

Ejercicio

Guillermo Ruiz

```
import pandas as pd
import matplotlib.pyplot as plt
```

El conjunto de datos **heart** incluye 13 características y una variable objetivo (target). Las características son:

1. age
2. sex
3. chest pain type (4 values)
4. resting blood pressure
5. serum cholestoral in mg/dl
6. fasting blood sugar > 120 mg/dl
7. resting electrocardiographic results (values 0,1,2)
8. maximum heart rate achieved
9. exercise induced angina
10. oldpeak = ST depression induced by exercise relative to rest
11. the slope of the peak exercise ST segment
12. number of major vessels (0-3) colored by flourosopy
13. thal: 0 = normal; 1 = fixed defect; 2 = reversable defect
14. **target**: 0 = no disease; 1 = disease

Para este ejercicios se debe hacer los siguiente:

1. Leer el conjunto de datos con Pandas.
2. Mostrar los registros de personas con más de 70 años.
3. Crear dos DataFrames, uno para las personas enfermas y otro para las sanas.
4. Mostrar el histograma de edad para las personas enfermas y otro para las sanas.
5. Graficar los valores de las variables **age** contra **trestbps**

1. Leer el conjunto de datos con Pandas.

```
url = "https://drive.google.com/uc?id=1E7wYL1bJgVs9UUTsbNhHKtEF3zp8vzRj"
heart = pd.read_csv(url)
heart.head(3)
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1

2. Mostrar los registros de personas con más de 70 años.

```
heart[heart['age']>70]
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
25	71	0	1	160	302	0	1	162	0	0.4	2	2	2	1
60	71	0	2	110	265	1	0	130	0	0.0	2	1	2	1
129	74	0	1	120	269	0	0	121	1	0.2	2	1	2	1
144	76	0	2	140	197	0	2	116	0	1.1	1	0	2	1
151	71	0	0	112	149	0	1	125	0	1.6	1	0	2	1
238	77	1	0	125	304	0	0	162	1	0.0	2	3	2	0

3. Crear dos DataFrames, uno para las personas enfermas y otro para las sanas.

```
heart_sanas = heart[heart['target'] == 0]
heart_enfermas = heart[heart['target'] == 1]
```

```
heart_sanas['age'].describe()
```

```
count    138.000000
mean      56.601449
std       7.962082
min       35.000000
```

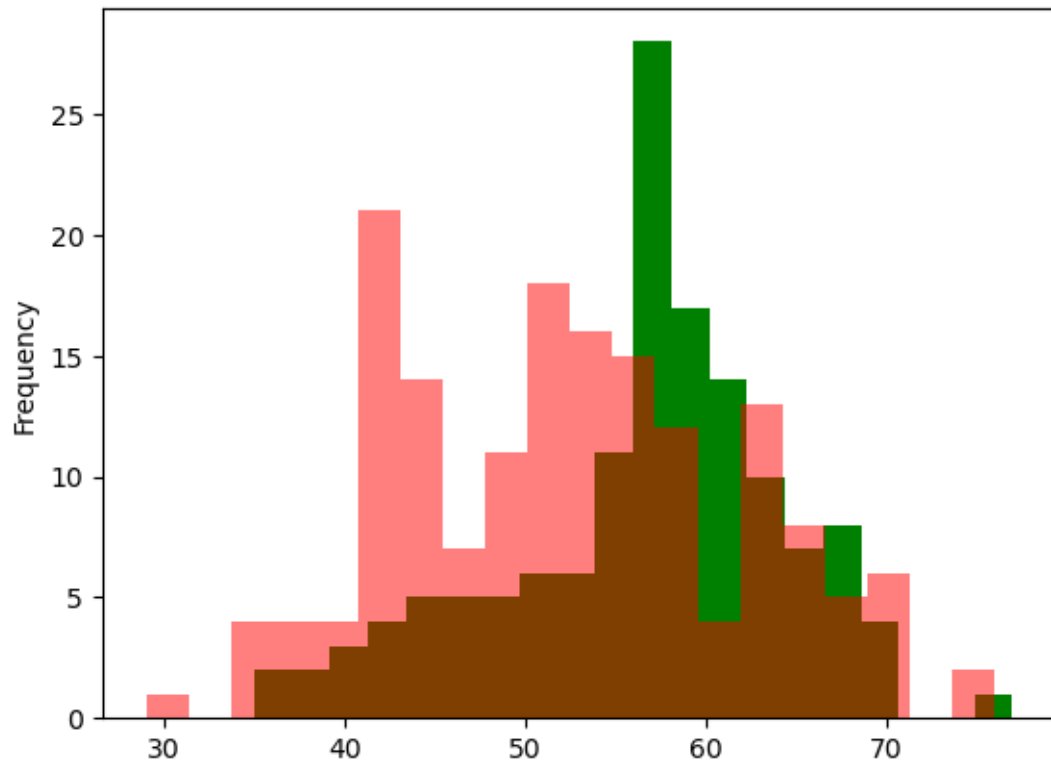
```
25%      52.000000
50%      58.000000
75%      62.000000
max       77.000000
Name: age, dtype: float64
```

```
heart_enfermas['age'].describe()
```

```
count      165.000000
mean       52.496970
std         9.550651
min        29.000000
25%        44.000000
50%        52.000000
75%        59.000000
max        76.000000
Name: age, dtype: float64
```

4. Mostrar el histograma de edad para las personas enfermas y otro para las sanas.

```
heart_sanas['age'].plot.hist(bins=20, color='g')
heart_enfermas['age'].plot.hist(bins=20, alpha=.5, color='r')
plt.show()
```



5. Graficar los valores de las variables age contra trestbps

```
heart.plot.scatter(x='age', y='trestbps')  
plt.title('Age vs presión arterial')  
plt.show()
```

