

unid2_clas4

May 10, 2019

```
#
Unidad 2, Clase 4
##
Mauricio Suárez DuránUnidad 2, Clase 4Introducción a Python
###
Departamento de Física y GeologíaUniversidad de PamplonaI Semestre, 2019
```

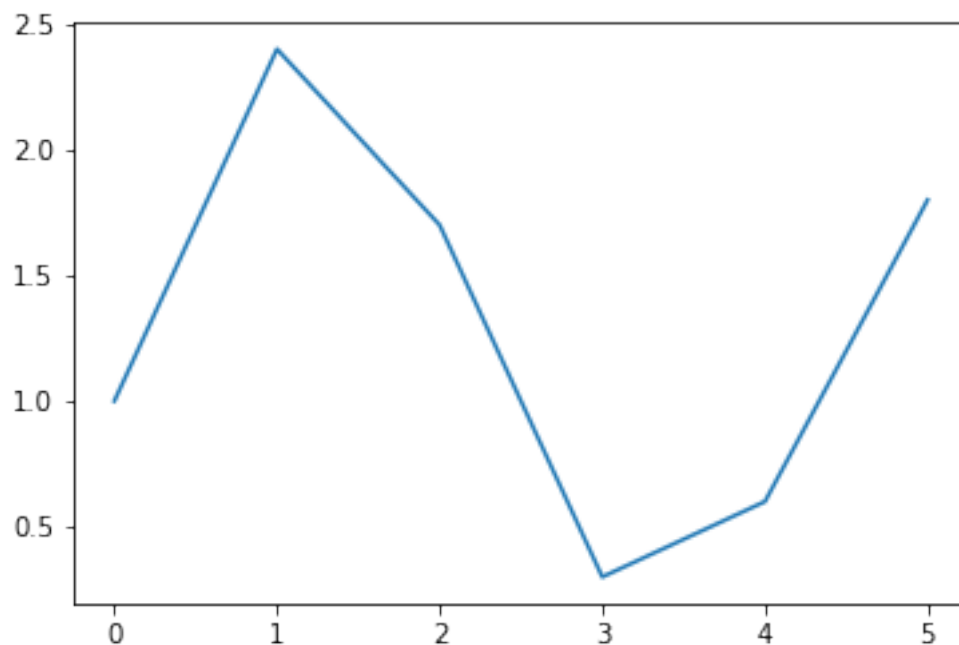
0.1 Gráficos:

- Vamos usar Pylab, un paquete que hace parte de uno más grande llamado matplotlib.
- Dos tipos de gráficos: lineales, dispersión, y gráficos de densidad.

```
In [1]: from pylab import plot, show
```

```
y = [1., 2.4, 1.7, 0.3, 0.6, 1.8]
```

