Geospatial Data

WORKING WITH GEOSPATIAL DATA IN PYTHON

Instructors

Joris Van den Bossche & Dani Arribas-Bel

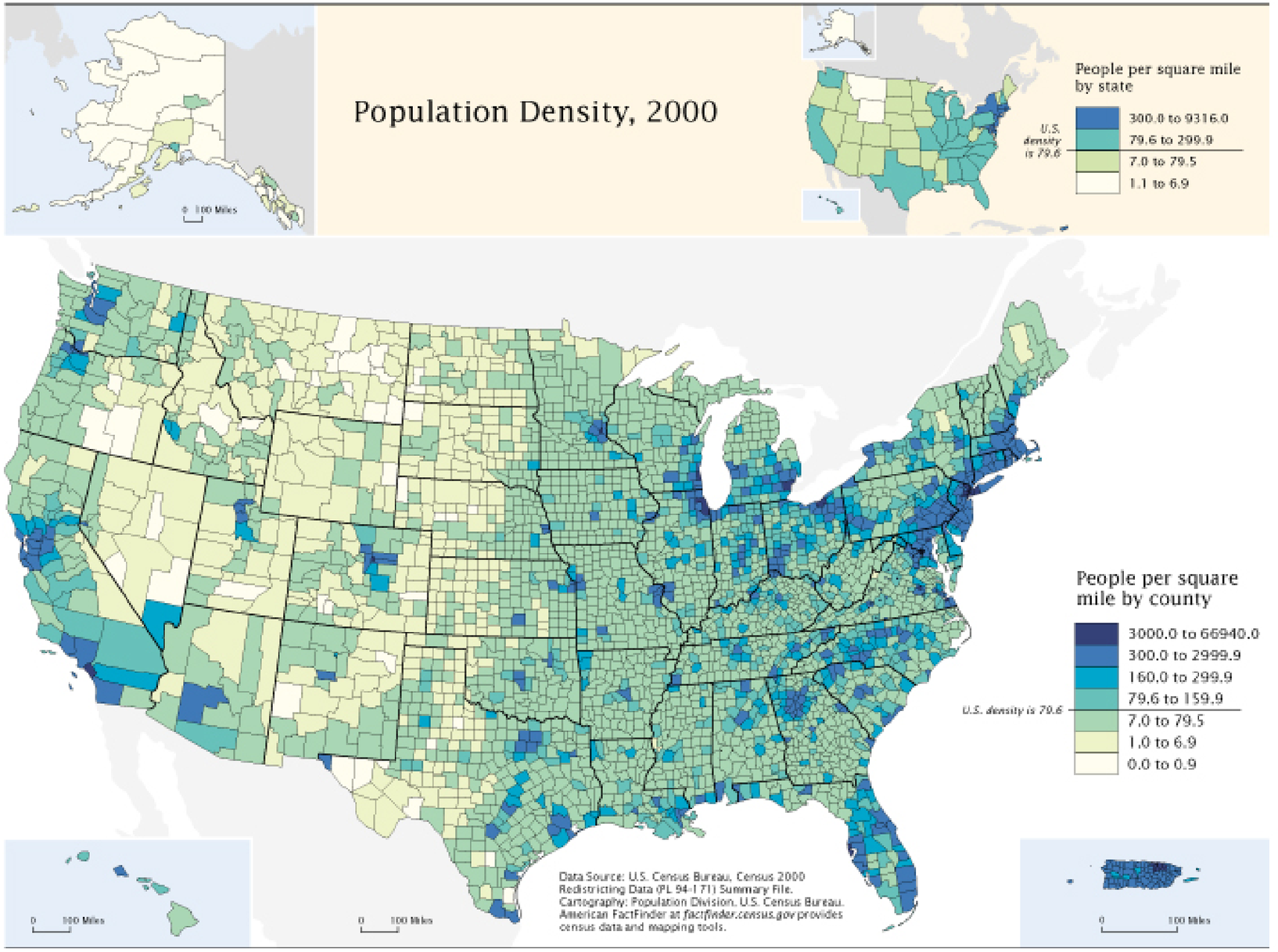
datacamp

What is Geospatial Data?

> Geospatial data are **data** with location information

What is Geospatial Data?

> Geospatial data are data with **location** information


Population Density, 2000
People per square mile by state
U.S. density is 79.6
300.0 to 9316.0
79.6 to 299.9
7.0 to 79.5
1.1 to 6.9
People per square mile by county
U.S. density is 79.6
3000.0 to 66940.0
300.0 to 2999.9
160.0 to 299.9
79.6 to 159.9
7.0 to 79.5
1.0 to 6.9
0.0 to 0.9
0 100 Miles
Data Source: U.S. Census Bureau, Census 2000 Redistricting Data (PL 94-171) Summary File.
Cartography: Population Division, U.S. Census Bureau.
American FactFinder at factfinder.census.gov provides census data and mapping tools.

