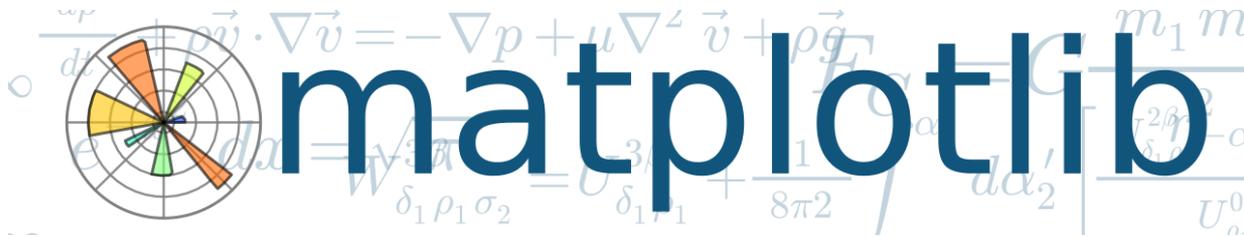


Creación de graficas en Python

Jairo Bernardo Viola Villamizar

Lógica y algoritmia

Introducción

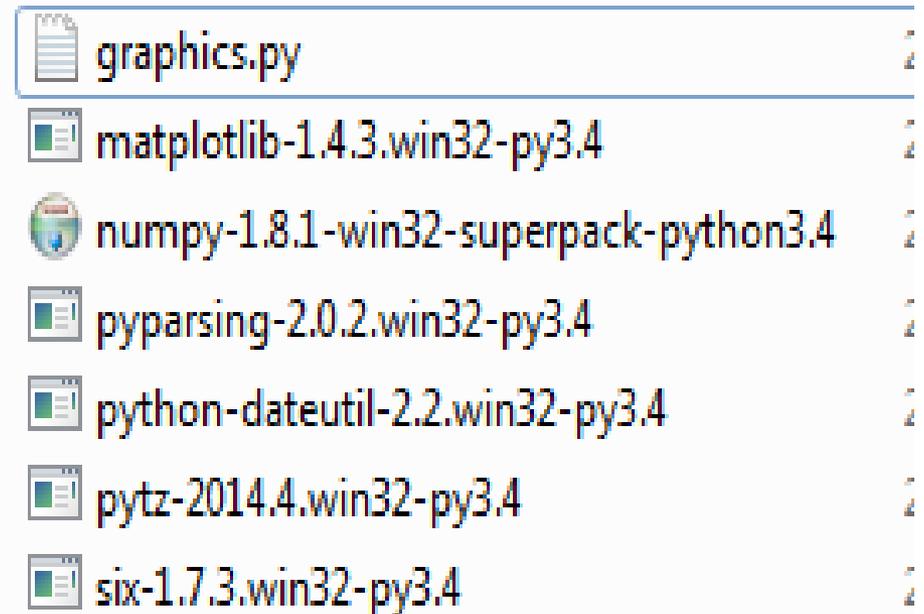


- Librería de código abierto para la creación de gráficos en Python
 - Interfaz de objeto y estilo MATLAB
 - open source
 - Disponible para Python 3.4
 - Soporte para edición de gráficos con texto en latex

