

Matplotlib

Gráficos combinados

CertiDevs

Índice de contenidos

1. Introducción	1
2. Creación de múltiples gráficos en una figura	1
3. Combinación de diferentes tipos de gráficos	1
4. Visualización de datos en múltiples dimensiones	2
5. Gráfico de dispersión (Scatter plot):	2
6. Gráfico de línea (Line plot):	3
7. Gráfico de barras (Bar plot):	3
8. Ejemplo de gráfico combinado: barras y líneas	3
9. Ejemplo de gráfico multivariante: diagrama de dispersión con tamaño y color	4
10. Ejemplo de gráfico de caja combinado con diagrama de dispersión	4
11. Ejemplo de gráfico de barras apiladas y gráfico de línea combinados	5
12. Más ejemplos	6

1. Introducción

Con Matplotlib es posible crear **gráficos avanzados** y **combinados** en Matplotlib.

Esto incluye la creación de **gráficos múltiples** en una figura, la combinación de diferentes tipos de gráficos y la visualización de datos en múltiples dimensiones utilizando mapas de color y tamaño.

2. Creación de múltiples gráficos en una figura

Para crear **múltiples gráficos** en una sola figura, utilizamos la función `subplots()` que devuelve una matriz de objetos Axes:

```
import matplotlib.pyplot as plt

fig, ax = plt.subplots(2, 2)

x = [0, 1, 2, 3, 4, 5]
y = [0, 2, 4, 6, 8, 10]

ax[0, 0].plot(x, y)
ax[0, 1].scatter(x, y)
ax[1, 0].bar(x, y)
ax[1, 1].hist(y, bins=5)
plt.show()
```

3. Combinación de diferentes tipos de gráficos

Puede **combinar diferentes tipos de gráficos** en un solo objeto Axes simplemente llamando a las funciones de trazado correspondientes en el mismo objeto Axes:

```
import matplotlib.pyplot as plt

x = [0, 1, 2, 3, 4, 5]
y1 = [0, 2, 4, 6, 8, 10]
y2 = [0, 1, 4, 9, 16, 25]

fig, ax = plt.subplots()

ax.plot(x, y1, label='Lineal')
ax.scatter(x, y2, label='Cuadrática', color='red', marker='x')
ax.legend()
```

```
plt.show()
```

4. Visualización de datos en múltiples dimensiones

Puede utilizar el **tamaño** y el **color** de los marcadores para representar datos adicionales en sus gráficos:

```
import matplotlib.pyplot as plt
import numpy as np

x = np.random.rand(50)
y = np.random.rand(50)
colors = np.random.rand(50)
sizes = 1000 * np.random.rand(50)

plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='viridis')
plt.colorbar()
plt.show()
```

En este ejemplo, los colores y los tamaños de los marcadores se generan aleatoriamente, pero en la práctica, podría utilizar estas propiedades para representar características adicionales de sus datos.

5. Gráfico de dispersión (Scatter plot):

Supongamos que quieres analizar la relación entre las horas de estudio y las calificaciones obtenidas por los estudiantes.

```
import numpy as np
import matplotlib.pyplot as plt

# Datos ficticios
horas_estudio = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9])
calificaciones = np.array([40, 50, 60, 62, 75, 80, 85, 90, 100])

# Crear un gráfico de dispersión
plt.scatter(horas_estudio, calificaciones)
plt.xlabel('Horas de estudio')
plt.ylabel('Calificaciones')
plt.title('Relación entre horas de estudio y calificaciones')
plt.show()
```

6. Gráfico de línea (Line plot):

Supongamos que quieres analizar la tendencia en las ventas mensuales de un producto a lo largo de un año.

```
# Datos ficticios
meses = np.arange(1, 13)
ventas_mensuales = np.array([50, 60, 65, 80, 85, 90, 95, 100, 120, 140, 160, 200])

# Crear un gráfico de línea
plt.plot(meses, ventas_mensuales)
plt.xlabel('Meses')
plt.ylabel('Ventas mensuales')
plt.title('Tendencia de ventas mensuales a lo largo del año')
plt.show()
```

