CSE 632: Analysis of Algorithms II **Fall 2017**

Lecture 20 (11/03/2017): Ellipsoid Method, Semidefinite Programming

Lecturer: Shi Li *Scribe: Yunus Esencayi*

20.1 Semidefinite Programming

A symmetric matrix $X \in R^{n\times n}$ is positive semi-definite(psd) if $\forall y \in R^n$, we have $y^T xy \geq 0$. Use $X \succeq 0$ to denote X is psd.

The following statements are equivalent if X is symmetric.
1. $X \succeq 0$
2. X has non-negative eigenvalues.
3. $X = V^T V$ for some $V \in R^{m\times n}, m \leq n$.
4. $X = \sum_{i=1}^{n} \lambda_i w_i w_i^T$ for some $\lambda_i \geq 0, \forall i, \{w_i\}_i$ is orthonormal basis.

Semi-definite Programming (SDP)
There are two forms.
Form 1:

$$\begin{aligned}
&\text{max/min } \sum_{i,j} c_{i,j}.x_{i,j} \\
&\text{such that } \sum_{i,j} a_{k,i,j}.x_{i,j} \geq b_k \ (\forall k) \\
&x_{i,j} = x_{j,i} \ (\forall i, j) \\
&X = (x_{i,j})_{i,j} \text{ is psd.}
\end{aligned}$$

To use ellipsoid method to solve SDP form 1, we need:
- a seperation oracle O
- if X is not psd, find y such that $y^T xy < 0$, return the constraint $(yy^T).x \geq 0$.

Lemma 20.1 *$X = (X_{i,j})_{i,j}$ is psd $\iff$ $\forall y \in R^n, (yy^T).X \geq 0$.*

Proof: Define $A.B = \sum_{i,j} A_{i,j} Bi, j$.
$y^T xy = \sum_{i,j} y_i.x_{i,j}.y_j = \sum_{i,j} y_i.y_j.x_{i,j} = (yy^T).X$ ■

Form 2: (Vector Programming)

$$\begin{aligned}
&\max/\min \sum_{i,j} c_{i,j} < v_i, v_j > \\
&\text{such that } \sum_{i,j} a_{k,i,j} < v_i, v_j > \geq b_k \ (\forall k) \\
&v_i \in R^n \ (\forall i = 1, 2, ..., n)
\end{aligned}$$

How to convert from SDP form 2 to form 1:

Given $v_1, v_2, ..., v_n$, define $x_{i,j} =< v_i, v_j >$,

$$\text{then } X = \begin{pmatrix} ...v_1... \\ ...v_2... \\ \vdots \\ ...v_n... \end{pmatrix} \cdot \begin{pmatrix} . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ v_1 & v_2 & ... & v_n \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \end{pmatrix} \text{ is psd.}$$

How to convert from SDP form 1 to form 2:

Given a psd X, $X = V^T V$,

$$\text{let } V = \begin{pmatrix} . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \\ v_1 & v_2 & ... & v_n \\ . & . & ... & . \\ . & . & ... & . \\ . & . & ... & . \end{pmatrix} \text{then, } v_1, v_2, ..., v_n \text{ gives a solution form 2.}$$

Maximum Weight Cut Problem

Given $G = (V, E)$, $\{w_e\}_{e \in E}$,
find $S \subseteq V$ so as to maximize $\sum_{(u,v)\ inE:|\{u,v\} \cap S|=1} w_{(u,v)}$.

Quadratic Programming for Maximum Weighted Cut

$\forall v \in V$, define

$$y_v = \begin{cases} 1, & \text{if } v \in S \\ -1 & \text{if } v \notin S \end{cases}$$

So, the quadratic programming will be as follows:

$$\max \frac{1}{2}. \sum_{(i,j)\in E} (1 - y_i y_j)$$
$$\text{such that } y_i^2 = 1 \ (\forall i \in V)$$

SDP relaxation can be formulated as follows:

$$\max \frac{1}{2}. \sum_{(i,j)\in E} (1 - < v_i, v_j >)$$
$$\text{such that } < v_i, v_j >= 1 \ (\forall i \in V)$$
$$v_i \in R^n \ (\forall i \in V)$$

Let Z_{up} be the value of SDP relaxation and OPT be the value of the optimum solution. Since SDP was defined on a larger set, we can say that $Z_{up} \geq OPT$.

Our goal is to find a solution S of value $\geq \alpha.Z_{up}$ for some α.

Choose $r_i \sim \mathcal{N}(0,1)$ for each $i \in V$.
$r = (r_1, r_2, ..., r_n)$
Let $i \in S$, if $< r, v_i > \geq 0$ and $i \notin S$, if $< r, v_i > < 0$.

$$\Rightarrow prob[(i,j) \text{ is cut}] = \frac{arccos < v_i, v_j >}{\pi}$$

Let P be the plane containing v_i and v_j, then the direction of r projected to P is uniform distributed.

$$\frac{arccos < v_i, v_j >}{(1 - < v_i, v_j >)/2} \geq \alpha_{GW} := inf_{x\in[-1,1)} \frac{arccos x/\pi}{(1-x)/2} \approx -0.878$$

$$\mathbb{E}[\text{value of } S] = \sum_{(i,j)\in E} w_{(i,j)}. \frac{arccos < v_i, v_j >}{\pi} \geq \alpha_{GW}. \sum_{(i,j)\in E} w_{(i,j)}. \frac{1 - < v_i, v_j >}{2} = \alpha_{GW}.Z_{up} \geq \alpha_{GW}.OPT$$