第5章 MATLAB线性规划

线性规划是运筹学中研究较早、发展较快、应用广泛且方法较成熟的一个重要分支，它是辅助人们进行科学管理的一种数学方法。

学习目标：

（1）了解 MATLAB 线性规划基本概念；

（2）了解 MATLAB 线性规划标准形式；

（3）掌握 MATLAB 中线性规划函数的应用；

（4）熟练掌握 MATLAB 线性规划问题求解。

5.1 线性规划的概念

线性规划是研究线性约束条件下线性目标函数极值问题的数学理论和方法，英文缩写为 LP。它是运筹学的一个重要分支，广泛应用于军事作战、经济分析、经营管理和工程技术等方面。为合理地利用有限的人力、物力、财力等资源做出最优决策，提供科学的依据。

线性规划模型首先是列出约束条件及目标函数，然后画出约束条件所表示的可行域，最后在可行域内求目标函数的最优解及最优值。线性规划模型求解流程图如图 5-1 所示。

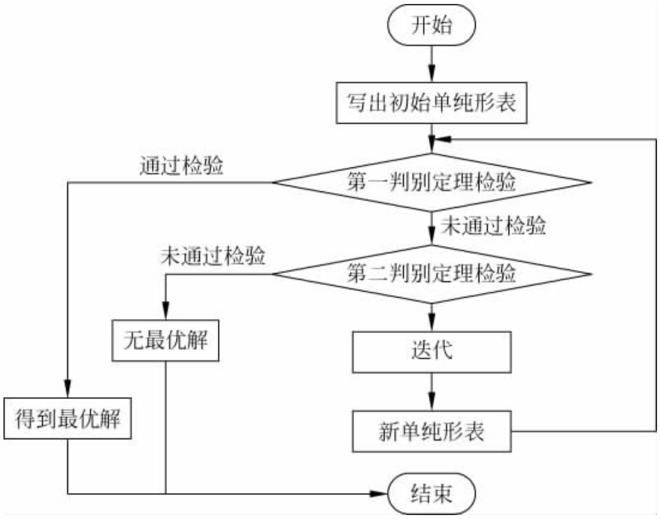

图 5-1　线性规划模型求解流程图

5.2 线性规划的标准形式

线性规划方法是在第二次世界大战中发展起来的一种重要的数学方法，它是处理线性目标函数和线性约束的一种较为成熟的方法，主要用于研究有限资源的最佳分配问题，即如何对有限的资源作出最佳方式地调配和最有利地使用，以便最充分地发挥资源的效能去获取最佳的经济效益。目前已经广泛应用于军事、经济、工业、农业、教育、商业和社会科学等方面。

线性规划问题的标准形式是

$$\begin{cases} \min z = c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \\ x_1, x_2, \cdots, x_n \geqslant 0 \end{cases}$$

或

$$\begin{cases} \min z = \sum_{j=1}^{n} c_jx_j \\ \sum_{j=1}^{n} a_{ij}x_j = b_i, i = 1,2,\cdots,m \\ x_j \geqslant 0, j = 1,2,\cdots,n \end{cases}$$

写成矩阵形式为

$$\begin{cases} \min z = \boldsymbol{CX} \\ \boldsymbol{AX} = \boldsymbol{b} \\ \boldsymbol{X} \geqslant 0 \end{cases}$$

线性规划的标准形式要求使目标函数最小化，约束条件取等式，变量 b 非负。不符合这几个条件的线性模型可以转化为标准形式。

从实际问题中建立数学模型一般有以下三个步骤：

(1) 根据影响所要达到目的的因素找到决策变量；

(2) 由决策变量和所要达到目的之间的函数关系确定目标函数；

(3) 由决策变量所受的限制条件确定决策变量所要满足的约束条件。

所建立的数学模型具有以下特点：

(1) 每个模型都有若干个决策变量($x_1, x_2, x_3, \cdots, x_n$)，其中 n 为决策变量个数。决策变量的一组值表示一种方案，同时决策变量一般是非负的。

(2) 目标函数是决策变量的线性函数，根据具体问题可以是最大化(max)或最小化(min)，二者统称为最优化(opt)。

(3) 约束条件也是决策变量的线性函数。

当得到的数学模型的目标函数为线性函数，约束条件为线性等式或不等式时，称此

数学模型为线性规划模型。

5.3 线性规划的 MATLAB 函数

在 MATLAB 中，用于 LP 的求解函数为 linprog，其调用格式为

```
x = linprog(f,A,b,Aeq,beq)
x = linprog(f,A,b,Aeq,beq,lb,ub)
x = linprog(f,A,b,Aeq,beq,lb,ub,x0)
x = linprog(f,A,b,Aeq,beq,lb,ub,x0,options)
[x,fval] = linprog(...)
[x,fval,exitflag] = linprog(...)
[x,fval,exitflag,output] = linprog(...)
[x,fval,exitflag,output,lambda] = linprog(...)
```

