计算机与信息工程学院实验报告

••••••••••••••••••••••••••••••••• 密 ••••••••••••••••••••••••••••••••• 封 ••••••••••••••••••••••••••••••••• 线 •••••••••••••••••••••••••••••••••

姓名：\_杨佳森\_学号：\_2112080106\_专业：\_数据科学与大数据技术年级：\_ 2021 \_

课程：数据科学与大数据技术 主讲教师：\_周黎明 辅导教师：\_无

实验时间： 2023 年\_3\_月\_15\_日\_上\_午\_10时至\_12\_时，实验地点\_\_计算机大楼201\_

实验题目：数据可视化基础

实验目的： 1、掌握 pyplot 常用的绘图参数 2、掌握常用图形的绘制方法 3、掌握 Seaborn 图表的用法

实验环境（硬件和软件） PC和Jupyter Notebook

实验内容：

（一）Matplotlib 数据可视化 1、设置 pyplot 的参数 2、绘制散点图、折线图、直方图、饼图、箱线图

（二）Seaborn 图表 1、Seaborn 图表基本设置 2、常用图表绘制

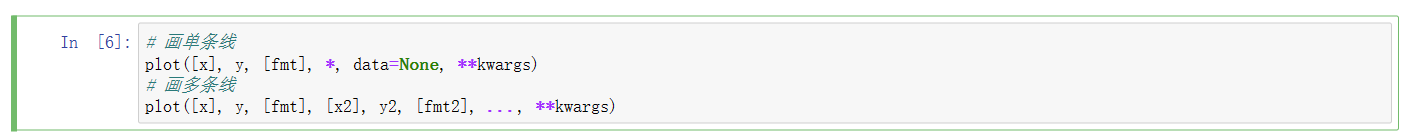
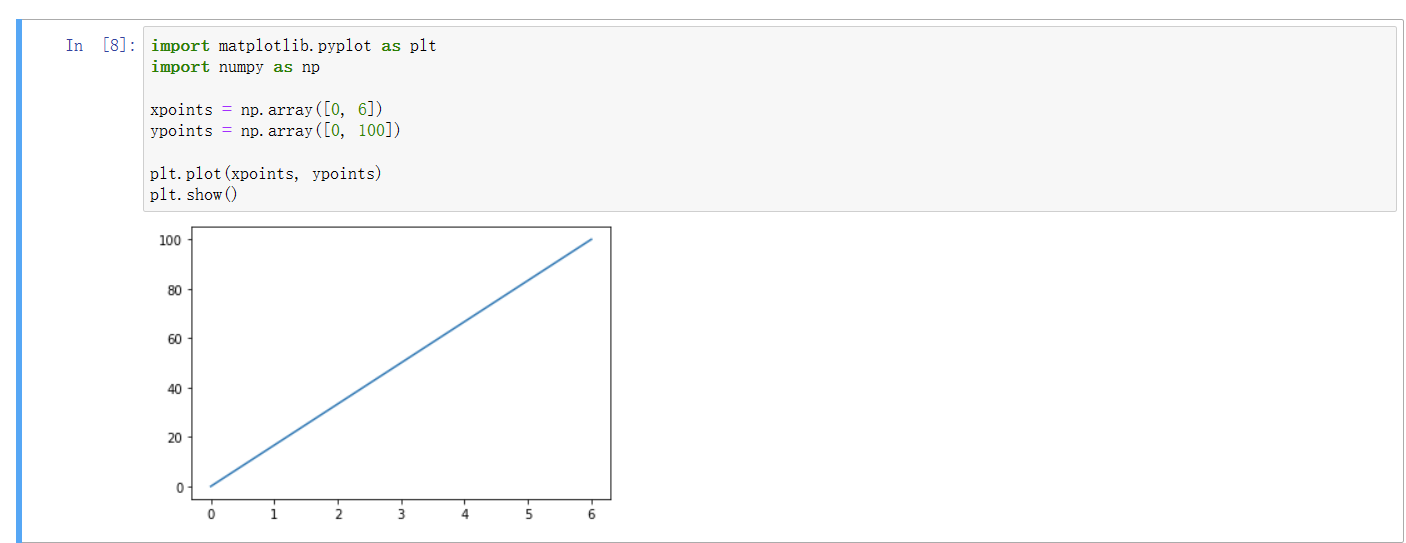
实验步骤：

1. Matplotlib数据可视化
2. 设置pyplot的参数
3. 绘制散点图、折线图、直方图、饼图、箱线图
4. Seaborn图表
5. Seaborn图表基本设置
6. 常用图表绘制

实验数据记录：

1. **Matplotlib数据可视化**
2. **设置pyplot的参数**

pyplot 包含一系列绘图函数的相关函数，每个函数会对当前的图像进行一些修改，例如：给图像加上标记，生新的图像，在图像中产生新的绘图区域等等。使用的时候，我们可以使用 import 导入 pyplot 库，并设置一个别名plt.



设置画布

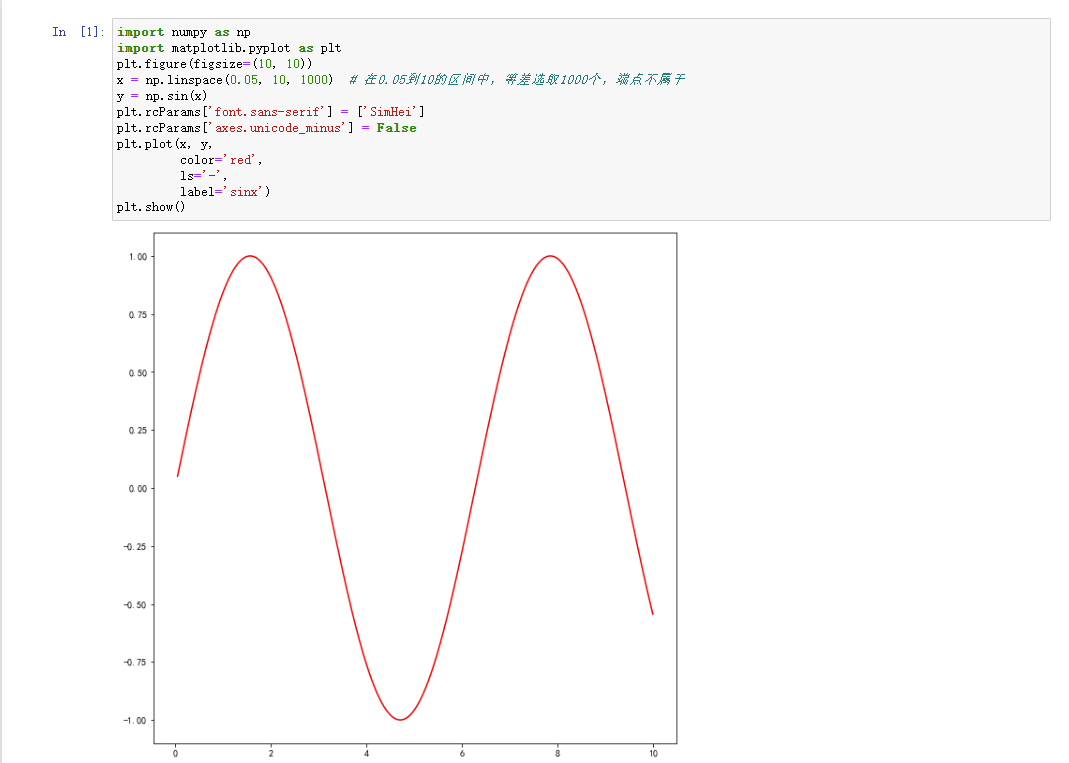
Matplotlib.pyplot.figure(num=none,figsize=none,dpi=none,facecolor=none,edgecolor=none,frameon=True)

num ：如果此参数没有提供，则一个新的 figure 对象将被创建，同时增加 figure 的计数数值，此数值被保存在 figure 对象的一个数字属性当中。如果有此参数，且存在对应 id 的 figure 对象，则激活对于 id 的 figure 对象。如果对应 id 的 figur 对象不存在，则创建它并返回它。如果 num 的值是字符串，则将窗口标题设置为此字符串

figsize：以英寸为单位的宽高，缺省值为 rc figure.figsize (1 英寸等于 2.54 厘米)

