Actividad 2 | Manejo de datos con Pandas.

> Por: Alexa Andivi Calderón Sánchez.

> Matrícula: A01637520.

Análisis de la base de datos de las flores Iris.

TUTORIAL 1

```
#Importar las librerias.
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
```

```
# Define where you are running the code: colab or local
RunInColab          = True     # (False: no  | True: yes)

# If running in colab:
if RunInColab:
    # Mount your google drive in google colab
    from google.colab import drive
    drive.mount('/content/drive')

    # Find location
    #!pwd
    #!ls
    #!ls "/content/drive/My Drive/Colab Notebooks/MachineLearningWithPython/"

    # Define path del proyecto
    Ruta            = "/content/drive/MyDrive/Colab Notebooks/MachineLearningWithPython/"

else:
    # Define path del proyecto
    Ruta            = ""
```

```
 Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

1. Load the iris.csv file in your computer and understand the dataset

```
#Cargar la base de datos
url = Ruta + "iris.csv"
df = pd.read_csv(url,names=['SepalLength', 'SepalWidth', 'PetalLength','PetalWidth','FlowerType'])
df
```

	SepalLength	SepalWidth	PetalLength	PetalWidth	FlowerType
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

2. How many observations (rows) are in total?

3. How many variables (columns) are in total? What do they represent?

```
#Imprimir el numero de filas
fila = len(df.index)

#Imprimir el número de columnas
columnas = df.columns
Ncolumnas= len(columnas)

print('Número·de·Columnas:',·Ncolumnas)
print('Número de Filas:', fila)
print('Las variables que se tienen son 4 númericas y 1 categorica')
```

```
Número de Columnas: 5
Número de Filas: 150
Las variables que se tienen son 4 númericas y 1 categorica
```

4. How many observations are for each type of flower?

```
#Valores unicos en la variable "FlowerType"
df.groupby(['FlowerType']).size()
```

```
FlowerType
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
dtype: int64
```

5. What is the type of data for each variable?

```
df.dtypes
```

```
SepalLength    float64
SepalWidth     float64
PetalLength    float64
PetalWidth     float64
FlowerType      object
dtype: object
```

6. What are the units of each variable?

> The units of our data set is cm.

TUTORIAL 2

1. Calculate the statistical summary for each quantitative variables. Explain the results
   - Identify the name of each column
   - Identify the type of each column
   - Minimum, maximum, mean, average, median, standar deviation

```
#Nombre de las variables.
tipo = df.dtypes
print('Nuestra base de datos contiene los siguientes datos:')
print(tipo)
```

```
Nuestra base de datos contiene los siguientes datos:
SepalLength    float64
SepalWidth     float64
PetalLength    float64
PetalWidth     float64
FlowerType      object
dtype: object
```

```
#La estadistica de nuestra base de datos es la siguiente.
df.describe()
```

	SepalLength	SepalWidth	PetalLength	PetalWidth
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

2. Are there missing data? If so, create a new dataset containing only the rows with the non-missing data

```
#Buscar el número de datos faltantes por variable.
non = df.notnull().sum()

print('La base de datos tiene un total de', fila, 'observaciones')
print('Nuestras variables presentan la siguiente cantidad observaciones:')
print(non)
print('Por lo que se muestra que se tienen observaciones completas')
```

```
La base de datos tiene un total de 150 observaciones
Nuestras variables presentan la siguiente cantidad observaciones:
SepalLength    150
SepalWidth     150
PetalLength    150
PetalWidth     150
FlowerType     150
dtype: int64
Por lo que se muestra que se tienen observaciones completas
```

3. Create a new dataset containing only the petal width and length and the type of Flower

```
#Seleccionar PetalLength and PetalWidth en una nueva base de datos.
petal = df.loc[:, ["PetalWidth", "PetalLength",]]
petal.head()
```

	PetalWidth	PetalLength
0	0.2	1.4
1	0.2	1.4
2	0.2	1.3
3	0.2	1.5
4	0.2	1.4

4. Create a new dataset containing only the setal width and length and the type of Flower

```
sepal = df.loc[:, ["SepalWidth", "SepalLength",]]
sepal.head()
```

	SepalWidth	SepalLength
0	3.5	5.1
1	3.0	4.9
2	3.2	4.7
3	3.1	4.6
4	3.6	5.0

5. Create a new dataset containing the setal width and length and the type of Flower encoded as a categorical numerical column

