Analysis and Recommendations for Investors who plan to start their Business in Hamburg (Germany)

1. Introduction

1.1. Background

Hamburg, the largest city in Germany after the capital Berlin, its location makes it an important link between the sea and Germany's network of inland waterways and numerous islands. The city is best known for its famous harbor area, the Port of Hamburg. In addition to being a major transportation hub, Hamburg has become one of Europe's most important cultural and commercial centers, as well as a major tourist destination.

The city is an excellent location for nascent entrepreneurs with clever ideas. More than 700 startup businesses are based here, with almost half of their total staff coming from abroad. Founders can benefit from Hamburg’s cosmopolitan flair, high quality of life, and optimum conditions for setting up a business.

1.2. Business Problem

Every international business starting in an unknown area especially in a new country is facing several problems:

- Where to find suitable offices and commercial spaces?
- What neighborhoods are best for it?

The business location plays a very important role and makes a great contribution to business success.

On the one hand, every business type has its optimal location, eg. restaurants succeed more in areas, that are visited by tourists, and a company office is better situated in a business district.

On the other hand, the crime rate of the neighborhood is also an important factor, that has an impact on business success.

International research has long shown evidence that crime makes communities decline (e.g. Skogan, 1990; Wilson & Kelling, 1982). This decline can be seen in the presence of crime in public places as well as in minor signs of physical and social disorder.

Shoplifting is the biggest concern, and the biggest problem, for most small-business owners. When the business is closed, burglary and breaking and entering become another concern in this criminal category.

Most businesses are sensitive to crime in their neighborhoods, especially jewelry shops, liquor stores, banks, hotels, etc.

1.3. Interest

The audience, who is interested in the information to the problems mentioned above are international companies or startups from foreign cities or countries intending to start or expand their business to Hamburg.

2. Data

2.1. Data Sources

To solve the business problem we need the following data:

1. The list of boroughs and neighborhoods can be found on Wikipedia (article “List of Districts and Neighborhoods of Hamburg”) [1] .

2. We can retreat the crime data from Hamburg Police Crime Statistics (PDF file, pages 16-19) [2].

3. To plot the boundaries of the neighborhoods of Hamburg with Choropleth maps we need a GEOJSON file. It can be downloaded from this source: [3].

4. We will use the neighborhood data, specifically the longitude and latitude to explore the venues in each neighborhood using the Foursquare API [4].

Then we will use machine learning to group the venues into a certain amount of clusters and plot them on the map.

We also will plot the police statistics to show the crime rate of each district.

Based on this information stakeholders can make a decision choosing the optimal location for their new business.

2.2. Data Cleaning

2.2.1. The list of boroughs and neighborhoods can be found on Wikipedia (article “List of Districts and Neighborhoods of Hamburg”). [1]

Example:

Stadtteil	Ortsteile	Bezirk	Fläche (km²)	Einwohner	Bevölkerungsdichte (Einwohner/km²)	Koordinaten	Karte
Hamburg-Altstadt	101–102	Hamburg-Mitte	2,4	2350	979	53° 33′ 0″ N, 10° 0′ 0″ O	
HafenCity	103–104	Hamburg-Mitte	2,2	4925	2239	53° 32′ 28″ N, 10° 0′ 1″ O	
Neustadt	105–108	Hamburg-Mitte	2,3	12.762	5549	53° 33′ 7″ N, 9° 59′ 8″ O	
St. Pauli	109–112	Hamburg-Mitte	2,5	22.097	8839	53° 33′ 25″ N, 9° 57′ 50″ O	
St. Georg	113–114	Hamburg-Mitte	2,4	11.358	4733	53° 33′ 18″ N, 10° 0′ 44″ O	
Hammerbrook	115–118	Hamburg-Mitte	3,0	4619	1540	53° 32′ 43″ N, 10° 1′ 50″ O	
Borgfelde	119–120	Hamburg-Mitte	0,8	8343	10429	53° 33′ 17″ N, 10° 2′ 4″ O	

I used the native Pandas read_html method to extract the tables from the webpage.

We got a list of 2 DataFrames, but only the second one had all the necessary data.

<table>
<tr><th>Dropped Features</th><th>Used Features</th></tr>
<tr><td>Ortsteile: numbers of neighborhoods
Fläche: area of the neighborhood in sq.km.
Bevölkerungsdiche: the density of population in residents per sq.km.
Karte: map fragment of the neighborhood.</td><td>Stadtteil: the names of the neighborhoods.
Bezirk: the names of boroughs.
Einwohner: the population of the neighborhood (must be converted to int)
Koordinaten: latitude and longitude in DMS format (must be converted to decimal format (float).</td></tr>
</table>

All the data were in as datatype string, so we had to convert the column “Einwohner” (population) to the type int.

From the neighborhood rows, we dropped the neighborhood “Neuwerk” since it is an unpopulated island.

So we got 103 neighborhoods.

But it was more complex to get the separate latitude and longitude coordinates in decimal format from the column “Koordinaten” since they were united in one column and in the DMS format (degree, minutes, seconds). So at first, I split the column into latitude and longitude columns using the Pandas .str.split method. Then I created a function, using regular expressions to extract the numbers and convert the latitude and longitude values from the DMS format into decimal to be able to use them later in the Foursquare API request.

2.2.2. Crime data from Hamburg Police Crime Statistics (PDF file, pages 16-19). [2]

Example:

3.3.3 Stadtteile

Bezirk Hamburg-Mitte

Stadtteile	2017	2018			Zu- / Abnahme	
	Fälle	Fälle	aufgeklärt	in %	absolut	in %
Altstadt	7.581	6.742	4.159	61,7	-839	-11,1
HafenCity	761	821	219	26,7	60	7,9
Neustadt	4.893	5.063	2.367	46,8	170	3,5
St. Pauli	18.289	18.790	8.275	44,0	501	2,7
St. Georg	19.167	20.047	14.193	70,8	880	4,6
Hammerbrook	2.591	2.359	1.135	48,1	-232	-9,0
Borgfelde	771	692	255	36,8	-79	-10,2
Hamm	3.175	2.872	1.092	38,0	-303	-9,5
Horn	3.633	3.165	1.443	45,6	-468	-12,9
Billstedt	7.771	7.442	3.651	49,1	-329	-4,2
Billbrook	663	699	274	39,2	36	5,4
Rothenburgsort	1.398	1.319	577	43,7	-79	-5,7
Veddel	709	967	612	63,3	258	36,4
Wilhelmsburg	6.671	6.432	2.533	39,4	-239	-3,6
Kleiner Grasbrook	299	251	117	46,6	-48	-16,1
Steinwerder	178	210	117	55,7	32	18,0
Waltershof	132	120	62	51,7	-12	-9,1
Finkenwerder	644	585	237	40,5	-59	-9,2
Insel Neuwerk	0	0	0	----	0	----
Bezirk Mitte	**79.326**	**78.576**	**41.318**	**52,6**	**-750**	**-0,9**