Dani Arribas-Bel – Ride

12:54 PM on Sunday, October 2, 2016

Lunch Ride

Add a description

STRAVA LABS

View Flybys ›

40.3 km	1:44:53	223 m
Distance	Moving Time	Elevation

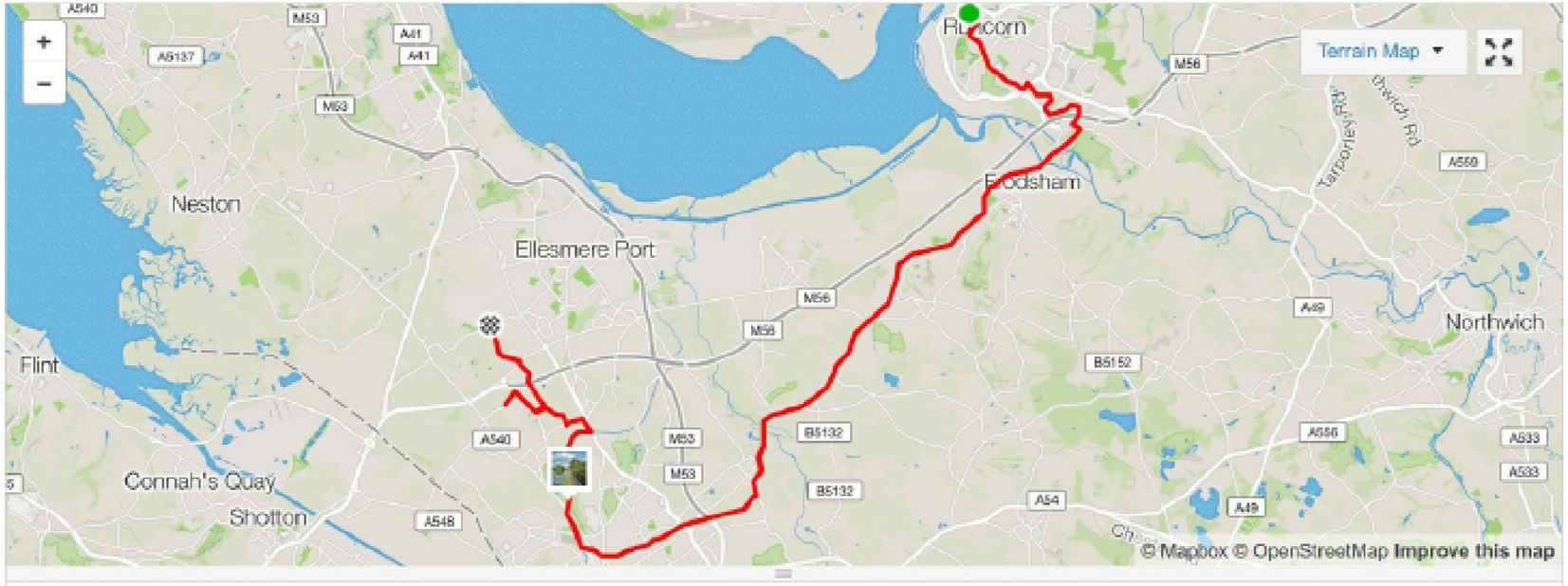
173 W	1,088 kJ
Estimated Avg Power	Energy Output

