Coursera_capstone_Week4_part2

May 17, 2020

1 The Battle of the Neighborhoods - (Week 2)

1.1 Sweden, Stockholm - Gym and Fitness offerings - by Tommy Hägvall

1.1.1 A. Introduction & Business problem:

1.1.2 A.1 Background:

Stockholm is the capital of Sweden which is well known for many historian landmarks and highly attractive among tourist with its many islands and its wide archipelages. The history from the Vikings and also heritage of being a monarchy has forged the culture and its people. Sweden represents a modern democratic and overtime has become a multi cultural and inclusive society.

It also has high ambition to create value and make the world a better and a more sustainable place to live. Sweden is also known for being a well advanced high technology innovative capabilities to create generation companies, industries and ecosystem. It is popular and attractive for both investors and also the people and skillset needed.

Though Sweden is a small country with around 10 million people they are known to be very active and succesful in sports and have had a history of being exercising and healthy. Over 2 million people lives in the Region Stockholm that is divided in to 26 municipals. There are also much smaller bourougs and subareas

Many people have over several generations had meberships in gyms and fitness center and the government is also supporting company staff with incentives and also pay a percentage of the gym memberships.

1.1.3 A.2 Description:

Now in these Corona pandemic many industries is having a hard time and had been forced to shutdown and people laydoff due to less customer and lower revenue and income.

The Gym have been affected as well - all had to close down, some has got clearance from the government to re-open again but constrained with many restrictions in atendance volumne and adjusting for Corona.

The number of members that visits the Gyms has drastically decreased, several Gym have had to shutdown completly.

The Government has also informed a public recommendation for people to try to stay fit, keep exercising and to maintain in good health.

Now the helthcare has announced warning for more illness and that more citizens especially younger people is less active and that affects their ability to focus in school and makes them in a less shape and more recepient for increased health issues.

People are trying to find other alternatives to exercise. One could be more outside activities, several clubs are now trying to shift to deliver other services for their members outside in a variety of places where a group can perform their exercises without being exposed to Corona.

1.1.4 A.3 Opportunity

There is a need to adjust the Gym and their offerings and to change their services and also the content and also the skills and number of staff they need.

Since some of them are closing down, or about to close down there is a room for other companies or forces to offer better aligned services to the citizens.

Some Gyms will take mergers and aquisition as an alternative

The Government is willing to invest in alternatives and can fund and innovate where needed but with a service which is more effective without exposing risks to the citizens

1.2 A.4 Target Audience:

The citizens, the governement , existing companies, actors and also new potential future companies are all interested in better insights and knowledge in how the services are offered and consumed differently over the Stockholm Region.

There are differences in Demographics, as well as how much services that are offered as well as potential customers in eq. citizens.

1.3 A.5 The questions asked that needs to be answered are:

Can we access enough data and get insights for understanding the situation in Stockholm, even its risks and opportunities ?

How is the Stockholm Region is shaped geographically and how is it divided in sub-areas with municipals and boroughs ?

What is the current Demographic - number of citizens over the Region ? Due to the islands and the bridges and water Sweden is also known for arranging its geographical areas in Postal Codes.

How does the current Gym alternatives and offerings look like - mapped over the regions and Postal codes ?

How is the offering vs the number of citizens differ between boroughs ?

What kind of Gym types are being offered in the Regions ?

Are there Regions that do not have some of the offerings ?

What is the nr offerings vs nr of consumers in the different boroughs ?

If a company will introduce a new Gym - wich regions is better suited to invest in - that still has a good customer base but is not too saturated with too many competitive offerings.

How are the different Gym companies spread over different regions ?

How many regions have multiple centers from the same Gym company ? Provide a top 3 list of ex. the lack of services or where the availability is lower.

The government and the municipals is also funding and supporting initiatives and investments in more outside activities and gyms that are free for people to use.

Outside Gyms is very popular though Corona but they have same constrained for Corona as the normal Gyms.

Is it possible to see how indivuals rate and like the different gyms ?

1.4 A.6 Success Criteria:

Informative insights and understandings of offerings compared to its potential consumers.

Show Top 3 regions to pick for introducing a new Company and Gym Provider.

Any information in numbers and metrics regarding possible saturation or the opposite opportunity would be a bonus.

1.5 A.7 Disclaimer:

We have no access to each Gyms total perfomance abilities, we can not see their membershipsstatuses, nor their nr of actuall attendes or even the financial outcome.

[ ]:

```

```

1.6 B. Methodology

A number of different mechanism was used to get different data structures from different sources and APIs.

There where a need to cleanse, normalise, correct and get the data in an understandable and usable shape for further processing.

Several mergers and joins of data and using different grouping and segmentation and sorting functionality.

Data was further refines and analysed due to its data and structures. Further data refinement and data wrangling activities needed to be adjusted.

Once data was in a good shape we could start to relate and crossreference data.

We also take advantage of K Cluster Mean grouping of related services in regards to the different areas.

Most of the data and questions stated beforehand as input could then be answered to and also provide insights and conclusions of issues, problems as well as more areas of interest for further exploration

1.7 B.1 Data description - input:

-

1.7.1 Geography logical structure for Stockholm - Importing MS Excel files

Open data not avialable Manual batch process - email request sent to government agency - replied with 2 Excel spreadsheets with Regions and Municipals. Changes in structures of regions, county and boroughs Regions and Municipals data needs to be fixed, cleansed merged and normalised

Data wrangle Remove columns Replace text with new changed texts Normalise Merge and join based on new Region naming

Solid base for plotting out the regions and bouroughs on a map

- ### 1.7.2 Population and density information for Region Stockholm - Screenscraping data from Wikipedia tables

  https://en.wikipedia.org/wiki/Stockholm_County There was some overlapping but different data between this data and also numbers for the population. Here we got area per square kilometer per municipal and aggregated that to region municipal data

- ### 1.7.3 Physical geography data with addresses, postal codes and coordinates for Stockholm - Importing MS Excel files

  Lack of available APIs, a paid batch oriented service exist.

Addresses for Stockholm Found at: URL ……… Downloaded file - 100 000 of records

Plot the zipcodes on the map .. .overlap the region/borough map

It turns out this address file is not needed and this report will be County specific in regards to Gym provided services in the Municipals in Region Stockholm

- ### 1.7.4 Nomanitim - Get Longitutde and Latitude service (coordinates for each municipal)

  We merged the coordinate data with the regional, municipal data with citizen, area and density information of the population

- ### 1.7.5 Foursquare APIs

We use a variety of the APIs to get hold of different data, venue, places, services and users.

- ### 1.7.6 Search & Explore

We will need to iterate through all address data and the addresses and coodinates to crosscheck and ask the Foursquare for a radius range of Gym providers and their offerings

-

1.7.7 Venue

We need to understand more details of each Venue and how they relate to the area and what services they provide. There where to few tips and comments provided by the services and we do not need this information

-

1.7.8 Users

If users have been liking and commenting the Venue services we can highlight those tips , recommendations or comments Since there where to few tips and comments provided by the service we do not need this information

-

1.7.9 Matplotlib, GeoPy and Folium

We show the number of citizen per borough in the region/borough file. We also plot a dempographic presence on a map and population in a shapemap

[ ]:
```

```

[ ]:
```

```

1.8 Fixing the Swedish Kommuner data for Stockholm

MS .xlsx Spreadsheets from SKR - Sveriges Kommuner och Regioner - (former SKL - Sveriges Kommuner och Landsting) (Data from 2020-04-18 but the provided population per municipal data is from 2018-09-30)

Describing the data structures, the dilemma, problem and issues and the solution

[1]:
```
%matplotlib inline
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

[ ]:
```
#1. Read in Kommuner.xls - as Dataframes and also stringify some of the␣
↪important fields
#2  Remove unneccessary columns from both Dataframes
#3. Read in Regioner.xls and Kommuner.xls - as Dataframes and also stringify␣
↪some of the important fields
#4. Remove unneccessary columns from both Dataframes
#5. Munge, prepare and cleanse the data before merging, joining and processing
#6. Fix a lot of data - namechanges kommuner and regions and also relationships␣
↪kommuner related to regions
#7. Save the Dataframes as New_Regioner.xls and New_Kommuner.xls - stored for␣
↪future potential use ...
#8. Merging, joining and processing to a new Dataframe
```

```
#9. Save a New_Kommuner_Regioner.xls file as a source for next step and input␣
↪for other projects
```

[2]:
```
#1.Read in the Kommuner.xls file and make sure the fields with KKod and PNr is␣
↪read as string text otherwise "01" will be 1int
kommuner = pd.read_excel('Data/Kommuner.xls', converters = {'KKod': str,'PNr':␣
↪str })
#pd.read_excel('Regioner.xls', converters = {'OrgKod': str,'PNr': str })
print(kommuner.shape)
kommuner.head(1)
```

```
(290, 11)
```

[2]:
```
   KKod                     Org  Adr    PNr            POrt              Län  \
0  0114  Upplands Väsby kommun  NaN  19480  UPPLANDS VÄSBY  Stockholms län

                                  E-post            Tfn  \
0  upplands.vasby.kommun@upplandsvasby.se  08-590 970 00

                     Hemsida       OrgNr  Inv 30/9 2018
0  http://www.upplandsvasby.se  2120000019          45237
```

[3]:
```
#2. We do not need all 11 columns - let us keep 5 columns - KKod, Org, PNr,␣
↪POrt and Län and drop the other 6 columns
#'Inv 30/9 2018'
kommuner.drop(['Adr', 'E-post', 'Hemsida', 'OrgNr','Tfn'], axis=1, inplace=True)
print(kommuner.shape)
kommuner.head(1)
```

```
(290, 6)
```

[3]:
```
   KKod                     Org    PNr            POrt              Län  \
0  0114  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län

   Inv 30/9 2018
0          45237
```

[4]:
```
#3. Read in the Regioner.xls file and make sure the fields with OrgKod and PNr␣
↪is read as string text otherwise "01" will 1int
regioner = pd.read_excel('Data/Regioner.xls', converters = {'OrgKod': str,'PNr':
↪ str })
print(regioner.shape)
regioner.head(1)
```

```
(21, 9)
```

[4]:   OrgKod   Org   Adr   PNr   POrt \
0   01 Region Stockholm Box 22550 10422 STOCKHOLM

                    E-post   Hemsida   Tfn   OrgNr
0 regionstockholm@sll.se http://www.sll.se 08-737 25 00 2321000016

```
[5]: #4. We do not need all 9 columns - let us keep 4 columns - OrgKod, Org and POrt␣
     ↪and drop the other 5 columns
     regioner.drop(['Adr', 'E-post', 'Hemsida', 'Tfn', 'OrgNr'], axis=1,␣
     ↪inplace=True)
     print(regioner.shape)
     regioner.head(1)
```

(21, 4)

[5]:   OrgKod   Org   PNr   POrt
0   01 Region Stockholm 10422 STOCKHOLM

```
[ ]: # This is not really a good data set and their is no normalised and direct␣
     ↪relationships between "Kommuner" and "Regioner"
     # ##### Just a first recap of the data so far ......
     # Lack of open data  - needs to be fixed !
     # Lack of open data APIs - needs to be fixed !

     # There is nothing in the "Kommuner" that relates directly to "Regioner"
     # There isndirect" link in the old naming, the new naming and also some changes␣
     ↪in segmentation due to mergers of regions
     # "Stockholms län" has become "Region Stockholm"
     # "Jänköpings län" ha nothing in "Regioner" that relates directly to "Kommuner"
     # SKLF has also started to change från "Län", "Landsting" to "Regions"
     # There is an "is become "Region Jänköping"
     # "Västra Götalands län" has become "Västra Götalandsregionen"

     # "Västra Götalandsregionen" is a new name and based on a merge of landsting in␣
     ↪Älvsborg, Skaraborg, Göteborg and Bohuslän.
     # ( Västra Götalands läns landsting where "landsting" in Västra Götalands län )
     # The 1st of januari 2021 the name will be change to "Region Västra Götaland"

     # 290 Kommuner
     # 21 Regioner but up to was before 25 in numbering - now 5 regions are merged

     # We will come back to a similar change for postal codes and its structure and␣
     ↪relationships to other definitions
     # 123 45 - 5 digits in a Swedish postalcode
     # Sometimes with a space between 3rd and 4th digit. This is getting gradually␣
     ↪simplified to not have spaces
```

```
# 2-digit postcode system
# 12x xx - the first 2 digits are a "postort"
# xx3 xx - the 3rd digit is "the form for delivery"
# xxx 45 - the 4th and 5th digit is the "geografic area"

# 3-digit postcode system
# 123 xx - the first 3 digits are "geographical area" for a "postort" ( this␣
↪used to be the post terminal )
# xxx 4x - the 4th digit is "the form for delivery"
# xxx x5 - the 5th digit is the "geografic area"

# There are some exceptions ... all is not covered here
# Stockholm, Göteborg, Malmö has received more and separated companyaddresses␣
↪from other addresses
# 10x xx - Ex Stockholm "Box- och företagsadresser"
# 11x xx - Ex Stockholm "Gatuadresser"

# 5-digit postcode system - not that common - used at smaller areas very far␣
↪and distant

# the postal codes is not easily mapped to areas - same street can have␣
↪different post code numbering
# there are actual streets that exists even in different "Kommuner", "Regioner"␣
↪by spanning the area borders

# geo mapping services have differences
# using Google and iOS geopositioning services we can see that there are␣
↪several zipcode/postalcode fields - these are populated very differently
# they are not that good updated to be trusted as a single source - it is␣
↪improving - it needs to be improved !
# it is easier to range zipcode/postalcodes in mapping tools to streetaddresses␣
↪and coordinates
```

[6]:

```
# We prepare a column that will have the new name for the Region which will␣
↪then become the link and key for relating to Regioner
kommuner['Region'] = kommuner['Län']
```

[7]:

```
#5. Fix the data

# 1 remove "s län" from string1 - changes 220 st of 290 st
kommuner.loc[(kommuner.Region.str.contains('s län')),'Region']=kommuner.Region.
↪str.replace('s län','')
# 2 remove " län" from string1 - changes the rest of 70 st
kommuner.loc[(kommuner.Region.str.contains(' län')),'Region']=kommuner.Region.
↪str.replace(' län','')
# Region Jämtland Härjedalen is new - was before "Jämtlands län"
```

```
kommuner.loc[(kommuner.Region.str.contains('Jämtland')),'Region']=kommuner.
↪Region.str.replace('Jämtland','Jämtland Härjedalen')
# 3 add prefix "Region " before existing text
kommuner.loc[:, 'Region'] = 'Region ' + kommuner['Region']
# Härjedalens kommun är en kommun i Jämtlands län. Delar av landskapen␣
↪Härjedalen, Hälsingland, Dalarna och Jämtland ingår i kommunen
# Exception is Västra Götalandsregionen which was merged from 5 län ( 49 st␣
↪kommuner )
# "Västra Götalandsregionen" kommer att ändras till "Region Västra Götaland" (␣
↪till 2021)
kommuner.loc[(kommuner.Region.str.contains('Region Västra␣
↪Götaland')),'Region']=kommuner.Region.str.replace('Region Västra␣
↪Götaland','Västra Götalandsregionen')

