

IMPORTACION DE LIBRERIAS

PANDAS -> Es indispensable en este proyecto porque nos ayuda con la manipulacion de las tablas de excel. NUMPY -> En este proyecto nos colabora con los calculos matematicos que se vayan analizar.

```
In [ ]: import pandas as pd
import numpy as np
import openpyxl
```

CARGA DE DATOS

Leemos el archivo en donde estan las ventas de la tienda de maquillaje (Beauty vibes) df = pd.read_excel -----> Nos ayuda a leer el archivo de excel. df.head -----> Nos muestra que el archivo se lea correctamente.

```
In [3]: df = pd.read_excel('./tests/productos_maquillaje.xlsx')
df.head(30)
```

Out[3]:

	cliente_id	producto	categoria	precio
0	1	base	maquillaje	35
1	2	corrector	maquillaje	30
2	3	labial mate	maquillaje	25
3	4	shampoo	cuidado_capilar	50
4	5	tonico	cuidado_capilar	45
5	6	tratamiento	cuidado_capilar	48
6	7	sombras	maquillaje	30
7	8	delineador	maquillaje	20
8	9	rubor	maquillaje	18
9	10	iluminador	maquillaje	15
10	11	acondicionador	cuidado_capilar	50
11	12	pestañina	maquillaje	20
12	13	polvo compacto	maquillaje	15
13	14	rubor	maquillaje	18
14	15	base	maquillaje	35
15	16	shampoo	cuidado_capilar	50
16	17	sombras	maquillaje	30
17	18	iluminador	maquillaje	15
18	19	corrector	maquillaje	30
19	20	polvo compacto	maquillaje	25
20	21	base	maquillaje	35
21	22	rubor	maquillaje	18
22	23	shampoo	cuidado_capilar	50
23	24	tratamiento	cuidado_capilar	45
24	25	delineador	maquillaje	15
25	26	pestañina	maquillaje	20
26	27	tonico	cuidado_capilar	30
27	28	polvo compacto	maquillaje	20
28	29	labial mate	maquillaje	15
29	30	iluminador	maquillaje	18

DIMENSION DEL DATA SET:

Esta linea nos muestra que el archivo esta conformado por 30 registros y 4 variables separados por filas y columnas en el excel y nos ayuda hacer una analisis exploratorio de las ventas obtenidas en la tienda de maquillaje.

```
In [4]: df.shape
```

```
Out[4]: (30, 4)
```

INFORMACION DE LOS DATOS:

Esta linea nos permite identificar el tipo de variable de cada columna y nos ayuda a identificar si existen valores nulos, esto es importante porque nos garantiza una mejor calidad antes de cualquier analisis.

```
In [5]: df.info()
```

```
<class 'pandas.DataFrame'>
RangeIndex: 30 entries, 0 to 29
Data columns (total 4 columns):
#   Column      Non-Null Count  Dtype
---  -
0   cliente_id  30 non-null    int64
1   producto    30 non-null    str
2   categoria   30 non-null    str
3   precio      30 non-null    int64
dtypes: int64(2), str(2)
memory usage: 1.1 KB
```

EVALUACION DE CALIDAD Y DATOS AUSENTES:

Esta linea de codigo nos ayuda a verificar los datos del archivo y cuenta si hacen falta valores en las filas y columnas

```
In [8]: df.isnull().sum()
```

```
Out[8]: cliente_id    0
        producto      0
        categoria     0
        precio        0
        dtype: int64
```

TRATAMIENTO DE DATOS:

Esta linea tiene como funcionalidad remplazar los valores que hacen falta por el promedio del precio

```
In [ ]: df["precio"]=df["precio"].fillna(df["precio"].mean())
```

Si el resultado no arroja nada es porque no hay valores nulos y no tenemos que hacer tecnica de imputacion. Esto se analiza porque el el ciclo de vida de ML no trabaja con datos ausentes.

