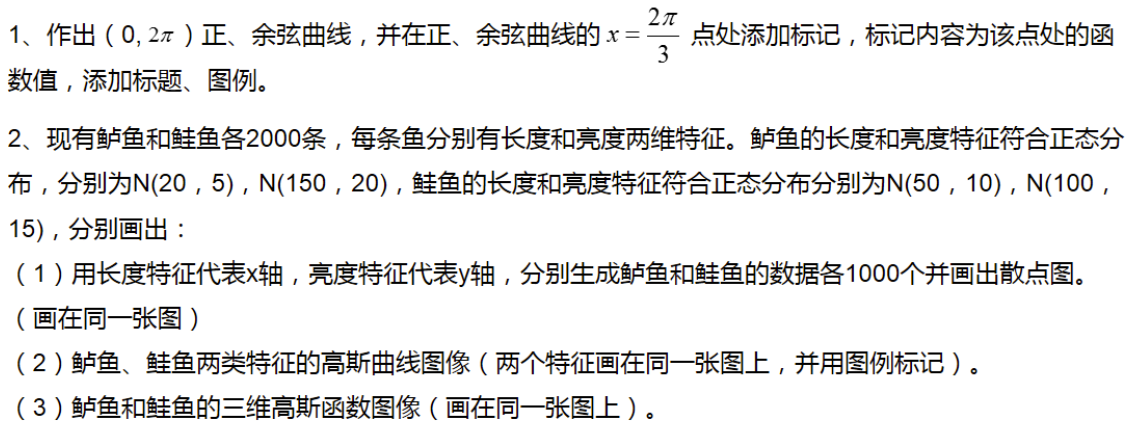
 

**智能信息系统综合实践**

**实验报告**

|  |  |
| --- | --- |
| **题 目：** | Python画图 |
| **年 级：** | **2020级** |
| **专 业：** | **软件工程** |
| **姓 名：** | **庞晓宇** |

1. **题目（原题目）**



1. **解题步骤（思路+代码）**

作出(0, 2pi)的正、余弦曲线，并在正、余弦曲线的x=2\*pi/3处添加标记，标记内容为该点的函数值，添加标题、图例。

* 导入需要的库：numpy和matplotlib.pyplot。
* 使用np.linspace函数生成0到2π之间的100个数据点，作为x坐标轴的值。
* 分别计算正弦和余弦的函数值，并存储在y\_sin和y\_cos数组中。
* 使用plt.plot函数绘制正弦和余弦曲线，并添加标签。
* 使用plt.scatter函数在x=pi/3处添加标记，其中color参数指定标记的颜色，marker参数指定标记的形状。
* 使用plt.text函数在标记旁边添加文本，其中第一个参数是文本的x坐标，第二个参数是文本的y坐标，第三个参数是文本的内容，其中{:.2f}表示保留两位小数。
* 使用plt.title函数添加标题。
* 使用plt.legend函数添加图例。
* 最后使用plt.show函数显示图形。

In [ ]:

**import** numpy **as** np

**import** matplotlib.pyplot **as** plt

*# 生成数据*

x **=** np**.**linspace(0, 2**\***np**.**pi, 100)

y\_sin **=** np**.**sin(x)

y\_cos **=** np**.**cos(x)

*# 绘制正弦曲线和余弦曲线*

plt**.**plot(x, y\_sin, label**=**'sin(x)')

plt**.**plot(x, y\_cos, label**=**'cos(x)')

*# 在x=pi/3处添加标记*

x\_mark **=** 2**\***np**.**pi**/**3

y\_mark\_sin **=** np**.**sin(x\_mark)

y\_mark\_cos **=** np**.**cos(x\_mark)

plt**.**scatter(x\_mark, y\_mark\_sin, color**=**'r', marker**=**'o')

plt**.**text(x\_mark**+**0.1, y\_mark\_sin, f'{y\_mark\_sin:.2f}')

plt**.**scatter(x\_mark, y\_mark\_cos, color**=**'b', marker**=**'o')

plt**.**text(x\_mark**+**0.1, y\_mark\_cos, f'{y\_mark\_cos:.2f}')