其中，f、A、b 是不可默认的输入变量，x 是不可默认的输出变量，它是问题的解。lb、ub 均是向量，分别表示 x 的下界和上界，x0 为 x 的起始点，options 为 optimset 函数中定义的参数的值，fval 是目标函数在解 x 处的值，lambda 为在解 x 处的 Lagrange 乘子。

lambda 参数的属性如下：

lambda. lower：lambda 的下界。

lambda. upper：lambda 的上界。

lambda. ineqlin：lambda 的线性不等式。

lambda. eqlin：lambda 的线性等式。

函数 linprog 的具体用例解释如下：

x=linprog(f,A,b)：求解问题 min $f^T x$，约束条件为 $Ax \leqslant b$。

x=linprog(f,A,b,Aeq,beq)：求解上面的问题，但增加等式约束，即 Aeqx＝beq。若没有不等式存在，则令 A=[]、b=[]。

x=linprog(f,A,b,Aeq,beq,lb,ub)：定义设计变量 x 的下界 lb 和上界 ub，使得 x 始终在该范围内。若没有等式约束，令 Aeq=[]、beq=[]。

x=linprog(f,A,b,Aeq,beq,lb,ub,x0)：设置初值为 x0。该选项只适用于中型问题，默认大型算法将忽略初值。

x=linprog(f,A,b,Aeq,beq,lb,ub,x0,options)：用 options 指定的优化参数进行最小化。

[x,fval]=linprog(…)：返回解 x 处的目标函数值 fval。

[x,lambda,exitflag]=linprog(…)：返回 exitflag 值，描述函数计算的退出条件。

[x,lambda,exitflag,output]=linprog(…)：返回包含优化信息的输出变量 output。

[x,fval,exitflag,output,lambda]=linprog(…)：将解 x 处的 Lagrange 乘子返回到 lambda 参数中。

【例 5-1】 有以下模型：

$$\min Z=-4a+b+7c$$

$$\text{s.t.}\begin{cases}a+b-c=5\\3a-b+c\leqslant 4\\a+b-4c\leqslant -7\\a,b\geqslant 0\end{cases}$$

请问 a,b,c 分别取何值时，Z 有最小值？

解：根据题中模型，编写以下代码：

```
clear all
clc
c=[-4 1 7];
A=[3 -1 1;1 1 -4];
b=[4; -7];
Aeq=[1 1 -1];
beq=[5];vlb=[0, 0];
vub=[];
[x,fval]=linprog(c,A,b,Aeq,beq,vlb,vub)
```

运行后得到结果如下：

```
Optimization terminated.

x =
    2.2500
    6.7500
    4.0000

fval =
   25.7500
```

即 a,b,c 分别取 2.2500、6.7500、4.0000 时，Z 有最小值 25.7500。

【例 5-2】 根据以下模型：

$$\begin{cases}x(1)+x(2)\leqslant 2\\x(1)+x(2)/4\leqslant 1\\x(1)-x(2)\leqslant 2\\-x(1)/4-x(2)\leqslant 1\\-x(1)-x(2)\leqslant -1\\-x(1)+x(2)\leqslant 2\end{cases}$$

求解 $\boldsymbol{X}$ 值。

解：根据题意，编写代码如下：

```
clear all
clc
A = [  1    1
       1  1/4
       1   - 1
     - 1/4 - 1
     - 1   - 1
     - 1    1];

b = [2  1  2  1  - 1  2];

Aeq = [1  1/4];
beq = 1/2;

lb = [ - 1, - 0.5];
ub = [1.5,1.25];
f = [ - 1  - 1/3];

x = linprog(f,A,b,Aeq,beq,lb,ub)
```

运行后得到结果如下：

```
Optimization terminated.
x =
    0.1875
    1.2500
```

【例 5-3】 求函数 $f(x)=-5x_1-4x_2-6x_3$ 的最小值并且满足以下条件：

$$\begin{cases} x_1-x_2+x_3\leqslant 20 \\ 3x_1+2x_2+4x_3\leqslant 42 \\ 3x_1+2x_2\leqslant 30 \\ x_1\geqslant 0 \\ x_2\geqslant 0 \\ x_3\geqslant 0 \end{cases}$$

