

Análisis espacial con Python

Clase 0

Objetivos

- Introducir las herramientas fundamentales del análisis espacial
- Hacerlo utilizando un abordaje programático o utilizando pensamiento computacional a través del lenguaje Python

A close-up portrait of Jon Snow from the TV series Game of Thrones. He is looking directly at the camera with a serious expression. He has dark hair and a beard, and is wearing a thick, dark fur cloak. The background is a snowy, out-of-focus landscape with falling snowflakes visible in the air.

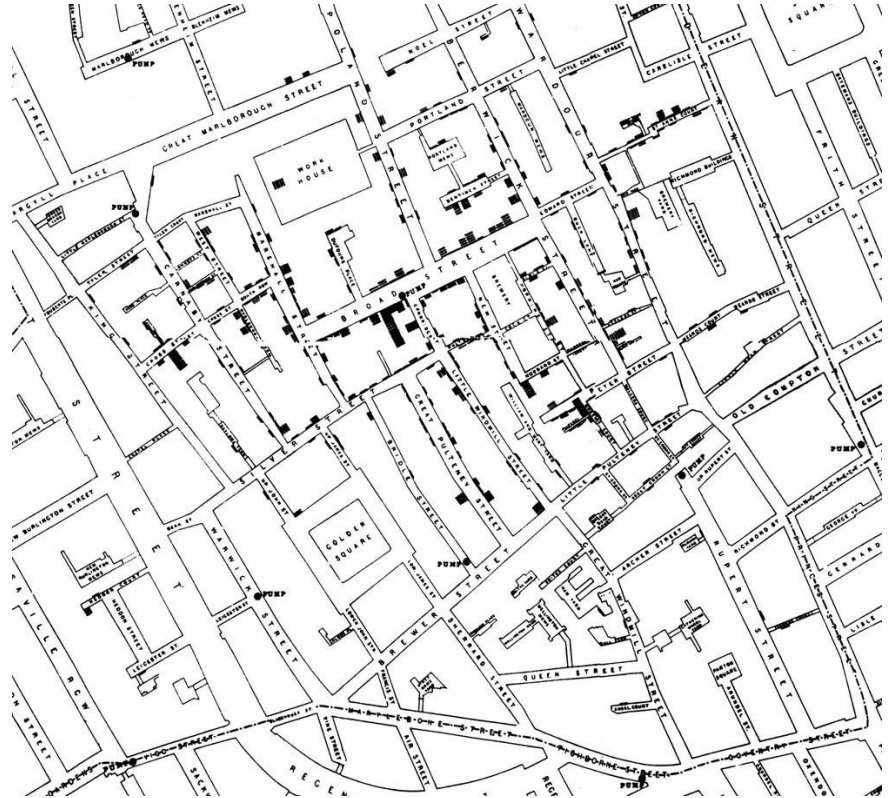
**Conozcan a uno de los pioneros del
análisis espacial**

... y su famoso mapa

En 1854, el Dr Jon Snow mapeo los casos de cólera en el barrio de Soho de Londres, descubriendo el patrón de contagio por agua a partir de un grifo y su ubicación.

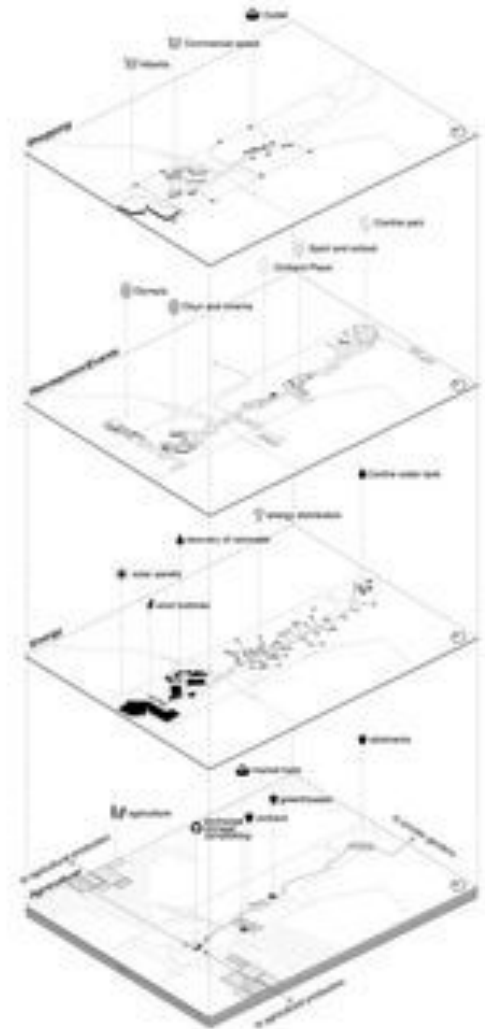
[Mapas Online 1](#)

