HW1-Linear Programming

Diet Problem

Analysis

- Decision Variable
  - 每人每天对A_j类食品的需求量，记为x_j
  - $x_j \quad j = 1, 2, \ldots, n$
- Objective Function
  - 每人每天所需食物的总费用，记为z
  - $min \quad z = \sum_{j=1}^{n} c_j \cdot x_j$
- Constraints
  - 显性约束
    1. 每人每天摄入B_i成分的量大于等于b_i
       - $\sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i \quad i = 1, 2, \ldots, m$
    2. 每人每天摄入A_j食品的量大于等于d_j
       - $x_j \geq d_j \quad j = 1, 2, \ldots, n$

Model

$$min \quad z = \sum_{j=1}^{n} c_j \cdot x_j$$

$$s.t. \begin{cases} \sum_{j=1}^{n} a_{ij} \cdot x_j \geq b_i \quad i = 1, 2, \ldots, m \\ x_j \geq d_j \quad j = 1, 2, \ldots, n \end{cases}$$

Solve

我使用了Python的PuLP库进行求解

- Data
  - Supply

Food	Cost per serving	Vitamin A	Calories	Low Bound
Corn	0.18	107	72	5
2% milk	0.23	500	121	3
Wheat Bread	0.05	0	65	7

  - Need

Vitamin A	Calories
3500	2340

- Model

```
Diet_Problem:
MINIMIZE
0.18*x1 + 0.23*x2 + 0.05*x3 + 0.0
SUBJECT TO
Nutrition_Constraints1: 107 x1 + 500 x2 >= 3500

Nutrition_Constraints2: 72 x1 + 121 x2 + 65 x3 >= 2340

VARIABLES
5 <= x1 Continuous
3 <= x2 Continuous
7 <= x3 Continuous
```

- Result

x1	x2	x3
5.00	5.93	19.42

Bomber Problem

Analysis

- Mathematical Description
  - 记"重型炸弹"为"1"类炸弹，"轻型炸弹"为"2"类炸弹
  - 记第i类炸弹轰炸第j个要害点成功的概率为p_{ij}
  - 由题可转化得，载"1"型炸弹每公里油耗为$\frac{1}{2}L$，载"2"型炸弹每公里油耗为$\frac{1}{3}L$，空载时每公里油耗为$\frac{1}{4}L$，则可以据此算出飞机载i型导弹往返j要害点的总油耗，记为c_{ij}
- Decision Variable
  - 第i类炸弹投到第j个要害的数量，记为x_{ij}
  - $x_{ij}\begin{cases} i=1,2 \\ j=1,2,3,4 \end{cases}$
- Objective Function
  - 有一个要害点轰炸成功的概率，记为z
    - 有一个要害点轰炸成功的概率=1 - 所有要害点均毫发无损的概率
    - 所有要害点均毫发无损的概率为：

      $$\prod_{j=1}^{4}\prod_{i=1}^{2}(1-p_{ij})^{x_{ij}}$$

      有一个要害点轰炸成功的概率:

      $$z=1-\prod_{j=1}^{4}\prod_{i=1}^{2}(1-p_{ij})^{x_{ij}}$$

      这是一个比较棘手的非线性项目，好在第二项的所有运算都为乘法，因此我们可以通过取对数，转化为一个线性规划问题：

      $$ln(1-z)=ln(\prod_{j=1}^{4}\prod_{i=1}^{2}(1-p_{ij})^{x_{ij}})=\sum_{j=1}^{4}\sum_{i=1}^{2}ln(1-p_{ij})\cdot x_{ij}$$

    - 由于lnx的单调性，$max\quad z\iff min\quad ln(1-z)$
  - $min\quad\sum_{j=1}^{4}\sum_{i=1}^{2}ln(1-p_{ij})\cdot x_{ij}$
- Constraints

- 显性约束
    1. "1"型炮弹数小于等于48
        - $\sum_{j=1}^{4} x_{1j} \leq 48$
    2. "2"型炮弹数小于等于32
        - $\sum_{j=1}^{4} x_{2j} \leq 32$
    3. 总油耗小于等于48000
        - $\sum_{i=1}^{2} \sum_{j=1}^{4} x_{ij} \cdot c_{ij} \leq 48000$
- 隐性约束
    1. 投弹数量为整数且大于等于0
        - $x_{ij} \in \mathbf{N}$

Model

$$min \quad \sum_{j=1}^{4} \sum_{i=1}^{2} ln(1-p_{ij}) \cdot x_{ij}$$

$$s.t. \begin{cases} \sum_{j=1}^{4} x_{1j} \leq 48 \\ \sum_{j=1}^{4} x_{2j} \leq 32 \\ \sum_{i=1}^{2} \sum_{j=1}^{4} x_{ij} \cdot c_{ij} \leq 48000 \\ x_{ij} \in \mathbf{N} \end{cases}$$

