Session 6: Introduction to geospatial data

Imperial College London

Learning objectives

After this lecture you should be able to

- Describe (statistically) the different geospatial data types
- Describe how geospatial data are represented in R
- Describe Coordinate Reference Systems (Geographical and Projected CRS)

The topics covered in this lecture are presented in:

- Chapter 2 of the book **Geocomputation with R** https://r.geocompx.org/index.html

Outline

1. The different types of spatial data

2. Spatial data in R

3. Coordinate Reference Systems

What are the different types of spatial data?

Terminology

- The data are seen as being a realization of a stochastic process, that is, of a set of random numbers each of which is associated with a spatial location.

  – A spatial process in d dimensions is denoted as:

$$\{Z(\boldsymbol{s}) : \boldsymbol{s} \in \mathcal{D} \subset \mathbb{R}^d\}$$

where

- Z is the attribute we observe (e.g. temperature, number of sudden infant deaths etc.)
- $\boldsymbol{s}$ is the location where Z is observed (e.g. coordinates such as latitude and longitude)
- $\mathcal{D}$ is the domain, and it is called the **index set** = possible locations

- Symbols are as follows: $\{\}$ a set (a collection of elements); $\in$ element of or belongs to; $\subset$ subset of; $\mathbb{R}$ real numbers set

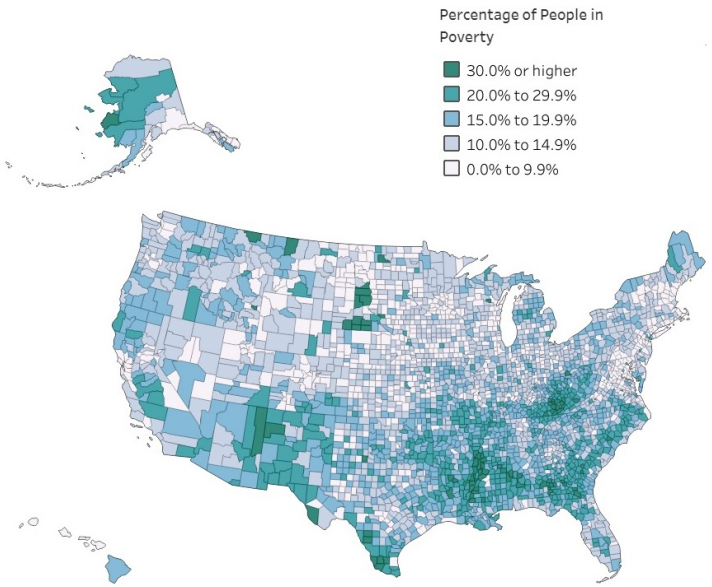
Type of spatial data

Cressie (1993) distinguishes three types of spatial data, based on the nature of the spatial domain $\mathcal{D}$:

- Areal data (also known as lattice data): $\mathcal{D}$ is fixed (of regular or irregular shape) and partitioned into a finite number of areal units (e.g. census tract, pixels) with well-defined boundaries.
- Geostatistical (or point-referenced) data: $\mathcal{D}$ is a continuous fixed set. By continuous we mean that $Z(\boldsymbol{s})$ can be observed everywhere within $\mathcal{D}$. By fixed we mean that the points in $\mathcal{D}$ are non-stochastic.
- Point pattern data: $\mathcal{D}$ is itself random. Its index set gives the locations of random **events** that are the spatial point pattern.

Example of areal data (irregular lattice) [1]

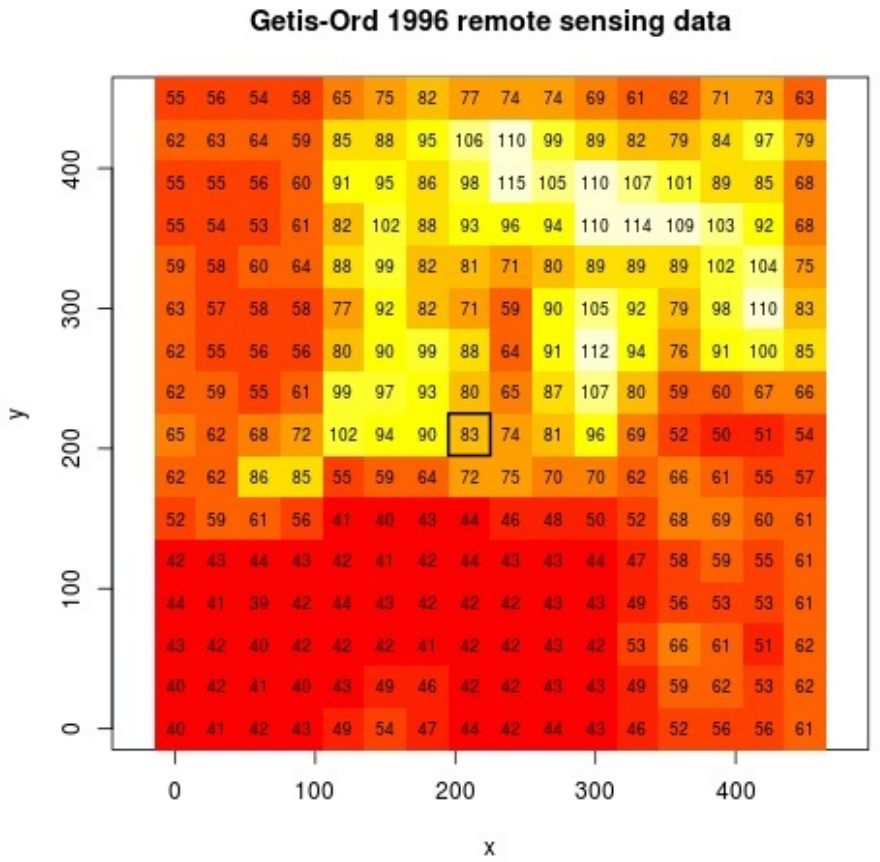
Lattice data can be either irregularly aligned or gridded, and occur in the form of aggregated observation over areas.



