Advanced Algorithms

Lecture 7: Semidefinite Programming

Lecturer: Nathan Klein

1 Positive Semidefinite Matrices

1.1 Recap

Last class we explored the following max cut formulation:

$$\begin{aligned} \max \quad & \sum_{\{u,v\}\in E} \frac{1}{2}(1-y_{uv}) \\ \text{s.t.} \quad & y_{uv} = x_u x_v && \forall u,v \in V \\ & y_{vv} = 1 && \forall v \in V \end{aligned}$$

Since the constraints $y_{uv} = x_u x_v$ were non-linear, we dropped those constraints and then tried to find a separation oracle from the following initial polytope:

$$\begin{aligned} \max \quad & \sum_{\{u,v\}\in E} \frac{1}{2}(1-y_{uv}) \\ \text{s.t.} \quad & y_{vv} = 1 && \forall v \in V \end{aligned}$$

We noticed that constraints *generated* from the inequalities $(\sum_{v\in V} c_v x_v)^2 \geq 0$ (for some $c \in \mathbb{R}^V$) led to some useful inequalities, such as $y_v \geq -1$ and $y_{uv} + y_{uw} + y_{vw} \geq -\frac{3}{2}$. Let's remember what "generated" means here by considering three variables u, v, w:

$$(x_u + x_v + x_w)^2 \geq 0 \iff x_u^2 + x_v^2 + x_w^2 + 2x_u x_v + 2x_u x_w + 2x_v x_w \geq 0$$

Now, replacing the squared terms with 1 and using $y_{uv} = x_u x_v$, this is equivalent to $y_{uv} + y_{uw} + y_{vw} \geq -\frac{3}{2}$. This says that we cannot cut all three edges in a triangle. So the question was: **can we separate over all constraints of the form $(\sum_{v\in V} c_v x_v)^2 \geq 0$?**

1.2 Linear Algebra Recap

$(\sum_{v\in V} c_v x_v)^2 \geq 0$ is equivalent to $0 \leq \sum_{u,v\in V} c_u c_v x_u x_v = \sum_{(u,v)\in V\times V} c_u c_v y_{uv}$ (where pairs u, v appear twice with the understanding that $y_{uv} = y_{vu}$). But letting $Y = (y_{uv})_{u,v\in V}$ be a matrix, this is equivalent to $c^T Y c \geq 0$.

It's reasonable for now to think of $c^T Y c = \sum_{(u,v)\in V\times V} c_u c_v y_{uv}$ as the definition of $c^T Y c$. But let's also derive it for ourselves for general matrices.

Lemma 1.1. *Let $A \in \mathbb{R}^{n\times n}$ and $x \in \mathbb{R}^n$. Then*

$$x^T A x = \sum_{i=1}^n \sum_{j=1}^n x_i x_j A_{ij}.$$

Proof. We have:

$$Ax = \sum_{j=1}^{n} x_j \cdot (\text{column } j \text{ of } A),$$

so each entry A_{ij} gets multiplied by x_j from the right. Adding in the left multiplication, we obtain:

$$x^T(Ax) = \sum_{i=1}^{n} x_i \cdot (\text{row } i \text{ of } Ax),$$

so each entry in row i gets multiplied by x_i from the left. Therefore each entry A_{ij} gets multiplied by $x_i x_j$, and summing over all entries gives

$$x^T Ax = \sum_{i,j} x_i x_j A_{ij}.$$ □

Remember that $\lambda \in \mathbb{R}$ is an eigenvalue of matrix $A \in \mathbb{R}^{n\times n}$ if there is a vector $x \in \mathbb{R}^n$ so that $Ax = \lambda x$. In this case, x is called an eigenvector of A and (x, λ) is an eigenvector, eigenvalue pair.

The criteria it turns out we want for Y is positive semidefiniteness. Remember that Y is symmetric since we assume $y_{uv} = y_{vu}$.

Definition 1.2 (Positive Semidefinite)**.** *A symmetric matrix is positive semidefinite (PSD) if all of its eigenvalues are non-negative.*

To warm up a bit more with linear algebra, we will show in a second that this is equivalent to asking that $c^T Yc \geq 0$ for all $c \in \mathbb{R}^n$. First, we need a key fact about symmetric matrices:

Theorem 1.3 (Spectral Theorem)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix. Then, there are n eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ with corresponding orthonormal eigenvectors $v_1, \ldots, v_n$ so that*

$$A = \sum_{i=1}^{n} \lambda_i v_i v_i^T = V\Lambda V^T$$

where V has columns $v_1, \ldots, v_n$ (so $V^T V = I$) and Λ is the diagonal matrix with $\Lambda_{ii} = \lambda_i$.

Remember that a set of vectors $v_1, \ldots, v_n$ is orthonormal if for any $i \neq j$ we have $\langle v_i, v_j\rangle = 0$, and $\|v_i\|_2 = \sum_{i=1}^n v_i^2 = 1$. These vectors $v_1, \ldots, v_n$ therefore form a basis for $\mathbb{R}^n$.

