

Leyendas, etiquetas, títulos

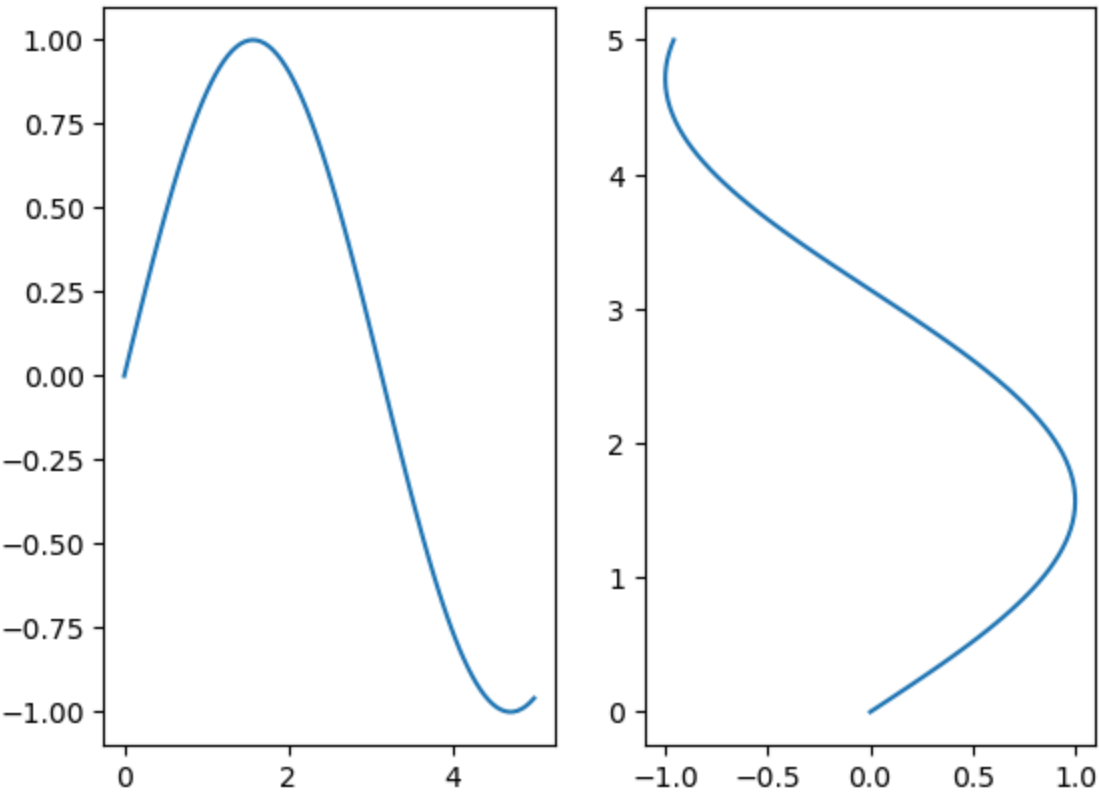
Son usados para darle contexto a nuestros gráficos

```
In [ ]: import matplotlib.pyplot as plt
import numpy as np
```

```
In [ ]: #Declarando x
x = np.linspace(0,5,100)
#Declarando y
y = np.sin(x)
```

```
In [ ]: #Graficando
fig, axes = plt.subplots(1,2)
axes[0].plot(x,y)
axes[1].plot(y,x)
```

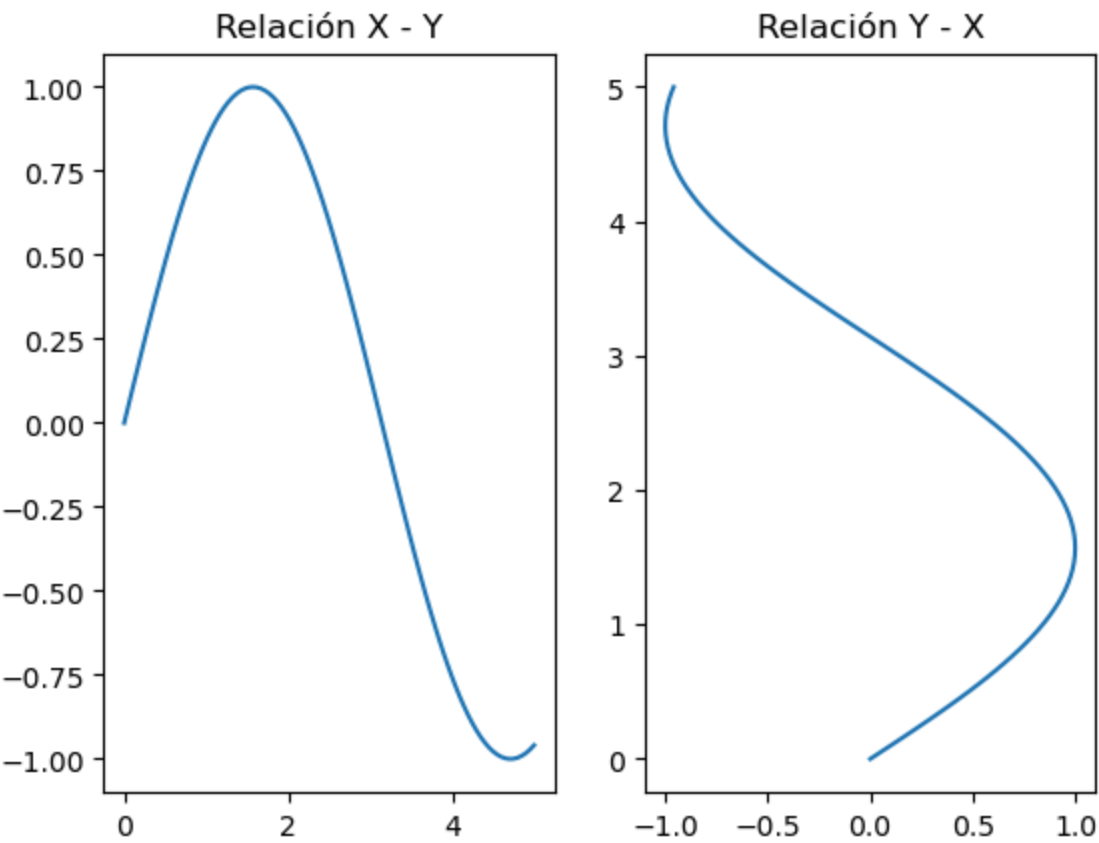
Out[]: [<matplotlib.lines.Line2D at 0x7f8fca47c890>]