[Mapas Online 2](#)



Análisis espacial

Entendemos por análisis espacial un caso particular de análisis de datos donde en el proceso de entender, describir, modelar y/o predecir, la ubicación en el espacio de las unidades de análisis (absoluta o relativa a otras) contiene información importante para dicho análisis.



... desde una perspectiva computacional

¿Por qué abordar un problema de análisis espacial utilizando un procedimiento computacional y programático en lugar del modo tradicional con point and click y software estilo QGIS o ARCGIS?

- Tareas sencillas pero muy reiterativas donde cambian pocos insumos (un mismo análisis que hay que actualizar con nuevos datos)
- Análisis customizados donde no existen las herramientas en los software tradicionales
- Tareas que impliquen una enorme cantidad de datos
- Garantizar la reproducibilidad y fiscalización del trabajo

Interactuar con computadoras



La interfaz gráfica

El nacimiento de la PC puso la computadoras en todos los hogares

Utilizó la interfaz gráfica para facilitar acceso

Potenció el acceso y las posibilidades de uso

Pero su uso exclusivo y dependencia, las limita



La interfaz gráfica



Fue útil por un tiempo, pero eventualmente para desbloquear el potencial de la herramienta, debemos dejarla de lado o por lo menos, no depender de ella de manera exclusiva

Pensamiento computacional

Uno de los objetivos es desarrollar el pensamiento computacional, no solo “aprender a programar” o aprender Python.

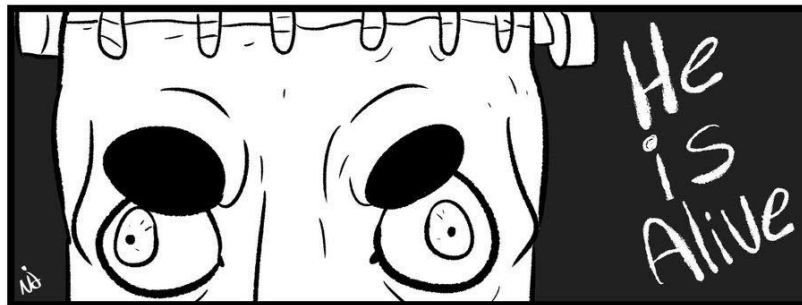
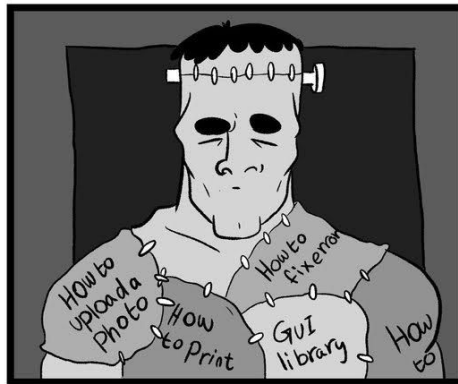
Los lenguajes, librerías y funciones cambian. No se trata de memorizar “como se hace tal proceso en Python”, sino a aprender a aprender, a obtener la gimnasia y los conocimientos generales para saber donde y como ir a buscar lo que no se sabe.

A veces saber programar bien es saber buscar bien en Google

**NOT SURE IF I AM A GOOD
PROGRAMMING**

**OR GOOD AT
GOOGLING**

Stackoverflow es
nuestro mejor amigo



Nunca desesperen, siempre va a haber errores... a nadie le sale en el primer intento

<https://thecodinglove.com/when-my-cast-function-works-on-first-tr>
y



Python



Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Fue desarrollado por Guido van Rossum

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional.

