ISyE6669 Homework Week 6
Spring 2021

1 Week 6

1. A machine shop has a drill press and a milling machine which are used to produce two parts A and B. The required time (in minutes) per unit part on each machine is shown in the table below.

	Drill press	Milling machine
A	3	4
B	5	3

Table 1: Data for required times in minutes

The shop must produce at least 50 units in total (both A and B) and at least 30 units of part A, and it can make at most 100 units of part A and 100 units of part B. You can assume that the shop can make fractional amount of the parts. Formulate an LP to minimize the absolute difference between the total running time of the drill press and that of the milling machine. You need to define clearly the meaning of the variables that you use in the LP. Finally, you need to implement your model in CVXPY and print out the optimal solution and objective value.

Solution:

Let's break this down.

Decision variables: x is the number of part A produced. y is the number of part B produced.

Constraints:

(a) Total A needs to be greater than 30: $x \geq 30$;

(b) Total A and B must be at least 50: $x + y \geq 50$;

(c) Total A and B can be at most 100: $x \leq 100, y \leq 100$.

Objective Function: Our objective function is to minimize the difference in the running times between the two machines. And the difference means the absolute value of the number of units produced by the drill press multiplied by how long it takes to manufacture those units (both

A and B) subtracted by the number of units produced by the milling machine multiplied by how long it takes to manufacture those units. Our optimization problem becomes:

$$\begin{aligned} \min \quad & |(3x+5y)-(4x+3y)| \\ \text{s.t.} \quad & x+y \geq 50, \\ & x \geq 30, \\ https://www.overleaf.com/project/5ded83061c17bc00011ec76a & x \leq 100, \\ & y \leq 100, \\ & x \geq 0, y \geq 0. \end{aligned}$$

Based on the lectures we need to convert our optimization problem into a linear problem. This is shown below:

$$\begin{aligned} \min \quad & z \\ \text{s.t.} \quad & z \geq -x+2y, \\ & z \geq x-2y, \\ & x+y \geq 50, \\ & x \geq 30, \\ & x \leq 100, \\ & y \leq 100, \\ & x \geq 0, y \geq 0. \end{aligned}$$

Optimal value: $0.000000000528472124841 49 \approx 0$

Optimal production of unit A: $51.85645822 \approx 52$

Optimal production of unit B: $25.92822911 \approx 26$

Note: This problem has multiple optimal solutions. As long as the formulation and reformulation are correct and optimal value is close to 0, solution should be accepted.

The code is shown below:

```
import cvxpy as cp
import numpy as np

x = cp.Variable(1)
y = cp.Variable(1)
z = cp.Variable(1)

objective = cp.Minimize(z)

constraints = [
```

```
    -x + 2*y <= z,
    -x + 2*y >= -z,
    x+y >= 50,
    x >= 30,
    x <= 100,
    y <= 100,
    x >= 0,
    y >= 0
]

prob = cp.Problem(objective, constraints)
print('Optimal objective value: {} \n '.format((prob.solve())))
print('Optimal production of unit A: {}'.format((x.value)))
print('Optimal production of unit B: {}'.format((y.value)))
```

2. I am a retailer of suitcases. I can purchase suitcases from two suppliers. Supplier 1 sells one suitcase for \$10. Supplier 2 sells suitcases in the following fashion:

   (a) It is a fixed cost of \$1200 for purchasing 100 or less suitcases. (Assume that Supplier 2 charges \$1200 even if no suitcases are purchased from supplier 2.)

   (b) For each suitcase more that 100 suitcases purchased, there is an additional charge of \$5 per suitcase.

   I want to buy 500 suitcases. Formulate a linear program to minimize the total cost of my purchase. You should treat the number of suitcases as a continuous variable. You need to define clearly the meaning of the variables that you use. Finally, implement and solve your LP in CVXPY and print out the optimal solution and objective value.

   Hint: You may encounter a nonlinear function involving the maximum function. But you can reformulate such nonlinear function using linear constraints.

   Solution:

   Our goal is to minimize the total cost of the purchase.

   Decision Variables: Let x be the number of suitcases purchased from Supplier 1 and let y be the number of suitcases purchased from Supplier 2.

   This can be formulated as a piece-wise optimization problem:

$$
\begin{aligned}
\min \quad & \begin{cases} 1200 + 10x, & 0 \leq y \leq 100 \\ 1200 + 10x + 5(y - 100), & y > 100 \end{cases} \\
\text{s.t.} \quad & x + y = 500 \\
& x \geq 0, y \geq 0.
\end{aligned}
$$

   The lectures state that convex piece-wise linear functions can be formulated as the max of a finite number of linear functions. This further translates into:

$$\begin{aligned} \min \quad & \max\{1200 + 10x,\ 700 + 10x + 5y\} \\ \text{s.t.} \quad & x + y = 500 \\ & x \geq 0, y \geq 0. \end{aligned}$$

We introduce a new decision variable z and the original objective variable is now incorporated as part of the constraints. Therefore:

$$\begin{aligned} \min \quad & z \\ \text{s.t.} \quad & 1200 + 10x \leq z, \\ & 700 + 10x + 5y \leq z, \\ & x + y = 500, \\ & x \geq 0, y \geq 0. \end{aligned}$$

Optimal value for total cost: $3200.0000057373554 \approx \$3200$

Number of suitcases purchased from Supplier 1: $0.00000115511929 \approx 0$

Number of suitcases purchased from Supplier 2: $499.99999884 \approx 500$

Note: There are multiple ways to reformulate original formulation into linear program. Depending from the reformulation solutions may be slightly different. However rounded to the nearest integers they should give same values as what is stated above.

Code is show below:

```
import cvxpy as cp
import numpy as np


x = cp.Variable(1)
y = cp.Variable(1)
x = cp.Variable(1)

objective = cp.Minimize(z)

constraints = [x + y == 500,
               z >= 1200 + 10*x,
               z >= 700 + 10*x + 5*y,
               x >= 0,
               y >= 0
               ]
```

```python
prob = cp.Problem(objective, constraints)
print('Optimal objective value: {} \n '.format((prob.solve())))
print('Suitcases purchased from supplier 1:{}'.format((x.value)))
print('Suitcases purchased from supplier 2:{}'.format((y.value)))
```