Decision Analytics for Business and Policy

Project Description

Due Dates Proposal Nov 13, Presentations Dec 4-6, Report Dec 8

Project Overview.

In this project, you will formulate an optimization model, implement it Gurobi, and then implement a natural language wrapper for it. The purpose of the language wrapper is to reduce the analytical burden of revising the model and performing what-if analyses. Imagine you have done a lot of work to clean data and implement an optimization model to come up with an optimal production plan for a client. In a meeting with the client, you are asked: "I wonder how the solution changes if the market demand drops by 20% in January, and our supplier X is disrupted for 2 weeks at the beginning of the monsoon season." This language wrapper, designed correctly, can help you answer the question on the spot.

We will provide a few sample questions for you to choose from, but you can choose your own setting as well. Once you have chosen the question, you will formulate an optimization model and implement it in Python and Gurobi. At this moment, ChatGPT can not reliably answer mathematical questions on its own, even for the very simple ones. So your Gurobi and Python implementation of a decision model will serve as an analytical engine. For the language wrapper, you are going to use the recently developed open-source tool from Microsoft Research: https://github.com/microsoft/OptiGuide. This tool requires ChatGPT APIs. For the scope of the project, the 18 dollar free credits (per account) should be sufficient.

Timeline and Deliverables:

- Nov 6: form your team (2 to 4 people), email the names and emails of your teammates to the TAs.
- Nov 13: project proposal submission, one per team (10%), clearly state the question you are addressing, show you can find the right data, formulate an optimization model (in mathematics, not code), and show that you can successfully use the OptiGuide tool.
- Dec 4 and Dec 6: in-class presentations (10%).
- Dec 8: final report submission, one per team (20%).

In total the project is worth 40% of the course grade.

Details about Deliverables. Details for the deliverable (one submission per team):

1. Proposal:
    - The proposal should be 2 pages long, with the following content: (1) a clearly defined question to be explored; (2) a mathematical formulation of a decision problem that addresses this question; (3) dataset that you can use to solve the problem; (4) a statement that you can successfully use OptiGuide, or, describe why you can not get it working and what you have tried. The goal of this submission is for us to give you feedback to move along.
2. Presentation:

- 10-20 min for each team, depending on how many teams we have eventually.
- Organize your presentation flow in the same way you organize the final report. Beware of the time constraints, and put emphasis on clearly describing what problem you are trying to solve, what assumptions you are making, data sources, and the analysis.
- Presentation should be balanced among the team members.

3. Final report should include these parts and be submitted in a zip file:

   - Written report:
     - Cover page: project title, team information (names, Andrew IDs, emails, contributions from each team member).
     - Executive summary (1 page)
     - Analytical model and data ($<$ 3 pages): It should include problem statement, analytical formulations, data summary.
     - Analyese with the language wrapper (10 pages): Design 10 interesting questions to analyze variations of the analytical model (change objective, constraints) or get insights about the optimal solutions (why x_1 is 0 but x_1 is 1 in the optimal solution). For each question prompt, ask it a few times to check the consistency of responses. You can look at sample questions from OptiGuide.
     - Appendix: the link to and transcript of chat.

   - Python code, data, and other files that are required to run everything.
     - If you have multiple files, include a "ReadMe" file to explain which files should be executed and in what order, and/or the structure of the source/data file directories.
     - Your files should be self-standing and executable on other computers (e.g., do not hard-copy your local address in your code, use relative addresses).

Potential Topics.

- Supply chain resilience: https://pubsonline.informs.org/doi/pdf/10.1287/inte.2015.0804
- Finance: Exercise 14.8, or 14.9, or 14.10 in https://people.eecs.berkeley.edu/~elghaoui/ExManual.pdf
- Matching: https://stanford.edu/~alroth/match.html and https://www.nobelprize.org/uploads/2018/06/popular-economicsciences2012.pdf
- Or your own topic!