Problem Set 2

Laker Newhouse
Collaborators: Evelyn Fu, Jacob Hansen

January 19, 2022

All code and raw results are available at `https://github.com/Arongil/6.S098/tree/main/pset2`.

1. (Linear programming with random cost vector) We are minimizing $c^T x$ subject to $Ax \preceq b$, where c is a random vector which is normally distributed with mean $\mathbb{E}\, c = c_0$ and $\mathbb{E}(c - c_0)(c - c_0)^T = \Sigma$.

   (a) To minimize the expected cost $\mathbb{E}\, c^T x$, we can equivalently minimize c_0^T because

   $$\mathbb{E}\, c^T x = E\left(\sum c_i x_i\right) = \sum \mathbb{E}(c_i x_i) = \sum \mathbb{E}(c_i) x_i = c_0^T x.$$

   (b) To minimize the risk-sensitive cost $\mathbb{E}\, c^T x + \gamma \mathbf{var}(c^T x)$, we can equivalently minimize

   $$c_0^T x + \gamma x^T \Sigma x$$

   because $x^T \Sigma x$ is a quadratic form which allocates the proper weights to pairs of entries in x based on the covariance matrix Σ. We know $c_0^T x$ is convex because it is affine, and $x^T \Sigma x$ is convex if and only if Σ is positive semidefinite. We know however that all covariance matrices are positive semidefinite. Therefore the problem is always convex given $\gamma \geq 0$.

   (c) If $\gamma < 0$, then the problem is convex if and only if Σ is negative semidefinite.

   (d) If we change the problem to that of minimizing β subject to $\mathbb{P}(c^T x \geq \beta) \leq \alpha$ and $Ax \preceq b$, then we need some more notation to describe when the problem is convex. Let

   $$\Phi(t) = \frac{1}{\sqrt{2\pi}} \int_t^\infty e^{-s^2/2} ds$$

   denote the probability that a Gaussian random variable with mean 0 and variance 1 is greater than t. Since each entry of x is Gaussian, the sum $c^T x$ is Gaussian, too. To account for the means and variances as given, we can say

   $$\mathbb{P}(c^T x \geq \beta) = \Phi\left(\frac{\beta - c_0^T x}{\|\Sigma^{1/2} x\|_2}\right).$$

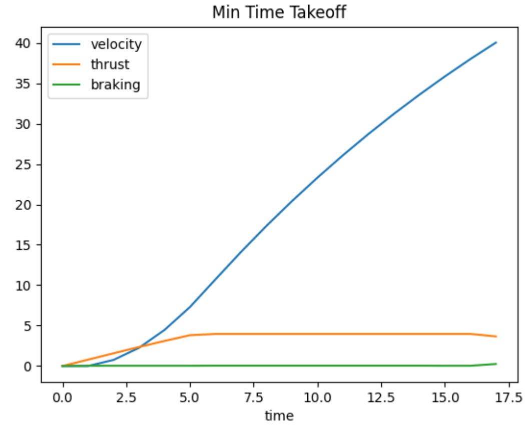
   Another way to think about $\|\Sigma^{1/2} x\|_2$ is as $(x^T \Sigma x)^{1/2}$. We divide by this term to account for the variance. Then, rearranging $\mathbb{P}(c^T x \geq \beta) \leq \alpha$, we find the equivalent condition that

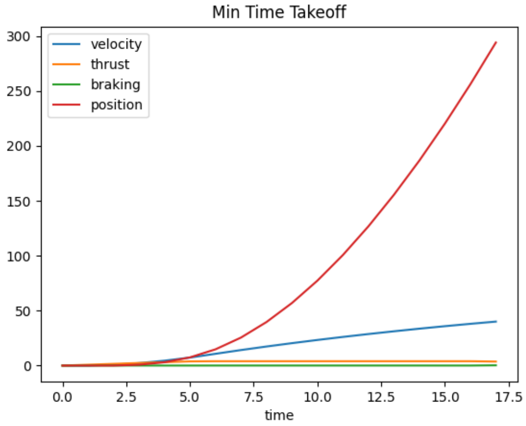
   $$c_0^T x + \Phi^{-1}(\alpha) \|\Sigma^{1/2} x\|_2 \geq \beta.$$

   The left side must be concave for the overall expression to be valid in a convex optimization problem. We know $\|\Sigma^{1/2} x\|_2$ is convex as a second order cone constraint, so the entire expression is concave so long as $\Phi^{-1} \leq 0$. That is equivalent to requiring that $\alpha \geq 1/2$. Therefore the problem is convex if and only if $\alpha \geq 1/2$. There is no way to configure α to seek risk, because fundamentally the problem is framed to put a limit on how often $c^T x \geq \beta$. For risk seeking, one would instead want the problem to, for example, put a limit on how often $c^T x \leq \beta$, and maximize β.

2. (Quickest takeoff) We describe how to find thrust and braking profiles that minimize the takeoff time. The idea is to solve many convex optimization problems. For $i = 1, 2, 3, \dots,$ we formulate a problem which *assumes* that we take off at time i. More precisely, the problem contains a constraint that $v_i \geq v^{\text{to}}$. We encode all of the other constraints directly. Then the first such problem which is feasible tells us the minimum takeoff time.

   When we solve the quickest takeoff problem with the given data, we find the quickest takeoff time is $T^{\text{to}} = 17$, and the takeoff position is $P^{\text{to}} = 294.05$. The plane brakes close to negligibly, while ramping up acceleration nearly as quickly as possible. Below is a graph of velocity, thrust, and braking. Another graph which includes position is also included, but it is separate because the large position values obscure detail in the other plots.





3. (House profit and imputed probabilities) A set of n participants bet on which of m outcomes, labeled $1, \dots, m$, will occur. Participant i offers to purchase up to $q_i > 0$ gambling contracts, at price $p_i > 0$, that the true outcome will be in the set $S_i \subset \{1, \dots, m\}$. The house then sells her x_i contracts, with $0 \leq x_i \leq q_i$. If the true outcome j is in S_i, then participant i reeives \$1 per contract, i.e. x_i. Otherwise, she loses and receives nothing. The house collects a total of $x_1 p_1 + \cdots + x_n p_n$ and pays out an amount that depends on the outcome j,

$$\sum_{i:j \in S_i} x_i.$$

   The difference is house profit.

(a) How should the house decide on x to maximize its worst case profit? We formulate the problem as a linear program. Immediately

$$\sum_{i:j\in S_i} x_i$$

is not clearly linear or convex, nor is optimizing over a discrete set $j \in \{1, \dots, m\}$. We reformulate the problem by introducing some new variables. First, call the above sum t, which is a number depending on the outcome j. Next, we organize the S_i into one $m \times n$ matrix S with

$$S_{ji} = \begin{cases} 1 & j \in S_i, \\ 0 & j \notin S_i. \end{cases}$$

The problem becomes to maximize $p^T x - t$ such that $x \geq 0$, $x \leq q$, and $Sx \leq t$. Here when we write $\leq t$ we mean that every entry of Sx is $\leq t$. Ideally we would set $Sx = t$, but t is not affine so instead we equivalently say $Sx \leq t$, which is the same because the objective maximizes $-t$, so increasing it is detrimental. Now we would like to collapse $p^T x - t$ into a single dot product. We define

$$c = \begin{bmatrix} p_1 & p_2 & \cdots & p_n & -1 \end{bmatrix}^T,$$
$$x' = \begin{bmatrix} x_1 & x_2 & \cdots & x_n & t \end{bmatrix}^T.$$

Now the objective becomes to maximize $c^T x'$. We collapse the constraints into a single matrix multiplication by defining the $(m+n) \times (n+1)$ matrix

$$A = \begin{bmatrix} I_n & 0 \\ S & -1 \end{bmatrix},$$

where I_n is the $n \times n$ identity matrix, S is stacked underneath it and is $m \times n$, and we pad on the right with a single column of n zeros to the right of I_n and m negative ones to the right of S. Define

$$b = \begin{bmatrix} q_1 & \cdots & q_n & 0 & \cdots & 0 \end{bmatrix}^T$$

to be $n + m \times 1$. Our constraint is

$$Ax' \leq b,$$

along with $x' \geq 0$. The first n rows of the inequality account for the constraints $x_i \leq q_i$, while the final m rows account for the constraints

$$t \geq \sum_{i:j\in S_i} x_i,$$

running over each possible j, and ensuring that t is greater than the maximum possible amount the house would need to pay out. Now an optimal x^* for this linear program is one which minimizes worst case house profits, as desired.

