**《数据统计与分析基础实验》**

实验三报告

|  |  |
| --- | --- |
| 班级： | 计224 |
| 姓名： | 王小康 |
| 学号： | 225432 |

**实验三 常规数学统计运算**

**一、实验目的(标题1，微软雅黑，小四，加粗)**

掌握各种常见的数学统计、数学分析运算的编程实现过程。

**二、实验内容**

1、随机生成一个10×15的高斯矩阵，均值为自己学号后两位，方差为1。对该矩阵分别进行LU、QR、奇异值，并展示分解结果。

2、利用软件求解优化问题：

min Z = - X1 - 0.8X2 - 1.2X3,

约束：X1 –X2 + X3 <= 30,

3X1 + 2X2 + 4X3 <=42,

3X1 + 2X2 <= 30,

X1, X2 , X3 >=0。

并找到其中有效约束。

3、统计自己过去12个月实际生活花费的数值，并拟合成一条一次、二次、三次曲线，三条曲线分别用不同颜色和线型展示在同一张图里。

4、某人进行射击，及每次命中的概率为0.1，独立射击50次，求击中10次以上且40次以下的概率。

5、假设自己过去12个月实际生活花费的数值服从正态分布，请求出其均值和方差的极大似然估计。

**三、实验源程序及运行结果（可附截图）**

1.程序如下：

import numpy as np

import scipy.linalg as la

import matplotlib.pyplot as plt

# 随机种子，确保结果可重复

np.random.seed(42)

student\_number\_last\_two\_digits = 32

# 生成10×15的高斯矩阵

mean = student\_number\_last\_two\_digits

std\_dev = 1

matrix = np.random.normal(mean, std\_dev, (10, 15))

# LU分解

P, L, U = la.lu(matrix)

# QR分解

Q, R = np.linalg.qr(matrix)

# 奇异值分解

U\_svd, S, Vt = np.linalg.svd(matrix)

# 绘制分解结果

fig, axes = plt.subplots(2, 2, figsize=(12, 12))

# LU分解的L和U矩阵

axes[0, 0].imshow(L, cmap='viridis')

axes[0, 0].set\_title('LU-L')

axes[0, 1].imshow(U, cmap='viridis')

axes[0, 1].set\_title('LU-U')

# QR分解的Q和R矩阵

axes[1, 0].imshow(Q, cmap='viridis')

axes[1, 0].set\_title('QR-Q')

axes[1, 1].imshow(R, cmap='viridis')

axes[1, 1].set\_title('QR-R')

