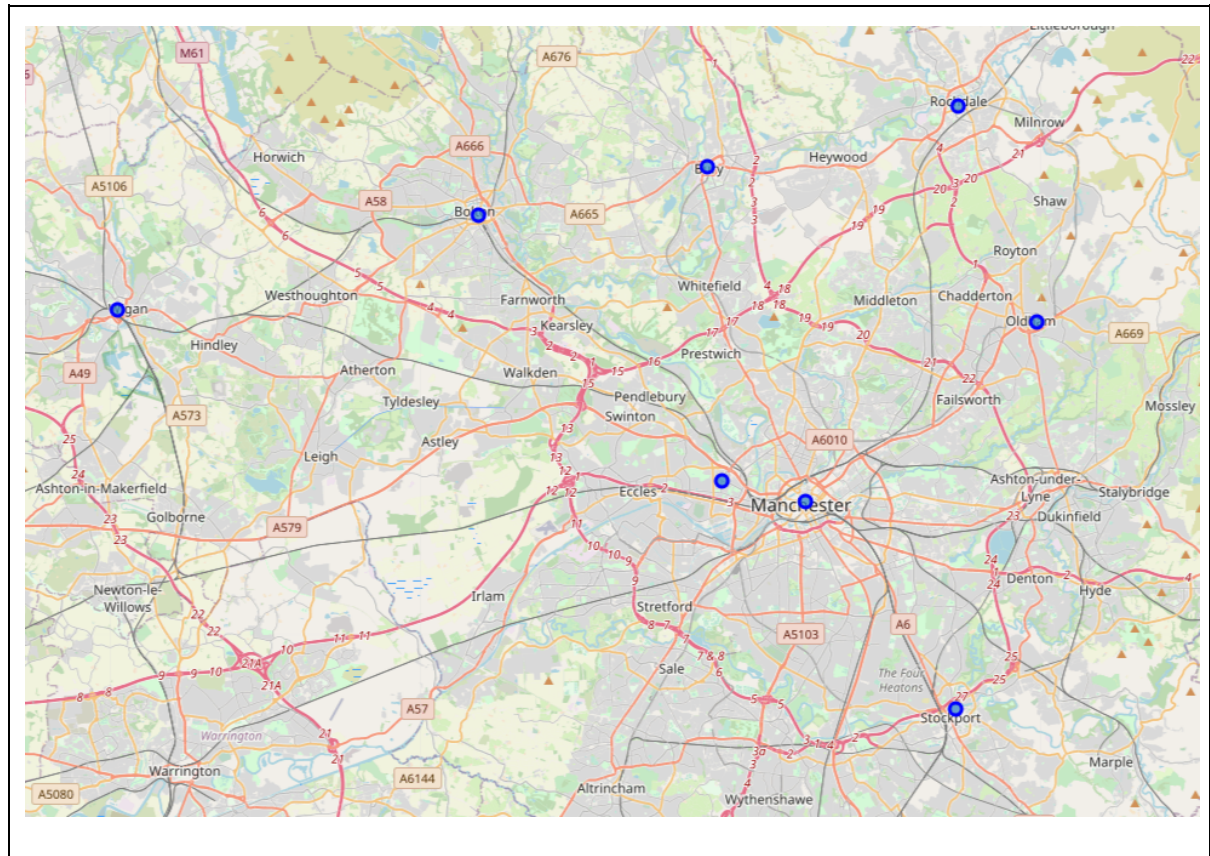
```
[ ] import requests
import lxml
from bs4 import BeautifulSoup
```

```
# Obtaining data from Wikipedia
source = requests.get('https://en.wikipedia.org/wiki/List_of_Greater_Manchester_settlements_by_population').text
soup = BeautifulSoup(source, 'lxml')
soup.encode("utf-8-sig")
```

# ↕	Settlement ↕	Population ^[1]	
		Census 2001 ↕	Census 2011 ↕
1	Manchester [‡] (see notes)	405,300	510,746
2	Bolton	182,980	194,189
3	Sale	122,990	134,022
4	Rochdale	102,080	107,926
5	Stockport	106,170	105,878
6	Salford	88,460	103,886
7	Wigan	98,480	103,608
8	Oldham	103,544	96,555
9	Bury	74,540	77,211
10	Leigh/Atherton	65,980	70,542

After performing some Data Wrangling the data for the boroughs for the geo-coordinates and demographics will be connected, this will then allow us to now pinpoint Manchester's boroughs on map.

