

Python程序设计综合训练

郭 宇

目录

01 爬虫练习

02 办公自动化

03 科学计算

04 数据可视化

04

数据可视化

主要实验内容

- 一. Matplotlib概念
- 二. 快速绘图
- 三. 常用绘图函数
- 四. 三维绘图

一、Matplotlib概念

1. 准备工作
2. 图形图像与元素

1.准备工作

➤matplotlib绘图库



Matplotlib is a **comprehensive** library for creating static, animated, and interactive **visualizations** in Python.

➤数据处理库

➤Numpy、pandas、sklearn...

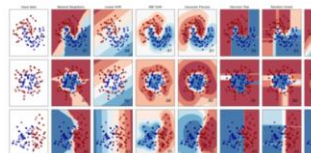


Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.

Algorithms: SVM, nearest neighbors, random forest, and more...

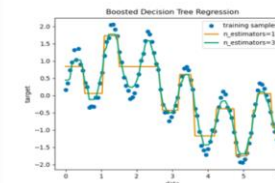


Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, Stock prices.

Algorithms: SVR, nearest neighbors, random forest, and more...



Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, Grouping experiment outcomes

Algorithms: k-Means, spectral clustering, mean-shift, and more...



1.图形对象与元素

figure对象: 整个图形即是一个figure对象。figure对象至少包含一个子图，也就是axes对象。figure对象包含一些特殊的对象，如图名 (title)、图例 (legend)。figure对象包含画布(canvas)对象。canvas对象一般不可见，通常无须直接操作该对象。

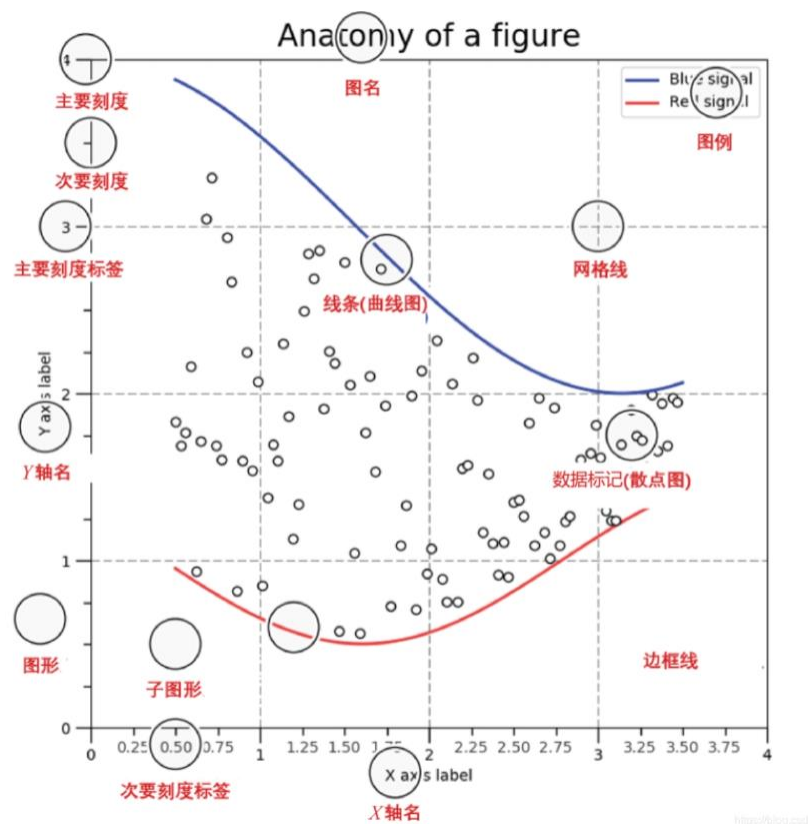
axes对象: 字面上理解，axes是axis (坐标轴) 的复数，但它并不是指坐标轴，而是子图对象可以这样理解，每一个子图都有X轴和Y轴，axes则用于代表这两个坐标轴所对应的一个子图对象。常用方法: **set_xlim()**及**set_ylim()**:设置子图X轴和Y轴对应的数据范围;**set_title()**:设置子图的图名;**set_xlabel()**以及**set_ylabel()**:设置子图X轴和Y轴名。

1.图形对象与元素

axis对象: axis是数据轴对象，主要用于控制数据轴上的刻度位置和显示数值。

tick对象:常见的二维直角坐标系 (axes) 都有两条坐标轴 (axis)，横轴 (X axis) 和纵轴 (Y axis)。每个坐标轴都包含两个元素: 刻度(容器类元素)，该对象里还包含刻度本身和刻度标签; 标签(基础类元素)，该对象包含的是坐标轴标签