dpi：图形分辨率，缺省值为 rc figure.dpi

facecolor：背景色

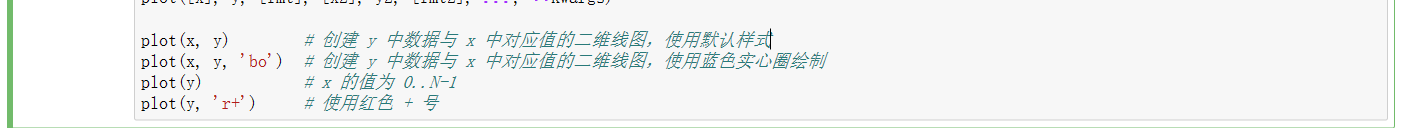


参数说明：

x, y：点或线的节点，x 为 x 轴数据，y 为 y 轴数据，数据可以列表或数组。

fmt：可选，定义基本格式（如颜色、标记和线条样式）。

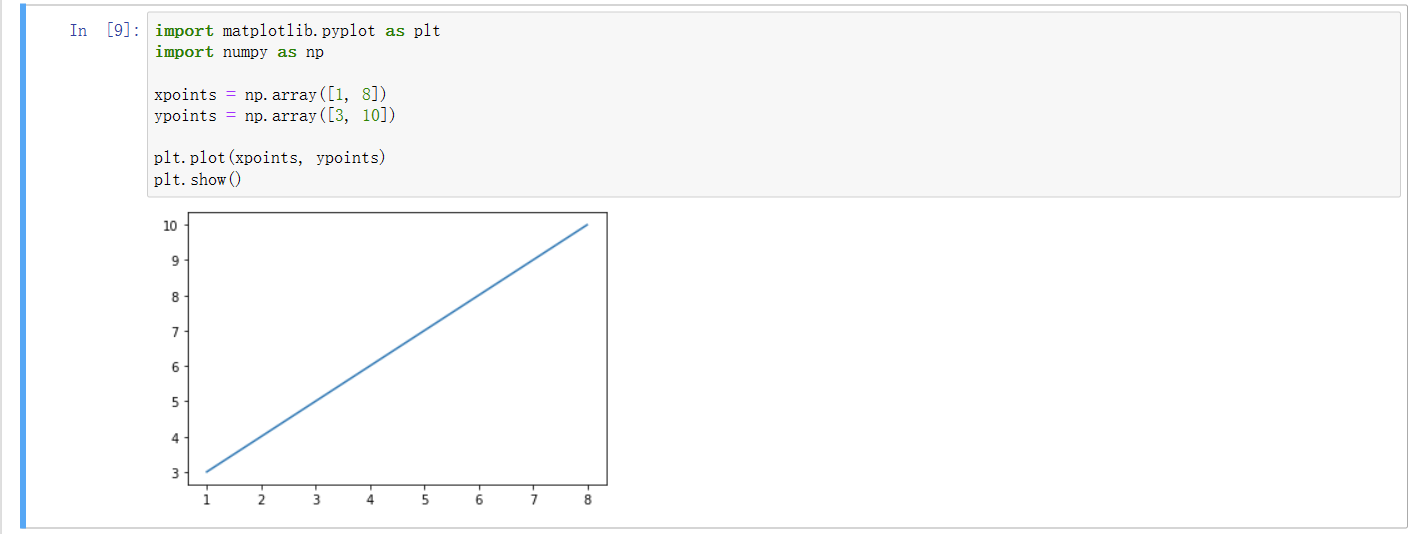
\*\*kwargs：可选，用在二维平面图上，设置指定属性，如标签，线的宽度等。



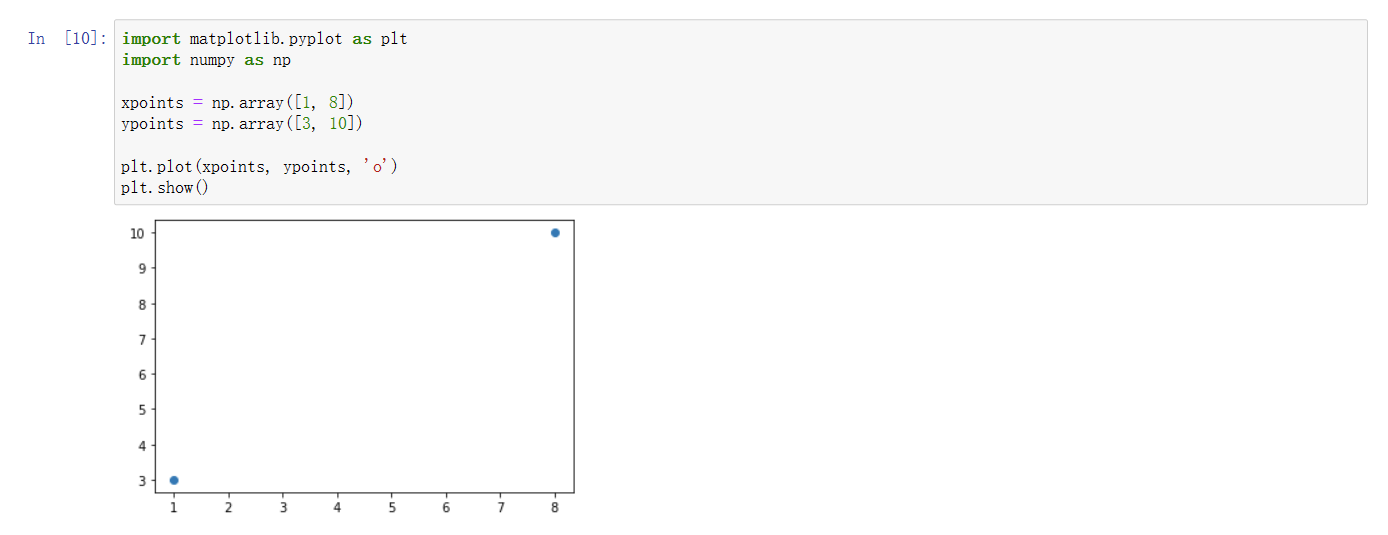
**颜色字符：**'b' 蓝色，'m' 洋红色，'g' 绿色，'y' 黄色，'r' 红色，'k' 黑色，'w' 白色，'c' 青绿色，'#008000' RGB 颜色符串。多条曲线不指定颜色时，会自动选择不同颜色。

**线型参数：**'‐' 实线，'‐‐' 破折线，'‐.' 点划线，':' 虚线。

**标记字符：**'.' 点标记，',' 像素标记(极小点)，'o' 实心圈标记，'v' 倒三角标记，'^' 上三角标记，'>' 右三角标记，'<' 左三角标记...等等。



如果我们只想绘制两个坐标点，而不是一条线，可以使用 o 参数，表示一个实心圈的标记:



也可以绘制任意数量的点，只需确保两个轴上的点数相同即可。

绘制一条不规则线，坐标为 (1, 3) 、 (2, 8) 、(6, 1) 、(8, 10)，对应的两个数组为：[1, 2, 6, 8] 与 [3, 8, 1, 10]。



Num:图像编号或名称

figsize:画布的宽和高

dpi:绘图对象的分辨率

facecolor：背景颜色

frameon:是否显示边框

设置坐标轴

X轴标题:xlabel()，Y轴标题:ylabel()

使用方法：plt.xlabel(fontsize, verticalalignment, horizontalalignment, rotation, bbox)

fontsize：数字或者（small，large，medium）

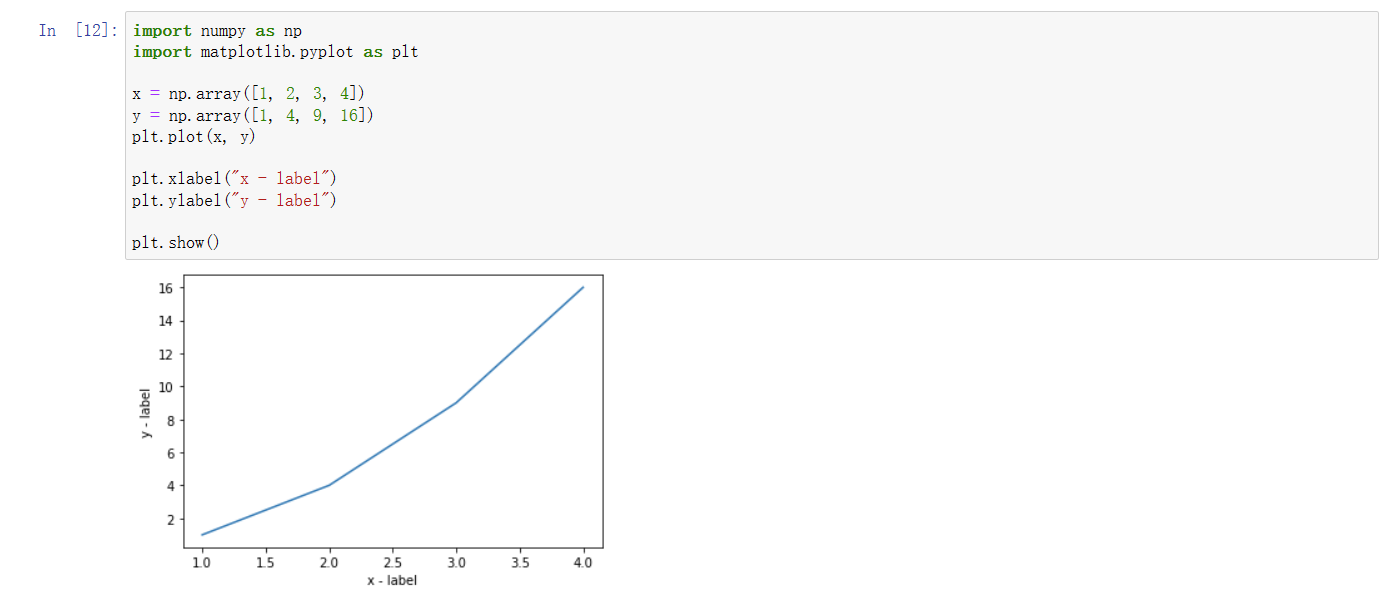
verticalalignment：距离坐标轴的位置（top，bottom，center，baseline）

