The Data Science Key to an Unpredictable Future:

Hands-on Guide to Solving Complex Challenges with Python and Gurobi

Ehsan Khodabandeh

Who Am I?

- Ehsan Khodabandeh
- Principal Operations Research Scientist at Decision Spot
- 7 years of industry experience
- PhD in Industrial Engineering
- Adjunct Lecturer at Northwestern MS in Analytics Program

Agenda

- What is optimization? Why should you know about it? What are its applications?
- How to frame business problems as optimization problems
- “Hello World” of optimization problems
- Case study
- Q&A

What is (Mathematical) Optimization?

- Finding the best solution, given limited resources, to meet the desired criteria by using math, analytics, and computer modeling.

Related Terms

- Optimization, Mathematical Programming, Decision Optimization, Decision Analytics, Prescriptive Analytics, Operations Research (OR)
    - Operations Research is the field where mathematical optimization is practiced

Mathematical Optimization vs Machine Learning

Machine Learning (ML)

- Goal: Descriptions
  - Insights from marketing campaigns
- Goal: Predictions
  - Predicting customer's purchase habits

- What should be done with your predictions?
  - This is where optimization works

Mathematical Optimization

- Goal: Prescriptions (Decisions)

Some Differences

ML

- You search for a model to fit the data
- Models are data-driven. They need to be re-trained over time with new data to stay fresh

Mathematical Optimization

- You build the model based on input data and search for the results
- Models are algorithm-based. Goals and requirements are stated algebraically

Source: https://www.gurobi.com/resource/mathematical-optimization-and-machine-learning

Examples of Collaboration Between ML and Optimization

- ML can help firms predict what supply chain issues might arise, and then Optimization can help them decide the least costly way to reroute shipments.
- ML can predict where the next cyberattack will originate, then Optimization can decide which investigators to assign based on their skills and the potential damage from an attack.
- ML can predict customer likelihood to buy more with targeted offers, then Optimization can decide how many discount coupons to offer to maximize revenue/profit.
- ML can predict machine failure, while Optimization can help decide when to stop production for maintenance to minimize costs and production disruptions.

Source: https://www.gurobi.com/resource/mathematical-optimization-and-machine-learning

Optimization Applications – A Few Examples

Supply Chain	Manufacturing	Airlines	Healthcare
Facility locations, truck routing, Container unloading, Inventory placement	Production planning and scheduling, Workforce planning	Crew scheduling, Flight to aircraft assignment	Nurse scheduling, Equipment and staff allocation

Energy	Finance	Sales & Marketing	Sports
Balancing supply and demand, Optimal energy resource planning to meet CO2 emission targets	Investment portfolio optimization, Cash allocation and management	Marketing campaign optimization, Sales territory allocation	NFL Scheduling, Marathon course design

Mathematical Optimization – Let's Get Mathy

- Mathematical Optimization covers a few different technologies. The most widely used of those technologies is Mixed Integer Programming (MIP).

What is MIP?

- Optimize an objective function over a set of decision variables subject to a set of constraints

Minimize $c^T x$

Subject to

$$Ax = b$$

$$l \leq x \leq u$$

Some or all x_j are integer

Framing Business Problems as Optimization Problems

- What you know → your data and parameters
- What you want to know → your decision variables
  - Examples: How many/much of …? Which product/location/worker/resource …?
- Your dependencies and limitations → your constraints
  - Examples: Capacity, staffing, raw material, time, budget, locations, regulatory restrictions, …
- Metrics, evaluation criteria, and success measures → your objectives
  - Examples: Costs, profits, service time, …

Mathematical Optimization Framework

Steps:

1. What needs to be maximized or minimized? Those are your objectives.
2. Think about your objectives again. What are your limitations? What stops you from having an infinitely good objective? Those are your constraints.
3. Given your constraints, what can you control to improve your objectives? Those are your decision variables.
4. Using these components, you create a system of linear equalities and inequalities.
5. You then solve them using a mathematical optimization solver
    - There are commercial solvers (such as Gurobi, CPLEX, XPRESS) or open-source solvers (such as COIN-OR, GLPK).

This process is iterative: model, solve, identify gaps, adjust, repeat

Diet Problem:
"Hello World" of Optimization Problems

- There are four categories of nutrients.
- Each person need to have somewhere between minimum and maximum recommended amount per day.
- You want to know how to get the required daily nutrition from a set of food while minimizing your cost.

