LECTURE 4

MAX-CUT, SDPs, ...
Lecture 4: Max-Cut, SDPs

The Problem:

Input: $G = (V, E)$: graph on $n$ vertices, $m$ edges
Goal: Compute $S \subseteq V$ s.t. $|E(S, \overline{S})|$ is maximized.

We’ll simply call it max-cut $\leftarrow$ “normalized max-cut($G$).”

Fact: Computing Max-cut ($G$) is NP-hard.

Pf: Reduction from Max Independent Set.

Remark: You may recall that min-cut ($G$) can be computed in polynomial time. What a difference min vs max makes!

Approximation Algorithms for Max-Cut

• $\frac{1}{2}$-approximation is easy.

Lemma: Let $S$ be a uniformly random set of vertices of $V$. Equivalently, choose each $v \in V$ to be in $S$ w.p. $\frac{1}{2}$ independently of others.
Then, $\mathbb{E}\left[\frac{1}{S} \frac{|E(C_S, \overline{S})|}{m}\right] \geq \frac{1}{2}$

Since $\text{max-cut}(G) \leq 1$, this gives a $\frac{1}{2}$ approx.

"Surrogate for $\text{max-cut}(G) = m$".

Proof: $\mathbb{E} [\text{edge } \{u,v\} \text{ is cut}] = \frac{1}{2}$. Apply linearity of expectations.

don't need to use randomized algo.

Local Search: 1) Start from any cut $S$
2) If there's a $v$ such that moving $v$ to the other side of the cut improves cut size, do it.
3) Stop when there's no such $v$ & return the resulting cut.

Analysis: Exercise.
Question: Is there a $> \frac{1}{2}$ factor approx. algo for Max-Cut?

Set Cover: LPs helped. What about here?  

→ quadratic program

QP for Max-Cut

$$\max \frac{1}{4m} \sum_{e \in \{41N\}} (x_u - x_v)^2$$

s.t. $x_u \in \{\pm 1\}$ for all $u \in V$

Every $x = (x_u)_{u \in V}$ is a "\pm 1" indicator of a set of vertices.

the objective function computes the size of the cut defined by $x$. 
Clearly, QPs are NP-hard to solve. Can we relax & round them?

- LPs for Max-Cut

must "linearize" the objective:

$$
\frac{1}{2} \sum_{m \in E \subseteq \{u, v\}} y_{u, v}
$$

$$y_{u, v} \in [0, 1]$$

Clearly useless.

Can we add additional constraints?

no cut can "cut" all 3 edges.
For every \(|u, v, w| = \Delta \text{ in } G\),
\[ y_{uv} + y_{vw} + y_{uw} \leq 2. \]

Do they help?

**NOPE.**

**Integrality Gap:** Let \( 2K = G \).

Then clearly \( y_{1,2} = y_{2,3} = \ldots = y_{5,1} = 1 \) is a solution.

In fact, we know that for some \( c \in [0,1] \), we need at least \( 2^{n^c} \) constraints to improve on \( \frac{1}{2} \) approx.
Fact: [K, Meka, Raghavendra’18]
Beating $\frac{1}{2}$ for Max-Cut requires $2^n$ size & extended formulations for some constant $c > 0$.

Brief Intuition: There are graphs $G_1, G_2$
st. $\text{max-cut}(G_1) \sim 1$ (almost bipartite)
$\text{max-cut}(G_2) \sim \frac{1}{2}$ ("minimal")
but local neighborhood around every vertex in both $G_1$ & $G_2$ looks exactly the same. Both are $d$-regular trees.

LPS, local algorithms seem unable to distinguish between such pairs.
Can we hope to add some non-linear constraints on $Y_{uv}$ that provide more power?

