Althea Lutz
MATH 381

Homework 1

Some of these problems are from *Operations Research, Fourth Edition* by Wayne Winston.
See the syllabus for a link to the pdf.
Some of these problems will require you to use a computer to solve the LP.
Please include a screenshot of your code and the output in your submission.
Two of the problems will be selected to be graded. You will not be told which two in advance.

① Winston, Problem 5, page 63. Solve the LP by hand.

Furnco manufactures desks and chairs. Each desk uses 4 units of wood, and each chair uses 3. A desk contributes \$40 to profit, and a chair contributes \$25. Marketing restrictions require that the number of chairs produced be at least twice the number of desks produced. If 20 units of wood are available, formulate an LP to maximize Furnco's profit. Then graphically solve the LP.

Variables:

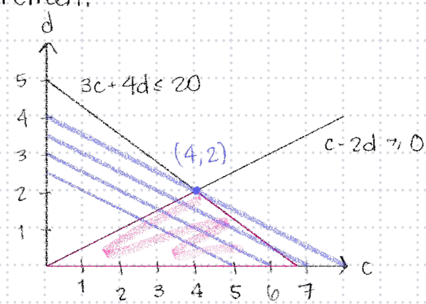
d = #/ desks
c = #/ chairs

Objective function:

$25c + 40d$ ← maximizing profit

Constraints:

$c \geq 2d \Rightarrow c - 2d \geq 0$ ← marketing requirement
$3c + 4d \leq 20$ ← available wood
$c, d \geq 0$ ← implicit



$$\begin{aligned} \max \quad & 25c + 40d \\ \text{s.t.} \quad & c - 2d \geq 0 \\ & 3c + 4d \leq 20 \\ & c, d \geq 0 \end{aligned}$$

$d = 0 \Rightarrow c \leq 20/3$
$c = 0 \Rightarrow d \leq 5$

$3c + 4d = 20$
$6d + 4d = 20$
$10d = 20$
$d = 2$

$c - 2d = 0$
$c = 2d$
$c = 4$
✓

$25(4) + 40(2) = 100 + 80 = 180$

optimal solution: $(4, 2)$
optimal value: 180

Furnco has a maximum profit of \$180 when they produce 4 chairs and 2 desks.

② Winston, Problem 13, page 115. Solve the LP using OR-Tools.

Feedco produces two types of cattle feed, both consisting totally of wheat and alfalfa. Feed 1 must contain at least 80% wheat, and feed 2 must contain at least 60% alfalfa. Feed 1 sells for \$1.50/lb, and feed 2 sells for \$1.30/lb. Feedco can purchase up to 1,000 lb of wheat at 50¢/lb and up to 800 lb of alfalfa at 40¢/lb. Demand for each type of feed is unlimited. Formulate an LP to maximize Feedco's profit.

Variables:

w_1 = lbs/ wheat in feed 1
a_1 = lbs/ alfalfa in feed 1
w_2 = lbs/ wheat in feed 2
a_2 = lbs/ alfalfa in feed 2

Objective function:

$(1.5-0.5)w_1 + (1.5-0.4)a_1 + (1.3-0.5)w_2 + (1.3-0.4)a_2$

$\Rightarrow w_1 + 1.1a_1 + 0.8w_2 + 0.9a_2 \leftarrow$ maximizing profit

Constraints:

$\frac{w_1}{w_1+a_1} \geq 0.8 \Rightarrow 0.2w_1 - 0.8a_1 \geq 0 \leftarrow$ feed 1 composition

$\frac{a_2}{w_2+a_2} \geq 0.6 \Rightarrow -0.6w_2 + 0.4a_2 \geq 0 \leftarrow$ feed 2 composition

