CapstoneProjectWeek4

October 17, 2019

1 Week 4 Presentation

[2]:

```
import numpy as np  # useful for many scientific computing in Python

import pandas as pd # primary data structure library
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

!pip install geopandas
import geopandas as gpd
import requests # library to handle requests

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors
import matplotlib.pyplot as plt


import json # library to handle JSON files

!conda install -c conda-forge geopy --yes # uncomment this line if you haven't␣
 ↪completed the Foursquare API lab

from geopy.geocoders import Nominatim # convert an address into latitude and␣
 ↪longitude values



#!conda install -c conda-forge folium=0.5.0 --yes
import folium
import shapely

from shapely.wkb import loads
#from shapely import wkt
from shapely.geometry import Point

# import k-means from clustering stage
```

```python
from sklearn.cluster import KMeans


print('Imported Libraries')
```

```
Requirement already satisfied: geopandas in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.6.1)
Requirement already satisfied: shapely in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopandas)
(1.6.4.post2)
Requirement already satisfied: fiona in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopandas)
(1.8.8)
Requirement already satisfied: pyproj in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopandas)
(1.9.6)
Requirement already satisfied: pandas>=0.23.0 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from geopandas)
(0.25.1)
Requirement already satisfied: cligj>=0.5 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (0.5.0)
Requirement already satisfied: munch in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (2.3.2)
Requirement already satisfied: click<8,>=4.0 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (7.0)
Requirement already satisfied: six>=1.7 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (1.12.0)
Requirement already satisfied: attrs>=17 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (19.1.0)
Requirement already satisfied: click-plugins>=1.0 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
fiona->geopandas) (1.1.1)
Requirement already satisfied: pytz>=2017.2 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
pandas>=0.23.0->geopandas) (2019.2)
Requirement already satisfied: numpy>=1.13.3 in
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from
pandas>=0.23.0->geopandas) (1.15.4)
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) (2.8.0)
Solving environment: done
```

```
==> WARNING: A newer version of conda exists. <==
  current version: 4.5.11
  latest version: 4.7.12

Please update conda by running

    $ conda update -n base -c defaults conda



# All requested packages already installed.

Imported Libraries
```

2 Introduction

In this presentation we will define a problem to solve using Data Science techniques managing and processing data from different sources, including data pulled using an API, in this case, from Foursquare.com

The problem selected for this example is to determine best places to open a Coffee Shop in the city of Buenos Aires in the proximity of subway stations.

There are a high number of variables to consider in such decision, but for the example of this practice, we will base our analysis in the data sets described below, leaving other important data, measurable or not, outside of the scope of the example.

3 Problem Description

The problem selected is based in the city of Buenos Aires, in particular near the subway stations that concentrates a high number of people traveling daily across the city. Considering Coffee Shops are particularly popular in Argentina as meeting points for joining with friends, coworkers, and also for business, we will analyze different sets of data to determine best possible new place to open a new Coffee Shop.

4 Data Section

The data sets used along this study are:

From the Government of the city of Buenos Aires, official information:

1. Neighborhood information, with geographical location
2. Population for each Neighborhood from same source, but based on 2010 official data
3. Subway stations information, including geographical location and line
4. Premetro stations information (additional stations complimenting the subway network)
5. Traffic information of the subway stations

From Foursquare.com

1. Exploration of all Venues near subways stations
2. Exploration of specific categories of Coffee Shops near subway stations

4.0.1 Buenos Aires city Neigborhood information with location data

```
[3]: borough=pd.read_csv('http://cdn.buenosaires.gob.ar/datosabiertos/datasets/
      ↪barrios/barrios.csv')
     borough.head()
```

```
[3]:                                                  WKT           barrio  \
     0  POLYGON ((-58.4528200492791 -34.5959886570639,…        CHACARITA
     1  POLYGON ((-58.4655768128541 -34.5965577078058,…         PATERNAL
     2  POLYGON ((-58.4237529813037 -34.5978273383243,…     VILLA CRESPO
     3  POLYGON ((-58.4946097568899 -34.6148652395239,…  VILLA DEL PARQUE
     4  POLYGON ((-58.4128700313089 -34.6141162515854,…          ALMAGRO

        comuna    perimetro          area
     0      15  7724.852955  3.115707e+06
     1      15  7087.513295  2.229829e+06
     2      15  8131.857075  3.615978e+06
     3      11  7705.389797  3.399596e+06
     4       5  8537.901368  4.050752e+06
```

```
[4]: borough=borough.rename(columns={'barrio':'Borough'}).
      ↪sort_values(by=['Borough']).reset_index(drop=True) #rename column and sort
     borough.head()
```

```
[4]:                                                  WKT    Borough  comuna  \
     0  POLYGON ((-58.4771156675186 -34.5951149914833,…  AGRONOMIA      15
     1  POLYGON ((-58.4128700313089 -34.6141162515854,…    ALMAGRO       5
     2  POLYGON ((-58.4119188098038 -34.5980030767748,…  BALVANERA       3
     3  POLYGON ((-58.3703353711449 -34.6329258371189,…   BARRACAS       4
     4  POLYGON ((-58.4505669109009 -34.5356104340406,…   BELGRANO      13

           perimetro          area
     0   6556.167772  2.122169e+06
     1   8537.901368  4.050752e+06
     2   8375.821811  4.342280e+06
     3  12789.791771  7.953453e+06
     4  20609.775397  7.999240e+06
```