## These are 4 other kommuner which has a Hybrid in their naming compared to␣
↪others - they have " län" as a suffix to new name
# Region Kalmar län
# Region Jönköping län
# Region Södermanland län ->>>> OBS Södermanland <-> Sörmland !!!
# Region Örebro län

kommuner.loc[(kommuner.Region.str.contains('Region Kalmar')),'Region']=kommuner.
↪Region.str.replace('Region Kalmar','Region Kalmar län')
kommuner.loc[(kommuner.Region.str.contains('Region␣
↪Jönköping')),'Region']=kommuner.Region.str.replace('Region␣
↪Jönköping','Region Jönköpings län')
kommuner.loc[(kommuner.Region.str.contains('Region␣
↪Södermanland')),'Region']=kommuner.Region.str.replace('Region␣
↪Södermanland','Region Sörmland')
kommuner.loc[(kommuner.Region.str.contains('Region Örebro')),'Region']=kommuner.
↪Region.str.replace('Region Örebro','Region Örebro län')
```

[8]:
```
kommuner.rename(columns={'Org': 'KOrg', 'PNr': 'KPNr', 'POrt': 'KPOrt', 'Län':␣
↪'KLan', 'Region': 'ROrg', 'Inv 30/9 2018': 'Citizens'}, inplace=True)
kommuner.head(1)
```

[8]:
```
   KKod                    KOrg   KPNr           KPOrt            KLan  \
0  0114  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län

   Citizens               ROrg
0     45237  Region Stockholm
```

[9]:
```
regioner.rename(columns={'OrgKod': 'ROrgKod', 'Org': 'ROrg','PNr': 'RPNr',␣
↪'POrt': 'RPOrt'}, inplace=True)
regioner.head()
```

```
[9]:   ROrgKod                   ROrg   RPNr      RPOrt
     0      01       Region Stockholm  10422  STOCKHOLM
     1      03         Region Uppsala  75125    UPPSALA
     2      04        Region Sörmland  61188   NYKÖPING
     3      05    Region Östergötland  58191  LINKÖPING
     4      06  Region Jönköpings län  55111  JÖNKÖPING
```

```
[ ]: #7. Let us save those to new Excel files - Ny_kommuner.xlsx and Ny_regioner.xlsx
     kommuner.to_excel('Data/Ny_kommuner.xlsx')
     regioner.to_excel('Data/Ny_regioner.xlsx')
```

```
[10]: #8. Merge with inner join to a new Dataframe
      kommuner_region_merged = pd.merge(kommuner, regioner, on='ROrg', how='left')
      kommuner_region_merged.head(1)
```

```
[10]:    KKod                   KOrg   KPNr           KPOrt            KLan  \
      0  0114  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län

         Citizens                ROrg ROrgKod   RPNr      RPOrt
      0     45237  Region Stockholm      01  10422  STOCKHOLM
```

```
[ ]: #9. Let us save a new merged Excel file - Ny_kommuner_region_merged.xlsx
     kommuner_region_merged.to_excel('Data/Ny_kommuner_region_merged.xlsx')
```

```
[11]: stockholm_kommun = kommuner[kommuner.ROrg == 'Region Stockholm']
```

```
[12]: stockholm_kommun = stockholm_kommun.sort_values('KOrg',ascending=True).
       ↪reset_index(drop=True)
```

```
[13]: stockholm_kommun.head(1)
```

```
[13]:    KKod             KOrg   KPNr  KPOrt            KLan  Citizens  \
      0  0127  Botkyrka kommun  14785  TUMBA  Stockholms län     92648

                     ROrg
      0  Region Stockholm
```

```
[ ]: ### Where are almost DONE !!! Data is fixed merged and we have the right␣
      ↪relationships between Kommuner and Regions
```

```
[ ]: #### We need to get the size of each municipal in the region.
     # https://en.wikipedia.org/wiki/Stockholm_County
```

```
[ ]:
```

```
[14]: !pip install beautifulsoup4
```

```
Collecting beautifulsoup4
  Downloading https://files.pythonhosted.org/packages/e8/b5/7bb03a696f2c9b
7af792a8f51b82974e51c268f15e925fc834876a4efa0b/beautifulsoup4-4.9.0-py3-none-
any.whl (109kB)
     |                    | 112kB 5.3MB/s eta 0:00:01
Collecting soupsieve>1.2 (from beautifulsoup4)
  Downloading https://files.pythonhosted.org/packages/6f/8f/457f4a5390eeae1cc3ae
ab89deb7724c965be841ffca6cfca9197482e470/soupsieve-2.0.1-py3-none-any.whl
Installing collected packages: soupsieve, beautifulsoup4
Successfully installed beautifulsoup4-4.9.0 soupsieve-2.0.1
```

[15]:
```
import requests
import json
from bs4 import BeautifulSoup
from IPython.display import Image
from IPython.core.display import HTML
from IPython.display import display_html
```

[16]:
```
url = "https://en.wikipedia.org/wiki/Stockholm_County"
source = requests.get(url).text
soup = BeautifulSoup(source, 'html.parser')
tables = soup.find_all('table', class_='sortable')
```

[ ]:

[ ]:

[17]:
```
# Exploring the table column names
table_headers = []
for row in tables[0].find_all('th'):
    table_headers.append(row.text.strip().replace('[5]','').replace('[6]',''))
    #print(row.text.strip())
print(len(table_headers), "Columns")
print(table_headers)
```

```
3 Columns
['municipality', 'pop. (2018)', 'area/km2']
```

[18]:
```
#df = pd.DataFrame(columns = column_names)
df_wiki = pd.DataFrame(columns = table_headers)

# Search all the postcode, borough, neighborhood
for tr_cell in tables[0].find_all('tr'):
    row_data=[]
    for td_cell in tr_cell.find_all('td'):
        row_data.append(td_cell.text.strip())
    if len(row_data)==3:
```

```
        df_wiki.loc[len(df_wiki)] = row_data
```

[19]:
```
# Upplands Väsby and Upplands-Bro where sorted incorrectly compared to the␣
↪other dataframe sorting
df_wiki = df_wiki.sort_values('municipality',ascending=True).
↪reset_index(drop=True)
```

[20]:
```
df_wiki.shape
```

[20]:
```
(26, 3)
```

[21]:
```
df_wiki.dtypes
```

[21]:
```
municipality    object
pop. (2018)     object
area/km2        object
dtype: object
```

[22]:
```
df_wiki
```

[22]:

	municipality	pop. (2018)	area/km2
0	Botkyrka	93,106	194
1	Danderyd	33,187	26
2	Ekerö	28,308	217
3	Haninge	89,989	458
4	Huddinge	111,722	131
5	Järfälla	78,480	54
6	Lidingö	47,818	31
7	Nacka	103,656	95
8	Norrtälje	61,769	2015
9	Nykvarn	10,923	153
10	Nynäshamn	28,290	359
11	Salem	16,786	54
12	Sigtuna	48,130	328
13	Sollentuna	72,528	53
14	Solna	80,950	19
15	Stockholm	962,154	187
16	Sundbyberg	50,564	9
17	Södertälje	97,381	525
18	Tyresö	48,004	69
19	Täby	71,397	61
20	Upplands Väsby	45,543	75
21	Upplands-Bro	28,756	235
22	Vallentuna	33,432	358
23	Vaxholm	12,023	58
24	Värmdö	44,397	448
25	Österåker	44,831	312

[23]:
```
stockholm_kommun.shape
```

[23]: (26, 7)

[ ]:
```
# The number of rows and order of each municipal is the same - instead of doing␣
↪a merge/join a we did earlier I will just concatenate the rows to the other␣
↪dataframe
# it preserves a very ggod normalisation of the structure with all of the␣
↪different overlapping and changing naming conventions
```

[24]:
```
# Join the two dataframes along columns
stockholm_kommun = pd.concat([stockholm_kommun, df_wiki], axis=1)
```

[25]:
```
stockholm_kommun.dtypes
```

[25]:
```
KKod             object
KOrg             object
KPNr             object
KPOrt            object
KLan             object
Citizens          int64
ROrg             object
municipality     object
pop. (2018)      object
area/km2         object
dtype: object
```

[26]:
```
stockholm_kommun.shape[0]
```

[26]: 26

[27]:
```
stockholm_kommun.loc[:,'pop. (2018)']=stockholm_kommun['pop. (2018)'].str.
↪replace(',','')
```

[28]:
```
stockholm_kommun['pop. (2018)'].astype('int').sum()
# The Wikipedia information shows total population of 2344124
# The offical authority data provided from same time shows total population of␣
↪2336404
# We will use the offical numbers regarding the population
```

[28]: 2344124

[ ]:
```
# density is also very important for analysis - number of citizens per square␣
↪kilometer
```

[29]:
```
# some columns was read in as strings (objects) -> change to integers
total_number_municipals = stockholm_kommun.shape[0]
```

```
total_region_area = stockholm_kommun['area/km2'].astype('int').sum()
total_region_population = stockholm_kommun['Citizens'].astype('int').sum()
total_region_average_density = (stockholm_kommun['Citizens'].astype('int').
↪sum()/stockholm_kommun['area/km2'].astype('int').sum()).round(0).astype(int)
print('{} number of Municipals in the Region Stockholm with an total area of {}␣
↪square kilometers'.format(total_number_municipals,total_region_area))
print('The total population for the Region Stockholm is {} citizens and with a␣
↪average density of {} citizens per square kilometer'.
↪format(total_region_population,total_region_average_density))
```

```
26 number of Municipals in the Region Stockholm with an total area of 6524
square kilometers
The total population for the Region Stockholm is 2336404 citizens and with a
average density of 358 citizens per square kilometer
```

[30]:
```
stockholm_kommun['density'] = (stockholm_kommun['Citizens'].astype('int')/
↪stockholm_kommun['area/km2'].astype('int')).round(0).astype(int)
```

[31]:
```
stockholm_kommun
```

[31]:
```
    KKod                KOrg   KPNr           KPOrt            KLan  \
0   0127     Botkyrka kommun  14785           TUMBA  Stockholms län
1   0162    Danderyds kommun  18205      DJURSHOLM  Stockholms län
2   0125        Ekerö kommun  17823           EKERÖ  Stockholms län
3   0136      Haninge kommun  13681         HANINGE  Stockholms län
4   0126     Huddinge kommun  14185        HUDDINGE  Stockholms län
5   0123     Järfälla kommun  17780        JÄRFÄLLA  Stockholms län
6   0186        Lidingö stad  18182         LIDINGÖ  Stockholms län
7   0182        Nacka kommun  13181           NACKA  Stockholms län
8   0188    Norrtälje kommun  76128       NORRTÄLJE  Stockholms län
9   0140     Nykvarns kommun  15580         NYKVARN  Stockholms län
10  0192   Nynäshamns kommun  14981       NYNÄSHAMN  Stockholms län
11  0128       Salems kommun  14480        RÖNNINGE  Stockholms län
12  0191      Sigtuna kommun  19585          MÄRSTA  Stockholms län
13  0163   Sollentuna kommun  19186      SOLLENTUNA  Stockholms län
14  0184          Solna stad  17186           SOLNA  Stockholms län
15  0180     Stockholms stad  10535       STOCKHOLM  Stockholms län
16  0183    Sundbybergs stad  17292      SUNDBYBERG  Stockholms län
17  0181   Södertälje kommun  15189      SÖDERTÄLJE  Stockholms län
18  0138       Tyresö kommun  13581          TYRESÖ  Stockholms län
19  0160         Täby kommun  18380            TÄBY  Stockholms län
20  0114 Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län
21  0139  Upplands-Bro kommun  19681      KUNGSÄNGEN  Stockholms län
22  0115   Vallentuna kommun  18686      VALLENTUNA  Stockholms län
23  0187        Vaxholms stad  18583         VAXHOLM  Stockholms län
24  0120       Värmdö kommun  13481      GUSTAVSBERG  Stockholms län
25  0117    Österåkers kommun  18486     ÅKERSBERGA  Stockholms län
```

	Citizens	ROrg	municipality	pop. (2018)	area/km2	density
0	92648	Region Stockholm	Botkyrka	93106	194	478
1	33193	Region Stockholm	Danderyd	33187	26	1277
2	28159	Region Stockholm	Ekerö	28308	217	130
3	89625	Region Stockholm	Haninge	89989	458	196
4	111385	Region Stockholm	Huddinge	111722	131	850
5	77922	Region Stockholm	Järfälla	78480	54	1443
6	47712	Region Stockholm	Lidingö	47818	31	1539
7	103191	Region Stockholm	Nacka	103656	95	1086
8	61513	Region Stockholm	Norrtälje	61769	2015	31
9	10841	Region Stockholm	Nykvarn	10923	153	71
10	28159	Region Stockholm	Nynäshamn	28290	359	78
11	16807	Region Stockholm	Salem	16786	54	311
12	47865	Region Stockholm	Sigtuna	48130	328	146
13	72393	Region Stockholm	Sollentuna	72528	53	1366
14	80851	Region Stockholm	Solna	80950	19	4255
15	960031	Region Stockholm	Stockholm	962154	187	5134
16	50205	Region Stockholm	Sundbyberg	50564	9	5578
17	96997	Region Stockholm	Södertälje	97381	525	185
18	47896	Region Stockholm	Tyresö	48004	69	694
19	71186	Region Stockholm	Täby	71397	61	1167
20	45237	Region Stockholm	Upplands Väsby	45543	75	603
21	28566	Region Stockholm	Upplands-Bro	28756	235	122
22	33321	Region Stockholm	Vallentuna	33432	358	93
23	11967	Region Stockholm	Vaxholm	12023	58	206
24	44126	Region Stockholm	Värmdö	44397	448	98
25	44608	Region Stockholm	Österåker	44831	312	143

1.9 Here we see the Wiki related data of the municipal sqare kilometer information and population merged together with regional, municipal data

[32]:
```
!pip install geopandas
```

```
Collecting geopandas
  Downloading https://files.pythonhosted.org/packages/83/c5/3cf9cdc39a6f25
52922f79915f36b45a95b71fd343cfc51170a5b6ddb6e8/geopandas-0.7.0-py2.py3-none-
any.whl (928kB)
     |                  | 931kB 21.7MB/s eta 0:00:01
Collecting shapely (from geopandas)
  Downloading https://files.pythonhosted.org/packages/20/fa/c96d3461fda99e
d8e82ff0b219ac2c8384694b4e640a611a1a8390ecd415/Shapely-1.7.0-cp36-cp36m-manylinu
x1_x86_64.whl (1.8MB)
     |                  | 1.8MB 34.8MB/s eta 0:00:01
Collecting fiona (from geopandas)
  Downloading https://files.pythonhosted.org/packages/ec/20/4e63bc5c6e62df
889297b382c3ccd4a7a488b00946aaaf81a118158c6f09/Fiona-1.8.13.post1-cp36-cp36m-man
```