Suppose x^* maximizes the worst case house profit. There must exist a probability distribution π over the m outcomes for which x^* also maximizes the expeted house profit. For example, we could choose π to be all zero except for a one in the column representing the worst case option. However, there is a more systematic way to find π. In a way, maximizing expected profit instead of worst case profit is like switching the order of the min and max in the original expression

$$\max_x \min_\pi \text{profit}.$$

The original expression represents finding the maximal worst case profit. Switching the order means finding the probabilities which maximize expected profit, given we know x^*. This suggests solving the dual problem for insight. The dual problem is

$$\begin{aligned} &\text{minimize } b^T y \\ &\text{subject to } A^T y \geq c,\ y \geq 0. \end{aligned}$$

We claim the final m entries of y will be our desired probability distribution π, maximizing expected profit given x^*. The last row of $A^Ty \geq c$ yields the constraint $\mathbb{1}^T\pi \leq 1$. Why $\mathbb{1}^T\pi \geq 1$ for π as the final m entries of y is not clear to me, except that it works in all the nondegenerate examples I've tried. Similarly, I do not understand why the expected profit

$$x^{*T}(p - S^T\pi)$$

from π given x^* is equal to the maximal worst case profit, but again I have confirmed it experimentally. For the example given, the optimal worst case house profit is \$3.50. Here, we would use

$$x^* = (5, 5, 5, 5, 10).$$

If all offers were accepted, the worst case profit would be −\$5.00 in the case of outcome 2. The imputed probabilities come from solving the dual problem. We find

$$\pi = (0.09, 0.41, 0.12, 0.19, 0.19).$$

The expected profit given π and x^* is indeed \$3.50, the same as the optimal worst case profit. If we take π as the probability distribution and ask what x^* maximizes expected profit, we get the same answer back. Fair prices in this example are

$$p_{\text{fair}} = (0.5, 0.19, 0.48, 0.6, 0.12).$$

4. (Robust linear program with polyhedral cost uncertainty) Consider the problem

$$\begin{array}{ll} \text{minimize} & \sup_{c\in\mathcal{C}} c^Tx \\ \text{subject to} & Ax \succeq b, \end{array}$$

with variable $x \in \mathbb{R}^n$, where $\mathcal{C} = \{c \mid Fc \preceq g\}$. Assume $\mathcal{C} \neq \emptyset$ and that $Ax \succeq b$ is feasible.

(a) The function $f(x) = \sup_{c\in\mathcal{C}} c^Tx$ is convex because its epigraph is convex. And its epigraph is convex because the epigraph of a supremum is the intersection of the epigraphs of all the functions over which the supremum is taken. But each c^Tx is convex and thus has convex epigraph, and the intersection of convex functions is convex. Therefore the supremum that defines $f(x)$ is convex.

(b) The dual of the problem

$$\begin{array}{ll} \text{maximize} & c^Tx \\ \text{subject to} & Fc \preceq g, \end{array}$$

with variable c, is

$$\begin{array}{ll} \text{minimize} & g^Ty \\ \text{subject to} & F^Ty \geq x, \; y \geq 0. \end{array}$$

The optimal value of the dual problem is the same as the optimal value of the primal, $f(x)$, as a result of strong duality. This linear program exhibits strong duality because the primal is feasible.

(c) We formulate a single linear program equivalent to the original robust linear program by combining the answer to (b) with the constraint $Ax \succeq b$:

$$\begin{array}{ll} \text{minimize} & g^Ty \\ \text{subject to} & F^Ty \geq x, \; y \geq 0, Ax \succeq b. \end{array}$$

(d) We solve the robust linear program as described. The worst case cost $f(x)$ for x optimal in the robust LP is 3.17. The nominal cost for x optimal in the robust LP is 2.52. The cost for the nominal problem (i.e. when $\mathcal{C} = \{c_{\text{nom}}\}$) is 2.11. The robust optimum is worse than the nominal optimum as expected, because the robust scenario must contend with a variety of possible values for c whatever it picks for x. When choosing x optimal for the nominal problem, we see further improvement as one would expect, seeing as now x can be selected without worrying about robustness at all. The robust solution is a tradeoff: worse expected outcomes, but with a guarantee that the worst won't be too bad.

5. I have two final project ideas. Each has a potential problem, so I've emailed Alexander Amice for input. More input would be great if you have it!

   (a) Implement the branch and bound algorithm for integer programming. The algorithm uses linear programs as a subroutine. Would you count this as sufficient for a final project? More importantly, do you think it would be a good learning experience to implement this algorithm compared to a problem with some kind of real world data?

   (b) Currency arbitrage. I was initially very interested in this option, but I looked into various APIs for getting real time currency exchange rates and all have a paywall for more than, say, 100 requests per day. This isn't enough to support real time arbitrage. I need n exchange rate calls to build the matrix for a single arbitrage of n currencies, so I'm stuck here on how to get data if I want to move forward with this project idea.

   Remaining questions on my end: is (1) or (2) is a better option? Are there other ways to gather data on exchange rates?