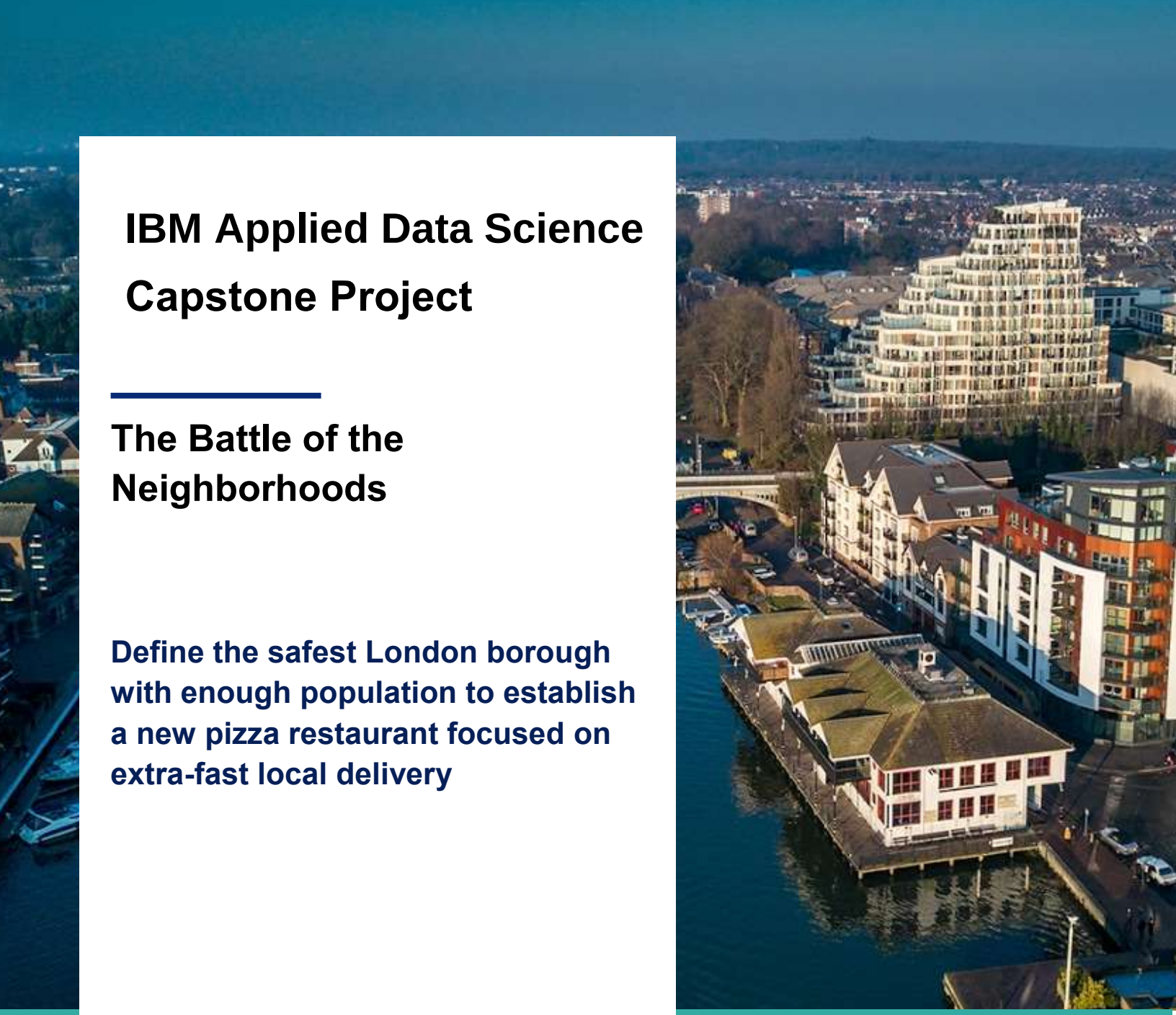


IBM Applied Data Science Capstone Project

The Battle of the Neighborhoods

Define the safest London borough with enough population to establish a new pizza restaurant focused on extra-fast local delivery

MARCH 3

Author: Daryna Manzheliivska

IBM Applied Data Science Specialization Capstone Project

The Battle of the Neighborhoods

Date: 3 March 2021

Author: Daryna Manzheliivska

A. Introduction

A.1. Description of the Business Problem to be solved

My client, an owner of a large Italian restaurants, is experiencing huge decline of business due to Covid-19 and quarantine measures. To escape bankruptcy my client decided to promptly reform his business from large restaurants to a network of smaller pizza places focused on extra-fast local delivery.

