ISE 3230: Systems Modeling and Optimization for Analytics

Homework 6

(Due Friday 12/04/2020 at 11:59pm)

Instructions

(1) For problems that require coding, your should use the CVXPY package in Python and *Gurobi* as the solver. Please include your code, the relevant part of the output, and comment on it. Do not include your code and/or its output as an appendix to your homework.

(2) If you are familiar with LaTex or a suitable markdown language for equations and willing to prepare your homework in Jupyter Notebook or Lab, you are welcome to do so.

(3) that the objectives of some of the problems are Maximization, hence the entering variable at each iteration of the simplex algorithm is the one with the most negative $z_j - c_j$ and the optimality condition is $z_j - c_j \geq 0$ for all j.

Problem 1. Consider the following problem.

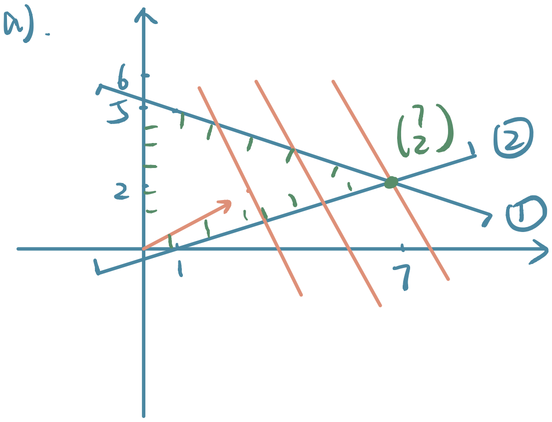
$$\begin{aligned} &\text{Maximize } z = 3x_1 + 2x_2 \\ &\text{subject to} \\ &\qquad x_1 + 2x_2 \leq 11 \\ &\qquad x_1 - 3x_2 \leq 1 \\ &\qquad x_1, x_2 \geq 0 \end{aligned}$$

(a) (5 pts) Solve this problem graphically.

(b) (10 pts) Write the canonical dual and solve the dual graphically. Compare the optimal objective values of the two problems.

(c) (5 pts) Solve the primal problem in Python using CVXPY and identify the optimal dual variables (shadow prices) on the program output.

(a).



Thus, the optimal solution is $\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 7 \\ 2 \end{pmatrix}$

$z_p = 25$.

(b)

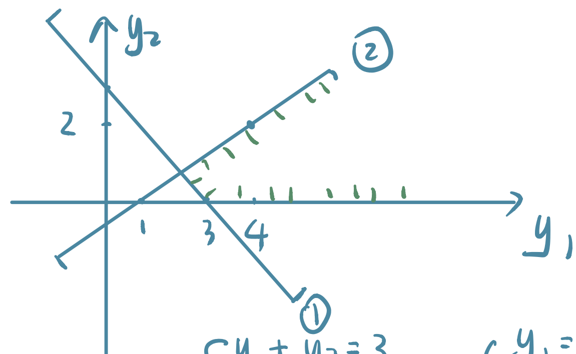
$$\min z = 11y_1 + y_2$$

$$\text{s.t. } y_1 + y_2 \geq 3 \quad ①$$

$$2y_1 - 3y_2 \geq 2 \quad ②$$

$$y_1, y_2 \geq 0$$

$\boxed{\text{Code Here}}$



$$\begin{cases} y_1 + y_2 = 3 \\ 2y_1 - 3y_2 = 2 \end{cases} \Rightarrow \begin{cases} y_1 = 2.2 \\ y_2 = 0.8 \end{cases}$$

$$z = 11 \cdot 2.2 + 0.8 = 25$$

```python
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Sat Dec  5 05:07:07 2020

@author: fionafei
"""


import cvxpy as cp

x = cp.Variable(2, nonneg = True) # vector variable



obj_func=3*x[0]+2*x[1]
#obj_func_neg=3*x[0]+2*x[1]

constraints = []
constraints.append(x[0]+2*x[1]<=11)
constraints.append(x[0]-3*x[1]<=1)
constraints.append(x[0]>=0)
constraints.append(x[1]>=0)



problem = cp.Problem(cp.Maximize(obj_func), constraints)
#problem = cp.Problem(cp.Minimize(obj_func_neg), constraints)

#problem.solve(solver=cp.CVXOPT,verbose = True)
#problem.solve(verbose = True)
problem.solve(solver=cp.GUROBI,verbose = True)

print("obj_func =")
print(obj_func.value)
#print(obj_func_neg.value)
print("x =")
print(x.value)

# Shadow prices or dual prices
print("optimal (x[0]+2*2<=11) dual variable", constraints[0].dual_value)
print("optimal (x[0]-3*2<=1) dual variable", constraints[1].dual_value)

# Reduced costs
print("reduced cost of the 1st primal variable x[0]", (-1)*constraints[2].dual_v
print("reduced cost of the 2nd primal variable x[1]", (-1)*constraints[3].dual_v

#output
Solved in 0 iterations and 0.01 seconds
Optimal objective -2.500000000e+01
```

```
obj_func =
25.0
x =
[7. 2.]
optimal (x[0]+2*2<=11) dual variable 2.2
optimal (x[0]-3*2<=1) dual variable 0.8
reduced cost of the 1st primal variable x[0] 0.0
reduced cost of the 2nd primal variable x[1] 0.0
```

