Making business decisions with public data & machine learning, a case study

Machine learning is probably one of the hottest buzz words around. Every single business seems to want to use machine learning and artificial intelligence to optimise their processes… but where do they start? Luckily, it's quite simple and cheap to start using machine learning! That's what I'll be proving in this blogpost, where I'll walk you through a possible use case.

Let's start by defining the question

You can't conduct any research without first defining a clear question. In reality, this is often done through workshops or meetings with the management. In this situation, let's say our client is a young, Belgian entrepreneur. His dream is to traditional Belgian food to New York City. But where should he open his restaurant? New York City is huge!

Determining the optimal location to start a business or open a new location is a question we're perfectly able to answer with the power of machine learning and some free, publicly available data. Let's start our search for the best location to open a Belgian restaurant in New York City!

How will we handle this analysis?

If we have to gather location data for the entirety of New York City, we'll end up with an enormous data set. While this isn't necessarily a problem, it would make things more complicated than they need to be. Why don't we split up our analysis into 2 parts?

First of all, let's narrow it down to the optimal borough to start a restaurant in. The boroughs of New York City still contain their own, diverse neighbourhoods. Once we've found the best borough, we drill down.

If you want to conduct simple data analysis, I highly recommend you have a look and Python's pandas library, as this is what I used over the course of this case study. In pandas, there's something called a 'dataframe'. Dataframes are, for all intents and purposes, tables.

Looking for our data

On a borough-level

For our ideal borough, we'd love to have a place with a sizeable population, a high GDP per capita and preferably a low crime rate. Of course, there are more data points we can look at, but let's stick to these for simplicity's sake.

Population and GDP data by borough can actually be consulted on Wikipedia! The article 'Boroughs of New York City' even presents them in a nice table. As for crime data, most police departments or governments actually publish crime statistics online. The NYPD is no different! On their website, I found an overview of crime by borough, including a breakdown across multiple types of crime.

On a neighbourhood-level

An ideal neighbourhood would be a neighbourhood with plenty of restaurants of different cuisines, but nothing too similar to Belgium. This would be a place locals, tourists and foodies alike visit for a bite to eat and to experience something new. Of course, real estate can't be too expensive in this neighbourhood either, since our young entrepreneur is only just starting out.

I managed to find data of neighbourhoods with the average housing prices on a website called 'Streeteasy'. If you're curious, you can consult their data yourself right here. At the bottom of the page, you can download it all as a .csv file.

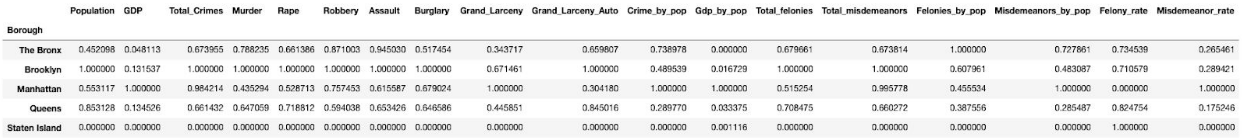
As for the number of restaurants and other business-related data, we're going to have to get technical. Foursquare's API can provide all the data we need for this analysis. If we query their API for venue names, venue categories and coordinates for max. 100 business in a radius of 1.25km of the neighbourhoods' centres, we have everything we need! A free Foursquare developer account gives you enough daily calls for this purpose. It won't cost you a single penny!

Preparing our borough-level data

Let's start building our dataset with our found data. The simplest way to do so would be through a Python library named 'Beautiful Soup'. This allows you to crawl a url, retrieve its HTML-structure and filter out the parts you don't want. With only a few lines of code, you can extract the table from the wikipedia article and turn it into a pandas dataframe.

As for the crime data, I simply merged the different Excel-files I found into one big file using Microsoft Excel. I then exported the data as a .csv and loaded it into a pandas dataframe with one simple command: df_crime = pd.read_csv(*{{filename}}*, sep=";", decimal=",").