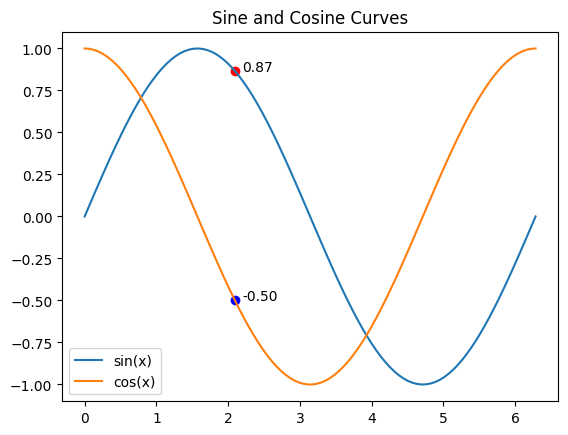
*# 添加标题和图例*

plt**.**title('Sine and Cosine Curves')

plt**.**legend()

*# 显示图形*

plt**.**show()



现有鲈鱼和鲤鱼各2000条，每条鱼分别有长度和亮度两维特征。鲈鱼的长度和亮度特征符合正态分布，分别为N(20, 5)和N(150, 20)；鲑鱼的长度和亮度特征符合正态分布，分别为N(50, 10)和N(100, 15)。用长度特征代表x轴，亮度特征代表y轴，分别生成鲈鱼和鲑鱼各1000个并画出散点图。

* 定义鲈鱼和鲑鱼的长度和亮度特征的均值和标准差。
* 使用np.random.normal函数分别生成1000个符合正态分布的鲈鱼和鲑鱼的长度和亮度特征数据。
* 使用plt.scatter函数绘制散点图，其中第一个参数是x坐标轴的值，第二个参数是y坐标轴的值，label参数指定图例中显示的标签。
* 使用plt.title函数添加标题，plt.xlabel函数添加x轴的标签，plt.ylabel函数添加y轴的标签，plt.legend函数添加图例。
* 最后使用plt.show函数显示图形。

In [ ]:

**import** numpy **as** np

**import** matplotlib.pyplot **as** plt

*# 鲈鱼长度和亮度特征*

mean\_lu, std\_lu **=** 20, 5

mean\_ly, std\_ly **=** 150, 20

*# 鲑鱼长度和亮度特征*

mean\_gu, std\_gu **=** 50, 10

mean\_gy, std\_gy **=** 100, 15

*# 生成鲈鱼和鲑鱼的数据*

lu\_length **=** np**.**random**.**normal(mean\_lu, std\_lu, 1000)

lu\_luminance **=** np**.**random**.**normal(mean\_ly, std\_ly, 1000)

gu\_length **=** np**.**random**.**normal(mean\_gu, std\_gu, 1000)

gu\_luminance **=** np**.**random**.**normal(mean\_gy, std\_gy, 1000)

*# 绘制散点图*

plt**.**scatter(lu\_length, lu\_luminance, label**=**'Lu')

plt**.**scatter(gu\_length, gu\_luminance, label**=**'Gu')

*# 添加标题和图例*

plt**.**title('Fish Scatter Plot')

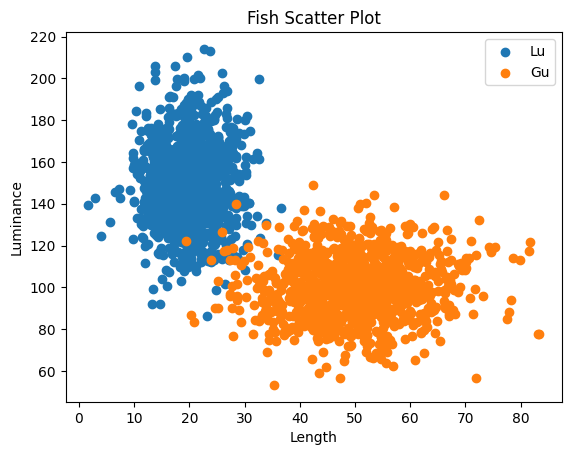
plt**.**xlabel('Length')

plt**.**ylabel('Luminance')

plt**.**legend()

*# 显示图形*

plt**.**show()



现有鲈鱼和鲤鱼各2000条，每条鱼分别有长度和亮度两维特征。鲈鱼的长度和亮度特征符合正态分布，分别为N(20, 5)和N(150, 20)；鲑鱼的长度和亮度特征符合正态分布，分别为N(50, 10)和N(100, 15)。鲈鱼、鲑鱼两类特征的高斯曲线图像，两个特征画在同一张图上，并使用图例标记。

