

数学实验：第六次作业

计算机系 计 73 2017011620 李家昊

2020 年 4 月 17 日

1 实验目的

- 掌握用 MATLAB 优化工具箱和 LINGO 解线性规划的方法。
- 练习建立实际问题的线性规划模型。
- 掌握用 MATLAB 优化工具箱和 LINGO 解非线性规划的方法。
- 练习建立实际问题的非线性规划模型。

2 问题求解

2.1 Chap8-Ex10 污水处理（应用题）

2.1.1 问题分析

题目给出江边的工厂、污水处理站和居民点的地理位置分布，需要根据工厂污水的流量和水污染浓度、处理站的处理系数、江水的原流量和水污染浓度、各段江水的自净系数等参数，求出在江水污染达到国家标准的情况下，污水处理站的最小总处理费用，这是一个线性规划问题。

2.1.2 模型假设

为了简化实际情况，模型基于以下假设，

1. 工厂是江水的唯一污染源。
2. 污水处理站处理前后，水流量不变。
3. 污水处理站能够将污水净化到所需的任意程度。
4. 污水与江水能在极短时间内混合均匀。

2.1.3 模型建立

设江边共有 n 个工厂，第 1 个工厂位于最上游，第 n 个工厂位于最下游，每个工厂的排污口处设有一个污水处理站，处理站的对岸是一个居民区。对于第 i ($i = 1, 2, \dots, n$) 个工厂，记其污水流量为 A_i ，污水浓度为 a_i ，处理站的处理系数为 c_i ，降低的污水浓度为 x_i ，处理站上游的污水流量为 U_i ，浓度为 u_i ，下游的污水流量为 D_i ，浓度为 d_i ，第 i 个到第 $i+1$ 个工厂之间的江水自净系数为 s_i ，国家标准规定的水污染浓度为 r 。其中 A_i, a_i, c_i, s_i ($i = 1, 2, \dots, n$) 和 U_1, u_1, r 为已知常数。

需要优化的目标函数为总处理费用，即，

$$\min f = \sum_{i=1}^n c_i A_i x_i \quad (1)$$

决策变量为处理站降低的污水浓度 x_1, x_2, \dots, x_n ，需要满足取值范围约束，

$$0 \leq x_i \leq a_i, \quad i = 1, 2, \dots, n \quad (2)$$

第 1 个工厂的上游水流量 U_1 和污水浓度 u_1 为已知，其余各工厂的上游水流量 U_i 和污水浓度 u_i 满足，

$$U_i = D_{i-1}, \quad u_i = s_{i-1} d_{i-1}, \quad i = 2, 3, \dots, n \quad (3)$$

各工厂的下游水流量 D_i 和污水浓度 d_i 满足，

$$D_i = U_i + A_i, \quad d_i = \frac{U_i u_i + A_i (a_i - x_i)}{U_i + A_i}, \quad i = 1, 2, \dots, n \quad (4)$$

对于第 (1) 问，需要使江面上所有地段的水污染达到国家标准，则有，

$$u_i \leq r, \quad d_i \leq r, \quad i = 1, 2, \dots, n \quad (5)$$

对于第 (2) 问，只要求居民点上游的水污染达到国家标准，则有，

$$u_i \leq r, \quad i = 1, 2, \dots, n \quad (6)$$

这是一个有约束优化问题，目标函数为方程 (1)，决策变量为 x_1, x_2, \dots, x_n ，第 (1) 问约束条件为方程 (2) 和方程 (5)，第 (2) 问约束条件为方程 (2) 和方程 (6)。由于目标函数和约束条件均为关于决策变量的线性表示，因此该模型也是一个线性规划模型。

2.1.4 算法设计

根据上述分析可知，这是一个线性规划问题，可利用单纯形法求解，对应的 MATLAB 命令为 `linprog`，也可利用 LINGO 软件求解。

2.1.5 MATLAB 和 LINGO 程序

编写了 MATLAB 版本和 LINGO 版本的程序，请参见附录4.1。

2.1.6 计算结果

经过实验，MATLAB 与 LINGO 的计算结果完全一致，由于 LINGO 的分析功能更加强大，因此这里主要叙述 LINGO 的计算结果。

第 (1) 问 令江面上所有地段的水污染均达到国家标准，LINGO 经过 1 次迭代计算得出，总处理费用 f 的最小值为 489.50 万元，各污水处理站净化的污水浓度 x_1, x_2, x_3 的计算结果如表 1所示，江面各处的水质约束情况如表 2所示。

