Linear and Convex Optimization Homework 12

Qiu Yihang, 2021/12/22-12/25

0. Preparation

Complete newton.py and LP.py. The completed code (with `newton_eq`, `centering_step` and `barrier` function) is enclosed in the zip file.

1.(a) *Solution:*

$$\nabla f(\boldsymbol{x}) = (e^{x_1}, 2e^{2x_2}, 2e^{2x_3})^T, \nabla^2 f(\boldsymbol{x}) = \begin{pmatrix} e^{x_1} & 0 & 0 \\ 0 & 4e^{2x_2} & 0 \\ 0 & 0 & 4e^{2x_3} \end{pmatrix}, \boldsymbol{A} = (1,1,1), b = 1.$$

Let the Newton direction $\boldsymbol{d} = (d_1, d_2, d_3)^T$. Thus, the KKT system is

$$\begin{pmatrix} e^{x_1} & 0 & 0 & 1 \\ 0 & 4e^{2x_2} & 0 & 1 \\ 0 & 0 & 4e^{2x_3} & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{d} \\ \lambda \end{pmatrix} = \begin{pmatrix} -e^{x_1} \\ -2e^{2x_2} \\ -2e^{2x_3} \\ 0 \end{pmatrix},$$

$$\text{i.e.} \begin{cases} e^{x_1} d_1 + \lambda = -e^{x_1} \\ 4e^{2x_2} d_2 + \lambda = -2e^{2x_2} \\ 4e^{2x_3} d_3 + \lambda = -2e^{2x_3} \\ d_1 + d_2 + d_3 = 0 \end{cases} \Rightarrow \begin{cases} d_1 = -1 + \frac{8e^{2x_2+2x_3}}{e^{x_1+2x_2}+e^{x_1+2x_3}+4e^{2x_2+2x_3}} \\ d_2 = -\frac{1}{2} + \frac{2e^{x_1+2x_3}}{e^{x_1+2x_2}+e^{x_1+2x_3}+4e^{2x_2+2x_3}} \\ d_3 = -\frac{1}{2} + \frac{2e^{x_1+2x_2}}{e^{x_1+2x_2}+e^{x_1+2x_3}+4e^{2x_2+2x_3}} \\ \lambda = -\frac{8e^{x_1+2x_2+2x_3}}{e^{x_1+2x_2}+e^{x_1+2x_3}+4e^{2x_2+2x_3}} \end{cases}$$

Thus, the Newton direction at a feasible $\boldsymbol{x} = (x_1, x_2, x_3)^T$ is

$$\boldsymbol{d} = \begin{pmatrix} -1 + \dfrac{8e^{2x_2+2x_3}}{e^{x_1+2x_2} + e^{x_1+2x_3} + 4e^{2x_2+2x_3}} \\ -\dfrac{1}{2} + \dfrac{2e^{x_1+2x_3}}{e^{x_1+2x_2} + e^{x_1+2x_3} + 4e^{2x_2+2x_3}} \\ -\dfrac{1}{2} + \dfrac{2e^{x_1+2x_2}}{e^{x_1+2x_2} + e^{x_1+2x_3} + 4e^{2x_2+2x_3}} \end{pmatrix}. \quad \blacksquare$$

(b) *Solution:*

The code (p1.py) is enclosed in the zip file. The output with initial point $(0,1,0)^T$ is as follows.

```
In [2]: runfile('D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12/
p1.py', wdir='D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12')

iteration 0: [0. 1. 0.]
iteration 1: [ 0.55783402  0.55270748 -0.1105415 ]
iteration 2: [0.74171111 0.22388047 0.03440841]
iteration 3: [0.83735858 0.09139269 0.07124873]
iteration 4: [0.8464719  0.07685719 0.07667091]
iteration 5: [0.84657358 0.07671322 0.0767132 ]
iteration 6: [0.84657359 0.0767132  0.0767132 ]
optimal value: 4.663287963194248
```

Fig.01. Results of Program 1

2.(a) ***Solution:***

The approximating equality constrained problem is

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \ \boldsymbol{c}^T\boldsymbol{x} - \frac{1}{t}\sum_{i=1}^{n}\log x_i$$

$$\text{s.t.} \ \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \qquad \blacksquare$$

(b) ***Solution:***

Let the objective function be f. Then $f(\boldsymbol{x}) = \boldsymbol{c}^T\boldsymbol{x} - \frac{1}{t}\sum_{i=1}^{n}\log x_i$.