Son gráficas sin sentido. Empezaremos a darle contexto

```
In [ ]: #Graficando
fig, axes = plt.subplots(1,2)
axes[0].plot(x,y)
axes[0].set_title('Relación X - Y')
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
```

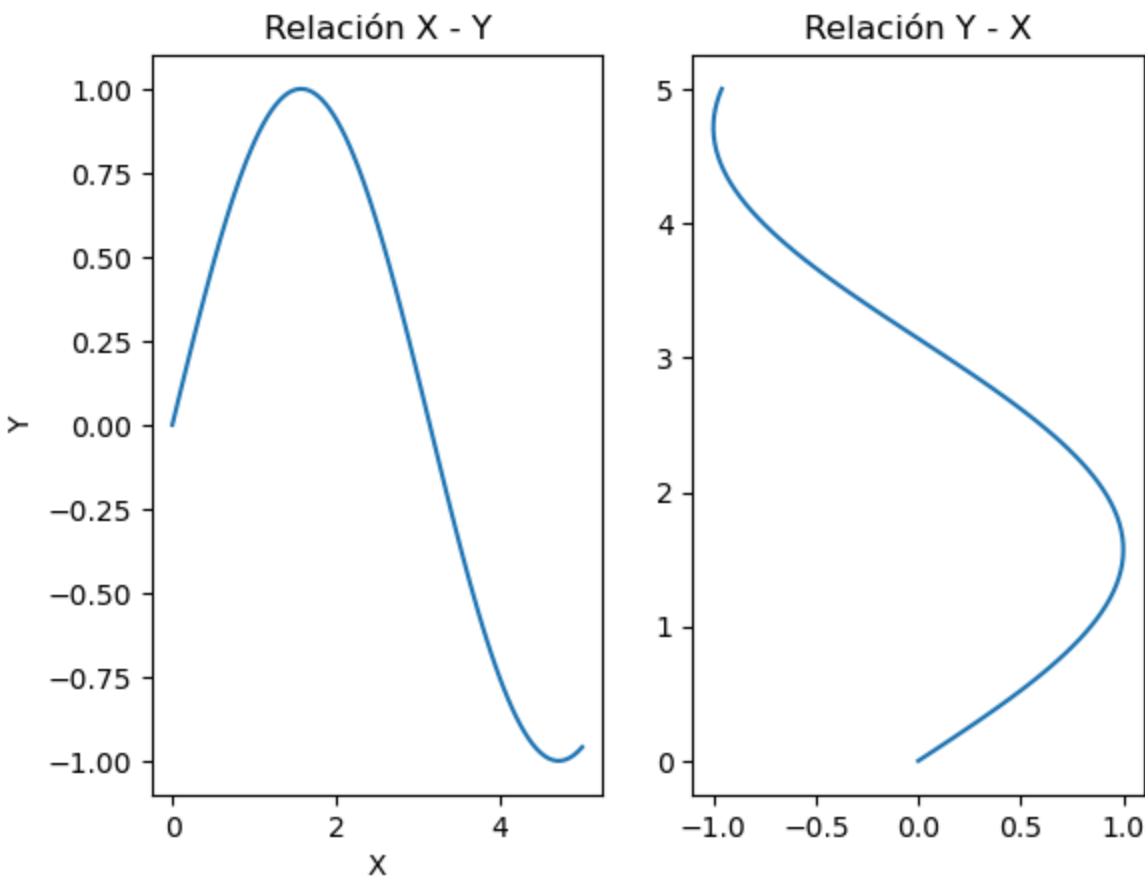
Out[]: Text(0.5, 1.0, 'Relación Y - X')



```
In [ ]: #Explotando mas opciones
#Graficando
fig, axes = plt.subplots(1,2)
axes[0].plot(x,y)
```

