

Graph with Matplotlib

by Jibo He

清华大学社会科学学院心理学系

hejibolaboratory@pku.org.cn

version:2022-05-22

画图重点与技巧

- i. 显示中文
- ii. 标题或者y-label过长，换行显示
- iii. 如何以期刊要求的300DPI保存图片
- iv. 如何改变线条颜色与风格
- v. 如何改变字体大小

为什么不用Excel画图？

- 确保图片是300DPI的清晰度
- 为了重用代码，不重复点击操作
- 为了确保图片的长宽比一致（aspect ratio）或者高度一致

期刊对图片的要求

以顶刊Accident Analysis and Prevention为例

source: <https://www.elsevier.com/journals/accident-analysis-and-prevention/0001-4575/guide-for-authors>

TIFF (or JPG): Color or grayscale photographs (halftones): always use a minimum of 300 dpi.

TIFF (or JPG): Bitmapped line drawings: use a minimum of 1000 dpi.

TIFF (or JPG): Combinations bitmapped line/half-tone (color or grayscale): a minimum of 500 dpi is required.

Please do not:

- Supply files that are optimized for screen use (e.g., GIF, BMP, PICT, WPG); the resolution is too low.

package installation

pip install chinesize-matplotlib

pip install matplotlib

pip install numpy

Introduction to Matplotlib

“make easy things easy and hard things possible”

- create simple plots with just a few commands
- “emulate” MATLABs plotting capabilities

matplotlib is conceptually divided into three parts

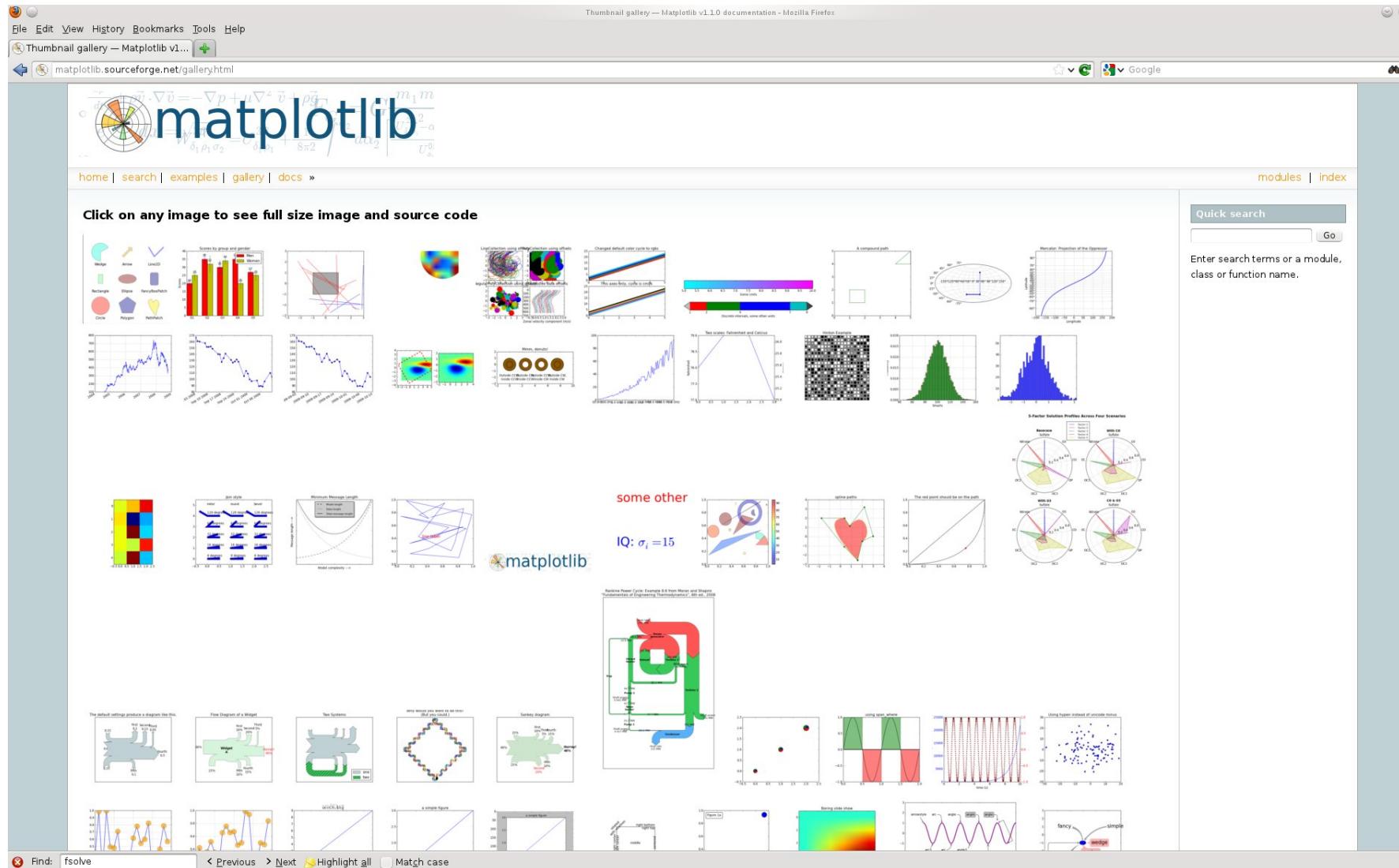
- Pylab interface : MATLAB like plotting
- Matplotlib API : abstract interface
- Backends : managing the output