Percentage of people in poverty by County: 2015-2019. Source: American Community Survey 2015-2019, 5-Year Data Release.

- Lattice data involves data measured at different areas, e.g., neighbourhoods, cities, provinces, states, etc.
- The question of primary interest is related to spatial variability of the phenomenon under study.
- Spatial dependence appears because neighbour areas will show similar values of the variable of interest.
- The *sites* $\boldsymbol{s} \in \mathcal{D}$ in this case are actually the areas themselves.

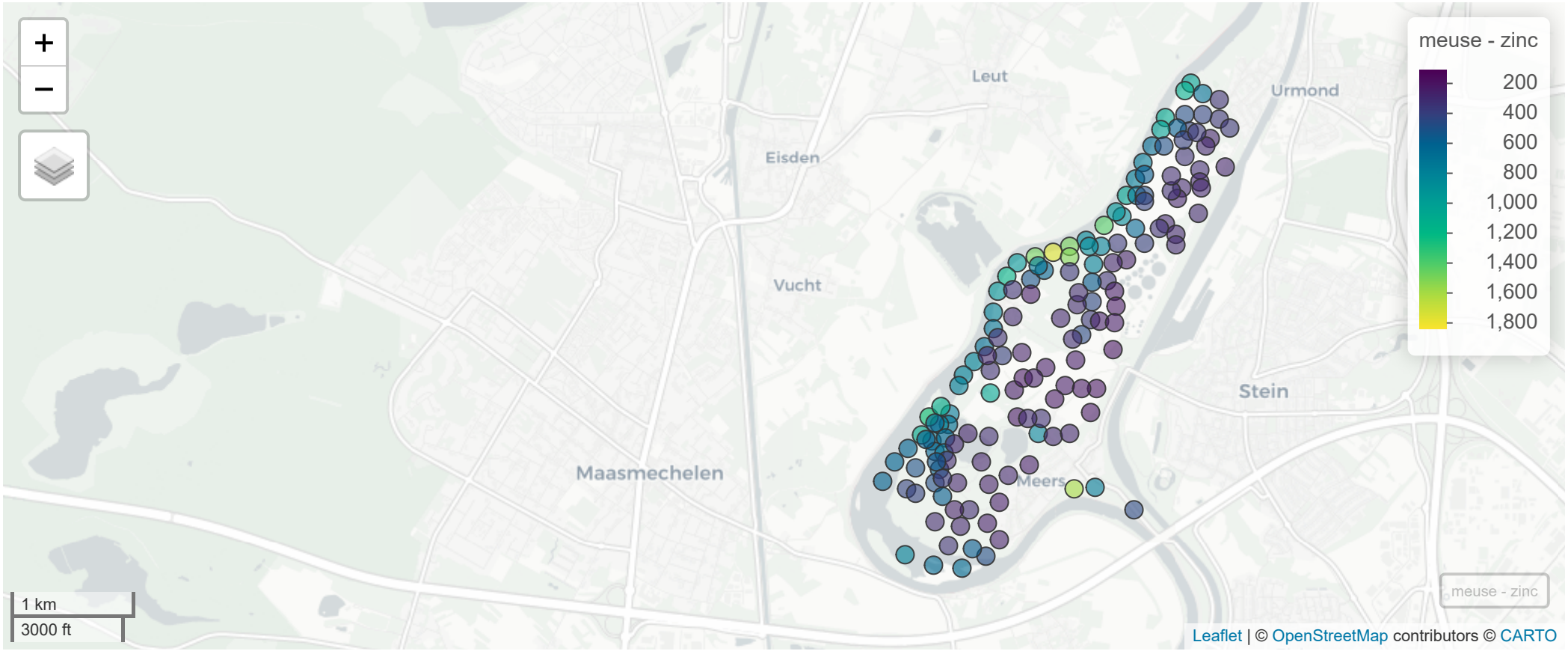
Example of regular lattice data [2]



Regular lattice: Getis-Ord remote sensing example data (from package spdep).

