

9. Introducción a la visualización de datos

La visualización de datos es una herramienta fundamental en el análisis de datos, que permite presentar de manera clara y concisa información compleja a través de gráficos y otras representaciones visuales. En este sentido, Python cuenta con diversas bibliotecas especializadas en la creación de gráficos y visualizaciones de datos, siendo una de las más populares la biblioteca Matplotlib.

a) Introducción a la biblioteca Matplotlib

Matplotlib es una biblioteca de gráficos para Python que permite crear una gran variedad de gráficos estáticos, animados e interactivos en 2D y 3D. Esta biblioteca se integra perfectamente con NumPy y Pandas, lo que permite manipular y visualizar datos de manera eficiente. Las funciones básicas de Matplotlib incluyen la creación de gráficos, configuración de sus propiedades, y la visualización de los mismos.

Además de las funciones básicas, Matplotlib cuenta con una gran cantidad de recursos online, tales como tutoriales, documentación y ejemplos que permiten a los usuarios explorar y aprender de manera sencilla y eficiente. Algunos de estos recursos son:

- La galería de Matplotlib, que presenta una gran variedad de gráficos con su código correspondiente.
- El tutorial oficial de Matplotlib, que cubre desde los conceptos básicos hasta las funciones más avanzadas.
- La documentación oficial de Matplotlib, que describe todas las funciones y parámetros de la biblioteca.

b) Diferentes tipos de gráficos

Matplotlib permite crear diferentes tipos de gráficos para visualizar datos de manera eficiente. Algunos de los gráficos más utilizados son:

i. Histogramas

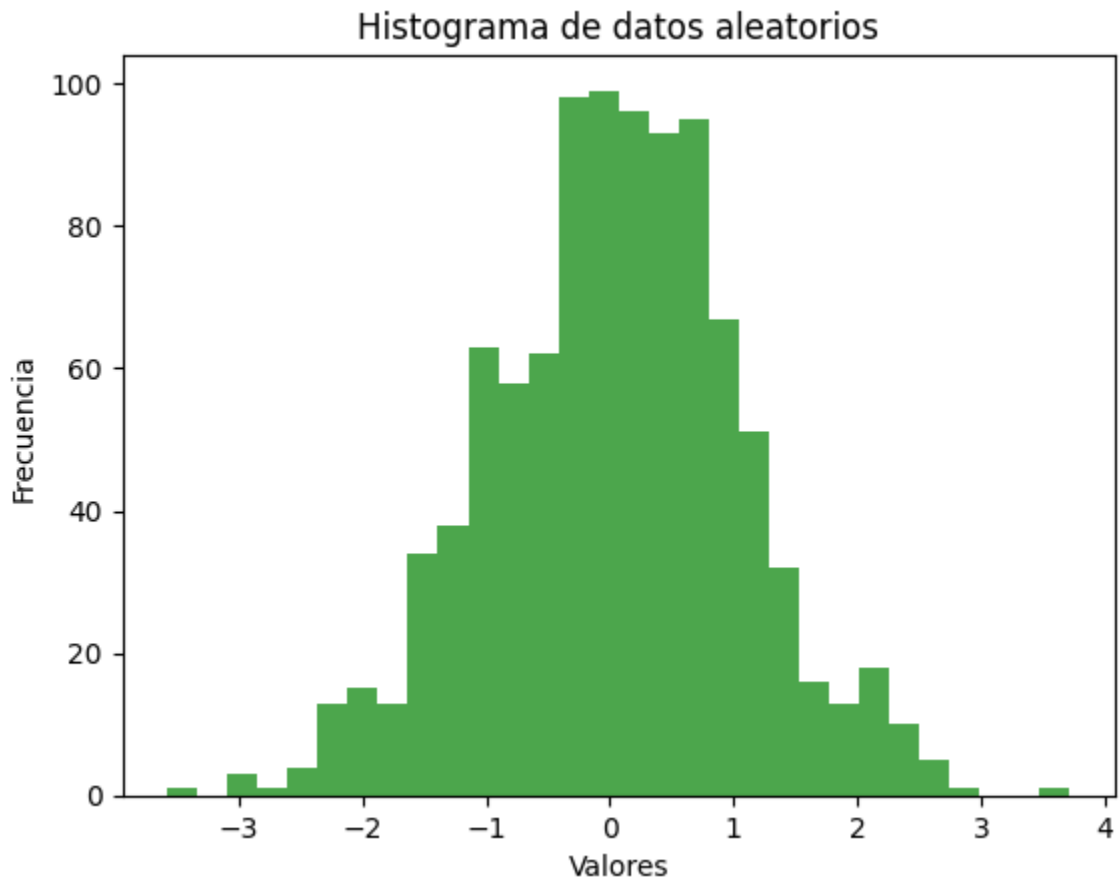
Estos gráficos muestran la distribución de frecuencia de un conjunto de datos.