Dropped Features	**Used Features**
• *Ortsteile:* numbers of neighborhoods • *2017 Fälle:* crime accidents in 2017 • *2018* aufgeklärt: crimes solved in 2018 • *in %*: in % • *Zu- / Abnahme absolut*: increase / decrease absolut • *in %:* in %	• *Stadtteil: the names of the neighborhoods.* • *2018 Fälle: crime accidents in 2018, must be converted to int)*

To read the pdf file into Pandas DataFrame I used the tabula.read_pdf module, that must be installed at first.

I read the pages 16-19, containing the necessary tables.

We got a list of DataFrames, with lists of neighborhoods separately for each borough, which had to be merged into one table.

The first table was read differently from other tables, so to make it ready for merging I had to reassign the two necessary columns.

After that, I used the Pandas .concat method to concatenate all the DataFrames into one DataFrame.

The column with crime data was in string format using "." as thousands separator, so it had to be removed, after that we could convert the values into the int format.

2.2.3. We created a request to the Foursquare API and got the following response in JSON format.

Example:


```
[{'reasons': {'count': 0,
   'items': [{'reasonName': 'globalInteractionReason',
     'summary': 'This spot is popular',
     'type': 'general'}]},
  'referralId': 'e-0-4db16e9bf7b1bd003adbeb06-0',
  'venue': {'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/german_',
      'suffix': '.png'},
     'id': '4bf58dd8d48988d10d941735',
     'name': 'German Restaurant',
     'pluralName': 'German Restaurants',
     'primary': True,
     'shortName': 'German'}],
   'id': '4db16e9bf7b1bd003adbeb06',
   'location': {'address': 'Estedeich 88',
    'cc': 'DE',
    'city': 'Hamburg',
    'country': 'Deutschland',
    'distance': 284,
    'formattedAddress': ['Estedeich 88', '21129 Hamburg', 'Deutschland'],
    'labeledLatLngs': [{'label': 'display',
      'lat': 53.534702,
      'lng': 9.778484}],
    'lat': 53.534702,
    'lng': 9.778484,
    'postalCode': '21129',
    'state': 'Hamburg'},
   'name': 'Gasthaus zur Post',
   'photos': {'count': 0, 'groups': []}}},
 {'reasons': {'count': 0,
   'items': [{'reasonName': 'globalInteractionReason',
     'summary': 'This spot is popular',
     'type': 'general'}]},
  'referralId': 'e-0-4db42e7493a017099dd33ecc-1',
  'venue': {'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/german_',
      'suffix': '.png'},
     'id': '4bf58dd8d48988d10d941735',
     'name': 'German Restaurant',
     'pluralName': 'German Restaurants',
     'primary': True,
     'shortName': 'German'}],
   'id': '4db42e7493a017099dd33ecc',
   'location': {'cc': 'DE',
    'city': 'Hamburg',
    'country': 'Deutschland',
    'distance': 258,
    'formattedAddress': ['21129 Hamburg', 'Deutschland'],
    'labeledLatLngs': [{'label': 'display',
      'lat': 53.534674,
      'lng': 9.779735}],
    'lat': 53.534674,
    'lng': 9.779735,
    'postalCode': '21129',
    'state': 'Hamburg'},
   'name': 'Altes Fährhaus',
   'photos': {'count': 0, 'groups': []}}}]
```


Used Features
• *Venue name* • *Venue category*

To extract the necessary features from that response we installed the json module, which made it able to convert it into a python dictionary.

2.2.4 GEOJSON file with boundaries of the neighborhoods of Hamburg

Example:

{"type": "FeatureCollection", "features": [{"type":"Feature","geometry":{"type":"MultiPolygon","coordinates":[[[[10.269134,53.467846],[10.269706,53.467598],
[10.270433,53.467282],[10.270698,53.467167],[10.270936,53.467061],[10.271144,53.466969],[10.271562,53.466785],[10.271902,53.466638],[10.271783,53.466518],
[10.271626,53.466361],[10.27154,53.466276],[10.271435,53.466171],[10.271411,53.466148],[10.271381,53.466118],[10.271209,53.465946],[10.270989,53.465727],
[10.270661,53.465398],[10.270238,53.464982],[10.270085,53.464831],[10.269661,53.464409],[10.269439,53.464174],[10.269361,53.464092],[10.269339,53.464069],
[10.269382,53.464054],[10.269548,53.463996],[10.269893,53.463885],[10.27007,53.463806],[10.270341,53.463711],[10.270393,53.463695],[10.270919,53.463516],
[10.271293,53.463394],[10.271684,53.463276],[10.271896,53.463208],[10.272192,53.46311],[10.272442,53.463025],[10.272816,53.462857],[10.272936,53.462819],
[10.273116,53.462803],[10.273429,53.462723],[10.273507,53.462691],[10.273579,53.462661],[10.273671,53.462613],[10.273914,53.462449],[10.274285,53.462187],
[10.274488,53.462045],[10.274617,53.46196],[10.274688,53.461851],[10.274707,53.461788],[10.274746,53.461739],[10.274825,53.461681],[10.275493,53.46135],[10.275665,53.461269],
[10.275736,53.46124],[10.275986,53.46112],[10.276131,53.46106],[10.27637,53.460947],[10.276546,53.460862],[10.276758,53.460764],[10.277037,53.46062],[10.277162,53.460574],
[10.277265,53.460524],[10.277282,53.460501],[10.277366,53.46044],[10.277661,53.460264],[10.277965,53.460116],[10.278232,53.460004],[10.278898,53.459751],
[10.279098,53.459683],[10.279351,53.459579],[10.279473,53.459529],[10.279681,53.459437],[10.279763,53.459398],[10.280077,53.459264],[10.280335,53.459173],
[10.280528,53.459089],[10.280749,53.458983],[10.280909,53.458881],[10.280985,53.458827],[10.281075,53.458754],[10.281186,53.458652],[10.281304,53.458556],
[10.281336,53.458534],[10.28144,53.458458],[10.281585,53.458324],[10.281803,53.458142],[10.281863,53.458084],[10.281924,53.45802],[10.281996,53.457939],[10.28234,53.457547],
[10.282367,53.457564],[10.282738,53.45779],[10.283038,53.457974],[10.283482,53.458253],[10.283646,53.458355],[10.283809,53.458461],[10.283883,53.45851],[10.28391,53.458516],
[10.283927,53.458512],[10.283966,53.458498],[10.28404,53.458454],[10.284056,53.458443],[10.284164,53.458376],[10.284331,53.458273],[10.284429,53.458215],
[10.284598,53.458116],[10.284657,53.458086],[10.28492,53.45797],[10.28499,53.457944],[10.285065,53.457915],[10.285224,53.457863],[10.285226,53.457862],[10.285433,53.457787],
[10.285648,53.457697],[10.285811,53.457622],[10.285961,53.457546],[10.286059,53.457494],[10.286167,53.457439],[10.286257,53.457384],[10.286439,53.457258]