To be on the safe side and to protect delivery workers from robbery and harassment, my client wants to establish a small pizza restaurant in the safest possible neighborhood of London, where he wants to move his family and his business.

Knowing little to nothing about London and its neighborhoods, my client decided to ask for a professional data science advice in this regard.

Business problem to be solved:

define the safest London borough but with enough population where it will be profitable to establish a new pizza restaurant focused on extra-fast local delivery, without competition with other Pizza places in chosen locality.

This data science project will consist of four stages of preparing and analysing data:

1. Analyze crime level in London boroughs.
2. Analyze population in each borough
3. Analyze combined crime to population ratio and define the most suitable borough.
4. Explore venues in chosen borough, analyze competition rating.

To analyze each stage, I will need corresponding data.

A.2. Data

• To identify the safest areas neighborhood we will use London crime data, which we downloaded from https://data.london.gov.uk/dataset/recorded_crime_summary.

• We will get data with London boroughs population, latitude and longitude from Wikipedia page: https://en.wikipedia.org/wiki/List_of_London_boroughs

• All data related to locations of Pizza restaurants will be obtained via the Foursquare API utilized via the Request library in Python. This API provides location search, location sharing and details about a local business. Foursquare users can also use photos, tips and reviews in many productive ways to add value to the results of data science projects.

• Work Flow: HTTP requests would be made to this Foursquare API server using latitude and longitude of London boroughs to collect the information of the nearby venues of the chosen boroughs.

• Visualization: To visualize the neighborhoods cluster distribution of chosen London boroughs we will use Folium - Python visualization library that creates an interactive map.

• Extensive comparative analysis to derive insights from the datasets so that to define the safest neighborhood will be carried out using python's scientific libraries Pandas, NumPy and Scikit-learn.

A.3. Python Libraries used

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

A.4. Methodology (summary of analysis)

• Data will be collected from provided above public internet resources, cleaned and processed into a dataframe.

• Foursquare data will be used to locate all venues and then filtered by venue category.

• Final data science advice will be provided based on combined analysis of datasets and venue data.

