

# matplotlib 函数手册

星猫译

函数	介绍
<a href="#">acorr</a>	Plot the autocorrelation of x. 绘制 X 的自相关
<a href="#">annotate</a>	Create an annotation: a piece of text referring to a data point. 用箭头在指定的一个数据点创建一个注释或一段文本。
<a href="#">arrow</a>	Add an arrow to the axes. 为 axes 添加一个箭头。
<a href="#">autoscale</a>	Autoscale the axis view to the data (toggle). 自动缩放轴视图的数据（切换）。
<a href="#">axes</a>	Add an axes to the figure. 为当前 figure 添加一个 axes。
<a href="#">axhline</a>	Add a horizontal line across the axis. 添加一条穿越 axis 的水平线。
<a href="#">axhspan</a>	Add a horizontal span (rectangle) across the axis. 添加一条穿越 axis 的水平矩形。
<a href="#">axis</a>	Convenience method to get or set axis properties. 获取或设置轴属性的便捷方法。
<a href="#">axvline</a>	Add a vertical line across the axes. 添加一条穿越 axes 的垂线。
<a href="#">axvspan</a>	Add a vertical span (rectangle) across the axes. 添加一个与 axes 交叉的垂直跨度（矩形）。
<a href="#">bar</a>	Make a bar plot. 做一个垂直条形图。
<a href="#">barbs</a>	Plot a 2-D field of barbs. 绘制一个倒钩的二维场（风场）。
<a href="#">barh</a>	Make a horizontal bar plot. 做一个横向条形图。
<a href="#">box</a>	Turn the axes box on or off. 打开或关闭主轴箱。
<a href="#">boxplot</a>	Make a box and whisker plot. 使一盒须图。
<a href="#">broken_barh</a>	Plot horizontal bars. 绘制水平杆。
<a href="#">cla</a>	Clear the current axes. 清除当前 axes。
<a href="#">clabel</a>	Label a contour plot. 为等值线图设标签
<a href="#">clf</a>	Clear the current figure. 清除当前 figure。
<a href="#">clim</a>	Set the color limits of the current image. 设置当前图像的颜色取值范围。
<a href="#">close</a>	Close a figure window. 关闭 figure 窗口。
<a href="#">cohere</a>	Plot the coherence between x and y. 绘制 X 和 Y 之间的相关性分析

<a href="#"><u>colorbar</u></a>	Add a colorbar to a plot. 为图添加彩条标值。
<a href="#"><u>contour</u></a>	Plot contours. 绘制等值线。
<a href="#"><u>contourf</u></a>	Plot contours. 绘制填充等值线。
<a href="#"><u>csd</u></a>	Plot cross-spectral density. 绘制交叉谱密度。
<a href="#"><u>delaxes</u></a>	Remove an axes from the current figure. 从目前的 figure 删除 axes。
<a href="#"><u>draw</u></a>	Redraw the current figure. 再次绘制当前 figure。
<a href="#"><u>errorbar</u></a>	Plot an errorbar graph. 绘制 errorbar 图。
<a href="#"><u>eventplot</u></a>	Plot identical parallel lines at specific positions. 在特定的位置绘制相同的平行线。
<a href="#"><u>figimage</u></a>	Adds a non-resampled image to the figure. 为 figure 添加一个非重采样图像。
<a href="#"><u>figlegend</u></a>	Place a legend in the figure. 为 figure 放置一个标注 (legend)
<a href="#"><u>figtext</u></a>	Add text to figure. 为 figure 添加文字。
<a href="#"><u>figure</u></a>	Creates a new figure. 创建一个新的 figure。
<a href="#"><u>fill</u></a>	Plot filled polygons. 绘制填充多边形。
<a href="#"><u>fill between</u></a>	Make filled polygons between two curves. 使两曲线间填充色彩。
<a href="#"><u>fill betweenx</u></a>	Make filled polygons between two horizontal curves. 使两水平线间填充色彩。
<a href="#"><u>findobj</u></a>	Find artist objects. 发现 artist 对象。
<a href="#"><u>gca</u></a>	Return the current axis instance. 返回当前轴实例 (axis instance)。
<a href="#"><u>gcf</u></a>	Return a reference to the current figure. 返回当前 figure 的序号。
<a href="#"><u>gci</u></a>	Get the current colorable artist. 获取当前彩条的 artist。
<a href="#"><u>get_figlabels</u></a>	Return a list of existing figure labels. 返回一个当前 figure 的标签列表。
<a href="#"><u>get_fignums</u></a>	Return a list of existing figure numbers. 返回一系列 figure 的序号。
<a href="#"><u>grid</u></a> (on/off)	Turn the axes grids on or off. 打开或关闭坐标网格。
<a href="#"><u>hexbin</u></a>	Make a hexagonal binning plot. 做一个六边形箱图。
<a href="#"><u>hist</u></a>	Plot a histogram. 绘制直方图。
<a href="#"><u>hist2d</u></a>	Make a 2D histogram plot. 使一个二维直方图。

<a href="#">hlines</a>	Plot horizontal lines. 绘制水平线。
<a href="#">hold</a>	Set the hold state. 设置 hold 状态。
<a href="#">imread</a>	Read an image from a file into an array. 读取一个图像，从图形文件中提取数组。
<a href="#">imsave</a>	Save an array as in image file. 保存为图像文件中的数组。
<a href="#">imshow</a>	Display an image on the axes. 在 axes 上显示图像。
<a href="#">ioff</a>	Turn interactive mode off. 关闭互动模式。
<a href="#">ion</a>	Turn interactive mode on. 开启互动模式。
<a href="#">ishold</a>	Return the hold status of the current axes. 返回当前 axes 的 hold 状态。
<a href="#">isinteractive</a>	Return status of interactive mode. 返回互动模式状态。
<a href="#">legend</a>	Place a legend on the current axes. 为当前 axes 放置标注。
<a href="#">locator_params</a>	Control behavior of tick locators. 控制轴刻度标签
<a href="#">loglog</a>	Make a plot with log scaling on both the x and y axis. 使 X、Y 轴为 log 刻度。
<a href="#">margins</a>	Set or retrieve autoscaling margins. 设置或检索自动缩放功能。
<a href="#">matshow</a>	Display an array as a matrix in a new figure window. 在新 figure 窗口显示数组矩阵
<a href="#">minorticks_off</a>	Remove minor ticks from the current plot. 移除当前数轴上的次刻度
<a href="#">minorticks_on</a>	Display minor ticks on the current plot. 显示当前数轴上的次刻度
<a href="#">over</a>	Call a function with hold(True). 调用一个函数，并且 hold 为 (True)。
<a href="#">pause</a>	Pause for interval seconds. Pause 的时间间隔（秒）。
<a href="#">pcolor</a>	Create a pseudocolor plot of a 2-D array. 创建一个二维阵列的伪彩色图。
<a href="#">pcolormesh</a>	Plot a quadrilateral mesh. 绘制一个四边形网格。
<a href="#">pie</a>	Plot a pie chart. 绘制一个饼图。
<a href="#">plot</a>	<a href="#">Plot lines and/or markers to the Axes.</a> 绘制当前 axes。
<a href="#">plot_date</a>	Plot with data with dates. 绘制数据日期。
<a href="#">plotfile</a>	Plot the data in in a file. 将图绘入文件
<a href="#">polar</a>	Make a polar plot. 绘制极坐标图。

<a href="#"><u>psd</u></a>	Plot the power spectral density. 绘制功率谱密度图。
<a href="#"><u>quiver</u></a>	Plot a 2-D field of arrows. 绘制二维箭头图（风矢量图）。
<a href="#"><u>quiverkey</u></a>	Add a key to a quiver plot. 为风矢量图绘制单位标量
<a href="#"><u>rc</u></a>	Set the current rc params. 设置当前的 RC 参数。
<a href="#"><u>rc_context</u></a>	Return a context manager for managing rc settings. 返回一个用于管理 RC 设置上下文管理器。
<a href="#"><u>rcdefaults</u></a>	Restore the default rc params. 恢复默认的 RC 参数。
<a href="#"><u>rgrids</u></a>	Get or set the radial gridlines on a polar plot. 获取或设置径向网格在极坐标图。
<a href="#"><u>savefig</u></a>	Save the current figure. 保存当前图。
<a href="#"><u>sca</u></a>	Set the current Axes instance to ax. 为 ax 设置当前（axes）轴实例。
<a href="#"><u>scatter</u></a>	Make a scatter plot of x vs y, where x and y are sequence like objects of the same lengths. 做一个 X 和 Y 的散点图，其中 X 和 Y 是相同长度的序列的对象。
<a href="#"><u>sci</u></a>	Set the current image. 设置当前的图像。
<a href="#"><u>semilogx</u></a>	Make a plot with log scaling on the x axis. 使 x 轴为 log 刻度。
<a href="#"><u>semilogy</u></a>	Make a plot with log scaling on the y axis. 使 y 轴为 log 刻度。
<a href="#"><u>set_cmap</u></a>	Set the default colormap. 设置默认的 colormap。
<a href="#"><u>setp</u></a>	Set a property on an artist object. 对 artist 对象设置属性。
<a href="#"><u>show</u></a>	Display a figure. 显示图。
<a href="#"><u>spectrogram</u></a>	Plot a spectrogram. 绘制谱图。
<a href="#"><u>spy</u></a>	Plot the sparsity pattern on a 2-D array. 绘制一个二维阵列的稀疏模式。
<a href="#"><u>stackplot</u></a>	Draws a stacked area plot. 绘制一个堆叠面积图。
<a href="#"><u>stem</u></a>	Create a stem plot. 创建一个 stem 图。
<a href="#"><u>step</u></a>	Make a step plot. 做一个步阶图
<a href="#"><u>streamplot</u></a>	Draws streamlines of a vector flow. 绘制一个流场图。
<a href="#"><u>subplot</u></a>	Return a subplot axes positioned by the given grid definition. 返回一个 subplot axes
<a href="#"><u>subplot2grid</u></a>	Create a subplot in a grid. 在网格中创建一个 subplot。
<a href="#"><u>subplot_tool</u></a>	Launch a subplot tool window for a figure. 获取 subplot 工具窗口。

<a href="#">subplots</a>	Create a figure with a set of subplots already made. 一个图中包含多个子图
<a href="#">subplots_adjust</a>	Tune the subplot layout. 调整 subplot 布局
<a href="#">suptitle</a>	Add a centered title to the figure. 在图添加一个中心标题。
<a href="#">switch_backend</a>	Switch the default backend. 交换机默认的后台。
<a href="#">table</a>	Add a table to the current axes. 当前 axes 添加 table。
<a href="#">text</a>	Add text to the axes. 在轴上添加文本。
<a href="#">thetagrids</a>	Get or set the theta locations of the gridlines in a polar plot. 设置极坐标网格线 $\theta$ 位置。
<a href="#">tick_params</a>	Change the appearance of ticks and tick labels. 改变刻度及刻度标签外观
<a href="#">ticklabel_format</a>	<a href="#">Change the ScalarFormatter used by default for linear axes.</a> 通过使用默认线性 axes 轴更改 scalarformatter 。
<a href="#">tight_layout</a>	Automatically adjust subplot parameters to give specified padding. 自动调节 subplot 参数进行指定填充。
<a href="#">title</a>	Set a title of the current axes. 设置当前 axes 标题。
<a href="#">tricontour</a>	Draw contours on an unstructured triangular grid. 在非结构三角形网格绘制等值线。
<a href="#">tricontourf</a>	Draw contours on an unstructured triangular grid. 在非结构三角形网格绘制填充等值线。
<a href="#">tripcolor</a>	Create a pseudocolor plot of an unstructured triangular grid. 创建一个非结构三角形网格伪彩色图。
<a href="#">triplot</a>	Draw a unstructured triangular grid as lines and/or markers. 画一个非结构三角形网格图（类似 plot 函数）。
<a href="#">twinx</a>	Make a second axes that shares the x-axis. 制作第二个 axes，并共用 x 轴。
<a href="#">twiny</a>	Make a second axes that shares the y-axis. 制作第二个 axes，并共用 y 轴。
<a href="#">vlines</a>	Plot vertical lines. 绘制垂直线。
<a href="#">xcorr</a>	Plot the cross correlation between x and y. 绘制 x 和 y 的相关性分析图
<a href="#">xkcd</a>	<a href="#">Turns on xkcd sketch-style drawing mode.</a> 开启 XKCD 草图风格绘画模。

<a href="#"><u>xlabel</u></a>	Set the x axis label of the current axis. 在当前轴设置 X 轴的标签。
<a href="#"><u>xlim</u></a>	Get or set the x limits of the current axes. 设置当前 axes 的 x 轴取值范围
<a href="#"><u>xscale</u></a>	Set the scaling of the x-axis. 设置 X 轴缩放。
<a href="#"><u>xticks</u></a>	Get or set the x-limits of the current tick locations and labels. 设置当前 x 轴刻度位置的标签和值。
<a href="#"><u>ylabel</u></a>	Set the y axis label of the current axis. 设置当前轴 y 轴标签。
<a href="#"><u>ylim</u></a>	Get or set the y-limits of the current axes. 设置当前 axes 的 y 轴取值范围
<a href="#"><u>yscale</u></a>	Set the scaling of the y-axis. 设置 Y 轴缩放。
<a href="#"><u>yticks</u></a>	Get or set the y-limits of the current tick locations and labels. 设置当前 y 轴刻度位置的标签和值。

函数：

1. matplotlib.pyplot.**acorr**(*x*, *hold=None*, **\*\*kwargs**)

自相关函数

```
acorr(x, normed=True, detrend=mlab.detrend_none, usevlines=True,
      maxlags=10, **kwargs) #x 的自相关
```

如果 *normed = True*, 通过零滞后自相关规范数据, *x* 通过非趋势可调用 (默认没有归一化) 的方式除趋势

数据绘制形如 `plot(lags, c, **kwargs)`

返回值是元组 (*lags*, *c*, *line*), 其中:

*lags* 是一个长度为  $2 \times \text{maxlags} + 1$  的滞后 (*lags*) 向量

*C* 是  $2 \times \text{maxlags} + 1$  的自相关向量

*line* 是一个 `line2d` 实例, 通过 `plot()` 返回

默认 *linestyle* 是 *None*, 默认的 *marker* 是 'o', 交叉相关性是通过 `numpy.correlate()` 函数、*mode = 2* 实现的。

若 *usevlines* 是 *True*, `vlines()` 将被调用 (而不调用 `plot()` 函数), 用来绘制从起点到 `acorr` 的垂线。否则, `plot()` 由 `Line2D properties` 属性参数 (*kwargs*) 决定。

*maxlags* 是正整数, 决定 *lags* 的显示数目。默认值 *None* 将返回  $(2 \times \text{len}(x) - 1)$  个 *lags*, 返回值是一个元组 (*lags*, *c*, *linecol*, *b*), 其中:

*linecol* 是 `linecollection`

*b* 是 *x-axis*

其他参数参见 [Line2D](#) 的 *kwargs* 属性

2. matplotlib.pyplot.**annotate**(*\*args*, **\*\*kwargs**)

创建一个文本注释: 从指定点指向目标点

```
annotate('注释内容', xy,
xytext=None, xycoords='data', textcoords='data', arrowprops=None,
**kwargs)
```

*xy*: 被注释点的位置, *xytext*: 注释文本的位置坐标

*xycoords* 和 *textcoords* 是字符串, 指示 *xy* 和 *xytext* 的坐标关系:

Property	Description	
'figure points'	points from the lower left corner of the figure	起点从 figure 左下角
'figure pixels'	pixels from the lower left corner of the figure	pixels (像素) 从 figure 左下角



'figure fraction'	0,0 is lower left of figure and 1,1 is upper right	(0, 0) 是 figure 的左下方, (1, 1) 是右上方
'axes points'	points from lower left corner of axes	起点从 axes 的左下方
'axes pixels'	pixels from lower left corner of axes	pixels (像素) 从 axes 左下角
'axes fraction'	0,0 is lower left of axes and 1,1 is upper right	(0, 0) 是 axes 的左下方, (1, 1) 是右上方
'data'	use the coordinate system of the object being annotated (default)	使用被注释对象的坐标系
'offset points'	Specify an offset (in points) from the xy value	从 xy 点指定一个偏移量
'polar'	you can specify theta, r for the annotation, even in cartesian plots. Note that if you are using a polar axes, you do not need to specify polar for the coordinate system since that is the native "data" coordinate system.	即使在直角坐标系, 也可以指定 (θ, r) 的值。如果使用极坐标, 不需要指定极点

通用箭头属性:

Key	Description	
width	the width of the arrow in points	箭头宽度
frac	the fraction of the arrow length occupied by the head	箭头尖端 (头部) 所占比例
headwidth	the width of the base of the arrow head in points	箭头尖端的尾部宽度
shrink	oftentimes it is convenient to have the arrowtip and base a bit away from the text and point being annotated. If d is the distance between the text and annotated point, shrink will shorten the arrow so the tip and base are shrink percent of the distance d away from the endpoints. ie, shrink=0.05 is 5%	缩短箭头, 以容纳文字。 Shrink=0.05 表示缩短 5%
?kwargs	any key for matplotlib.patches.polygon	通用绘图参数, 包括 facecolor、alpha 等, 可查