解：根据目标函数及限制条件，编写代码如下：

```
clear all
clc

f = [ - 5; - 4; - 6];

A = [ 1   - 1  1
      3    2  4
      3    2  0];
b = [20;42;30];

lb = zeros(3,1);
```

```
Aeq = [];
beq = [];

[x,fval,exitflag,output,lambda] = linprog(f,A,b,Aeq,beq,lb);

x,lambda.ineqlin,lambda.lower

A * x
```

运行后得到结果如下：

```
Optimization terminated.

x =
    0.0000
   15.0000
    3.0000
ans =
    0.0000
    1.5000
    0.5000
ans =
    1.0000
    0.0000
    0.0000
ans =
  - 12.0000
    42.0000
    30.0000
```

5.4 线性规划问题求解方法

求解线性规划问题的基本方法是单纯形法。为了提高解题速度，又有改进单纯形法、对偶单纯形法、原始对偶方法、分解算法和各种多项式时间算法。对于只有两个变量的简单的线性规划问题，也可采用图解法求解。

线性规划包括单纯形线性规划和多目标线性规划。

5.4.1 单纯形线性规划问题求解

单纯形法是从所有基本可行解的一个较小部分中通过迭代过程选出最优解，其迭代过程的一般描述如下：

(1) 将线性规划化为典范形式，从而可以得到一个初始基本可行解 $x(0)$（初始顶点），将它作为迭代过程的出发点，其目标值为 $z(x(0))$。

（2）寻找一个基本可行解 $x(1)$，使 $z(x(1))\leqslant z(x(0))$。方法是通过消去法将产生 $x(0)$ 的典范形式化为产生 $x(1)$ 的典范形式。

（3）继续寻找较好的基本可行解 $x(2)$，$x(3)$，…使目标函数值不断改进，即 $z(x(1))\geqslant z(x(2))\geqslant z(x(3))\geqslant\cdots$ 当某个基本可行解再也不能被其他基本可行解改进时，它就是所求的最优解。

MATLAB 采用投影法求解线性规划问题，该方法是单纯形法的变种。MATLAB 中用来求解线性规划的函数是 linprog。

【例 5-4】 求函数的最小值 $f(x)=5x_1+3x_2-7x_3$，其中 $\boldsymbol{x}$ 满足条件

$$\begin{cases} x_1-x_2+x_3\leqslant 23 \\ 3x_1+2x_2+4x_3\leqslant 40 \\ 3x_1+2x_2\leqslant 32 \\ 0\leqslant x_1,0\leqslant x_2,0\leqslant x_3 \end{cases}$$

解：首先将变量按顺序排好，然后用系数表示目标函数，即

```
f = [5;3; -7];
```

因为没有等式条件，所以 Aeq、beq 都是空矩阵，即

```
Aeq = [ ] ;
beq = [ ] ;
```

不等式条件的系数为

$$\boldsymbol{A}=\begin{bmatrix} 1 & -1 & 1 \\ 3 & 2 & 4 \\ 3 & 2 & 0 \end{bmatrix},\quad \boldsymbol{b}=\begin{bmatrix} 23 \\ 40 \\ 32 \end{bmatrix}$$

由于没有上限要求，故 lb、ub 设为

$$\boldsymbol{lb}=\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},\quad \boldsymbol{ub}=\begin{bmatrix} \text{inf} \\ \text{inf} \\ \text{inf} \end{bmatrix}$$

根据以上分析，编写 MATLAB 代码如下：

```
clear all
clc
f = [5; 3; -7];                          %目标函数的系数
A = [ 1  -1  1
      3   2  4
      3   2  0];
b = [23; 40; 32];
lb = [0;0;0];                            %各变量的下限
ub = [inf;inf;inf];                      %各变量的上限
[x,fval] = linprog(f,A,b,[ ],[ ],lb,[ ]); %求解运算
x
fval
```

运行程序得到结果如下：

```
Optimization terminated.

x =
     0.0000
     0.0000
    10.0000
fval =
    -70.0000
```

【例 5-5】 求解下列优化问题：

$$f(x) = -6x_1 + 5x_2 - 4x_3$$

$$\begin{cases} x_1 - x_2 + x_3 \leqslant 19 \\ 3x_1 + 2x_2 + 4x_3 \leqslant 36 \\ 3x_1 + 2x_2 \leqslant 25 \\ 0 \leqslant x_1, \quad 0 \leqslant x_2, \quad 0 \leqslant x_3 \end{cases}$$

解：在 MATLAB 命令窗口输入以下代码：

```
clear all
clc
f=[-6;5;-4];
A=[1 -1 1;3 2 4;3 2 0];
b=[19;36;25];
lb=zeros(3,1);
[x,fval,exitflag,output,lambda]=linprog(f,A,b,[],[],lb)
```