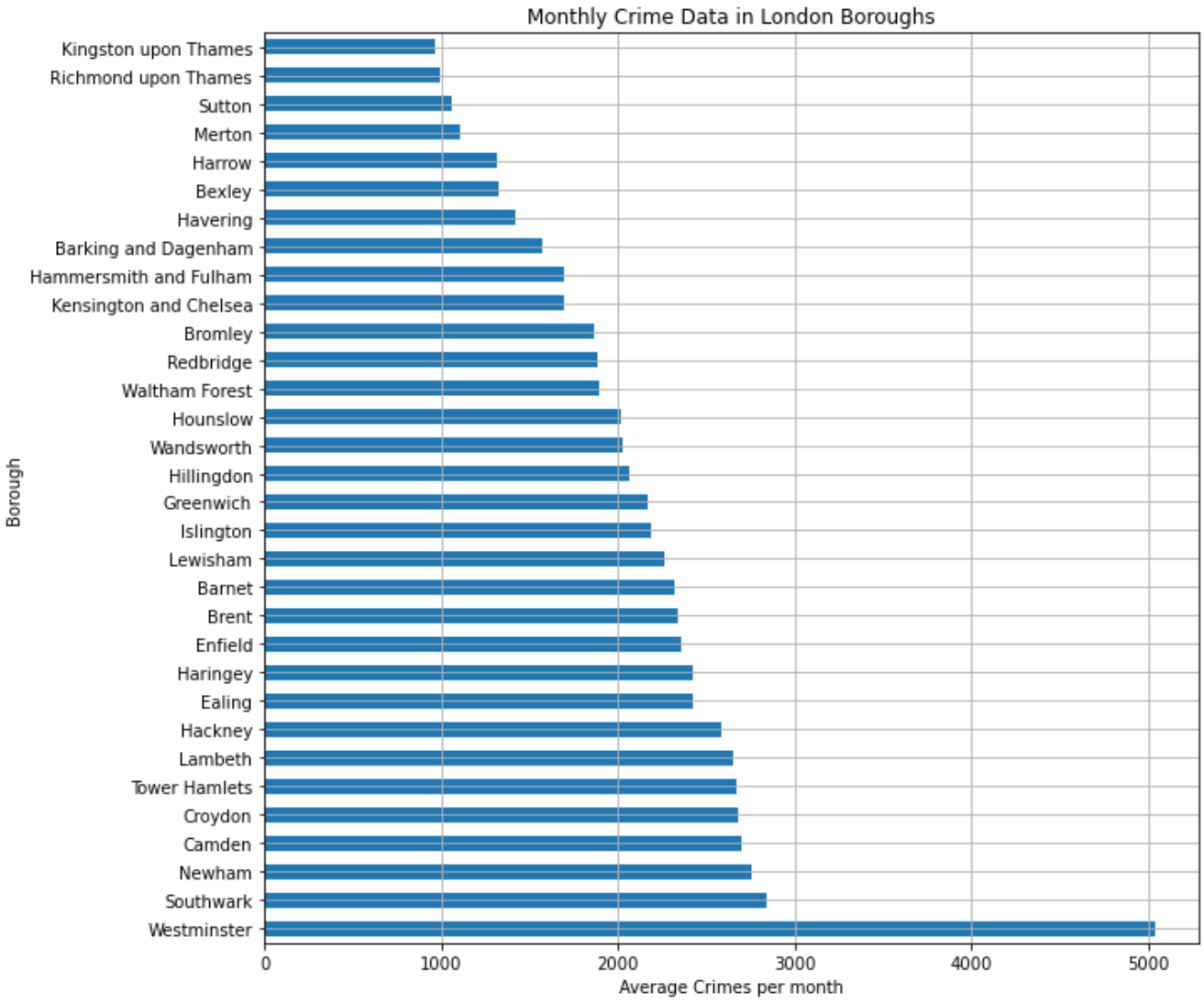
B. Data Analysis

B1. Part 1 of data analysis

In the first part of data analysis I will scrape London crime records for the past two years by boroughs from London Datastore.

url: https://data.london.gov.uk/dataset/recorded_crime_summary

There were 1546 rows and 26 columns in a raw dataset. After extensive data wrangling and cleansing I acquired a dataframe with the crime data. Using matplotlib I prepared a visualization of processed data:



I have received an initial data on borough safety. But to better understand these criteria I will need a ration of crime rates per population, since total amount of crimes is not enough for my analysis. Suitable neighborhood needs to be populated enough for the local pizza delivery service to be profitable.

B2. Part 2 of data analysis

To obtain population data I will scrape Wikipedia page with the list of London boroughs.
url: https://en.wikipedia.org/wiki/List_of_London_boroughs

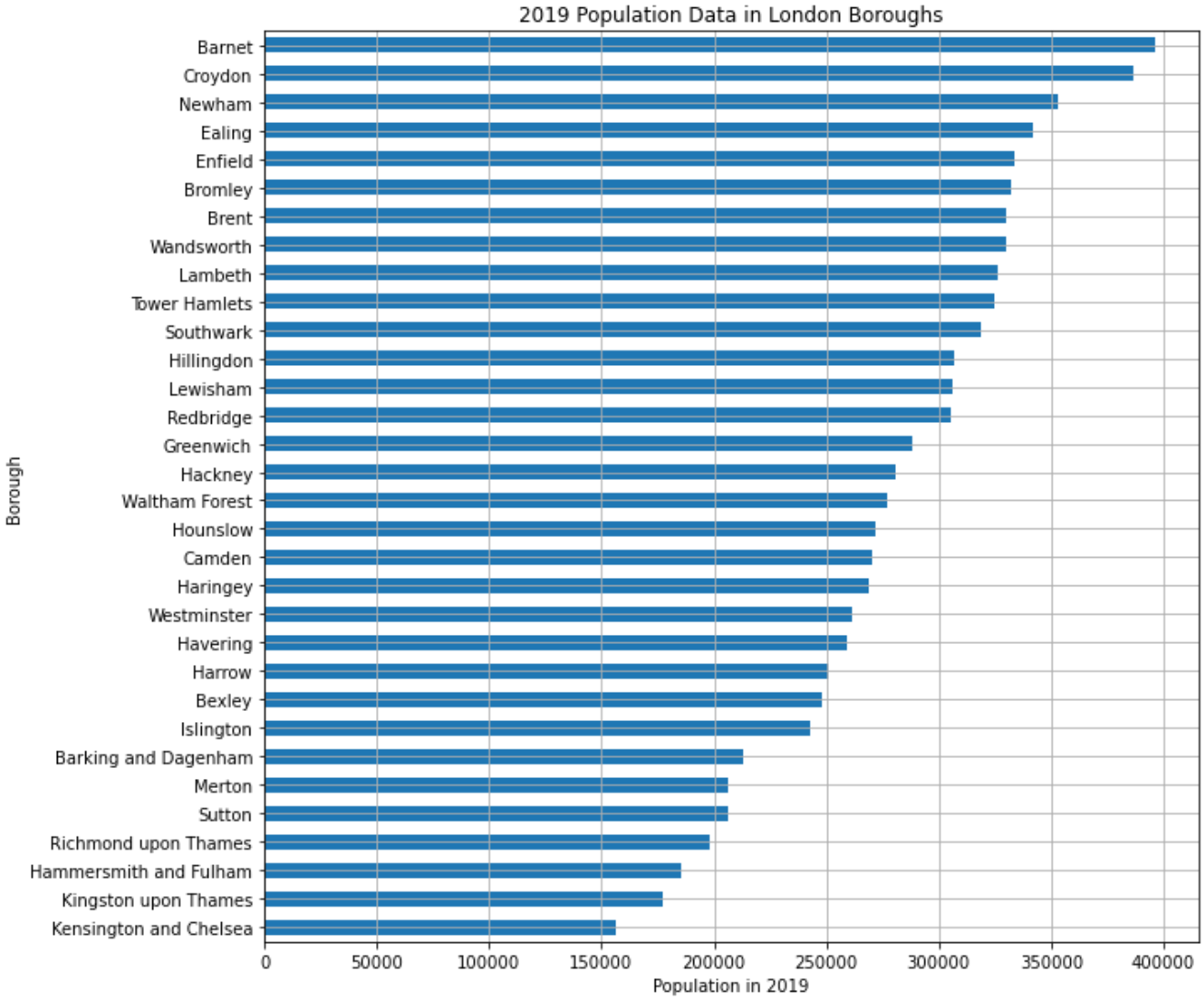
This is how raw dataset from the Wikipedia page looks like:

List of boroughs and local authorities

Borough	Inner	Status	Local authority	Political control	Headquarters	Area (sq mi)	Population (2019 est)[1]	Co-ordinates	
Barking and Dagenham[note 1]			Barking and Dagenham London Borough Council	Labour	Town Hall, 1 Town Square	13.93	212,906	51.5607°N 0.1557°E	25
Barnet			Barnet London Borough Council	Conservative	Barnet House, 2 Bristol Avenue, Colindale	33.49	396,896	51.6252°N 0.1517°W	31
Bexley			Bexley London Borough Council	Conservative	Civic Offices, 2 Watling Street	23.38	248,287	51.4549°N 0.1505°E	23
Brent			Brent London Borough Council	Labour	Brent Civic Centre, Engineers Way	16.70	329,771	51.5588°N 0.2817°W	12
Bromley			Bromley London Borough Council	Conservative	Civic Centre, Stockwell Close	57.97	332,336	51.4039°N 0.0198°E	20
Camden	✓		Camden London Borough Council	Labour	Camden Town Hall, Judd Street	8.40	270,029	51.5290°N 0.1255°W	11
Croydon			Croydon London Borough Council	Labour	Bernard Weatherill House, Mint Walk	33.41	386,710	51.3714°N 0.0977°W	19
Ealing			Ealing London Borough Council	Labour	Perceval House, 14-16 Uxbridge Road	21.44	341,806	51.5130°N 0.3089°W	13
Enfield			Enfield London Borough Council	Labour	Civic Centre, Silver Street	31.74	333,794	51.6538°N 0.0799°W	30
Greenwich[note 2]	✓[note 3]	Royal	Greenwich London Borough Council	Labour	Woolwich Town Hall, Wellington Street	18.28	287,942	51.4892°N 0.0648°E	22
Hackney	✓		Hackney London Borough Council	Labour	Hackney Town Hall, Mare Street	7.36	281,120	51.5450°N 0.0553°W	9
Hammersmith and Fulham[note 4]	✓		Hammersmith and Fulham London Borough Council	Labour	Town Hall, King Street	6.33	185,143	51.4927°N 0.2339°W	4
Haringey	[note 3]		Haringey London Borough Council	Labour	Civic Centre, High Road	11.42	268,847	51.6000°N 0.1119°W	29
Harrow			Harrow London Borough Council	Labour	Civic Centre, Station Road	19.48	251,160	51.5898°N 0.3346°W	32
Havering			Havering London Borough Council	Conservative (council NOC)	Town Hall, Main Road	43.35	259,552	51.5812°N 0.1837°E	24
Hillingdon			Hillingdon London Borough Council	Conservative	Civic Centre, High Street	44.67	306,870	51.5441°N 0.4760°W	33
Hounslow			Hounslow London Borough Council	Labour	Hounslow House, 7 Bath Road	22.61	271,979	51.4746°N	14

From this raw dataset for my further data analysis I will only need population and coordinates. Population – for a more precise understanding of crime ration. Coordinates – to pull from Foursquare API information on venues in the selected suitable borough.