		询 kwargs 表格。
--	--	--------------

花式箭头参数：

Key	Description	
arrowstyle	the arrow style	箭头风格
connectionstyle	the connection style	
relpos	default is (0.5, 0.5)	默认 (0.5, 0.5)
patchA	default is bounding box of the text	默认边界框
patchB	default is None	默认无
shrinkA	default is 2 points	默认 2
shrinkB	default is 2 points	默认 2
mutation_scale	default is text size (in points)	默认为字体大小
mutation_aspect	default is 1.	默认为 1
?	any key for matplotlib.patches.PathPatch	

其他参数参见 kwargs

3. matplotlib.pyplot.**arrow**(*x, y, dx, dy, hold=None, \*\*kwargs*)

为 axes 添加箭头

**arrow**(*x, y, dx, dy, \*\*kwargs*)

绘制指定的箭头从 (X, Y) 指向 (X + Y + dx, dy)。常用的参数：

width (箭头尾部宽度) :	浮点数 (默认: 0.001)
length_includes_head (length 值是否包含 head[箭头尖部]) :	[True   False] (默认: False)
head_width (箭头 head 宽度) :	float (浮点数) 或 None (默认: 3*width)
head_length:	float (浮点数) 或 None (默认: 1.5 * head_width)
shape (箭头形状) :	['full'('完整的'), 'left'('左半部分'), 'right'('右半部分')] (default: 'full')
overhang (箭头尖端形状) :	float (浮点数) (默认: 0)可以是负值, 也可以大于 1
head_starts_at_zero:	[True   False] (默认: False)如果为 True, 箭头尖部起点在坐标 0 点, 而 False 则终点在坐标 0 点

其他参数参见 **matplotlib.patches.Polygon** 实例。

4. matplotlib.pyplot.**autoscale**(*enable=True, axis='both', tight=None*)

自动缩放轴视图的数据（切换），方便简单的轴视图自动缩放的方法。

enable:	[True   False   None]	True (默认) 开启自动缩放, False 关闭, None 离开 (自动缩放状态不改变)
axis:	['x'   'y'   'both']	选择缩放的轴线, 默认 'both'

tight:	[True   False   None]	如果 True，根据 data 范围设置视角（view）范围；若为 False，可扩大视角范围；若为 None，只有一个 data 时，按 True 处理，其他按 False 对待
返回 None		

#### 5. matplotlib.pyplot.autumn()

对 `autumn()` 函数设置默认的 colormap，并应用于当前图像。更多信息查询

`help(colormaps)`。

#### 6. matplotlib.pyplot.axes(\*args, \*\*kwargs)

对 figure 增加一个 axes

由 `rect` 指定 axes 添加的位置：

<code>axes()</code>	创建一个默认的完整的 subplot (111) 图轴。	
<code>axes(rect, axisbg='w')</code>	其中 <code>rect = [left, bottom, width, height]</code> （四个参数取值范围为[0,1]） <code>axisbg</code> 指背景颜色，默认为 white	
<code>axes(h)</code>	其中 h 指 axes 实例，包含下列参数：	
kwarg	Accepts	Description
<code>axisbg</code>	color	Axes 图轴背景色
<code>frameon</code>	[True False]	是否绘制图像框架
<code>sharex</code>	otherax	当前 axes 图轴是否与其它图轴共用 x 轴
<code>sharey</code>	otherax	当前 axes 图轴是否与其它图轴共用 y 轴
<code>polar</code>	[True False]	使用极坐标轴？

#### 7. matplotlib.pyplot.axhline(y=0, xmin=0, xmax=1, hold=None, \*\*kwargs)

添加一条穿越 axis 的水平线。

```
axhline(y=0, xmin=0, xmax=1, **kwargs)
```

绘制一条从 xmin 到 xmax 的水平线，其中 0=left, 0.5=middle, 1.0=right，而水平线的长度视水平轴长度而定。

该函数返回值为 **Line2D**（二维曲线）参量，详情查询 **Line2D** 属性。

#### 8. matplotlib.pyplot.axhspan(ymin, ymax, xmin=0, xmax=1, hold=None, \*\*kwargs)

添加一条穿越 axis 的水平 span（矩形）。

其中 x 轴方向： 0=left, 0.5=middle, 1.0=right，水平线的长度视水平轴长度而定。而 y 轴则根据坐标点确定。

返回值为 **matplotlib.patches.Polygon** 实例，详情查询 **matplotlib.patches.Polygon**

## 9.matplotlib.pyplot.axis(\*v, \*\*kwargs)

获取或设置轴属性的便捷方法。没有输入调用变量:

```
>>> axis()          #返回当前 axes 范围 (limits) [xmin, xmax, ymin, ymax].
```

通过输入 `v = [xmin, xmax, ymin, ymax]`. 设置 `x`、`y` 的取值范围

```
>>> axis(v)
.....
>>> axis('off')      #关闭轴线及其标签
.....
>>> axis('equal')     #使 x、y 轴长度一致
.....
>>> axis('scaled')    #调整图框的尺寸（而不是改变坐标轴取值范围），使 x、y 轴
长度一致
.....
>>> axis('tight')     #改变 X 和 Y 轴的限制，使所有数据被展示。如果所有的数据
已经显示，它将移动到图形的中心而不修改（XMAX - XMIN）或（YMAX - Ymin）。
.....
>>> axis('image')     #缩放 axis 范围 (limits)，等同于对 data 缩放范围
.....
>>> axis('auto')
>>> axis('normal')
```

不推荐使用。他们恢复默认状态；轴限的自动缩放以使数据合适地在图中。

## 10.matplotlib.pyplot.axvline(x=0, ymin=0, ymax=1, hold=None, \*\*kwargs)

添加一条穿越 axes 的垂线。

```
axvline(x=0, ymin=0, ymax=1, **kwargs)
```

绘制一条从 `ymin` 到 `ymax` 的水平线，其中 `0=bottom`, `0.5=middle`, `1.0=top`，而垂线的长度视水平轴长度而定。

该函数返回值为 `Line2D`（二维曲线）参量，与 `Line2D` 属性一致：

## 11.matplotlib.pyplot.axvspan(xmin, xmax, ymin=0, ymax=1, hold=None, \*\*kwargs)

添加一个与 axes 交叉的垂直跨度（矩形）。

```
axvspan(xmin, xmax, ymin=0, ymax=1, **kwargs)
```

其中 `x` 轴方向： `0=bottom`, `0.5=middle`, `1.0=top`，垂线的长度视水平轴长度而定。

而 `x` 轴则根据坐标点确定。

返回值为 `matplotlib.patches.Polygon` 实例：

## 12.matplotlib.pyplot.bar(left, height, width=0.8, bottom=None, hold=None, \*\*kwargs)

绘制条形图

矩形边界条形图: `left, left + width, bottom, bottom + height(left, right, bottom and top edges)`

left :	标量或数组	bar 左端的 x 轴坐标
--------	-------	---------------

height :	标量或数组	bar 高度
width :	标量或数组	bar 宽度
bottom :	标量或数组, 可选, 默认为 None	bar 的底部高度
color :	色彩、标量或数组, 可选	bar 内部颜色
edgecolor :	色彩、标量或数组, 可选	bar 边缘颜色
linewidth :	标量或数组, 可选, 默认为 None	bar 边缘宽度, 若为 None, 取默认宽度; 若为 0, 则无边缘
xerr :	标量或数组, 可选, 默认为 None	若不为 None, 将生成 errorbar(s)
yerr :	标量或数组, 可选, 默认为 None	若不为 None, 将生成 errorbar(s)
ecolor :	标量或数组, 可选, 默认为 None	指定 errorbar(s)的颜色
capsize :	整数, 可选, 默认值: 3	指定 errorbar(s)的 cap (帽子) 的长度
error_kw :	字典	errorbar 字典, ecolor 和 capsize 可在此指定, 而不作为独立参量
align :	['edge'   'center'],可选, 默认: 'edge'	若为'edge',对于竖条形图, 按 bar 左边缘排列; 对于横条形图, 按 bar 下边缘排列。若为'center', 把左端的参数作为 bar 的中心坐标。
orientation :	'vertical'   'horizontal',可选, 默认: 'vertical'	选择竖条形图、横条形图
log :	True、False (0、1) , 可选, 默认为 False	如果为 True, 为对数轴

返回:class (类别):`matplotlib.patches.Rectangle`实例。

其他可选参量, 参见**matplotlib.patches.Polygon**实例(完全一致)。

### 13. matplotlib.pyplot.barbs(\*args, \*\*kw)

绘制二维风向杆

barb(X, Y, U, V, C, **kw)		
X, Y:	风向杆的坐标位置, 默认为 barb 的头部	
U, V:	风向的 x、y 方向分量	
C:	风向杆颜色	
length:	风向杆长度, 默认为 9	
pivot (支点) :	['tip'   'middle']默认'tip'	风向杆围绕着指定点旋转
barbcolor:	[ color   color 序列、数组 ]	指定除了 flags 部分之外的颜色, 会覆

		盖 facecolor
flagcolor	[ color   color 序列、数组 ]	指定 flags 颜色，会覆盖 facecolor
sizes: (该参量为字典)	‘spacing’ - flags 间距 ‘height’ - 高度 (shaft 到 top 距离) ‘width’ - flag 宽度, full barb (完整 barb) 的两倍宽度 ‘emptybarb’ - radius of the circle used for low magnitudes 用于低幅度的圆半径	
fill_empty:	A flag on whether the empty barbs (circles) that are drawn should be filled with the flag color. If they are not filled, they will be drawn such that no color is applied to the center. Default is False	
rounding:	A flag to indicate whether the vector magnitude should be rounded when allocating barb components. If True, the magnitude is rounded to the nearest multiple of the half-barb increment. If False, the magnitude is simply truncated to the next lowest multiple. Default is True	
barb_increments: (该参量为字典)	‘half’ - half barbs (Default is 5) ‘full’ - full barbs (Default is 10) ‘flag’ - flags (default is 50)	
flip_barb:	Either a single boolean flag or an array of booleans. Single boolean indicates whether the lines and flags should point opposite to normal for all barbs. An array (which should be the same size as the other data arrays) indicates whether to flip for each individual barb. Normal behavior is for the barbs and lines to point right (comes from wind barbs having these features point towards low pressure in the Northern Hemisphere.) Default is False	

额外参量参见**PolyCollection**

14.matplotlib.pyplot.**barh**(bottom, width, height=0.8, left=None, hold=None, \*\*kwargs)

绘制横条形图

```
barh(bottom, width, height=0.8, left=0, **kwargs)#
```

返回: `class` (类别): `matplotlib.patches.Rectangle` 实例。必要参量:

<code>bottom</code>	Bars 的垂直位置的底部边缘
<code>width</code>	Bars 的长度

可选参量:

Keyword	描述
<code>height</code>	bars 的高度
<code>left</code>	bars 左边缘 x 轴坐标值
<code>color</code>	bars 颜色
<code>edgecolor</code>	bars 边缘颜色
<code>linewidth</code>	bar 边缘宽度 ; None 表示默认宽度 ; 0 表示不 i 绘制边缘
<code>xerr</code>	若不为 None, 将在 bar 图上生成 errobars
<code>yerr</code>	若不为 None, 将在 bar 图上生成 errobars
<code>ecolor</code>	指定 errorbar 颜色
<code>capsize</code>	指定 errorbar 的顶部 (cap) 长度
<code>align</code>	'edge' (默认)   'center': 'edge' 以底部为准对齐 ; 'center' 以 y 轴作为中心 ?
<code>log</code>	[False True] False (默认), 若为 True, 使用 log 坐标

其他参数参见 `matplotlib.patches.Polygon` 实例

15. `matplotlib.pyplot.box(on=None)`

设置 axes 边框是否打开, ['on' 或 'off'], 也可以是布尔数组

16. `matplotlib.pyplot.boxplot(x, notch=False, sym='b+', vert=True, whis=1.5, positions=None, widths=None, patch_artist=False, bootstrap=None, usermedians=None, conf_intervals=None, hold=None)`

绘制方框和线

```
boxplot(x, notch=False, sym='+', vert=True, whis=1.5, positions=None, widths=None, patch_artist=False, bootstrap=None, usermedians=None, conf_intervals=None)
```

<code>x :</code>		表示向量的数组或列表
<code>notch :</code>	[ False (默认)   True ]	若为 False (默认) , 绘制一个矩形框。若为 True, 生成一个有缺口的矩形框
<code>sym :</code>	[ 默认'b+' ]	flier points (飞点) 的点符号, 若不显示飞点, 输入空字符串 ('')

vert :	[ False   True (默认) ]	若为 True (默认) , 使 box 竖放。若为 False, 使横框。
whis :	[ default 1.5 ]	定义 whisker (box 的延伸线) 范围? 具体参阅说明
bootstrap :	[ None (默认)   integer (整数) ]	具体参阅说明
usermedians :	[ 默认 None ]	具体参阅说明
conf_intervals :	[ 默认 None ]	具体参阅说明
positions :	[默认 1,2,...,n ]	设置水平位置, 刻度和取值范围 自动匹配
widths :	[默认 0.5 ]	设置每个 box 的宽度, 通过输入一个标量或矢量。默认值 0.5 或 0.15*(两端距离)
patch_artist :	[ False (默认)   True ]	False 对应 Line2D 属性, True 对应 Patch 属性

返回值包含参量:

boxes:	the main body of the boxplot showing the quartiles and the median's confidence intervals if enabled.
medians:	horizontal lines at the median of each box.
whiskers:	the vertical lines extending to the most extreme, n-outlier data points.
caps:	the horizontal lines at the ends of the whiskers.
fliers:	points representing data that extend beyond the whiskers (outliers).

17. `matplotlib.pyplot.broken_barh(xranges, yrange, hold=None, **kwargs)`

绘制水平方框

```
broken_barh(self, xrange, yrange, **kwargs)
```

必要参量:

xranges	(xmin, xwidth)序列
yrange	(ymin, ywidth)序列

其他参数参考 `matplotlib.collections.BrokenBarHCollection` 属性。

18. `matplotlib.pyplot.cla()`

清除当前 axes (图轴)

19. `matplotlib.pyplot.clabel(CS, *args, **kwargs)`

为等值线图添加标签



`clabel(cs, v, **kwargs)`

**cs** 是等值线函数（**contour**）返回的 **ContourSet** 目标，而 **v** 包含 **label** 标签信息。

可选参量：

<b>fontsize:</b> 字体大小	用标量表示，或名称表示'smaller', 'x-large'
<b>colors:</b>	若为 <b>None</b> ，每个标签的颜色相匹配对应的轮廓颜色；若为一种颜色，如 <b>colors='r'</b> 或 <b>colors='red'</b> ，则所以标签显示这种颜色；若为一系列颜色（色彩字符串或数组），则不同 <b>label</b> 按顺序绘制不同色彩
<b>inline:</b>	[ <b>True</b> (默认), <b>False</b> ]表示标签下面的等值线是否被并且覆盖
<b>inline_spacing:</b>	默认为 5，值越大，标签位置越靠近等值线较平直的部位
<b>fmt:</b>	指定标签格式的字符串，默认：'%1.3f'。更多信息参见手册
<b>manual:</b>	若为 <b>True</b> ，可用鼠标或键盘确定标签位置
<b>rightside_up:</b>	若为 <b>True</b> （默认），标签将竖直放置
<b>use_clabeltext:</b>	默认 <b>False</b> 。若为 <b>True</b> ，等值线变化时，标签可随等值线的变化而变化

20. `matplotlib.pyplot.clf()`

清除当前 figure

21. `matplotlib.pyplot.clim(vmin=None, vmax=None)`

设置当前 image 的色彩取值范围

22. `matplotlib.pyplot.close(*args)`

关闭 figure 窗口

<code>close()</code>	关闭当前窗口
<code>close(h)</code>	其中 h 指 figure 实例，关闭那个 figure
<code>close(num)</code>	关闭第 num 个 figure
<code>close(name)</code>	关闭名为 name 的 figure
<code>close('all')</code>	关闭所有 figure

23. `matplotlib.pyplot.cohere(x, y, NFFT=256, Fs=2, Fc=0, detrend=<function detrend_none at 0x2635de8>, window=<function window_hanning at 0x2635b90>, noverlap=0, pad_to=None, sides='default', scale_by_freq=None, hold=None, **kwargs)`

绘制 x 与 y 之间的相关性

```
cohere(x, y, NFFT=256, Fs=2, Fc=0, detrend = mlab.detrend_none, window =
mlab.window_hanning, noverlap=0, pad_to=None, sides='default',
scale_by_freq=None, **kwargs)
```

相关性函数  $C_{xy} = \frac{|P_{xy}|^2}{P_{xx}P_{yy}}$ ，关键参数：

NFFT:	integer	每排的 FFT 数据点数。必须均匀，2 的平方最有效，默认值 256。不能用于获取零填充（补零），否则结果错误，使用 pad_to 替代。
Fs:	scalar 标量	采样频率（样品每时间单位[time unit]）。它是用来计算傅立叶频率,freqs,周期中每时间单位(in cycles per time unit)。默认值为 2
detrend:	callable ?	应用在每个 fft 之前的函数，旨在消除平均或线性趋势。不像在 MATLAB，detrend 参数是一个向量，而在 matplotlib 是一个函数。Pylab 模块定义了 detrend_none(), detrend_mean(), and detrend_linear()函数，也可以用户自定义。
window:	callable 或 ndarray (python 的某种数组)	一个函数或向量的傅立叶变换长度。创建窗口向量出现 window_hanning(), window_none(), NumPy.blackman(), NumPy.hamming(), NumPy.bartlett(), SciPy.signal(), scipy.signal.get_window(),等。默认的是 window_hanning(), 如果一个函数作为参数传递，它必须以一个数据段作为参数和返回段的窗口版本。
pad_to:	integer	执行 FFT 时，数据段垫点的个数。不同于 NFFT 指定使用的数据点的数目。在不增加 PSD 的实际分辨率（分辨峰之间的最小距离）的情况下，可以在图中获取更多的点。这对应于调用 fft()中的 n 参数，默认为 None，可设置 pad_to 使等同于 NFFT。
sides:	['default'   'onesided'   'twosided']	指定返回哪一侧的 PSD，默认'default'，它返回 one-sided 真实数据和复杂的数据。'one-sided'返回一侧，'twosided'返回两侧
scale_by_freq:	boolean 布尔数	指定是否返回由缩放频率缩放的密度值，单位 Hz <sup>-1</sup> 。这允许集成在返回的频率值。MATLAB 兼容默认是 True 的。
noverlap:	integer	重叠的块之间的点的数量。默认值是 0（无重叠）

		。
Fc:	integer	X 的中心频率（缺省为 0）。offsets the x extents of the plot to reflect the frequency range used when a signal is acquired and then filtered and downsampled to baseband.