表 1: 第 (1) 问各决策变量的最优值和减少费用

决策变量	x_1	x_2	x_3
最优值	59.00	38.90	0.00
减少费用	0.00	0.00	5.00

表 2: 第 (1) 问各水质约束的松弛变量和对偶价格

约束条件	$u_1 \leq r$	$u_2 \leq r$	$u_3 \leq r$	$d_1 \leq r$	$d_2 \leq r$	$d_3 \leq r$
松弛变量	0.20	0.10	0.40	0.00	0.00	0.16
对偶价格	0.00	0.00	0.00	100.50	1010.00	0.00

第 (2) 问 只要求居民点上游的水污染达到国家标准，LINGO 经过 0 次迭代计算得出，总处理费用 f 的最小值为 183.33 万元，各污水处理站净化的污水浓度 x_1, x_2, x_3 的计算结果如表 3所示，江面各处的水质约束情况如表 4所示。

表 3: 第 (2) 问各决策变量的最优值和减少费用

决策变量	x_1	x_2	x_3
最优值	36.67	0.00	0.00
减少费用	0.00	5.00	5.00

表 4: 第 (2) 问各水质约束的松弛变量和对偶价格

约束条件	$u_1 \leq r$	$u_2 \leq r$	$u_3 \leq r$
松弛变量	0.20	0.00	0.22
对偶价格	0.00	1116.67	0.00

2.1.7 结果的数学分析

在第 (1) 问中, x_1, x_2 的减少费用为零, 为基变量, 约束 $d_1 \leq r$ 和 $d_2 \leq r$ 的松弛变量为零, 起约束作用。

在第 (2) 问中, x_1 的减少费用为零, 为基变量, 约束 $u_2 \leq r$ 的松弛变量为零, 起约束作用。由于此时去掉了第 (1) 问中两个起作用的约束 $d_1 \leq r$ 和 $d_2 \leq r$, 因此最优值比第 (1) 问更优。若想进一步优化最优值, 可以提高工厂 1 到 2 之间的江水净化能力, 即降低 s_1 , 使得约束 $u_2 \leq r$ 更容易被满足。

2.1.8 结果的实际意义

由计算结果可以看出, 仅要求居民点上游的水质满足国家标准时, 所花费的处理费用显著减少, 原因是该方案充分利用了江水的自净机制, 减轻了污水处理厂的处理压力。在实际的城市规划中也有类似的现象, 例如自来水厂通常建在河流的上游, 而工厂通常建在下游, 使得居民能够用到未污染的水, 且河流能充分发挥自净作用。

该计算结果具有一定的实际意义, 可作为制定污水处理方案的参考。在实际情况下, 还需考虑更多的因素, 例如工厂污水排放对河流自净能力的破坏, 居民点生活污水的排放对河流的污染, 居民取水对水流量的影响等等。在该污水处理方案下, 实际的水质是否符合国家标准, 还需要通过实验来验证。

2.1.9 结论

为了使江面上所有地段的水污染达到国家标准, 最少需要花费 489.50 万元; 如果只要求三个居民点上游的水污染达到国家标准, 最少需要花费 183.33 万元。

2.2 Chap9-Ex4 液体混合 (应用题)

2.2.1 问题分析

题目设置了一个生产场景, 给定所需原料的价格、成分和最大供应量, 最终产品的生产流程、成分指标和最大需求量, 需要确定最优的生产方案。一般来说, 生产以利润为目的, 因此只需在给定约束条件下确定最大利润即可, 这是一个非线性规划问题。

2.2.2 模型假设

1. 两种原料混合后，不生物料损失，且硫的总含量不变。
2. 混合物不易分解，能够稳定存在。
3. 该生产过程中，原料采购是唯一成本来源，产品销售是唯一利润来源。

2.2.3 模型建立

第 (1) 问 设公司购买甲原料 x 吨，乙原料 y 吨，丙原料 z 吨，甲乙混合后浓度为 $w\%$ ，甲乙混合物中有 p 吨用于制造 A 产品，有 q 吨用于制造 B 产品，丙原料中有 r 吨用于制造 A 产品，有 s 吨用于制造 B 产品，则 A 产品的产量为 $p + r$ 吨，B 产品的产量为 $q + s$ 吨。

则原料采购量满足供应量约束，

$$0 \leq x \leq 500, \quad 0 \leq y \leq 500, \quad 0 \leq z \leq 500 \quad (7)$$

甲乙混合物满足浓度约束，

$$w = \frac{3x + y}{x + y} \quad (8)$$

甲乙混合物满足数量约束，

$$p \geq 0, \quad q \geq 0, \quad p + q \leq x + y \quad (9)$$

丙原料满足数量约束，

$$r \geq 0, \quad s \geq 0, \quad r + s \leq z \quad (10)$$

A, B 产品需要满足含硫量约束，

$$\frac{wp + 2r}{p + r} \leq 2.5, \quad \frac{wq + 2s}{q + s} \leq 1.5 \quad (11)$$

