

# Módulo: Visualización de Datos

DATA  
ANALÍTICA





# Reglas del Juego

**Mantener el micrófono  
apagado en caso de que  
no vayamos a hablar.**



**Nos encantaría verte.  
Ten tu cámara encendida y  
conozcámonos  
virtualmente.**



**Preguntar en caso que  
tengan dudas.**



**Disfruta de este espacio.  
Desconecta del resto y  
participa.**



**Por cada clase tendremos  
10 min o 15 min de receso.**



# Agenda

1. Introducción a la visualización de datos
2. Cómo usar las herramientas básicas de visualización, que incluyen gráficos de líneas, gráficos de barras e histogramas
3. Cómo usar herramientas de visualización especializadas, incluidos gráficos de caja, gráficos de dispersión, gráficos circulares y gráficos de burbujas
4. Gráficas con Matplotlib
5. Gráficas usando Seaborn

# Visualización de datos

“La **representación gráfica de datos o conceptos**, que tiene como resultado una imagen mental o un artefacto externo **que ayude a la toma de decisiones**”

**Colin Ware**

“La representación visual de información diseñada para permitir la comunicación, el análisis, el descubrimiento y la exploración”

**Alberto Cairo**

# Visualización de datos

La visualización no es solo el último paso en análisis de datos, en ocasiones es el comienzo.

El valor principal de la visualización está en representar los insights que se encuentran en los datos. Para ello hay que trabajar en 3 pasos:

1. Organizar los datos (Data Wrangling o Data Cleaning)
2. Desarrollar el gráfico
3. Estilizarlo (darle el aspecto visual que queremos)

# Visualización de datos

Matplotlib es una biblioteca basada en NumPy que permite crear gráficos a partir de datos contenidos en listas o arrays.

Tiene una API, `matplotlib.pyplot`, que consiste en una colección de funciones similares a las empleadas en MATLAB.

Cada función en `pyplot` realiza una acción: crear la figura, crea un área de dibujo, traza líneas en el área, añade etiquetas...

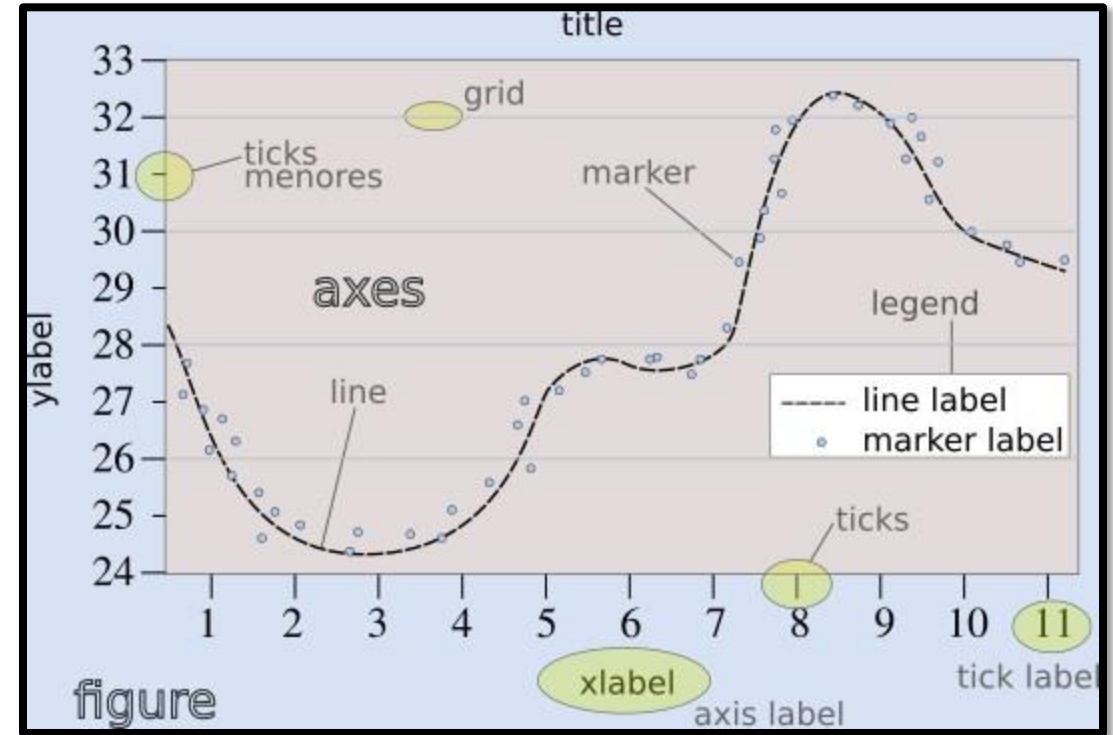
Para crear un gráfico con matplotlib es habitual seguir los siguientes pasos:

1. Importar el módulo pyplot.
2. Definir la figura que contendrá el gráfico, que es la region (ventana o página) donde se dibujará y los ejes sobre los que se dibujarán los datos. Para ello se utiliza la función `subplots()`.
3. Dibujar los datos sobre los ejes. Para ello se utilizan distintas funciones dependiendo del tipo de gráfico que se quiera.
4. Personalizar el gráfico. Para ello existen multitud de funciones que permiten añadir un título, una leyenda, una rejilla, cambiar colores o personalizar los ejes.
5. Guardar el gráfico. Para ello se utiliza la función `savefig()`.
6. Mostrar el gráfico. Para ello se utiliza la función `show()`



# Interfaz Pyplot

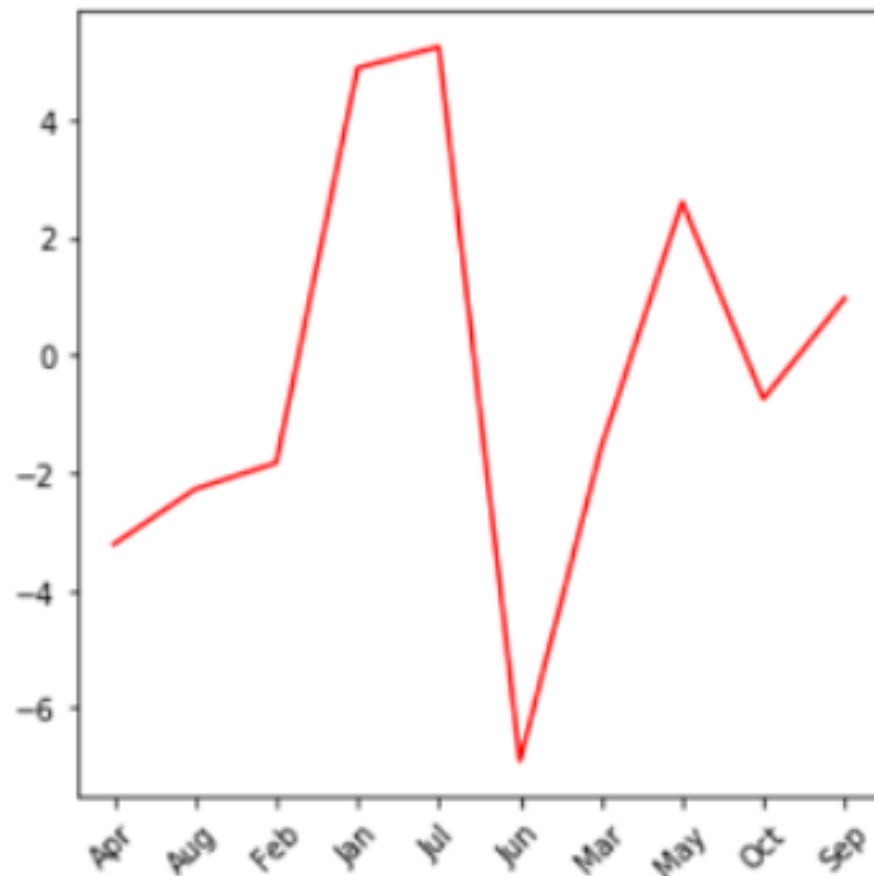
- Permite llevar a cabo la función plot: listas, arrays.
- Personalizar las líneas, colores, leyendas, etc.
- Ofrece un conjunto de tipos de graficas: hist(), scatter(), boxplot(), etc.
- Identificar los componentes:
  - Datos: Listas, ndarray, Pandas Series
  - Elementos: Titulos, nombre de los ejes, etc. Se accede a los elementos básicos con **.plt**



# Gráfico de Lineas

Nos permite proyectar la tendencia general de un conjunto de datos que forma una serie temporal.

