INEGI_Asymm_NL

June 4, 2020

1 import libraries

[1]:
```
import pandas as pd
import requests
import json
import datetime
import folium
from folium import plugins
import time
import matplotlib
```

2 Read the geo json with the geometric shape of the regions

[2]:
```
# file name, for all the municipalities in the Mexican state of Nuevo León
file_geo_json_nl = 'Munxentidad19_MOD.json' # nuevo leon

# read nl
with open(file_geo_json_nl) as f:
    geo_json_nl = json.load(f)
```

[3]:
```
# list of codes for municipalities
mun_code_list = []
mun_name_list = []

for municipio in geo_json_nl['features']:
    mun_code_list.append( municipio['properties']['mun_code'] )
    mun_name_list.append( municipio['properties']['mun_name'] )

n_mun = len(mun_code_list)
```

[4]:
```
# adjust strings with only one digit to have two digits
for i in list(range(0,len(mun_code_list))):
    if len(mun_code_list[i])==1:
        mun_code_list[i] = '0'+mun_code_list[i]
```

3 Read from INEGI data bank

3.1 my tokens and test urls

[7]:
```
# look at
# 'https://www.inegi.org.mx/app/api/indicadores/desarrolladores/jsonxml/
 ↪INDICATOR/1002000002,1002000003/en/19026/true/BISE/2.0/'+key+'?type=json'
# there, the 19 is for Nuevo León State and the 026 is for Guadalupe␣
 ↪Municipality. I will look through all the municipalities in the text

key = 'XXXXXXX'

url_Ini = 'https://www.inegi.org.mx/app/api/indicadores/desarrolladores/jsonxml/
 ↪INDICATOR/'
indicadores = '1002000002,1002000003'
```

3.2 read

[8]:
```
json_mun_list = []

print('begin---',datetime.datetime.now())

#for i, mun_code in zip( list(range(0,len(mun_code_list))) , mun_code_list ):
for mun_code in mun_code_list:
    time.sleep(0.0025)
    resp_tmp = requests.get( url_Ini + indicadores + '/en/' + '190' + mun_code␣
 ↪+ '/true/BISE/2.0/'+key+'?type=json' )
    if resp_tmp.status_code==200:
        json_tmp = json.loads(resp_tmp.content)
        json_mun_list.append(json_tmp)

print('end-----',datetime.datetime.now())
```

```
begin--- 2020-05-19 08:15:20.589266
end----- 2020-05-19 08:15:46.512670
```

[9]:
```
# dataframe with Nuevo Leon State valid Municipalities
df_nl = pd.DataFrame(columns=('Municipality', 'Code', 'People', 'Males',␣
 ↪'Females', 'Asymmetry'))

n_valid_mun = 0;
for mun_name, mun_code, data_mun in zip( mun_name_list, mun_code_list,␣
 ↪json_mun_list ):
    if data_mun['Series'] != None:
        if (
```

```
            (
                data_mun['Series'][0]['INDICADOR']=='1002000002'
            ) and (
                data_mun['Series'][0]['OBSERVATIONS'][-1]['TIME_PERIOD']=='2010'
            ) and (
                data_mun['Series'][1]['INDICADOR']=='1002000003'
            ) and (
                data_mun['Series'][1]['OBSERVATIONS'][-1]['TIME_PERIOD']=='2010'
            )
        ):
            df_nl.loc[n_valid_mun] = [
                mun_name,#
                mun_code,# string with two digit number
                # total
                (
                    (
                        int(float(␣
↪data_mun['Series'][0]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                    ) + (
                        int(float(␣
↪data_mun['Series'][1]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                    )
                ),
                # males
                int(float(␣
↪data_mun['Series'][0]['OBSERVATIONS'][-1]['OBS_VALUE'] )),
                # females
                int(float(␣
↪data_mun['Series'][1]['OBSERVATIONS'][-1]['OBS_VALUE'] )),
                # asymmetry males - females. If negative there are more females␣
↪than males
                (
                    (
                        (
                            int(float(␣
↪data_mun['Series'][0]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                        ) - (
                            int(float(␣
↪data_mun['Series'][1]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                        )
                    )/(
                        (
                            int(float(␣
↪data_mun['Series'][0]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                        ) + (
```

```
                            int(float(␣
↪data_mun['Series'][1]['OBSERVATIONS'][-1]['OBS_VALUE'] ))
                        )
                    )
                )
            ]
        else:
            df_nl.loc[n_valid_mun] = [
                mun_name,#
                mun_code,# string with two digit number,
                'ERROR','ERROR','ERROR'
            ]
        n_valid_mun = n_valid_mun+1
```

[10]:
```
# store csv with municipalities
df_nl.to_csv('datos_municipios_nl.csv')
```

[11]:
```
# set 'Code' as the index of the dataframe
df_nl.set_index('Code',inplace=True)
```

[12]:
```
df_nl.shape
```

[12]:
```
(51, 5)
```

[13]:
```
df_nl.dtypes
```

[13]:
```
Municipality     object
People           object
Males            object
Females          object
Asymmetry       float64
dtype: object
```

[16]:
```
df_nl.head(n_valid_mun)
```

[16]:

Code	Municipality	People	Males	Females	Asymmetry
26	Guadalupe	678006	336731	341275	-0.006702
31	Juárez	256970	129324	127646	0.006530
41	Pesquería	20843	10737	10106	0.030274
49	Santiago	40469	20341	20128	0.005263
18	García	143668	72640	71028	0.011220
48	Santa Catarina	268955	134388	134567	-0.000666
39	Monterrey	1135550	561656	573894	-0.010777
19	San Pedro Garza García	122659	57622	65037	-0.060452
10	Carmen	16092	8175	7917	0.016033
21	Gral. Escobedo	357937	180332	177605	0.007619

45	Salinas Victoria	32660	16800	15860	0.028781
25	Gral. Zuazua	55213	27787	27426	0.006538
06	Apodaca	523370	263374	259996	0.006454
12	Ciénega de Flores	24526	12476	12050	0.017369
47	Hidalgo	16604	8318	8286	0.001927
01	Abasolo	2791	1406	1385	0.007524
13	China	10864	5585	5279	0.028166
20	Gral. Bravo	5527	2876	2651	0.040709
03	Los Aldamas	1374	708	666	0.030568
15	Dr. Coss	1716	889	827	0.036131
51	Villaldama	4113	2073	2040	0.008023
44	Sabinas Hidalgo	34671	17214	17457	-0.007009
46	San Nicolás de los Garza	443273	219337	223936	-0.010375
08	Bustamante	3773	1890	1883	0.001855
33	Linares	78669	39104	39565	-0.005860
50	Vallecillo	1971	1065	906	0.080670
43	Rayones	2628	1363	1265	0.037291
42	Los Ramones	5359	2720	2639	0.015115
40	Parás	1034	512	522	-0.009671
38	Montemorelos	59113	29370	29743	-0.006310
37	Mina	5447	2774	2673	0.018542
36	Mier y Noriega	7095	3555	3540	0.002114
35	Melchor Ocampo	862	433	429	0.004640
34	Marín	5488	2808	2680	0.023324
32	Lampazos de Naranjo	5349	2700	2649	0.009534
30	Iturbide	3558	1844	1714	0.036537
29	Hualahuises	6914	3424	3490	-0.009546
28	Higueras	1594	829	765	0.040151
27	Los Herreras	2030	1001	1029	-0.013793
24	Gral. Zaragoza	5942	3052	2890	0.027264
23	Gral. Treviño	1277	626	651	-0.019577
22	Gral. Terán	14437	7234	7203	0.002147
17	Galeana	39991	20237	19754	0.012078
16	Dr. González	3345	1713	1632	0.024215
14	Dr. Arroyo	35445	17849	17596	0.007138
11	Cerralvo	7855	4049	3806	0.030936
09	Cadereyta Jiménez	86445	43939	42506	0.016577
07	Aramberri	15470	7869	7601	0.017324
05	Anáhuac	18480	9219	9261	-0.002273
04	Allende	32593	16440	16153	0.008806
02	Agualeguas	3443	1777	1666	0.032239

4 Colors

[17]:
```
# this uses
# see https://www.codespeedy.com/convert-rgb-to-hex-color-code-in-python/
# import matplotlib
```

[18]:
```
# color function red for -max of the abs of the values, blue for +max of the␣
 ↪abs of the values
def color_rwb(x,xMax,xMin):
    xAbsMax = max( abs(xMax), abs(xMin) )

    if 0<x:
        xCol = matplotlib.colors.to_hex(
            [
                1-x/xAbsMax,
                1-x/xAbsMax,
                1
            ]
        )

    if x<0:
        xCol = matplotlib.colors.to_hex(
            [
                1,
                1+x/xAbsMax,
                1+x/xAbsMax
            ]
        )

    return xCol
```

[19]:
```
#color_rb(0.006538315251842863,max_Asymm,min_Asymm)
```

[20]:
```
#import branca # works
```

[21]:
```
#import geopandas # works
```

5 Make map

[22]:
```
latCen_nl = +024.95
lonCen_nl = -100.33

# define a map
map_nl = folium.Map(location=[latCen_nl, lonCen_nl], width=300, height=400,␣
 ↪zoom_start=6, min_zoom=6, max_zoom=14)
```

[23]:
```
# see
# https://github.com/python-visualization/folium/blob/master/examples/
 ↪GeoJSON_and_choropleth.ipynb

max_Asymm = df_nl['Asymmetry'].max()
min_Asymm = df_nl['Asymmetry'].min()

# nuevo intento con la asimetría
folium.GeoJson(
    r'Munxentidad19_MOD.json',
    style_function=lambda feature: {
#        'fillColor': 'blue' if ( df_nl['Asymmetry'].loc[␣
 ↪feature['properties']['mun_code'] ]>0 ) else 'red',
        'fillColor': color_rwb( df_nl['Asymmetry'].loc[␣
 ↪feature['properties']['mun_code'] ] ,max_Asymm,min_Asymm),
        'fillOpacity': 0.6,
        'color': 'black',
        'weight': 1
    }
).add_to(map_nl)
```

[23]:
```
<folium.features.GeoJson at 0x7fa06847aa58>
```

[24]:
```
map_nl
```

[24]:
```
<folium.folium.Map at 0x7fa0684c9b00>
```

[25]:
```
map_nl.save('INEGI_Asymm_NL.html')
```

[ ]:
```

```