运行代码得到结果如下：

```
Optimization terminated.

x =
    8.3333
    0.0000
    2.7500
fval =
  -61.0000
exitflag =
     1

output =
  包含以下字段的 struct:
         iterations: 5
          algorithm: 'interior-point-legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
     constrviolation: 0
```

```
      firstorderopt: 2.7701e-07

lambda =
  包含以下字段的 struct:
    ineqlin: [3×1 double]
      eqlin: [0×1 double]
      upper: [3×1 double]
      lower: [3×1 double]
```

exitflag = 1 表示过程正常收敛于解 x 处。

5.4.2 多目标线性规划问题求解

多目标线性规划是多目标最优化理论的重要组成部分,由于多个目标之间的矛盾性和不可公度性,要求使所有目标均达到最优解是不可能的,因此多目标规划问题往往只是求其有效解。

目前求解多目标线性规划问题有效解的方法包括理想点法、线性加权和法、最大最小法、目标规划法。

多目标线性规划有两个以及两个以上的目标函数,且目标函数和约束条件全是线性函数,其数学模型表示为

$$\max\begin{cases} z_1 = c_{11}x_1 + c_{12}x_2 + \cdots + c_{1n}x_n \\ z_2 = c_{21}x_1 + c_{22}x_2 + \cdots + c_{2n}x_n \\ \quad\vdots \\ z_r = c_{r1}x_1 + c_{r2}x_2 + \cdots + c_{rn}x_n \end{cases}$$

约束条件为

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \leqslant b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \leqslant b_2 \\ \quad\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n \leqslant b_m \\ x_1, x_2, \cdots, x_n \geqslant 0 \end{cases}$$

上述多目标线性规划可用矩阵形式表示为

$$\max \boldsymbol{Z} = \boldsymbol{C}x$$

约束条件为

$$\begin{cases} \boldsymbol{A}x \leqslant \boldsymbol{b} \\ x \geqslant 0 \end{cases}$$

1. 理想点法

$\max \boldsymbol{Z}=\boldsymbol{C}x$

在 $\begin{cases} \boldsymbol{A}x\leqslant\boldsymbol{b} \\ x\geqslant 0 \end{cases}$ 中,先求解 r 个单目标问题:$\min\limits_{x\in D} Z_j(x), j=1,2,\cdots,r$。设其最优值为

Z_j^*，称 $\boldsymbol{Z}^*$ 为值域中的一个理想点。于是在期望的某种度量下，寻求距离 $\boldsymbol{Z}^*$ 最近的 $\boldsymbol{Z}$ 作为近似值。一种最直接的方法就是最短距离理想点法，构造评价函数 $\varphi(Z)=\sqrt{\sum_{i=1}^{r}[Z_i-Z_i^*]^2}$，然后极小化 $\varphi[Z(x)]$，即求解 $\min\limits_{x\in D}\varphi[Z(x)]=\sqrt{\sum_{i=1}^{r}[Z_i(x)-Z_i^*]^2}$，maz$\boldsymbol{Z}=\boldsymbol{C}x$ 并将它的最优解 x^* 作为 $\begin{cases}\boldsymbol{A}x\leqslant\boldsymbol{b}\\x\geqslant 0\end{cases}$ 在这种意义下的最优解。

【例 5-6】 利用理想点法求解

$$\max f_1(x)=-3x_1-4x_2$$
$$\max f_2(x)=-5x_1+2x_2$$
$$\text{s. t}\begin{cases}2x_1-3x_2\leqslant 19\\3x_1+x_2\leqslant 11\\x_1,x_2\geqslant 0\end{cases}$$

解：先分别对单目标求解。

求解 $f_1(x)$ 最优解的 MATLAB 代码如下：

```
clear all
clc
f = [ - 3; - 4];
A = [2, - 3;3,1];
b = [19;11];
lb = [0;0];
[x,fval] = linprog(f,A,b,[],[],lb)
```

结果输出如下：

```
Optimization terminated.
x =
     0.0000
    11.0000
fval =
   - 44.0000
```

即最优解为 44。

求解 $f_2(x)$ 最优解的 MATLAB 代码如下：

```
f = [ - 5; - 5];
A = [2, - 3;3,1];
b = [19;11];
lb = [0;0];
[x,fval] = linprog(f,A,b,[],[],lb)
```

结果输出如下：

```
Optimization terminated.
x2 =
    0.0000
   11.0000
fval =
  -22.0000
```

即最优解为 22。

于是得到理想点：(44,22)。

然后求如下模型的最优解：

$$\min_{x\in D}\varphi[f(x)] = \sqrt{[f_1(x)-44]^2+[f_2(x)-22]^2}$$

$$\text{s. t}\quad \begin{cases} 2x_1-3x_2\leqslant 19 \\ 3x_1+x_2\leqslant 11 \\ x_1,x_2\geqslant 0 \end{cases}$$