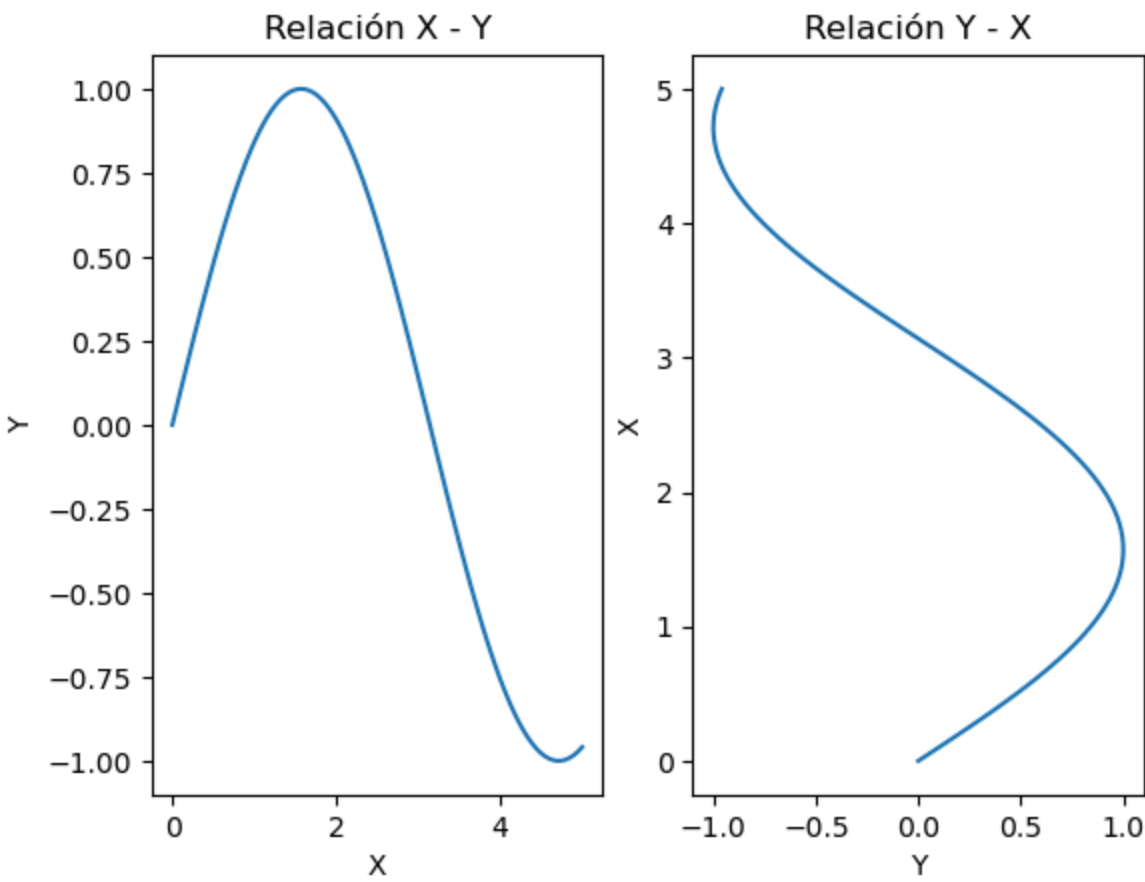
```
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
```

Out[]: Text(0.5, 1.0, 'Relación Y - X')



```
In [ ]: #Explotando mas opciones
#Graficando
#Creando un lienzo
fig, axes = plt.subplots(1,2)
#Grafica 1
axes[0].plot(x,y)
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')
#Grafica 2
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
axes[1].set_xlabel('Y')
axes[1].set_ylabel('X')
```

Out[]: Text(0, 0.5, 'X')



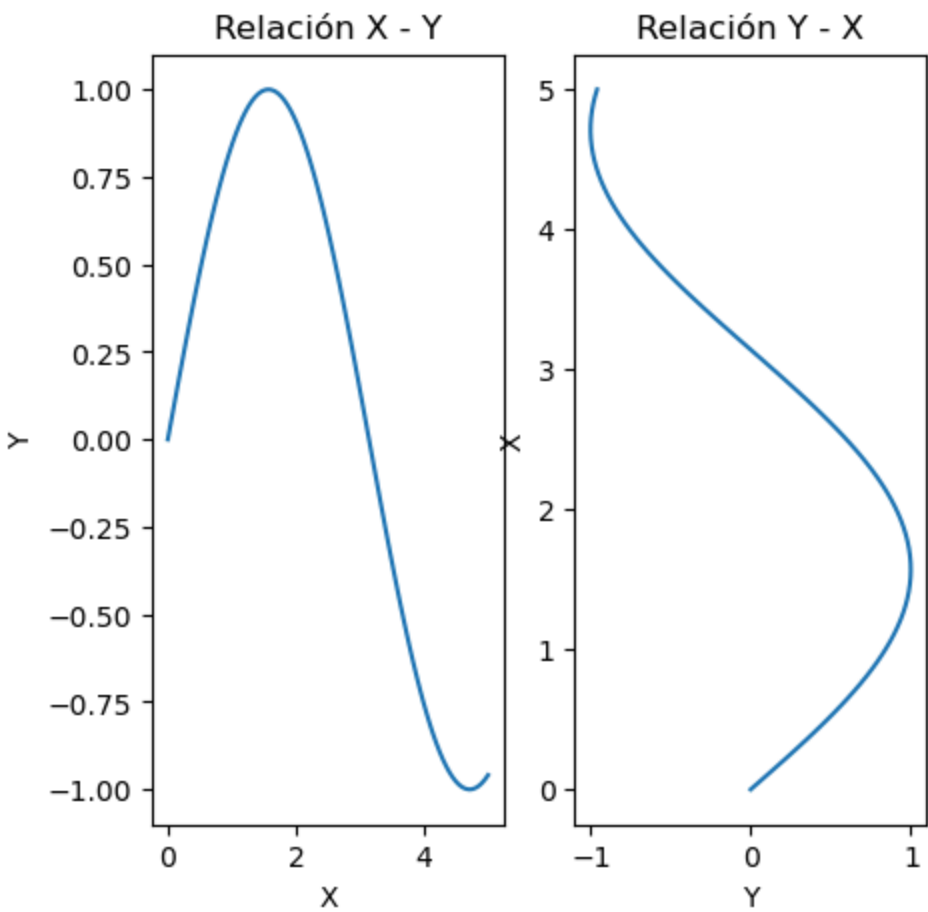
```
In [ ]: #Cambiando tamaño de figura

#Creando un lienzo
fig, axes = plt.subplots(1,2,figsize=(5,5))

#Grafica 1
axes[0].plot(x,y)
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')
```

```
#Grafica 2
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
axes[1].set_xlabel('Y')
axes[1].set_ylabel('X')
```