返回元组( $C_{xy}$ ,  $f$ ),  $f$  是相关性向量的频率。

其他参数参见 [Line2D](#) 属性。

24.matplotlib.pyplot.colorbar(*mappable=None, cax=None, ax=None, \*\*kw*)

对一张 plot 添加一个彩条 (colorbar )

colorbar(mappable, ax=ax, \*\*kwargs)

mappable	应用于该 colorbar 的参量, 包括 Image, ContourSet 等。该参量对于 colorbar() 方法是必须的, 但对该函数是可选的, 默认当前 image。
cax	None   该 colorbar 将被绘制的 axes 目标
ax	None (默认)   与 axes 相关联的 colorbar
use_gridspec	False (默认)   若 <i>cax=None</i> , 将创建一个新的 axes 实例; 若 <i>ax</i> 是一个 Subplot 实例、并且 <i>use_gridspec=True</i> , 将创建一个 Subplot 使用的 grid_spec 模块实例, 提供给 <i>cax</i>

axes 属性:

Property	Description
orientation	vertical 或 horizontal (垂直或水平)
fraction	0.15; colorbar 所占 axes 的百分比
pad	0.05 (vertical 垂直), 0.15 (horizontal 水平); 两个 axes 的间距
shrink	1.0; colorbar 收缩比例
aspect	20; ratio of long to short dimensions (长宽比?)
anchor	(0.0, 0.5)[vertical 垂直]; (0.5, 1.0)[horizontal 水平]; colorbar 在 axes 的位置
panchor	(1.0, 0.5)[vertical 垂直]; (0.5, 0.0)[horizontal 水平]; colorbar 的源 axes(parent axes) 所在位置。若为 False, 源 axes 的位置不会改变

colorbar 属性:

Property	Description	
extend	[ 'neither'   'both'   'min'   'max' ]	若不是 'neither', 将不显示 colorbar 的尖端。该参数也可通过 set_under 和

		set_over 方法设置
extendfrac	[ None   'auto'   length   lengths ]	若为 None,两端的三角尖端长度占 colorbar 长度的 5%（默认设置）；若为 'auto', 三角尖端长度与 colorbar 平直部分长度一样（当 spacing[间距]设置为 'proportional'）；若为一个标量, 表征三角尖端长度占 colorbar 平直部分长度比例；若为[a,b](两个元素序列), 表征两个三角尖端分别占 colorbar 平直部分长度比例。
extendrect	[ False   True ]	若为 False, colorbar 将去掉平直部分, 只剩三角尖端；若为 True, 将去掉三角尖端, 只剩 colorbar 平直部分
spacing	[ 'uniform'   'proportional' ]	'uniform'使 colorbar 的每个色彩均匀间隔, 'proportional'则根据 data（数据）指定间隔
ticks	[ None   list of ticks   Locator object ]	若为 None, 刻度将根据导入的数据自动分配
format	[ None   format string   Formatter object ]	若为 None, 采用 ScalarFormatter；也可以输入格式字符串, 如: '%.3f'；也可以是 Formatter object
drawedges	[ False   True ]	是否绘制彩条边界

以下可能仅在索引 color 的情况下使用（当 norm=NoNorm()或其他非同寻常的情况下）：

boundaries	None 或者 sequence（一个序列）
values	None 或者 sequence（一个序列, 该序列的值低于 boundaries 的值）

## 25. matplotlib.pyplot.colors()

通用颜色：

Alias	Color
'b'	blue
'g'	green
'r'	red
'c'	cyan
'm'	magenta

'y'	yellow
'k'	black
'w'	white

其他色彩可通过 html 十六进制字符串指定，或者 html 颜色名称，如：

```
color = '#eeffff'
或者 color = 'burlywood'
color = 'chartreuse'
```

也可以通过 R,G,B 元组，其中每个 R,G,B 取值范围[0,1]

26.matplotlib.pyplot.contour(\*args, \*\*kwargs)

绘制等值线

contour(X, Y, Z, N) #N 表示等值线数量

可选重要参量：

colors:	[ None   string   (mpl_colors) ]	若为 None，颜色分配（colormap）由 cmap 指定；若为一个色彩字符串，如 colors='r'或 colors='red'，则所以标签显示这种颜色；若为一系列颜色（色彩字符串或数组），则不同等值线按顺序绘制不同色彩
alpha:	0-1	透明度
cmap:	[ None   Colormap ]	一个 colormap 实例，若 cmap=None 且 colors=None，则使用默认默认的 colormap
norm:	[ None   Normalize ]	一个 matplotlib.colors.Normalize 实例，将 data 缩放成色彩，若 norm=None 且 colors=None，使用默认线性缩放（linear scaling）
vmin, vmax:	[ None   scalar(标量) ]	若不为 None，其他值将被 matplotlib.colors.Normalize 实例支持，根据 levels 覆盖默认缩放色彩
levels:	[level0, level1, ..., leveln]	绘制一系列指定值等值线，例如： levels=[0]只绘制 0 等值线
origin:	[ None   'upper'   'lower'   'image' ]	若为 None，Z 的第一个值将对应的左下角 (0,0)位置。若为'image'，将使用 image.origin 对应的 rc 值。若 x、y 被

		contour () 函数调用, 该参数不会被激活
extent:	[ None   (x0,x1,y0,y1) ]	若 origin 为 None, extent 值来自 matplotlib.pyplot.imshow() : 提供像素坐标位置, 这种情况下, Z[0,0]是像素中心位置, 不是角落; 若 origin=None, (x0,y0)是 Z[0,0]位置,(x1,y1)是 Z[-1,-1]位置。若 x、y 被 contour () 函数调用, 该参数不会被激活
locator:	[ None   ticker.Locator subclass ]	若 locator=None, 使用默认 MaxNLocator。如果没有指定 levels, 该参数用于指定等值线 levels
extend:	[ 'neither'   'both'   'min'   'max' ]	Unless this is 'neither', contour levels are automatically added to one or both ends of the range so that all data are included. These added ranges are then mapped to the special colormap values which default to the ends of the colormap range, but can be set via matplotlib.colors.Colormap.set_under() and matplotlib.colors.Colormap.set_over() methods.
xunits, yunits:	[ None   registered units ]	通过指定 matplotlib.units.ConversionInterface 实例, 覆盖 axis 单位
antialiased:	[ True   False ]	抗锯齿

仅本函数有的重要参量:

linewidths:	[None number 一系列元组或列表]	若为 None, 使用默认线宽; 若为 number, 采用统一指定线宽; 若为列表或元组, 不同线宽按列表顺序排列
linestyles:	[ None   'solid'   'dashed'   'dashdot'   'dotted' ]	

## 27. matplotlib.pyplot.contourf(\*args, \*\*kwargs)

与 contourf()函数类似:

colors:	[ None   string   (mpl_colors) ]	若为 None，颜色分配 (colormap) 由 cmap 指定；若为一个色彩字符串，如 colors='r' 或 colors='red'，则所以标签显示这种颜色；若为一系列颜色（色彩字符串或数组），则不同等值线按顺序绘制不同色彩
alpha:	0-1	透明度
cmap:	[ None   Colormap ]	一个 colormap 实例，若 cmap=None 且 colors=None，则使用默认默认的 colormap
norm:	[ None   Normalize ]	一个 matplotlib.colors.Normalize 实例，将 data 缩放成色彩，若 norm=None 且 colors=None，使用默认线性缩放 (linear scaling)
vmin, vmax:	[ None   scalar(标量) ]	若不为 None，其他值将被 matplotlib.colors.Normalize 实例支持，根据 levels 覆盖默认缩放色彩
levels:	[level0, level1, ..., leveln]	绘制一系列指定值等值线，例如： levels=[0]只绘制 0 等值线
origin:	[ None   'upper'   'lower'   'image' ]	若为 None，Z 的第一个值将对应的左下角 (0,0)位置。若为'image'，将使用 image.origin 对应的 rc 值。若 x、y 被 contour () 函数调用，该参数不会被激活
extent:	[ None   (x0,x1,y0,y1) ]	若 origin 为 None，extent 值来自 matplotlib.pyplot.imshow()：提供像素坐标位置，这种情况下，Z[0,0]是像素中心位置，不是角落；若 origin=None，(x0,y0)是 Z[0,0]位置,(x1,y1)是 Z[-1,-1]位置。若 x、y 被 contour () 函数调用，该参数不会被激活
locator:	[ None   ticker.Locator subclass ]	若 locator=None，使用默认 MaxNLocator。如果没有指定 levels，该参数用于指定等值线 levels
extend:	[ 'neither'   'both'   'min'   'max' ]	Unless this is 'neither', contour levels are automatically added to one or both ends of

		the range so that all data are included. These added ranges are then mapped to the special colormap values which default to the ends of the colormap range, but can be set via matplotlib.colors.Colormap.set_under() and matplotlib.colors.Colormap.set_over() methods.
xunits, yunits:	[ None   registered units ]	通过指定 matplotlib.units.ConversionInterface 实例, 覆盖 axis 单位
antialiased:	[ True   False ]	抗锯齿

仅本函数有的重要参量:

nchunk:	[ 0   integer ]	若为 0, 区域不进行细分。表征填充区域划分精度
hatches:		一系列交叉填充图案。若为 None, 没有 hatch 添加到 contour。Hatching 仅被 PostScript, PDF, SVG 和 Agg backends 支持

28. matplotlib.pyplot.csd(x, y, NFFT=256, Fs=2, Fc=0, detrend=<function detrend\_none at 0x2635de8>, window=<function window\_hanning at 0x2635b90>, noverlap=0, pad\_to=None, sides='default', scale\_by\_freq=None, hold=None, \*\*kwargs)

绘制交叉谱密度。

```
csd(x, y, NFFT=256, Fs=2, Fc=0, detrend=mlab.detrend_none, window=mlab.window_hanning, noverlap=0, pad_to=None, sides='default', scale_by_freq=None, **kwargs)
```

交叉谱密度  $P_{xy}$  基于 Welch's 平均周期图法。

NFFT:	integer	每排的 FFT 数据点数。必须均匀, 2 的平方最有效, 默认值 256。不能用于获取零填充 (补零), 否则结果错误, 使用 pad_to 替代。
Fs:	scalar 标量	采样频率 (样品每时间单位[time unit])。它是用来计算傅立叶频率,freqs,周期中每时间单位(in cycles per time unit)。默认值为 2
detrend:	callable ?	应用在每个 fft 之前的函数, 旨在消除平均或线性



		趋势。不像在 MATLAB, <code>detrend</code> 参数是一个向量, 而在 <code>matplotlib</code> 是一个函数。Pylab 模块定义了 <code>detrend_none()</code> , <code>detrend_mean()</code> , and <code>detrend_linear()</code> 函数, 也可以用户自定义。
<code>window:</code>	callable 或 ndarray (python 的某种数组)	一个函数或向量的傅立叶变换长度。创建窗口向量出现 <code>window_hanning()</code> , <code>window_none()</code> , <code>NumPy.blackman()</code> , <code>NumPy.hamming()</code> , <code>NumPy.bartlett()</code> , <code>SciPy.signal()</code> , <code>scipy.signal.get_window()</code> , 等。默认的是 <code>window_hanning()</code> , 如果一个函数作为参数传递, 它必须以一个数据段作为参数和返回段的窗口版本。
<code>pad_to:</code>	integer	执行 FFT 时, 数据段垫点的个数。不同于 NFFT 指定使用的数据点的数目。在不增加 PSD 的实际分辨率 (分辨峰之间的最小距离) 的情况下, 可以在图中获取更多的点。这对应于调用 <code>fft()</code> 中的 <code>n</code> 参数, 默认为 <code>None</code> , 可设置 <code>pad_to</code> 使等同于 NFFT。
<code>sides:</code>	['default'   'onesided'   'twosided']	指定返回哪一侧的 PSD, 默认 'default', 它返回 one-sided 真实数据和复杂的数据。'one-sided' 返回一侧, 'twosided' 返回两侧
<code>scale_by_freq:</code>	boolean 布尔数	指定是否返回由缩放频率缩放的密度值, 单位 $\text{Hz}^{-1}$ 。这允许集成在返回的频率值。MATLAB 兼容默认是 <code>True</code> 的。
<code>noverlap:</code>	integer	重叠的块之间的点的数量。默认值是 0 (无重叠)。
<code>Fc:</code>	integer	X 的中心频率 (缺省为 0)。offsets the x extents of the plot to reflect the frequency range used when a signal is acquired and then filtered and downsampled to baseband.

其他参数参见 [Line2D](#) 属性。

## 29. matplotlib.pyplot.delaxes(\*args)

删除当前 figure 的 axes

## 30. matplotlib.pyplot.draw()

重新绘制当前 figure, 用于交互式环境。

```
fig.canvas.draw()
```

31. `matplotlib.pyplot.errorbar(x, y, yerr=None, xerr=None, fmt='-',  
' , ecolor=None, elinewidth=None, capsize=3, barsabove=False, lolims=False,  
uplims=False, xlolims=False, xuplims=False, errorevery=1, capthick=None, h  
old=None, **kwargs)`

绘制 errorbar 图表，如图：

```
errorbar(x, y, yerr=None, xerr=None, fmt='-', ecolor=None,  
elinewidth=None, capsize=3, barsabove=False, lolims=False, uplims=False,  
xlolims=False, xuplims=False, errorevery=1, capthick=None)
```

yerr 表示垂直方向的误差线，xerr 表示水平方向的误差线。

可选重要参量：

xerr/yerr:	[ scalar   N, Nx1, or 2xN array-like ]	若为标量(scalar)、N、Nx1 数组， errorbar 绘制各个点对应的+/-值；若为 2xN，errorbar 绘制为-row1 和 row2(row1 为 2xN 的第一列，row2 为 2xN 的第二列)
fmt:	'-'	errorbar 绘制线型符号
ecolor:	[ None   mpl color ]	matplotlib 色彩
elinewidth:	scalar	errorbar 的线宽，类似与 linewidth
capsize:	scalar	errorbar 棒长
capthick:	scalar	errorbar 棒 cap 的宽度
barsabove:	[ True   False ]	errorbar 图层在 plot 图层的上(True)、 下(False)
lolims / uplims / xlolims / xuplims:	[ False   True ]	These arguments can be used to indicate that a value gives only upper/lower limits. In that case a caret symbol is used to indicate this. lims-arguments may be of the same type as xerr and yerr.
errorevery:	positive integer(正整数)	errorbar 长度

其他参量参见 [Line2D](#)属性

32`matplotlib.pyplot.eventplot(positions, orientation='horizontal', lineoffsets=1, linelengths=1, linewidths=None, colors=None, linestyle='solid', hold=None, **kwargs)`

绘制条形线、条形码。

```
eventplot(positions, orientation='horizontal', lineoffsets=0,
linelengths=1, linewidths=None, color =None, linestyle='solid')
```

在给定的位置绘制平行线，位置由一维或二维数组指定，每行对应一个 row 或 column。

用于离散事件分析。

orientation :	[ 'horizontal'   'vertical' ]水平或垂直方向	'horizontal'使每一个条形横放、整体呈垂直排列；'vertical'使每一个条形竖放、整体呈水平排列
lineoffsets :	float(浮点数)或浮点型数组序列	线性偏移
linelengths :	float(浮点数)或浮点型数组序列	线长
linewidths :	float(浮点数)或浮点型数组序列	线宽
colors	RGBA 元组序列、及列表数组	
linestyles :	[ 'solid'   'dashed'   'dashdot'   'dotted' ]或这些参数列表	

