Lecture 13: Semidefinite programming

Rajat Mittal

IIT Kanpur

Semidefinite programming is a class of convex optimization where optimization function is linear and the constraints are either linear equalities/inequalities or generalized inequalities with respect to semidefinite cone. Hence, it is linear programming with the additional power of generalized inequalities for positive semidefinite cone ($\mathcal{S}_n$).

1 Definition

A semidefinite program in standard form looks like,

$$\begin{aligned} \max \quad & C \circ X \\ \text{s.t.} \quad & A_i \circ X = b_i, \ \forall i = 1, \cdots, m \\ & X \succeq 0. \end{aligned}$$

Here X is the variable matrix of dimension $n \times n$. The matrix C is called the cost or objective matrix. A_i's are the constraint matrices. C and A_i's have the same dimension as X $(n \times n)$. b_i's are scalars and the vector b (with b_i) as components is known as constraint matrix. Remember that $A \circ B = tr(A^T B)$ is the hadamard product between two matrices.

Many of the standard tricks used in linear programming to convert non-standard form into standard form can also be used here. For example, converting inequalities into equalities, changing minimum to maximum and change of variables.

Suppose the program is,

$$\begin{aligned} \max \quad & Tr(X) \\ \text{s.t.} \quad & X = \begin{pmatrix} 1 & x \\ 1 & x \end{pmatrix} \succeq 0. \end{aligned}$$

Exercise 1. Find the value of this semidefinite program.

Look at another similar program,

$$\begin{aligned} \inf / \min \quad & x_1 \\ \text{s.t.} \quad & \begin{pmatrix} x_1 & 1 \\ 1 & x_2 \end{pmatrix} \succeq 0. \end{aligned}$$

Exercise 2. Find the value of this semidefinite program.

2 Equivalent definitions

2.1 Form with positive semidefinite constraints

Another standard form for semidefinite programming is:

$$\begin{aligned} & \min b^T y \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^m y_i A_i - C \succeq 0 \end{aligned}$$

Let us take a look at why these two forms are equivalent. Suppose the matrix $\sum_i y_i A_i - C = Z$. This will be our new variable matrix. Now the variables will be Z and scalars y_i's. The linear constraints will be $\forall\ i, j;\ z_{ij} = \sum_k y_k (A_k)_{ij}$ (where z_{ij} are the entries of matrix Z). Hence the program becomes,

$$\begin{gathered} \min b^T y \\ \text{s.t. } \ \forall\ i, j;\ z_{ij} = \textstyle\sum_k y_k (A_k)_{ij} \\ Z \succeq 0 \end{gathered}$$

It almost looks like the standard form but variables y do not occur in the semidefinite constraint. Notice the old trick of converting unrestricted variables to positive variables. Say $y_i = y_i' - y_i''$ and $y_i', y_i'' \geq 0$. Then these variables y_i', y_i'' can be put in a separate block and included in the semidefinite constraint.

$$\begin{gathered} \min b^T y \\ \text{s.t.} \qquad \forall\ i, j;\ z_{ij} = \textstyle\sum_k (y_k' - y_k'')(A_k)_{ij} \\ Z = \begin{pmatrix} Z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & y_1' & \cdots & 0 & 0 & 0 & 0 \\ 0 & \vdots & \ddots & \vdots & 0 & 0 & 0 \\ 0 & 0 & 0 & y_m' & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_1'' & \cdots & 0 \\ 0 & 0 & 0 & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & y_m'' \end{pmatrix} \succeq 0 \end{gathered}$$

Exercise 3. Prove that we don't need to put the explicit constraint that off-diagonal entries (blocks) are zero.

2.2 Gram matrix formulation

We know that any positive semidefinite matrix can be written as the gram matrix of vectors. Suppose X can be expressed as the gram matrix of $u_1, \cdots, u_n \in \mathbb{R}^k$. Then the semidefinite program looks like,

$$\begin{gathered} \max \quad \textstyle\sum_{ij} C_{ij} u_i^T u_j \\ \text{s.t. } \ \textstyle\sum_{ij} A_{ij}^{(k)} u_i^T u_j = b_k,\ \forall k = 1, \cdots, m \end{gathered}$$

Notice that we don't need the $X \succeq 0$ constraint now. This form can be also be understood as the program where constraints and objective functions are linear in inner-products of the vectors. This form is really useful when we want our variables in optimization problem to be able to assume vector values.

Remark: This is not a linear program, since constraints are on the inner-products.

3 Examples

3.1 Minimizing the maximum eigenvalue

Suppose we are give a matrix $M(x)$, which depends affinely on the variables in x. That means every entry in $M(x)$ can be written as an affine function of variables in x (say $M(x)_{ij} = a_1 x_1 + \cdots + a_n x_n + b$). The problem is to minimize the maximum eigenvalue of $M(x)$ over all x, i.e.,

$$\min_x \max_i \lambda_i(M(x))$$

Clearly this is not in the standard form of SDP. The trick here is to introduce another variable η to change $\min\max$ to only min. Suppose $\lambda_{\max}(M)$ represents the maximum eigenvalue of M, then

$$\begin{aligned} &\min \eta \\ \text{s.t.}\ \ &\eta \geq \lambda_{\max}(M(x)) \end{aligned}$$

Now use the fact that $\lambda_{\max} I - M(x) \succeq 0$. Hence,

$$\begin{aligned} &\min \eta \\ \text{s.t.}\ \ &\eta I - M(x) \succeq 0 \end{aligned}$$

This is one of the alternative form discussed in the last section (why?). Here the variables are (η, x).

Exercise 4. Write an SDP to find the maximum eigenvalue of a matrix M.