Again, after extensive data wrangling and cleansing I received a sorted dataset with population data. Using matplotlib I visualized this dataset from top populated boroughs down to the least populated.



At the end of the second stage I obtained two cleansed datasets:

1) with initial crime data analysis of London boroughs, and

2) with London boroughs total population (as of 2019).

B3. Part 3 of data analysis

At this stage I merged two cleared datasets into one using pd.merge method.

First five rows of the new combined dataset looked like this:

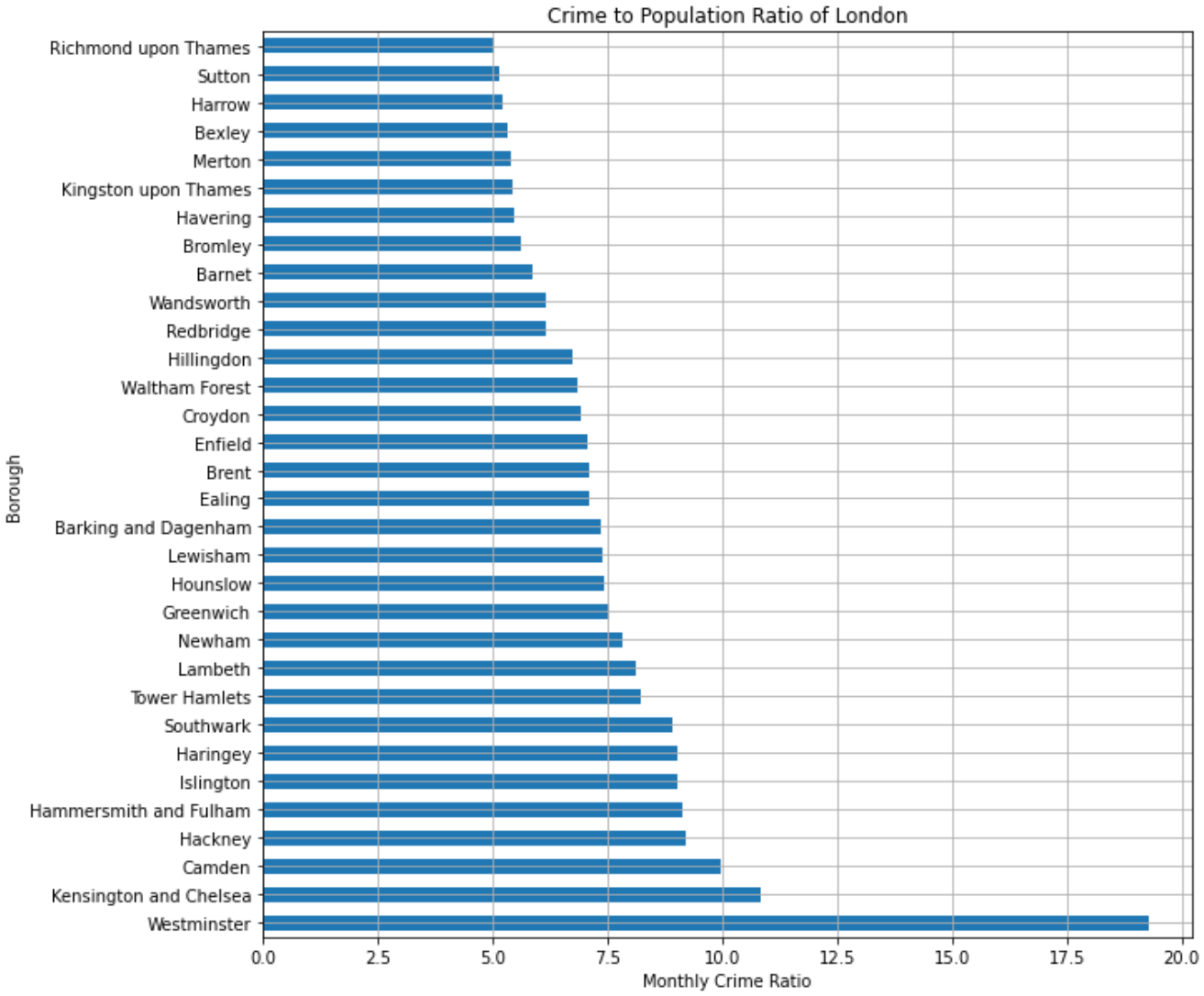
	Borough	Avg crimes per month	2019 Population	Latitude	Longitude
0	Kingston upon Thames	967.833333	177507	51.4096	-0.306262
1	Richmond upon Thames	996.041667	198019	51.4406	-0.307639
2	Sutton	1064.708333	206349	51.3656	-0.1963
3	Merton	1112.791667	206548	51.4109	-0.188097
4	Harrow	1313.083333	251160	51.5968	-0.337316