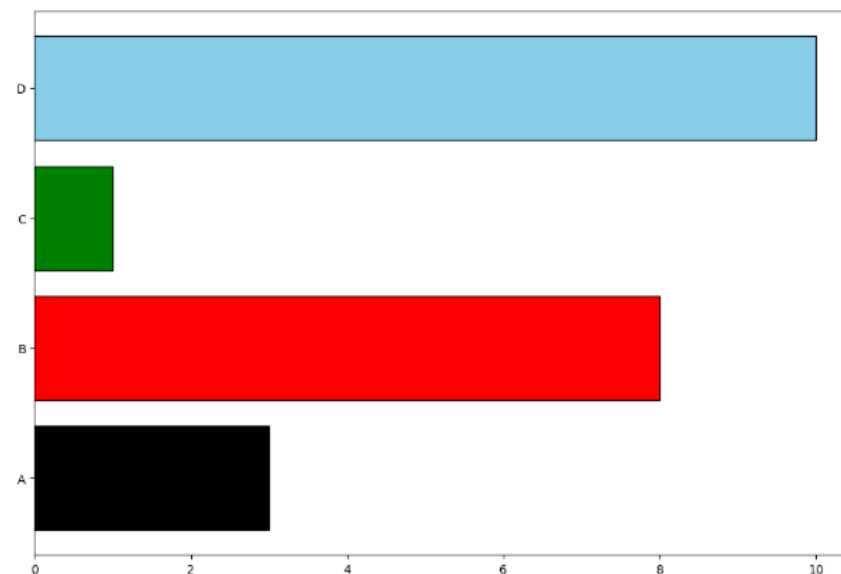
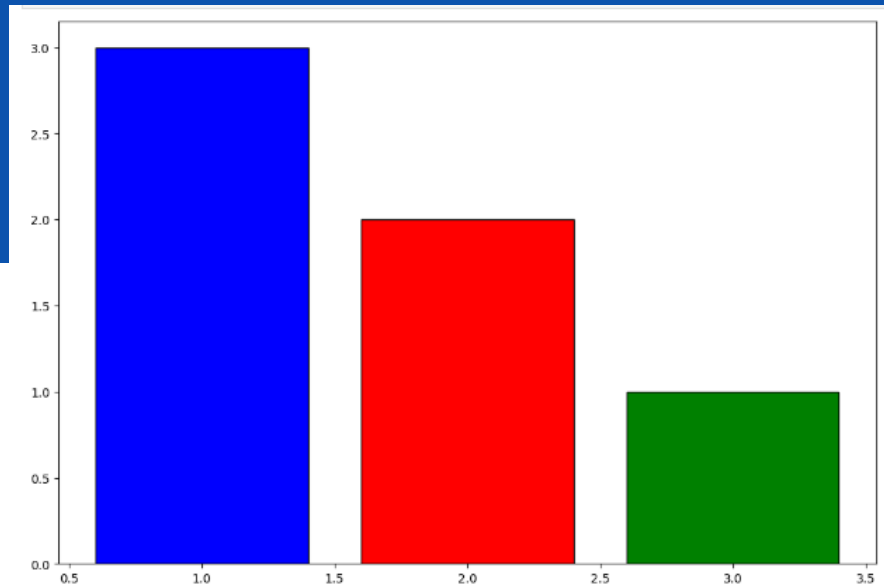
x: Cualitativa Ordinal , y: Cuantitativo