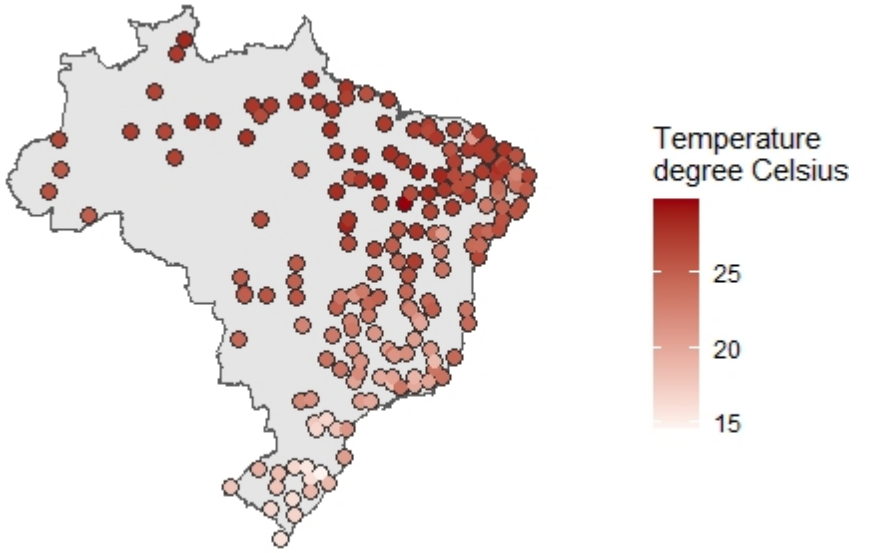
Example of geostatistical data [1]

Zinc measured in the top soil in a flood plain along the river Meuse (obtained from sp R package)



Example of geostatistical data [2]

Average air temperature in degree Celsius (March-November 2018)



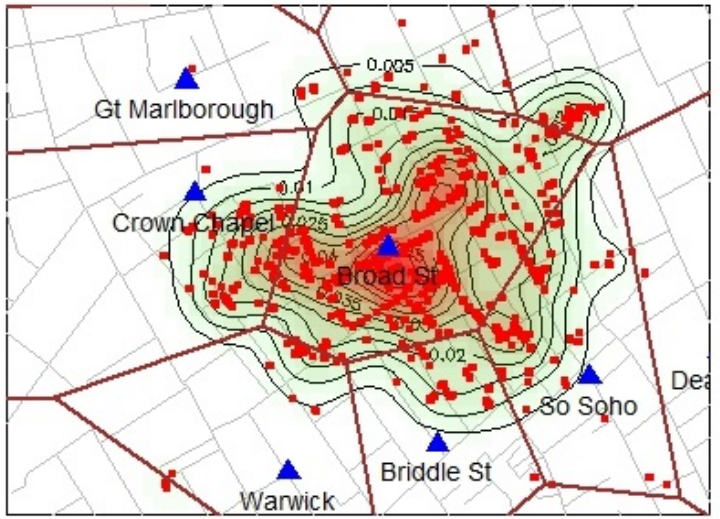
Observed

Predicted

- The question of primary interest concerns the reconstruct a surface from noisy observations taken at a finite number of spatial locations.

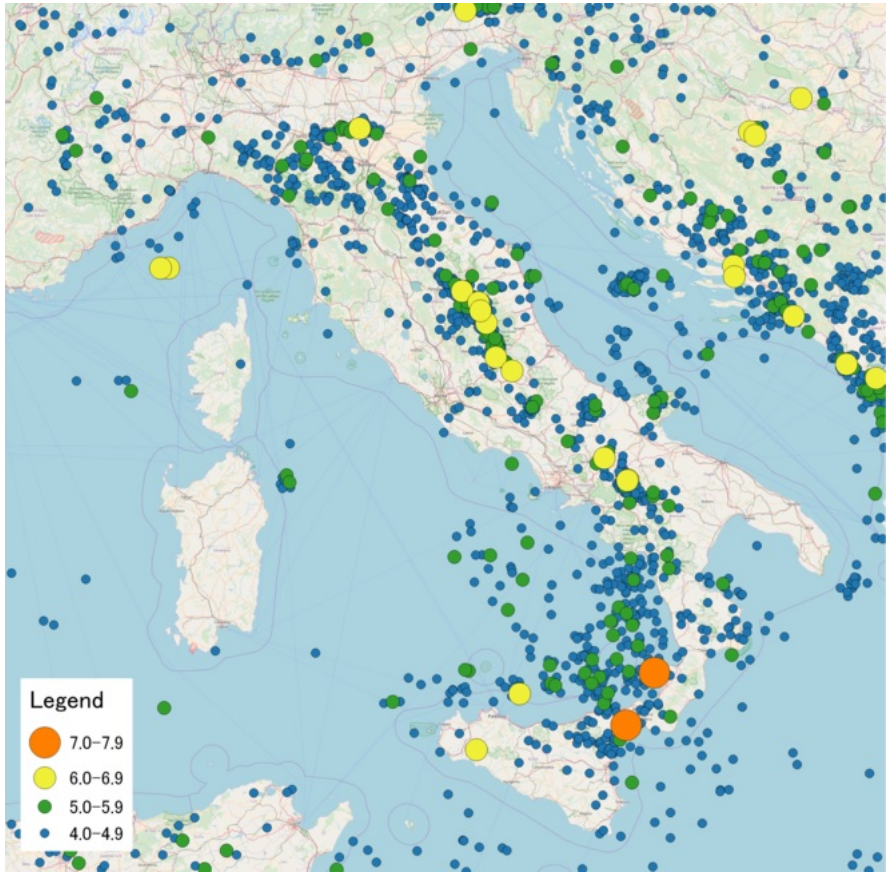
Examples of point pattern data [1]