But my aim was to define a more precise understanding of crime situation per each borough. For this, I have added an additional column with crime ratio obtained by division of average crimes per month by total population for each borough.

After this data modification my new dataset looked like this:

	Borough	Avg crimes per month	2019 Population	Latitude	Longitude	Crime Ratio
0	Kingston upon Thames	967.833333	177507	51.4096	-0.306262	5.452367
1	Richmond upon Thames	996.041667	198019	51.4406	-0.307639	5.030031
2	Sutton	1064.708333	206349	51.3656	-0.1963	5.159746
3	Merton	1112.791667	206548	51.4109	-0.188097	5.387569
4	Harrow	1313.083333	251160	51.5968	-0.337316	5.228075

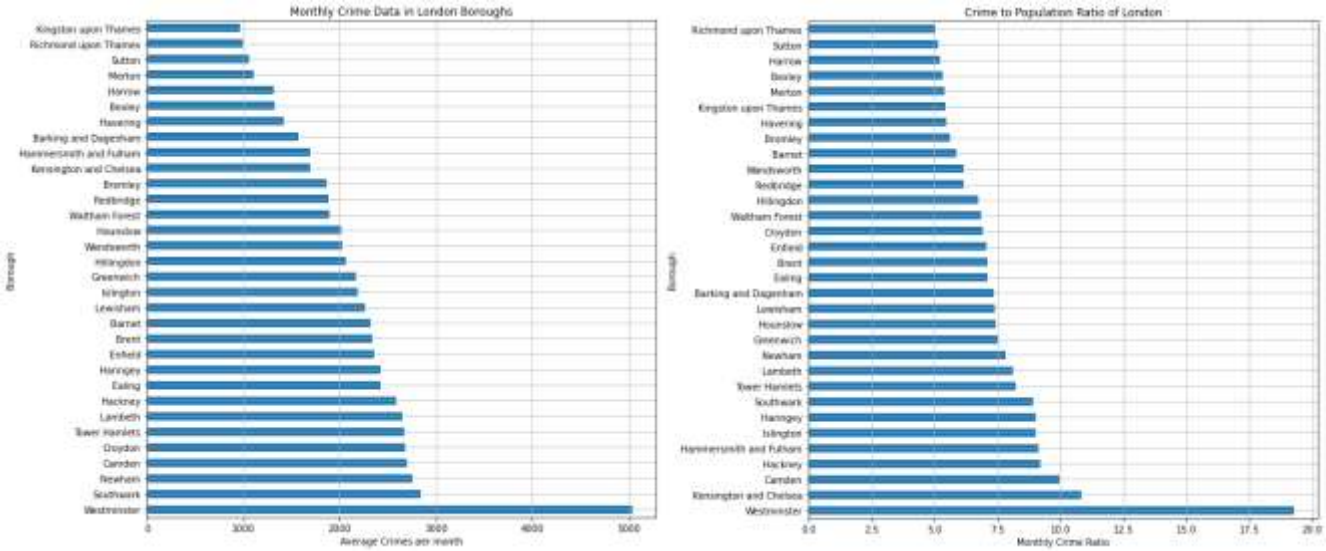
Next I sorted updated dataset with crime rates and using matplotlib I visualized crime ratio for all boroughs:



After my 1st data analysis stage the “safest” borough by total crimes was Kingston upon Thames. But after stage 3 of data analysis I got a more precise data, and now Richmond on Thames is the safest borough.

There is a difference between two datasets. Decision to get the third stage of analysis was correct. A more precise data is always a better choice for a data scientist, since we must always focus on providing prediction as precise as possible.