NORMALIZACION DE DATOS:

La normalizacion se hace para ajustar las variables en una misma escala. Ejemplo (0 a 1) Es importantante analizarla en este proyecto para la variable precio ya que las otras son variables de texto

La normalización de la variable precio se realizó como parte del ciclo de vida de Machine Learning, con el objetivo de escalar los datos a una misma dimensión y evitar inconsistencias durante el análisis y el modelado.

```
In [11]: df["precio_normalizado"]=(df["precio"]-df["precio"].min()/(df["precio"].max()-df["df.head(30)
```

Out[11]:

	cliente_id	producto	categoria	precio	precio_normalizado
0	1	base	maquillaje	35	0.571429
1	2	corrector	maquillaje	30	0.428571
2	3	labial mate	maquillaje	25	0.285714
3	4	shampoo	cuidado_capilar	50	1.000000
4	5	tonico	cuidado_capilar	45	0.857143
5	6	tratamiento	cuidado_capilar	48	0.942857
6	7	sombras	maquillaje	30	0.428571
7	8	delineador	maquillaje	20	0.142857
8	9	rubor	maquillaje	18	0.085714
9	10	iluminador	maquillaje	15	0.000000
10	11	acondicionador	cuidado_capilar	50	1.000000
11	12	pestañina	maquillaje	20	0.142857
12	13	polvo compacto	maquillaje	15	0.000000
13	14	rubor	maquillaje	18	0.085714
14	15	base	maquillaje	35	0.571429
15	16	shampoo	cuidado_capilar	50	1.000000
16	17	sombras	maquillaje	30	0.428571
17	18	iluminador	maquillaje	15	0.000000
18	19	corrector	maquillaje	30	0.428571
19	20	polvo compacto	maquillaje	25	0.285714
20	21	base	maquillaje	35	0.571429
21	22	rubor	maquillaje	18	0.085714
22	23	shampoo	cuidado_capilar	50	1.000000
23	24	tratamiento	cuidado_capilar	45	0.857143
24	25	delineador	maquillaje	15	0.000000
25	26	pestañina	maquillaje	20	0.142857
26	27	tonico	cuidado_capilar	30	0.428571
27	28	polvo compacto	maquillaje	20	0.142857
28	29	labial mate	maquillaje	15	0.000000
29	30	iluminador	maquillaje	18	0.085714

```
In [12]: df["precio_normalizado"].min(), df["precio_normalizado"].max()
```

```
Out[12]: (np.float64(0.0), np.float64(1.0))
```

Se aplicó una normalización Min-Max a la variable Precio con el fin de escalar sus valores entre 0 y 1. Como resultado del proceso de normalización, se creó una nueva variable denominada precio_normalizado, cuyos valores se encuentran en un rango entre 0 y 1. Este resultado confirma que la normalización se aplicó correctamente y que los datos están preparados para un posible uso en modelos de Machine Learning.

ANALISIS UNIVARIADO.

Se realizó un análisis univariado de la variable Precio, calculando la media, mediana y moda, con el objetivo de comprender la distribución de los precios de los productos de la tienda de maquillaje. Este análisis permite identificar el precio promedio, el valor central y el precio más frecuente, información clave para la toma de decisiones y para las etapas posteriores del ciclo de vida de Machine Learning.

```
In [ ]: df["precio"].mode()[0] # La moda devuelve el valor que más se repite
```

```
Out[ ]: np.int64(15)
```

```
In [ ]: df["precio"].median() # La mediana es el valor central
```

```
Out[ ]: np.float64(27.5)
```

```
In [ ]: df["precio"].mean() # La media es el promedio
```

```
Out[ ]: np.float64(29.0)
```

Mediante la función value_counts() se analizó la distribución de la variable categoría, lo que permitió identificar la frecuencia de cada tipo de producto dentro de la tienda de maquillaje.

```
In [23]: df["categoria"].value_counts()
```

```
Out[23]: categoria
maquillaje      22
cuidado_capilar   8
Name: count, dtype: int64
```

A partir del análisis univariado de la variable categoría, se identificó que la mayoría de los productos pertenecen a la categoría maquillaje, seguida por el cuidado capilar.

