CVXPY x NASA Course 2024

Philipp Schiele Steven Diamond Parth Nobel Akshay Agrawal

July 8, 2024

Geometric Programming and Aircraft Design

Outline

Linear measurements with IID noise

linear measurement model

$$y_i = a_i^T x + v_i, \quad i = 1, \ldots, m$$

- $x \in \mathbf{R}^n$ is vector of unknown parameters
- v_i is IID measurement noise, with density $p(z)$
- y_i is measurement: $y \in \mathbf{R}^m$ has density $p_x(y) = \prod_{i=1}^m p(y_i - a_i^T x)$

maximum likelihood estimate: any solution x of

$$\text{maximize} \quad l(x) = \sum_{i=1}^m \log p(y_i - a_i^T x)$$

(y is observed value)

MLE with exponentially distributed noise

Show how to solve the ML estimation problem when the noise is exponentially distributed, with density

$$p(z) = \begin{cases} (1/a)e^{-z/a} & z \geq 0 \\ 0 & z < 0, \end{cases}$$

where $a > 0$.

Solution

- Substitute

$$p(z) = \begin{cases} (1/a)e^{-z/a} & z \geq 0 \\ 0 & z < 0, \end{cases}$$

  into MLE problem

$$\text{maximize} \quad l(x) = \textstyle\sum_{i=1}^{m} \log p(y_i - a_i^T x)$$

- Let $\lambda = 1/a$. The log-likelihood l is

$$\begin{aligned} l(x) &= \sum_{i=1}^{m} \log(\lambda \cdot e^{(a_i^T x - y_i)\lambda} \cdot \mathbf{1}[y_i - a_i^T x \geq 0]) \\ &= \sum_{i=1}^{m} \log \lambda + (a_i^T x - y_i)\lambda + \log \mathbf{1}[y_i - a_i^T x \geq 0]) \end{aligned}$$

Solution

- To maximize $l(x)$, maximize

$$\sum_{i=1}^{m}(a_i^T x - y) = \mathbf{1}^T(Ax - y), \quad Ax \leq y$$

  (can ignore constant λ)

- Compute the maximum likelihood estimate of x by solving the LP

$$\begin{array}{ll} \text{maximize} & \mathbf{1}^T(Ax - y) \\ \text{subject to} & Ax \preceq y \end{array}$$

Comparing regularizers

`https://marimo.app/l/dh54bd`

Outline

Homework review

This lecture

Geometric programming

Aircraft design

Other GP compatible models

Monomial function fitting

This lecture

- This lecture is about a specific kind of convex optimization problem: the **geometric program (GP)**
- Geometric programs are widely used in engineering design: chemical engineering, circuit design, transformer design, communications, aircraft design, ...
- They are nonconvex in their natural form, but can be made convex via a change of variables
- Popularized in aero/astro by NASA astronaut Woody Hoburg
- Example: size a wing to minimize the total drag

Readings

- A Tutorial on Geometric Programming [Boyd, et al]
- Geometric Programming for Aircraft Design Optimization [Hoburg, Abbeel]
- (Optional) Hyperloop System Optimization [Kirschen, Burnell]

Motivation from aircraft design

- Design optimization requires solving many problems, sweeping out trade-off curves
- Requires reliable and efficient optimization methods
- Geometric programs well-suited to physics-based models, and can be readily solved with convex optimization (in contrast to nonlinear or black-box optimization methods)
- GPs can't model arbitrary nonlinear relationships ...
- ... but are surprisingly expressive, and can employ effective approximation techniques

Outline

Geometric programming

- **monomial function**:

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \quad \mathbf{dom} f = \mathbf{R}^n_{++}$$

  with $c > 0$, $a_i \in \mathbf{R}$.

- **posynomial function**: sum of monomials

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \quad \mathbf{dom} f = \mathbf{R}^n_{++}$$

- **geometric program (GP)**

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, m \\ & g_i(x) = 1, \quad i = 1, \ldots, p \end{array}$$

  with f_i posynomial, g_i monomial.