- In 1854, cholera hit the city of London. No one knew where the disease started. So, British physician Snow started mapping the outbreak.
- But it wasn't just the disease; he also mapped out roads, property boundaries, and water lines, discovering that cholera cases were only along one water line.
- Breakthrough that connected geography to public health safety.
- The question of primary interest is whether, and if so where and when, statistically unusual local concentrations of cases occur.



Snow's map of the 1854 London cholera outbreak (obtained from `HistData` `R` package)

Examples of point pattern data [2]



Location of earthquakes in Italy 1900-2017 (moment magnitude scale).

Other examples in which points are the location of an event of interest:

- Location of crimes
- Location of trees

Data we will work with

In this course, we will work with:

- Areal or lattice data
- Geostatistical or spatial-referenced data

Spatial Data in R

Spatial data in R: Vectors

The are two fundamental distinctions: spatial vector data and raster.

- Vector represents the world using points, lines and polygons or combinations of those, where:
- *Point*, is a single point location, such as a geocoded address or a temperature sensor or the location of a bus stop;
- *Line*, is a set of ordered points, connected by straight line segments such as route travel or connections between locations;
- *Polygon*, is an area, marked by one or more enclosing lines such as local authority districts or census tracts.

```
> library(leaflet)
> popup = "Imperial College"
> leaflet() %>%
+ setView(lng = -0.1769, lat = 51.4983, zoom = 2.
+ addProviderTiles("NASAGIBS.ViirsEarthAtNight201
+ addMarkers(lng = -0.1769, lat = 51.4983,
+             popup = popup)
```

Spatial data in R: Rasters

- Raster: typically divides the surface into cells (also called pixels) of constant size. The cell of one raster hold a value, which might be numeric or categorical.

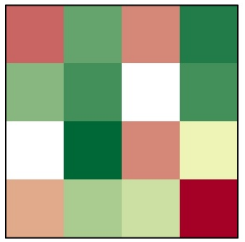
A. Cell IDs

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

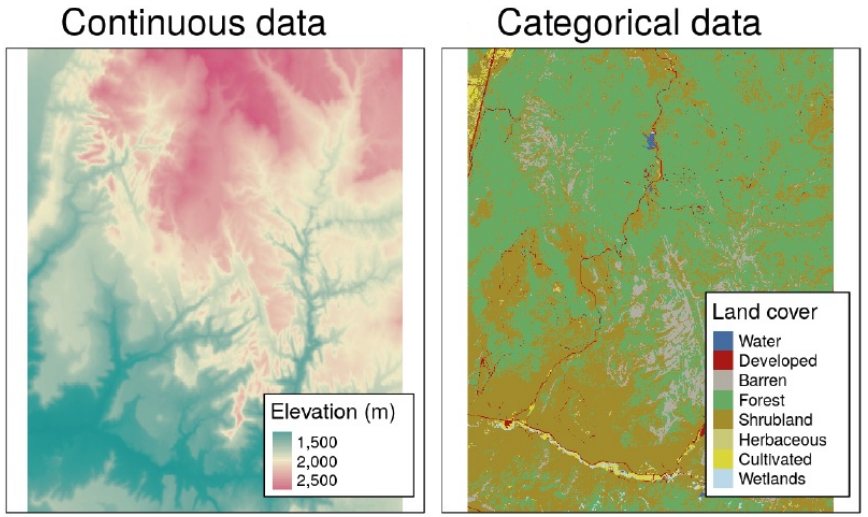
B. Cell values

22	74	28	91
72	84	NA	85
NA	92	24	53
31	62	56	5

C. Colored cell values



- Examples of continuous and categorical raster data.



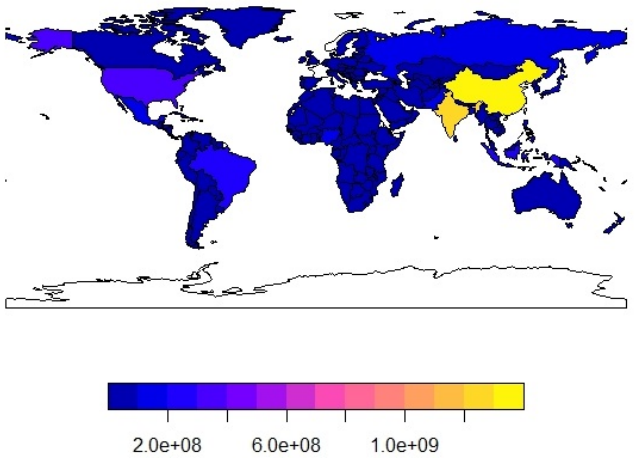
Source: Lovelace, Nowosad, and Muenchow (2019), Section 2.3.

