

练习 3

1、问题描述：

现有 50 万元基金用于投资三种股票 A、B、C。A 每股年期望收益为 5 元（标准差 2 元），目前市价 20 元；B 每股年期望收益 8 元（标准差 6 元），目前市价 25 元；C 每股年期望收益为 10 元（标准差 10 元），目前市价 30 元；股票 A、B 收益的相关系数为 5/24，股票 A、C 收益的相关系数为 -0.5，股票 B、C 收益的相关系数为 -0.25。假设基金不一定要用完（不计利息或贬值），风险通常用收益的方差或标准差衡量。

- (1) 期望今年得到至少 20% 的投资回报，应如何投资？
- (2) 投资回报率与风险的关系如何？

2、建模

(1) 不同与例题给出的 12 年收益率样本数据，本次练习给出的是收益，单位是“元”；并且给出了相关系数；这使得我们需要按照相关系数公式：

$$r(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X)Var(Y)}}$$

去计算两只股票之间的协方差：

$$Cov(A,B) = 5/24 * 2 * 6 = 2.5$$

$$Cov(A,C) = -0.5 * 2 * 10 = -10$$

$$Cov(B,C) = -0.25 * 6 * 10 = -15$$

也是基于此，我们令 $X = (X_1, X_2, X_3)^T$ 分别表示购买 A、B、C 三种股票的份额数；矩阵

$Y = (20, 25, 30)$ 表示购买价格矩阵；矩阵 $Mu = (5, 8, 10)$ 表示平均每股收益；协方差矩阵为：

$$F = \begin{pmatrix} 4 & 2.5 & -10 \\ 2.5 & 36 & -15 \\ -10 & -15 & 100 \end{pmatrix}$$

依照题意可以构建出模型：

$$\begin{aligned} \min : & X^T F X \\ & \begin{cases} X \cdot Y \leq 500000 \\ Mu \cdot X \geq 100000 \\ X_i \geq 0 (i=1,2,3) \end{cases} \end{aligned}$$

结果:

投资: 367084.31016164186

收益: 99999.77164741016

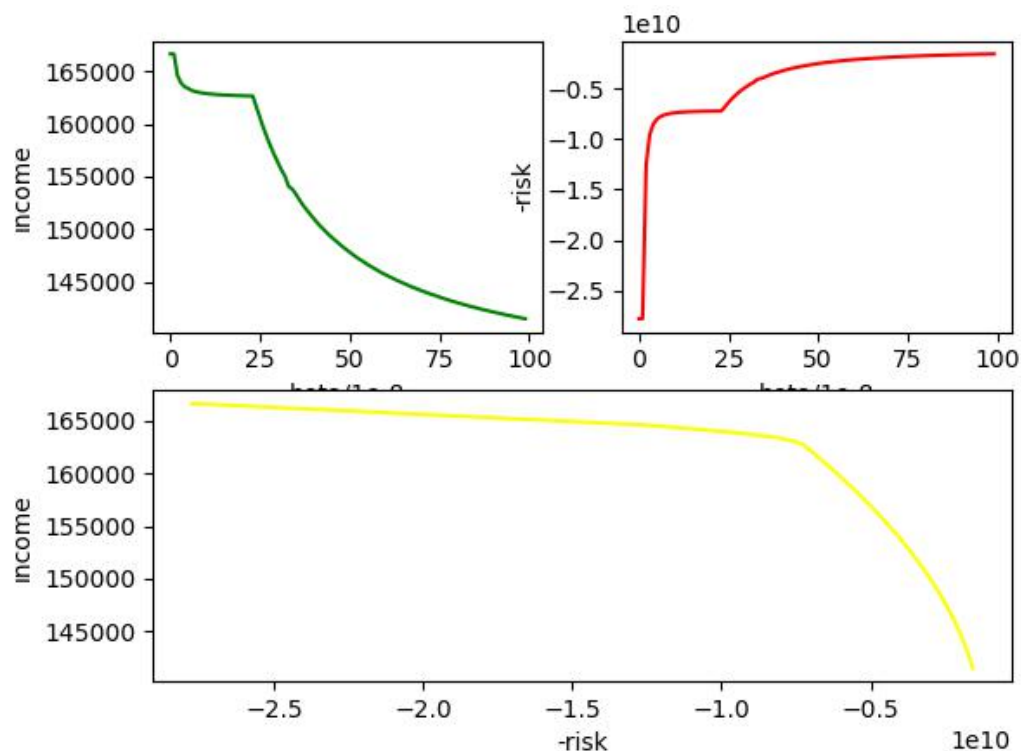
最优解: [13111.3038994 1528.47572243 2221.5446371]

(2) 对于问题 2, 引入变量 β : 表示投资者对风险的偏好程度; 考虑目标函数为:

$$\min Z = \beta Z_2 - Z_1 = \beta(4x_1^2 + 36x_2^2 + 100x_3^2 + 5x_1x_2 - 20x_1x_3 - 30x_2x_3) - (5x_1 + 8x_2 + 10x_3).$$

如果 $\beta = 0$ 表明投资者是冒险型, β 越大, 投资者越保守。

结果:



3、代码

```
from scipy.optimize import minimize
import numpy as np
import matplotlib.pyplot as plt
# 协方差矩阵
F = np.array([[4,5/2,-10],
              [5/2,36,-15],
              [-10,-15,100]])
m1 = np.array([20, 25, 30])
m2 = np.array([25,33,40])
m3 = np.array([5,8,10])
beta = 0
# 问题 1 优化目标函数
```

```

def func(x):
    return x @ F @ x.T

# 问题一求解
def investment():
    con1={'type': 'ineq', 'fun': lambda x: 500000-m1@x}
    con2={'type': 'ineq', 'fun': lambda x: m3@x-100000}
    b = (0, 500000)
    bound = (b, b, b)
    res = minimize(func, np.ones(3)+1e-5, constraints=[con1,con2],bounds=bound)
    print(res) #输出解的信息
    print('投资:',m1 @ res['x'])
    print("收益:",m3 @ res['x'])
    print('最优值:',func(res['x']))
    print('最优解: ',res['x'])

# 求解固定 beta 时问题二的目标函数最优值
def cov():
    con1 = {'type': 'ineq', 'fun': lambda x: 500000 - m1 @ x}
    b = (1,500000)
    bound = (b,b,b)
    res = minimize(func2, np.ones(3) + 1e-5, constraints=con1,bounds=bound)
    return m2 @ res['x'] - 500000,-res['x']@F@res['x']

# 问题二优化目标函数
def func2(x):
    return beta*(x @ F @ x.T)- (m2@x)

if __name__ == '__main__':
    investment()
    income_list = []
    risk_list = []
    # 画出关系图
    for i in np.arange(0,100,1):
        beta=i*0.0000001
        income,risk = cov()
        income_list.append(income)
        risk_list.append(risk)
    plt.subplot(221)
    x = np.arange(0,100,1)
    plt.plot(x,income_list,color='green')
    plt.xlabel('beta/1e-8')
    plt.ylabel('income')
    plt.subplot(222)
    plt.xlabel('beta/1e-8')

```

```
plt.ylabel('-risk')
plt.plot(x,risk_list,color='red')
plt.subplot(212)
plt.xlabel('-risk')
plt.ylabel('income')
plt.plot(risk_list,income_list,color='yellow')
plt.show()
```