4.0.2 Buenos Aires Population by Borough Dataset

[5]:

```
population=pd.read_csv('http://cdn.buenosaires.gob.ar/datosabiertos/datasets/
 ↪barrios/caba_pob_barrios_2010.csv')
population=population.rename(columns={'BARRIO':'Borough', 'POBLACION':
 ↪'Population'}) #rename column
population.head()
```

[5]:

```
     Borough  Population
0  AGRONOMIA       13912
1    ALMAGRO      131699
2  BALVANERA      138926
3   BARRACAS       89452
4   BELGRANO      126267
```

[6]:

```
#Join both Borough datasets together
borough=borough.join(population.set_index('Borough'), on='Borough')
```

[7]:

```
print(borough.shape)
borough.head()
```

```
(48, 6)
```

[7]:

```
                                                 WKT    Borough  comuna  \
0  POLYGON ((-58.4771156675186 -34.5951149914833,…  AGRONOMIA      15
1  POLYGON ((-58.4128700313089 -34.6141162515854,…    ALMAGRO       5
2  POLYGON ((-58.4119188098038 -34.5980030767748,…  BALVANERA       3
3  POLYGON ((-58.3703353711449 -34.6329258371189,…   BARRACAS       4
4  POLYGON ((-58.4505669109009 -34.5356104340406,…   BELGRANO      13

      perimetro          area  Population
0   6556.167772  2.122169e+06       13912
1   8537.901368  4.050752e+06      131699
2   8375.821811  4.342280e+06      138926
3  12789.791771  7.953453e+06       89452
4  20609.775397  7.999240e+06      126267
```

4.0.3 Subway Information dataset

[8]:

```
#Read subway stations names with geo locations
sub=pd.read_csv('http://cdn.buenosaires.gob.ar/datosabiertos/datasets/
 ↪subte-estaciones/estaciones-de-subte.csv')
print(sub.shape)
sub.head()
```

```
(90, 5)
```

```
[8]:         long        lat   id                    estacion linea
     0 -58.398928 -34.635750  1.0                     CASEROS     H
     1 -58.400970 -34.629376  2.0  INCLAN - MEZQUITA AL AHMAD     H
     2 -58.402323 -34.623092  3.0                 HUMBERTO 1°     H
     3 -58.404732 -34.615242  4.0                   VENEZUELA     H
     4 -58.406036 -34.608935  5.0      ONCE - 30 DE DICIEMBRE     H
```

```
[9]: sub=sub.rename(columns={'estacion':'Station Name', 'linea':'Station Line'})␣
      ↪#rename column
     sub=sub.drop(['id'],axis=1)
     print(sub.shape)
     sub.head()
```

```
(90, 4)
```

```
[9]:         long        lat                Station Name Station Line
     0 -58.398928 -34.635750                     CASEROS            H
     1 -58.400970 -34.629376  INCLAN - MEZQUITA AL AHMAD            H
     2 -58.402323 -34.623092                 HUMBERTO 1°            H
     3 -58.404732 -34.615242                   VENEZUELA            H
     4 -58.406036 -34.608935      ONCE - 30 DE DICIEMBRE            H
```