**Idea:**

Objective: $\frac{1}{4m} \sum_{e=\{u,v\}} (x_u - x_v)^2$

$$= \frac{1}{4m} \sum_{e=\{u,v\}} 2 - 2x_u \cdot x_v$$

$$= \frac{1}{2} - \frac{1}{2m} \sum_{e=\{u,v\}} x_u \cdot x_v$$
Observation: 

\[ M = \begin{bmatrix} \cdot x_u^2 = 1 \\ y \cdot x_u \cdot x_v \end{bmatrix} = xx^T \geq 0 \]

nxn matrix with diag = 1

positive semidefinite

A new relaxation:

\[ \max \frac{1}{2} - \frac{1}{2m} \sum_{e \in \{u,v\}} X_{u,v} \]

s.t. \ diag(X) = 1 \rightarrow \ linear constraint

¤ "non linear constraint" \n
Relaxation because we forgot "rank1" constraint
Detour: Semidefinite Programs

A class of convex programs.

Linear objective $\max \sum c_i x_i$

subject to $x \in K$

Convex program

Convex subset of $\mathbb{R}^n$

Caution: Convex programming in general is NP-hard.

Convex $\neq$ easy

But some convex programs are "solvable" in polynomial time.
E.g. Linear Programs

\[ K = \text{intersection of linear inequalities} \]

\[ = \{ x \mid A \leq b \) \text{ bitsism} \]

Def (Semidefinite Programs)

An SDP in \( n \times n \) matrix valued variable \( X \) is a convex program where

\[ K = \{ X \geq 0 | \langle A_i, X \rangle_F \leq b_i \} \text{ for } 1 \leq i \leq m \]

\[ \sum_{j,k} A_{i}(j,k) \cdot X(j,k) = \text{Frobenius inner product} \]
SDPs can be solved approximately via ellipsoid method. We'll study this in more detail in the last two weeks of this course.

SDPs for Max-Cut

\[ G(V,E) \quad \text{max} \quad \frac{1}{2} - \frac{1}{2m} \sum_{u,v \in E} X_{u,v} \]

s.t. \( \text{diag}(X) = 1 \)

\[ X \succeq 0 \]
Rounding (Goemans-Williamson's 95)

Theorem: There's a poly time randomized algorithm that takes input $x: x \geq 0$ & $x_{ii} = 1 + i$ and outputs an $x \in \{\pm 1\}^n$ s.t. for every $1 \leq i, j \leq n$,

if $\left(\frac{1}{2} - \frac{1}{2} x_{uv}\right) \geq 1 - 3$,

then $\mathbb{E}_x \left[\frac{1}{2} - \frac{1}{2} x_{uv} - x_{uv}\right] \geq 1 - 2\sqrt{3}$

Further: $\mathbb{E}_x \left[\frac{1}{2} - \frac{1}{2} x_{uv}\right] \geq 0.878...$
**Corollary:** There's a (4/5) approx. algo for Max-Cut for \( C = 1-\varepsilon \) & \( S = 1-2\sqrt{\varepsilon} \) for every \( \varepsilon > 0 \). Further, the approx. ratio of this algo is \( \geq 0.878 \ldots \)

**Proof:** Let \( G \) be a graph and \( \text{SDP}(G) = C \) with optimal solution \( X \).

Then: \( \text{SDP}(G) \succeq \text{Max-Cut}(G) \).

Why? \( \text{SDP}(G) \) is a relaxation. \( [X = x^*x^*^T \text{ is \ "feasible"}] \)
On the other hand, theorem implies that we can find $x$ such that

$$E \left[ 1 - \frac{1}{2} x u \cdot x v \right] \geq 1 - 2\sqrt{3}$$

we take average of LHS over $\{u,v\} \in E$ of $G$.

then $E \left[ \text{cut}_G (x) \right] \geq 1 - 2\sqrt{3}$

Similar argument for approx ratio.
**BASIC FACTS** (may not prove in lecture)

Lemma (Cholesky Factorization)

For every $X \in \mathbb{R}^{n \times n}$, $X \succ 0$, there is a $Z \in \mathbb{R}^{n \times n}$ s.t.

$$X = ZZ^T$$

**Proof:**

$X = U \cdot \Lambda \cdot U^T$

**diagonal**

Eigenvalue

**matrix decomposition**

$X \succ 0 \iff \Lambda$ has non-negative diagonals.

Let $\Lambda^{1/2} =$ entrywise square root
Set $Z = (U \cdot \Lambda^{1/2})$. Then $X = ZZ^T$.