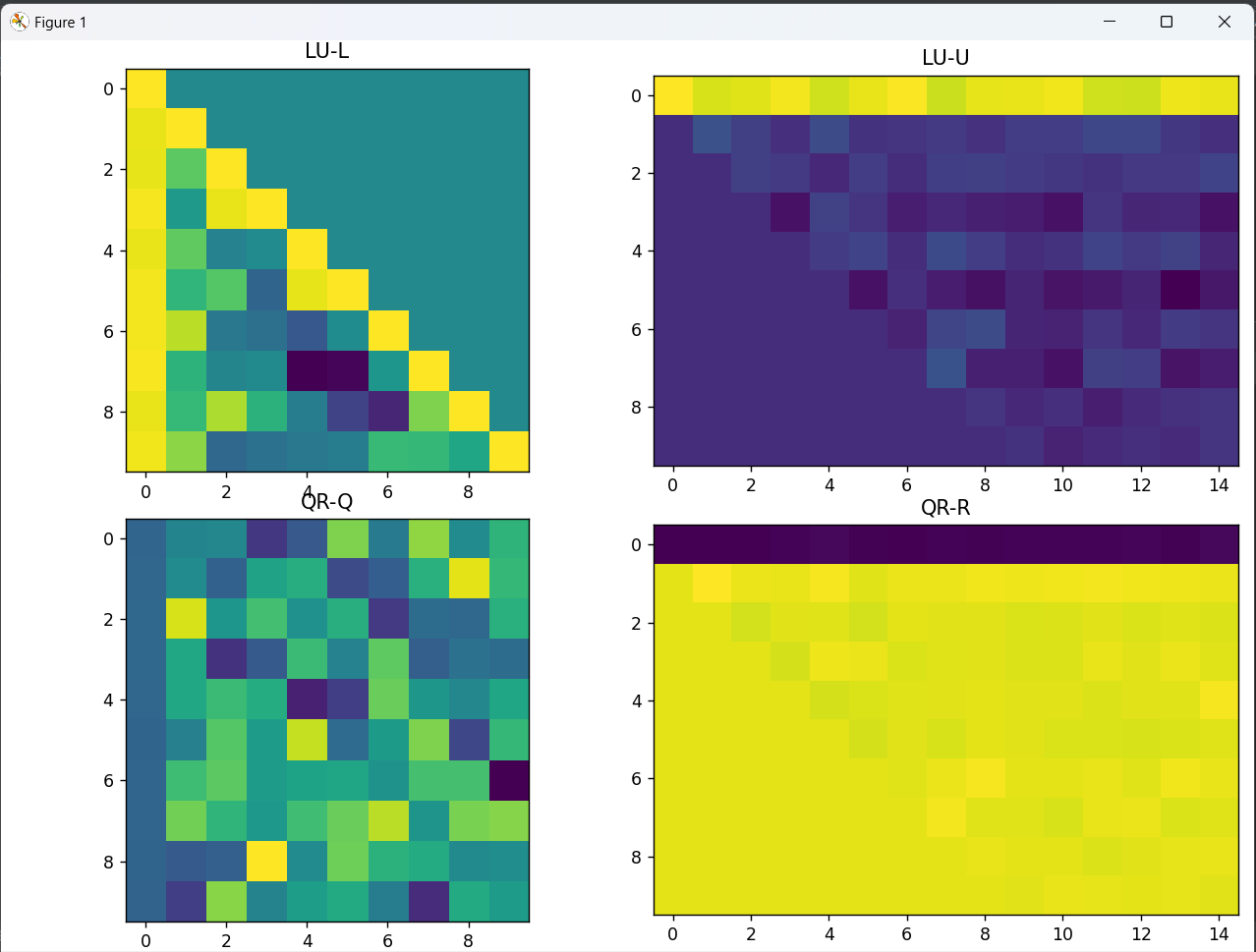
plt.tight\_layout()

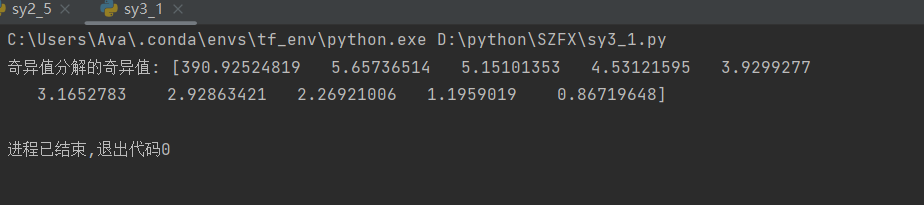
plt.show()

# 显示奇异值

print("奇异值分解的奇异值:", S)

结果如下：





2.程序如下：

from scipy.optimize import linprog

# 目标函数的系数

c = [-1, -0.8, -1.2]  # 因为我们要最小化 -X1 - 0.8X2 - 1.2X3

# 不等式约束的系数矩阵

A = [

    [1, -1, 1],

    [3, 2, 4],

    [3, 2, 0]

]

# 不等式约束的右侧常数

b = [30, 42, 30]

# 变量的上下界

x\_bounds = (0, None)

bounds = [x\_bounds, x\_bounds, x\_bounds]

# 求解优化问题

res = linprog(c, A\_ub=A, b\_ub=b, bounds=bounds, method='highs')

# 输出结果

print("最优解:", res.x)

print("最优目标函数值:", res.fun)

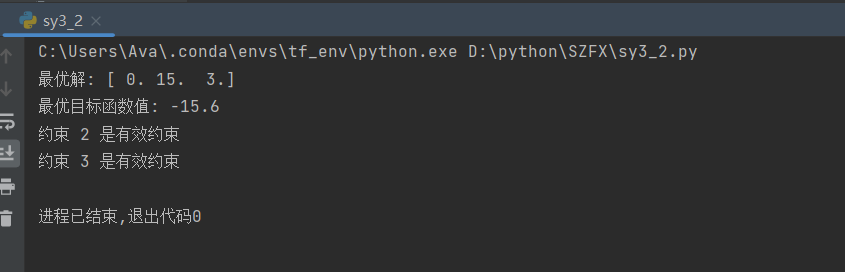
# 找到有效约束

for i, slack in enumerate(res.slack):

    if slack == 0:  # 如果松弛变量为0，说明该约束是有效的

        print(f"约束 {i + 1} 是有效约束")

