
Clase N° 23

— Breve Introducción al Trabajo
con Datos Georreferenciados —

Motivación

A lo largo de la materia, vimos que estamos rodeados de datos y constantemente seguimos generándolos.



Motivación

A lo largo de la materia, vimos que estamos rodeados de datos y constantemente seguimos generándolos.

Algunos de estos datos tienen la peculiaridad de estar asociados a una determinada localización geográfica.

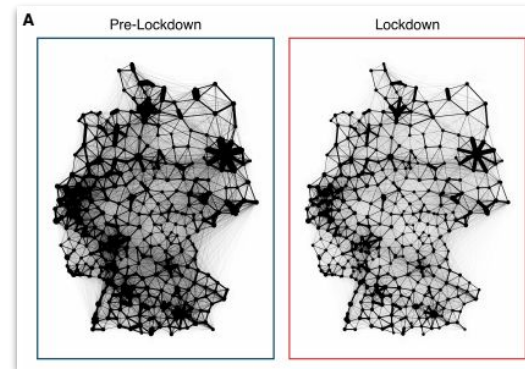
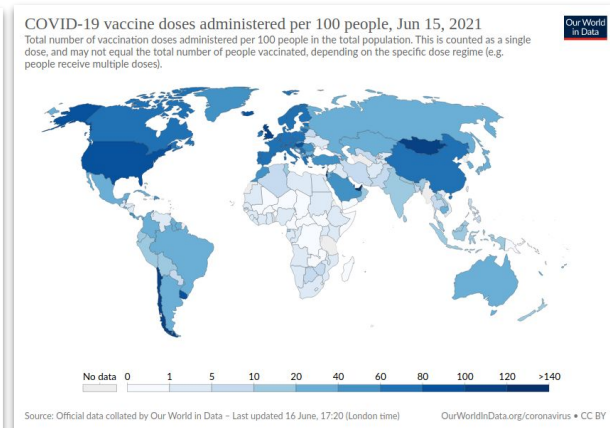
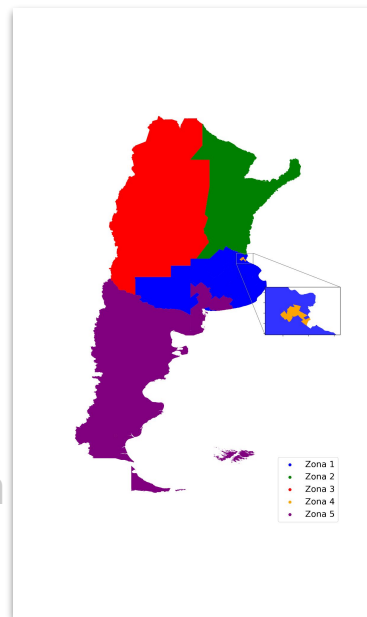


Motivación

A lo largo de la materia, vimos que estamos rodeados de datos y constantemente seguimos generándolos.

Algunos de estos datos tienen la peculiaridad de estar asociados a una determinada localización geográfica.

Explotar esta relación puede ser una herramienta interesante cuando estamos trabajando en estos problemas



Generalidades

A la hora de trabajar con datos georeferenciados, es probable que nos enfrentemos con:

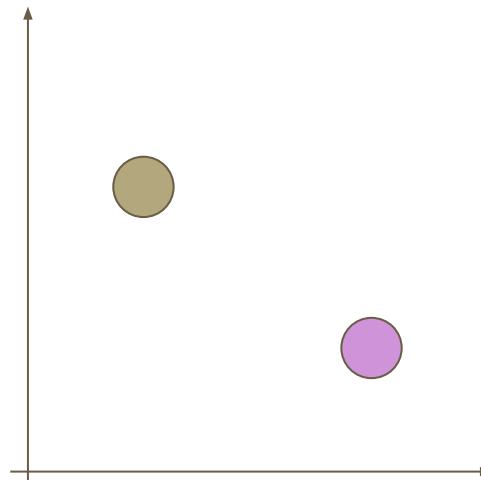
- Figuras geométricas
- Operaciones entre las mismas
- Proyecciones en las cuales nuestros datos están inmersos

Geometrías

Nos vamos a enfrentar con:

- Puntos
 - Par de coordenadas x-y

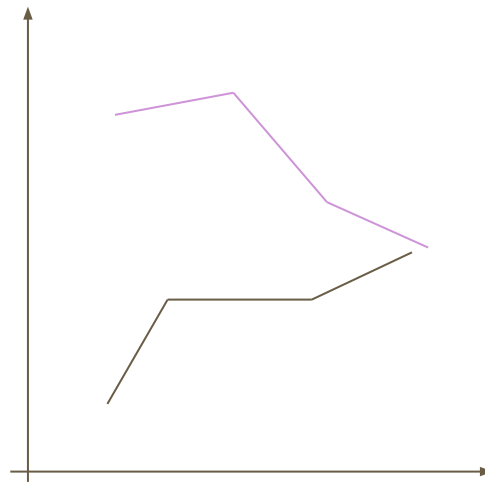
A *Point* has an *interior* set of exactly one point, a *boundary* set of exactly no points, and an *exterior* set of all other points. A *Point* has a topological dimension of 0.