available at (including many examples)

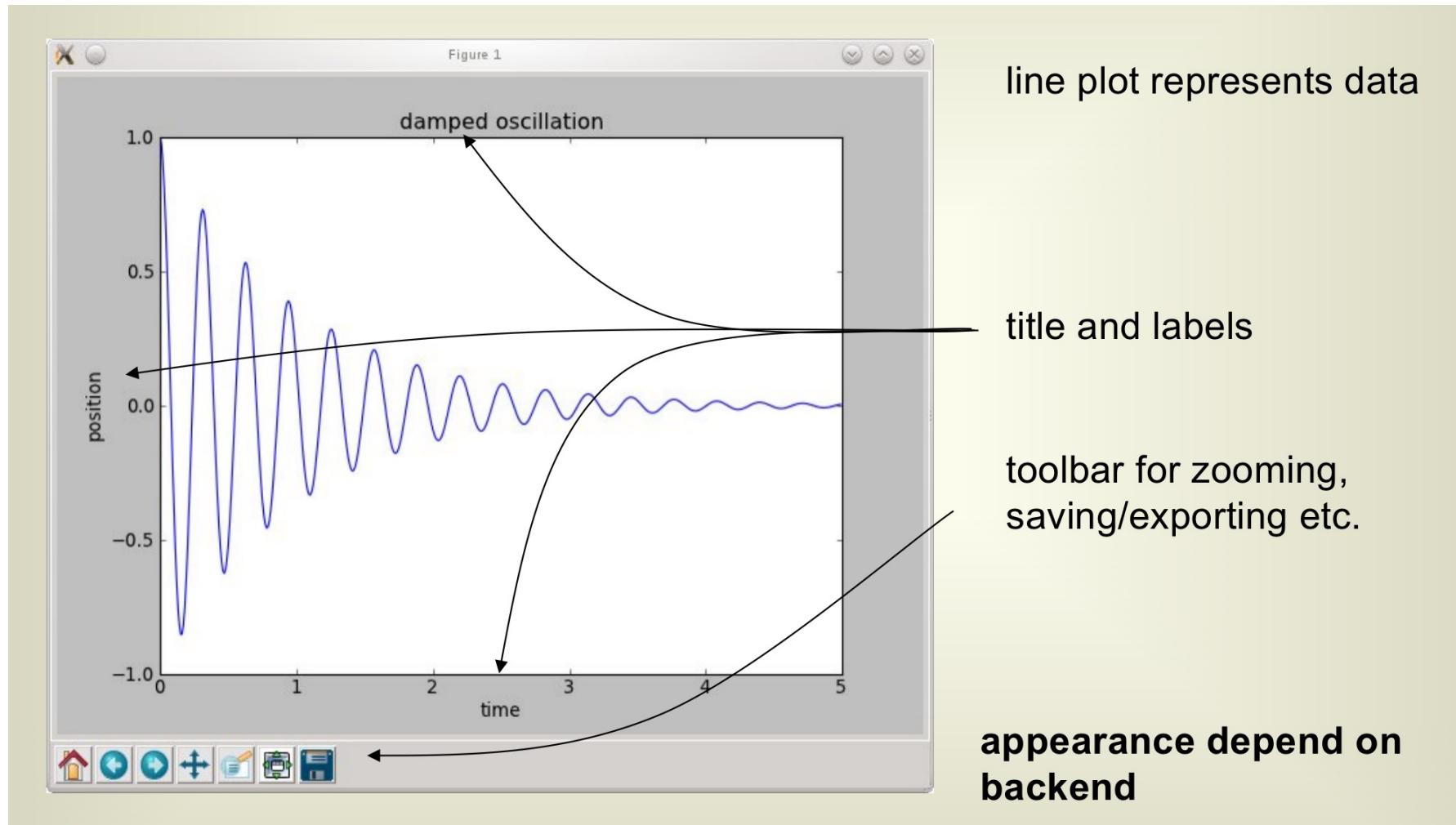
<http://matplotlib.sourceforge.net/>

The Matplotlib Gallery

<http://matplotlib.sourceforge.net/gallery.html>



Basic 2D - plotting



Basic 2D - plotting

```
import numpy as np          # import numpy
import pylab as pl           # import pylab interface

times = np.arange ( 0, 5, 0.01 )      # define x-vector