hoizontalalignment：位置（center，right，left）

ratation：位置（vertical，horizontal，vertical）

bbox：添加边框

可以使用xlabel（）和ylabel（）方法来设置 x 轴和 y 轴的标签。

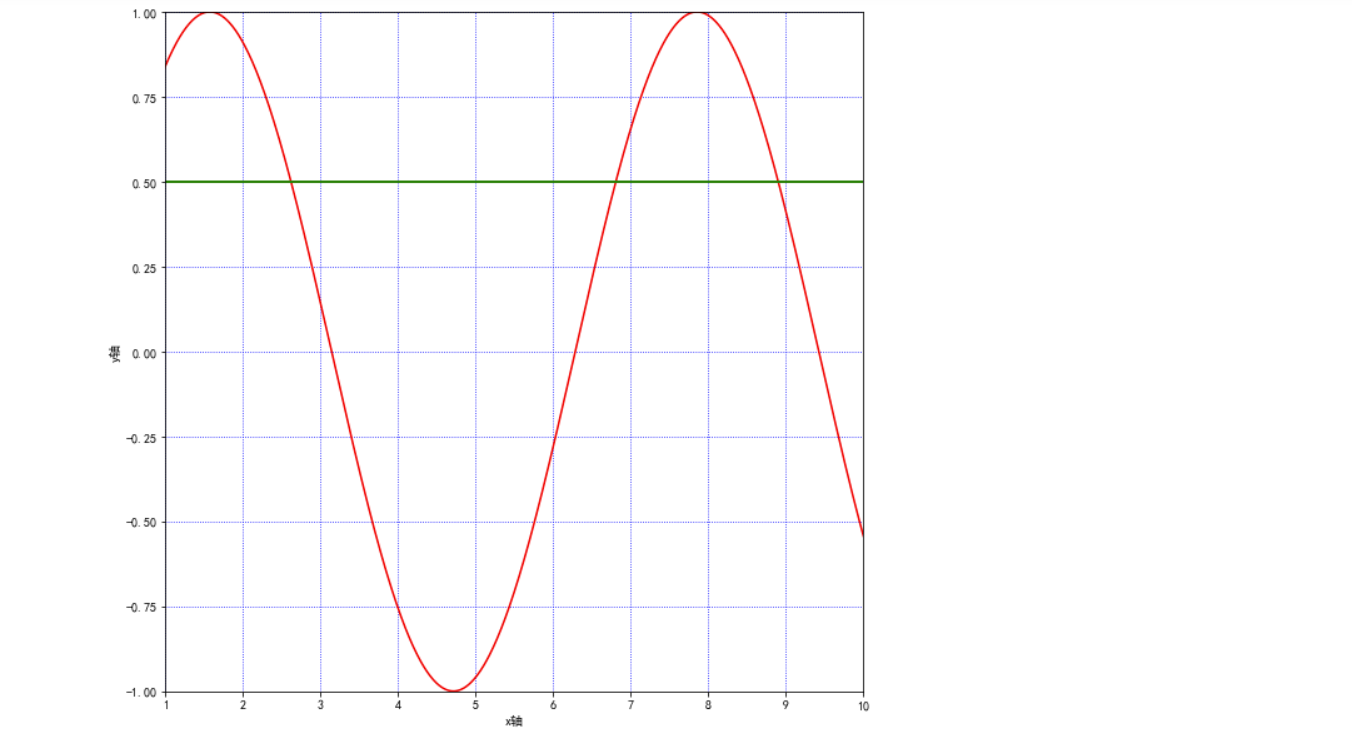


绘制刻度线的网格线**plt.grid(linestyle, color)**

绘制平行于x轴额度水平参考线**plt.axhline(y, c, ls, lw, label)**



xticks(),yticks()获取或设置当前 x 轴或 y 轴刻度位置和标签（即设置 x 或 y 轴的标 签）可以理解为设置xilim和ylim一样的效果，但可以指定范围和距离



tick\_params()函数

axis: 可选"x","y","both",默认"both"，分别代表，对x轴操作，对y轴操作，对两个轴都操作。 direction: 可选 "in","out","inout"代表，刻度线显示在坐标轴里面，坐标轴外边，双边

length: 刻度线长度，上面图里的刻度线长度，我设置的为6

color: 刻度线颜色，上面图里的刻度线颜色，我设置的为“r”

width： 刻度线宽度

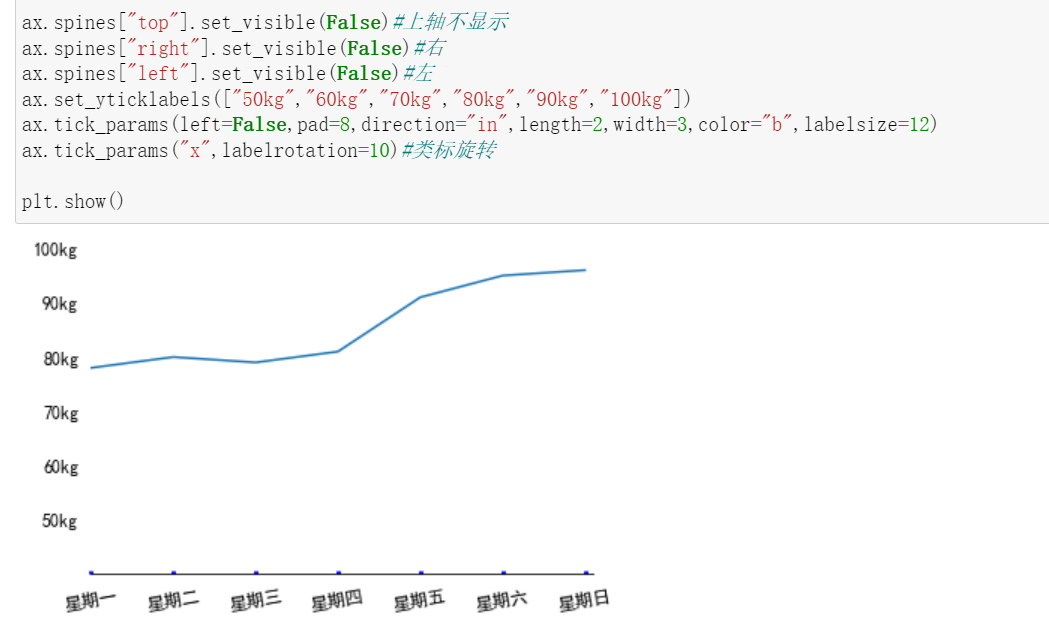
pad： 刻度线与刻度标签之间的间隔

bottom, top, left, right四个参数对应四个边框，它们的取值为布尔类型，True 表示显示对应边框上的刻度线，False，代表不显示，默认True

labelbottom, labeltop, labelleft, labelright,与上面四个对应，代表的是四个边框上的类标的设置，取值为布尔类型，True代表显示对应边框上的类标，False代表不显示。

labelsize：类标大小的设置参数，可取浮点型数值，也可去"medium","large","small"

labelrotation：旋转类标一定的角度，与在set\_xticklabels（）中的参数rotation作用相同



图例添加legend()

loc:可取"best",1或者"upper right",2或"upper left",3或"lower left",4或"lower right",代表放不同位置

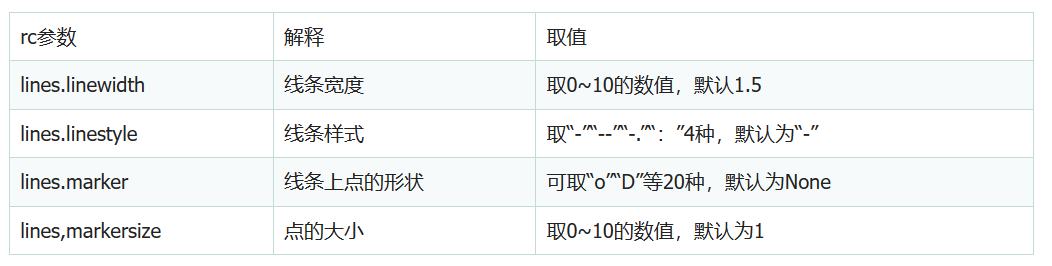
fontsize： int或float或{‘xx-small’, ‘x-small’, ‘small’, ‘medium’, ‘large’, ‘x-large’, ‘xx-large’}，字体大小

shadow: 是否为图例边框添加阴影

labelspacing: 图例中条目之间的距离

handlelength: 图例句柄的长度

函数title（）添加图形内容的标题：plt.title("正弦函数")



