

Lecture 5

Semidefinite Programming(SDP)

You should all be familiar with linear programs, the theory of duality, and the use of linear programs to solve a variety of problems in polynomial time; flow or routing problems are typical examples. While LPs are very useful, and can be solved very efficiently, they don't cover all possible situations. In this lecture we study an extension of linear programming, semidefinite programming, which is much more general but still solvable in polynomial time.

5.1 MAXCUT

Consider an undirected graph $G = (V, E)$. Then

$$\text{MAXCUT}(G) := \max_{x_i \in \{\pm 1\}, i \in V} \sum_{(i,j) \in E} \frac{1 - x_i x_j}{2}$$

is precisely the number of edges that are cut by the largest cut in the graph (a cut is a partition of the vertices into two sets). Replacing the constraints $x_i \in \{\pm 1\}$ by $x_i \in [-1, 1]$ does not change the optimum. This makes all the constraints linear, but the objective function itself is not; it is quadratic. A natural idea is to introduce auxiliary variables z_{ij} and reformulate the problem as

$$\begin{aligned} \max \quad & \sum_{(i,j) \in E} \frac{1 + z_{ij}}{2} \\ \text{s.t.} \quad & z_{ij} = x_i x_j \\ & -1 \leq x_i \leq 1. \end{aligned}$$

Now the objective function has been linearized, but the constraints are quadratic! Of course it shouldn't be too much of a surprise that we're having a hard time finding a simple form for MAXCUT(G), as the problem is NP-hard.

In particular this is not even a convex program. To see why, observe that both $z = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$ and $z' = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$ are feasible (for the first, take $x_1 = x_2 = 1$; for the second take

$x_1 = 1, x_2 = -1$), but $(z + z')/2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ is not (for instance, because it has rank 2). But here's an interesting observation. What if we allow x_i to be a vector \vec{x}_i , of norm at most 1? Then we can factor $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = (\vec{e}_i \cdot \vec{e}_j)_{i,j}$. More generally, if $z_{ij} = \vec{x}_i \cdot \vec{x}_j$ and $z'_{ij} = \vec{x}'_i \cdot \vec{x}'_j$ then $\frac{1}{2}(z_{ij} + z'_{ij}) = \frac{1}{\sqrt{2}} \begin{pmatrix} \vec{x}_i & \vec{x}'_i \end{pmatrix} \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} \vec{x}_j \\ \vec{x}'_j \end{pmatrix}$. This may sound strange, but let's see what we get. Since we allow a larger set for the variables we have a *relaxation*

$$\begin{aligned} \max \quad & \sum_{(i,j) \in E} \frac{1 + z_{ij}}{2} \\ \text{s.t.} \quad & z_{ij} = \vec{x}_i \cdot \vec{x}_j \\ & \|\vec{x}_i\|^2 \leq 1 \quad \forall i \\ & \vec{x}_i \in \mathbb{R}^d. \end{aligned} \tag{5.1}$$

What did we do? We relaxed the constraint that $x_i \in [-1, 1]$, i.e. x_i is a one-dimensional vector of norm at most 1, to allowing a vector \vec{x}_i of arbitrary dimension (we use the \vec{x} notation to emphasize the use of vectors, but soon we'll omit the arrow and simply write $x \in \mathbb{R}^d$ as usual). Note in particular that the feasible region for this problem is unbounded, and a priori it could be that higher and higher-dimensional vectors would give better and better objective values.

To see that this is not the case, we're going to rewrite (5.1) as one that only involves a finite number of variables using the theory of positive semidefinite (PSD) matrices. Recall the following equivalent definitions for a PSD matrix Z :

Lemma 5.1. *A symmetric matrix $Z \in \mathbb{R}^{n \times n}$ is positive semidefinite if and only if any of the following equivalent conditions holds:*

- (1) $\forall x \in \mathbb{R}^n, x^T Z x \geq 0$;
- (2) $\exists X \in \mathbb{R}^{n \times d}$ such that $Z = X^T X$;
- (3) All eigenvalues of Z are nonnegative;
- (4) $Z = U D U^T$, where U is orthogonal and D is diagonal with nonnegative entries;
- (5) $Z = \sum_i \lambda_i u_i u_i^T$, $\lambda_i \geq 0$, $\{u_i\}$ orthonormal basis of \mathbb{R}^n .

We denote $\mathcal{P}(\mathbb{R}^n) := \{Z \in \mathbb{R}^{n \times n} | Z \succeq 0\}$ the positive semidefinite cone (a cone is a convex set C such that $x \in C, \lambda \geq 0 \implies \lambda x \in C$).

Exercise 1. Prove that the five conditions in the definition are equivalent. You may use the spectral theorem for real symmetric matrices.

For $A, B \in \mathbb{R}^{n \times n}$ symmetric we'll write $A \succeq B$ when $A - B \succeq 0$. For $Z \in \mathbb{R}^{n \times n}$, we will use the notation

$$Y \bullet Z := \text{Tr}(Y^T Z) = \sum_{i,j} Y_{ji} Z_{ji},$$

which defines an inner product on real $n \times n$ matrices.