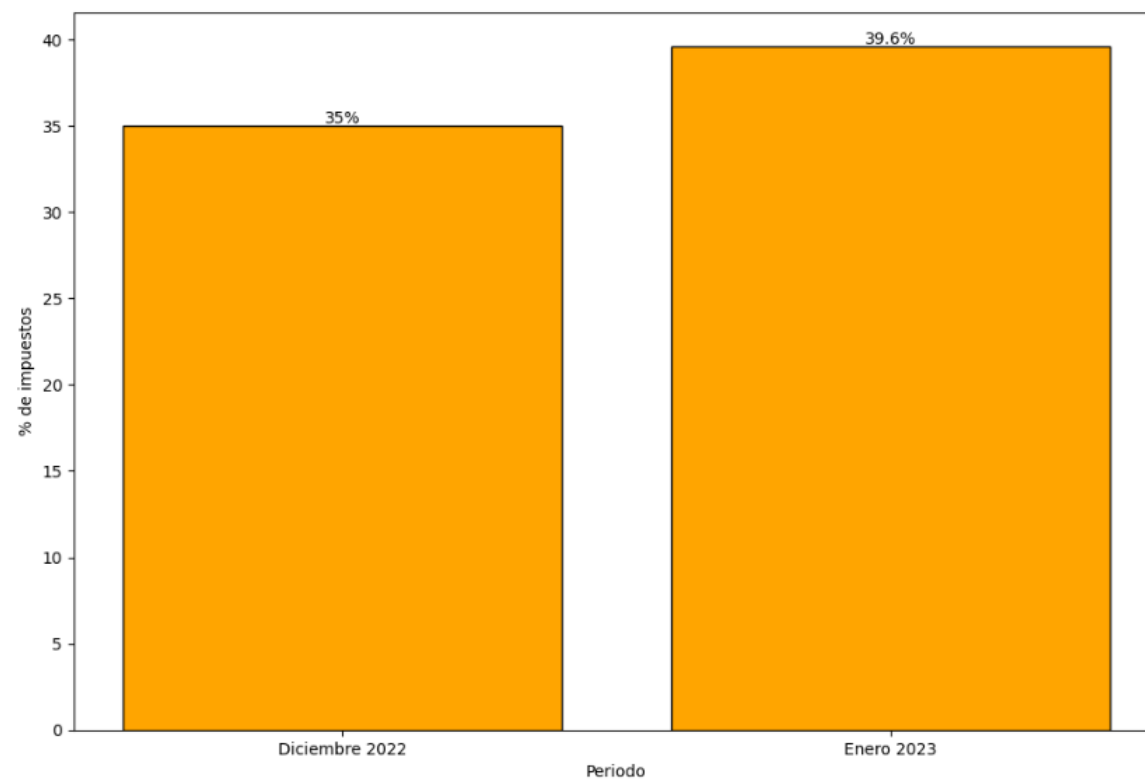
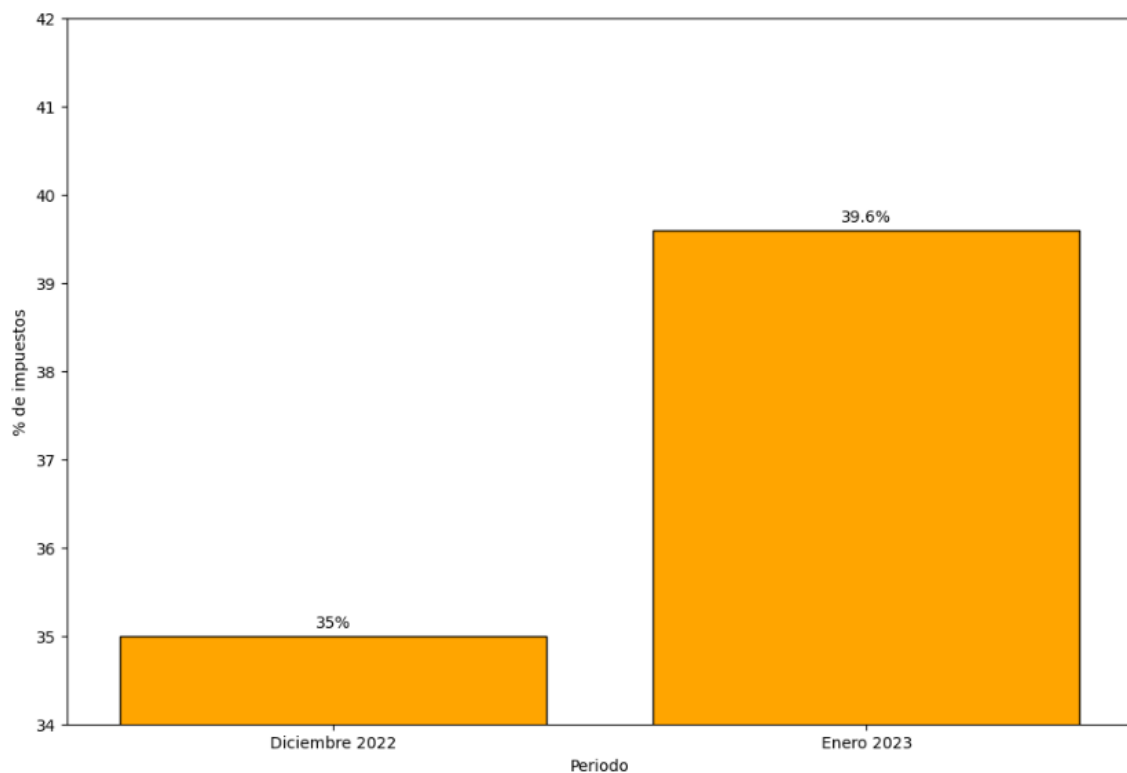
# Gráficos de Barras

Presenta datos categóricos con barras rectangulares con alturas o longitudes proporcionales a los valores que representan. Las barras se pueden trazar vertical u horizontalmente.