4.0.4 Premetro Data set

Additional metro stations

```
[10]: #Read the external extension to the subway lines called "Premetro"
      sub_pre=pd.read_csv('http://cdn.buenosaires.gob.ar/datosabiertos/datasets/
       ↪premetro/estaciones-premetro.csv')
      sub_pre=sub_pre.drop(['id'],axis=1)
      sub_pre=sub_pre.rename(columns={'nombre':'Station Name', 'linea':'Station␣
       ↪Line'}) #rename column
      print(sub_pre.shape)
      sub_pre.head()
```

```
(18, 4)
```

```
[10]:         long        lat Station Line                          Station Name
      0 -58.461491 -34.643757     PREMETRO                    INTENDENTE SAGUIER
      1 -58.456415 -34.648601     PREMETRO  BALBASTRO (Cementerio de Flores)
      2 -58.450047 -34.659499     PREMETRO                       ANA MARÍA JANER
      3 -58.446607 -34.662225     PREMETRO                  FERNANDEZ DE LA CRUZ
      4 -58.448513 -34.665368     PREMETRO                  PTE. ILLIA (Lacarra)
```

```
[11]: #Join subway and premetro information
      sub= pd.concat([sub,sub_pre], sort=False).sort_values('Station Name').
       ↪reset_index(drop=True)
      print(sub.shape)
```

```
sub.head()
```

```
(108, 4)
```

```
[11]:         long        lat       Station Name Station Line
      0 -58.380574 -34.604245         9 DE JULIO            D
      1 -58.436429 -34.618280             ACOYTE            A
      2 -58.407161 -34.591628             AGÜERO            D
      3 -58.401208 -34.609834            ALBERTI            A
      4 -58.420962 -34.603165  ALMAGRO - MEDRANO            B
```

```
[ ]:
```

4.0.5 Stations Traffic information

YTD 2019 from http://cdn.buenosaires.gob.ar/datosabiertos/datasets/subte-viajes-molinetes/molinetes-2019.zip

```
[12]: #Read uncompressed dataset already loaded
      sub_use=pd.read_csv(r'datahistorica082019.csv')
      sub_use.head(15)
```

```
[12]:     periodo       fecha     desde     hasta   linea  \
      0    201901  2019-01-01  08:00:00  08:15:00  LineaA
      1    201901  2019-01-01  08:00:00  08:15:00  LineaA
      2    201901  2019-01-01  08:00:00  08:15:00  LineaA
      3    201901  2019-01-01  08:00:00  08:15:00  LineaA
      4    201901  2019-01-01  08:00:00  08:15:00  LineaA
      5    201901  2019-01-01  08:00:00  08:15:00  LineaA
      6    201901  2019-01-01  08:00:00  08:15:00  LineaA
      7    201901  2019-01-01  08:00:00  08:15:00  LineaA
      8    201901  2019-01-01  08:00:00  08:15:00  LineaA
      9    201901  2019-01-01  08:00:00  08:15:00  LineaB
      10   201901  2019-01-01  08:00:00  08:15:00  LineaB
      11   201901  2019-01-01  08:00:00  08:15:00  LineaB
      12   201901  2019-01-01  08:00:00  08:15:00  LineaB
      13   201901  2019-01-01  08:00:00  08:15:00  LineaB
      14   201901  2019-01-01  08:00:00  08:15:00  LineaB

                               molinete          estacion  pax_pagos  \
      0             LineaA_Lima_N_Turn02              Lima        1.0
      1            LineaA_Loria_N_Turn03             Loria        3.0
      2    LineaA_Miserere_Q_HALL_Turn01   Plaza Miserere        3.0
      3         LineaA_Miserere_S_Turn01   Plaza Miserere        6.0
      4         LineaA_Miserere_S_Turn03   Plaza Miserere       10.0
      5             LineaA_Peru_S_Turn03              Peru        2.0
      6           LineaA_PJunta_S_Turn04     Primera Junta        9.0
```

```
7   LineaA_SanPedrito_Oeste_Turn05       San Pedrito        2.0
8          LineaA_SnzPena_N_Turn01     Saenz PeÃ Â±a        1.0
9             LineaB_Alem_N_Turn03   Leandro N. Alem        1.0
10          LineaB_Gardel_S_Turn02      Carlos Gardel        1.0
11     LineaB_JMRosas_Oeste_Turn08              Rosas        1.0
12         LineaB_Lacroze_E_Turn02  Federico Lacroze        1.0
13         LineaB_Lacroze_O_Turn04  Federico Lacroze        0.0
14         LineaB_Medrano_N_Turn03            Medrano       10.0

    pax_pases_pagos  pax_franq  total
0               0.0        0.0    1.0
1               0.0        0.0    3.0
2               0.0        0.0    3.0
3               0.0        0.0    6.0
4               0.0        0.0   10.0
5               0.0        0.0    2.0
6               0.0        0.0    9.0
7               0.0        1.0    3.0
8               0.0        0.0    1.0
9               0.0        0.0    1.0
10              0.0        0.0    1.0
11              0.0        0.0    1.0
12              0.0        0.0    1.0
13              0.0        0.0    0.0
14              0.0        0.0   10.0
```