After removing some unnecessary columns and rows, merging the two dataframes and calculating a few extra columns, we see the following table.



	Population	GDP	Total_Crimes	Murder	Rape	Robbery	Assault	Burglary	Grand_Larceny	Grand_Larceny_Auto	Crime_by_pop	Gdp_by_pop	Total_felonies	Total_misdemeanors	Felonies_by_pop	Misdemeanors_by_pop	Felony_rate	Misdemeanor_rate
Borough																		
The Bronx	0.452098	0.048113	0.673955	0.788235	0.661386	0.871003	0.945030	0.517454	0.343717	0.659807	0.738978	0.000000	0.679661	0.673814	1.000000	0.727861	0.734539	0.265461
Brooklyn	1.000000	0.131537	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.671461	1.000000	0.489539	0.016729	1.000000	1.000000	0.607961	0.483087	0.710579	0.289421
Manhattan	0.553117	1.000000	0.984214	0.435294	0.528713	0.757453	0.615587	0.679024	1.000000	0.304180	1.000000	1.000000	0.515254	0.995778	0.455534	1.000000	0.000000	1.000000
Queens	0.853128	0.134526	0.661432	0.647059	0.718812	0.594038	0.653426	0.646586	0.445851	0.845016	0.289770	0.033375	0.708475	0.660272	0.387556	0.285487	0.824754	0.175246
Staten Island	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.001116	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000

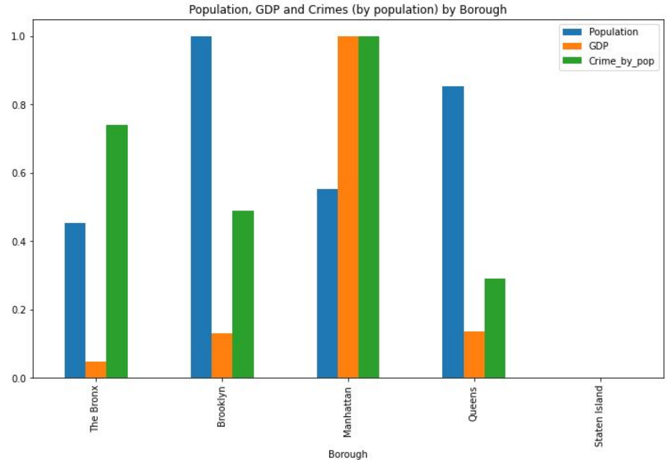
You might have noticed that the data for Staten Island has seemingly disappeared. That's due to a process called 'normalisation'. Normalisation is when you rescale your data so that every column uses the same scale. Otherwise it would be hard to graph a large number like 'population' alongside a small number like crime rate!

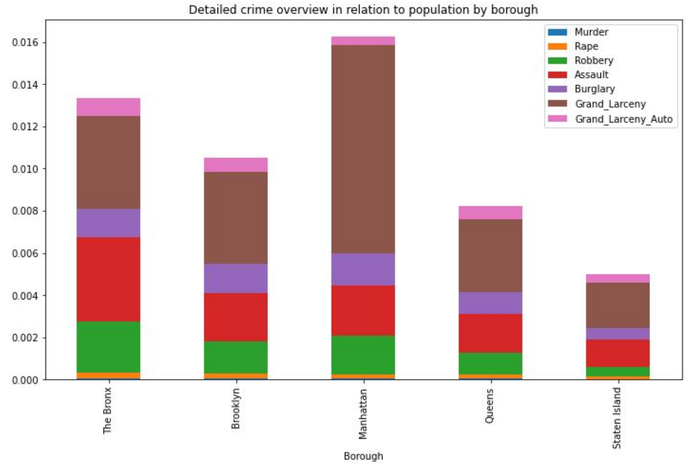
In this case I used min-max scaling, which means the minimum value of a column becomes 0, the maximum value becomes 1 and everything else is properly scaled in between those numbers. You can do this by using the following mathematical formula:

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

So what's our best borough?