1.图形对象与元素



组成元素： 图形 (figure)、坐标图形 (axes)、图名 (title)、图例 (legend)、主要刻度(major tick)、次要刻度 (minor tick)、主要刻度标签 (maior tick label)、次要刻度标签 (minor tick label)、Y轴名 (Y axis label)、X轴名 (Xaxis label)、边框图 (line)、数据标记 (markers)、网格 (grid)线等。

二、快速绘图

1. Pyplot模块
2. 属性设置
3. 绘制子图

1. Pyplot模块

matplotlib.pyplot

```
import matplotlib.pyplot as plt
```

1. 载入pyplot模块;
2. 创建一个Figure对象;
3. 调用`plot()`绘图;
4. 指定颜色线形;
5. 设置坐标轴;
6. 显示窗口。

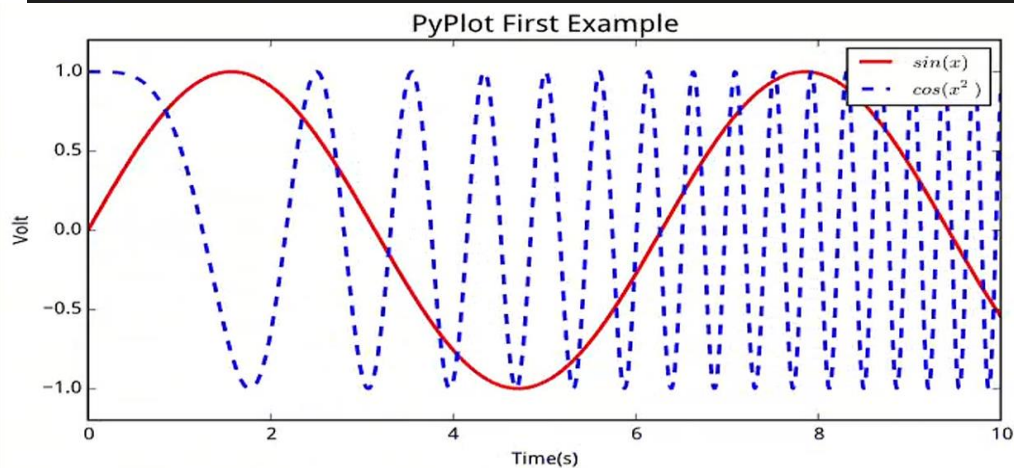
```
x = np.linspace(0, 10, 1000)
y = np.sin(x)
z = np.cos(x**2)

plt.figure(figsize=(8,4)) #②

plt.plot(x,y,label="$sin(x)$",color="red",linewidth=2) #③
plt.plot(x,z,"b--",label="$cos(x^2)$") #④

plt.xlabel("Time(s)") #⑤
plt.ylabel("Volt")
plt.title("PyPlot First Example")
plt.ylim(-1.2,1.2)
plt.legend()

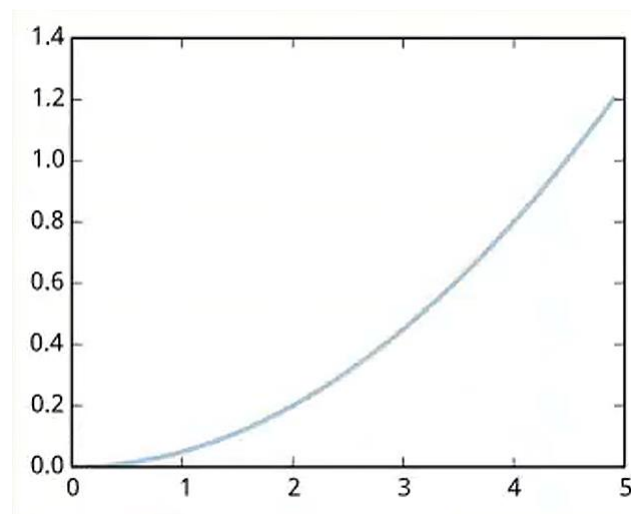
plt.show() #⑥
```



