线性规划模型

Linear Programming

September 25, 2021

目录

线性规划模型介绍

在人们的生产实践中，经常会遇到如何利用有限的资源来安排生产，以取得最大经济效益的问题。此类问题构成了运筹学的一个重要分支:数学规划，而线性规划(Linear Programming, LP) 则是数学规划中最基础的模型，也是一种十分常用的最优化模型。
随着计算机的发展，线性规划的方法已经被应用于广泛的领域，已成为数学建模里最为经典，最为常用的模型之一。线性规划模型可用于求解利润最大，成本最小，路径最短等最优化问题。

线性规划的引入

一个线性规划模型，包含三个部分：**决策变量、目标函数和约束条件**

- **决策变量**: 决策变量是指问题中可以改变的量，例如生产多少货物，选择哪条路径等；线性规划的目标就是找到最优的决策变量。在线性规划中决策变量包括实数变量，整数变量，0-1变量等。
- **目标函数**: 目标函数就是把问题中的决策目标量化，一般分为最大化目标函数和最小化目标函数。在线性规划中，目标函数为一个包含决策变量的线性函数，例如$\max z = x_1 + x_2$或者 $\min z = 4x_1 - x_2$。
- **约束条件**: 约束条件是指问题中各种时间，空间，人力，物力等限制。在线性规划中，约束条件一般表示为一组包含决策变量的线性不等式，例如$x_1 + 2x_2 \leq 10$或者$4x_1 + 3x_2 \leq 24$。此外，部分决策变量有固定的取值范围，也形成约束条件，例如$x_1 \geq 0, x_2 \geq 0$。

Definition (线性规划)

如果一个规划问题的目标函数是决策变量的**线性**函数，约束条件也都是决策变量的**线性方程或线性不等式**的规划问题，则称为线性规划问题。

请思考，什么是线性，你能举几个例子?

线性规划的引入

Example (1)

某机床厂生产甲、乙两种机床，每台销售后的利润分别为4千元与3千元。已知：

- 生产甲机床需用A、B机器加工，加工时间分别为每台2小时和1小时
- 生产乙机床需用A、B、C三种机器加工，加工时间为每台各一小时

若每天可用于加工的机器时数分别为A机器10小时、B机器8小时和C机器7小时。问该厂应生产甲、乙机床各几台，才能使总利润最大?

这个问题是一个十分典型的线性规划问题，首先对问题提取出关键信息：

- **决策变量**：生产x_1台甲机床，x_2台乙机床
- **优化目标**：总利润最大
- **约束**：生产机床的使用时间有限

线性规划的引入

将上述三个要素写成数学表达式，就是一个典型的线性规划模型：

$$
\max \quad z = 4x_1 + 3x_2
$$

$$
\text{s.t.} \begin{cases} 2x_1 + x_2 \leq 10 \\ x_1 + x_2 \leq 8 \\ x_2 \leq 7 \\ x_1, x_2 \geq 0 \end{cases}
$$

实际上，在这个问题中，由于甲乙机床生产台数必须是整数，我们可以通过穷举法，对所有可能的生产方案进行计算。除此之外，对于较为简单且只有两个决策变量的线性规划问题可以使用图解法进行求解。

图解法求解线性规划

$$\max \quad z = 4x_1 + 3x_2$$

$$\text{s.t.} \begin{cases} 2x_1 + x_2 \le 10 \\ x_1 + x_2 \le 8 \\ x_2 \le 7 \\ x_1, x_2 \ge 0 \end{cases}$$

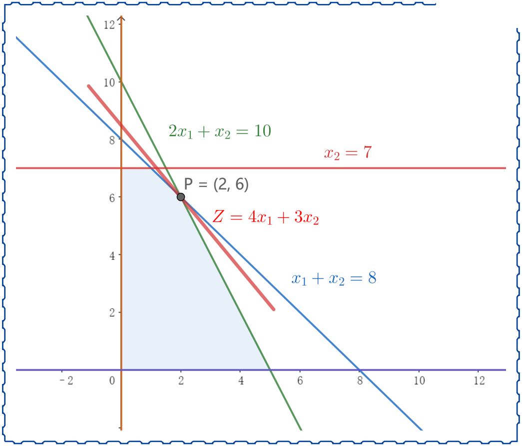

Figure: 图解法求解线性规划

以决策变量 x_1 为 x 轴，决策变量 x_2 为 y 轴，可以将可行域表示为如下图1所示的多边形，其中多边形的每一个边界为一个约束条件，目标函数则为一条直线，优化目标为使该条直线在 y 轴上的截距最大。求解结果为：当 $x_1 = 2, x_2 = 6$ 时，最大利润为26千元。