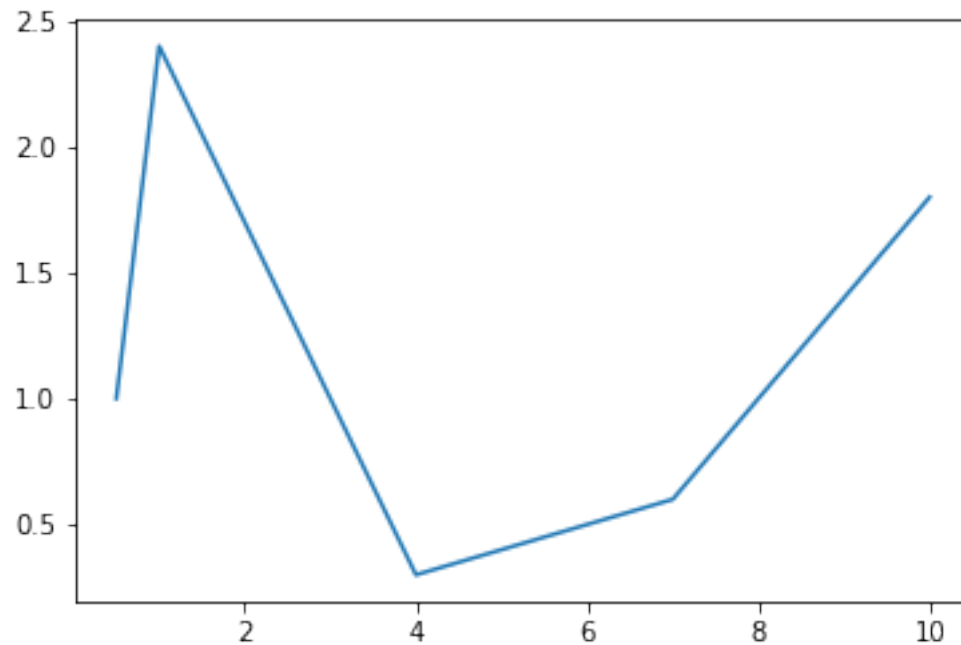
```
plot(y)
show()
```



En este caso Python asignó la escala del eje X. Para especificar esta escala:

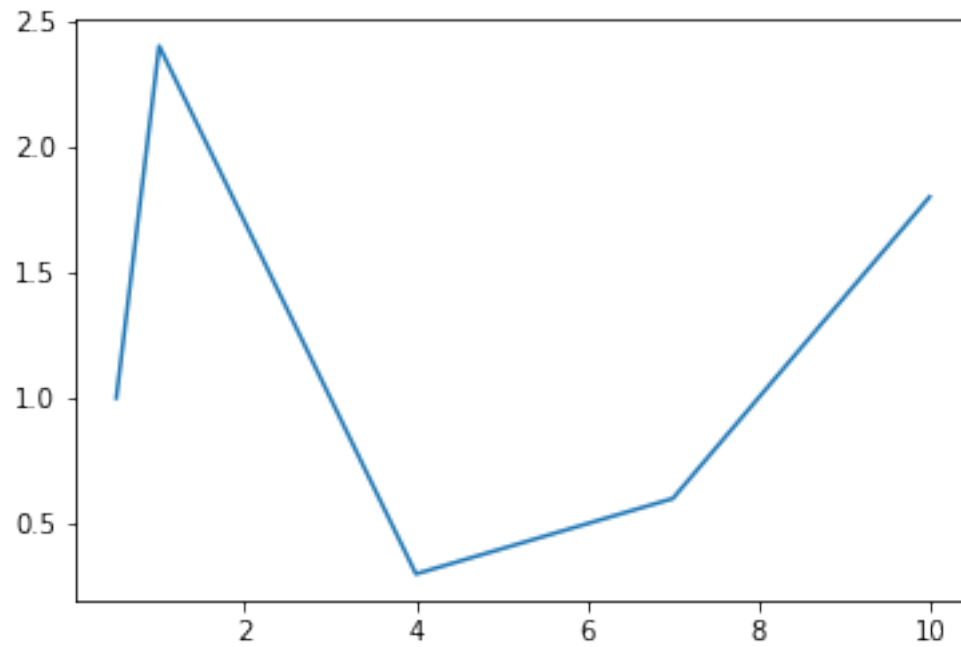
```
In [2]: x = [0.5, 1., 2., 4., 7., 10.]
```

```
plot(x,y)  
show()
```



Con la opción plot() podemos insertar varios plot en el mismo gráfico. Para salvar el gráfico en un archivo:

```
In [3]: from pylab import savefig  
plot(x, y)  
savefig('destination_path.eps', format='eps', dpi=1000)  
show()
```



Para graficar una función conocida, por ejemplo la función Sin:

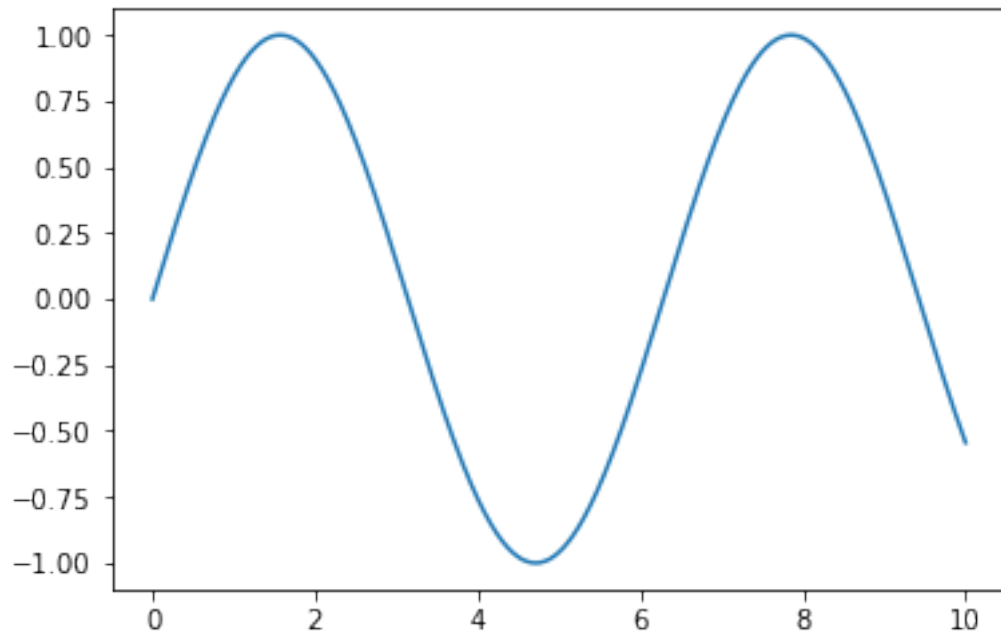
```
In [4]: from numpy import linspace, sin, cos
```

```
x = linspace(0, 10, 1000)
```

```
y = sin(x)
```

```
plot(x,y)
```

```
show()
```

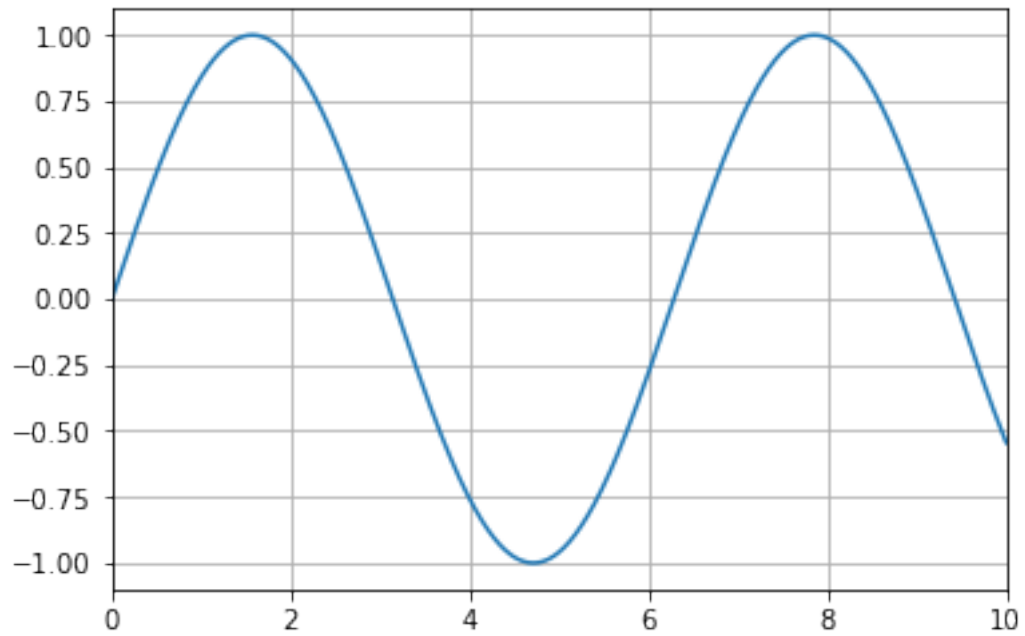


Note que en este caso, la función `sin` actuó sobre cada elemento del array `x`. De manera equivalente a `map(sin, x)`