Out[]: Text(0, 0.5, 'X')



Cambiando tamaño

`fig, axes = plt.subplots(1,2,figsize=(5,5))`
El método usa `figsize=(x,y)` para cambiar el tamaño. En donde `x` cambia el tamaño horizontal y `y` cambia el tamaño vertical.

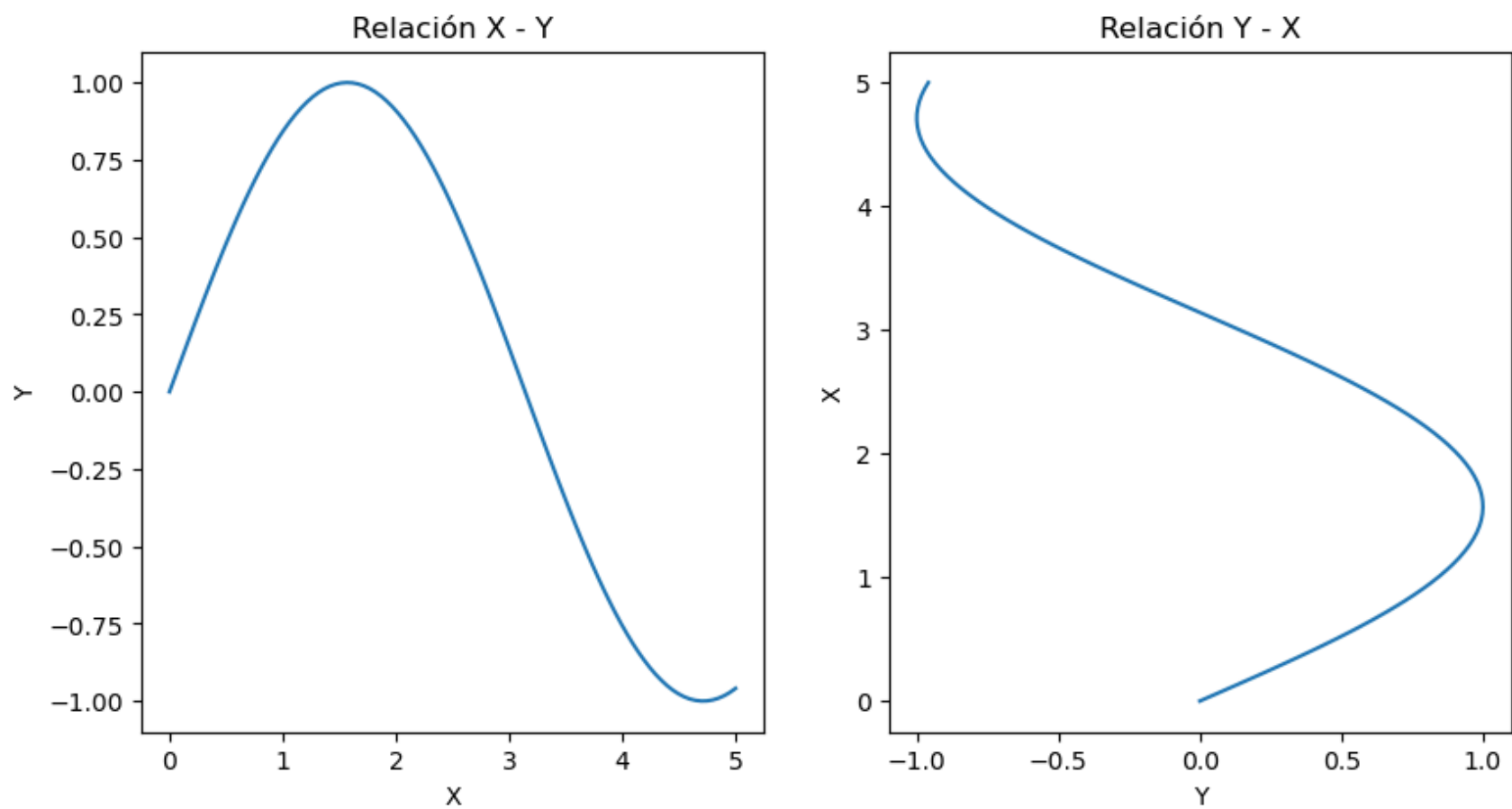
```
In [ ]: #Cambiando tamaño de figura

#Creando un lienzo
fig, axes = plt.subplots(1,2,figsize=(10,5))

#Grafica 1
axes[0].plot(x,y)
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')

#Grafica 2
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
axes[1].set_xlabel('Y')
axes[1].set_ylabel('X')
```

Out[]: Text(0, 0.5, 'X')