fun  = lambda x : np.cos (20 *x) * np.exp (- pl.absolute(x) )
                                # define some function fun (x)

pl.plot ( times, fun(times) )    # plot fun (t) vs. t
pl.xlabel ('time' )              # creating x-label
pl.ylabel ('position')          # creating y-label

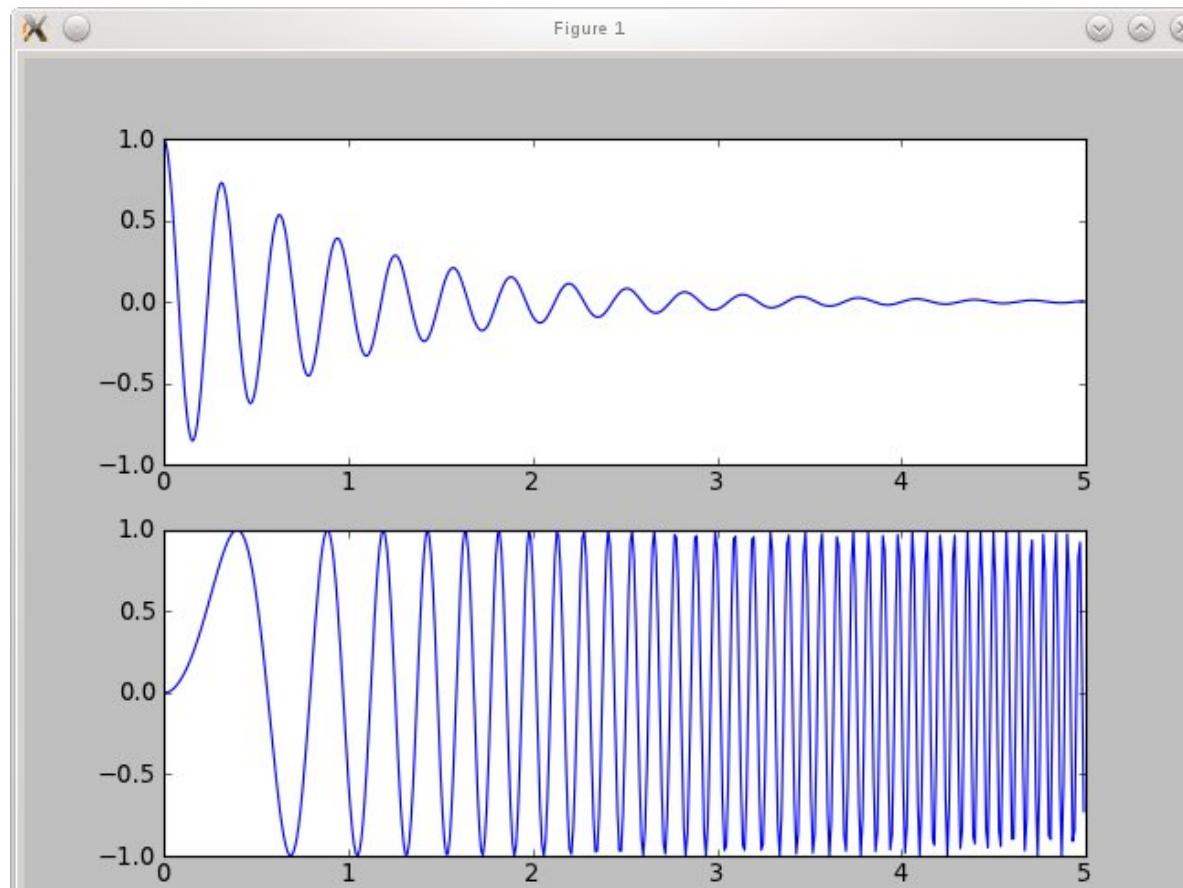
pl.title ( 'damped oscillation' ) # setting the title
pl.show()                         # show the plot
```

subplot (2,1,1) :

