Income Inequality Data Analytics

This notebook visualizes income inequality data geospatially, comparing various years altogether.

In [ ]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import geopandas as gpd
import matplotlib.pyplot as plt
```

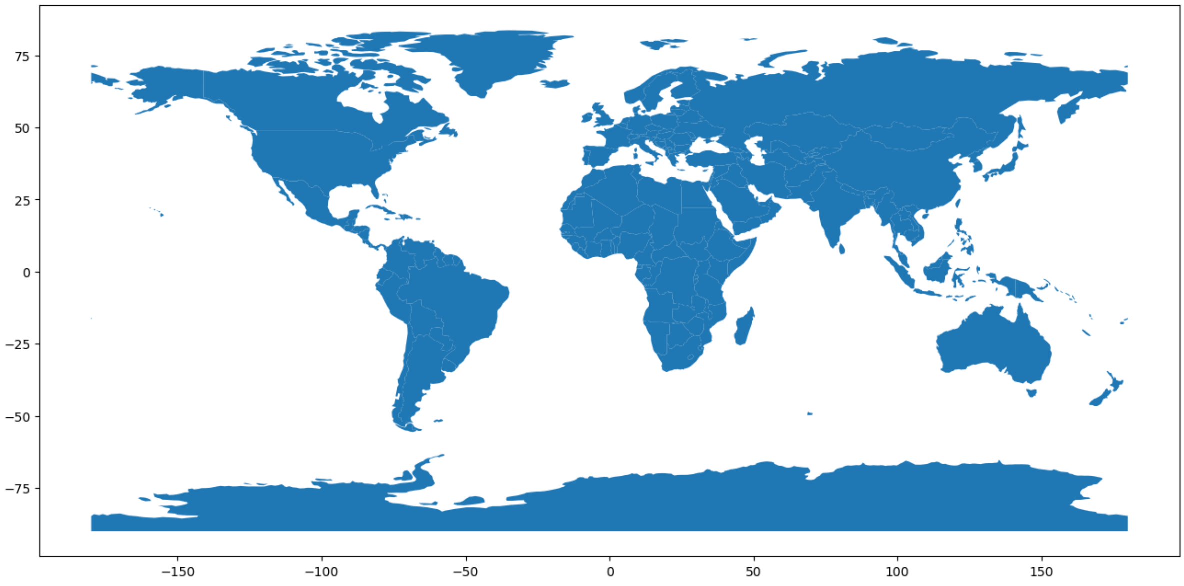
In [ ]:
```
world_data = gpd.read_file('../world-data.geojson', driver='geojson', crs=43
# Updates to names so merge finds as many countries as possible
world_data['name'] = np.where(world_data['name'].str.contains('United States
world_data['name'] = np.where(world_data['name'].str.contains('Dem. Rep. Con
world_data['name'] = np.where(world_data['name'].str.contains('Congo'), 'Con
world_data['name'] = np.where(world_data['name'].str.contains("Côte d'Ivoire
world_data['name'] = np.where(world_data['name'].str.contains("eSwatini"), '
world_data['name'] = np.where(world_data['name'].str.contains("Slovakia"), '
world_data['name'] = np.where(world_data['name'].str.contains("Bosnia and He
world_data['name'] = np.where(world_data['name'].str.contains("Laos"), "Lao'
world_data['name'] = np.where(world_data['name'].str.contains("Central Afric
world_data['name'] = np.where(world_data['name'].str.contains("Czechia"), "C
world_data['name'] = np.where(world_data['name'].str.contains("Kyrgyzstan"),
world_data['name'] = np.where(world_data['name'].str.contains("Macedonia"),
world_data['name'] = np.where(world_data['name'].str.contains("S. Sudan"), '
world_data['name'] = np.where(world_data['name'].str.contains("Eq. Guinea"),
world_data['name'] = np.where(world_data['name'].str.contains("Dominican Rep
world_data['name'] = np.where(world_data['name'].str.contains("Solomon Is.")