2. 属性设置

➤Matplotlib所绘制图表的每个组成部分都对应于一个对象，可调用这些对象的属性设置方法**`set_*`**(***或*****`setp`**())设置其属性。

```
##fig[1x3]=配置绘图对象的属性  
plt.figure(figsize=(4, 3))  
x = np.arange(0, 5, 0.1)  
line = plt.plot(x, 0.05*x*x)[0] # plot返回一个列表  
line.set_alpha(0.5) # 调用Line2D对象的set_*( )方法设置属性值
```

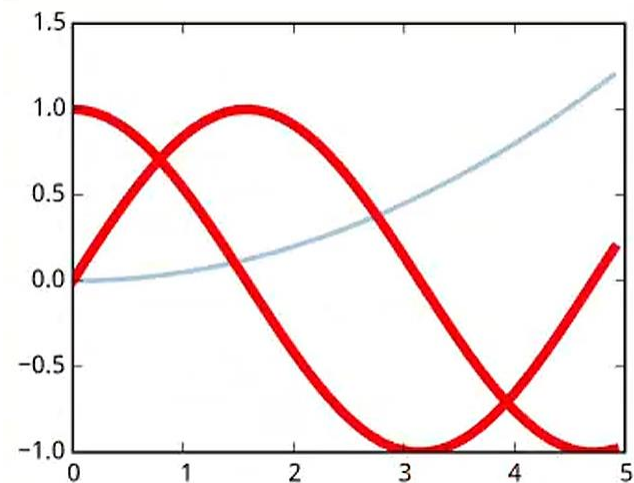
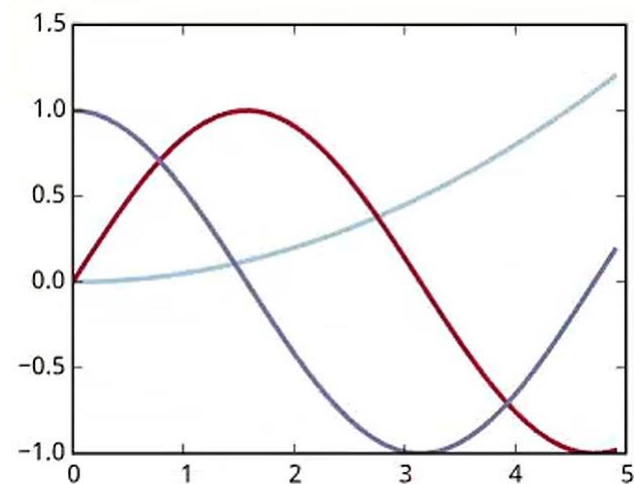


2. 属性设置

```
lines = plt.plot(x, np.sin(x), x, np.cos(x))
```



```
plt.setp(lines, color="r", linewidth=4.0);
```



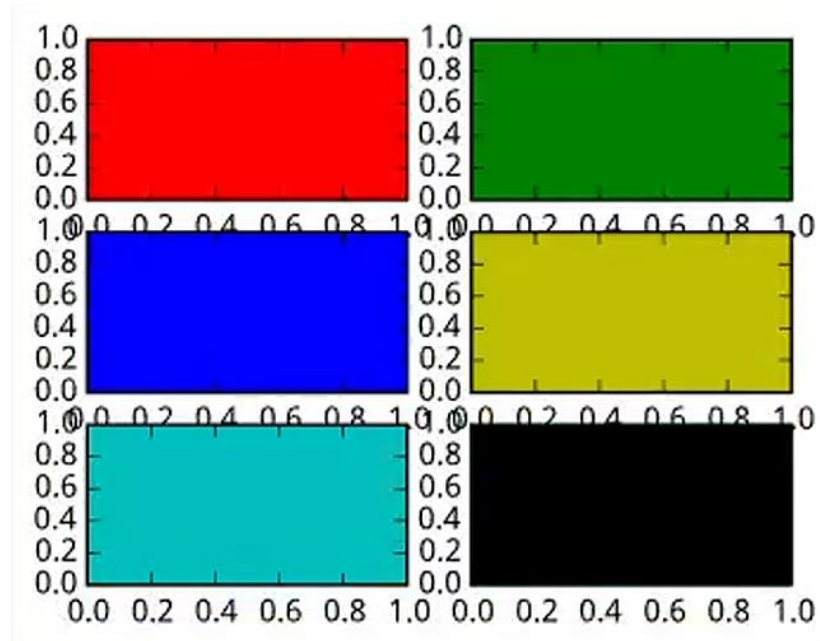
3. 绘制子图

一个figure对象可以包含多个子图，一个子图对象表示一个绘图区域，可用

subplot(numR,numC,plotNum)