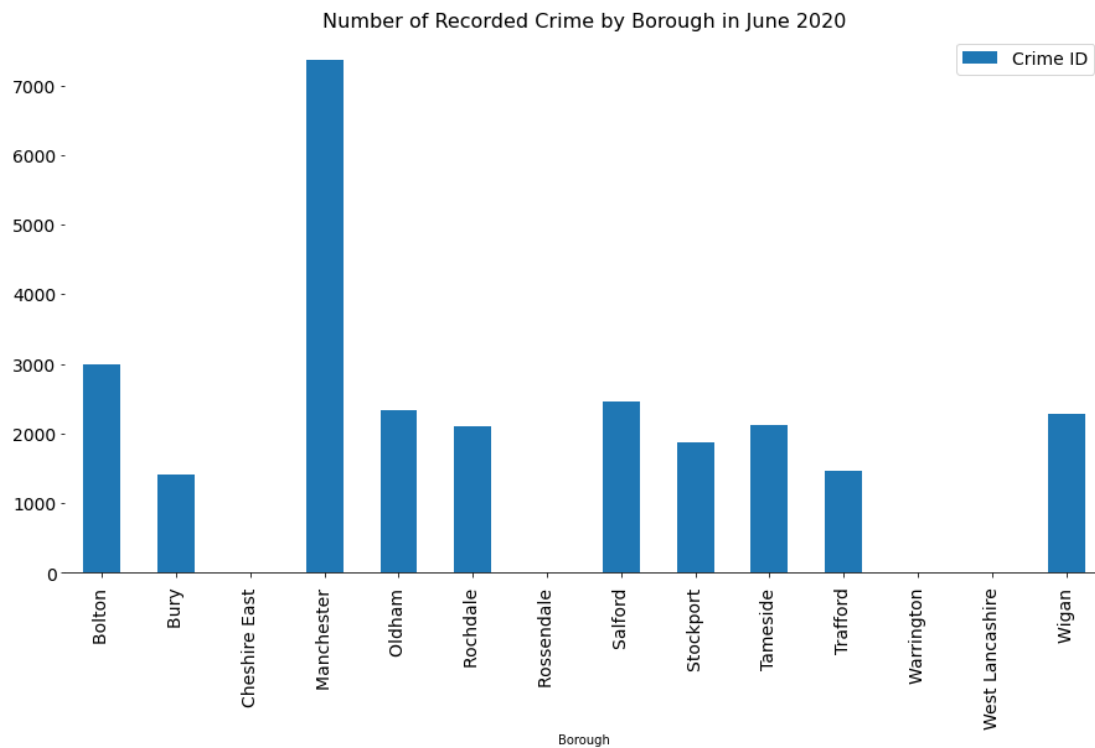
Folium- Python visualization library would be used to visualize the neighborhoods cluster distribution of Manchester. An example as shown below:



Crime and Safety

In order to have an understanding of the safety levels of each borough we shall extract the recorded crime rates in Greater Manchester, this information will be extracted from the UK police database - <https://data.police.uk/> and this information will be in the form of a csv file which will be converted into a dataframe using the Pandas library.

We will then later convert this information into a visualisation of the information either using a Chart or a folium map showing the clusters of crime in their respective areas.



Number of Restaurants

We will use the Foursquare API gather information in relation to venues and places in and around the Greater Manchester boroughs. Foursquare is a social location service that allows users to explore the world around them.

