# Notebook 4: Interactive Map for National Migration Outflow and Their Paths

- side note: because we tried to do some interactive maps that contain many layers of information, we decided to take each notebook to focus on one or two specific maps so that the flow and the organization of the project are more straightforward. The cells for this map are usually compiled into single cells in the end.

In this notebook, only one map is focused on – the interactive map that shows where the migration goes to and where are the outflows of migration. The note begins with processing geodata for all metros. Then, we processed the migration flow census data for the outflow numbers for each county. After matching the census and geodata, we were able to produce an very interactive map in the end.

```
In [1]:  # Import all modules I will be using in this notebook

         import pandas as pd
         import geopandas as gpd
         import contextily as ctx
         import matplotlib.pyplot as plt
         import plotly.express as px
         import plotly.graph_objects as go
         from plotly.subplots import make_subplots
         import numpy as np
         import fiona
```

/opt/conda/lib/python3.8/site-packages/geopandas/_compat.py:106: UserWarning: The Shapely GEOS version (3.8.1-CAPI-1.13.3) is incompatible with the GEOS version PyGEOS was compiled with (3.9.0-CAPI-1.16.2). Conversions between both will be slow.
  warnings.warn(

## Process Metro Geo Data

```
In [2]:  # Import the geo data SHP file

         metro_geodata_rawdata = gpd.read_file('GeoData/01_Basemap_metroborder/tl_2019_us_cbsa.shp')
```

```
In [3]:  # Take a look to see if all data are valid and in the right data type.
         # There's one missing a lot of values, but it's okay because it's not important to our questions

         metro_geodata_rawdata.head()
```
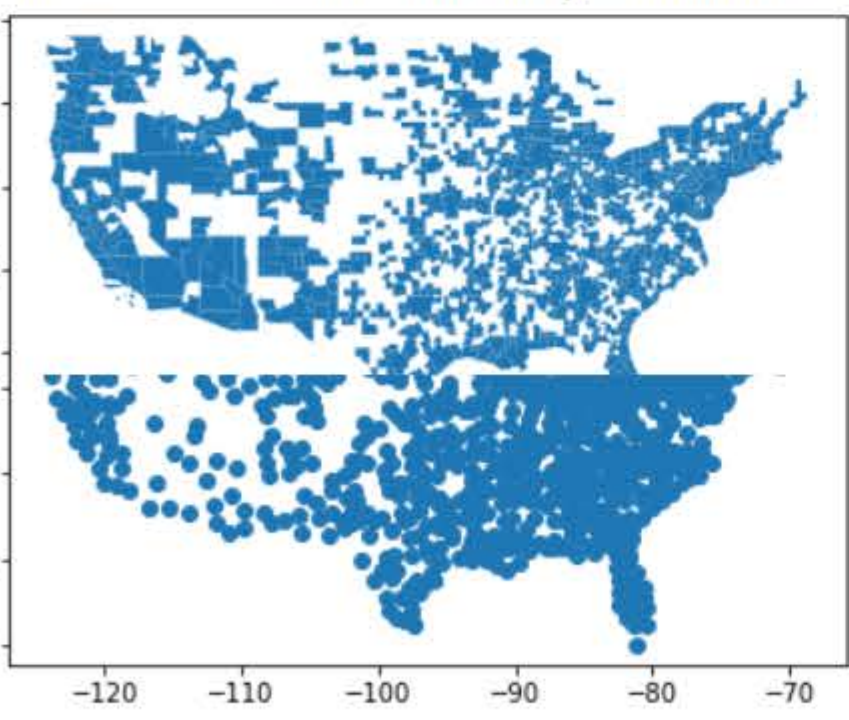
Out[3]:

| | CSAFP | CBSAFP | GEOID | NAME | NAMELSAD | LSAD | MEMI | MTFCC | ALAND | AWATER | INTPTLAT | INTPTLON | geometry |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 122 | 12020 | 12020 | Athens-Clarke County, GA | Athens-Clarke County, GA Metro Area | M1 | 1 | G3110 | 2654601832 | 26140309 | +33.9439840 | -083.2138965 | POLYGON ((-83.53739 33.96591, -83.53184 33.968... |

see the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  super(GeoDataFrame, self).__setitem__(key, value)

```
In [5]:  metro_geodata_ready.head()
```