其他属性为[LineCollection](#)属性，参见[PolyCollection](#)实例。

### 33. matplotlib.pyplot.figimage(\*args, \*\*kwargs)

为 figure 添加非重采样图像(non-resampled image)

```
figimage(X, xo, yo)
```

X 必须是浮点型数组

若 X is MxN, 默认亮度（灰度）

若 X is MxNx3, 默认 RGB

若 X is MxNx4, 默认 RGBA

Keyword	描述
xo or yo	整数数，x、y 方向像素偏移量
cmap	Colormap 实例，例如 cm.jet.默认为 rc image.cmap
norm	Normalize 实例，默认 normalization(), 该标量亮度 0-1
vmin vmax	用于缩放图像以 0-1 的亮度。若为 None，亮度值的最小值和最大值将被使用。注意，如果你通过一个规范的实例，为 Vmin 和 Vmax 的设置将被忽略。
alpha	$\alpha$ -混合值，默认为 None
origin	[ 'upper'   'lower' ], 指定[0,0]点的位置为左上角或左下角

### 34. matplotlib.pyplot.figlegend(handles, labels, loc, \*\*kwargs)

为 figure 添加标题

labels	一系列字符串
handles	一系列 Line2D 或 Patch 实例（参见 Line2D 实例）
loc	指定标签位置，可以是整数数，也可以是形如'upper right'的字符串

loc 参数:

Location String	Location Code
'best'	0
'upper right'	1
'upper left'	2
'lower left'	3
'lower right'	4
'right'	5
'center left'	6
'center right'	7
'lower center'	8
'upper center'	9
'center'	10

35. matplotlib.pyplot.**figtext**(\*args, \*\*kwargs)

为 figure 添加 text:

```
text(x, y, s, fontdict=None, **kwargs)
```

x、y 坐标（0-1 表示位于 figure 位置）表示 text 所在位置。

具体参数查看 **Text**属性

36.

matplotlib.pyplot.**figure**(num=None, figsize=None, dpi=None, facecolor=None, edgecolor=None, frameon=True, FigureClass=<class 'matplotlib.figure.Figure'>, \*\*kwargs)

创建一个新 figure

num	整数或字符串，可选，默认 None。Num=n 选择图表 n 或创建图表 n
figsize	创建指定大小的图像，单位：尺寸。Figuresize=(m,n)
dpi	每英寸的像素数，默认 80
facecolor	背景颜色
edgecolor	边框颜色

返回 **Figure** 参数。

37. matplotlib.pyplot.**fill**(\*args, \*\*kwargs)

绘制填充图，填充 x 轴和曲线 y 之间区域。

```
ax.fill(x1, y1, 'g', x2, y2, 'r')
```

返回 [Polygon](#) 属性，其他参数可见 [Polygon](#) 实例。

38. `matplotlib.pyplot.fill_between(x, y1, y2=0, where=None, interpolate=False, hold=None, **kwargs)`

绘制填充图，填充 x 区间内不同曲线之间区域。

```
fill_between(x, y1, y2=0, where=None, **kwargs)
```

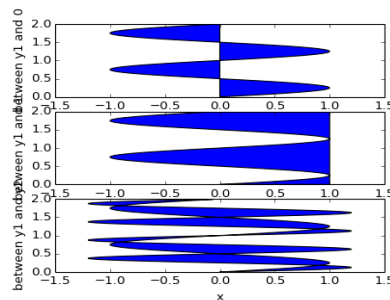
x 指定填充区间，填充 y1 和 y2 之间部分。其中 `where` 是填充条件，如 `where=y1>y2`，则仅绘制 y1 与 y2 之间区域；若为 `None`，则对数组 x 指定范围全部填充；若为布尔数组，则填充 `True` 部分。

`interpolate` 表示插入额外区域，默认为 `False`，针对的是 x 轴区间范围。

其它参数参见 [Polygon](#) 属性。

39. `matplotlib.pyplot.fill_betweenx(y, x1, x2=0, where=None, hold=None, **kwargs)`

绘制填充图，填充 y 区间内不同 x 的函数曲线之间区域，如图：



```
fill_betweenx(y, x1, x2=0, where=None, **kwargs)
```

y 指定填充区间，填充 x1 和 x2 之间部分。其中 `where` 是填充条件，如 `where=y1>y2`，则仅绘制 y1 与 y2 之间区域；若为 `None`，则对数组 x 指定范围全部填充；若为布尔数组，则填充 `True` 部分。

`interpolate` 表示插入额外区域，默认为 `False`，针对的是 x 轴区间范围。

其它参数参见 [Polygon](#) 属性。

40. `matplotlib.pyplot.findobj(o=None, match=None, include_self=True)`

寻找 artist 目标。递归寻找所有包含在 self 的 Artist 实例。

match	None: 返回包含 artist 的所有目标；利用布尔函数 <code>boolean = match(artist)</code> 过滤匹配；class 实例，例如 <code>Line2D</code> ，只返回 class 类型
-------	--

若 `include_self = True`，将确认匹配项的 self 属性

41. `matplotlib.pyplot.gca(**kwargs)`

返回当前 axes 实例，可用于控制 axis 属性、或通过 set 函数设置

```
plot(t,s)
set(gca(), 'xlim', [0,10])
```

或

```
plot(t,s)
a = gca()
a.set_xlim([0,10])
```

42. matplotlib.pyplot.gcf()

返回当前 figure 实例。

43. matplotlib.pyplot.gci()

获取当前色彩 artist, 返回当前 **ScalarMappable**实例(image 或 patch collection)

44. matplotlib.pyplot.get\_figlabels()

返回 figure 标签列表

45. matplotlib.pyplot.get\_fignums()

返回所有 figure 数字排列序列

46. matplotlib.pyplot.grid(b=None, which='major', axis='both', \*\*kwargs)

是否显示图轴网格

```
grid(self, b=None, which='major', axis='both', **kwargs)
```

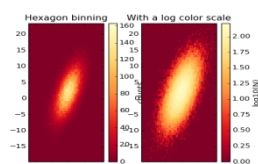
主要参数:

b	[True False]或者是布尔数组、或['on','off']	表示网格是否开启
which	[major(默认) minor both]	选择主、次网格开启方式
axis	[both(默认) x y]	选择使用网格的数轴

其他参数见 Line2D 属性。

47.matplotlib.pyplot.hexbin(x, y, C=None, gridsize=100, bins=None, xscale='linear', yscale='linear', extent=None, cmap=None,norm=None, vmin=None, vmax=None, alpha=None, linewidths=None, edgecolors='none', reduce\_C\_function=<function mean at 0x16ea d70>, mincnt=None, marginals=False, hold=None, \*\*kwargs)

绘制 hexagonal binning 图:



```
hexbin(x, y, C = None, gridsize = 100, bins = None,xscale = 'linear',
yscale = 'linear',cmap=None, norm=None, vmin=None,
vmax=None,alpha=None, linewidths=None,
edgecolors='none',reduce_C_function = np.mean, mincnt=None,
marginals=True,**kwargs)
```

必须参数:

x、y	一维等长数组	
C	[None(默认) 与 x、y 等长的一维数组]	若 C=None, 在(x[i],y[i])生成对应柱状图; 若 C 为指定值, 则根据相关函数指定(x[i],y[i])点值, 这些值根据 reduce_C_function 运算

可选参数:

gridsize:	[ 100(默认)   integer ]	x 方向的 hexagons 数, 默认 100, 而 y 方向自动调整与 x 方向相近似。另外, 也可以指定一个两个元素的元组(a,b),分别指定 x、y 方向
bins:	[ None   'log'   integer   sequence ]	若为 None, 不进行分级, 每个 hexagon 的色彩值直接对应计算值; 若为'log', 使用对数刻度的 color map, 用来确定 hexagon 内部颜色; 若为 interger, 按指定值划分 bins 值, 并自动分配 hexagons 颜色; 若为一系列值, 将使用 bins 下边界值
xscale:	[ 'linear'   'log' ]	指定水平轴线性刻度、指数刻度
yscale:	[ 'linear'   'log' ]	指定垂直轴线性刻度、指数刻度
mincnt:	[ None   一个正整数 ]	若不为 None,只显示超过 mincnt 数的 cell
marginals:	[ True   False ]	若为 True, 根据 colormap 矩形框、沿着 x 轴(底部)和 y 轴(左侧)绘制边缘密度
extent:	[ None   scalars (left, right, bottom, top) ]	bins 的限制 (取值范围), 默认根据 gridsize, x, y, xscale,yscale 分配

其他参量控制 color mapping 和 normalization 参量:

cmap	Colormap 实例, 例如 cm.jet.默认为 rc image.cmap
norm	Normalize 实例, 默认 normalization(), 该标量亮度 0-1
vmin vmax	用于缩放图像以 0-1 的亮度。若为 None, 亮度值的最小值和最大值将被使用。注意, 如果你通过一个规范的实例, 为 Vmin 和 Vmax 的设置将被忽略。
alpha	$\alpha$ -混合值, 默认为 None
linewidths:	[ None   scalar ]

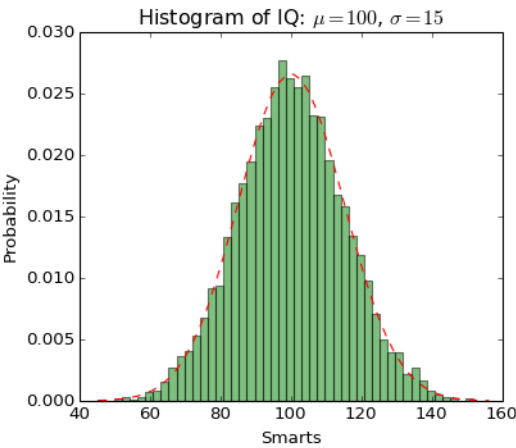
其他参量控制 Collection 属性:

edgecolors:	[ None   'none'   mpl color   color sequence ]	若为'none', 边缘绘制相同颜色; 若为 None, 使用默认颜色填充 outlines; 若为 matplotlib 颜色或序列 RGBA 元组, 用指定颜色绘制 outlines
-------------	--	---

其他参量参见[PolyCollection](#)实例。

```
48.matplotlib.pyplot.hist(x, bins=10, range=None, normed=False, weights=
None, cumulative=False, bottom=None,histtype='bar', align='mid', orientatio
n='vertical', rwidth=None, log=False, color=None, label=None, stacked=False,
hold=None,**kwargs)
```

绘制直方图



x :	array_like, shape (n, )	形如 shape(n,)的数组
bins :	integer 或 array_like, 可选, 默认: 10	若给定一个整数, 返回一个 bins+1 个条柱
range :	tuple(元组), 可选, 默认: None	bins(横坐标)的最大、最小值, 默认 range=(x.min(), x.max())
normed :	布尔量, 可选, 默认 : False	若为 True, 按正常方式处理直方图
weights :	形如 shape(n,)的数组, 可选, 默认: None	权重数组, 与 x 数组长度一致
cumulative :	布尔量, 可选, 默认	
histtype :	['bar'   'barstacked'   'step'   'stepfilled'], optional	'bar'是一个杆式直方图; "barstacked"是一个 条形直方图有多个数据堆叠在彼此顶部; 'step'生成悬空未填充线条; "stepfilled"产 生填充线条
align :	['left'   'mid'   'right'],可选, 默认: 'mid'	指定条(bar)的起点位置, 左侧、中间、右 侧
orientation :	['horizontal'   'vertical'], optional	
rwidth :	scalar, optional, default: None	相对宽度
log :	boolean, optional, default	x 轴是否为 log 刻度



color :	color or array_like of colors, 可选, 默认: None	
label :	字符串, 可选, 默认 : "	
stacked :	布尔参量, 可选, 默认	

其他参量参见 [matplotlib.patches.Polygon](#) 实例。

49.matplotlib.pyplot.hist2d(x, y, bins=10, range=None, normed=False, weights=None, cmin=None, cmax=None, hold=None, \*\*kwargs)

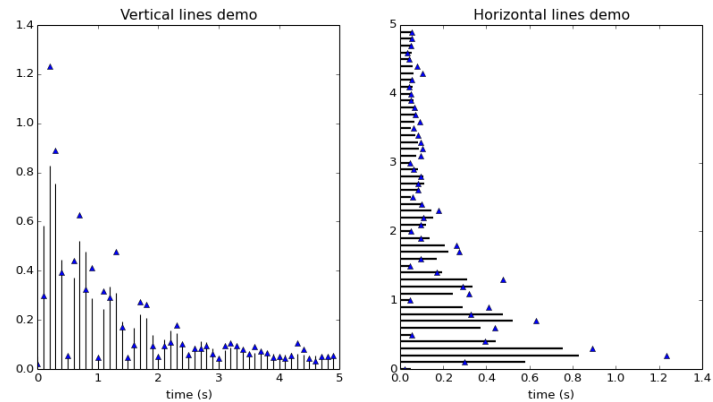
绘制二维直方图

x, y:	array_like, shape (n, ) :	形如 shape(n,)的数组
bins:	[None   int   [int, int]   array_like   [array, array]] :	若为 int, 指定两维的 bins(n <sub>x</sub> =n <sub>y</sub> =bins) ; 若为 [int,int]分别指定 x、y 方向 bins(n <sub>x</sub> , n <sub>y</sub> = bins) ; 若为数组(array_like), 指定两维的 bin:(x <sub>edges</sub> =y <sub>edges</sub> =bins) ; 若为 [array,array],指定 x、y 方向 bins : (x <sub>edges</sub> , y <sub>edges</sub> = bins)
range :	形如 shape(2,2)的数组, 可选, 默认: None	指定区间[[xmin, xmax], [ymin, ymax]], 这个范围以外的所有的值都被认为是不吻合直方图的异常值
normed :	布尔型, 可选, 默认 : False	归一化直方图
weights :	形如 shape(n,)的数组, 可选, 默认 : None	每个(x <sub>i</sub> , y <sub>i</sub> )点对应的权重值
cmin :	标量, 可选, 默认 : None	所有低于 cmin 值的 bins 将不被显示, 并且返回值被设为 nan upon
cmax :	标量, 可选, 默认 : None	所有高于 cmax 值的 bins 将不被显示, 并且返回值被设为 nan upon

返回值为: '(counts, xedges, yedges, Image)'

50.matplotlib.pyplot.hlines(y, xmin, xmax, colors='k', linestyle='solid', label="", hold=None, \*\*kwargs)

从每个 y 区间的 xmin 到 xmax 范围绘制水平线



参数:

y:	标量或一维数组	指定绘图区间
xmin, xmax:	标量或一维数组	指定每条线的起始值、终止值, 若给定了标量: 所有线长度相同
colors:	形如数组颜色, 可选, 默认: 'k'	
linestyles:	['solid'   'dashed'   'dashdot'   'dotted'], 可选	
label:	字符串, 可选, 默认: ''	

其他参数见 [LineCollection](#) 实例。

#### 51. matplotlib.pyplot.hold(b=None)

b=None 默认为 True。若为 True, 之后的命令也会被添加到当前 axes; 若为 False, 当前 axes 命令会被清除, 而只绘制接下来输入的命令。

#### 52. matplotlib.pyplot.imread(\*args, \*\*kwargs)

读取 image 图像文件, 将信息转化为数组。

fname	文件路径	必须以二进制形式打开
读取形式	None,r+,r,w+,c	

#### 53. matplotlib.pyplot.imsave(\*args, \*\*kwargs)

参数:

fname:	文件路径及文件名
arr:	一个 M×N (亮度), mxnx3 (RGB) 或 mxnx4 (RGBA) 数组。

重要参数:

vmin/vmax:	[ None   scalar ]	设置颜色范围区间
cmap:		Colormap 实例, 若为 None, 默认 rc image.cmap
format:		一个由后台支持的文件扩展名。大多数的后台支持 PNG, PDF, PS, EPS 和 SVG。

origin	['upper'   'lower']	指定[0,0]点的位置为左上、左下
dpi		存储在文件元数据的 DPI。这并不影响输出图像的分辨率。

54matplotlib.pyplot.imshow(X, cmap=None, norm=None, aspect=None, interpolation=None, alpha=None, vmin=None, vmax=None, origin=None, extent=None, shape=None, filternorm=1, filterrad=4.0, imlim=None, resample=None, url=None, hold=None, \*\*kwargs)

在 axes 上显示 image 图像

cmap :	Colormap,可选, 默认 : None	若为 None, 默认 rc image.cmap 值。 当 x 包含 RGB(A)信息, cmap 则被忽略
aspect :	['auto'   'equal'   scalar],可选, 默认 : None	若为'auto', 改变图像的纵横比以匹配当前 axes ; 若为'equal', (1)且 extent=None, 改变图像的纵横比以匹配当前 axes,(2)若 extent 不为 None, 改变 axes 的纵横比以匹配当前图像 ; 若为 None,默认 rc image.aspect 值
interpolation :	None(默认),字符串,可选: 'none', 'nearest', 'bilinear', 'bicubic', 'spline16', 'spline36', 'hanning', 'hamming', 'hermite', 'kaiser', 'quadric', 'catrom', 'gaussian', 'bessel', 'mitchell', 'sinc', 'lanczos'	若 interpolation=None(无差值),默认为 rc image.interpolation。详情参见手册
norm :	Normalize,可选,默认: None	通过 Normalize 实例, 使亮度标准化为 0-1;若为 None,使用默认 func:normalize 标准化方法。 Norm 仅用于 x 是浮点数组
vmin, vmax :	标量, 可选, 默认 : None	指定亮度范围, 若 norm 实例已指定, 该参数无效
alpha :	标量, 可选, 默认 : None	0-1
origin :	['upper'   'lower'],可选, 默认 : None	指定[0,0]点的位置为左上、左下

extent :	标量(left, right, bottom, top), 可选, 默认 : None	指定 x、y 像素中心
shape :	标量(columns, rows), 可选, 默认 : None	详情见参考书
filtnorm :	标量, 可选, 默认 : 1	详情见参考书
filtrrad :	标量, 可选, 默认 : 4	详情见参考书