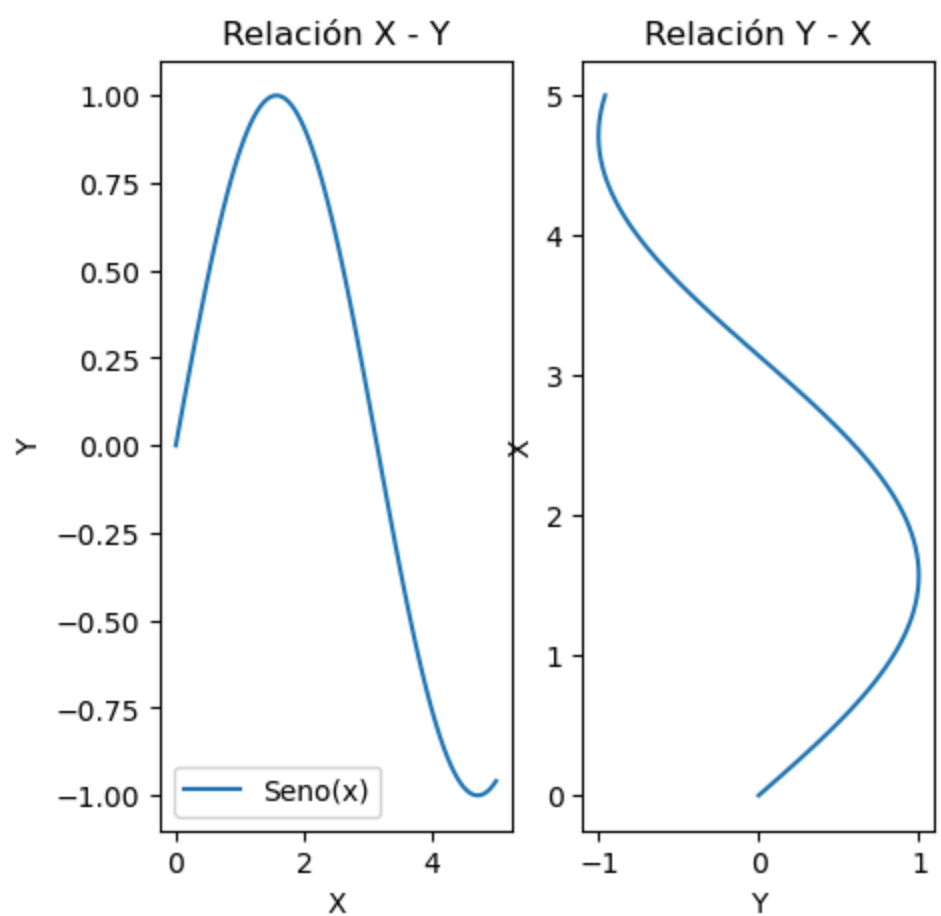
```
In [ ]: #Etiquetas
#Cambiando tamaño de figura

#Creando un lienzo
fig, axes = plt.subplots(1,2,figsize=(5,5))

#Grafica 1
axes[0].plot(x,y,label='Seno(x)')
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')
axes[0].legend()

#Grafica 2
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
axes[1].set_xlabel('Y')
axes[1].set_ylabel('X')
```

Out[]: Text(0, 0.5, 'X')



Labels - Etiquetas

Nos sirven para explicar lo que estamos haciendo. Mediante:

```
axes[0].plot(x,y,label='Seno(x)')
.
.
.
axes[0].legend()
```

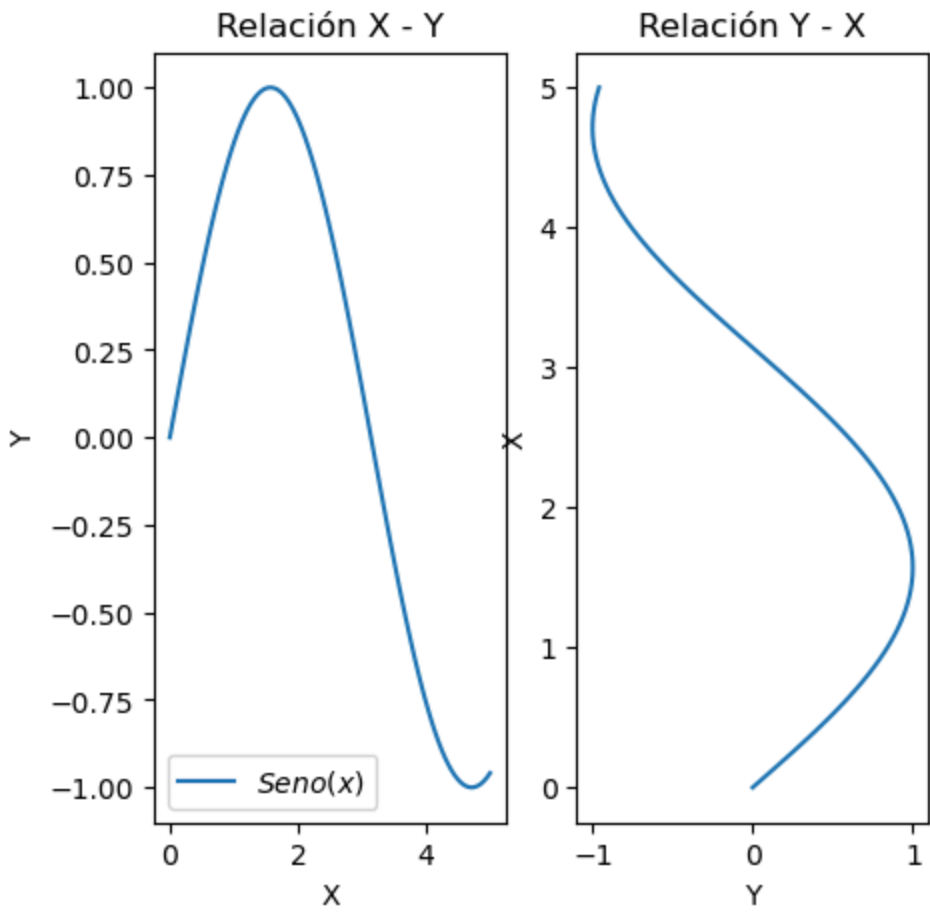
```
In [ ]: #Otra forma con $
#Etiquetas
#Cambiando tamaño de figura

#Creando un lienzo
fig, axes = plt.subplots(1,2,figsize=(5,5))

#Grafica 1
axes[0].plot(x,y,label='$Seno(x)$')
axes[0].set_title('Relación X - Y')
axes[0].set_xlabel('X')
axes[0].set_ylabel('Y')
axes[0].legend()

#Grafica 2
axes[1].plot(y,x)
axes[1].set_title('Relación Y - X')
axes[1].set_xlabel('Y')
axes[1].set_ylabel('X')
```