A, B 产品需要满足需求量约束，

$$p + r \leq 100, \quad q + s \leq 200 \quad (12)$$

利润最高的生产策略是最优的，因此需要最大化生产利润 f ，

$$\max f = 9(p + r) + 15(q + s) - 6x - 16y - 10z \quad (13)$$

综上，对于第 (1) 问，目标函数为方程 (13)，决策变量为 x, y, z, w, p, q, r, s ，约束条件为方程 (7)，方程 (8)，方程 (9)，方程 (10)，方程 (11)，以及方程 (12)。由于方程 (8) 和方程 (11) 为非线性约束，因此这是一个非线性规划模型。

第 (2) 问 在第 (1) 问基础上, 只需将方程 (12) 的需求量约束修改为,

$$p + r \leq 600, \quad q + s \leq 200 \quad (14)$$

第 (3) 问 在第 (1),(2) 问基础上, 只需将方程 (13) 的目标函数修改为,

$$\max f = 9(p + r) + 15(q + s) - 6x - 13y - 10z \quad (15)$$

2.2.4 算法设计

这是一个非线性规划问题, 可采用逐步二次规划法 (SQP) 求解, 对应的 MATLAB 命令为 `fmincon`, 但 MATLAB 只能求出局部最优解, 不同的初值条件下求出的局部最优解不同, 因此需要对初值进行搜索, 才能大致确定全局最优解; 也可以采用 LINGO 软件求解, 只需开启全局优化选项, 就可以求得全局最优解。

2.2.5 MATLAB 和 LINGO 程序

编写了 MATLAB 版本和 LINGO 版本的程序, 请参见附录4.2。

2.2.6 计算结果

第 (1) 问 LINGO 经过 3997 次迭代, 找到了全局最优解, 利润 f 的最大值为 400 千元, 各决策变量的最优值为,

$$x = 0, \quad y = 100, \quad z = 100, \quad w = 1, \quad p = 0, \quad q = 100, \quad r = 0, \quad s = 100 \quad (16)$$

起作用的约束为甲乙混合物的浓度约束8和数量约束9, 丙原料的数量约束10, A,B 产品的含硫量约束11, 以及 B 产品的需求量约束12。

用 MATLAB 求解时, 当初值 $y_0 = 100$, 其余变量初值均为零时, 得到的结果与 LINGO 全局最优解相同。

第 (2) 问 当产品 A 的最大市场需求量增长为 600 吨时, LINGO 经过 2393 次迭代, 找到了全局最优解, 利润 f 的最大值为 600 千元, 各决策变量的最优值为,

$$x = 300, \quad y = 0, \quad z = 300, \quad w = 3, \quad p = 300, \quad q = 0, \quad r = 300, \quad s = 0 \quad (17)$$

起作用的约束为甲乙混合物的浓度约束8和数量约束9, 丙原料的数量约束10, A,B 产品的含硫量约束11, 以及 A 产品的需求量约束12。

用 MATLAB 求解时, 当初值 $x_0 = 300$, 其余变量初值均为零时, 得到的结果与 LINGO 全局最优解相同。

第 (3) 问 当乙的进货价格下降为 13 千元/吨时, 考虑第 (1) 问的情况, LINGO 经过 2036 次迭代, 找到了全局最优解, 利润 f 的最大值为 750 千元, 各决策变量的最优值如下, 起作用的约束同第 (1) 问。用 MATLAB 求解时, 当初值 $y_0 = 150$, 其余变量初值均为零时, 得到的结果与 LINGO 全局最优解相同。

$$x = 50, y = 150, z = 0, w = 1.5, p = 0, q = 200, r = 0, s = 0 \quad (18)$$

考虑第 (2) 问的情况, LINGO 经过 3099 次迭代, 找到了全局最优解, 利润 f 的最大值为 750 千元, 各决策变量的最优值如下, 起作用的约束同第 (1) 问。用 MATLAB 求解时, 当初值 $y_0 = 150$, 其余变量初值均为零时, 得到的结果与 LINGO 全局最优解相同。

$$x = 50, y = 150, z = 0, w = 1.5, p = 0, q = 200, r = 0, s = 0 \quad (19)$$

此外, 在第 (2) 问的情况下, 当初值化 x 为 450 时, LINGO 经过 4594 次迭代, 找到了另一个全局最优解, 利润 f 的最大值也为 750 千元, 各决策变量的最优值如下, 起作用的约束同第 (2) 问。用 MATLAB 求解时, 当初值 $x_0 = 450$, 其余变量初值均为零时, 得到的结果与 LINGO 全局最优解相同。

$$x = 450, y = 150, z = 0, w = 2.5, p = 600, q = 0, r = 0, s = 0 \quad (20)$$

2.2.7 结果的数学分析

用 MATLAB 求解时, 只能找到局部极小值, 不同的初值条件下的求解结果不同, 一般来说, 当初值越接近全局最优解, 则越容易求得全局最优解。

用 LINGO 求解时, 当目标函数有多个全局最优值时, 在不同的初值条件下, 可能找到不同的全局最优值, 它们对应的目标函数值相等, 但决策变量值不同。

2.2.8 结果的现实意义

该计算结果具有一定的现实意义, 可作为制定实际生产方案的重要参考。然而, 该模型仍相对简单, 在实际应用中, 还需要考虑仓储物流、人力资源、价格波动等因素, 才能制定出合理的生产方案。