2 columns, 1 row

choose first subplot

! Indexing starts with 1



subplots

```
import numpy as np          # import numpy
import pylab as pl           # import pylab interface

times = np.arange ( 0, 5, 0.01 )      # define x-vector

fun   = lambda x : np.cos (20 *x) * np.exp (- pl.absolute(x) )
fun2  = lambda x : np.sin (10 *x**2)    # define two functions

pl.subplot (2,1,1)                # choose a subplot ( rows, columns, idx)
pl.plot ( times, fun(times) )     # plot fun(t)

pl.subplot (2,1,2)                # choose a subplot ( rows, columns, idx)
pl.plot ( times, fun2(times) )    # plot fun2(t)

pl.show()
```

Subplots Exercise

How to make the figures side by side?

Other basic plotting commands

pl.bar () # box plot

pl.errorbar() # plot with errorbars

pl.loglog() # logarithmically scaled axis

pl.semilogx () # x-axis logarithmically scaled

pl.semilogy () # y-axis logarithmically scaled

Histograms

```
import numpy as np          # import numpy
import pylab as pl           # import pylab interface

data = 3. + 3. * np.random.randn(100000)
    # generate normally distributed randomnumbers

pl.subplot(2,1,1)
pl.hist(data, 100)          # make histogram with 100 bins

pl.subplot(2,1,2)
pl.hist(data, bins = np.arange(3, 25, 0.1))
    # make histogram with given bins

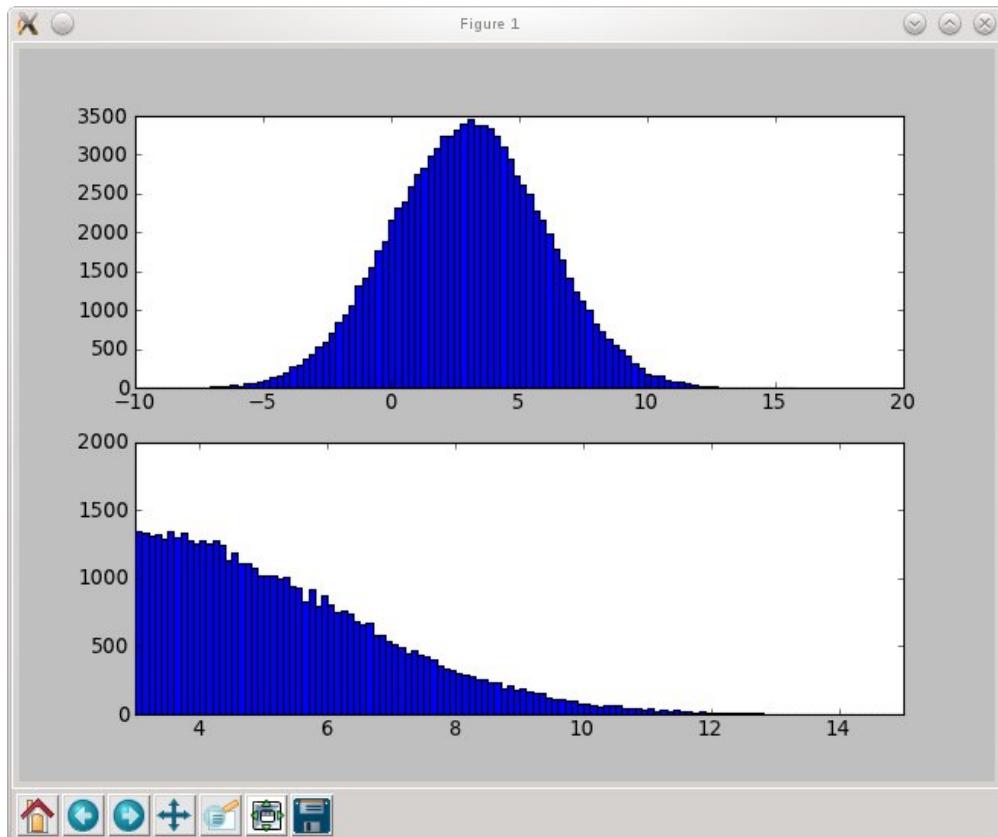
pl.axis((3, 15, 0, 2000))   # specify axis (x1,x2,y1,y2)

pl.show()
```

