

# Datos estadísticos

November 20, 2019

## 1 Medidas básicas de la estadística descriptiva

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

## 2 Medidas de centralización

Nos sirve para ver cómo se sitúan los datos. Son la **media**, **mediana**, **percentiles** y **moda**.

Vamos a generar una lista de números

```
[2]: # Generar números aleatorios
import random
random.seed(0)
A= (random.sample(range(100000), 60))
A
```

```
[2]: [50494,
99346,
55125,
5306,
33936,
67013,
63691,
53075,
39755,
62468,
46930,
76465,
28631,
66150,
18254,
36941,
18316,
99064,
12429,
81050,
```

32834,  
69804,  
92428,  
78892,  
19262,  
40651,  
12945,  
95660,  
9665,  
89651,  
43279,  
61884,  
73375,  
13199,  
46372,  
56907,  
41444,  
80070,  
83941,  
26801,  
72420,  
62522,  
58024,  
68334,  
34143,  
8163,  
71919,  
1840,  
12225,  
94333,  
52274,  
93094,  
87576,  
81954,  
149,  
80202,  
64694,  
43664,  
31969,  
95719]

```
[3]: mean=np.mean(A)#Cálculo media  
mean
```

```
[3]: 53312.01666666667
```

```
[4]: np.median(A)# Cálculo mediana
```

```
[4]: 56016.0
```

```
[5]: from scipy import stats # Cálculo moda
stats.mode(A)
```

```
[5]: ModeResult(mode=array([149]), count=array([1]))
```

```
[6]: # percencil 100=p1 0=p0, 0.25 1er cuantil 0.75 3er cuantil
np.percentile(A, 25) # return 0.25 percentile, e.g 1er cuantil.
```

```
[6]: 32617.75
```

### 3 Medidas de dispersión

La varianza y desviación típica, nos indica si los valores se desplazan mucho o poco con respecto de la media.

La **varianza** es como se aleja cada valor de la media. La varianza eleva los valores al cuadrado... nos introduce en una nueva dimensión... puede no tener sentido.

La **desviación típica** es la raíz cuadrada de la varianza. Con la desviación típica volvemos a la dimensión original.

**Coefficiente de variación:** nos mide la variabilidad relativa entre la desviación típica entre la media.

```
[7]: np.var(A) #varianza
```

```
[7]: 810711216.7830557
```

```
[8]: std=np.std(A) #desviación típica
std
```

```
[8]: 28472.991005215023
```

```
[9]: # coeficiente de variacion std/mean*100
# variabilidad relativa entre la media y la std, si hay mucha variabilidad será
    ↪ grande el coeficiente.
std/mean*100
```

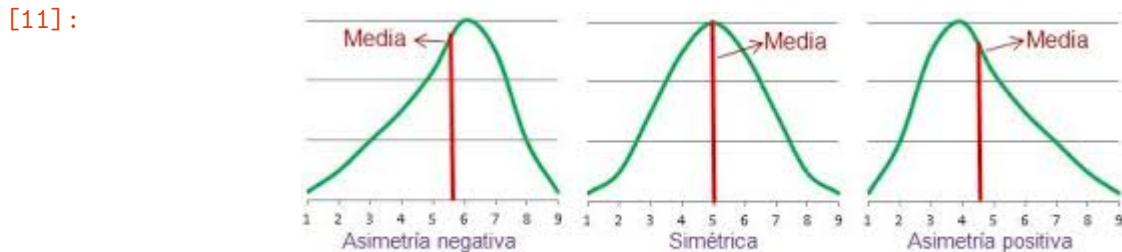
```
[9]: 53.40820472660482
```

### 4 Medidas de asimetría

Momento de orden r, respecto a la media El momento de orden r. son los momentos de distribución respecto a la media.

```
[10]: import scipy.stats as sp # para modelos de asimetria
      from IPython.display import Image # para incluir imagenes local
```

```
[11]: # Asimetría de Fisher
      Image(filename="asimetria/asimetria.png")
```



### Asimetría de Fisher

- Si el coeficiente es  $= 0$ ; Significa que vuestra función es perfectamente simétrica, se distribuye igual, por ejemplo la distribución normal. Raro es que salga cero
- Si el coeficiente es  $>0$ ; Significa que cuánto más positivo es este valor más desplazada está la distribución hacia la izquierda, de modo que tenemos una asimetría positiva, nos queda la media muy por encima de la distribución.
- Si el coeficiente es  $<0$ ; Significa que cuánto más negativo es este valor más desplazado está la distribución hacia la derecha, de modo que tenemos una asimetría negativa, nos queda la media muy por debajo de la distribución.

```
[12]: # 3rd Moment: Asimetría de Fisher
      Asimetria=sp.skew(A)
      Asimetria
```

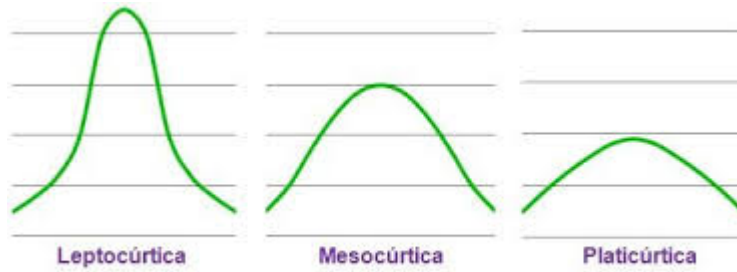
[12]: -0.17284864155126514

### Curtosis

- $=0$  Mesocúrtica Distribución perfecta, asemejada a la distribución normal en forma, no en valores. Está compensado tanto el centro como las colas.
- $>0$  Leptocúrtica Distribución donde se le concentran mucho los datos en el valor central, y apenas tiene cola.
- $<0$  Platicúrtica Distribución donde hay pocos valores que se concentren respecto al valor central (media) y hay muchos que aparecen hacia las colas, se concentran más en los laterales. Existe valor central, pero también hay mucha presencia de colas directamente en la distribución de nuestros datos.

```
[13]: # Kurtosis
Image(filename="asimetria/kurtosis.png")
```

[13]:



```
[14]: # 4th Moment: Kurtosis
curtosis=sp.kurtosis(A)
curtosis
```

[14]: -1.0800693127939995

## Histogramas

```
[15]: import matplotlib.pyplot as plt
%matplotlib inline
```

```
[16]: plt.hist(A)
```

```
[16]: (array([5., 7., 2., 6., 6., 6., 8., 6., 6., 8.]),
      array([ 149. , 10068.7, 19988.4, 29908.1, 39827.8, 49747.5, 59667.2,
              69586.9, 79506.6, 89426.3, 99346. ]),
      <a list of 10 Patch objects>)
```