Out[]: Text(0, 0.5, 'X')



Nota:

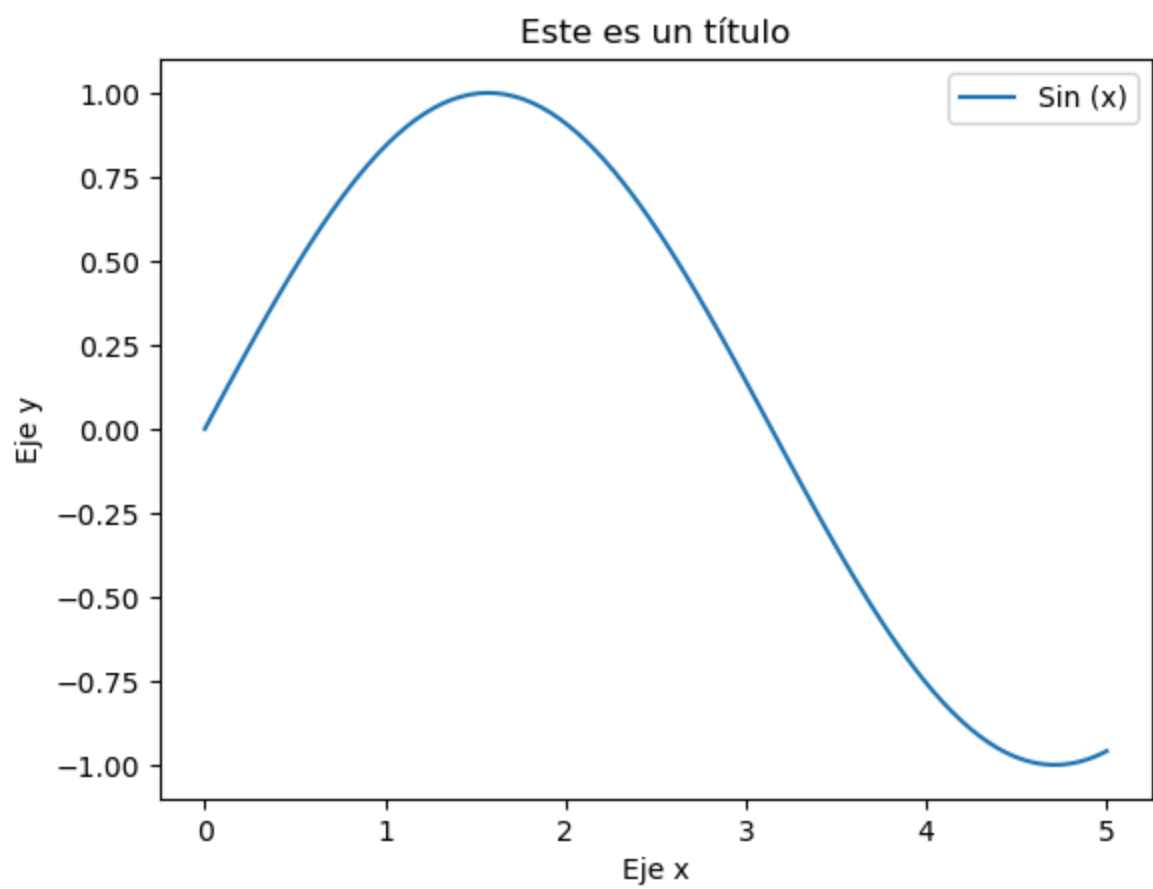
si ponemos `$` dentro del label, nos ayuda a visualizarlo como notación matematica `axes[0].plot(x,y,label='$Seno(x)$')`

Otra forma de graficar

Sin el método orientado a objetos

```
In [ ]: plt.plot(x,y,label='Sin (x)')
plt.title('Este es un título')
plt.xlabel('Eje x')
plt.ylabel('Eje y')
plt.legend()
```

