

🔑 python para Ciencia de Datos: Hoja de Referencia



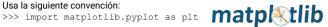
Aprende Python para Ciencia de Datos en www.datademia.es



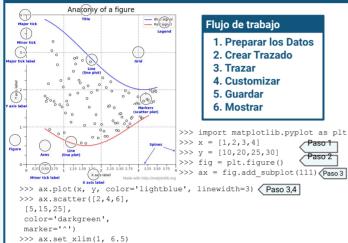
Matplotlib

Matplotlib es una biblioteca de Python que produce figuras con calidad de publicación en una variedad de formatos impresos y entornos interactivos.

Usa la siguiente convención:



Anatomía y flujo de trabajo



Crea el trazado

>>> plt.show()

>>> plt.savefig('foo.png')

>>> fig = plt.figure() >>> fig2 = plt.figure(figsize=plt.figaspect(2.0))

Ejes

Todo el trazado se realiza con respecto a los eies. En la mayoría de los casos, el subtrazado se ajustará a tus necesidades. Un subtrazado esta sobre unos ejes sistema de cuadrícula.

Paso 6

```
>>> fig = plt.figure()
>>> fig.add axes()
>>> ax1 = fig.add_subplot(221) # row-col-num
>>> ax3 = fig.add_subplot(212)
>>> fig3, axes = plt.subplots(nrows=2,ncols=2)
>>> fig4, axes2 = plt.subplots(ncols=3)
```

Trazado

Datos de una dimensión >>> lines = ax.plot(x,y)

Dibuja puntos desconectados, >>> ax.scatter(x, y) escalados o coloreados >>> axes[0,0].bar([1,2,3],[3,4,5]) constante) >>> axes[1,0].barh([0.5,1,2.5],[0,1,2]) Dibuja una línea horizontal a través de >>> axes[1,1].axhline(0.45) los ejes >>> axes[0,1].axvline(0.65) Dibuja polígonos rellenos >>> ax.fill(x.v.color='blue')

Datos de dos dimensiones o imágenes

>>> fig, ax = plt.subplots()
>>> im = ax.imshow(img, cmap='gist_earth', interpolation='nearest', vmin=-2, vmax=2)

>>> axes2[0].pcolor(data2) >>> axes2[0].pcolormesh(data) >>> CS = plt.contour(Y,X,U)

>>> axes2[2].contourf(data1) >>> axes2[2]= ax.clabel(CS)

Campos Vectoriales

>>> axes[0,1].arrow(0,0,0.5,0.5) >>> axes[1,1].quiver(y,z)
>>> axes[0,1].streamplot(X,Y,U,V)

Distribuciones de datos

>>> ax1.hist(v) >>> ax3.boxplot(y) >>> ax3.violinplot(z) Dibuja puntos con líneas o marcadores que los conectan

Traza rectángulos verticales (ancho

Traza rectángulos horizontales (altura

Dibuja una línea vertical a través de los

>>> ax.fill_between(x,y,color='yellow') Rellena entre valores y y 0

Matriz de colores o RGB

Gráfico de pseudocolor de matriz de 2D

Gráfico de pseudocolor de matriz de 2D Trazar contornos Trazar contornos rellenos Etiquetar un trazado de contornos

Añade una flecha a los ejes Trazar un campo 2D de flechas Trazar campos vectoriales 2D

Traza un histograma Haz un diagrama de caja y bigotes Haz un diagrama de violín

Customizar

Colores, barras de colores y mapas de colores

```
>>> plt.plot(x, x, x, x**2, x, x**3)
>>> ax.plot(x, y, alpha = 0.4)
>>> ax.plot(x, y, c='k')
>>> fig.colorbar(im, orientation='horizontal')
>>> im = ax.imshow(img, cmap='seismic')
```

Marcadores

```
>>> fig, ax = plt.subplots()
>>> ax.scatter(x, y, marker=".
>>> ax.plot(x,y,marker="o")
```

Estilo de líneas

```
>>> plt.plot(x,y,linewidth=4.0)
>>> plt.plot(x,y,ls='solid')
>>> plt.plot(x,y,ls='--')
>>> plt.plot(x,y,'--',x**2,y**2,'-.')
>>> plt.setp(lines,color='r',linewidth=4.0)
```

Texto y anotaciones

```
>>> ax.text(1,-2.1, 'Example Graph', style='italic')
>>> ax.annotate("Sine",
                 xy = (8, 0),
                 xycoords='data'.
                 xytext=(10.5, 0),
                 textcoords='data',
                 \verb|arrowprops=dict(arrowstyle="->",
                 connectionstyle="arc3"),)
```

Texto Matemático

>>> plt.title(r'\$sigma i=15\$', fontsize=20)

Límites, leyendas y diseños

Límites v escalado automático

Agregar relleno a una figura >>> ax.margins(x=0.0,y=0.1) Establece la relación de aspecto de >>> ax.axis('equal') la figura a 1 Establece límites para los ejes xyy >>>ax.set(xlim=[0,10.5],ylim=[-1.5,1.5]) Establece límites para el eje x >>> ax.set xlim(0,10.5)

>>> ax.set(title='An Example Axes', ylabel='Y-Axis', xlabel='X-Axis') >>> ax.legend(loc='best')

Establecer un título y etiquetas de Elementos no superpuestos

Marcas en los ejes

>>> ax.xaxis.set(ticks=range(1,5), ticklabels=[3,100,-12,"foo"]) >>> ax.tick_params(axis='y', direction='inout', length=10) Establecer manualmente las marcas Hacer las marcas en el eie v más

largas y que entren y salgan

Espaciado de subtrazados

>>> fig3.subplots_adjust(wspace=0.5, hspace=0.3, left=0.125, right=0.9. top=0.9, bottom=0.1) >>> fig.tight_layout()

Aiuste el espacio entre subtrazados

Ajusta subtrazado en el área de la

Espinas del eje

>>> ax1.spines['top'].set visible(False) >>> ax1.spines['bottom'].set_position(('outward',10))

Haz invisible la línea del eje superior Mueve la línea del eje inferior hacia afuera

Guardar

Guardar Figura

>>> plt.savefig('foo.png')

Guardar figura transparente

>>> plt.savefig('foo.png', transparent=True)

Mostrar

>>> plt.show()

Cerrar y despejar

>>> plt.clf() >>> plt.close() Despeiar el eie Despejar la figura entera Cerrar la ventana



Aprende Python para Ciencia de Datos en www.datademia.es