结果如下：



3.程序如下：

import numpy as np

import matplotlib.pyplot as plt

# 过去12个月的花费数据（这里是示例数据，可以替换成你的实际数据）

months = np.arange(1, 13)  # 12个月

expenditure\_WeiXin = [3742.13, 2257.99, 4980.12, 1805.65, 2526.10, 2002.59, 3579.46, 1142.75, 2285.99, 1071.80, 719.48, 1462.10]

# 拟合一次、二次、三次曲线

coeffs\_1 = np.polyfit(months, expenditure\_WeiXin, 1)

coeffs\_2 = np.polyfit(months, expenditure\_WeiXin, 2)

coeffs\_3 = np.polyfit(months, expenditure\_WeiXin, 3)

# 创建拟合曲线

fit\_1 = np.polyval(coeffs\_1, months)

fit\_2 = np.polyval(coeffs\_2, months)

fit\_3 = np.polyval(coeffs\_3, months)

# 绘制数据和拟合曲线

plt.plot(months, expenditure\_WeiXin, 'bo', label='actual expend')  # 原始数据

plt.plot(months, fit\_1, 'r-', label='Fit the curve once')

plt.plot(months, fit\_2, 'g--', label='Quadratic fitting curves')

plt.plot(months, fit\_3, 'b-.', label='Cubic fitting curves')

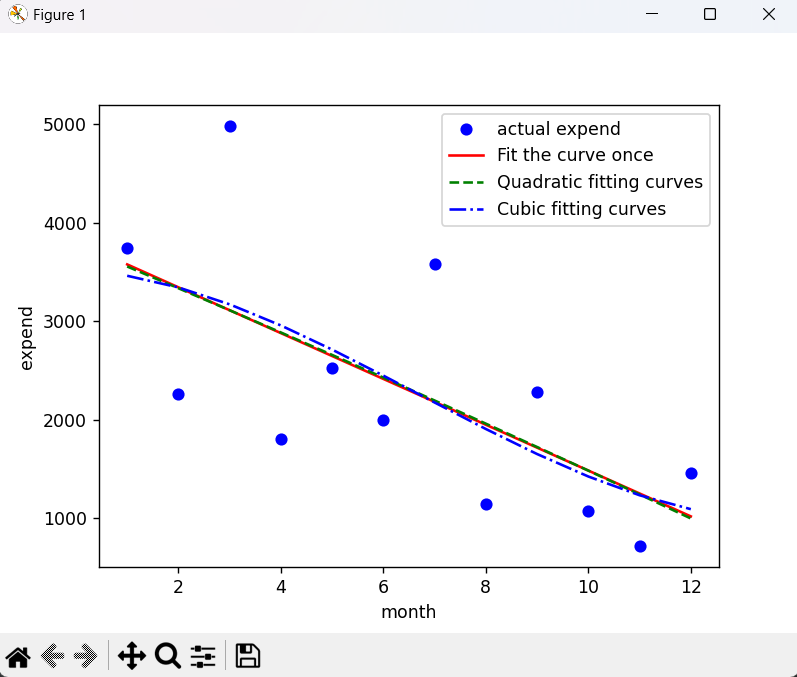
plt.xlabel('month')

plt.ylabel('expend')

plt.legend()

plt.show()

结果如下：



4.程序如下：

from scipy.stats import binom

# 设定参数

n = 50  # 射击次数

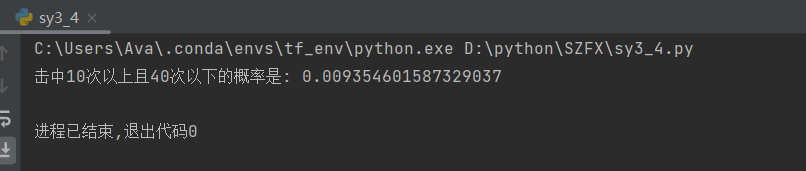
p = 0.1  # 每次射击的命中概率

# 计算击中10次以上且40次以下的概率

prob = binom.cdf(40, n, p) - binom.cdf(10, n, p)

print(f"击中10次以上且40次以下的概率是: {prob}")