返回值: **image** : **AxesImage**

55. matplotlib.pyplot.**ioff()**

关闭绘图交互模式

56 matplotlib.pyplot.**ion()**

打开绘图交互模式

57. matplotlib.pyplot.**ishold()**

返回当前 axes 的 hold 状态

58. matplotlib.pyplot.**isinteractive()**

返回绘图交互模式状态

59. matplotlib.pyplot.**legend(\*args, \*\*kwargs)**

在当前 axes 放置 legend 标签

```
legend((line1, line2, line3), ('label1', 'label2', 'label3'), loc=2,
bbox_to_anchor = (0.5, 0.5))
```

其中(line1, line2, line3)和('label1', 'label2', 'label3')一一对应。loc 表示标签位置:

Location String	Location Code
'best'	0
'upper right'	1
'upper left'	2
'lower left'	3
'lower right'	4
'right'	5
'center left'	6
'center right'	7
'lower center'	8
'upper center'	9
'center'	10

而 **bbox\_to\_anchor=(m,n)** 指定具体位置:m 表示横坐标位置(0-1 之间), n 表示纵坐标位置(0-1 之间)。

numpoints:	整数	legend 图例中 line 的点数
scatterpoints:	整数	scatter 图 legend 图例中点数
scatteryoffsets:	浮点数列表	yoffsets 列表
markerscale:	[ None   scalar ]	相对于原图例标记的大小
frameon:	[ True   False ]	图例边框是否开启
fancybox:	[ None   False   True ]	图例边框是否花边
shadow:	[ None   False   True ]	图例阴影是否开启
framealpha:	[None   float]	图例透明度
ncol :	整数	列数, 默认 : 1
mode :	[ “expand”   None ]	若为'expand', 图例将沿横轴 扩展到最长
bbox_to_anchor:	BboxBase 实例或 2or4 个字符型 元组	?
bbox_transform :	[ an instance of Transform   None ]	?
title		注释标题

图例内部各个元素的参数设置:

<b>Keyword</b>	
borderpad	边框和说明文字间距
labelspacing	图例和行距
handlelength	图例中线条(legend handles)的长度
handletextpad	线条(legend handles)和文字的间距(文字和符号间距)
borderaxespad	图例和最靠近的两条坐标轴间距
columnspacing	不同列的间距

60. matplotlib.pyplot.locator\_params(axis='both', tight=None, \*\*kwargs)

控制轴刻度标签

关键参数:

axis	[ 'x'   'y'   'both' ]	选择坐标轴, 默认 : 'both'
tight	[ True   False   None ]	默认为 None, 不改变。将参量传输到 autoscale_view()

其他参量直接访问[set\\_params\(\)](#)。

## 61. matplotlib.pyplot.loglog(\*args, \*\*kwargs)

x、y 轴均为指数刻度

重要参数：

basex/basey:	scalar > 1	x、y 轴刻度的底数
subsx/subsy:	[ None   sequence ]	x、y 轴刻度最小值。详情参见 matplotlib.axes.Axes.set_xscale() / matplotlib.axes.Axes.set_yscale()
nonposx/nonposy :	['mask' (屏蔽)  'clip'(转换)]	屏蔽小于 0 的数；或转换负数转换为接近 0 的 数

其他参量参见[Line2D](#)属性。

## 62. matplotlib.pyplot.margins(\*args, \*\*kw)

设置或检索自动缩放功能。返回值 xmargin, ymargin

```
margins(margin)
margins(xmargin, ymargin)
margins(x=xmargin, y=ymargin)
margins(..., tight=False)
```

## 63. matplotlib.pyplot.matshow(A, fignum=None, \*\*kw)

将数组以矩阵的形式在 figure 窗口显示。

fignum:	[ None   integer   False ]	布尔型数组，默认创建一个新的窗口；若为 False 或 0，则不创建
origin	['upper'   'lower'],可选,默认,None	指定[0,0]的位置

## 64. matplotlib.pyplot.minorticks\_off()

关闭轴线次刻度

## 65. matplotlib.pyplot.minorticks\_on()

开启轴线次刻度

## 66. matplotlib.pyplot.over(func, \*args, \*\*kwargs)

以 hold(True)的方式调用函数，并保存当前状态。

## 67. matplotlib.pyplot.pause(interval)

暂停指定时间，单位：秒

## 68. matplotlib.pyplot.pcolor(\*args, \*\*kwargs)

绘制二维数组的伪彩色图。

```
pcolor(C, **kwargs)
pcolor(X, Y, C, **kwargs)
```

C 为颜色值。重要参数：

cmap :	Colormap,可选, 默认 : None	若为 None, 默认 rc image.cmap 值。当 x 包含 RGB(A)信息, cmap 则被忽略
--------	---------------------------	---

norm :	Normalize,可选,默认: None	通过 Normalize 实例, 使亮度标准化为 0-1;若为 None,使用默认 func:normalize 标准化方法。 Norm 仅用于 x 是浮点数组
vmin, vmax :	标量, 可选, 默认: None	指定亮度范围, 若 norm 实例已指定, 该参数无效
shading:	[ 'flat'   'faceted' ]	若为 'faceted', 在每个矩形绘制一个黑色的网格 ; 若为'flat', 边缘将不绘制。
edgecolors:	[ None   'none'   color   color sequence]	若为 None,使用默认 rc 设置 ; 若为'none', 边缘不可见 ; 一个 MPL 序列色彩设置边缘的颜色
alpha:	0 <= scalar <= 1 或 None	透明度混合值

其他参数见 [PolyCollection](#)实例。

69. matplotlib.pyplot.**pcolormesh**(\*args, \*\*kwargs)

绘制一个四边形网格, 创建一个二维阵列的伪彩色图。

```

pcolormesh(C)
pcolormesh(X, Y, C)
pcolormesh(C, **kwargs)

```

cmap :	Colormap,可选, 默认: None	若为 None, 默认 rc image.cmap 值。当 x 包含 RGB(A)信息, cmap 则被忽略
norm :	Normalize,可选,默认: None	通过 Normalize 实例, 使亮度标准化为 0-1;若为 None,使用默认 func:normalize 标准化方法。 Norm 仅用于 x 是浮点数组
vmin, vmax :	标量, 可选, 默认: None	指定亮度范围, 若 norm 实例已指定, 该参数无效
shading:	[ 'flat'   'faceted' ]	若为 'faceted', 在每个矩形绘制一个黑色的网格 ; 若为'flat', 边缘将不绘制。
edgecolors:	[ None   'none'   color   color sequence]	若为 None,使用默认 rc 设置 ; 若为'none', 边缘不可见 ; 一个 MPL 序列色彩设置边缘的颜色
alpha:	0 <= scalar <= 1 或 None	透明度混合值

其他参数见[PolyCollection](#)实例:

70.matplotlib.pyplot.**pie**(x, explode=None, labels=None, colors=None, autopct=None, pctdistance=0.6, shadow=False,labeldistance=1.1, startangle=No

*ne, radius=None, hold=None)*

绘制饼图

```
pie(x, explode=None, labels=None, colors=('b', 'g', 'r', 'c', 'm', 'y',  
'k', 'w'), autopct=None, pctdistance=0.6, shadow=False,  
labeldistance=1.1, startangle=None, radius=None)
```

关键参数:

explode:	[ None   len(x) sequence ]	指定饼图分片数
colors:	[ None   color sequence ]	matplotlib 颜色
labels:	[ None   len(x) sequence of strings ]	指定每个分片的标签
autopct:	[ None   format string   format function ]	标签中数字格式
pctdistance:	标量	饼图和标签比例? 默认 0.6
labeldistance:	标量	标签距饼中心距离
shadow:	[ False   True ]	饼图阴影
startangle:	[ None   Offset angle ]	若不为 None, 从 x 轴起逆时针旋转
radius:	[ None   scalar ]	饼的半径, 默认为: 1

71. matplotlib.pyplot.**plot**(\*args, \*\*kwargs)

详情参见 [Line2D](#)属性

72.matplotlib.pyplot.**plot\_date**(x, y, fmt='bo', tz=None, xdate=True, ydate=False, hold=None, \*\*kwargs)

绘图添加日期。该函数类似于 plot() 函数, 除了 x 或 y (或者两者) 为日期, 并且轴坐标标签也是日期。

关键参数:

fmt:	string	图格式字符串
tz:	[ None   timezone(时区) string   tzinfo instance ]	时区使用标记的日期; 若为 None, 默认 rc 值
xdate:	[ True   False ]	若为 True, x 轴标记日期
ydate:	[ True   False ]	若为 True, y 轴标记日期

其他参数见[Line2D](#)属性。

73.matplotlib.pyplot.**plotfile**(fname, cols=(0, ), plotfuncs=None, comments='#', skiprows=0, checkrows=5, delimiter=',', names=None, subplots=True, newfig=True, \*\*kwargs)

直接绘制文件数据(跳过读取文件命令)。



*cols*: 用于鉴别 column 内容。若为 int，表征列数；若为字符串，表征列标题(label)，列标题的空格符将被'\_'下划线取代，并删除所以非法字符。

若 `len(cols) == 1`，仅绘制指定列(column)的数据；若 `len(cols) > 1`，第一列(column)数据作为 x 轴数据，其他列数据分别为 y 轴数据；若 `subplots=True`，分别将各个曲线绘制在多个子图；若 `subplots=False`，多个曲线只绘制在一个图。

plotfuncs	'plot','semilogy','fill', 'bar',等	选择绘图函数
newfig	[ True   False ]	若为 True，每次 plot 都创建新绘图窗口；若为 False，在原窗口绘图

*comments*, *skiprows*, *checkrows*, *delimiter*, 和 *names* 通过

`matplotlib.pyplot.csv2rec()`将数据载入到记录数组。

74. `matplotlib.pyplot.polar(*args, **kwargs)`

绘制极坐标图

```
polar(theta, r, **kwargs)
```

theta 表示旋转角度，r 表示旋转半径

75.`matplotlib.pyplot.psd(x, NFFT=256, Fs=2, Fc=0, detrend=<function detrend_none at 0x2635de8>, window=<function window_hanning at 0x2635b90>, noverlap=0, pad_to=None, sides='default', scale_by_freq=None, hold=None, **kwargs)`

绘制功率谱密度图

```
psd(x, NFFT=256, Fs=2, Fc=0, detrend=mlab.detrend_none,  
window=mlab.window_hanning, noverlap=0, pad_to=None,sides='default',  
scale_by_freq=None, **kwargs)
```

向量 x 分割进入 NFFT 长度分段，每个分段通过 *detrend* 函数和 *window* 函数去趋势，*nooverlap* 为每段重叠区域长度。 $|fft(i)|^2$  通过每段*i*的平均值来计算 *Pxx*，通过一个标准来纠正功率损耗。*Fs* 是采样频率。关键参数：

NFFT:	integer	每排的 FFT 数据点数。必须均匀，2 的平方最有效，默认值 256。不能用于获取零填充（补零），否则结果错误，使用 <code>pad_to</code> 替代。
Fs:	scalar 标量	采样频率（样品每时间单位[time unit]）。它是用来计算傅立叶频率,freqs,周期中每时间单位(in cycles per time unit)。默认值为 2
detrend:	callable ?	应用在每个 fft 之前的函数，旨在消除平均或线性趋势。不像在 MATLAB，detrend 参数是一个向量，而在 matplotlib 是一个函数。Pylab 模块定

		义了 <code>detrend_none()</code> , <code>detrend_mean()</code> , and <code>detrend_linear()</code> 函数，也可以用户自定义。
window:	callable 或 ndarray (python 的某种数组)	一个函数或向量的傅立叶变换长度。创建窗口向量出现 <code>window_hanning()</code> , <code>window_none()</code> , <code>NumPy.blackman()</code> , <code>NumPy.hamming()</code> , <code>NumPy.bartlett()</code> , <code>SciPy.signal()</code> , <code>scipy.signal.get_window()</code> , 等。默认的是 <code>window_hanning()</code> ，如果一个函数作为参数传递，它必须以一个数据段作为参数和返回段的窗口版本。
pad_to:	integer	执行 FFT 时，数据段垫点的个数。不同于 NFFT 指定使用的数据点的数目。在不增加 PSD 的实际分辨率（分辨峰之间的最小距离）的情况下，可以在图中获取更多的点。这对应于调用 <code>fft()</code> 中的 <code>n</code> 参数，默认为 <code>None</code> ，可设置 <code>pad_to</code> 使等同于 NFFT。
sides:	[ 'default'   'onesided'   'twosided' ]	指定返回哪一侧的 PSD，默认 'default'，它返回 one-sided 真实数据和复杂的数据。'one-sided' 返回一侧，'twosided' 返回两侧
scale_by_freq:	boolean 布尔数	指定是否返回由缩放频率缩放的密度值，单位 $\text{Hz}^{-1}$ 。这允许集成在返回的频率值。MATLAB 兼容默认是 <code>True</code> 的。
noverlap:	integer	重叠的块之间的点的数量。默认值是 0（无重叠）。
Fc:	integer	X 的中心频率（缺省为 0）。offsets the x extents of the plot to reflect the frequency range used when a signal is acquired and then filtered and downsampled to baseband.

其他参数参见 [Line2D](#) 属性。

#### 76. matplotlib.pyplot.quiver(\*args, \*\*kw)

绘制二维箭头

`quiver(X, Y, U, V, C, **kw)`

X, Y:	箭头的位置坐标(默认是箭头尾部)
-------	------------------

U, V:	箭头在 x、y 方向的分量
C:	箭头颜色，可选参量

重要参量：

units:	[ 'width'   'height'   'dots'   'inches'   'x'   'y'   'xy' ]	箭头的尺寸单位；除了长度都是这个单位的倍数。 'width'或'height':axes 的宽度或长度；'dots'或 'inches':基于 figure 的像素或英寸；'x','y',或'xy':X, Y,或 $\sqrt{X^2+Y^2}$ 的数据单位
--------	---	--

若 *units* = 'width'或 'height'，调整坐标轴的长或宽，箭头大小也随之变化；若

*units*='dots'或 'inches'，调整坐标轴长或宽不影响箭头大小。

angles:	[ 'uv'   'xy'   array ]	默认'uv'，箭头横纵方向比例为 1 : 1；若 为'xy'，箭头由(x,y)指向(x+u, y+v)；可用 数组指定箭头方向。
scale:	[None float]	浮点型数据，指定一个长度为单位长度， 如：scale=2inches(单位由 scale_units 确定) ，则风速为 2m/s 在图中显示为 1 英寸 (inches)
scale_units:	None,或其它单位,如： 'inches','width','x'	若为'x'，则和 x 轴单位一样；若为 'width',单位为 axes 宽度
width:	scalar, 默认 0.005	箭头宽度，与 scale_units 和 scale 的选取有 关，默认为 0.005*图轴宽度
headwidth:	scalar, 默认 3	尖端宽度，默认 3
headlength:	scalar, 默认 5	尖端长度，默认 5
headaxislength :	scalar, 默认 4.5	箭头两翼长度，默认 4.5
minshaft:	scalar, 默认 1	长度低于 scale 值的箭头，尖端长度设置为 默认值 1。尽量不要低于 1，否则小箭头很 难看。
minlength:	scalar, 默认 1	长度低于该值的箭头不被绘制，而只绘制 一个(六边形)点
pivot:	[ 'tail'   'middle'   'tip' ]	支点位置，箭头围绕该点旋转

其他参量参见[PolyCollection](#)属性。

77. matplotlib.pyplot.**quiverkey**(\*args, \*\*kw)

为 quiver 图像添加注释标签

```
quiverkey(Q, X, Y, U, label, **kw)
```

参量:

Q:	quiver 函数的返回值
X, Y:	注释标签的位置
U:	标准风速值对应的箭头。如 U=10，则出现表示 10m/s 的箭头
label:	字符串标签

其他参量:

coordinates	['axes' 'figure'  'data'   'inches']	坐标系，‘axes’和‘figure’表示(0,0)点 为坐标左下角，(1,1)点为坐标左上角； ‘data’根据 quiver 图坐标确定； ‘inches’：通过该参量表征点在图的位置， (0,0)为左下角
color:	matplotlib 颜色	图例箭头颜色
labelpos	['N' 'S' 'E' 'W']	标签相对于图例箭头'->'的位置
labelsep:	默认：0.1	箭头和标签之间的距离，单位：英寸， 默认值：0.1
labelcolor:	matplotlib 颜色	字体颜色
fontproperties:	字体字典：family, style, variant, size, weight	FontProperties 参量

78. matplotlib.pyplot.rc(\*args, \*\*kwargs)

设置当前 rc 参量，包括 lines.linewidth 群的 lines，axes.facecolor 群的 axes 等。如：

```
rc('lines', linewidth=2, color='r')
rcParams['lines.linewidth'] = 2
rcParams['lines.color'] = 'r'
font = {'family' : 'monospace',
        'weight' : 'bold',
        'size'   : 'larger'}
rc('font', **font) # pass in the font dict as kwargs
```