ANALISIS BIVARIADO

Se realizó un análisis bivariado entre las variables categoría y precio con el fin de identificar el comportamiento de los precios según el tipo de producto ofrecido en la tienda de maquillaje. Para ello, se agruparon los datos por categoría y se calcularon medidas estadísticas como el promedio, la mediana, el valor mínimo y el valor máximo del precio. Este análisis permite comprender diferencias de precios entre categorías y apoya la etapa de análisis exploratorio dentro del ciclo de vida de un proyecto de Machine Learning.

```
In [42]: df.groupby(["categoria", "producto"])["precio"].agg(["mean", "median", "max", "min"])
```

```
Out[42]:
```

		mean	median	max	min
categoria	producto				
cuidado_capilar	acondicionador	50.0	50.0	50	50
	shampoo	50.0	50.0	50	50
	tonico	45.0	45.0	45	45
	tonico	30.0	30.0	30	30
	tratamiento	46.5	46.5	48	45
maquillaje	base	35.0	35.0	35	35
	corrector	30.0	30.0	30	30
	corrector	30.0	30.0	30	30
	delineador	17.5	17.5	20	15
	iluminador	16.0	15.0	18	15
	labial mate	20.0	20.0	25	15
	pestañina	20.0	20.0	20	20
	pestañina	20.0	20.0	20	20
	polvo compacto	20.0	20.0	25	15
	rubor	18.0	18.0	18	18
	sombras	30.0	30.0	30	30

ANALISIS MULTIVARIADO

Se realizó un análisis multivariado utilizando las variables categoría, producto y precio, con el fin de identificar patrones de precios dentro de cada tipo de producto y categoría en la tienda de maquillaje.

```
In [43]: df.groupby(["categoria", "producto"])["precio"].mean()
```

```

Out[43]: categoria    producto    precio
cuidado_capilar  acondicionador  50.0
                shampoo          50.0
                tonico            45.0
                tonico            30.0
                tratamiento       46.5
maquillaje      base             35.0
                corrector         30.0
                corrector         30.0
                delineador        17.5
                iluminador        16.0
                labial mate       20.0
                pestañina         20.0
                pestañina         20.0
                polvo compacto    20.0
                rubor             18.0
                sombras           30.0
Name: precio, dtype: float64

```

Este análisis permite comprender cómo interactúan múltiples variables simultáneamente y aporta información relevante para la etapa de análisis exploratorio de datos dentro del ciclo de vida de un proyecto de Machine Learning.

```

In [45]: df.groupby(["cliente_id", "producto"])["precio"].mean()

```



```
Out[45]: cliente_id producto      precio
1         base      35.0
2         corrector  30.0
3         labial mate 25.0
4         shampoo    50.0
5         tonico      45.0
6         tratamiento 48.0
7         sombras    30.0
8         delineador  20.0
9         rubor       18.0
10        iluminador  15.0
11        acondicionador 50.0
12        pestañina   20.0
13        polvo compacto 15.0
14        rubor       18.0
15        base      35.0
16        shampoo    50.0
17        sombras    30.0
18        iluminador  15.0
19        corrector  30.0
20        polvo compacto 25.0
21        base      35.0
22        rubor       18.0
23        shampoo    50.0
24        tratamiento 45.0
25        delineador  15.0
26        pestañina   20.0
27        tonico      30.0
28        polvo compacto 20.0
29        labial mate 15.0
30        iluminador  18.0
Name: precio, dtype: float64
```

El análisis multivariado permitió evaluar la relación entre múltiples variables del dataset, específicamente cliente, producto y precio, con el objetivo de identificar comportamientos de compra y variaciones de precios dentro de la tienda de maquillaje.