

Clase 03: Fundamentos de Datos Geoespaciales

Modelos Vectorial y Raster

Profesor: Francisco Parra O.

20 de agosto, 2025

USACH - Ingeniería Civil en Informática

Objetivos de Aprendizaje

- **Comprender** los modelos fundamentales de datos espaciales
- **Diferenciar** entre representaciones vectorial y raster
- **Identificar** formatos de archivos y sus aplicaciones
- **Manipular** datos geoespaciales con Python
- **Seleccionar** el modelo apropiado según el problema

Introducción a los Modelos de Datos

Datos Vectoriales: Puntos, Líneas, Polígonos

Datos Raster: Grillas y Resolución

Introducción a los Modelos de Datos

¿Cómo representamos el mundo real?

El desafío:

- Mundo continuo e infinito
- Computadores discretos y finitos
- Múltiples fenómenos simultáneos
- Diferentes escalas y precisiones



Los Dos Paradigmas Fundamentales

Modelo Vectorial

- Objetos discretos
- Fronteras definidas
- Coordenadas exactas
- Ejemplos: calles, edificios, límites

Modelo Raster

- Campos continuos
- Grilla regular
- Valores en celdas
- Ejemplos: temperatura, elevación, NDVI

"La elección del modelo determina las operaciones posibles"

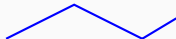
Datos Vectoriales: Puntos, Líneas, Polígonos

Geometrías Vectoriales Básicas



Punto

(x, y)



Línea

$[(x_1, y_1), (x_2, y_2), \dots]$



Polígono

Anillo cerrado

Geometría	Dimensión	Ejemplo en Chile
Punto	0D	Estaciones de Metro
Línea	1D	Red vial, ríos
Polígono	2D	Comunas, lagos

Puntos: Localización Exacta

Características:

- Coordenadas (x, y) o (lon, lat)
- Opcionalmente coordenada z
- Sin área ni longitud
- Atributos asociados

Aplicaciones:

- Ubicación de sensores
- Puntos de interés (POI)
- Eventos (delitos, accidentes)
- Muestras de campo

```
1 import geopandas as gpd
2 from shapely.geometry import Point
3
4 # Crear puntos de estaciones
5 estaciones = [
6     Point(-70.6693, -33.4489), # Santiago
7     Point(-70.6483, -33.4372), # La Moneda
8     Point(-70.6506, -33.4183), # Santa Ana
9 ]
10
11 # Crear GeoDataFrame
12 gdf = gpd.GeoDataFrame(
13     {'nombre': ['Santiago', 'La Moneda',
14                'Santa Ana'],
15     'linea': [1, 1, 2]},
16     geometry=estaciones,
17     crs='EPSG:4326'
18 )
19
20 # Operaciones
21 print(gdf.distance(gdf.iloc[0].geometry))
22 buffer = gdf.buffer(0.001) # 100m aprox
23
```

Listing 1: Puntos en GeoPandas

Líneas: Conectividad y Redes

Características:

- Secuencia ordenada de puntos
- Longitud medible
- Dirección (opcional)
- Topología de red

Aplicaciones:

- Redes de transporte
- Hidrografía
- Límites lineales
- Trayectorias GPS

```
1 from shapely.geometry import LineString
2 import osmnx as ox
3
4 # Crear línea simple
5 ruta = LineString([
6     (-70.669, -33.449),
7     (-70.665, -33.445),
8     (-70.660, -33.440)
9 ])
10
11 # Propiedades
12 print(f"Longitud: {ruta.length}")
13 print(f"Punto medio: {ruta.centroid}")
14
15 # Red vial con OSMnx
16 G = ox.graph_from_place(
17     'Providencia, Chile',
18     network_type='drive'
19 )
20
21 # Análisis de red
22 centrality = ox.betweenness centrality(G)
23 shortest_path = ox.shortest_path(
24     G, orig, dest, weight='length'
25 )
26
```

Listing 2: Líneas y redes

Polígonos: Áreas y Regiones

Características:

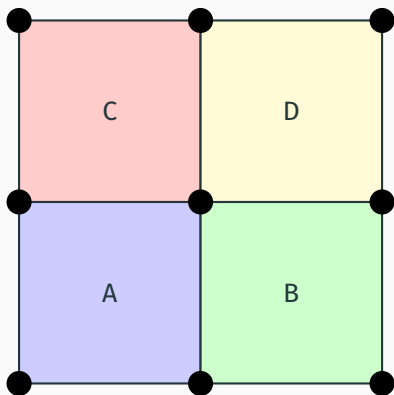
- Anillo exterior cerrado
- Posibles huecos internos
- Área y perímetro
- Relaciones topológicas

Aplicaciones:

- Límites administrativos
- Parcelas/predios
- Zonas de cobertura
- Áreas de influencia

```
1 from shapely.geometry import Polygon
2
3 # Crear polígono
4 manzana = Polygon([
5     (-70.650, -33.440),
6     (-70.648, -33.440),
7     (-70.648, -33.438),
8     (-70.650, -33.438),
9     (-70.650, -33.440)
10 ])
11
12 # Propiedades geométricas
13 print(f"Área: {manzana.area}")
14 print(f"Perímetro: {manzana.length}")
15
16 # Operaciones espaciales
17 edificio = Point(-70.649, -33.439)
18 print(manzana.contains(edificio)) # True
19
20 # Overlay de polígonos
21 interseccion = poligono1.intersection(poligono2)
22 union = poligono1.union(poligono2)
23 diferencia = poligono1.difference(poligono2)
24
```

Listing 3: Polígonos y análisis



Reglas topológicas:

- Sin gaps
- Sin overlaps
- Nodos compartidos
- Arcos compartidos

Datos Raster: Grillas y Resolución