1. **绘制散点图、折线图、直方图、饼图、箱线图**

绘制散点图

matplotlib.pyplot.scatter(x,y,s=None,marker=None,cmap=None,norm=None,vmin=None,alpha=None,linewidths=None,verts=None,edgecolors=None,data=None)

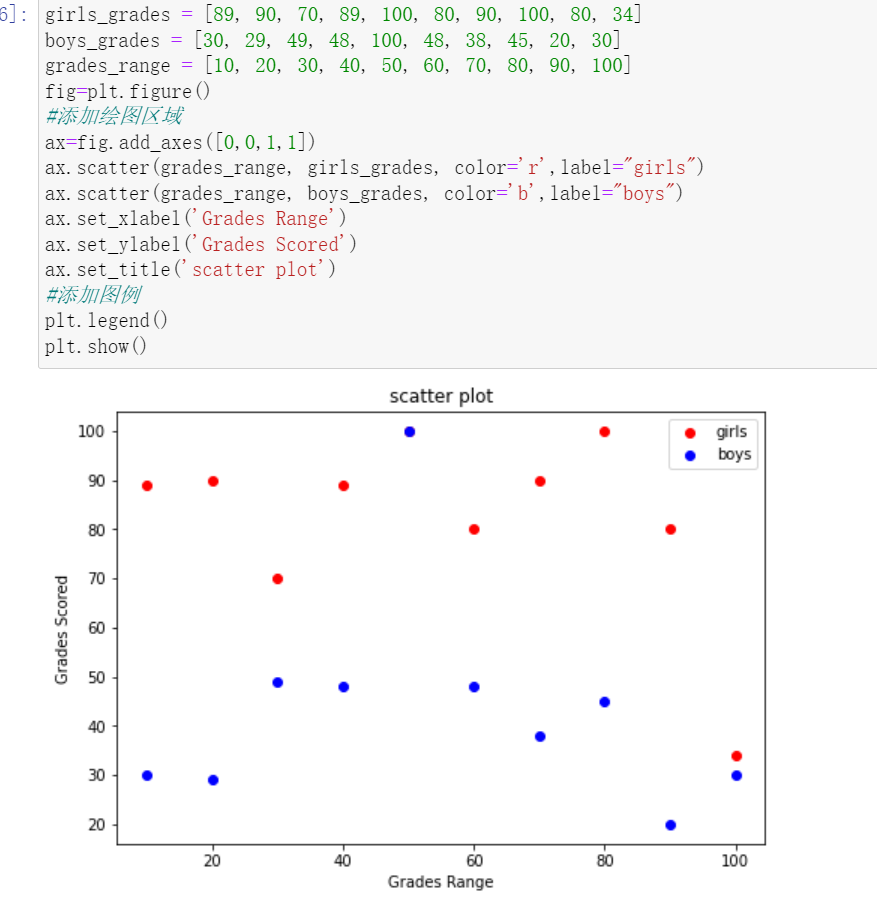
x,y接收array,表示x,y轴对应的数据；

s接收数值或一维array，指定点的大小，若传入一维数组，则表示每个点的大小，默认为None

c接收颜色或一维array，指定点的颜色，若传入一维数组，则表示每个点的颜色，默认为None

marker接受特定的string，表示绘制的点的类型，默认为None

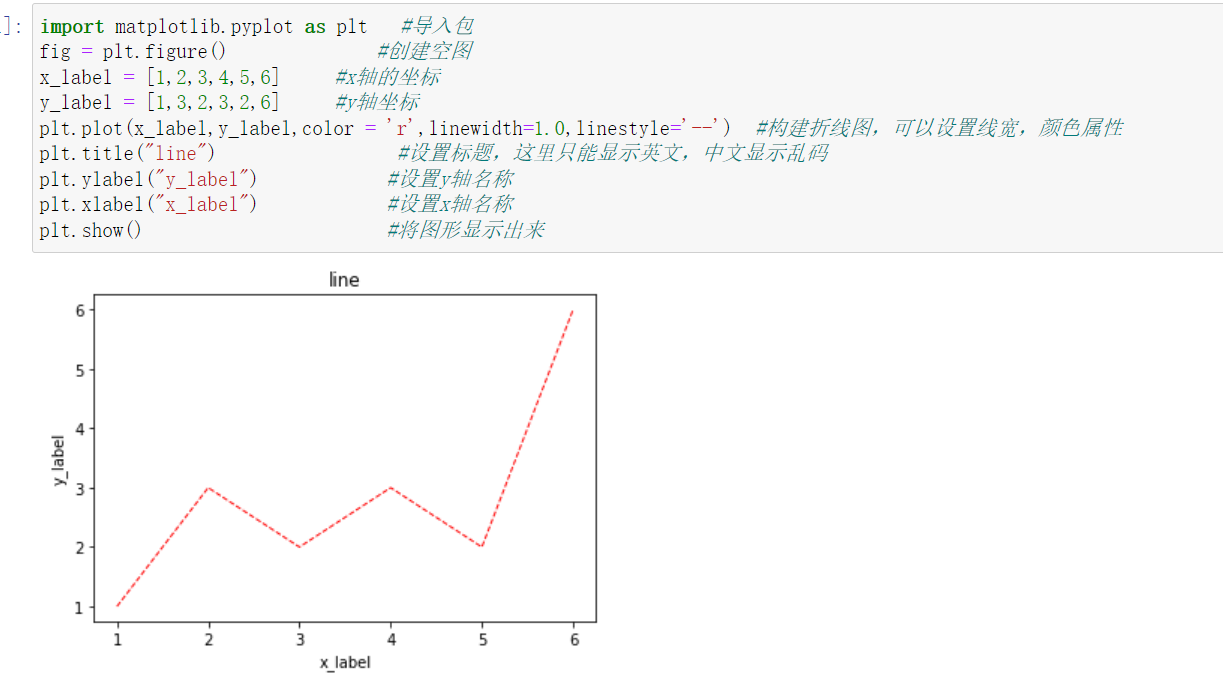
alpha接收0-1的小数，表示点的透明度，默认为None



绘制折线图

折线图：matplotlib.pyplot.**plot**(\*args,\*\*kwargs)

x,y接收array,表示x,y轴对应的数据；color 接收特定string，指定线条颜色，默认为None ，linestyle 接收特定string，指定线条类型，默认为marker 接收特定string，指定绘制的点的类型，默认为None，alpha 接受0-1间的小数，表点的透明度，默认None常见颜色：b-blue，g-green, r-red, c-青色，m-品红，y-yellow, k-black, w-white



绘制直方图

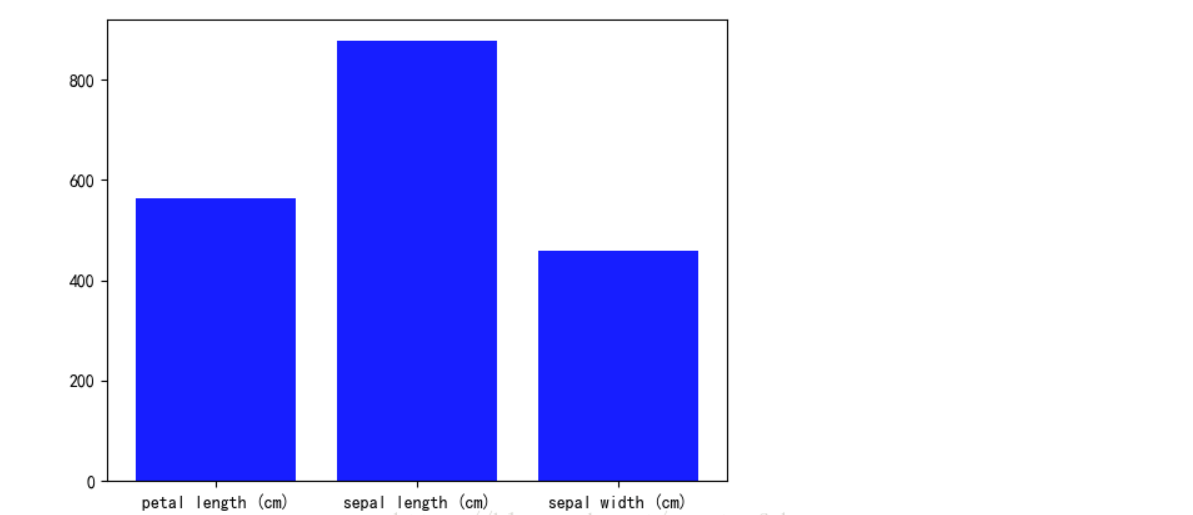
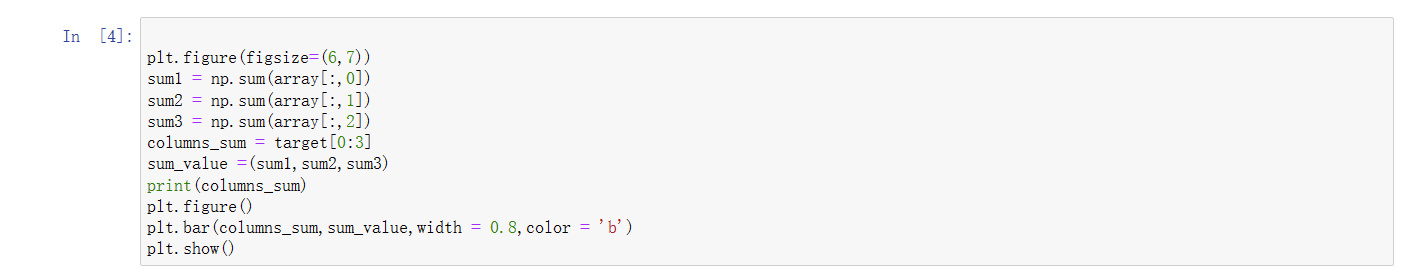
绘制直方图函数：pyplot中绘制直方图函数为bar(left,height,width=0.8,bottom=None,hold=None,data=None,\*\*kwargs)，

