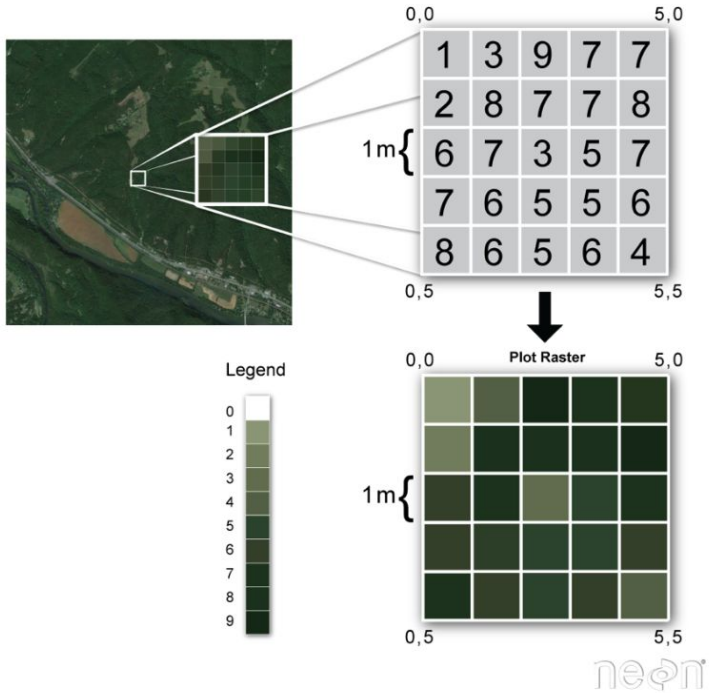
A Python Intro to Rasters

Carmen Galaz García
National Center for Ecological Analysis and Synthesis (NCEAS)

Workshop at Central Coast Python Group

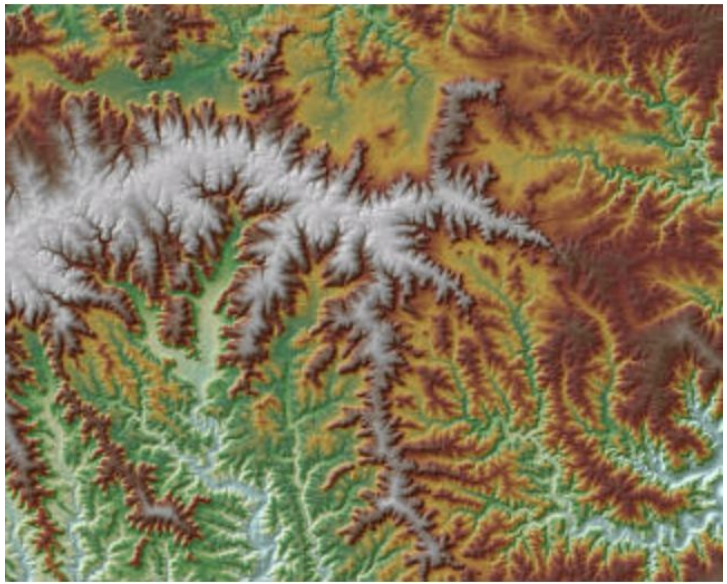
Raster Data

Raster data is any pixelated (or gridded) data where each pixel is associated with a specific geographic location.