[13]:
```
#Discard not needed columns
sub_use=sub_use.
 ↪drop(['desde','hasta','pax_pagos','pax_pases_pagos','pax_franq'],1)
print('Subway 2019 passengers records dataset: ',sub_use.shape)
```

```
Subway 2019 passengers records dataset:  (8300108, 6)
```

[14]:
```
sub_use.tail()
```

[14]:
```
          periodo       fecha   linea                      molinete  \
8300103    201908  2019-08-31  LineaD       LineaD_9Julio_S_Turn02
8300104    201908  2019-08-31  LineaD     LineaD_Pueyrredon_Turn01
8300105    201908  2019-08-31  LineaH     LineaH_Once_Norte_Turn04
8300106    201908  2019-08-31  LineaH       LineaH_Once_Sur_Turn02
8300107    201908  2019-08-31  LineaH  LineaH_Venezuela_Sur_Turn01

              estacion  total
8300103    9 de julio    4.0
8300104  Pueyrredon.D    1.0
8300105          Once    1.0
8300106          Once    1.0
```

```
8300107     Venezuela    2.0
```

[15]:

```
start=sub_use.iloc[0,1]
end=sub_use.iloc[-1,1]
print('Subway passengers initial date: ',start,' final date measured: ',end)

sub_use.groupby(['estacion','linea']).sum()[['total']]
```

```
Subway passengers initial date:  2019-01-01  final date measured:  2019-08-31
```

[15]:

```
                                   total
estacion             linea
9 de julio           LineaD    1832363.0
Acoyte               LineaA    2891038.0
AgÃ Â¼ero           LineaD    450586.0
AgÃ¼ero              LineaD    1160895.0
Agüero               LineaD     630504.0
Alberti              LineaA     958847.0
Angel Gallardo       LineaB    3048723.0
Avenida La Plata     LineaE     984719.0
Avenida de Mayo      LineaC    1104846.0
Boedo                LineaE     875852.0
Bolivar              LineaE    2497981.0
Bulnes               LineaD    3502431.0
Callao               LineaD    3219082.0
Callao.B             LineaB    3522456.0
Carabobo             LineaA    2340735.0
Carlos Gardel        LineaB    3210229.0
Carlos Pellegrini    LineaB    4406126.0
Caseros              LineaH    1387393.0
Castro Barros        LineaA    1854662.0
Catalinas            LineaE     323371.0
Catedral             LineaD    6039256.0
Congreso             LineaA    2624919.0
Congreso de Tucuman  LineaD    7285741.0
Constitucion         LineaC   16634004.0
Cordoba              LineaH    1522160.0
Correo Central       LineaE     462713.0
Corrientes           LineaH    1233007.0
Diagonal Norte       LineaC    1029999.0
Dorrego              LineaB    2051506.0
Echeverria           LineaB    1340491.0
Emilio Mitre         LineaE    1096100.0
Entre Rios           LineaE     533657.0
Facultad de Derecho  LineaH    1224406.0
Facultad de Medicina LineaD    3325412.0
Federico Lacroze     LineaB    6449850.0
```

```
Flores               LineaA   2356193.0
Florida              LineaB   4004692.0
General Belgrano     LineaE    574222.0
General San Martin   LineaC   1734714.0
Hospitales           LineaH   2712652.0
Humberto I           LineaH   1249960.0
Inclan               LineaH   1118686.0
Independencia        LineaC   1227333.0
Independencia.H      LineaE   1573716.0
Jose Hernandez       LineaD   2504747.0
Jose Maria Moreno    LineaE    749174.0
Jujuy                LineaE    632354.0
Juramento            LineaD   3607521.0
Las Heras            LineaH   2568933.0
Lavalle              LineaC   1904969.0
Leandro N. Alem      LineaB   5101373.0
Lima                 LineaA   1540111.0
Loria                LineaA   1642711.0
Los Incas            LineaB   2107479.0
Malabia              LineaB   3864007.0
Mariano Moreno       LineaC   1324711.0
Medalla Milagrosa    LineaE    370347.0
Medrano              LineaB   3409340.0
Ministro Carranza    LineaD   2576018.0
Olleros              LineaD   2731207.0
Once                 LineaH   2990236.0
Palermo              LineaD   3505322.0
Pasco                LineaA    707200.0
Pasteur              LineaB   2457707.0
Patricios            LineaH   2244552.0
Peru                 LineaA   2636778.0
Pichincha            LineaE    445478.0
Piedras              LineaA   1518580.0
Plaza Italia         LineaD   2617519.0
Plaza Miserere       LineaA   2822642.0
Plaza de Mayo        LineaA   3359463.0
Primera Junta        LineaA   2697752.0
Puan                 LineaA   1507711.0
Pueyrredon           LineaB   2434220.0
Pueyrredon.D         LineaD   2048789.0
Pza. de los Virreyes LineaE   1638090.0
Retiro               LineaC   7141916.0
Retiro E             LineaE    333457.0
Rio de Janeiro       LineaA   2108867.0
Rosas                LineaB   5649560.0
Saenz PeÃ Â±a        LineaA    398262.0
Saenz PeÃ±a          LineaA    876515.0
Saenz Peña           LineaA    485050.0
San Jose             LineaE    537189.0
San Juan             LineaC   1136550.0
San Pedrito          LineaA   5674922.0
Santa Fe             LineaH   2332929.0
Scalabrini Ortiz     LineaD   2668685.0
Tribunales           LineaD   2976193.0
Tronador             LineaB   1145765.0
Urquiza              LineaE    599984.0
Uruguay              LineaB   3103812.0
Varela               LineaE    433517.0
Venezuela            LineaH   1635804.0
```