left 接收array，表示x轴数据，无默认，

height接收array，表示x轴所代表数据的数量，无默认，

width 接收0-1之间的float，指定直方图的宽度，默认0.8，

color 接收特定string，或者包含颜色字符串的array，表示直方图颜色，默认为None



绘制饼图

matplotlib.pyplot.pie(x,explode=None,colors=None,autopct=None,pctdistance=0.6shadow=False,labeldistance=1.1,startangle=None,radius=None,counterclock=True,wedgeprops=None,textprops=None,center=(0,0),frame=False,hole=None,data=None)

x 接收array，表示绘制饼图的数据，无默认； labels 接收array，指定每一项的名称，默认为None

explode 接收array，表示指定项距离饼图圆心为n个半径，默认为None

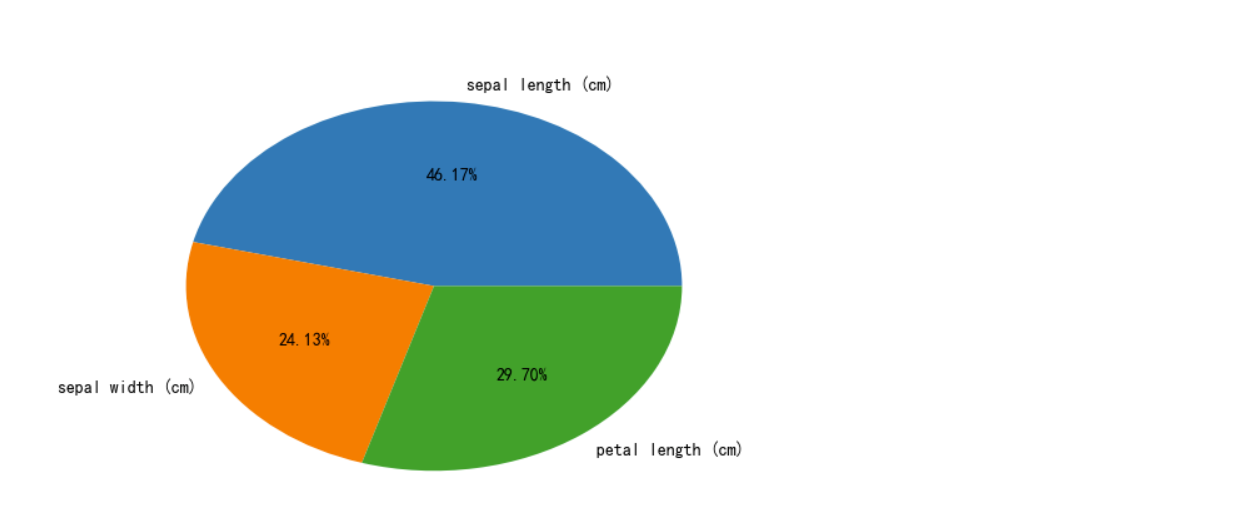
color 接收特定string，或者包含颜色字符串的array，表示饼图颜色，默认为None

autopct 接收特定string，指定数值的显示方式，默认为None

pctdistance 接收float，指定每一项的比例autopct和距离圆心的半径，默认为0.6

labeldistance 接收float，指定每一项的名称labels和距离圆心的半径，默认为1.1

radius 接收float，表示饼图的半径，默认为1



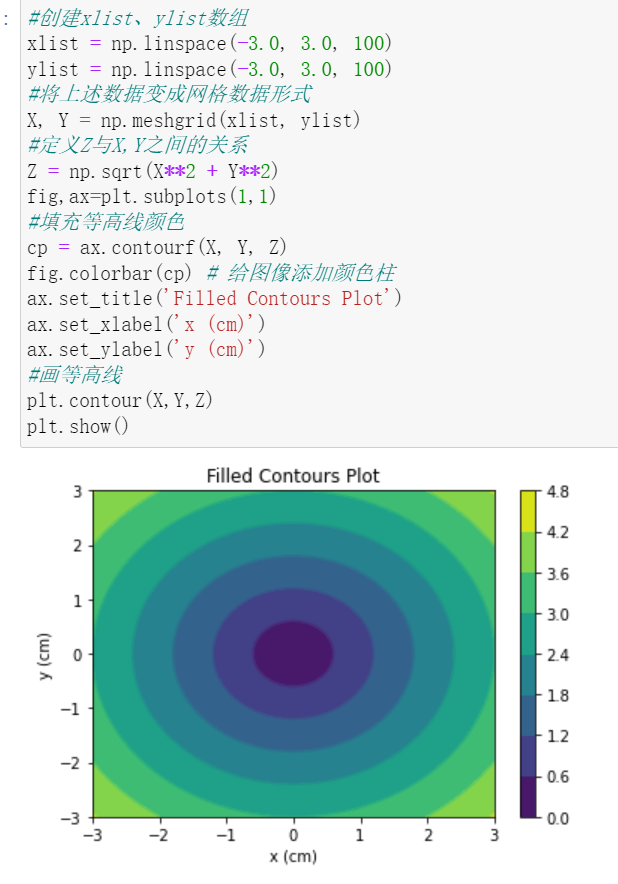
绘制箱线图

绘制箱线图函数：plt.boxplot(最大值，上四分位数，中位数，下四分位数，最小值)

plt.boxplot(x,notch=None,sym=None,vert=None,whis=None,positions=None)



绘制等高线图



绘制面积图

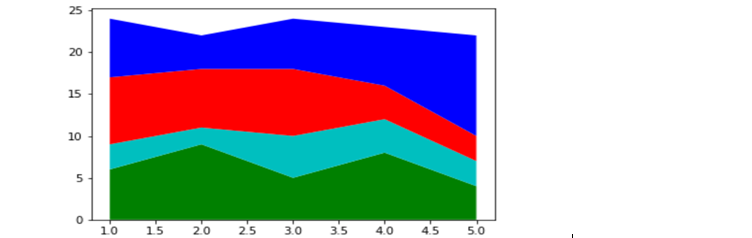
Matplotlib.pyplot.stackplot(x,\*args,data=Node,\*\*kwargs)

X:x轴数据

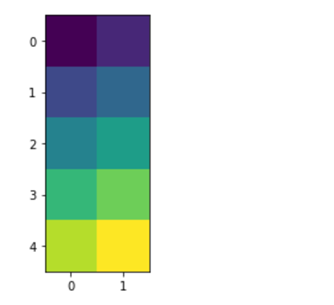
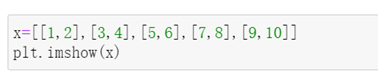
\*args:当传入的参数个数未知时使用\*args,这里指y轴数据可以传入y轴

Data:给定一个数据参数，所有位置和关键字参数将被替换

\*\*kwargs:其他可选参数例如，color,alpha



绘制热力图



1. **Seaborn图表**
   1. **Seaborn图表基本设置**

Seaborn通过set函数实现风格设置：seaborn.set(context='notebook',style='darkgrid',palette='deep',font='sans-serif',font\_scale=1,color\_codes=True,rc=None)