```
ylinux1_x86_64.whl (14.7MB)
     |                | 14.7MB 1.0MB/s  eta 0:00:01
Collecting pyproj>=2.2.0 (from geopandas)
  Downloading https://files.pythonhosted.org/packages/e5/c3/071e080230ac4b
6c64f1a2e2f9161c9737a2bc7b683d2c90b024825000c0/pyproj-2.6.1.post1-cp36-cp36m-man
ylinux2010_x86_64.whl (10.9MB)
     |                | 10.9MB 35.1MB/s eta 0:00:01
| 3.9MB 35.1MB/s eta 0:00:01
Requirement already satisfied: pandas>=0.23.0 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopandas)
(1.0.3)
Requirement already satisfied: six>=1.7 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (1.14.0)
Collecting cligj>=0.5 (from fiona->geopandas)
  Downloading https://files.pythonhosted.org/packages/e4/be/30a58b4b0733850280d0
1f8bd132591b4668ed5c7046761098d665ac2174/cligj-0.5.0-py3-none-any.whl
Collecting click-plugins>=1.0 (from fiona->geopandas)
  Downloading https://files.pythonhosted.org/packages/e9/da/824b92d9942f4e472702
488857914bdd50f73021efea15b4cad9aca8ecef/click_plugins-1.1.1-py2.py3-none-
any.whl
Collecting munch (from fiona->geopandas)
  Downloading https://files.pythonhosted.org/packages/cc/ab/85d8da5c9a45e072301b
eb37ad7f833cd344e04c817d97e0cc75681d248f/munch-2.5.0-py2.py3-none-any.whl
Collecting click<8,>=4.0 (from fiona->geopandas)
  Downloading https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4
f8d338c2fd15c8d33fdd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl
(82kB)
     |                | 92kB 30.1MB/s eta 0:00:01
Requirement already satisfied: attrs>=17 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (19.3.0)
Requirement already satisfied: pytz>=2017.2 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
pandas>=0.23.0->geopandas) (2020.1)
Requirement already satisfied: python-dateutil>=2.6.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
pandas>=0.23.0->geopandas) (2.8.1)
Requirement already satisfied: numpy>=1.13.3 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
pandas>=0.23.0->geopandas) (1.18.4)
Installing collected packages: shapely, click, cligj, click-plugins, munch,
fiona, pyproj, geopandas
  Found existing installation: pyproj 1.9.6
    Uninstalling pyproj-1.9.6:
      Successfully uninstalled pyproj-1.9.6
Successfully installed click-7.1.2 click-plugins-1.1.1 cligj-0.5.0
fiona-1.8.13.post1 geopandas-0.7.0 munch-2.5.0 pyproj-2.6.1.post1 shapely-1.7.0
```

[33]:
```
import numpy as np
import pandas as pd
import shapefile as shp
import matplotlib.pyplot as plt
import seaborn as sns
import geopandas as gpd
```

[34]:
```
!pip install descartes
```

```
Collecting descartes
  Downloading https://files.pythonhosted.org/packages/e5/b6/1ed2eb03989ae5745846
64985367ba70cd9cf8b32ee8cad0e8aaeac819f3/descartes-1.1.0-py3-none-any.whl
Requirement already satisfied: matplotlib in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from descartes)
(3.1.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
matplotlib->descartes) (2.4.7)
Requirement already satisfied: python-dateutil>=2.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
matplotlib->descartes) (2.8.1)
Requirement already satisfied: numpy>=1.11 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
matplotlib->descartes) (1.18.4)
Requirement already satisfied: kiwisolver>=1.0.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
matplotlib->descartes) (1.2.0)
Requirement already satisfied: cycler>=0.10 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
matplotlib->descartes) (0.10.0)
Requirement already satisfied: six>=1.5 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from python-
dateutil>=2.1->matplotlib->descartes) (1.14.0)
Installing collected packages: descartes
Successfully installed descartes-1.1.0
```

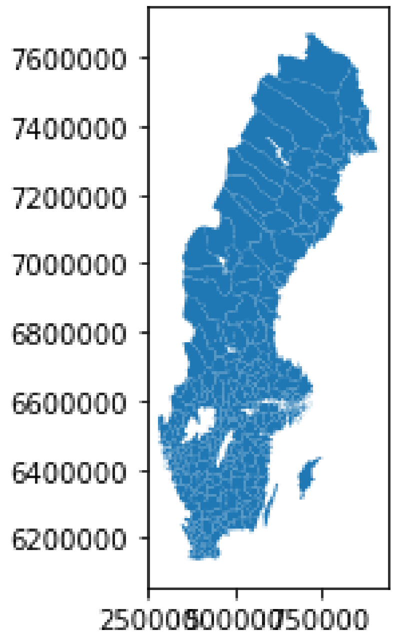
[35]:
```
fname = 'Kommun_Sweref99TM_region.shp'
nil = gpd.read_file(fname)
nil.head()
```

[35]:

	KnKod	KnNamn	geometry
0	0114	Upplands Väsby	POLYGON ((665740.728 6599291.303, 664510.300 6…
1	0115	Vallentuna	POLYGON ((682869.466 6601057.394, 682007.866 6…
2	0117	Österåker	MULTIPOLYGON (((702182.380 6606673.621, 700428…
3	0120	Värmdö	MULTIPOLYGON (((697991.594 6577581.542, 699316…
4	0123	Järfälla	POLYGON ((658883.748 6594701.728, 659038.094 6…

[190]:
```
nil.plot()
```

[190]: <matplotlib.axes._subplots.AxesSubplot at 0x7f439c66add8>



1.10 Here is the Geography data set showing Sweden

[ ]:
```
# This is way to much  - we onky need the Region Stockholm - we will later on␣
↪when we merge and joing data just focus on the rows for Stockholm Region
```

[37]:
```
nil.rename(columns={'KnKod': 'KKod'}, inplace=True)
nil.head()
```

[37]:

	KKod	KnNamn	geometry
0	0114	Upplands Väsby	POLYGON ((665740.728 6599291.303, 664510.300 6…
1	0115	Vallentuna	POLYGON ((682869.466 6601057.394, 682007.866 6…
2	0117	Österåker	MULTIPOLYGON (((702182.380 6606673.621, 700428…
3	0120	Värmdö	MULTIPOLYGON (((697991.594 6577581.542, 699316…
4	0123	Järfälla	POLYGON ((658883.748 6594701.728, 659038.094 6…

[38]:
```
#8. Merge with inner join to a new Dataframe
kommuner_region_merged = pd.merge(stockholm_kommun, nil, on='KKod', how='left')
kommuner_region_merged.head(1)
```

[38]:

	KKod	KOrg	KPNr	KPOrt	KLan	Citizens	\
0	0127	Botkyrka kommun	14785	TUMBA	Stockholms län	92648	

```
            ROrg municipality pop. (2018) area/km2  density    KnNamn  \
0  Region Stockholm     Botkyrka       93106      194      478  Botkyrka

                                             geometry
0  POLYGON ((663734.053 6553815.790, 660815.535 6…
```

[39]:

```
#8. Merge with inner join to a new Dataframe
geo_kommuner_region_merged = pd.merge(nil, stockholm_kommun, on='KKod',␣
↪how='right')
geo_kommuner_region_merged.head(1)
```

[39]:

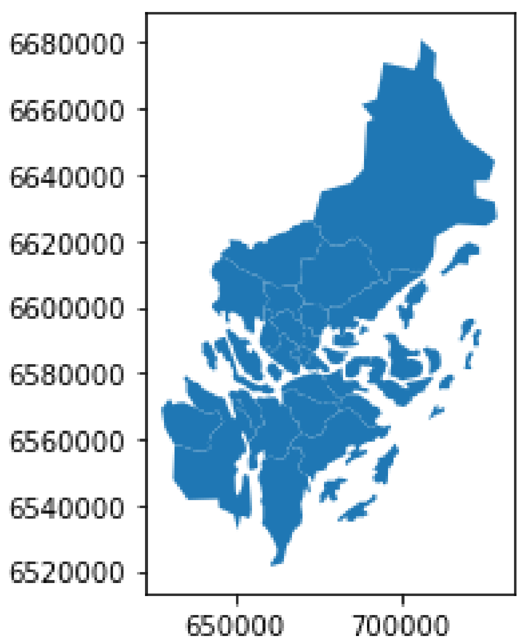
```
   KKod          KnNamn                                          geometry  \
0  0114  Upplands Väsby  POLYGON ((665740.728 6599291.303, 664510.300 6…

                 KOrg   KPNr           KPOrt           KLan  Citizens  \
0  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län     45237

            ROrg    municipality pop. (2018) area/km2  density
0  Region Stockholm  Upplands Väsby       45543       75      603
```

[ ]:

[40]:

```
geo_kommuner_region_merged.plot()
```

[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f4410e7a8d0>



1.11 Here we have filtered down the Geography set to Region Stockholm

[ ]:

[41]:
```python
type(geo_kommuner_region_merged)
```

[41]:
```
geopandas.geodataframe.GeoDataFrame
```

[ ]:

[188]:
```python
# set a variable that will call whatever column we want to visualise on the map
#variable = 'Citizens'
# set the range for the choropleth
#vmin, vmax = 120, 220
# create figure and axes for Matplotlib
#fig, ax = plt.subplots(1, figsize=(10, 6))
```

[44]:
```python
geo_kommuner_region_merged.head()
```

[44]:
```
   KKod          KnNamn                                          geometry  \
0  0114  Upplands Väsby  POLYGON ((665740.728 6599291.303, 664510.300 6…
1  0115      Vallentuna  POLYGON ((682869.466 6601057.394, 682007.866 6…
2  0117       Österåker  MULTIPOLYGON (((702182.380 6606673.621, 700428…
3  0120          Värmdö  MULTIPOLYGON (((697991.594 6577581.542, 699316…
4  0123        Järfälla  POLYGON ((658883.748 6594701.728, 659038.094 6…

                      KOrg   KPNr           KPOrt             KLan  Citizens  \
0  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län     45237
1      Vallentuna kommun  18686      VALLENTUNA  Stockholms län     33321
2      Österåkers kommun  18486      ÅKERSBERGA  Stockholms län     44608
3          Värmdö kommun  13481     GUSTAVSBERG  Stockholms län     44126
4        Järfälla kommun  17780        JÄRFÄLLA  Stockholms län     77922

               ROrg    municipality pop. (2018) area/km2  density
0  Region Stockholm  Upplands Väsby       45543       75      603
1  Region Stockholm      Vallentuna       33432      358       93
2  Region Stockholm       Österåker       44831      312      143
3  Region Stockholm          Värmdö       44397      448       98
4  Region Stockholm        Järfälla       78480       54     1443
```

[45]:
```python
print(geo_kommuner_region_merged['geometry'][0])
```

```
POLYGON ((665740.7282246178 6599291.303340398, 664510.2998304499
6598357.503412673, 663103.1465013289 6598472.370031904, 662872.0154857106
6597499.728010375, 660764.3199627302 6596518.267783766, 660959.0125924041
6598925.21785604, 658867.0501403406 6599774.614963715, 659198.5578185747
```

```
6600797.466765189, 660059.5971258126 6601767.759264957, 660177.4714700246
6602516.066949092, 659007.715998187 6604238.526347462, 658574.3144219866
6605511.017335602, 658715.0792299644 6607417.389299102, 661254.6838313277
6607002.383772276, 661803.7086362473 6606253.196267227, 662948.5286749337
6605840.200331355, 665472.4084512243 6606471.811768673, 667507.9280990815
6606318.61295185, 668411.8375508145 6606641.557217479, 668727.9255002149
6606054.506709912, 668240.0188932177 6605232.713983444, 670622.3422604781
6602461.158125624, 670983.7493701297 6601600.703268722, 670977.548495219
6600877.740425717, 669332.5685249114 6599172.027618626, 670057.8400058148
6597413.135636003, 669820.4339026872 6595805.512862969, 668821.2189825805
6595919.339693847, 667901.5269848648 6596482.065152707, 667984.867733165
6597603.896807118, 667140.6252107582 6598540.466171132, 666365.4581260195
6598729.007794031, 665740.7282246178 6599291.303340398))
```

[46]:
```
!pip install geopy
```

```
Collecting geopy
  Downloading https://files.pythonhosted.org/packages/ab/97/25def417bf5db4
cc6b89b47a56961b893d4ee4fec0c335f5b9476a8ff153/geopy-1.22.0-py2.py3-none-any.whl
(113kB)
     |                  | 122kB 6.4MB/s eta 0:00:01
Collecting geographiclib<2,>=1.49 (from geopy)
  Downloading https://files.pythonhosted.org/packages/8b/62/26ec95a98ba642991631
99e95ad1b0e34ad3f4e176e221c40245f211e425/geographiclib-1.50-py3-none-any.whl
Installing collected packages: geographiclib, geopy
Successfully installed geographiclib-1.50 geopy-1.22.0
```

[47]:
```
import time
import geopy
from geopy.geocoders import Nominatim
```

[48]:
```
geo_kommuner_region_merged['lat']=0
```

[49]:
```
geo_kommuner_region_merged['lon']=0
```

[50]:
```
for ind in geo_kommuner_region_merged.index:
    print(ind)
    geolocator = Nominatim(user_agent="foursquare_agent")
    location = geolocator.geocode(geo_kommuner_region_merged['KOrg'][ind])
    geo_kommuner_region_merged.loc[ind,'lat'] = location.latitude
    geo_kommuner_region_merged.loc[ind,'lon'] = location.longitude
    time.sleep(2)
```

```
0
1
2
3
4
```

```
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
```

[51]:
```python
geo_kommuner_region_merged.head(1)
```

[51]:
```
   KKod          KnNamn                                          geometry  \
0  0114  Upplands Väsby  POLYGON ((665740.728 6599291.303, 664510.300 6…

                   KOrg   KPNr           KPOrt            KLan  Citizens  \
0  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län     45237

             ROrg    municipality pop. (2018) area/km2  density        lat  \
0  Region Stockholm  Upplands Väsby       45543       75      603  59.516693

        lon
0  17.91566
```

1.12 Here we see the Longitude, Latitude data from GeoPy, Nominatim merged with the other existing data

[53]:
```python
geo_kommuner_region_merged['lon'].values
```

[53]:
```
array([17.91565995, 18.22121967, 18.59772226, 19.05742161, 17.82201282,
       17.60780045, 18.05343229, 17.86017305, 17.7283887 , 18.45467268,
       18.34325578, 17.67386182, 17.40808067, 18.0708782 , 18.06192264,
       17.94001845, 18.1032723 , 17.49085301, 18.22503457, 18.00166496,
       18.013991  , 18.133142  , 18.331331  , 18.7       , 17.84374997,
       17.89202337])
```

[ ]:
```
print(np.array(geo_kommuner_region_merged['lon']))
```

[54]:
```
nil.crs
```

[54]:
```
<Projected CRS: EPSG:3006>
Name: SWEREF99 TM
Axis Info [cartesian]:
- N[north]: Northing (metre)
- E[east]: Easting (metre)
Area of Use:
- name: Sweden
- bounds: (10.03, 54.96, 24.17, 69.07)
Coordinate Operation:
- name: SWEREF99 TM
- method: Transverse Mercator
Datum: SWEREF99
- Ellipsoid: GRS 1980
- Prime Meridian: Greenwich
```

[55]:
```
geo_kommuner_region_merged.head(1)
```

[55]:
```
   KKod          KnNamn                                          geometry  \
0  0114  Upplands Väsby  POLYGON ((665740.728 6599291.303, 664510.300 6…

                    KOrg   KPNr           KPOrt           KLan  Citizens  \
0  Upplands Väsby kommun  19480  UPPLANDS VÄSBY  Stockholms län     45237

               ROrg    municipality pop. (2018)  area/km2  density        lat  \
0  Region Stockholm  Upplands Väsby       45543        75      603  59.516693

        lon
0  17.91566
```

[56]:
```
address = 'Botkyrka kommun, Stockholm'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
59.1565469 17.860173050874476
```

[57]:
```
address = 'Stockholm Kommun'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
```

```
print(latitude, longitude)
```

59.3251172 18.0710935

[58]:
```
address = 'Salems kommun'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

59.23870475 17.728388699012335

[59]:
```
address = 'Värmdö kommun'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

59.1822255 19.057421611711707

[ ]:
```
# Salems kommun - the coordinates are incorrect
# The right are - 59.229354, 17.695501  - to the center

# Botkyrka kommun the coordinates are incorrect
# The right are - 59.161714, 17.843802 - to the center
```

[60]:
```
#address = '102 North End Ave, New York, NY'
address = 'Haninge kommun'
geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

