Comparing Neighborhoods in London and Brooklyn

By Claire Masse

IBM Data Science Professional Certificate Capstone Project Write-up, July 2020

Jupyter notebook with complete code available on my GitHub page.

I. Introduction:

I used to live in Brooklyn, New York and am hoping to move to London in the next few months. For my project, I decided to compare neighborhoods in Brooklyn and London based on location data (venues such as restaurants, bars, gyms, grocery stores…) to see what neighborhoods in the city I hope to move to most resemble the one I used to live in, Bedford-Stuyvesant, Brooklyn. Given how large and spread out both these cities are, I decided to cluster different neighborhoods based on similarities in venues and visualize them on a map. My hope was that I would be able to narrow down neighborhoods in London.

II. Data:

I used publicly available neighborhood data from Wikipedia for Brooklyn and London to extrapolate latitude and longitude values using the geopy library.

I used Foursquare data to get venue information for neighborhoods in both cities based on the geospatial data I mentioned above.

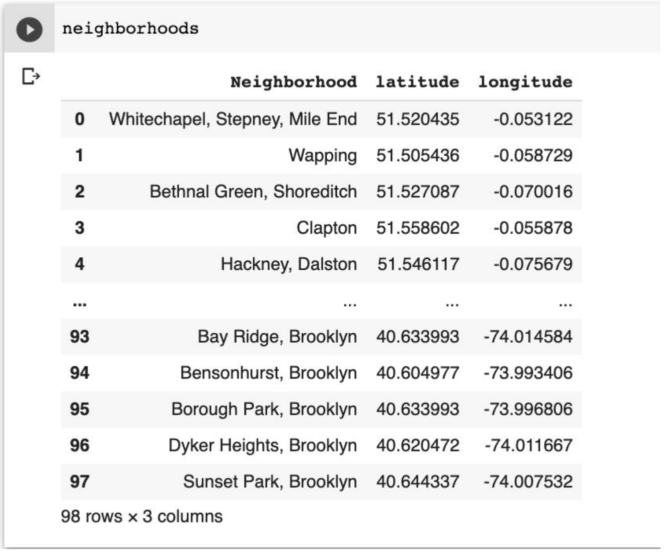
I used folium to create interactive maps of the cities.

III. Methodology:

1. Data Cleaning & Exploratory Data Analysis

I started with some data cleaning and exploratory data analysis. I had to manipulate the neighborhood data I linked above to get it into the format I wanted before getting latitude and longitude information.

I rearranged and dropped unnecessary data from the London data table to make it easier to work with. Once both tables were in a consistent format, I combined them and used geopy's geolocator function to get longitude and latitude data for each neighborhood. I ended up with the below table:



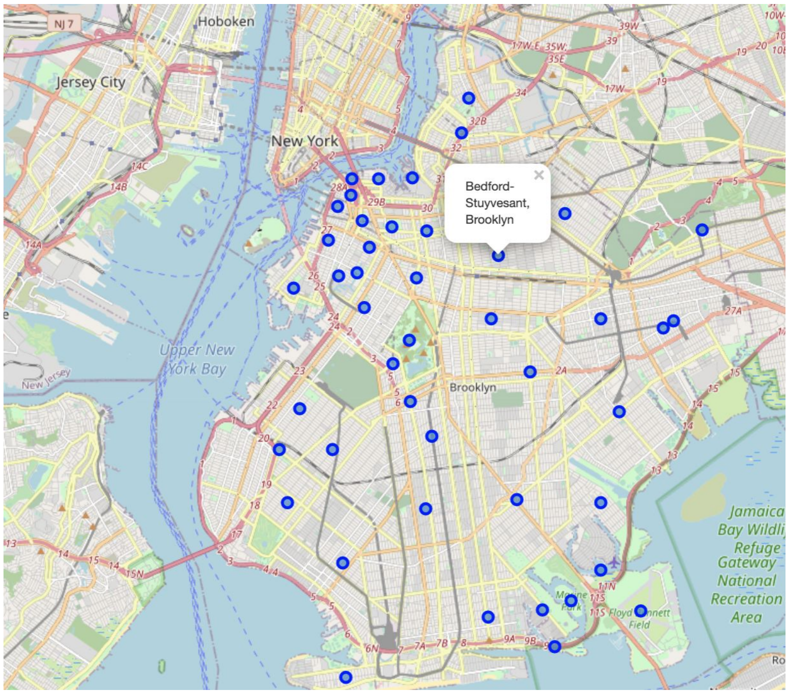
```
neighborhoods
```

	Neighborhood	latitude	longitude
0	Whitechapel, Stepney, Mile End	51.520435	-0.053122
1	Wapping	51.505436	-0.058729
2	Bethnal Green, Shoreditch	51.527087	-0.070016
3	Clapton	51.558602	-0.055878
4	Hackney, Dalston	51.546117	-0.075679
...	...	...	...
93	Bay Ridge, Brooklyn	40.633993	-74.014584
94	Bensonhurst, Brooklyn	40.604977	-73.993406
95	Borough Park, Brooklyn	40.633993	-73.996806
96	Dyker Heights, Brooklyn	40.620472	-74.011667
97	Sunset Park, Brooklyn	40.644337	-74.007532

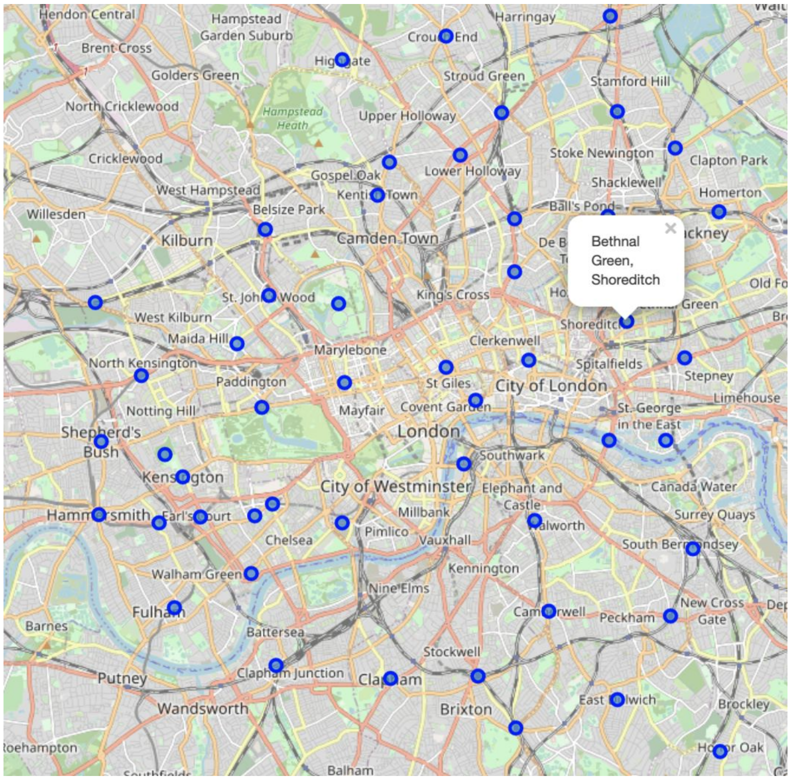
98 rows × 3 columns

Cleaned lat/lon data for London & Brooklyn by neighborhood

Then, I created maps using folium and added clickable markers for each neighborhood in the data table.

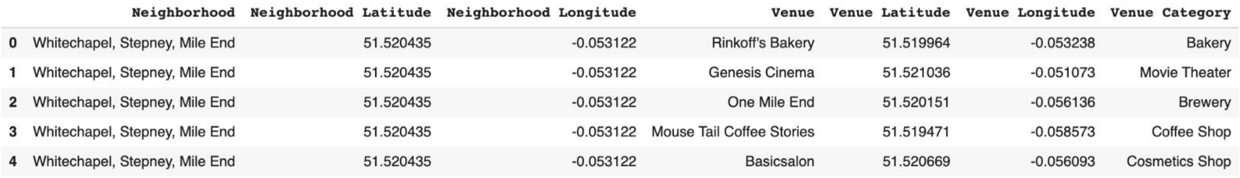


Folium map of Brooklyn Neighborhoods



Map of London

Once I had all of the neighborhoods defined and marked, I used the Foursquare API to get information on the venues in each of these neighborhoods. I created a function to get all the closest venues for each neighborhood (up to 50 per neighborhood). In total, I ended up with a dataframe of 3340 locations from 319 unique categories such as Bakery, Coffee Shop, Movie Theater, etc.

