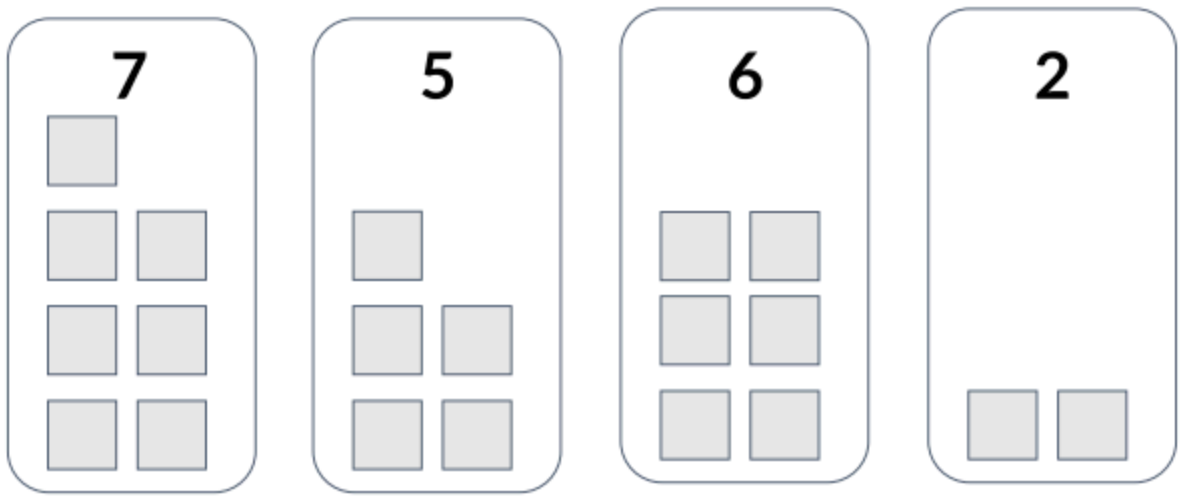
Estadística descriptiva aplicada: medidas de tendencia central.

En esta sesión veremos medidas que nos ayudaran a entender el comportamiento de los datos.
Tenemos 3 medidas de tendencia central principales:

1. Media (promedio).
2. Mediana (dato central).
3. Moda (dato que más se repite).

Comenzando con una serie de bloques, cada uno tiene minibloques dentro de ellos de color gris. Como se puede ver la cantidad dentro de ellos es diferente, vamos a formular una pregunta:

> **¿Cómo podrías distribuir equitativamente los cuadros dentro de cada bloque?**

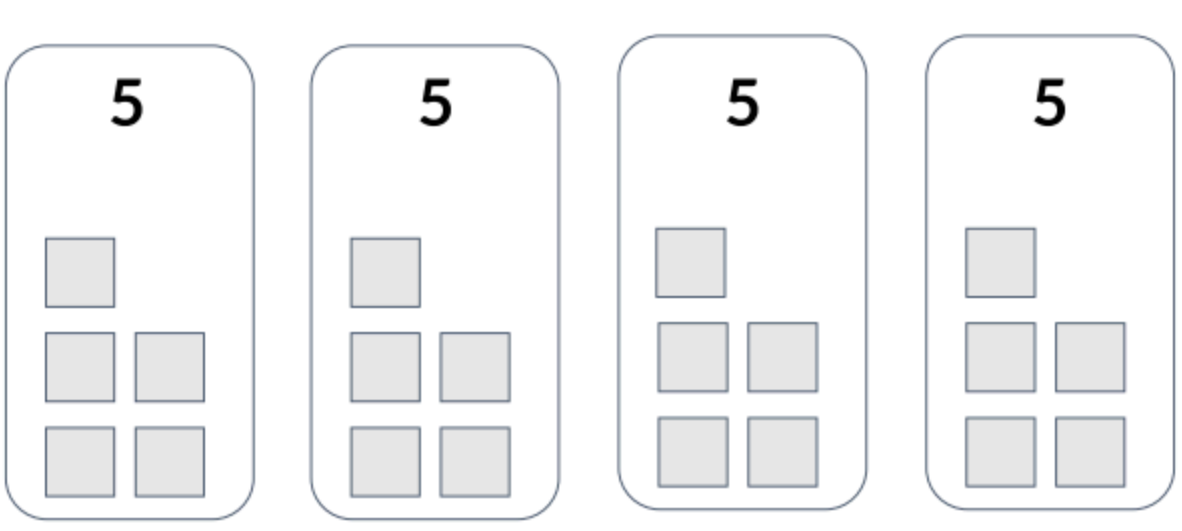


Mi respuesta: Con el valor promedio, es decir sumar el numero de bloques entre el numero de cajas. Así de esta manera tendrá la misma cantidad.