58.90266925 18.454672677008574

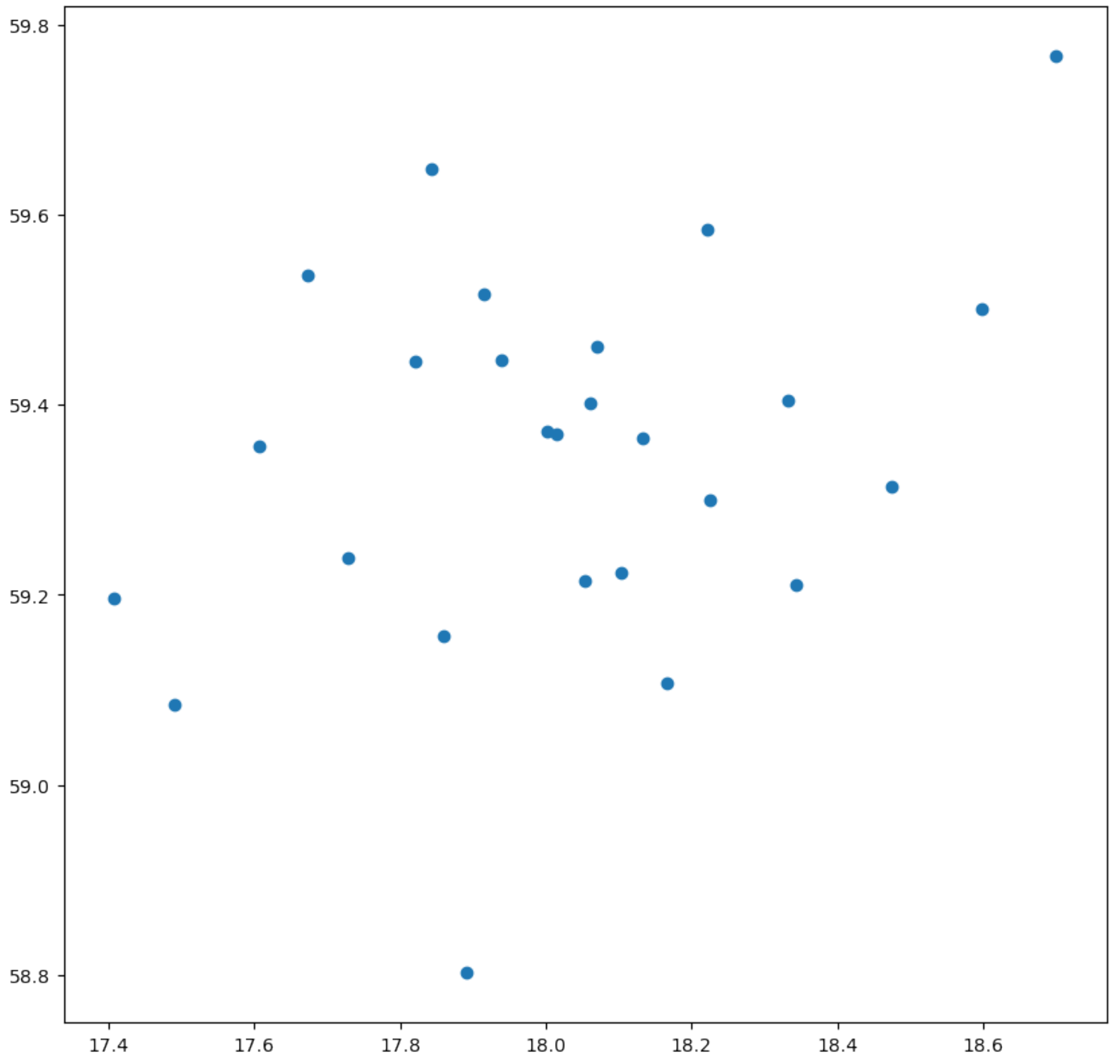
[61]:
```
# Haninge kommun gives incorrectly 58.90266925 18.454672677008574 -> Out in the␣
 ↪water !
# Should be 59.107406, 18.165635
geo_kommuner_region_merged.loc[(geo_kommuner_region_merged.KOrg.str.
 ↪contains('Haninge kommun'), 'lat')] =59.107406
geo_kommuner_region_merged.loc[(geo_kommuner_region_merged.KOrg.str.
 ↪contains('Haninge kommun'), 'lon')] =18.165635

# Värmdö kommun gives incorrectly 59.107406, 18.165635 -> Out in the water !
```

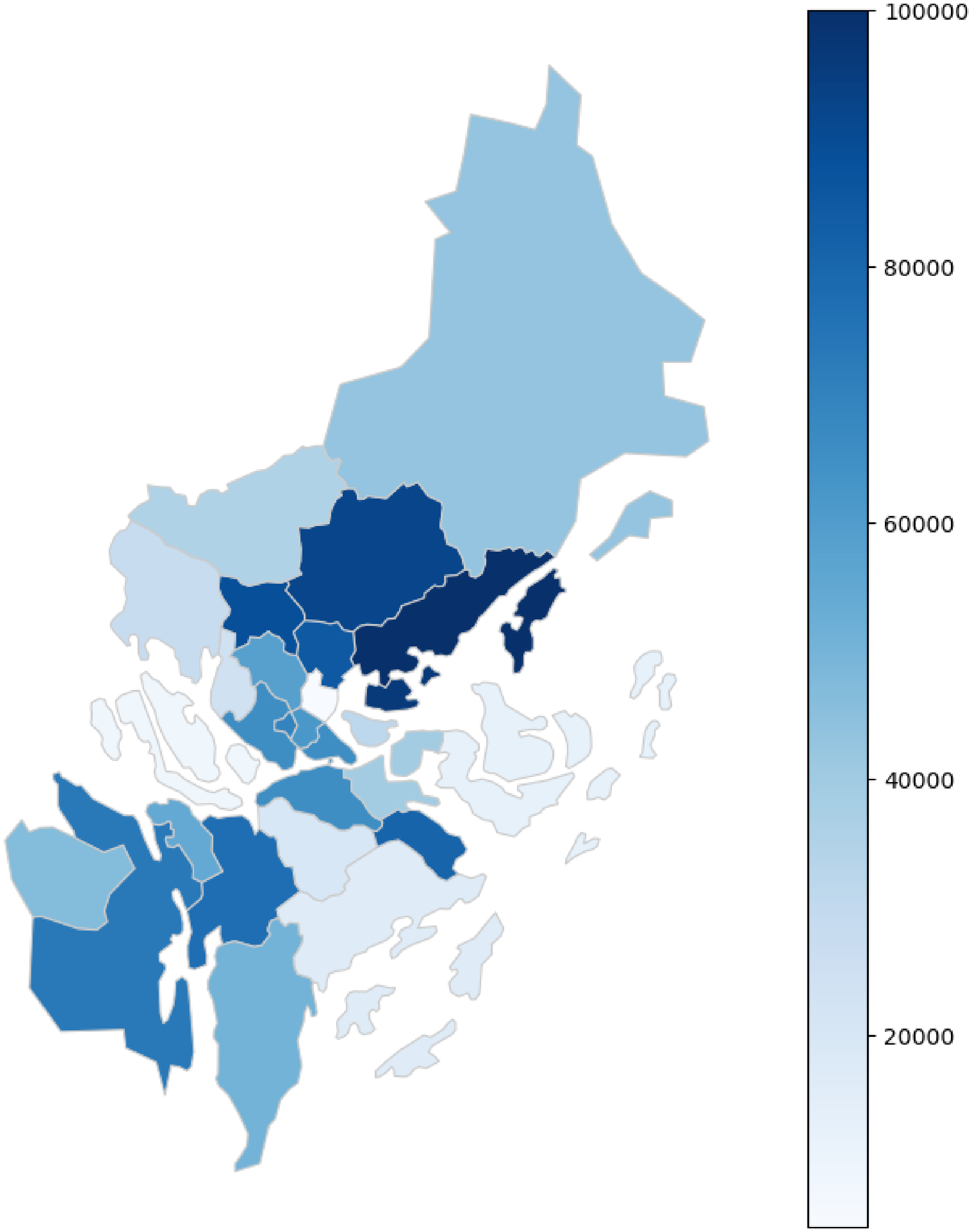
```
# Should be 59.313522, 18.473878
geo_kommuner_region_merged.loc[(geo_kommuner_region_merged.KOrg.str.
↪contains('Värmdö kommun'), 'lat')] =59.313522
geo_kommuner_region_merged.loc[(geo_kommuner_region_merged.KOrg.str.
↪contains('Värmdö kommun'), 'lon')] =18.473878
```

[62]:

```
# set a variable that will call whatever column we want to visualise on the map
variable = 'KPOrt'
# set the range for the choropleth
vmin, vmax = 5000, 100000
# create figure and axes for Matplotlib

#fig = plt.figure()
#ax = plt.axes()
#plt.axis([15,56,21,61])

fig2, ax2 = plt.subplots(1, figsize=(10, 10), dpi=100)
#Ax2=testing.fig2([0,0,70,20])
lat_x  = np.array(geo_kommuner_region_merged['lat'], dtype=float)
lon_x = np.array(geo_kommuner_region_merged['lon'], dtype=float)
ax2.scatter(lon_x, lat_x)

fig, ax = plt.subplots(1, figsize=(10, 10), dpi=100)
#fig.add_axes([0,0,70,20])
# remove the axis
#ax.set_title('Population of Stockholm, Sweden', fontdict={'fontsize': '25',␣
↪'fontweight' : '3'})
#plt.xticks([15,16,17,19,20,21])
ax.axis('off')
# create an annotation for the data source
#ax.annotate('Source: Tommy Hägvall, 2020',xy=(0.1, .08),  xycoords='figure␣
↪fraction', horizontalalignment='right', verticalalignment='top',␣
↪fontsize=18, color='#555555')

# Create colorbar as a legend
sm = plt.cm.ScalarMappable(cmap='Blues', norm=plt.Normalize(vmin=vmin,␣
↪vmax=vmax))
# empty array for the data range
sm._A = []
# add the colorbar to the figure
cbar = fig.colorbar(sm)

#test_axes = ([15,16,17,19,20,21])
#ax.set_xlim(15,20)
#ax.set_ylim(56,61)
#fig.add_axes([0,0,70,20])
```

```
# create map
#ax.plot(kind="scatter", x="lon", y="lat", alpha=0.4)
#geo_kommuner_region_merged.plot(column=variable, cmap='Blues', linewidth=0.8,␣
↪ax=ax, edgecolor='0.8')
geo_kommuner_region_merged.plot(column=variable, cmap='Blues', linewidth=0.8,␣
↪ax=ax, edgecolor='0.8')
#lat_x  = np.array(geo_kommuner_region_merged['lat'], dtype=float)
#lon_x = np.array(geo_kommuner_region_merged['lon'], dtype=float)
#ax.scatter(lon_x, lat_x)

plt.plot()
plt.show()
```




100000
80000
60000
40000
20000

2 Here we show the population over the different areas in the region

[ ]:

```

```

[64]:

```
!pip install folium
```

```
Requirement already satisfied: folium in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.5.0)
Requirement already satisfied: requests in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium)
(2.23.0)
Requirement already satisfied: six in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium)
(1.14.0)
Requirement already satisfied: branca in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium)
(0.4.1)
Requirement already satisfied: jinja2 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from folium)
(2.11.2)
Requirement already satisfied: chardet<4,>=3.0.2 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->folium) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->folium) (2020.4.5.1)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->folium) (1.25.9)
Requirement already satisfied: idna<3,>=2.5 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
requests->folium) (2.9)
Requirement already satisfied: MarkupSafe>=0.23 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
jinja2->folium) (1.1.1)
```

[65]:

```
import folium
```

[66]:

```
# create map of New York using latitude and longitude values
map_stockholm = folium.Map(location=[latitude, longitude], zoom_start=8)

# add markers to map
for lat, lon, KOrg, KnNamn in zip(geo_kommuner_region_merged['lat'],␣
 ↪geo_kommuner_region_merged['lon'], geo_kommuner_region_merged['KOrg'],␣
 ↪geo_kommuner_region_merged['KnNamn']):
    label = '{}, {}'.format(KnNamn, KOrg)
```

```
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_stockholm)

map_stockholm
```

[66]:
```
<folium.folium.Map at 0x7f44029ed160>
```

2.1 Here we show the centers of the different municipals in the region

[102]:
```
population_stockholm = geo_kommuner_region_merged.
↪sort_values('Citizens',ascending=False).reset_index(drop=True)
```

[103]:
```
population_stockholm.head(3)
```

[103]:
```
   KKod     KnNamn                                           geometry  \
0  0180  Stockholm  MULTIPOLYGON (((674490.509 6580174.564, 674492…
1  0126   Huddinge  POLYGON ((674526.568 6571525.665, 675888.254 6…
2  0182      Nacka  MULTIPOLYGON (((680137.107 6573386.446, 679296…

             KOrg   KPNr      KPOrt            KLan  Citizens  \
0  Stockholms stad  10535  STOCKHOLM  Stockholms län    960031
1  Huddinge kommun  14185   HUDDINGE  Stockholms län    111385
2     Nacka kommun  13181      NACKA  Stockholms län    103191

              ROrg municipality pop. (2018) area/km2  density        lat  \
0  Region Stockholm    Stockholm      962154      187     5134  59.222639
1  Region Stockholm     Huddinge      111722      131      850  59.214934
2  Region Stockholm        Nacka      103656       95     1086  59.299990

         lon
0  18.103272
1  18.053432
2  18.225035
```

[73]:
```
population_stockholm.describe()
```

[73]:
```
            Citizens      density        lat        lon
count      26.000000    26.000000  26.000000  26.000000
mean     89861.692308  1049.230769  59.347010  18.026115
```

```
std    179778.681261  1543.433894   0.200036   0.315632
min     10841.000000    31.000000  58.802419  17.408081
25%     33225.000000   133.250000  59.216860  17.847856
50%     47880.500000   394.500000  59.366864  18.033712
75%     80118.750000  1249.500000  59.457919  18.207324
max    960031.000000  5578.000000  59.766667  18.700000
```

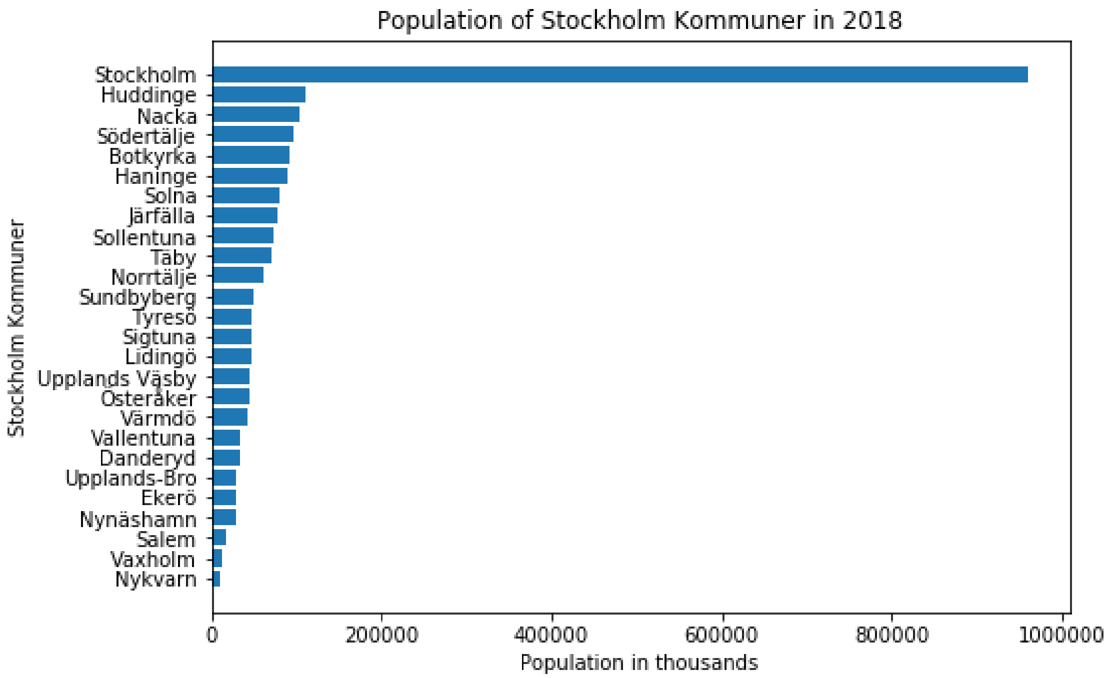
[115]:
```
population_stockholm = geo_kommuner_region_merged.
 ↪sort_values('Citizens',ascending=False).reset_index(drop=True)
```