Instalar matplotlib

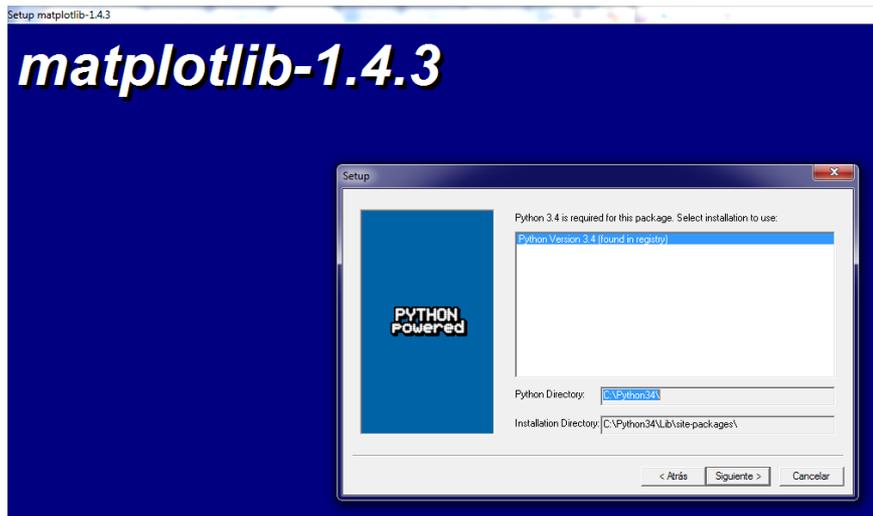
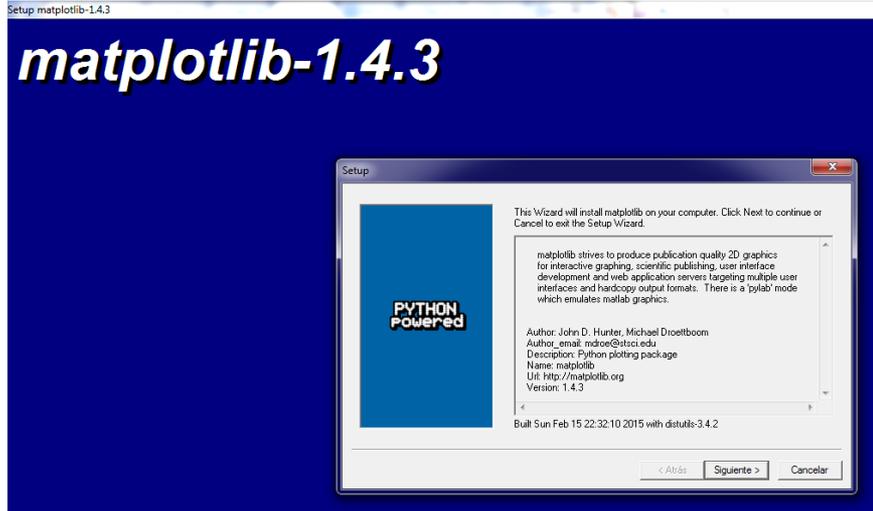
- Librerías requeridas (Python 3.4):

- Matplotlib 1.4.3
- Numpy 1.81
- Pyparsing 2.02
- Python dateutil
- Pytz
- six



<http://matplotlib.org/index.html>

Instalar matplotlib



Tomar cada uno de los paquetes anteriormente mencionados e instalarlos mediante el asistente

Es similar para todos los archivos

Modos de operación de matplotlib

MATLAB

- Comandos similares a Matlab
- Uso mas simple
- Restricciones de modificación de propiedades