Histograms

(automatic) histogram with 100 bins

histogram for data between 3. and 25. with binsize 0.1
axis set to (3,15,0,2000)



画图技巧 1. 显示中文

chineseize-matplotlib 自动设置 matplotlib 中文字体

```
import matplotlib.pyplot as plt
import chineseize_matplotlib

plt.plot([1, 2, 3, 4])
plt.xlabel('简单图表')
plt.show()
```

<https://pypi.org/project/chineseize-matplotlib/>

画图技巧 2.通过换行显示一个很长的title或者label

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4])
#plt.xlabel('I am a loooooooooooooooong title')
plt.xlabel('I am a looooooooooooo\nooooooooooooong title')

plt.show()
```

画图技巧 3.如何以期刊要求的300DPI保存图片

```
import numpy as np          # import numpy
import pylab as pl           # import pylab interface
fig = pl.figure()
times = np.arange ( 0, 5, 0.01 )      # define x-vector
fun  = lambda x : np.cos (20 *x) * np.exp (- pl.absolute(x) )
                           # define some function fun (x)
pl.plot ( times, fun(times) )    # plot fun (t) vs. t
pl.xlabel ('time' )              # creating x-label
pl.ylabel ('position')          # creating y-label
pl.title ( 'damped oscillation') # setting the title

fig.set_size_inches(30.,18.)
## ensuare the figure is saved in 300 dpi, high resolution enough for publication purposes.
pl.savefig('/Users/ucdlab/Desktop/Course /Python - 大数据 /清华Python 2022/Graph-matplotlib/300dpi.png', dpi=300)
pl.show()
```

画图技巧 4. 如何改变线条颜色与风格

支持的线条风格

'-'

'--'

'-. '

'::'

'None'

' '

' ''

'solid'

'dashed'

'dashdot'

'dotted'

画图技巧 4. 如何改变线条颜色与风格

```
import matplotlib.pyplot as plt

plt.plot([1, 2, 3, 4],color = 'black',linestyle="dashed")

plt.plot([ 3, 4,1, 2],color = 'black',linestyle="dotted")

#supported values are '-','--','-.',':', 'None', ' ', '','solid', 'dashed', 'dashdot', 'dotted'

plt.ylabel('I am a ylabel')
plt.xlabel('I am a looooooooooooo\nooooooooooooong title')

plt.show()
```

画图技巧 5. 如何改变字体大小

```
import matplotlib.pyplot as plt

LegiableFontSize = 20
plt.rc('font', size=LegiableFontSize) #controls default text size
plt.rc('axes', titlesize=LegiableFontSize) #fontsize of the title
plt.rc('axes', labelsize=LegiableFontSize) #fontsize of the x and y labels
plt.rc('xtick', labelsize=LegiableFontSize) #fontsize of the x tick labels
plt.rc('ytick', labelsize=LegiableFontSize) #fontsize of the y tick labels
plt.rc('legend', fontsize=LegiableFontSize) #fontsize of the legend

plt.plot([1, 2, 3, 4])
plt.ylabel('I am a looooooooooooo\nooooooooooooong title')
plt.xlabel('I am a looooooooooooo\nooooooooooooong title')