* 定义鲈鱼和鲑鱼的长度和亮度特征的均值和标准差。
* 使用np.linspace函数定义x轴范围，并生成1000个等间隔的x坐标轴数据。
* 使用高斯分布公式计算鲈鱼和鲑鱼两类特征的概率密度函数，并将结果保存在pdf\_lu\_length、pdf\_lu\_luminance、pdf\_gu\_length和pdf\_gu\_luminance变量中。
* 使用plt.plot函数绘制鲈鱼和鲑鱼两类特征的高斯曲线，其中第一个参数是x坐标轴的值，第二个参数是y坐标轴的值，label参数指定图例中显示的标签。
* 使用plt.title函数添加标题，plt.xlabel函数添加x轴的标签，plt.ylabel函数添加y轴的标签，plt.legend函数添加图例。
* 最后使用plt.show函数显示图形。

In [ ]:

**import** numpy **as** np

**import** matplotlib.pyplot **as** plt

*# 鲈鱼长度和亮度特征*

mean\_lu, std\_lu **=** 20, 5

mean\_ly, std\_ly **=** 150, 20

*# 鲑鱼长度和亮度特征*

mean\_gu, std\_gu **=** 50, 10

mean\_gy, std\_gy **=** 100, 15

*# 定义x轴范围*

x\_min, x\_max **=** 0, 80

x **=** np**.**linspace(x\_min, x\_max, 1000)

*# 计算鲈鱼和鲑鱼两类特征的高斯分布*

pdf\_lu\_length **=** 1 **/** (std\_lu **\*** np**.**sqrt(2 **\*** np**.**pi)) **\*** np**.**exp(**-**(x **-** mean\_lu)**\*\***2 **/** (2 **\*** std\_lu**\*\***2))

pdf\_lu\_luminance **=** 1 **/** (std\_ly **\*** np**.**sqrt(2 **\*** np**.**pi)) **\*** np**.**exp(**-**(x **-** mean\_ly)**\*\***2 **/** (2 **\*** std\_ly**\*\***2))

pdf\_gu\_length **=** 1 **/** (std\_gu **\*** np**.**sqrt(2 **\*** np**.**pi)) **\*** np**.**exp(**-**(x **-** mean\_gu)**\*\***2 **/** (2 **\*** std\_gu**\*\***2))

pdf\_gu\_luminance **=** 1 **/** (std\_gy **\*** np**.**sqrt(2 **\*** np**.**pi)) **\*** np**.**exp(**-**(x **-** mean\_gy)**\*\***2 **/** (2 **\*** std\_gy**\*\***2))

*# 绘制鲈鱼和鲑鱼两类特征的高斯曲线*

plt**.**plot(x, pdf\_lu\_length, label**=**'Lu Length')

plt**.**plot(x, pdf\_lu\_luminance, label**=**'Lu Luminance')

plt**.**plot(x, pdf\_gu\_length, label**=**'Gu Length')

plt**.**plot(x, pdf\_gu\_luminance, label**=**'Gu Luminance')

*# 添加标题和图例*

plt**.**title('Fish Gaussian Distribution')

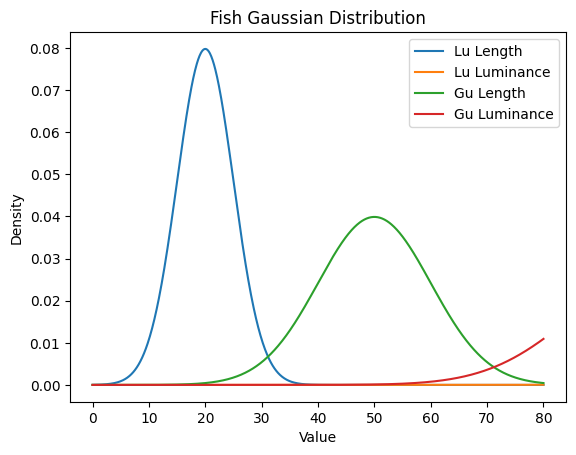
plt**.**xlabel('Value')

plt**.**ylabel('Density')

plt**.**legend()

*# 显示图形*

plt**.**show()