单纯型法求解线性规划*[1]

对于决策变量大于两个的线性规划模型，图解法不再适用。这样的问题可以采用单纯形法进行求解。单纯形法是1947 年G. B. Dantzig提出的一种十分有效的求解方法，极大地推广了线性规划的应用，直到今日也在一些线性规划的求解器中使用。
从前面图解法的例子中，我们可以看出，约束条件所围成的区域为一个凸多边形，当决策变量多于两个时，约束条件围城的区域为一个凸多面体，称之为可行域。其中每一个面（称之为超平面）即代表一个约束条件。

定理

线性规划的最优解一定在可行域的边界上

[1]标*的内容为拓展内容，不要求掌握

单纯型法求解线性规划*

单纯形法的思路就是在可行域的一个顶点处找到一个初始可行解，判断该解是不是最优，若不是，则迭代到下一个顶点处进行重复判断。因为最优解的搜索范围从整个可行域缩小到了可行域的有限个顶点，算法的效率得到了极大的提升。
具体的找初始可行解的方法，判断解是否最优的条件，以及如何进行迭代这里不做详细展开，有兴趣的同学可以查阅相关资料。此外，求解线性规划的方法还有椭球法、卡玛卡算法、内点法等。其中内点法因为求解效率更高，在决策变量多，约束多的情况下能取得更好的效果，目前主流线性规划求解器都是使用的内点法。

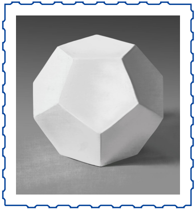

Figure: 凸多面体

线性规划的标准形式

为了能够使用计算机求解线性规划问题，我们首先介绍线性规划的标准型。例如，对于一个线性规划模型：

$$\min \quad z = x_1 + x_2$$
$$\text{s.t.} \quad \begin{cases} x_1 + 2x_2 \leq 10 \\ 4x_1 + 3x_2 \leq 24 \\ x_1, x_2 \geq 0 \end{cases}$$

其中$s.t.$为subject to的缩写。上述模型可以写成如下的矩阵形式：

$$\min c^T x$$
$$\text{s.t.} \quad \begin{cases} Ax \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases}$$

线性规划的标准形式

$$\min \quad z = x_1 + x_2$$

$$\text{s.t.} \begin{cases} x_1 + 2x_2 \leq 10 \\ 4x_1 + 3x_2 \leq 24 \\ x_1, x_2 \geq 0 \end{cases}$$

勺缩写。上述模型可以写成如下

$$\min c^T x$$

$$\text{s.t.} \begin{cases} Ax \leq b \\ Aeq \cdot x = beq \\ lb \leq x \leq ub \end{cases}$$

$$c = [1, 1]^T$$

$$x = [x_1, x_2]^T$$

$$A = \begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix}$$

$$b = \begin{bmatrix} 10 \\ 24 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 4 & 3 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$1 \times x_1 + 2 \times x_2$$

$$4 \times x_1 + 3 \times x_2$$

$$Aeq = [](\text{没有等式约束，因此为空})$$

$$beq = [](\text{没有等式约束，因此为空})$$

$$lb = [0, 0]^T$$

$$ub = [+\infty, +\infty]^T$$

线性规划的标准形式

线性规划的目标函数可能是最大化，也可能是最小化，约束条件的符号可能是小于等于，也可能是大于等于，甚至还有等于。因此为了编程方便，一般统一为目标函数最小化，不等式约束写为小于等于约束。等式约束不变。具体来说，最大化目标函数可以添加负号变为最小化约束：