OBJETO

- Mayor cantidad de código
- Control total sobre los objetos

Modo MATLAB

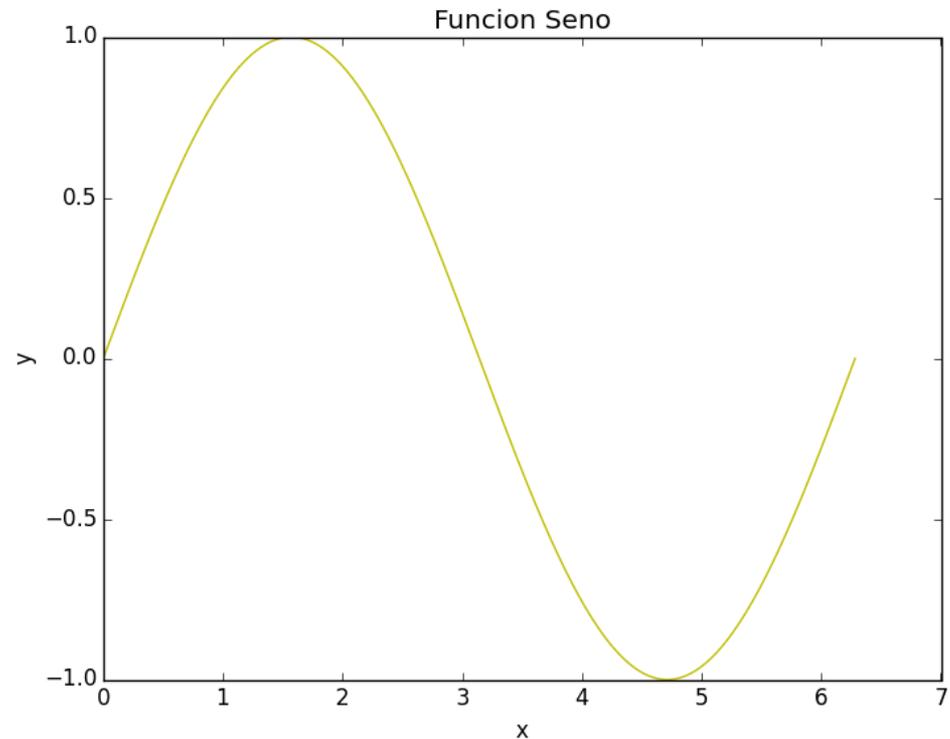
```
#librerias classicas
import random
import math
#librerias de graicas
from matplotlib.pyplot import *
from numpy import *

#inicializacion del vector
vec1=[0 for i in range(360)]
time=linspace(0,360,360)      #ge
time=radians(time)          #conv

#llenado del vector
for i in range(360):
    vec1[i]=math.sin(time[i])

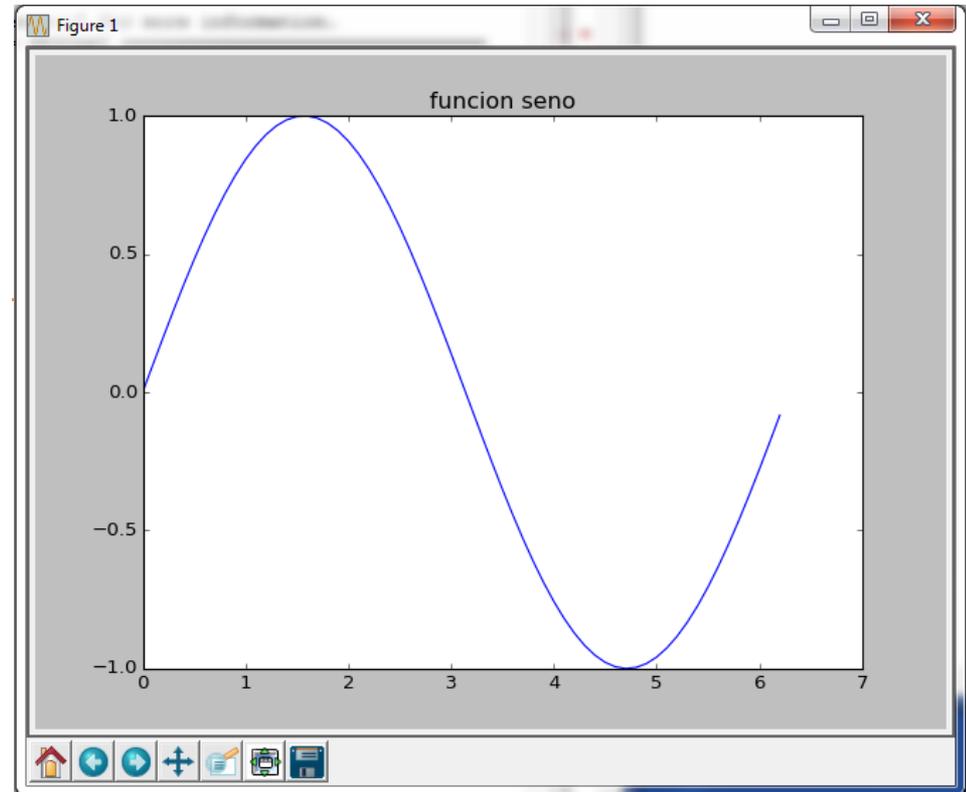
plot(time,vec1,'y')
xlabel('x')
ylabel('y')
title(u'Funcion Seno') # el carac
show() # muestra el gráfico en ur

print(x)
```



Modo Objeto

```
#trabajar una grafica como objeto  
  
import numpy as np  
import matplotlib.pyplot as plt  
  
x = np.arange(0, 6.28, 0.1); #crea  
y = np.sin(x)  
  
plt.plot(x, y)  
plt.title('funcion seno')  
plt.show()
```