A major advantage of using pandas is how easy it is to generate graphs based on the dataframes you've created. Some examples of the graphs I generated for this case study are:





Now what does this tell us? For starters, if we look at the crime rates, The Bronx and Manhattan are out. Staten Island is also a bad fit due to its low population and GDP. This just leaves Queens and Brooklyn. Due to both safety and GDP being in its favour, I would definitely recommend Queens over Brooklyn.

Preparing our neighbourhood-level data

We start by loading our housing data, as a csv, into our dataframe. However, if we want to run queries, we're going to need coordinates for each neighbourhood. This can be done by simply running the names we have through geopy's geolocator. Keep in mind that it's a great idea to further specify neighbourhoods with a city and/or country when you use the geolocator.

Now that I have my neighbourhoods and their locations, it's time to start making Foursquare API calls. The way this works, is that you create a query URL. This url contains specific types of data you want to request. When you go to this URL, it will return the data in a json format.

I used a python function to loop through my list of neighbourhoods and return the venues as a dataframe with the columns: neighbourhood, lat, lng, venue, venue latitude, venue longitude and venue category.

As a result, I could add some more interesting columns to my dataframe: the total number of restaurants retrieved, the total number of venues retrieved, the restaurant ratio and the total number of unique restaurant categories by neighbourhood.

Time to bust out the machine learning algorithm

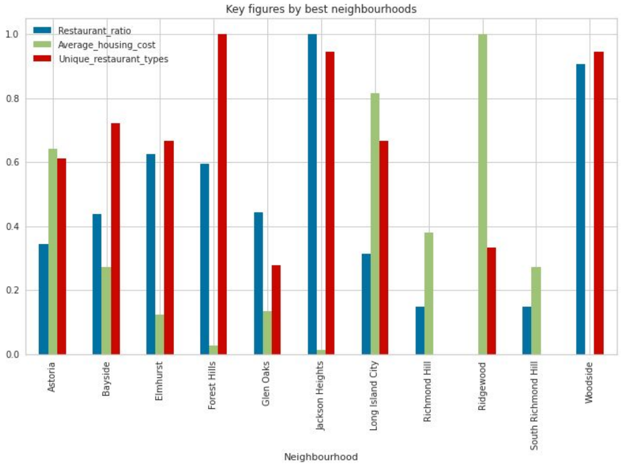
There has to be a trend in these neighbourhoods. The kmeans clustering algorithm is perfect for identifying trends and labelling data that was previously unlabelled. This makes the form of unsupervised learning a great fit.

Using the yellowbrick and scikitlearn libraries, I was able to divide all of those neighbourhoods over 9 different clusters in only a few lines of code. If you want a full look at the code behind this, make sure you have a look at my Jupyter Notebook on github. That cut my analysis time down significantly! Now we just need to identify the optimal cluster.

As it turns out, cluster 4 contained neighbourhoods with a high restaurant density and many different kinds of restaurants.

Analysing the optimal cluster

Now that we've been able to filter everything down to a cluster of 11 neighbourhoods, it's time to look at the specific data for those 11 neighbourhoods. This graph makes our decision a lot easier.