MATLAB 代码如下：

```
A = [2, - 3;3,1];
b = [19;11];
x0 = [1;1];
lb = [0;0];
x = fmincon('((3 * x(1) - 4 * x(2) - 44)^2 + (5 * x(1) + 2 * x(2) - 22)^2)^(1/2)',x0,A,b,[],[],
lb,[])
```

结果输出如下：

```
x =
    3.6667
    0.0000
```

2. 线性加权和法

在具有多个指标的问题中，人们总希望对那些相对重要的指标给予较大的权系数，因而将多目标向量问题转化为所有目标的加权求和的标量问题。

基于上述设计，构造如下评价函数，即

$$\min_{x\in D} Z(x) = \sum_{i=1}^{r}\omega_i Z_i(x)$$

$$\max \boldsymbol{Z} = \boldsymbol{Cx}$$

将它的最优解 $\boldsymbol{x}^*$ 作为 $\begin{cases}\boldsymbol{A}x\leqslant\boldsymbol{b}\\ x\geqslant 0\end{cases}$ 在线性加权和意义下的最优解。其中 ω_i 为加权因子，其选取的方法很多，有专家打分法、容限法和加权因子分解法等。

【例 5-7】 对例 5-6 采用线性加权和法求解。(权系数分别取 $\omega_1=0.5$，$\omega_2=0.5$)

解：构造如下评价函数，即求如下模型的最优解：

$$\min\{-4\times(-3x_1+4x_2)+1\times(-5x_1-2x_2)\}$$

$$\text{s.t}\quad\begin{cases}2x_1-3x_2\leqslant 19\\3x_1+x_2\leqslant 11\\x_1,x_2\geqslant 0\end{cases}$$

MATLAB 代码如下：

```
clear all
clc
f = [ - 4;1];
A = [2, - 3;3,1];
b = [19;11];
lb = [0;0];
x = linprog(f,A,b,[],[],lb)
```

结果输出如下：

```
Optimization terminated.

x =
    3.6667
    0.0000
```

3. 最大最小法

在决策的时候，采取保守策略是稳妥的，即在最坏的情况下，寻求最好的结果，按照此想法，可以构造如下评价函数，即

$$\varphi(Z)=\max_{1\leqslant i\leqslant r}Z_i$$

然后求解

$$\min_{x\in D}\varphi[Z(x)]=\min_{x\in D}\max_{1\leqslant i\leqslant r}Z_i(x)$$

$$\max\boldsymbol{Z}=\boldsymbol{Cx}$$

并将它的最优解 $\boldsymbol{x}^*$ 作为 $\begin{cases}\boldsymbol{Ax}\leqslant\boldsymbol{b}\\\boldsymbol{x}\geqslant 0\end{cases}$ 在最大最小意义下的最优解。

【例 5-8】 对例 5-6 进行最大最小法求解。

解：首先编写目标函数的 M 文件：

```
function f = ex58(x)
f(1) = - 3 * x(1) + 4 * x(2);
f(2) = - 5 * x(1) - 2 * x(2);
```

然后编写 MATLAB 代码如下：

```
clear all
clc
x0 = [1;1];
```

```
A = [2, - 3;3,1];
b = [19;11];
lb = zeros(2,1);
[x,fval] = fminimax('ex58',x0,A,b,[],[],lb,[])
```

结果输出如下：

```
x =
    3.6667
    0.0000
fval =
  - 11.0000   - 18.3333
```

多目标线性规划是优化问题的一种，由于其存在多个目标，要求各目标同时取得较优的值，使得求解的方法与过程都相对复杂。通过将目标函数进行模糊化处理，可将多目标问题转化为单目标，借助工具软件，从而达到较易求解的目标。

5.5 线性规划实例

在企业的各项管理活动中，如计划、生产、运输、技术等问题，线性规划是指从各种限制条件的组合中，选出最为合理的计算方法，建立线性规划模型从而求得最佳结果。本节将从企业活动中的实例出发讲解线性规划的求解。

5.5.1 生产决策问题

【例 5-9】 某厂生产甲、乙两种产品，已知制成一吨产品甲需要资源 A 5t，资源 B $4m^3$，资源 C 1 个单位；制成 1t 产品乙需要资源 A 2t，资源 B $6m^3$，资源 C 7 个单位。若 1t 产品甲和乙的经济价值分别为 9 万元和 4 万元，3 种资源的限制量分别为 85t、$210m^3$ 和 250 个单位。试分析应生产这两种产品各多少吨才能使创造的总经济价值最高？