Problem 2. Consider the following linear programming problem.

$$
\begin{aligned}
&\text{Maximize } z = x_1 + 2x_2 - 3x_3 \\
&\text{subject to} \\
&\qquad -3x_1 + x_2 + 2x_3 = 16 \quad y_1 \\
&\qquad 2x_1 + 4x_2 + 3x_3 \geq 20 \quad y_2 \\
&\qquad x_1 \geq 0 \\
&\qquad x_2 \leq 0 \\
&\qquad x_3 \text{ unrestricted}
\end{aligned}
$$

(a) (5 pts) Using the table provided in the duality lecture, write the general dual .

(b) (10 pts) Transform the given problem into the canonical form. Write the canonical dual and verify its equivalence to that found in part (a).

Problem 3. Consider the following linear programming problem.

$$
\begin{aligned}
&\text{Maximize } z = 3x_1 + 10x_2 + 5x_3 + 11x_4 + 6x_5 + 14x_6 \\
&\text{subject to} \\
&\qquad x_1 + 7x_2 + 3x_3 + 4x_4 + 2x_5 + 5x_6 = 42 \\
&\qquad x_j \geq 0, \text{ for all } j
\end{aligned}
$$

(a) (5 pts) Write the dual problem.

(b) (10 pts) Solve the dual problem by inspection.

Problem 4. Consider the following the following problem.

$$
\begin{aligned}
&\text{Maximize } z = x_1 + 2x_2 - 9x_3 + 8x_4 - 36x_5 \\
&\text{subject to} \\
&\qquad 2x_2 - x_3 + x_4 - 3x_5 \leq 40 \\
&\qquad x_1 - x_2 + 2x_4 - 2x_5 \leq 10 \\
&\qquad x_j \geq 0, \text{ for all } j
\end{aligned}
$$

Problem #2.

(a) general dual.

$\min z = 16y_1 + 20y_2$

s.t. $-3y_1 + 2y_2 \geq 1$

$y_1 + 4y_2 \leq 2$

$2y_1 + 3y_2 = -3$

y_1 unrestricted.

$y_2 \leq 0$

(b) canonical form

$x_2^* = -x_2 \geq 0$

$x_3 = x_3^+ - x_3^-$, $\quad x_3^+, x_3^- \geq 0$

$\max z = x_1 - 2x_2^* - 3x_3^+ + 3x_3^-$

s.t. $3x_1 + x_2^* - 2x_3^+ + 2x_3^- \leq -16 \qquad y_1$

$-3x_1 - x_2^* + 2x_3^+ - 2x_3^- \leq 16 \qquad y_2$

$-2x_1 + 4x_2^* - 3x_3^+ + 3x_3^- \leq -20 \qquad y_3$

$x_1, x_2^*, x_3^+, x_3^- \geq 0$.

$\min z = 16w_1 - 16w_2 - 20w_3$

s.t. $-3w_1 + 3w_2 - 2w_3 \geq 1$ ①

$-w_1 + w_2 + 4w_3 \geq -2$ ②

$2w_1 - 2w_2 - 3w_3 \geq -3$ ③

$-2w_1 + 2w_2 + 3w_3 \geq 3$ ④ combine.

$w_1, w_2, w_3 \geq 0$.

③ + ④:

$$2w_1 - 2w_2 - 3w_3 = -3.$$

let $w_1^* = w_2 - w_1$

$w_2^* = -w_3 \leq 3$.

Thus, the formulation from part (a) is equivalent with the formulation in part (b).

Problem #3

(a). $\min \quad z = 42y_1$

s.t. $y_1 \geq 3$	$y_1 \geq 3$
$7y_1 \geq 10$	$y_1 \geq \frac{10}{7}$
$3y_1 \geq 5$	$y_1 \geq \frac{5}{3}$
$4y_1 \geq 11$	$y_1 \geq \frac{11}{4}$
$2y_1 \geq 6$	$y_1 \geq 3$
$5y_1 \geq 14$.	$y_1 \geq \frac{14}{5}$

y_1 unrestricted.

(b). By inspection, in order to get minimal objective value.

$$y_1 = \min\{3, \underline{\frac{10}{7}}, \frac{5}{3}, \frac{11}{4}, 3, \frac{14}{5}\} = \frac{10}{7}.$$

$$\therefore y^* = \frac{10}{7}, \quad z = \cancel{42}^{6} \times \frac{10}{\cancel{7}} = 60.$$

Problem #4.

