

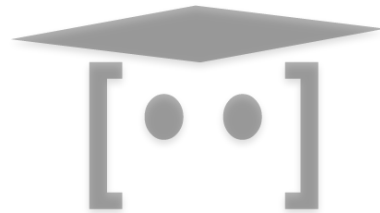
ANÁLISIS DE DATOS

Profesor: Nahuel Meza

Cod.Ar[••]

REPASO

¿Que vimos la clase anterior?



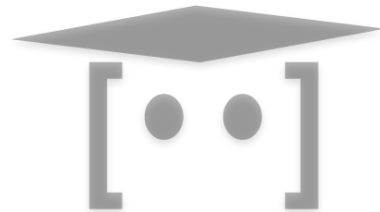
Dos grandes librerías

Dentro del mundo del análisis, existen dos librerías muy utilizadas. Que permiten una versatilidad a la hora de ver datos presentados o recaudados



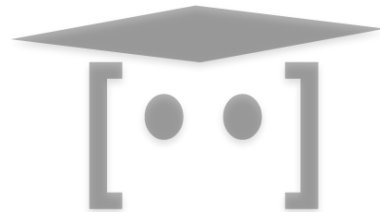
Dos grandes librerías

- Pandas 
- Matplotlib 



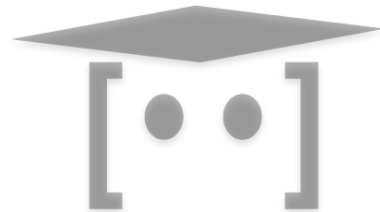
Librería especializada en la visualización de datos

```
import matplotlib.pyplot as plt
```



Nos permite crear distintos tipos de gráficos en base a los datos que le demos.

Esto nos da la capacidad de representar de manera entendible lo que queramos presentar.

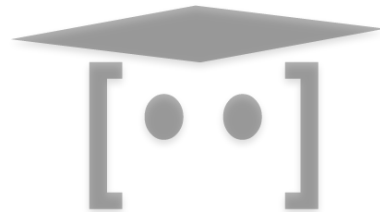


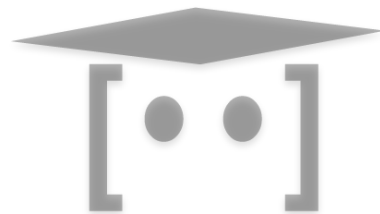
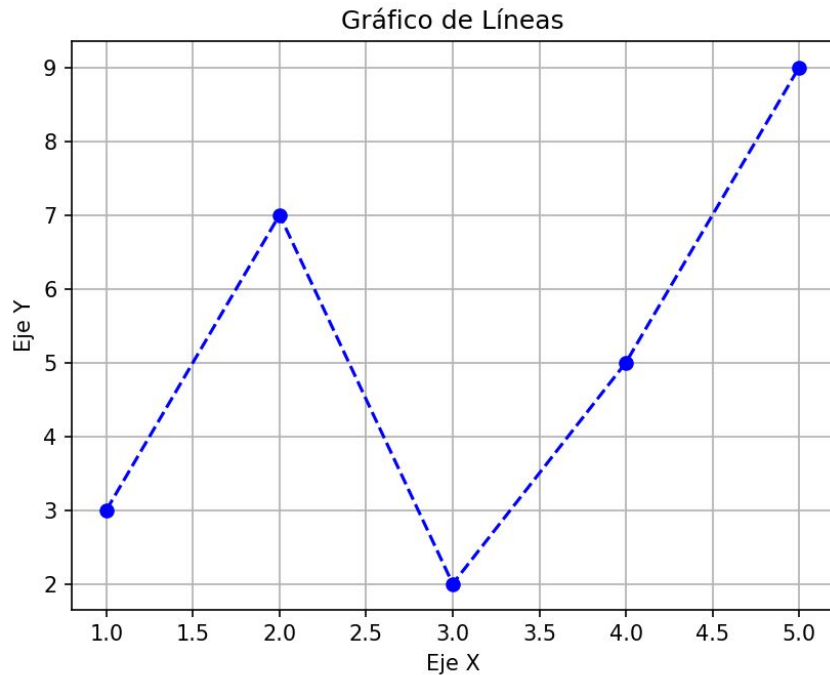
Matplotlib

```
import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5]
y = [3, 7, 2, 5, 9]

plt.plot(x, y, color='blue', marker='o', linestyle='--')
plt.title('Gráfico de Líneas')
plt.xlabel('Eje X')
plt.ylabel('Eje Y')
plt.grid()
plt.show()
```





Función

`plt.plot()`

`plt.bar()`

`plt.barh()`

`plt.scatter()`

`plt.hist()`

`plt.pie()`

`plt.subplot()`

`plt.title(), plt.xlabel(), plt.ylabel()`

`plt.legend()`

`plt.grid()`

`plt.savefig()`

Descripción

Gráfico de líneas

Gráfico de barras

Barras horizontales

Dispersión

Histograma

Gráfico de pastel

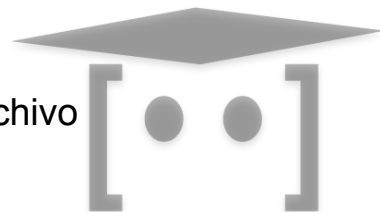
Múltiples subgráficos

Títulos y etiquetas

Leyenda

Cuadrícula

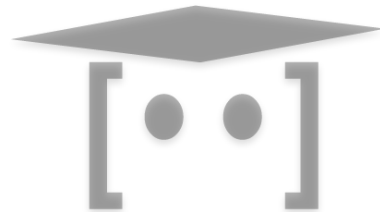
Guardar gráfico en archivo



Una de las librerías más utilizadas para la manipulación y análisis de datos.

Ofrece herramientas para:

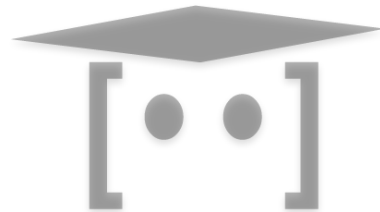
- Leer y escribir archivos
- Ordenar, filtrar y agrupar datos
- Limpiar y transformar información
- Trabajar con fechas y datos faltantes



Ejemplificamos

Para tener un mejor entendimiento de ambas librerías, tendremos ejemplos detallados dentro del IDE.

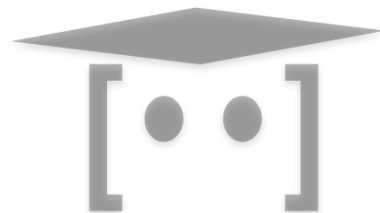
Antes de esto, es importante aclarar que ambas librerías son EXTERNAS. Es decir que es necesario instalarlas antes de poder utilizarlas



Instalación de entorno

Python nos provee de un entorno virtual en donde podemos instalar las librerías necesarias para nuestro proyecto.

En unos pocos comandos podemos tener ese entorno listo.



Instalación de entorno

```
python -m venv nombre_del_entorno
```

comúnmente llamado .venv

◦ En Windows:

Código

```
nombre_del_entorno\Scripts\activate
```

En macOS y Linux. [🔗](#)

Código

```
source nombre_del_entorno/bin/activate
```



Instalación de entorno

```
pip install nombre_libreria
```

En nuestro caso sería

- matplotlib
- pandas

```
deactivate
```

para cerrar el espacio virtual

Con eso estaríamos listos para pasar al código!