Muestra comparaciones entre categorías.



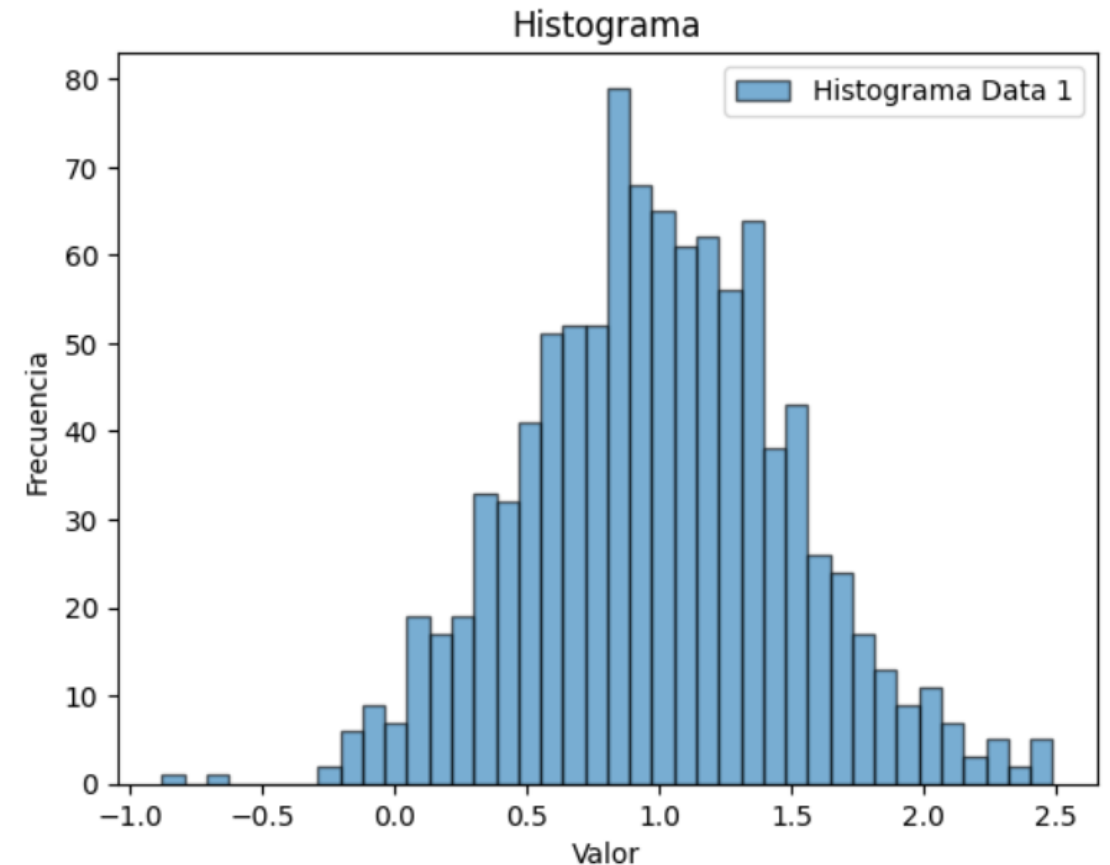
# Los gráficos de barras deben tener un valor base de cero porque sino obtendremos una interpretación visual falsa