Some useful R packages

- `sf` is a key package for spatial vector data (points, lines, polygons etc.). It refers to a formal standard that describes how objects in the real world can be represented in computers, with emphasis on the spatial geometry of these objects
- `sp` precedes `sf`. `sp` was first release on CRAN in 2005 and provides classes and methods to create points, lines, polygons, and grids and to operate on them. Some R packages still depends on the `sp` package
- `spdep` includes functions and tests for evaluating spatial patterns and autocorrelation
- `terra` contains classes and methods representing raster objects (but it has also its own vector data classes). It allows raster data to be loaded, processed and saved.
- `stars` contains classes and methods representing spatio-temporal data (raster and vector data cubes)
- `ggplot2`, `tmap`, `leaflet`, `mapview` and `gganimate` for visualization and maps

sp and sf objects

```
> library(spData); library(sf); library(tidyverse)
>
> # world is a sf object containing a world map data
> # from Natural Earth with a few variables from World Bank
> world = st_read(system.file("shapes/world.gpkg", package="spData"))
> plot(world["pop"])                    # plot world population
>
> world_sp = as(world, "Spatial")  # from sf to sp object
> class(world_sp)                       # "SpatialPolygonsDataFrame"
>
> world_sf = st_as_sf(world_sp)    # from sp to sf object
> class(world_sf)                       # "sf"  "data.frame"
```

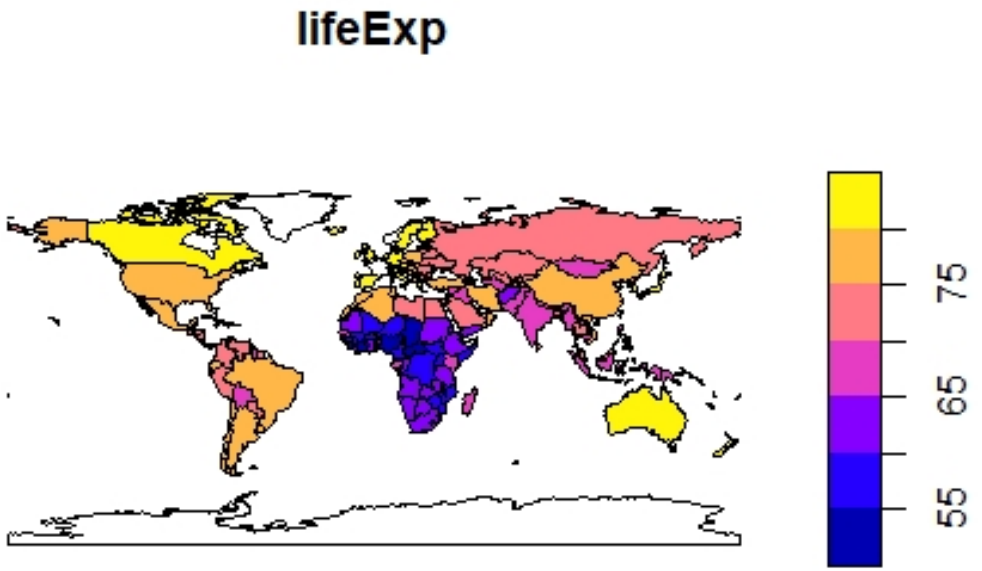


sp and sf

`sf` works well with the `tidyverse` collection of R packages.

For example, functions can be combined using the pipe operator $\%>\%$ given that both packages are loaded

```
> # Select and plot information for a single attribute
> world %>% select(lifeExp) %>% plot()
```

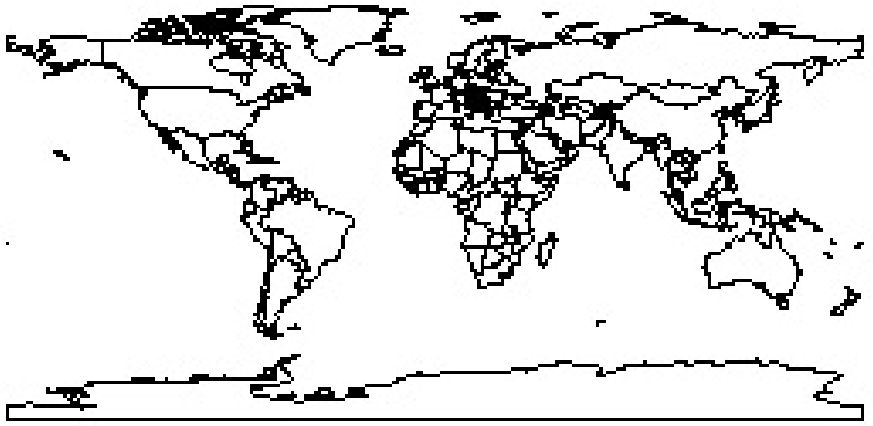


sp and sf

`sf` object includes spatial metadata like the coordinate reference system (CRS), which are stored in a list column.