Compare two datasets: for total crimes and for crime ration:

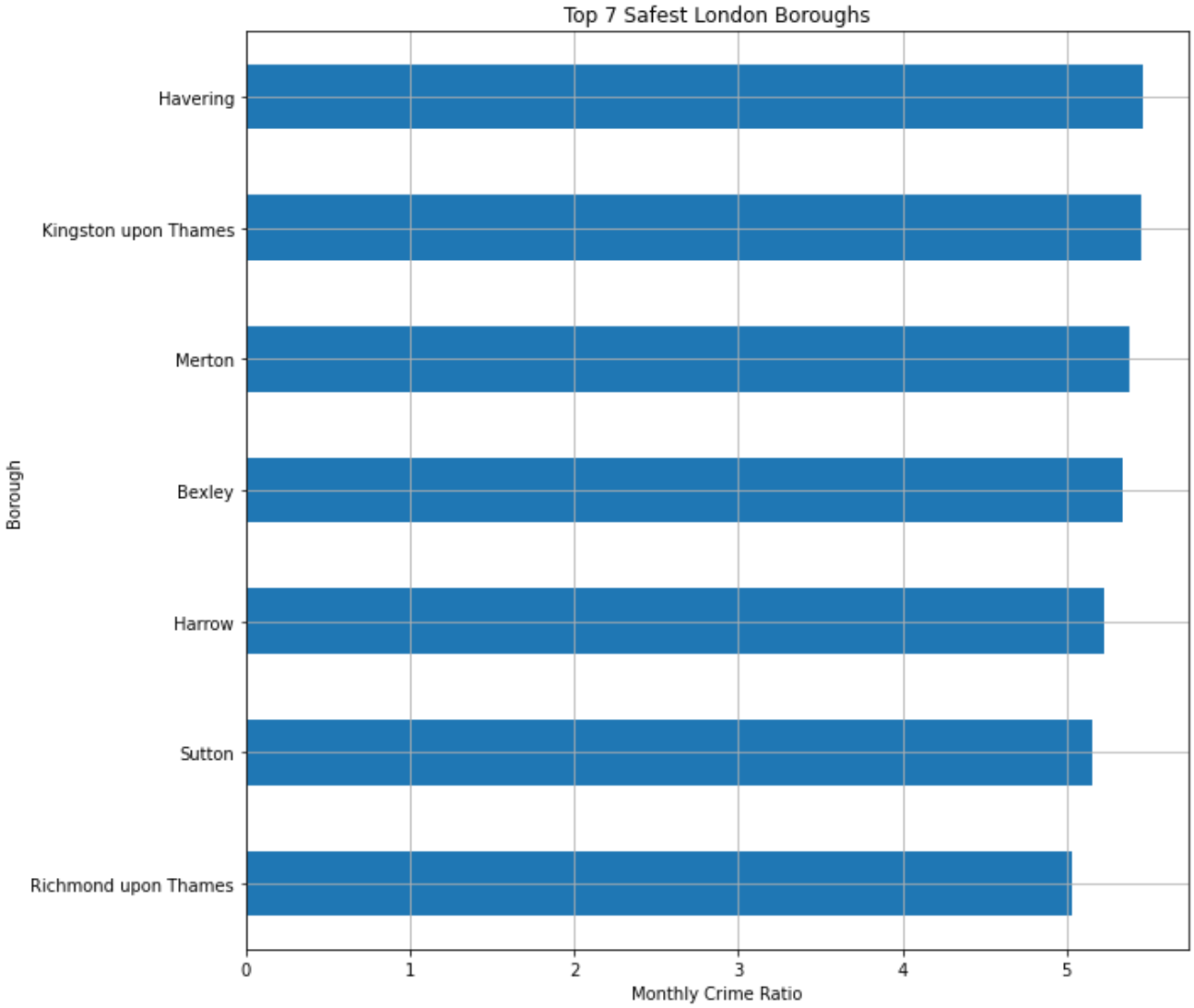


Since my client urged me to define the safest boroughs (so that to be o the safe side and to protect his delivery workers from robbery or offence), I will further focus only top 7 safest areas.

