

# Gráficas con la librería Matplotlib para Python

Jhon Jairo Padilla A., PhD.

# Pyplot

- `matplotlib.pyplot` es una colección de funciones estilo comandos que hacen que `matplotlib` sea similar a `matlab`.
- Cada función de `pyplot` hace algunos cambios a la figura:
  - Crear la figura
  - Crear un área de impresión en una figura
  - Imprimir líneas en el área de impresión
  - Agregar etiquetas a las figuras
  - Etc.

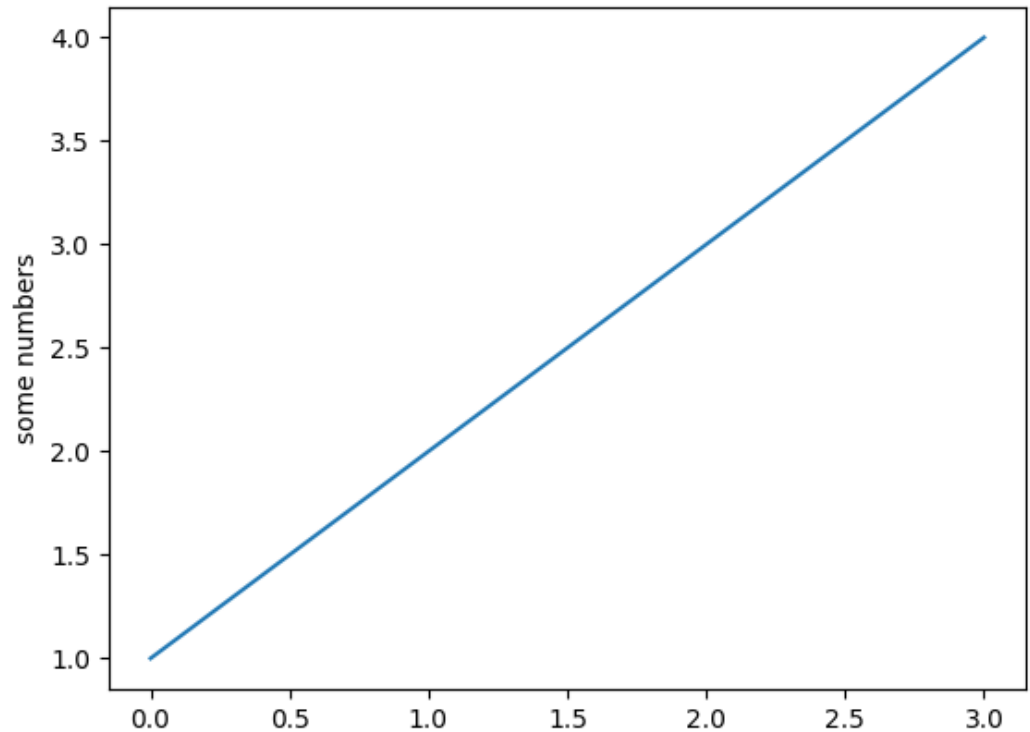
# Ejemplo

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

```
plt.plot([1, 2, 3, 4])
```