Foursquare has other information such as reviews and photos giving a more layered view of the venues in a certain area. By gathering this information we will be able to get a more rounded view of what amenities and venues are available in a certain borough such as restaurants and cinemas.

HTTP requests would be made to this Foursquare API server using the location of Manchester's boroughs neighborhoods to pull the location information (Latitude and Longitude).

Foursquare API search feature would be enabled to collect the nearby places of the neighborhoods. The number of places per neighborhood parameter would be set to 100 and the radius parameter would be set to 2000m as the boroughs are quite large.

Extracting this information can give us a broad view on whether a particular neighbourhood is primarily residential or more of a mixed use borough with restaurants.

We are able to extract a list of venues in the Boroughs and this can be further filtered down to the number of Restaurants per Borough.

```
#Get top 100 venues in 3000m radius of the center of each Borough
LIMIT = 100
venues = getNearbyVenues(names=McrCoord['Borough'],
                          latitudes=McrCoord['Latitude'],
                          longitudes=McrCoord['Longitude']
                          )
```

```
Manchester
Bolton
Rochdale
Stockport
Salford
Wigan
Oldham
Bury
```

```
print(venues.shape)
venues.head()
```

```
(639, 7)
```

	BoroughName	Borough Latitude	Borough Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Manchester	53.480759	-2.242631	Café Beermoth	53.481359	-2.241902	Bar
1	Manchester	53.480759	-2.242631	Royal Exchange Theatre	53.482555	-2.244770	Theater
2	Manchester	53.480759	-2.242631	Manchester Art Gallery	53.478882	-2.241817	Art Gallery
3	Manchester	53.480759	-2.242631	Northern Soul • Grilled Cheese	53.483074	-2.238303	Sandwich Place
4	Manchester	53.480759	-2.242631	Hotel Gotham	53.480767	-2.242235	Hotel

House within Budget

We would also need to be realistic and consider that the cost of a property must be taken into consideration, as the best neighbourhoods often come at a price. There is a fine balance between a reasonably priced home and a nice neighbourhood to live in.

We shall therefore extract the median household prices paid in Greater Manchester from a csv extracted from <https://mappinggm.org.uk/metadata/> which has recorded prices paid in all areas of Greater Manchester across a number of years, however we will use the latest prices paid in 2020 to analyse.

This information will be converted into a dataframe using Pandas and visualised in a graph using Matplotlib.

	Borough	Population	Latitude	Longitude	Median-PricePaid	Mean-PricePaid
0	Manchester	510,746	53.480759	-2.242631	165000.0	226208.930079
1	Bolton	194,189	53.576866	-2.428219	128000.0	165499.729730
2	Rochdale	107,926	53.613600	-2.156100	125000.0	148055.183024
3	Stockport	105,878	53.410600	-2.157500	230050.0	282065.896332
4	Salford	103,886	53.487500	-2.290100	150010.0	193393.930185
5	Wigan	103,608	53.545100	-2.632500	125000.0	172105.265248
6	Oldham	96,555	53.540932	-2.111366	125000.0	146423.039216
7	Bury	77,211	53.593460	-2.298540	160000.0	181432.515850