Actualmente tiene dos versiones 2.7 y 3

Su licencia es libre y compatible con GPL (GNU Public License)

El Zen de Python (por Tim Peters)

- Lindo es mejor que feo.
- **Explícito es mejor que implícito.**
- Simple es mejor que complejo.
- Complejo es mejor que complicado.
- Plano es mejor que anidado.
- Espaciado es mejor que denso.
- La legibilidad es importante.
- Los casos especiales no son lo suficientemente especiales como para romper las reglas.
- **Sin embargo la practicidad le gana a la pureza.**

El Zen de Python

- **Los errores nunca deberían pasar silenciosamente.**
- A menos que se silencien explícitamente.
- **Frente a la ambigüedad, evitá la tentación de adivinar.**
- **Debería haber una, y solamente una, manera obvia de hacerlo.**
- A pesar de que no sea obvio a menos que seas Holandés (como GvR)
- **Ahora es mejor que nunca.**
- **A pesar de que nunca es muchas veces mejor que *justo ahora*.**
- Si la implementación es difícil de explicar, es una mala idea.
- **Si la implementación es fácil de explicar, quizás sea una buena idea.**
- Los espacios de nombres son una gran idea, ¡tenemos más de esos!

Jupyter notebooks

Se estructuran en celdas y en cada celda podremos ir ejecutando instrucciones y órdenes en código Python y ver el resultado en el momento.

También podemos introducir texto e imágenes.



Librerías de Python

Las librerías de Python pueden considerarse como plug-ins, funcionalidades adicionales desarrolladas por la comunidad que le suman herramientas al programa original.

Por eso la importancia de usar software libre, puede canalizarse la energía de una comunidad dispersa a nivel mundial con enorme potencial

NumPy



NumPy es una extensión de Python, que le agrega mayor soporte para vectores y matrices, constituyendo una biblioteca de funciones matemáticas de alto nivel para operar con esos vectores o matrices. El ancestro de NumPy, Numeric, fue creado originalmente por Jim Hugunin con algunas contribuciones de otros desarrolladores.

En 2005, Travis Oliphant creó NumPy incorporando características de Numarray en NumPy con algunas modificaciones. NumPy es software **LIBRE**

Scipy



SciPy es una biblioteca **LIBRE** de herramientas y algoritmos matemáticos para Python que nació a partir de la colección original de Travis Oliphant que consistía de módulos de extensión para Python, lanzada en 1999.

SciPy contiene módulos para optimización, álgebra lineal, integración, interpolación, funciones especiales, FFT, procesamiento de señales y de imagen, resolución de ODEs y otras tareas para la ciencia e ingeniería.

Geopandas

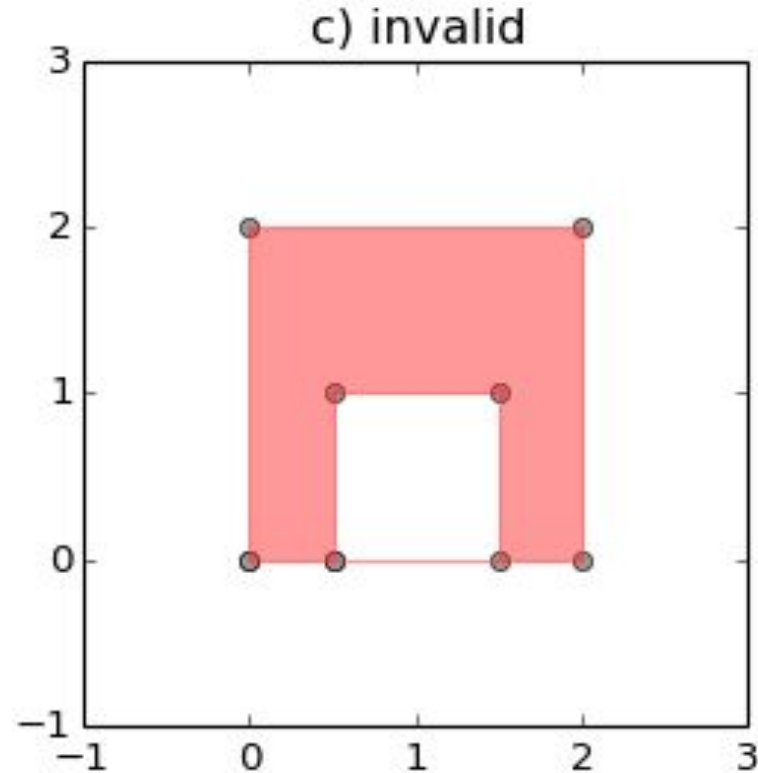
El objetivo de GeoPandas es facilitar el trabajo con datos geoespaciales en Python. Combina las capacidades de los pandas y las formas, proporcionando operaciones geoespaciales en pandas y una interfaz de alto nivel para múltiples geometrías a formas. GeoPandas le permite realizar fácilmente operaciones en Python que de otra manera requerirían una base de datos espacial como PostGIS.



