SEMIDEFINITE PROGRAM BASICS

BRIAN AXELROD

Abstract. A introduction to the basics of Semidefinite programs.

Contents

1. Definitions and Preliminaries

Before going into Semidefinate programs we define a few essential concepts.

1.1. Linear Algebra.

The following are equivalent:

- A is positive semidefinite
- $A = T^T T$ for some matrix T (i.e. A has a square root)
- $A = Q \Lambda Q^{-1}$ for some Q orthogal, Λ diagonal with $\Lambda_{ii} \geq 0$.
- $x^T A x \geq 0, \forall x$
- $A \succeq 0$

As are the following:

- A is positive definite
- $A = T^T T$ for some **nonsingular** matrix T (i.e. A has a square root)
- $A = Q \Lambda Q^{-1}$ for some Q orthogal, Λ diagonal with $\Lambda_{ii} > 0$.
- $x^T A x > 0, \forall x$
- $A \succ 0$

Definition 1.1. $A \bullet X = Tr(AX) = \sum A_{ij} X_{ij}$

Definition 1.2. A bilinear form $< x, y >_S$, $S \succ 0$ is defined as $x^T S y$. If we don't specify S we assume that $S = I$, i.e. the standard square norm.

Date: DEADLINE AUGUST 26, 2011.

Theorem 1.3. *A bilinear form has the following properties:*

- *A bilinear form is a norm*
- $< x, y >=< y, x >$
- $< \lambda_1 x, \lambda_2 y >= \lambda_1 \lambda_2 < x, y >$
- $< x, a + b >=< x, a > + < x, b >$

Definition 1.4. An adjoint mapping of a linear map A is defined as the unique linear map $A^\star$ such that:

$$(A^\star y)^T x = y^T (Ax), \forall x, y$$

1.2. **Convex Analysis (on $\mathbb{R}^n$).**

Definition 1.5. We say $A \subset \mathbb{R}^n$ is convex if for all $x_1, x_2 \in A$ $\lambda x_1 + (1 - \lambda) x_2 \in A$ for $\lambda \in [0, 1]$.

Definition 1.6. We say a set C is a proper cone if:

- C is convex
- $0 \in C$
- $x \in C \to \lambda x \in C$ for $\lambda \in [0, \infty)$ (this is the definition of a cone)
- The interior of C is not empty
- C is closed

Definition 1.7. The dual $V*$ to a vector space V is the space of real-valued linear functionals on that space. [Parrilo]. It is also a vector space.

Definition 1.8. The dual cone $C^\star$ to a cone C is defined as $C^\star = \{y | x^t y \geq 0, \forall x \in C\}$

Theorem 1.9. *The positive orthant,* $\{(x_1 ... x_n) | x_i \geq 0\}$ *is a proper cone.*

Theorem 1.10. *The set of positive semidefinite matrices* S^+ *is a proper cone in the set of symmetric matrices.*

Proof. First we show that S^+ is a cone. Take any $X \in S^+$. Let $Q \Lambda Q^{-1}$ be it's eigendecomposition. For $\lambda \in [0, \infty)$, $\lambda X = \lambda Q \Lambda Q^{-1} = Q \lambda \Lambda Q^{-1}$. Thus $\lambda \Lambda_{ii} \geq 0$ and $\lambda X \in S^+$.

Next we observe that the zero matrix, $0 \in S^+$. $x^t 0 x = 0, \forall x$. Thus $0 \in S^+$.

Finally we show that the interior of S^+ is nonempty. We claim that $I \in Int(S^+)$. We show there exists an ϵ for all X such that $I + \epsilon X \in S^+$. Take λ the smallest eigenvalue of X. Let $\epsilon = \frac{|\lambda|}{2}$. Then the eigenvalues of $I - \epsilon X$ are all positive and S^+ has a nonempty interior, meaning that S^+ is a proper cone. □

Theorem 1.11. *The set of positive semidefinite matrices is equal to the interior of the set of positive definite matrices.*

2. SDP Basics

2.1. Formulation.
We define a semidefinite program as follows:

$$\begin{aligned} \text{minimize} \quad & C \bullet X \\ \text{subject to} \quad & A_i \bullet X = b_i \\ & X \succeq 0 \end{aligned}$$

Note that if we consider x as the vector representaiton of X, $C \bullet X = c \cdot x$ and the linear constraints can be expressed as $Ax = b$.

The solution spaces to SDPs, slices of the PSD cone, can be quite complicated and nonintuitive. We show one example in figure 1.

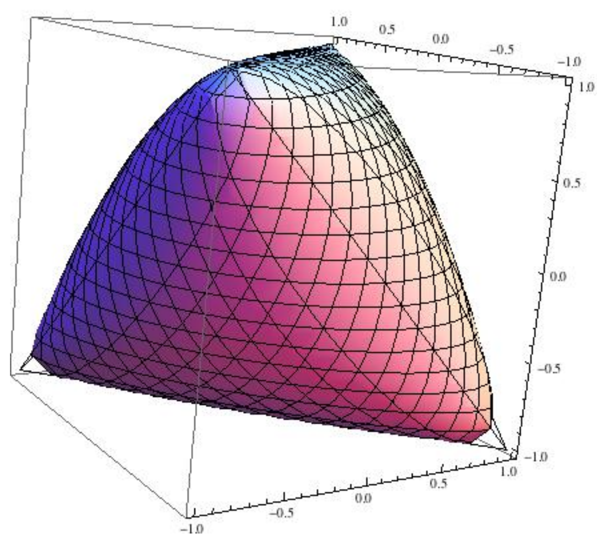

Figure 1. http://i.stack.imgur.com/UeqY1.jpg

It turns out that solutions to certain SDPs require exponentially many bits. This forces us to relax our requirements on SDP solvers. Instead we look for algorithms polynomial in n, m, where n is the size of the input and m is the required precision. One way of checking the precision of the output of an SDP solver is by examining primal-dual pairs.

2.2. Conic (and SDP) Duality. This presentation of conic duality comes directly from Parrilo's 6.256 lecture notes
The following is a primal-dual pair for conic programs.

$$\begin{aligned} \text{minimize} \quad & c \cdot x \\ & Ax = b \\ & x \in \mathcal{K} \end{aligned} \qquad \begin{aligned} \text{maximize} \quad & y \cdot b \\ & c - A^\star y \in \mathcal{K} \end{aligned}$$

Theorem 2.1. *The primal serves as an upper bound for the dual and the dual serves as a lower bound for the primal. This allows us to certify the precision of a solution.*

Proof.

$$\begin{aligned} c \cdot x - y \cdot b &= c \cdot x - y \cdot (Ax) \\ &= c \cdot x - (A^\star y) \cdot x \\ &= (c - A^\star y) \cdot x \geq 0 \end{aligned}$$

□

Theorem 2.2. *The duality gap is zero for strictly feasible SDPs. The proof is beyond the scope of this lecture.*

Theorem 2.3. *If the primal is infeasible, i.e. there does not exist an $x \in \mathcal{K}$ s.t. $Ax = b$, then there exists a y s.t. $y \cdot b < 0, A^\star y \in K^\star$.*

Proof. This is equivalent to saying that b is not in the image of $\mathcal{K}$ under A.
We know that $\mathcal{K}$ is convex so $A(\mathcal{K})$ is convex.
If b is not in the set we take a linear functional y such that y corresponds to a seperating hyperplane positive on $A(\mathcal{K})$ and negative on b.

$$\begin{aligned} y \cdot A(x) &\geq 0, x \in \mathcal{K} \\ (A^\star y) \cdot &\geq 0 \\ A^\star(y) &\in \mathcal{K}^\star \end{aligned}$$

Thus the y that satisfies the above relations acts as a seperating hyperplane between $A(\mathcal{K})$ and b. It is called a certificate of infeasibility. □

3. Applications

3.1. **Euclidean Embedding.** Given a graph g, wich weighted edges corresponding to distances, does there exist a set of points $\{x_i\}$ corresponding to the nodes v_i s.t. $||x_i - x_j|| = w_{ij}$ where w_{ij} is the weight of the edge connecting nodes i, j or zero if no such edge exists.

Asnwer: It exists iff the weight matrix is negative semidefinite orthogonal to $e = [1, 1, ...]$, and in such cases the x_i's are computable via an SDP.
Gram matrix $= [x_1...x_n]^T[x_1, ...x_n]$.
$D = diag(G) \cdot e^T + ediag(G)^T - 2G$.
By changing coordinates such that $x_1 = 0$ we get:
$G_{ij} = \frac{1}{2}(d_{1i}^2 + d_{1j}^2 - d_{ij}^2)$

References

[1] Pablo Parrilo lgebraic techniques and semidefinite optimization - Lecture 2
[2] Harald Rcke Theory of Metric Embeddings - Notes
[3] Gartner, B; Matousek, J. Approximation Algorithms and Semidefinite Programming