7. Gráfico de barras (Bar plot):

Supongamos que quieres comparar el número de ventas de diferentes categorías de productos en tu tienda.

```
# Datos ficticios
categorias = ['A', 'B', 'C', 'D']
ventas_categorias = [100, 150, 120, 200]

# Crear un gráfico de barras
plt.bar(categorias, ventas_categorias)
plt.xlabel('Categorías')
plt.ylabel('Ventas')
plt.title('Comparación de ventas por categoría')
plt.show()
```

8. Ejemplo de gráfico combinado: barras y líneas

En este ejemplo, combinaremos un **gráfico de barras** y uno de **líneas** en una sola figura.

Supongamos que queremos comparar las ventas de un producto a lo largo del tiempo con su precio promedio mensual:

```
import matplotlib.pyplot as plt

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

```

ventas = [25, 32, 34, 20, 41, 35, 30, 28, 33, 45, 38, 40]
precios = [100, 98, 102, 101, 99, 100, 97, 96, 95, 94, 93, 92]

fig, ax1 = plt.subplots()

# Gráfico de barras
ax1.bar(meses, ventas, color='blue', alpha=0.5, label='Ventas')

# Gráfico de línea
ax2 = ax1.twinx()
ax2.plot(meses, precios, color='red', label='Precio promedio', marker='o')

# Leyendas
fig.legend(loc='upper left')
plt.show()

```

9. Ejemplo de gráfico multivariante: diagrama de dispersión con tamaño y color

En este ejemplo, utilizaremos un conjunto de datos con cuatro variables (x, y, tamaño y color) y los representaremos en un diagrama de dispersión:

```

import matplotlib.pyplot as plt
import numpy as np

np.random.seed(42)

x = np.random.rand(50)
y = np.random.rand(50)
sizes = 500 * np.random.rand(50)
colors = np.random.rand(50)

plt.scatter(x, y, c=colors, s=sizes, alpha=0.5, cmap='viridis')
plt.colorbar()
plt.show()

```

10. Ejemplo de gráfico de caja combinado con diagrama de dispersión

En este ejemplo, combinaremos un **gráfico de caja** (box plot) con un **diagrama de dispersión** para mostrar la distribución de los datos y los valores individuales:

```

import matplotlib.pyplot as plt
import numpy as np

```

```

np.random.seed(42)

data = np.random.randn(100)

fig, ax1 = plt.subplots()

# Gráfico de caja
ax1.boxplot(data, vert=False, widths=0.5, positions=[0])

# Diagrama de dispersión
ax2 = ax1.twinx()
ax2.scatter(data, np.zeros_like(data), alpha=0.5)

# Configurar límites del eje y
ax1.set_ylim(-0.5, 0.5)
ax2.set_ylim(-0.5, 0.5)
plt.show()

```

11. Ejemplo de gráfico de barras apiladas y gráfico de línea combinados

En este ejemplo, combinaremos un gráfico de **barras apiladas** y un **gráfico de línea** para mostrar la composición de las ventas totales por categoría y el total mensual:

```

import matplotlib.pyplot as plt

meses = ['Ene', 'Feb', 'Mar', 'Abr', 'May', 'Jun', 'Jul', 'Ago', 'Sep', 'Oct', 'Nov', 'Dic']
ventas_a = [25, 32, 34, 20, 41, 35, 30, 28, 33, 45, 38, 40]
ventas_b = [15, 25, 28, 18, 30, 26, 22, 20, 28, 32, 29, 30]

ventas_totales = [sum(x) for x in zip(ventas_a, ventas_b)]

fig, ax1 = plt.subplots()

# Gráfico de barras apiladas
ax1.bar(meses, ventas_a, label='Categoría A')
ax1.bar(meses, ventas_b, bottom=ventas_a, label='Categoría B')

# Gráfico de línea
ax2 = ax1.twinx()
ax2.plot(meses, ventas_totales, color='red', label='Ventas totales', marker='o')

# Leyendas
fig.legend(loc='upper left')
plt.show()

```

12. Más ejemplos

Ver gráficos en <https://www.python-graph-gallery.com/>