IOE 691: Approximation Algorithms Date: 4/3/2017

Lecture Notes: Maximum Cut (Semidefinite Programming)

Instructor: Viswanath Nagarajan Scribe: Cheng Sun

1 Semidefinite Programming

Recall the following equivalent conditions for an $n \times n$ real symmetric matrix X to be *positive semidefinite* (denoted $X \succeq 0$).

- $v^t X v \geq 0$ for all $v \in \mathbb{R}^n$.
- all eigenvalues of X are non-negative ($\lambda \in \mathbb{R}$ is an eigenvalue of X if there is some $v \in \mathbb{R}^n$ with $Xv = \lambda v$).
- $X = V^t V$ for some matrix $V \in \mathbb{R}_{k \times n}$ where $k \leq n$.

Semidefinite programs (SDPs) form an important class of convex programs and can be solved efficiently to any desired level of accuracy. Here we discuss two equivalent definitions of SDPs.

$$\begin{aligned} \max \quad & \sum_{i,j} w_{ij} \cdot x_{ij} \\ \text{s.t.} \quad & \sum_{i,j} a_{ij}^r \cdot x_{ij} \geq b_r, \quad \forall r \in I \\ & X \succeq 0 \end{aligned}$$

Above, w_{ij}, a_{ij}^r and b_r are given numbers and $X = (x_{ij})_{1 \leq i,j \leq n}$ denote the decision variables. Note that the objective and the first set of constraints are linear in the variables.

Note that since X is positive semidefinite, we can write $X = V^t V$ for some $k \times n$ matrix V. If $v_1, \cdots v_n \in \mathbb{R}^k$ denote the columns of V then $x_{ij} = \langle v_i, v_j \rangle$. (We use the standard notation $\langle u.v \rangle$ for the inner product of vectors.) As $k \leq n$, we can equivalently write the above problem as:

$$\begin{aligned} \max \quad & \sum_{i,j} w_{ij} \langle v_i, v_j \rangle \\ \text{s.t.} \quad & \sum_{i,j} a_{ij}^r \langle v_i, v_j \rangle \geq b_r, \quad \forall r \in I \\ & v_i \in \mathbb{R}^n, \ \forall i \in [n] \end{aligned}$$

Using the ellipsoid algorithm one can obtain in polynomial time, a solution to any SDP that has a small additive violation on each constraint. This violation will not be important in our application since we aim for an approximation algorithm anyway.

2 SDP for Max-Cut

Given an undirected graph $G = (V, E)$ with weights $w_{ij} \geq 0$ for edges $(i, j) \in E$, the goal is to find a subset $S \subseteq V$ that maximizes $\sum\limits_{i \in S, j \in V \setminus S} w_{ij}$. We assume that every pair of distinct vertices is connected by a unique edge (possibly of weight zero). We will show:

Theorem 2.1 ([1]). *There is a* 0.878*-approximation algorithm for maximum cut.*

It is also known that any polynomial-size linear program (from a certain general class of LPs) for max-cut has integrality gap no greater than 1/2.

When we apply SDP to solve the max-cut problem, we use variables v_i for each $i \in V$. Ideally, we want the solution to only have two directions, either $v_i = e$ (corresponding to $i \in S$) or $v_i = -e$ (corresponding to $i \notin S$), where e is a fixed unit vector. In the ideal solution, all the vertices whose $v_i = e$ belong to the solution subset S, and the rest vertices (with $v_i = -e$) belong to the complement of S. We can set up the SDP as follows.

$$\begin{array}{ll} \max & \sum\limits_{i,j \in V} \frac{1}{2} w_{ij}(1 - \langle v_i, v_j \rangle) \\ \text{s.t.} & \langle v_i, v_i \rangle = 1, \quad \forall i \in V \\ & v_i \in \mathbb{R}^n, \quad \forall i \in V \end{array}$$

A simple observation is that the optimal solution of SDP gives a value at least as good as the optimal solution to the original max-cut problem. This uses the "ideal" vector solution discussed above.

However, the optimal SDP solution need not have the form of the ideal solution. So we need to round this general vector solution. The rounding is performed in a simple randomized manner. Choose a uniformly random unit vector $r \in \mathbb{R}^n$, and define $S = \{i \in V : \langle r, v_i \rangle \geq 0\}$.

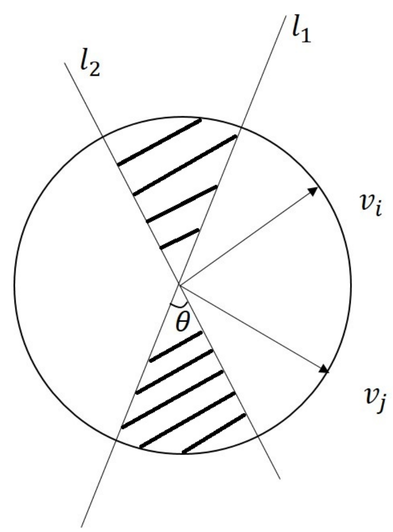

Figure 1: v_i, v_j in 2d plane

Lemma 2.1. *For each* $i, j \in V$, $\Pr[i, j \text{ separated by } S] = \Pr[|\{i, j\} \cap S| = 1] \geq 0.878 \frac{1 - \langle v_i, v_j \rangle}{2}$

Proof. Since only two vectors v_i and v_j are involved, consider the 2-dimensional plane H where

$v_i \in H, v_j \in H$. See Figure 1. Let $\bar{r}$ be the projection of r on H. Note that the normalization $\bar{r}/\|\bar{r}\|$ is a uniform unit vector in $\mathbb{R}^2$. It is also clear that $\langle r, v_i\rangle = \langle \bar{r}, v_i\rangle$. See Figure 1. Then we have

$$Pr[i, j \text{ separated by } S] = Pr[\langle \bar{r}, v_i\rangle \text{ and } \langle \bar{r}, v_j\rangle \text{ have opposite sign}].$$

If $\bar{r}$ falls into the the shadow area in Figure 1, $\langle \bar{r}, v_i\rangle$ and $\langle \bar{r}, v_j\rangle$ have opposite sign. So it follows that $\Pr[i, j \text{ separated by } S] = \frac{2\theta}{2\pi} = \frac{1}{\pi}\cos^{-1}\langle v_i, v_j\rangle$. Now,

$$\begin{aligned}
\Pr[i, j \text{ separated by } S] &= \frac{1-\langle v_i, v_j\rangle}{\pi} \cdot \frac{\cos^{-1}\langle v_i, v_j\rangle}{1-\langle v_i, v_j\rangle} \\
&\geq \frac{1-\langle v_i, v_j\rangle}{2} \cdot \frac{2}{\pi}\left(\min_{x\in[-1,1]} \frac{\cos^{-1} x}{1-x}\right) \\
&\geq 0.878 \cdot \frac{1-\langle v_i, v_j\rangle}{2}
\end{aligned}$$

The last inequality can be seen numerically. □

By taking the sum over all pairs $i, j \in V$ and linearity of expectation, it follows that the expected value of the solution S is at least 0.878 times the SDP optimum.

It is also known that this approximation ratio is the best possible under the "Unique Games Conjecture" [2].

References

[1] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

[2] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan ODonnell. Optimal inapproximability results for max-cut and other 2-variable csps? *SIAM Journal on Computing*, 37(1):319–357, 2007.