The problem in this file was that it included the German special characters (Umlaute) “ö, ü, ä, ß: “, which were not recognized properly in the ASCI format and had to be replaced in the file with ASCI “friendly” characters “oe, ue, ae, ss”.

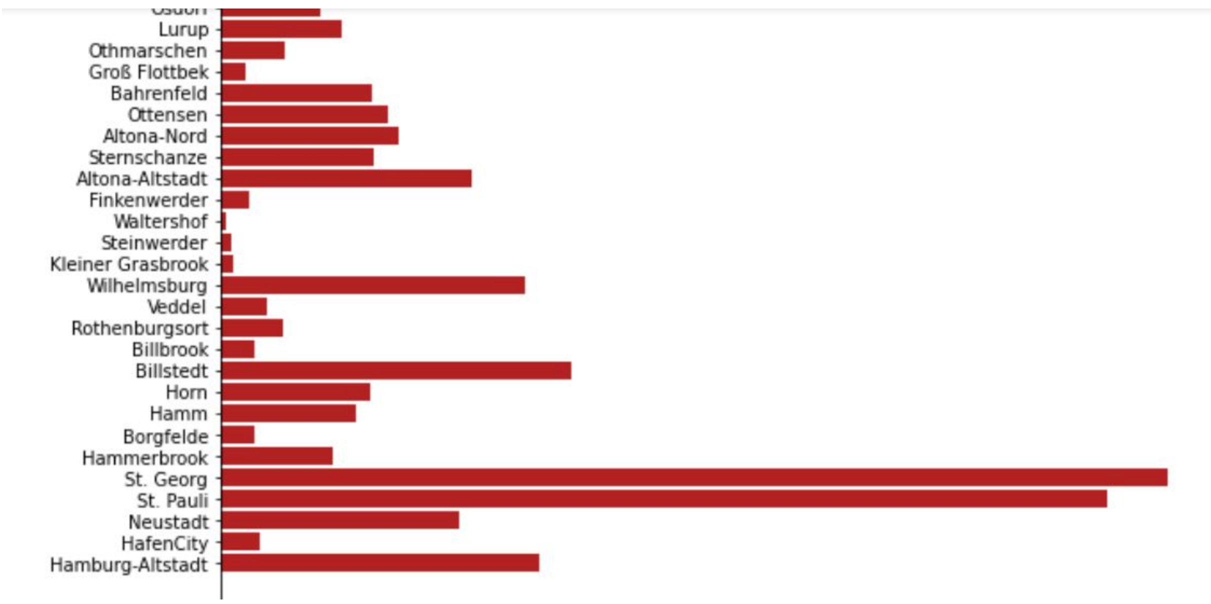
3. Methodology

3.1. Exploratory Data Analysis

3.1.1. The Crime Situation in the Neighborhoods of Hamburg

After getting the crime data we can make a bar plot, showing the crime rate in each neighborhood.

Clip:



As we see, our "crime leaders" of Hamburg are the neighborhoods St. Pauli and St. Georg. Their crime rate exceeds significantly all other neighborhoods.

We use the Pandas .join method to join the neighborhood DataSet and the crime dataset on the column “Neighborhood”.

Now we can create another necessary column “Crimes per Capita” simply dividing the data from the column “Crimes in 2018” by the column “Population”.

It gives us a better understanding of the relative crime rate since the population of the neighborhoods varies from 142 to 92087 persons.

Example:

	Neighborhood	Borough	Population	Latitude	Longitude	Crimes in 2018	Crimes per Capita
0	Hamburg-Altstadt	Hamburg-Mitte	2350	53.550000	10.000000	6742	2.868936
1	HafenCity	Hamburg-Mitte	4925	53.541111	10.000278	821	0.166701
2	Neustadt	Hamburg-Mitte	12762	53.551944	9.985556	5063	0.396725
3	St. Pauli	Hamburg-Mitte	22097	53.556944	9.963889	18790	0.850342
4	St. Georg	Hamburg-Mitte	11358	53.555000	10.012222	20047	1.765011
...	...	...	...	...	...	...	...
99	Hausbruch	Harburg	17036	53.466667	9.883333	942	0.055295
100	Neugraben-Fischbek	Harburg	31589	53.483333	9.850000	2112	0.066859
101	Francop	Harburg	715	53.508056	9.852778	20	0.027972
102	Neuenfelde	Harburg	4927	53.514722	9.795556	174	0.035316
103	Cranz	Harburg	804	53.536944	9.780556	24	0.029851

3.1.2. Gathering the Information about Venues in Neighborhoods

To get the venues of the Neighborhoods we use the Foursquare API passing into the GET request following parameters:

- CLIENT_ID
- CLIENT_SECRET
- VERSION
- Latitude
- Longitude
- Radius
- LIMIT

The credentials (CLIENT_ID and CLIENT_SECRET) we get by registering to the Foursquare API.

We use the version “20180605”.

Also, the latitude and longitude from the corresponding DataSet are used to assemble the request URL for each neighborhood.

The radius was set to 600 meters to avoid overlapping of the same venues by scanning different neighborhoods.

The limit was set to 1000 to get the most venus.

Example of the URL:

https://api.foursquare.com/v2/venues/explore?&client_id=L1MM1ECQMJI4EWWW0SA2YOXXXD4PI2A1NHENJX2B5XXXXXX&client_secret=KL1DMABXU2XXXXXXGJ3FSD2SE2KXULSRHKXXXXXXXXXHI2&v=20180605&ll=53.5369444,9.7805556&radius=600&limit=1000

As a response, we get a JSON file, which can be converted by using the json module to a dictionary.