Converting geometric programs to convex form

- The standard form of the geometric program is non-convex
- Can be made convex via a **log-log transformation**:
- 1. Change variables $y_i = \log x_i$
- 2. Take logarithm of objective, constraints

Log-log transformation of monomials

- monomial function:

$$f(x) = c x_1^{a_1} x_2^{a_2} \cdots x_n^{a_n}, \quad \mathbf{dom}\, f = \mathbf{R}^n_{++}$$

  with $c > 0$, $a_i \in \mathbf{R}$.
- Substitute $y_i = \log x_i$:

$$f(x) = f(e^{y_1}, \ldots, e^{y_n}) = c e^{y_1 a_1} \cdots e^{y_n a_n}$$

- Take logarithm of f:

$$\log f(x) = \log f(e^{y_1}, \ldots, e^{y_n}) = a^T y + b, \quad b = \log c$$

- monomial functions are **affine** in $y \in \mathbf{R}^n$ after log-log transformation

Log-log transformation of posynomials

- **posynomial function**: sum of monomials

$$f(x) = \sum_{k=1}^{K} c_k x_1^{a_{1k}} x_2^{a_{2k}} \cdots x_n^{a_{nk}}, \quad \mathbf{dom}\, f = \mathbf{R}^n_{++}$$

- Substitute $y_i = \log x_i$:

$$f(x) = f(e^{y_1}, \ldots, e^{y_n}) = \sum_{k=1}^{K} e^{a_k^T y + b_k}, \quad b_k = \log c_k$$

- Take logarithm of f:

$$\log f(x) = \log f(e^{y_1}, \ldots, e^{y_n}) = \log \sum_{k=1}^{K} e^{a_k^T y + b_k}$$

- posynomial functions are **convex** in $y \in \mathbf{R}^n$ after log-log transformation

Geometric program in convex form

- Start with a GP

$$
\begin{array}{ll}
\text{minimize} & f_0(x) \\
\text{subject to} & f_i(x) \le 1, \quad i = 1, \ldots, m \\
& g_i(x) = 1, \quad i = 1, \ldots, p
\end{array}
$$

  where f_i are posynomials and g_i are monomials.

- Taking log-log transformation of objective and constraints converts geometric program into a convex optimization problem:

$$
\begin{array}{ll}
\text{minimize} & \log \sum_{k=1}^{K} \exp(a_{0k}^T y + b_{0k}) \\
\text{subject to} & \log \sum_{k=1}^{K} \exp(a_{ik}^T y + b_{ik}) \le 0, \quad i = 1, \ldots, m \\
& Gy + d = 0.
\end{array}
$$

Example with CVXPY

`https://marimo.app/l/uttpdr`

Trade-off Analysis

- Consider perturbed GP

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq u_i, \quad i = 1, \ldots, m \\ & g_i(x) = v_i, \quad i = 1, \ldots, p \end{array}$$

  where u_i, v_i are positive constants, with optimal value $p(u, v)$.
- for $u_i > 1$, ith inequality constraint loosened by $(100)(u_i - 1)$ percent
- for $u_i > 1$, ith inequality constraint tightened by $(100)(u_i - 1)$
- Can study optimal trade-off between constraints and objective by resolving problem while perturbing one constraint, holding others constant.

Sensitivity Analysis

- The optimal dual variable $\lambda^\star$ of the inequality constraint functions satisfies

$$\lambda_i^\star = \frac{\partial \log p}{\partial u_i}\bigg|_{u=1,v=1}$$

- Because

$$\frac{\partial \log p(u,v)}{\partial u_i} = \frac{1}{p(u,v)} \frac{\partial p(u,v)}{\partial u_i}$$

for small perturbations,

$$\lambda_i^\star \approx \Delta p / p^\star,$$

that is, the sensitivity $\lambda_i^\star$ gives the *fractional* change in the optimal value per fractional change in inequality i