$$\max z = x_1 + x_2 \Longrightarrow \min -z = -x_1 - x_2$$

大于等于约束可以两边乘以-1 变为小于等于约束：

$$x_1 + 2x_2 \geq 10 \Longrightarrow -x_1 - 2x_2 \leq -10$$

~~等于约束可以变为一个大于等于约束和一个小于等于约束，但在编程中一般支持直接写等式约束,可以不进行转换：~~

$$x_1 + 2x_2 = 10 \Longrightarrow x_1 + 2x_2 \leq 10, x_1 + 2x_2 \geq 10$$

请写出以下两个线性规划的标准型

$c=[5\ 4\ 6]\quad x=[x_1\ x_2\ x_3]$

$$A=\begin{bmatrix}-3 & -2 & +1\\ -4 & -3 & +2\\ 1 & 1 & 1\end{bmatrix}\quad b=\begin{bmatrix}-10\\ -24\\ 8\end{bmatrix}$$

$$Aeq=\begin{bmatrix}5 & 2 & 0\\ 0 & 2 & 3\end{bmatrix}\quad beq=\begin{bmatrix}10\\ 6\end{bmatrix}$$

$lb=[0\ 0\ 0]\quad ub=[\infty\ \infty\ \infty]$

$$\min \quad z = 5x_1 + 4x_2 + 6x_3$$

$$\text{s.t.}\begin{cases} 3x_1 + 2x_2 - x_3 \geq 10 \\ 4x_1 + 3x_2 - 2x_3 \geq 24 \\ x_1 + x_2 + x_3 \leq 8 \\ 5x_1 + 2x_2 = 10 \\ 2x_2 + 3x_3 = 6 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

$$\min \quad -z = -x_1 - x_2 - 3x_3 - 2x_4$$

$$\max \quad z = x_1 + x_2 + 3x_3 + 2x_4$$

$$\text{s.t.}\begin{cases} -(3x_1 + 2x_2 - x_3) \leq -5 \\ -(4x_1 + 3x_2 - 2x_3 + x_4) \leq -12 \\ x_1 + x_2 + x_3 + x_4 \leq 7 \\ 5x_1 + 2x_2 = 5 \\ 2x_2 + 3x_4 = 4 \\ x_1, x_2, x_3, x_4 \geq 0 \end{cases}$$

$c=[-1\ -1\ -3\ -2]^T$

$$A=\begin{bmatrix}-3 & -2 & 1 & 0\\ -4 & -3 & 2 & -1\\ 1 & 1 & 1 & 1\end{bmatrix}\quad b=\begin{bmatrix}-5\\ -12\\ 7\end{bmatrix}$$

$$Aeq=\begin{bmatrix}5 & 2 & 0 & 0\\ 0 & 2 & 0 & 3\end{bmatrix}\quad beq=\begin{bmatrix}5\\ 4\end{bmatrix}$$

$lb=[0\ 0\ 0\ 0]^T$

$ub=[\infty\ \infty\ \infty\ \infty]^T$

使用Python求解简单线性规划模型

Example (2)

考虑如下线性规划问题

$$\max z = 2x_1 + 3x_2 - 5x_3$$
$$\text{s.t.} \quad \begin{cases} x_1 + x_2 + x_3 = 7 \\ 2x_1 - 5x_2 + x_3 \geq 10 \\ x_1 + 3x_2 + x_3 \leq 12 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

使用Python求解简单线性规划模型

```
import numpy as np
from scipy import optimize as op # 导入相关库
x1=(0,7)
x2=(0,7)
x3=(0,7) # 给出变量取值范围
c=np.array([-2,-3,5]) # 目标函数系数,3 *1 列向量
A=np.array([[-2,5,-1],[1,3,1]]) # 不等式约束系数，A2维矩阵*3
b=np.array([-10,12]) # 等式约束系数b, 维列向量2*1
A_eq=np.array([[1,1,1]]) # 等式约束系数，Aeq3维列向量*1
b_eq=np.array([7]) # 等式约束系数，beq1数值*1
res=op.linprog(c,A,b,A_eq,b_eq,bounds=(x1,x2,x3)) # 调用函数进行求解
```

求解得当 $x_1 = 6.43, x_2 = 0.57, x_3 = 0$ 时，目标函数取得最大值 $z = 14.57$。

课堂练习

请使用python编写程序，求解以下线性规划问题

$$
\min z = 2x_1 + 3x_2 + x_3
$$
$$
\text{s.t.} \quad \begin{cases} x_1 + 4x_2 + 2x_3 \geq 8 \\ 3x_1 + 2x_2 \geq 6 \\ x_1, x_2, x_3 \geq 0 \end{cases}
$$

当 $x_1 = \cancel{0.8}, x_2 = \cancel{1.8}, x_3 = \cancel{0}$ 时，目标函数取得最小值 $z = 7$（请自行验证）。