Respuesta del profesor: Es el promedio, y representa o simboliza la cantidad en la cual podemos distribuir los bloques grises de tal manera que los contenedores tengan la misma cantidad. En este caso diríamos que el promedio es 5:

$$\text{Promedio } [\mu] = \frac{\text{Número de cajas}}{\text{Número de contenedores}} = \frac{7+5+6+2}{4} = 5$$

> **Media (promedio)**



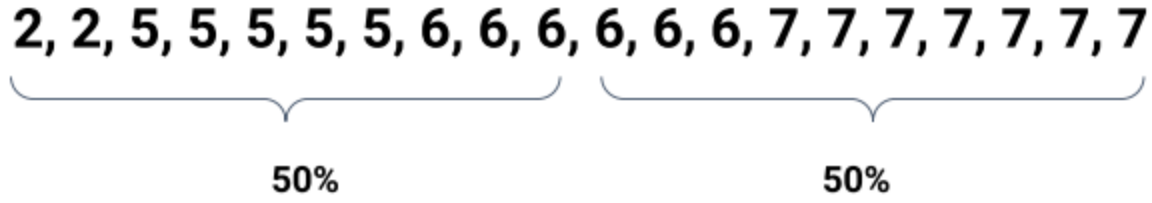
¿Cual es el valor que divide los datos a la mitad?

En este caso es el 50% abajo y 50% arriba.
Mi respuesta: Es la mediana.

Hay que representarlos y contar

2, 2, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6, 6, 7, 7, 7, 7, 7, 7, 7

¿Cuál es el valor que divide a los datos?



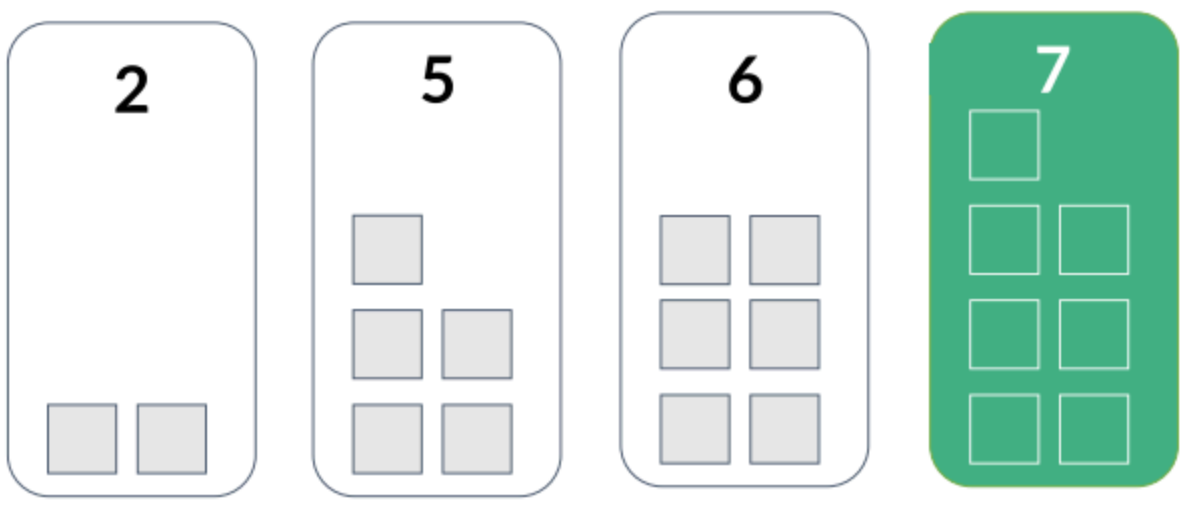
Nos damos cuenta que el valor es 6. En este caso tenemos una cantidad de **números par**, entonces la representación es el promedio de esos números.

$$\text{Mediana } = \frac{6+6}{2} = 6$$

En el caso que la cantidad de números sea impar, el valor que quede justamente en el centro, es la **mediana**. Este será el número que dividirá los bloques inferior y superior.

Ahora hagamos otra pregunta.

¿Cuál es el valor que más se repite en los datos?



El valor que se repite más en la imagen es el número 7 y con esto, entendemos el concepto de **moda**.

En esta clase repasamos los estadísticos más usados, sin embargo no son los únicos, existen otros tipos de estadísticos.

- Media (promedio).
- Mediana (dato central).
- Moda (dato que más se repite).
- Media ponderada.
- Media armónica.
- Media geométrica.

In [ ]:
```
# Importando librerías
import empiricaldist
import janitor
import matplotlib.pyplot as plt
import numpy as np
import palmerpenguins
import pandas as pd
import scipy.stats
import seaborn as sns
import sklearn.metrics
import statsmodels.api as sm
import statsmodels.formula.api as smf
import statsmodels.stats as ss
import session_info
```

Establecer apariencia general de las gráficas

In [ ]:
```
%matplotlib inline
sns.set_style(style='whitegrid')
sns.set_context(context='notebook')
plt.rcParams['figure.figsize'] = (11, 9.4)

penguin_color = {
    'Adelie': '#ff6602ff',
    'Gentoo': '#0f7175ff',
    'Chinstrap': '#c65dc9ff'
}
```