快速绘制包含多个子图的图表。

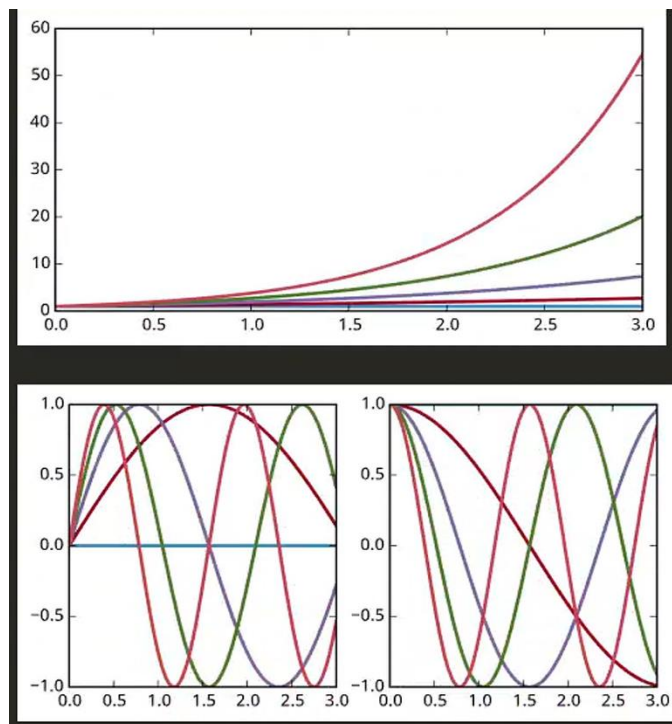
```
for idx, color in enumerate("rgbyck"):
    plt.subplot(321+idx, axisbg=color)
```



3. 绘制子图

```
#!/fig[1x2]=同时多幅图表、多个子图中进行绘图
plt.figure(1) # 创建图表1
plt.figure(2) # 创建图表2
ax1 = plt.subplot(121) # 在图表2中创建子图1
ax2 = plt.subplot(122) # 在图表2中创建子图2

x = np.linspace(0, 3, 100)
for i in xrange(5):
    plt.figure(1) # 选择图表1
    plt.plot(x, np.exp(i*x/3))
    plt.sca(ax1) # 选择图表2的子图1
    plt.plot(x, np.sin(i*x))
    plt.sca(ax2) # 选择图表2的子图2
    plt.plot(x, np.cos(i*x))
```