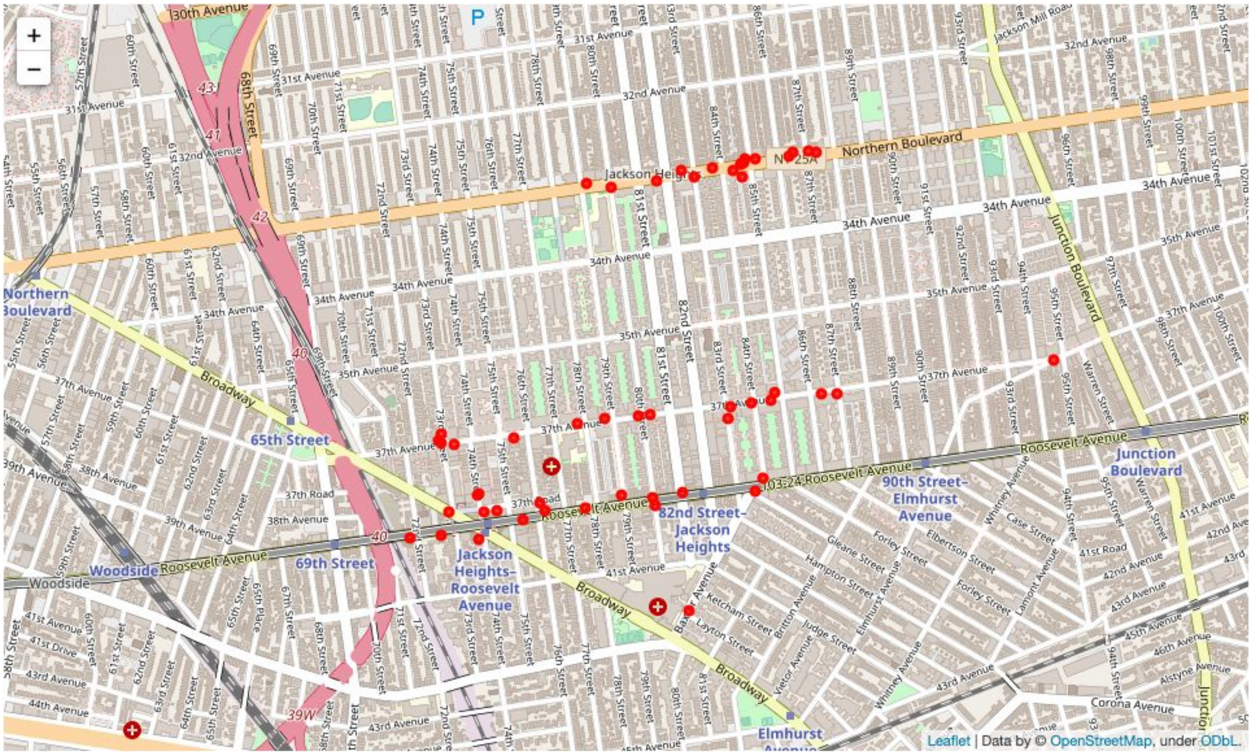


Two neighbourhoods stand out with their high restaurant density and wide variety of cuisines and low housing costs: Jackson Heights or Woodside.

If we dive deeper into the types of restaurants in each neighbourhood, we'll notice that Jackson Heights has mostly latino restaurants with some asian and european restaurants mixed in. Woodside, on the other hand, has mostly asian restaurants with some latino and european restaurants mixed in. As Woodside already seems to have a French restaurant, which is quite similar to a Belgian restaurant, Jackson Heights is the most suitable neighborhood.

Time to go even further beyond

Jackson Heights as a neighbourhood might be a good fit, but we need a good location in this neighbourhood. If we map out all the retrieved restaurants in Jackson Heights, we notice that they are all centred around 2 areas. We know the restaurants in those areas, so let's use geopy's geolocator.reverse to find the addresses that match the coordinates we have.


+
−
Northern Boulevard
Jackson Heights
Roosevelt Avenue
65th Street
69th Street
Woodside
Jackson Heights-Roosevelt Avenue
82nd Street-Jackson Heights
90th Street-Elmhurst Avenue
Junction Boulevard
Broadway
Elmhurst Avenue
Leaflet | Data by © OpenStreetMap, under ODbL.

So... What is the best location?

According to our data, the best areas to set up a Belgian restaurant in New York City would be in Jackson Heights. More specifically in the areas of 84-19, Northern Boulevard or 76-02, 37th Road. We've managed to answer this question without spending any money, with only publicly available data, some Python code and a Jupyter Notebook. The possibilities are endless for any business!