Cargar los datos

Datos crudos

In [ ]:
```
raw_penguins_df = pd.read_csv('dataset/penguins_raw.csv')
```

Datos Preprocesados

NOTA: Puede que no usemos estos datos

In [ ]:
```
preprocessed_penguins_df = pd.read_csv('dataset/penguins.csv')
```

Medidas de tendencia central

Media $[\mu]$

In [ ]:
```
    preprocessed_penguins_df
    .bill_depth_mm.mean()
```

```
  Cell In[7], line 2
    .bill_depth_mm.mean()
    ^
SyntaxError: invalid syntax
```

El error anterior se debe a que la sintaxis no es correcta, esto es debido a que necesitamos colocar paréntesis, para que reconozca de manera correcta la estructura multi-linea.

```
(
    preprocessed_penguins_df
    .bill_depth_mm.mean()
)
```

In [ ]:
```
(
    preprocessed_penguins_df
    .bill_depth_mm.mean()
)
```

Out[ ]:
```
17.151169590643274
```

OJO: este numero puede existir o no dentro del conjunto de datos, dependiendo como se manejen los datos.

El significado del resultado anterior es:

> La altura promedio de los picos de los pingüinos de todos los que tenemos es de: **17 mm**. Con esto ya tenemos un comportamiento general que explica la altura de los picos de los pingüinos.

El calculo anterior lo hicimos a traves de la **API** de Pandas. Pero podemos hacerlo también a través de NumPy.

In [ ]:
```
np.mean(preprocessed_penguins_df.bill_depth_mm)
```

Out[ ]:
```
17.151169590643274
```

Fijémonos que obtenemos el mismo resultado, esto es debido a que **Pandas** está construido sobre **Numpy**, así que todo esto está perfecto.

Lo anterior es solo para una variable, pero debido a que tenemos muchas, también podemos realizar la operación en conjunto de datos.

De la siguiente manera:

In [ ]:
```
preprocessed_penguins_df.mean()
```

```
/tmp/ipykernel_83006/1101382982.py:1: FutureWarning: The default value of numeric_only in DataFrame.mean is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
  preprocessed_penguins_df.mean()
```

Out[ ]:
```
bill_length_mm        43.921930
bill_depth_mm         17.151170
flipper_length_mm    200.915205
body_mass_g         4201.754386
year                2008.029070
dtype: float64
```

Ahora si podemos ver que:

- La altura promedio de pico es de: 17.15 mm
- La longitud de pico es de: 43.92 mm
- El peso promedio de los pingüinos en gr es de: 4201.75 gr ≈ 4Kg.

Ahora que ya sabemos como calcular los estadísticos, podemos hacerlo de manera similar con los demás.

Mediana

In [ ]:
```
preprocessed_penguins_df.median()
```

```
/tmp/ipykernel_83006/1600558650.py:1: FutureWarning: The default value of numeric_only in DataFrame.median is deprecated. In a future version, it will default to False. In addition, specifying 'numeric_only=None' is deprecated. Select only valid columns or specify the value of numeric_only to silence this warning.
  preprocessed_penguins_df.median()
```

Out[ ]:
```
bill_length_mm        44.45
bill_depth_mm         17.30
flipper_length_mm    197.00
body_mass_g         4050.00
year                2008.00
dtype: float64
```

Esto nos dice que tenemos aproximadamente un 50% de nuestros datos (pingüinos) que tiene un:

- 44.45 mm de longitud de pico.
- 17.30 mm de altura de pico.

Es decir 50% esta en estos valores y el otro 50% está por encima de esos valores.

Moda

In [ ]:
```
preprocessed_penguins_df.mode()
```

Out[ ]:

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex	year
0	Adelie	Biscoe	41.1	17.0	190.0	3800.0	male	2009

Aquí hay algo muy interesante, porque ya se incluyen variables categóricas, esto nos dice que variables se repiten más.

Entonces de este conjunto de datos sabemos que los valores de las categorías que más se repiten son:

- Especie: Adelie.
- Isla: Biscoe.
- Longitud de pico: 41.1 mm.
- Altura de pico: 17.0 mm.
- Peso: 3800.0 gr.
- Sexo: Macho (Male).
- Año: 2009.

Generalmente la moda se verá aplicada mucho en variables categóricas, podríamos estar interesados en ver las variables categóricas.

General

Otra forma de ver el conjunto de datos es mediante la función `describe()`. Está función me arrojará un poco más de información sobre mis variables categóricas.

In [ ]:
```
preprocessed_penguins_df.describe(include=object)
```

Out[ ]:

	species	island	sex
count	344	344	333
unique	3	3	2
top	Adelie	Biscoe	male
freq	152	168	168

Fijémonos que tenemos una columna por cada variable. Podemos ver el conteo de observaciones, cuantos valores **únicos** existen para cada variable categórica, la moda o **top** y su número de frecuencia. Es decir:

- 152 de la especie Adelie de 344.
- 168 de la isla Biscoe de 344.
- 168 del sexo Macho de 333.

Así podemos darnos una idea sobre las medidas de tendencia central en mis variables categóricas.