

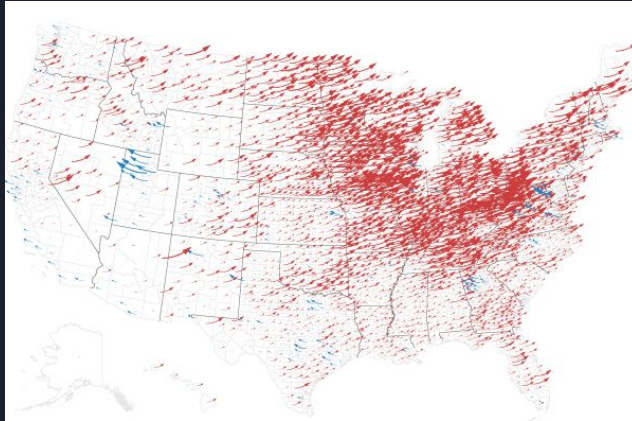


# Interactive Geospatial Data Visualization in Python

Pradyoth Velagapudi

# Election Data

- One of the biggest stories of the 2024 election: DATA
- How is election data different from the data we dealt with in class?
  - Inextricably linked with a geospatial component



[How Trump Reshaped the Election Map - The New York Times](#)

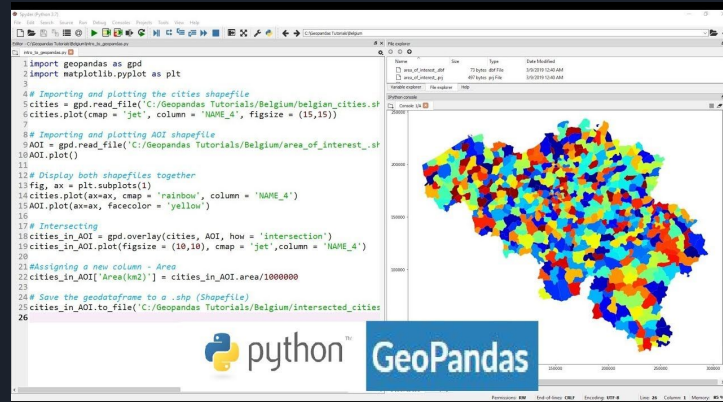
Guiding question: How can I use Python to represent data with a geospatial element?

# GeoPandas

Powerful open-source library that allows seamless integration of geospatial data

Compatible with Pandas and matplotlib

Reads shapefiles to create geodataframes (analogous to Pandas dataframes) containing polygon geometry

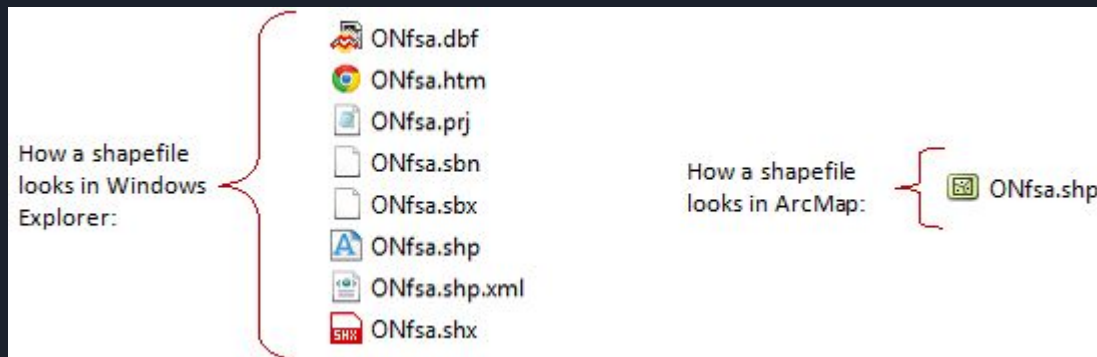


# Shapefiles

File format used in GIS (geographic information systems) to store location, shape, and attributes of geographic features as a collection of points, lines, and polygons

Used by industry standard technologies such as arcGIS

Consists of a collection of files dependent on one another, including .shp, .shx, .dbf, and .prj files