$w_1 + w_2 \leq 1000 \leftarrow$ wheat limit

$a_1 + a_2 \leq 800 \leftarrow$ alfalfa limit

$w_1, a_1, w_2, a_2 \geq 0 \leftarrow$ implicit

$$
\begin{aligned}
\max \quad & w_1 + 1.1a_1 + 0.8w_2 + 0.9a_2 \\
\text{s.t.} \quad & 0.2w_1 - 0.8a_1 \geq 0 \\
& -0.6w_2 + 0.4a_2 \geq 0 \\
& w_1 + w_2 \leq 1000 \\
& a_1 + a_2 \leq 800 \\
& w_1, a_1, w_2, a_2 \geq 0
\end{aligned}
$$

```python
from ortools.linear_solver import pywraplp

# Construct the LP solver. We will use GLOP, a simplex method solver.
solver = pywraplp.Solver.CreateSolver("GLOP")

# Create the variables c1, c2, w1, w2
w1 = solver.NumVar(0, solver.infinity(), "w1")
a1 = solver.NumVar(0, solver.infinity(), "a1")
w2 = solver.NumVar(0, solver.infinity(), "w2")
a2 = solver.NumVar(0, solver.infinity(), "a2")

# Create the constraints
solver.Add(0.2*w1 - 0.8*a1 >= 0)
solver.Add(-0.6*w2 + 0.4*a2 >= 0)
solver.Add(w1 + w2 <= 1000)
solver.Add(a1 + a2 <= 800)

# Set the objective
solver.Maximize(w1 + 1.1*a1 + 0.8*w2 + 0.9*a2)

# Solve
status = solver.Solve()
val = solver.Objective().Value() # optimal value

# Print solution
if status == pywraplp.Solver.OPTIMAL:
    print(f"Optimal value = {val:0.2f}")
    print(f"w1 = {w1.solution_value():0.2f}")
    print(f"a1 = {a1.solution_value():0.2f}")
    print(f"w2 = {w2.solution_value():0.2f}")
    print(f"a2 = {a2.solution_value():0.2f}")
else:
    print("The problem does not have an optimal solution.")
```

```
Optimal value = 1770.00
w1 = 1000.00
a1 = 250.00
w2 = 0.00
a2 = 550.00
```

To reach a maximum profit of \$1,770, Feedco must use 1,000 pounds of wheat and 250 pounds of alfalfa for feed 1, and 550 pounds of alfalfa only for feed 2.

(3) Winston, Problem 14, page 115. Solve the LP using OR-Tools.

Feedco has decided to give its customer (assume it has only one customer) a quantity discount. If the customer purchases more than 300 lb of feed 1, each pound over the first 300 lb will sell for only \$1.25/lb. Similarly, if the customer purchases more than 300 pounds of feed 2, each pound over the first 300 lb will sell for \$1.00/lb. Modify the LP of problem 13 to account for the presence of quantity discounts. (*Hint*: Define variables for the feed sold at each price.)

Variables:

f_1 = lbs/ full price feed 1 sold
df_1 = lbs/ discounted feed 1 sold
f_2 = lbs/ full price feed 2 sold
df_2 = lbs/ discounted feed 2 sold
w_1 = lbs/ wheat in feed 1
a_1 = lbs/ alfalfa in feed 1
w_2 = lbs/ wheat in feed 2
a_2 = lbs/ alfalfa in feed 2

Objective function:

$$1.5f_1 + 1.25df_1 + 1.3f_2 + df_2 - 0.5w_1 - 0.5w_2 - 0.4a_1 - 0.4a_2$$

Constraints:

$f_1 + df_1 = w_1 + a_1$
$f_2 + df_2 = w_2 + a_2$
$df_1 - f_1 \geq 300$
$df_2 - f_2 \geq 300$
$0.2w_1 - 0.8a_1 \geq 0$
$-0.6w_2 + 0.4a_2 \geq 0$
$w_1 + w_2 \leq 1000$ ← wheat limit
$a_1 + a_2 \leq 800$ ← alfalfa limit
$f_1, df_1, f_2, df_2, w_1, a_1, w_2, a_2 \geq 0$ ← implicit