2.2.9 结论

在默认情况下, 应当分别采购乙原料和丙原料 100 吨, 不采购甲原料, 所有原料全部用于生产 B 产品, 此时利润最大, 为 40 万元。

如果产品 A 的最大市场需求量增长为 600 吨, 则应当分别采购甲原料和丙原料 300 吨, 不采购乙原料, 所有原料全部用于生产 A 产品, 此时利润最大, 为 60 万元。

如果乙的进货价格下降为 13 千元/吨, 则应当采购甲原料 50 吨, 乙原料 150 吨, 不采购丙原料, 所有原料全部用于生产 B 产品, 此时利润最大, 为 75 万元。在此基础上, 如果产品 A 的最大市场需求量增长为 600 吨, 则有两种生产方案: 方案一为保持原有生产方案不变, 此时利润最大为 75 万元; 方案二为采购甲原料 450 吨, 乙原料 150 吨, 不采购丙原料, 所有原料全部用于生产 A 产品, 此时利润最大也为 75 万元。

2.3 Chap9-Ex8 股票投资 (应用题)

2.3.1 问题分析

题目给出了三支股票的历史收益数据, 在达到目标年收益率的前提下, 需要确定最优投资策略。根据投资组合模型可知, 这是一个二次规划问题。

2.3.2 模型假设

为了简化实际情况, 模型基于以下假设,

1. 投资收益可用股票收益的期望衡量, 投资风险可用股票收益的方差衡量。
2. 除了股票和国库券, 没有其他投资方式。
3. 政治经济文化环境相对稳定, 股票市场也相对稳定。

2.3.3 模型建立

第 (0) 问 这是一个典型的投资组合模型, 记股票 A,B,C 的投资比例为 $\mathbf{x} = (x_1, x_2, x_3)^T$, 年收益率为随机变量 $\mathbf{S} = (S_1, S_2, S_3)^T$, 则投资人资金的年收益率为 $R = \mathbf{S}^T \mathbf{x}$, 设投资人需要达到的目标年收益率为 r 。

投资人期望的收益可用 R 的期望衡量, 即,

$$ER = (E\mathbf{S})^T \mathbf{x} \quad (21)$$

投资人面临的风险可用 R 的方差衡量, 记 \mathbf{S} 的协方差矩阵为 Σ , 则有,

$$DR = \mathbf{x}^T \Sigma \mathbf{x} \quad (22)$$

对投资比例有范围约束, 这里假设投资人不一定把所有的资金投入股市, 未投资的资金闲置不用, 则有,

$$\sum_{i=1}^3 x_i \leq 1, \quad x_i \geq 0, \quad i = 1, 2, 3 \quad (23)$$

目标年收益率约束为,

$$ER \geq r \quad (24)$$

在上述约束下，需要最小化风险 f ，

$$\min f = DR \quad (25)$$

这是一个二次规划模型，目标函数为方程 (25)，决策变量为 \mathbf{x} ，约束条件为方程 (23) 和方程 (24)。

第 (1) 问 在第 (0) 问的基础上，令目标年收益率 r 在 10%-100% 之间取值，对于每一个固定的 r ，求解第 (0) 问模型即可。需要注意的是，若投资人的目标年收益率超过了股票的最高期望年收益率，问题将无解。

第 (2) 问 在第 (0) 问的基础上，设购买国库券的资金比例为 x_4 ，国库券的期望收益为 S_4 ，则有 $\mathbf{x} = (x_1, x_2, x_3, x_4)^T$ ， $\mathbf{S} = (S_1, S_2, S_3, S_4)^T$ ，需要修改取值范围约束为，

$$\sum_{i=1}^4 x_i \leq 1, \quad x_i \geq 0, \quad i = 1, 2, 3, 4 \quad (26)$$

对于第 (2) 问，目标函数为方程 (25)，决策变量为 \mathbf{x} ，约束条件为方程 (26) 和方程 (24)。

第 (3) 问 在第 (0) 问基础上，设当前持有股票 A,B,C 占总资金比例为 $\mathbf{c} = (c_1, c_2, c_3)^T$ ，交易费按交易额的 α 倍收取，应当买入股票 A,B,C 的数量为 $\mathbf{y} = (y_1, y_2, y_3)^T$ ，应当卖出的数量分别为 $\mathbf{z} = (z_1, z_2, z_3)^T$ ，则需要修改范围约束为，

$$\sum_{i=1}^3 (x_i + \alpha y_i + \alpha z_i) \leq 1, \quad x_i, y_i, z_i \geq 0, \quad i = 1, 2, 3 \quad (27)$$