The gradient and the Hessian matrix are as follows.

$$\nabla f(\boldsymbol{x}) = \boldsymbol{c} - \frac{1}{t}\left(\frac{1}{x_1},\frac{1}{x_2},\dots\frac{1}{x_n}\right), \nabla^2 f(\boldsymbol{x}) = \frac{1}{t}\operatorname{diag}\left(\frac{1}{x_1^2},\frac{1}{x_2^2},\dots\frac{1}{x_n^2}\right).$$

$\blacksquare$

(c) ***Solution:*** Complete LP.py. The code is enclosed in the zip file.

(d) ***Solution:***

Convert the original form.

$$\min_{\boldsymbol{x}\in\mathbb{R}^4} \ -x_1 - 3x_2$$

$$\text{s.t.} \ \ x_1 + x_2 + x_3 = 6, \quad -x_1 + 2x_2 + x_4 = 8,$$

$$\boldsymbol{x} = (x_1, x_2, x_3, x_4) \geq 0.$$

The standard form is

$$\min_{\boldsymbol{x}\in\mathbb{R}^4} \ \boldsymbol{c}^T\boldsymbol{x}$$

$$\text{s.t.} \ \ \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$$

$$\boldsymbol{x} \geq \boldsymbol{0}$$

$$\text{where } \boldsymbol{c} = (-1,-3,0,0), \boldsymbol{A} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 2 & 0 & 1 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} 6 \\ 8 \end{pmatrix}.$$

Implement the barrier method. The code (p2.py) is enclosed in the zip file.

We can find a feasible initial point $\boldsymbol{x}_0 = (2,1,3,8)^T$. The output of p2.py is as follows.

```
In [1]: runfile('D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12/
p2.py', wdir='D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12')

iteration 0: [2 1 3 8]
iteration 1: [1.63307563 3.90402064 0.46290373 1.82503435]
iteration 2: [1.34600004 4.59597677 0.05802319 0.1540465 ]
iteration 3: [1.3343604  4.65966009 0.00597951 0.01504022]
iteration 4: [1.33343360e+00 4.66596660e+00 5.99794358e-04 1.50040181e-03]
iteration 5: [1.33334334e+00 4.66659667e+00 5.99948876e-05 1.49996563e-04]
iteration 6: [1.33333433e+00 4.66665967e+00 5.99937452e-06 1.49985317e-05]
iteration 7: [1.33333343e+00 4.66666597e+00 5.99939244e-07 1.49984906e-06]
iteration 8: [1.33333334e+00 4.66666660e+00 5.93972887e-08 1.48493250e-07]
iteration 9: [1.33333333e+00 4.66666666e+00 3.99866757e-09 9.99667352e-09]
optimal value: -15.333333320004444
```

Fig.02. Results of Program 2

The visualization of the projection of the barrier iterates onto x_1, x_2 coordinates is given below.

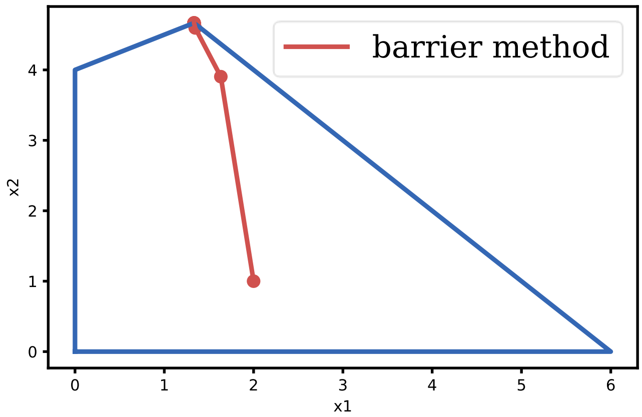

Fig.03. The Visualization of the Projection of the Barrier Iterates of Problem (2)

3.(a) ***Solution:***

The dual LP of 2(d) is

$$\max_{\boldsymbol{\mu}\in\mathbb{R}^4} \ -6\mu_1 - 8\mu_2$$

$$\textbf{s.t.} \ -\mu_1 + \mu_2 + \mu_3 = -1,$$

$$-\mu_1 - 2\mu_2 + \mu_4 = -3$$

$$\boldsymbol{\mu} = (\mu_1, \mu_2, \mu_3, \mu_4) \geq \mathbf{0}$$

Thus, the standard form of the dual LP of 2(d) is

$$\max_{\boldsymbol{\mu}\in\mathbb{R}^4} \ \boldsymbol{c}^T\boldsymbol{\mu}$$

$$\textbf{s.t.} \ \boldsymbol{A\mu} = \boldsymbol{b}$$

$$\boldsymbol{\mu} \geq \mathbf{0}$$

where $\boldsymbol{c} = (-6, -8, 0, 0), \boldsymbol{A} = \begin{pmatrix} -1 & 1 & 1 & 0 \\ -1 & -2 & 0 & 1 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} -1 \\ -3 \end{pmatrix}$. ■

(b) ***Solution:***

The symmetric dual LP is

$$\max_{\boldsymbol{\mu}\in\mathbb{R}^2} \ -6\mu_1 - 8\mu_2$$

$$\textbf{s.t.} \ -\mu_1 + \mu_2 \leq -1,$$

$$\mu_1 + 2\mu_2 \geq 3$$

$$\boldsymbol{\mu} = (\mu_1, \mu_2) \geq \mathbf{0}$$ ■

(c) ***Solution:***

The visualization of the feasible set and the level set of the objective function in 3(b) is shown in the Fig.04 (in the next page).