三、常用绘图函数

1. 对数坐标图

2. 柱状图

3. 散列图

4. 图像

5. 等值线

6. 三维绘图

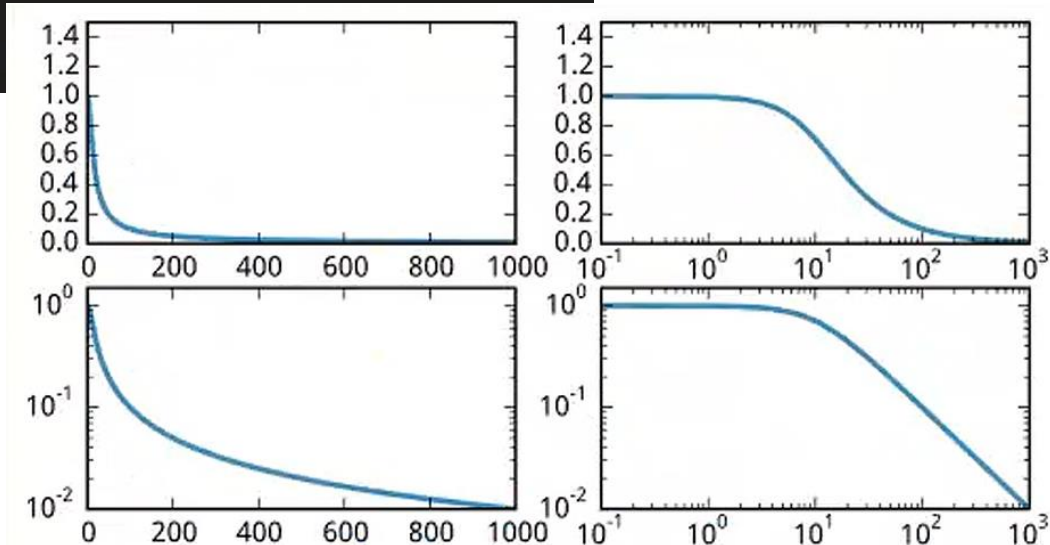
1. 对数坐标图

```
##fig=低通滤波器的频率响应: 算术坐标 (左上)、x轴对数坐标 (右上)、y轴对数坐标 (左下)、双对数坐标 (右上)
w = np.linspace(0.1, 1000, 1000)
p = np.abs(1/(1+0.1j*w)) # 计算低通滤波器的频率响应

fig, axes = plt.subplots(2, 2)

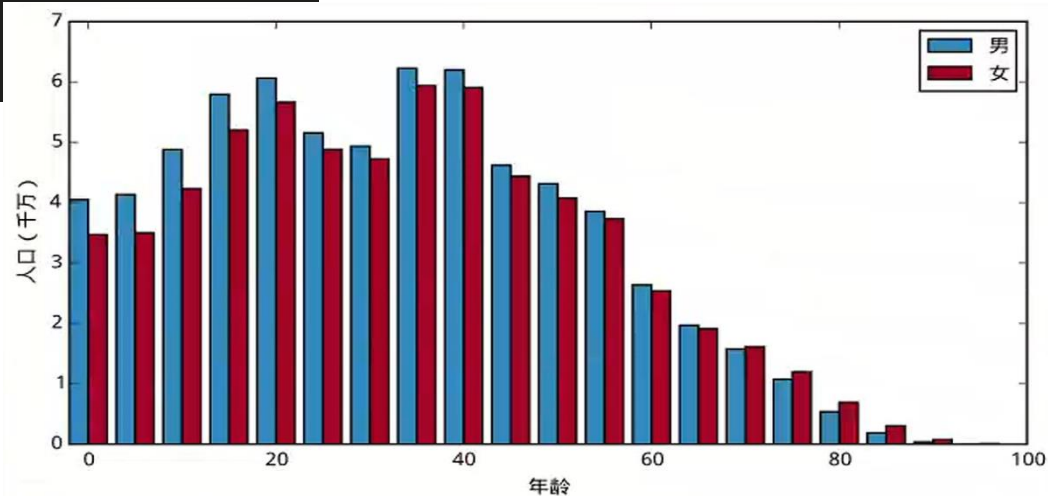
functions = ("plot", "semilogx", "semilogy", "loglog")

for ax, fname in zip(axes.ravel(), functions):
    func = getattr(ax, fname)
    func(w, p, linewidth=2)
    ax.set_ylim(0, 1.5)
```



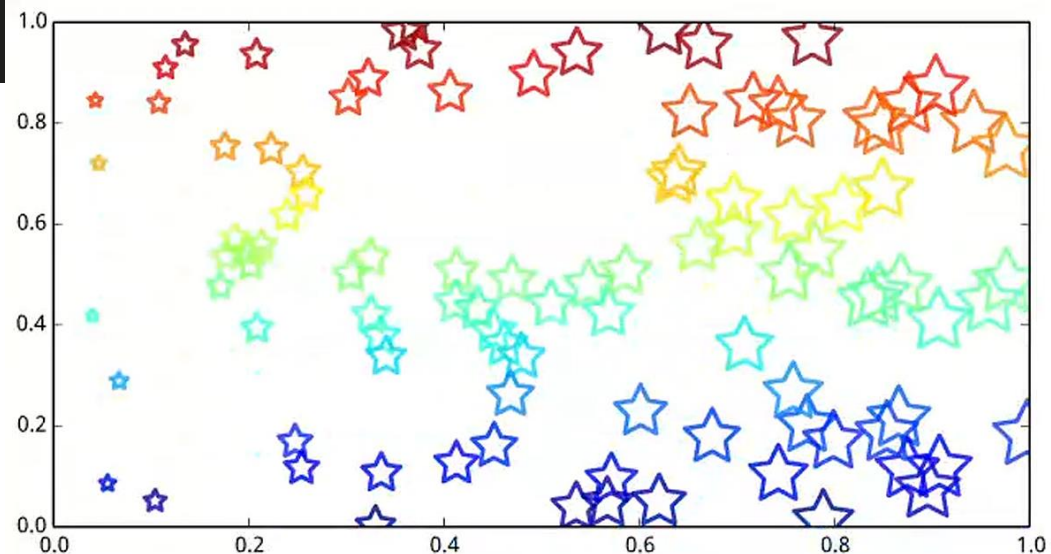
2. 柱状图

```
##fig=中国男女人口的年龄分布图
data = np.loadtxt("china_population.txt")
width = (data[1,0] - data[0,0])*0.4 #①
plt.figure(figsize=(8, 4))
c1, c2 = plt.rcParams['axes.color_cycle'][:2]
plt.bar(data[:,0]-width, data[:,1]/1e7, width, color=c1, label=u"男") #②
plt.bar(data[:,0], data[:,2]/1e7, width, color=c2, label=u"女") #③
plt.xlim(-width, 100)
plt.xlabel(u"年龄")
plt.ylabel(u"人口（千万）")
plt.legend();
```