	Avg	Max
Speed	23.1 km/h	46.8 km/h
Elapsed Time	2:08:22	

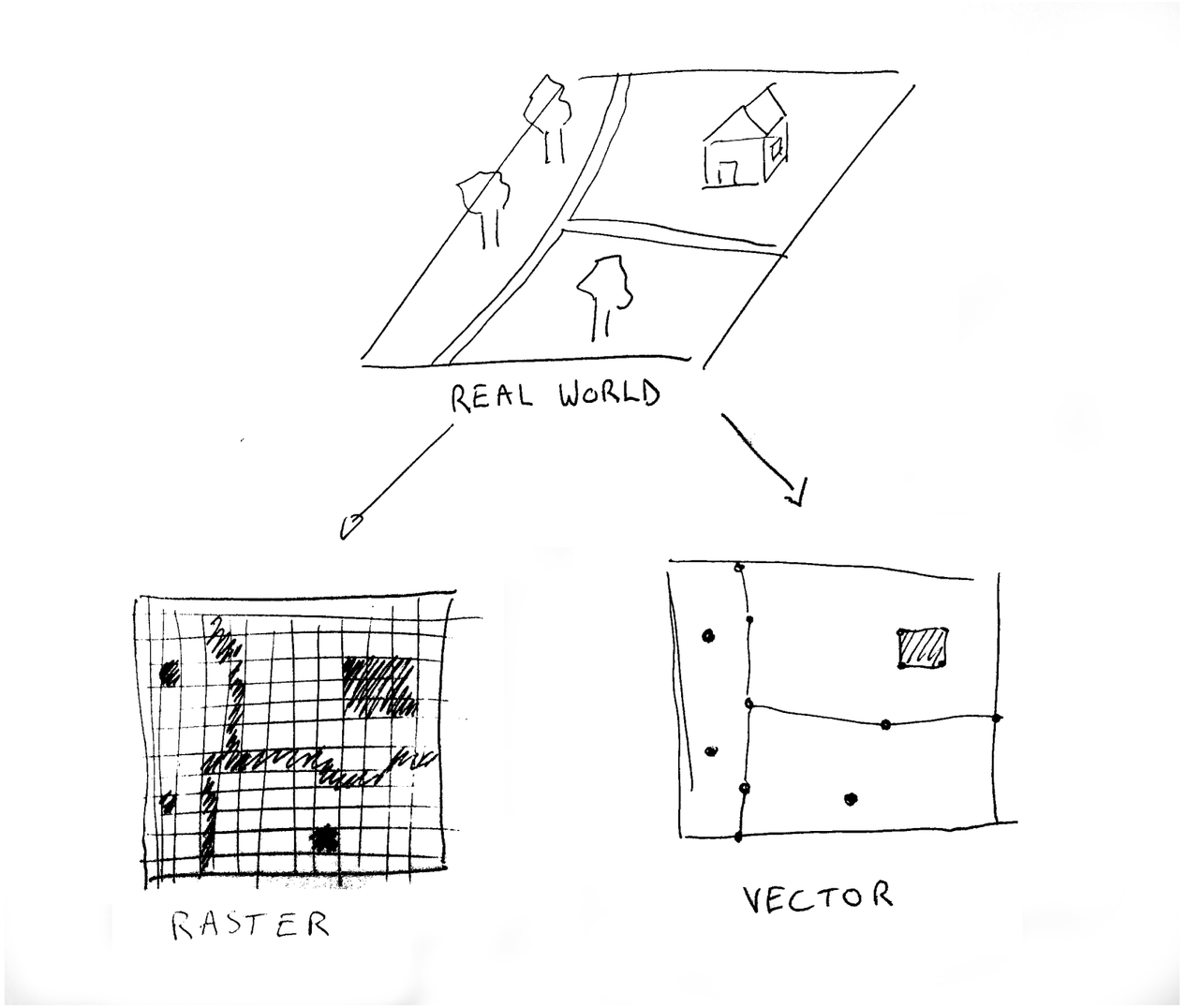
Show More

Device: Strava Android App

Bike: CrossBeast

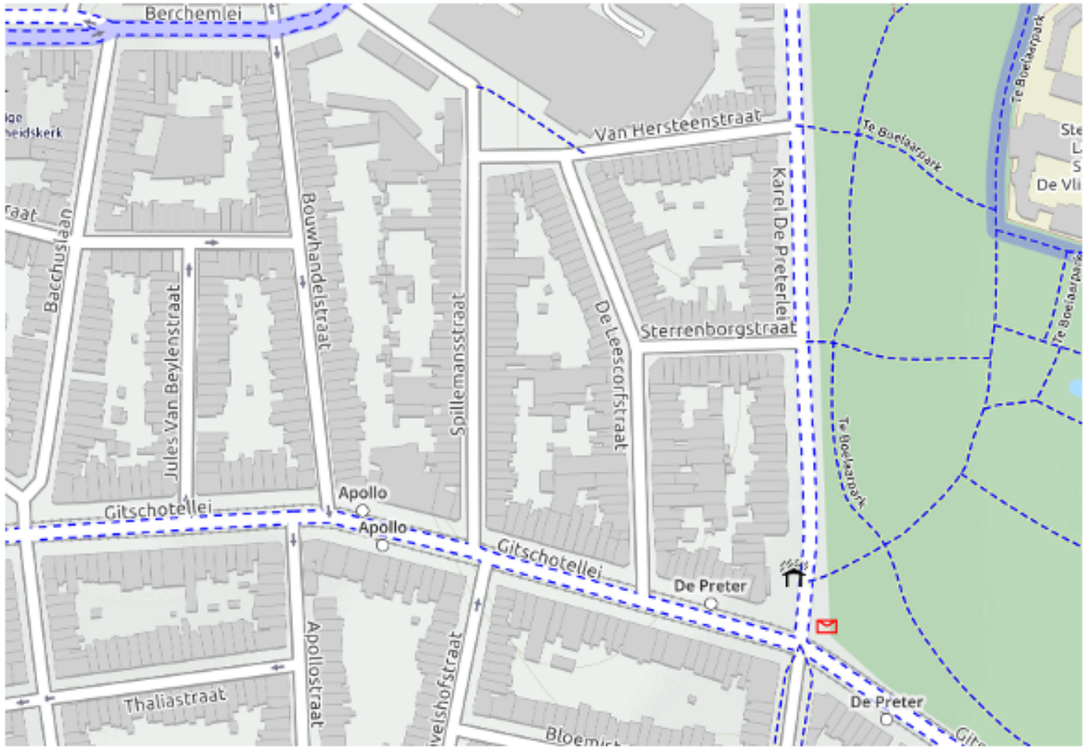


How we record the real world



Raster versus vector data

Raster



Vector

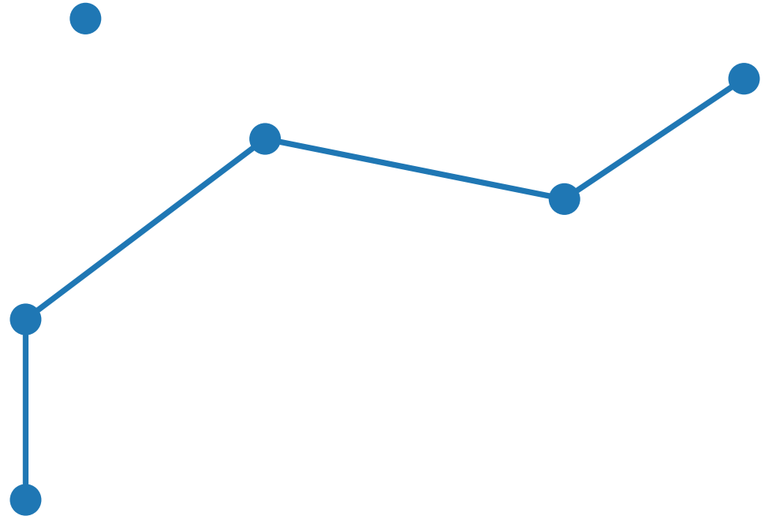
Vector features

"Discrete" representations that turn the world into:

Point(2, 10)

Vector features

"Discrete" representations that turn the world into:

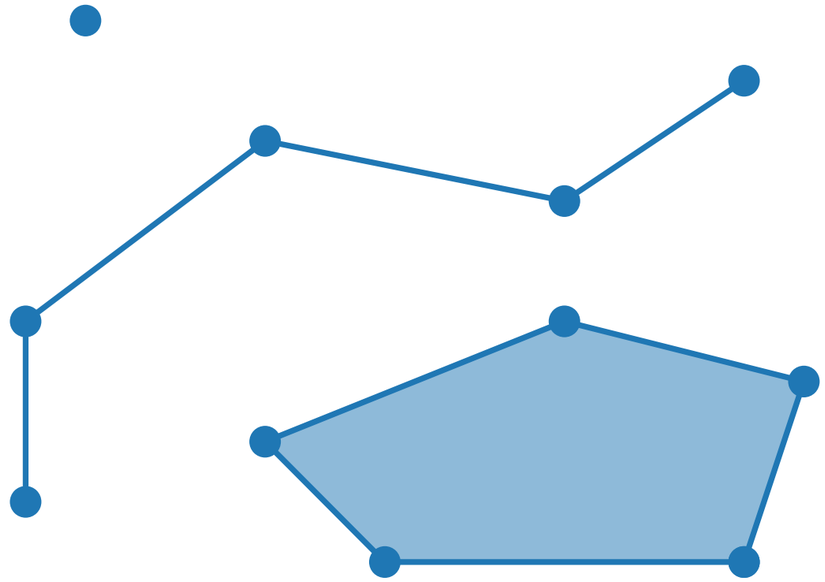


Point(2, 10)

LineString([(1, 2), (1, 5), ...])

Vector features

"Discrete" representations that turn the world into:

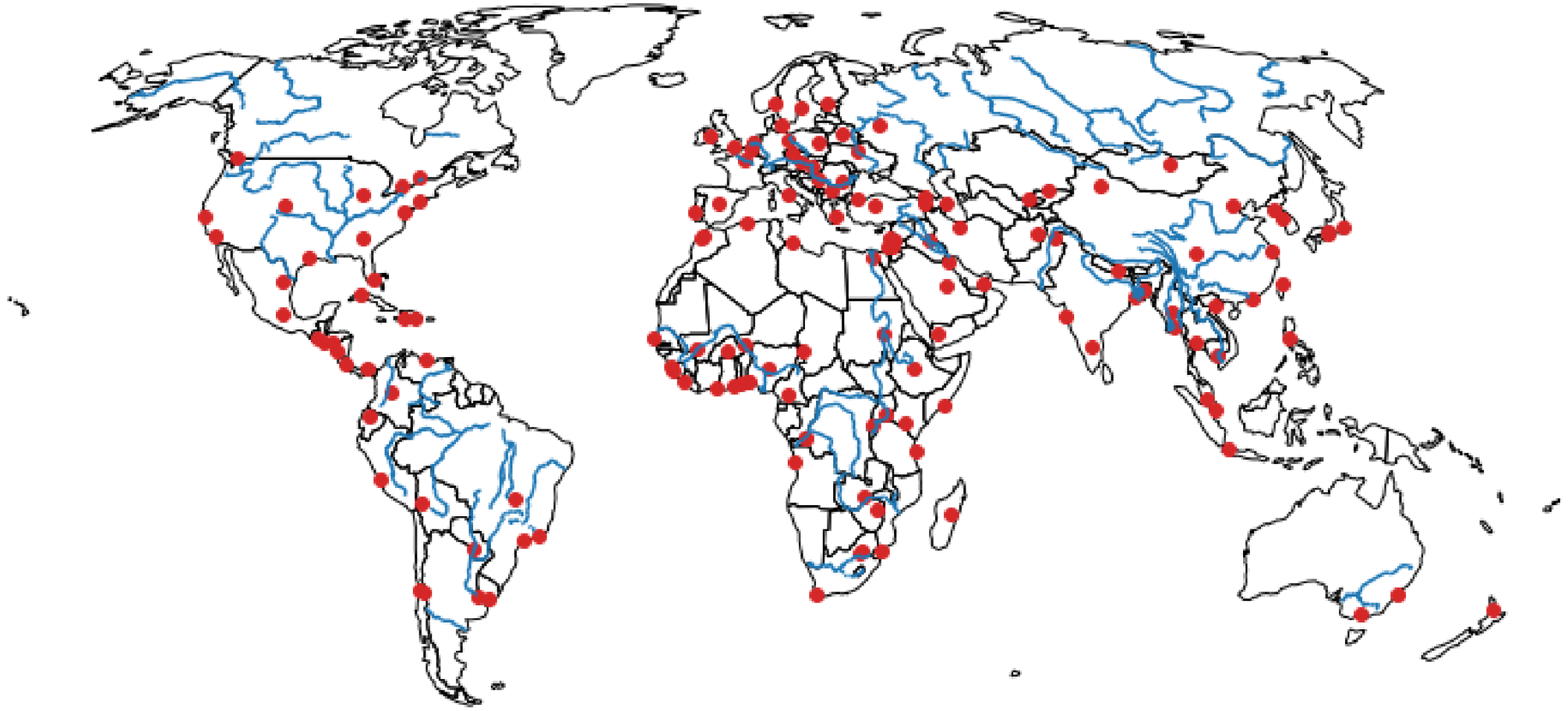


Point(2, 10)

LineString([(1, 2), (1, 5), ...])