Nutrients	Minimum Nutrition	Maximum Nutrition
Calories	1800	2200
Protein	91	-
Fat	0	65
Sodium	0	1779

Food	Cost
hamburger	2.49
chicken	2.89
hot dog	1.5
fries	1.89
macaroni	2.09
pizza	1.99
salad	2.49
milk	0.89
ice cream	1.59

Source: https://www.gurobi.com/documentation/9.5/examples/diet_py.html

Diet Problem:
“Hello World” of Optimization Problems

		Nutrients			
		Calories	Protein	Fat	Sodium
Food	hamburger	2.49	410	24	26
	chicken	2.89	420	32	10
	hot dog	1.5	560	20	32
	fries	1.89	380	4	19
	macaroni	2.09	320	12	10
	pizza	1.99	320	15	12
	salad	2.49	320	31	12
	milk	0.89	100	8	2.5
	ice cream	1.59	330	8	10

Source: *https://www.gurobi.com/documentation/9.5/examples/diet_py.html*

Diet Problem - Definition

Objective

- Minimizing total cost of buying food

Constraints

- The minimum and maximum levels of each nutrient should be met

Decision Variables

- How much of each food to buy

Diet Problem - Formulation

Sets:

- I: Set of foods
- J: Set of nutrients

Parameters:

- a_{ij}: amount of nutrient j in each serving of food i
- c_i: cost of each serving of food i
- M_j: minimum required level of nutrient j
- N_j: maximum required level of nutrient j

Decision Variables:

- x_i: amount (number of servings) of food i to buy

$$\min \sum_{i \in I} c_i x_i$$

Minimize total cost

Subject to:

$$\sum_{i \in I} a_{ij} x_i \geq M_j \quad \forall j \in J$$

$$\sum_{i \in I} a_{ij} x_i \leq N_j \quad \forall j \in J$$

For each nutrient category, total amount of nutrition that we get from the foods, should fall between the min and max required level.

$$x_i \geq 0 \quad \forall i \in I$$

We buy non-negative amount of food

Network Optimization Case Study

- A manufacturer has several plants around the US
- They are looking at adding a few warehouses to reduce cost and provide better services to their customers
- You have access to the locations of their plants and the total demand of last year
- We consider the plant locations as potential warehouse locations, a common practice
- Question: Where should they open the warehouses?

Network Optimization Case Study – Definition

Objective

- Reduce transportation cost and improve service → be closer to high-demand customers
  - Minimizing total weighted distance from the warehouses to customers

Constraints

- There is a limit on the number of warehouses to open
- All demands must be satisfied
- Each customer must be assigned to one warehouse

Decision Variables

- Where should we open a warehouse
- Which warehouse serves which customers

Network Optimization Case Study - Formulation

Sets:

- I: Set of warehouses
- J: Set of customers

Parameters:

- q_j: demand of customer j
- d_{ij}: distance between warehouse i and customer j
- P: maximum number of warehouses that can be used

Decision Variables:

- x_i: 1, if a warehouse is opened at location i, 0 otherwise
- y_{ij}: 1, if warehouse i is assigned to customer j, 0 otherwise

$$\min \sum_{i\in I, j\in J} q_j d_{ij} y_{ij}$$

Minimize total weighted distance

Subject to:

$$\sum_{i\in I} y_{ij} = 1 \quad \forall j \in J$$

Each customer is assigned to one warehouse

$$\sum_{i\in I} x_i \le P$$

Limit on the number of warehouses

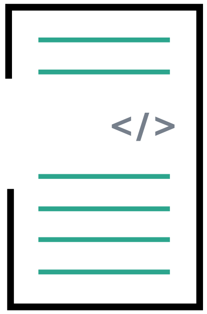
$$y_{ij} \le x_i \quad \forall i \in I, \forall j \in J$$

A customer can only be assigned to a warehouse, if the warehouse is open

$$x_i \in \{0,1\} \quad \forall i \in I$$
$$y_{ij} \in \{0,1\} \quad \forall j \in J$$

Variables are binary

Let’s see the model in Python



The code is available in this GitHub repo: https://github.com/decision-spot/net_opt

Resources & References

- “3 Reasons why optimization should be included in every data science graduate program”

  by Lindsay Montanari: https://tinyurl.com/t3xrduyn
- Supply Chain Network Design

  by Michael Watson, Sara Lewis, Peter Cacioppi, and Jay Jayaraman: http://networkdesignbook.com/
- INFORMS: https://www.informs.org/

  INFORMS: The Institute for Operations Research and the Management Sciences
- Gurobi: https://www.gurobi.com/
- Operations Research (OR) Stack Exchange: https://or.stackexchange.com/
- Burrito’s Game: https://www.gurobi.com/lp/academics/burrito-optimization-game/