ISyE 6673 HW4 (Due: 11:59pm on 11/1/2022)

Please write your answers neatly or (recommended) type them up in LaTeX. Code should be implemented in a Jupyter notebook file with your name and GT ID clearly indicated. Please submit the homework as a PDF via Gradescope, including a PDF of your Jupyter notebook and please make sure to follow all Gradescope instructions matching page numbers to questions (if we have to put unreasonable effort into matching, we may choose to take off points or leave problems ungraded). Additionally, please submit your Jupyter notebook separately on Canvas as a backup. For modeling problems, clearly indicate your decision variables, constraints, and objectives. All notation should be clearly explained. Homework should be done independently. Please abide by the Georgia Tech Honor Code. Published 10/22/2022.

Problem 1: Cutting stock (40 pts)

In this problem we will implement the column generation algorithm for the cutting stock problem. We begin with an unlimited supply of rolls of width $W = 30$, and we are trying to satisfy the demand of six customers, described in Table 1.

Customer i	Width w_i	Demand b_i
1	3	120
2	4	280
3	7	130
4	9	110
5	13	80
6	16	150

Table 1: Data for the cutting stock problem

We will build up the elements of a column generation algorithm in JuMP. As in lecture, we represent a cutting pattern as a vector $\mathbf{a}$ of length 6, where a_i indicates the number of rolls of width i in the pattern.

This is the first time we will create and solve multiple models in JuMP. When using Gurobi on past assignments, you probably noticed that the solver outputs a log of what it's doing. It's very nice, but sometimes we just need the solver to be quiet and let us work in peace. We can silence the solver log by doing the following.

1. To silence most of the log, we can add the following line of code after creating the model:
   ```
   model = Model(Gurobi.Optimizer)
   set_optimizer_attribute(model, "OutputFlag", 0)
   ```

2. The above line will keep Gurobi mostly quiet. It will still output an annoying academic license warning, however. We can also silence that by doing the following:

   - Add the following line at the top of the program (right underneath `using Gurobi`):
     ```
     GRB_ENV = Gurobi.Env()
     ```
   - Anytime you create a model, replace the line
     ```
     model = Model(Gurobi.Optimizer)
     ```
     with
     ```
     model = Model(() -> Gurobi.Optimizer(GRB_ENV))
     ```

(a) **Initializing the set of patterns.** To initialize the restricted problem, we need to create an initial set of cutting patterns that produces a feasible solution for the problem. The simplest solution is to produce one pattern per customer, with pattern i corresponding to cutting as many rolls of width w_i as possible out of the roll of width W.

(i) (2 pts) Provide a general form for the i-th pattern as a function of W and w_i.

(ii) (3 pts) In a table, indicate the starting patterns corresponding to the data in Table 1. In your table, the rows should indicate patterns and the columns should indicate widths/customer types.

(iii) (2 pts) Write the set of starting patterns in Julia as a vector, where each element of the vector is a pattern (and therefore also a vector of integers). Print out the list of starting patterns in your Jupyter notebook.

(b) **Solving the restricted problem.** We would now like to solve the restricted problem over an arbitrary set of patterns. The formulation for the restricted problem was provided in lecture. The objective is to minimize the number of rolls used.

(i) (8 pts) Implement a function in Julia called `solve_restricted_problem` which takes in as arguments the set of widths $\mathbf{w}$, the set of demands $\mathbf{b}$, the starting width W, and an arbitrary set of patterns, and returns the optimal primal and dual solutions. Please comment your code clearly.

(ii) (2 pts) Run your code on the starting set of patterns you created in part (a). What is the total number of rolls we need to use according to this restricted optimal solution?

(c) **Solving the subproblem.** Recall the formulation of the column generation subproblem in lecture. We know that this is a knapsack problem which can be solved via dynamic programming. However, it can also be solved using integer optimization. We'll talk about integer optimization later on in the class, but for now all you need to know is how to create non-negative integer variables in JuMP: `@variable(model, x[1:n] >= 0, Int)`.

(i) (7 pts) Implement a function in Julia called `solve_subproblem` which takes in as arguments the set of widths $\mathbf{w}$, the starting width W, and the optimal dual variables of the restricted problem $\mathbf{p}$. Your function should return the optimal objective of the subproblem as well as the newly created pattern which attains this optimal objective.

(ii) (3 pts) Run your code using the optimal dual solution obtained in part (a-ii). What is the optimal solution of the subproblem? Is the solution of the restricted problem optimal? Which pattern does the subproblem generate?

(d) **Putting it all together.**

(i) (5 pts) Implement a function in Julia called `solve_with_column_generation` which takes in as arguments the set of widths $\mathbf{w}$, the set of demands $\mathbf{b}$, and the starting width W, and returns the optimal patterns and the number of times each pattern is used.

(ii) (3 pts) Run your code on the data from Table 1. In a table, summarize the optimal solution: one row per pattern, indicating the number of times each pattern is used, and the number of rolls of each width each pattern contains. No need to include patterns that are used 0 times in the table.

(iii) (2 pts) What is the optimal objective value? Compare with the value obtained in part (b-ii). What is the percentage gain from using optimization?

(iv) (3 pts) The optimal solution obtained in the previous part potentially requires a fractional number of patterns, but this number should be integral. Report the integral solution obtained by rounding up any fractional number of patterns. What is the optimality gap of this solution relative to the continuous optimal solution?

Problem 2: Analyzing three markets (27 pts)

Consider the following three markets. All of them involve two periods, the present period $t = 0$ and the future period $t = 1$. The market in the future period has uncertainty, described by a set of scenarios. Each market has a risk-free asset, which is always indexed as the first asset. The current price of the risk-free asset is always $p_1 = 1$ and an interest rate of r. The payoff of asset j in scenario i is denoted as S_{ij}. So the risk-free asset's payoffs are $S_{i1} = 1 + r$ for all scenarios i.