Subplot

```
#trabajar una grafica como objeto

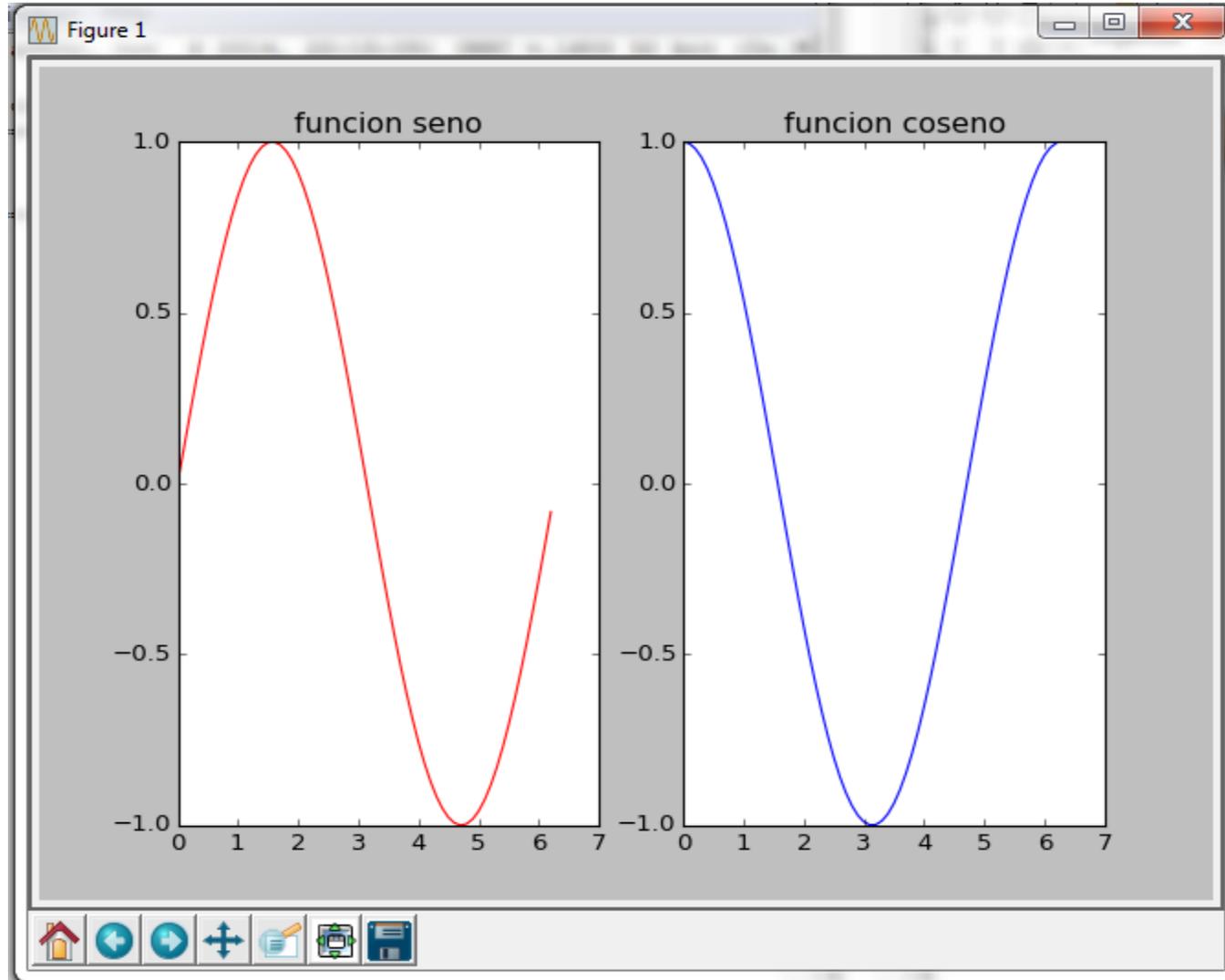
import numpy as np
import matplotlib.pyplot as plt

x = np.arange(0, 6.28, 0.1); #crear array
y = np.sin(x)
z = np.cos(x)

plt.subplot(1,2,1)
plt.plot(x, y, 'r')
plt.title('funcion seno')
plt.subplot(1,2,2)
plt.plot(x, z, 'b')
plt.title('funcion coseno')

plt.show()
```

Subplot



Graficar datos desde fichero

```

#trabajar una grafica como objeto
import math
import numpy as np
import matplotlib.pyplot as plt

#funcion para leer un fichero

def  abrir_fichero(nombre):

    crs = open(nombre, "r")                #abrir el archivo
    columna1=[]                            #vector donde voy
    columna2=[]                            #vector donde voy
    columna3=[]                            #vector donde voy
    #saco los datos de cada columna y los guardo en vectores
    for columns in ( raw.strip().split() for raw in crs ):
        columna1.append(float(columns[0]))    #empleando float
        columna2.append(float(columns[1]))
        columna3.append(float(columns[2]))
    #cierro el fichero para no sobrescribirlo
    crs.close()

    datos=[columna1,columna2,columna3]      #concateno los datos

    return datos
    
```

```

#programa principal
#extraer informacion del fichero de datos
info=abrir_fichero("datos_test1.txt")
    
```

```

#programa principal
#extraer informacion del fichero de datos
info=abrir_fichero("datos_test1.txt")

w=info[0]
y=info[1]
z=info[2]
x = np.arange(0, len(w), 1);
    
```

```

plt.figure(1)
plt.subplot(2,2,1)
plt.plot(x, w, 'r')
plt.title('data 1')
plt.subplot(2,2,2)
plt.plot(x, y, 'b')
plt.title('data 2')
plt.subplot(2,2,3)
plt.plot(x, z, 'g')
plt.title('data 3')
    
```

```

plt.figure(2)
plt.plot(x,w,'r',x,y,'b',x,z,'g')
plt.title('combinados')
    
```