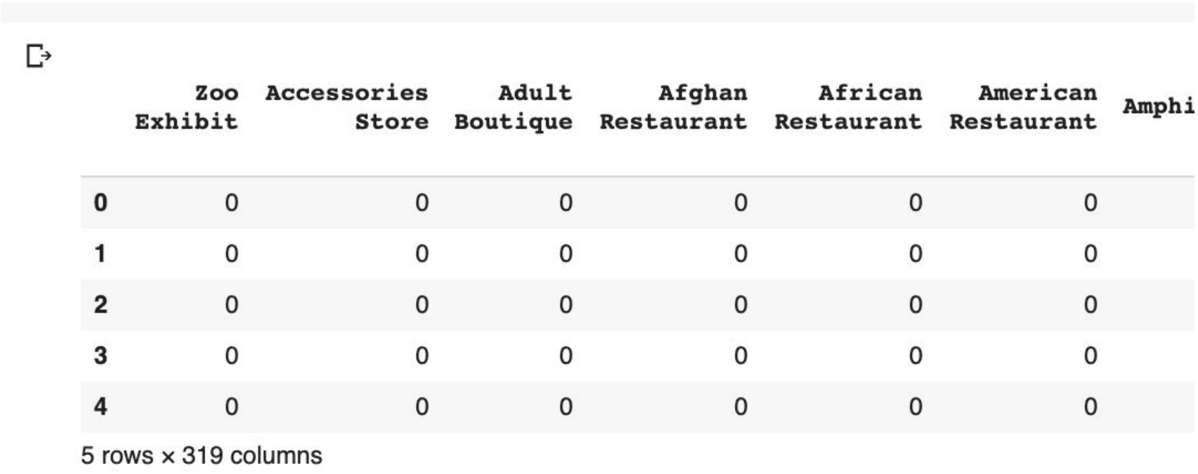


	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Whitechapel, Stepney, Mile End	51.520435	-0.053122	Rinkoff's Bakery	51.519964	-0.053238	Bakery
1	Whitechapel, Stepney, Mile End	51.520435	-0.053122	Genesis Cinema	51.521036	-0.051073	Movie Theater
2	Whitechapel, Stepney, Mile End	51.520435	-0.053122	One Mile End	51.520151	-0.056136	Brewery
3	Whitechapel, Stepney, Mile End	51.520435	-0.053122	Mouse Tail Coffee Stories	51.519471	-0.058573	Coffee Shop
4	Whitechapel, Stepney, Mile End	51.520435	-0.053122	Basicsalon	51.520669	-0.056093	Cosmetics Shop