还有其他方法：

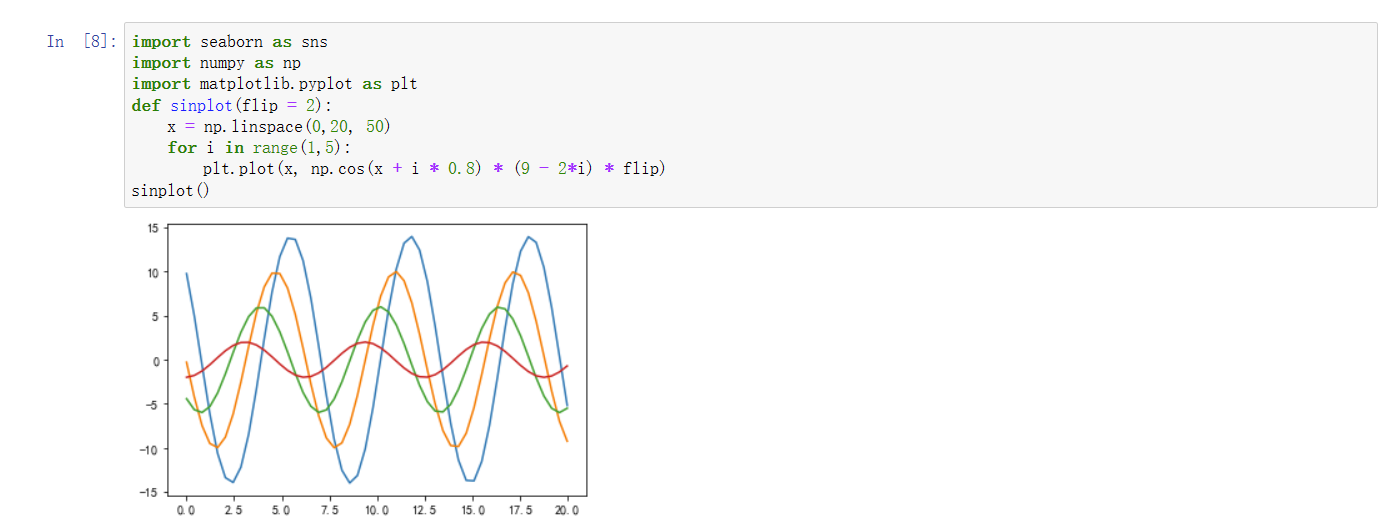
set\_style，风格专用设置接口，设置后全局风格随之改变

axes\_style，设置当前图（axes级）的风格，同时返回设置后的风格系列参数，支持with关键字用法

颜色配置的方法有多种，常用方法包括以下两个：

color\_palette，基于RGB原理设置颜色的接口，可接收一个调色板对象作为参数，同时可以设置颜色数量

hls\_palette，基于Hue(色相)、Luminance(亮度)、Saturation(饱和度)原理设置颜色的接口，除了颜色数量参数外，另外3个重要参数即是hls



Seaborn有五个预设好的主题： darkgrid， whitegrid，dark，white，和 ticks，默认为darkgrid

Seaborn将matplotlib的参数划分为两个独立的组合。第一组是设置绘图的外观风格的，第二组主要将绘图的各种元素按比例缩放的，以至可以嵌入到不同的背景环境中。控制这些参数的接口主要有两对方法

**控制风格：axes\_style(), set\_style()**

**缩放绘图：plotting\_context(), set\_context()**

每对方法中的第一个方法（axes\_style(), plotting\_context()）会返回一组字典参数，而第二个方法（set\_style(), set\_context()）会设置matplotlib的默认参数

 在seaborn中，可以利用despine（）方法移除绘图中顶部和右侧的轴线

sinplot()

sns.despine()

despine（）方法中可以利用offset参数讲轴线进行偏置，另外，当刻度没有完全覆盖整个坐标轴的的范围时，利用trim参数限制已有坐标轴的范围。

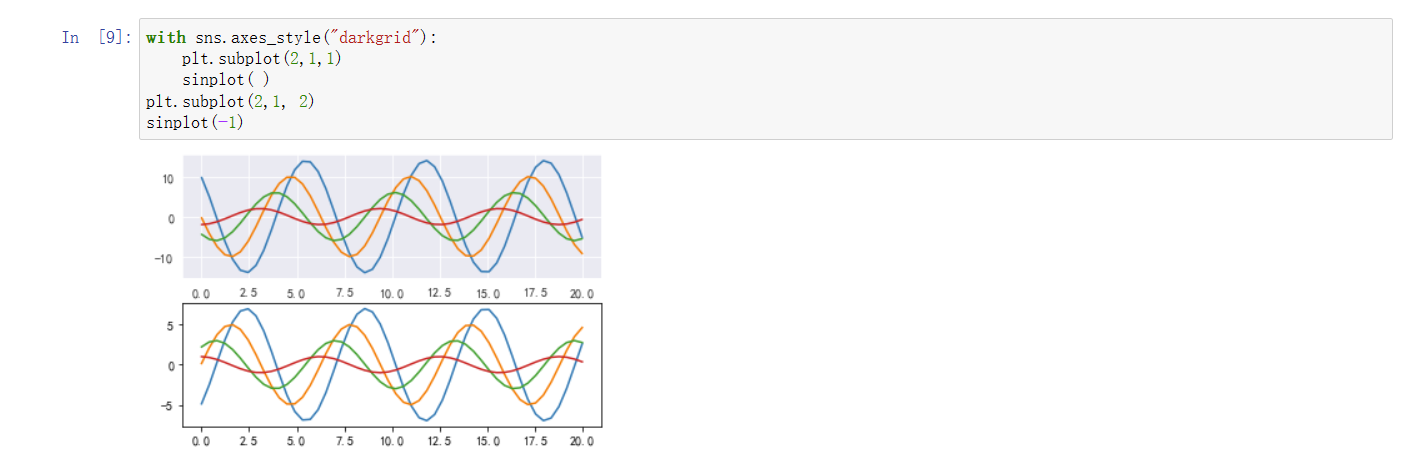
sinplot()

sns.despine(offset = 20, trim = True)

也可以通过despine()控制哪个脊柱将被移除

 除了选用预设的风格外，可以利用with 语句使用axes\_style()方法设置临时绘图参数，可以在一张图片中采用多种绘图风格

设置临时绘图参数

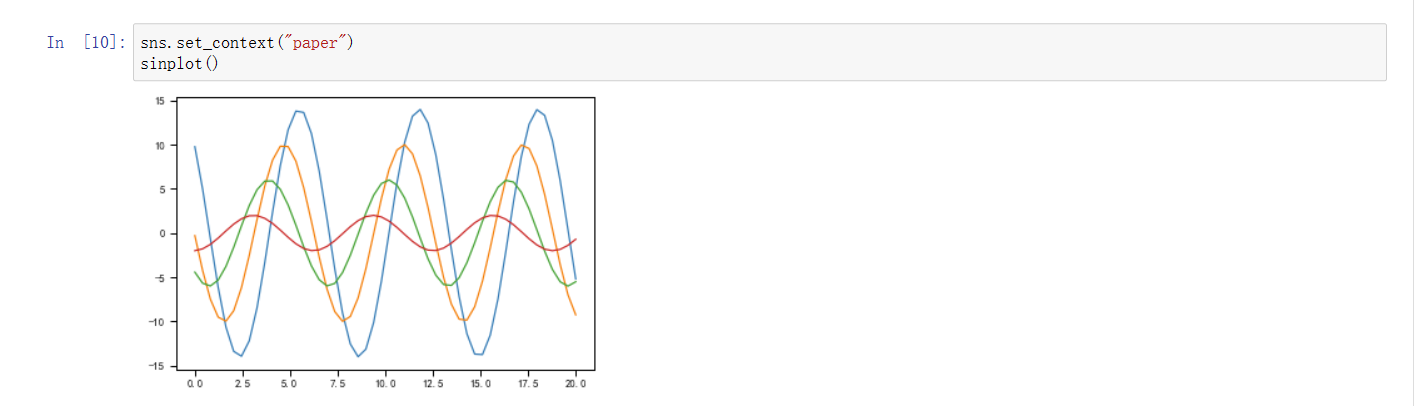


利用字典传递参数

sns.set\_style("darkgrid", {"axes.facecolor": ".9"})

sinplot()