world_data['name'] = np.where(world_data['name'].str.contains("N. Cyprus"),
```

In [ ]:
```
world_data[world_data['name'].str.contains('Congo')].head()
```

Out[ ]:

	iso_a3	name	continent	geometry
11	COD	Congo, Rep.	Africa	POLYGON ((29.34000 -4.49998, 29.51999 -5.41998...
67	COG	Congo, Rep.	Africa	POLYGON ((18.45307 3.50439, 18.39379 2.90044, ...

In [ ]:
```
world_data.plot(figsize=(16,8))
plt.show()
```



In [ ]:
```
continents = pd.read_csv('../CSV/continent_country.csv')
continents[continents['country'].str.contains('K')].head()
```

Out[ ]:

	continent	country
24	Africa	Kenya
79	Asia	Hong Kong, China
80	Asia	Korea, Dem. Rep.
81	Asia	Korea, Rep.
99	Asia	Kuwait

In [ ]:
```
gini_data = pd.read_csv('../CSV/gini.csv')
```

In [ ]:
```
gini_data.head()
```

Out[ ]:

	country	1800	1801	1802	1803	1804	1805	1806	1807	1808	...	2031	2032
0	Afghanistan	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	...	36.8	36.8
1	Albania	38.9	38.9	38.9	38.9	38.9	38.9	38.9	38.9	38.9	...	29.0	29.0
2	Algeria	56.2	56.2	56.2	56.2	56.2	56.2	56.2	56.2	56.2	...	27.6	27.6
3	Andorra	40.0	40.0	40.0	40.0	40.0	40.0	40.0	40.0	40.0	...	40.0	40.0