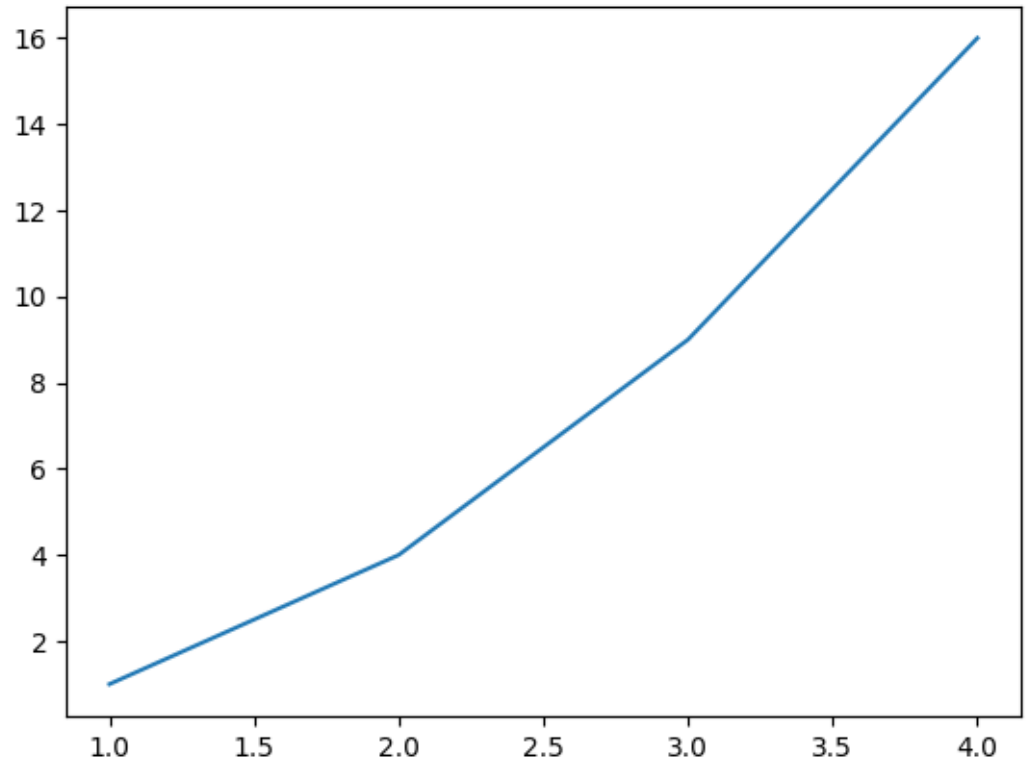
```
plt.ylabel('some numbers')
```

```
plt.show()
```



# Ejemplo

```
import matplotlib.pyplot as plt  
plt.plot([1, 2, 3, 4], [1, 4, 9, 16])  
plt.show()
```



# Ejemplo

```
import numpy as np
```

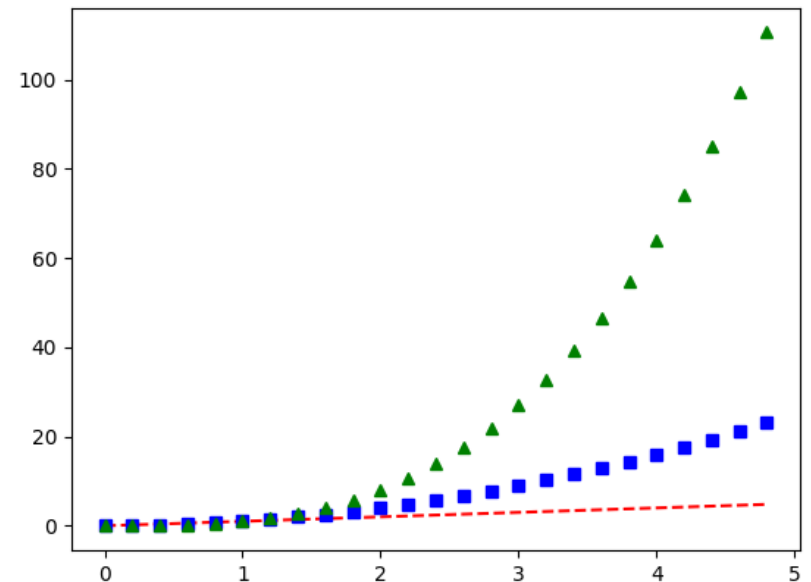
```
# generar puntos cada 0.2 en t
```

```
t = np.arange(0., 5., 0.2)
```

```
# líneas rojas, cuadros azules y triángulos verdes
```

```
plt.plot(t, t, 'r--', t, t**2, 'bs', t, t**3, 'g^')
```

```
plt.show()
```



# Graficos con variables categóricas

```
names = ['group_a', 'group_b', 'group_c']
```

```
values = [1, 10, 100]
```

```
plt.figure(1, figsize=(9, 3))
```

```
plt.subplot(131)
```

```
plt.bar(names, values)
```

```
plt.subplot(132)
```