Example:


```
[{'reasons': {'count': 0,
   'items': [{'reasonName': 'globalInteractionReason',
     'summary': 'This spot is popular',
     'type': 'general'}]},
  'referralId': 'e-0-4db16e9bf7b1bd003adbeb06-0',
  'venue': {'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/german_',
      'suffix': '.png'},
     'id': '4bf58dd8d48988d10d941735',
     'name': 'German Restaurant',
     'pluralName': 'German Restaurants',
     'primary': True,
     'shortName': 'German'}],
   'id': '4db16e9bf7b1bd003adbeb06',
   'location': {'address': 'Estedeich 88',
    'cc': 'DE',
    'city': 'Hamburg',
    'country': 'Deutschland',
    'distance': 284,
    'formattedAddress': ['Estedeich 88', '21129 Hamburg', 'Deutschland'],
    'labeledLatLngs': [{'label': 'display',
      'lat': 53.534702,
      'lng': 9.778484}],
    'lat': 53.534702,
    'lng': 9.778484,
    'postalCode': '21129',
    'state': 'Hamburg'},
   'name': 'Gasthaus zur Post',
   'photos': {'count': 0, 'groups': []}}},
 {'reasons': {'count': 0,
   'items': [{'reasonName': 'globalInteractionReason',
     'summary': 'This spot is popular',
     'type': 'general'}]},
  'referralId': 'e-0-4db42e7493a017099dd33ecc-1',
  'venue': {'categories': [{'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/german_',
      'suffix': '.png'},
     'id': '4bf58dd8d48988d10d941735',
     'name': 'German Restaurant',
     'pluralName': 'German Restaurants',
     'primary': True,
     'shortName': 'German'}],
   'id': '4db42e7493a017099dd33ecc',
   'location': {'cc': 'DE',
    'city': 'Hamburg',
    'country': 'Deutschland',
    'distance': 258,
    'formattedAddress': ['21129 Hamburg', 'Deutschland'],
    'labeledLatLngs': [{'label': 'display',
      'lat': 53.534674,
      'lng': 9.779735}],
    'lat': 53.534674,
    'lng': 9.779735,
    'postalCode': '21129',
    'state': 'Hamburg'},
   'name': 'Altes Fährhaus',
   'photos': {'count': 0, 'groups': []}}}]
```

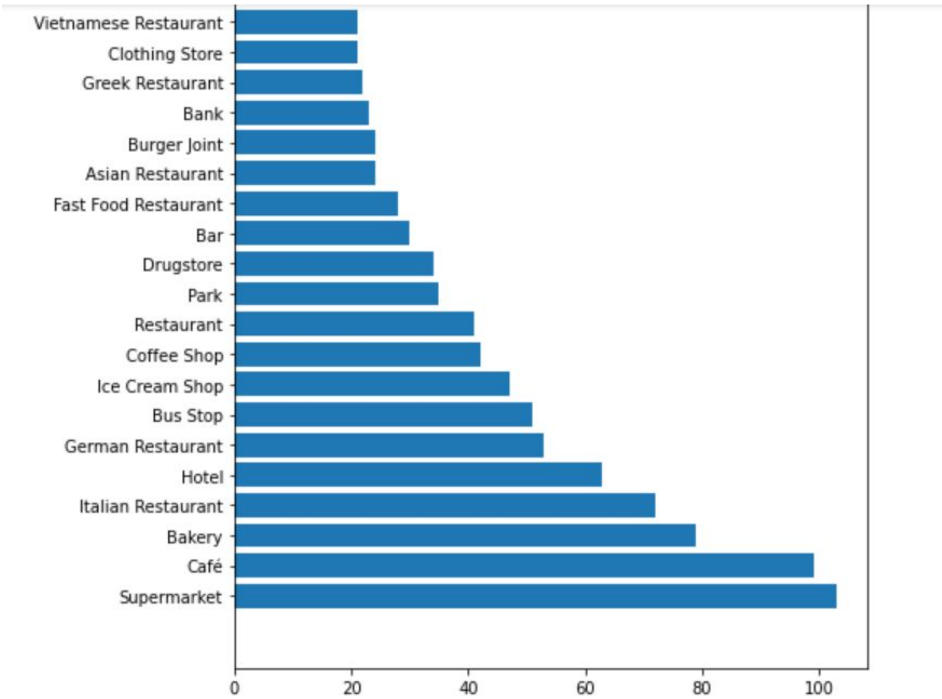

From that dictionary, we can extract the necessary information, namely the venue name and the venue category.
Using the extracted data we build the following DataFrame.

Example:

	Neighborhood	Neighborhoods Latitude	Neighborhoods Longitude	Venue Name	Venue Category
0	Hamburg-Altstadt	53.550000	10.000000	GOOT - Finest Cuts	Comfort Food Restaurant
1	Hamburg-Altstadt	53.550000	10.000000	Mi Chii	Vietnamese Restaurant
2	Hamburg-Altstadt	53.550000	10.000000	O-ren Ishii	Vietnamese Restaurant
3	Hamburg-Altstadt	53.550000	10.000000	Thalia Theater	Theater
4	Hamburg-Altstadt	53.550000	10.000000	Manufactum	Furniture / Home Store
...	...	...	...	...	...
1828	Neuenfelde	53.514722	9.795556	Lidl	Supermarket
1829	Neuenfelde	53.514722	9.795556	Bäckerei Rundt	Bakery
1830	Neuenfelde	53.514722	9.795556	LOTTO Hamburg	Lottery Retailer

We can create a bar chart from the top venues in Hamburg.

Clip:



3.2. Machine Learning and inferential statistical Testing

We will cluster our neighborhoods by its venues using the feature “Venue Category”. But since this feature is categorical we cannot use in in the KMeans clustering algorithm. So the venue categories must be converted using one-hot encoding.

Therefore we use the Pandas .get_dummies method.

We get the following DataSet

Clip:

ATM	Accessories Store	Afghan Restaurant	Airport	American Restaurant	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Austrian Restaurant	Auto Dealership
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0

Then we group our one-hot DataSet by the column 'Neighborhood' and apply the Pandas .mean method.