4	Angola	57.2	57.2	57.2	57.2	57.2	57.2	57.2	57.2	57.2	...	42.6	42.6

5 rows × 242 columns

In [ ]:
```
gini_data = gini_data.melt(id_vars=['country'], var_name='year', value_name=
```

In [ ]:
```
#gini_data.query('country == "France"').head()
#gini_data[gini_data['country'].str.contains('State')].head()
#gini_data['country'].unique()
```

In [ ]:
```
gini_world_data = world_data.merge(gini_data, left_on='name', right_on='cou
```

In [ ]:
```
gini_world_data[gini_world_data['country'].isna()]
```

Out[ ]:

	iso_a3	name	continent	geometry	country	year	gini
482	ESH	W. Sahara	Africa	POLYGON ((-8.66559 27.65643, -8.66512 27.58948...	NaN	NaN	NaN
4580	FLK	Falkland Is.	South America	POLYGON ((-61.20000 -51.85000, -60.00000 -51.2...	NaN	NaN	NaN
4822	GRL	Greenland	North America	POLYGON ((-46.76379 82.62796, -43.40644 83.225...	NaN	NaN	NaN
4823	ATF	Fr. S. Antarctic Lands	Seven seas (open ocean)	POLYGON ((68.93500 -48.62500, 69.58000 -48.940...	NaN	NaN	NaN
9885	PRI	Puerto Rico	North America	POLYGON ((-66.28243 18.51476, -65.77130 18.426...	NaN	NaN	NaN
31094	NCL	New Caledonia	Oceania	POLYGON ((165.77999 -21.08000, 166.59999 -21.7...	NaN	NaN	NaN