案例分析：投资的收益和风险

市场上有n种资产$s_i(i=1,2,,n)$可以选择，现用数额为M的相当大的资金作一个时期的投资。购买这n种资产的收益率为r_i，风险损失率为q_i，投资越分散，总的风险越少，总体风险可以用投资s_i中最大的一个风险来度量。
此外，购买s_i时还要付交易费（费率为p_i），当购买额不超过给定值u_i时，交易费按购买额u_i计算。另外，假定同期银行存款利率为r_0，既无交易费又无风险（$r_0=5\%$）。已知$n=4$时相关数据如下表

s_i	$r_i(\%)$	q_i	$p_i(\%)$	u_i(RMB)
s_1	28	2.5	1	103
s_2	21	1.5	2	198
s_3	23	5.5	4.5	52
s_4	25	2.6	6.5	40

试给该公司设计一种投资组合方案，用给定的资金M，有选择地购买若干种资产或者银行生息，使净收益最大，总体风险最小。

案例分析：投资的收益和风险

符号规定

- s_i :第i种投资项目，如股票、债券
- r_i, p_i, q_i：分别为s_i的平均收益率，交易费率，风险损失率
- u_i ：s_i的交易定额
- r_0 :同期银行利率
- x_i ：投资项目s_i的资金
- a ：投资风险度
- Q ： 总体收益

模型假设

- 投资数额M相当大，为了便于计算，假设$M = 1$
- 投资越分散，总风险越小，总体风险用投资项目s_i中最大的一个风险来度量
- n种资产s_i之间是相互独立的
- 在投资的这一时期内，r_i, p_i, q_i, r_0为定值

案例分析：投资的收益和风险

根据模型假设和符号规定，我们可以写出模型的第一个优化目标为总体风险尽可能小,而总体风险是所有投资中风险最大的一个

第i项投资的风险 总体风险

$$\min\left[\max\{\underline{q_i x_i}|i=1,2,...n\}\right]$$

第二个优化目标为净收益尽可能大。根据题意，交易费用为一个分段函数（非线性函数），因此需要进行简化：由于题目给定的定值 u_i 相对于总投资额 M 很小，可以忽略不计，因此将交易费简化为 $p_i x_i$,所以目标函数为

$$\max\sum_{i=1}^{n}(r_i-p_i)x_i$$

案例分析：投资的收益和风险

对于一个多目标优化模型，常用的考虑方式为先固定其中一个目标，再优化另一个目标。在本题中，可以给定一个投资者能够承受的风险界限a, 使得最大投资风险下损失比例小于 a, 即 $q_i x_i < Ma$ ，将其作为新的约束，就可以把多目标优化转化为单目标优化，即：

$$\max \sum_{i=1}^{n} (r_i - p_i) x_i$$

$$s.t. \begin{cases} q_i x_i < Ma & \rightarrow \text{总体风险最小} \\ \sum_{i=1}^{n} (1 + p_i) x_i = M \\ x_i \geq 0, i = 1, 2, \ldots n \end{cases}$$

其中，a 反映了投资者对风险的偏好程度，从 $a = 0$ 开始，以步长为 0.001 进行循环搜索，使用 python 编写代码如下

```python
import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize as op #导入相关库
a = 0 #定义的取值a
profit_list = [] #记录最大收益
a_list = [] #记录的取值a
while a<0.05:
    x1=(0,None) #定义决策变量取值范围
    c=np.array([-0.05,-0.27,-0.19,-0.185,-0.185]) #定
        义目标函数系数
    A = np.hstack((np.zeros((4,1)),np.diag
        ([0.025,0.015,0.055,0.026]))) #定义不等式约束
        条件左边系数
```

案例分析：投资的收益和风险

```
    b=a*np.ones((4,1)); #定义不等式约束条件右边系数
    Aeq=np.array([[1,1.01,1.02,1.045,1.065]]) #定义等式约束条件左边系数
    beq=np.array([1]); #定义等式约束条件右边系数
   #求解
    res=op.linprog(c,A,b,Aeq,beq,bounds=(x1,x1,x1,x1,x1))
    profit = -res.fun
    profit_list.append(profit)
    a_list.append(a)
    a = a + 0.001
plt.figure(figsize=(10,7))
plt.plot(a_list,profit_list)
plt.xlabel('a');plt.ylabel('Profit') #绘制风险偏好与最大收益的曲线图a
```