# Loading Data

Can load shapefiles using GeoPandas to create a geodataframe

Geodataframes can be merged with regular Pandas dataframes on a common variable. This allows you to append a geographic component to existing data.

```
#load election data
election_data_2020 = pd.read_csv("az20.csv")
election_data_2024 = pd.read_csv("az24.csv")

#load county map shapefile data
shapefile_path = "az_county/az_county.shp"
counties = gpd.read_file(shapefile_path)
```

```
#clean data on county name column to match between datasets
counties["county"] = counties["NAME"].str.upper().str.strip()
election_data_2020["county"] = election_data_2020["county"].str.upper().str.strip()
election_data_2024["county"] = election_data_2024["county"].str.upper().str.strip()

#merge shapefile with election results data on the county column
counties_2020 = counties.merge(election_data_2020, on="county")
counties_2024 = counties.merge(election_data_2024, on="county")
```

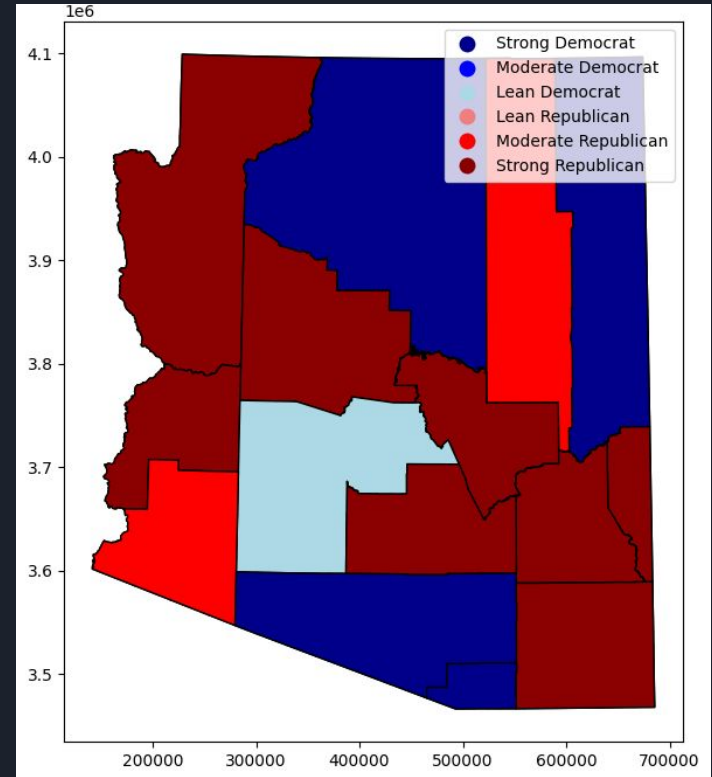
# Plotting Data

Use matplotlib to plot the geodataframe

I put the % margin of victory into discrete bins and defined a custom colormap to color each county according to its results.

```
#plot function to update map based on selected year  
fig, ax = plt.subplots(1, 1, figsize=(12, 8))
```

```
#display data on map  
plot = counties_data.plot(  
    column="percentage_margin_category",  
    cmap=cmap,  
    legend=True,  
    ax=ax,  
    edgecolor='black'  
)  
ax.set_title(f"Arizona {year} Presidential Election Results by County (Percentage Margin)")
```