3. 散列图

```
#%fig=可指定点的颜色和大小的散列  
plt.figure(figsize=(8, 4))  
x = np.random.random(100)  
y = np.random.random(100)  
plt.scatter(x, y, s=x*1000, c=y, marker=(5, 1),  
            alpha=0.8, lw=2, facecolors="none")  
plt.xlim(0, 1)  
plt.ylim(0, 1);
```



4. 图像

```
##fig=用imread()和imshow()显示图像
img = plt.imread("lena.jpg")
fig, axes = plt.subplots(2, 4, figsize=(11, 4))
fig.subplots_adjust(0, 0, 1, 1, 0.05, 0.05)

axes = axes.ravel()

axes[0].imshow(img) #①
axes[1].imshow(img, origin="lower") #②
axes[2].imshow(img * 1.0) #③
axes[3].imshow(img / 255.0) #④
axes[4].imshow(np.clip(img / 200.0, 0, 1)) #⑤

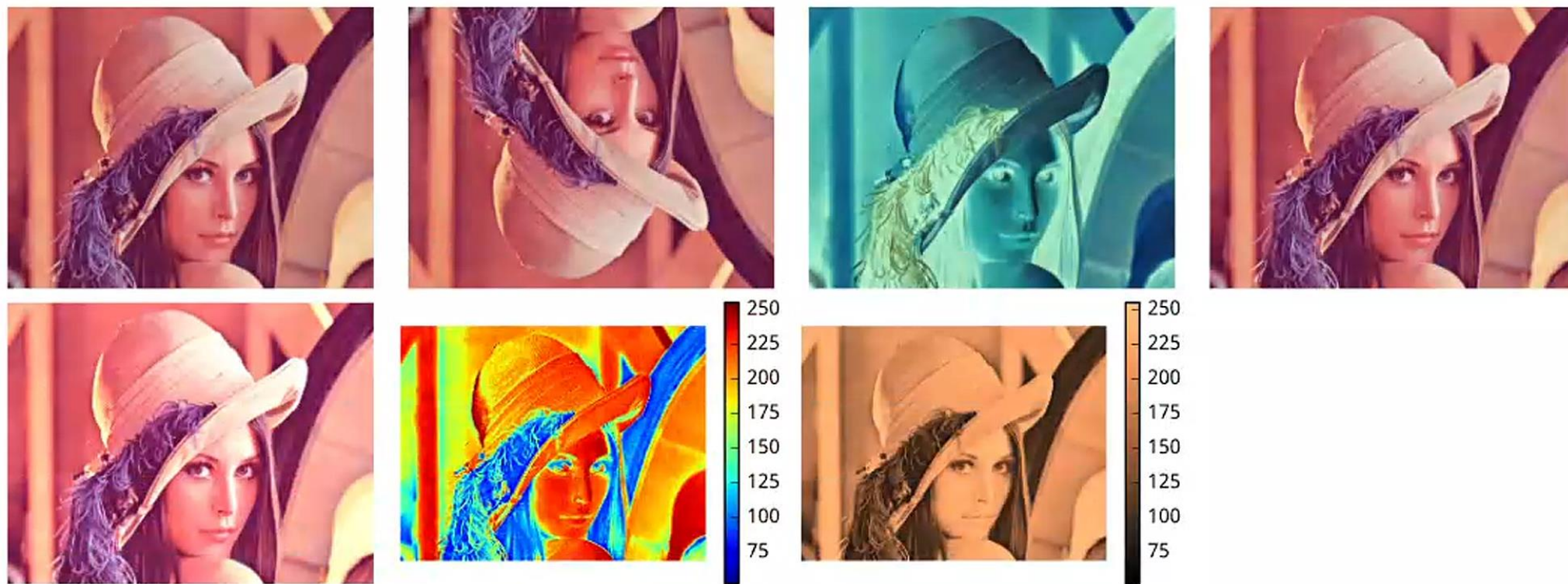
axe_img = axes[5].imshow(img[:, :, 0]) #⑥
plt.colorbar(axe_img, ax=axes[5])

axe_img = axes[6].imshow(img[:, :, 0], cmap="copper") #⑦
plt.colorbar(axe_img, ax=axes[6])

for ax in axes:
    ax.set_axis_off()
```