We can extract and plot only the geometry with the function `st_geometry()`

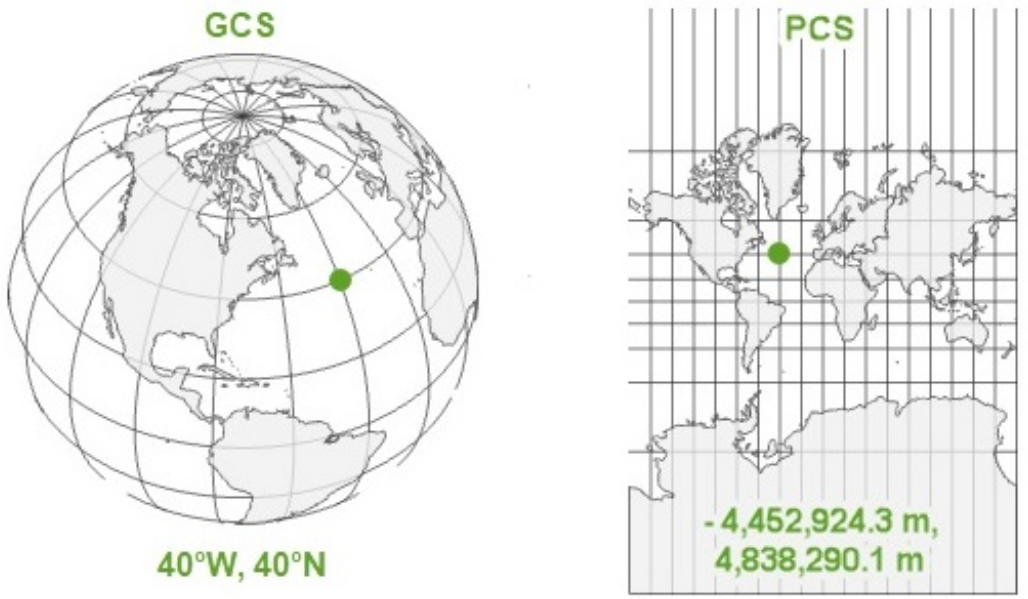
```
> # Extract geometry
> worlg_geo = st_geometry(world)
> # Extract and plot out only the geometries
> world %>% st_geometry() %>% plot()
```



Coordinate Reference Systems

Coordinate Reference Systems (CRS)

- A CRS defines how the spatial elements of the data relate to the surface of the Earth.
- There are two main types of CRS:
  1. Geographic coordinate reference systems (GCRS): coordinate systems that identify any location on the Earth's surface using longitude and latitude, with units in decimal degrees or degrees.
  2. Projected coordinate reference systems (PCRS): coordinate systems that provide various mechanisms to project maps of the Earth's ellipsoid shape onto a two-dimensional Cartesian coordinate plan, with units on feet or meter. Projected coordinate systems are referred to as map projections.

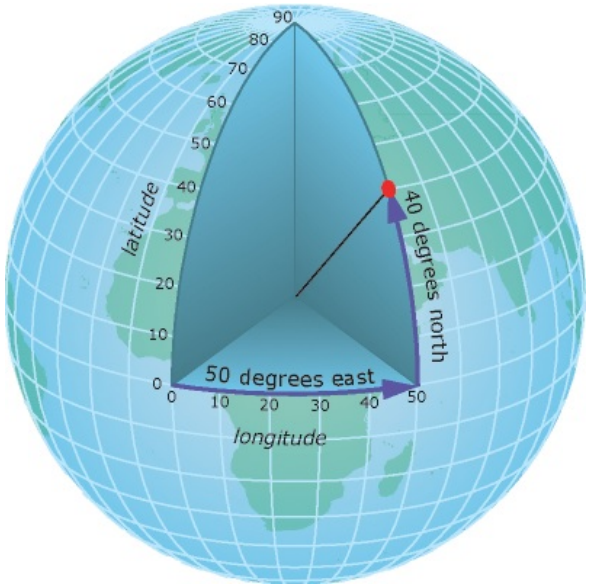


Source: https://www.esri.com/arcgis-blog/products/arcgis-pro/mapping/gcs_vs_pcs/

GCRS [1]

Geographic CRS refers to a point on the Earth by its latitude and longitude values, which are angles measured from the Earth's center to a point on the Earth's surface.