```
#Creación de una nueva base de datos con 3 variables diferentes.
sepalFlower = df.loc[:, ["SepalWidth", "SepalLength","FlowerType"]]
sepalFlower.head()
```

	SepalWidth	SepalLength	FlowerType
0	3.5	5.1	Iris-setosa
1	3.0	4.9	Iris-setosa
2	3.2	4.7	Iris-setosa
3	3.1	4.6	Iris-setosa
4	3.6	5.0	Iris-setosa

```
sepalFlower1 = sepalFlower.replace({'Iris-setosa':1,'Iris-versicolor':2,'Iris-virginica':3})
sepalFlower1.head()
```

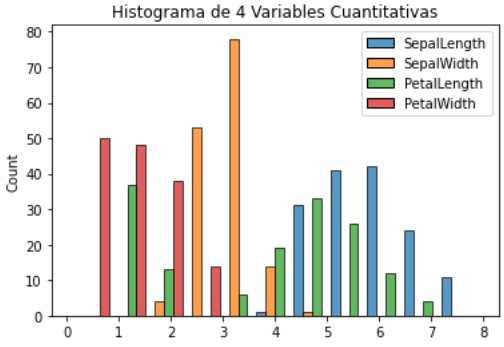
	SepalWidth	SepalLength	FlowerType
0	3.5	5.1	1
1	3.0	4.9	1
2	3.2	4.7	1
3	3.1	4.6	1
4	3.6	5.0	1

TUTORIAL 3

1. Plot the histograms for each of the four quantitative variables

```
p = sns.histplot(data = df,multiple="dodge")
p.set_title("Histograma de 4 Variables Cuantitativas")
```

```
Text(0.5, 1.0, 'Histograma de 4 Variables Cuantitativas')
```



+ Código + Texto

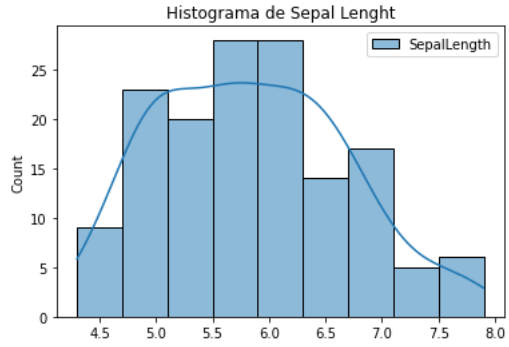
2. Plot the histograms for each of the quantitative variables

```
Sw = df.loc[:, ["SepalWidth"]]
p1 = sns.histplot(data=Sw,kde= True)
p1.set_title("Histograma de Sepal Width")
```

```
Text(0.5, 1.0, 'Histograma de Sepal Width')
```

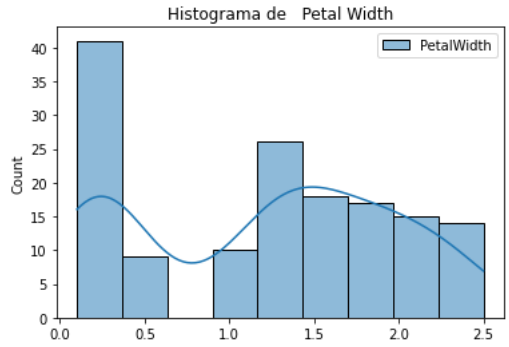

```
Sl = df.loc[:, ["SepalLength"]]
p2 = sns.histplot(data = Sl,kde= True)
p2.set_title("Histograma de Sepal Lenght")
```

```
Text(0.5, 1.0, 'Histograma de Sepal Lenght')
```



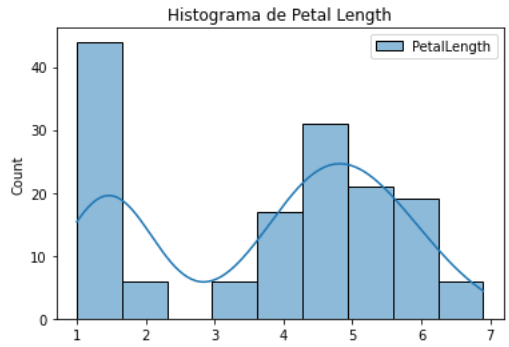

```
Pw = df.loc[:, ["PetalWidth"]]
p3 = sns.histplot(data = Pw,kde= True)
p3.set_title("Histograma de   Petal Width")
```