修改收益约束为，

$$ER - \alpha \sum_{i=1}^3 (y_i + z_i) \geq r \quad (28)$$

且增加交易数量约束，

$$x_i = c_i + y_i - z_i, \quad i = 1, 2, 3 \quad (29)$$

对于第 (3) 问，目标函数为方程 (25)，决策变量为 $\mathbf{x}, \mathbf{y}, \mathbf{z}$ ，约束条件为方程 (27)，方程 (28) 和方程 (29)。

2.3.4 算法设计

对于二次规划问题，可以通过 MATLAB 的 `quadprog` 求解，使用的方法是凸规划的内点法，也可以通过 LINGO 软件求解。其中，题目给出了过去 12 年内股票的年末与年初价值之比，注意年末价值包括了股票的收益和本金，这些历史数据可用于估算股票年收益率的期望和协方差矩阵。

2.3.5 MATLAB 和 LINGO 程序

编写了 MATLAB 版本和 LINGO 版本的程序，请参见附录4.3。

2.3.6 计算结果

通过股票的历史数据，计算出 S 的期望和协方差矩阵为，

$$ES = \begin{pmatrix} 0.0891 \\ 0.2137 \\ 0.2346 \end{pmatrix}, \quad \Sigma = \begin{pmatrix} 0.0108 & 0.0124 & 0.0131 \\ 0.0124 & 0.0584 & 0.0554 \\ 0.0131 & 0.0554 & 0.0942 \end{pmatrix} \quad (30)$$

第 (0) 问 当目标年收益率为 15% 时，用 LINGO 求解时，经过 82 次迭代，找到了全局最优解，风险 f 的最小值为 0.0224，各决策变量的最优值如下，所有约束条件 (23, 24) 均起约束作用。

$$\mathbf{x} = (0.5302, 0.3565, 0.1133)^T \quad (31)$$

用 MATLAB 求解时，也得到了相同的结果。

第 (1) 问 当目标年收益率在 10%-100% 变化时，由股票年收益率的期望向量 ES 可以看出，股票 C 的期望收益率最高，为 23.46%，因此，投资人的目标年收益率不能超过 23.46%，否则问题无解。当目标年收益率在 10%-23.46% 之间变化时，用 MATLAB 求解得到，三支股票的投资比例以及总投资比例的变化情况如图 1，投资人面临风险的变化情况如图 2所示。

