CS 524: Introduction to Optimization
Lecture 32 : Conic optimization

Michael Ferris

Computer Sciences Department
University of Wisconsin-Madison

November 17, 2023

New Problem: Sylvester

Round-Up

- Given points $\{p^1, p^2, \ldots p^m\}$ with $p_i \in \mathbb{R}^n$, find the smallest sphere that encloses all the points

New Problem: Sylvester (see `sylvester.gms`)

Round-Up

- Given points $\{p^1, p^2, \ldots p^m\}$ with $p_i \in \mathbb{R}^n$, find the smallest sphere that encloses all the points

- Let $x \in \mathbb{R}^n$ be the center coordinates
- r be the radius, use constraints (1) or (2)

$$\min r$$

$$r \geq \sqrt{\sum_{j=1}^{n} (p_j^i - x_j)^2} \quad \forall i = 1 \ldots m \tag{1}$$

$$r^2 \geq \sum_{j=1}^{n} (p_j^i - x_j)^2 \quad \forall i = 1 \ldots, m, \; r \geq 0 \tag{2}$$

Extension: Second-order cone

A second-order cone is the set of points $x \in \mathbb{R}^n$ satisfying:

$$\|Ax + b\| \leq c^T x + d$$

Special cases:

- If $A = 0$, we have a linear constraint (hyperplane)
- If $c = 0$, $d \geq 0$, can square both sides (ellipsoid)

Counter example:

In general you cannot square both sides.
If $A = [1\ 0]$ and $c^T = [0\ 1]$, $b = d = 0$, we have:

$$|x_1| \leq x_2$$

Squaring both sides, we get a nonconvex quadratic constraint:

$$x_1^2 - x_2^2 \leq 0$$

Implementation details

A second-order cone program (SOCP) has the form:

$$\begin{aligned} \min_x \quad & c^T x \\ \text{subject to:} \quad & \|A_i x + b_i\| \le c_i^T x + d_i, \text{for } i = 1, \ldots, m \\ & Fx = g \end{aligned}$$

- Every LP is an SOCP (just make each $A_i = 0$)
- Every convex QP and QCQP is an SOCP (see later)
- In GAMS, you can specify SOCP constraints directly in QCP. Solvers include Mosek, Gurobi, Cplex (see `32introsocp.ipynb`)
- Solvers often can recognize convex constraint structure automatically (see `32convexqcp.ipynb`)

Ellipsoids

- For linear constraints, the set of x satisfying $c^T x = b$ is a **hyperplane** and the set $c^T x \leq b$ is a **halfspace**.
- For quadratic constraints, what is the set $x^T Q x \leq b$?

If $Q \succ 0$, the set $x^T Q x \leq b$ is an **ellipsoid**.

Ellipsoids

Suppose $Q \succ 0$. We know from before that: $x^\mathsf{T} Q x = z^\mathsf{T} \Lambda z$ where we defined the new coordinates $z = V^\mathsf{T} x$.

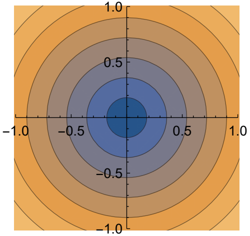
The set of x satisfying $x^\mathsf{T} Q x \leq 1$ corresponds to the set of z satisfying $\lambda_1 z_1^2 + \cdots + \lambda_n z_n^2 \leq 1$.

- In the z coordinates, this is a stretched sphere (ellipsoid). In the z_i direction, it is stretched by $\frac{1}{\sqrt{\lambda_i}}$.
- In the x_i coordinates, it is just a rotated ellipsoid, since the relationship between x and z coordinates is an isometry.
- The principal axis in the z_i direction corresponds to:

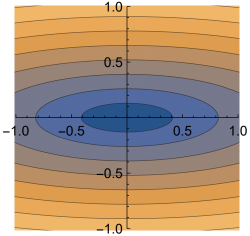
$$x = Vz = \begin{bmatrix} v_1 & \dots & v_n \end{bmatrix} e_i = v_i$$

Degenerate and ill conditioned ellipsoids

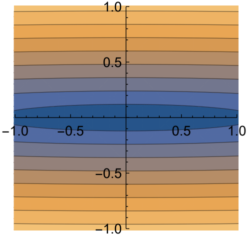
Ellipsoid axes have length $\frac{1}{\sqrt{\lambda_i}}$. When an eigenvalue is close to zero, contours are stretched in that direction.



$x^2 + y^2$



$\frac{1}{10}x^2 + y^2$



$\frac{1}{100}x^2 + y^2$

- Warmer colors = larger values
- If $\lambda_i = 0$, then $Q \succeq 0$ and ellipsoid $x^T Qx \leq 1$ is degenerate (stretches out to infinity in direction v_i)
- Degenerate (or ill-conditioned - large variation in λ_i) problems are harder to solve numerically (solvers may take many iterations)

Norm constraints

Constraints of the form $\|Ax - b\|^2 \leq c$ are (possibly degenerate) ellipsoids.

Proof: When we expand the square, we get the quadratic $x^\mathsf{T} A^\mathsf{T} A x - 2b^\mathsf{T} A x + b^\mathsf{T} b$. But notice that:

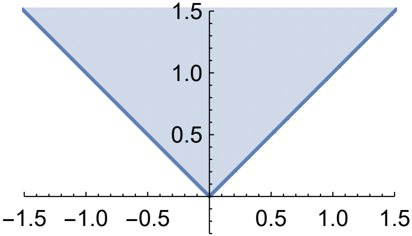
$$x^\mathsf{T} A^\mathsf{T} A x = \|Ax\|^2 \geq 0$$

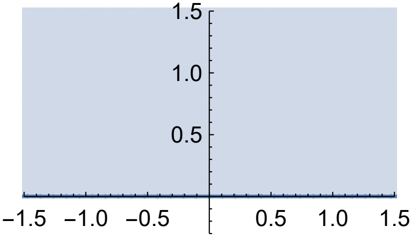
Therefore, $A^\mathsf{T} A \succeq 0$, so we must have an ellipsoid. In the case where $A^\mathsf{T} A$ is invertible (A is tall with linearly independent columns), the ellipsoid will be non-degenerate.

Conic Programming; $\mathcal{C}$ is a (convex) cone

$$\min c^T x$$
$$\text{subject to: } Ax = b$$
$$x \in \mathcal{C}$$

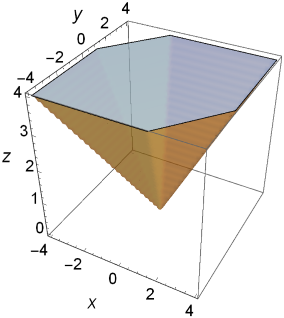
- $\mathcal{C}$ is a cone if $x \in \mathcal{C}, \alpha > 0 \Rightarrow \alpha x \in \mathcal{C}$
- A cone is *convex* if in addition
  - $x + y \in C$ whenever $x \in C$ and $y \in C$
- Similar to a subspace, but $\alpha > 0$ (not $\alpha \in \mathbb{R}$ - a critical difference!)
- Simple examples: $|x| \leq y$ and $y \geq 0$

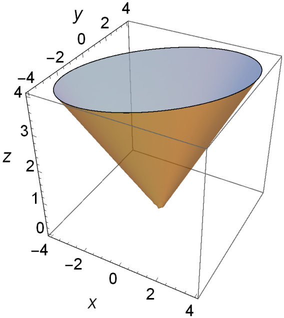


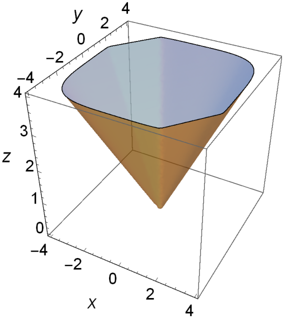


What is a cone?

- A *slice* of a cone is its intersection with a linear manifold
- We are interested in *convex cones* (all slices are convex)
- Can be polyhedral, ellipsoidal, or something else...







Special Cases of Conic Programming

$$\min_x \{c^T x \mid x \in \mathcal{C},\ Ax = b\}$$

- $\mathcal{C} = \mathcal{C}_\ell^n \stackrel{\text{def}}{=} \{x \in \mathbb{R}^n : \ x \geq 0\} \Rightarrow$ Linear Programming
- $\mathcal{C} = \mathcal{C}_q^n \stackrel{\text{def}}{=} \{(x, z) \in \mathbb{R}^n \times \mathbb{R} : \ z \geq \|x\|_2 = \sqrt{\sum_{j=1}^n x_j^2}\} \Rightarrow$ Second Order Cone Programming
- $\mathcal{C} = \mathcal{C}_r^n(\alpha) \stackrel{\text{def}}{=} \{(x, y, z) \in \mathbb{R}^n \times \mathbb{R}_+^2 : \alpha yz \geq \|x\|_2^2 = x^T x\} \Rightarrow$ (Rotated) Second Order Cone Programming
- $\mathcal{C} = \mathcal{S}_+^n \stackrel{\text{def}}{=} \{X = X^T : X \succeq 0\}$ (Semidefinite Programming)

Notes: other desirable properties that these cones have are closed, pointed ($\mathcal{C} \cap -\mathcal{C} = \{0\}$), convex (closed under addition), self dual ($C = -C^\circ = C^*$ for $\mathcal{C}_r^n(2)$) and have non-empty interior.

Second order cone programming fits our conic model by adding variables (w, z):

$$w = Ax + b, \; z = c^T x + d, \; \|w\| \leq z \Leftrightarrow (w, z) \in \mathcal{C}_q^n$$

Note that $d \geq 0$ is needed to square both sides in SOC because:

$$\begin{aligned} \mathcal{C}_q^n &:= \left\{ (x, z) \in \mathbb{R}^n \times \mathbb{R} : z \geq \|x\|_2 = \sqrt{\sum_{j=1}^n x_j^2} \right\} \\ &= \left\{ (x, z) \in \mathbb{R}^n \times \mathbb{R} : z \geq 0, \; z^2 \geq \sum_{j=1}^n x_j^2 \right\} \end{aligned}$$

In GAMS, we thus specify SOC using the second formulation (with $z \geq 0$)

Could also formulate for SOCP solvers (adding equations)

While constraints (2) allow specification as QCP in GAMS, some solvers require special forms to recognize the second order cones. e.g.

$$\min r$$
$$d_{ij} = p^i_j - x_j \quad \forall i \in M, j \in N$$
$$r^2 \geq \|d_{ij}\|_2^2 \quad \forall i \in M, j \in N,\ r \geq 0$$

GAMS Code

- `positive variable y(M)`
- `sqr(y(M)) =G= sum(N,sqr(d(M,N)))`
- Use problem type `qcp`
- Use `option qcp=mosek`
- Variables cannot appear in more than one cone (for mosek)