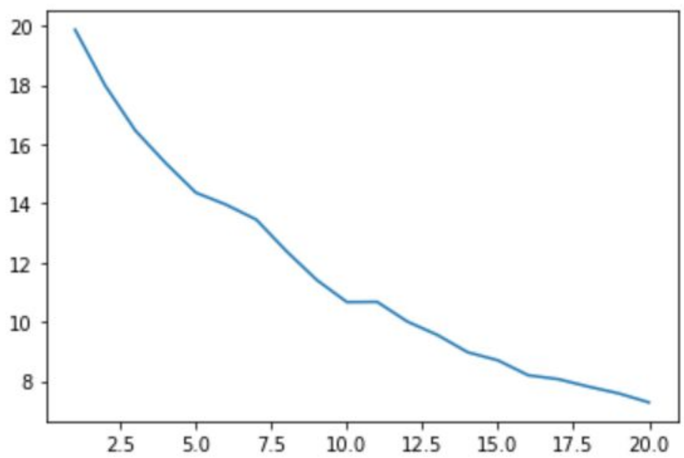
Clip:

	Neighborhood	ATM	Accessories Store	Afghan Restaurant	Airport	American Restaurant	Arcade	Argentinian Restaurant	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Athletics & Sports	Austrian Restaurant
0	Allermöhe	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
1	Alsterdorf	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
2	Altenwerder	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
3	Altona-Altstadt	0.0	0.0	0.0	0.0	0.0	0.017857	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
4	Altona-Nord	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.025000	0.000000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
92	Wellingsbüttel	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
93	Wilhelmsburg	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.062500	0.000000
94	Wilstorf	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.000000	0.000000	0.000000
95	Winterhude	0.0	0.0	0.0	0.0	0.0	0.000000	0.0	0.0	0.0	0.0	0.034483	0.034483	0.034483

Now we can use these data in the KMeans clustering algorithm.

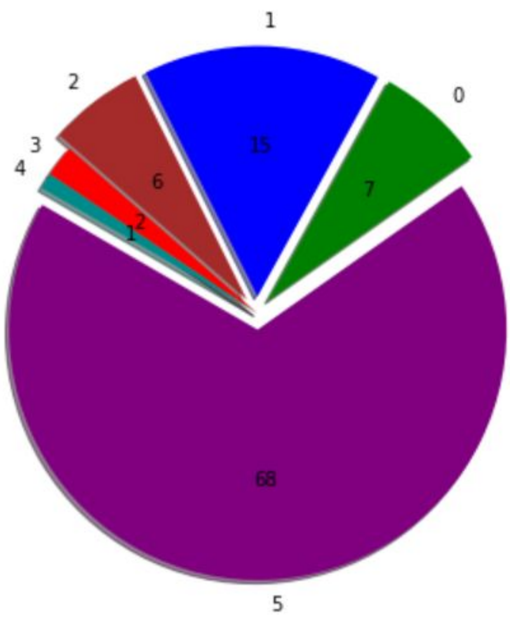
As an output, we get the labels for each neighborhood.

To define the optimal number of clusters I tried to use the elbow method, but it didn't answer the question since I didn't observe an "elbow".



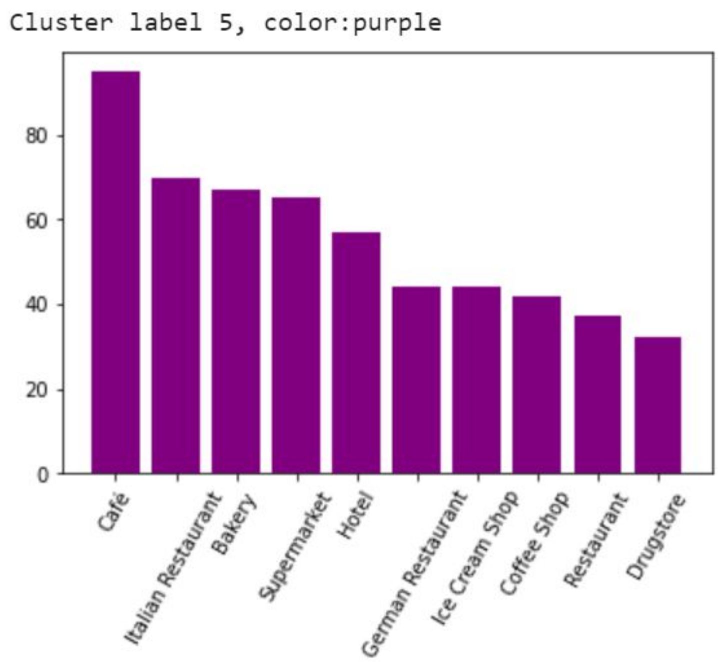
So experimentally I decided to assign the number of clusters to 6.

And we get the following distribution of our neighborhoods in 6 clusters:

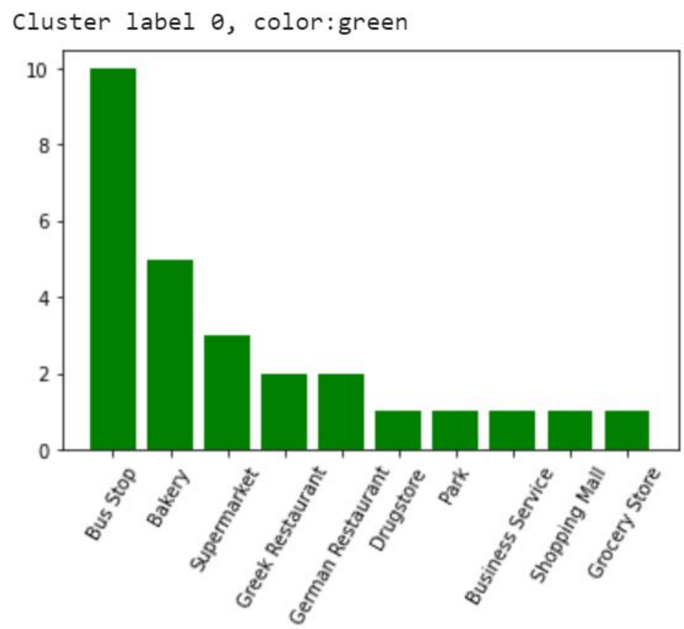


We see, our main clusters are 0, 1, 2, and 5.

