# **Código para generar gráficos**

## Gráfico de sectores (Pie chart) - pastel o torta

import matplotlib.pyplot as plt

# Datos

sistemas = ['Hogar', 'Servicios', 'Comidas', 'Supermercado'] # Categorías

porcentajes = [73260, 34000, 24000, 123000] # Valores

colores = ['#fbf989','#ff9999','#99ff99','#99bbff'] # Colores

explode = [0, 0, 0, 0.05] # Separacion (Opc)

# Crear gráfico de pastel

plt.figure(figsize=(8, 5))

plt.pie(porcentajes,

explode=explode,

labels=sistemas,

colors=colores,

autopct='%1.1f%%', # Mostrar porcentajes

)

# Personalización

plt.title('Gastos Mayo 2025',

fontsize=14)

plt.show()

## Gráfico de barras

import matplotlib.pyplot as plt

import pandas as pd

# Datos de ejemplo: Ventas mensuales (en miles de dólares)

datos = {

'Mes': ['Enero', 'Febrero', 'Marzo', 'Abril', 'Mayo', 'Junio'],

'Ventas': [45, 78, 56, 89, 67, 92]

}

# Crear un DataFrame con pandas

df = pd.DataFrame(datos)

# Configurar el gráfico de barras

plt.figure(figsize=(8, 5)) # Tamaño del gráfico (ancho, alto)

plt.bar(df['Mes'], df['Ventas'], color='skyblue', edgecolor='black')

# Personalizar el gráfico

plt.title('Ventas Mensuales (Primer Semestre)', fontsize=14, fontweight='bold')

plt.xlabel('Mes', fontsize=12)

plt.ylabel('Ventas (en miles de USD)', fontsize=12)

plt.grid(axis='y', linestyle='--', alpha=0.7) # Líneas de guía horizontales

# Mostrar el gráfico

plt.show()

## Gráfico de puntos (Dot Plot)

import matplotlib.pyplot as plt

import numpy as np

data = [1,1, 2, 2,2, 3, 3, 3, 3, 3, 3, 4, 4, 5, 5]

unique\_vals, counts = np.unique(data, return\_counts=True)

y\_coords = np.concatenate([np.arange(1, count + 1) for count in counts])

x\_coords = np.repeat(unique\_vals, counts)

plt.figure(figsize=(4, 2))

plt.scatter(x\_coords, y\_coords, color='blue')

# setea valores de los ejes

plt.xticks(unique\_vals)

plt.yticks(np.arange(1, max(counts) + 1))

plt.title("Ejemplo Dot Plot")

plt.xlabel("Valores")

plt.ylabel("Frequencia")

# Líneas de guía horizontales

plt.grid(axis='y', linestyle='--', alpha=0.3)

plt.show()

## Gráfico de Bastón

import matplotlib.pyplot as plt

import numpy as np

# Datos de ejemplo

datos = [42, 44, 56, 56, 58, 64, 69, 71, 72, 88]

# Crear el gráfico de bastón

plt.stem(datos)

plt.xlabel("Valores")

plt.ylabel("Frecuencia Absoluta")

plt.title("Gráfico de Bastón")

plt.show()

## Histograma con rangos de valores

import matplotlib.pyplot as plt

import numpy as np

# Datos de edades

edades = [12, 15, 22, 25, 27, 28, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 45, 47, 50]

tam\_intervalo = 5

bordes = np.arange(10, 60, tam\_intervalo)

# Crear histograma

plt.figure(figsize=(10, 5))

plt.hist(edades,

bins=bordes, # Bordes personalizados

color='#FFA07A', # Color de las barras

edgecolor='black')

# Personalización títulos

plt.title('Distribución con Intervalos Personalizados', fontsize=14)

plt.xlabel('Rango de Edad (años)', fontsize=12)

plt.ylabel('Frecuencia', fontsize=12)

plt.show()

Código extra: para agregar etiquetas centradas  
# Genera etiqueta  
etiquetas = [f"{bordes[i]:.0f}-{bordes[i+1]:.0f}" for i in range(len(bordes)-1)]  
  
# Centrar etiquetas  
plt.xticks(bordes[:-1] + tam\_intervalo/2, etiquetas, rotation=0)

## Histograma con valores discretos

import matplotlib.pyplot as plt

import numpy as np

# Datos de ejemplo (valores discretos)

calificaciones = [2, 3, 5, 7, 8, 4, 6, 5, 9, 10,

3, 4, 5, 6, 7, 8, 2, 3, 4, 5,

6, 7, 8, 9, 1, 2, 3, 4, 5, 6,

7, 8, 9, 10, 4, 5, 6, 7, 8, 2,

3, 4, 5, 6, 7, 8, 9, 10, 5, 6]