$$
\begin{array}{ll}
\max & 1.5f_1 + 1.25df_1 + 1.3f_2 + df_2 - 0.5w_1 - 0.5w_2 - 0.4a_1 - 0.4a_2 \\
\text{s.t.} & f_1 + df_1 = w_1 + a_1 \\
& f_2 + df_2 = w_2 + a_2 \\
& df_1 - f_1 \geq 300 \\
& df_2 - f_2 \geq 300 \\
& 0.2w_1 - 0.8a_1 \geq 0 \\
& -0.6w_2 + 0.4a_2 \geq 0 \\
& w_1 + w_2 \leq 1000 \\
& a_1 + a_2 \leq 800 \\
& f_1, df_1, f_2, df_2, w_1, a_1, w_2, a_2 \geq 0
\end{array}
$$

```python
from ortools.linear_solver import pywraplp

# Construct the LP solver. We will use GLOP, a simplex method solver.
solver = pywraplp.Solver.CreateSolver("GLOP")

# Create the variables
f1 = solver.NumVar(0, solver.infinity(), "f1")
df1 = solver.NumVar(0, solver.infinity(), "df1")
f2 = solver.NumVar(0, solver.infinity(), "f2")
df2 = solver.NumVar(0, solver.infinity(), "df2")
w1 = solver.NumVar(0, solver.infinity(), "w1")
a1 = solver.NumVar(0, solver.infinity(), "a1")
w2 = solver.NumVar(0, solver.infinity(), "w2")
a2 = solver.NumVar(0, solver.infinity(), "a2")

# Create the constraints
solver.Add(f1 + df1 == w1 + a1)
solver.Add(f2 + df2 == w2 + a2)
solver.Add(df1 - f1 >= 300)
solver.Add(df2 - f2 >= 300)
solver.Add(0.2*w1 - 0.8*a1 >= 0)
solver.Add(-0.6*w2 + 0.4*a2 >= 0)
solver.Add(w1 + w2 <= 1000)
solver.Add(a1 + a2 <= 800)

# Set the objective
solver.Maximize(1.5*f1 + 1.25*df1 + 1.3*f2 + df2 - 0.5*w1 - 0.5*w2 - 0.4*a1 - 0.4*a1)

# Solve
status = solver.Solve()
val = solver.Objective().Value() # optimal value

# Print solution
if status == pywraplp.Solver.OPTIMAL:
    print(f"Optimal value = {val:0.2f}")
    print(f"f1 = {f1.solution_value():0.2f}")
    print(f"df1 = {df1.solution_value():0.2f}")
    print(f"f2 = {f2.solution_value():0.2f}")
    print(f"df2 = {df2.solution_value():0.2f}")
    print(f"w1 = {w1.solution_value():0.2f}")
    print(f"a1 = {a1.solution_value():0.2f}")
    print(f"w2 = {w2.solution_value():0.2f}")
    print(f"a2 = {a2.solution_value():0.2f}")
else:
    print("The problem does not have an optimal solution.")
```

```
Optimal value = 1712.50
f1 = 350.00
df1 = 650.00
f2 = 250.00
df2 = 550.00
w1 = 1000.00
a1 = 0.00
w2 = 0.00
a2 = 800.00
```

Feedco makes a maximum profit of $1,712.50 by using 1,000 pounds of wheat only for feed 1 and 800 pounds of alfalfa only for feed 2. $350 is made from full-price feed 1, $650 is made from discounted feed 1, $250 is made from full-price feed 2, and $550 is made from discounted feed 2.

④ Winston, Problem 46, page 120. Please have the computer solve over the real numbers, not the integers. You will not get credit if you use an integer problem solver. After you solve this LP, you may need to do some additional work to make a realistic recommendation. You do not have to prove that your recommendation is the best possible.

Gotham City National Bank is open Monday-Friday from 9 A.M.-5 P.M. From past experience, the bank knows that it needs the number of tellers shown in the table. The bank hires two types of tellers. Full-time tellers work 9-5 five days a week, except for 1 hour off for lunch. (The bank determines when a full-time employee takes lunch hour, but each teller must go between noon and 1 P.M. or between 1 P.M. and 2 P.M.) Full-time employees are paid (including fringe benefits) \$8/hour (this includes payment for lunch hour). The bank may also hire part-time tellers. Each part-time teller must work exactly 3 consecutive hours each day. A part-time teller is paid \$5/hour (and receives no fringe benefits). To maintain adequate quality of service, the bank has decided that at most five part-time tellers can be hired. Formulate an LP to meet the teller requirements at minimum cost. Solve the LP on a computer. Experiment with the LP answer to determine an employment policy that comes close to minimizing labor cost.

Time Period		Tellers Required
0	9-10	4
1	10-11	3
2	11-12	4
3	12-1	6
4	1-2	5
5	2-3	6
6	3-4	8
7	4-5	8

min $72f_0 + 15p_i$

s.t.
$f_0 + p_0 \geq 4$
$f_0 + p_0 + p_1 \geq 3$
$f_0 + p_0 + p_1 + p_2 \geq 4$
$f_0 + p_1 + p_2 + p_3 - l_3 \geq 6$
$f_0 + p_2 + p_3 + p_4 - l_4 \geq 5$
$f_0 + p_3 + p_4 + p_5 \geq 6$
$f_0 + p_4 + p_5 \geq 8$
$f_0 + p_5 \geq 8$
$f_0 = l_3 + l_4$
$f_0, p_i, l_4, l_5 \geq 0 \ \forall i$

Variables

f_0 = #/ full-time tellers starting shift at period 0

l_3 = #/ full-time tellers taking lunch during period 3

l_4 = #/ full-time tellers taking lunch during period 4

p_i = #/ part-time tellers starting shift at period i

Objective function:

$(8\cdot9)f_0 + (3\cdot5)p_i = 72f_0 + 15p_i$ ← minimize cost

Constraints:

$f_0 + p_0 \geq 4$
$f_0 + p_0 + p_1 \geq 3$
$f_0 + p_0 + p_1 + p_2 \geq 4$
$f_0 + p_1 + p_2 + p_3 - l_3 \geq 6$
$f_0 + p_2 + p_3 + p_4 - l_4 \geq 5$
$f_0 + p_3 + p_4 + p_5 \geq 6$
$f_0 + p_4 + p_5 \geq 8$
$f_0 + p_5 \geq 8$
$f_0 = l_3 + l_4$ $\quad f_0, p_i, l_4, l_5 \geq 0 \ \forall i$
$p_0 + p_1 + p_2 + p_3 + p_4 + p_5 \leq 5$

```python
from ortools.linear_solver import pywraplp

# Construct the LP solver. We will use GLOP, a simplex method solver.
solver = pywraplp.Solver.CreateSolver("GLOP")

# Create the variables c1, c2, w1, w2
f0 = solver.NumVar(0, solver.infinity(), "f0")
l3 = solver.NumVar(0, solver.infinity(), "l3")
l4 = solver.NumVar(0, solver.infinity(), "l4")
p0 = solver.NumVar(0, solver.infinity(), "p0")
p1 = solver.NumVar(0, solver.infinity(), "p1")
p2 = solver.NumVar(0, solver.infinity(), "p2")
p3 = solver.NumVar(0, solver.infinity(), "p3")
p4 = solver.NumVar(0, solver.infinity(), "p4")
p5 = solver.NumVar(0, solver.infinity(), "p5")

# Create the constraints
solver.Add(f0 + p0 >= 4)
solver.Add(f0 +p0 + p1 >= 3)
solver.Add(f0 + p0 + p1 + p2 >= 4)
solver.Add(f0 + p1 + p2 + p3 - l3 >= 6)
solver.Add(f0 + p2 + p3 + p4 - l4 >= 5)
solver.Add(f0 + p3 + p4 + p5 >= 6)
solver.Add(f0 + p4 + p5 >= 8)
solver.Add(f0 + p5 >= 8)
solver.Add(f0 == l3 + l4)

# Set the objective
solver.Minimize(72*f0 + 15*p0 + 15*p1 + 15*p2 + 15*p3 + 15*p4 + 15*p5)

# Solve
status = solver.Solve()
val = solver.Objective().Value() # optimal value

# Print solution
if status == pywraplp.Solver.OPTIMAL:
    print(f"Optimal value = {val:0.2f}")
    print(f"f0 = {f0.solution_value():0.2f}")
    print(f"l3 = {l3.solution_value():0.2f}")
    print(f"l4 = {l4.solution_value():0.2f}")
    print(f"p0 = {p0.solution_value():0.2f}")
    print(f"p1 = {p1.solution_value():0.2f}")
    print(f"p2 = {p2.solution_value():0.2f}")
    print(f"p3 = {p3.solution_value():0.2f}")
    print(f"p4 = {p4.solution_value():0.2f}")
    print(f"p5 = {p5.solution_value():0.2f}")
else:
    print("The problem does not have an optimal solution.")
```

```
Optimal value = 483.00
f0 = 5.67
l3 = 2.33
l4 = 3.33
p0 = 0.00
p1 = 0.00
p2 = 0.00
p3 = 2.67
p4 = 0.00
p5 = 2.33
```

adjusting to integers

→ Optimal value = 507

$f_0 = 6$

$l_3 = 3$

$l_4 = 3$

$p_0 = 0$

$p_1 = 0$

$p_2 = 0$

$p_3 = 3$

$p_4 = 0$

$p_5 = 2$

Gotham city bank spends a minimizes labor costs at \$507/day by having 6 full-time tellers (3 having lunch from 12-1, 3 having lunch from 1-2) and 5 part-time tellers (3 starting at 12, 2 starting at 5).

5. Recall the corn/wheat farm example we did in lecture. To increase profit, the farmer is considering buying a little more land, hiring a little more labor, or both. Make a recommendation on what the farmer should do. Your recommendation should take into account that buying land and labor costs money.

$$
\left.
\begin{array}{ll}
\min & 100c_1 + 120c_2 + 90w_1 + 80w_2 \\
\text{s.t.} & c_1 + w_1 \le 100 \\
 & c_2 + w_2 \le 100 \\
 & 500c_1 + 650c_2 \ge 7000 \\
 & 400w_1 + 350w_2 \ge 11000 \\
 & c_1, w_1, c_2, w_2 \ge 0 \\
 & \\
\multicolumn{2}{l}{\text{optimal solution} = (0, 10.77, 27.50, 0)} \\
\multicolumn{2}{l}{\text{optimal value} = 3767.31}
\end{array}
\right\} \text{original problem}
$$

In the original problem, the farmer is only using 10.77 of the available 100 acres in farm 1 and 27.5 of the available 100 acres in farm 2. Since they are not yet using all the alotted land, it would not benefit them to purchase more land. Instead, the farmer should spend more money on labor, to maximize the land they already own.