Exercise 2. Show that if $A \succeq B$ and $C \succeq 0$ then $A \bullet C \geq B \bullet C$.

Then

$$\sum_{(i,j) \in E} \frac{1 + z_{ij}}{2} = \frac{|E|}{2} + \frac{1}{2} G \bullet Z,$$

where $G \in \{0, 1\}^{n \times n}$ is the adjacency matrix of the graph. The constraints $z_{ij} = \vec{x}_i \cdot \vec{x}_j$ are equivalent to the factorization $Z = X^T X$, where $X \in \mathbb{R}^{d \times n}$ has the \vec{x}_i as its columns. The constraint $\|\vec{x}_i\|^2 \leq 1$ is equivalent to $E_i \bullet Z \leq 1$, where E_i is a matrix with a 1 in the (i, i) -th entry and 0 everywhere else. Using the characterization of PSD matrices given in Lemma 5.1, the problem (5.1) can be reformulated as

$$\begin{aligned} \max \quad & \frac{|E|}{2} + \frac{1}{2} G \bullet Z \\ \text{s.t.} \quad & Z \succeq 0 \\ & E_i \bullet Z \leq 1 \quad \forall i \\ & Z \in \mathbb{R}^{n \times n}, \end{aligned}$$

where here $Z \succeq 0$ means that Z is PSD (we'll soon simply write $Z \geq 0$ instead of $Z \succeq 0$ to mean that Z is PSD).

5.2 Semidefinite programs

In general a semidefinite program is the optimization of a linear function under linear and semidefinite constraints. Let's see some examples.

1. For any symmetric matrix B , its largest eigenvalue can be expressed as

$$\begin{aligned} \min \quad & x_1 \\ \text{s.t.} \quad & x_1 \mathbb{I} - B \succeq 0 \end{aligned}$$

2. The following SDP

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

is equivalent to $x_1, x_2 \geq 0$ and $x_1 x_2 \geq 1$. The optimum is 0, but this optimal value is not attained at any feasible point. This is an important difference with LPs. From now on

we'll have to be careful and write “inf” or “sup” instead of “min” or “max” whenever we're writing an SDP for which we're not sure whether the optimum is attained.

3. This SDP

$$\begin{aligned}
 \text{inf } & x_n \\
 \text{s.t. } & x_0 \geq 2 \\
 & \begin{pmatrix} 1 & x_0 \\ x_0 & x_1 \end{pmatrix} \succeq 0 \\
 & \begin{pmatrix} 1 & x_1 \\ x_1 & x_2 \end{pmatrix} \succeq 0 \\
 & \vdots \\
 & \begin{pmatrix} 1 & x_{n-1} \\ x_{n-1} & x_n \end{pmatrix} \succeq 0
 \end{aligned}$$

evaluates to 2^{2^n} . Here even writing down the optimum requires a number of bits (2^n) that is exponential in the instance size ($O(n)$ bits). This could not happen for LPs either.

The last example demonstrates that an SDP cannot always be solved exactly in polynomial time, even if it is both feasible and bounded. Two additional conditions will allow us to give polynomial-time algorithms. First, we will only solve SDPs approximately. This takes care of the second example: we will only require the solver to return a feasible point that achieves an objective value at least $\text{opt} - \varepsilon$, for any $\varepsilon > 0$ (the running time will depend on ε). Second, the SDP solver will require as input an a priori bound on the size of the solution. This gets rid of the third example. Finally, we will also need to require that the SDP is *strictly feasible*, meaning that there is a feasible point X that is strictly positive.

Under these three conditions it is possible to show that SDPs can be solved efficiently. An algorithm that works well is the ellipsoid algorithm — the same used to solve LPs. In a lecture or two we'll see another algorithm based on a matrix variant of the MW algorithm. For now let's state one of the best results known:

Theorem 5.2. *For any $\epsilon > 0$ and any SDP such that the feasible region K is such that $\exists r, R > 0, \vec{O}$ with*

$$B(\vec{O}, r) \subset K \subset B(\vec{O}, R),$$

a feasible X such that $B \bullet X \geq \text{OPT}(SDP) - \epsilon$ can be computed in time $\text{poly}(\log \frac{R}{r} + |SDP| + \log \frac{1}{\epsilon})$, where $|SDP|$ denotes the number of bits required to completely specify the SDP instance.

5.3 Dual of a SDP

Just as linear programs, every SDP has a *canonical form* as follows:

$$\begin{aligned}
 (\mathcal{P}) \quad & \sup \quad B \bullet X \\
 & \text{s.t.} \quad A_i \bullet X \leq c_i \quad \forall i \in \{1, \dots, m\} \\
 & \quad \quad X \succeq 0,
 \end{aligned} \tag{5.2}$$

where $B \in \mathbb{R}^{n \times n}$, $A_1, \dots, A_m \in \mathbb{R}^{n \times n}$, $c_i \in \mathbb{R}$.