- Market A has one risky asset and two scenarios in period $t = 1$. Interest rate $r = 1/9$. The risky asset's current price is $p_2 = 5$. The risky asset's payoffs in period 1 are given by $S_{12} = 20/3$ and $S_{22} = 49/9$.
- Market B has one risky asset and three scenarios in period $t = 1$. Interest rate $r = 1/9$. The risky asset's current price is $p_2 = 5$. The risky asset's payoffs in period 1 are given by $S_{12} = 20/3, S_{22} = 49/9, S_{32} = 10/3$.
- Market C has two risky assets and three scenarios in period $t = 1$. Interest rate $r = 1/9$. The risky assets' current prices are $p_2 = 5, p_3 = 10$. Risky asset 2's payoffs in period 1 are given by $S_{12} = 20/3, S_{22} = 20/3, S_{32} = 40/9$. Risky asset 3's payoffs in period 1 are given by $S_{13} = 40/3, S_{23} = 80/9, S_{33} = 80/9$.

Answer the following questions for each market.

(a) (6 pts) Write down the vector of current prices $\boldsymbol{p}$ and the payoff matrix $\boldsymbol{S}$. Note $\boldsymbol{p}$ should be a column vector of n rows and $\boldsymbol{S}$ should be a matrix of size $m \times n$, where m is the number of scenarios and n is the number of assets (both risk-free and risky).

(b) (3 pts) Is the market a complete market? Why? If the market is not complete, find a payoff vector (i.e. a contingent claim) that is not attainable by any portfolio composed of the assets in the market.

(c) (6 pts) Does the Law of One Price hold in the market? Why?

(d) (3 pts) If the Law of One Price holds, find the unique (for complete markets) linear payoff pricing function $q(\boldsymbol{z}) = \boldsymbol{q}^\top \boldsymbol{z}$ where $\boldsymbol{z}$ is a payoff vector. If the market is not complete, can you find two different representations of the payoff pricing function, i.e. two different vectors $\boldsymbol{q}_1, \boldsymbol{q}_2$ such that $q(\boldsymbol{z}) = \boldsymbol{q}_1^\top \boldsymbol{z} = \boldsymbol{q}_2^\top \boldsymbol{z}$ for all payoff vectors $\boldsymbol{z}$?

(e) (3 pts) Does the market have a strong arbitrage opportunity? Argue by using the pricing function you found in (c). If the market has a strong arbitrage opportunity, find a strong-arbitrage portfolio.

(f) (3 pts) Does the market have a weak arbitrage opportunity? Argue by using the pricing function you found in (c). If the market has a weak arbitrage opportunity, find such a portfolio.

(g) (3 pts) If the market is arbitrage-free, find a risk-neutral probability distribution. If there are multiple risk-neutral probability distributions, find two distinct ones.

Problem 3: Contingent claims in incomplete markets (33 pts)

In this exercise, we will see that in an incomplete market, some contingent claims may not have a unique arbitrage-free price.

Consider Market B in Problem 2 again. Now someone introduces a new contingent claim with payoff vector $\boldsymbol{C}$ given by $\boldsymbol{C} = [3.5, 5, 2]^\top$, which lists the payoff of the contingent claim in each of the three scenarios in period $t = 1$. Answer the following questions.

(a) (4 pts) Can we build a portfolio using the two existing assets in the market to completely replicate the new contingent claim? If yes, find such a portfolio. If not, explain why.

(b) As an investor in this market, you would like to buy the new contingent claim $\boldsymbol{C}$.

(i) (4 pts) Formulate a linear optimization problem to find the maximum arbitrage-free price you would be willing to pay for the claim.

(ii) (2 pts) Write your optimization model in JuMP and solve it using Gurobi. Report the optimal objective value, corresponding to the maximum arbitrage-free price for the claim.

(iii) (4 pts) Now formulate another linear optimization problem to find the minimum cost of building a portfolio out of the two existing assets such that the payoff of the portfolio always dominates (i.e. greater than or equal to) that of the contingent claim.

(iv) (2 pts) Using JuMP and Gurobi, report such a portfolio. What is its price?

(v) (2 pts) What do you notice about your answers in part (ii) and (iv). Why?

(c) As an investor in this market, you would like to sell the new contingent claim $\boldsymbol{C}$.

(i) (4 pts) Formulate a linear optimization problem to find the minimum arbitrage-free price you would be willing to pay for the claim.

(ii) (2 pts) Write your optimization model in JuMP and solve it using Gurobi. Report the optimal objective value, corresponding to the minimum arbitrage-free price for the claim.

(iii) (4 pts) Now formulate another linear program that finds the highest price of a portfolio composed of the two existing assets whose payoffs are inferior to (i.e. less than or equal to) the new contingent claim.

(iv) (2 pts) Using JuMP and Gurobi, report such a portfolio. What is its price?

(v) (3 pts) Explain why your answers in part (b) and (c) differ.

(Extra Credit) Problem 4: Pricing call options (10 pts)

Consider a market consisted of a risk-free asset with interest rate $r = 0$ and a risky asset. The risky asset's current price is $p = 100$. Its future payoff is either 120 or 90 in two scenarios. Suppose both scenarios would happen with a positive probability. Answer the following questions.

(a) (2 pts) If an investor buys a unit of the risky asset in time 0, what is the return of the investment in each of the scenarios in time 1? Return is computed as the ratio $(p_{future} - p_{current})/p_{current}$.

(b) (8 pts) If an investor instead buys a unit of a call option on the risky asset with strike price $K = 100$, what are possible returns on the investment? You should see the call option amplifies both the gains and losses, comparing to purely holding the risky asset.