# Histograma

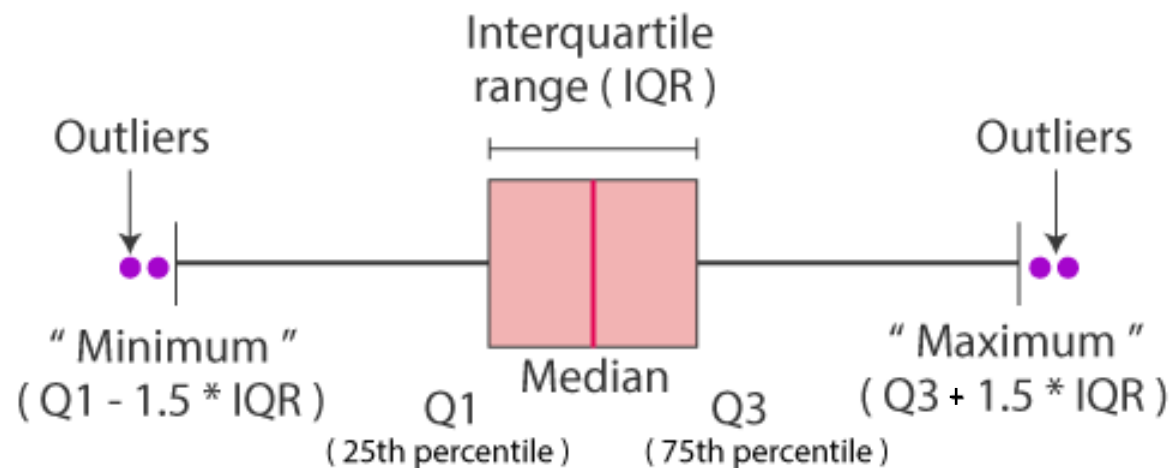
Es una representación gráfica de puntos de datos organizados en rangos especificados por el usuario. De apariencia similar a un gráfico de barras. **Representa la distribución de frecuencia** de las variables en un conjunto de datos.

**Se usa para variables cuantitativas continuas.**

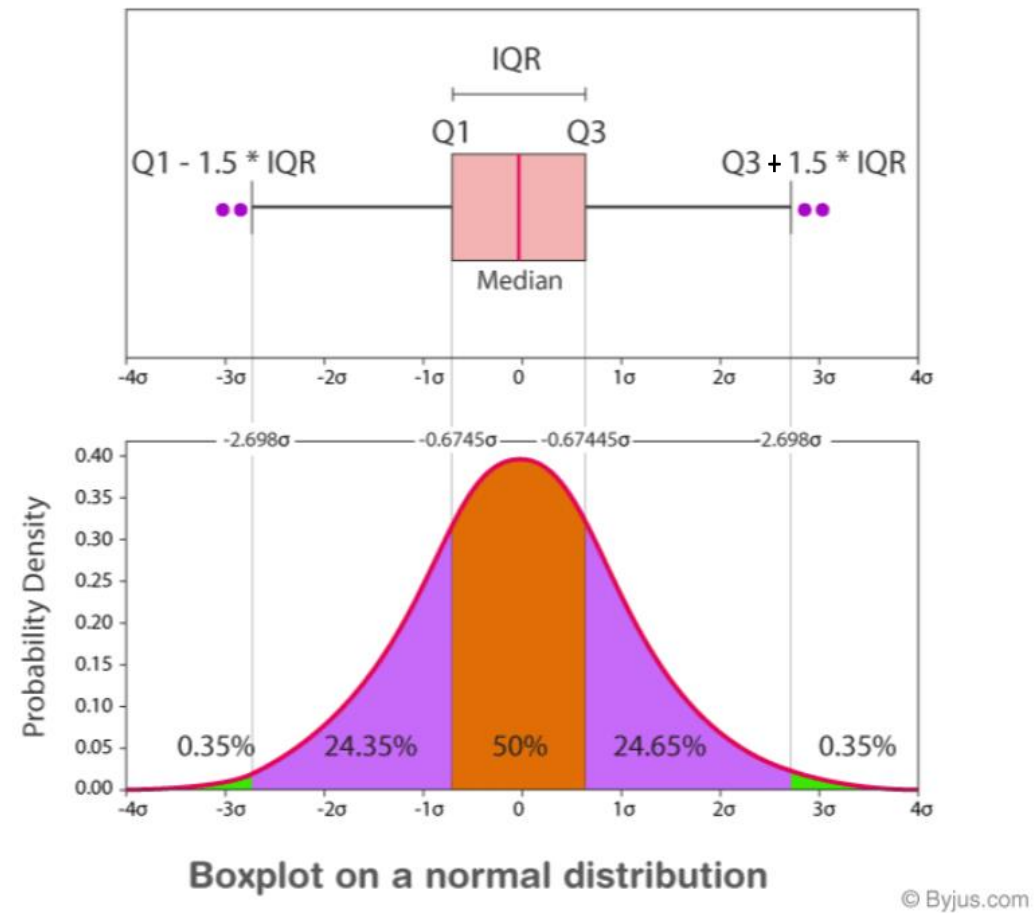


# Gráficos de caja

Es un método para demostrar gráficamente la dispersión y asimetría de los grupos de los datos numéricos a través de sus cuartiles. También conocido como Box Plot.