Shapely

Es una librería que permite la creación, edición y manipulación de objetos espaciales.

```
ext = [(0, 0), (0, 2), (2, 2), (2, 0), (0, 0)]  
int = [(0.5, 0), (1.5, 0), (1.5, 1), (0.5, 1), (0.5, 0)]  
polygon = Polygon(ext, [int])
```



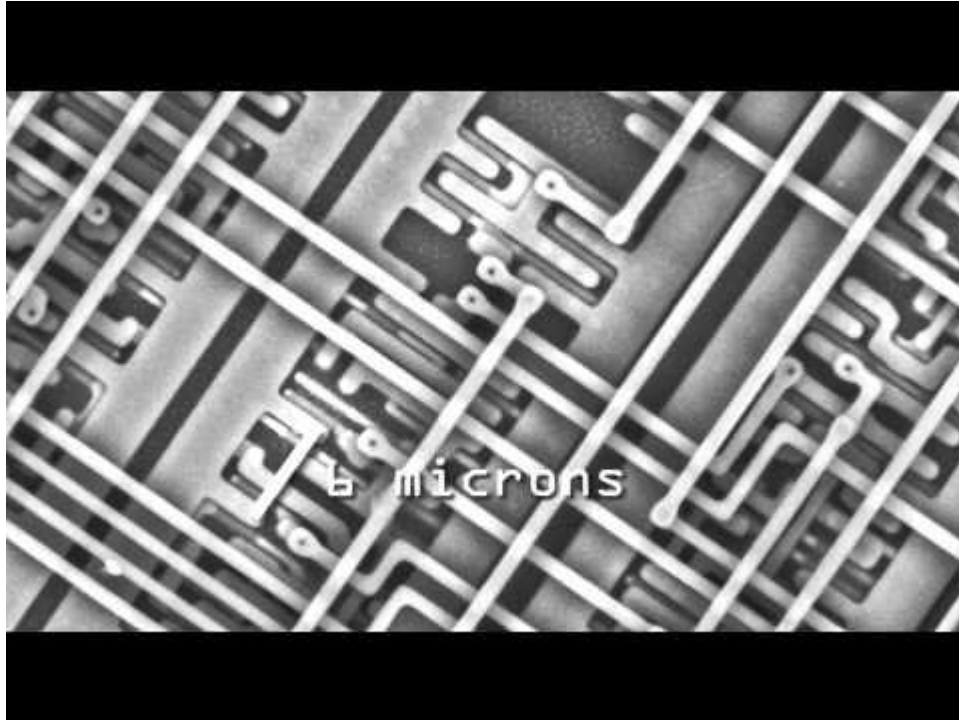
Pedagogía de muñecas rusas

Hay diferentes niveles de abstracción que debemos manejar

Debemos saber hasta qué muñeca rusa queremos llegar



Niveles de abstracción: alto o bajo nivel



Empecemos...

Durante parte de esta clase y la próxima, por momentos podemos sentir que solamente hacemos ejercicios de encerar y pulir, pero progresivamente irá cobrando sentido



Caja de herramientas

Utilizaremos fundamentalmente el entorno de Google Drive. Tendrán acceso a un directorio compartido donde podrán ver e interactuar con los jupyter notebooks.

Por un lado veremos el código en estos notebooks utilizando Colab.