[116]:
```
population_stockholm.head(3)
```

[116]:
```
   KKod     KnNamn                                           geometry  \
0  0180  Stockholm  MULTIPOLYGON (((674490.509 6580174.564, 674492…
1  0126   Huddinge  POLYGON ((674526.568 6571525.665, 675888.254 6…
2  0182      Nacka  MULTIPOLYGON (((680137.107 6573386.446, 679296…

              KOrg   KPNr      KPOrt            KLan  Citizens  \
0  Stockholms stad  10535  STOCKHOLM  Stockholms län    960031
1  Huddinge kommun  14185   HUDDINGE  Stockholms län    111385
2     Nacka kommun  13181      NACKA  Stockholms län    103191

                RОrg municipality pop. (2018) area/km2  density        lat  \
0  Region Stockholm    Stockholm      962154       187     5134  59.222639
1  Region Stockholm     Huddinge      111722       131      850  59.214934
2  Region Stockholm        Nacka      103656        95     1086  59.299990

         lon
0  18.103272
1  18.053432
2  18.225035
```

[117]:
```
population_stockholm = population_stockholm.
 ↪sort_values('Citizens',ascending=True).reset_index(drop=True)
```

[191]:
```
import matplotlib.pyplot as plt
fig = plt.figure()

#plt.subplots(1, figsize=(10, 10), dpi=100)
ax = fig.add_axes([0,0,1,1])

ax.barh(population_stockholm['KnNamn'],population_stockholm['Citizens'])
plt.xlabel('Population in thousands')
plt.ylabel('Stockholm Kommuner')
plt.title('Population of Stockholm Kommuner in 2018')

plt.show()
```



2.2 Here we show the population of Stockholm among the different municipals

Almost half of the citizens live in Stockholm main area

[121]:
```
density_stockholm = geo_kommuner_region_merged.
↪sort_values('density',ascending=False).reset_index(drop=True)
```

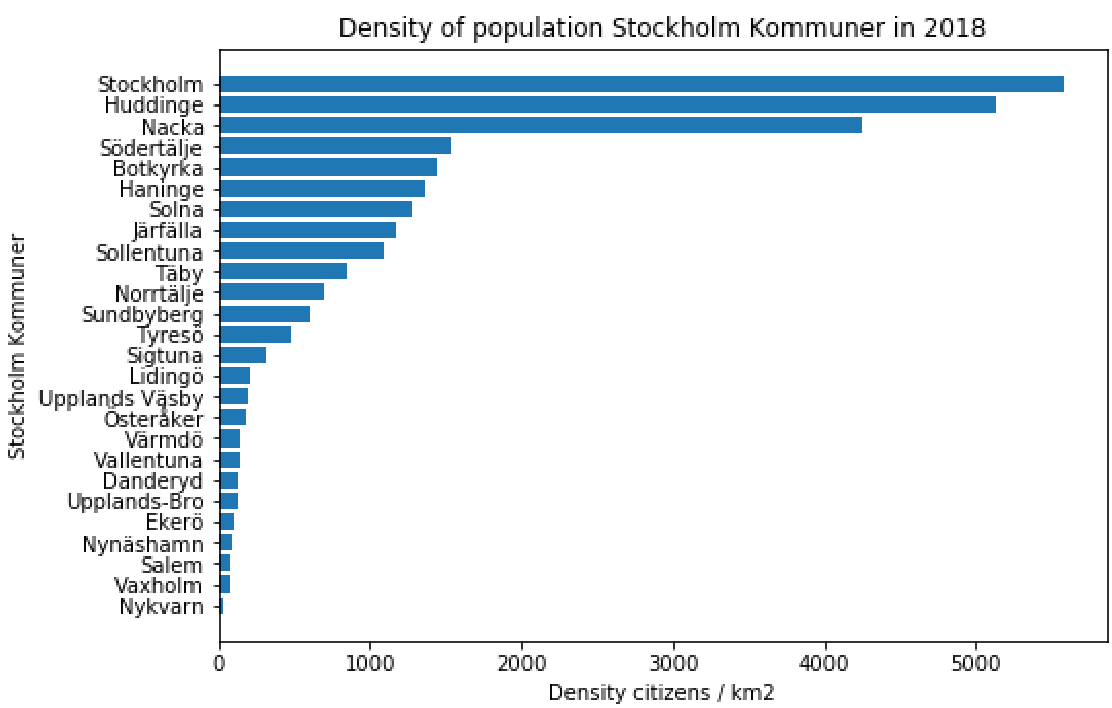
[123]:
```
density_stockholm.head(3)
```

[123]:
```
   KKod      KnNamn                                          geometry  \
0  0183  Sundbyberg  POLYGON ((668201.758 6584775.698, 666575.356 6…
1  0180   Stockholm  MULTIPOLYGON (((674490.509 6580174.564, 674492…
2  0184       Solna  POLYGON ((672828.724 6584467.951, 672974.337 6…

              KOrg   KPNr       KPOrt            KLan  Citizens  \
0  Sundbybergs stad  17292  SUNDBYBERG  Stockholms län     50205
1   Stockholms stad  10535   STOCKHOLM  Stockholms län    960031
2        Solna stad  17186       SOLNA  Stockholms län     80851

              ROrg municipality pop. (2018) area/km2  density        lat  \
0  Region Stockholm   Sundbyberg      50564        9     5578  59.372504
1  Region Stockholm    Stockholm     962154      187     5134  59.222639
2  Region Stockholm        Solna      80950       19     4255  59.369343

         lon
```

```
0   18.001665
1   18.103272
2   18.013991
```

[124]:
```
density_stockholm = geo_kommuner_region_merged.
 ↪sort_values('density',ascending=True).reset_index(drop=True)
```

[125]:
```
import matplotlib.pyplot as plt
fig = plt.figure()

#plt.subplots(1, figsize=(10, 10), dpi=100)
ax = fig.add_axes([0,0,1,1])

ax.barh(population_stockholm['KnNamn'],density_stockholm['density'])
plt.xlabel('Density citizens / km2')
plt.ylabel('Stockholm Kommuner')
plt.title('Density of population Stockholm Kommuner in 2018')

plt.show()
```



2.3 Here we show the density and the number of citizens per square kilometer

There are much more citizens per square kilometer in mainly 3 large areas

[126]:
```
CLIENT_ID = 'JPNXEO4N4LG1H3XMK01IAKNRFHOZU4XQHROSJX3WY0PTVKJI' # your␣
↪Foursquare ID
CLIENT_SECRET = 'XYSTHZ44KCYRFUXKSBBENHM5ARXFGXFE2A3JSDRUOJOSZX4Z' # your␣
↪Foursquare Secret
VERSION = '20180605' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: JPNXEO4N4LG1H3XMK01IAKNRFHOZU4XQHROSJX3WY0PTVKJI
CLIENT_SECRET:XYSTHZ44KCYRFUXKSBBENHM5ARXFGXFE2A3JSDRUOJOSZX4Z
```

[ ]:
```
population_stockholm.loc[16, 'KOrg']
```

[127]:
```
neighborhood_latitude = population_stockholm.loc[16, 'lat'] # neighborhood␣
↪latitude value
neighborhood_longitude = population_stockholm.loc[16, 'lon'] # neighborhood␣
↪longitude value

neighborhood_name = population_stockholm.loc[16, 'KOrg'] # neighborhood name

print('Latitude and longitude values of {} are {}, {}.'.
↪format(neighborhood_name,
                                                                           ␣
↪neighborhood_latitude,
                                                                           ␣
↪neighborhood_longitude))
```

```
Latitude and longitude values of Täby kommun are 59.4615531, 18.0708782046907.
```

[ ]:
```
# type your answer here

LIMIT = 100 # limit of number of venues returned by Foursquare API

radius = 5000
search_query = 'Gym / Fitness Center'

#url = 'https://api.foursquare.com/v2/venues/explore?
↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
## No hits at all using search for any of the coordinates
url = 'https://api.foursquare.com/v2/venues/explore?
↪client_id={}&client_secret={}&ll={},{}&v={}&query={}&radius={}&limit={}'.
↪format(
    CLIENT_ID, CLIENT_SECRET, neighborhood_latitude, neighborhood_longitude,␣
↪VERSION, search_query, radius, LIMIT)
```

```
url # display URL
```

[128]:
```
import json
import requests
from pandas.io.json import json_normalize
```

[ ]:
```
#results = requests.get(url).json()
#results
```

[ ]:
```
# assign relevant part of JSON to venues items
#venues = results['response']['groups'][0]['items']

# tranform venues into a dataframe
#dataframe = json_normalize(venues)
#dataframe.head()
```

[ ]:
```
#dataframe.info()
```

[ ]:
```
#results['response']['groups'][0]['items']
```

[ ]:
```
# print('{} venues were returned by Foursquare.'.format(dataframe.shape[0]))
```

[ ]:
```
#dataframe.groupby('venue.location.city').count()
```

[ ]:
```

```

[129]:
```
LIMIT = 1000 # limit of number of venues returned by Foursquare API

radius = 5000
search_query = 'Gym / Fitness Center'
```

[130]:
```
def getNearbyVenues(names, latitudes, longitudes, radius=5000):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?
↪&client_id={}&client_secret={}&v={}&ll={},{}&query={}&radius={}&limit={}'.
↪format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
```

```
            lng,
            search_query,
            radius,
            LIMIT)

        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']

        # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

    nearby_venues = pd.DataFrame([item for venue_list in venues_list for item␣
↪in venue_list])
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

[131]:
```
# type your answer here

stockholm_venues = getNearbyVenues(names=population_stockholm['KnNamn'],
                                   latitudes=population_stockholm['lat'],
                                   longitudes=population_stockholm['lon']

                                  )
```

```
Nykvarn
Vaxholm
Salem
Nynäshamn
Ekerö
Upplands-Bro
Danderyd
Vallentuna
Värmdö
```

```
Österåker
Upplands Väsby
Lidingö
Sigtuna
Tyresö
Sundbyberg
Norrtälje
Täby
Sollentuna
Järfälla
Solna
Haninge
Botkyrka
Södertälje
Nacka
Huddinge
Stockholm
```

[ ]:

```

```

[140]:

```
print(stockholm_venues.shape)
stockholm_venues
#.sort_values('Neighborhood',ascending=True)
```

```
(407, 7)
```

[140]:

```
    Neighborhood  Neighborhood Latitude  Neighborhood Longitude  \
0        Nykvarn              59.196527               17.408081
1        Vaxholm              59.404060               18.331331
2        Vaxholm              59.404060               18.331331
3        Vaxholm              59.404060               18.331331
4        Vaxholm              59.404060               18.331331
..           …                    …                      …
402    Stockholm              59.222639               18.103272
403    Stockholm              59.222639               18.103272
404    Stockholm              59.222639               18.103272
405    Stockholm              59.222639               18.103272
406    Stockholm              59.222639               18.103272

                          Venue  Venue Latitude  Venue Longitude  \
0                The Alfort gym       59.183253        17.440904
1      Niana Fitness Vaxö Skola       59.403463        18.358818
2                        Fysio+       59.402574        18.340298
3                Accédo Fons HB       59.393115        18.279122
4                 Danderyds Gym       59.433331        18.319870
..                          …             …               …
402              Puls & Träning       59.185581        18.136363
```

```
403              Puls & Träning       59.262810        18.082840
404  Elit Sports Club Skarpnäck       59.265069        18.126948
405              Nordic Wellness       59.242067        18.090105
406                     Farsta IP       59.240071        18.082942

           Venue Category
0                     Gym
1     Gym / Fitness Center
2                     Gym
3     Gym / Fitness Center
4                     Gym
..                    …
402   Gym / Fitness Center
403   Gym / Fitness Center
404                   Gym
405   Gym / Fitness Center
406                 Track

[407 rows x 7 columns]
```

2.4 Here we see a variety of different Gym related categories

[143]:
```
stockholm_venues.groupby('Neighborhood').count().
 ↪sort_values('Venue',ascending=False)
```

[143]:
```
                 Neighborhood Latitude  Neighborhood Longitude  Venue  \
Neighborhood
Solna                               95                      95     95
Sundbyberg                          90                      90     90
Lidingö                             55                      55     55
Danderyd                            34                      34     34
Sollentuna                          22                      22     22
Täby                                17                      17     17
Stockholm                           14                      14     14
Huddinge                            13                      13     13
Nacka                               13                      13     13
Järfälla                            12                      12     12
Upplands Väsby                       7                       7      7
Sigtuna                              7                       7      7
Vaxholm                              5                       5      5
Värmdö                               5                       5      5
Norrtälje                            4                       4      4
Upplands-Bro                         3                       3      3
Haninge                              3                       3      3
Botkyrka                             3                       3      3
Salem                                2                       2      2
Tyresö                               1                       1      1
```

```
Nykvarn                                  1                        1      1
Vallentuna                               1                        1      1

                 Venue Latitude  Venue Longitude  Venue Category
Neighborhood
Solna                        95               95              95
Sundbyberg                   90               90              90
Lidingö                      55               55              55
Danderyd                     34               34              34
Sollentuna                   22               22              22
Täby                         17               17              17
Stockholm                    14               14              14
Huddinge                     13               13              13
Nacka                        13               13              13
Järfälla                     12               12              12
Upplands Väsby                7                7               7
Sigtuna                       7                7               7
Vaxholm                       5                5               5
Värmdö                        5                5               5
Norrtälje                     4                4               4
Upplands-Bro                  3                3               3
Haninge                       3                3               3
Botkyrka                      3                3               3
Salem                         2                2               2
Tyresö                        1                1               1
Nykvarn                       1                1               1
Vallentuna                    1                1               1
```

2.5 Here we see the number of Gym related sevices being offered in different municipals

[144]:

```
print('There are {} uniques categories.'.format(len(stockholm_venues['Venue␣
↪Category'].unique())))
```

```
There are 18 uniques categories.
```

[146]:

```
stockholm_venues.groupby('Venue Category').count().
↪sort_values('Venue',ascending=False)
```

[146]:

```
                      Neighborhood  Neighborhood Latitude  \
Venue Category
Gym / Fitness Center           278                    278
Gym                             68                     68
Gym Pool                        11                     11
Martial Arts Dojo               11                     11
Track                            7                      7
Yoga Studio                      6                      6
```

```
Weight Loss Center                4                      4
Climbing Gym                      4                      4
Sporting Goods Shop               3                      3
Spa                               3                      3
Track Stadium                     2                      2
Stadium                           2                      2
Pool                              2                      2
Rehab Center                      2                      2
Pilates Studio                    1                      1
Outdoor Gym                       1                      1
Medical Center                    1                      1
Athletics & Sports                1                      1

                      Neighborhood Longitude  Venue  Venue Latitude  \
Venue Category
Gym / Fitness Center                     278    278             278
Gym                                       68     68              68
Gym Pool                                  11     11              11
Martial Arts Dojo                         11     11              11
Track                                      7      7               7
Yoga Studio                                6      6               6
Weight Loss Center                         4      4               4
Climbing Gym                               4      4               4
Sporting Goods Shop                        3      3               3
Spa                                        3      3               3
Track Stadium                              2      2               2
Stadium                                    2      2               2
Pool                                       2      2               2
Rehab Center                               2      2               2
Pilates Studio                             1      1               1
Outdoor Gym                                1      1               1
Medical Center                             1      1               1
Athletics & Sports                         1      1               1