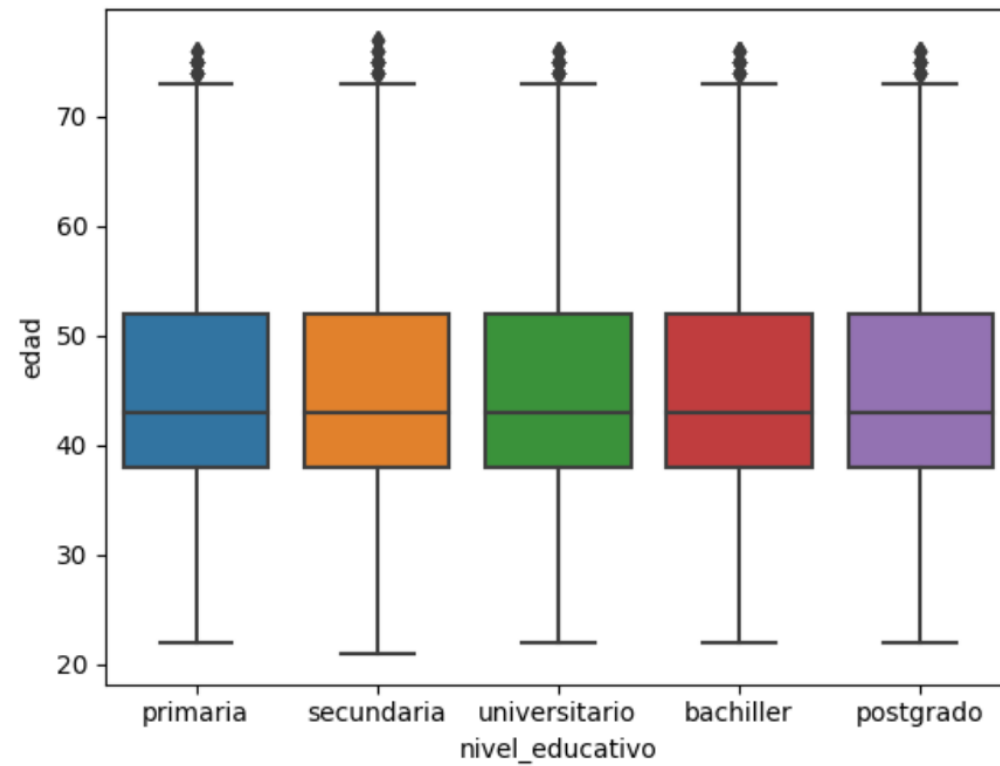


# Gráficos de caja



# Gráficos de caja

Comparación entre dos o más grupos de la distribución de una variable cuantitativa





# Gráficos de caja

## Normal Distribution

$$(\text{Quartile 3} - \text{Quartile 2}) = (\text{Quartile 2} - \text{Quartile 1})$$



## Positive Skew

$$(\text{Quartile 3} - \text{Quartile 2}) > (\text{Quartile 2} - \text{Quartile 1})$$



## Negative Skew

$$(\text{Quartile 3} - \text{Quartile 2}) < (\text{Quartile 2} - \text{Quartile 1})$$



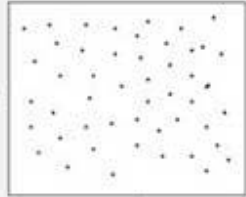
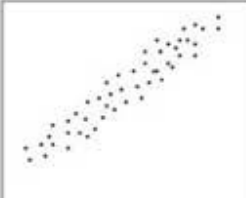
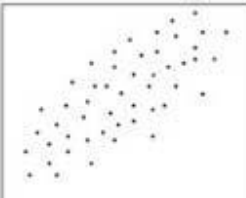
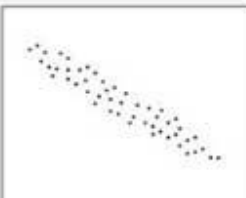
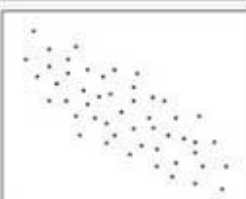
**Simétrico:** se dice que el diagrama de caja es simétrico si la mediana es equidistante de los valores máximo y mínimo.

**Asimetría positiva:** si la distancia desde la mediana hasta el máximo es mayor que la distancia desde la mediana hasta el mínimo, entonces el diagrama de caja tiene un asimetría positiva.