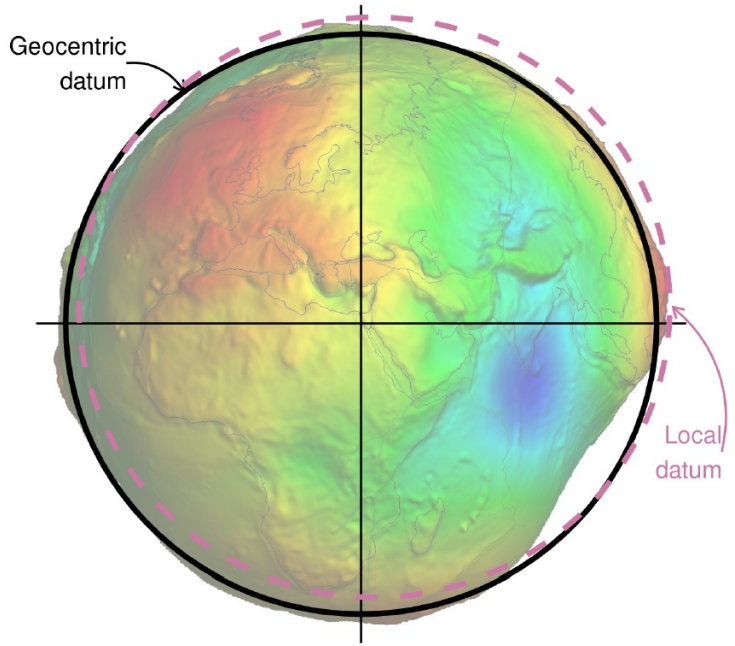
- The latitude of a point on Earth's surface is the angle between the equatorial plane and the line that passes through that point and the center of the Earth. Latitude values are measured relative to the equator (0 degrees) and range from -90 degrees at the south pole to 90 degrees at the north pole.
- The longitude of a point on the Earth's surface is the angle west or east of the prime meridian to another meridian that passes through that point. Longitude values range from -180 degrees when running west to 180 degrees when running east.



Source: https://annefou.github.io/metos_python/03-crs_proj/

GCRS [2]

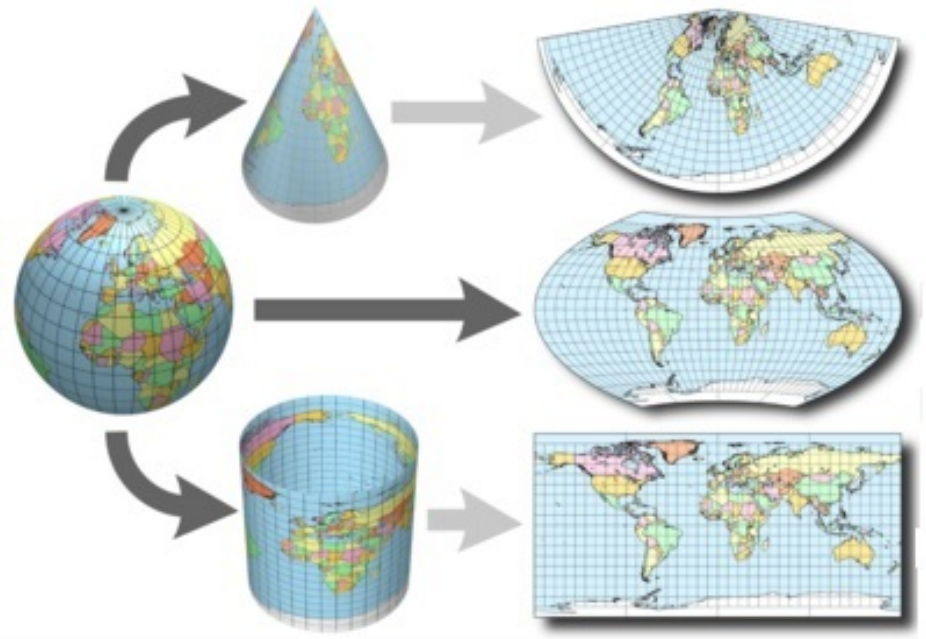
- Obviously we cannot actually measure these angles, but we can estimate them. To do so, we need a model of the shape of the Earth. Such a model is called as the datum.
- Datum contains information on what **ellipsoid** is used to approximate the Earth's shape and the relationship between the coordinate system and location on the Earth's surface; it also describes the origin of a coordinate system.
- There are two types of datum: geocentric and local



Source: Lovelace, Nowosad, and Muenchow (2019), Section 2.4.

PCRS [1]

