5 Semidefinite programming

The vector space $\mathbf{S}^n$ of $n \times n$ real symmetric matrices is endowed with the *trace inner product*:

$$\langle A, B\rangle := \mathrm{Tr}(AB) = \sum_{1\leq i,j\leq n} A_{ij}B_{ij}.$$

Recall the notation:

$$A \succeq 0 \iff A \text{ positive semidefinite.}$$

We are also going to use the following convenient notation:

$$A \succeq B \iff A - B \succeq 0.$$

A *semidefinite program* (SDP) is an optimisation problem of the form

$$\begin{array}{ll} \underset{X\in\mathbf{S}^n}{\text{minimise}} & \langle C, X\rangle \\ \text{subject to} & \mathcal{A}(X) = b \\ & X \succeq 0. \end{array} \tag{1}$$

where $C \in \mathbf{S}^n$, $\mathcal{A} : \mathbf{S}^n \to \mathbb{R}^m$ is a linear map and $b \in \mathbb{R}^m$. The optimisation variable is $X \in \mathbf{S}^n$. The constraint $X \succeq 0$ means that X is positive semidefinite. A semidefinite program (1) is entirely specified by the data $\mathcal{A} : \mathbf{S}^n \to \mathbb{R}^m, b \in \mathbb{R}^m, C \in \mathbf{S}^n$.

Recognising semidefinite programs It is important to be able to recognise semidefinite programs. We now give some examples:

1. Consider the problem

$$\underset{X\in\mathbf{S}^3}{\text{minimise}} \quad X_{12} + X_{13} \quad \text{s.t.} \quad X \succeq 0, \;\; \mathrm{diag}(X) = (1,1,1). \tag{2}$$

   This is a semidefinite program of the form (1) where

$$C = \frac{1}{2}\begin{bmatrix} 0 & 1 & 1\\ 1 & 0 & 0\\ 1 & 0 & 0\end{bmatrix}, \quad \mathcal{A} : X \mapsto (X_{11}, X_{22}, X_{33}), \quad b = (1,1,1).$$

2. Consider

$$\underset{x_1,x_2\in\mathbb{R}}{\text{minimise}} \quad x_1 + 2x_2 \quad \text{s.t.} \quad \begin{bmatrix} 1-x_1 & x_2\\ x_2 & 1+x_1\end{bmatrix} \succeq 0. \tag{3}$$

   This can be put in the form (1) as follows: if we call X the matrix $\left[\begin{smallmatrix} 1-x_1 & x_2\\ x_2 & 1+x_1\end{smallmatrix}\right]$ the only constraint we have on X is that $X_{11} + X_{22} = 2$. In terms of the entries of X the cost function can be written as $(X_{22} - X_{11})/2 + X_{12} + X_{21}$. It is thus not difficult to see that our problem (3) is "equivalent" to (1) with:

$$C = \begin{bmatrix} -1/2 & 1\\ 1 & 1/2\end{bmatrix}, \quad \mathcal{A} : X \mapsto X_{11} + X_{22}, \quad b = 2. \tag{4}$$

3. Some problems may be cast in the form (1) even though they do not look like semidefinite programs. Consider for example the problem:

$$\underset{x_1,x_2\in\mathbb{R}}{\text{minimise}} \quad x_1 + 2x_2 \quad \text{s.t.} \quad x_1^2 + x_2^2 \leq 1. \tag{5}$$

As written, (5) is *not* a semidefinite program of the form (1). The following claim however shows that it can be expressed in the form (1).

Claim 5.1. *For any $x_1, x_2 \in \mathbb{R}$ we have*

$$x_1^2 + x_2^2 \leq 1 \quad \iff \quad \begin{bmatrix} 1 - x_1 & x_2 \\ x_2 & 1 + x_1 \end{bmatrix} \succeq 0. \tag{6}$$

Proof. We know that a 2×2 symmetric matrix is positive semidefinite if and only if its trace and determinant are nonnegative: the trace of the 2×2 matrix in (6) is $2 \geq 0$ and its determinant is $(1 - x_1)(1 + x_1) - x_2^2 = 1 - x_1^2 - x_2^2 \geq 0$ if and only if $x_1^2 + x_2^2 \leq 1$. □

The equivalence (6) shows that problem (5) is the same as (3) which we saw is a semidefinite program.

The main take-away point from these examples is: semidefinite optimisation problems can be more or less difficult to recognise. From now on in the course, it will be taken for granted that problems like (2) and (3) (or (7), see exercise below) are semidefinite optimisation problems, and we will not write down the identifications (4) (which should be "mechanical"). Example (5) however is more difficult and requires justification of the form (6).

Exercise 5.1. *Let $A_0, A_1, \ldots, A_m$ be $n \times n$ real symmetric matrices and let $c \in \mathbb{R}^m$. Show how to put the optimisation problem*

$$\underset{x\in\mathbb{R}^m}{\textit{minimise}} \quad c^T x \quad \textit{s.t.} \quad A_0 + x_1 A_1 + \cdots + x_m A_m \succeq 0 \tag{7}$$

into standard semidefinite form (1).

Exercise 5.2. *Show that any linear programming problem is also a semidefinite program.*

Solving semidefinite programs What makes the class of *semidefinite optimisation problems* interesting is that there are efficient computer algorithms to solve them, given the input data $\mathcal{A}, b, c$ (e.g., *interior-point algorithms*). We will not discuss these algorithms in this course. These algorithms have been implemented and interfaced with user-friendly modelling languages. In practice one can solve instances of (1) for n up to ≈ 100 in a reasonable amount of time on a personal computer. Larger instances can be solved by exploiting structure or using specialized algorithms. We recommend the MATLAB packages CVX (`http://www.cvxr.com`) and YALMIP (`https://yalmip.github.io/`) as a starting point. For example the semidefinite program (3) can be solved using the following code on MATLAB, using the modelling language CVX:

```
cvx_begin sdp
   variables x1 x2
   minimize x1 + 2*x2
   subject to
    [1-x1   x2;
       x2  1+x1] >= 0;
cvx_end
```

CVX will do the identification (4) automatically and will call a semidefinite programming solver.

Example: (symmetric) matrix completion Consider the following (symmetric) *matrix completion*: we observe certain entries of an unknown symmetric matrix and the goal is to recover the symmetric matrix with the smallest *trace norm*. The trace norm of a symmetric matrix X is defined as the sum of the absolute values of the eigenvalues:

$$\|X\|_{\text{tr}} = \sum_{i=1}^{n} |\lambda_i(X)|$$

where $\lambda_1(X), \ldots, \lambda_n(X)$ are the eigenvalues of X. The *trace norm* is also sometimes called the *nuclear norm* of X, or the *Schatten 1-norm*. It can be interpreted as the ℓ_1 norm of the eigenvalues of X.

Exercise 5.3. *Show that the trace norm is the* dual norm *of the* spectral norm, *i.e., for any symmetric matrix X we have:*

$$\|X\|_{\text{tr}} = \max_{Y \in \mathbf{S}^n : \|Y\| \leq 1} \langle X, Y \rangle.$$

where $\|Y\|$ is the spectral norm of Y.

Let $\Omega \subseteq \{1, \ldots, n\} \times \{1, \ldots, n\}$ be the subset of entries that we observe and let D_{ij} be the values that we observe. We want to solve the problem:

$$\underset{X \in \mathbf{S}^n}{\text{minimise}} \quad \|X\|_{\text{tr}} \quad \text{s.t.} \quad X_{ij} = D_{ij} \; \forall (i,j) \in \Omega. \tag{8}$$

We now show how to formulate (8) as a semidefinite program. Consider the following semidefinite program:

$$\underset{X, Y \in \mathbf{S}^n}{\text{minimise}} \quad \text{Tr}(Y) \quad \text{s.t.} \quad X_{ij} = D_{ij} \; \forall (i,j) \in \Omega, \quad Y - X \succeq 0, \quad Y + X \succeq 0. \tag{9}$$

The next claim shows that the two problems (8) and (9) are "equivalent".

Claim 5.2. *Assume X is feasible for (8). Then there exists $Y \in \mathbf{S}^n$ such that (X, Y) is feasible for (9) and $\text{Tr}(Y) \leq \|X\|_{\text{tr}}$. Conversely if (X, Y) is feasible for (9) then $\|X\|_{\text{tr}} \leq \text{Tr}(Y)$. As a consequence, the optimal values of (8) and (9) are equal.*

Proof. For the first direction, assuming $X = \sum_{i=1}^{n} \lambda_i v_i v_i^T$ is an eigendecomposition of X, we let $Y = \sum_{i=1}^{n} |\lambda_i| v_i v_i^T$. Note that $\text{Tr}(Y) = \|X\|_{\text{tr}}$. We need to show that $Y - X \succeq 0$ and $Y + X \succeq 0$. Note that $Y \pm X = \sum_{i=1}^{n} (|\lambda_i| \pm \lambda_i) v_i v_i^T$ and thus is positive semidefinite since $|\lambda_i| \pm \lambda_i \geq 0$.

For the converse, assume (X, Y) satisfy the constraints of (9). We need to show that $\|X\|_{\mathrm{tr}} \leq \mathrm{Tr}(Y)$. Let $X = \sum_{i=1}^{n} \lambda_i v_i v_i^T$ be an eigenvalue decomposition of X. Let $P^+ = \sum_{i:\lambda_i \geq 0} v_i v_i^T$ and $P^- = \sum_{i:\lambda_i < 0} v_i v_i^T$. Note that $P^+ + P^- = I_n$ and $\mathrm{Tr}(X(P^+ - P^-)) = \|X\|_{\mathrm{tr}}$. Since $Y - X \succeq 0$ and $P^+ \succeq 0$ we have $\mathrm{Tr}((Y - X)P^+) \geq 0$. Similarly we have $\mathrm{Tr}((Y + X)P^-) \geq 0$. Adding these two inequalities we get $\mathrm{Tr}(Y) - \|X\|_{\mathrm{tr}} \geq 0$ which is what we wanted. $\square$