Polygon([(13, 1), (14, 4), ...])

Feature consisting of multiple geometries: eg `MultiPolygon`



Vector attribute data

Vector features can have information associated that describe them: **attributes**

Tabular vector data:

	name	capital	population	geometry
0	Afghanistan	Kabul	34124811.0	POLYGON ((61.21081709172574 35.65007233330923,...
1	Angola	Luanda	29310273.0	(POLYGON ((23.90415368011818 -11.7222815894063...
2	Albania	Tirana	3047987.0	POLYGON ((21.0200403174764 40.84272695572588, ...
...	...	...	...	...
174	South Africa	Cape Town	54841552.0	POLYGON ((19.89576785653443 -24.76779021576059...
175	Zambia	Lusaka	15972000.0	POLYGON ((23.21504845550606 -17.52311614346598...
176	Zimbabwe	Harare	13805084.0	POLYGON ((29.43218834810904 -22.09131275806759...

Let's practice!

WORKING WITH GEOSPATIAL DATA IN PYTHON

Introduction to GeoPandas

WORKING WITH GEOSPATIAL DATA IN PYTHON

Joris Van den Bossche

Open source software developer and teacher, GeoPandas maintainer

datacamp

Spatial specific data formats

```
restaurants = pd.read_csv("datasets/paris_restaurants.csv")
restaurants.head()
```

	type	x	y
0	Restaurant européen	259641.6	6251867.4
1	Restaurant traditionnel français	259572.3	6252030.2
2	Restaurant traditionnel français	259657.2	6252143.8
3	Restaurant indien, pakistanais et Moyen Orient	259684.4	6252203.6
4	Restaurant traditionnel français	259597.9	6252230.0

In the rest of the course:

- spatial file formats (Shapefiles, GeoJSON, GeoPackage, ...)
- GeoPandas: pandas dataframes with support for spatial data

Importing geospatial data with GeoPandas

```
import geopandas
```

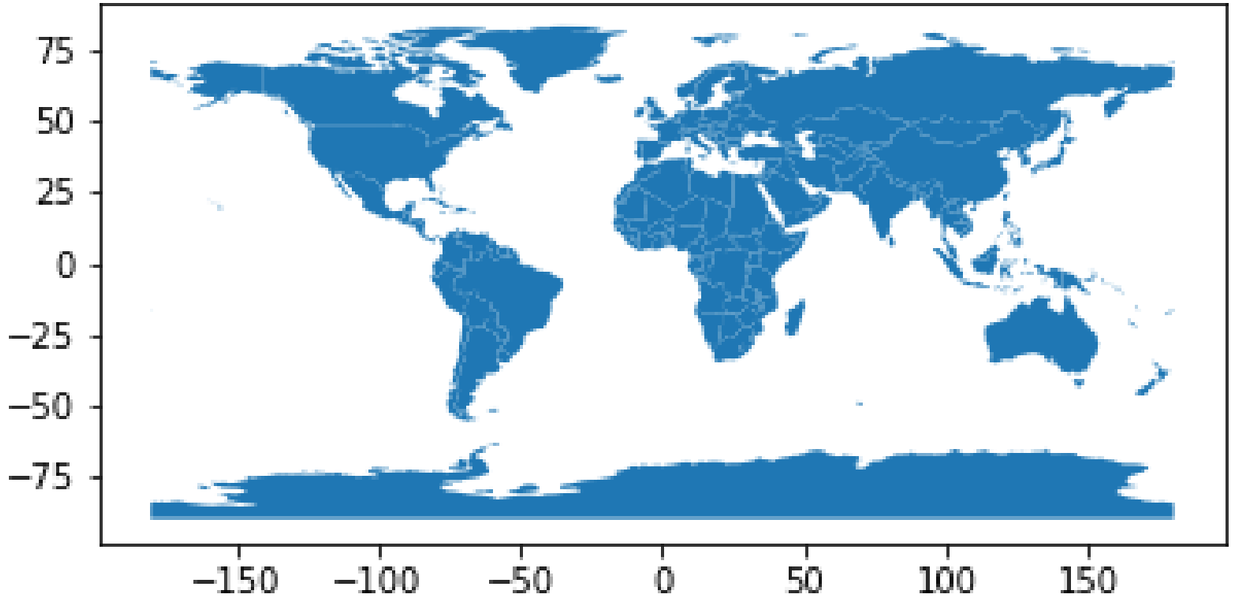
```
countries = geopandas.read_file("countries.geojson")
```

```
countries.head()
```

```
          name      continent       gdp                                geometry
0  Afghanistan           Asia   64080.0  POLYGON ((61.21 35.65, 62.23 35...
1       Angola         Africa  189000.0  MULTIPOLYGON (((23.90 -11.72, 2...
2      Albania         Europe   33900.0  POLYGON ((21.02 40.84, 21.00 40...
4    Argentina  South America  879400.0  MULTIPOLYGON (((-66.96 -54.90, ...
5      Armenia           Asia   26300.0  POLYGON ((43.58 41.09, 44.97 41...
```