4.0.6 Foursquare Venues Explore Function

[16]:
```
CLIENT_ID = 'JK4SBLGN3YU1DKHQS40VSJ2RUTBDQ0BF0VE2X2J0E2HSDR3M' # your␣
 ↪Foursquare ID
CLIENT_SECRET = 'QXOOOJBSDA10TTVT50BRHURLN2NEBGMZEKAWC3XPHPS4XD2M' # your␣
 ↪Foursquare Secret
VERSION = '20190605' # Foursquare API version

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

```
Your credentails:
CLIENT_ID: JK4SBLGN3YU1DKHQS40VSJ2RUTBDQ0BF0VE2X2J0E2HSDR3M
CLIENT_SECRET:QXOOOJBSDA10TTVT50BRHURLN2NEBGMZEKAWC3XPHPS4XD2M
```

[17]:
```
def getNearbyVenues(names, line, latitudes, longitudes, radius=800):
    LIMIT = 200 # limit of number of venues returned by Foursquare API
    radius = 800 # define radius
    venues_list=[]
    for name, line, lat, lng in zip(names, line, latitudes, longitudes):
         # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?
 ↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            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([(
            line,
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['location']['distance'],
            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 = ['Station Line',
                  'Station Name',
                  'Station Latitude',
                  'Station Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Distance',
                  'Venue Category']
    return(nearby_venues)
```

[18]:

```
LIMIT = 30000 # limit of number of venues returned by Foursquare API
radius = 15000 # define radius
         # create the API request URL
latitudes='-34.619139'
longitudes='-58.443683'
url = 'https://api.foursquare.com/v2/venues/explore?
 ↪&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            latitudes,
            longitudes,
            radius,
            LIMIT)
```

[20]:

```
#Create dataframe with Venues of Buenos Aires
bsas_venues = getNearbyVenues(names=sub['Station Name'],
                                   line=sub['Station Line'],
                                   latitudes=sub['lat'],
                                   longitudes=sub['long'])
```

```
#To execute only once and avoid Foursquare limite save to disk
bsas_venues.to_csv (r'venues_sub.csv', index = None, header=True)
```

[21]:

```
#Readl from disk
bsas_venues=pd.read_csv(r'venues_sub.csv')
print(bsas_venues.shape)
bsas_venues.head()
```

```
(7246, 9)
```

[21]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574

                          Venue  Venue Latitude  Venue Longitude  \
0  Obelisco - Plaza de la República      -34.603736       -58.381632
1                 Teatro Gran Rex      -34.603237       -58.378928
2                          On Fit      -34.602386       -58.379985
3                 All Saints Cafe      -34.603559       -58.378070
4               Tostado Café Club      -34.603494       -58.380793

   Venue Distance        Venue Category
0             112    Monument / Landmark
1             188                Theater
2             213  Gym / Fitness Center
3             241            Coffee Shop
4              85            Coffee Shop
```