Solve

虽然没要求求解，不过感觉既然都写出来了，我顺便锻炼一下PuLP语法的熟练程度，顺便验证一下结果对不对

- Data
    - Possibility of success

p_{ij}	**1**	**2**	**3**	**4**
1	0.10	0.2	0.15	0.25
2	0.08	0.16	0.12	0.20

- Cost of oil

以c_{11}为例，往返的油耗为$c_{11} = 100 + 450 \times 0.5 + 450 \times 0.25 + 100 = 537.5$，其余计算均同理，结果列于下表

c_{ij}	**1**	**2**	**3**	**4**
1	537.5	560	605	650
2	462.5	480	515	550

- Model

```
Bomber_Problem:
MINIMIZE
```

```
-0.10536051565782628*x11 + -0.2231435513142097*x12 + -0.16251892949777494*x13
+ -0.2876820724517809*x14 + -0.08338160893905101*x21 +
-0.1743533871447778*x22 + -0.12783337150988489*x23 + -0.2231435513142097*x24
+ 0.0
SUBJECT TO
Big_Bomb_Number_Constraints: x11 + x12 + x13 + x14 <= 48

Small_Bomb_Number_Constraints: x21 + x22 + x23 + x24 <= 32

Oil_Constraints: 537.5 x11 + 560 x12 + 605 x13 + 650 x14 + 462.5 x21 + 480
x22
 + 515 x23 + 550 x24 <= 48000

VARIABLES
0 <= x11 Integer
0 <= x12 Integer
0 <= x13 Integer
0 <= x14 Integer
0 <= x21 Integer
0 <= x22 Integer
0 <= x23 Integer
0 <= x24 Integer
```

- Result
  - Process

    ```
    Objective value:                -20.54832236
    Enumerated nodes:               10
    Total iterations:               133
    Time (CPU seconds):             0.02
    Time (Wallclock seconds):       0.02
    ```

  - Result

x_{ij}	**1**	**2**	**3**	**4**
1	0	1	0	46
2	0	1	0	31

    看起来还是挺合理的

附录

Prob1 Code

```
from pulp import *

## Data
m = 2                               # m种营养成分(i)
n = 3                               # n种食品(j)
```

```
b = [None,3500,2340]            # 第i种营养的每日必需量(b_i)
c = [None,0.18,0.23,0.05]       # 第j种食品的cost(c_j)
d = [None,5,3,7]                # 第j种食品的每日必需量(d_j)
a = [   None,
        [None,107,500,0],
        [None,72,121,65]
    ]                           # 第j种食品含有的i类成分(a_ij)


## Problem
problem = LpProblem('Diet_Problem', LpMinimize)
## Variables
x = [None]
for j in range(1,n+1):
    xj = LpVariable("x"+str(j),lowBound=d[j])
    x.append(xj)

## Objective Function
obj = 0
for j in range(1,n+1):
    obj += c[j]*x[j]
problem += obj, 'Objective Function'

## Constraints
for i in range(1,m+1):
    cons1i = lpSum(a[i][j]*x[j] for j in range(1,n+1))
    problem += cons1i >= b[i] , 'Nutrition_Constraints'+str(i)

## Solve
print(problem)
problem.solve()

## Print
for i in range(1,n+1):
    print("x"+str(i)+":",x[i].varValue)
```

Prob2 Code

```
from pulp import *
import numpy as np

## Data
m = 2                           # m种炸弹(i)
n = 4                           # n个要害(j)

p = [
        [0.1,0.2,0.15,0.25],
        [0.08,0.16,0.12,0.20]
    ]                           # 第i种炸弹轰炸要害j成功的可能性(p_ij)
```

```
c = [
        [537.5,560,605,650],
        [462.5,480,515,550]
    ]                           # 第i种炸弹轰炸要害j花费的油量(c_ij)


## Problem

problem = LpProblem('Bomber_Problem', LpMinimize)
## Variables

x = [ [LpVariable("x"+str(i+1)+str(j+1),lowBound=0,cat=const.LpInteger) for j
in range(n)] for i in range(m)]


## Objective Function
obj = lpSum(np.log(1-p[i][j])*x[i][j] for j in range(n) for i in range(m))
problem += obj, 'Objective Function'



## Constraints
problem += lpSum(x[0][j] for j in range(n)) <= 48
,'Big_Bomb_Number_Constraints'
problem += lpSum(x[1][j] for j in range(n)) <= 32
,'Small_Bomb_Number_Constraints'
problem += lpSum(x[i][j]*c[i][j] for j in range(n) for i in range(m)) <= 48000
,'Oil_Constraints'
## Solve
print(problem)
problem.solve()

## Print
for i in range(m):
    for j in range(n):
        print(x[i][j].varValue)
```