Quickly visualizing spatial data with GeoPandas

```
countries.plot()
```



The GeoDataFrame

```
countries.head()
```

```
          name       continent        gdp                                geometry
0  Afghanistan            Asia    64080.0  POLYGON ((61.21 35.65, 62.23 35...
1       Angola          Africa   189000.0  MULTIPOLYGON (((23.90 -11.72, 2...
2      Albania          Europe    33900.0  POLYGON ((21.02 40.84, 21.00 40...
...
```

```
type(countries)
```

```
geopandas.geodataframe.GeoDataFrame
```

The GeoDataFrame

```
countries.head()
```

```
          name       continent         gdp                                  geometry
0  Afghanistan            Asia     64080.0  POLYGON ((61.21 35.65, 62.23 35...
1       Angola          Africa    189000.0  MULTIPOLYGON (((23.90 -11.72, 2...
2      Albania          Europe     33900.0  POLYGON ((21.02 40.84, 21.00 40...
...
```

A GeoDataFrame represents a tabular, geospatial vector dataset:

- a **'geometry' column**: that holds the geometry information
- other columns: **attributes** describe each of the geometries

The 'geometry' attribute

```
countries.geometry
```

```
0      POLYGON ((61.21 35.65, 62.23 35...
1      MULTIPOLYGON (((23.90 -11.72, 2...
                    ...
175    POLYGON ((23.22 -17.52, 22.56 -...
176    POLYGON ((29.43 -22.09, 28.79 -...
Name: geometry, Length: 176, dtype: object
```

```
type(countries.geometry)
```

```
geopandas.geoseries.GeoSeries
```

Spatial aware DataFrame

```
countries.geometry.area
```

```
0        63.593500
1       103.599439
2         3.185163
            ...
174     112.718524
175      62.789498
176      32.280371
Length: 177, dtype: float64
```

Summary

A `GeoDataFrame` is like a pandas DataFrame:

- all features of normal pandas DataFrames still work

but supercharged with spatial functionality:

- `plot()` method
- `geometry` attribute (GeoSeries)
- spatial-specific attributes and methods (e.g. `area`)

Let's practice!

WORKING WITH GEOSPATIAL DATA IN PYTHON

Exploring and visualizing spatial data and its attributes

WORKING WITH GEOSPATIAL DATA IN PYTHON

Joris Van den Bossche
Open source software developer and teacher, GeoPandas maintainer

datacamp

Filtering data

```
countries.head()
```

```
          name      continent       gdp                              geometry
0  Afghanistan           Asia   64080.0  POLYGON ((61.21 35.65, 62.23 35...
1       Angola         Africa  189000.0  MULTIPOLYGON (((23.90 -11.72, 2...
...
```

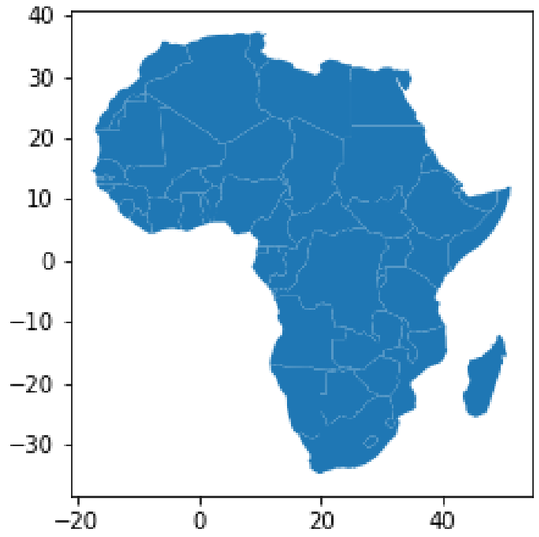
```
countries['continent'] == 'Africa'
```

```
0      False
1       True
         ...
175     True
176     True
Name: continent, Length: 177, dtype: bool
```