Geometrías

Nos vamos a enfrentar con:

- Puntos
 - Par de coordenadas x-y
- Líneas
 - Conjunto de coordenadas x-y
 - No necesariamente cerrada



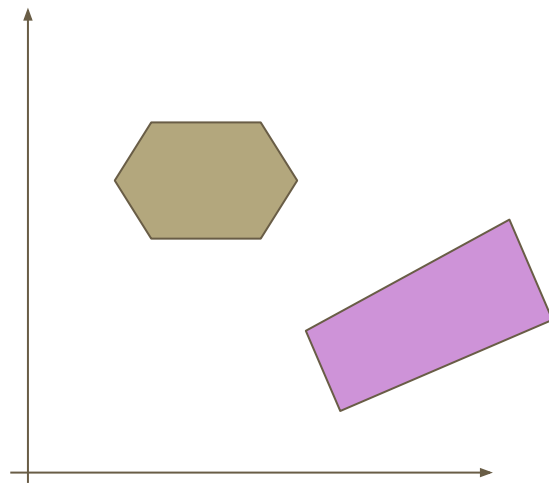
A *Curve* has an *interior* set consisting of the infinitely many points along its length (imagine a *Point* dragged in space), a *boundary* set consisting of its two end points, and an *exterior* set of all other points. A *Curve* has a topological dimension of 1.

Geometrías

Nos vamos a enfrentar con:

- Puntos
 - Par de coordenadas x-y
- Líneas
 - Conjunto de coordenadas x-y
 - No necesariamente cerrada
- Polígonos
 - Conjunto de coordenadas x-y
 - Cerrado

A *Surface* has an *interior* set consisting of the infinitely many points within (imagine a *Curve* dragged in space to cover an area), a *boundary* set consisting of one or more *Curves*, and an *exterior* set of all other points including those within holes that might exist in the surface. A *Surface* has a topological dimension of 2.



Geometrías - Operaciones

Podemos trabajar las distintas figuras a partir de:

- Distancias
 - Dependiendo el método que usemos, podremos trabajar con una distancia perpendicular o bien con la distancia entre los puntos más cercanos

Geometrías - Operaciones

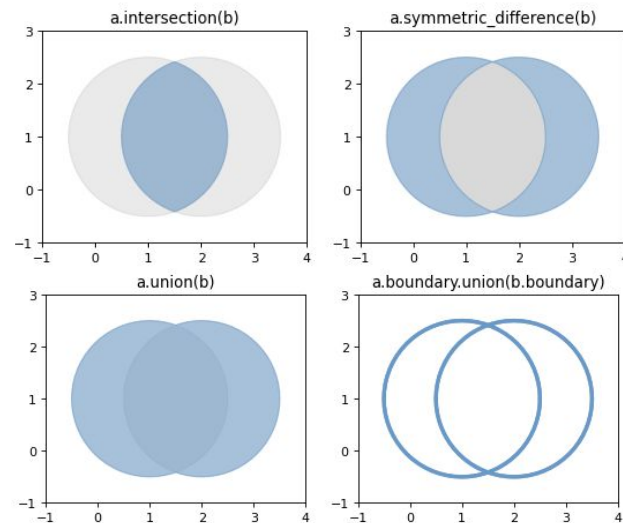
Podemos trabajar las distintas figuras a partir de:

- Distancias
- Tamaños (perímetro-largo/área)
 - Los objetos con los que trabajemos tendrán estos atributos

Geometrías - Operaciones

Podemos trabajar las distintas figuras a partir de:

- Distancias
- Tamaños (perímetro-largo/área)
- Intersecciones/Uniones
 - Como estamos trabajando con figuras geométricas, las mismas pueden trabajarse como conjuntos, y por lo tanto, existen las operaciones de intersección y unión



Geometrías - ¿y si la vida no es euclídea?

Trabajando con datos reales, nos enfrentaremos al siguiente problema: LA TIERRA NO ES PLANA



Geometrías - ¿y si la vida no es euclídea?

Trabajando con datos reales, nos enfrentaremos al siguiente problema: LA TIERRA NO ES PLANA

Entonces, debemos tratar ahora a nuestras figuras de forma distinta. Ya no estamos inmersos en un espacio euclídeo, sino que debemos dar cuenta de cómo se mapean las figuras según la curvatura terrestre.

Geometrías - ¿y si la vida no es euclídea?

Trabajando con datos reales, nos enfrentaremos al siguiente problema: LA TIERRA NO ES PLANA

Entonces, debemos tratar ahora a nuestras figuras de forma distinta. Ya no estamos inmersos en un espacio euclídeo, sino que debemos dar cuenta de cómo se mapean las figuras según la curvatura terrestre.

Osea, las distancias y superficies se van a ver alterados. Necesitaremos usar programas que puedan trabajar con estas curvaturas

Bases de Datos Referenciadas

Pensemos que tenemos un conjunto de datos, cada registro al margen de otras categorías, presenta una/s correspondiente/s a la ubicación.