解：这里可以令生产产品甲的数量为 x_1，生产产品乙的数量为 x_2。根据题意，编码代码如下：

```
clear all
clc
f = [ - 9; - 4];
A = [5 2
     4 7
     1 6];
b = [85; 210; 250];
lb = zeros(2,1);
```

然后调用 linprog 函数：

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,[],[],lb)
```

最优化结果如下：

```
Optimization terminated.

x =
    6.4815
   26.2963
fval =
 - 163.5185
exitflag =
     1
output =
  包含以下字段的 struct:
         iterations: 5
          algorithm: 'interior - point - legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
    constrviolation: 0
      firstorderopt: 1.6144e - 11

lambda =
  包含以下字段的 struct:
    ineqlin: [3 × 1 double]
      eqlin: [0 × 1 double]
      upper: [2 × 1 double]
      lower: [2 × 1 double]
```

由上可知，生产甲种产品 6.4815t、乙种产品 26.2963t 可使创造的总经济价值最高，最高经济价值为 163.5185 万元。exitflag＝1 表示过程正常收敛于解 x 处。

5.5.2 工作人员计划安排问题

【例 5-10】 某昼夜服务的公共交通系统每天各时间段（每 4 小时为一个时间段）所需的值班人数如表 5-1 所示，这些值班人员在某一时段开始上班后要连续工作 8 小时（包括轮流用餐时间），问该公共交通系统至少需要多少名工作人员才能满足值班的需要？

表 5-1　各时段所需值班人数

班次	时间段	所需人数
1	5:00～9:00	40
2	9:00～13:00	35
3	13:00～17:00	60
4	17:00～21:00	70
5	21:00～1:00	45
6	1:00～5:00	30

解：这里可以设 x_i 为第 i 个时段开始上班的人员数。

根据题意，编写代码如下：

```
clear all
clc
f = [1;1;1;1;1;1];
A = [ - 1 0 0 0 0  - 1
   - 1  - 1 0 0 0 0
   0  - 1  - 1 0 0 0
   0 0  - 1  - 1 0 0
   0 0 0  - 1  - 1 0
   0 0 0 0  - 1  - 1];
b = [ - 40; - 35; - 60; - 70; - 45; - 30];
lb = zeros(6,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,[],[],lb)
```

最优化结果如下：

```
Optimization terminated.

x =
   20.6959
   18.1954
   41.8046
   30.6180
   14.3820
   19.3041
fval =
  145.0000
exitflag =
     1
output =
  包含以下字段的 struct:
         iterations: 5
          algorithm: 'interior - point - legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
    constrviolation: 0
      firstorderopt: 1.8243e - 07

lambda =
  包含以下字段的 struct:
    ineqlin: [6 × 1 double]
      eqlin: [0 × 1 double]
```

```
        upper: [6×1 double]
        lower: [6×1 double]
```

可见，只要6个时段分别安排21人、18人、42人、31人、14人和19人就可以满足值班的需要，共计145人，并且计算结果 exitflag =1 是收敛的。

5.5.3 投资问题

【例 5-11】 某单位有一批资金用于4个工程项目的投资，用于各工程项目时所得的净收益(投入资金的百分比)如表 5-2 所示。

表 5-2 工程项目收益

工程项目	A	B	C	D
收益(%)	18	10	9	12

由于某种原因，决定用于项目A的投资不大于其他各项投资之和；而用于项目B和C的投资要大于项目D的投资。试确定使该单位收益最大的投资分配方案。

解：这里可以用 x_1、x_2、x_3 和 x_4 分别代表用于项目A、B、C和D的投资百分数，由于各项目的投资百分数之和必须等于100%，所以 $x_1+x_2+x_3+x_4=1$

根据题意，编写代码如下：

```
clera all
clc
f = [-0.18;-0.1;-0.09;-0.12];
A = [1 -1 -1 -1
      0 -1 -1 1];
b = [0; 0];
Aeq=[1 1 1 1];
beq=[1];
lb = zeros(4,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,Aeq,beq,lb)
```

结果如下：

```
Optimization terminated.

x =
    0.5000
    0.2500
    0.0000
    0.2500
fval =
```

```
   - 0.1450
exitflag =
     1
output =
  包含以下字段的 struct:
         iterations: 9
          algorithm: 'interior - point - legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
     constrviolation: 4.4409e - 16
       firstorderopt: 2.4035e - 10

lambda =
  包含以下字段的 struct:
    ineqlin: [2 × 1 double]
      eqlin: 0.1450
      upper: [4 × 1 double]
      lower: [4 × 1 double]
```

上面的结果说明，项目 A、B、C、D 投入资金的百分比分别为 50%、25%、0、25%时，该单位收益最大。

5.5.4 工件加工任务分配问题

【例 5-12】 某车间有两台机床甲和乙，可用于加工 3 种工件。假定这两台机床的可用台时数分别为 500 和 800，3 种工件的数量分别为 300、600 和 700，且已知用两台不同机床加工单位数量的不同工件所需的台时数和加工费用如表 5-3 所示。问怎样分配机床的加工任务，才能既满足加工工件的要求，又使总加工费用最低？

表 5-3 机床加工情况

机床类型	单位工作所需加工台时数/小时			单位工件的加工费用/元			可用台时数/小时
	工件 1	工件 2	工件 3	工件 1	工件 2	工件 3	
甲	0.4	1.1	1.4	13	11	12	600
乙	0.5	1.2	1.3	13	10	9	900

解：这里可以设在甲机床上加工工件 1、2 和 3 的数量分别为 x_1、x_2和 x_3，在乙机床上加工工件 1、2 和 3 的数量分别为 x_4、x_5和 x_6。根据 3 种工种的数量限制，则有

$$x_1 + x_4 = 400 \quad (对工件 1)$$
$$x_2 + x_5 = 600 \quad (对工件 2)$$
$$x_3 + x_6 = 500 \quad (对工件 3)$$

根据题意，编写代码如下：

```
clear all
clc
```

可见，在甲机床上加工 349 个工件 1、600 个工件 2，在乙机床上加工 700 个工件 3 时，可在满足条件的情况下使总加工费用最小。最小费用为 17442 元，收敛正常。


```
f = [13;11;12;13;10;9];
A = [0.4 1.1 1.4 0 0 0
      0 0 0 0.5 1.2 1.3];
b = [500; 800];
Aeq = [1 0 0 1 0 0
     0 1 0 0 1 0
     0 0 1 0 0 1];
beq = [300 600 700];
lb = zeros(6,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,Aeq,beq,lb)
```

结果如下：

```
x =
  349.3976
  599.9658
    0.0080
    0.0003
    0.0340
  699.9920

fval =
   1.7442e+04
exitflag =
    -2

output =
  包含以下字段的 struct:
         iterations: 8
          algorithm: 'interior-point-legacy'
       cgiterations: 0
            message: 'Exiting: One or more of the residuals, duality gap, or total relative
error…'
    constrviolation: 299.7327
      firstorderopt: 2.7307e+21

lambda =
  包含以下字段的 struct:
    ineqlin: [2×1 double]
      eqlin: [3×1 double]
      upper: [6×1 double]
      lower: [6×1 double]
```

可见，在甲机床上加工 349 个工件 1、600 个工件 2，在乙机床上加工 700 个工件 3 时，可在满足条件的情况下使总加工费用最小。最小费用为 17442 元，收敛正常。

5.5.5 厂址选择问题

【例 5-13】 A、B、C 三地，每地都出产一定数量的产品，也消耗一定数量的原料，如表 5-4 所示。已知制成每吨产品需 3t 原料，各地之间的距离如下：A—B，150km；A—C，100km；B—C，200km。假定每万吨原料运输 1km 的运价是 5000 元，每万吨产品运输 1km 的运价是 6000 元。由示于地区条件的差异，在不同地点设厂的生产费用也不同。试问究竟在哪些地方设厂，规模多大，才能使总费用最小？另外，由于其他条件限制，在 B 处建厂的规模（生产的产品数量）不能超过 7 万 t。

表 5-4 A、B、C 三地出产产品、消耗原料情况

地点	年产原料/万 t	年销产品/万 t	生产费用/(万元/万 t)
A	22	7	150
B	16	14	120
C	25	0	100

解：这里可令 x_{ij} 为由 i 地运到 j 地的原料数量（万 t），y_{ij} 为由 i 地运往 j 地的产品数量（万 t），$i,j=1,2,3$（分别对应 A、B、C 三地）。

根据题意，编写代码如下：

```
clear all
clc
f = [75;75;50;50;100;100;150;240;210;120;160;220];
A=[1 -1 1 -1 0 0 3 3 0 0 0 0
  -1 1 0 0 1 -1 0 0 3 3 0 0
   0 0 -1 1 -1 1 0 0 0 0 3 3
   0 0 0 0 0 0 0 0 1 1 0 0];
b=[22;16;25;7];
Aeq=[0 0 0 0 0 0 1 0 1 0 1 0
     0 0 0 0 0 0 0 1 0 1 0 1];
beq=[7;14];
lb = zeros(12,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,Aeq,beq,lb)
```

结果如下：

