

OR 第四周上机作业

20123101 李昀哲

2022.12.27

1、给定：A 里面包含一个单位矩阵，利用对偶单纯形法进行求解。

函数接口：[x_opt,fx_opt,iter] = DSimplex_eye(A,b,c)

其中 x_opt 为最优解，fx_opt 为最优函数值，iter 为迭代次数。

例：
 $A = \begin{bmatrix} -1 & -2 & -1 & 1 & 0 \\ -2 & 1 & -3 & 0 & 1 \end{bmatrix};$
 $b = [-3 \ -4]';$
 $c = [-2 \ -3 \ -4 \ 0 \ 0]';$

准备部分：

使用 PPT 上的一道例题作为测试

```
% Operational Research
% @author 李昀哲 20123101
% Dec 27, 2022


---


%% 数据准备
A=[-1,-2,-1,1,0;
   -2,1,-3,0,1];
b=[-3,-4]';
c=[-2,-3,-4,0,0]';
m=2;n=5;
format rat;
```

DSimplex_eye 函数：

格式、容器等的准备

```
% Operational Research
% @author 李昀哲 20123101
% Dec 27, 2022


---


function [x_opt,fx_opt,iter]=DSimplex_eye(A,b,c,n,m)


---


%% 选出初始单位阵
iter=0;
% n列中选出m列
origin_eye=nchoosek(1:1:n,m);
[row,col]=size(origin_eye);
```

找单位阵：

```

% 遍历所有行找单位阵
for i = 1:row
    tmp = origin_eye(i,:);
    init_eye=A(:,tmp);
%% 确定是否找到eye
    a = init_eye;
    flag=1;
    [row_a,col_a]=size(a);
    tmp_a=zeros(row_a);%对应的列只有一个1,其他全是0, 否则flag=false;
    for i1 =1:row_a
        for j1 = 1:col_a
            if a(i1,j1)==1 && tmp_a(j1)==0
                tmp_a(j1)=1;
            elseif a(i1,j1)==0
                continue;
            else
                flag=0;
                break;
            end
        end
    end
    for i1 =1:row_a
        if tmp_a(i1)==0
            flag=0;
            break;
        end
    end
    found_eye = flag;
%%
    if found_eye == 1
        basic_vector = tmp;
        break;
    end
end
end

```

开始计算：

```

%% 循环迭代求解
sigma = zeros(n);
CB = (zeros(m))';
tmp_A = A;
found_eye = 1;
while found_eye
    % 更新b
    min_b = 0;
    for i = 1:m
        if b(i) < min_b
            min_b = b(i);
            min_b_pos = basic_vector(i);
            min_b_pos_tmp = i;
        end
    end
end

```

```

% >=0表示已经找到最优解
if min_b >= 0
    x_opt = (zeros(n))';
    x_opt(basic_vector,1) = b(:,1);
    fx_opt = 0;
    for pos_x = 1:n
        fx_opt=fx_opt+c(pos_x)*x_opt(pos_x,1);
    end
    found_eye = 0;
    break;
else
    % 更新sigma,用sigma/y,找最小的, 确定换入基
    for pos_c = 1:n
        sigma(pos_c,1)=c(pos_c);
        for pos_tmp = 1:m
            sigma(pos_c,1)=sigma(pos_c,1)-CB(pos_tmp,1)*tmp_A(pos_tmp,pos_c);
        end
    end
    min_sigma=-1;

    for i=1:n
        if i==min_b_pos
            continue;
        end
        flag_basic = 1;
        for j=1:m
            if basic_vector(j)==i
                flag_basic=0;
                break;
            end
        end
        if flag_basic==1 && tmp_A(min_b_pos_tmp,i)<0
            if min_sigma==-1||min_sigma>sigma(i,1)/tmp_A(min_b_pos_tmp,i)
                min_sigma=sigma(i,1)/tmp_A(min_b_pos_tmp,i);
                min_sigma_pos=i;
            end
        end
    end
end
end

```

```

if min_sigma== -1
    found_eye=0;
    disp("存在无界解!");
    break;
end
out_x_pos=min_b_pos; % 换出基
in_x_pos=min_sigma_pos; % 换入基
for in_out = 1:m
    if basic_vector(in_out)==out_x_pos
        basic_vector(in_out)=in_x_pos;
        CB(in_out,1)=c(in_x_pos);
        break;
    end
end
% 对矩阵执行初等变换
tmp_beishu=tmp_A(in_out,basic_vector(in_out));
for change_1 = 1:n
    tmp_A(in_out,change_1)=tmp_A(in_out,change_1)/tmp_beishu;
end
b(in_out,1)=b(in_out,1)/tmp_beishu;

    % 矩阵内部的初等变换
    for change_pos = 1:m
        if change_pos~=in_out
            beishu=tmp_A(change_pos,in_x_pos);
            for tt=1:n
                tmp_A(change_pos,tt)=tmp_A(change_pos,tt)-beishu*tmp_A(in_out,tt);
            end
            b(change_pos,1)=b(change_pos,1)-beishu*b(in_out,1);
        end
    end
end
iter=iter+1;
end
end

```

函数调用：

```

%% 函数调用
[x_opt,fx_opt,iter]=DSimplex_eye(A,b,c,n,m);
x_opt, fx_opt, iter

```

运行结果：

```

x_opt =

    11/5         0         0         0         0
     2/5         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0
         0         0         0         0         0

```

```

fx_opt =

    -28/5

```

```

iter =

     2

```