0.2 La escala de los ejes se puede modificar

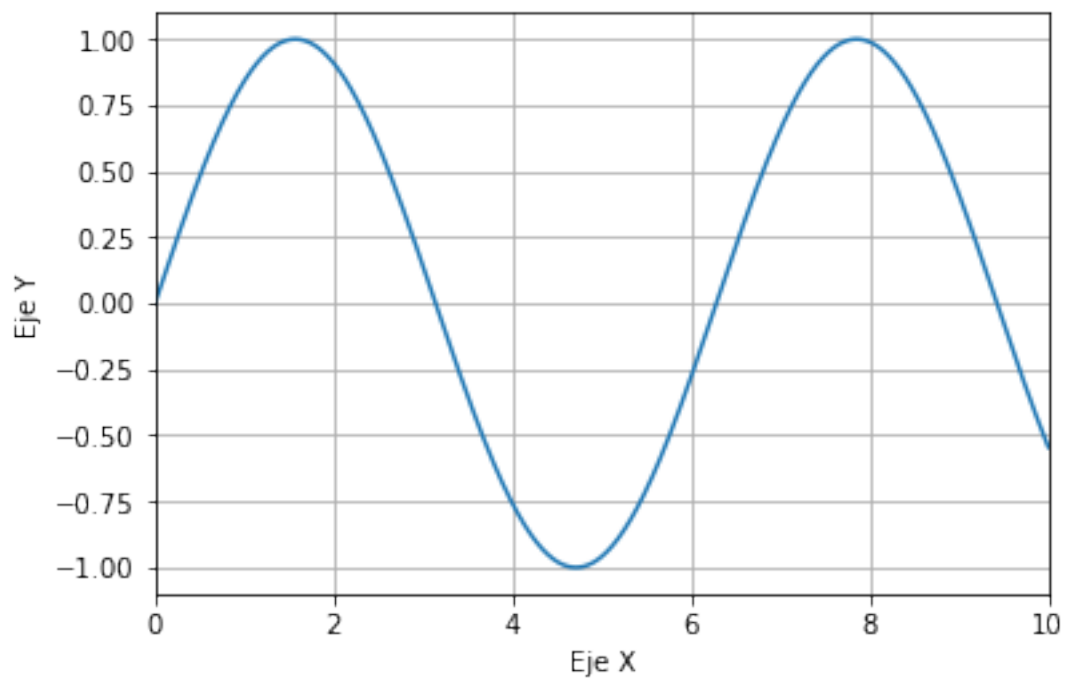
```
In [5]: from pylab import ylim, xlim, grid
        x = linspace(0, 10, 1000)
        y = sin(x)

        plot(x,y)
        ylim(-1.1, 1.1)
        xlim(0., 10)
        grid("on")
        show()
```



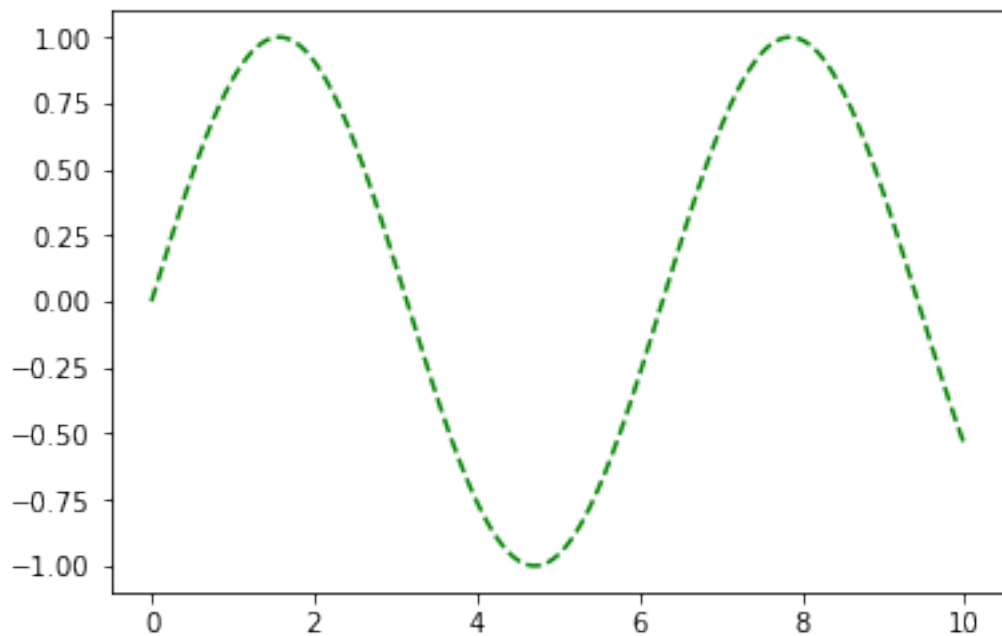
- También podemos insertar las etiquetas de los ejes

```
In [6]: from pylab import xlabel, ylabel
        plot(x,y)
        ylim(-1.1, 1.1)
        xlim(0., 10)
        grid("on")
        xlabel("Eje X")
        ylabel("Eje Y")
        show()
```