79. matplotlib.pyplot.rc\_context(rc=None, fname=None)

返回一个用于管理 RC 设置的文管理器。

参见有关资料

80. matplotlib.pyplot.rcdefaults()

恢复默认的 RC 参数。

81. matplotlib.pyplot.rgrids(\*args, \*\*kwargs)

获取或设置极坐标图的径向辐射网格线。

```
lines, labels = rgrids()
lines, labels = rgrids(radial, labels=None, angle=22.5, **kwargs)
```

当不调用参数，`rgrid()`仅返回元组(`lines`, `labels`)，其中 `lines` 是辐射网格数组 (`Line2D`实例)，`labels` 是一个标签数组(`Text`实例)。当调用参数，标签会在指定的位置出现(radial distances and angles)。 `labels` 包含 `len(radial)`组标签字符串。例子：

```
# set the locations of the radial gridlines and labels
lines, labels = rgrids( (0.25, 0.5, 1.0) )

# set the locations and labels of the radial gridlines and labels
lines, labels = rgrids( (0.25, 0.5, 1.0), ('Tom', 'Dick', 'Harry')
)
```

## 82. matplotlib.pyplot.savefig(\*args, \*\*kwargs)

保存当前 figure

```
savefig(fname, dpi=None, facecolor='w', edgecolor='w',
        orientation='portrait', papertype=None, format=None,
        transparent=False, bbox_inches=None, pad_inches=0.1,
        frameon=None)
```

fname: 文件路径及文件名

关键参数:

dpi:	[ None   scalar > 0 ]	每英寸的分辨率
facecolor, edgecolor:	matplotlib 色彩	figure 图像颜色
orientation:	[ 'landscape'   'portrait' ]	不支持所有的后台；目前只能在 PostScript 输出
papertype:	'letter', 'legal', 'executive', 'ledger', 'a0' through 'a10', 'b0' through 'b10'等	仅支持 PostScript 输出
format:	文件扩展名	png, pdf, ps, eps 和 svg 等.
transparent:	透明度？	？
frameon:	[True False]	背景图是否显示
bbox_inches:	'tight'	边缘空白区域大小设置
pad_inches:	？	？
bbox_extra_artists:	？	？

## 83. matplotlib.pyplot.sca(ax)

设置当前 axes 实例

## 84.matplotlib.pyplot.scatter(x, y, s=20, c='b', marker='o', cmap=None, norm=None, vmin=None, vmax=None, alpha=None, linewidths=None, verts=N

`one, hold=None, **kwargs)`

绘制散点图，其中 X 和 Y 是相同长度的数组序列

<code>x, y :</code>	形如 <code>shape (n,)</code> 数组	输入数据
<code>s :</code>	标量或形如 <code>shape (n,)</code> 数组，可选，默认：20	<code>size in points^2</code>
<code>c :</code>	色彩或颜色序列，可选，默认	注意 C 不应是一个单一的 RGB 数字或 RGBA 序列，因为不便区分。C 可以是一个 RGB 或 RGBA 二维行数组
<code>marker :</code>	<code>MarkerStyle</code> , 可选，默认: 'o'	详情参阅 <code>markers</code> 属性
<code>cmap :</code>	<code>Colormap</code> 可选，默认：None	<code>Colormap</code> 实例
<code>norm :</code>	<code>Normalize</code> 可选，默认：None	数据亮度 0-1, float 数据
<code>vmin, vmax :</code>	标量，可选，默认：None	亮度设置，若 <code>norm</code> 实例已使用，该参数无效
<code>alpha :</code>	标量，可选，默认：None	0-1
<code>linewidths :</code>	标量或数组，默认：None	.....

85. `matplotlib.pyplot.sci(im)`

设置当前图像

86. `matplotlib.pyplot.semilogx(*args, **kwargs)`

使图像 x 轴为对数刻度。

<code>basex:</code>	<code>scalar &gt; 1</code>	x 轴对数底数
<code>subsx:</code>	[ None   sequence ]	刻度序列
<code>nonposx:</code>	['mask' (屏蔽)  'clip'(转换)]	屏蔽小于 0 的数；或转换负数转换为接近 0 的数

其他参数见 [Line2D](#) 属性。

87. `matplotlib.pyplot.semilogy(*args, **kwargs)`

使图像 y 轴为对数刻度。

<code>basey:</code>	<code>scalar &gt; 1</code>	y 轴对数底数
<code>subsy:</code>	[ None   sequence ]	刻度序列
<code>nonposy:</code>	['mask' (屏蔽)  'clip'(转换)]	屏蔽小于 0 的数；或转换负数转换为接近 0 的数

其他参数见 [Line2D](#) 属性。

88. `matplotlib.pyplot.set_cmap(cmap)`

设置应用于当前 image 的 colormap。cmap 参数必须是一个 [Colormap](#)

89. `matplotlib.pyplot.setp(*args, **kwargs)`

设置 property 属性

90. `matplotlib.pyplot.show(*args, **kw)`

显示 figure

91. `matplotlib.pyplot.specgram(x, NFFT=256, Fs=2, Fc=0, detrend=<function detrend_none at 0x2635de8>, window=<function window_hanning at 0x2635b90>, noverlap=128, cmap=None, xextent=None, pad_to=None, sides='default', scale_by_freq=None, hold=None, **kwargs)`

绘制光谱图，计算和绘制图谱在 x 的数据位长度的数据分成段，每段的 PSD 计算。窗函数的窗口被施加到每个段，每一段的重叠量由 `nooverlap` 指定。

NFFT:	integer	每排的 FFT 数据点数。必须均匀，2 的平方最有效，默认值 256。不能用于获取零填充（补零），否则结果错误，使用 <code>pad_to</code> 替代。
Fs:	scalar 标量	采样频率（样品每时间单位[time unit]）。它是用来计算傅立叶频率,freqs,周期中每时间单位(in cycles per time unit)。默认值为 2
detrend:	callable ?	应用在每个 fft 之前的函数，旨在消除平均或线性趋势。不像在 MATLAB，detrend 参数是一个向量，而在 matplotlib 是一个函数。 Pylab 模块定义了 <code>detrend_none()</code> , <code>detrend_mean()</code> , and <code>detrend_linear()</code> 函数，也可以用户自定义。
window:	callable 或 ndarray (python 的某种数组)	一个函数或向量的傅立叶变换长度。创建窗口向量出现 <code>window_hanning()</code> , <code>window_none()</code> , <code>NumPy.blackman()</code> , <code>NumPy.hamming()</code> , <code>NumPy.bartlett()</code> , <code>SciPy.signal()</code> , <code>scipy.signal.get_window()</code> , 等。默认的是 <code>window_hanning()</code> ，如果一个函数作为参数传递，它必须以一个数据段作为参数和返回段的窗口版本。
pad_to:	integer	执行 FFT 时，数据段垫点的个数。不同于 NFFT 指定使用的数据点的数目。在不增加 PSD 的实际分辨率（分辨峰之间的最小距离）的情况下，可以在图中获取更多的点。这对应于调用 <code>fft()</code> 中的 <code>n</code> 参数，默认为 <code>None</code> ，可设置 <code>pad_to</code> 使等同于 NFFT。
sides:	['default']	指定返回哪一侧的 PSD，默认 'default'，它返

	'onesided'   'twosided' ]	回 one-sided 真实数据和复杂的数据。'one-sided'返回一侧， 'twosided'返回两侧
scale_by_freq:	boolean 布尔数	指定是否返回由缩放频率缩放的密度值，单位 Hz <sup>-1</sup> 。这允许集成在返回的频率值。 MATLAB 兼容默认是 True 的。
noverlap:	integer	重叠的块之间的点的数量。默认值是 0（无重叠）。
Fc:	integer	X 的中心频率（缺省为 0）。offsets the x extents of the plot to reflect the frequency range used when a signal is acquired and then filtered and downsampled to baseband.
cmap:	指定 colormap 实例	一个 matplotlib.colors.Colormap 实例；若为 None，使用默认 rc 实例
xextent:	(xmin,xmax)	xextent = (xmin,xmax)，指定 image 在 x 轴显示范围。默认(0,max(bins))，其中 bins 是 specgram()的返回值

返回值为(Pxx, freqs, bins, im):

bins	光谱计算对应时间点
freqs	频率数组
Pxx	an array of shape (len(times), len(freqs)) of power
im	AxesImage 实例

92.matplotlib.pyplot.**spy**(Z, precision=0, marker=None, markersize=None, aspect='equal', hold=None, \*\*kwargs)

```
spy(Z, precision=0, marker=None, markersize=None,
    aspect='equal', **kwargs)
```

spy(Z)绘制二维数组 Z 的稀疏模式(sparsity pattern)。

precision	float、'present'	只绘制 z >precision 的数据；若为'present'，所有数据都被绘制
aspect	'equal'(默认)、 'auto'、标量	若为'equal'，数组元素在空间均匀分配；若为'auto'，图像可能会充满绘图框；若为标量，直接指定每个元素占用空间比例

其它参量参见[Line2D](#)和[Line2D](#)属性。

93. matplotlib.pyplot.**stackplot**(x, \*args, \*\*kwargs)

绘制区域堆叠(stacked area )图？



x	一维数组，元素个数为 N
y	MxN 型二维数组，或 1xN 型一维数组

数据默认非堆叠，调用形式：

```
stackplot(x, y) # where y is MxN
stackplot(x, y1, y2, y3, y4) # where y1, y2, y3, y4, are all 1xNm
```

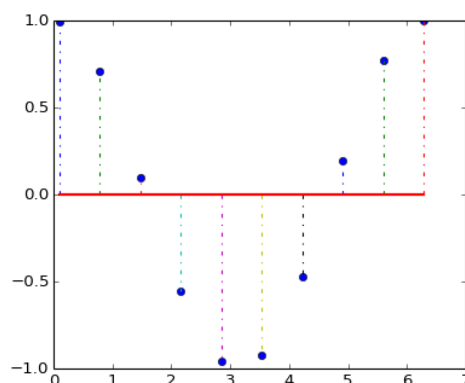
参量：

baseline :	['zero', 'sym', 'wigggle', 'weighted_wigggle']	['zero'为简单的堆积图；'sym'关于 0 对称，或称为 ThemeRiver(河流图)；'wigggle'斜度最小平方和；'weighted_wigggle'每层根据大小考虑重量。
colors :	一系列色彩元组	这些值用于循环填充堆积区域

其它参量与 `fill_between()` 相同。返回值 *r* : 一系列 [PolyCollection](#) 属性。

#### 94. matplotlib.pyplot.stem(\*args, \*\*kwargs)

绘制 stem 图。



```
stem(y, linefmt='b-', markerfmt='bo', basefmt='r-')
stem(x, y, linefmt='b-', markerfmt='bo', basefmt='r-')
```

绘制 x-y 函数曲线的每个点到水平基线(y=0 轴线)的垂线，*linefmt* 表示垂线线型，*markerfmt* 表示 x-y 函数曲线的每个点的点标 marker，*basefmt* 表示基线的线型。返回值为(*markerline*, *stemlines*, *baseline*)元组。

#### 95. matplotlib.pyplot.step(x, y, \*args, \*\*kwargs)

绘制阶图？

```
step(x, y, *args, **kwargs)
```

x、y 为一维数组，并且假设 x 均匀增长。

where:	['pre'   'post'   'mid']	若为 'pre'，从 x[i]到 x[i+1]为 y[i+1]插值？若为 'post'，插值为
--------	--------------------------	---

		y[i] ? 若为‘mid’，取平均值？
--	--	----------------------

其他参数与`plot()`相同。

96. `matplotlib.pyplot.streamplot(x, y, u, v, density=1, linewidth=None, color=None, cmap=None, norm=None, arrowsize=1, arrowstyle='>', minlength=0.1, transform=None, hold=None)`

绘制流线图

x, y :	一维数组	生成一个均匀分布的网格
u, v :	二维数组	列数与 x 长度匹配；行数与 y 长度匹配
density :	字符串或	控制流线之间的紧密程度(流线密度)，对于不同方向的密度，使用[density_x, density_y]
linewidth :	数字或二维数组	改变线宽
color :	matplotlib 色彩代码，或二维数组	color 值转化为色彩使用 cmap 参数
cmap :	Colormap	用于将数组转化为颜色
norm :	0-1	亮度数据，用于 color 值为数组时
arrowsize :	float	表征箭头大小
arrowstyle :	箭头字符串	箭头样式，参见 FancyArrowPatch
minlength :	float	流线最短长度

97. `matplotlib.pyplot.subplot(*args, **kwargs)`

在一个图表中绘制多个子图。

`subplot(nrows, ncols, plot_number, axisbg, polar, projection)`

绘图区分为 `nrows` 行和 `ncols` 列，从左到右、从上到下顺序对每个区域进行编号，左上区编号为 1。`nrows` 表示图的行数，`ncols` 表示图的列数，`plot_number` 表示第几个子图。上述三个参数均小于 10 时，中间逗号可以省略。

`axisbg=color`，表示所选子图背景颜色。`polar=False`，是否为极地投影，默认为否。

`projection`：使用指定的投影方式，该方式使用之前需要已经注册(registered)。

98. `matplotlib.pyplot.subplot2grid(shape, loc, rowspan=1, colspan=1, **kwargs)`

在网格中创建 subplot。网格由 `shape` 指定，`loc` 指定位置，`rowspan`, `colspan` 生成行和列。

99. `matplotlib.pyplot.subplot_tool(targetfig=None)`

打开图像的 subplot 工具窗口。

100. `matplotlib.pyplot.subplots(nrows=1, ncols=1, sharex=False, sharey=False, squeeze=True, subplot_kw=None, **fig_kw)`

绘制已包含多个子图的图。

nrows :	int	subplot 网格行数, 默认 : 1
ncols :	int	subplot 网格列数, 默认 : 1
sharex :	布尔参量或字符串 (“row”, “col”, “all”, or “none”)	若为 True : 对所有子图共用 x 轴(最后一行除外), 'all' 等同于 True ; ‘None’ 等同于 False ; 若为 ‘col’ 每列子图共享 x 轴 ; 若为 'row' 每行子图共享 x 轴
sharey :	布尔参量或字符串 (“row”, “col”, “all”, or “none”)	若为 True : 对所有子图共用 y 轴(最后一行除外), 'all' 等同于 True ; ‘None’ 等同于 False ; 若为 ‘col’ 每列子图共享 y 轴 ; 若为 'row' 每行子图共享 y 轴
squeeze :	bool	若为 True, 额外维度被剔除 ; 若为 False, 没有改变, 返回二维数组
subplot_kw :	字典 dict	查询 add_subplot()
fig_kw :	字典 dict	figure()

返回值(fig, ax)均为元组。

101. matplotlib.pyplot.subplots\_adjust(\*args, \*\*kwargs)

调整 subplot 布局。

```
subplots_adjust(left=None, bottom=None, right=None, top=None,  
                wspace=None, hspace=None)
```

各个参数取值 0-1.