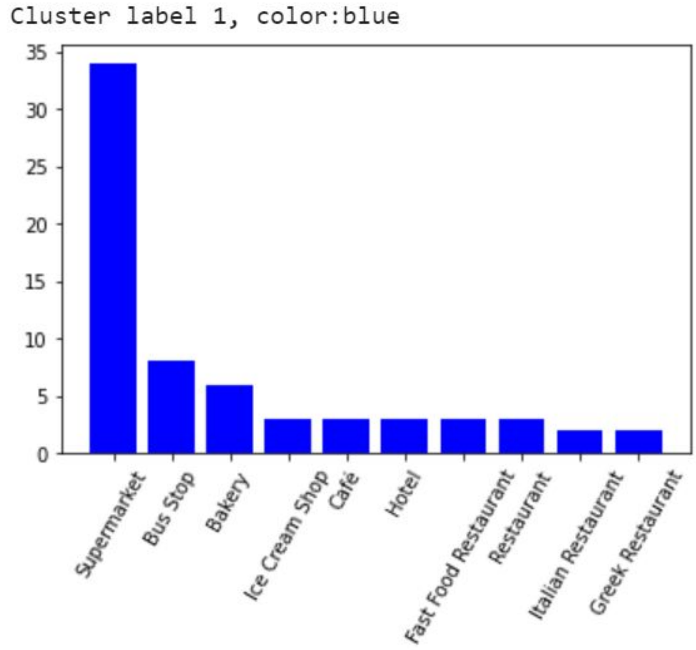
Now we can find out the main venue categories for each cluster, analyze and name them.



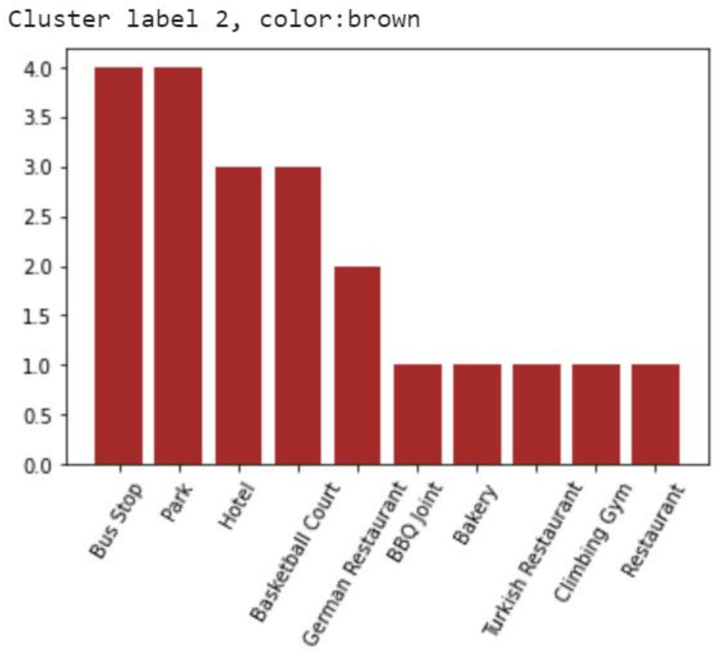
Cluster 5 (purple) is our biggest cluster.
The most common venues here are restaurants, cafes but also supermarkets. A lot of cafes and restaurants are mostly in tourist areas and supermarkets are mostly in residential areas. So, we name cluster 5 **Tourist and Residential area.**



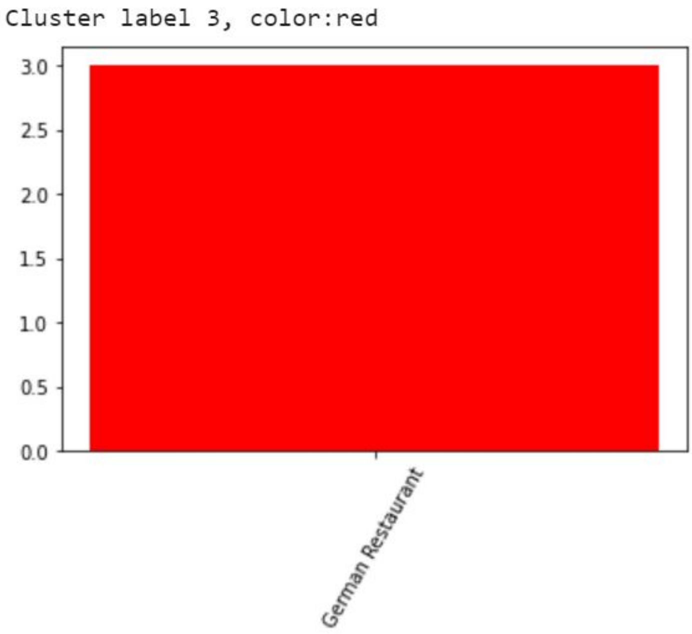
In **cluster 0** (green) the most common venues are bakeries, supermarkets, cafes and there are some business centers.
So, we name cluster 0 **Residential and Business area.**



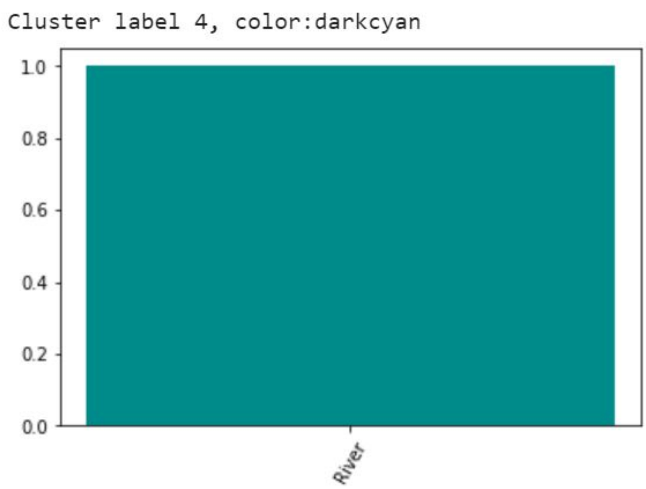
In **cluster 1 (blue)** the most common venues are supermarkets, cafes, and restaurants.
Supermarkets are the most common venue here. So, we name the cluster 1 **Dynamic Residential Area.**



In **cluster 2** (brown) the most common venues are parks, hotels, and sport places.
So, let's call the cluster 2 **Recreation area.**