- 1.显示图像;
- 2.圆点设置在左下角;
- 3.转为浮点数, 超过了1.0, 颜色异常;
- 4.归一化到0.0—1.0, 显示正常;
- 5.超出范围的值限定在0—1之间;
- 6.取第一通道(Red), 并映射到colorbar;
- 7.更换映射方案。

4. 图像



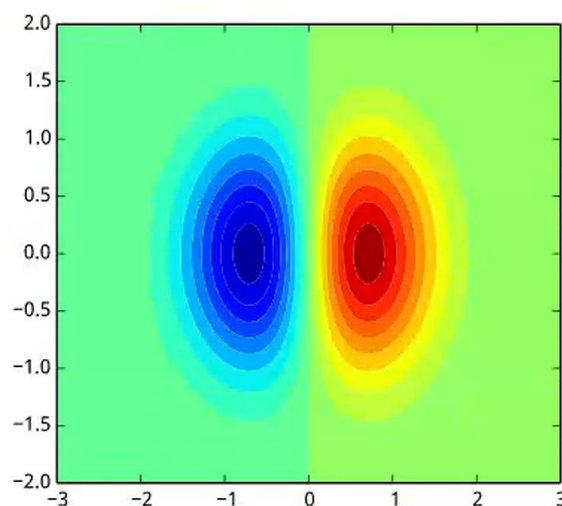
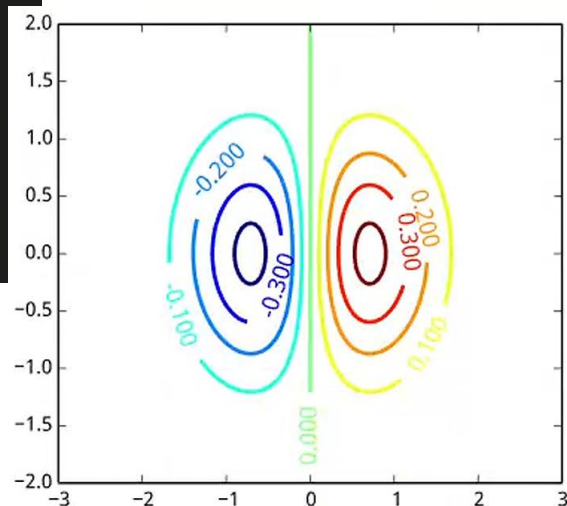
5.等值线

```
##fig=用contour(左)和contourf(右)描绘等值线图
y, x = np.ogrid[-2:2:200j, -3:3:300j] #0
z = x * np.exp(- x**2 - y**2)

extent = [np.min(x), np.max(x), np.min(y), np.max(y)]

plt.figure(figsize=(10,4))
plt.subplot(121)
cs = plt.contour(z, 10, extent=extent) #2
plt.clabel(cs) #3
plt.subplot(122)
plt.contourf(x.reshape(-1), y.reshape(-1), z, 20) #4;
```