Exercise

- Warm up for aircraft design problem
- In this exercise, we maximize the shape of a box with height h, width w, and depth d, with limits on the wall area $2(hw + hd)$ and the floor area wd, subject to bounds on the aspect ratios h/w and w/d. The optimization problem is

$$\begin{array}{ll} \text{maximize} & hwd \\ \text{subject to} & 2(hw + hd) \leq A_{\text{wall}}, \\ & wd \leq A_{\text{flr}}, \\ & \alpha \leq h/w \leq \beta, \\ & \gamma \leq d/w \leq \delta. \end{array}$$

with variables $h > 0$, $w > 0$, and $d > 0$

Exercise

`https://marimo.app/l/g8julj`

Exercise solution

`https://marimo.app/l/b2hvro`

Outline

Wing sizing

- Our goal is to size a wing with total area S, span b, and aspect ratio $A = b^2/S$. These parameters should be chosen to minimize the total drag $D = 1/2\rho V^2 C_D S$, where ρ is the density of air, V is the cruising speed, and C_D is the drag coefficient.
- The drag coefficient C_D is modeled as the sum of the fuselage parasite drag, wing parasite drag, and induced drag:

$$C_D = (\text{CDA}_0)/S + kC_f + S_{\text{wet}}/S + \frac{C_L^2}{\pi A e},$$

where $(\text{CDA}_0)/S$ is the fuselage drag area, k is a form factor for pressure drag, S_{wet}/S is the wetted area ratio, C_L is the lift coefficient, and e is the Oswald efficiency factor.

Wing sizing

- The skin friction C_f can be approximated as

$$C_f = 0.074/\text{Re}^2$$

where $Re = \rho V/\mu\sqrt{(}S/A)$ is the Reynolds number at mean cord $c = \sqrt{S/A}$.

- The total aircraft weight W is the sum of a fixed weight W_0 and the wing weight W_w:

$$W = W_0 + W_w.$$

Wing sizing

- The wing weight is

$$W_w = 45.42S + 8.71 \cdot 10^{-5} N_{\text{ult}} A^{3/2} \sqrt{W_0 W}/\tau,$$

  where N_{ult} is the ultimate load factor for structural sizing, and τ is the airfoil thickenss to chord ratio.
- The weight equations are coupled to the drag equations by the constraint that lift equals weight,

$$W = 1/2 \rho V^2 C_L S.$$

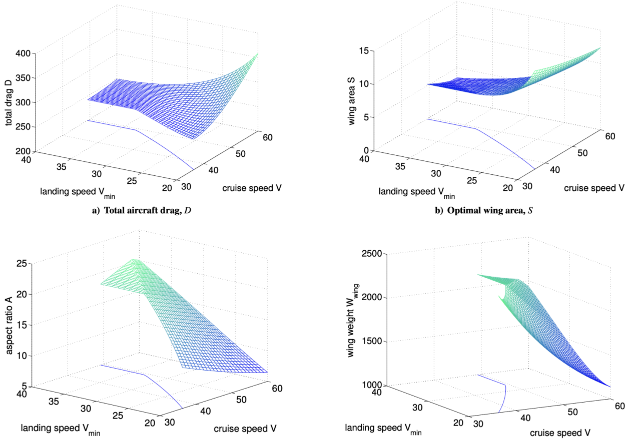
Wing sizing

- Finally, for a safe landing, the aircraft should be capable of flying at a reduced speed V_{min} subject to a stall constraint,

$$\frac{2W}{\rho V_{\text{min}}^2 S} \leq C_{L,\max}.$$

Tradeoff surfaces

Solve 775 problems in 3 seconds to compute tradeoff surfaces:



a) **Total aircraft drag,** D

b) **Optimal wing area,** S

c) **Optimal aspect ratio,** A

d) **Optimal wing weight,** W_w

Relaxation

- Consider the following program

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, m \\ & h(x) = 1 \end{array}$$

  where the f_i and h are posynomials.
- This is **not** a GP, because $h(x) = 1$ is a posynomial equality constraint.

Relaxation

- If there exists an index r such that
  - f_0 is increasing in x_r,
  - f_i are nondecreasing in x_r,
  - h is decreasing in x_r,

  then any solution $x^\star$ of the *relaxed* problem

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 1, \quad i = 1, \ldots, m \\ & h(x) \leq 1 \end{array}$$

  satisfies $h(x^\star) = 1$. This means we can solve the original nonconvex problem with the relaxed GP.