                      Venue Longitude
Venue Category
Gym / Fitness Center              278
Gym                                68
Gym Pool                           11
Martial Arts Dojo                  11
Track                               7
Yoga Studio                         6
Weight Loss Center                  4
Climbing Gym                        4
Sporting Goods Shop                 3
Spa                                 3
Track Stadium                       2
```

```
Stadium                          2
Pool                             2
Rehab Center                     2
Pilates Studio                   1
Outdoor Gym                      1
Medical Center                   1
Athletics & Sports               1
```

2.6 Here we see that there are 2 categories which seems to be popular.

We can also see that there are not that many other specific categories. I know that there are more than the API result provided for us

[ ]:

```

```

[147]:

```
# one hot encoding
stockholm_onehot = pd.get_dummies(stockholm_venues[['Venue Category']],␣
↪prefix="", prefix_sep="")

# add neighborhood column back to dataframe
stockholm_onehot['Neighborhood'] = stockholm_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [stockholm_onehot.columns[-1]] + list(stockholm_onehot.columns[:
↪-1])
stockholm_onehot = stockholm_onehot[fixed_columns]

stockholm_onehot.head()
```

[147]:

```
  Neighborhood  Athletics & Sports  Climbing Gym  Gym  Gym / Fitness Center  \
0      Nykvarn                   0             0    1                     0
1      Vaxholm                   0             0    0                     1
2      Vaxholm                   0             0    1                     0
3      Vaxholm                   0             0    0                     1
4      Vaxholm                   0             0    1                     0

   Gym Pool  Martial Arts Dojo  Medical Center  Outdoor Gym  Pilates Studio  \
0         0                  0               0            0               0
1         0                  0               0            0               0
2         0                  0               0            0               0
3         0                  0               0            0               0
4         0                  0               0            0               0

   Pool  Rehab Center  Spa  Sporting Goods Shop  Stadium  Track  \
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

   Track Stadium  Weight Loss Center  Yoga Studio
0              0                   0            0
1              0                   0            0
2              0                   0            0
3              0                   0            0
4              0                   0            0
```

[148]:
```
stockholm_onehot.shape
```

[148]:
```
(407, 19)
```

[149]:
```
stockholm_grouped = stockholm_onehot.groupby('Neighborhood').mean().
 ↪reset_index()
stockholm_grouped
```

[149]:
```
      Neighborhood  Athletics & Sports  Climbing Gym       Gym  \
0         Botkyrka            0.000000      0.000000  0.000000
1         Danderyd            0.000000      0.000000  0.088235
2          Haninge            0.000000      0.000000  0.333333
3         Huddinge            0.000000      0.000000  0.153846
4          Järfälla            0.000000      0.000000  0.083333
5           Lidingö            0.000000      0.000000  0.181818
6            Nacka            0.000000      0.000000  0.153846
7         Norrtälje            0.000000      0.000000  0.000000
8          Nykvarn            0.000000      0.000000  1.000000
9            Salem            0.000000      0.000000  0.000000
10         Sigtuna            0.000000      0.000000  0.142857
11      Sollentuna            0.000000      0.045455  0.000000
12           Solna            0.000000      0.010526  0.210526
13       Stockholm            0.000000      0.000000  0.214286
14      Sundbyberg            0.011111      0.011111  0.188889
15           Tyresö            0.000000      0.000000  0.000000
16             Täby            0.000000      0.000000  0.117647
17  Upplands Väsby            0.000000      0.000000  0.000000
18    Upplands-Bro            0.000000      0.000000  0.333333
19      Vallentuna            0.000000      0.000000  0.000000
20          Vaxholm            0.000000      0.200000  0.400000
21           Värmdö            0.000000      0.000000  0.400000

    Gym / Fitness Center  Gym Pool  Martial Arts Dojo  Medical Center  \
0               0.666667  0.000000           0.000000        0.000000
1               0.735294  0.029412           0.029412        0.000000
2               0.666667  0.000000           0.000000        0.000000
3               0.692308  0.000000           0.000000        0.000000
```

4	0.833333	0.000000	0.000000	0.000000
5	0.581818	0.036364	0.036364	0.018182
6	0.769231	0.000000	0.076923	0.000000
7	1.000000	0.000000	0.000000	0.000000
8	0.000000	0.000000	0.000000	0.000000
9	1.000000	0.000000	0.000000	0.000000
10	0.714286	0.142857	0.000000	0.000000
11	0.818182	0.045455	0.000000	0.000000
12	0.642105	0.031579	0.031579	0.000000
13	0.714286	0.000000	0.000000	0.000000
14	0.677778	0.022222	0.033333	0.000000
15	1.000000	0.000000	0.000000	0.000000
16	0.705882	0.000000	0.058824	0.000000
17	1.000000	0.000000	0.000000	0.000000
18	0.333333	0.333333	0.000000	0.000000
19	1.000000	0.000000	0.000000	0.000000
20	0.400000	0.000000	0.000000	0.000000
21	0.600000	0.000000	0.000000	0.000000

	Outdoor Gym	Pilates Studio	Pool	Rehab Center	Spa	\
0	0.000000	0.000000	0.000000	0.000000	0.000000	
1	0.000000	0.000000	0.000000	0.000000	0.029412	
2	0.000000	0.000000	0.000000	0.000000	0.000000	
3	0.000000	0.000000	0.000000	0.000000	0.000000	
4	0.000000	0.000000	0.000000	0.000000	0.000000	
5	0.018182	0.018182	0.000000	0.000000	0.000000	
6	0.000000	0.000000	0.000000	0.000000	0.000000	
7	0.000000	0.000000	0.000000	0.000000	0.000000	
8	0.000000	0.000000	0.000000	0.000000	0.000000	
9	0.000000	0.000000	0.000000	0.000000	0.000000	
10	0.000000	0.000000	0.000000	0.000000	0.000000	
11	0.000000	0.000000	0.045455	0.000000	0.000000	
12	0.000000	0.000000	0.010526	0.010526	0.010526	
13	0.000000	0.000000	0.000000	0.000000	0.000000	
14	0.000000	0.000000	0.000000	0.011111	0.011111	
15	0.000000	0.000000	0.000000	0.000000	0.000000	
16	0.000000	0.000000	0.000000	0.000000	0.000000	
17	0.000000	0.000000	0.000000	0.000000	0.000000	
18	0.000000	0.000000	0.000000	0.000000	0.000000	
19	0.000000	0.000000	0.000000	0.000000	0.000000	
20	0.000000	0.000000	0.000000	0.000000	0.000000	
21	0.000000	0.000000	0.000000	0.000000	0.000000	

	Sporting Goods Shop	Stadium	Track	Track Stadium	\
0	0.000000	0.000000	0.333333	0.000000	
1	0.000000	0.000000	0.029412	0.029412	
2	0.000000	0.000000	0.000000	0.000000	

3	0.000000	0.000000	0.076923	0.000000
4	0.000000	0.000000	0.083333	0.000000
5	0.018182	0.000000	0.000000	0.000000
6	0.000000	0.000000	0.000000	0.000000
7	0.000000	0.000000	0.000000	0.000000
8	0.000000	0.000000	0.000000	0.000000
9	0.000000	0.000000	0.000000	0.000000
10	0.000000	0.000000	0.000000	0.000000
11	0.000000	0.000000	0.000000	0.000000
12	0.010526	0.010526	0.010526	0.000000
13	0.000000	0.000000	0.071429	0.000000
14	0.011111	0.011111	0.011111	0.000000
15	0.000000	0.000000	0.000000	0.000000
16	0.000000	0.000000	0.000000	0.058824
17	0.000000	0.000000	0.000000	0.000000
18	0.000000	0.000000	0.000000	0.000000
19	0.000000	0.000000	0.000000	0.000000
20	0.000000	0.000000	0.000000	0.000000
21	0.000000	0.000000	0.000000	0.000000

	Weight Loss Center	Yoga Studio
0	0.000000	0.000000
1	0.029412	0.000000
2	0.000000	0.000000
3	0.076923	0.000000
4	0.000000	0.000000
5	0.000000	0.090909
6	0.000000	0.000000
7	0.000000	0.000000
8	0.000000	0.000000
9	0.000000	0.000000
10	0.000000	0.000000
11	0.045455	0.000000
12	0.000000	0.010526
13	0.000000	0.000000
14	0.000000	0.000000
15	0.000000	0.000000
16	0.058824	0.000000
17	0.000000	0.000000
18	0.000000	0.000000
19	0.000000	0.000000
20	0.000000	0.000000
21	0.000000	0.000000

[150]:
```
stockholm_grouped.shape
```

[150]: (22, 19)

[151]:
```python
num_top_venues = 5

for hood in stockholm_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = stockholm_grouped[stockholm_grouped['Neighborhood'] == hood].T.
↪reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).
↪head(num_top_venues))
    print('\n')
```

```
----Botkyrka----
                  venue  freq
0  Gym / Fitness Center  0.67
1                 Track  0.33
2    Athletics & Sports  0.00
3          Rehab Center  0.00
4     Weight Loss Center  0.00


----Danderyd----
                  venue  freq
0  Gym / Fitness Center  0.74
1                   Gym  0.09
2                   Spa  0.03
3              Gym Pool  0.03
4      Martial Arts Dojo  0.03


----Haninge----
                  venue  freq
0  Gym / Fitness Center  0.67
1                   Gym  0.33
2    Athletics & Sports  0.00
3          Rehab Center  0.00
4     Weight Loss Center  0.00


----Huddinge----
                  venue  freq
0  Gym / Fitness Center  0.69
1                   Gym  0.15
2     Weight Loss Center  0.08
3                 Track  0.08
```

```
4    Athletics & Sports  0.00


----Järfälla----
                  venue  freq
0  Gym / Fitness Center  0.83
1                   Gym  0.08
2                 Track  0.08
3    Athletics & Sports  0.00
4          Rehab Center  0.00


----Lidingö----
                  venue  freq
0  Gym / Fitness Center  0.58
1                   Gym  0.18
2           Yoga Studio  0.09
3              Gym Pool  0.04
4     Martial Arts Dojo  0.04


----Nacka----
                  venue  freq
0  Gym / Fitness Center  0.77
1                   Gym  0.15
2     Martial Arts Dojo  0.08
3    Athletics & Sports  0.00
4                   Spa  0.00


----Norrtälje----
                  venue  freq
0  Gym / Fitness Center   1.0
1    Athletics & Sports   0.0
2          Rehab Center   0.0
3    Weight Loss Center   0.0
4         Track Stadium   0.0


----Nykvarn----
                venue  freq
0                 Gym   1.0
1  Athletics & Sports   0.0
2        Rehab Center   0.0
3  Weight Loss Center   0.0
4       Track Stadium   0.0
```

```
----Salem----
                  venue  freq
0  Gym / Fitness Center   1.0
1    Athletics & Sports   0.0
2          Rehab Center   0.0
3    Weight Loss Center   0.0
4         Track Stadium   0.0


----Sigtuna----
                  venue  freq
0  Gym / Fitness Center  0.71
1                   Gym  0.14
2              Gym Pool  0.14
3    Athletics & Sports  0.00
4                   Spa  0.00


----Sollentuna----
                  venue  freq
0  Gym / Fitness Center  0.82
1                  Pool  0.05
2    Weight Loss Center  0.05
3              Gym Pool  0.05
4          Climbing Gym  0.05


----Solna----
                  venue  freq
0  Gym / Fitness Center  0.64
1                   Gym  0.21
2              Gym Pool  0.03
3     Martial Arts Dojo  0.03
4                  Pool  0.01


----Stockholm----
                  venue  freq
0  Gym / Fitness Center  0.71
1                   Gym  0.21
2                 Track  0.07
3    Athletics & Sports  0.00
4          Rehab Center  0.00


----Sundbyberg----
                  venue  freq
0  Gym / Fitness Center  0.68
```

```
1                  Gym  0.19
2     Martial Arts Dojo  0.03
3              Gym Pool  0.02
4    Athletics & Sports  0.01


----Tyresö----
                  venue  freq
0  Gym / Fitness Center   1.0
1    Athletics & Sports   0.0
2          Rehab Center   0.0
3    Weight Loss Center   0.0
4         Track Stadium   0.0


----Täby----
                  venue  freq
0  Gym / Fitness Center  0.71
1                   Gym  0.12
2    Weight Loss Center  0.06
3     Martial Arts Dojo  0.06
4         Track Stadium  0.06


----Upplands Väsby----
                  venue  freq
0  Gym / Fitness Center   1.0
1    Athletics & Sports   0.0
2          Rehab Center   0.0
3    Weight Loss Center   0.0
4         Track Stadium   0.0


----Upplands-Bro----
                  venue  freq
0                   Gym  0.33
1  Gym / Fitness Center  0.33
2              Gym Pool  0.33
3    Athletics & Sports  0.00
4                   Spa  0.00


----Vallentuna----
                  venue  freq
0  Gym / Fitness Center   1.0
1    Athletics & Sports   0.0
2          Rehab Center   0.0
3    Weight Loss Center   0.0
```

```
4         Track Stadium   0.0


----Vaxholm----
                  venue  freq
0                   Gym   0.4
1   Gym / Fitness Center   0.4
2          Climbing Gym   0.2
3     Athletics & Sports   0.0
4                   Spa   0.0


----Värmdö----
                  venue  freq
0   Gym / Fitness Center   0.6
1                   Gym   0.4
2     Athletics & Sports   0.0
3          Rehab Center   0.0
4     Weight Loss Center   0.0
```

[152]:
```
def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

[153]:
```
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Neighborhood']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = stockholm_grouped['Neighborhood']

for ind in np.arange(stockholm_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] =␣
 ↪return_most_common_venues(stockholm_grouped.iloc[ind, :], num_top_venues)
```

```
neighborhoods_venues_sorted.head()
```

[153]:
```
  Neighborhood 1st Most Common Venue 2nd Most Common Venue  \
0     Botkyrka  Gym / Fitness Center                 Track
1     Danderyd  Gym / Fitness Center                   Gym
2      Haninge  Gym / Fitness Center                   Gym
3     Huddinge  Gym / Fitness Center                   Gym
4     Järfälla  Gym / Fitness Center                 Track

  3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
0           Yoga Studio            Outdoor Gym          Climbing Gym
1         Track Stadium                  Track                   Spa
2           Yoga Studio     Weight Loss Center          Climbing Gym
3                 Track     Weight Loss Center           Yoga Studio
4                   Gym            Yoga Studio           Outdoor Gym

  6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
0                   Gym               Gym Pool     Martial Arts Dojo
1              Gym Pool     Weight Loss Center     Martial Arts Dojo
2              Gym Pool      Martial Arts Dojo        Medical Center
3           Outdoor Gym           Climbing Gym              Gym Pool
4          Climbing Gym               Gym Pool     Martial Arts Dojo

  9th Most Common Venue 10th Most Common Venue
0        Medical Center        Pilates Studio
1           Yoga Studio        Medical Center
2           Outdoor Gym        Pilates Studio
3     Martial Arts Dojo        Medical Center
4        Medical Center        Pilates Studio
```

[154]:
```
# import k-means from clustering stage
from sklearn.cluster import KMeans
```

[155]:
```
# set number of clusters
kclusters = 5

stockholm_grouped_clustering = stockholm_grouped.drop('Neighborhood', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).
 ↪fit(stockholm_grouped_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]
```

[155]:
```
array([3, 3, 1, 3, 0, 3, 3, 0, 4, 0], dtype=int32)
```

[161]:
```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

stockholm_merged = stockholm_venues

# merge stockholm_grouped with toronto_data to add latitude/longitude for each␣
↪neighborhood
stockholm_merged = stockholm_merged.join(neighborhoods_venues_sorted.
↪set_index('Neighborhood'), on='Neighborhood')

stockholm_merged.head() # check the last columns!
```