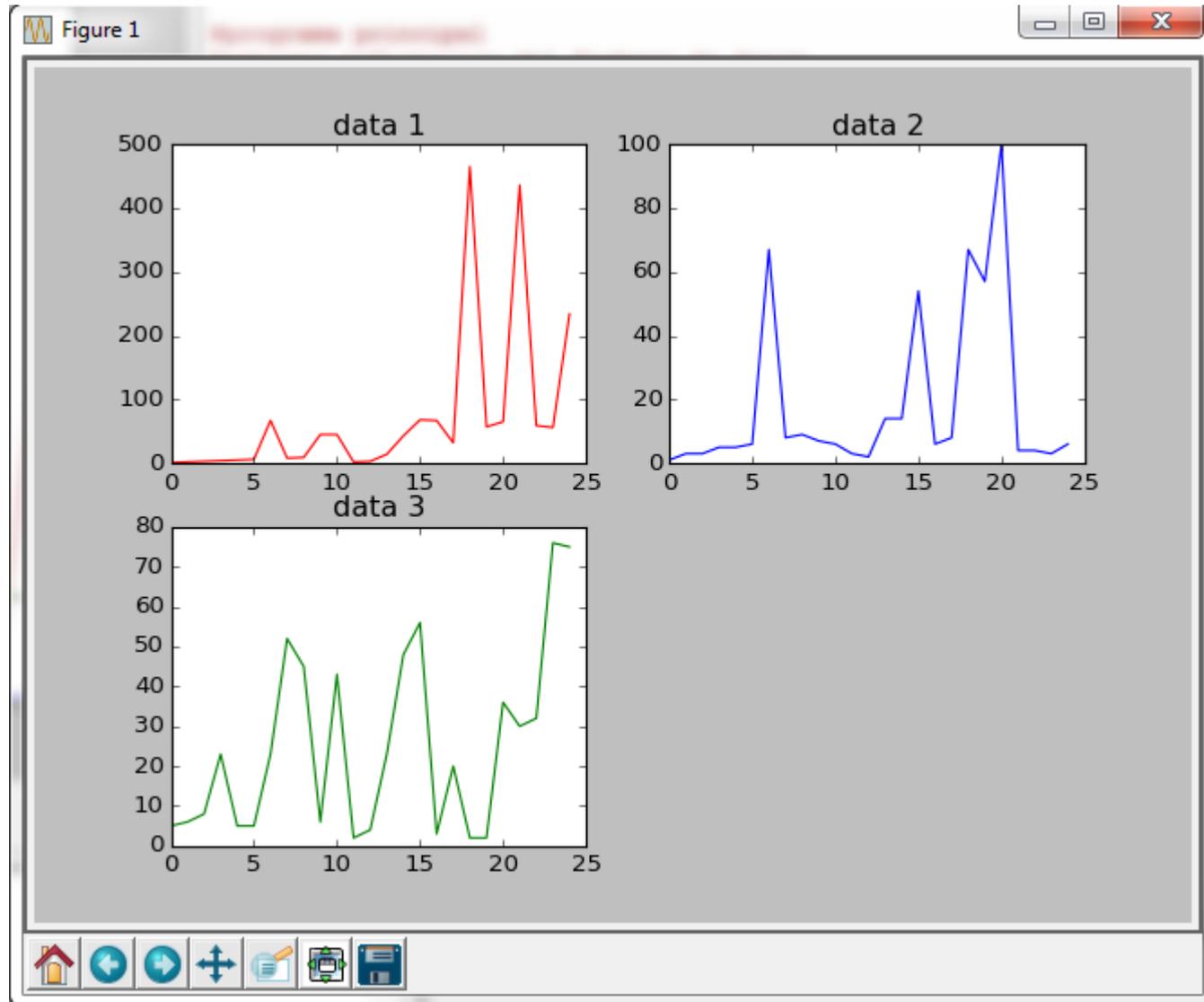
```

plt.savefig("grafica.png")
plt.savefig("grafica.pdf")
    
```