Exercise 3. Write each of the three SDPs from the previous section in canonical form (i.e. specify what the matrices B , A_i , and the reals c_i should be).

Let's develop the duality theory for SDPs. What is the dual of (\mathcal{P}) given in (5.2)? Let's proceed in the same way as one derives the dual of an LP: form linear combinations of the constraints in order to prove upper bounds on the objective value. More precisely, for any $y_1, \dots, y_m \geq 0$, if

$$y_1 A_1 + \dots + y_m A_m \succeq B$$

then for any primal feasible X

$$B \bullet X \preceq (y_1 A_1 + \dots + y_m A_m) \bullet X = y^T c,$$

where the second inequality uses $A \leq Z \implies A \bullet X \leq Z \bullet X$ for any $X \succeq 0$. We obtain the dual

$$\begin{aligned}
 (\mathcal{D}) \quad & \inf \quad y^T c \\
 & \text{s.t.} \quad y_1, \dots, y_m \geq 0 \\
 & \quad \quad y_1 A_1 + \dots + y_m A_m - B \succeq 0,
 \end{aligned}$$

and we just showed:

Theorem 5.3 (Weak Duality). *If both the primal and the dual problems are feasible and bounded, then*

$$\text{OPT}(\mathcal{P}) \leq \text{OPT}(\mathcal{D}).$$

While weak duality always holds under the same conditions as for LPs, strong duality can fail dramatically!

Example 5.4. Consider the optimization problem

$$\begin{aligned}
 \inf \quad & -y_1 \\
 \text{s.t.} \quad & \begin{pmatrix} 0 & y_1 & 0 \\ y_1 & y_2 & 0 \\ 0 & 0 & 1 - y_1 \end{pmatrix} \succeq 0 \\
 & y_1, y_2 \geq 0.
 \end{aligned} \tag{5.3}$$

The determinant of a positive semidefinite matrix should be no less than 0, thus $0 \times y_2 - y_1^2 \geq 0$, and the optimum of the above SDP is 0. You can check that its dual is given by

$$\begin{aligned} \sup \quad & -X_{33} \\ \text{s.t.} \quad & X_{12} + X_{21} - X_{33} \leq -1 \\ & X_{22} \leq 0 \\ & X \succeq 0. \end{aligned}$$

Since X_{22} is 0, the positivity condition implies $X_{12} = X_{21} = 0$, so $-X_{33} \leq -1$ and the optimum is -1 .

In spite of this strong duality does hold as long as both the primal and dual SDPs are *strictly feasible*:

Theorem 5.5 (Strong Duality). *Suppose both the primal and the dual are strictly feasible and bounded, then*

$$\text{OPT}(P) = \text{OPT}(D).$$

Proof. Suppose for contradiction that $\alpha < \beta$. Let $\alpha' = \frac{\alpha + \beta}{2}$ be the mid-point. Define the set

$$K = \left\{ \begin{pmatrix} \sum_i y_i A_i - B \\ \alpha' - c^\top y \end{pmatrix} \in \mathbb{R}^{n^2+1}, y \in \mathbb{R}^m \right\}$$

Then K does not intersect the positive cone $\text{Pos}(\mathbb{R}^n) \times \mathbb{R}_+$. By definition K is an affine subspace, hence there exists an affine hyperplane $\begin{pmatrix} Z \\ \mu \end{pmatrix} \in \mathbb{R}^{n^2+1}, \delta \in \mathbb{R}$ that includes K but does not intersect $\text{Pos}(\mathbb{R}^n) \times \mathbb{R}_+$:

- $\forall y \in \mathbb{R}^m, Z \bullet (\sum y_i A_i - B) + \mu(\alpha' - c^\top y) = \delta$
- $\forall Q \succeq 0, \forall q \geq 0, Z \bullet Q + \mu q > \delta$

Taking $Q = 0, q = 0$ in the second constraint shows that necessarily $\delta < 0$. Suppose Z has a negative eigenvalue. For any value of δ , taking $q = 0$ and Q as a large enough multiple of the projection on the eigenvector of Z corresponding to this eigenvalue would contradict the second inequality. Thus $Z \succeq 0$.

Setting $y = 0$ in the first constraint gives $Z \bullet B = \alpha' \mu - \delta$. Setting $y = e_i$, we obtain $Z \bullet A_i = \mu c_i$. Finally, taking y to be any strictly feasible point, meaning $\begin{cases} \sum_i y_i A_i - B \succ 0 \\ \alpha' - c^\top y < 0 \end{cases}$

(since any feasible solution will satisfy $c^\top y \geq \beta > \alpha'$), gives $\mu > 0$.

Define $Z' = \frac{Z}{\mu}$. Then Z' is primal feasible: $Z' \succeq 0$ since both $Z \succeq 0$ and $\mu > 0$, and $Z' \bullet A_i = c_i$ for each i . We can also compute the objective value $Z' \bullet B = \alpha' - \frac{\delta}{\mu} > \alpha$ since $\delta < 0, \mu > 0$. This is a contradiction with our original assumption that the primal optimum was α . \square