(a). $\min\ z = 40y_1 + 10y_2$

s.t. $y_2 \geq 1$ ①

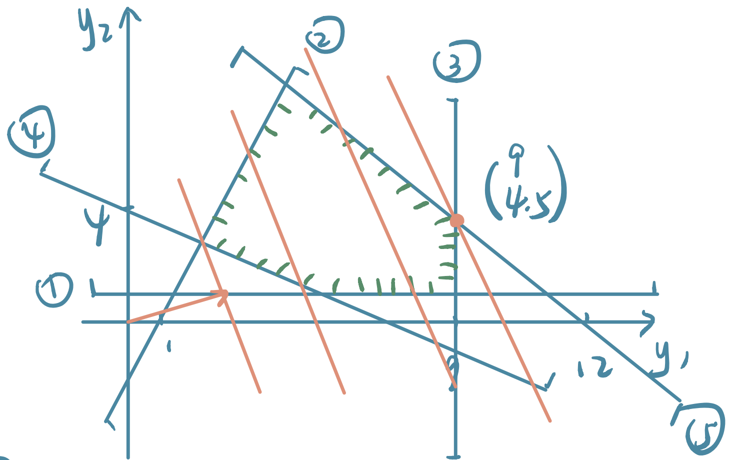
$2y_1 - y_2 \geq 2$ ②

$-y_1 \geq -9$ ③

$y_1 + 2y_2 \geq 8$ ④

$-3y_1 - 2y_2 \geq -36$ ⑤

$y_1 \geq 0,\ y_2 \geq 0.$



$$\begin{cases} y_1^* = 9 \\ y_2^* = 4.5 \end{cases}$$

(b)

$x_1(y_2 - 1) = 0$ ①

$x_2(2y_1 - y_2 - 2) = 0$ ②

$x_3(-y_1 + 9) = 0$ ③

$x_4(y_1 + 2y_2 - 8) = 0$ ④

$x_5(-3y_1 - 2y_2 + 36) = 0$ ⑤

$y_1(2x_2 - x_3 + x_4 - 3x_5 - 40) = 0$ ⑥

$y_2(x_1 - x_2 + 2x_4 - 2x_5 - 10) = 0$ ⑦

$x_1 = x_3 = x_5 = 0.$

⑥ + ⑦.

$$\begin{cases} 2x_2 + x_4 - 40 = 0 \\ -x_2 + 2x_4 - 10 = 0 \end{cases} \rightarrow \begin{cases} x_2 = 14 \\ x_4 = 12. \end{cases}$$

$\therefore z^* = \boxed{124}$

(a) (5 pts) Write the dual problem and solve it graphically.

(b) (10 pts) Using Complementary Slackness (CS) conditions and the optimal dual solution found in part (a), find an optimal solution to the primal problem.

Problem 5. Consider the following the following problem.

$$\begin{aligned}
&\text{Maximize } z = 3x_1 - x_2 + 6x_3 \\
&\text{subject to} \\
&\qquad 5x_1 + x_2 + 4x_3 \leq 42 \\
&\qquad 2x_1 - x_2 + 2x_3 \leq 18 \\
&\qquad x_1, x_2, x_3 \geq 0
\end{aligned}$$

(a) (5 pts) Write the dual problem.

(b) (10 pts) Solve the primal problem by the primal simplex algorithm. Identify both the optimal primal and optimal dual solutions from the final tableau.

(c) (5 pts) At each iteration in part (b), identify the dual solution and indicate which dual constraints are violated. Also, at each iteration, identify the 2×2 primal basis matrix and the 3×3 dual basis matrix.

(d) (5 pts) Write the complementary slackness conditions and verify that these conditions are satisfied by the optimal solutions found in part (b).

(e) (10 pts) Solve the problem in Python using CVXPY. Identify the shadow prices. Identify the reduced costs for the original variables. Interpret the two shadow prices and the three reduced costs.

Problem #5.

(a). $\min z = 42y_1 + 18y_2$

$$\text{s.t. } \begin{aligned} 5y_1+2y_2 &\geq 3 \\ y_1-y_2 &\geq -1 \\ 4y_1+2y_2 &\geq 6. \end{aligned} \quad \rightarrow \quad \begin{aligned} 5y_1+2y_2-w_1 &= 3 \\ y_1-y_2-w_2 &= -1 \\ 4y_1+2y_2-w_3 &= 6. \end{aligned}$$

$$y_1, y_2, w_1, w_2, w_3 \geq 0.$$

(b)

