12 Semidefinite Programming

Again consider the standard form (2.2) of a linear program,

$$\min\{\, c^T x : Ax = b, x \geqslant 0 \,\}.$$

The goal in linear programming is to optimize a linear objective over the intersection of the non-negative orthant $\mathbb{R}^n_+ = \{x \in \mathbb{R}^n : x \geqslant 0\}$ with an affine space, described by the linear equation $Ax = b$. The non-negative orthant is a *convex cone*, i.e., a set $C \subseteq \mathbb{R}^n$ for which $\alpha x + \beta y \in C$ for all $\alpha, \beta \in \mathbb{R}$ with $\alpha, \beta \geqslant 0$ and all $x, y \in C$.

Semidefinite programming replaces the non-negative orthant with a different convex cone. Let $\mathbb{S}^n = \{X \in \mathbb{R}^{n\times n} : X^T = X\}$ be the set of all symmetric $n \times n$ matrices, and call $A \in \mathbb{S}^n$ *positive semidefinite*, denoted $A \succeq 0$, if $z^T A z \geqslant 0$ for all $z \in \mathbb{R}^n$. Let $\mathbb{S}^n_+ = \{A \in \mathbb{S}^n : A \succeq 0\}$. It is easy to see that $\mathbb{S}^n_+$ is a convex cone, henceforth called the convex cone of positive semidefinite matrices, or simply the positive semidefinite cone.

A linear function of $X \in \mathbb{S}^n$ can be expressed in terms of the inner product

$$\langle C, X\rangle = \operatorname{tr}(CX) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}$$

for some $C \in \mathbb{S}^n$. A *semidefinite program* (SDP) therefore has the form

$$\begin{aligned} &\text{minimize} && \langle C, X\rangle \\ &\text{subject to} && \langle A_i, X\rangle = b_i \quad \text{for all } i = 1,\dots,m \\ & && X \succeq 0, \end{aligned} \tag{12.1}$$

where $C, A_1, \dots, A_m \in \mathbb{S}^n$ and $b \in \mathbb{R}^m$.

An equivalent formulation, which is sometimes more convenient, is to

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && B_0 + x_1 B_1 + \cdots + x_k B_k \succeq 0, \end{aligned}$$

where $B_0, B_1, \dots, B_k \in \mathbb{S}^n$ and $c \in \mathbb{R}^k$. A problem of this type can be brought into the form of (12.1) by setting $X = B_0 + x_1 B_1 + \cdots + x_k B_k$. The entries of X then depend in a linear way on the variables $x_1, \dots, x_k$, which leads to linear relationships between the former when the latter are eliminated.

To see that linear programming is a special case of semidefinite programming, observe that $v \geqslant 0$ for a vector $v \in \mathbb{R}^n$ if and only if

$$\operatorname{diag}(v) = \begin{pmatrix} v_1 & 0 & \cdots & 0 \\ 0 & v_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & v_n \end{pmatrix}$$

is positive semidefinite. The linear program (2.2) can thus be written as

$$\begin{array}{ll} \text{minimize} & \langle \mathrm{diag}(c), \mathrm{diag}(x) \rangle \\ \text{subject to} & \langle \mathrm{diag}(a_i), \mathrm{diag}(x) \rangle = b_i \quad \text{for all } i = 1, \dots, m \\ & \mathrm{diag}(x) \succeq 0, \end{array}$$

where $a_i = (a_{i1}, \dots, a_{in})^T$, for $i = 1, \dots, m$, is a vector consisting of the elements of the ith row of A. This problem can be brought into the form of (12.1) by replacing the diagonal matrix $\mathrm{diag}(x)$ by a general symmetric matrix X, and adding linear constraints to ensure that the off-diagonal entries of X are zero.

SDPs can be viewed as having an infinite number of linear constraints on X, namely, $z^T X z \geqslant 0$ for all $z \in \mathbb{R}^n$. As a consequence, there are optimization problems that can be written as an SDP, but not as an LP.

There are good reasons to study semidefinite programming. It includes important classes of convex optimization problems as special cases, for example linear programming and quadratically constrained quadratic programming. In a later lecture we will see that it can be used to obtain approximate solutions to hard combinatorial and non-convex optimization problems. Moreover, SDPs can often be solved very efficiently, both in theory and in practice.

12.1 SDP Duality

The Lagrangian of (12.1) can be written as

$$L(X, \lambda, Z) = \langle C, X \rangle - \sum_{i=1}^{m} \lambda_i (\langle A_i, X \rangle - b_i) - \langle Z, X \rangle,$$

where the last term takes account of the constraint $X \succeq 0$. This works because for any $Y \in \mathbb{S}^n$, $\max_{Z \succeq 0} -\langle Z, Y \rangle$ is finite if and only if $Y \succeq 0$, so (12.1) is equivalent to the unconstrained problem $\min_{X \in \mathbb{S}^n} \max_{\lambda \in \mathbb{R}^m, Z \succeq 0} L(X, \lambda, Z)$. Then,

$$g(\lambda, Z) = \inf_{X \in \mathbb{S}^n} L(X, \lambda, Z) = \begin{cases} \lambda^T b & \text{if } C - \sum_{i=1}^m \lambda_i A_i - Z = 0, \\ -\infty & \text{otherwise.} \end{cases}$$

By eliminating Z, we obtain the following dual of (12.1), which is itself an SDP:

$$\begin{array}{ll} \text{maximize} & \lambda^T b \\ \text{subject to} & C - \sum_{i=1}^m \lambda_i A_i \succeq 0. \end{array}$$

Primal and dual SDP satisfy weak duality, because

$$\langle C, X \rangle - \lambda^T b = \langle C, X \rangle - \sum_{i=1}^{m} \lambda_i b_i = \langle C, X \rangle - \sum_{i=1}^{m} \lambda_i \langle A_i, X \rangle = \langle C - \sum_{i=1}^{m} \lambda_i A_i, X \rangle \geqslant 0,$$

where the last inequality holds because both $C - \sum_{i=1}^{m} \lambda_i A_i$ and X are positive semidefinite. If the *duality gap* $\langle C, X\rangle - \lambda^T b$ is zero, then X and λ are optimal solutions of the primal and dual, respectively. Unlike in the case of LPs, strong duality might not hold. Consider for example the SDP to

$$\begin{array}{ll} \text{minimize} & x_1 \\ \text{subject to} & \begin{pmatrix} x_1 & 1 \\ 1 & x_2 \end{pmatrix} \succeq 0. \end{array}$$

The positive semidefiniteness condition is equivalent to the constraints $x_1 \geqslant 0$, $x_2 \geqslant 0$, and $x_1 x_2 \geqslant 1$, which in turn are satisfied if and only if $x_1 > 0$ and $x_2 \geqslant 1/x_1$. The SDP thus has an optimum of 0, but the optimum is not attained. There exist SDPs whose minimum duality gap is strictly positive or even infinite. Strong duality does hold, on the other hand, if both primal and dual have a feasible solution that is *positive definite*, i.e., lies in the interior of the positive semidefinite cone.

While no algorithm is known for solving SDPs in a finite number of steps, they can be solved approximately in polynomial time, for example by a variant of the ellipsoid method. We will now briefly discuss a different class of methods that run in polynomial time in the worst case and are also very efficient in practice.

12.2 Primal-Dual Interior-Point Methods

We discuss the method for the primal and dual linear programs

$$\min\{\, c^T x : Ax = b, x \geqslant 0 \,\} \qquad \text{and} \qquad \max\{\, b^T\lambda : A^T\lambda + z = c, z \geqslant 0 \,\}.$$

The reason why these optimization problems cannot be solved using Newton's method are the inequality constraints $x \geqslant 0$ and $z \geqslant 0$. The idea behind *barrier methods* is to drop the inequality constraints and instead augment the objective by a so-called barrier function that penalizes solutions close to the boundary of, or outside, the feasible set. *Primal-dual interior-point methods* apply this idea to both the primal and the dual and try to solve them simultaneously. In the case of the above linear programs we add a *logarithmic barrier* and obtain the modified primal and dual problems

$$\min\{\, c^T x - \mu \sum_{i=1}^{n} \log x_i : Ax = b \,\} \quad \text{and} \quad \max\{\, b^T\lambda + \mu \sum_{j=1}^{m} \log z_j : A^T\lambda + z = c \,\},$$

for a parameter $\mu > 0$. The constraint $X \succeq 0$ in an SDP can be handled analogously using the barrier

$$-\mu \sum_{i=1}^{n} \log(\kappa_i(X)) = -\mu \log\left(\prod_{i=1}^{n} \kappa_i(X)\right) = -\mu \log(\det(X)),$$

where κ_i is the ith eigenvalue of X. This works because $X \succeq 0$ if and only if $\kappa_i \geqslant 0$ for all $i = 1, \ldots, n$.

By considering the Lagrangian, it can be shown that (x, λ, z) is optimal for the modified primal and dual problems if

$$\begin{aligned} Ax = b, &\quad x \geqslant 0, \\ A^T\lambda + z = c, &\quad z \geqslant 0, \\ x_i z_i = \mu &\quad \text{for all } i = 1, \dots, n. \end{aligned} \tag{12.2}$$

Note that for $\mu = 0$, the last constraint is identical to the usual complementary slackness condition, which ensures optimality for the original problem.

A solution to the modified problems can be found using Newton's method, and provides a better and better approximation to a solution of the original problems as μ tends to zero. When μ is small, however, the modified objective is hard to optimize using Newton's method because its second-order partial derivatives vary rapidly near the boundary of the feasible set. Primal-dual interior-point methods circumvent this problem by solving a sequence of problems, decreasing μ in each iteration and starting each Newton minimization at the solution obtained in the previous round. The procedure terminates when $\mu < \epsilon$ for some desired accuracy $\epsilon > 0$. Suppose that we have found a solution (x, λ, z) that satisfies (12.2) for a given value of μ. If $\mu < \epsilon$, we stop. Otherwise we update μ, for example to $(x^T z)/(2n)$, and use Newton's method to compute a solution $(x, \lambda, z)_k + (\delta x, \delta\lambda, \delta z)$ that satisfied (12.2) for the new value of μ. Then we proceed with the next round. It can be shown that for an appropriate choice of the parameters, the method decreases the duality gap from ϵ_0 to ϵ in time $O(\sqrt{n}\log(\epsilon_0/\epsilon))$.