

Python para Análisis de Datos

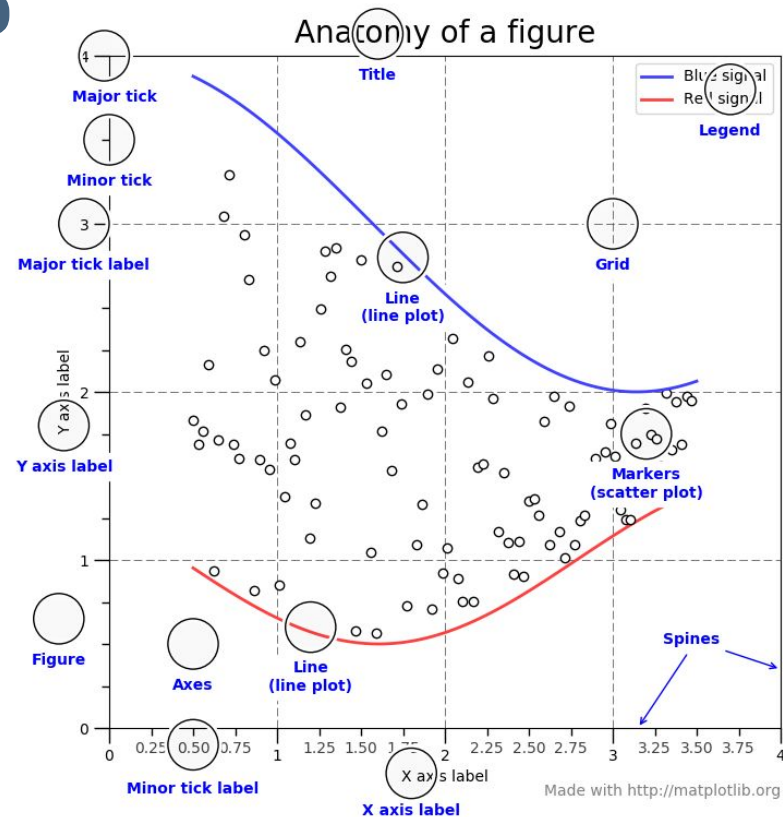
Módulo 06 – Matplotlib 3

Matplotlib

Anatomía de un gráfico

Un gráfico de matplotlib se puede componer de muchas partes, cada una representada por un tipo de objeto particular, constituyendo una jerarquía de objetos.

El más grande de ellos es la **figura**, que puede contener uno o más **subgráficos** y estos a su vez pueden contener otros objetos como *líneas*, *puntos*, *legendas*, etc.



Anatomía de un gráfico

Para controlar las propiedades de la figura se usa la función `figure`. Esta función crea una nueva figura. Cuando usamos solamente la función `plot` la figura se crea implícitamente con los valores por defecto. Al crearla explícitamente podemos controlar sus parámetros, como por ejemplo el tamaño de la figura.

El parámetro `figsize` recibe una tupla de números con las dimensiones de ancho y alto total de la figura medido en pulgadas.



Figura activa

La interfaz de pyplot trabaja manteniendo un registro de cuál es la figura activa, de modo que las funciones que se invocan actúan sobre ella. Por ejemplo, si llamamos a la función `plt` muchas veces seguidas actúan sobre el mismo gráfico. Para que actúen sobre figuras distintas hay que crear una nueva figura mediante `figure`.

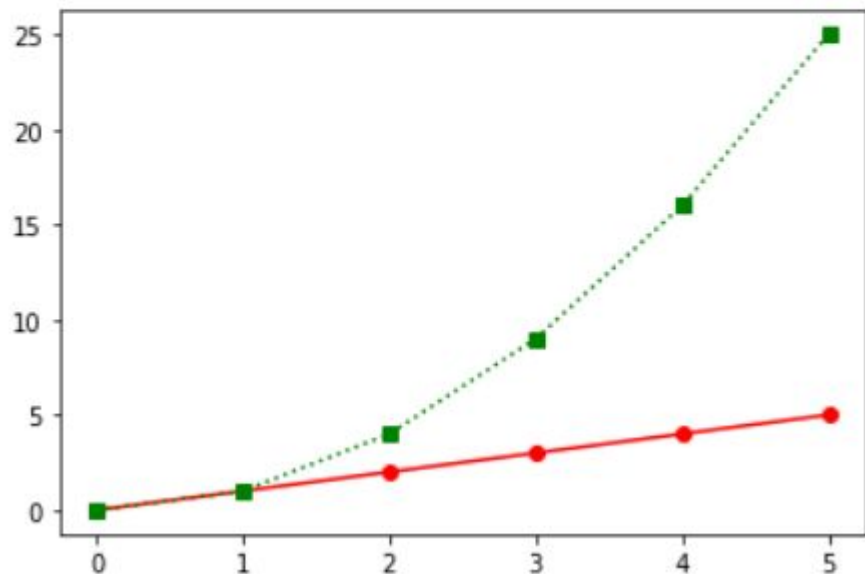
Así podemos representar varios conjuntos de datos en una o varias figuras.