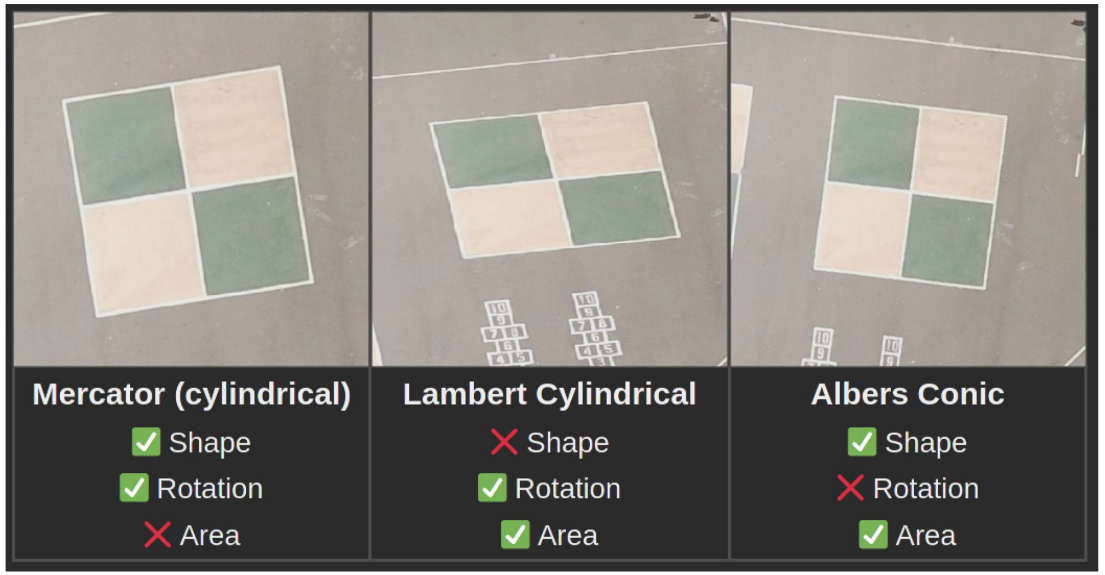
- Projected CRS are based on a GCRS and rely on map projections to convert the three-dimensional surface of the Earth into x and y values.
- The transition leads to some distortions: there is not one best projection. Some projections can be used for a map of the whole world; other projections are appropriate for small areas only.
- There are three main projection types: conic, cylindrical, and planar:



Source: https://www.earthdatascience.org/courses/earth-analytics/spatial-data-r/geographic-vs-projected-coordinate-reference-systems-UTM/

PCRS [2]

- Projected CRS is optimized to best represent the shape, and/or the rotation, and/or the area of features in a data set.
- There is not a single system that does a great job at optimizing all these elements: shape, rotation and area.



Source: https://ihatecoordinatesystems.com/

A funny and explanatory YouTube video on the trade off between GCRS and PCRS is at this link: https://www.youtube.com/watch?v=kIID5FDi2JQ

Re-projecting

Re-projecting is the process of changing the representation of locations from one CRS to another. When should data be transformed?

- The projections of locations on the Earth into a two-dimensional plane are distortions, the projection that is best for an application may be different from the projection associated with the data we import. In these cases, data can be re-projected.
- Another case is when two objects with different CRSs must be compared or combined. Thus, in the case of dealing with multiple data, re-projection permits to transform all data to a common CRS.
- In R, to transform data to a different projection, we can use `st_transform()` function of the `sf` package.

CRS in R (before 2020)

- Spatial R packages support a wide range of CRSs and they use the `PROJ` library to perform conversions between cartographic projections.

- Before 2020, there are two ways of defining a CRS:
  - the proj4string definition, which specifies attributes such as the projection, the ellipsoid and the datum, for example the WGS84 longitude and latitude projection is specified as:
    $+proj = longlat + ellps = WGS84 + datum = WGS84 + no\_defs$
  - the EPSG numeric code (EPSG stands for European Petroleum Survey Group). The code refers to only one, well-defined coordinate reference system, for example the EPSG code of the WGS84 projection is 4326 (Note that the EPSG code could not be available for a particular coordinate system; https://spatialreference.org/ref/epsg/)

CRS in R (after 2020)

- There are new developments at level of `PROJ` (https://proj.org/), and shift from `proj4string` to the `OGC WKT2` representation (OGC stands for Open Geospatial Consortium, and WKT stands for Well-known Text formats)

- The CRS is represented by:
  – a set of mathematical rules for specifying how coordinates are to be assigned to points;
  – a set of parameters defining the position of the origin, the scale, and the orientation of a coordinate system (datum)
- There is capability of direct transformations from CRS to CRS
- The use of `proj4string` is discouraged (some `proj4string` string elements are not longer supported; also only three ellipsoids can be used). `proj4string` still can be used for coordinate operations (transformations between CRS).
- In `sf`, we can retrieve the CRS of an object using `st_crs()`

Example

```
> library(spData)
> library(sf)
> # Extract the CRS information from the sf object nz (New Zealand)
> st_crs(nz) # here EPSG:2193
>
> # Re-project
> nz_wgs = st_transform(nz, 4326)
> st_crs(nz_wgs) # now EPSG:4326
> # Extract the CRS information from the sf object world
> st_crs(world) #EPSG:4326
>
> # Re-project using the CRS of the sf object world
> nz_wgs = st_transform(nz, st_crs(world))
> st_crs(nz_wgs)
>
> # Remove the CRS from nz_wgs
> nz_wgs_NULL_crs = st_set_crs(nz_wgs, NA)
>
> # Plot
> par(mfrow = c(1, 3))
> plot(st_geometry(nz))
> plot(st_geometry(nz_wgs))
> plot(st_geometry(nz_wgs_NULL_crs))
```

Example

Plots of the results:

References

Cressie, N. (1993). *Statistics for Spatial Data*. John Wiley & Sons, Inc.

Lovelace, R., J. Nowosad, and J. Muenchow (2019). *Geocomputation with R*. Chapman and Hall/CRC.