# Configurar el histograma

plt.figure(figsize=(10, 6))

plt.hist(calificaciones,

bins=np.arange(0.5, 11.5, 1), # Ajuste val. discretos

edgecolor='black',

color='#4CAF50')

# Personalización del diagrama

plt.title('Distribución de Calificaciones', fontsize=14, fontweight='bold')

plt.xlabel('Calificación (puntos)', fontsize=12)

plt.ylabel('Frecuencia', fontsize=12)

plt.xticks(range(1, 11)) # Etiquetas enteras en el eje X

plt.grid(axis='y', linestyle='--', alpha=0.5)

# Mostrar el gráfico

plt.show()

## Gráfico de líneas (un grupo)

import matplotlib.pyplot as plt

# Datos de ejemplo (temperaturas mensuales en °C)

meses = ['Ene', 'Feb', 'Mar', 'Abr', 'May', 'Jun', 'Jul', 'Ago', 'Sep', 'Oct', 'Nov', 'Dic']

temperaturas = [22, 21, 19, 16, 13, 10, 9, 11, 13, 16, 18, 20]

# Crear gráfico de líneas

plt.figure(figsize=(10, 5))

plt.plot(meses, temperaturas,

marker='o', # Puntos en cada dato

linestyle='-', # Línea sólida

color='#E74C3C', # Color rojo

linewidth=2, # Grosor de línea

label='Temperatura (°C)')

# Personalización

plt.title('Temperatura Promedio Mensual (2023)', fontsize=14, pad=20)

plt.xlabel('Mes', fontsize=12)

plt.ylabel('Temperatura (°C)', fontsize=12)

plt.grid(axis='y', linestyle='--', alpha=0.7) # Cuadrícula horizontal

plt.legend() # Mostrar leyenda

plt.show()

## Gráfico de líneas (dos grupos)

import numpy as np

import matplotlib.pyplot as plt

# Data

x = [2010, 2011, 2012, 2013]

y1 = [10, 15, 25, 18]

y2 = [43, 30, 6, 12]

# Line chart

fig, ax = plt.subplots()

ax.plot(x, y1, marker = "o", label = "2023")

ax.plot(x, y2, marker = "o", label = "2024")

ax.legend()

plt.xticks(np.arange(2010, 2014, step=1))

plt.show()

## Diagrama de cajas (Box plot)

import matplotlib.pyplot as plt

# Datos de ejemplo: notas de 3 grupos

grupo\_A = [12, 14, 15, 16, 16, 17, 18, 20]

grupo\_B = [5, 8, 10, 10, 11, 12, 14, 15]

grupo\_C = [2, 3, 6, 8, 10, 12, 14, 18]

# Crear el boxplot

plt.boxplot([grupo\_A, grupo\_B, grupo\_C], # Lista de grupos

labels=["Grupo A", "Grupo B", "Grupo C"], # Etiquetas

patch\_artist=True) # Para colorear las cajas

# Personalización mínima

plt.title("Distribución de Notas por Grupo", fontsize=12)

plt.ylabel("Puntuación (0-20)")

plt.grid(axis='y', linestyle='--', alpha=0.4)

plt.show()

## Diagrama de Dispersión (Scatter plot)

import matplotlib.pyplot as plt

import numpy as np

# Datos precargados

edades = [25, 30, 35, 40, 45, 50, 55, 60, 65, 70, 75, 80, 25, 30, 35, 40, 45, 50, 55, 60]

presion\_arterial = [120, 122, 125, 130, 132, 138, 140, 145, 150, 155,

160, 162, 118, 121, 124, 128, 130, 135, 142, 144]

# Configuración del gráfico

plt.figure(figsize=(10, 6))

plt.scatter(

edades,

presion\_arterial,

c='#e74c3c', # Color rojo de los puntos

s=80, # Tamaño de los puntos

edgecolor='black', # Borde negro

)

# Personalización

plt.title('Relación entre Edad y Presión Arterial', fontsize=14, pad=20)

plt.xlabel('Edad (años)', fontsize=12)

plt.ylabel('Presión Arterial (mmHg)', fontsize=12)

plt.grid(linestyle='--', alpha=0.3)

# Añadir línea de tendencia

coeficientes = np.polyfit(edades, presion\_arterial, 1)

linea\_tendencia = np.poly1d(coeficientes)

plt.plot(edades, linea\_tendencia(edades), 'b--',

label=f'Tendencia: y = {coeficientes[0]:.2f}x + {coeficientes[1]:.2f}')

plt.legend()

plt.show()