For this I created a new dataframe, containing only top 7 safest boroughs:

	Borough	Avg crimes per month	2019 Population	Latitude	Longitude	Crime Ratio
1	Richmond upon Thames	996.041667	198019	51.4406	-0.307639	5.030031
2	Sutton	1064.708333	206349	51.3656	-0.1963	5.159746
4	Harrow	1313.083333	251160	51.5968	-0.337316	5.228075
5	Bexley	1325.916667	248287	51.4416	0.150488	5.340258
3	Merton	1112.791667	206548	51.4109	-0.188097	5.387569
0	Kingston upon Thames	967.833333	177507	51.4096	-0.306262	5.452367
6	Havering	1419.833333	259552	51.5444	-0.144307	5.470323

And visualized the new dataset using matplotlib:



At this point I decided to plot top 7 safest boroughs on London map using Folium.

Longitude and latitude for London I defined by the following geolocator Nonimatim code:

```
address = 'London, UK'
geolocator = Nominatim(user_agent="explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('Latitude and longitude values of London are {}, {}.'.format(latitude, longitude))
```

Map of London with the top 7 safest boroughs (defined by crime ratio per population).

At this point I found an exciting insight:

two safe boroughs were very close to each other! These boroughs were Richmond and Kingston – the two most eastern dots on the map.

I decided to check if combined population and average crime ratio for this combined zone will still satisfy crime ratio of the 7 top safest boroughs. I created a new dataset just with Richmond and Kingston:

	Borough	Avg crimes per month	2019 Population	Latitude	Longitude	Crime Ratio
1	Richmond upon Thames	996.041667	198019	51.440553	-0.307639	5.030031
0	Kingston upon Thames	967.833333	177507	51.409627	-0.306262	5.452367

Combined Population: 375526
Combined Avg Crime: 1963.875
Combined Crime Ratio: 5.229664523894485

That's fantastic! Thanks to thorough analysis and visualization I have found a large populated area but with the low crime rates. This might be our sweet spot for a small pizza restaurant focused on extra-fast local delivery.

Now I needed to check the amount of pizza restaurants in this area.

C. Exploring Selected Neighborhood

C1. Getting Foursquare credentials

At this stage I have set up connection to Foursquare API using my unique client ID and client secret codes, as well as version (the date of API request).

C2. Exploring Kingston and Richmond

Using Kingston on Thames latitude and longitude coordinates I pulled a request for 100 closet venues within a radius of 500 meters. Obtained results from the Foursquare API I saved into a .json file.

To extract category of venues I used a function, next I cleared json and structured obtained info into a dataframe.

There were total 61 unique categories of venues in this area.

C3. Analyzing venues

To find out most popular (and competitive) venues I created a new onehot dataframe, which contained 104 rows (locations) and 62 columns (different venues). After grouping all locations by borough (i.e. Kingston and Richmond), I obtained a clean dataset with two rows (boroughs) and 62 columns (different venues).

The next step was to define top 10 most common venues:

	Borough	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	Kingston upon Thames	Coffee Shop	Pub	Café	Italian Restaurant	Clothing Store	Hotel	Bakery	Sushi Restaurant	Department Store	Sandwich Place
1	Richmond upon Thames	Trail	German Restaurant	Garden Center	Bus Station	Video Game Store	Cosmetics Shop	Gift Shop	Gastropub	French Restaurant	Fast Food Restaurant

After examining this sorted dataset, I discovered that though there was an Italian restaurant on the 4th place of most common venues in Kingston, there were **no Pizza places** focused on delivery services in both boroughs among top 10 common venues!

D. Results and Discussion

Upon 3 analysis I discovered the best area based on the criteria I received from my client: safety and enough populated.

To get a better understanding of this criteria I combined two initial datasets into third with a more precise crime per population ratio.

I need to mention that this last dataset differed from the general crime rates per borough.

After plotting top 7 safest boroughs on map I discovered an **excellent insight**: two safe boroughs were located super-close to each other: mainly Kingston and Richmond.

After analyzing venues data though Foursquare API for these to boroughs I discovered that there were no Pizza places among the top 10 most common venues.

D. Conclusion

From my analysis I discovered 7 safest boroughs in London as this was the primarily provision of my client.

Next I have discovered a promising insight: two safe boroughs were very close to each other, giving an excellent opportunity to establish pizza restaurant with super-fast local delivery service in this area.

Moreover, since there were no Pizza delivery places among the top 10 most common venues in this area, me client would only compete with Italian restaurants, taking 4th place among top 10 most common venues (yet not really a competitor for our client, since most probably they do not focus on super-fast delivery of pizza (up to 20 minutes).

Expected further clustering was not necessary since both boroughs were located super-close to each other and amount of pizza delivery places satisfied our data analysis condition.