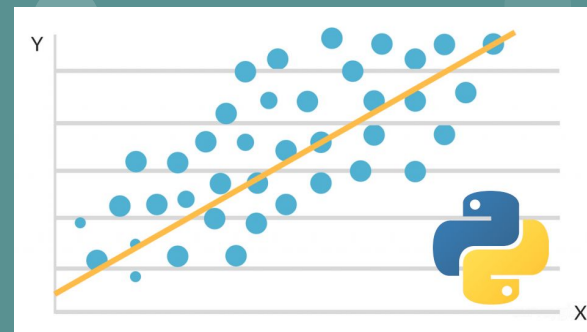


TL 1 - Regresión Lineal.



Docente: Ing. D'Angiolo Federico

- Cabot Lucas
- Calonge Federico
- Lew Imanol



1a) y 1b) Dataset utilizado



Objetivos:

1a) Analizar la media y el desvío standard

1b) Realizar modelo matemático basado en la Regresión Lineal

Datos_Presión.csv 15.6 KB	
1	1000.83,1000.33
2	1000.64,1000.24
3	1000.81,1000.48
4	1000.80,1000.56
5	1001.28,1001.01
6	1001.69,1001.47
7	1002.02,1001.77
8	1001.64,1001.44
9	1001.17,1001.06
10	1001.52,1001.31
11	1001.58,1001.35
12	1001.54,1001.50
13	1001.56,1001.41
14	1001.25,1001.04
15	1001.24,1001.02
16	1001.26,1001.20
17	1001.03,1000.92
18	1000.87,1000.29
19	1000.65,1000.50
20	1000.68,1000.55

1a) Media y Desvío Estándar



Para obtener la media y desvío standard de los valores de nuestro dataset, primero corroboramos que este pueda ser obtenido y leído, por lo que *printeamos* las primeras 10 filas del dataset.

```
In [2]: my_csv = "Datos_Presión.csv"
names = ['presionA', 'presionB']
dataset_presion = pd.read_csv(my_csv, names=names)
```

```
In [4]: dataset_presion.head(10)
```

Out[4]:

	presionA	presionB
0	1000.83	1000.33
1	1000.64	1000.24
2	1000.81	1000.48
3	1000.80	1000.56
4	1001.28	1001.01
5	1001.69	1001.47
6	1002.02	1001.77
7	1001.64	1001.44
8	1001.17	1001.06
9	1001.52	1001.31

1a) Media y Desvío Estándar



Luego, gracias a la librería “Pandas” podemos simplemente pedirle que nos muestre varios datos estadísticos de nuestro conjunto:

```
print(dataset_presion.describe())
```

Lo que nos devuelve:

	presionA	presionB
count	999.000000	999.000000
mean	1016.580450	1016.841832
std	6.962852	7.058162
min	1000.560000	1000.240000
25%	1015.375000	1015.685000
50%	1017.940000	1018.280000
75%	1019.190000	1019.485000
max	1025.880000	1026.150000

La media de la presionA es 1016,58. El desvío estándar es 6,96, lo que significa que la mayoría de las mediciones de la presionA se encuentran entre 1009,62 y 1023,54.

#La media de la presionB es 1016,84. El desvío estándar es 7,05, lo que significa que la mayoría de las mediciones #de la presionB se encuentran entre 1009.79 y 1023.89

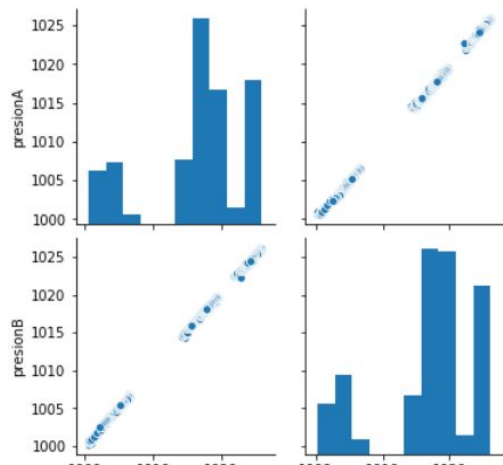
1b) Modelo Matemático basado en Regresión Lineal

Primero, para realizar la regresión lineal debemos verificar que el modelo tenga una relación lineal (puede ser positiva o negativa).

Esto lo verificamos por medio de la librería “seaborn”, que nos permite graficar y observar la relación entre variables.

```
In [5]: sns.pairplot(dataset_presion)  
#Como el output resulta ser una Relación lineal positiva , podemos hacer uso del algoritmo Regresion Lineal.
```

```
Out[5]: <seaborn.axisgrid.PairGrid at 0x7f30c0414dd0>
```



1b) Modelo Matemático basado en Regresión Lineal (Cont)

$$f(x_i) = \alpha_0 + \sum_{i=1}^m \alpha_i x_i \quad \text{donde } A = \{\alpha_0, \alpha_1, \dots, \alpha_m\}$$

2a) y 2b) Dataset utilizado



Para esta sección se utilizan 3 datasets, que si bien contienen valores distintos, la estructura es la misma. Se cuenta con datasets para casos confirmados, muertes, y casos de recuperaciones.