First 5 of 3340 venues returned

2. One-hot encoding

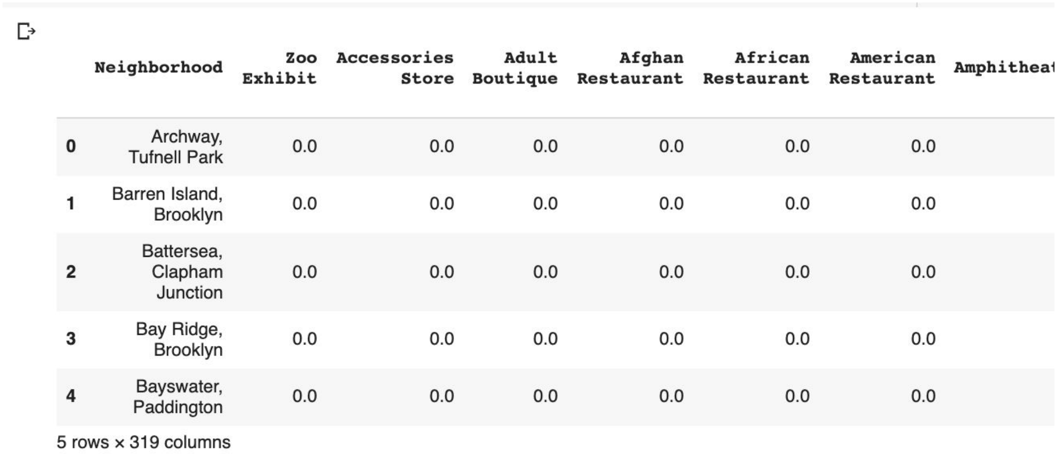
I used one-hot encoding to convert the data into dummy variables. I broke up all the locations into distinct columns in a dataframe, with a 1 or 0 in each row depending on if that specific venue matched the category of that column.



	Zoo Exhibit	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Amphi
0	0	0	0	0	0	0	
1	0	0	0	0	0	0	
2	0	0	0	0	0	0	
3	0	0	0	0	0	0	
4	0	0	0	0	0	0	

5 rows × 319 columns

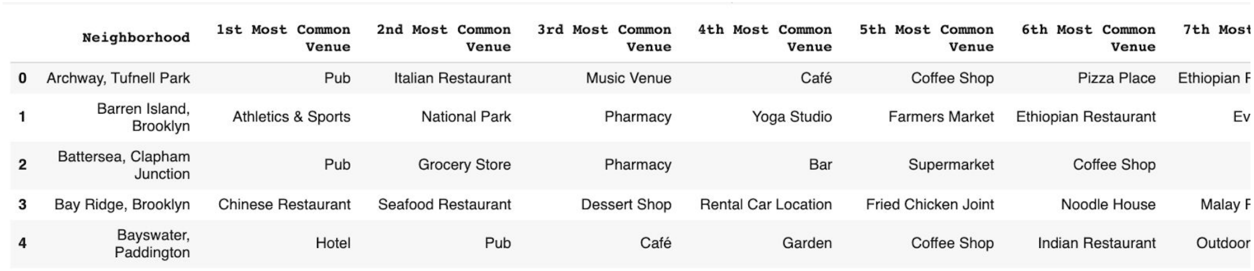
Then, I grouped by neighborhood and took the mean of the frequency of occurrences for each category



	Neighborhood	Zoo Exhibit	Accessories Store	Adult Boutique	Afghan Restaurant	African Restaurant	American Restaurant	Amphitheat
0	Archway, Tufnell Park	0.0	0.0	0.0	0.0	0.0	0.0	
1	Barren Island, Brooklyn	0.0	0.0	0.0	0.0	0.0	0.0	
2	Battersea, Clapham Junction	0.0	0.0	0.0	0.0	0.0	0.0	
3	Bay Ridge, Brooklyn	0.0	0.0	0.0	0.0	0.0	0.0	
4	Bayswater, Paddington	0.0	0.0	0.0	0.0	0.0	0.0	

5 rows × 319 columns

I took the top 5 venues in each neighborhood and then put it all into a Pandas dataframe:



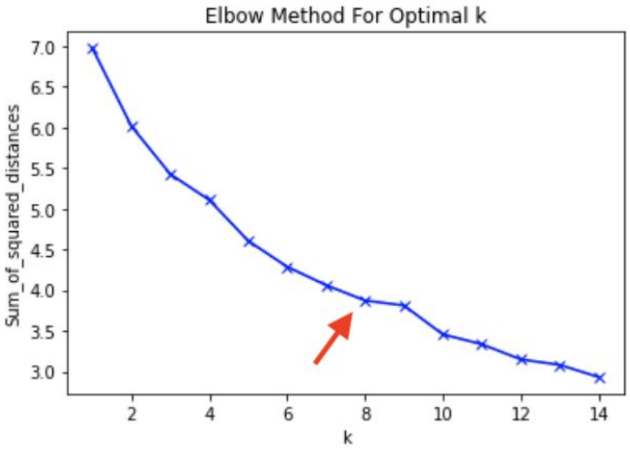
	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most
0	Archway, Tufnell Park	Pub	Italian Restaurant	Music Venue	Café	Coffee Shop	Pizza Place	Ethiopian F
1	Barren Island, Brooklyn	Athletics & Sports	National Park	Pharmacy	Yoga Studio	Farmers Market	Ethiopian Restaurant	Ev
2	Battersea, Clapham Junction	Pub	Grocery Store	Pharmacy	Bar	Supermarket	Coffee Shop	
3	Bay Ridge, Brooklyn	Chinese Restaurant	Seafood Restaurant	Dessert Shop	Rental Car Location	Fried Chicken Joint	Noodle House	Malay F
4	Bayswater, Paddington	Hotel	Pub	Café	Garden	Coffee Shop	Indian Restaurant	Outdoor

