

Creación de DataFarme

```
In [3]: import numpy as np
import pandas as pd
iris = pd.read_csv('Iris.csv')
iris
```

```
Out[3]:
```

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
1	4.9	3.0	1.4	0.2	Setosa
2	4.7	3.2	1.3	0.2	Setosa
3	4.6	3.1	1.5	0.2	Setosa
4	5.0	3.6	1.4	0.2	Setosa
...
145	6.7	3.0	5.2	2.3	Virginica
146	6.3	2.5	5.0	1.9	Virginica
147	6.5	3.0	5.2	2.0	Virginica
148	6.2	3.4	5.4	2.3	Virginica
149	5.9	3.0	5.1	1.8	Virginica

150 rows x 5 columns

1 Determine en una sola tabla la media, desviación estándar y la mediana (percentil 50) para el ancho del pétalo, por cada tipo de flor para aquellos registros cuyo ancho del sépalo es mayor o igual a 3.5 y el largo del sépalo es mayor que 5.

```
filtro_sepal = iris[(iris['sepal.length'] > 5) & (iris['sepal.width'] >= 3.5)]
filtro_sepal
```

	sepal.length	sepal.width	petal.length	petal.width	variety
0	5.1	3.5	1.4	0.2	Setosa
5	5.4	3.9	1.7	0.4	Setosa
10	5.4	3.7	1.5	0.2	Setosa
14	5.8	4.0	1.2	0.2	Setosa
15	5.7	4.4	1.5	0.4	Setosa
16	5.4	3.9	1.3	0.4	Setosa
17	5.1	3.5	1.4	0.3	Setosa
18	5.7	3.8	1.7	0.3	Setosa
19	5.1	3.8	1.5	0.3	Setosa
21	5.1	3.7	1.5	0.4	Setosa
27	5.2	3.5	1.5	0.2	Setosa
32	5.2	4.1	1.5	0.1	Setosa
33	5.5	4.2	1.4	0.2	Setosa
36	5.5	3.5	1.3	0.2	Setosa

Determinamos en una sola tabla la media, desviación estándar y la mediana

```
def Media(columna):  
    return columna.mean()  
  
def DesvEst(columna):  
    return columna.std()  
  
def Mediana(columna):  
    return columna.median()  
  
iris_tipo = filtro_sepal.groupby('variety')['petal.width'].agg([Media, DesvEst, Mediana])  
iris_tipo
```

	Media	DesvEst	Mediana
variety			
Setosa	0.270588	0.098518	0.2
Virginica	2.233333	0.251661	2.2

```
# Calcule el número de registros que se utilizaron para obtener los resultados del problema 1.  
len(filtro_sepal)
```

20

Agregue una nueva columna al DataFrame generado al problema 1, que sea resultado de estandarizar los valores del ancho del pétalo para cada registro.

Denomine a dicha columna “petal.width.st” y calcúlela de la siguiente manera:

$$\text{petal.width} - \text{Media}(\text{petal.width}) / \text{Desv. Estándar}(\text{petal.width})$$

```
def Media(columna):  
    return columna.mean()  
  
def DesvEst(columna):  
    return columna.std()  
  
estandar_petal = filtro_sepal['petal.width'] - filtro_sepal['petal.width'].agg(Media) / filtro_sepal['petal.width'].agg(DesvEst)
```

Agregamos la nueva columna al DF

```
filtro_sepal['petal.width.st'] = estandar_petal  
filtro_sepal
```

C:\Users\Isaac\AppData\Local\Temp\ipykernel_16364\781291095.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/1d_internals.html

```
filtro_sepal['petal.width.st'] = estandar_petal
```

:

	sepal.length	sepal.width	petal.length	petal.width	variety	petal.width.st
0	5.1	3.5	1.4	0.2	Setosa	-0.574723
5	5.4	3.9	1.7	0.4	Setosa	-0.374723
10	5.4	3.7	1.5	0.2	Setosa	-0.574723
14	5.8	4.0	1.2	0.2	Setosa	-0.574723
15	5.7	4.4	1.5	0.4	Setosa	-0.374723
16	5.4	3.9	1.3	0.4	Setosa	-0.374723
17	5.1	3.5	1.4	0.3	Setosa	-0.474723
18	5.7	3.8	1.7	0.3	Setosa	-0.474723
19	5.1	3.8	1.5	0.3	Setosa	-0.474723

Renombramos las columnas 'variety' y 'petal.width.st'

```
filtro_sepal.rename(columns = {'variety': 'variedad', 'petal.width.st': 'Ancho Est. Pétalo'})  
filtro_sepal
```

:

	sepal.length	sepal.width	petal.length	petal.width	variedad	Ancho Est. Pétalo
0	5.1	3.5	1.4	0.2	Setosa	-0.574723
5	5.4	3.9	1.7	0.4	Setosa	-0.374723
10	5.4	3.7	1.5	0.2	Setosa	-0.574723
14	5.8	4.0	1.2	0.2	Setosa	-0.574723
15	5.7	4.4	1.5	0.4	Setosa	-0.374723
16	5.4	3.9	1.3	0.4	Setosa	-0.374723
17	5.1	3.5	1.4	0.3	Setosa	-0.474723
18	5.7	3.8	1.7	0.3	Setosa	-0.474723
19	5.1	3.8	1.5	0.3	Setosa	-0.474723
21	5.1	3.7	1.5	0.4	Setosa	-0.374723
27	5.2	3.5	1.5	0.2	Setosa	-0.574723
32	5.2	4.1	1.5	0.1	Setosa	-0.674723
33	5.5	4.2	1.4	0.2	Setosa	-0.574723
36	5.5	3.5	1.4	0.2	Setosa	-0.574723

```
: # .Exporte el resultado del DataFrame obtenido en el problema 4 a un archivo denominado "Iris2.csv".  
filtro_sepal.to_csv('Iris2.csv')
```

```
# Investigue cómo generar un gráfico de dispersión para la base original Iris  
# que relacione la variable "petal.width" con "petal.length". Sugerencia: Revisar la instrucción .plot(x,y,kind)  
  
import numpy as np  
import matplotlib.pyplot as plt  
  
x = [iris['petal.width']]  
y = [iris['petal.length']]  
  
plt.scatter(x,y)  
plt.xlabel('Ancho del Petalo')  
plt.ylabel('Largo del Petalo')  
plt.title('Grafico de Dispersion DataFrame Iris')  
plt.show()
```