A useful consequence of the spectral theorem is we can now apply functions to symmetric matrices and still get a handle on the resulting matrix. For example:

Lemma 1.4.

$$A^2 = \sum_{i=1}^{n} \lambda_i^2 v_i v_i^T$$

Proof. By the spectral theorem,

$$A^2 = (\sum_{i=1}^{n} \lambda_i v_i v_i^T)^2 = \sum_{i,j\in[n]\times[n]} \lambda_i \lambda_j v_i v_j = \sum_{i=1}^{n} \lambda_i^2 v_i v_i^T$$

where in the last equation we used that $v_i v_j = 0$ for all $i \neq j$. □

Fact 1.5. *A symmetric matrix $A \in \mathbb{R}^{n\times n}$ has no negative eigenvalues if and only if $x^T A x \geq 0$ for all $x \in \mathbb{R}^n$.*

Proof. First assume A has no negative eigenvalues so that $\lambda_i \geq 0$ for all i. Then:

$$x^T A x = x^T(\sum_{i=1}^{n} \lambda_i v_i v_i^T)x = \sum_{i=1}^{n} \lambda_i x^T v_i v_i^T x = \sum_{i=1}^{n} \lambda_i (x^T v_i)^2 \geq 0$$

For the other direction, assume $x^T A x \geq 0$ for all $x \in \mathbb{R}^n$. Then, A cannot have a negative eigenvalue since for a negative eigenvalue v, we would have $v^T A v = v^T(\lambda v) = \lambda \|v\|^2 < 0$. □

2 Semidefinite Programming

So: getting back to Max Cut, we want to design a separation oracle which given Y, either asserts that it is PSD or returns a violated constraint. By the above, we now know all we need to do for this is find the smallest eigenvalue, eigenvector pair λ_1, v_1. If $\lambda_1 \geq 0$, the matrix is PSD and we are finished. Otherwise, $\lambda_1 < 0$. But then $v_1^T Y v_1 = v_1^T(\lambda_1 v_1) < 0$, so this is a violated constraint, and we can continue. Finding the smallest eigenvalue, eigenvector pair can be done in polynomial time via, for example, the Cholesky Decomposition. So, we have a separation oracle.

The one catch is that there are some bit complexity issues here. To say we separate *all* inequalities $c^T Y c \geq 0$, we need to deal with some possibly pretty huge numbers in the vectors c. So, in the end, we can only solve this up to some additive precision ϵ in time polynomial in $\log(1/\epsilon)$. This leads us to the following semidefinite program:

$$\begin{aligned}
\max \quad & \sum_{\{u,v\}\in E} \frac{1}{2}(1 - y_{uv}) \\
\text{s.t.} \quad & (y_{uv})_{u,v\in V} \succeq 0 \\
& y_{vv} = 1 \qquad\qquad \forall v \in V
\end{aligned}$$

Now comes the intuition behind this PSD criterion:

Second Moments

We want a distribution over the signs $x \in \{-1,1\}^n$, and instead of just looking at the *first moments* of this distribution, i.e. $\mathbb{E}[x]$ (which is what an LP usually encodes), we are also looking at the *second moments*, $\mathbb{E}\left[xx^T\right]$. And what do we know about $\mathbb{E}\left[xx^T\right]$? It must be PSD, as for any distribution μ over $x \in \{-1,1\}^n$ we must have:

$$c^T \mathbb{E}\left[xx^T\right] c = c^T(\sum_{y\sim\mu} \mathbb{P}[y]\, yy^T)c = \sum_{y\sim\mu} \mathbb{P}[y]\,(c^T y)^2 \geq 0$$

Let's see how to round a solution given the covariance matrix $\mathbb{E}\left[xx^T\right]$. Note that the covariance matrix is technically the matrix $\mathbb{E}\left[(X - \mathbb{E}[X])(X - \mathbb{E}[X])^T\right]$, we will be dealing with vectors with $\mathbb{E}[X] = \mathbf{0}$, so this is the same thing.

2.1 Rounding

Let's continue with our usual relax-and-round framework. Given a solution $Y = (y_{uv})_{u,v \in V}$, how do we actually round it?

Lemma 2.1. *Let $r_1, \ldots, r_n$ be independent Gaussians with mean 0 and variance 1. Then, the covariance matrix of $x = Y^{1/2} r$ is Y.*

Proof.
$$\mathbb{E}\left[xx^T\right] = \mathbb{E}\left[Y^{1/2} r r^T Y^{1/2}\right] = Y^{1/2} \mathbb{E}\left[rr^T\right] Y^{1/2} = Y \qquad \square$$

Unfortunately, $x = Y^{1/2} r$ will not be in $\{-1, +1\}^n$. If we could somehow sample from μ, get covariance matrix Y, and ensure x was in $-1, 1$, we would get a 1-approximation.

Fact 2.2. *Let μ be a distribution over vectors in $\{-1, +1\}^n$ and suppose $\mathbb{E}\left[x_u x_v\right] = y_{uv}$ for all $u, v \in V$. Then, we obtain a 1-approximation.*

Proof. By linearity of expectation, the expected cost of our algorithm is $\sum_{\{u,v\} \in E} \frac{1}{2}(1 - \mathbb{E}\left[x_u x_v\right]) = \sum_{\{u,v\} \in E} \frac{1}{2}(1 - y_{uv})$ which is the objective function of our LP. $\square$

So, just like when rounding LPs, we must lose something when we get an integer distribution. What's the most natural way to round in this setting?