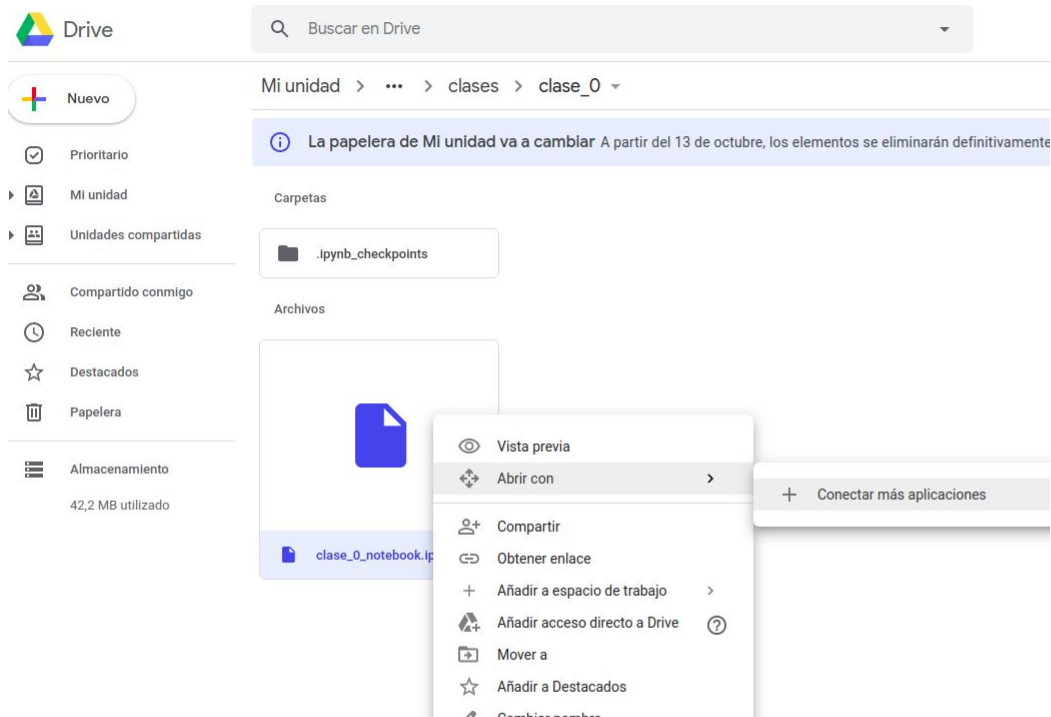
Los archivos de datos que usemos estarán en el Drive compartido. Pero es importante que cada uno tenga **una copia propia** en su Drive personal.



Usar Colab para abrir un notebook por primera vez

Haciendo botón derecho sobre el notebook /
Abrir con...

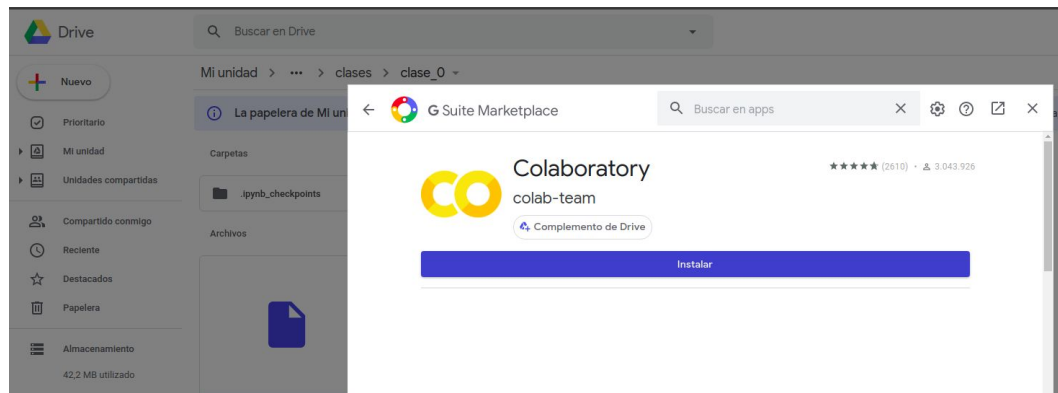
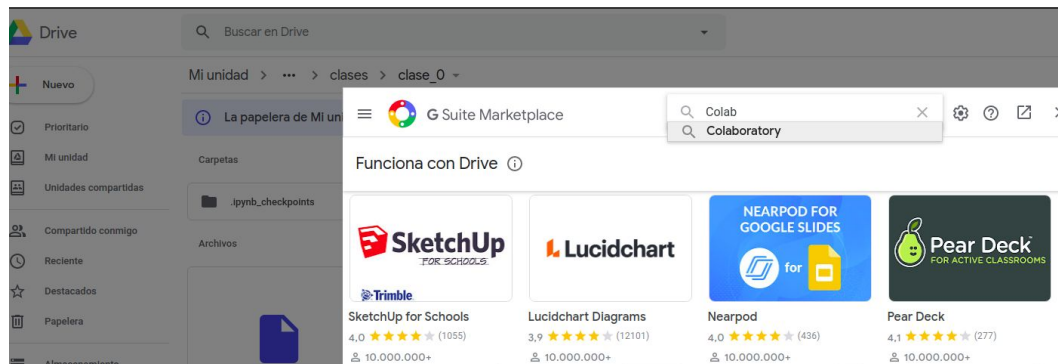
Si Colab no esta
instalado en Drive, hay
que instalarlo con
*Conectar más
aplicaciones*