Find and remove duplicated Venues

[23]:

```
bsas_venues=bsas_venues.sort_values(by=['Station Name','Venue','Venue␣
 ↪Latitude','Venue Longitude','Venue Distance']).reset_index(drop=True)
print(bsas_venues.shape)
bsas_venues.head(10)
```

```
(7246, 9)
```

[23]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574
5            D   9 DE JULIO        -34.604245         -58.380574
6            D   9 DE JULIO        -34.604245         -58.380574
```

```
7            D   9 DE JULIO        -34.604245         -58.380574
8            D   9 DE JULIO        -34.604245         -58.380574
9            D   9 DE JULIO        -34.604245         -58.380574

                    Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0          180 Burger Bar      -34.604848       -58.380574              67
1  474 Buenos Aires Hotel      -34.602514       -58.374013             631
2   725 Continental Hotel      -34.606234       -58.376724             416
3                  Alimac      -34.601557       -58.374829             605
4         All Saints Cafe      -34.603559       -58.378070             241
5                Arcangel      -34.607182       -58.379870             333
6      Asador La Estancia      -34.602488       -58.380096             200
7  BIGG CrossFit Downtown      -34.604621       -58.376212             401
8                Banchero      -34.603985       -58.385112             416
9             Bar The Bar      -34.608303       -58.379013             473

           Venue Category
0            Burger Joint
1                   Hotel
2                   Hotel
3        Video Game Store
4             Coffee Shop
5              Restaurant
6  Argentinian Restaurant
7    Gym / Fitness Center
8             Pizza Place
9              Restaurant
```

[24]:

```
#Remove duplicated keeping closest Venue to the Station
bsas_venues=bsas_venues.drop_duplicates(subset=['Venue','Venue Latitude','Venue␣
 ↪Longitude'], keep='first').reset_index(drop=True)
print(bsas_venues.shape)
bsas_venues.head(10)
```

```
(2804, 9)
```

[24]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574
5            D   9 DE JULIO        -34.604245         -58.380574
6            D   9 DE JULIO        -34.604245         -58.380574
7            D   9 DE JULIO        -34.604245         -58.380574
8            D   9 DE JULIO        -34.604245         -58.380574
9            D   9 DE JULIO        -34.604245         -58.380574
```

```
                    Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0          180 Burger Bar      -34.604848       -58.380574              67
1  474 Buenos Aires Hotel      -34.602514       -58.374013             631
2   725 Continental Hotel      -34.606234       -58.376724             416
3                  Alimac      -34.601557       -58.374829             605
4          All Saints Cafe     -34.603559       -58.378070             241
5                Arcangel      -34.607182       -58.379870             333
6       Asador La Estancia     -34.602488       -58.380096             200
7  BIGG CrossFit Downtown      -34.604621       -58.376212             401
8                Banchero      -34.603985       -58.385112             416
9             Bar The Bar      -34.608303       -58.379013             473

           Venue Category
0            Burger Joint
1                   Hotel
2                   Hotel
3        Video Game Store
4             Coffee Shop
5              Restaurant
6  Argentinian Restaurant
7    Gym / Fitness Center
8             Pizza Place
9              Restaurant
```

4.0.7 Add Borough to each Venue

[25]:
```
bsas_venues['Borough'] =  'NaN'
bsas_venues.head(5)
```

[25]:
```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574

                    Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0          180 Burger Bar      -34.604848       -58.380574              67
1  474 Buenos Aires Hotel      -34.602514       -58.374013             631
2   725 Continental Hotel      -34.606234       -58.376724             416
3                  Alimac      -34.601557       -58.374829             605
4          All Saints Cafe     -34.603559       -58.378070             241

     Venue Category Borough
0      Burger Joint     NaN
1             Hotel     NaN
```

```
2             Hotel     NaN
3  Video Game Store     NaN
4       Coffee Shop     NaN
```

[26]:
```
for i in range(len(bsas_venues)):
    #print('Latitude', bsas_venues.iloc[i,5])
    #print('Longitude', bsas_venues.iloc[i,6])
    point=Point(bsas_venues.iloc[i,6],bsas_venues.iloc[i,5])
    for b in range(len(borough)):
        poly = shapely.wkt.loads(borough.iloc[b,0])
        if (poly.contains(point)):
            bsas_venues.iloc[i,9]=borough.iloc[b,1]
bsas_venues.head()
```

[26]:
```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D    9 DE JULIO        -34.604245         -58.380574
1            D    9 DE JULIO        -34.604245         -58.380574
2            D    9 DE JULIO        -34.604245         -58.380574
3            D    9 DE JULIO        -34.604245         -58.380574
4            D    9 DE JULIO        -34.604245         -58.380574

                  Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0        180 Burger Bar      -34.604848       -58.380574              67
1  474 Buenos Aires Hotel    -34.602514       -58.374013             631
2   725 Continental Hotel    -34.606234       -58.376724             416
3                 Alimac     -34.601557       -58.374829             605
4        All Saints Cafe     -34.603559       -58.378070             241

     Venue Category      Borough
0      Burger Joint  SAN NICOLAS
1             Hotel  SAN NICOLAS
2             Hotel  SAN NICOLAS
3  Video Game Store  SAN NICOLAS
4       Coffee Shop  SAN NICOLAS
```

[28]:
```
print(bsas_venues.shape)
```

```
(2804, 10)
```

[29]:
```
bsas_venues.head()
```

[29]:
```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D    9 DE JULIO        -34.604245         -58.380574
1            D    9 DE JULIO        -34.604245         -58.380574
2            D    9 DE JULIO        -34.604245         -58.380574
3            D    9 DE JULIO        -34.604245         -58.380574
4            D    9 DE JULIO        -34.604245         -58.380574
```

```
                     Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0           180 Burger Bar      -34.604848       -58.380574              67
1   474 Buenos Aires Hotel      -34.602514       -58.374013             631
2    725 Continental Hotel      -34.606234       -58.376724             416
3                   Alimac      -34.601557       -58.374829             605
4           All Saints Cafe     -34.603559       -58.378070             241

     Venue Category      Borough
0      Burger Joint  SAN NICOLAS
1             Hotel  SAN NICOLAS
2             Hotel  SAN NICOLAS
3  Video Game Store  SAN NICOLAS
4       Coffee Shop  SAN NICOLAS
```

4.0.8 Explore by Category to get only Cafes of Coffe Shops

Since Venues per station reached Foursqueare maximum hits, will develop new explore query to bring only Cafe or Coffe Shopps Venues for a more detailed set of data

```
[33]: #Define new function with category explore for Caffe or Coffe Shops
def getNearbyVenuesCat(names, line, latitudes, longitudes):
    LIMIT = 200 # limit of number of venues returned by Foursquare API
    radius = 800 # define radius
    venues_list=[]
    for name, line, lat, lng in zip(names, line, latitudes, longitudes):
         # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?
 ↪&client_id={}&client_secret={}&v={}&ll={},{}&categoryId=4bf58dd8d48988d1e0931735&radius={}&
 ↪format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            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([(
            line,
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['location']['distance'],
            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 = ['Station Line',
                  'Station Name',
                  'Station Latitude',
                  'Station Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Distance',
                  'Venue Category']
    return(nearby_venues)
```

[34]:

```
#Create dataframe with Venues category of Caffe or Coffe Shops in Buenos Aires␣
 ↪near subway stations
bsas_cafes = getNearbyVenuesCat(names=sub['Station Name'],
                                   line=sub['Station Line'],
                                   latitudes=sub['lat'],
                                   longitudes=sub['long']
                                   )
bsas_cafes.to_csv (r'cafes.csv', index = None, header=True)
```

[35]:

```
bsas_cafes=pd.read_csv(r'cafes.csv')
print(bsas_cafes.shape)
bsas_cafes.head()
```

```
(3699, 9)
```

[35]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574

                 Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0      All Saints Cafe      -34.603559       -58.378070             241
1    Tostado Café Club      -34.603494       -58.380793              85
2            Starbucks      -34.603542       -58.378715             187
3  NEGRO. Cueva de café      -34.600848       -58.379612             388
4            Starbucks      -34.604619       -58.381821             121
```

```
  Venue Category
0    Coffee Shop
1    Coffee Shop
2    Coffee Shop
3    Coffee Shop
4    Coffee Shop
```

[36]:

```
bsas_cafes["Venue Category"].replace("Coffee Shop","Café",inplace=True)
bsas_cafes.head()
```

[36]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574

                   Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0        All Saints Cafe      -34.603559       -58.378070             241
1      Tostado Café Club      -34.603494       -58.380793              85
2              Starbucks      -34.603542       -58.378715             187
3  NEGRO. Cueva de café      -34.600848       -58.379612             388
4              Starbucks      -34.604619       -58.381821             121

  Venue Category
0           Café
1           Café
2           Café
3           Café
4           Café
```