It turns out the simplest idea works here. After sampling from our distribution μ over $\mathbb{R}^n$ with covariance matrix Y, for each coordinate x_i, let $x_i = 1$ if $x_i \geq 0$ and -1 otherwise.

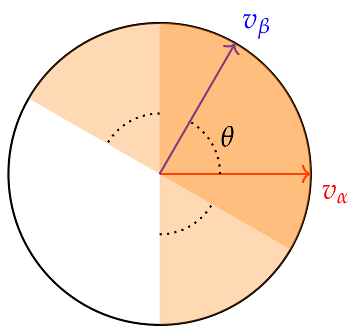

Figure 1: A visual proof that the regions in which the signs differ have total angle 2θ. In the solid yellow region, both signs will be positive, and in the white region, both will be negative. The remainder is the desired region with total angle 2θ.

Lemma 2.3 (Sheppard's Formula [She98]). *Let $\alpha, \beta \sim \mathcal{N}(0, 1)$ be two (one-dimensional) correlated Gaussians such that $\mathbb{E}\left[\alpha\beta\right] = \rho$. Then, $\mathbb{P}\left[sign(\alpha) \neq sign(\beta)\right] = \frac{\arccos(\rho)}{\pi}$.*

Proof. (α, β) is a multivariate Gaussian with covariance matrix $\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ and mean 0. This uniquely defines the distribution. A standard way to sample a multivariate Gaussian with covariance matrix C is to sample r where each $r_i \sim \mathcal{N}(0, 1)$ independently and then output $C^{1/2} r$. We have already noticed this produces the desired covariance matrix.

Now, let v_α be the first row of $C^{1/2}$ and v_β the second. Then, $\alpha = \langle v_\alpha, r\rangle$ and $\beta = \langle v_\beta, r\rangle$. α will be positive if the angle between v_α and r is in the range $[-\frac{\pi}{2}, \frac{\pi}{2}]$ and similarly for β. The angle of r is uniformly random because r is a Gaussian (this is the only time we use this). So, the signs will be different according to Fig. 1, with probability equal to to twice the angle between v_α and v_β divided by 2π. Now:

$$\rho = \langle v_\alpha, v_\beta\rangle = \|v_\alpha\|\|v_\beta\| \cos(\theta) = \cos(\theta),$$

where in the first equality we used that the covariance matrix is as above. This completes the proof. $\square$

We can now use this to finish the proof. We will use the following computational lemma, which we sketch by picture (see Fig. 2):

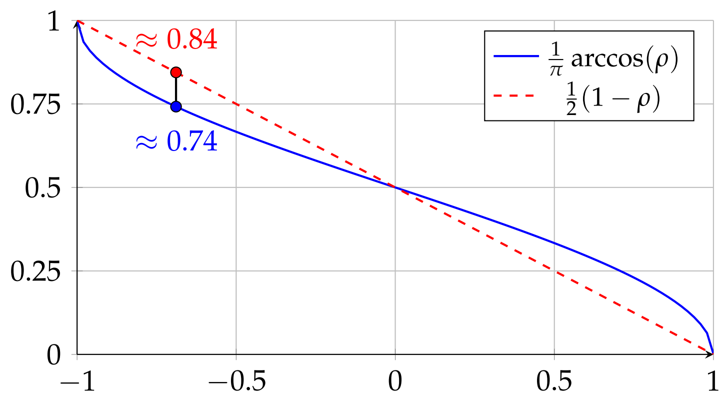

Figure 2: This can be verified mathematically, but one can check that the largest deviation occurs at approximately -0.689 and has a ratio of approximately 0.878.

Lemma 2.4. *For $\rho \in [-1, 1]$, we have*

$$\frac{\arccos(\rho)}{\pi} \geq 0.878 \cdot \frac{1}{2}(1-\rho)$$

But now we're done, as the expected number of edges cut is (by linearity of expectation):

$$\mathbb{E}\left[|\delta(S)|\right] = \sum_{\{u,v\}\in E} \frac{\arccos(y_{uv})}{\pi} \geq 0.878 \cdot \sum_{\{u,v\}\in E} \frac{1}{2}(1-y_{uv}) = 0.878$$

As mentioned, this is also the integrality gap and this ratio can be improved unless the Unique Games Conjecture is false [Kho+07].

References

[Kho+07] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. "Optimal Inapproximability Results for MAX-CUT and Other 2-Variable CSPs?" In: *SIAM Journal on Computing* 37.1 (2007), pp. 319–357. DOI: 10.1137/S0097539705447372. eprint: https://doi.org/10.1137/S0097539705447372 (cit. on p. 5).

[She98] W. F. Sheppard. "On the Application of the Theory of Error to Cases of Normal Distribution and Normal Correlation". In: *Philosophical Transactions of the Royal Society A* 192 (1898), pp. 101–167. DOI: 10.1098/RSTA.1899.0003 (cit. on p. 4).