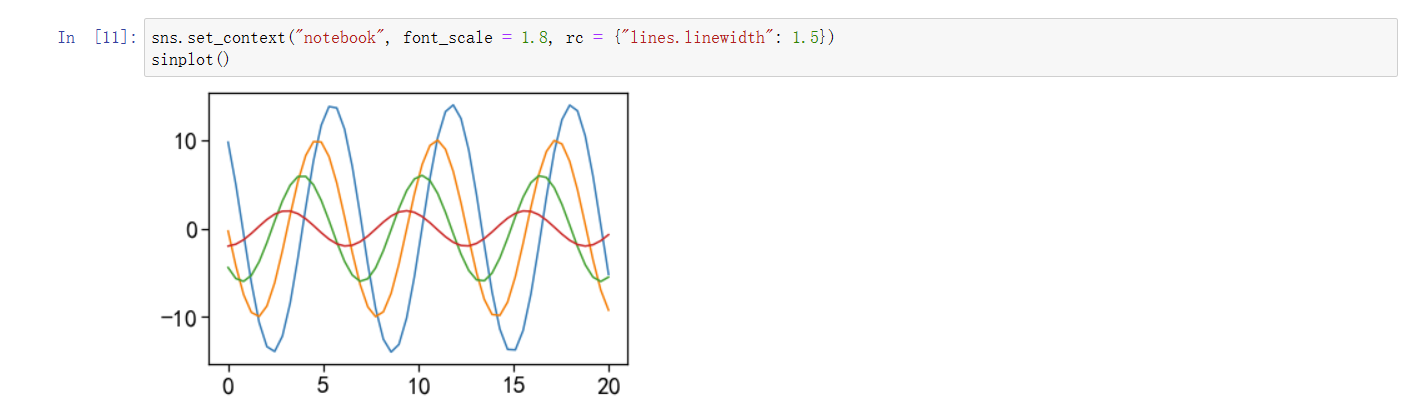
 seaborn中通过set\_context()设置缩放参数，预设的参数有paper, notebook, talk, poster。默认为notebook



 设置绘图元素比例notebook

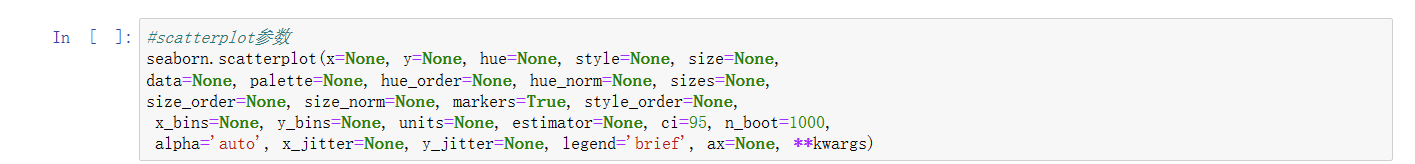
sns.set\_context("notebook", font\_scale = 1.8, rc = {"lines.linewidth": 1.5})

sinplot()



* 1. **常用图表绘制**

散点图



x，y：容易理解就是你需要传入的数据，一般为dataframe中的列；

hue：也是具体的某一可以用做分类的列，作用是分类；

data：是你的数据集，可要可不要，一般都是dataframe；

style：绘图的风格（后面单独介绍）；

size：绘图的大小（后面介绍）；

palette：调色板（后面单独介绍）；

markers：绘图的形状（后面介绍）；

ci：允许的误差范围（空值误差的百分比，0-100之间），可为‘sd’，则采用标准差（默认95）；

n\_boot（int）：计算置信区间要使用的迭代次数；

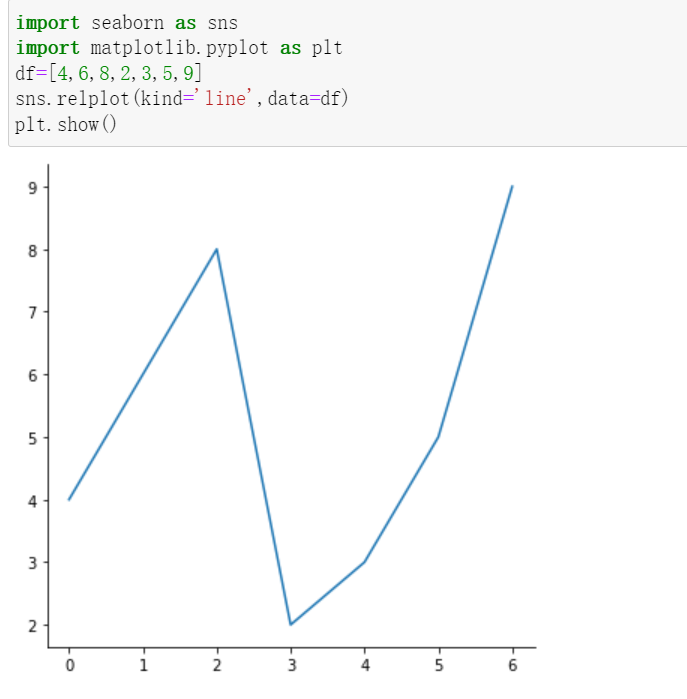
alpha：透明度；

x\_jitter，y\_jitter：设置点的抖动程度。

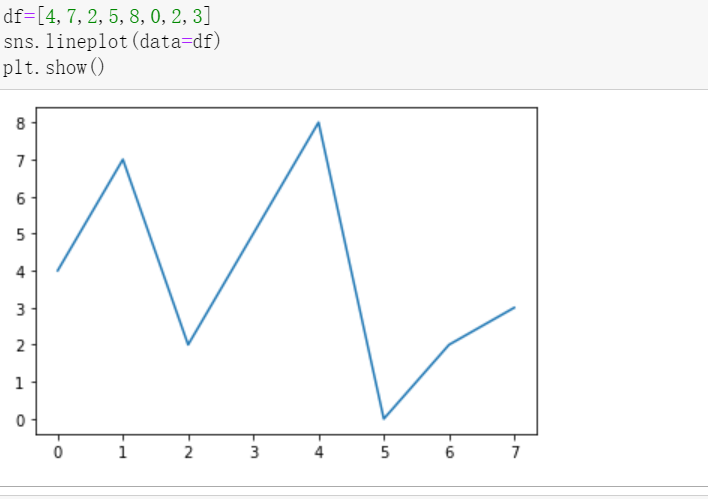


折线图

Replot()



Lineplot()



lmplot（回归图）

许多数据集都有着众多连续变量。数据分析的目的经常就是衡量变量之间的关系，lmplot() 是一个非常有用的方法，它会在绘制二维散点图时，自动完成回归拟合。具体参数如下：

col，row：和前面一样，根据所指定属性在列，行上分类；

col\_wrap：指定每行的列数，最多等于col参数所对应的不同类别的数量；

aspect：控制图的长宽比；

x\_jitter，y\_jitter：给x，y轴随机增加噪音点，设置这两个参数不影响最后的回归直线；

order：多项式回归，控制进行回归的幂次，设定指数，可以用多项式拟合；

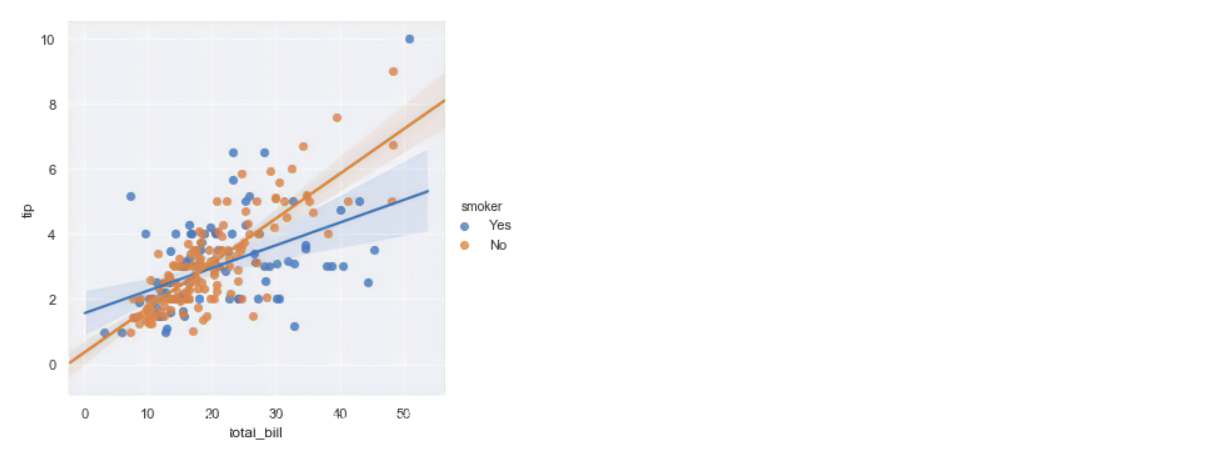
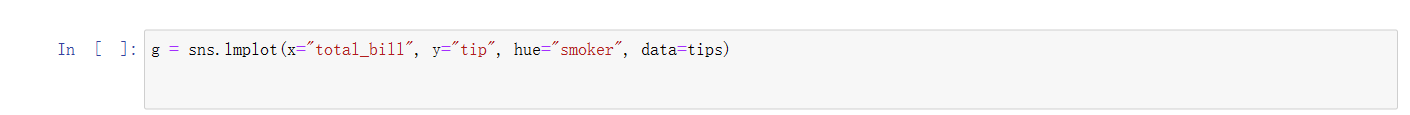
logistic：逻辑回归；

robust：如果是True，使用statsmodels来估计一个稳健的回归（鲁棒线性模型）。这将减少异常值。请注意 logistic回归和robust回归相较于简单线性回归需要更大的计算量，其置信区间的产生也依赖于bootstrap采样，你可以关掉置信区间估计来提高速度（ci=None）；