plt.show()
```



数据可视化—Matplotlib模块快速查表Ⅱ

图表绘制专属函数	说 明	图表绘制专属函数	说 明
acorr()	绘制x的自相关图	phase_spectrum()	绘制相位谱
angle_spectrum()	绘制角度谱	pie()	绘制饼形图
bar()	绘制柱形图	plot()	绘制折线图
barh()	绘制水平条形图	plot_date()	绘制时间序列图
boxplot()	绘制箱形图	polar()	绘制极坐标图
broken_barh()	绘制水平断条图	psd()	绘制功率谱密度图
contourf/contour()	绘制等高线图(填充/不填充)	scatter()	绘制散点图
csd()	绘制交叉谱密度图	specgram()	绘制声谱图
fill()	绘制填充多边形	stackplot()	绘制堆叠面积图
hexbin()	绘制二维六角形多色柱状图	step()	绘制阶梯图
hist()	绘制直方图	subplot()	绘制子图表(比较灵活)
magnitude_spectrum()	绘制震级谱	subplots()	绘制子图表
pcolor()	绘制二维阵列的伪彩色图	table()	绘制表格
pcolormesh	绘制四边形网格	violinplot()	绘制小提琴图

图表设置函数	说 明	图表设置函数	说 明
annotate()	注释	imread()	读取图像
arrow()	在坐标轴上添加一个箭头	imsave()	保存图像
axex()	添加一个轴	imshow()	显示图像
axis()	获取或设置axis属性的方法	legend()	图例
box()	打开/关闭“坐标轴”框	savefig()	保存当前画布(图表)
cla()	清除当前轴	show()	显示所有画布(图表)
clabel()	等高线图标签	subplots_adjust()	图表与画布边缘间距
clf()	清除当前画布	subtitle()	子图标题
colorbar()	在绘图中添加一个颜色条	text()	文本标签
draw()	重新绘制当前图形	title()	图表标题
errorbar()	绘制误差线	twinx() / twiny()	共享x轴/y轴
figtext()	向图中添加文本	xcorr()	绘制相关性
figure()	创建新的画布	xlabel() / ylabel()	x轴标题/y轴标题
grid()	为图表设置网格线	xlim() / ylim()	x轴/y轴的坐标范围
hlines()	绘制水平线(横线)	xscale() /yscale()	x轴/y轴刻度



数据可视化—Matplotlib模块速查表Ⅰ

常用颜色	blue	green	red	cyan	magenta	yellow	black	white
	b	g	r	c	m	y	k	w

通过十六进制字符串或者颜色名称设置颜色：

浮点形式的RGB或RGBA元组。例如：[(0.5,0.2,0.7),(0.3,0.1,0.9),(0.4,0.1,0.6),(0.6,0.1,0.8,0.3)]

16进制的RGB或RGBA字符串。例如：['#00FF7F','#3CB371','#E8B571','#0FFF00']

0~1的小数作为的灰度值。例如：[0.88,0.65,0.47,0.24]

X11/CSS4规定中的颜色名称。例如：{'xkcd:sky blue','xkcd:flat blue','xkcd:baby blue','xkcd:mid blue'}

Tableau T10调色板颜色。例如：{'tab:blue','tab:orange','tab:green','tab:purple'}

CN格式颜色循环。例如：['#9467bd','#8c564b','#e377c2','#7f7f7f']

线型	说明	标记符	说明	标记符	说明	标记符	说明
- —	实线	· ●	点	1 ♪	下花三角	h ♡	竖六边形
— — —	双划线	, ■	像素	2 ▲	上花三角	H ◆	横六边形
- · - -	点划线	o ●	实心圆	3 ↖	左花三角	+ +	加号
: - - -	虚线	v ▼	倒三角	4 ↘	右花三角	x X	叉号
		^ ▲	上三角	8 ●	八边形	X ✗	叉号(满)
		> ►	右三角	s ■	实心正方	D ♦	大菱形
		< ▲	左三角	p ♦	实心五角星	d ♦	小菱形
		* ★	星形	P +	加号(满)		垂直线
						- —	水平线

图例位置字符串	位置代码	说明	图例位置字符串	位置代码	说明
best	0	自适应	center left	6	左侧中间位置
upper right	1	右上方	center right	7	右侧中间位置
upper left	2	左上方	lower center	8	上方中间位置
lower left	3	左下方	upper center	9	下方中间位置
lower right	4	右下方	center	10	正中央
right	5	右侧	自定义图例位置: bbox_to_anchor=(0.5, 0., 0.5, 0.5)		

图表与画布边缘间距：

位置	建议值	说明	位置	建议值	说明
top	0.9	距画布顶部的距离, 值越大, 空白越少	left	0.125	距画布左边的距离, 值越小, 空白越少
bottom	0.1	距画布底部的距离, 值越小, 空白越少	right	0.9	距画布右边的距离, 值越大, 空白越少
wspace	0.2	预留空白的宽度	hspace	0.2	预留空白的高度

示例代码：