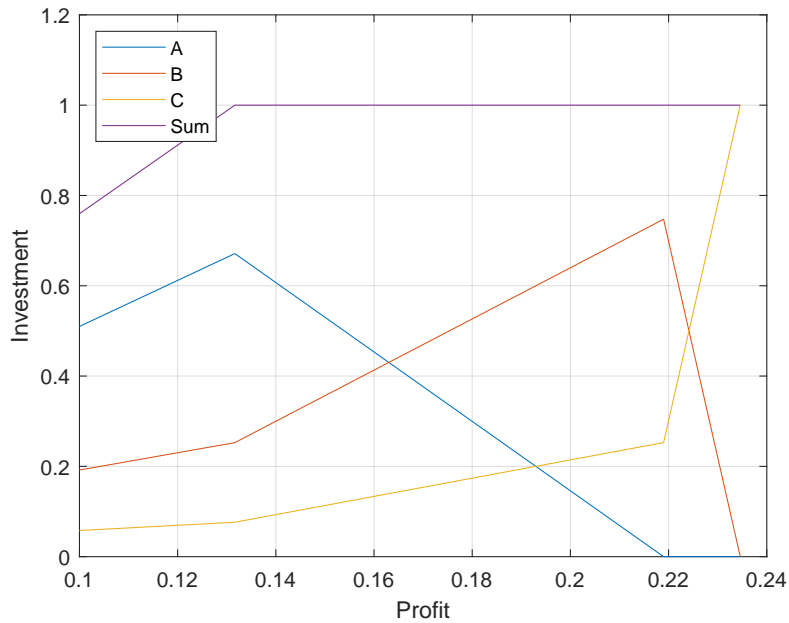


图 1: 各股投资金额及投资总额占总资金比例随目标收益率的变化曲线

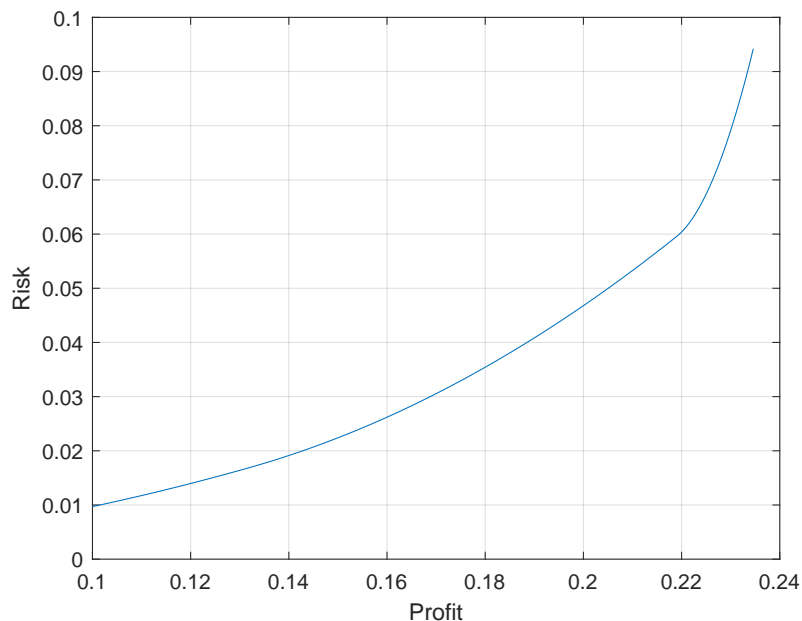


图 2: 投资风险随目标收益率的变化曲线

第 (2) 问 假设可以购买年收益率为 5% 的无风险国库券, 用 LINGO 求解时, 经过 94 次迭代, 找到了全局最优解, 风险 f 的最小值为 0.0208, 各决策变量的最优值如下, 所有约束条件 (26, 24) 均起约束作用。

$$\mathbf{x} = (0.0870, 0.4282, 0.1435, 0.3412)^T \quad (32)$$

用 MATLAB 求解时, 也得到了相同的结果。

第 (3) 问 假设目前持有的股票比例为: 股票 A 占 50%, B 占 35%, C 占 15%, 每次股票买卖的交易费为交易额 1%, 用 LINGO 求解时, 经过 421 次迭代, 找到了全局最优解, 风险 f 的最小值为 0.0226, 各决策变量的最优值如下, 所有约束条件 (27, 28, 29) 均起约束作用。

$$\mathbf{x} = \begin{pmatrix} 0.5266 \\ 0.3500 \\ 0.1229 \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} 0.0266 \\ 0.0000 \\ 0.0000 \end{pmatrix}, \quad \mathbf{z} = \begin{pmatrix} 0.0000 \\ 0.0000 \\ 0.0271 \end{pmatrix} \quad (33)$$

2.3.7 结果的数学分析

由图 2 可以看出, 当期望收益越高, 相应的风险也越高, 体现了高风险高回报的市场规律。由图 1 可以看出, 当目标收益率较低时, 投资者并未将所有的资金投入股市, 并倾向于投资低风险低收益的股票 A, 当目标收益率较高时, 投资者将所有资金投入股市, 并倾向于投资高风险高收益的股票 C。

投资的目标是以最小的风险赚取最大的收益，这是一个多目标优化问题，本题中固定了目标收益率，求最小化风险，实际上也可以在固定的风险承受范围内，求最大的收益，抑或是引入一个平衡因子 β 来统一优化收益和风险，即，

$$\min f = \beta DR - ER \quad (34)$$

其中 β 越大，表明投资人越厌恶风险。

2.3.8 结果的实际意义

在稳定的股票市场中，投资组合模型可作为制定投资策略的重要参考。然而，其局限性也不可忽视，仅仅依靠股票的历史收益序列来估算未来的收益和风险是远远不够的，例如在突如其来的新冠疫情面前，或者瑞幸咖啡的虚假财报面前，该模型将完全失效。

事实上，股票市场往往受到政治经济文化等多方面的影响，除了考虑历史收益，还要考虑公司的运营状况，产业的发展情况，是否有重大利好或利空消息，大盘的走势，广大投资者的心理预期等等，这样才能制定出合适的投资策略。

2.3.9 结论

在默认情况下，目标年收益率为 15%，应当以 53.02% 的资金投资股票 A，35.65 的资金投资股票 B，11.33% 的资金投资股票 C，此时风险最小，为 0.0224。

当目标年收益率在 10%-23.46% 之间变化时，投资组合的变化如图 1 所示，相应风险的变化如图 2 所示，当目标年收益率在 23.47%-100% 之间变化时，没有合适的投资组合。

当可以购买年收益率为 5% 的无风险国库券时，应当以 8.70% 的资金投资股票 A，42.82% 的资金投资股票 B，14.35% 的资金投资股票 C，34.12% 的资金投资国库券，此时风险最小，为 0.0208。

