

Graficar en Python con Matplotlib y NumPy

Código fuente

Trazar gráficas en Python es muy sencillo, para ello necesita tener instaladas las librerías matplotlib y numpy, las cuales pueden encontrarse y descargar de la red sin mayores obstáculos. Debe asegurarse que la versión de las librerías sea compatible con la versión de Python.

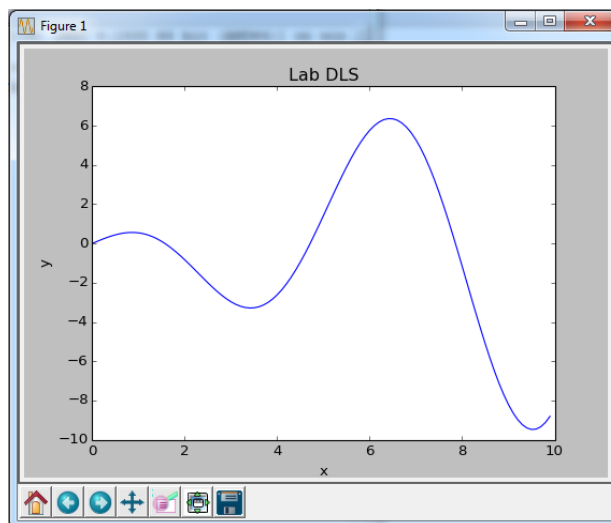
Si alguna vez ha utilizado MATLAB para estos mismos fines, las instrucciones le parecerán demasiado familiar.

Enseguida os adjunto un código sencillo y el resultado que produce:

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

x = np.arange(0,10,0.1)
y = x*np.cos(x)

plt.plot(x,y)
plt.xlabel('x')
plt.ylabel('y')
plt.title('Lab DLS')
plt.show()
```



Las primeras dos líneas sirven para importar las librerías que se utilizarán. Recuerde que en Python pueden utilizarse seudónimos al cargar una librería (en este caso plt para matplotlib.pyplot y np para numpy).

En la línea se 4 se define el vector de la variable independiente utilizando la instrucción `arange` de la librería `numpy`, el cual crea un vector especificando el valor inicial, el valor final y el incremento como argumentos. La línea 5 crea simplemente un vector dependiente del primero.

La instrucción `plot` gráfica los vectores creados con anterioridad, teniendo como primer argumento el vector de la variable independiente. Con `xlabel`, `ylabel` y `title` se muestran las etiquetas correspondientes a los ejes horizontal, vertical y el título en la parte superior respectivamente. Finalmente, la instrucción `show()` sirve para mostrar la ventana gráfica creada y poder visualizar lo que se ha trazado.

Modificando la presentación...

Grosor de línea

Para modificar el grosor de línea basta con incluir como argumento adicional en `plot` la propiedad `linewidth`, tal como se muestra enseguida:

```
plt.plot(x,y,linewidth=4)
```

Color de línea

El siguiente ejemplo configura el color de la línea, en este caso rojo.

```
plt.plot(x,y,color='r')
```

También puede especificar el color utilizando una tupla de 3 elementos (R,G,B) en el intervalo [0 1], por ejemplo:

```
plt.plot(x,y,color=(0.8,0.9,0))
```

Agregando rejilla

Para agregar una rejilla debe incluirse la instrucción `grid()` como se muestra:

```
plt.grid()
```

Mostrar más de una gráfica

Para mostrar más de una gráfica en la misma ventana, utilice `hold(True)` después de haber creado la primer gráfica o bien antes de ella, tal como se muestra en el código siguiente:

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

x = np.arange(0,10,0.2)
y1 = np.cos(x)
y2 = np.sin(x)

plt.plot(x,y1,'o',linewidth=3,color=(0.2,0.1,0.4))
plt.hold(True)
plt.plot(x,y2,'-',linewidth=2,color='g')
plt.grid()
plt.axis('equal')
plt.xlabel('x')
plt.ylabel('y')
plt.title('Lab DLS')
plt.show()
```