```
plt.subplots_adjust(left=0.2, right=0.9, top=0.9, bottom=0.2)
```

重塑/排序/置换/数据加入与合并	
DataFrame	append() 增加数据
Pandas	pivot() 重塑数据(数据透视表)
	sort() 数据排序
	stack() 列索引转换成最内层的行索引
	unstack() 最内层的行索引转换成列索引
	T属性 行列数据转换
	crosstab() 交叉表是用于统计分组频率的特殊透视表
	melt() 将数据从宽表转换为长表
	merge()/concat() 按列名相同的列合并DataFrame数据/根据不同方式合并DataFrame数据
	pivot_table() 数据透视表

日期处理与时间序列	
Pandas	Series.dt() 获取日期中的年、月、日、星期数、季节等
	date_range() 生成时间序列
	period_range() 根据指定频率创建时期范围
	to_datetime() 将数据转换为日期时间格式
	to_timedelta() 计算两个日期数据之间的时间差
	asfreq() 将指定的日期数据转换为一定频率的数据
	resample() 数据重采样
	to_period() 将时间戳转换为时期
	to_timestamp() 将时期转换为时间戳

数据分析常用方法速查表

GroupBy分组与函数应用	
DataFrame	apply() 应用函数
Pandas	applymap() 将函数应用于要操作的DataFrame
DataFrame	groupby() 数据分组

数据统计			
Pandas	cut() 数据面元化(即将数据按照一定的区间进行分割)	mean()	平均值
	qcut 把一组数字按大小区间进行分割	median()	中位数
	get_dummies() 实现one-hot编码(例如, 性别男、女转换为0、1)	min()/max()	最小值/最大值
	corr() 相关系数	mode()	众数
	corrwith() 行或列两两相关	pct_change()	百分比
	count() 计算非空值或空值数量	quantile()	分位数
	cov() 协方差	rank()	数据排名
	describe() 描述性统计	skew()	峰度
	diff() 数据移位(一阶差分操作)	sum()	求和
	kurt() 偏度	std()	标准差
	mad() 平均绝对偏差	var()	方差

DataFrame绘制图表	
Data	boxplot() 绘制箱线图
	hist() 绘制直方图



数据处理常用方法速查表

数据输入/输出

Pandas	read_csv()	将CSV(逗号分隔)文件读入DataFrame对象
	read_excel()	将一个Excel表读入DataFrame对象
	read_html()	将HTML表读入DataFrame对象
	read_json()	将JSON字符串转换为DataFrame对象
	read_sql_table()	将SQL数据库中的表读入DataFrame对象
	read_sql_query()	将SQL查询读入DataFrame对象
	read_sql()	将SQL查询或数据库中的表读入DataFrame对象
	read_stata()	将Stata(统计学软件)文件读入DataFrame对象
	StataWriter.write_file()	写入Stata文件
DataFrame	to_csv()	输出数据为CSV文件
	to_excel()	将DataFrame数据写入Excel工作表
	to_html()	将DataFrame数据输出为HTML网页格式
	to_json()	将DataFrame数据转换为JSON字符串
	to_sql()	将DataFrame数据写入SQL数据库
	to_stata()	将DataFrame数据输出为Stata(统计学软件)文件

索引与迭代

DataFrame	at属性	以行名和列名获取单个数据。例如: a行a列, df.at['a', 'a']
	iat属性	以行列索引获取单个数据。例如: 0行0列, df.iat[0, 0]
	iloc属性	以行列索引获取多个数据。例如: df.iloc[0:4]
	loc属性	以行名和列名获取多个数据。例如: df.loc['明日':'二月二']
	insert()	在指定位置插入一列数据
	iteritems()	返回列名和序列的迭代器
	iterrows()	返回索引和序列的迭代器
	reindex()	重新设置索引
	reset_index()	常用于数据清洗后, 对数据重新设置连续行索引
	set_index()	设置索引

数据筛选/数据清洗

DataFrame	dropna()	缺失值删除
	drop()	删除数据返回新的DataFrame对象
	drop_duplicates()	删除重复行的DataFrame数据
	duplicated()	布尔值, 判断每一行数据是否重复
	fillna()	缺失值填充
	filter()	通过指定条件筛选数据
	isin()	返回符合条件的DataFrame。例如: A列包含8, df[df['A'].isin([8])]
	rename()	重命名行或列的标题
	replace()	批量替换数据(替换全部或某一行)
Series	str.contains()	数据中是否包含指定的字符串
	str.join()	连接字符串
	str.lower()/str.upper()	转换为小写字母/转换大写字母
	str.lstrip()/str.rstrip()	去除字符串左边/右边的空格