**Asimetría negativa:** si la distancia desde la mediana hasta el mínimo es mayor que la distancia desde la mediana hasta el máximo, entonces el diagrama de caja tiene un asimetría negativa.

# Gráficos de Dispersión

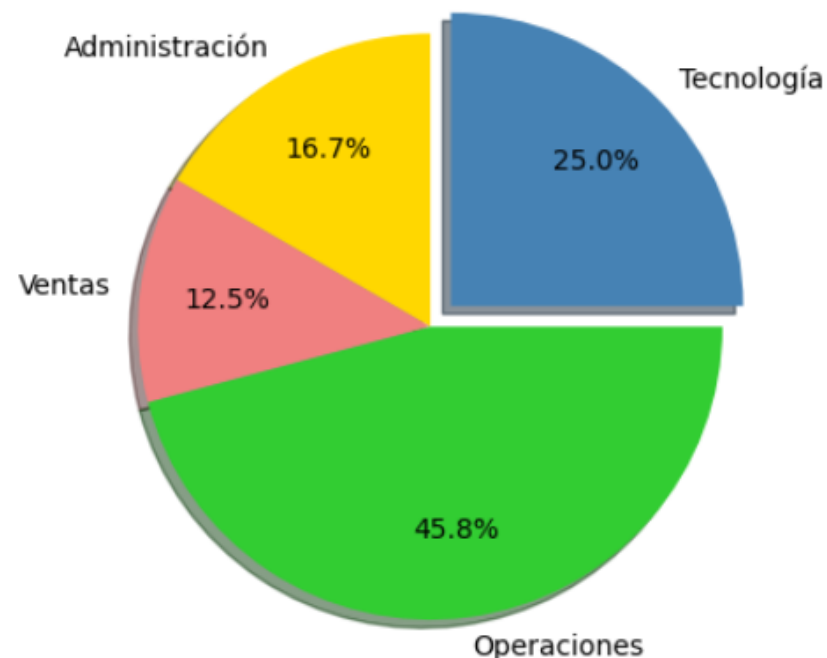
Nos permite analizar si existe algún tipo de **relación entre dos variables cuantitativas**.

Diagrama	Tipo de relación
	Sin relación. No se aprecia ninguna correlación entre las dos variables.
	Alta correlación positiva. El valor de Y se incrementa nítidamente a medida que el valor de X aumenta.
	Baja correlación positiva. El valor de X aumenta ligeramente a medida que aumenta el valor de Y.
	Fuerte correlación negativa. El valor de X claramente disminuye a medida que aumenta el valor de Y.
	Débil correlación negativa. El valor de X disminuye ligeramente a medida que aumenta el valor de Y.

# Gráficos Circulares

Es un diagrama que muestra los datos en sectores fáciles de entender. Cada sector representa una categoría de datos y el tamaño es proporcional a la cantidad que representa.

Los gráficos circulares son populares, ya que son fáciles de crear y comprender, pero solo **son efectivos cuando se comparan entre 3 y 5 categorías que sean fáciles de diferenciar.**



# QUIZ TIME





# **Gráficas con Matplotlib**

## Challenge 2:

De forma individual hagan 5 gráficos usando matplotlib usando la data que van a trabajar en su proyecto final.

Enviar por correo con el asunto: Challenge 2 – Visualización de Datos – [Apellidos y Nombres]

Correo: [team@dataanalitica.net](mailto:team@dataanalitica.net)

Fecha entrega: 29-marzo hasta las 10 pm

# **Gráficas con Seaborn**

# Ventajas de usar Seaborn

- Fácil de usar
- Trabaja bien con estructuras de datos Pandas
- Esta basado en matplotlib



# Objeto FacetGrid vs AxesSubplot

Tipo de Objeto	Plot Types	Características
FacetGrid	relplot() , catplot()	Puede crear subplots
AxesSubplot	scatterplot() , countplot() , etc.	Solo crea un solo gráfico

## Challenge 3:

De forma individual hagan 5 gráficos usando Seaborn usando la data que van a trabajar en su proyecto final.

Enviar por correo con el asunto: Challenge 3 – Visualización de Datos – [Apellidos y Nombres]

Correo: [team@dataanalitica.net](mailto:team@dataanalitica.net)

Fecha entrega: 04-abril hasta las 10 pm

# GRACIAS

DATA  
ANALÍTICA