案例分析：投资的收益和风险

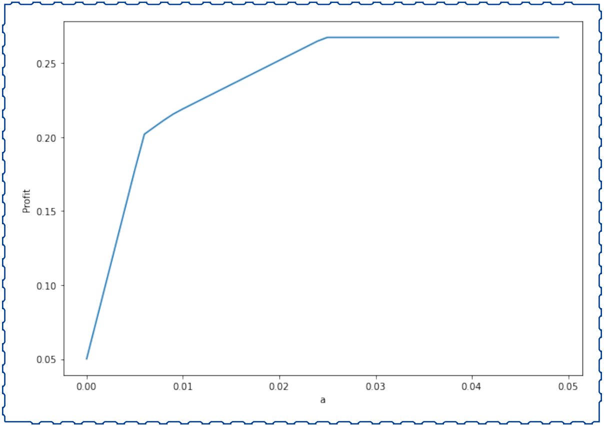

Figure: 风险偏好a与最大收益关系曲线

案例分析：投资的收益和风险

从上图中可以看出

1. 风险越大，收益也就越大；
2. 当投资越分散时，投资者承担的风险越小，这与题意一致。即：冒险的投资者会出现集中投资的情况，保守的投资者则尽量分散投资。
3. 在 $a = 0.006$ 附近有一个转折点，在这点左边，风险增加很少时，利润增长很快。在这一点右边，风险增加很大时，利润增长很缓慢，所以对于风险和收益没有特殊偏好的投资者来说，应该选择曲线的拐点作为最优投资组合，大约是 $a = 0.6\%$，总体收益为 $Q = 20\%$，所对应投资方案为：风险度 $a = 0.006$，收益 $Q = 0.2019$，$x_0 = 0$fi$x_1 = 0.24$fi$x_2 = 0.4$fi$x_3 = 0.1091, x_4 = 0.2212$。

案例分析：投资的收益和风险

在上面的例子中，我们使用固定风险水平来最大化收益的方法来将多目标转化为单目标，也可考虑其他思路：

- 在总盈利在水平 k 以上的情况下，寻找风险最低的投资方案，即：

$$\min \{\max q_i x_i\}$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{n} (r_i - p_i) x_i \geq k \\ \sum_{i=1}^{n} (1 + p_i) x_i = M \\ x_i \geq 0, i = 1, 2, \ldots n \end{cases}$$

- 对风险和收益赋予权重 $s(0 \leq s \leq 1)$ 和 1`s, s为投资偏好系数，

$$\min \quad s \max\{q_i x_i\} - (1 - s) \sum_{i=1}^{n} (r_i - p_i) x_i$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{n} (1 + p_i) x_i = M \\ x_i \geq 0, i = 1, 2, \ldots n \end{cases}$$

课后作业

- 请使用Python scipy库 的optimize.linprog方法，求解以下线性规划问题,并通过图解法验证。

$$\max z = 4x_1 + 3x_2$$
$$\text{s.t.} \quad \begin{cases} 2x_1 + x_2 \leq 10 \\ x_1 + x_2 \leq 8 \\ x_1, x_2 \geq 0 \end{cases}$$

- 请使用Python scipy库 的optimize.minimize方法，求解以下非线性规划问题

$$\min z = x_1^2 + x_2^2 + x_3^2$$
$$\text{s.t.} \quad \begin{cases} x_1 + x_2 + x_3 \geq 9 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

课后作业

- 某农场 I,II,III 等耕地的面积分别为 $100hm^2$、$300hm^2$ 和 $200hm^2$，计划种植水稻、大豆和玉米，要求三种作物的最低收获量分别为190000kg、130000kg和350000kg。I,II,III 等耕地种植三种作物的单产如下表所示。

	I等耕地	II等耕地	III等耕地
水稻	11000	9500	9000
大豆	8000	6800	6000
玉米	14000	12000	10000

若三种作物的售价分别为水稻1.20元/kg，大豆1.50元/kg，玉米0.80元/kg。那么，

  - 如何制订种植计划才能使总产量最大?
  - 如何制订种植计划才能使总产值最大?
  - 要求：写出规划问题的标准型，并合理采用本课程学到的知识，进行求解。

Thank You

- 线性规划模型