Out[5]:

| | GEOID | Metro_Name | geometry | State |
|---|---|---|---|---|
| 0 | 12020 | Athens-Clarke County, GA Metro Area | POLYGON ((-83.53739 33.96591, -83.53184 33.968... | GA |



| | | Area2 | Rico | - |
|---|---|---|---|---|

| | 10180 | 10740 | Abilene, TX Metro Area | 167649 | 130384 | 21221 | 11121 | 3883 | 1040 | Albuquerque, NM Metro Area | 897771 | 773804 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 10180 | 10740 | Abilene, TX Metro Area | 167649 | 130384 | 21221 | 11121 | 3883 | 1040 | Albuquerque, NM Metro Area | 897771 | 773804 |
| 1 | 10180 | 11100 | Abilene, TX Metro Area | 167649 | 130384 | 21221 | 11121 | 3883 | 1040 | Amarillo, TX Metro Area | 262324 | 212065 |
| 2 | 10180 | 12060 | Abilene, TX Metro Area | 167649 | 130384 | 21221 | 11121 | 3883 | 1040 | Atlanta-Sandy Springs- | 5667591 | 4846011 |

```
 #   Column                                              Non-Null Count  Dtype
---  ------                                              --------------  -----
 0   Current Residence Metro Code1                       49739 non-null  object
 1   Residence 1 Year Ago Metro Code1                    49739 non-null  object
 2   Metropolitan Statistical Area of Current Residence  49739 non-null  object
 3   Population 1 Year and Over                           49739 non-null  int64
 4   Nonmovers                                           49739 non-null  int64
 5   Movers within Same Metropolitan Statistical Area    49739 non-null  int64
 6   Movers from Different Metropolitan Statistical Area2 49739 non-null  int64
```

| 49735 | 48300 | Wenatchee, WA Metro Area | 49740 | Yuma, AZ Metro Area | 10 |
|---|---|---|---|---|---|
| 49736 | 48660 | Wichita Falls, TX Metro Area | 49740 | Yuma, AZ Metro Area | 3 |
| 49737 | 49420 | Yakima, WA Metro Area | 49740 | Yuma, AZ Metro Area | 18 |
| 49738 | 49700 | Yuba City, CA Metro Area | 49740 | Yuma, AZ Metro Area | 22 |

49739 rows × 5 columns

```
In [25]:  Migration2018_total_outflow_trimmed = Migration2018_total_outflow_trimmed.sort_values(by='Total_Outflow_Population',
                                                                                              ascending=False,
                                                                                              ignore_index = True)
```

```
In [26]:  Migration2018_total_outflow_trimmed
```

Out[26]:

| | GEOID | Total_Outflow_Population | | Metro | | geometry | lon | lat |
|---|---|---|---|---|---|---|---|---|
| 1 | 10740 | Albuquerque, NM Metro Area | 10900 | Allentown-Bethlehem-Easton, PA-NJ Metro Area | 14.0 | Albuquerque, NM Metro Area | POINT (-106.47158 35.12229) | -106.471584 | 35.12229 |
| 2 | 10740 | Albuquerque, NM Metro Area | 11100 | Amarillo, TX Metro Area | 153.0 | Albuquerque, NM Metro Area | POINT (-106.47158 35.12229) | -106.471584 | 35.12229 |
| 3 | 10740 | Albuquerque, NM Metro Area | 11260 | Anchorage, AK Metro Area | 133.0 | Albuquerque, NM Metro Area | POINT (-106.47158 35.12229) | -106.471584 | 35.12229 |

```
In [36]:  Migration2018_Trimmed2_geo = Migration2018_Trimmed2_geo.drop(['Metro'],axis=1)
```

```
In [37]:  Migration2018_Trimmed2_geo
```

Out[37]:

| | Previous_GEOID | Previous_Metro | GEOID | Current_Metro | Migration_Population | Previous_lon | Previous_lat | geometry | lon | lat |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 10740 | Albuquerque, NM Metro Area | 10180 | Abilene, TX Metro Area | 3.0 | -106.471584 | 35.122290 | POINT (-99.71768 32.44969) | -99.717678 | 32.449690 |

```
In [40]:  Migration2018_top = Migration2018_top.reset_index()
```

```
In [41]:  Migration2018_top = Migration2018_top.drop(['index'],axis=1)
```

```
In [42]:  Migration2018_top
```

Out[42]:

| 173 | 24660 | High Point, NC Metro Area | 49180 | Salem, NC Metro Area | 5916.0 | -79.791602 | 36.025861 | (-80.34581 36.07664) | -80.345813 | 36.076644 |
|---|---|---|---|---|---|---|---|---|---|---|

174 rows × 10 columns

```
In [ ]:
```

```
In [43]:  fig = go.Figure()

                  marker = dict(
                      size = Migration2018_total_outflow_trimmed_sub['Total_Outflow_Population']/scale,
                      color = colors[i],
                      line_color='rgb(40,40,40)',
                      line_width=0.5,
                      sizemode = 'area'
                  ),
                  name = '{0} - {1}'.format(lim[0],lim[1])))

          # Format
                          )
                      )
                  )

          fig.update_layout(title_font_size=15)

          fig.update_geos(fitbounds="locations")

          fig.show("notebook")
```

We are glad the interactive map works out. We tried to add many layers of information into one map. The indexing part is challenging, especially within a for loop code, but we learned a lot from this code.

The map itself presents where the "hot spots" for outflow migration are and how the flows in between. New York City and LA lost many people and the major flows are within those big metros as well.

```
In [ ]:
```