[161]:
```
  Neighborhood  Neighborhood Latitude  Neighborhood Longitude  \
0      Nykvarn              59.196527               17.408081
1      Vaxholm              59.404060               18.331331
2      Vaxholm              59.404060               18.331331
3      Vaxholm              59.404060               18.331331
4      Vaxholm              59.404060               18.331331

                      Venue  Venue Latitude  Venue Longitude  \
0            The Alfort gym       59.183253        17.440904
1  Niana Fitness Vaxö Skola       59.403463        18.358818
2                    Fysio+       59.402574        18.340298
3             Accédo Fons HB       59.393115        18.279122
4              Danderyds Gym       59.433331        18.319870

          Venue Category  Cluster Labels 1st Most Common Venue  \
0                    Gym               4                   Gym
1  Gym / Fitness Center               1                   Gym
2                    Gym               1                   Gym
3  Gym / Fitness Center               1                   Gym
4                    Gym               1                   Gym

  2nd Most Common Venue 3rd Most Common Venue 4th Most Common Venue  \
0          Yoga Studio    Weight Loss Center          Climbing Gym
1  Gym / Fitness Center          Climbing Gym           Yoga Studio
2  Gym / Fitness Center          Climbing Gym           Yoga Studio
3  Gym / Fitness Center          Climbing Gym           Yoga Studio
4  Gym / Fitness Center          Climbing Gym           Yoga Studio

  5th Most Common Venue 6th Most Common Venue 7th Most Common Venue  \
0  Gym / Fitness Center              Gym Pool     Martial Arts Dojo
1    Weight Loss Center              Gym Pool     Martial Arts Dojo
2    Weight Loss Center              Gym Pool     Martial Arts Dojo
3    Weight Loss Center              Gym Pool     Martial Arts Dojo
4    Weight Loss Center              Gym Pool     Martial Arts Dojo
```

```
  8th Most Common Venue 9th Most Common Venue 10th Most Common Venue
0        Medical Center           Outdoor Gym         Pilates Studio
1        Medical Center           Outdoor Gym         Pilates Studio
2        Medical Center           Outdoor Gym         Pilates Studio
3        Medical Center           Outdoor Gym         Pilates Studio
4        Medical Center           Outdoor Gym         Pilates Studio
```

[162]:
```
# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors
```

[163]:
```
# create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(stockholm_merged['Venue Latitude'],␣
 ↪stockholm_merged['Venue Longitude'], stockholm_merged['Neighborhood'],␣
 ↪stockholm_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

[163]:
```
<folium.folium.Map at 0x7f43ff943f60>
```

2.7 Here we see clusters of similar offerings

[164]:
```
stockholm_merged.loc[stockholm_merged['Cluster Labels'] == 0, stockholm_merged.
 ↪columns[[1] + list(range(5, stockholm_merged.shape[1]))]]
```

[164]:
```
     Neighborhood Latitude  Venue Longitude        Venue Category  \
6                59.238705        17.766051  Gym / Fitness Center
7                59.238705        17.767149  Gym / Fitness Center
```

45	59.584996	18.211618	Gym / Fitness Center
51	59.516693	17.925242	Gym / Fitness Center
52	59.516693	17.919924	Gym / Fitness Center
53	59.516693	17.910370	Gym / Fitness Center
54	59.516693	17.909956	Gym / Fitness Center
55	59.516693	17.917923	Gym / Fitness Center
56	59.516693	17.913920	Gym / Fitness Center
57	59.516693	17.913542	Gym / Fitness Center
120	59.210855	18.281395	Gym / Fitness Center
211	59.766667	18.700714	Gym / Fitness Center
212	59.766667	18.687450	Gym / Fitness Center
213	59.766667	18.669980	Gym / Fitness Center
214	59.766667	18.703678	Gym / Fitness Center
232	59.447015	17.951135	Gym / Fitness Center
233	59.447015	17.952040	Gym Pool
234	59.447015	17.951978	Gym / Fitness Center
235	59.447015	17.952290	Gym / Fitness Center
236	59.447015	17.938103	Gym / Fitness Center
237	59.447015	17.941371	Gym / Fitness Center
238	59.447015	17.965874	Gym / Fitness Center
239	59.447015	17.954289	Gym / Fitness Center
240	59.447015	17.946253	Gym / Fitness Center
241	59.447015	17.941157	Gym / Fitness Center
242	59.447015	17.913542	Gym / Fitness Center
243	59.447015	17.913920	Gym / Fitness Center
244	59.447015	17.971590	Gym / Fitness Center
245	59.447015	17.947024	Weight Loss Center
246	59.447015	17.963146	Gym / Fitness Center
247	59.447015	17.945293	Gym / Fitness Center
248	59.447015	17.948034	Gym / Fitness Center
249	59.447015	17.946620	Gym / Fitness Center
250	59.447015	17.944537	Gym / Fitness Center
251	59.447015	17.922227	Pool
252	59.447015	17.915347	Climbing Gym
253	59.447015	17.948611	Gym / Fitness Center
254	59.445019	17.833913	Gym / Fitness Center
255	59.445019	17.837646	Gym / Fitness Center
256	59.445019	17.836654	Gym / Fitness Center
257	59.445019	17.831026	Gym / Fitness Center
258	59.445019	17.825908	Gym / Fitness Center
259	59.445019	17.802792	Gym / Fitness Center
260	59.445019	17.864950	Gym / Fitness Center
261	59.445019	17.863855	Gym / Fitness Center
262	59.445019	17.848132	Gym / Fitness Center
263	59.445019	17.848391	Gym / Fitness Center
264	59.445019	17.812909	Gym
265	59.445019	17.797617	Track

	Cluster Labels	1st Most Common Venue	2nd Most Common Venue \
6	0	Gym / Fitness Center	Yoga Studio
7	0	Gym / Fitness Center	Yoga Studio
45	0	Gym / Fitness Center	Yoga Studio
51	0	Gym / Fitness Center	Yoga Studio
52	0	Gym / Fitness Center	Yoga Studio
53	0	Gym / Fitness Center	Yoga Studio
54	0	Gym / Fitness Center	Yoga Studio
55	0	Gym / Fitness Center	Yoga Studio
56	0	Gym / Fitness Center	Yoga Studio
57	0	Gym / Fitness Center	Yoga Studio
120	0	Gym / Fitness Center	Yoga Studio
211	0	Gym / Fitness Center	Yoga Studio
212	0	Gym / Fitness Center	Yoga Studio
213	0	Gym / Fitness Center	Yoga Studio
214	0	Gym / Fitness Center	Yoga Studio
232	0	Gym / Fitness Center	Climbing Gym
233	0	Gym / Fitness Center	Climbing Gym
234	0	Gym / Fitness Center	Climbing Gym
235	0	Gym / Fitness Center	Climbing Gym
236	0	Gym / Fitness Center	Climbing Gym
237	0	Gym / Fitness Center	Climbing Gym
238	0	Gym / Fitness Center	Climbing Gym
239	0	Gym / Fitness Center	Climbing Gym
240	0	Gym / Fitness Center	Climbing Gym
241	0	Gym / Fitness Center	Climbing Gym
242	0	Gym / Fitness Center	Climbing Gym
243	0	Gym / Fitness Center	Climbing Gym
244	0	Gym / Fitness Center	Climbing Gym
245	0	Gym / Fitness Center	Climbing Gym
246	0	Gym / Fitness Center	Climbing Gym
247	0	Gym / Fitness Center	Climbing Gym
248	0	Gym / Fitness Center	Climbing Gym
249	0	Gym / Fitness Center	Climbing Gym
250	0	Gym / Fitness Center	Climbing Gym
251	0	Gym / Fitness Center	Climbing Gym
252	0	Gym / Fitness Center	Climbing Gym
253	0	Gym / Fitness Center	Climbing Gym
254	0	Gym / Fitness Center	Track
255	0	Gym / Fitness Center	Track
256	0	Gym / Fitness Center	Track
257	0	Gym / Fitness Center	Track
258	0	Gym / Fitness Center	Track
259	0	Gym / Fitness Center	Track
260	0	Gym / Fitness Center	Track
261	0	Gym / Fitness Center	Track

262	0	Gym / Fitness Center	Track
263	0	Gym / Fitness Center	Track
264	0	Gym / Fitness Center	Track
265	0	Gym / Fitness Center	Track

	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	\
6	Weight Loss Center	Climbing Gym	Gym	
7	Weight Loss Center	Climbing Gym	Gym	
45	Weight Loss Center	Climbing Gym	Gym	
51	Weight Loss Center	Climbing Gym	Gym	
52	Weight Loss Center	Climbing Gym	Gym	
53	Weight Loss Center	Climbing Gym	Gym	
54	Weight Loss Center	Climbing Gym	Gym	
55	Weight Loss Center	Climbing Gym	Gym	
56	Weight Loss Center	Climbing Gym	Gym	
57	Weight Loss Center	Climbing Gym	Gym	
120	Weight Loss Center	Climbing Gym	Gym	
211	Weight Loss Center	Climbing Gym	Gym	
212	Weight Loss Center	Climbing Gym	Gym	
213	Weight Loss Center	Climbing Gym	Gym	
214	Weight Loss Center	Climbing Gym	Gym	
232	Gym Pool	Pool	Weight Loss Center	
233	Gym Pool	Pool	Weight Loss Center	
234	Gym Pool	Pool	Weight Loss Center	
235	Gym Pool	Pool	Weight Loss Center	
236	Gym Pool	Pool	Weight Loss Center	
237	Gym Pool	Pool	Weight Loss Center	
238	Gym Pool	Pool	Weight Loss Center	
239	Gym Pool	Pool	Weight Loss Center	
240	Gym Pool	Pool	Weight Loss Center	
241	Gym Pool	Pool	Weight Loss Center	
242	Gym Pool	Pool	Weight Loss Center	
243	Gym Pool	Pool	Weight Loss Center	
244	Gym Pool	Pool	Weight Loss Center	
245	Gym Pool	Pool	Weight Loss Center	
246	Gym Pool	Pool	Weight Loss Center	
247	Gym Pool	Pool	Weight Loss Center	
248	Gym Pool	Pool	Weight Loss Center	
249	Gym Pool	Pool	Weight Loss Center	
250	Gym Pool	Pool	Weight Loss Center	
251	Gym Pool	Pool	Weight Loss Center	
252	Gym Pool	Pool	Weight Loss Center	
253	Gym Pool	Pool	Weight Loss Center	
254	Gym	Yoga Studio	Outdoor Gym	
255	Gym	Yoga Studio	Outdoor Gym	
256	Gym	Yoga Studio	Outdoor Gym	
257	Gym	Yoga Studio	Outdoor Gym	

258	Gym	Yoga Studio	Outdoor Gym
259	Gym	Yoga Studio	Outdoor Gym
260	Gym	Yoga Studio	Outdoor Gym
261	Gym	Yoga Studio	Outdoor Gym
262	Gym	Yoga Studio	Outdoor Gym
263	Gym	Yoga Studio	Outdoor Gym
264	Gym	Yoga Studio	Outdoor Gym
265	Gym	Yoga Studio	Outdoor Gym

	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue \
6	Gym Pool	Martial Arts Dojo	Medical Center
7	Gym Pool	Martial Arts Dojo	Medical Center
45	Gym Pool	Martial Arts Dojo	Medical Center
51	Gym Pool	Martial Arts Dojo	Medical Center
52	Gym Pool	Martial Arts Dojo	Medical Center
53	Gym Pool	Martial Arts Dojo	Medical Center
54	Gym Pool	Martial Arts Dojo	Medical Center
55	Gym Pool	Martial Arts Dojo	Medical Center
56	Gym Pool	Martial Arts Dojo	Medical Center
57	Gym Pool	Martial Arts Dojo	Medical Center
120	Gym Pool	Martial Arts Dojo	Medical Center
211	Gym Pool	Martial Arts Dojo	Medical Center
212	Gym Pool	Martial Arts Dojo	Medical Center
213	Gym Pool	Martial Arts Dojo	Medical Center
214	Gym Pool	Martial Arts Dojo	Medical Center
232	Yoga Studio	Outdoor Gym	Gym
233	Yoga Studio	Outdoor Gym	Gym
234	Yoga Studio	Outdoor Gym	Gym
235	Yoga Studio	Outdoor Gym	Gym
236	Yoga Studio	Outdoor Gym	Gym
237	Yoga Studio	Outdoor Gym	Gym
238	Yoga Studio	Outdoor Gym	Gym
239	Yoga Studio	Outdoor Gym	Gym
240	Yoga Studio	Outdoor Gym	Gym
241	Yoga Studio	Outdoor Gym	Gym
242	Yoga Studio	Outdoor Gym	Gym
243	Yoga Studio	Outdoor Gym	Gym
244	Yoga Studio	Outdoor Gym	Gym
245	Yoga Studio	Outdoor Gym	Gym
246	Yoga Studio	Outdoor Gym	Gym
247	Yoga Studio	Outdoor Gym	Gym
248	Yoga Studio	Outdoor Gym	Gym
249	Yoga Studio	Outdoor Gym	Gym
250	Yoga Studio	Outdoor Gym	Gym
251	Yoga Studio	Outdoor Gym	Gym
252	Yoga Studio	Outdoor Gym	Gym
253	Yoga Studio	Outdoor Gym	Gym

254	Climbing Gym	Gym Pool	Martial Arts Dojo
255	Climbing Gym	Gym Pool	Martial Arts Dojo
256	Climbing Gym	Gym Pool	Martial Arts Dojo
257	Climbing Gym	Gym Pool	Martial Arts Dojo
258	Climbing Gym	Gym Pool	Martial Arts Dojo
259	Climbing Gym	Gym Pool	Martial Arts Dojo
260	Climbing Gym	Gym Pool	Martial Arts Dojo
261	Climbing Gym	Gym Pool	Martial Arts Dojo
262	Climbing Gym	Gym Pool	Martial Arts Dojo
263	Climbing Gym	Gym Pool	Martial Arts Dojo
264	Climbing Gym	Gym Pool	Martial Arts Dojo
265	Climbing Gym	Gym Pool	Martial Arts Dojo

	9th Most Common Venue	10th Most Common Venue
6	Outdoor Gym	Pilates Studio
7	Outdoor Gym	Pilates Studio
45	Outdoor Gym	Pilates Studio
51	Outdoor Gym	Pilates Studio
52	Outdoor Gym	Pilates Studio
53	Outdoor Gym	Pilates Studio
54	Outdoor Gym	Pilates Studio
55	Outdoor Gym	Pilates Studio
56	Outdoor Gym	Pilates Studio
57	Outdoor Gym	Pilates Studio
120	Outdoor Gym	Pilates Studio
211	Outdoor Gym	Pilates Studio
212	Outdoor Gym	Pilates Studio
213	Outdoor Gym	Pilates Studio
214	Outdoor Gym	Pilates Studio
232	Martial Arts Dojo	Medical Center
233	Martial Arts Dojo	Medical Center
234	Martial Arts Dojo	Medical Center
235	Martial Arts Dojo	Medical Center
236	Martial Arts Dojo	Medical Center
237	Martial Arts Dojo	Medical Center
238	Martial Arts Dojo	Medical Center
239	Martial Arts Dojo	Medical Center
240	Martial Arts Dojo	Medical Center
241	Martial Arts Dojo	Medical Center
242	Martial Arts Dojo	Medical Center
243	Martial Arts Dojo	Medical Center
244	Martial Arts Dojo	Medical Center
245	Martial Arts Dojo	Medical Center
246	Martial Arts Dojo	Medical Center
247	Martial Arts Dojo	Medical Center
248	Martial Arts Dojo	Medical Center
249	Martial Arts Dojo	Medical Center