Out[]: <matplotlib.legend.Legend at 0x7f8fc94a6420>

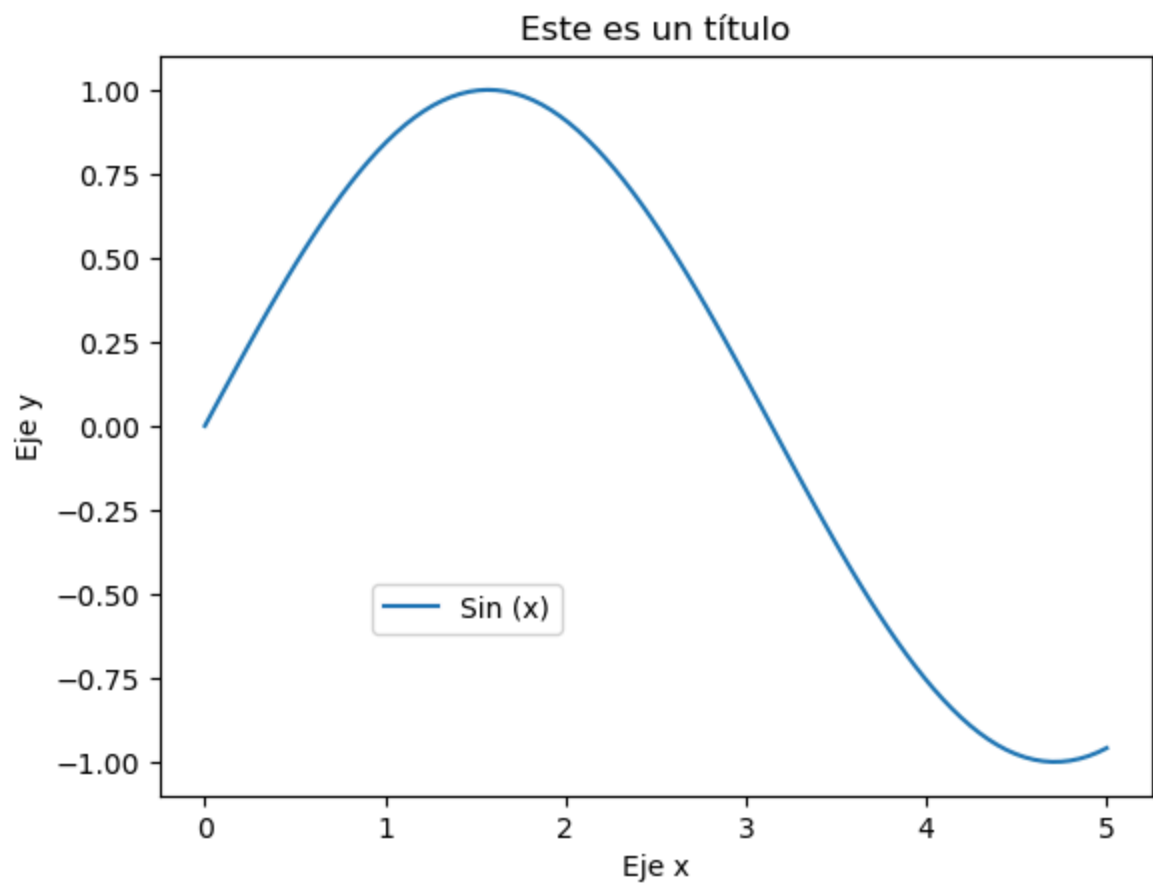


También

Podemos decirle conde se ubique el legend, que tamaño

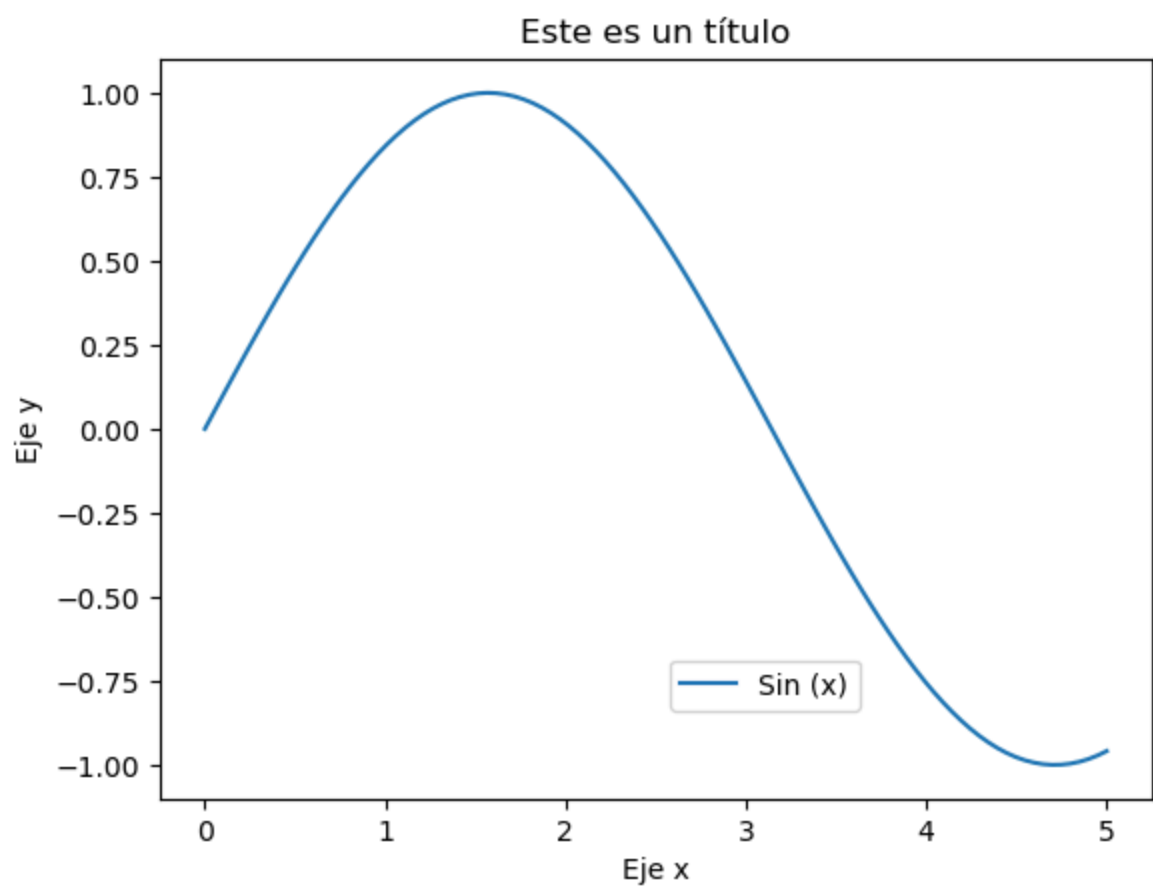
```
In [ ]: plt.plot(x,y,label='Sin (x)')
plt.title('Este es un título')
plt.xlabel('Eje x')
plt.ylabel('Eje y')
plt.legend(loc='lower left',bbox_to_anchor=(0.2,0.2))
```

```
Out[ ]: <matplotlib.legend.Legend at 0x7f8fc91addc0>
```



```
In [ ]: plt.plot(x,y,label='Sin (x)')
plt.title('Este es un título')
plt.xlabel('Eje x')
plt.ylabel('Eje y')
plt.legend(loc='lower left',bbox_to_anchor=(0.5,0.1))
```