102. matplotlib.pyplot.suptitle(\*args, \*\*kwargs)

为 figure 添加总标题。参数均为Text属性, 可查询Text。

默认参数:

x :	0.5	文本横坐标
y :	0.98	文本纵坐标
horizontalalignment :	‘center’	文本的水平对齐方式
verticalalignment :	‘top’	文本的垂直对齐

103. matplotlib.pyplot.switch\_backend(newbackend)

交换默认后台。

104. matplotlib.pyplot.table(\*\*kwargs)

为当前 axes 添加 table。

```
table(cellText=None, cellColours=None, cellLoc='right', colWidths=None,  
      rowLabels=None, rowColours=None, rowLoc='left', colLabels=None,  
      colColours=None, colLoc='center', loc='bottom', bbox=None):
```

参数:

agg_filter	unknown
alpha 透明度	float (取值范围 0.0-1.0 )
animated 动画	[True   False]
axes	an Axes instance
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
contains	一个可调用函数
figure	a matplotlib.figure.Figure instance 设置 figure 示例
fontsize	a float in points
gid	一个 id 字符串
label 标签	以'%s'形式显示的字符串及可显示的参数
lod	[True   False]
path_effects	unknown
picker	[None float boolean callable]
rasterized	[True   False   None]
sketch_params	unknown
snap	unknown
transform	Transform instance
url	a url string
visible	[True   False]
zorder	any number

105.matplotlib.pyplot.**text**(x, y, s, fontdict=None, withdash=False, \*\*kwargs)

在 axes 图轴添加注释。通用参量:

s :	字符串	注释内容
x, y :	scalars 标量	text 注释坐标
fontdict :	字典, 可选, 默认 : 无	用于重写默认文本属性。Fontdict=None, 缺省值是由 rc 参数确定。
withdash :	布尔参量, 可选, 默认 :	创建一个 TextWithDash 实例, 而不是 Text 实

	False	例
--	-------	---

其他参量见Text属性。

106. matplotlib.pyplot.thetagrids(\*args, \*\*kwargs)

设置极坐标网格  $\theta$  位置。

```
lines, labels = thetagrids()
```

或:

```
lines, labels = thetagrids(angles, labels=None, fmt='%d', frac = 1.1)
```

*lines* 为Line2D实例, *labels* 为Text实例

107. matplotlib.pyplot.tick\_params(axis='both', \*\*kwargs)

改变刻度及刻度标签的外形。参量:

axis :	['x'   'y'   'both']	选择操作的轴线，默认 : both
reset :	[True   False]	默认 False，若为 True，所有参量返回默认状态
which :	['major'   'minor'   'both']	默认为'major'，选择主次刻度
direction :	['in'   'out'   'inout']	刻度放置在 axes 内侧、外侧、两侧都有
length	float	刻度线长度
width	float	刻度线宽度
color	颜色	刻度线颜色
pad	float	刻度与标签间距
labelsize	float 或名称，如 : 'large'	刻度标签字体大小
labelcolor	颜色	标签颜色
colors	颜色	同时指定刻度和标签颜色
zorder	int	顺序
bottom, top, left, right :	[bool   'on'   'off']	控制各个轴线是否显示刻度
labelbottom, labeltop, labelleft, labelright	布尔参量或['on'   'off']	控制各个轴线是否显示刻度标签

108. matplotlib.pyplot.ticklabel\_format(\*\*kwargs)

更改标签数据格式。可选参量:

Keyword	Description	
---------	-------------	--

style	[ 'sci' (or 'scientific')   'plain' ]	'plain'关闭科学计算法
scilimits	(m, n), pair of integers	(m, n)为一对整数。若 style='sci'，则大于 10 的数采用科学计数法；
useOffset	[True   False   offset]	若为 True，将根据需要计算偏移量；若为 False，不使用偏移量；也可以手动指定偏移量
axis	[ 'x'   'y'   'both' ]	选择轴线
useLocale	[True   False]	若为 True，数字格式根据当前情况而定；若为 False，采用 C 语言格式

109.matplotlib.pyplot.**tight\_layout**(pad=1.08, h\_pad=None, w\_pad=None, rect=None)

自动调节 subplot 参数进行指定填充。参数：

pad :	float	填充 figure 和子图边缘部分
h_pad, w_pad :	float	填充相邻子图边缘部分
rect :	(left, bottom, right, top)	指定子图填充区域，默认：(0, 0, 1, 1)

110. matplotlib.pyplot.**title**(s, \*args, \*\*kwargs)

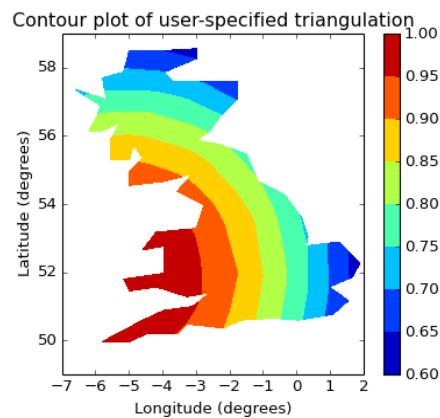
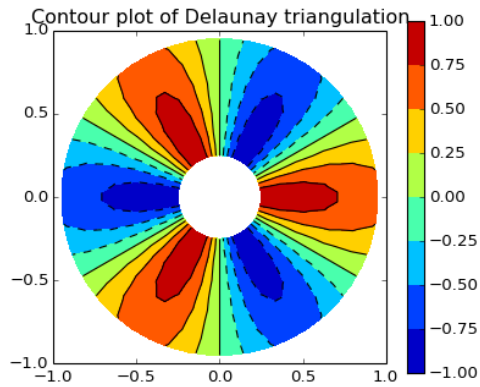
设置当前子图的标题。

label :	字符串	标题文本
fontdict :	字典	控制标题外观，如：{'fontsize': rcParams['axes.titlesize'],'verticalalignment': 'baseline', 'horizontalalignment': loc}
loc :	{'center', 'left', 'right'},字符串, 可选	标题位置，默认 center

返回text参量，详情参见text属性。

111. matplotlib.pyplot.**tricontour**(\*args, \*\*kwargs)

在三角形非结构网格绘制等值线。



```
tricontour(x, y, triangles, mask=mask, ...)
```

参量:

colors:	[ None   string   (mpl_colors) ]	若为 None，颜色分配 (colormap) 由 cmap 指定；若为一个色彩字符串，如 colors='r' 或 colors='red'，则所以标签显示这种颜色；若为一系列颜色（色彩字符串或数组），则不同等值线按顺序绘制不同色彩
alpha:	0-1	透明度
cmap:	[ None   Colormap ]	一个 colormap 实例，若 cmap=None 且 colors=None，则使用默认默认的 colormap
norm:	[ None   Normalize ]	一个 matplotlib.colors.Normalize 实例，将 data 缩放成色彩，若 norm=None 且 colors=None，使用默认线性缩放 (linear scaling)
vmin, vmax:	[ None   scalar(标量) ]	若不为 None，其他值将被 matplotlib.colors.Normalize 实例支持，根据 levels 覆盖默认缩放色彩
levels:	[level0, level1, ..., leveln]	绘制一系列指定值等值线，例如： levels=[0] 只绘制 0 等值线
origin:	[ None   'upper'   'lower'   'image' ]	若为 None，Z 的第一个值将对应的左下角 (0,0) 位置。若为 'image'，将使用 image.origin 对应的 rc 值。若 x、y 被 contour () 函数调用，该参数不会被激活
extent:	[ None   (x0,x1,y0,y1) ]	若 origin 为 None，extent 值来自

		matplotlib.pyplot.imshow() : 提供像素坐标位置, 这种情况下, Z[0,0]是像素中心位置, 不是角落; 若 origin=None, (x0,y0)是 Z[0,0]位置,(x1,y1)是 Z[-1,-1]位置。若 x、y 被 contour () 函数调用, 该参数不会被激活
locator:	[ None   ticker.Locator subclass ]	若 locator=None, 使用默认 MaxNLocator。如果没有指定 levels, 该参数用于指定等值线 levels
extend:	[ 'neither'   'both'   'min'   'max' ]	Unless this is 'neither', contour levels are automatically added to one or both ends of the range so that all data are included. These added ranges are then mapped to the special colormap values which default to the ends of the colormap range, but can be set via matplotlib.colors.Colormap.set_under() and matplotlib.colors.Colormap.set_over() methods.
xunits, yunits:	[ None   registered units ]	通过指定 matplotlib.units.ConversionInterface 实例, 覆盖 axis 单位

tricontour 函数仅有参量 :

linewidths:	[None number 一系列元组或列表]	若为 None, 使用默认线宽; 若为 number, 采用统一指定线宽; 若为列表或元组, 不同线宽按列表顺序排列
linestyles:	[ None   'solid'   'dashed'   'dashdot'   'dotted' ]	

## 112. matplotlib.pyplot.tricontourf(\*args, \*\*kwargs)

在三角形非结构网格绘制等值线填充图。参量:

colors:	[ None   string   (mpl_colors) ]	若为 None, 颜色分配 (colormap) 由 cmap 指定; 若为一个色彩字符串, 如 colors='r'或 colors='red', 则所以标签显
---------	----------------------------------	---



		示这种颜色；若为一系列颜色（色彩字符串或数组），则不同等值线按顺序绘制不同色彩
alpha:	0-1	透明度
cmap:	[ None   Colormap ]	一个 colormap 实例，若 cmap=None 且 colors=None，则使用默认默认的 colormap
norm:	[ None   Normalize ]	一个 matplotlib.colors.Normalize 实例，将 data 缩放成色彩，若 norm=None 且 colors=None，使用默认线性缩放（linear scaling）
vmin, vmax:	[ None   scalar(标量) ]	若不为 None，其他值将被 matplotlib.colors.Normalize 实例支持，根据 levels 覆盖默认缩放色彩
levels:	[level0, level1, ..., leveln]	绘制一系列指定值等值线，例如： levels=[0]只绘制 0 等值线
origin:	[ None   'upper'   'lower'   'image' ]	若为 None，Z 的第一个值将对应的左下角 (0,0)位置。若为'image'，将使用 image.origin 对应的 rc 值。若 x、y 被 contour () 函数调用，该参数不会被激活
extent:	[ None   (x0,x1,y0,y1) ]	若 origin 为 None，extent 值来自 matplotlib.pyplot.imshow()：提供像素坐标位置，这种情况下，Z[0,0]是像素中心位置，不是角落；若 origin=None，(x0,y0)是 Z[0,0]位置,(x1,y1)是 Z[-1,-1]位置。若 x、y 被 contour () 函数调用，该参数不会被激活
locator:	[ None   ticker.Locator subclass ]	若 locator=None，使用默认 MaxNLocator。如果没有指定 levels，该参数用于指定等值线 levels
extend:	[ 'neither'   'both'   'min'   'max' ]	Unless this is 'neither', contour levels are automatically added to one or both ends of the range so that all data are included. These added ranges are then mapped to the special colormap values which default to

		the ends of the colormap range, but can be set via matplotlib.colors.Colormap.set_under() and matplotlib.colors.Colormap.set_over() methods.
xunits, yunits:	[ None   registered units ]	通过指定 matplotlib.units.ConversionInterface 实例, 覆盖 axis 单位

仅本函数有的重要参量:

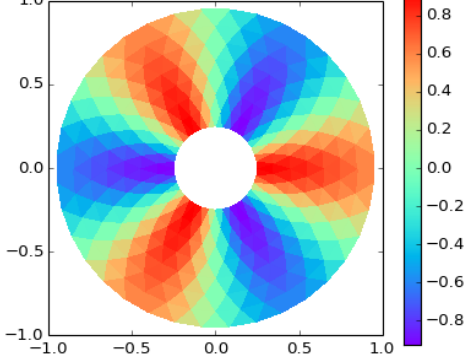
nchunk:	[ 0   integer ]	若为 0，区域不进行细分。表征填充区域 划分精度
hatches:		一系列交叉填充图案。若为 None，没有 hatch 添加到 contour。Hatching 仅被 PostScript, PDF, SVG 和 Agg backends 支持

113. matplotlib.pyplot.tripcolor(\*args, \*\*kwargs)

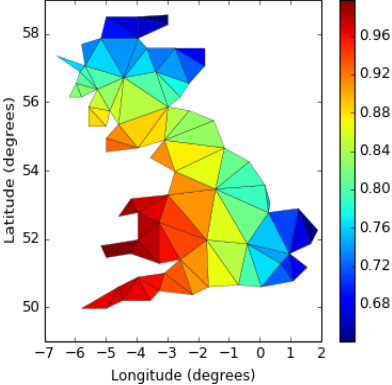
绘制非结构三角形网格伪彩色图。

tripcolor(x, y, triangles, mask=mask, ...)

tripcolor of Delaunay triangulation, flat shading



tripcolor of user-specified triangulation

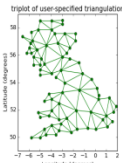
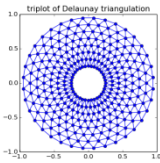


具体参数参见[pcolor\(\)](#)。

114. matplotlib.pyplot.triplot(\*args, \*\*kwargs)

绘制非结构三角形网格点线图。

triplot(x, y, triangles, mask=mask, ...)



具体参数参见[plot\(\)](#)。

115. `matplotlib.pyplot.twinx(ax=None)`

绘制与之前 `axes` 图轴共用 `x` 轴的 `axes` 图，新的 `axes` 图在右侧放置。若 `ax=None`，新的 `axes` 将覆盖 `ax`，返回 `ax2` 实例。

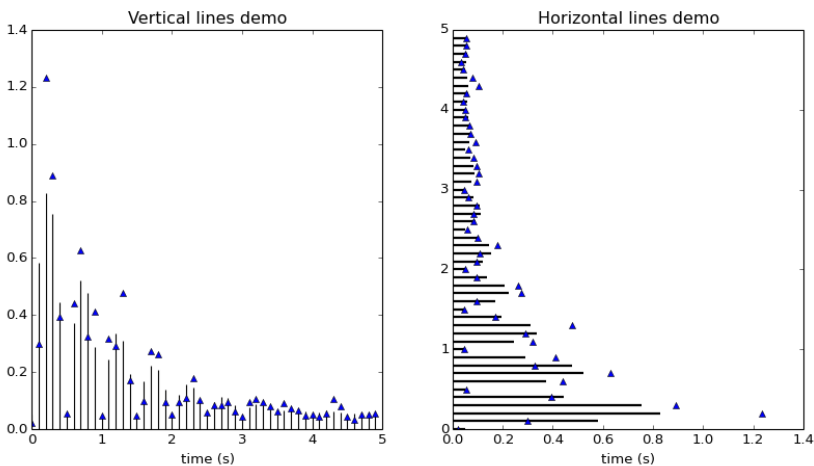
116. `matplotlib.pyplot.twinx(ax=None)`

绘制与之前 `axes` 图轴共用 `y` 轴的图，新的 `axes` 图在上方放置。若 `ax=None`，新的 `axes` 将覆盖 `ax`，返回 `ax2` 实例。

117. `matplotlib.pyplot.vlines(x, ymin, ymax, colors='k', linestyle='solid', label="", hold=None, **kwargs)`

绘制垂线，类似于 `hlines()` 函数(第 50 号)。

从每个 `x` 区间的 `ymin` 到 `ymax` 范围绘制水平线



参数：

<code>x</code> ：	标量或一维数组	指定绘图区间
<code>ymin</code> , <code>ymax</code> ：	标量或一维数组	指定每条线的起始值、终止值，若给定了标量：所有线长度相同
<code>colors</code> ：	形如数组颜色，可选，默认：'k'	
<code>linestyles</code> ：	['solid'   'dashed'   'dashdot'   'dotted'], 可选	
<code>label</code> ：	字符串，可选，默认：''	

其他参数见 [LineCollection](#) 实例。

118. `matplotlib.pyplot.xcorr(x, y, normed=True, detrend, usevlines=True, maxlags=10, hold=None, **kwargs)`

绘制 `x`、`y` 之间的相关性。

```
xcorr(self, x, y, normed=True, detrend=mlab.detrend_none, usevlines=True, maxlags=10, **kwargs)
```

若 `normed = True`，如果 `normed = True`，通过零滞后自相关规范数据，`x` 和 `y` 通过

detrend 调用（默认没有归一化）的方式除趋势。x 和 y 必须等长。

数据绘制形如 `plot(lags, c, **kwargs)`, 返回值是元组 `(lags, c, line)`, 其中:

\$lags 是一个长度为  $2 \times \text{maxlags} + 1$  的滞后 (lags) 向量

\$C 是  $2 \times \text{maxlags} + 1$  的自相关向量

\$line 是一个 `line2d` 实例, 通过 `plot()` 返回

默认 `linestyle` 是 `None`, 默认的 `marker` 是 'o', 交叉相关性是通过 `numpy.correlate()` 函数、`mode = 2` 实现的。

若 `usevlines` 是 `True`, `vlines()` 将被调用（而不调用 `plot()` 函数），用来绘制从起点到 `acorr` 的垂线。否则, `plot()` 由 `Line2D properties` 属性参数 (`kwargs`) 决定。

`maxlags` 是正整数, 决定 `lags` 的显示数目。默认值 `None` 将返回  $(2 \times \text{len}(x) - 1)$  个 `lags`, 返回值是一个元组 `(lags, c, linecol, b)`, 其中:

`linecol` 是 `linecollection`

`b` 是 `x-axis`

其他参数参见 `Line2D` 的 `kwargs` 属性

119. `matplotlib.pyplot.xkcd(scale=1, length=100, randomness=2)`

`matplotlib` 模块不包含的字体。

120. `matplotlib.pyplot.xlabel(s, *args, **kwargs)`

设置 x 轴标签

```
override = {
    'fontsize'          : 'small',
    'verticalalignment' : 'top',
    'horizontalalignment' : 'center'
}
```

详情参见 `text()` 函数及属性。

121. `matplotlib.pyplot.xlim(*args, **kwargs)`

设置当前轴线的取值范围。

```
xmin, xmax = xlim()    # return the current xlim
xlim( (xmin, xmax) )   # set the xlim to xmin, xmax
xlim( xmin, xmax )    # set the xlim to xmin, xmax
```

122. `matplotlib.pyplot.xscale(*args, **kwargs)`

设置 x 轴的缩放

```
xscale(scale, **kwargs)
```

scales 的可选参数: `['linear' | 'log' | 'symlog']`

'linear'		
'log'	basex/basey:	x、y 轴对数底数

	nonposx/nonposy:	['mask'   'clip'] : 若为'mask', 屏蔽非正数 ; 若为'clip', 将负数转换成接近 0 的正数
	subsx/subsy:	指定主刻度的标签及刻度, 如 : [2, 3, 4, 5, 6, 7, 8, 9]
'symlog'	basex/basey:	x、y 轴对数底数
	linthreshx/linthreshy:	指定一个区域为线性图, 避免在 0 附近达到无穷大
	subsx/subsy:	指定主刻度的标签及刻度, 如 : [2, 3, 4, 5, 6, 7, 8, 9]
	linscalex/linscaley:	将线性区域拉伸到对数区域, 使用底数为 10?