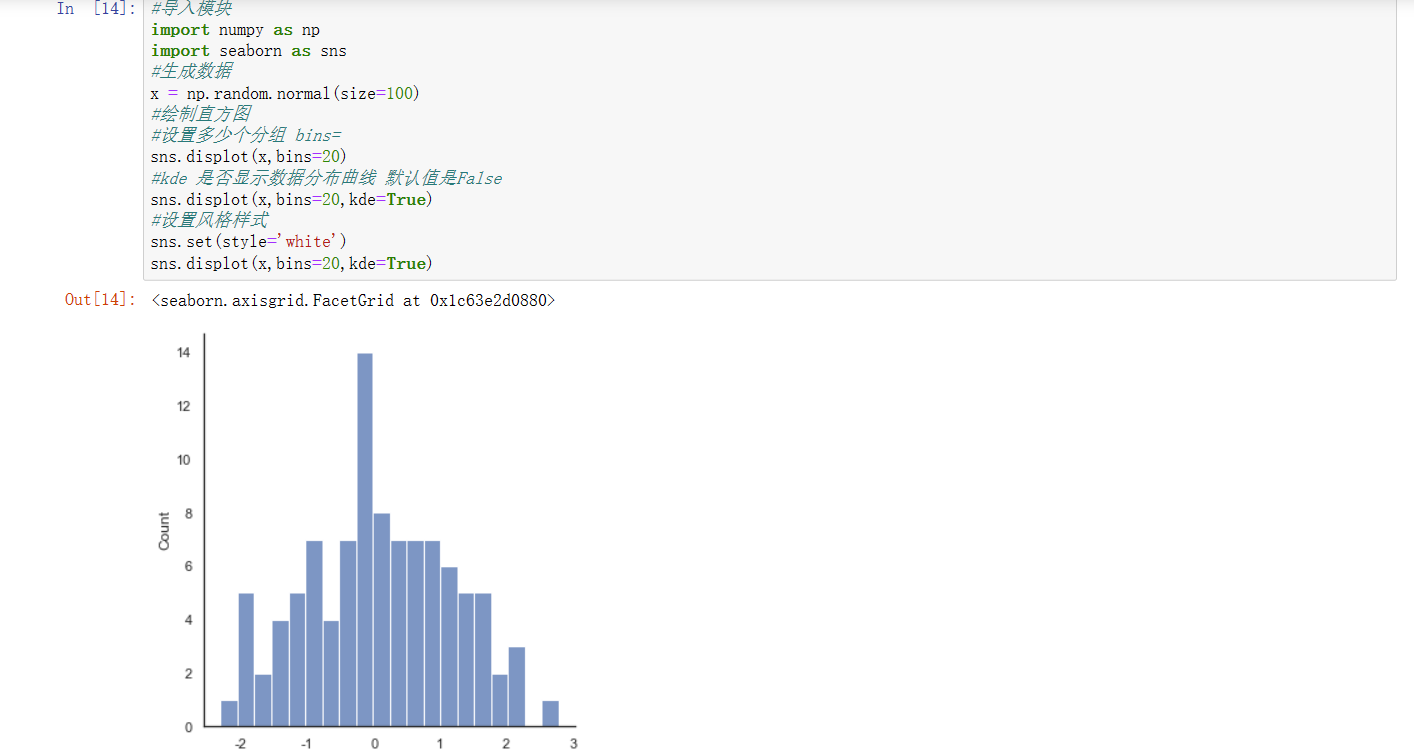
lowess：如果是True，使用statsmodels来估计一个非参数的模型(局部加权线性回归)。这种方法具有最少的假设，尽管它是计算密集型的，但目前无法为这类模型绘制置信区间；

logx：转化为log(x)

truncate：默认情况下，绘制散点图后绘制回归线以填充x轴限制。如果为True，则它将被数据限制所限制；

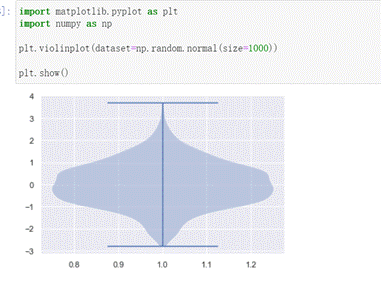


直方图

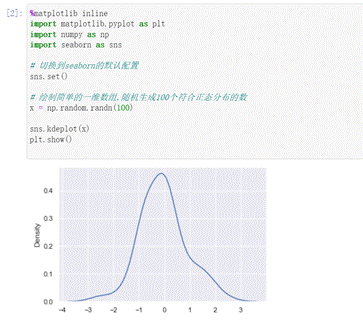


小提琴图

小提琴图是箱图和密度图的一种结合图形。左右越宽代表当前数据 量越密集。



绘制核密度图



热力图

热力图（heatmap）是以矩阵的形式表示，数据值在图形中以颜色 的深浅来表示数量的多少，并可以快速到到大值的与最小值所在位 置。

seaborn.heatmap(data, vmin=None, vmax=None, cmap=None, center=None,

robust=False, annot=None, fmt='.2g', annot\_kws=None, linewidths=0, linecolor='white',

cbar=True, cbar\_kws=None, cbar\_ax=None, square=False, xticklabels='auto',

yticklabels='auto', mask=None, ax=None, \*\*kwargs)

data：矩阵数据集，可以使numpy的数组（array），如果是pandas的dataframe，则df的index/column信息会分别对应到heatmap的columns和rows；

vmax,vmin：图例中最大值和最小值的显示值，没有该参数时默认不显示；

cmap：从数字到色彩空间的映射，取值是matplotlib包里的colormap名称或颜色对象，或者表示颜色的列表；

center：数据表取值有差异时，设置热力图的色彩中心对齐值。通过设置center值，可以调整生成的图像颜色的整体深浅；设置center数据时，如果有数据溢出，则手动设置的vmax、vmin会自动改变 ；

robust：默认取值False；如果是False，且没设定vmin和vmax的值，热力图的颜色映射范围根据具有鲁棒性的分位数设定，而不是用极值设定；

annot(annotate的缩写)：默认取值False；如果是True，在热力图每个方格写入数据；如果是矩阵，在热力图每个方格写入该矩阵对应位置数据；

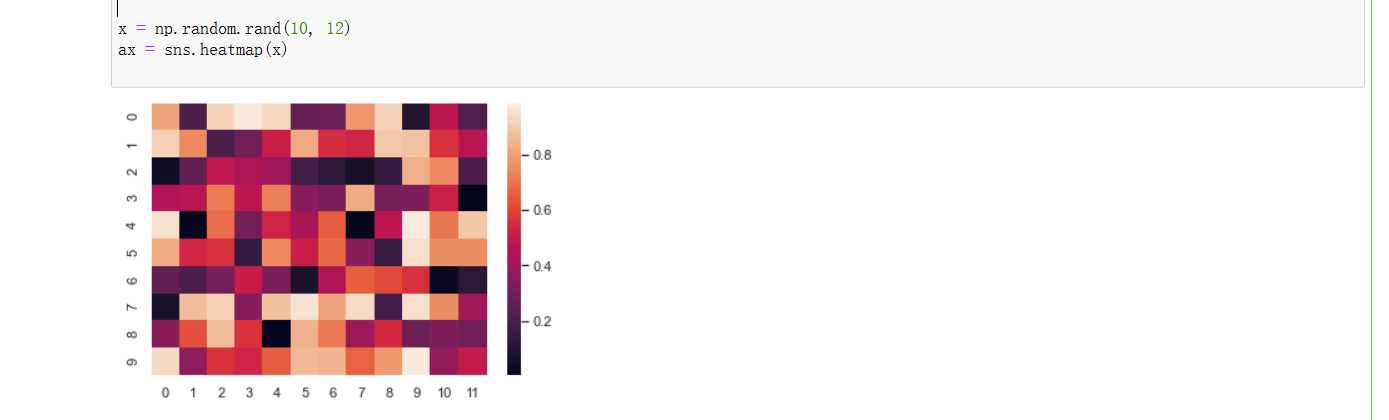
fmt：字符串格式代码，矩阵上标识数字的数据格式，比如保留小数点后几位数字；

annot\_kws:默认取值False；如果是True，设置热力图矩阵上数字的大小颜色字体；

square:设置热力图矩阵小块形状，默认值是False；

xticklabels, yticklabels:控制每行列标签名的输出。默认值是auto，自动选择标签的标注间距，将标签名不重叠的部分(或全部)输出。如果是True，则以DataFrame的列名作为标签名；

mask:控制某个矩阵块是否显示出来。默认值是None。如果是布尔型的DataFrame，则将DataFrame里True的位置用白色覆盖掉。



问题讨论：

1、符号显示问题

plt.rcParams['font.sans-serif']=['SimHei'] # 正常显示中文

plt.rcParams['axes.unicode\_minus']=False # 正常显示负号

2、