

INGENIERÍA DE EJECUCIÓN EN INFORMÁTICA MENCIÓN DESARROLLO DE SISTEMAS - PEV - BELLAVISTA

**LIBRERÍA PYTHON**

**Autores**

Carlos Alexander Andrade Lavin

Carlos Luis Chourio Ortiz

Rodolfo Esteban Hernández Ramos

**Docente**

Segundo Galdames Henriquez

**Introducción a Matplotlib**

matplotlib es una biblioteca de gráficos 2D de Python que produce cifras de calidad de publicación en una variedad de formatos impresos y entornos interactivos en todas las plataformas. matplotlib se puede utilizar en scripts de python, el shell de python e ipython (como MATLAB®\* o Mathematica®), servidores de aplicaciones web y seis kits de herramientas de interfaz gráfica de usuario. matplotlib intenta hacer que las cosas fáciles sean fáciles y las difíciles sean posibles. Puede generar diagramas, histogramas, espectros de potencia, gráficos de barras, gráficos de error, diagramas de dispersión, etc., con solo unas pocas líneas de código.

En este tutorial, vamos a enseñarte a instalar la librería matplotlib, y después veremos unos cuantos ejemplos de uso.

**Instalando matplotlib**

Instalar matplotlib es muy simple. Actualmente estoy trabajndo en una máquina Mac OS X, por lo tanto te enseñaré cómo instalar la librería en dicho sistema operativo. Por favor, échale un vistazo a la [**página de instalación de matplotlib**](https://matplotlib.org/users/installing.html) para más información sobre cómo instalar matplotlib en otros sistemas operativos.

Matplotlib puede ser instalada ejecutando los siguientes comandos en tu terminal. Yo utilizaré **[pip](https://pypi.python.org/pypi/pip" \t "_blank)**, pero tú puedes utilizar otras herramientas.

curl - O https : //bootstrap.pypa.io/get-pip.py   
python get - pip . py   
pip instalar matplotlib

Esto es todo. Ya tienes matplotlib listo y funcionando. ¿A que ha sido muy fácil?

**Dibujando gráficos simples**

Ahora vamos a ver cómo funciona matplotlib con algunos ejemplos. Vamos a ver cómo dibujar ciertos gráficos con la librería.

**Gráfico de líneas**

Vamos a crear un gráfico de líneas utilizando la librería matplotlib. En este caso, vamos a utilizar matplotlib.pyplot, que nos provee de un framework de gráficos parecido a MATLAB. En otras palabras, nos proporciona una colección de funciones que permiten que matplotlib funcione como MATLAB.

Digamos que queremos dibujar una línea en el siguiente conjunto de puntos:

x =(4,8,13,17,20)y =(54,67,98,78,45)

Esto puede realizarse utilizando el siguiente script.

importar matplotlib . pyplot como plt   
plt . trama ([ 4 , 8 , 13 , 17 , 20 ],[ 54 , 67 , 98 , 78 , 45 ]) plt . mostrar ()

Date cuenta que representamos los puntos x e y como listas.

En este caso, el resultado será el siguiente:

Gráfico, Gráfico de líneas

Descripción generada automáticamente

La línea de la imagen de arriba es la línea que se genera por defecto, en términos de forma y color. Podemos personalizarla cambiando el color y la forma de la línea utilizando ciertos símbolos (especificadores) que puedes obtener de la [**documentación de MATLAB**](https://www.mathworks.com/help/matlab/ref/plot.html?s_tid=gn_loc_drop). Digamos que quieres pintar una línea verde punteada, con marcadores de diamante. Estos especificadores, en este caso, serán: 'g--d'. En nuestro anterior script, colocamos los especificadores tal que así:

por favor trama ([ 4 , 8 , 13 , 17 , 20 ], [ 54 , 67 , 98 , 78 , 45 ], 'g - d' )

En cuyo caso, la imagen lucirá tal que así:

Interfaz de usuario gráfica, Gráfico, Gráfico de líneas

Descripción generada automáticamente

**Gráfico de dispersión**

Un gráfico de dispersión es un gráfico que muestra la relación entre dos conjuntos de datos, como la relación entre edad y peso, por ejemplo. Voy a mostraros cómo podemos dibujar un gráfico de dispersión utilizando matplotlib.

Cogeremos dos conjuntos de datos, x e y, de los cuales queremos encontrar sus relaciones (gráficos de dispersión).

x =[2,4,6,7,9,13,19,26,29,31,36,40,48,51,57,67,69,71,78,88]y =[54,72,43,2,8,98,109   
 ,5,35,28,48,83,94,84,73,11,464,75,200,54]

Si quieres pintar un gráfico de dispersión con dichos puntos, el código será:

import matplotlib.pyplot as plt  
x =[2,4,6,7,9,13,19,26,29,31,36,40,48,51,57,67,69,71,78,88]y =[54,72,43,   
 2 , 8 , 98 , 109 , 5 , 35 , 28 , 48 , 83 , 94 , 84 , 73 , 11 , 464 , 75 , 200 , 54 ]   
plt . dispersión ( x , y )   
plt . mostrar ()

La salida de este script será:

Gráfico, Gráfico de dispersión

Descripción generada automáticamente

Claro está, puede cambiar el color de los marcadores tal y como hemos hecho antes.

**Histogramas**

Un histograma es un gráfico que muestra la frecuencia de los datos utilizando barras, donde los números son agrupados en rangos. En otras palabras, la frecuencia de cada elemento en la listase mostrará utilizando histogramas. Los números agrupados en los rangos se llaman bins. Vamos a ver un ejemplo para entender esto un poco más.

Digamos que tenemos una lista de datos que queremos mostrar en un histograma como el siguiente:

x =[2,4,6,5,42,543,5,3,73,64,42,97,63,76,63,8,73,97,23,45,56,89,45,3,23,2,5,78, 23,56,67,78,8,3,78,34,67,23,324,234,43,544,54,33,223,443,444,234,76,432,233,23,232,243,222,221,254,222,276,300,353,354,387,364,309]

El script de Python que podemos utilizar para mostrar el histograma para los datos anteriores será:

importar matplotlib . pyplot como plt   
x = [ 2 , 4 , 6 , 5 , 42 , 543 , 5 , 3 , 73 , 64 , 42 , 97 , 63 , 76 , 63 , 8 , 73 , 97 , 23 , 45 , 56 , 89 , 45 , 3 , 23 ,2,5,78,23,56,67,78,8,3,78,34,67,23,324,234,43,544,54,33,223,443,444,234,76,432,233,23,232,243, 222 , 221 , 254 , 222 , 276 , 300 , 353 , 354 , 387 , 364 , 309 ]   
num\_bins = 6 n , bins , parches = plt . hist ( x , num\_bins , facecolor = 'verde' ) plt . mostrar ()

Si ejecutas el anterior script, obtendrás un resultado parecido a este:

Gráfico, Histograma

Descripción generada automáticamente

**Bibliografía**

<https://matplotlib.org/stable/index.html>

<https://programacion.net/articulo/introduccion_a_la_libreria_matplotlib_de_python_1599>