Bases de Datos Referenciadas

Pensemos que tenemos un conjunto de datos, cada registro al margen de otras categorías, presenta una/s correspondiente/s a la ubicación.

Entonces, no sólo tenemos información asociada al registro como teníamos antes, sino que también ahora sabemos a dónde corresponde cada registro.

Bases de Datos Referenciadas

Pensemos que tenemos un conjunto de datos, cada registro al margen de otras categorías, presenta una/s correspondiente/s a la ubicación.

Entonces, no sólo tenemos información asociada al registro como teníamos antes, sino que también ahora sabemos a dónde corresponde cada registro.

Ejemplos:

- Casos diarios de COVID, registros telefónicos
- Capitales, Países
- Ubicación de servicios
- Tránsito, recorridos de colectivos, etc.

Proyecciones

Ahora bien, como mencionamos antes, si estos datos son de la Tierra, entonces, necesitamos decir de qué forma estamos trabajando las coordenadas.

El sistema de referencia de coordenadas que usemos indica cómo hay que proyectar los datos con los que trabajamos.

Es importante tener en cuenta esto, sobre todo cuando trabajamos con múltiples fuentes de datos georeferenciados.

Proyecciones

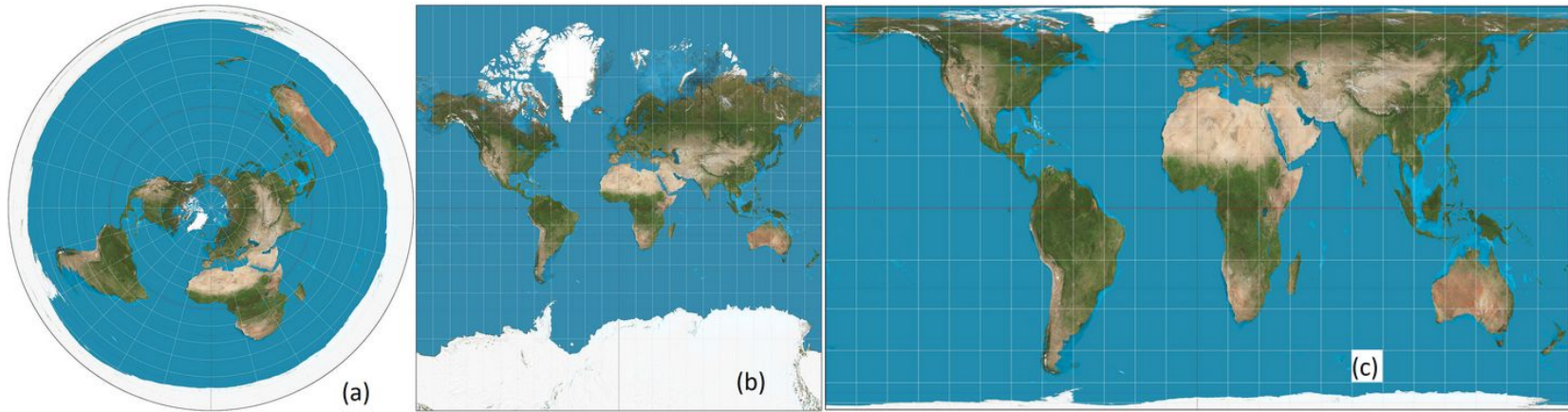


Figure 5: Map of the world in three different projections: (a) is in azimuthal projection that preserves distance from the center point, (b) is in aMercator projection that preserves shape, and (c) is in cylindrical equal-area projection that preserves area. Source: [Wikipedia](#).

Operaciones

Cuando trabajemos bases de datos de este tipo, podremos aplicar todas las operaciones válidas para las geometrías que ya vimos.

No obstante, hay que tener en cuenta lo último que vimos, ya que algunas librerías tratan de operar en geometrías planas y si nuestros datos no lo son, debemos acudir a otras.

Además, existen operaciones que rápidamente nos permiten obtener información entre dos set de datos distintos. Por ejemplo, hay veces que queremos hacer un merge entre datos, usando la columna de geometría.

Geometrías - GeoDataFrames en Python

En Python, las librerías que más suelen usarse son:

- [geopandas](#): manejo de bases de datos georeferenciadas
- [shapely](#): manejo de figuras geométricas en el plano
- [pyproj](#): manejo de figuras en otras geometrías
- [geopy](#): buscar ubicación lat-long de lugares. Labura con Internet, así que paciencia
- [dash/folium](#): manejo de mapas interactivos

Algunos recursos interesantes fuera de Python

Por fuera de python, podemos mencionar:

- [Leaflet](#): librería JS para manejo de mapas (las de Python interactúan con esta)
- Muchas páginas que permiten trabajar con datos georeferenciados:
 - <http://apps.headwallphotonics.com/> muy útil para ver y modificar polígonos con los que trabajemos
 - <https://www.arcgis.com/> creo que es de las herramientas más usadas en el área