Instalar Colab

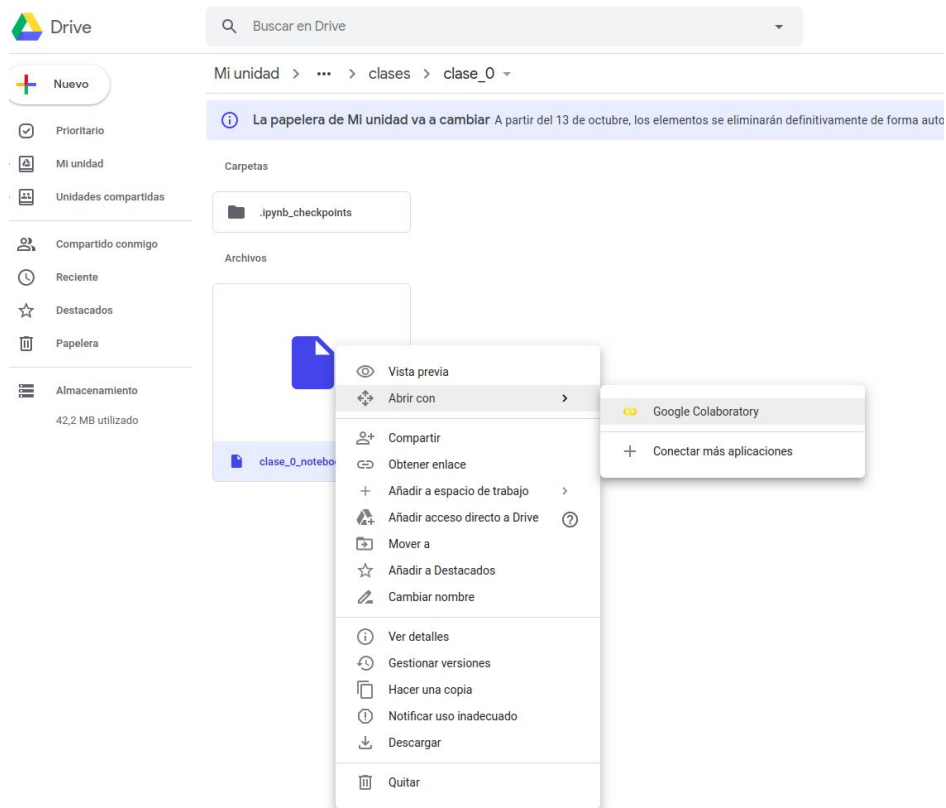
Buscamos Colaboratory en la tienda de aplicaciones de Google Suites.

Y lo instalamos



Usar Colab para abrir un notebook por primera vez

Una vez instalado, nos aparecerá como opción para abrir el notebook



Github

En Github van a poder encontrar todos los notebooks y lo datos en una estructura clase por clase donde podrán descargar todo junto e ir subiendo a Drive

Link:

https://github.com/alephcero/untref_analisis_especial_2020

The screenshot shows the GitHub interface for the repository `alephcero/untref_analisis_especial_2020`. The repository is described as "Repositorio de datos para el Seminario Analisis Espacial - MGAIE UNTREF". It has 887754 files, 19 hours ago, and 5 commits. The file list includes:

File	Commit	Time
carto	primer borrador clase 0	16 days ago
clases/clase_0	limpiar notebook	19 hours ago
LICENSE	Initial commit	16 days ago
README.md	Initial commit	16 days ago

The README.md file is visible, showing the repository title `untref_analisis_especial_2020` and the description "Repositorio de datos para el Seminario Analisis Espacial - MGAIE UNTREF".

Tarea para el hogar

Alto nivel

Cuando en la clase 2 veamos datos tabulares y Pandas, podremos explorar diferentes datasets por ejemplo en data.buenosaires.gob.ar. Pero pueden ir buscando set de datos y generar preguntas que les gustaría resolver

Bajo nivel

Hay muchos ejercicios online para ir probando, cómo definir una función para calcular el IVA de un producto. Este es un video donde pueden ver varios:

