Convex Conic Optimization, and SDP

Robert M. Freund

April, 2014

©2014 Massachusetts Institute of Technology. All rights reserved.
1 Introduction

Conic optimization is a natural generalization of linear optimization that leads to elegant and symmetric formulations of many optimization problems. Conic optimization allows for the representation of many important types of constraints as set inclusions in a way that is natural for many problems and that is in synch with modern interior-point algorithms for convex optimization.

2 Cones and Proper Cones

We say that $K \subset \mathbb{R}^n$ is a convex cone if:

$$x, y \in K \text{ and } \alpha, \beta \geq 0 \implies \alpha x + \beta y \in K.$$ 

Some examples of convex cones that are useful to us are:

(i) the nonnegative orthant: $\mathbb{R}^n_+ := \{x \in \mathbb{R}^n \mid x_j \geq 0, \ j = 1, \ldots, n\}$

(ii) the second-order cone: $Q^n = \left\{ x \in \mathbb{R}^n \mid \sqrt{\sum_{j=1}^{n-1} x_j^2} \leq x_n \right\}$. This cone is also sometimes called the Lorentz cone or the ice-cream cone (honestly)

(iii) real $n$-dimensional space: $\mathbb{R}^n$

(iv) any linear subspace: $\{x \in \mathbb{R}^n : Ax = 0\}$

(v) the origin in $\mathbb{R}^n$: $\{0\} \in \mathbb{R}^n$

(vi) the semidefinite cone: $S^{k \times k}_+ := \{X \in S^{k \times k} \mid v^T X v \geq 0 \text{ for all } v \in \mathbb{R}^k\}$ is the set of $k \times k$ symmetric positive semi-definite matrices

(vii) any polyhedral cone: $\{x \in \mathbb{R}^n : Ax \geq 0\}$, or/also in the format $\{x \in \mathbb{R}^n : x = M w \text{ for some } w \geq 0\}$

(viii) a cross-product of cones: $K = K_1 \times K_2 \times \cdots \times K_l$ where $K_j$ is a convex cone, $j = 1, \ldots, l$. 

2
We say that a cone $K$ is a *proper* cone if $K$ satisfies:

1. $K$ is closed and convex,

2. $K$ is *solid*, i.e., there exists $x$ and $r$ for which $B(x, r) \subset K$ where $B(x, r)$ is the ball centered at $x$ with radius $r$, and

3. $K$ is *pointed*, i.e., $K$ contains no line. This means that there is no non-zero $x \in K$ for which $-x \in K$. Put yet another way, $K \cap -K = \{0\}$.

### 3 Standard Primal Conic Optimization Problem

We consider the following convex optimization problem in *standard primal conic form*:

$$\begin{align*}
\text{CP} : \quad z^* &= \min_{x} \quad c^T x \\
\text{s.t.} \quad Ax &= b \\
&\quad x \in K,
\end{align*}$$

where $K$ is a closed convex cone.

When $K = \mathbb{R}^n_+$, then CP is called a *linear optimization problem* (LO or LP).

When $K$ is the cross-product of second-order cones, namely $K = Q^{n_1} \times Q^{n_2} \times \cdots \times Q^{n_q}$, then CP is called a *second-order cone problem* (SOCP).

When $K$ is the cross-product of semidefinite cones, namely $K = S^{n_1 \times n_1}_+ \times S^{n_2 \times n_2}_+ \times \cdots \times S^{n_s \times n_s}_+$, then CP is called a *semidefinite optimization problem* or a semidefinite programming problem (SDO or SDP).

**Remark 1** We can use the standard primal conic format to model any linear optimization problem directly, whether it contains equality constraints, inequality constraints, etc. We accomplish this by having $K$ be the cross-
product of appropriate half-lines and zero-vectors. To see this, consider the linear optimization problem:

\[
\text{LP : } z^* = \min_x c^T x \\
\text{s.t.} \quad \begin{align*}
\tilde{A}x &= \tilde{b} \\
\hat{A}x &\geq \hat{b} \\
\hat{A}x &\leq \hat{b},
\end{align*}
\]

where \(\tilde{A}, \hat{A}, \text{ and } \hat{A}\) have \(k, l, \text{ and } m\) rows each, respectively. Let

\[
A = \begin{bmatrix}
\tilde{A} \\
\hat{A}
\end{bmatrix}, \quad b = \begin{bmatrix}
\tilde{b} \\
\hat{b}
\end{bmatrix},
\]

and let

\[
K = \mathbb{R}^n \times \{0\}^k \times (-\mathbb{R}_+^l) \times \mathbb{R}_+^m.
\]

Then we can reformat LP to be:

\[
\text{CP : } z^* = \min_{x,s} c^T x + 0^T s \\
\text{s.t.} \quad Ax + Is = b \\
(x, s) \in K.
\]

4 Dual Cones

Let \(K^*\) denote the dual cone of the closed convex cone \(K \subset \mathbb{R}^n\), defined by:

\[
K^* := \{ y \in \mathbb{R}^n \mid y^T x \geq 0 \text{ for all } x \in K \}.
\]

**Proposition 4.1** If \(K\) is a nonempty closed convex cone, then \(K^*\) is a nonempty closed convex cone.
Proof: Notice that $0 \in K^*$, which shows that $K^* \neq \emptyset$. If $y^1, y^2 \in K^*$, then for every $x \in K$ and every $\alpha, \beta \geq 0$ we have $(\alpha y^1 + \beta y^2)^T x \geq 0$, which shows that $K^*$ is a convex cone. Suppose that $y^1, y^2, \ldots \in K^*$ and $\lim_{j \to \infty} y^j = \bar{y}$. Then for every $x \in K$ we have $(y^j)^T x \geq 0$ and so $\bar{y}^T x \geq 0$, whereby $\bar{y} \in K^*$, which shows that $K^*$ is closed. 

Proposition 4.2 If $K$ is a nonempty closed convex cone, then $(K^*)^* = K$.

Proof: We have

$$K^* := \{ y \in \mathbb{R}^n \mid y^T x \geq 0 \text{ for all } x \in K \}$$

and

$$(K^*)^* := \{ z \in \mathbb{R}^n \mid z^T y \geq 0 \text{ for all } y \in K^* \}.$$ 

For every $x \in K$ we have $x^T y \geq 0$ for all $y \in K^*$, which shows that $K \subset (K^*)^*$. Suppose that $K \neq (K^*)^*$. Then there exists $\bar{z} \in (K^*)^*$ for which $\bar{z} \notin K$. Since $\bar{z} \notin K$ and $K$ is a closed convex set, there exists a hyperplane that strictly separates $\bar{z}$ from $K$. Thus there exists $y \neq 0$ and $\alpha$ for which $y^T \bar{z} < \alpha$ and $y^T x > \alpha$ for all $x \in K$, and in particular this implies that $\alpha < 0$. It also follows that $y^T x \geq 0$ for all $x \in K$, and so $y \in K^*$. However, $y^T \bar{z} < \alpha < 0$, which implies that $\bar{z} \notin (K^*)^*$, which is a contradiction. Therefore $(K^*)^* = K$. 

Here are the dual cones associated with the above examples:

(i) $(\mathbb{R}_+^n)^* = \mathbb{R}_+^n$

(ii) $(Q^n)^* = Q^n$

(iii) $(\mathbb{R}^n)^* = \{0\}^n$

(iv) $\{ x \in \mathbb{R}^n : Ax = 0 \}^* = \{ s \in \mathbb{R}^n : s = A^T \lambda \text{ for some } \lambda \in \mathbb{R}^m \}$

(v) $(\{0\})^* = \mathbb{R}^n$

(vi) $(S_+^{k \times k})^* = S_+^{k \times k}$

(vii) $\{ x \in \mathbb{R}^n : Ax \geq 0 \}^* = \{ s \in \mathbb{R}^n : s = A^T \lambda \text{ for some } \lambda \geq 0 \}$, and

$\{ x \in \mathbb{R}^n : x = Mw \text{ for some } w \geq 0 \}^* = \{ s \in \mathbb{R}^n : M^T s \geq 0 \}$

(viii) $(K_1 \times K_2 \times \cdots \times K_l)^* = K_1^* \times K_2^* \times \cdots \times K_l^*$
5 The Conic Dual Problem

We define the dual problem of CP as follows:

\[ \text{CD : } v^* = \max_{u,s} b^T u \]

\[ \text{s.t. } A^T u + s = c \]

\[ s \in K^* . \]

(This dual problem can also be derived from first principles using an appropriate Lagrangian function.)

**Proposition 5.1 (Weak Duality)** If \( x \) is feasible for CP and \( (u,s) \) is feasible for CD, then \( c^T x \geq b^T u \). Consequently, \( z^* \geq v^* \).

**Proof:** If \( x \) is feasible for CP and \( (u,s) \) is feasible for CD, then \( c^T x - b^T u = u^T Ax + s^T x - b^T u = s^T x \geq 0 \) because \( x \in K \) and \( s \in K^* \).

**Corollary 5.1** If \( x \) is feasible for CP and \( (u,s) \) is feasible for CD, and \( c^T x = b^T u \), then \( x \) is an optimal solution of CP and \( (u,s) \) is an optimal solution of CD.

**Remark 2** The conic dual of CD can be rearranged to look exactly like the original primal problem CP. This is exactly like linear optimization where “the dual of the dual is the primal.”

6 Formulations and Applications of SOCP

6.1 Location Problems with the Euclidean Norm

6.1.1 The Minimum Norm Problem

Given a vector \( c \), we would like to find the closest point to \( c \) that also satisfies the linear inequalities \( Ax \leq b \), see Figure 1.
This problem can be formulated as:

\[
\text{MNP : minimize}_x \quad \|x - c\|
\]

\[\text{s.t.} \quad Ax \leq b\]

\[x \in \mathbb{R}^n,\]

and can be re-written as an SOCP problem as:

\[
\text{MNP : minimize}_{x,\theta} \quad \theta
\]

\[\text{s.t.} \quad Ax \leq b\]

\[(x - c, \theta) \in Q^{n+1}\]

\[x \in \mathbb{R}^n, \quad \theta \in \mathbb{R}.\]
6.1.2 The Fermat-Weber Problem

We are given $m$ points $c^1, \ldots, c^m \in \mathbb{R}^n$. We would like to determine the location of a distribution center at the point $x \in \mathbb{R}^n$ that minimizes the sum of the distances from $x$ to each of the points $c^1, \ldots, c^m \in \mathbb{R}^n$. This problem is illustrated in Figure 2. It has the following formulation:

\[
\text{FWP : } \minimize_x \sum_{i=1}^{m} \|x - c^i\| \quad \text{s.t.} \quad x \in \mathbb{R}^n,
\]

and can be re-formulated as the following SOCP problem:
FWP : minimize \( x, \theta_1, ..., \theta_m \)

\[
\sum_{i=1}^{m} \theta_i
\]

s.t.

\[
(x - c^i, \theta_i) \in Q^{n+1}
\]

\( x \in \mathbb{R}^n, \theta_i \in \mathbb{R}, i = 1, \ldots, m \).

6.1.3 The Ball Circumscription Problem

We are given \( m \) points \( c^1, \ldots, c^m \in \mathbb{R}^n \). We would like to determine the location of a distribution center at the point \( x \in \mathbb{R}^n \) that minimizes the maximum distance from \( x \) to any of the points \( c^1, \ldots, c^m \in \mathbb{R}^n \). This problem is illustrated in Figure 3. It has the following formulation:

Figure 3: Illustration of the ball circumscription problem.
BCP : minimize_{x, \delta} \quad \delta \\
\text{s.t.} \\
\|x - c_i\| \leq \delta, \quad i = 1, \ldots, m \\
x \in \mathbb{R}^n , \\

and can be re-formulated as the following SOCP problem:

BCP : minimize_{x, \delta} \quad \delta \\
\text{s.t.} \\
(x - c_i, \delta) \in Q^{n+1}, \quad i = 1, \ldots, m \\
x \in \mathbb{R}^n, \quad \delta \in \mathbb{R} .

6.2 (Convex) Quadratic Function Level Sets

Suppose we wish to model the following “level set” constraint in the decision variables $x, t$:

$$
x^T Q x + q^T x + g \leq t ,
$$

where $Q = M^T M$ and $M$ is known or is otherwise easily computed. This can be modeled using the second-order cone as:

$$
\|(M x, \frac{1}{2} (1 + q^T x + g - t))\| \leq \frac{1}{2} (1 - (q^T x + g - t)) ,
$$

which we can write down as the following SOCP inclusion:

$$
(M x, \frac{1}{2} (1 + q^T x + g - t), \frac{1}{2} (1 - (q^T x + g - t))) \in Q^{n+2} . \quad (1)
$$
This form of the constraint fits nicely into the standard conic dual form. To instead put this constraint in conic primal form, we can write:

\[
\begin{pmatrix}
M \\
\frac{1}{2}(q^T) \\
-\frac{1}{2}(q^T)
\end{pmatrix} x +
\begin{pmatrix}
0 \\
-\frac{1}{2} \\
\frac{1}{2}
\end{pmatrix} t +
\begin{pmatrix}
-I & 0 & 0 \\
0 & -1 & 0 \\
0 & 0 & -1
\end{pmatrix}
\begin{pmatrix}
y \\
y_{n+1} \\
y_{n+2}
\end{pmatrix} =
\begin{pmatrix}
0 \\
\frac{1}{2}(-1 - g) \\
\frac{1}{2}(g - 1)
\end{pmatrix}
\]

\((x, t, y, y_{n+1}, y_{n+2}) \in \mathbb{R}^n \times \mathbb{R} \times Q^{n+2}\).

### 6.3 Quadratically Constrained (Convex) Quadratic Problems

A (convex) quadratically constrained quadratic program is a problem of the form:

\[
\text{QCQP} \quad \begin{align*}
\text{minimize} & \quad x^T Q_0 x + q_0^T x + c_0 \\
\text{s.t.} & \quad x^T Q_i x + q_i^T x + c_i \leq 0, \quad i = 1, \ldots, m,
\end{align*}
\]

where \(Q_i \succeq 0, \quad i = 0, 1, \ldots, m\), are symmetric and positive semi-definite. This problem is the same as:

\[
\text{QCQP} \quad \begin{align*}
\text{minimize} & \quad \theta \\
\text{s.t.} & \quad x^T Q_0 x + q_0^T x + c_0 - \theta \leq 0 \\
& \quad x^T Q_i x + q_i^T x + c_i \leq 0, \quad i = 1, \ldots, m.
\end{align*}
\]

We can factor each \(Q_i\) into
\[ Q_i = M_i^T M_i \]

for some matrix \( M_i \). Then note the equivalence developed in (1):

\[
(M_i x, \frac{1}{2} (q_i^T x + c_i - t + 1), \frac{1}{2} (1 - (q_i^T x + c_i - t))) \in Q^{n+2} \iff x^T Q_i x + q_i^T x + c_i \leq t.
\]

We can therefore write QCQP as:

\[
\begin{align*}
\text{QCQP} & \quad \text{minimize} \quad \theta \\
& \quad x, \theta \\
& \quad \text{s.t.} \quad (M_0 x, \frac{1}{2} (q_0^T x + c_0 - \theta + 1), \frac{1}{2} (1 - (q_0^T x + c_0 - \theta))) \in Q^{n+2} \\
& \quad (M_i x, \frac{1}{2} (q_i^T x + c_i + 1), \frac{1}{2} (1 - (q_i^T x + c_i))) \in Q^{n+2} \quad \text{for} \ i = 1, \ldots, m.
\end{align*}
\]

### 6.4 Modeling a Hyperbola Constraint

Suppose we have constraints in \( x, y, z \) of the specific form:

\[ xy \geq z^2, \quad x \geq 0, \quad y \geq 0. \]

We can model this with the second-order cone as:

\[ (x - y, 2z, x + y) \in Q^3. \]

To see why this is true, write out the above as:

\[ \sqrt{(x - y)^2 + (2z)^2} \leq x + y. \]

Squaring both sides yields:

\[ x^2 - 2xy + y^2 + 4z^2 \leq x^2 + 2xy + y^2. \]
Rearranging and dividing by 4 yields \( xy \geq z^2 \). Also notice from the SOCP inclusion that \( x + y \geq |x - y| \), which implies that \( x + y \geq -x + y \) (and hence \( x \geq 0 \)) and also \( x + y \geq x - y \) (and hence \( y \geq 0 \)).

### 6.5 Products of Nonnegative Variables

Here we show a general method for modeling products of nonnegative variables using SOCP. We illustrate this with the following specific example in variables \( x_1, x_2, \ldots, x_{16} \) and \( t \). Suppose we wish to model:

\[
x_1 x_2 x_3 \cdots x_{16} \geq t^{16}, \quad x_1 \geq 0, \ x_2 \geq 0, \ \ldots, \ x_{16} \geq 0.
\]  

(2)

We will add 14 = 16 − 2 additional variables, namely \( y_1, \ldots, y_8, \ z_1, \ldots, \ z_4, \) and \( w_1, w_2 \), and write the following 15 = 16 − 1 constraints:

\[
\begin{align*}
x_1 x_2 & \geq y_1^2 \\
x_3 x_4 & \geq y_2^2 \\
x_5 x_6 & \geq y_3^2 \\
x_7 x_8 & \geq y_4^2 \\
x_9 x_{10} & \geq y_5^2 \\
x_{11} x_{12} & \geq y_6^2 \\
x_{13} x_{14} & \geq y_7^2 \\
x_{15} x_{16} & \geq y_8^2 \\
y_1 y_2 & \geq z_1^2 \\
y_1 y_2 & \geq z_2^2 \\
y_3 y_4 & \geq z_3^2 \\
y_3 y_4 & \geq z_4^2 \\
z_1 z_2 & \geq w_1^2 \\
z_1 z_2 & \geq w_2^2 \\
z_1 z_2 & \geq w_3^2 \\
z_1 z_2 & \geq w_4^2 \\
w_1 w_2 & \geq t^2
\end{align*}
\]

in nonnegative variables

\[
x_1 \geq 0, \ x_2 \geq 0, \ \ldots, \ x_{16} \geq 0, \ y_1 \geq 0, \ldots, y_8 \geq 0, \ z_1 \geq 0, \ldots, z_4 \geq 0, \ w_1 \geq 0, \ w_2 \geq 0.
\]

Notice first that each of the above 15 hyperbola constraints can be re-written as an SOCP constraint using the construction in Subsection 6.4. Notice second that these constraints imply:

\[
x_1 x_2 x_3 \cdots x_{16} \geq (y_1 y_2 y_3 \cdots y_8)^2 \geq (z_1 z_2 z_3 z_4)^4 \geq (w_1 w_2)^8 \geq t^{16}.
\]

Conversely, if \( x_1, x_2, \ldots, x_{16} \) and \( t \) satisfy (2), then it is straightforward to construct nonnegative values of \( y_1, y_2, \ldots, y_8, \ z_1, z_2, z_3, z_4, \) and \( w_1, w_2 \) for which the above system of inequalities is satisfied.
7 Slater Points and Strong Duality

We know that when $K$ is the nonnegative orthant, i.e., $K = \mathbb{R}_+^n$, then CP is in fact a linear optimization problem, and hence the values of the primal and dual problems are identical and both the primal and dual problems attain their respective optima. However, when $K$ is not the nonnegative orthant, there is no guarantee of strong duality for general convex conic optimization. Consider the following example of an SOCP instance with a duality gap:

\begin{align*}
P: & \quad z^* = \min_x -x_2 \\
& \quad \text{s.t.} \\
& \quad x_1 - x_3 = 0 \\
& \quad (x_1, x_2, x_3) \in Q^3,
\end{align*}

whose dual is:

\begin{align*}
D: & \quad v^* = \max_u 0 \cdot u \\
& \quad \text{s.t.} \\
& \quad (-u, -1, u) \in Q^3.
\end{align*}

Looking at the primal problem, we have $x_1 = x_3$, whereby the second-order cone constraint $\sqrt{x_1^2 + x_2^2} \leq x_3$ implies that $x_2 = 0$ and hence $z^* = 0$ and this is attained at $(x_1, x_2, x_3) = (\alpha, 0, \alpha)$ for any $\alpha \geq 0$.

Looking at the dual, the second-order cone constraint states that for any feasible $u$ it must hold that $\sqrt{u^2 + 1} \leq u$ which is not satisfiable, whereby $D$ is infeasible and so $v^* = -\infty$. This shows that it is possible for $z^* \neq v^*$, unless we assume some other qualifying condition holds that will guarantee strong duality. The most common such qualifying condition, both from a practical and theoretical perspective, is that of a “Slater point” of the feasible region, which we now develop.

A Slater point for CP is a point $x^0$ that satisfies:
\[ Ax^0 = b \quad \text{and} \quad x^0 \in \text{int}K. \]

**Theorem 7.1 (Strong Duality Theorem for CP)** If CP has a Slater point, then \( v^* = z^* \) and the dual attains its optimum.  

This theorem is proved in the chapter on duality theory.

A **Slater point** for CD is a point \((u^0, s^0)\) that satisfies:

\[ A^T u^0 + s^0 = c \quad \text{and} \quad s^0 \in \text{int}K^*. \]

**Theorem 7.2 (Strong Duality Theorem for CD)** If CD has a Slater point, then \( v^* = z^* \) and the primal attains its optimum.

**Corollary 7.1** If both CP and CD have a Slater point, then then \( v^* = z^* \) and both the primal and the dual attain their respective optima.
8 Semidefinite Programming (SDP)

8.1 SDP Introductory Material

Semidefinite programming (SDP) is perhaps one of the most far-reaching developments in optimization since the identification of linear optimization in 1947. SDP has applications in such diverse fields as traditional convex constrained optimization, control theory, and combinatorial optimization. Because SDP is solvable via interior point methods and exhibits both theoretical and practical computational performance that is about as good as that of linear optimization problems of comparable size, most SDP applications can usually be solved very efficiently in practice as well as in theory.

Simply stated, SDP is a conic optimization problem wherein the underlying primal cone $K$ is the set of $n \times n$ symmetric positive semi-definite matrices, namely, $K = S_{++}^{n\times n} := \{ X \in S^{n\times n} | v^T X v \geq 0 \text{ for all } v \in \mathbb{R}^k \}$. Thus we can “naively” write down an SDP instance as:

$$\begin{align*}
\text{SDP} : \quad z^* &= \text{minimum}_{x} \quad c^T x \\
\text{s.t.} \quad &Ax = b \\
&x \in S_{++}^{n\times n}.
\end{align*}$$

We can also naively write down the dual of this SDP, which we refer to as SDD for “semidefinite dual,” using the conic dual developed in Section 5:

$$\begin{align*}
\text{SDD} : \quad v^* &= \text{maximum}_{u,s} \quad b^T u \\
\text{s.t.} \quad &A^T u + s = c \\
&s \in (S_{++}^{n\times n})^*.
\end{align*}$$

These two formats are quite naive because it is more natural to think of elements of the cone $K = S_{++}^{n\times n}$ as matrices rather than as vectors or points with $n^2$ coefficients in $\mathbb{R}^{n\times n}$ (and which must lie in the subspace of $\mathbb{R}^{n\times n}$...
defined by the \( n(n-1)/2 \) equations \( x_{ij} = x_{ji} \) for \( 1 \leq i < j \leq n \). Furthermore, it is not yet clear what is the form of the dual cone \((S^n_{n \times n})^*\). We will resolve these issues in the following subsection.

Before leaving this introductory subsection, we introduce some useful notation and collect some other useful facts about symmetric matrices and the SDP cone. Let \( S^n_{n \times n} \) denote the set of symmetric \( n \times n \) matrices, and let \( S^+_n \) denote the set of positive semidefinite (psd) \( n \times n \) symmetric matrices. Similarly let \( S^{++}_n \) denote the set of positive definite (pd) \( n \times n \) symmetric matrices.

Let \( X \) and \( Y \) be any symmetric matrices. We write “\( X \succeq 0 \)” to denote that \( X \) is symmetric and positive semidefinite, and we write “\( X \succeq Y \)” to denote that \( X - Y \succeq 0 \). We write “\( X \succ 0 \)” to denote that \( X \) is symmetric and positive definite, etc.

**Proposition 8.1** The following properties of symmetric matrices and the semidefinite cone hold:

- (i) \( S^n_{n \times n} = \{ X \in S^n \mid X \succeq 0 \} \) is a closed convex cone in \( \mathbb{R}^{n^2} \) of dimension \( n \times (n+1)/2 \).
- (ii) If \( X \in S^n_{n \times n} \), then \( X = QDQ^T \) for some orthonormal matrix \( Q \) and some diagonal matrix \( D \). (Recall that \( Q \) is orthonormal means that \( Q^{-1} = Q^T \), and that \( D \) is diagonal means that the off-diagonal entries of \( D \) are all zeros.)
- (iii) If \( X = QDQ^T \) as above, then the columns of \( Q \) form a set of \( n \) orthogonal eigenvectors of \( X \), whose eigenvalues are the corresponding diagonal entries of \( D \).
- (iv) \( X \succeq 0 \) if and only if \( X = QDQ^T \) where the eigenvalues (i.e., the diagonal entries of \( D \)) are all nonnegative.
- (v) \( X \succ 0 \) if and only if \( X = QDQ^T \) where the eigenvalues (i.e., the diagonal entries of \( D \)) are all positive.
- (vi) If \( X \succeq 0 \) and if \( x_{ii} = 0 \), then \( x_{ij} = x_{ji} = 0 \) for all \( j = 1, \ldots, n \).
Consider the matrix \( M \) defined as follows:

\[
M = \begin{pmatrix}
P & v \\
v^T & d
\end{pmatrix},
\]

where \( P > 0 \), \( v \) is a vector, and \( d \) is a scalar. Then \( M > 0 \) if and only if \( d - v^T P^{-1} v > 0 \).

### 8.2 A More Natural Representation of SDP and SDD

Let us re-write the general standard primal conic problem as:

\[
\begin{align*}
\text{CP} \quad & \text{minimize}_x \quad c \cdot x \\
\text{s.t.} \quad & a_i \cdot x = b_i, \quad i = 1, \ldots, m
\end{align*}
\]

\[ x \in K. \tag{3} \]

Here, just to be exact, \( x \) is a vector of \( n \) variables, and we write “\( c \cdot x \)” for the inner-product “\( \sum_{j=1}^{n} c_j x_j \)”, etc.

In words, CP is the following problem:

“Minimize the linear function \( c \cdot x \), subject to the condition that \( x \) must solve \( m \) given equations \( a_i \cdot x = b_i \), \( i = 1, \ldots, m \), and that \( x \) must lie in the closed convex cone \( K \).”

Let us now re-format the standard primal SDP primal with our variable being a matrix \( X \in S^{n \times n} \). We can think of \( X \) as a symmetric matrix, or equivalently, as an array of \( n^2 \) components of the form \( (x_{11}, \ldots, x_{nn}) \). Of course, we can also just think of \( X \) as an object (a vector) in the space \( S^{n \times n} \). All three different equivalent ways of looking at \( X \) will be useful.

What will a linear function of \( X \) look like? If \( C(X) \) is a linear function of \( X \), then \( C(X) \) can be written as \( C \cdot X \), where
\[ C \cdot X := \sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} X_{ij} . \]

If \( X \) is a symmetric matrix, there is no loss of generality in assuming that the matrix \( C \) is also symmetric. With this notation, we can rewrite the standard SDP primal problem as:

\[
\begin{align*}
\text{SDP :} & \quad \text{minimize} \quad C \cdot X \\
& \quad \text{s.t.} \quad A_i \cdot X = b_i , \ i = 1, \ldots, m \\
& \quad \quad X \succeq 0 .
\end{align*}
\]

Notice that the variable is the matrix \( X \), but it is also helpful to think of \( X \) as an array of \( n^2 \) numbers or simply as an object (a vector) in \( S^{n \times n} \). The objective function is the linear function \( C \cdot X \) and there are \( m \) linear equations that \( X \) must satisfy, namely \( A_i \cdot X = b_i , \ i = 1, \ldots, m. \) The variable \( X \) also must lie in the (closed convex) cone of positive semidefinite symmetric matrices \( S_{+}^{n \times n} \). Note that the data for SDP consists of the symmetric matrix \( C \) (which is the data for the objective function) and the \( m \) symmetric matrices \( A_1, \ldots, A_m \), and the \( m \)-dimensional vector \( b \), which form the \( m \) linear equations.

Let us see an example of an SDP for \( n = 3 \) and \( m = 2 \). Define the following matrices:

\[
A_1 = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 3 & 7 \\ 1 & 7 & 5 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 2 & 8 \\ 2 & 6 & 0 \\ 8 & 0 & 4 \end{pmatrix}, \quad \text{and} \quad C = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 9 & 0 \\ 3 & 0 & 7 \end{pmatrix},
\]

19
and \( b_1 = 11 \) and \( b_2 = 19 \). Then the variable \( X \) will be the \( 3 \times 3 \) symmetric matrix:

\[
X = \begin{pmatrix}
  x_{11} & x_{12} & x_{13} \\
  x_{21} & x_{22} & x_{23} \\
  x_{31} & x_{32} & x_{33}
\end{pmatrix},
\]

and so, for example,

\[
C \cdot X = x_{11} + 2x_{12} + 3x_{13} + 2x_{21} + 9x_{22} + 0x_{23} + 3x_{31} + 0x_{32} + 7x_{33}
\]

\[
= x_{11} + 4x_{12} + 6x_{13} + 9x_{22} + 0x_{23} + 7x_{33}.
\]

since, in particular, \( X \) is symmetric. Therefore the SDP can be written as:

\[
\text{SDP : minimize } \begin{bmatrix}
  x_{11} + 4x_{12} + 6x_{13} + 9x_{22} + 0x_{23} + 7x_{33}
\end{bmatrix}
\]

s.t.

\[
\begin{align*}
x_{11} + 0x_{12} + 2x_{13} + 3x_{22} + 14x_{23} + 5x_{33} &= 11 \\
0x_{11} + 4x_{12} + 16x_{13} + 6x_{22} + 0x_{23} + 4x_{33} &= 19
\end{align*}
\]

\[
X = \begin{pmatrix}
  x_{11} & x_{12} & x_{13} \\
  x_{21} & x_{22} & x_{23} \\
  x_{31} & x_{32} & x_{33}
\end{pmatrix} \succeq 0.
\]

Notice that SDP looks remarkably similar to a linear optimization problem. However, the standard LP constraint that \( x \) must lie in the nonnegative orthant is replaced by the condition that the variable \( X \) must lie in the cone of positive semidefinite matrices. Just as “\( x \geq 0 \)” states that each of the \( n \) components of \( x \) must be nonnegative, it is sometimes useful to think of “\( X \succeq 0 \)” as stating that each of the \( n \) eigenvalues of \( X \) must be nonnegative.

Now let us re-format the dual problem similarly. Towards this goal, let us first examine the dual cone \((S_+^{n \times n})^*\). Notice that a generic dual cone is of the form:
\[ K^* := \{ s : s^T x \geq 0 \text{ for all } x \in K \} . \]

With the understanding that \( s^T x \) is a linear functional specified by the object \( s \), we can write down the format for the dual of the semidefinite cone as:

\[ (S_{n \times n}^+)^* := \{ S \in S_{n \times n}^+ : S \cdot X \geq 0 \text{ for all } X \in S_{n \times n}^+ \} . \]

It turns out that \( (S_{n \times n}^+)^* \) has a very simple form, namely it is self-dual: \( (S_{n \times n}^+)^* = S_{n \times n}^+ \). Before proving this result, we first develop some simple properties of the trace of a matrix \( M \), which is the sum of its diagonal entries. We begin as follows:

**Definition 8.1** For a (square) matrix \( M \in \mathbb{R}^{n \times n} \), define:

\[ \text{trace}(M) := \sum_{j=1}^{n} M_{jj} . \]

**Definition 8.2** For two matrices \( A, B \in \mathbb{R}^{k \times l} \) define

\[ A \cdot B := \sum_{i=1}^{k} \sum_{j=1}^{l} A_{ij} B_{ij} . \]

**Proposition 8.2** For two matrices \( A, B \in \mathbb{R}^{k \times l} \), it holds that:

1. \( A \cdot B = \text{trace}(A^T B) \).
2. \( \text{trace}(A^T B) = \text{trace}(BA^T) \).

In Exercise 3 you are asked to prove Proposition 8.2 plus some other properties as well. Proposition 8.2 enables a simple proof that the cone \( S_{n \times n}^+ \) is self-dual.

**Lemma 8.1** The semidefinite cone is self-dual, namely, \( (S_{n \times n}^+)^* = S_{n \times n}^+ \).
Proof: First suppose that $S \in S_n^{n \times n}$. Then $S = M^T M$ for some matrix $M$, and for any $X \succeq 0$ it holds that:

$$S \cdot X = \text{trace}(S^T X) = \text{trace}(S X) = \text{trace}(M^T M X) = \text{trace}(M X M^T) \geq 0,$$

where the first and fourth equalities follow from Proposition 8.2, and the inequality follows since $X \succeq 0$ implies that $M X M^T \succeq 0$ and hence its diagonal entries are nonnegative. Therefore $S \in (S_n^{n \times n})^*$. Next suppose that $S \notin S_n^{n \times n}$. Therefore there exists $\bar{v}$ for which $\bar{v}^T S \bar{v} < 0$. Let $X = \bar{v} \bar{v}^T$, and note that $X \in S_n^{n \times n}$. However,

$$S \cdot X = \text{trace}(S X) = \text{trace}(S \bar{v} \bar{v}^T) = \text{trace}(\bar{v}^T S \bar{v}) < 0,$$

whereby we see that $S \notin (S_n^{n \times n})^*$. □

Let us continue with our goal of writing down the dual of SDP. Borrowing the notation in (3), we can write the general standard conic dual problem as:

$$\text{CD : maximize}_{u,s} \quad \sum_{i=1}^m u_i b_i$$

s.t. \quad \sum_{i=1}^m u_i a_i + s = c$$

$$s \in K^*.$$

In words, CD is the following problem:

"Find multipliers $u_1, \ldots, u_m$ in order to maximize $b_1 u_1 + \cdots + b_m u_m$, subject to the condition that $u_1 a_1 + \cdots + u_m a_m + s = c$ where $s$ must lie in the closed convex cone $K^*$."

Noting for SDP that the matrix $C$ plays the role of the vector $c$ and the matrix $A_i$ plays the role of $a_i$, we write down the following format for the dual problem SDD:
SDD : \[ v^* = \max \sum_{i=1}^{m} u_i b_i \]

\[ \text{s.t.} \quad \sum_{i=1}^{m} u_i A_i + S = C \]

\[ S \succeq 0 . \]

One convenient way of thinking about this problem is as follows. Given multipliers \( u_1, \ldots, u_m \), the objective is to maximize the linear function \( \sum_{i=1}^{m} u_i b_i \). The constraints of SDD state that the matrix \( S \) defined as

\[ S = C - \sum_{i=1}^{m} u_i A_i \]

must be positive semidefinite. That is,

\[ C - \sum_{i=1}^{m} u_i A_i \succeq 0 . \]

We illustrate this construction with the example presented earlier. The dual problem is:

SDD : maximize \( 11u_1 + 19u_2 \)

\[ \text{s.t.} \quad u_1 \begin{pmatrix} 1 & 0 & 1 \\ 0 & 3 & 7 \\ 1 & 7 & 5 \end{pmatrix} + u_2 \begin{pmatrix} 0 & 2 & 8 \\ 2 & 6 & 0 \\ 8 & 0 & 4 \end{pmatrix} + S = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 9 & 0 \\ 3 & 0 & 7 \end{pmatrix} \]

\[ S \succeq 0 , \]

which we can rewrite in the following form:
SDD: maximize \[ 11u_1 + 19u_2 \]
\[
\text{s.t.} \begin{pmatrix} 1 - 1u_1 - 0u_2 & 2 - 0u_1 - 2u_2 & 3 - 1u_1 - 8u_2 \\ 2 - 0u_1 - 2u_2 & 9 - 3u_1 - 6u_2 & 0 - 7u_1 - 0u_2 \\ 3 - 1u_1 - 8u_2 & 0 - 7u_1 - 0u_2 & 7 - 5u_1 - 4u_2 \end{pmatrix} \succeq 0.
\]

It is often easier to “see” and work with a semidefinite program when it is presented in the format of the dual SDD, since the variables are the \( m \) multipliers \( u_1, \ldots, u_m \).

Similar to linear optimization, we can switch from one format of SDP (primal or dual) to any other format with great ease, and there is no loss of generality in assuming a particular specific format for the primal or the dual.

Using this more natural format for SDP and SDD, we can restate the duality results, namely Proposition 5.1, Corollary 5.1, Theorem 7.1, Theorem 7.2, and Corollary 7.1 as follows:

**Proposition 8.3 (Weak Duality for SDP and SDD)** If \( X \) is feasible for SDP and \( (u, S) \) is feasible for SDD, then the duality gap is \( C \cdot X - \sum_{i=1}^{m} u_ib_i = S \cdot X \geq 0 \). Consequently, \( z^* \geq v^* \).

**Corollary 8.1** If \( X \) is feasible for SDP and \( (u, S) \) is feasible for SDD, and \( C \cdot X - \sum_{i=1}^{m} u_ib_i = 0 \), then \( X \) and \( (u, S) \) are optimal solutions to SDP and SDD, respectively.

A **Slater point** for SDP is a point \( X^0 \) that satisfies:
\[
A_i \cdot X^0 = b_i, \; i = 1, \ldots, m, \text{ and } X^0 \succ 0.
\]

**Theorem 8.1 (Strong Duality Theorem for SDP)** If SDP has a Slater point, then \( v^* = z^* \) and the dual attains its optimum. ■
A Slater point for SDD is a point \((u^0, S^0)\) that satisfies:

\[
\sum_{i=1}^{m} u_i^0 A_i + S^0 = C \quad \text{and} \quad S^0 > 0 .
\]

**Theorem 8.2 (Strong Duality Theorem for SDD)** If SDD has a Slater point, then \(v^* = z^*\) and the primal attains its optimum.

**Corollary 8.2** If both SDP and SDD have a Slater point, then \(v^* = z^*\) and both the primal and the dual attain their respective optima.

In the absence of a Slater condition or other suitable qualifying condition, SDP and SDD can exhibit finite or infinite duality gaps, and the primal and/or the dual might not attain its optimum. To illustrate this last point, consider the following SDD instance:

\[
\text{SDD : } v^* = \text{maximum}_{u_1, u_2} -u_1 \\
\text{s.t.} \\
\begin{pmatrix} u_1 & 1 \\ 1 & u_2 \end{pmatrix} \succeq 0 .
\]

Notice that \((u_1, u_2)\) is feasible if and only if \(u_1 > 0, u_2 > 0,\) and \(u_1u_2 \geq 1.\) Thus \((u_1, u_2) = (\varepsilon, 1/\varepsilon)\) is feasible for all \(\varepsilon > 0,\) whereby \(v^* = 0,\) but this optimal value is not attained by any feasible solution.

### 8.3 SDP for Constraints (and Optimization) on Eigenvalues

There are many types of eigenvalue optimization problems that can be formulated as SDPs. A prototypical eigenvalue optimization problem is to create a matrix

\[
S := B - \sum_{i=1}^{k} w_i A_i
\]
given symmetric data matrices $B$ and $A_i$, $i = 1, \ldots, k$, using weights $w_1, \ldots, w_k$, in such a way to minimize the difference between the largest and smallest eigenvalue of $S$. This problem can be written as:

$$\text{EOP : minimize } \lambda_{\max}(S) - \lambda_{\min}(S)$$

$$w, S$$

s.t. $$S = B - \sum_{i=1}^{k} w_i A_i,$$

where $\lambda_{\min}(S)$ and $\lambda_{\max}(S)$ denote the smallest and the largest eigenvalue of $S$, respectively. We now show how to convert this problem into an SDP.

Recall that $S$ can be factored into $S = QDQ^T$ where $Q$ is an orthonormal matrix (i.e., $Q^{-1} = Q^T$) and $D$ is a diagonal matrix consisting of the eigenvalues of $S$. The conditions:

$$\lambda I \preceq S \preceq \mu I$$

can be rewritten as:

$$Q(\lambda I)Q^T \preceq QDQ^T \preceq Q(\mu I)Q^T.$$ 

After premultiplying the above by $Q^T$ and postmultiplying by $Q$, these conditions become:

$$\lambda I \preceq D \preceq \mu I$$

which are equivalent to:

$$\lambda \leq \lambda_{\min}(S) \text{ and } \lambda_{\max}(S) \leq \mu.$$ 

Therefore EOP can be written as:

$$\text{EOP : minimize } \mu - \lambda$$

$$w, S, \mu, \lambda$$

s.t. $$S = B - \sum_{i=1}^{k} w_i A_i$$

$$\lambda I \preceq S \preceq \mu I.$$ 

This last problem is a semidefinite program.
Note that the above methodology shows the following modeling equivalences:

(i) $\lambda_{\min}(S) \geq t \iff S \succeq tI$, and 
(ii) $\lambda_{\max}(S) \leq t \iff S \preceq tI$.

It turns out that very many other types of eigenvalue constraints can be modeled using SDP. Let us order the eigenvalues of $S$ in increasing index values as: $\lambda_1(S) \leq \lambda_2(S) \leq \cdots \leq \lambda_n(S)$. Then the following types of constraints can be modeled with SDP:

(i) $\lambda_1(S) + \lambda_2(S) + \cdots + \lambda_k(S) \geq t$, i.e., the sum of the $k$ smallest eigenvalues of $S$ must be at least $t$.

(ii) $\lambda_n(S) + \lambda_{n-1}(S) + \cdots + \lambda_{n-k+1}(S) \leq t$, i.e., the sum of the $k$ largest eigenvalues of $S$ must be at most $t$.

(iii) $\sqrt{\sum_{j=1}^{n}(\lambda_j(S))^2} \leq t$, i.e., the Euclidean norm of the vector of eigenvalues of $S$ must be at most $t$.

(iv) Let $M \in \mathbb{R}^{k \times l}$, and let $\lambda_i$, $i = 1, \ldots, r$, denote the (nonnegative) singular values of $M$. Then using SDP we can model $\lambda_1(S) + \lambda_2(S) + \cdots + \lambda_r(S) \leq t$, i.e., the sum of the singular values of $M$ must be at most $t$.

The formulation for (iii) above is very straightforward. However, the formulations for (i), (ii), and (iii) above are not so straightforward and in fact require a bit of cleverness to construct.

8.4 SDP in Combinatorial Optimization

SDP has wide applicability in combinatorial optimization, particular in approximation algorithms for NP-hard problems, where one solves an SDP problem that is a convex relaxation of a given combinatorial optimization and the solution of the SDP is then converted to a feasible solution of the
given combinatorial optimization problem. In many instances, the SDP relaxation is very tight in practice, and in certain instances in particular, the optimal solution to the SDP relaxation can be converted to a feasible solution for the original problem with provably good objective value. An example of the use of SDP in combinatorial optimization is given below for the now-famous MAXCUT problem.

Let $G$ be an undirected graph with nodes $N = \{1, \ldots, n\}$, and edge set $E$. Let $w_{ij} = w_{ji}$ be the weight on edge $(i, j)$, for $(i, j) \in E$. We assume that $w_{ij} \geq 0$ for all $(i, j) \in E$. The MAXCUT problem is to determine a subset $S$ of the nodes $N$ for which the sum of the weights of the edges that cross from $S$ to its complement $\bar{S}$ is maximized (where $\bar{S} := N \setminus S$).

Let us formulate MAXCUT as an integer program as follows. Let $x_j = 1$ for $j \in S$ and $x_j = -1$ for $j \in \bar{S}$. Then our formulation is:

$$\text{MAXCUT} : \maximize_{x} \frac{1}{4} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (1 - x_i x_j)$$

s.t. $x_j \in \{-1, 1\}, \ j = 1, \ldots, n$.

Now let

$$Y = xx^T,$$

whereby

$$Y_{ij} = x_i x_j \quad i = 1, \ldots, n, \ j = 1, \ldots, n.$$ 

Also let $W$ be the matrix whose $(i, j)^{th}$ element is $w_{ij}$ for $i = 1, \ldots, n$ and $j = 1, \ldots, n$. Then MAXCUT can be equivalently formulated as:

$$\text{MAXCUT} : \maximize_{y, x} \frac{1}{4} \left( \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \cdot Y \right)$$

s.t. $x_j \in \{-1, 1\}, \ j = 1, \ldots, n$

$$Y = xx^T.$$ 

Notice in this problem that the first set of constraints are equivalent to
\[ Y_{jj} = 1, j = 1, \ldots, n. \] We therefore obtain:

\[
\text{MAXCUT} : \quad \text{maximize}_{Y,x} \quad \frac{1}{4} \left( \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \cdot Y \right)
\]

s.t. \[ Y_{jj} = 1, \quad j = 1, \ldots, n \]
\[ Y = xx^T. \]

Last of all, notice that the matrix \( Y = xx^T \) is a symmetric rank-1 positive semidefinite matrix. If we relax this condition by removing the rank-1 restriction, we obtain the following relaxation of MAXCUT, which is a semidefinite program:

\[
\text{RELAX} : \quad \text{maximize}_{Y} \quad \frac{1}{4} \left( \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \cdot Y \right)
\]

s.t. \[ Y_{jj} = 1, \quad j = 1, \ldots, n \]
\[ Y \succeq 0. \]

It is therefore easy to see that RELAX provides an upper bound on MAXCUT, i.e.,
\[ \text{MAXCUT} \leq \text{RELAX}. \]

As it turns out, one can also prove without too much effort that:
\[ 0.87856 \leq \text{RELAX} \leq \text{MAXCUT} \leq \text{RELAX}. \]

This is an impressive result, in that it states that the value of the semidefinite relaxation is guaranteed to be no more than 14% higher than the value of NP-hard problem MAXCUT. What is so remarkable about this result is that it does not depend either on the data (the nonnegative weights \( w_{ij}, i,j = 1,\ldots, n \)), nor on the number of vertices (\( n \)) or edges (\( m \)).
8.5 SDP in Control Theory

A variety of control and system problems can be cast and solved as instances of SDP. However, this topic is beyond the scope of this monograph.

8.6 More SDP Formulations and Applications in Convex Optimization

As stated earlier, SDP has very wide applications in convex optimization. The types of constraints that can be modeled in the SDP framework include: linear inequalities, convex quadratic inequalities, lower bounds on matrix norms, lower bounds on determinants of symmetric positive semidefinite matrices, lower bounds on the geometric mean of a nonnegative vector, plus many others. Using these and other constructions, the following problems (among many others) can be cast in the form of a semidefinite program: linear programming, optimizing a convex quadratic form subject to convex quadratic inequality constraints, minimizing the volume of an ellipsoid that covers a given set of points and ellipsoids, maximizing the volume of an ellipsoid that is contained in a given polytope, plus a variety of maximum eigenvalue and minimum eigenvalue problems. In the subsections below we show some of these SDP formulations.

8.6.1 SDP for an Alternative Representation of Convex Quadratically Constrained Quadratic Programming

Recall the (convex) quadratically constrained quadratic optimization problem:

\[
\begin{align*}
\text{QCQP} : \quad & \text{minimize} \quad x^T Q_0 x + q_0^T x + c_0 \\
& \text{s.t.} \quad x^T Q_i x + q_i^T x + c_i \leq 0, \quad i = 1, \ldots, m,
\end{align*}
\]

where \( Q_i \succeq 0, \quad i = 0, 1, \ldots, m \), are symmetric and positive semi-definite. This problem is the same as:
QCQP: minimize $\theta$

\[ x, \theta \]

s.t. \[ x^T Q_0 x + q_0^T x + c_0 - \theta \leq 0 \]
\[ x^T Q_i x + q_i^T x + c_i \leq 0 \quad i = 1, \ldots, m \]

In Section 6.3 we showed that this problem could be conveniently modeled using SOCP. Here we present an alternative representation using SDP. We begin with the following elementary equivalence:

\[ M := \begin{pmatrix} W & x \\ x^T & 1 \end{pmatrix} \succeq 0 \iff W \succeq xx^T, \quad (4) \]

which follows as a special case of a more general result in Exercise 2. Using (4), it is straightforward to show that QCQP is equivalent to the following SDP:

QCQP: minimize $\theta$

\[ x, W, \theta \]

s.t.

\[ \begin{pmatrix} Q_0 & \frac{1}{2} q_0 \\
\frac{1}{2} q_0^T & c_0 - \theta \end{pmatrix} \cdot \begin{pmatrix} W & x \\ x^T & 1 \end{pmatrix} \leq 0 \]

\[ \begin{pmatrix} Q_i & \frac{1}{2} q_i \\
\frac{1}{2} q_i^T & c_i \end{pmatrix} \cdot \begin{pmatrix} W & x \\ x^T & 1 \end{pmatrix} \leq 0 \quad i = 1, \ldots, m \]

\[ \begin{pmatrix} W & x \\ x^T & 1 \end{pmatrix} \succeq 0. \]

Notice in this formulation that there are now \( \frac{(n+1)(n+2)}{2} \) variables and one semidefinite cone inclusion. However, the constraints are all linear inequalities as opposed to second-order cone inclusions, and there is no longer the need to factorize the matrices $Q_i$, $i = 1, \ldots, m$, before constructing the optimization formulation.
8.6.2 A Hierarchy among the Nonnegative Orthant, the Second-Order Cone, and the Semidefinite Cone

Both from a practical as well as a theoretical viewpoint, it is relevant to ask if, for example, linear inequality constraints can be modeled with second-order cone constraints or semidefinite cone constraints? Or if second-order cone constraints can be modeled with semidefinite cone constraints? It turns out that there is a hierarchy among these conic classes as follows:

(i) a linear inequality constraint can be modeled with second-order cone constraints, and

(ii) a second-order cone constraint can be modeled with a semidefinite cone constraint.

In this way we observe that the semidefinite cone can be used to model both linear inequalities and second-order cone constraints. Let us see how these modeling constructions can be done.

Modeling Linear Inequalities with Second-Order Cone Constraints

We need to show that the nonnegative orthant $\mathbb{R}_+^n$ can be modeled as a “slice” of cross-products of second-order cones. (Here we use the term “slice” to denote the intersection of a linear subspace with the cone.) To see this, notice that we can model the conditions

$$x_j \geq 0, \ j = 1, \ldots, n$$

as

$$ (y_j, x_j) \in Q^2, \ y_j = 0, \ j = 1, \ldots, n.$$

It therefore follows that

$$\mathbb{R}_+^n = \{(x_1, \ldots, x_n) : ((y_1, x_1), \ldots, (y_n, x_n)) \in Q^2 \times \cdots \times Q^2 \text{ and } y_1 = 0, \ldots, y_n = 0\}.$$

Notice in the above identity that the right-hand side is the intersection of the cone $Q^2 \times \cdots \times Q^2$ and the subspace $\{(y_1, x_1), \ldots, (y_n, x_n)) \in \mathbb{R}^{2n} : y_1 = 0, \ldots, y_n = 0\}.$
Modeling Second-Order Cone Constraints with the Semidefinite Cone

Here we show that the second-order cone is a slice of the semidefinite cone. Let \( \|v\| = \sqrt{v^T v} \) denote the Euclidean norm. First note the following equivalence:

\[
\|x\| \leq t \iff X := \begin{pmatrix} tI & x \\ x^T & t \end{pmatrix} \succeq 0.
\] (5)

To see why this is true, observe that each of the left and right statements imply that \( t \geq 0 \). If \( t = 0 \), then each statement implies that \( x = 0 \) and so the other statement is true. If \( t > 0 \), then the result follows as a consequence of Exercise 1.

Let us identify \((x, t)\) with the last column of the matrix \( X \) in (5). Then we can write \( Q^{n+1} = \{(x, t) \in \mathbb{R}^n \times \mathbb{R} : \|x\| \leq t\} \) as

\[
Q^{n+1} = \left\{ (X_{1,n+1}, \ldots, X_{n,n+1}, X_{n+1,n+1}) \left| \begin{array}{l} X \succeq 0 \\
X_{i,i} = X_{n+1,n+1}, \ i = 1, \ldots, n \\
X_{i,j} = 0, \ 1 \leq i < j \leq n \end{array} \right. \right\}.
\]

Of course, it is also easy to directly model any linear optimization problem as a special instance of an SDP. To see one way of doing this, suppose that \((c, a_1, \ldots, a_m, b_1, \ldots, b_m)\) comprise the data for the LP. Then define:

\[
A_i = \begin{pmatrix} a_{i1} & 0 & \ldots & 0 \\
0 & a_{i2} & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & a_{in} \end{pmatrix}, \ i = 1, \ldots, m, \quad \text{and} \quad C = \begin{pmatrix} c_1 & 0 & \ldots & 0 \\
0 & c_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & c_n \end{pmatrix}.
\]

Then the LP can be written as:
SDP: \[ z^* = \text{minimum} \ C \cdot X \]

subject to \[ A_j \cdot X = b_i, \ i = 1, \ldots, m \]
\[ X_{ij} = 0, \ i = 1, \ldots, n, \ j = i + 1, \ldots, n \]
\[ X \succeq 0, \]

where the second group of equations enforce that \( X \) is a diagonal matrix:

\[
X = \begin{pmatrix}
x_1 & 0 & \ldots & 0 \\
0 & x_2 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & x_n
\end{pmatrix}.
\]

Of course, in practice one would never want to convert an instance of LP into an instance of SDP (or an instance of second-order cone optimization). The above constructions merely shows that SDP and second-order cone optimization includes linear optimization as a special case.

There are some very important differences between semidefinite optimization and linear optimization (LP), which include the following:

- In SDP, there may be a finite or infinite duality gap. The primal and/or dual may or may not attain their optima. But, as noted above in Corollary 8.2, both programs will attain their common optimal value if both programs have feasible solutions in the interior of their respective semidefinite cones.

- There is no finite algorithm for solving SDP. There is a simplex algorithm, but it is not a finite algorithm, and there is no direct analog of a “basic feasible solution” for SDP.

- Given rational data, the feasible region of SDP may have no rational
solutions. The optimal solution may not have rational components or rational eigenvalues.

- Given rational data whose binary encoding is size $L$, the norms of any SDP feasible and/or optimal solutions or eigenvalues may exceed $2^{2^L}$ (or worse).
- Given rational data whose binary encoding is size $L$, the nonzero values of any SDP feasible and/or optimal solutions or eigenvalues may be less than $2^{-2^L}$ (or worse).

### 8.6.3 SDP for the Smallest Circumscribed Ellipsoid Problem

A given matrix $R > 0$ and a given point $z$ can be used to define an ellipsoid in $\mathbb{R}^n$:

$$E_{R,z} := \{ y \mid (y - z)^T R (y - z) \leq 1 \} .$$

One can prove that the volume of $E_{R,z}$ is proportional to $\det(R^{-\frac{1}{2}})$.

Suppose we are given a convex set $C \subset \mathbb{R}^n$ described as the convex hull of $k$ points $c_1, \ldots, c_k$. We would like to find an ellipsoid circumscribing these $k$ points that has minimum volume. Our problem can be written in the following form:

$$\text{MCP} : \text{minimize} \quad \text{vol}(E_{R,z})$$
$$\text{subject to} \quad c_i \in E_{R,z} , \quad i = 1, \ldots, k ,$$

which is equivalent to:

$$\text{MCP} : \text{minimize} \quad -\frac{1}{2} \ln(\det(R))$$
$$\text{subject to} \quad (c_i - z)^T R (c_i - z) \leq 1 , \quad i = 1, \ldots, k$$
$$R > 0 .$$

Now factor $R = M^2$ where $M > 0$ (that is, $M$ is a square root of $R$), and
now MCP becomes:

\[
\text{MCP : minimize } -\frac{1}{2} \ln(\det(M^2)) \\
\text{ s.t. } (c_i - z)^T M^T M(c_i - z) \leq 1, \quad i = 1, \ldots, k \\
M \succ 0.
\]

We can write MCP as:

\[
\text{MCP : minimize } -\ln(\det(M)) \\
\text{ s.t. } \|M(c_i - z)\| \leq 1, \quad i = 1, \ldots, k \\
M \succ 0.
\]

Last of all, we make the substitution \(y = Mz\) to obtain:

\[
\text{MCP : minimize } -\ln(\det(M)) \\
\text{ s.t. } \|Mc_i - y\| \leq 1, \quad i = 1, \ldots, k \\
M \succ 0.
\]

Notice that this last program involves second-order cone constraints that are linear in \(M\) and \(y\), and a semidefinite inclusion, where all of the relevant coefficients are linear functions of the variables \(M\) and \(y\). However, the objective function is not a linear function. It is possible to convert this problem further into a genuine instance of SDP, because there is a way to model constraints of the form

\[-\ln(\det(X)) \leq \theta\]

by using SDP and SOCP inclusions, see Section 8.6.5. However, this is not necessary, either from a theoretical or a practical viewpoint, because it turns out that the function \(f(X) = -\ln(\det(X))\) is extremely well-behaved and is very easy to optimize (both in theory and in practice).

Finally, note that after solving the formulation of MCP above, we can recover the matrix \(R\) and the center \(z\) of the optimal ellipsoid by computing

\[R = M^2 \quad \text{and} \quad z = M^{-1}y.\]
8.6.4 SDP for the Largest Inscribed Ellipsoid Problem

Recall that a given matrix $R \succ 0$ and a given point $z$ can be used to define an ellipsoid in $\mathbb{R}^n$:

$$E_{R,z} := \{ x \mid (x-z)^TR(x-z) \leq 1 \},$$

and that the volume of $E_{R,z}$ is proportional to $\det(R^{-\frac{1}{2}})$.

Suppose we are given a convex set $\mathcal{C} \subset \mathbb{R}^n$ described as the intersection of $k$ halfspaces $\{x \in \mathbb{R}^n \mid (a_i)^T x \leq b_i\}$, for $i = 1, \ldots, k$, that is,

$$\mathcal{C} = \{ x \in \mathbb{R}^n \mid Ax \leq b \},$$

where the $i^{th}$ row of the matrix $A$ consists of the entries of the vector $a_i$ for $i = 1, \ldots, k$. We would like to find an ellipsoid inscribed in $\mathcal{C}$ of maximum volume. Our problem can be written in the following form:

$$\text{MIP : maximize } \text{vol } (E_{R,z})$$

$$\text{s.t. } E_{R,z} \subset \mathcal{C},$$

which is equivalent to:

$$\text{MIP : maximize } \det(R^{-\frac{1}{2}})$$

$$\text{s.t. } E_{R,z} \subset \{ x \mid (a_i)^T x \leq b_i \}, \quad i = 1, \ldots, k$$

$$R \succ 0,$$
which is equivalent to:

\[
\begin{align*}
\text{MIP} : & \quad \text{maximize} \quad \frac{1}{2} \ln(\det(R^{-1})) \\
& \quad R, z \\
& \quad \text{s.t.} \quad \max_x \{ a_i^T x \mid (x - z)^T R (x - z) \leq 1 \} \leq b_i, \quad i = 1, \ldots, k \\
& \quad \quad R > 0.
\end{align*}
\]

For a given \( i = 1, \ldots, k \), the solution to the optimization problem in the \( i^{th} \) constraint is:

\[
x^* = z + \frac{R^{-1} a_i}{\sqrt{a_i^T R^{-1} a_i}}
\]

with optimal objective function value

\[
a_i^T z + \sqrt{a_i^T R^{-1} a_i},
\]

and so MIP can be rewritten as:

\[
\begin{align*}
\text{MIP} : & \quad \text{maximize} \quad \frac{1}{2} \ln(\det(R^{-1})) \\
& \quad R, z \\
& \quad \text{s.t.} \quad a_i^T z + \sqrt{a_i^T R^{-1} a_i} \leq b_i, \quad i = 1, \ldots, k \\
& \quad \quad R > 0.
\end{align*}
\]

Now factor \( R^{-1} = M^2 \) where \( M > 0 \) (that is, \( M \) is a square root of \( R^{-1} \)), whereby MIP becomes:

\[
\begin{align*}
\text{MIP} : & \quad \text{maximize} \quad \frac{1}{2} \ln(\det(M^2)) \\
& \quad M, z \\
& \quad \text{s.t.} \quad a_i^T z + \sqrt{a_i^T M^T M a_i} \leq b_i, \quad i = 1, \ldots, k \\
& \quad \quad M > 0.
\end{align*}
\]

which we can re-write as:

\[
\begin{align*}
\text{MIP} : & \quad \text{maximize} \quad \ln(\det(M)) \\
& \quad M, z \\
& \quad \text{s.t.} \quad \| M a_i \| \leq (b_i - a_i^T z), \quad i = 1, \ldots, k \\
& \quad \quad M > 0.
\end{align*}
\]
Notice that this last program involves second-order cone constraints and a semidefinite inclusion where all of the appropriate coefficients are linear functions of the variables $M$ and $z$. However, the objective function is not a linear function. It is possible to convert this problem further into a genuine instance of SDP, because there is a way to model constraints of the form

$$-\ln(\det(X)) \leq \theta$$

by using SDP and SOCP inclusions, see Section 8.6.5. However, this is not necessary, either from a theoretical or a practical viewpoint, because it turns out that the function $f(X) = -\ln(\det(X))$ is extremely well-behaved and is very easy to optimize (both in theory and in practice).

Finally, note that after solving the formulation of MIP above, we can recover the matrix $R$ of the optimal ellipsoid by computing

$$R = M^{-2}.$$ 

### 8.6.5 SDP for Modeling Lower Bounds on Determinants of PSD Matrices

Let $X \in S^{n \times n}$ and $t > 0$ be given, and suppose we want to model the following constraint:

$$X \succeq 0 \text{ and } \ln \det(X) \geq 2n \ln(t). \quad (6)$$

Here we show how to model (6) using SDP and SOCP inclusions. We illustrate this modeling construction with the case $n = 8$, i.e., $X \in S^{8 \times 8}$. Consider the following constraints in $X$, $t$, an additional matrix variable $L$, and six additional nonnegative scalar variables $a, b, c, d, e, f$: 

39
\[
\begin{pmatrix}
I & L^T \\
L & X
\end{pmatrix} \succeq 0
\]

\[L_{ij} = 0, \quad 1 \leq i < j \leq 8\]

\[
\begin{align*}
L_{11}L_{22} & \geq a^2 \\
L_{33}L_{44} & \geq b^2 \\
L_{55}L_{66} & \geq c^2 \\
L_{77}L_{88} & \geq d^2 \\
ab & \geq e^2 \\
cd & \geq f^2 \\
ef & \geq t^2
\end{align*}
\]

(7)

It follows from Exercise 14 and the first semidefinite inclusion above that \(X \succeq LL^T\), and it follows from Exercise 8 that \(\det(X) \geq \det(LL^T) = \det(L)^2\). Furthermore, the equation group above enforces that \(L\) is lower triangular, which implies that \(\det(L) = L_{11}L_{22}\cdots L_{88}\). We therefore have:

\[
\begin{align*}
\det(X) & \geq \det(LL^T) \\
& = \det(L)^2 \\
& = (L_{11}L_{22}\cdots L_{88})^2 \\
& \geq (abcd)^4 \\
& \geq (ef)^8 \\
& \geq (t)^{16}
\end{align*}
\]

Taking logarithms of both sides proves that the above constraints imply (6). In order to prove the converse, we use the fact that any positive semidefinite symmetric matrix \(X\) can be factorized into \(X = LL^T\) where \(L\) is lower triangular. The result then follows by using \(L\) and appropriate values of \(a, b, c, d, e, f\). Finally, note that the conditions (7) are comprised of an SDP inclusion, \(n(n-1)/2\) linear equations, and \(n - 1 = 7\) hyperbola constraints.
that can be re-written as SOCP inclusions using the methodology of Section 6.4.

8.6.6 SDP for Univariate Polynomial Optimization

This section is under construction.

8.6.7 Standard Problem Formats for Conic Optimization Software

This section is under construction.

9 Interior-point Methods for SDP

At the heart of an interior-point method is a barrier function that exerts a repelling force from the boundary of the feasible region. For SDP, we need a barrier function whose value approaches +\infty as points \( X \) approach the boundary of the semidefinite cone \( S_{n \times n}^+ \).

Let \( X \in S_{n \times n}^+ \). Then \( X \) will have \( n \) eigenvalues, say \( \lambda_1(X), \ldots, \lambda_n(X) \) (possibly counting multiplicities). We can characterize the interior of the semidefinite cone as follows:

\[
\text{int} S_{n \times n}^+ = \{ X \in S_{n \times n}^+ \mid \lambda_1(X) > 0, \ldots, \lambda_n(X) > 0 \}.
\]

A natural barrier function to use to repel \( X \) from the boundary of \( S_{n \times n}^+ \) then is
Consider the logarithmic barrier problem $BSDP(\theta)$ parameterized by the positive barrier parameter $\theta$:

$$BSDP(\theta) : \text{minimize} \quad C \cdot X - \theta \ln(\det(X))$$

s.t. \quad A_i \cdot X = b_i, \ i = 1, \ldots, m,

\[ X \succ 0. \]

Let $f_\theta(X)$ denote the objective function of $BSDP(\theta)$. Then it is not too difficult to derive:

$$\nabla f_\theta(X) = C - \theta X^{-1}, \quad (8)$$

and so the Karush-Kuhn-Tucker conditions for $BSDP(\theta)$ are:

$$\left\{ \begin{array}{l}
A_i \cdot X = b_i, \ i = 1, \ldots, m \\
X \succ 0 \\
C - \theta X^{-1} = \sum_{i=1}^{m} u_i A_i.
\end{array} \right. \quad (9)$$

Because $X$ is symmetric, we can factorize $X$ into $X = LL^T$. We then can define
\[ S = \theta X^{-1} = \theta L^{-T} L^{-1} , \]

which implies

\[ \frac{1}{\theta} L^T S L = I , \]

and we can rewrite the Karush-Kuhn-Tucker conditions as:

\[
\begin{align*}
A_i \cdot X &= b_i , \quad i = 1, \ldots, m \\
X &> 0 , \quad X = LL^T \\
\sum_{i=1}^{m} u_i A_i + S &= C \\
I - \frac{1}{\theta} L^T S L &= 0 .
\end{align*}
\]

(10)

From the equations of (10) it follows that if \((X, u, S)\) is a solution of (10), then \(X\) is feasible for SDP, \((u, S)\) is feasible for SDD, and the resulting duality gap is

\[ S \cdot X = \text{trace}(S^T X) = \text{trace}(SX) = \text{trace}(\theta I) = n\theta . \]

This suggests that we try solving \(BSDP(\theta)\) for a variety of values of \(\theta\) as \(\theta \to 0\).
However, we cannot usually solve (10) exactly, because the fourth equation group is not linear in the variables. We will instead define a “β-approximate solution” of the Karush-Kuhn-Tucker conditions (10). Before doing so, we introduce the following norm on matrices, called the Frobenius norm:

$$\|M\| := \sqrt{M \cdot M} = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} M_{ij}^2}.$$ 

For some important properties of the Frobenius norm, see the last subsection of this section. A “β-approximate solution” of $BSDP(\theta)$ is defined as any solution $(X,u,S)$ of

$$\begin{align*}
A_i \cdot X &= b_i, \quad i = 1, \ldots, m \\
X &> 0, \quad X = LL^T \\
\sum_{i=1}^{m} u_i A_i + S &= C \\
\|I - \frac{1}{\theta} L^T SL\| &\leq \beta.
\end{align*}$$

(11)

**Lemma 9.1** If $(\bar{X}, \bar{u}, \bar{S})$ is a β-approximate solution of $BSDP(\theta)$ and $\beta < 1$, then $\bar{X}$ is feasible for $SDP$, $(\bar{u}, \bar{S})$ is feasible for $SDD$, and the duality gap satisfies:

$$n\theta(1 - \beta) \leq C \cdot X - \sum_{i=1}^{m} u_i b_i = \bar{X} \cdot \bar{S} \leq n\theta(1 + \beta).$$

(12)

**Proof:** Primal feasibility is obvious. To prove dual feasibility, we need to show that $\bar{S} \succeq 0$. To see this, define
\[ R = I - \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L} \]  

and note that \( \|R\| \leq \beta < 1 \). Rearranging (13), we obtain
\[ \bar{S} = \theta \bar{L}^{-T} (I - R) \bar{L}^{-1} > 0 \]

because \( \|R\| < 1 \) implies that \( I - R > 0 \), see Proposition 9.4, part (4.). We also have \( \bar{X} \cdot \bar{S} = \text{trace}(\bar{X} \bar{S}) = \text{trace}(\bar{L} \bar{L}^T \bar{S}) = \theta \text{trace}(I - R) = \theta (n - \text{trace}(R)) \). However, \( |\text{trace}(R)| \leq \sqrt{n} \|R\| \leq n \beta \), whereby we obtain
\[ n \theta (1 - \beta) \leq \bar{X} \cdot \bar{S} \leq n \theta (1 + \beta) . \]

**9.1 The Algorithm**

Based on the analysis just presented, we are motivated to develop the following algorithm:

**Step 0. Initialization.** Data is \((X^0, u^0, S^0, \theta^0)\). \( k = 0 \). Assume that \((X^0, u^0, S^0)\) is a \( \beta \)-approximate solution of \( BSDP(\theta^0) \) for some known value of \( \beta \) that satisfies \( \beta < 1 \).

**Step 1. Set Current values.** \((\bar{X}, \bar{u}, \bar{S}) = (X^k, u^k, S^k), \theta = \theta^k \).  

**Step 2. Shrink \( \theta \).** Set \( \theta' = \alpha \theta \) for some \( \alpha \in (0,1) \). In fact, it will be appropriate to set
\[ \alpha = 1 - \frac{\sqrt{\beta} - \beta}{\sqrt{\beta} + \sqrt{n}} \]

**Step 3. Compute Newton Direction and Multipliers.** Compute
the Newton step $D'$ for $BSDP(\theta')$ at $X = \bar{X}$ by factoring $\bar{X} = \bar{L}\bar{L}^T$ and solving the following system of equations in the variables $(D, u)$:

$$
\begin{align*}
C - \theta' \bar{X}^{-1} + \theta' \bar{X}^{-1}D\bar{X}^{-1} &= \sum_{i=1}^{m} u_i A_i \\
A_i \cdot D &= 0, \quad i = 1, \ldots, m.
\end{align*}
$$

(14)

Denote the solution to this system by $(D', u')$.

**Step 4. Update All Values.**

$$
\begin{align*}
X' &= \bar{X} + D' \\
S' &= C - \sum_{i=1}^{m} u'_i A_i
\end{align*}
$$

**Step 5. Reset Counter and Continue.** $(X^{k+1}, u^{k+1}, S^{k+1}) = (X', u', S')$. $\theta^{k+1} = \theta'$. $k \leftarrow k + 1$. Go to Step 1.

Some of the unresolved issues regarding this algorithm include:

- how to set the fractional decrease parameter $\alpha$
- the derivation of the Newton step $D'$ and the multipliers $u'$
- whether or not successive iterative values $(X^k, u^k, S^k)$ are $\beta$-approximate solutions to $BSDP(\theta^k)$, and
- how to get the method started in the first place.

### 9.2 The Newton Step

Suppose that $\bar{X}$ is a feasible solution to $BSDP(\theta)$:
\[ \text{BSDP}(\theta) : \text{minimize} \quad C \cdot X - \theta \ln(\det(X)) \]

\[ \text{s.t.} \quad A_i \cdot X = b_i, \quad i = 1, \ldots, m, \quad X \succ 0. \]

Let us denote the objective function of \( \text{BSDP}(\theta) \) by \( f_\theta(X) \), i.e.,

\[ f_\theta(X) = C \cdot X - \theta \ln(\det(X)). \]

Then we can derive:

\[ \nabla f_\theta(\bar{X}) = C - \theta \bar{X}^{-1} \]

and the quadratic approximation of \( \text{BSDP}(\theta) \) at \( X = \bar{X} \) can be derived as:

\[ \text{minimize} \quad f_\theta(\bar{X}) + (C - \theta \bar{X}^{-1}) \cdot (X - \bar{X}) + \frac{1}{2} \theta \bar{X}^{-1} (X - \bar{X}) \cdot \bar{X}^{-1} (X - \bar{X}) \]

\[ \text{s.t.} \quad A_i \cdot X = b_i, \quad i = 1, \ldots, m. \]

Letting \( D = X - \bar{X} \), this is equivalent to:

\[ \text{minimize} \quad (C - \theta \bar{X}^{-1}) \cdot D + \frac{1}{2} \theta \bar{X}^{-1} D \cdot \bar{X}^{-1} D \]

\[ \text{s.t.} \quad A_i \cdot D = 0, \quad i = 1, \ldots, m. \]
The solution to this program will be the Newton direction. The Karush-Kuhn-Tucker conditions for this program are necessary and sufficient, and are:

\[
\begin{cases}
C - \theta \bar{X}^{-1} + \theta \bar{X}^{-1} D \bar{X}^{-1} = \sum_{i=1}^{m} u_i A_i \\
A_i \cdot D = 0, \quad i = 1, \ldots, m .
\end{cases}
\]

These equations are called the Normal Equations. Let \( D' \) and \( u' \) denote the solution to the Normal Equations. Note in particular from the first equation in (15) that \( D' \) must be symmetric. Suppose that \((D', u')\) is the (unique) solution of the Normal Equations (15). We obtain the new value of the primal variable \( X \) by taking the Newton step, i.e.,

\[ X' = \bar{X} + D' . \]

We can produce new values of the dual variables \((u, S)\) by setting the new value of \( u \) to be \( u' \) and by setting \( S' = C - \sum_{i=1}^{m} u' A_i \). Using (15), then, we have:

\[ S' = \theta \bar{X}^{-1} - \theta \bar{X}^{-1} D' \bar{X}^{-1} . \]

We have the following very powerful convergence theorem which demonstrates the quadratic convergence of Newton’s method for this problem, with an explicit guarantee of the range in which quadratic convergence takes place.

**Theorem 9.1 (Explicit Quadratic Convergence of Newton’s Method.)**

Suppose that \((\bar{X}, \bar{u}, \bar{S})\) is a \( \beta \)-approximate solution of BSDP(\( \theta \)) and \( \beta < 1 \).
Let \((D', u')\) be the solution to the Normal Equations (15), and let

\[
X' = \bar{X} + D'
\]

and

\[
S' = \theta \bar{X}^{-1} - \theta \bar{X}^{-1} D' \bar{X}^{-1}.
\]

Then \((X', u', S')\) is a \(\beta^2\)-approximate solution of \(BSDP(\theta)\).

**Proof:** Our current point \(\bar{X}\) satisfies:

\[
A_i \cdot \bar{X} = b_i, \ i = 1, \ldots, m, \ \bar{X} = \bar{L} \bar{L}^T > 0
\]

\[
\sum_{i=1}^{m} \bar{u}_i A_i + \bar{S} = C
\]

\[
\|I - \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L}\| \leq \beta < 1.
\]

Furthermore the Newton direction \(D'\) and multipliers \(u'\) satisfy:

\[
A_i \cdot D' = 0, \ i = 1, \ldots, m
\]

\[
\sum_{i=1}^{m} u'_i A_i + S' = C
\]

\[
X' = \bar{X} + D' = \bar{L}(I + \bar{L}^{-1} D' \bar{L}^{-T}) \bar{L}^T
\]

\[
S' = \theta \bar{X}^{-1} - \theta \bar{X}^{-1} D' \bar{X}^{-1} = \theta \bar{L}^{-T} (I - \bar{L}^{-1} D' \bar{L}^{-T}) \bar{L}^{-1}.
\]

We will first show that \(\|\bar{L}^{-1} D' \bar{L}^{-T}\| \leq \beta\). It turns out that this is the crucial fact from which everything will follow nicely. To prove this, note that

\[
\sum_{i=1}^{m} \bar{u}_i A_i + \bar{S} = C = \sum_{i=1}^{m} u'_i A_i + S' = \sum_{i=1}^{m} u'_i A_i + \theta \bar{L}^{-T} (I - \bar{L}^{-1} D' \bar{L}^{-T}) \bar{L}^{-1}.
\]

Taking the inner product with \(D'\) yields:

\[
\bar{S} \cdot D' = \theta \bar{L}^{-T} \bar{L}^{-1} \cdot D' - \theta \bar{L}^{-T} \bar{L}^{-1} D' \bar{L}^{-T} \bar{L}^{-1} \cdot D',
\]

49
which we can rewrite as:
\[ \bar{L}^T \bar{S} \bar{L} \cdot \bar{L}^{-1} D' \bar{L}^{-T} - \theta I \cdot \bar{L}^{-1} D' \bar{L}^{-T} \cdot \bar{L}^{-1} D' \bar{L}^{-T} , \]
which we finally rewrite as:
\[ \|\bar{L}^{-1} D' \bar{L}^{-T}\|^2 = \left(I - \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L}\right) \cdot \bar{L}^{-1} D' \bar{L}^{-T} . \]
Invoking the Cauchy-Schwartz inequality we obtain:
\[ \|\bar{L}^{-1} D' \bar{L}^{-T}\|^2 \leq \|I - \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L}\| \|\bar{L}^{-1} D' \bar{L}^{-T}\| \leq \beta \|\bar{L}^{-1} D' \bar{L}^{-T}\| , \]
from which we see that \( \|\bar{L}^{-1} D' \bar{L}^{-T}\| \leq \beta . \)

It therefore follows that
\[ X' = \bar{L}(I + \bar{L}^{-1} D' \bar{L}^{-T})\bar{L}^T > 0 \]
and
\[ S' = \theta \bar{L}^{-T}(I - \bar{L}^{-1} D' \bar{L}^{-T})\bar{L}^{-1} > 0 , \]
since \( \|\bar{L}^{-1} D' \bar{L}^{-T}\| \leq \beta < 1 , \) which guarantees that \( I \pm \bar{L}^{-1} D' \bar{L}^{-T} > 0 . \)

Next, factorize
\[ I + \bar{L}^{-1} D' \bar{L}^{-T} = M^2 , \]
(where \( M = M^T \)) and note that
\[ X' = \bar{L}M M \bar{L}^T = \bar{L}' (\bar{L}')^T \]
where we define \( \bar{L}' = \bar{L}M . \) Then note that
\[ I - \frac{1}{\theta} (\bar{L}')^T S' \bar{L}' = I - \frac{1}{\theta} M \bar{L}^T S' \bar{L}M \]
\[ = I - \frac{1}{\theta} M \bar{L}^T (\theta \bar{L}^{-T}(I - \bar{L}^{-1} D' \bar{L}^{-T})\bar{L}^{-1})\bar{L}M = I - M(I - \bar{L}^{-1} D' \bar{L}^{-T})M \]
\[ = I - MM + M(\bar{L}^{-1} D' \bar{L}^{-T})M = I - MM + M(MM - I)M = (I - MM)(I - MM) \]
\[ = (\bar{L}^{-1} D' \bar{L}^{-T})(\bar{L}^{-1} D' \bar{L}^{-T}) . \]
From this we obtain:
\[ \|I - \frac{1}{\theta} (\bar{L}')^T S' \bar{L}'\| = \|(\bar{L}^{-1} D' \bar{L}^{-T})(\bar{L}^{-1} D' \bar{L}^{-T})\| \leq \|(\bar{L}^{-1} D' \bar{L}^{-T})\|^2 \leq \beta^2 , \]
where the middle inequality above uses Proposition 9.5. This shows that \( (X', u', S') \) is a \( \beta^2 \)-approximate solution of \( BSDP(\theta) . \)
9.3 Complexity Analysis of the Algorithm

**Theorem 9.2 (Relaxation Theorem).** Suppose that $(\bar{X}, \bar{u}, \bar{S})$ is a $\beta$-approximate solution of BSDP($\theta$) and $\beta < 1$. Let

$$\alpha = 1 - \frac{\sqrt{\beta} - \beta}{\sqrt{\beta} + \sqrt{n}}$$

and let $\theta' = \alpha \theta$. Then $(\bar{X}, \bar{u}, \bar{S})$ is a $\sqrt{\beta}$-approximate solution of BSDP($\theta'$).

**Proof:** The triplet $(\bar{X}, \bar{u}, \bar{S})$ satisfies $A_i \cdot \bar{X} = b_i, i = 1, \ldots, m, \bar{X} > 0$, and $\sum_{i=1}^{m} \bar{u}_i A_i + \bar{S} = C$, and so it remains to show that

$$\left\| \frac{1}{\theta'} \bar{L}^T \bar{S} \bar{L} - I \right\| \leq \sqrt{\beta},$$

where $\bar{X} = \bar{L} \bar{L}^T$. We have

$$\left\| \frac{1}{\theta'} \bar{L}^T \bar{S} \bar{L} - I \right\| = \left\| \frac{1}{\alpha} \bar{L}^T \bar{S} \bar{L} - I \right\|$$

$$= \left\| \frac{1}{\alpha} \left( \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L} - I \right) - (1 - \frac{1}{\alpha}) I \right\|$$

$$\leq \left( \frac{1}{\alpha} \right) \left\| \frac{1}{\theta} \bar{L}^T \bar{S} \bar{L} - I \right\| + \left| \frac{1-\alpha}{\alpha} \right| \| I \|$$

$$\leq \frac{\beta}{\alpha} + \left( \frac{1-\alpha}{\alpha} \right) \sqrt{n}$$

$$= \frac{\beta + \sqrt{n}}{\alpha} - \sqrt{n}$$

$$= \sqrt{\beta} + \sqrt{n} - \sqrt{n} = \sqrt{\beta}.$$

\[ \blacksquare \]

**Theorem 9.3 (Convergence Theorem).** Suppose that $(X^0, u^0, S^0)$ is a $\beta$-approximate solution of BSDP($\theta^0$) and $\beta < 1$. Then for all $k = 1, 2, 3, \ldots$, $(X^k, u^k, S^k)$ is a $\beta$-approximate solution of BSDP($\theta^k$).
**Proof:** By induction, suppose that the theorem is true for iterates 0, 1, 2, ..., $k$.

Then $(X^k, u^k, S^k)$ is a $\beta$-approximate solution of $BSDP(\theta^k)$.

From the Relaxation Theorem, $(X^k, u^k, S^k)$ is a $\sqrt{\beta}$-approximate solution of $BSDP(\theta^{k+1})$ where $\theta^{k+1} = \alpha \theta^k$.

From the Quadratic Convergence Theorem, $(X^{k+1}, u^{k+1}, S^{k+1})$ is a $\beta$-approximate solution of $BSDP(\theta^{k+1})$.

Therefore, by induction, the theorem is true for all values of $k$. ■

**Theorem 9.4 (Complexity Theorem).** Suppose that $(X^0, u^0, S^0)$ is a $\beta = \frac{1}{4}$-approximate solution of $BSDP(\theta^0)$. In order to obtain primal and dual feasible solutions $(X^k, u^k, S^k)$ with a duality gap of at most $\varepsilon$, one needs to run the algorithm for at most

$$k = \left\lceil 6\sqrt{n} \ln \left( \frac{1.25 X^0 \bullet S^0}{0.75 \varepsilon} \right) \right\rceil$$

iterations.

**Proof:** Let $k$ be as defined above. Note that

$$\alpha = 1 - \frac{\sqrt{\beta} - \beta}{\sqrt{\beta} + \sqrt{n}} = 1 - \frac{1}{2 + \frac{1}{2 + \frac{1}{2 + 4\sqrt{n}}}} \leq 1 - \frac{1}{6\sqrt{n}}.$$

Therefore

$$\theta^k \leq \left( 1 - \frac{1}{6\sqrt{n}} \right)^k \theta^0.$$

This implies that

$$C \bullet X^k - \sum_{i=1}^{m} b_i u_i^k = X^k \bullet S^k \leq \theta^k n(1 + \beta) \leq \left( 1 - \frac{1}{6\sqrt{n}} \right)^k (1.25n\theta^0).$$
from (12). Taking logarithms, we obtain

\[ \ln \left( C \cdot X^k - \sum_{i=1}^{m} b_i u_i^k \right) \leq k \ln \left( 1 - \frac{1}{6\sqrt{n}} \right) + \ln \left( \frac{1.25}{0.75} X^0 \cdot S^0 \right) \]

\[ \leq \frac{-k}{6\sqrt{n}} + \ln \left( \frac{1.25}{0.75} X^0 \cdot S^0 \right) \]

\[ \leq -\ln \left( \frac{1.25}{0.75} \frac{X^0 \cdot S^0}{\varepsilon} \right) + \ln \left( \frac{1.25}{0.75} X^0 \cdot S^0 \right) = \ln(\varepsilon) . \]

Therefore \( C \cdot X^k - \sum_{i=1}^{m} b_i u_i^k \leq \varepsilon . \]

### 9.4 How to Start the Method from a Strictly Feasible Point

The algorithm and its performance relies on having a starting point \((X^0, u^0, S^0)\) that is a \(\beta\)-approximate solution of the problem \(BSDP(\theta^0)\). In this subsection, we show how to obtain such a starting point, given a positive definite feasible solution \(X^0\) of SDP.

We suppose that we are given a target value \(\theta^0\) of the barrier parameter, and we are given \(X = X^0\) that is feasible for \(BSDP(\theta^0)\), that is, \(A_i \cdot X^0 = b_i, \quad i = 1, \ldots, m, \quad X^0 \succ 0\). We will attempt to approximately solve \(BSDP(\theta^0)\) starting at \(X = X^0\), using the Newton direction at each iteration. The formal statement of the algorithm is as follows:

**Step 0. Initialization.** Data is \((X^0, \theta^0)\). \(k = 0\). Assume that \(X^0\) satisfies \(A_i \cdot X^0 = b_i, \quad i = 1, \ldots, m, \quad X^0 \succ 0\).

**Step 1. Set Current values.** \(\bar{X} = X^k\). Factor \(\bar{X} = LL^T\).

**Step 2. Compute Newton Direction and Multipliers.** Compute the Newton step \(D\) for \(BSDP(\theta^0)\) at \(X = \bar{X}\) by solving the following system of equations in the variables \((D, u)\):
\[
\begin{aligned}
C - \theta^0 \bar{X}^{-1} + \theta^0 \bar{X}^{-1}D\bar{X}^{-1} &= \sum_{i=1}^{m} u_i A_i \\
A_i \cdot D &= 0, \quad i = 1, \ldots, m.
\end{aligned}
\] (17)

Denote the solution to this system by \((D', u')\). Set

\[
S' = C - \sum_{i=1}^{m} u'_i A_i.
\]

**Step 3. Test the Current Point.** If \(\|\bar{L}^{-1}D'\bar{L}^{-T}\| \leq \frac{1}{4}\), stop. In this case, \(\bar{X}\) is a \(\frac{1}{4}\)-approximate solution of \(BSDP(\theta^0)\), along with the dual values \((u', S')\).

**Step 4. Update Primal Point.**

\[
X' = \bar{X} + \alpha D'
\]

where

\[
\alpha = \frac{0.2}{\|L^{-1}D' L^{-T}\|}.
\]

Alternatively, \(\alpha\) can be computed by a line-search of \(f_{\theta^0}(\bar{X} + \alpha D')\).

**Step 5. Reset Counter and Continue.** \(X^{k+1} \leftarrow X', \ k \leftarrow k + 1\). Go to Step 1.

The following proposition validates Step 3 of the algorithm:

**Proposition 9.1** Suppose that \((D', u')\) is the solution of the Normal equations (17) for the point \(\bar{X}\) for the given value \(\theta^0\) of the barrier parameter, and that

\[
\|\bar{L}^{-1}D'\bar{L}^{-T}\| \leq \frac{1}{4}\,.
\]

Then \(\bar{X}\) is a \(\frac{1}{4}\)-approximate solution of \(BSDP(\theta^0)\).
Proof: We must exhibit values \((u, S)\) that satisfy \(\sum_{i=1}^{m} u_i A_i + S = C\) and

\[
\left\| I - \frac{1}{\theta^0} L^T S L \right\| \leq \frac{1}{4}.
\]

Let \((D', u')\) solve the Normal equations (17), and let \(S' = C - \sum_{i=1}^{m} u'_i A_i\).

Then we have from (17) that

\[
I - \frac{1}{\theta^0} L^T S' \bar{L} = I - \frac{1}{\theta^0} L^T (\theta^0 (L^{-T} L^{-1} - \bar{L}^{-T} L^{-1} D' \bar{L}^{-T} L^{-1} )) \bar{L} = \bar{L}^{-1} D' \bar{L}^{-T},
\]

whereby

\[
\left\| I - \frac{1}{\theta^0} L^T S' \bar{L} \right\| = \left\| \bar{L}^{-1} D' \bar{L}^{-T} \right\| \leq \frac{1}{4}.
\]

The next proposition shows that whenever the algorithm proceeds to Step 4, then the objective function \(f_{\theta^0}(X)\) decreases by at least 0.025\(\theta^0\):

**Proposition 9.2** Suppose that \(\bar{X}\) satisfies \(A_i \bullet \bar{X} = b_i, \ i = 1, \ldots, m\), and \(\bar{X} > 0\). Suppose that \((D', u')\) is the solution of the Normal equations (17) for the point \(\bar{X}\) for a given value \(\theta^0\) of the barrier parameter, and that

\[
\left\| \bar{L}^{-1} D' \bar{L}^{-T} \right\| > \frac{1}{4}.
\]

Then for all \(\gamma \in [0, 1)\),

\[
f_{\theta^0} \left( \bar{X} + \frac{\gamma}{\left\| \bar{L}^{-1} D' \bar{L}^{-T} \right\|} D' \right) \leq f_{\theta^0}(\bar{X}) + \theta^0 \left( -\gamma \left\| \bar{L}^{-1} D' \bar{L}^{-T} \right\| + \frac{\gamma^2}{2(1 - \gamma)} \right).
\]

In particular,

\[
f_{\theta^0} \left( \bar{X} + \frac{0.2}{\left\| \bar{L}^{-1} D' \bar{L}^{-T} \right\|} D' \right) \leq f_{\theta^0}(\bar{X}) - 0.025\theta^0. \quad (18)
\]

In order to prove this proposition, we will need two powerful facts about the logarithm function:

55
Fact 1. Suppose that $|x| \leq \delta < 1$. Then

$$\ln(1 + x) \geq x - \frac{x^2}{2(1 - \delta)}.$$ 

**Proof:** We have:

$$\ln(1 + x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \ldots$$

$$\geq x - \frac{|x|^2}{2} - \frac{|x|^3}{3} - \frac{|x|^4}{4} - \ldots$$

$$\geq x - \frac{|x|^2}{2} - \frac{|x|^3}{2} - \frac{|x|^4}{2} - \ldots$$

$$= x - \frac{x^2}{2}(1 + |x| + |x|^2 + |x|^3 + \ldots)$$

$$= x - \frac{x^2}{2(1 - |x|)}$$

$$\geq x - \frac{x^2}{2(1 - \delta)}.$$ 

Fact 2. Suppose that $R \in S^n$ and that $\|R\| \leq \gamma < 1$. Then

$$\ln(\det(I + R)) \geq I \cdot R - \frac{\gamma^2}{2(1 - \gamma)}.$$ 

**Proof:** Factor $R = QDQ^T$ where $Q$ is orthonormal and $D$ is a diagonal matrix of the eigenvalues of $R$. Then first note that $\|R\| = \sqrt{\sum_{j=1}^{n} D_{jj}^2}$. We
then have:

\[
\ln(\det(I + R)) = \ln(\det(I + QDQ^T)) = \ln(\det(I + D)) = \sum_{j=1}^{n} \ln(1 + D_{jj}) \geq \sum_{j=1}^{n} \left( D_{jj} - \frac{D_{jj}^2}{2(1-\gamma)} \right) = I \bullet D - \frac{\|R\|^2}{2(1-\gamma)} \geq I \bullet Q^T R - \frac{\gamma^2}{2(1-\gamma)} = I \bullet R - \frac{\gamma^2}{2(1-\gamma)}.
\]

Proof of Proposition 9.2. Let

\[
\alpha = \frac{\gamma}{\|L^{-1}D\tilde{L}^{-T}\|},
\]

and notice that

\[
\|\alpha L^{-1}D\tilde{L}^{-T}\| = \gamma.
\]
Then
\[
f_{\theta^0} \left( \tilde{X} + \frac{\gamma}{\|L^{-1}D'L^{-T}\|} D' \right) = f_{\theta^0} \left( \tilde{X} + \alpha D' \right)
\]
\[
= C \bullet \tilde{X} + \alpha C \bullet D' - \theta^0 \ln(\det(I + \alpha L^{-1} D'L^{-T} L^T))
\]
\[
= C \bullet \tilde{X} - \theta^0 \ln(\det(\tilde{X})) + \alpha C \bullet D' - \theta^0 \ln(\det(I + \alpha L^{-1} D'L^{-T}))
\]
\[
\leq f_{\theta^0}(\tilde{X}) + \alpha C \bullet D' - \theta^0 \alpha I \bullet \tilde{X}^{-1} L^{-T} + \theta^0 \frac{\gamma^2}{2(1 - \gamma)}
\]
\[
= f_{\theta^0}(\tilde{X}) + \alpha C \bullet D' - \theta^0 \alpha L^{-T} \tilde{X}^{-1} \bullet D' + \theta^0 \frac{\gamma^2}{2(1 - \gamma)}
\]
\[
= f_{\theta^0}(\tilde{X}) + \alpha (C - \theta^0 \tilde{X}^{-1}) \bullet D' + \theta^0 \frac{\gamma^2}{2(1 - \gamma)}.
\]

Now, \((D', u')\) solve the Normal equations:

\[
\begin{cases}
C - \theta^0 \tilde{X}^{-1} + \theta^0 \tilde{X}^{-1} D' \tilde{X}^{-1} = \sum_{i=1}^{m} u_i' A_i \quad \text{(19)} \\
A_i \bullet D' = 0, \quad i = 1, \ldots, m.
\end{cases}
\]

Taking the inner product of both sides of the first equation above with \(D'\) and rearranging yields:

\[
\theta^0 \tilde{X}^{-1} D' \bullet \tilde{X}^{-1} D' = -(C - \theta^0 \tilde{X}^{-1}) \bullet D' .
\]

Substituting this in our inequality above yields:
Substituting $\gamma = 0.2$ and $\|\bar{L}^{-1}D'\bar{L}^{-T}\| > \frac{1}{4}$ yields the final result.

Last of all, we prove a bound on the number of iterations that the algorithm will need in order to find a $\frac{1}{4}$-approximate solution of $BSDP(\theta^0)$:

**Proposition 9.3** Suppose that $X^0$ satisfies $A_i \cdot X^0 = b_i$, $i = 1, \ldots, m$, and $X^0 \succ 0$. Let $\theta^0$ be given and let $f^*_\theta$ be the optimal objective function value of $BSDP(\theta^0)$. Then the algorithm initiated at $X^0$ will find a $\frac{1}{4}$-approximate solution of $BSDP(\theta^0)$ in at most

$$k = \left\lceil \frac{f^0(\bar{X}) - f^*_\theta}{0.025\theta^0} \right\rceil$$

iterations.

**Proof:** This follows immediately from (18). Each iteration that is not a $\frac{1}{4}$-approximate solution decreases the objective function $f^0(X)$ of $BSDP(\theta^0)$ by at least $0.025\theta^0$. Therefore, there cannot be more than

$$\left\lceil \frac{f^0(\bar{X}) - f^*_\theta}{0.025\theta^0} \right\rceil$$

iterations that are not $\frac{1}{4}$-approximate solutions of $BSDP(\theta^0)$.
9.5 Some Properties of the Frobenius Norm

**Proposition 9.4** If $M \in S^n$, then

1. \[ \|M\| = \sqrt{\sum_{j=1}^{n} (\lambda_j(M))^2}, \text{ where } \lambda_1(M), \lambda_2(M), \ldots, \lambda_n(M) \text{ is an enumeration of the } n \text{ eigenvalues of } M. \]

2. If $\lambda$ is any eigenvalue of $M$, then $|\lambda| \leq \|M\|$.

3. $|\text{trace}(M)| \leq \sqrt{n}\|M\|$.

4. If $\|M\| < 1$, then $I + M > 0$.

**Proof:** We can factorize $M = QDQ^T$ where $Q$ is orthonormal and $D$ is a diagonal matrix of the eigenvalues of $M$. Then

\[ \|M\| = \sqrt{M \cdot M} = \sqrt{QDQ^T \cdot QDQ^T} = \sqrt{\text{trace}(QDQ^TQDQ^T)} \]

\[ = \sqrt{\text{trace}(Q^TQDQ^TD)} = \sqrt{\text{trace}(DD)} = \sqrt{n \sum_{j=1}^{n} (\lambda_j(M))^2}. \]

This proves the first two assertions. To prove the third assertion, note that

\[ \text{trace}(M) = \text{trace}(QDQ^T) = \text{trace}(Q^TQD) \]

\[ = \text{trace}(D) = \sum_{j=1}^{n} \lambda_j(M) \leq \sqrt{n} \sqrt{\sum_{j=1}^{n} (\lambda_j(M))^2} = \sqrt{n}\|M\|. \]

To prove the fourth assertion, let $\lambda'$ be an eigenvalue of $I + M$. Then $\lambda' = 1 + \lambda$ where $\lambda$ is an eigenvalue of $M$. However, from the second assertion, $\lambda' = 1 + \lambda \geq 1 - \|M\| > 0$, and so $M > 0$. ■

**Proposition 9.5** If $A, B \in S^n$, then $\|AB\| \leq \|A\|\|B\|$. 

60
Proof: We have
\[ \|AB\| = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \left( \sum_{k=1}^{n} A_{ik} B_{kj} \right)^2} \]
\[ \leq \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} \left( \sum_{k=1}^{n} A_{ik}^2 \right) \left( \sum_{k=1}^{n} B_{kj}^2 \right)} \]
\[ = \sqrt{\left( \sum_{i=1}^{n} \sum_{k=1}^{n} A_{ik}^2 \right) \left( \sum_{j=1}^{n} \sum_{k=1}^{n} B_{kj}^2 \right)} = \|A\| \|B\|. \]

10 Exercises

1. Consider the matrix \( M \) defined as follows:

\[ M = \begin{pmatrix} P & v \\ v^T & d \end{pmatrix}, \]

where \( P \succ 0 \), \( v \) is a vector, and \( d \) is a scalar. Prove that \( M \succ 0 \) if and only if \( d - v^T P^{-1} v > 0 \). Prove that \( M \succeq 0 \) if and only if \( d - v^T P^{-1} v \geq 0 \).

2. Consider the matrix \( M \) defined as follows:

\[ M = \begin{pmatrix} W & x \\ x^T & t \end{pmatrix}, \]

where \( W \) is a symmetric matrix, \( x \) is a vector, and \( t \) is a scalar. Prove that

\[ M \succeq 0 \iff t \geq 0, \ W \succeq 0, \ \text{and} \ tW \succeq xx^T. \]
3. For a (square) matrix $M \in \mathbb{R}^{n \times n}$, define $\text{trace}(M) = \sum_{j=1}^{n} M_{jj}$, and for two matrices $A, B \in \mathbb{R}^{k \times l}$ define

$$A \bullet B := \sum_{i=1}^{k} \sum_{j=1}^{l} A_{ij}B_{ij}.$$ 

Prove that:

(a) $A \bullet B = \text{trace}(A^T B)$.

(b) $\text{trace}(A^T B) = \text{trace}(BA^T)$.

(c) If $M \in S^{n \times n}$, then $\text{trace}(M) = \sum_{j=1}^{n} \lambda_j(M)$.

(d) If $M \in S^{n \times n}$, then $M \bullet M = \sum_{j=1}^{n} (\lambda_j(M))^2$.

4. Let $S^{n \times n}$ denote the set of all symmetric $n \times n$ matrices. Let $K := \{X \in S^{n \times n} \mid X \succeq 0\}$ and define

$$K^* := \{S \in S^{n \times n} \mid S \bullet X \succeq 0 \text{ for all } X \in K\}.$$ 

Prove that $K^* = K$.

5. Show that if $K$ is a solid cone, then $K^*$ is a pointed cone. Show that if $K$ is a pointed cone, then $K^*$ is a solid cone.

6. Suppose that $X \succeq 0$ and $S \succeq 0$, and $S \bullet X = 0$. Prove that $SX = 0$, i.e., the ordinary matrix product of these two matrices is the matrix of all zeroes.

7. Let $\|v\|$ denote the regular Euclidean norm, namely $\|v\| := \sqrt{v^T v}$. Use Exercise 1 to show the following equivalence:

$$\|x\| \leq t \iff \begin{pmatrix} tI & x \\ x^T & t \end{pmatrix} \succeq 0.$$ 

8. Suppose $K \succeq M \succeq 0$. Prove that $\det(K) \geq \det(M)$.

9. Suppose $K \succeq M$, and let $\lambda_1(K), \ldots, \lambda_n(K)$ and $\lambda_1(M), \ldots, \lambda_n(M)$ denote the ordered eigenvalues of $K$ and $M$ in increasing order. Prove that $\lambda_i(K) \geq \lambda_i(M)$ for all $i = 1, \ldots, n$. 

62
10. Suppose $M \in S^{n \times n}$ and $f \in \mathbb{R}^n$ are given.

(a) Show that $f^T x$ is constant for all $x$ that satisfy $M x = f$.

(b) Show that

$$
\begin{bmatrix}
M & f \\
f^T & \theta
\end{bmatrix} \succeq 0 \iff \begin{cases}
M \succeq 0, \\
\text{the system } M x = f \text{ has a solution, and} \\
\theta \geq f^T x \text{ for all } x \text{ satisfying } M x = f.
\end{cases}
$$

11. Let $\lambda_{\min}$ denote the smallest eigenvalue of the symmetric matrix $Q$.

Show that the following three optimization problems each have optimal objective function value equal to $\lambda_{\min}$.

(P1): minimize $d^T Q d$

s.t. $d^T I d = 1$.

(P2): maximize $\lambda$

s.t. $Q \succeq \lambda I$.

(P3): minimize $Q \cdot X$

s.t. $I \cdot X = 1$

$X \succeq 0$.

12. Consider the problem:

(P): minimize $d^T Q d$

s.t. $d^T M d = 1$.

If $M > 0$, show that (P) is equivalent to:

(S): minimize $Q \cdot X$

s.t. $M \cdot X = 1$

$X \succeq 0$.
What is the SDP dual of (S)?

13. Suppose that $Q \succeq 0$. Prove the following:

$$x^T Q x + q^T x + c \leq 0$$

if and only if there exists $W$ for which

$$
\begin{pmatrix}
Q & \frac{1}{2}q \\
\frac{1}{2}q^T & c
\end{pmatrix} \cdot 
\begin{pmatrix}
W & x^T \\
x & 1
\end{pmatrix} \leq 0 \quad \text{and} \quad 
\begin{pmatrix}
W & x^T \\
x & 1
\end{pmatrix} \succeq 0.
$$

Hint: use the equivalence stated in Exercise 2.

14. Consider the matrix

$$M = \begin{pmatrix} A & B \\ B^T & C \end{pmatrix},$$

where $A, C$ are symmetric matrices and $A \succ 0$. Prove that $M \succeq 0$ if and only if $C - B^T A^{-1} B \succeq 0$. Prove that $M \succ 0$ if and only if $C - B^T A^{-1} B \succ 0$.

15. For a symmetric matrix $V \in S^{n \times n}$ we define the “Frobenius norm” of $V$ to be:

$$
\|V\| := \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} V_{ij}},
$$

which is simply the Euclidean norm of the $n^2$-component vector formed by writing out all of the components of $V$ as vector in $\mathbb{R}^{n^2}$. Using Exercise 14 and Exercise 9, show the equivalence:

$$
\|V\|^2 \leq t \iff \begin{pmatrix} Y & V \\ V & I \end{pmatrix} \succeq 0, \ I \cdot Y \leq t.
$$

16. For a given diagonal matrix $D$, let $D^+$ and $D^-$ denote the positive and negative parts of $D$, so that $D = D^+ - D^-$ and $D^+ \cdot D^- = 0$. Let $\bar{X} \in S^{n \times n}$ be given. Use the result in Exercise 15 above to show that the projection of $\bar{X}$ onto $S^+_{n \times n}$ is given by $Q D^Q DQ^T$ where $\bar{X} = QDQ^T$ expresses the eigendecomposition of $\bar{X}$ into appropriate orthonormal matrices and a diagonal matrix. (Here the projection of $\bar{X}$ onto $S^+_{n \times n}$ is the point $W \in S^+_{n \times n}$ that minimizes $\|W - \bar{X}\|$, measured where the norm here is the Frobenius norm defined in Exercise 15.)