当目前持有的股票比例为：股票 A 占 50%，B 占 35%，C 占 15%，每次股票买卖的交易费为交易额的 1% 时，应当以 2.66% 的资金买入股票 A，以 2.71% 的资金卖出股票 C，持有比例转变为：股票 A 占 52.66%，B 占 35.00%，C 占 12.29%，此时风险最小，为 0.0226。

3 收获与建议

在本次实验中，我掌握了 MATLAB 优化工具箱和 LINGO 软件求解线性规划和非线性规划的基本方法，用线性规划和非线性规划方法建立了实际问题的模型，并进行求解，在解决实际问题的过程中，我对数学方法的原理和应用有了更深刻的理解。

希望助教能对每次的实验进行详细的解答，希望老师在未来的课堂上介绍更多数学应用的前沿知识。

4 附录：程序代码

4.1 Chap8-Ex10

MATLAB 程序如下，

```
1 U = [1000 1005 1010]';
2 D = [1005 1010 1015]';
3 U1 = U(1);
4 u1 = 0.8;
5 A = [5 5 5]';
6 a = [100 60 50]';
7 c = [1 1 1]';
8 s = [0.9 0.6]';
9 r = 1;
10
11 % x = [x1, x2, x3, u2, u3, d1, d2, d3]'
12 f = [c.*A; zeros(5,1)];
13 B = [];
14 b = [];
15 Beq = [0      0      0      1      0      -s(1) 0      0;
16        0      0      0      0      1      0      -s(2) 0;
17        A(1) 0      0      0      0      D(1) 0      0;
18        0      A(2) 0      -U(2) 0      0      D(2) 0;
19        0      0      A(3) 0      -U(3) 0      0      D(3)];
20 beq = [0; 0; U1*u1 + A(1)*a(1); A(2)*a(2); A(3)*a(3)];
21 lb = zeros(8,1);
22 ub = [a; r * ones(5,1)];
23 options = [];
24 [x,fval,exitflag,output] = linprog(f, B, b, Beq, beq, lb, ub, options
    );
```

LINGO 程序如下，

```
1 sets:
2 n3/1..3/: aa, a, c, uu, u, dd, d, x;
3 n2/1..2/: s;
4 endsets
5
6 data:
7 uu = 1000, , ;
8 u = 0.8, , ;
```

```

9 aa = 5 5 5;
10 a = 100 60 50;
11 c = 1 1 1;
12 s = 0.9 0.6;
13 r = 1;
14 enddata
15
16 [obj] min = @sum(n3: c * aa * x);
17
18 @for(n3: [range] x <= a);
19 @for(n2(i): [up_flow] uu(i+1) = dd(i));
20 @for(n2(i): [up_percent] u(i+1) = s(i) * d(i));
21 @for(n3: [down_flow] dd = uu + aa);
22 @for(n3: [down_percent] d = (uu*u + aa*(a-x)) / dd);
23
24 @for(n3: [up_std] u <= r);
25 @for(n3: [down_std] d <= r);

```

4.2 Chap9-Ex4

MATLAB 程序如下,

```

1 % X = [x, y, z, w, p, q, r, s]'
2 X0 = [450 0 0 0 0 0 0 0]';
3 A = [-1 -1 0 0 1 1 0 0;
4       0 0 -1 0 0 0 1 1;
5       0 0 0 0 1 0 1 0;
6       0 0 0 0 0 1 0 1];
7 b = [0 0 600 200]';
8 lb = [0 0 0 0 0 0 0 0]';
9 ub = [500 500 500 inf inf inf inf inf]';
10 options = optimoptions('fmincon', 'Algorithm', 'sqp');
11 [X,fval,exitflag,output] = fmincon(@fun,X0,A,b,[],[],lb,ub,@nonlincon
    ,options);
12
13 function [c,ceq] = nonlincon(X)
14     [x,y,z,w,p,q,r,s] = decompose(X);
15     ceq(1) = w * (x + y) - (3*x + y);
16     c(1) = w*p + 2*r - 2.5 * (p + r);
17     c(2) = w*q + 2*s - 1.5 * (q + s);
18 end
19
20 function [x,y,z,w,p,q,r,s] = decompose(X)
21     x = X(1); y = X(2); z = X(3); w = X(4);
22     p = X(5); q = X(6); r = X(7); s = X(8);

```

```

23 end
24
25 function z = fun(X)
26     [x,y,z,w,p,q,r,s] = decompose(X);
27     profit = 9*(p+r) + 15*(q+s) - 6*x - 13*y - 10*z;
28     z = -profit;
29 end