Schools

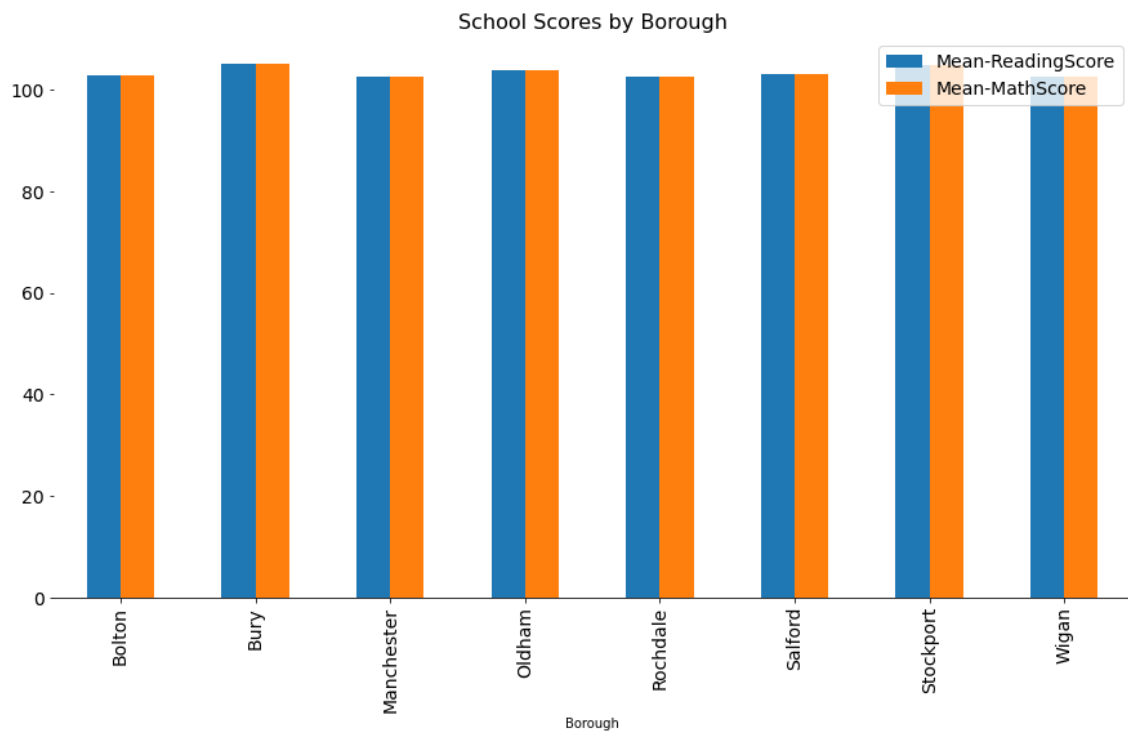
Finding a good school for the Willis family is also a key requirement. The information source that we will be dealing with is a csv file downloaded from the gov.uk website:

- <https://www.compare-school-performance.service.gov.uk/download-data?currentstep=year®iontype=beforeStep&la=&downloadYear=2018-2019>

This information will then be converted into a dataframe and merged with the other dataframes, in order to associate the schools to a borough coordinate.

	Borough	Mean-ReadingScore	Mean-MathScore	Population	Latitude	Longitude	Median-PricePaid	Mean-PricePaid
0	Bolton	102.720000	102.720000	194,189	53.576866	-2.428219	128000.0	165499.729730
1	Bury	105.200000	105.200000	77,211	53.593460	-2.298540	160000.0	181432.515850
2	Manchester	102.535836	102.535836	510,746	53.480759	-2.242631	165000.0	226208.930079
3	Oldham	103.812500	103.812500	96,555	53.540932	-2.111366	125000.0	146423.039216
4	Rochdale	102.534884	102.534884	107,926	53.613600	-2.156100	125000.0	148055.183024
5	Salford	103.172414	103.172414	103,886	53.487500	-2.290100	150010.0	193393.930185
6	Stockport	104.800000	104.800000	105,878	53.410600	-2.157500	230050.0	282065.896332
7	Wigan	102.594595	102.594595	103,608	53.545100	-2.632500	125000.0	172105.265248

We can then also start making score comparisons between the schools of various boroughs.



Analysis

Once all the dataframes have been set up an extensive comparative analysis of the boroughs would be carried out to derive the desirable insights from the outcomes using python's scientific libraries Pandas, NumPy and Scikit-learn.

The primary display formats that will be utilised are folium maps to visualise the collated information on a map of Manchester and the supported using matplotlib charts to display information that could be seen to be better represented as.

