Optimization
Finance Project 1 – Linear Programming

Deliverables
One well-written pdf file and one Python code file (.py or .ipynb), submitted to Canvas. Your report should go into some detail about how you solved the problem, include some graphs that explain your results, and include relevant code chunks in the final output. Your report can be created by taking screenshots of the code/graph and assembling it in a word document, then export as a pdf file.

Problem Description
Bonds are a form of debt issued by governments and companies. You can purchase a bond for a given price with maturity date in the future at which point you will receive the face value of the bond. For example, you could purchase a bond for $98 (price) and at maturity the bond issuer will pay you $100 (face value). Coupon bonds are the bonds that offer a periodic payment during the life of the bond. Let's say you purchase a bond for $102 with an annual coupon payment of $5, a face value of $100 and maturity three years from now. Then you will pay $102 immediately to purchase the bond, receive $5 coupon payments at the end of years 1, 2 and finally $5+$100=$105 at the end of year 3. For additional information see http://guides.wsj.com/personal-finance/investing/what-is-a-bond/.

Bond forwards are similar to bonds, except the purchase of the bond will happen in the future, at terms agreed upon today. For example, if you enter a forward bond contract that starts at year 1 at a forward price of $98 with a maturity at year 4 and $3 coupon, then today no money exchanges hands, one year from today you must pay $98, two years from today you get a $3 payment, three years from today you get a $3 payment and 4 years from today you get a $103 payment. There is no optionality in this contract, you can choose not to enter the contract, but once you enter the contract you MUST pay the $98 next year.

Dedication or cash-flow matching is a technique used to fund known liabilities in the future. The intent is to form a portfolio of assets whose cash inflows will exactly match cash outflows. The liabilities will be paid off as they become due without the need to sell or buy assets in the future. The portfolio is formed today and held until all liabilities are paid off. Dedicated portfolios usually only consist of risk-free non-callable bonds and bond forwards since future cash inflows need to be known when the portfolio is constructed. Dedicated portfolios eliminate the risk caused by change in interest rates. Corporates, municipalities and pension funds routinely use such strategies. For example, corporates and municipalities sometimes want to fund liabilities stemming from projects that they might have initiated. To simplify the problem, assume that one can buy a fraction of a bond and forward, but cannot short bonds or forwards (there is therefore no opportunity for arbitrage).

In this project, we will be writing and using Python to construct a dedicated portfolio.

Specifics

1) Assume that your company is facing a stream of liabilities that it wants to be guaranteed to be able to pay using its cash reserves today. As the chief financial officer, you have decided to purchase a dedicated portfolio of bonds and forwards to hedge against interest rate risk. Formulate this portfolio construction problem as a linear programming problem. If you choose to enter any forward contracts, you must use your coupon and face payments from other bonds

and forwards that you purchase to make the future payment on the bond. You cannot hold money over at time 0 to pay the forward entrance prices.

2) Use gurobi to find the optimal mixture of bonds and forwards.
3) Two csv files are attached to this project, one with a list of information about the bonds, and one with information about your liabilities. Perform your analysis on these two sets of data.
4) Write a pdf file that does all of this and summarizes your analysis with graphs, text, and code chunks. The first few lines of the code in Python code file, in the first code chunk at the beginning, should include pd.read_csv call that read the csv files that include the liability stream and the available bonds and forwards. Be sure to include a noticeable comment that lets us know where you read the csv files. The template csv files fit the correct format. To be graded, we will change these pd.read_csv calls to load new csv files and re-run your code to see if you get the right answer on new data. Failure to run will automatically reduce your grade by 10 percentage points! Be sure that all your analysis in the Python code is generalized, so when we load the new csv files the output will be for the new data, instead of the template data. That means you should not hard code any numbers. Instead you should reference variable names for your output.
5) Some of your liabilities also are approximate due to the fact that some of your employees may quit without notice. This could reduce your liabilities, but some employees that quit must be replaced and new hires may demand a higher salary than your current employees, so this could also increase your liabilities. Be sure to provide a discussion of the flexibility of the portfolio in the event that liabilities change somewhat.
6) Finally, go get some real data on bond prices from https://www.wsj.com/market-data/bonds/treasuries . Notice that bond coupons are paid semi-annually on these bonds, so a coupon of 2.0, pays $1 in 6 months and $1 in 1 year, instead of $2 in 1 year. For simplicity, you can ignore this and just assume that all coupons are paid annually. There are no forward bonds in this data set, so you can ignore forwards for this part. We will pretend that today is 9/15/2022 and you will only buy bonds that mature on 9/15/… in future years. Write down the price (ask price), maturity and coupon for each relevant bond, be sure to note which day you accessed the webpage to get the prices. Use the bonds to create a dedicated portfolio on the same liabilities as loaded in the csv file from part 3; year 1 liabilities correspond to 9/15/2023, which can be paid using coupons on all bonds, and the face value payment from bonds that mature on 9/15/2023.
7) Grading will be based 66% on whether you get the right answer or not when we re-run your analysis with new data. If you don't get the right answer or your python code file doesn't run, we will go through your code and give partial credit accordingly. The easier it is to read your code the easier it is for us to understand what you're doing, so use a lot of comments in your code! The remaining 34% of your grade is based on the quality of your analysis and presentation of results in the pdf file. Write this as if you were actually going to present it to your boss.