32300	TWN	Taiwan	Asia	POLYGON ((121.77782 24.39427, 121.17563 22.790...	NaN	NaN	NaN
36639	ATA	Antarctica	Antarctica	MULTIPOLYGON (((-48.66062 -78.04702, -48.15140...	NaN	NaN	NaN
38327	ABV	Somaliland	Africa	POLYGON ((48.94820 11.41062, 48.94820 11.41062...	NaN	NaN	NaN
39774	OSA	Kosovo	Europe	POLYGON ((20.59025 41.85541, 20.52295 42.21787...	NaN	NaN	NaN

In [ ]:

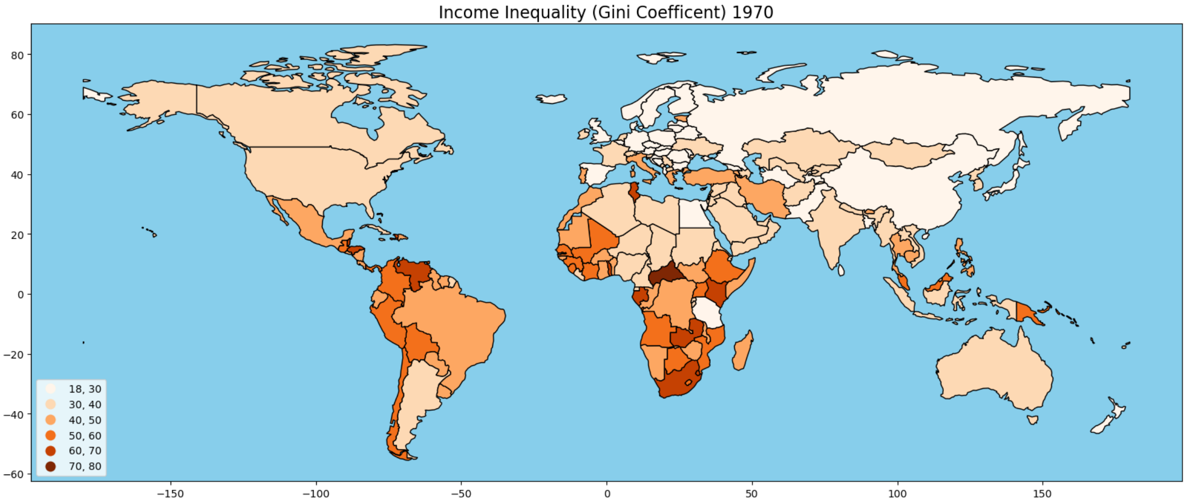
```
gini_world_data.query('year == "2002"').head()
#gini_world_data[gini_world_data['name'].str.contains('State')].head()
#gini_world_data.head()
```

Out[ ]:

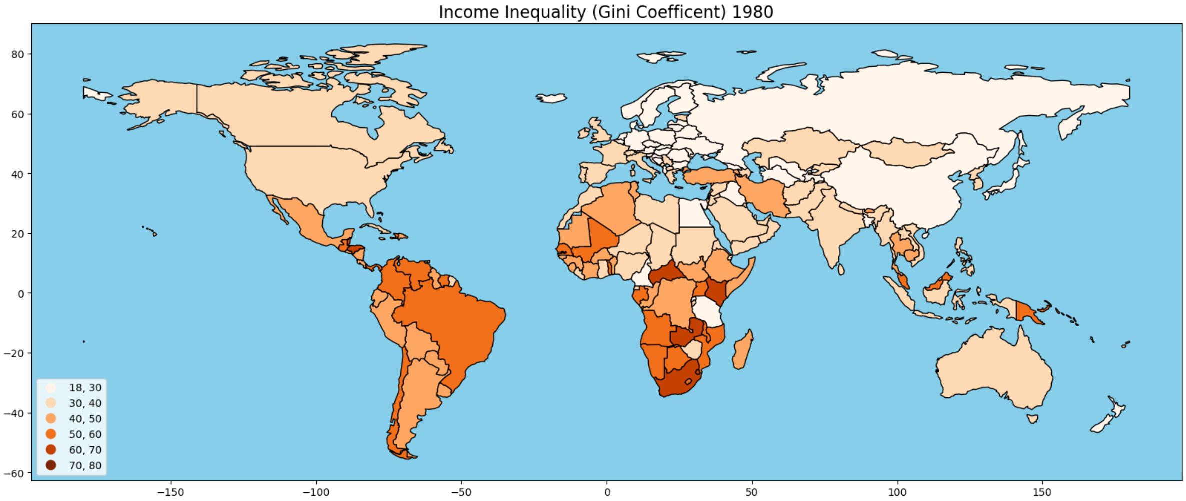
	iso_a3	name	continent	geometry	country	year	gini
202	FJI	Fiji	Oceania	MULTIPOLYGON (((180.00000 -16.06713, 180.00000...	Fiji	2002	38.2
443	TZA	Tanzania	Africa	POLYGON ((33.90371 -0.95000, 34.07262 -1.05982...	Tanzania	2002	37.9
685	CAN	Canada	North America	MULTIPOLYGON (((-122.84000 49.00000, -122.9742...	Canada	2002	33.5
926	USA	United States	North America	MULTIPOLYGON (((-122.84000 49.00000, -120.0000...	United States	2002	40.5
1167	KAZ	Kazakhstan	Asia	POLYGON ((87.35997 49.21498, 86.59878 48.54918...	Kazakhstan	2002	34.5

In [ ]:
```