$\min z = -3x_1 + x_2 - 6x_3$

$$\text{s.t. } \begin{aligned} 5x_1+x_2+4x_3+x_4 &= 42 \\ 2x_1-x_2+2x_3+x_5 &= 18 \\ x_i \geq 0, \quad i &= 1,2,\dots,5 \end{aligned}$$

	z	x_1	x_2	x_3 ↓	x_4	x_5	RHS	
z	0	3	-1	6	0	0	0	
x_4	0	5	1	4	1	0	42	$42/4$
→ x_5	0	2	-1	2	0	1	18	$18/2 \rightarrow \min$

$B_P \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad B_D = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$

dual constraint violated.

$$5y_1+2y_2 \geq 3$$

	z	x_1	x_2	x_3	x_4	x_5	RHS	
z	0	-3	2	0	0	-3	-54	
x_4	0	1	3	0	1	-2	6	$6/3$
x_3	0	1	$-\frac{1}{2}$	1	0	$\frac{1}{2}$	9	$9/-\frac{1}{2}$

min = 2.

$B_P = \begin{pmatrix} 1 & 4 \\ 0 & 2 \end{pmatrix}$ $B_D = \begin{pmatrix} -1 & 0 & 2 \\ 0 & -1 & -1 \\ 0 & 0 & 2 \end{pmatrix}$

violate constraint $y_1 - y_2 \geq -1$.

	z	x_1	x_2	x_3	x_4	x_5	RHS
z	0	$-11/3$	0	0	$-2/3$	$-5/3$	-58
x_2	0	$1/3$	1	0	$1/3$	$-2/3$	2
x_3	0	$7/6$	0	1	$1/6$	$1/6$	10.

$B_P = \begin{pmatrix} 1 & 4 \\ -1 & 2 \end{pmatrix}$ $B_D = \begin{pmatrix} -1 & 5 & 2 \\ 0 & 1 & -1 \\ 0 & 4 & 2 \end{pmatrix}$

no violations.

$\therefore \; x_1^* = \frac{11}{3}, \quad x_2^{*\prime} = 0, \quad x_3^* = 0.$

(d). $x_1 w_1 = x_2 w_2 = x_3 w_3 = x_4 w_1 = x_5 w_2 = 0.$

$x_1(5y_1 + 2y_2 - 3) = 0 \qquad x_1^* = 0$

$y_1(5x_1 + x_2 + 4x_3 - 42) = 0 \qquad x_2^* + 4x_3 = 42$

$y_2(2x_1 - x_2 + 2x_3 - 18) = 0 \qquad -x_2^* + 2x_3^* = 18$

$\left.\begin{matrix} x_2^* + 4x_3 = 42 \\ -x_2^* + 2x_3^* = 18 \end{matrix}\right\} \quad x_2^* = 2, \quad x_3^* = 10$

$x_4(y_1 - 0) = 0 \quad x_4^* = 0$

$x_5(y_2 - 0) = 0 \quad x_5^* = 0.$

Therefore, all conditions are satisfied.

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Sat Dec  5 05:07:07 2020

@author: fionafei
"""


import cvxpy as cp

x = cp.Variable(3, nonneg = True) # vector variable



obj_func=3*x[0]-x[1]+6*x[2]
#obj_func_neg=3*x[0]+2*x[1]

constraints = []
constraints.append(5*x[0]+x[1]+4*x[2]<=42)
constraints.append(2*x[0]-x[1]+2*x[2]<=18)
constraints.append(x[0]>=0)
constraints.append(x[1]>=0)
constraints.append(x[2]>=0)


problem = cp.Problem(cp.Maximize(obj_func), constraints)
#problem = cp.Problem(cp.Minimize(obj_func_neg), constraints)

#problem.solve(solver=cp.CVXOPT,verbose = True)
#problem.solve(verbose = True)
problem.solve(solver=cp.GUROBI,verbose = True)

print("obj_func =")
print(obj_func.value)
#print(obj_func_neg.value)
print("x =")
print(x.value)

# Shadow prices or dual prices
print("optimal (x[0]+2*x[1]<=11) dual variable", constraints[0].dual_value)
print("optimal (x[0]-3*x[1]<=1) dual variable", constraints[1].dual_value)

print("optimal (x[0]>=0) dual variable", constraints[2].dual_value)
print("optimal (x[1]>=0) dual variable", constraints[3].dual_value)
print("optimal (x[2]>=0) dual variable", constraints[4].dual_value)

Solved in 1 iterations and 0.01 seconds
Optimal objective -5.800000000e+01
obj_func =
```

```
58.0
x =
[ 0.  2. 10.]
optimal (x[0]+2*x[1]<=11) dual variable 0.6666666666666666
optimal (x[0]-3*x[1]<=1) dual variable 1.6666666666666667
optimal (x[0]>=0) dual variable -0.0
optimal (x[1]>=0) dual variable -0.0
optimal (x[2]>=0) dual variable -0.0
```