### 123. matplotlib.pyplot.xticks(\*args, \*\*kwargs)

设置刻度及对应标签。xticks(array,'a','b','c'...)其中 array 中每个元素与字符串'a','b'...一一对应, 使标签显示为指定字符串。

```
xticks( arange(5), ('Tom', 'Dick', 'Harry', 'Sally', 'Sue') )
```

其他参量参见[Text](#)属性。

### 124. matplotlib.pyplot.ylabel(s, \*args, \*\*kwargs)

设置 y 轴标签。

```
override = {
    'fontsize'          : 'small',
    'verticalalignment' : 'center',
    'horizontalalignment' : 'right',
    'rotation'='vertical' : }
```

详情参见[text\(\)](#)函数及属性。

### 125. matplotlib.pyplot.ylim(\*args, \*\*kwargs)

设置当前轴线的取值范围。

```
ymin, ymax = ylim()    # return the current ylim
ylim( (ymin, ymax) )   # set the ylim to ymin, ymax
ylim( ymin, ymax )     # set the ylim to ymin, ymax
```

### 126. matplotlib.pyplot.yscale(\*args, \*\*kwargs)

设置 y 轴的缩放

```
yscale(scale, **kwargs)
```

yscales 的可选参数: ['linear' | 'log' | 'symlog']

'linear'		
'log'	basex/basey:	x、y 轴对数底数

	nonposx/nonposy:	['mask'   'clip'] : 若为'mask', 屏蔽非正数 ; 若为'clip', 将负数转换成接近 0 的正数
	subsx/subsy:	指定主刻度的标签及刻度, 如 : [2, 3, 4, 5, 6, 7, 8, 9]
'symlog'	basex/basey:	x、y 轴对数底数
	linthreshx/linthreshy:	指定一个区域为线性图, 避免在 0 附近达到无穷大
	subsx/subsy:	指定主刻度的标签及刻度, 如 : [2, 3, 4, 5, 6, 7, 8, 9]
	linscalex/linscaley:	将线性区域拉伸到对数区域, 使用底数为 10?

#### 127. matplotlib.pyplot.yticks(\*args, \*\*kwargs)

```
# return locs, labels where locs is an array of tick locations and
# labels is an array of tick labels.
locs, labels = yticks()
# set the locations of the yticks
yticks( arange(6) )
# set the locations and labels of the yticks
yticks( arange(5), ('Tom', 'Dick', 'Harry', 'Sally', 'Sue') )
```

相关参数参见[Text](#)属性。

#### 128. matplotlib.pyplot.colormaps()

matplotlib 提供一系列彩色表, 并可通过[register\\_cmap\(\)](#)函数添加。可通过 image, pcolor, scatter, 等函数设置 colormap。例如:

```
imshow(X, cmap=cm.hot)
```

或者使用[set\\_cmap\(\)](#)函数:

```
imshow(X)
pyplot.set_cmap('hot')
pyplot.set_cmap('jet')
```

有几种通用颜色方案:

Sequential 方案	对单极性数据, 从低到高推进
Diverging 方案	双极性数据, 从中心向两级通过正负偏差
Cyclic 方案	用于绘制, 值在端点处环绕, 如相位角, 风的方向, 或一天的时间
Qualitative 方案	数据本身没有内在顺序, 颜色只是用来区分类别

附录一 **Line2D**属性的 kwargs:

Property	Description
agg_filter	unknown
alpha 透明度	float (0.0 transparent through 1.0 opaque)
animated 动画	[True   False]
antialiased or aa抗锯齿	[True   False]
axes	an Axes instance
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
color or c可选颜色	任何 matplotlib 颜色
contains	一个可调用函数
dash_capstyle [设置为虚线的 cap (帽子) 样式 (linestyles) ]	['butt' 一头大  'round' 圆的  'projecting'突出的]
dash_joinstyle [设置虚线的连接方式]	['miter' 斜接  'round' 圆的  'bevel'倾斜]
dashes	<p>一系列包含[on/off]的列表，表示每个像素点是否显示</p> <p>设置虚线序列，通过元组（或列表）的形式，设置每个像素点是否显示。</p>
drawstyle	<p>['default'   'steps'   'steps-pre'   'steps-mid'   'steps-post']</p> <p>Default 表示直线，</p>
figure	<p>a matplotlib.figure.Figure instance</p> <p>设置 figure 示例</p>
fillstyle填充方式	<p>['full'   'left'   'right'   'bottom'   'top'   'none']</p> <p>设置 marker 点的填充方式，full 表示全填充，none 表示不填充，其他为半填充</p>
gid	一个 id 字符串（设置 artist 属性的 id）
label标签	以'%s'形式显示的字符串及可显示的参数
linestyle or ls	['-'   '--'   '-.'   ':'   'None'   ''   '']

linewidth or lw线宽	用浮点型数据表示
lod	[True   False]
marker点属性	见下表 maker 属性
markeredgcolor or mec	任何 matplotlib 颜色 (marker 边缘颜色)
markeredgewidth or mew	用浮点型数据表示 (marker 边缘宽度)
markerfacecolor or mfc	任何 matplotlib 颜色 (marker 内部颜色)
markerfacecoloralt or mfcalt (marker 交替颜色)	任何 matplotlib 颜色 (marker 的交替颜色颜色)
markersize or ms	浮点型数据 (marker 大小)
markevery	None   integer   (startind, stride)
path_effects	unknown
picker	float distance in points or callable pick function fn(artist, event)
pickradius	float distance in points
rasterized	[True   False   None]
sketch_params	unknown
snap	unknown
solid_capstyle (为 solid 线的 cap[帽子]进行设置)	['butt' 一头大  'round' 圆的  'projecting'突出的]
solid_joinstyle[设置 solid 线连接方式]	['miter' 斜接  'round' 圆的  'bevel'倾斜]
transform	a matplotlib.transforms.Transform instance
url	a url string
visible	[True   False]
xdata	1D array
ydata	1D array
zorder	any number



附录二 matplotlib.patches.Polygon 实例：

Property	Description
agg_filter	unknown
alpha 透明度	float (取值范围 0.0-1.0 )
animated 动画	[True   False]
antialiased or aa抗锯齿	[True   False]
axes	an Axes instance
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
color or c可选颜色	任何 matplotlib 颜色
contains	一个可调用函数
edgecolor or ec[边缘颜色]	Matplotlib 色彩, or None 为默认, or 'none' 表示无色彩
facecolor or fc	Matplotlib 色彩, or None 为默认, or 'none' 表示无色彩
figure	a matplotlib.figure.Figure instance 设置 figure 示例
fill (填充)	[True   False]
gid	一个 id 字符串
hatch	['/'   '\'   ' '   '-'   '+'   'x'   'o'   'O'   '.'   '*']
label标签	以'%s'形式显示的字符串及可显示的参数
linestyle or ls	['solid'   'dashed'   'dashdot'   'dotted']
linewidth or lw	float or None for default
lod	[True   False]
path_effects	unknown
picker	[None float boolean callable]
rasterized	[True   False   None]
sketch_params	unknown
snap	unknown
transform	Transform instance
url	a url string
visible	[True   False]

zorder	any number
--------	------------

附录三 PolyCollection实例：

Property	Description
agg_filter	unknown
alpha	float (取值范围 0.0-1.0 )
animated	[True   False]
antialiased or antialiaseds (抗锯齿)	[True   False]或布尔数组 (表示 True   False)
array	unknown
axes	an Axes instance
clim	a length 2 sequence of floats
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
cmap	一个 colormap 或者已注册的 colormap 名字
color	任何 matplotlib 颜色
contains	一个可调用函数
edgecolor or edgecolors	matplotlib 颜色或一系列 RGBA 元组
facecolor or facecolors	matplotlib 颜色或一系列 RGBA 元组
figure	a matplotlib.figure.Figure instance
gid	一个 ID 字符串
hatch	[ '/'   '\'   ' '   '-'   '+'   'x'   'o'   'O'   ':'   '*' ]
label	string or anything printable with '%s' conversion.
linestyle or linestyles or dashes	['solid'   'dashed', 'dashdot', 'dotted'   (offset, on-off-dash-seq) ]
linewidth or lw or linewidths	float or sequence of floats
lod	[True   False]
norm	unknown

Property	Description
offset_position	unknown
offsets	float or sequence of floats
path_effects	unknown
picker	[None float boolean callable]
pickradius	unknown
rasterized	[True   False   None]
sketch_params	unknown
snap	unknown
transform	Transform instance
url	a url string
urls	unknown
visible	[True   False]
zorder	any number

Property	Description
agg_filter	unknown
alpha 透明度	float (0.0 transparent through 1.0 opaque)
animated 动画	[True   False]
antialiased or aa抗锯齿	[True   False]
array	unknown
axes	an Axes instance
clim	a length 2 sequence of floats
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
cmap	a colormap or registered colormap name
color or c可选颜色	任何 matplotlib 颜色
contains	一个可调用函数
edgecolor or edgecolors	matplotlib color arg or sequence of rgba tuples
facecolor or facecolors	matplotlib color arg or sequence of rgba tuples
figure	a matplotlib.figure.Figure instance 设置 figure 示例
gid	一个 id 字符串（设置 artist 属性的 id）
hatch	['/'   '\'   ' '   '-'   '+'   'x'   'o'   'O'   ':'   '*']
label标签	以'%s'形式显示的字符串及可显示的参数
linestyle or ls	['-'   '--'   '-'   ':'   'None'   ''   '']
linewidth or lw线宽	用浮点型数据表示
lod	[True   False]
norm	unknown
offset_position	unknown
offsets	float or sequence of floats

path_effects	unknown
picker	float distance in points or callable pick function fn(artist, event)
pickradius	float distance in points
rasterized	[True   False   None]
sketch_params	unknown
snap	unknown
transform	a matplotlib.transforms.Transform instance
url	a url string
urls	unknown
visible	[True   False]
zorder	any number

附录五      **Text属性:**

Property	Description
agg_filter	unknown
alpha透明度	float (0.0-1.0)
animated动画	[True   False]
axes	an Axes instance
backgroundcolor背景颜色	any matplotlib color
bbox	rectangle prop dict
clip_box	a matplotlib.transforms.Bbox instance
clip_on	[True   False]
clip_path	[ (Path, Transform)   Patch   None ]
color	any matplotlib color
contains	一个可调用函数
family or fontfamily or fontname or name	[FONTNAME   'serif'   'sans-serif'   'cursive'   'fantasy'   'monospace' ]
figure	a matplotlib.figure.Figure instance
fontproperties or font_properties	a matplotlib.font_manager.FontProperties instance
gid	an id string
horizontalalignment or ha	[ 'center'   'right'   'left' ]
label标签	string or anything printable with '%s' conversion.
linespacing	float (multiple of font size)
lod	[True   False]
multialignment	['left'   'right'   'center' ]
path_effects	unknown
picker	[None float boolean callable]
position	(x,y)
rasterized	[True   False   None]
rotation	[ angle in degrees   'vertical'   'horizontal' ]
rotation_mode	unknown

size or fontsize	[size in points   'xx-small'   'x-small'   'small'   'medium'   'large'   'x-large'   'xx-large' ]
sketch_params	unknown
snap	unknown
stretch or fontstretch	[a numeric value in range 0-1000   'ultra-condensed'   'extra-condensed'   'condensed'   'semi-condensed'   'normal'   'semi-expanded'   'expanded'   'extra-expanded'   'ultra-expanded' ]
style or fontstyle	[ 'normal'   'italic'   'oblique' ]
text	string or anything printable with '%s' conversion.
transform	Transform instance
url	a url string
variant or fontvariant	[ 'normal'   'small-caps' ]
verticalalignment or va or ma	[ 'center'   'top'   'bottom'   'baseline' ]
visible	[True   False]
weight or fontweight	[a numeric value in range 0-1000   'ultralight'   'light'   'normal'   'regular'   'book'   'medium'   'roman'   'semibold'   'demibold'   'demi'   'bold'   'heavy'   'extra bold'   'black' ]
x	float
y	float
zorder	any number



附录六 Marker 属性:

marker	description	描述
"."	point	点
","	pixel	像素
"o"	circle	圈
"v"	triangle_down	倒三角形
"^"	triangle_up	正三角形
"<"	triangle_left	左三角形
">"	triangle_right	又三角形
"1"	tri_down	tri_down
"2"	tri_up	tri_up
"3"	tri_left	tri_left
"4"	tri_right	tri_right
"8"	octagon	八角形
"s"	square	正方形
"p"	pentagon	五角
"*"	star	星星
"h"	hexagon1	六角 1
"H"	hexagon2	六角 2
"+"	plus	加号
"x"	x	x 号
"D"	diamond	钻石
"d"	thin_diamond	细钻
" "	vline	v 线
"_"	hline	H 线
TICKLEFT	tickleft	左刻度
TICKRIGHT	tickright	右刻度
TICKUP	tickup	上刻度
TICKDOWN	tickdown	下刻度
CARETLEFT	caretleft	caretleft
CARETRIGHT	caretright	caretright
CARETUP	caretup	caretup
CARETDOWN	caretdown	caretdown

“None”	nothing	无
None	nothing	无
” “	nothing	无
'''	nothing	无
'\$...\$'	render the string using mathtext.	使用 <b>mathtext</b> 渲染的字符串。
verts	a list of (x, y) pairs used for Path vertices.	用于路径顶点 (X, Y) 对的列表。
path	a Path instance.	一个路径实例。
(numsides, style, angle)	see below	

## 附录七

`datetime` 模块的所有类的对象都是不可变的。

### **`datetime.date` 类**

表示日期，由年、月、日组成。

`date` 类的实例具有 3 个只读属性：`year`、`month`、`day`

`date(year, month, day)`构造方法

`date.today()`返回一个表示今天的日期的 `date` 对象。

`d.timetuple()`返回一个对应于日期 `d` 和时间 00:00:00(午夜)的时间元组。

`d.weekday()`以整数形式返回日期 `d` 是一个星期的第几天，0 表示星期一，6 表示星期天。

`d.replace(year=None, month=None, day=None)`返回一个新的 `date` 对象。

`d.strftime(fmt)`格式化日期对象，返回字符串。

### **`datetime.time` 类**

`time` 类的实例表示一天中的时间。

`time(hour=0, minute=0, second=0, microsecond=0, tzinfo=None)`构造方法。

`t.replace(hour=None, minute=None, second=None, microsecond=None[, tzinfo])`返回一个新的 `time` 对象。

`t.strftime(fmt)`格式化时间对象，返回字符串。

### **`datetime.datetime` 类**

`datetime` 类的实例表示一个时刻（日期和这个日期的特定时间）。`datetime` 类是 `date` 类的子类，并添加了 `time` 的属性。

`datetime(year, month, day, hour=0, minute=0, second=0, microsecond=0, tzinfo=None)`构造方法。

`datetime.now(tz=None)`返回一个表示当前本地日期和时间的 `datetime` 对象。

`datetime.today()`返回一个表示当前本地日期和时间的 `datetime` 对象。

`d.date()`返回一个表示与 `d` 对象的日期相同的 `date` 对象。

`d.time()`返回一个表示与 `d` 对象的时间相同的 `time` 对象。

`d.replace(year=None, month=None, day=None, hour=None, minute=None, second=None, microsecond=None[, tzinfo])`返回一个新的 `datetime` 对象。

`d.weekday()`

`d.strftime(fmt)`格式化，返回字符串。

`datetime.strptime(str, fmt)`将字符串 `str` 按照格式 `fmt`, 返回 `datetime` 对象。

`d.timetuple()`返回一个对应时刻 `d` 的时间元组。

`datetime.combine(date, time)`根据 `date` 对象和 `time` 对象, 创建 `datetime` 对象。

## **datetime.timedelta 类**

表示时间间隔, 使用 3 个只读属性: `days`、`seconds` 和 `microseconds`

`timedelta(days=0, seconds=0, microseconds=0, milliseconds=0, minutes=0, hours=0, weeks=0)`构造方法

从一个 `timedelta` 对象计算对应的时间间隔, 单位为秒: `t.days*86400 + d.seconds + milliseconds/1000000.0`

**date 对象和 datetime 对象可与 timedelta 对象进行加减运算:**

**`date2 = date1 +timedelta #日期对象加上时间间隔, 返回新日期对象`**

`date` 对象之间、`datetime` 对象之间、`time` 对象可进行比较 (`>`~`<`)。

格式字符 意义

`%a` 星期的简写。如 星期三为 `Web`

`%A` 星期的全写。如 星期三为 `Wednesday`

`%b` 月份的简写。如 4 月份为 `Apr`

`%B` 月份的全写。如 4 月份为 `April`

`%c`: 日期时间的字符串表示。(如: `04/07/10 10:43:39`)

`%d`: 日在这个月中的天数 (是这个月的第几天)

`%f`: 微秒 (范围[0,999999])

`%H`: 小时 (24 小时制, [0, 23])

`%I`: 小时 (12 小时制, [0, 11])

`%j`: 日在年中的天数 [001,366] (是当年的第几天)

`%m`: 月份 ([01,12])

`%M`: 分钟 ([00,59])

`%p`: AM 或者 PM

`%S`: 秒 (范围为[00,61], 为什么不是[00, 59], 参考 python 手册~\_~)

`%U`: 周在当年的周数 (当年的第几周), 星期天作为周的第一天

`%w`: 今天在这周的天数, 范围为[0, 6], 6 表示星期天

`%W`: 周在当年的周数 (是当年的第几周), 星期一作为周的第一天

`%x`: 日期字符串 (如: `04/07/10`)

`%X`: 时间字符串 (如: `10:43:39`)

`%y`: 2 个数字表示的年份

**%Y:** 4 个数字表示的年份

**%z:** 与 **utc** 时间的间隔 （如果是本地时间，返回空字符串）

**%Z:** 时区名称 （如果是本地时间，返回空字符串）

**%%:** %% => %