years = [1970, 1980, 1990, 2000, 2010]
classifications = dict(bins=[30,40,50,60,70,80])

fig, axs = plt.subplots(len(years), 1, figsize=(20, 10*len(years)))

for ndx, yr in enumerate(years):
    ax1 = axs[ndx]
    ax1.set_facecolor('skyblue')
    ax1.set_title(f'Income Inequality (Gini Coefficent) {yr}', fontsize=16)
    gini_world_data.query(f'year == "{yr}"').plot(
        ax=ax1,
        column='gini',
        edgecolor='black',
        categorical=False,
        scheme='User_Defined',
        classification_kwds=classifications,
        #k=6,
        #scheme='fisher_jenks',
        cmap='Oranges',
        legend=True,
        legend_kwds={'loc': 'lower left', 'fmt': '{:,.0f}'}
        #,
        #missing_kwds={'color':'red','edgecolor': 'brwon', 'hatch': '///',
    )
plt.show()
```

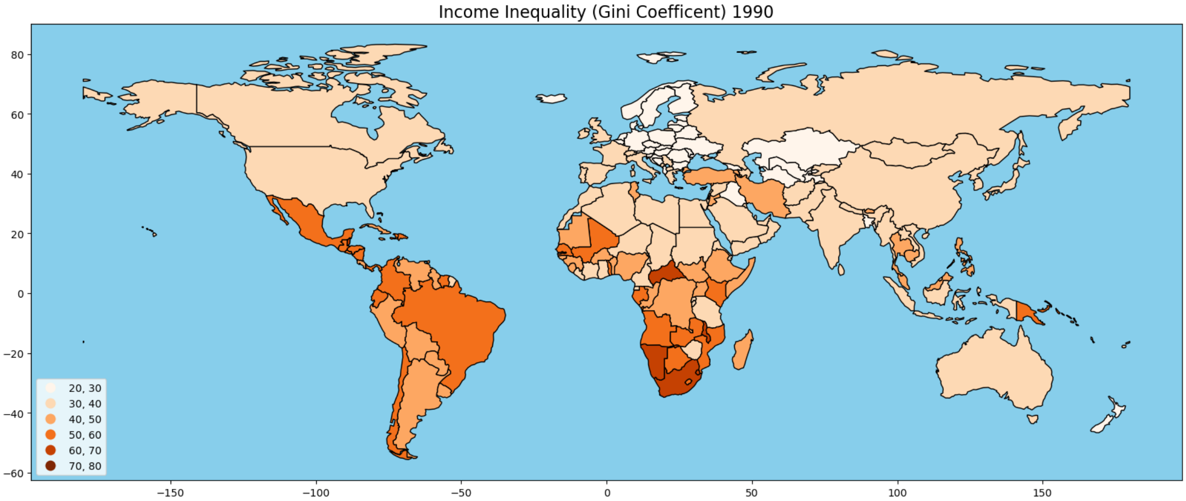
Income Inequality (Gini Coefficent) 1970



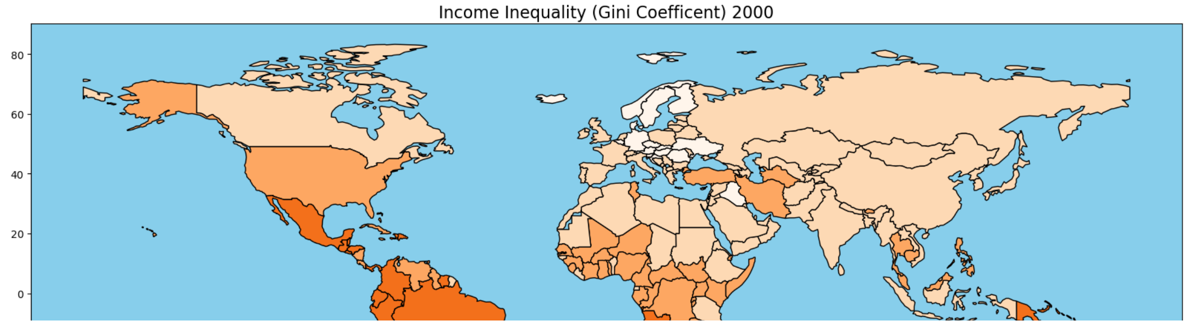
Income Inequality (Gini Coefficent) 1980

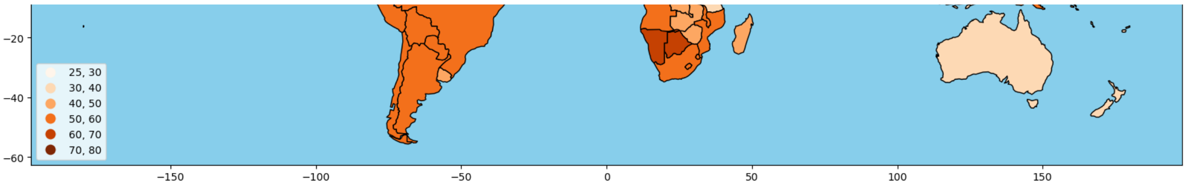


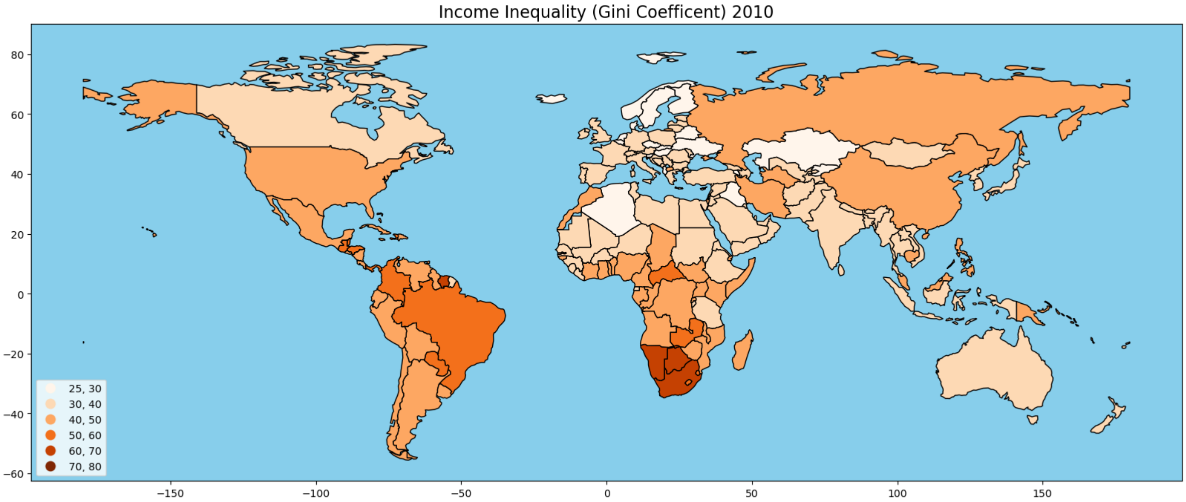
Income Inequality (Gini Coefficent) 1990



Income Inequality (Gini Coefficent) 2000

