전기전자공학수학

Computer Simulation HW3

2018142023 조성민

16.21 Write a MATLAB routine that implements the two-phase simplex method. It may be useful to use the MATLAB function of Exercise 16.20. Test the routine on the problem in Example 16.5.

※참고※

Example 16.5 Consider solving the following LP problem using the revised simplex method:

$$\begin{aligned} \text{maximize} \quad & 3x_1 + 5x_2 \\ \text{subject to} \quad & x_1 + x_2 \leq 4 \\ & 5x_1 + 3x_2 \geq 8 \\ & x_1, x_2 \geq 0. \end{aligned}$$

16.20 Write a simple MATLAB function that implements the simplex algorithm. The inputs are $\boldsymbol{c}$, $\boldsymbol{A}$, $\boldsymbol{b}$, and $\boldsymbol{v}$, where $\boldsymbol{v}$ is the vector of indices of basic columns. Assume that the augmented matrix $[\boldsymbol{A}, \boldsymbol{b}]$ is already in canonical form; that is, the v_ith column of $\boldsymbol{A}$ is $[0, \ldots, 1, \ldots, 0]^\top$, where 1 occurs in the ith position. The function should output the final solution and the vector of indices of basic columns. Test the MATLAB function on the problem in Example 16.2.

Example 16.2 Consider the following linear program (see also Exercise 15.10):

$$\begin{aligned} \text{maximize} \quad & 2x_1 + 5x_2 \\ \text{subject to} \quad & x_1 \leq 4 \\ & x_2 \leq 6 \\ & x_1 + x_2 \leq 8 \\ & x_1, x_2 \geq 0. \end{aligned}$$

We solve this problem using the simplex method.

<Theory>

Simplex algorithm

1. Form a canonical augmented matrix corresponding to an initial basic feasible solution.
2. Calculate the reduced cost coefficients corresponding to the nonbasic variables.
3. If $r_j \geq 0$ for all j, stop - the current basic feasible solution is optimal.
4. Select a q such that $r_q < 0$.
5. If no $y_{iq} > 0$, stop - the problem is unbounded; else, calculate $p = \mathrm{arg}min_i \{ y_{i0}/y_{iq} : y_{iq} > 0 \}$.
6. Update the canonical augmented matrix by pivoting about the $(p,q)^{th}$ element.\
7. Go to step 2.

Two-Phase Simplex method

-Delete the columns corresponding to the artificial variables in the last tableau in phase 1 and revert back to the original objective function.

ex)

$$
\begin{array}{cccccc}
 & \boldsymbol{a}_1 & \boldsymbol{a}_2 & \boldsymbol{a}_3 & \boldsymbol{a}_4 & \boldsymbol{b} \\
 & 1 & 0 & -\frac{2}{7} & \frac{1}{7} & \frac{18}{7} \\
 & 0 & 1 & \frac{1}{14} & -\frac{2}{7} & \frac{6}{7} \\
\boldsymbol{c}^\top & 2 & 3 & 0 & 0 & 0
\end{array}
\Rightarrow
\begin{array}{ccccc}
1 & 0 & -\frac{2}{7} & \frac{1}{7} & \frac{18}{7} \\
0 & 1 & \frac{1}{14} & -\frac{2}{7} & \frac{6}{7} \\
0 & 0 & \frac{5}{14} & \frac{4}{7} & -\frac{54}{7}
\end{array}
$$

Canonical form Optimal

<Implementation>

maximize $3x_1 + 5x_2$

subject to $x_1 + x_2 \leq 4$

$5x_1 + 3x_2 \geq 8$

$x_1, x_2 \geq 0$

Minimize $c^T x$ subject to $Ax = b, x \geq 0$,

where $A = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 5 & 3 & 0 & -1 \end{bmatrix}$, $b = [4\ 8]^T$, $c = [-3\ -5\ 0\ 0]^T$

<MATLAB code>

1. foptions

```
function OPTIONS=foptions(parain)
if nargin<1
   parain = [];
end
sizep=length(parain);
OPTIONS=zeros(1,18);
OPTIONS(1:sizep)=parain(1:sizep);
default_options=[0,1e-4,1e-4,1e-6,0,0,0,0,0,0,0,0,0,0,0,1e-8,0.1,0];
OPTIONS=OPTIONS+(OPTIONS==0).*default_options;
```