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

# Datos aleatorios
x = np.random.randn(1000)

# Crear histograma
```

```
plt.hist(x, bins=30, color='green', alpha=0.7)
plt.title('Histograma de datos aleatorios')
plt.xlabel('Valores')
plt.ylabel('Frecuencia')
plt.show()
```



ii. Gráficos de línea

Estos gráficos muestran la evolución de un conjunto de datos a lo largo del tiempo.

```
import matplotlib.pyplot as plt
```

```
# Datos de ejemplo
```

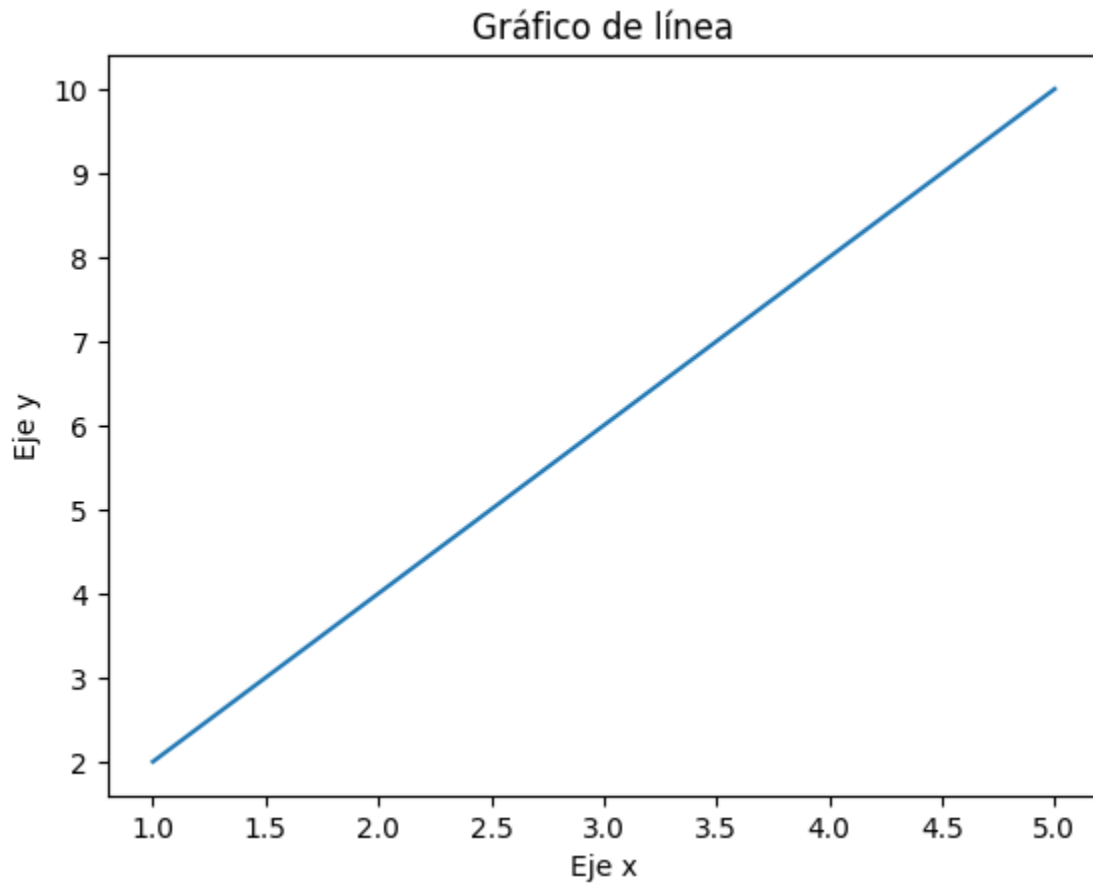
```
x = [1, 2, 3, 4, 5]
```