现有鲈鱼和鲤鱼各2000条，每条鱼分别有长度和亮度两维特征。鲈鱼的长度和亮度特征符合正态分布，分别为N(20, 5)和N(150, 20)；鲑鱼的长度和亮度特征符合正态分布，分别为N(50, 10)和N(100, 15)。给出鲈鱼和鲑鱼的三维高斯函数图像。

In [ ]:

**import** numpy **as** np

np**.**random**.**seed(42)

*# 生成鲈鱼的数据*

sea\_bass\_length **=** np**.**random**.**normal(loc**=**20, scale**=**5, size**=**2000)

sea\_bass\_brightness **=** np**.**random**.**normal(loc**=**150, scale**=**20, size**=**2000)

sea\_bass\_data **=** np**.**array([sea\_bass\_length, sea\_bass\_brightness])**.**T

*# 生成鲑鱼的数据*

salmon\_length **=** np**.**random**.**normal(loc**=**50, scale**=**10, size**=**2000)

salmon\_brightness **=** np**.**random**.**normal(loc**=**100, scale**=**15, size**=**2000)

salmon\_data **=** np**.**array([salmon\_length, salmon\_brightness])**.**T

In [ ]:

**from** mpl\_toolkits.mplot3d **import** Axes3D

**import** matplotlib.pyplot **as** plt

*# 创建三维坐标系*

fig **=** plt**.**figure()

ax **=** fig**.**add\_subplot(111, projection**=**'3d')

*# 定义高斯函数*

**def** gaussian(x, mu, sigma):

**return** 1 **/** (sigma **\*** np**.**sqrt(2 **\*** np**.**pi)) **\*** np**.**exp(**-**(x **-** mu)**\*\***2 **/** (2 **\*** sigma**\*\***2))

*# 创建一个点的网格来绘制函数*

x **=** np**.**linspace(0, 80, 100)

y **=** np**.**linspace(80, 220, 100)

X, Y **=** np**.**meshgrid(x, y)

*# 计算鲈鱼的概率密度函数*

Z\_sea\_bass **=** gaussian(X, 20, 5) **\*** gaussian(Y, 150, 20)

*# 鲑鱼概率密度函数的计算*

Z\_salmon **=** gaussian(X, 50, 10) **\*** gaussian(Y, 100, 15)

*# 将函数绘制为曲面*

ax**.**plot\_surface(X, Y, Z\_sea\_bass, cmap**=**'Blues', alpha**=**0.8, label**=**'Sea bass')

ax**.**plot\_surface(X, Y, Z\_salmon, cmap**=**'Oranges', alpha**=**0.8, label**=**'Salmon')

*# 添加图例和轴标签*

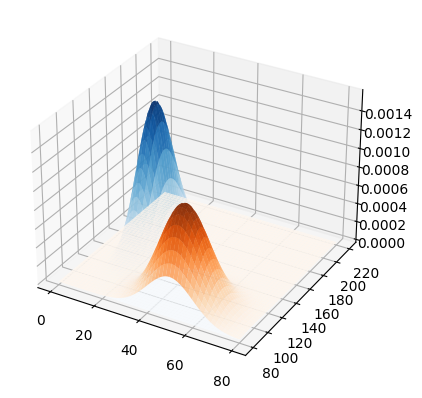
ax**.**legend()

ax**.**set\_xlabel('Length')

ax**.**set\_ylabel('Brightness')

ax**.**set\_zlabel('Probability density')

plt**.**show()



1. **总结（心得体会）**

通过编写上述代码，学习看Python中matplotlib和numpy的使用，体会到了Python在数据分析和可视化方面的强大功能。首先，我使用NumPy库生成了符合正态分布的随机数据，并使用Matplotlib库绘制了正弦、余弦曲线和散点图，还生成了高斯分布的概率密度函数图像。这些图像使我更好地理解了这些概念，并能够更好地对数据进行可视化。同时，我也学会了使用Matplotlib库对图像进行美化和自定义，包括修改线条颜色、添加标题和图例、调整坐标轴范围等等，这些技能能够帮助我更好地呈现数据。最后，我还掌握了使用mpl\_toolkits.mplot3d库生成三维高斯函数图像的方法，这可以帮助我更好地理解多维数据分布规律，并且可以帮助我更好地展示数据的多维特征。