```
250     Martial Arts Dojo         Medical Center
251     Martial Arts Dojo         Medical Center
252     Martial Arts Dojo         Medical Center
253     Martial Arts Dojo         Medical Center
254        Medical Center         Pilates Studio
255        Medical Center         Pilates Studio
256        Medical Center         Pilates Studio
257        Medical Center         Pilates Studio
258        Medical Center         Pilates Studio
259        Medical Center         Pilates Studio
260        Medical Center         Pilates Studio
261        Medical Center         Pilates Studio
262        Medical Center         Pilates Studio
263        Medical Center         Pilates Studio
264        Medical Center         Pilates Studio
265        Medical Center         Pilates Studio
```

[165]:

```python
stockholm_merged.loc[stockholm_merged['Cluster Labels'] == 1, stockholm_merged.
↪columns[[1] + list(range(5, stockholm_merged.shape[1]))]]
```

[165]:

```
     Neighborhood Latitude  Venue Longitude        Venue Category  \
1                59.404060        18.358818  Gym / Fitness Center
2                59.404060        18.340298                   Gym
3                59.404060        18.279122  Gym / Fitness Center
4                59.404060        18.319870                   Gym
5                59.404060        18.306164          Climbing Gym
46               59.313522        18.421012  Gym / Fitness Center
47               59.313522        18.421690                   Gym
48               59.313522        18.392014  Gym / Fitness Center
49               59.313522        18.394950                   Gym
50               59.313522        18.421370  Gym / Fitness Center
361              59.107406        18.102943  Gym / Fitness Center
362              59.107406        18.122127  Gym / Fitness Center
363              59.107406        18.123676                   Gym

     Cluster Labels 1st Most Common Venue 2nd Most Common Venue  \
1                 1                   Gym  Gym / Fitness Center
2                 1                   Gym  Gym / Fitness Center
3                 1                   Gym  Gym / Fitness Center
4                 1                   Gym  Gym / Fitness Center
5                 1                   Gym  Gym / Fitness Center
46                1  Gym / Fitness Center                   Gym
47                1  Gym / Fitness Center                   Gym
48                1  Gym / Fitness Center                   Gym
49                1  Gym / Fitness Center                   Gym
50                1  Gym / Fitness Center                   Gym
361               1  Gym / Fitness Center                   Gym
```

```
362               1  Gym / Fitness Center                   Gym
363               1  Gym / Fitness Center                   Gym

    3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
1             Climbing Gym           Yoga Studio    Weight Loss Center
2             Climbing Gym           Yoga Studio    Weight Loss Center
3             Climbing Gym           Yoga Studio    Weight Loss Center
4             Climbing Gym           Yoga Studio    Weight Loss Center
5             Climbing Gym           Yoga Studio    Weight Loss Center
46             Yoga Studio    Weight Loss Center          Climbing Gym
47             Yoga Studio    Weight Loss Center          Climbing Gym
48             Yoga Studio    Weight Loss Center          Climbing Gym
49             Yoga Studio    Weight Loss Center          Climbing Gym
50             Yoga Studio    Weight Loss Center          Climbing Gym
361            Yoga Studio    Weight Loss Center          Climbing Gym
362            Yoga Studio    Weight Loss Center          Climbing Gym
363            Yoga Studio    Weight Loss Center          Climbing Gym

    6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
1                 Gym Pool     Martial Arts Dojo        Medical Center
2                 Gym Pool     Martial Arts Dojo        Medical Center
3                 Gym Pool     Martial Arts Dojo        Medical Center
4                 Gym Pool     Martial Arts Dojo        Medical Center
5                 Gym Pool     Martial Arts Dojo        Medical Center
46                Gym Pool     Martial Arts Dojo        Medical Center
47                Gym Pool     Martial Arts Dojo        Medical Center
48                Gym Pool     Martial Arts Dojo        Medical Center
49                Gym Pool     Martial Arts Dojo        Medical Center
50                Gym Pool     Martial Arts Dojo        Medical Center
361               Gym Pool     Martial Arts Dojo        Medical Center
362               Gym Pool     Martial Arts Dojo        Medical Center
363               Gym Pool     Martial Arts Dojo        Medical Center

    9th Most Common Venue 10th Most Common Venue
1              Outdoor Gym         Pilates Studio
2              Outdoor Gym         Pilates Studio
3              Outdoor Gym         Pilates Studio
4              Outdoor Gym         Pilates Studio
5              Outdoor Gym         Pilates Studio
46             Outdoor Gym         Pilates Studio
47             Outdoor Gym         Pilates Studio
48             Outdoor Gym         Pilates Studio
49             Outdoor Gym         Pilates Studio
50             Outdoor Gym         Pilates Studio
361            Outdoor Gym         Pilates Studio
362            Outdoor Gym         Pilates Studio
363            Outdoor Gym         Pilates Studio
```

[166]:
```
stockholm_merged.loc[stockholm_merged['Cluster Labels'] == 2, stockholm_merged.
↪columns[[1] + list(range(5, stockholm_merged.shape[1]))]]
```

[166]:
```
    Neighborhood Latitude  Venue Longitude       Venue Category  \
8               59.535708        17.628522  Gym / Fitness Center
9               59.535708        17.642369                   Gym
10              59.535708        17.642510              Gym Pool

    Cluster Labels 1st Most Common Venue 2nd Most Common Venue  \
8                2                   Gym  Gym / Fitness Center
9                2                   Gym  Gym / Fitness Center
10               2                   Gym  Gym / Fitness Center

   3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
8               Gym Pool           Yoga Studio    Weight Loss Center
9               Gym Pool           Yoga Studio    Weight Loss Center
10              Gym Pool           Yoga Studio    Weight Loss Center

   6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
8           Climbing Gym     Martial Arts Dojo        Medical Center
9           Climbing Gym     Martial Arts Dojo        Medical Center
10          Climbing Gym     Martial Arts Dojo        Medical Center

   9th Most Common Venue 10th Most Common Venue
8             Outdoor Gym         Pilates Studio
9             Outdoor Gym         Pilates Studio
10            Outdoor Gym         Pilates Studio
```

[167]:
```
stockholm_merged.loc[stockholm_merged['Cluster Labels'] == 3, stockholm_merged.
↪columns[[1] + list(range(5, stockholm_merged.shape[1]))]]
```

[167]:
```
     Neighborhood Latitude  Venue Longitude       Venue Category  \
11               59.401807        18.038546  Gym / Fitness Center
12               59.401807        18.069981  Gym / Fitness Center
13               59.401807        18.011597                   Gym
14               59.401807        18.005941  Gym / Fitness Center
15               59.401807        18.095700  Gym / Fitness Center
..                     …              …                      …
402              59.222639        18.136363  Gym / Fitness Center
403              59.222639        18.082840  Gym / Fitness Center
404              59.222639        18.126948                   Gym
405              59.222639        18.090105  Gym / Fitness Center
406              59.222639        18.082942                 Track

     Cluster Labels 1st Most Common Venue 2nd Most Common Venue  \
11                3  Gym / Fitness Center                   Gym
12                3  Gym / Fitness Center                   Gym
```

```
13                3  Gym / Fitness Center                   Gym
14                3  Gym / Fitness Center                   Gym
15                3  Gym / Fitness Center                   Gym
..              …                   …                   …
402               3  Gym / Fitness Center                   Gym
403               3  Gym / Fitness Center                   Gym
404               3  Gym / Fitness Center                   Gym
405               3  Gym / Fitness Center                   Gym
406               3  Gym / Fitness Center                   Gym

    3rd Most Common Venue 4th Most Common Venue 5th Most Common Venue  \
11          Track Stadium                 Track                   Spa
12          Track Stadium                 Track                   Spa
13          Track Stadium                 Track                   Spa
14          Track Stadium                 Track                   Spa
15          Track Stadium                 Track                   Spa
..                      …                     …                     …
402                 Track           Yoga Studio           Outdoor Gym
403                 Track           Yoga Studio           Outdoor Gym
404                 Track           Yoga Studio           Outdoor Gym
405                 Track           Yoga Studio           Outdoor Gym
406                 Track           Yoga Studio           Outdoor Gym

    6th Most Common Venue 7th Most Common Venue 8th Most Common Venue  \
11               Gym Pool    Weight Loss Center     Martial Arts Dojo
12               Gym Pool    Weight Loss Center     Martial Arts Dojo
13               Gym Pool    Weight Loss Center     Martial Arts Dojo
14               Gym Pool    Weight Loss Center     Martial Arts Dojo
15               Gym Pool    Weight Loss Center     Martial Arts Dojo
..                      …                     …                     …
402          Climbing Gym              Gym Pool     Martial Arts Dojo
403          Climbing Gym              Gym Pool     Martial Arts Dojo
404          Climbing Gym              Gym Pool     Martial Arts Dojo
405          Climbing Gym              Gym Pool     Martial Arts Dojo
406          Climbing Gym              Gym Pool     Martial Arts Dojo

    9th Most Common Venue 10th Most Common Venue
11            Yoga Studio         Medical Center
12            Yoga Studio         Medical Center
13            Yoga Studio         Medical Center
14            Yoga Studio         Medical Center
15            Yoga Studio         Medical Center
..                      …                      …
402        Medical Center        Pilates Studio
403        Medical Center        Pilates Studio
404        Medical Center        Pilates Studio
405        Medical Center        Pilates Studio
```

```
406        Medical Center        Pilates Studio

[341 rows x 14 columns]
```

[ ]:

[187]:
```
stockholm_venues.groupby('Venue').count().
  ↪sort_values('Neighborhood',ascending=False)
```

[187]:
```
                                Neighborhood  Neighborhood Latitude  \
Venue
SATS                                      59                     59
Puls & Träning                            45                     45
Nordic Wellness                           33                     33
Fitness24Seven                            30                     30
Actic                                     15                     15
…                                        …                    …
Friskis & Svettis - Stinsen                1                      1
Friskis & Svettis Abrahamsberg             1                      1
Friskis & Svettis Gärdet                   1                      1
Lifeguide Bee Thufvesson                   1                      1
Össeby-Garns Skytteklubb                   1                      1

                                Neighborhood Longitude  Venue Latitude  \
Venue
SATS                                                59              59
Puls & Träning                                      45              45
Nordic Wellness                                     33              33
Fitness24Seven                                      30              30
Actic                                               15              15
…                                                  …             …
Friskis & Svettis - Stinsen                          1               1
Friskis & Svettis Abrahamsberg                       1               1
Friskis & Svettis Gärdet                             1               1
Lifeguide Bee Thufvesson                             1               1
Össeby-Garns Skytteklubb                             1               1

                                Venue Longitude  Venue Category
Venue
SATS                                         59              59
Puls & Träning                               45              45
Nordic Wellness                              33              33
Fitness24Seven                               30              30
Actic                                        15              15
…                                           …             …
Friskis & Svettis - Stinsen                   1               1
Friskis & Svettis Abrahamsberg                1               1
```

```
Friskis & Svettis Gärdet                         1               1
Lifeguide Bee Thufvesson                         1               1
Össeby-Garns Skytteklubb                         1               1

[149 rows x 6 columns]
```

[ ]:
```
# Most of the questions are now answered and with more insights
```

[ ]:
```
Results section where you discuss the results.
Discussion section where you discuss any observations you noted and any␣
↪recommendations you can make based on the results.
Conclusion section where you conclude the report.
```

3 C. Results

Results section where you discuss the results.

Answers to the stated questions:

Can we access enough data and get insights for understanding the situation in Stockholm, even its risks and opportunities ? - Yes we where able to see relationships between potential consumers and existing provided services in different regions

How is the Stockholm Region is shaped geographically and how is it divided in sub-areas with municipals and boroughs ? - Yes we provided several lists of list of municipal and listed them according to its area size , its population and its density of number of citizens per square kilometers

What is the current Demographic - number of citizens over the Region ? Due to the islands and the bridges and water Sweden is also known for arranging its geographical areas in Postal Codes. - Yes, we plotted these information in different types of maps - where you can see it is a lot of water and differences how many people live and where the live

How does the current Gym alternatives and offerings look like - mapped over the regions and Postal codes ? - Yes, we where able to find the different services providers and narrowed in to different Gym categories being offered a bit different over the areas.

How is the offering vs the number of citizens differ between boroughs ? - Yes, we provides different viewpoint on this. Some bigger Gym providers has several venues in several municipals. A fewer boroughs have more services

What kind of Gym types are being offered in the Regions ? - Yes, we segment and clustered the different services related to the different areas

Are there Regions that do not have some of the offerings ? - Yes there are regions which does not have many categories and there are Gym providers that does not exist in all areas

What is the nr offerings vs nr of consumers in the different boroughs ? - Yes, we provided different viewpoints that highlights the differences

If a company will introduce a new Gym - wich regions is better suited to invest in - that still has a good customer base but is not too saturated with too many competitive offerings. - Due to Corona

- the situation has changed - before the citizens visited a gym close to their work. Now with Corona we can see a difference that some areas where people lice does not have that many services. The opposite is true as well some areas where people worked is now having difficulties in surviving as a company due to much less customers

How are the different Gym companies spread over different regions ? - Yes, we list how the different providers offers their services and we can easily see a top 3 list which has multiple venues in different areas.

How many regions have multiple centers from the same Gym company ? Provide a top 3 list of ex. the lack of services or where the availability is lower. - Yes , we provided a list

The government and the municipals is also funding and supporting initiatives and investments in more outside activities and gyms that are free for people to use. - Yes, this data is of importance to all actors interesting in insights for further decisions. There where only 1 Outside Gym in the result data. We know that many more exists in several areas

Outside Gyms is very popular though Corona but they have same constrained for Corona as the normal Gyms.

Is it possible to see how indivuals rate and like the different gyms ? - Yes, but we where only able to see a handful of users that commented or tipped a specific venue

4 D. Discussion

Discussion section where you discuss any observations you noted and any recommendations you can make based on the results.

5 E. Conclusion

This was a really interesting and value added journey of exploring the data. First I was able to spot a lack data from different sources that I first expected. I was then succesful and able to get hold of more data sources and aggregate the data in to a solid base for future usage that can be shared. I will also get be to several actors and suggest improvements of their data exposed or lack of data exposed. The changes in Swedens Regional, Counties amd address took much more time than expected and affects all ecosystems.

The Geographic structure is really difficult in Stockholm due to bridges, islands and water.

As a future iteration to get more in to details where each individual lives and get down to presence coordinates. I have a huge addressdata file from Stockholm addresses which could be used to merged data from the region and municipal data with its postal code centric data.

Another future iteration is to map the API and presence API from the Region Stockholm Transportation APIs. Access to buses, busstops, stations.

More and more alternative outside services are coming and they are more situational and in different places.

The data from Nominatim for getting Longitud and Latitud provided some coordinates that was located in the wrong region as well as in the water. I needed to get very detailed and more precise and instead asked for the municipal coordinate - which gave me often the center of the municipal.

But there where differences. I needed to adjust the coordinate for three municipals manually with code.

When I used a variety if location coordinates using the search and explore I got not many results back from Foursquare. When I changed to the center of the municipals and increased the range and also provided a search phrase for "Gym" the results improved and was useful.

[ ]:
```
All of the data sources needs to improve but it was good enoug to get the␣
↪insights needed to answer the questions.
```

Now we probably understood there are need for improvements both in what questions we are interested in and also got new insights in new opportunities or even spoted problems we did not exist

The Corona siutation will change this Gym provider its actors, what services they provide , what consumers wants and where they live.

As a future iteration we could include data from other sources ex. providers member lists, their attendance, their popularity and their fincancial situation services vs price and cost for operations