Wing sizing and relaxation

- Many of the constraints in the wing sizing problem are posynomial equality constraints, making it not a GP
- Relaxing these constraints yields an equivalent problem!
- For example,

$$C_D = (\mathrm{CDA}_0)/S + kC_f + S_{\mathrm{wet}}/S + \frac{C_L^2}{\pi A e},$$

can be replaced with

$$C_D \geq (\mathrm{CDA}_0)/S + kC_f + S_{\mathrm{wet}}/S + \frac{C_L^2}{\pi A e},$$

since the objective (total drag) $D = 1/2\rho V^2 C_D S$ is increasing in C_D (exercise: check that the other conditions hold).

Homework

- Formulate the wing sizing problem as a geometric program, and solve it using CVXPY.
- Will require use of relaxations.
- Notebook has problem data and variable definitions.
- `https://marimo.app/l/xnvuat`
- Reference: §3 of Geometric Programming for Aircract Design Optimization
- `https://people.eecs.berkeley.edu/~pabbeel/papers/2012_gp_design.pdf`

Outline

Homework review

This lecture

Geometric programming

Aircraft design

Other GP compatible models

Monomial function fitting

Steady level flight relations

- The steady level flight relations constrain lift to equal the aircraft's total weight and thrust to equal force of drag:

$$
\begin{aligned}
L &= W \\
T &= D \\
L &= 0.5\rho V^2 C_L S \\
D &\geq 1/2 \rho V^2 C_D S
\end{aligned}
$$

  where L is the lift force, W is the total aircraft weight, T is the thrust force, ρ the density of air, V the flight speed, C_L the lift coefficient, S the wing area, and C_D the total drag coefficient
- These are each monomial constraints

Total aircraft weight

- Total aircraft weight W is a sum of component weights
- For example
  $$W_{zfw} \geq W_{fixed} + W_{payload} + W_{wing} + W_{tail} + \cdots$$
  and
  $$W \geq W_{zfw}(1 + \theta_{\text{fuel}},$$
  where θ_{fuel} is the fuel mass fraction W_{fuel}/W_{zfw}.
- These are posynomial constraints
- Weights can be further broken down

Chain of efficiencies

- Chain of efficiencies η relate cruise thrust power to fuel power:

$$TV \leq P_{\text{fuel}}\eta_{\text{eng}}\eta_{\text{prop}}$$

  where $P_{\text{fuel}} = mh$ is the is the mass flow rate times heating value, η_{eng} is the engine's fuel power to shaft power conversion efficiency, and η_{prop} is the propulsive shaft power to thrust power conversion efficiency.
- Monomial constraint

Outline

Monomial function fitting

- Can fit monomial (or posynomial) functions that approximately match observed data
- Useful when design constraints not from known laws of physics but from observations
- Given data points
$$(x^{(i)}, f^{(i)}), \quad i = 1, \ldots, N,$$
where $x^{(i)} \in \mathbf{R}^n_{++}$ are positive vectors and $f^{(i)}$ are positive constants.
- Goal is to fit the data with a monomial $f(x) = cx_1^{a_1} \cdots x_n^{a_n}$
- Find $c > 0$ and $a_1, \ldots, a_n$ so that
$$f(x^{(i)}) \approx f^{(i)}$$

Monomial function fitting

- Can fit via standard regression techniques like least squares
- Let $y^{(i)} = \log x^{(i)}$
- Seek $\log f^{e^{y^{(i)}}} \approx \log f^{(i)}$, so that

$$\log c + a^T y^{(i)} \approx \log f^{(i)}, \quad i = 1, \ldots, N.$$

- Fit with least squares

$$\text{minimize} \quad \textstyle\sum_{i=1}^N (\log c + a^T y^{(i)} - \log f^{(i)})^2$$

or any other regression technique.

Posynomial function fitting

- Also possible to fit posynomial functions to data
- See §8.5 of A Tutorial on Geometric Programming

Summary

- Geometric programs are nonconvex optimization problems on postive variables that become convex after a change of variables and log transformation
- Wide variety of engineering design problems naturally formulated as geometric programs
- Tricks like relaxations and function approximation can be used when problems are almost but not quite geometric programs
- Geometric programs can be solved quickly (ms) and reliably, making it possible to use them in tight inner loops when exploring tradeoff surfaces
- Optimal dual variables convey percent change in optimal value given percent change in constraints