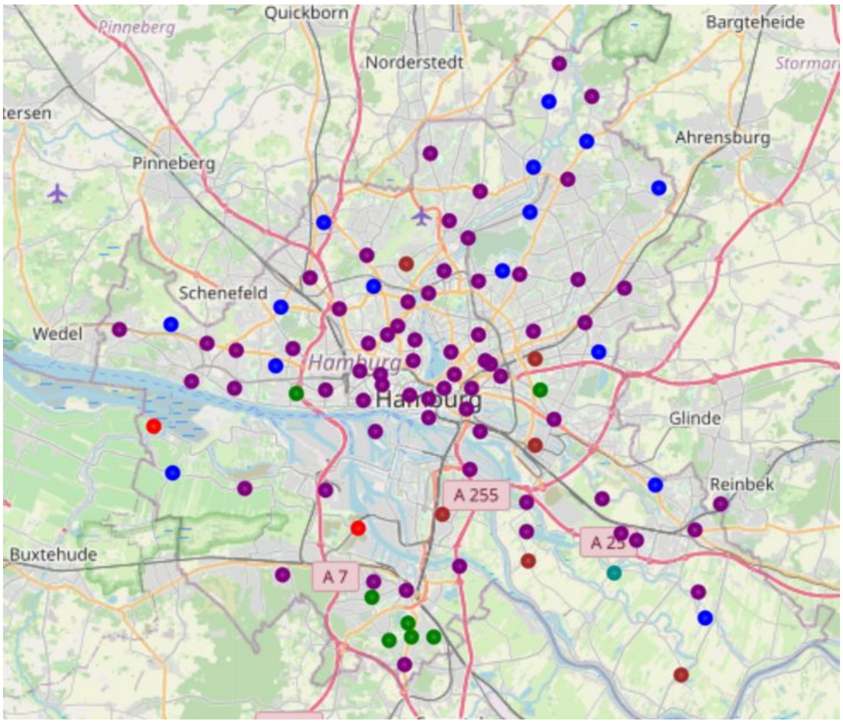


Cluster 3 (red) consists only of one category: German Restaurant. So, it seems to be a quiet residential area with no tourism. Let's call this cluster **Traditional Residential Area.**



Cluster 4 (dark cyan) is the smallest, it's the nature reserve of the neighborhood "Reitbrock". So, let's call cluster 4 **Nature Reserve**.

Now we can plot our clustered neighborhoods on a map.

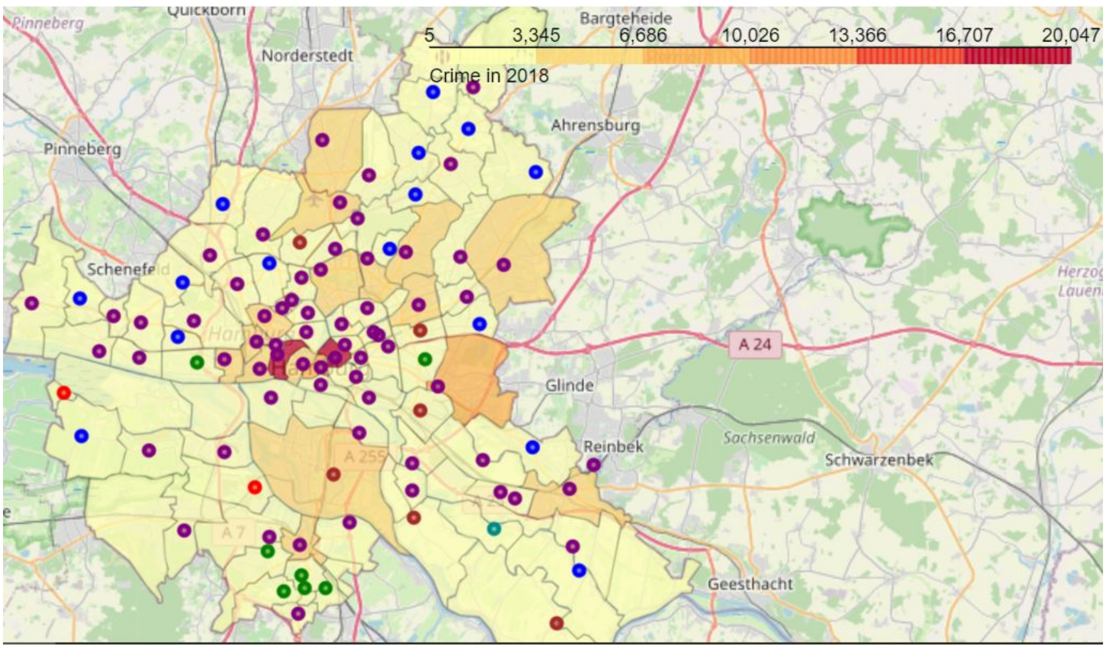


Now we analyze the crime situation in the city.

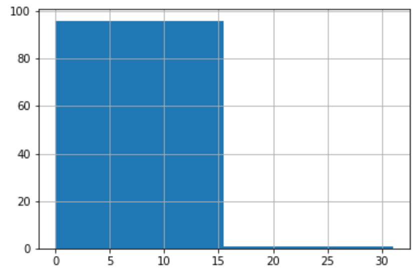
Here is the description of the data:

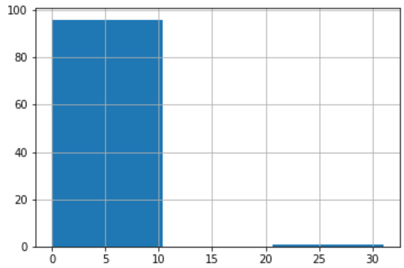
```
count    97.000000
mean      0.533579
std       3.207340
min       0.023352
25%       0.049032
50%       0.071674
75%       0.109977
max      31.000000
Name: Crimes per Capita, dtype: float64
```

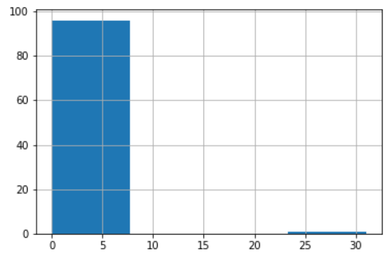
We can create a map showing both: the crime situation and the neighborhood clusters.



Let’s create a data of the crime situation distribution trying different amounts of bins.







The histogram with 2 bins makes more sense since other bins are anyway empty, so I decided to describe the crime situation in the city in 2 categories: "LOW" and "HIGH".

Therefore I created in our DataSet a column "Crime Situation" with this categorical variable using the simple formula: “HIGH” is everything above the mean of “Crimes per Capita”: 0.534.