3. Machine Learning — k-means clustering algorithm

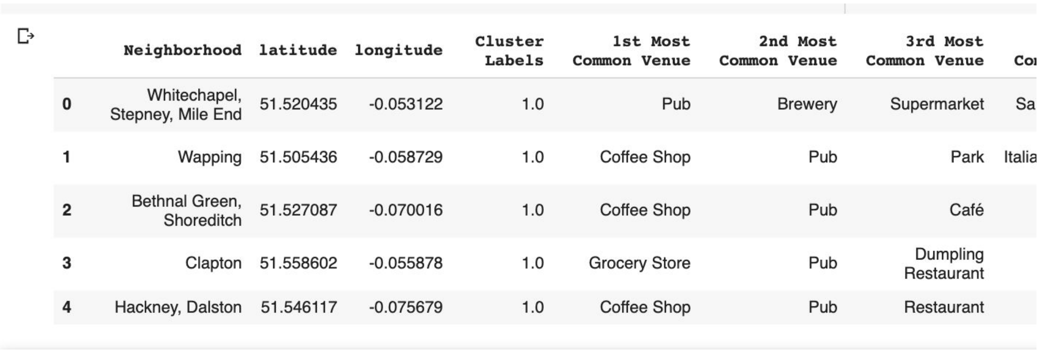
To compare the neighborhoods, I used the k-means clustering algorithm. It is one of the most popular unsupervised machine learning algorithms and its objective is to group similar data points together and discover underlying patterns.

In the context of this project, the clusters would be made up of the neighborhoods with similar patterns based on the venues within them.

Before clustering the neighborhoods, I used the ‘elbow method’ to determine the optimal number of clusters, or ‘k’.