```
y = [2, 4, 6, 8, 10]
```

```
# Gráfico de línea
```

```
plt.plot(x, y)
```

```
plt.title('Gráfico de línea')
plt.xlabel('Eje x')
plt.ylabel('Eje y')
plt.show()
```



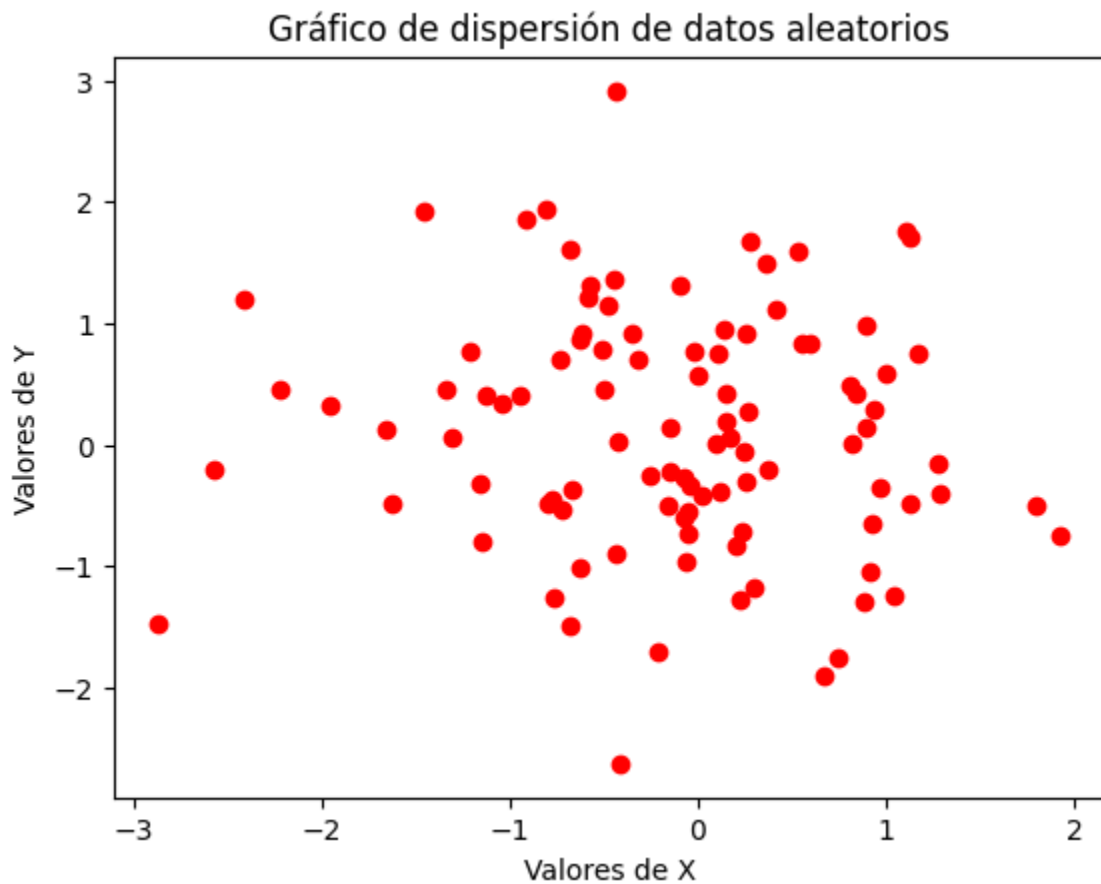
iii. Gráficos de dispersión

Estos gráficos muestran la relación entre dos variables.

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

# Datos aleatorios
x = np.random.randn(100)
y = np.random.randn(100)
```

```
# Crear gráfico de dispersión
plt.scatter(x, y, color='red')
plt.title('Gráfico de dispersión de datos aleatorios')
plt.xlabel('Valores de X')
plt.ylabel('Valores de Y')
plt.show()
```



iv. Gráficos de barras

Estos gráficos muestran la comparación entre diferentes categorías.

```
import matplotlib.pyplot as plt

# Datos
```

```
etiquetas = ['Manzanas', 'Naranjas', 'Peras']  
valores = [10, 15, 5]
```

```
# Crear gráfico de barras
```