```
Optimization terminated.

x =
    0.0000
    0.0000
    0.0000
    0.0000
```

```
     0.0000
     0.0000
     7.0000
     0.3333
     0.0000
     5.3333
     0.0000
     8.3333
fval =
   3.6033e+03
exitflag =
     1

output =
  包含以下字段的 struct:
         iterations: 8
          algorithm: 'interior-point-legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
    constrviolation: 7.1054e-15
      firstorderopt: 1.4588e-11

lambda =

  包含以下字段的 struct:
    ineqlin: [4×1 double]
      eqlin: [2×1 double]
      upper: [12×1 double]
      lower: [12×1 double]
```

可见要使总费用最小，A、B、C 三地的建厂规模分别为 7 万 t、5.3 万 t 和 8.3 万 t。最小总费用为 3603.3 万元。

5.5.6 确定职工编制问题

【例 5-14】 某工厂每日 8 小时的产量不低于 1800 件。为了进行质量控制，计划聘请两个不同水平的检验员。一级检验员的速度为 25 件/h，正确率 98%，计时工资 5 元/h；二级检验员的速度为 15 件/h，正确率 95%，计时工资 3 元/h。检验员每错检一次，工厂要损失 2 元。现有可供厂方聘请的检验员人数为一级 6 人和二级 7 人。为使总检验费用最省，该工厂应聘请一级、二级检验员各多少名？

解：可以设需要一级和二级检验员的人数分别为 x_1 名和 x_2 名。由题意，编写代码如下：

```
clear all
clc
f = [40;36];
```

```
A=[1 0
   0 1
   -5 -3];
b=[6;7;-45];
lb = zeros(2,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,[],[],lb)
```

结果如下：

```
Optimization terminated.

x =
    6.0000
    5.0000
fval =
  420.0000
exitflag =
     1

output =
  包含以下字段的 struct:
         iterations: 5
          algorithm: 'interior-point-legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
    constrviolation: 0
      firstorderopt: 3.1060e-07

lambda =
  包含以下字段的 struct:

    ineqlin: [3×1 double]
      eqlin: [0×1 double]
      upper: [2×1 double]
      lower: [2×1 double]
```

可见，聘请一级检验员 6 名、二级检验员 5 名可使总检验费用最省，约为 420.00 元，计算收敛。

5.5.7 生产计划的最优化问题

【例 5-15】 某工厂生产 A 和 B 两种产品，它们需要经过 3 种设备的加工，其工时如表 5-5 所示。设备一、二和三每天可使用的时间分别不超过 11 小时、9 小时和 12 小时。产品 A 和 B 的利润随市场的需求有所波动，如果预测未来某个时期内 A 和 B 的利润分

别为 6000 元/t 和 5000 元/t，试问在哪个时期内，每天应生产产品 A、B 各多少 t，才能使工厂获利最大？

表 5-5　生产产品工时

产品	设备一	设备二	设备三
A/(小时/t)	7	5	6
B/(小时/t)	3	3	2
设备每天最多可工作时数/h	12	10	8

解：这里可以设每天应安排生产产品 A 和 B 分别为 x_1t 和 x_2t。

由题意，编写代码如下：

```
clear all
clc
f = [-6;-5];
A=[7 3
   5 3
   6 2];
b=[12;10;8];
lb = zeros(2,1);
```

然后调用 linprog 函数

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,[],[],lb)
```

结果如下：

```
Optimization terminated.

x =
    0.0000
    3.3333
fval =
  -16.6667
exitflag =
     1
output =
  包含以下字段的 struct:
         iterations: 5
          algorithm: 'interior-point-legacy'
       cgiterations: 0
            message: 'Optimization terminated.'
     constrviolation: 0
      firstorderopt: 7.4706e-09
lambda =
```

```
包含以下字段的 struct:
  ineqlin: [3×1 double]
    eqlin: [0×1 double]
    upper: [2×1 double]
    lower: [2×1 double]
```

所以每天生产 A 产品 0t、B 产品 3.3333t 可使工厂获得最大利润 16666.7 元/t。

本章小结

线性规划是运筹学中研究较早、发展较快、应用广泛、方法较成熟的一个重要分支，它是辅助人们进行科学管理的一种数学方法。在经济管理、交通运输、工农业生产等经济活动中，提高经济效果是人们不可缺少的需求，而线性规划所研究的就是在一定条件下，合理安排人力物力等资源，使经济效果达到最好。

本章首先介绍了线性规划的概念和标准形式，然后对 MATLAB 线性规划函数进行了举例说明，最后对线性规划的求解方法和线性规划实例做了详细分析。