Clip:

Neighborhood	Borough	Population	Latitude	Longitude	Crimes in 2018	Crimes per Capita	Label	Cluster	Crime Situation
Hamburg-Altstadt	Hamburg-Mitte	2350	53.550000	10.000000	6742	2.868936	5	Tourist and Residetial Area	HIGH
HafenCity	Hamburg-Mitte	4925	53.541111	10.000278	821	0.166701	5	Tourist and Residetial Area	LOW
Neustadt	Hamburg-Mitte	12762	53.551944	9.985556	5063	0.396725	5	Tourist and Residetial Area	LOW
St. Pauli	Hamburg-Mitte	22097	53.556944	9.963889	18790	0.850342	5	Tourist and Residetial Area	HIGH
St. Georg	Hamburg-Mitte	11358	53.555000	10.012222	20047	1.765011	5	Tourist and Residetial Area	HIGH
...	...	...	...	...	...	...	...	...	...
Altenwerder	Harburg	3	53.506944	9.917778	93	31.000000	5	Tourist and Residetial Area	HIGH

We’ve got the following data distribution:

```
LOW     92
HIGH     5
Name: Crime Situation, dtype: int64
```

So there are only 5 neighborhoods with “HIGH” crime situation:

	Neighborhood	Borough	Population	Latitude	Longitude	Label	Cluster	Crime Situation
0	Hamburg-Altstadt	Hamburg-Mitte	2350	53.550000	10.000000	5	Tourist and Residetial Area	HIGH
3	St. Pauli	Hamburg-Mitte	22097	53.556944	9.963889	5	Tourist and Residetial Area	HIGH
4	St. Georg	Hamburg-Mitte	11358	53.555000	10.012222	5	Tourist and Residetial Area	HIGH
15	Steinwerder	Hamburg-Mitte	33	53.534444	9.957222	5	Tourist and Residetial Area	HIGH
98	Altenwerder	Harburg	3	53.506944	9.917778	5	Tourist and Residetial Area	HIGH

Neighborhoods *Steinwerder* and *Altenwerder* got their “HIGH” status due to their low population, in fact, they are unpopulated, so we can set their status to ”LOW”.

4. Results

In this study, we clustered the neighborhoods of Hamburg based on their venues. We combined the neighborhood dataset and the crime information for each neighborhood to provide recommendations to stakeholders for choosing an appropriate but also a safe location for their business needs.

Our analysis shows that the highest concentration of restaurants and cafes is in the central area of the city, which is popular among tourists. But there are also 3 neighborhoods with a high crime rate in the central area: St. Pauli, St. Georg, and Hamburg-Altstadt. We must consider this factor making recommendations for investors.

Let's assume our customer is a company who operates a chain of Chinese restaurants that wants to start their business in Hamburg.

It's looking for tourist neighborhoods to open restaurants there and for a good location for their business office. Safety is one of the important aspects.

What can we offer them?

Our data give an answer to this question. We can find 63 neighborhoods in the “Residential and Tourist Area” with a low crime situation for our potential restaurants.

Clip:

	Neighborhood	Borough	Population	Latitude	Longitude	Label	Cluster	Crime Situation
1	HafenCity	Hamburg-Mitte	4925	53.541111	10.000278	5	Tourist and Residetial Area	LOW
2	Neustadt	Hamburg-Mitte	12762	53.551944	9.985556	5	Tourist and Residetial Area	LOW
5	Hammerbrook	Hamburg-Mitte	4619	53.545278	10.030556	5	Tourist and Residetial Area	LOW
6	Borgfelde	Hamburg-Mitte	8343	53.554722	10.034444	5	Tourist and Residetial Area	LOW
7	Hamm	Hamburg-Mitte	38330	53.560833	10.057778	5	Tourist and Residetial Area	LOW
...	...	...	...	...	...	...	...	...
93	Sinstorf	Harburg	4201	53.423889	9.980556	5	Tourist and Residetial Area	LOW
96	Heimfeld	Harburg	22421	53.463889	9.956111	5	Tourist and Residetial Area	LOW
98	Altenwerder	Harburg	3	53.506944	9.917778	5	Tourist and Residetial Area	LOW
99	Hausbruch	Harburg	17036	53.466667	9.883333	5	Tourist and Residetial Area	LOW
101	Francop	Harburg	715	53.508056	9.852778	5	Tourist and Residetial Area	LOW

63 rows × 8 columns

And 7 suitable neighborhoods for the business office.

Clip:

	Neighborhood	Borough	Population	Latitude	Longitude	Label	Cluster	Crime Situation
8	Horn	Hamburg-Mitte	38373	53.553889	10.090000	0	Residential and Business Area	LOW
25	Othmarschen	Altona	15737	53.552778	9.894444	0	Residential and Business Area	LOW
90	Wilstorf	Harburg	17658	53.443611	9.984167	0	Residential and Business Area	LOW
91	Roenneburg	Harburg	3436	53.437500	10.004444	0	Residential and Business Area	LOW
92	Langenbek	Harburg	4038	53.437222	9.986111	0	Residential and Business Area	LOW
94	Marmstorf	Harburg	8960	53.435833	9.968611	0	Residential and Business Area	LOW
95	Eissendorf	Harburg	24999	53.455833	9.954444	0	Residential and Business Area	LOW

5. Discussion

As I mentioned before Hamburg is a one of the largest cities in Germany and it is one of Europe's most important cultural and commercial centers, as well as a major tourist destination.

So, in further studies, it would be interesting to analyze also other data and include them in our previous study. For example, real estate prices, the number of investments made in the development of each neighborhood, etc.

We also used a generic radius of 600 m from the center of each neighborhood exploring the venues, to make it simple, which provides us only an approximate picture. But the neighborhoods have different areas and are differently shaped, so we could work on the method of gathering this information to make it more precise.

This study was only a starting point for more detailed analysis.

6. Conclusion

The purpose of this project was to identify Hamburg areas according to characteristics in order to help investors to narrow down their search for their optimal business location.

The stakeholders can achieve better outcomes through access to such information giving them various options for their purposes by minimizing risks of their investments.

But the final decision should be made based on additional factors like levels of noise, proximity to major roads, real estate availability and prices, social and economic dynamics of every neighborhood, etc.

Not only for investors but also for city managers can make use of these analyses to pay attention to potential problems of the city and make wise decisions of its proper development.

7. References

1. https://de.wikipedia.org/wiki/Liste_der_Bezirke_und_Stadtteile_Hamburgs

2. https://www.polizei.hamburg/contentblob/12289868/49b59e72073b7c5e82c8800d36df8734/data/pks-2018-jahrbuch-do.pdf

3. https://rolbednarz.carto.com/tables/stadtteile_hamburg/public

4. https://developer.foursquare.com

Here is a link to complete project files on Github:

https://github.com/oleksandr-kushnir/Coursera_Capstone

Thank you for your interest!