1	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20	1/26/20	1/27/20
2		Afghanistan	33.0	65.0	0	0	0	0	0	0
3		Albania	41.1533	20.1683	0	0	0	0	0	0
4		Algeria	28.0339	1.6596	0	0	0	0	0	0
5		Andorra	42.5063	1.5218	0	0	0	0	0	0
6		Angola	-11.2027	17.8739	0	0	0	0	0	0
7		Antigua and Barbuda	17.0608	-61.7964	0	0	0	0	0	0
8		Argentina	-38.4161	-63.6167	0	0	0	0	0	0
9		Armenia	40.0691	45.0382	0	0	0	0	0	0
10	Australian Capital Territory	Australia	-35.4735	149.0124	0	0	0	0	0	0
11	New South Wales	Australia	-33.8688	151.2093	0	0	0	0	3	4
12	Northern Territory	Australia	-12.4634	130.8456	0	0	0	0	0	0
13	Queensland	Australia	-28.0167	153.4	0	0	0	0	0	0

2a) Casos de infecciones, muertes y recuperaciones - Argentina



Codigo:

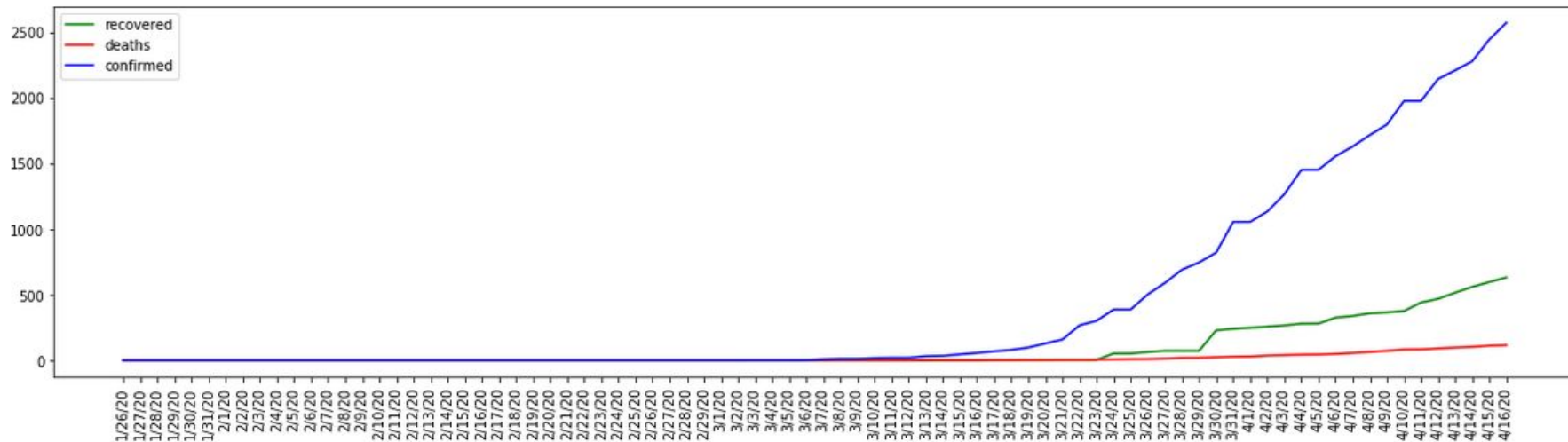
```
#asignamos la URL que contiene el CSV a una variable
url_recovered = "https://raw.githubusercontent.com/CS
#creamos nuestro dataframe a partir de la URL
dataset_recovered = pd.read_csv(url_recovered)
#filtramos en nuestro dataframe segun un pais deseado, que estara contenido en la variable "pais"
df_recovered = dataset_recovered[dataset_recovered['Country/Region']==pais]
#filtramos en nuestro dataframe unicamente las columnas en el slot 4 en adelante
df_recovered=df_recovered[df_recovered.columns[4:]]
dic_recovered=df_recovered.to_dict(orient='records')[0]

plt.figure(figsize=(20,5))
plt.plot(*zip(*(dic_recovered.items()))).label='recovered'.color='a')
plt.legend(loc='upper left')
plt.xticks(rotation=90)
plt.show()
```


2a) Casos de infecciones, muertes y recuperaciones - Argentina (Cont)



Salida:



2b) Top 3 países con casos confirmados