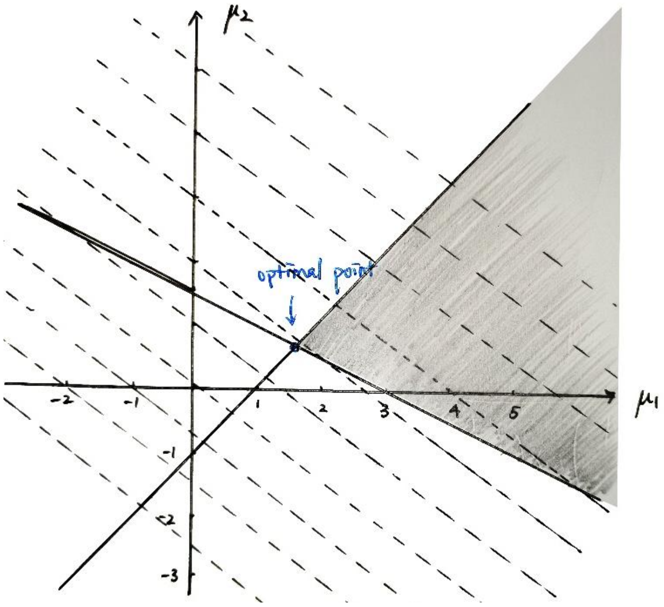

Fig.04. The Visualization Used to Solve Problem 3(b) Graphically

Thus, the dual optimal point is $\boldsymbol{\mu}^\star = \left(\frac{5}{3}, \frac{2}{3}\right)$ and the dual optimal value is $\psi(\boldsymbol{\mu}^\star) = -\frac{46}{3}$.

By slide 15 of §5-1, we know the primal optimal point is $\left(\frac{4}{3}, \frac{14}{3}\right)$ with primal optimal value $-\frac{46}{3}$.

The primal optimal value and the dual optimal value are **exactly the same**. ■

(d) ***Solution:***

The problem in 3(a) can be converted to

$$\min_{\boldsymbol{\mu}\in\mathbb{R}^4} \ \boldsymbol{c}^T\boldsymbol{\mu}$$

$$\textbf{s.t.} \ \boldsymbol{A}\boldsymbol{\mu} = \boldsymbol{b}$$

$$\boldsymbol{\mu} \geq \boldsymbol{0}$$

$$\text{where } \boldsymbol{c} = (6,8,0,0), \boldsymbol{A} = \begin{pmatrix} -1 & 1 & 1 & 0 \\ -1 & -2 & 0 & 1 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} -1 \\ -3 \end{pmatrix}.$$

Solve the problem implementing barrier method with initial point $\boldsymbol{\mu}_0 = (4,1,2,3)^T$. The output is as follows. Ignoring numerical errors, $\boldsymbol{\mu}^\star = \left(\frac{5}{3}, \frac{2}{3}\right)$ and $\psi(\boldsymbol{\mu}^\star) = -\frac{46}{3}$.

```
In [2]: runfile('D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12/
p3.py', wdir='D:/Textbooks/2021-2022-1/Linear and Convex Optimization/hw12')

iteration 0: [4. 1. 2. 3.]
iteration 1: [2.1602764  0.54793489 0.61234151 0.25614619]
iteration 2: [1.72344885 0.64915465 0.0742942  0.02175816]
iteration 3: [1.67237818 0.66488395 0.00749423 0.00214608]
iteration 4: [1.66723807e+00 6.66488125e-01 7.49943601e-04 2.14317865e-04]
iteration 5: [1.66672380e+00 6.66648812e-01 7.49918806e-05 2.14267532e-05]
iteration 6: [1.66667238e+00 6.66664881e-01 7.49923738e-06 2.14264427e-06]
iteration 7: [1.66666723e+00 6.66666490e-01 7.42466015e-07 2.12133296e-07]
iteration 8: [1.66666672e+00 6.66666651e-01 6.66555489e-08 1.90444589e-08]
iteration 9: [1.66666667e+00 6.66666666e-01 9.38122456e-10 2.68017088e-10]
dual optimal value: -15.333333335834926
```

Fig.05. Results of Program 2