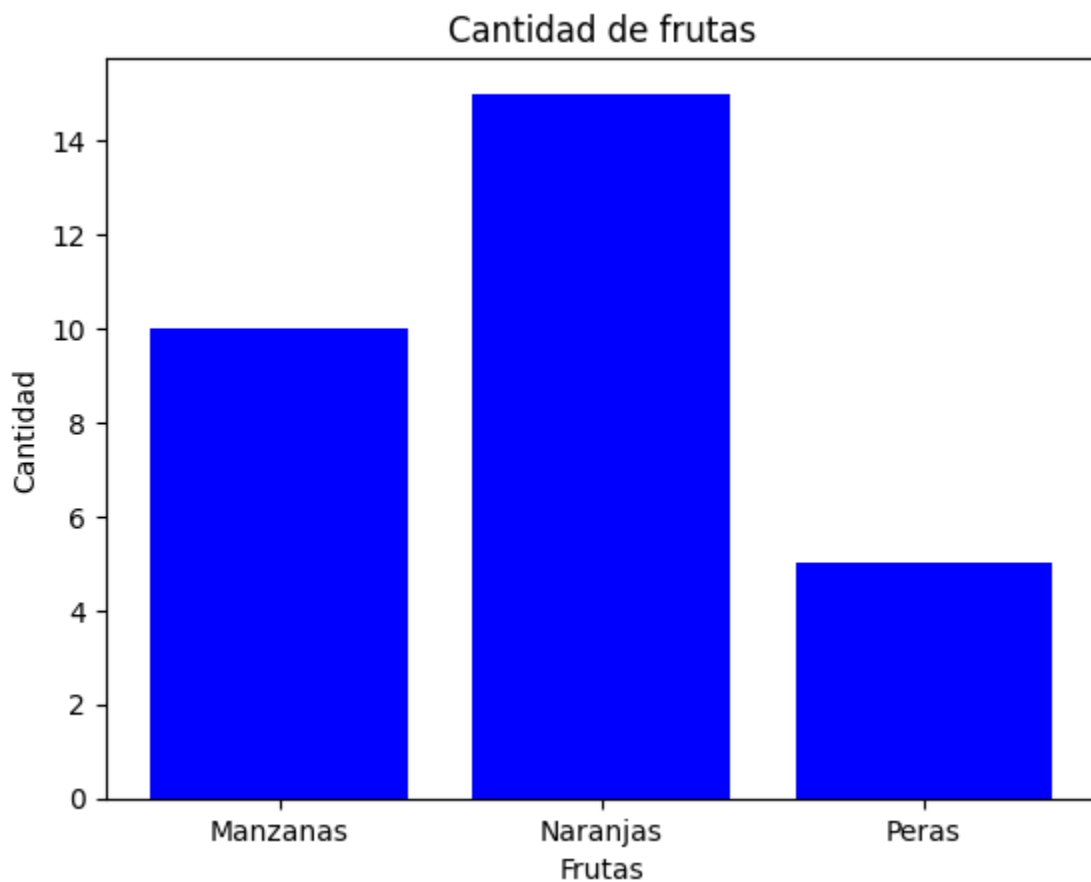
```
plt.bar(etiquetas, valores, color='blue')
```

```
plt.title('Cantidad de frutas')
```

```
plt.xlabel('Frutas')
```

```
plt.ylabel('Cantidad')
```

```
plt.show()
```



v. Gráficos de pastel

Estos gráficos muestran la proporción de cada categoría dentro de un conjunto de datos.

```
import matplotlib.pyplot as plt
# Gráfico de pastel
frutas = ['Manzanas', 'Naranjas', 'Plátanos', 'Peras']
cantidades = [20, 10, 15, 5]
colores = ['red', 'orange', 'yellow', 'green']

plt.pie(cantidades, labels=frutas, colors=colores)
plt.title('Gráfico de pastel')
plt.show()
```



En resumen, la biblioteca Matplotlib es una herramienta esencial para la visualización de datos en Python, permitiendo crear diferentes tipos de gráficos para presentar de manera clara y concisa información compleja.