Filtering data

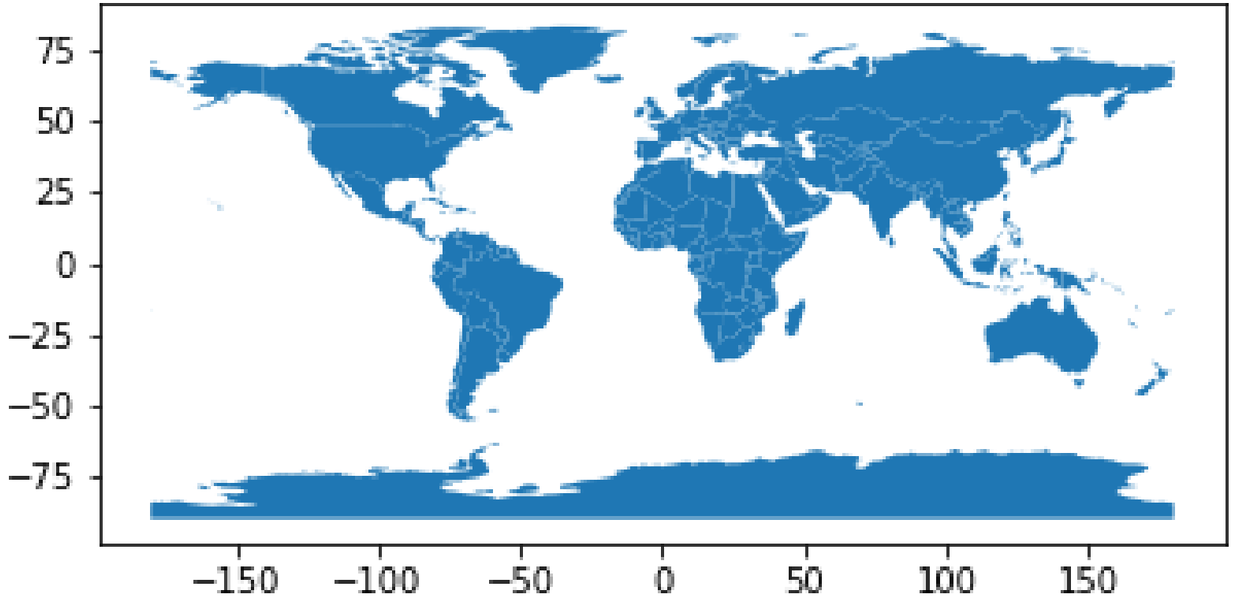
```
countries_africa = countries[countries['continent'] == 'Africa']
```

```
countries_africa.plot()
```



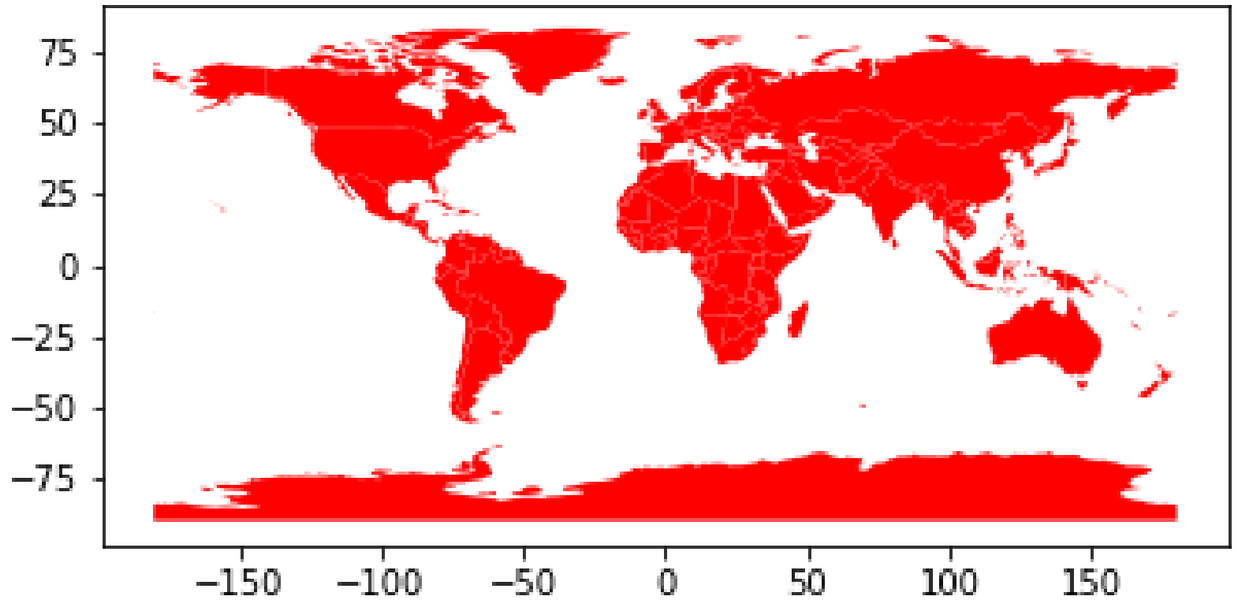
Visualizing spatial data

```
countries.plot()
```