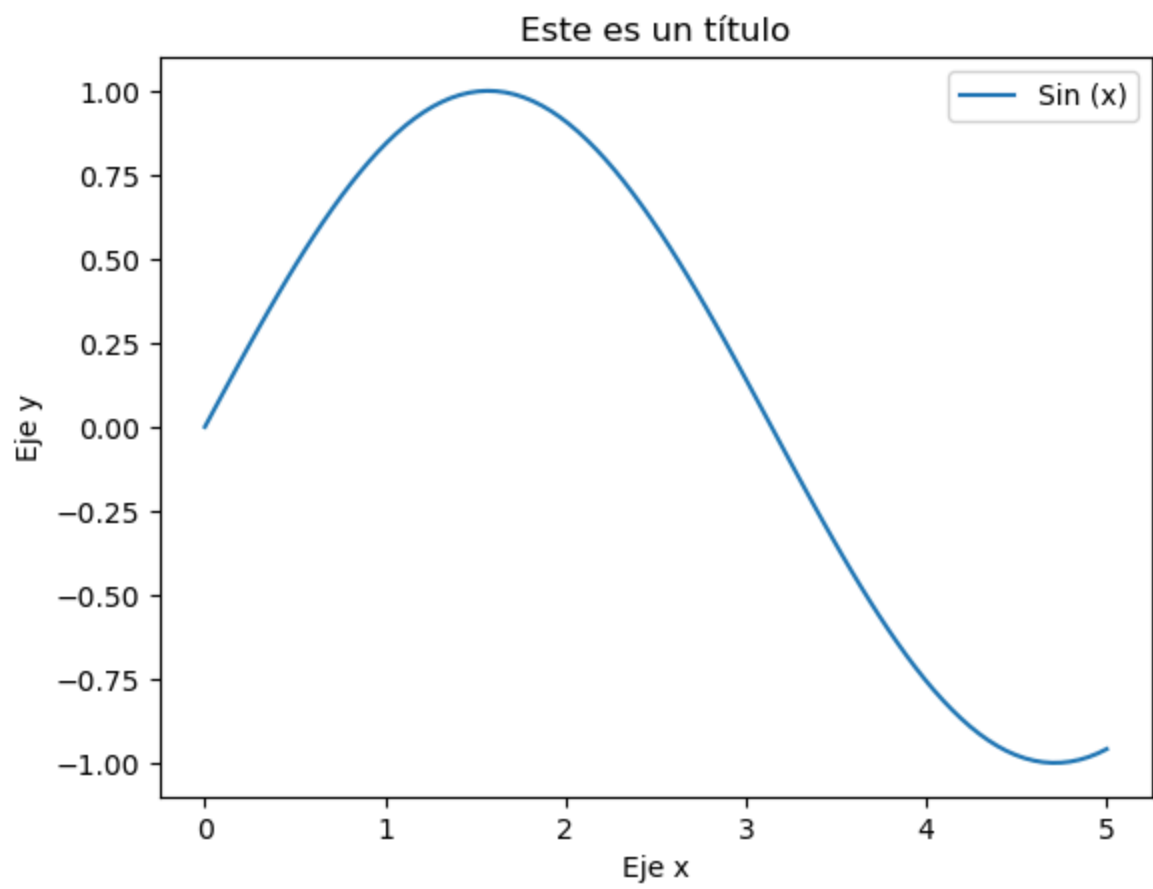
```
Out[ ]: <matplotlib.legend.Legend at 0x7f8fc93179b0>
```



Por default los labels los acomoda donde parezcan mas apropiados para ver.

```
In [ ]: plt.plot(x,y,label='Sin (x)')
plt.title('Este es un título')
plt.xlabel('Eje x')
plt.ylabel('Eje y')
plt.legend()
```

```
Out[ ]: <matplotlib.legend.Legend at 0x7f8fc9228bf0>
```



Referencias:

- [Matplotlib legend](#)
- [LaTeX en Matplot](#)