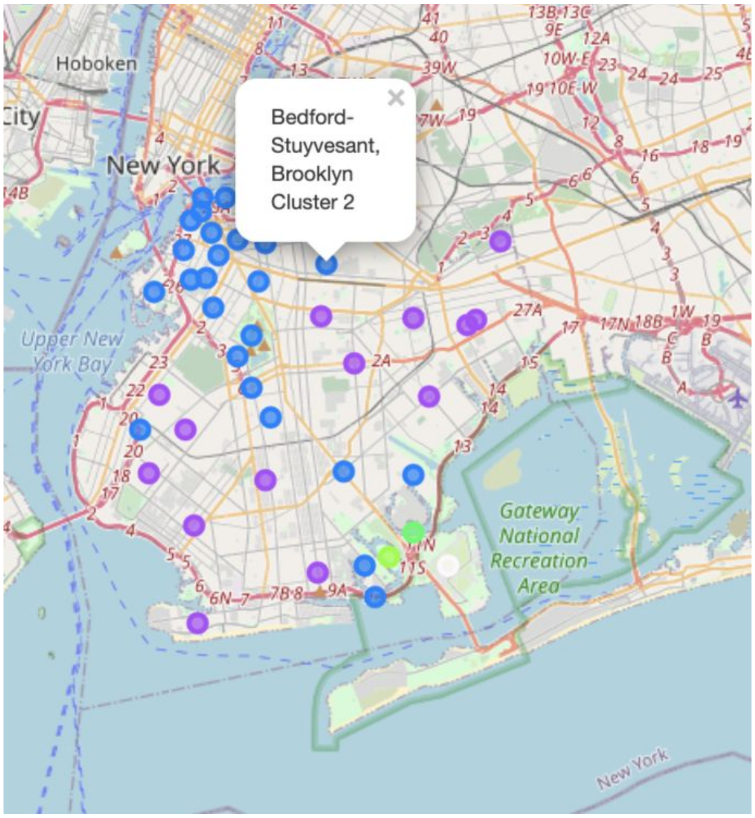


Based on the resulting chart, it appears that the optimal number of clusters is around 8. So, I set k = 8 and ran the kMeans clustering algorithm from scikit-learn to assign each neighborhood in my table to a specific cluster.

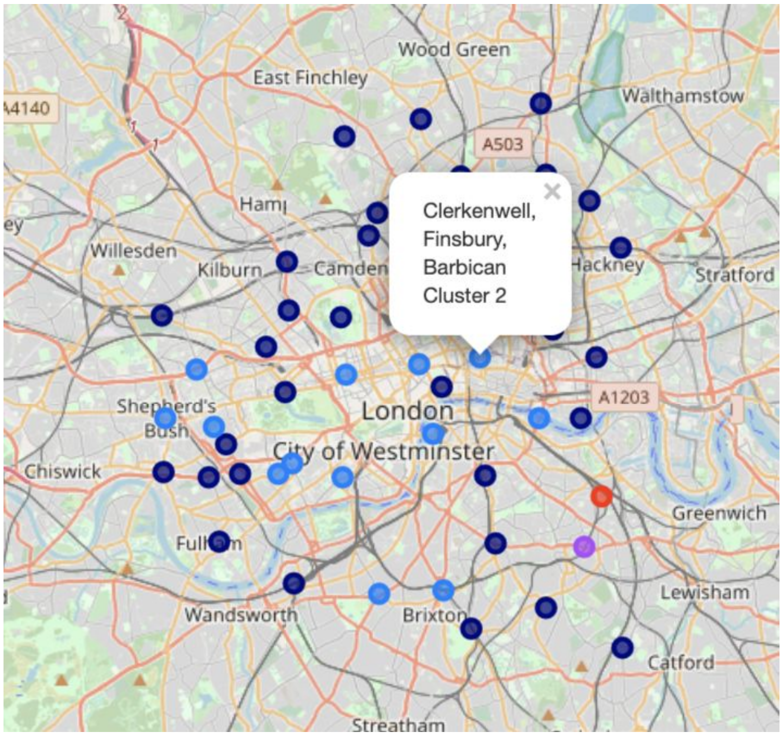


	Neighborhood	latitude	longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	Co
0	Whitechapel, Stepney, Mile End	51.520435	-0.053122	1.0	Pub	Brewery	Supermarket	Sa
1	Wapping	51.505436	-0.058729	1.0	Coffee Shop	Pub	Park	Italia
2	Bethnal Green, Shoreditch	51.527087	-0.070016	1.0	Coffee Shop	Pub	Café	
3	Clapton	51.558602	-0.055878	1.0	Grocery Store	Pub	Dumpling Restaurant	
4	Hackney, Dalston	51.546117	-0.075679	1.0	Coffee Shop	Pub	Restaurant	