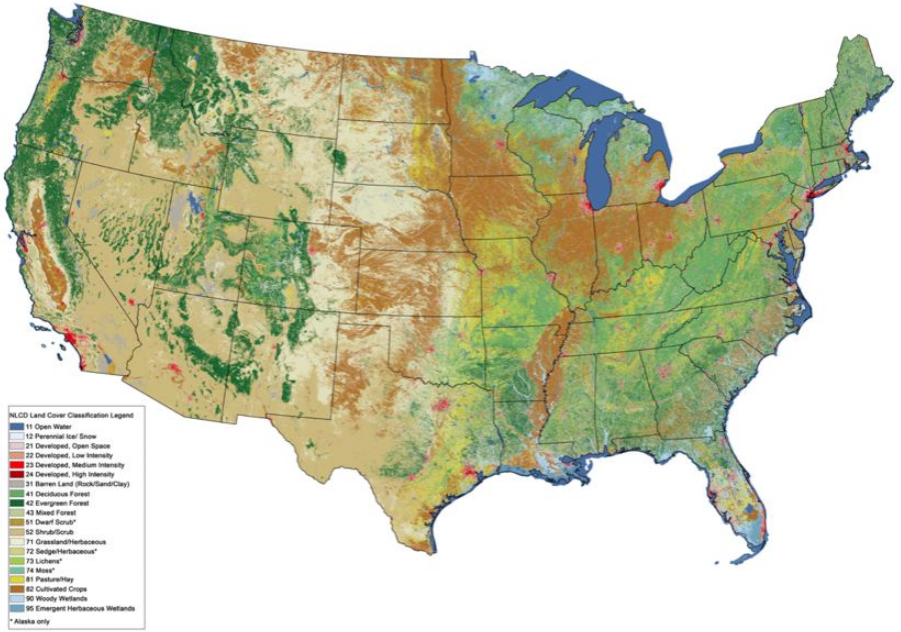


Source: National Ecological Observatory Network (NEON)

Raster Data



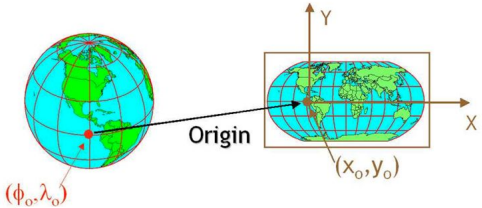
(Source: Elevation Data for Minnesota https://www.mngeo.state.mn.us/chouse/elevation/)



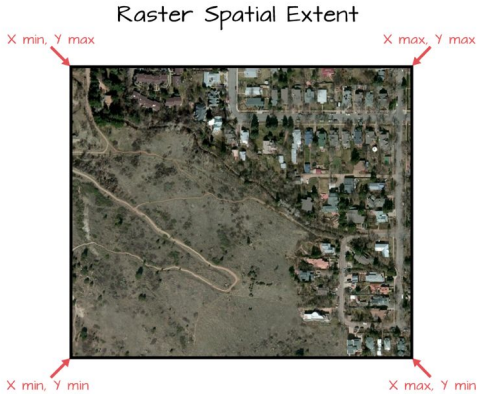
(Source: Homer, C.G., et al., 2015, Completion of the 2011 National Land Cover Database for the conterminous United States-Representing a decade of land cover change information. Photogrammetric Engineering and Remote Sensing, v. 81, no. 5, p. 345-354)

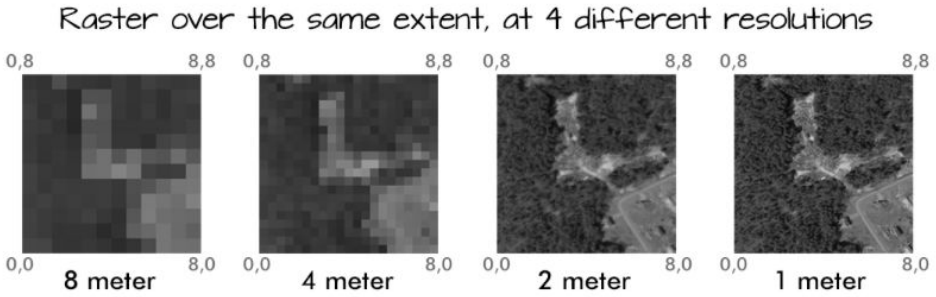
Raster File Format

Raster data can come in many different formats. For this workshop, we will use the GeoTIFF format which has the extension `.tif`. A GeoTIFF is a standard `.tif` image format with additional spatial (georeferencing) information embedded in the file. This includes the following raster metadata:



Source: http://ayresriverblog.com

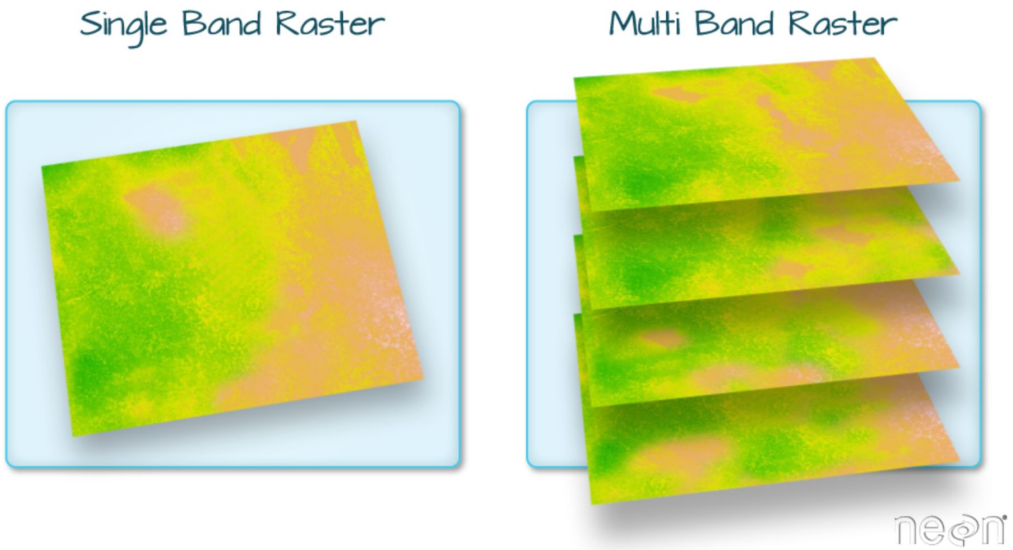




(Source: National Ecological Observatory Network (NEON))

Multi-band Raster Data

A raster can contain one or more bands. One type of multi-band raster dataset that is familiar to many of us is a color image.



Vector Data

Vector data structures represent specific features on the Earth's surface, and assign attributes to those features. Vectors are composed of discrete geometric locations (x, y values) known as vertices that define the shape of the spatial object. The organization of the vertices determines the type of vector that we are working with: point, line or polygon.

- Each geometry feature can carry multiple attributes instead of just one, e.g. a database of cities can have attributes for name, country, population, etc

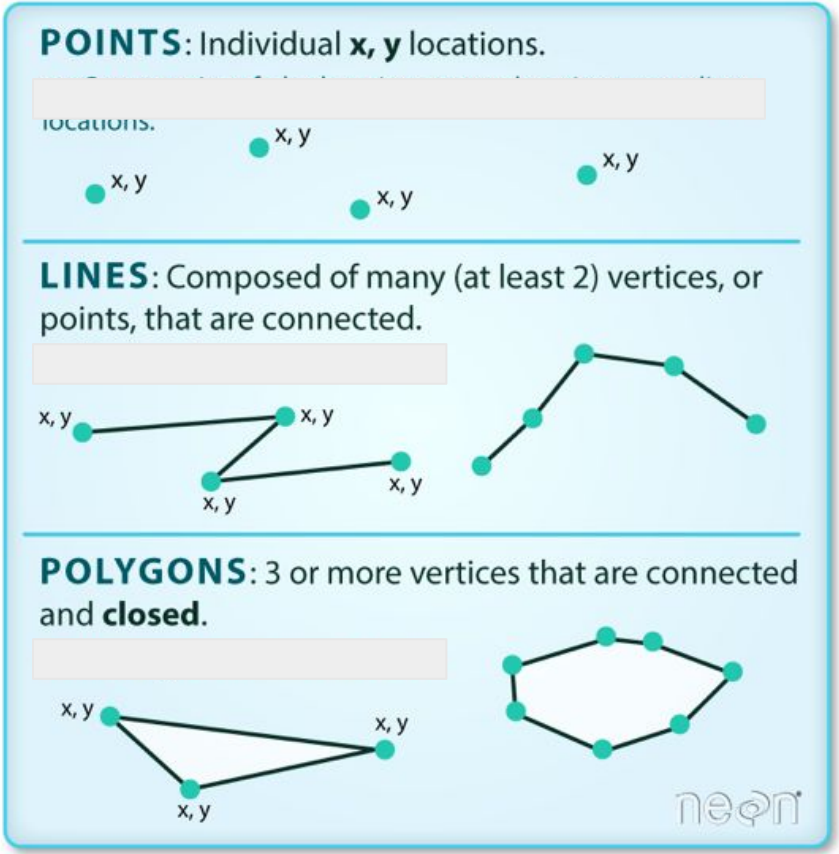


Image Source: National Ecological Observatory Network (NEON)