结果如下：



5.程序如下：

import numpy as np

months = np.arange(1, 13)  # 12个月

# 过去12个月的花费数据

data = np.array([3742.13, 2257.99, 4980.12, 1805.65, 2526.10, 2002.59, 3579.46, 1142.75, 2285.99, 1071.80, 719.48, 1462.10])

# 最大似然估计：均值和方差

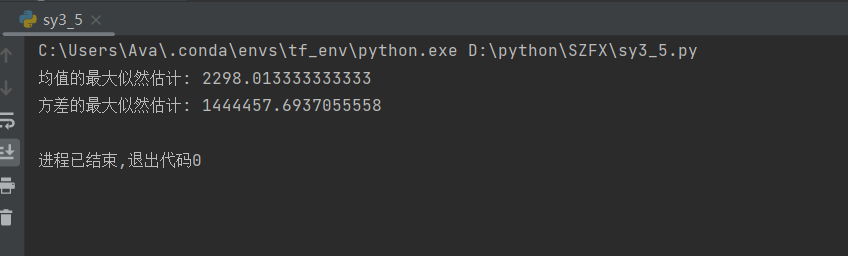
mu\_mle = np.mean(data)

sigma\_mle = np.std(data, ddof=0)

print(f"均值的最大似然估计: {mu\_mle}")

print(f"方差的最大似然估计: {sigma\_mle \*\* 2}")

结果如下：



**四、实验感想**

这次实验涉及了多个数学和计算机科学领域的内容，包括线性代数、优化问题、统计分析、概率论和数据拟合。通过实现这些任务，我加深了对相关知识的理解，并在实践中应用了许多数学模型和算法。以下是我对每个实验的感想：

#### 1. ****高斯矩阵分解：LU、QR、奇异值分解****

在这部分实验中，通过对高斯矩阵进行LU分解、QR分解和奇异值分解，我进一步理解了这些分解在矩阵计算中的重要性。LU分解和QR分解常用于解线性方程组，而奇异值分解（SVD）则在数据降维、特征提取等方面有着广泛应用。通过可视化这些分解结果，我可以直观地感受到矩阵的不同特性和分解的意义。尤其是奇异值分解，它能够揭示矩阵的秩和数据的最重要方向，这是在处理高维数据时非常有用的工具。

#### 2. ****优化问题求解****

优化问题的求解是运筹学和工业工程中常见的任务。通过使用 scipy.optimize.linprog 进行线性规划求解，我体验了如何将实际问题转化为数学模型，并利用现有工具求解最优解。通过观察约束条件的有效性，我了解到在实际应用中，不是所有的约束都会影响最终解，某些约束可能是冗余的。这个过程不仅让我更加理解线性规划的求解过程，也让我在实践中体会到数学建模的实际意义。

#### 3. ****数据拟合****

在统计分析部分，通过拟合一次、二次和三次曲线来拟合过去12个月的花费数据，我对数据拟合有了更深刻的理解。通过拟合不同阶数的多项式曲线，我们能够比较不同模型的拟合效果，并选择最适合的数据趋势。这一过程使我更加明白如何在现实生活中处理和预测数据，以及如何利用统计学工具进行模型选择和参数估计。

#### 4. ****射击问题的概率计算****

射击问题涉及到二项分布，这让我重新认识了概率论在实际问题中的应用。通过计算在50次射击中，命中次数在10次到40次之间的概率，我更深刻地理解了二项分布的性质以及如何通过概率分布来解决实际问题。这个任务让我意识到，概率论的工具可以帮助我们分析和预测各种随机事件的结果，具有广泛的实际应用价值。

#### 5. ****极大似然估计****

最后，通过对花费数据进行极大似然估计，我学到了如何根据数据推断出正态分布的参数（均值和方差）。这不仅加深了我对最大似然估计的理解，也让我体会到在实际应用中如何通过数据推断出潜在的分布模型。这种方法是统计学中的基本工具，广泛应用于金融、医学、工程等领域的数据分析。

### 总结

通过这次实验，我不仅复习了相关的数学知识，还实际操作了不同的算法和工具，进一步提升了我的编程能力和数学建模能力。在处理复杂问题时，能够灵活地运用不同的数学工具是非常重要的。每个任务都有其独特的数学背景和实际应用，使我更加清晰地认识到数学和计算机科学在现实世界中的重要性。未来，我希望能够在这些基础上深入学习，更好地应用这些工具解决实际问题。