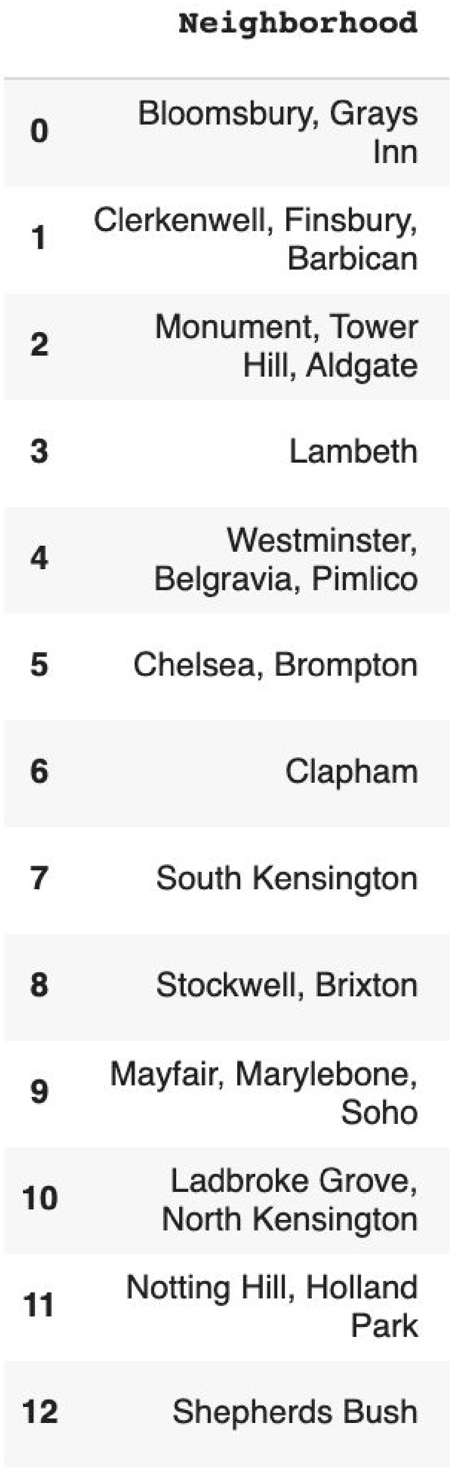
Then, I added these cluster labels to my maps. My old neighborhood of Bed-Stuy is located in cluster 2.



Brooklyn clusters



London clusters



	Neighborhood
0	Bloomsbury, Grays Inn
1	Clerkenwell, Finsbury, Barbican
2	Monument, Tower Hill, Aldgate
3	Lambeth
4	Westminster, Belgravia, Pimlico
5	Chelsea, Brompton
6	Clapham
7	South Kensington
8	Stockwell, Brixton
9	Mayfair, Marylebone, Soho
10	Ladbroke Grove, North Kensington
11	Notting Hill, Holland Park
12	Shepherds Bush

List of neighborhoods in London that are in cluster 2.

IV. Result:

Based on my analysis, I should consider living in one of the 12 neighborhoods in the above table if I wanted to live in a neighborhood that most closely resembles my old Brooklyn neighborhood. Something to keep in mind is that the clusters I ended up with were pretty imbalanced (40 neighborhoods in cluster 2 vs. just 1 n in clusters 3,4,5,6) and Bedford-Stuyvesant happened to be in that largest cluster.

V. Discussion:

There are definitely some caveats that I'd like to bring up and would take into account if I chose to take my analysis further. Namely, it would have been helpful to include other variables such as rent prices or proximity to public transportation, to name a few.

Something else to keep in mind regarding the accuracy of the clusters, it is possible that the data was a little skewed based on different naming conventions for similar establishments in the United Kingdom vs. in the United States. For example, in London, there were many "pubs" whereas in Brooklyn, there were many more "bars". In the context of this project, these could have been considered the same type of venue. It is likely that taking into account such nuances when building my model would have resulted in different clusters than those that I ended up with. There are also many other clustering methods I could have used instead of k-means which would have yielded different results.

VI. Conclusion:

There is a never-ending list of factors to consider when choosing a neighborhood to live in. I chose to explore one of these — nearby venues. When moving to a new city, let alone a new country, having familiar types of locations in the neighborhood can be a huge source of comfort. As discussed in the above section, there are many ways I could improve this model to get a more accurate picture of similarities within neighborhoods.

VII. Sources & Documentation:

1. London Data: https://en.wikipedia.org/wiki/List_of_areas_of_London
2. Brooklyn Data: https://en.wikipedia.org/wiki/List_of_Brooklyn_neighborhoods#By_geographical_region
3. Folium: https://python-visualization.github.io/folium/
4. Geopy: https://geopy.readthedocs.io/en/stable/
5. Foursquare: https://developer.foursquare.com/
6. Scikit learn k-means: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html