Following table shows the venues reached the limit of 100 venues from foursquare

[37]:

```
#Check quantity if venues
bsas_cafes.set_index(["Station Name", "Station Line"]).count(level="Station␣
 ↪Name").head()
```

[37]:

```
                   Station Latitude  Station Longitude  Venue  Venue Latitude  \
Station Name
9 DE JULIO                      100                100    100             100
ACOYTE                           30                 30     30              30
AGÜERO                           58                 58     58              58
ALBERTI                          23                 23     23              23
ALMAGRO - MEDRANO                27                 27     27              27

                   Venue Longitude  Venue Distance  Venue Category
```

```
Station Name
9 DE JULIO                     100              100              100
ACOYTE                          30               30               30
AGÜERO                          58               58               58
ALBERTI                         23               23               23
ALMAGRO - MEDRANO               27               27               27
```

but still need to cleanup duplicates so lets

4.0.9 Eliminate Venues duplicates

[38]:
```
bsas_cafes=bsas_cafes.sort_values(by=['Station Name','Venue','Venue␣
 ↪Latitude','Venue Longitude','Venue Distance']).reset_index(drop=True)
print(bsas_cafes.shape)
bsas_cafes.head()
```

```
(3699, 9)
```

[38]:
```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574
1            D   9 DE JULIO        -34.604245         -58.380574
2            D   9 DE JULIO        -34.604245         -58.380574
3            D   9 DE JULIO        -34.604245         -58.380574
4            D   9 DE JULIO        -34.604245         -58.380574

             Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0  All Saints Cafe      -34.603559       -58.378070             241
1       Alma Café      -34.604919       -58.382356             179
2          Anamora      -34.602653       -58.385472             482
3         Bar Dado      -34.605032       -58.387940             680
4    Bernardo Café      -34.609498       -58.380731             584

  Venue Category
0           Café
1           Café
2           Café
3           Café
4           Café
```

[39]:
```
#Remove duplicated keeping closest Venue to the Station
bsas_cafes=bsas_cafes.drop_duplicates(subset=['Venue','Venue Latitude','Venue␣
 ↪Longitude'], keep='first').reset_index(drop=True)
print(bsas_cafes.shape)
bsas_cafes.head(5)
```

```
(893, 9)
```

[39]:

```
  Station Line Station Name  Station Latitude  Station Longitude  \
0            D   9 DE JULIO        -34.604245         -58.380574   
1            D   9 DE JULIO        -34.604245         -58.380574   
2            D   9 DE JULIO        -34.604245         -58.380574   
3            D   9 DE JULIO        -34.604245         -58.380574   
4            D   9 DE JULIO        -34.604245         -58.380574   

            Venue  Venue Latitude  Venue Longitude  Venue Distance  \
0  All Saints Cafe      -34.603559       -58.378070             241   
1        Alma Café      -34.604919       -58.382356             179   
2          Anamora      -34.602653       -58.385472             482   
3         Bar Dado      -34.605032       -58.387940             680   
4    Bernardo Café      -34.609498       -58.380731             584   

  Venue Category  
0           Café  
1           Café  
2           Café  
3           Café  
4           Café  
```

[40]:

```
bsas_cafes.set_index(["Station Name", "Station Line"]).count(level="Station␣
 ↪Name").head()
```

[40]:

```
                   Station Latitude  Station Longitude  Venue  Venue Latitude  \
Station Name                                                                    
9 DE JULIO                      100                100    100             100   
ACOYTE                           30                 30     30              30   
AGÜERO                           58                 58     58              58   
ALBERTI                          23                 23     23              23   
ALMAGRO - MEDRANO                27                 27     27              27   

                   Venue Longitude  Venue Distance  Venue Category  
Station Name                                                        
9 DE JULIO                     100             100             100  
ACOYTE                          30              30              30  
AGÜERO                          58              58              58  
ALBERTI                         23              23              23  
ALMAGRO - MEDRANO               27              27              27  
```

[41]:

```
bsas_cafes[['Station Name','Station Line','Venue']].groupby(['Station␣
 ↪Name','Station Line']).count().sort_values('Venue',ascending=False).head()
```

[41]:

```
                           Venue
Station Name Station Line       
9 DE JULIO   D               100
CALLAO       D                80
```

```
AGÜERO      D                 58
CATALINAS   E                 55
AV. DE MAYO C                 40
```

4.1 ————-