```
Text(0.5, 1.0, 'Histograma de   Petal Width')
```




```
Pl = df.loc[:, ["PetalLength"]]
p4 = sns.histplot(data = Pl,kde= True)
p4.set_title("Histograma de Petal Length")
```

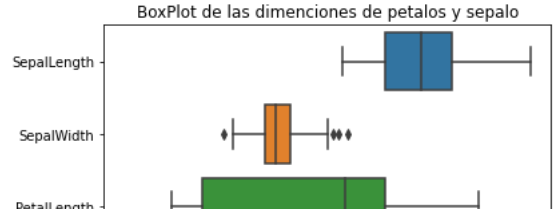
```
Text(0.5, 1.0, 'Histograma de Petal Length')
```



3. Plot the boxplots for each of the quantitative variables

```
Pb= sns.boxplot(data=df, orient = 'h')
Pb.set_title("BoxPlot de las dimenciones de petalos y sepalo")
```

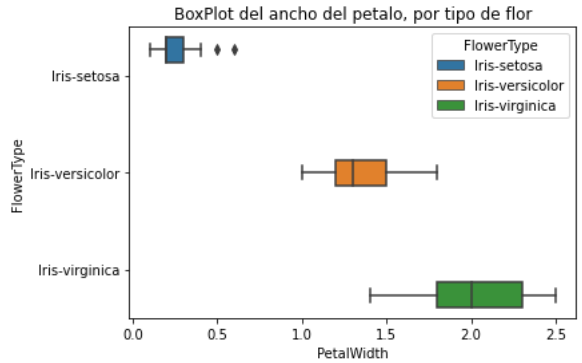
```
Text(0.5, 1.0, 'BoxPlot de las dimenciones de petalos y sepalo')
```



4. Plot the boxplots of the petal width grouped by type of flower

```
pw1= df.loc[:, ["PetalWidth","PetalLength","FlowerType"]]
pwb= sns.boxplot(data= pw1, x='PetalWidth',y='FlowerType',hue='FlowerType',orient='h')
pwb.set_title("BoxPlot del ancho del petalo, por tipo de flor")
```

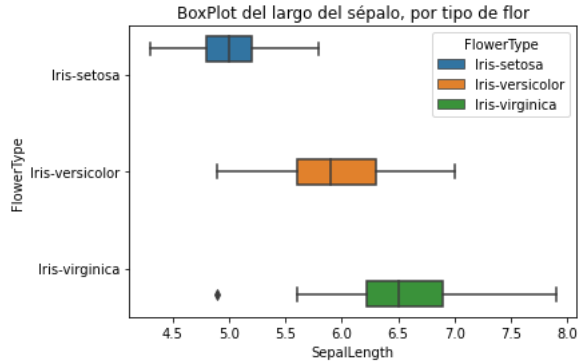
```
Text(0.5, 1.0, 'BoxPlot del ancho del petalo, por tipo de flor')
```



5. Plot the boxplots of the setal length grouped by type of flower

```
pl1= df.loc[:, ["SepalLength","FlowerType"]]
plb= sns.boxplot(data= pl1, x='SepalLength',y='FlowerType',hue='FlowerType',orient='h')
plb.set_title("BoxPlot del largo del sépalo, por tipo de flor")
```

```
Text(0.5, 1.0, 'BoxPlot del largo del sépalo, por tipo de flor')
```



6. Provide a description (explaination from your observations) of each of the quantitative variable.

La gráfica de bloxplot del ancho de los petalos muestra que los petalos de la planta de catalogada como Iris-Setosa son pequeños, aparte de que sus sepálos muestra que su rango de valores es el más pequeño del conjunto de datos; por lo tanto podemos concluir que ese tipo de variable de Iris es la más pequeña en comparación de las tres.

En el BloxPlot de las dimenciones muestra que la varibale con más rango de datos es el largo del petalo, mientras que el ancho de los pétalos son los valores más péqueños, por otro lado su histograma muestra una distribución más uniforme.

Las variables relacionadas con el sépalo presnetan una distribución normal.

Tambien podemos determinar que los tipos de iris catalogadas como versicolor y virginica tienen dimensiones más parecidas; sus boxplot tienden a sobreponerse, mostrando que puede ser más dificil difernciarlas en base a sus dimensiones.

✓ 0 s se ejecutó 22:02