```

LINGO 程序如下，

```

1  [obj] max = 9*(p+r) + 15*(q+s) - 6*x - 16*y - 10*z;
2  [supply_x] x <= 500;
3  [supply_y] y <= 500;
4  [supply_z] z <= 500;
5  [mixed_xy] w = (3*x+y) / (x+y);
6  [total_mixed] p + q <= x + y;
7  [total_z] r + s <= z;
8  [sulphur_a] (w*p + 2*r) <= 2.5 * (p+r);
9  [sulphur_b] (w*q + 2*s) <= 1.5 * (q+s);
10 [demand_a] p + r <= 100;
11 [demand_b] q + s <= 200;

```

4.3 Chap9-Ex8

MATLAB 程序如下，

```

1  data = [
2      1943 1.300 1.225 1.149 1.258997;
3      1944 1.103 1.290 1.260 1.197526;
4      1945 1.216 1.216 1.419 1.364361;
5      1946 0.954 0.728 0.922 0.919287;
6      1947 0.929 1.144 1.169 1.057080;
7      1948 1.056 1.107 0.965 1.055012;
8      1949 1.038 1.321 1.133 1.187925;
9      1950 1.089 1.305 1.732 1.317130;
10     1951 1.090 1.195 1.021 1.240164;
11     1952 1.083 1.390 1.131 1.183675;
12     1953 1.035 0.928 1.006 0.990108;
13     1954 1.176 1.715 1.908 1.526236;
14 ];
15
16 data = data(:, 2:4);
17
18 ES = mean(data - 1);
19 DS = cov(data);

```

```

20 ES = round(ES, 4);
21 DS = round(DS, 4);
22
23 r = 0.15;
24 H = 2 * DS;
25 f = [];
26 A = [1 1 1;
27      -ES];
28 Aeq = [];
29 beq = [];
30 b = [1; -r];
31 lb = [0 0 0];
32 ub = [1 1 1];
33 x0 = [0.3 0.3 0.3];
34 options = [];
35
36 [x,fval,exitflag,output,lambda] = ...
37     quadprog(H,f,A,b,Aeq,beq,lb,ub,x0,options);

```

第 (0) 问和第 (1) 问的 LINGO 程序如下,

```

1 sets:
2 stock/1..3/: x, ES;
3 cov(stock, stock): DS;
4 endsets
5
6 data:
7 ES = 0.0891 0.2137 0.2346;
8 DS = 0.0108 0.0124 0.0131
9      0.0124 0.0584 0.0554
10     0.0131 0.0554 0.0942;
11 r = 0.15;
12 enddata
13
14 [obj] min = DS(1,1)*x(1)^2 + DS(2,2)*x(2)^2 + DS(3,3)*x(3)^2 +
15         2*DS(1,2)*x(1)*x(2) + 2*DS(1,3)*x(1)*x(3) + 2*DS(2,3)*x(2)*x(3);
16
17 [range] @sum(stock: x) <= 1;
18 [profit] @sum(stock: ES * x) >= r;

```

第 (2) 问的 LINGO 程序如下,

```

1 sets:
2 stock/1..4/: x, ES;
3 cov(stock, stock): DS;
4 endsets
5

```



```

6 data:
7 ES = 0.0891 0.2137 0.2346 0.0500;
8 DS = 0.0108 0.0124 0.0131 0.0000
9      0.0124 0.0584 0.0554 0.0000
10     0.0131 0.0554 0.0942 0.0000
11     0.0000 0.0000 0.0000 0.0000;
12 r = 0.15;
13 enddata
14
15 [obj] min = DS(1,1)*x(1)^2 + DS(2,2)*x(2)^2 + DS(3,3)*x(3)^2 +
16         2*DS(1,2)*x(1)*x(2) + 2*DS(1,3)*x(1)*x(3) + 2*DS(2,3)*x(2)*x(3);
17
18 [range] @sum(stock: x) <= 1;
19 [profit] @sum(stock: ES * x) >= r;

```

第 (3) 问的 LINGO 程序如下,

```

1 sets:
2 stock/1..3/: ES, x, y, z, c;
3 cov(stock, stock): DS;
4 endsets
5
6 data:
7 ES = 0.0891 0.2137 0.2346;
8 DS = 0.0108 0.0124 0.0131
9      0.0124 0.0584 0.0554
10     0.0131 0.0554 0.0942;
11 r = 0.15;
12 c = 0.50 0.35 0.15;
13 enddata
14
15 [obj] min = DS(1,1)*x(1)^2 + DS(2,2)*x(2)^2 + DS(3,3)*x(3)^2 +
16         2*DS(1,2)*x(1)*x(2) + 2*DS(1,3)*x(1)*x(3) + 2*DS(2,3)*x(2)*x(3);
17
18 [range] @sum(stock: x + 0.01*y + 0.01*z) <= 1;
19 [profit] @sum(stock: ES * x - 0.01*y - 0.01*z) >= r;
20 @for(stock: [trans] x = c + y - z);

```