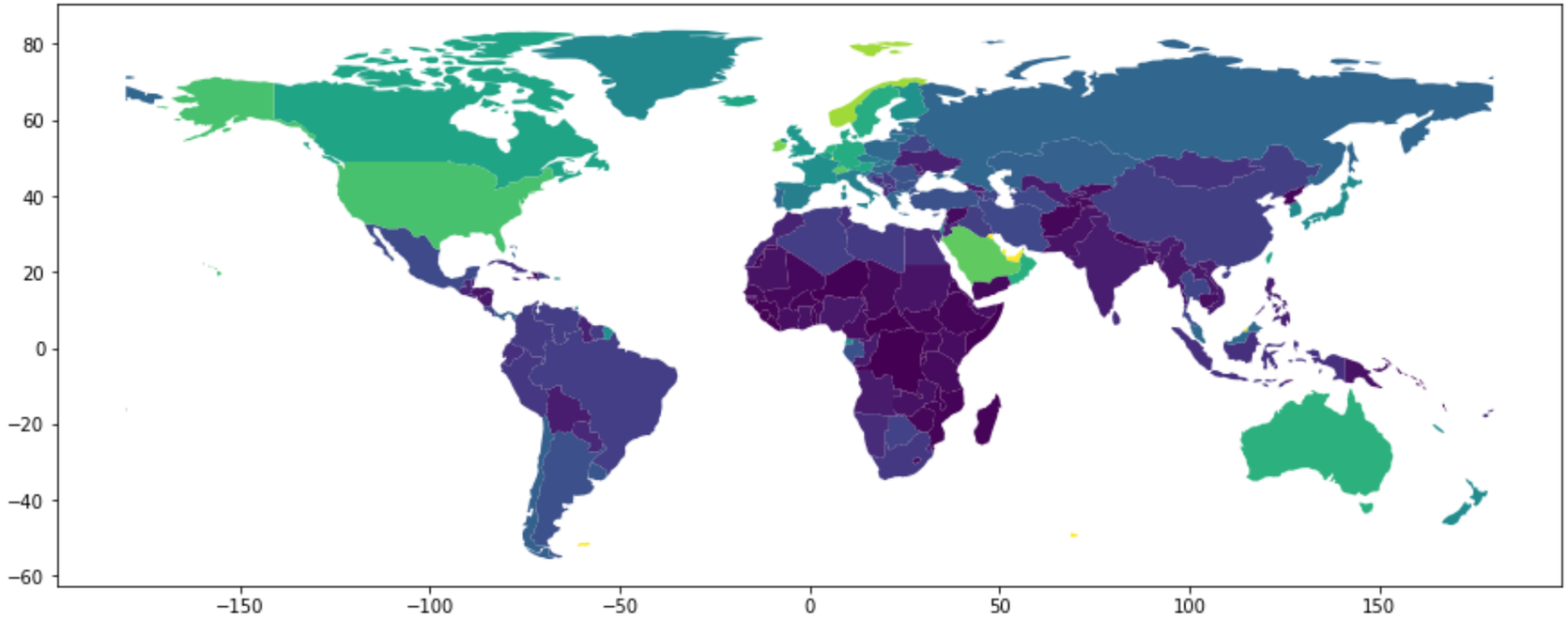
Adjusting the color: uniform color

```
countries.plot(color="red")
```



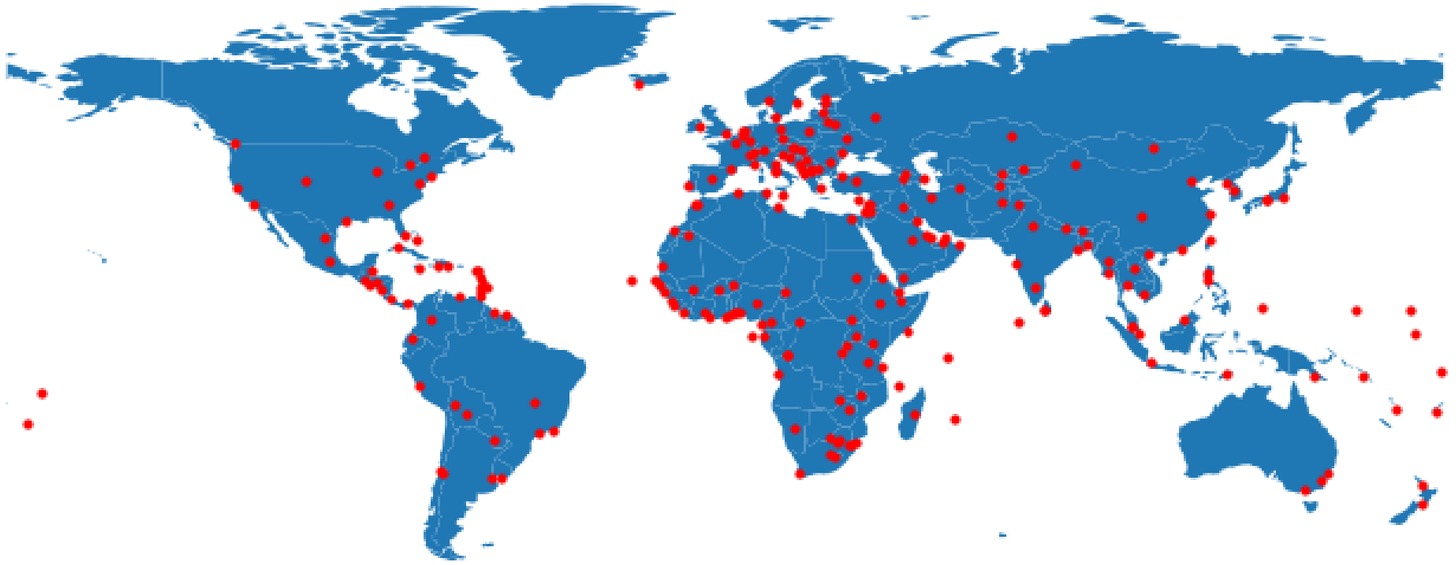
Adjusting the color: based on attribute values

```
countries.plot(column='gdp_per_cap')
```



Multi-layered plot

```
fig, ax = plt.subplots(figsize=(12, 6))
countries.plot(ax=ax)
cities.plot(ax=ax, color='red', markersize=10)
ax.set_axis_off()
```



Let's practice!

WORKING WITH GEOSPATIAL DATA IN PYTHON