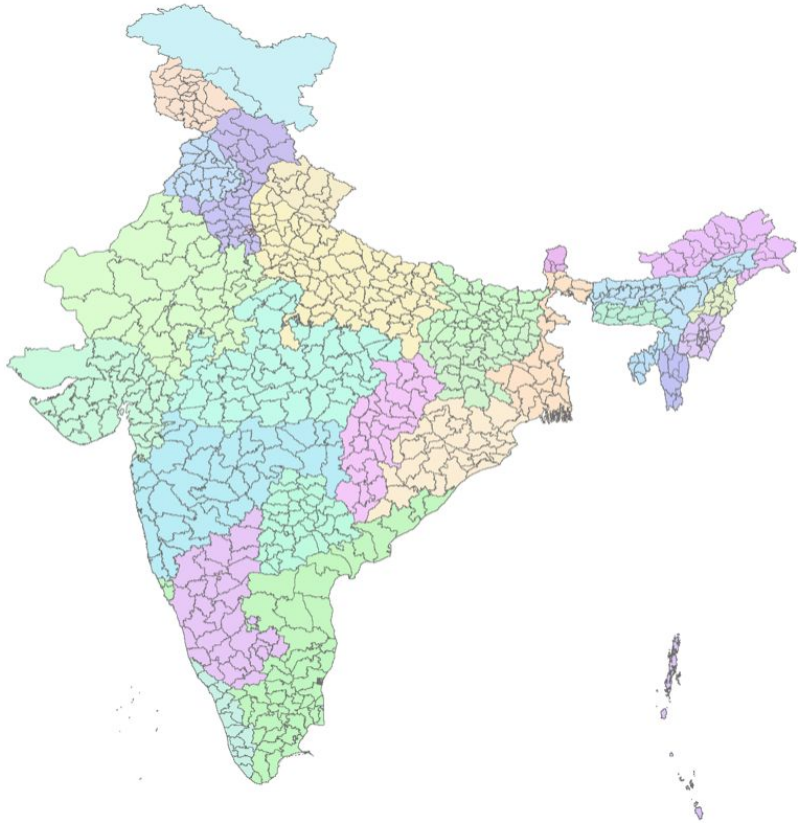
Vector Data Format

A Shapefile format consists of multiple files in the same directory:

- The `.shp` file stores the feature geometry itself
- `.shx` is a positional index of the feature geometry to allow quickly searching forwards and backwards the geographic coordinates of each vertex in the vector
- `.dbf` contains the tabular attributes for each shape.
- `.prj` file indicates the Coordinate reference system (CRS)
- `.shp.xml` contains the Shapefile metadata.

Together, the Shapefile includes the following information:

- **Extent** - the spatial extent of the shapefile. The spatial extent for a shapefile represents the combined extent for all spatial objects in the shapefile.
- **Object type** - whether the shapefile includes points, lines, or polygons.
- **Coordinate reference system (CRS)**
- **Other attributes** - for example, a line shapefile that contains the locations of streams, might contain the name of each stream.



India Administrative Boundaries Shapefile 2019
Source: https://revolutionarygis.wordpress.com

Coordinate Reference Systems (CRS)

A data structure cannot be considered geospatial unless it is accompanied by coordinate reference system (CRS) information.

The CRS associated with a dataset tells your mapping software (for example Python) where the raster is located in geographic space.

For the purposes of this workshop, what is important to understand is that **data from the same location but saved in different projections will not line up in any GIS or other program.** Thus, it's important when working with spatial data to identify the coordinate reference system applied to the data and retain it throughout data processing and analysis.

The above image shows maps of the United States in different projections. Notice the differences in shape associated with each projection. These differences are a direct result of the calculations used to flatten the data onto a 2-dimensional map. (Source: opennews.org)

Let’s look at some code!

References

Some of the information on the slides is based on the workshop *Introduction to Geospatial Raster and Vector Data with Python* from the The Carpentries Incubator. This is a great resource to go deeper into working with geospatial data with Python.