2. pivot

```
function Mnew=pivot(M,p,q)
for i=1:size(M,1)
    if i==p
        Mnew(p,:)=M(p,:)/M(p,q);
    else
        Mnew(i,:)=M(i,:)-M(p,:)*(M(i,q)/M(p,q));
    end
end
```

3. simplex

```
function [x,v]=simplex(c,A,b,v,options)
if nargin ~= 5
    options = [];
    if nargin ~= 4
        disp('Wrong number of arguments.');
        return;
    end
end
format compact;
options = foptions(options);
print = options(1);
n=length(c);
m=length(b);
cB=c(v(:));
r = c.'-cB.'*A;
cost = -cB.'*b;
tabl=[A b;r cost];
if print
    disp('Initial tableau:');
    disp(tabl);
end
while ones(1,n)*(r.' >= zeros(n,1)) ~= n
    if options(5) == 0
        [r_q,q] = min(r);
    else
        q=1;
        while r(q) >= 0
            q=q+1;
        end
    end
    min_ratio = inf;
    p=0;
    for i=1:m
        if tabl(i,q)>0
            if tabl(i,n+1)/tabl(i,q) < min_ratio
                min_ratio = tabl(i,n+1)/tabl(i,q);
                p = i;
            end
        end
    end
    if p == 0
        disp('Problem unbounded');
        break;
    end
    tabl=pivot(tabl,p,q);
    if print
        fprintf("\n");
        disp('Pivot point:');
        disp([p,q]);
        fprintf("\n");
        disp('New tableau:');
        disp(tabl);
    end
    v(p) = q;
    r = tabl(m+1,1:n);
end
x=zeros(n,1);
x(v(:))=tabl(1:m,n+1);
```

4. tpsimplex

```
function [x,v]=tpsimplex(c,A,b,options)
if nargin ~= 4
    options = [];
        if nargin ~= 3
        disp('Wrong number of arguments.');
        return;
    end
end
clc;
format compact;
options = foptions(options);
print = options(1);
n=length(c);
m=length(b);
if print
    fprintf("₩n");
    disp('Phase 1');
end
v=n*ones(m,1);
for i=1:m
    v(i)=v(i)+i;
end
[x,v]=simplex([zeros(n,1);ones(m,1)],[A eye(m)],b,v,options);
if all(v<=n)
    if print
        fprintf("₩n");
        disp('Phase 2');
        disp('Basic columns:')
        disp(v.')
    end
    Binv=inv(A(:,[v]));
    A=Binv*A;
    b=Binv*b;
    [x,v]=simplex(c,A,b,v,options);
    if print
        fprintf("₩n");
        disp('Final solution:');
        disp(x.');
    end
else
    disp('Terminating: problem has no feasible solution.');
end
```

<Result & Analysis>

-입력

```
>> A=[1 1 1 0; 5 3 0 -1];
>> b=[4;8];
>> c=[-3;-5;0;0];
>> options(1)=1;
>> format rat;
>> tpsimplex(c,A,b,options);
```

-Phase 1

Phase 1
Initial tableau:

1	1	1	0	1	0	4
5	3	0	-1	0	1	8
-6	-4	-1	1	0	0	-12

Pivot point:

2	1

New tableau:

0	2/5	1	1/5	1	-1/5	12/5
1	3/5	0	-1/5	0	1/5	8/5
0	-2/5	-1	-1/5	0	6/5	-12/5

Pivot point:

1	3

New tableau:

0	2/5	1	1/5	1	-1/5	12/5
1	3/5	0	-1/5	0	1/5	8/5
0	*	0	*	1	1	*

Pivot point:

2	2

New tableau:

-2/3	0	1	1/3	1	-1/3	4/3
5/3	1	0	-1/3	0	1/3	8/3
*	0	0	*	1	1	*

Pivot point:

1	4

New tableau:

-2	0	3	1	3	-1	4
1	1	1	0	1	0	4
*	0	*	0	1	1	*

-Phase 2

Phase 2
Basic columns:

4	2

Initial tableau:

-2	0	3	1	4
1	1	1	0	4
2	0	5	0	20

Final solution:

0	4	0	4