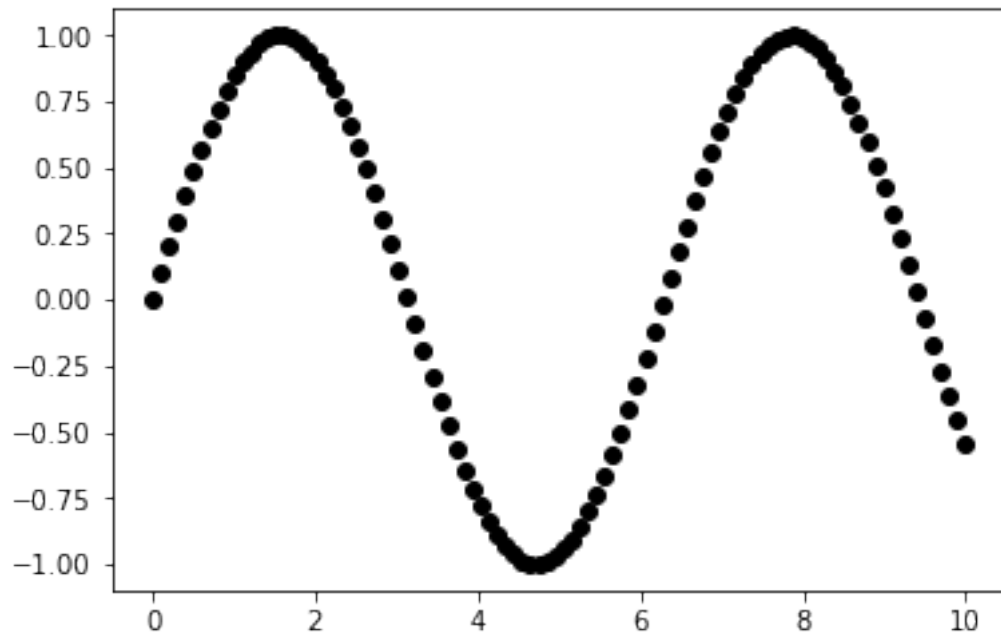
0.3 Diferentes formas de graficar

In [7]: `plot(x,y,'g--')`
`show()`

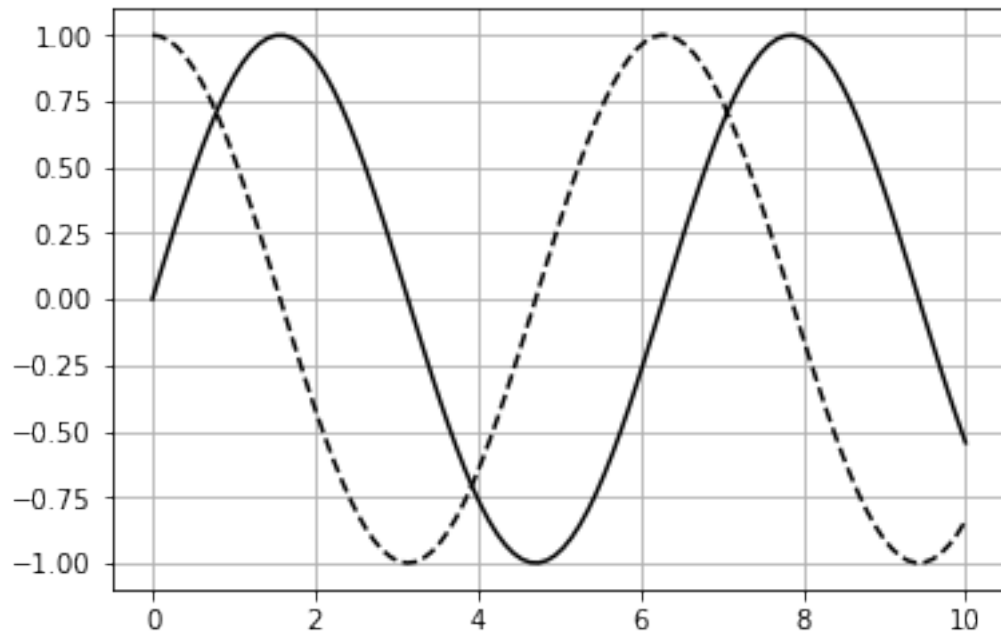


```
In [8]: x = linspace(0, 10, 100)
        y = sin(x)
```

```
plot(x,y, 'ko')
show()
```



```
In [9]: y = sin(x)
        plot(x,y, 'k-')
        y = cos(x)
        plot(x,y, 'k--')
        grid("on")
        show()
```



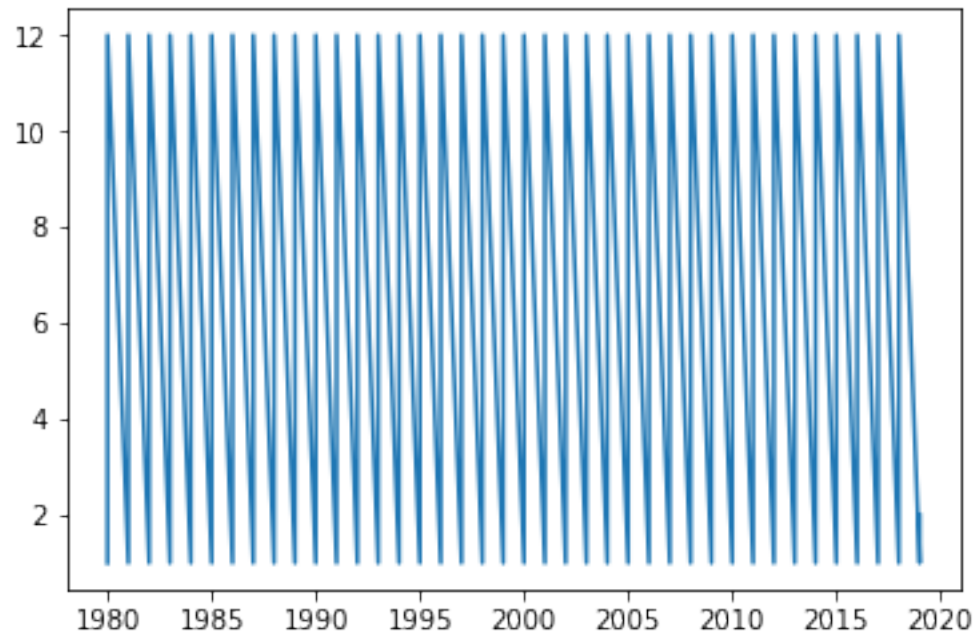
0.4 Ejercicio: Graficando datos desde un archivo

Ir a la página: https://www.esrl.noaa.gov/gmd/ccgg/trends/gl_full.html y descargar el registro histórico de la tendencia de CO2 en la atmósfera

```
In [10]: from numpy import loadtxt

dataC02 = loadtxt("../data/co2_mm_gl.txt", float)

plot(dataC02[:,0], dataC02[:,1])
show()
```

```
In [11]: plot(dataC02[:,3], dataC02[:,4])  
show()
```

