Spatial II

Peter Ganong and Maggie Shi

October 30, 2024

Table of contents I

Introduction to data structures in geopandas (6.2)

Geopandas roadmap

In practice, we won't be coding our geodata by hand... Instead we are going to use shapefiles!

```
import geopandas as gpd
```

Roadmap

- Vocabulary
- File formats
- Read in data
- Preview data

Define vocabulary

Vocabulary

- A `GeoDataFrame` is basically like a `pandas.DataFrame` that contains dedicated columns for storing geometries.
  - We will start with examples with a single column and later teach you how to use more than one column
- That column is called a `GeoSeries`. This can be any of data types (point, line, polygon) from the prior section. All of the methods you saw in the last section can also be used on a `GeoSeries`

File format I: Shapefile

- consists of at least three files `.shp` has feature geometrics, `.shx` has a positional index, `.dbf` has attribute information
- Usually also have `.prj` which describes the Coordinate Reference System (CRS)
- When you read in `map.shp` it automatically reads the rest of them as well to give you proper GeoDataFrame composed of geometry, attributes and projection.

Coordinate Reference Systems

- Coordinate Reference System (CRS) is a combination of:
  - "Datum": origin of latitude and longitude
  - "Project": representation of curved surface onto flat map
- Most common CRS: WGS84 (used for GPS)
- All coordinates are consistent *within* a CRS, but not always *across* CRS's
- Different CRS's suit different needs
  - optimized for local vs. global accuracy
  - different approaches to approx. shape of the earth
  - distance is measured in different units: degrees, miles, meters
- Each system is associated with a unique *EPSG code*. Searchable on https://epsg.io
  - (Aside: EPSG stands for European Petroleum Survey Group)
  - These codes are used to convert one CRS into another

Reading a Shapefile `.shp`

```
#in same dir:  `.shx` and `.dbf`
filepath = "/Users/nasser.alshaya/Desktop/Fall-2024/PPHA-3(
data = gpd.read_file(filepath)
```

File format II: GeoPackage

- single file `.gpkg`
- Supports both raster and vector data
- Efficiently decodable by software, particularly in mobile devices

GeoPackage is more modern, but you will encounter shapefiles everywhere you look so good to be familiar with it.

Reading a GeoPackage gpkg

```
filepath = "data/austin_pop_2019.gpkg"
data = gpd.read_file(filepath)
type(data)
```

```
geopandas.geodataframe.GeoDataFrame
```

Previewing a GeoDataFrame

```
data.head()
```

	pop2019	tract	geometry
0	6070.0	002422	POLYGON ((615643.487 3338728.496, 615645.4
1	2203.0	001751	POLYGON ((618576.586 3359381.053, 618614.3
2	7419.0	002411	POLYGON ((619200.163 3341784.654, 619270.8
3	4229.0	000401	POLYGON ((621623.757 3350508.165, 621656.2
4	4589.0	002313	POLYGON ((621630.247 3345130.744, 621717.9