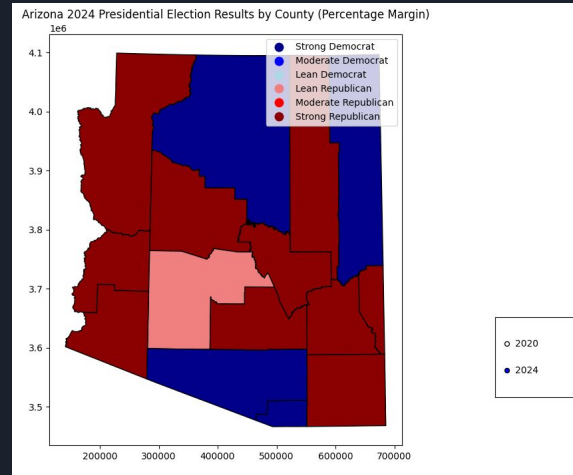
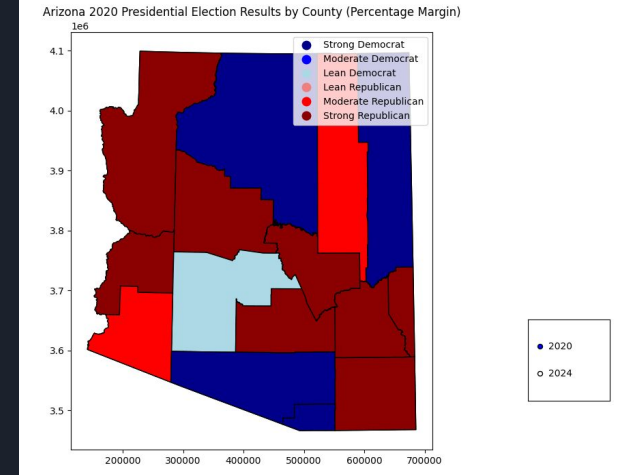


# Adding Interactivity

I used matplotlib widgets to add a radio button to toggle the year

```
#create radio buttons to switch between 2020 and 2024 election results
def radio_func(label):
    #switch between datasets based on selected radio button
    if label == "2020":
        update_plot(2020)
    elif label == "2024":
        update_plot(2024)

radio = RadioButtons(
    ax=plt.axes([0.85, 0.2, 0.1, 0.15]),
    labels=["2020", "2024"],
    active=1, #default to 2024
)
radio.on_clicked(radio_func)
```



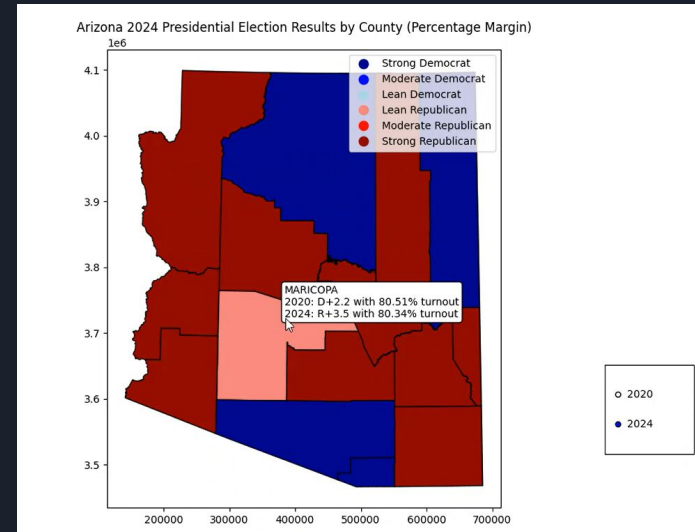
# Adding Interactivity

I wanted to add hover functionality that showed further details when the user hovers their cursor above a county

GeoPandas has `points_from_xy()` function that returns geometry given an x/y coordinate. Mouse coords can be found using matplotlibs. The geodataframe can be searched to return the county whose geometry contains that point. Event handling in matplotlib can attach a function to an event

```
73
74 #mouse hover event handler
75 def on_hover(event):
76     if event.inaxes == ax: #ensure the click is within the map
77         #convert cursor coordinates to GeoDataFrame CRS
78         coords = gpd.points_from_xy([event.xdata], [event.ydata], crs=counties.crs)
79         point = coords[0] # Only one point
80
81         #determine which county was clicked
82         clicked_2020 = counties_2020[counties_2020.geometry.contains(point)]
83         clicked_2024 = counties_2024[counties_2024.geometry.contains(point)]
```

```
100
109 fig.canvas.mpl_connect("motion_notify_event", on_hover)
```







# Demo

Source code available at <https://github.com/PradyothVelagapudi/Election-Data-Visualization>

Thank You!