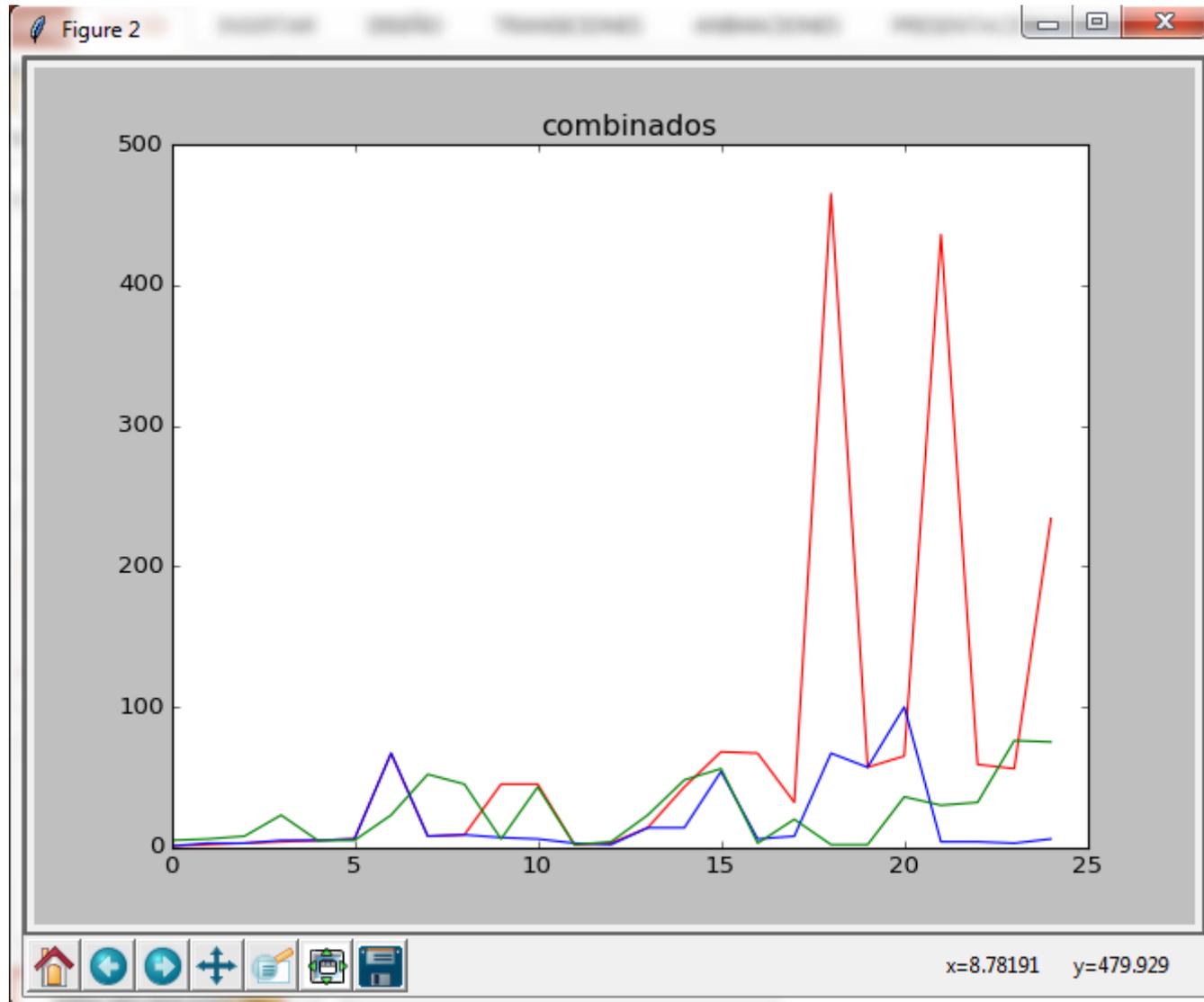
```

plt.show()
    
```

Graficar datos desde fichero



Graficar datos desde fichero



Guardar la figura

Formatos soportados

PNG

SVG

PDF

EPS

RGBA

RAW