- 1.设定xy的范围和等分数量;
- 2.第二个参数为取值范围等分为10个区间;
- 3.为等值线标对应数值
- 4.具有填充效果的等值线图。



6. 三维绘图

```
##fig=使用matplotlib3D绘制的三维曲面图
import mpl_toolkits.mplot3d #0

x, y = np.mgrid[-2:2:20j, -2:2:20j] #0
z = x * np.exp( - x**2 - y**2)

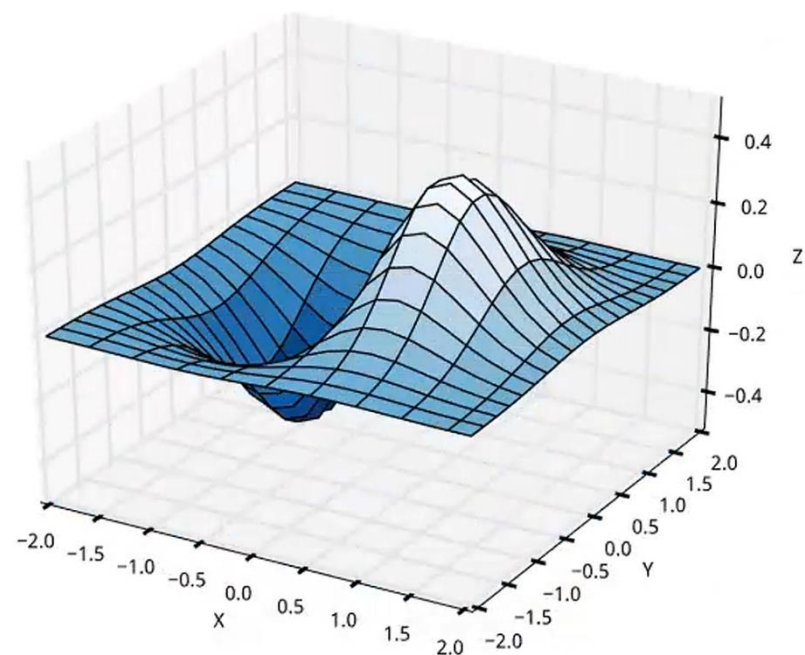
fig = plt.figure(figsize=(8, 6))
ax = plt.subplot(111, projection='3d') #0
ax.plot_surface(x, y, z, rstride=2, cstride=1, cmap = plt.cm.Blues_r) #0
ax.set_xlabel("X")
ax.set_ylabel("Y")
ax.set_zlabel("Z");
```

1.载入mplot3d;

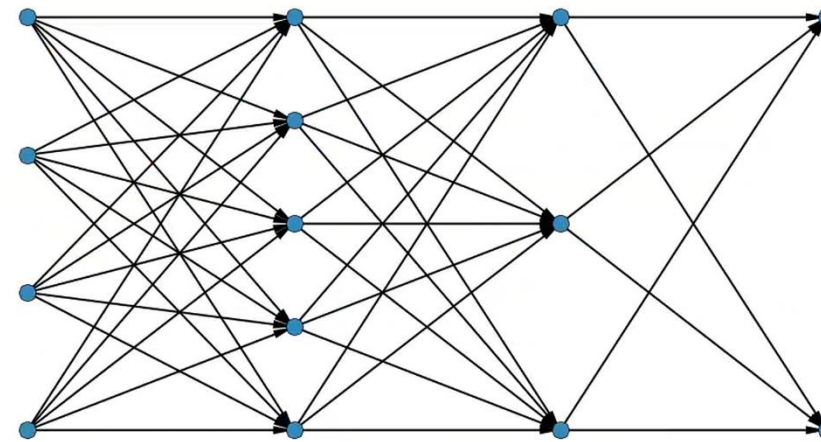
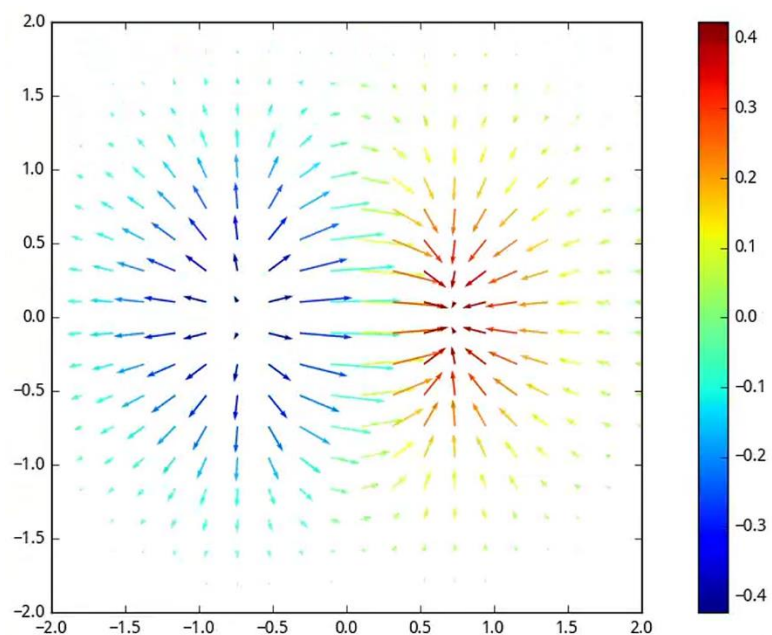
2.创建网格;

3.使用projection指定子图投影模式;

4.绘制三维曲面图，stride是间隔。



探索1：箭头图 *quiver()*



探索2： 波形图 *stackplot* ()