Ejemplos

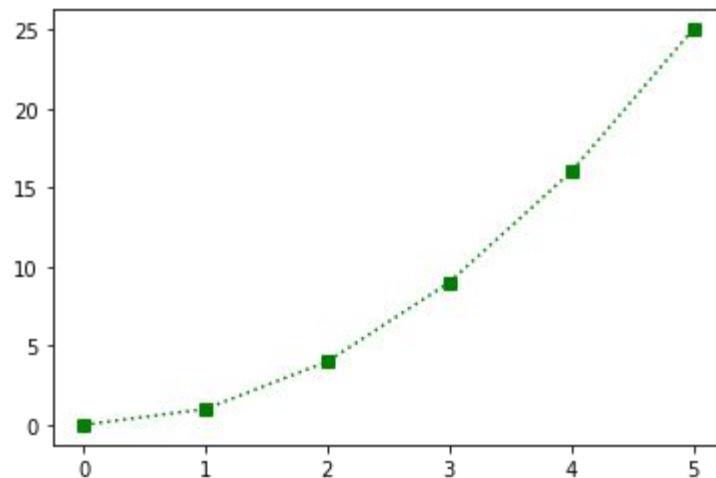
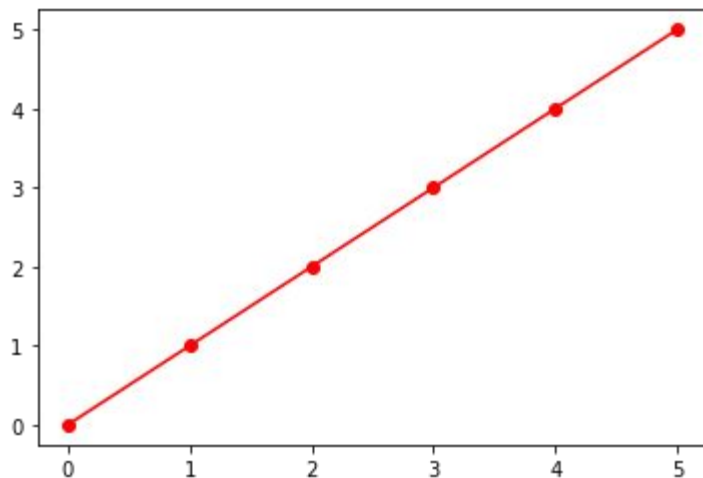
```
x = np.array([0,1,2,3,4,5])  
plt.plot(x, "o-r")  
plt.plot(x**2, "s:g")
```

[<matplotlib.lines.Line2D at 0x7f0ef5f548e0>]



Ejemplos

```
x = np.array([0,1,2,3,4,5])  
plt.plot(x, "o-r")  
plt.figure()  
plt.plot(x**2, "s:g")
```



Otros elementos del gráfico

Del mismo modo podemos agregar más elementos siempre teniendo en cuenta que las funciones actúan sobre la figura activa. Por ejemplo la función `title` permite agregarle un título a la figura. Las funciones `xlabel` e `ylabel` permiten definir el texto de los ejes.

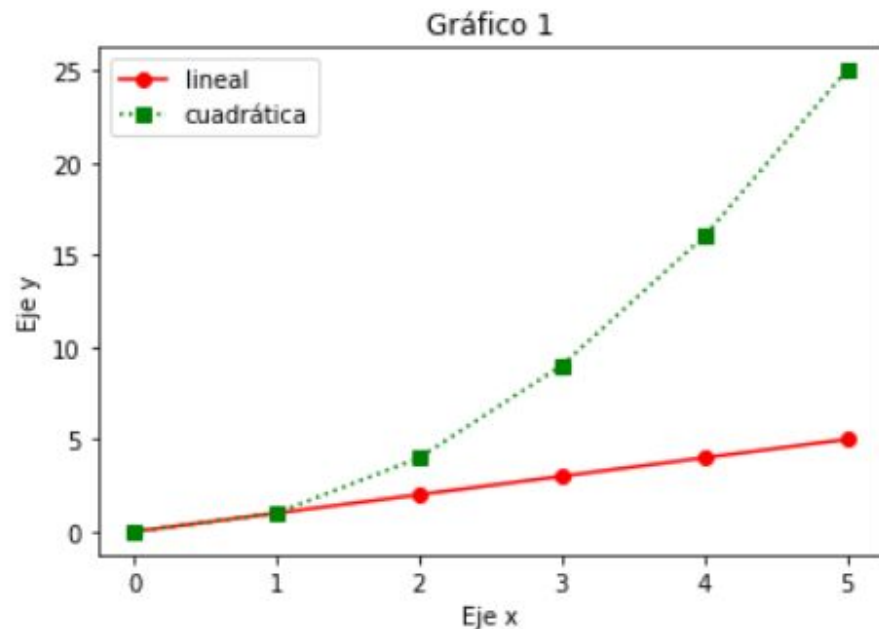
Si cada conjunto de datos tiene definido un label, la función `legend` permite activar el recuadro de leyendas. El label se define dentro de la función que construye el gráfico con el parámetro del mismo nombre.

No importa el orden en que se vayan llamando a las distintas funciones.



Ejemplos

```
x = np.array([0,1,2,3,4,5])
plt.plot(x, "o-r", label="lineal")
plt.plot(x**2, "s:g", label="cuadrática")
plt.title("Gráfico 1")
plt.xlabel("Eje x")
plt.ylabel("Eje y")
plt.legend()
```



Más elementos

Se pueden agregar muchos elementos más llamando a las funciones adecuadas. Por ejemplo podemos agregar textos en cualquier parte del gráfico con la función `text`, líneas verticales u horizontales con `axvline` y `axhline`.

También se puede activar una grilla con la función `grid` o activar y desactivar las líneas pequeñas sobre los ejes con `minorticks_on` / `minorticks_off`.

En general, todas las funciones tienen numerosos parámetros para controlar cómo se muestra cada elemento.



¡Muchas gracias!

¡Sigamos trabajando!