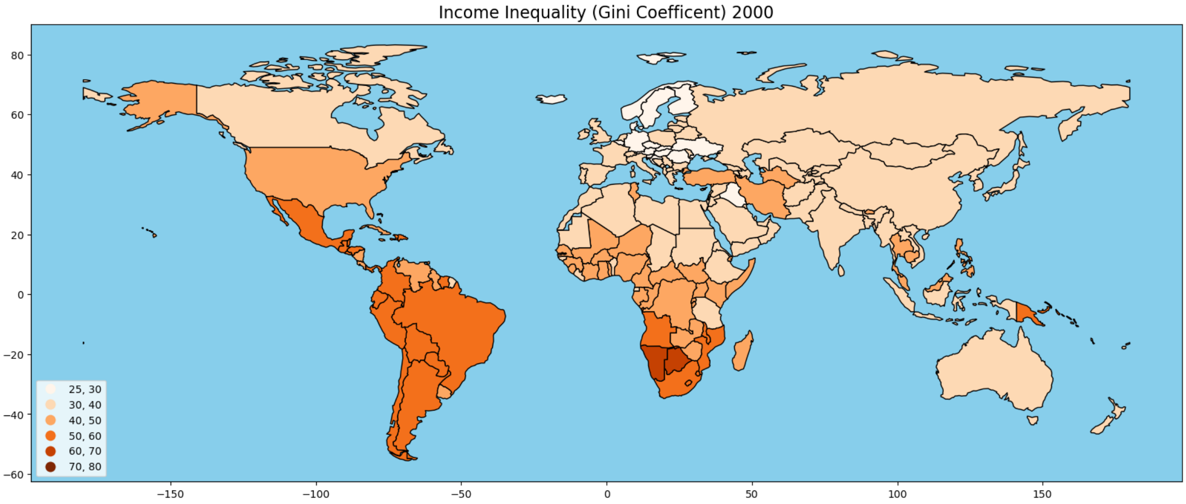


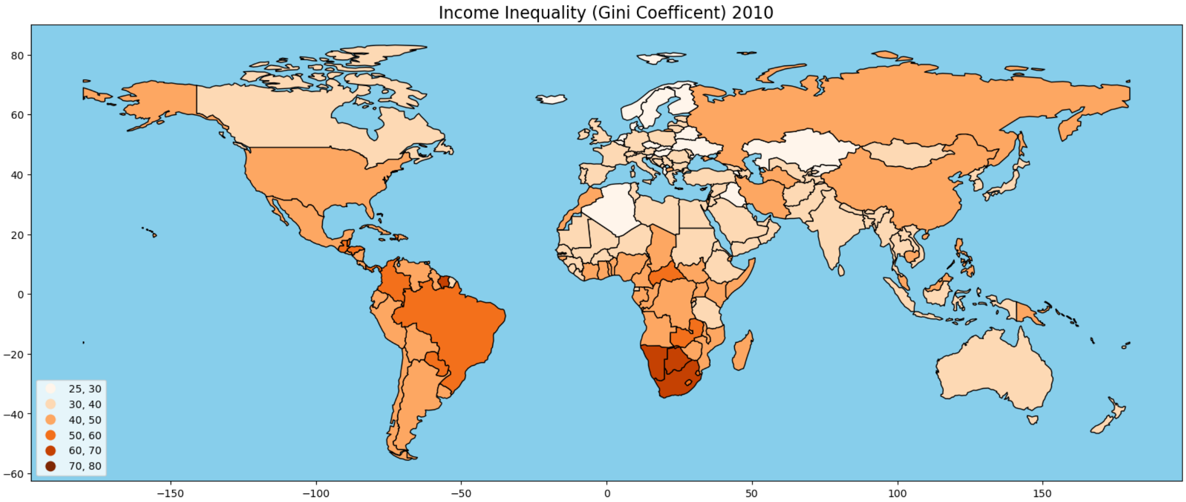




In [ ]:

```
years = [2000, 2010]
classifications = dict(bins=[30,40,50,60,70,80])

fig, axs = plt.subplots(len(years), 1, figsize=(20, 10*len(years)))

for ndx, yr in enumerate(years):
    ax1 = axs[ndx]
    ax1.set_facecolor('skyblue')
    ax1.set_title(f'Income Inequality (Gini Coefficent) {yr}', fontsize=16)
    gini_world_data.query(f'year == "{yr}"').plot(
        ax=ax1,
        column='gini',
        edgecolor='black',
        categorical=False,
        scheme='User_Defined',
        classification_kwds=classifications,
        #k=6,
        #scheme='fisher_jenks',
        cmap='Oranges',
        legend=True,
        legend_kwds={'loc': 'lower left', 'fmt': '{:,.0f}'}
        #,
        #missing_kwds={'color':'red','edgecolor': 'brwon', 'hatch': '///',
    )
plt.show()
```


Income Inequality (Gini Coefficent) 2000
80
60
40
20
0
−20
−40
−60
−150
−100
−50
0
50
100
150
25, 30
30, 40
40, 50
50, 60
60, 70
70, 80


Income Inequality (Gini Coefficent) 2010
80
60
40
20
0
−20
−40
−60
−150
−100
−50
0
50
100
150
25, 30
30, 40
40, 50
50, 60
60, 70
70, 80