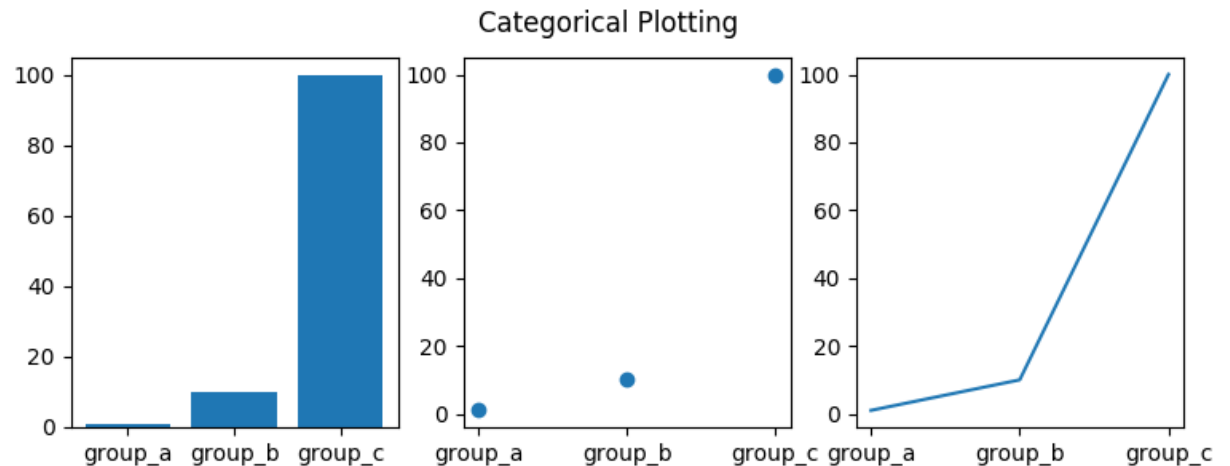
```
plt.scatter(names, values)
```

```
plt.subplot(133)
```

```
plt.plot(names, values)
```

```
plt.suptitle('Categorical Plotting')
```

```
plt.show()
```

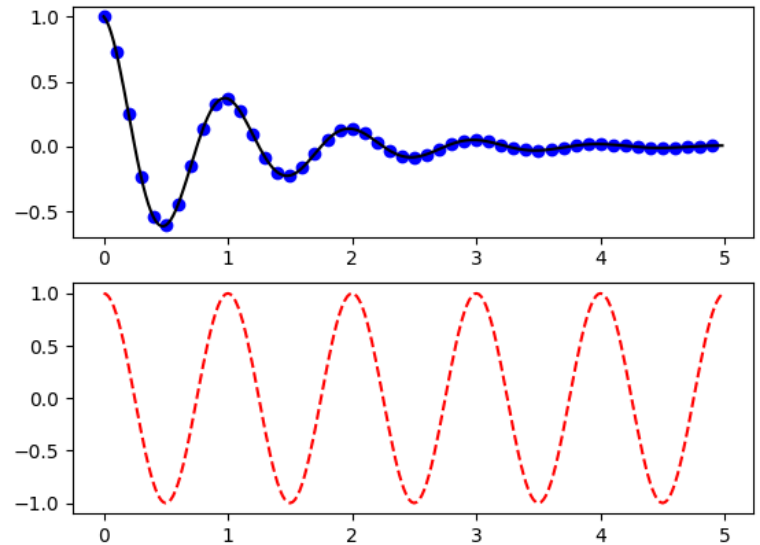


# Comando subplot

- Especifica num\_filas, num\_columnas, numplot
- Se puede escribir subplot(211) ó subplot(2,1,1)

# Ejemplo

```
def f(t):  
    return np.exp(-t) * np.cos(2*np.pi*t)  
t1 = np.arange(0.0, 5.0, 0.1)  
t2 = np.arange(0.0, 5.0, 0.02)  
plt.figure(1)  
plt.subplot(211)  
plt.plot(t1, f(t1), 'bo', t2, f(t2), 'k')  
plt.subplot(212)  
plt.plot(t2, np.cos(2*np.pi*t2), 'r--')  
plt.show()
```





# Manejo de varias figuras y subplots

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

```
plt.figure(1) # la primera figura
```

```
plt.subplot(211) # el primer subplot en la primera figura
```

```
plt.plot([1, 2, 3])
```

```
plt.subplot(212) # el segundo subplot en la primera figura
```

```
plt.plot([4, 5, 6])
```

```
plt.figure(2) # Una segunda figura
```

```
plt.plot([4, 5, 6]) # crea un subplot (111) por defecto
```

```
plt.figure(1) # figura 1 es la actual; subplot(212) aun es el actual
```

```
plt.subplot(211) # crea subplot(211) en figura 1 actual
```

```
plt.title('Tan fácil como 1, 2, 3') # título del subplot 211
```

# Bibliografía

- <https://matplotlib.org/tutorials/introductory/pyplot.html#sphx-glr-tutorials-introductory-pyplot-py>