**Def. (Gaussian)**

Std. Gaussian distribution: $PDF(X) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

**Prop. (Rotation Invariance)**

If $H \in \mathbb{R}^{n \times n}$, orthogonal (i.e., $HH^T = H^TH = I$)

$g : \text{std. gaussian vector in } \mathbb{R}^n$. 

independent std. gaussians.
Then, Hg has same distr as g.

\[ \text{Pf: PDF of } g = \frac{1}{(\sqrt{2\pi})^n} e^{-\frac{\|x\|^2}{2}} \]

\[ \|x\|^2 = \|Hx\|^2 \text{ (l}_2 \text{ norm is not invariant).} \]

Corollary 1: Let \((g_1, g_2)\) be std. 2D gaussian vector. Then the point \((\frac{g_1}{\|g_1\|}, \frac{g_2}{\|g_2\|})\) is uniformly distributed on the unit circle.

\[ \text{Pf: Let } u = (u_1, u_2), \ v = (v_1, v_2) \text{ s.t. } \]

\[ u_1^2 + u_2^2 = v_1^2 + v_2^2 = 1. \]
Then there's a $H$, orthogonal s.t. $H u = v$. By rotation invariance PDF at $u = PDF$ at $v$. □

Corollary 2: Let $z_1, z_2 \in \mathbb{R}^n$ be unit vectors s.t. $\beta = \langle z_1, z_2 \rangle$

$g = (g_1, \ldots, g_n)$: n-D std. gaussian

Then $\begin{pmatrix} \langle z_1, g \rangle \\ \langle z_2, g \rangle \end{pmatrix} \sim \begin{pmatrix} g_1 \\ \sqrt{1-\beta^2} g_2 + \beta g_1 \end{pmatrix}$

Proof: There's an orthogonal matrix $H$ s.t. $HZ_1 = e_1$

$Hz_2 = \sqrt{1-\beta^2} e_2 + \beta e_1$
Proof of Theorem

Rounding:

1) Compute $Z : X = ZZ^T$

2) Generate $g = (g_1, \ldots, g_n)$ from std. gaussian dist.

3) For each $u$, set

$$x_u = \text{sign}(\langle g, z_i \rangle)$$
Analysis of Rounding

\( g = (g_1, \ldots, g_n) : \) std. gaussian vector

Fix \( u, v \in V \) then

\( \langle Xu, Xv \rangle \) has same dist as \( \text{sgn} \left( \langle g_i, z_n \rangle, \langle g_i, z_i \rangle \right) \)

We care about the random variable

\[
\frac{1}{2} - \frac{1}{2} xu \cdot xv = \left\{ \begin{array}{ll}
0 & \text{if } Xu = Xv \\
1 & \text{if } Xu \neq Xv.
\end{array} \right.
\]

Thus we are interested in the following elementary question:
**Question:** Let \( (g_u, g_v) \) be jointly distributed as a 2-D Gaussian with mean \( 0 \), \( \text{cov} = \begin{pmatrix} 1 & X_{uv} \\ X_{uv} & 1 \end{pmatrix} \)

What's the chance that \( X_u \neq X_v \)?

**Lemma (Sheppard's lemma):**

Let \( Z_u, Z_v \) be \( n \)-dim unit vectors such that \( \langle Z_u, Z_v \rangle = X_{uv} = -1 + \eta \)

Let \( \mathbf{g} = (g_1, \ldots, g_n) = \text{std. gaussian vector} \)

\[
\Pr[ X_u \neq X_v ] = 1 - \frac{\sqrt{2n}}{\pi} + O(n^{3/2})
\geq 1 - 2\sqrt{n} + \eta.
\]

**Aside:** If \( X \) is PSD, \( X_{uu} = X_{vv} = 1 \) then: \( |X_{uv}| \leq 1 \). To see why use \( w^T X w \geq 0 \) for \( w = e_u + e_v \) and \( w = e_u - e_v \) and rearrange.
Proof: "Reduce to 2d geometry:"

\[ Z = (z_1, \ldots, z_u, \ldots, z_v, \ldots, z_n) \]

\[ \text{rows of } Z \]

\[ g = (g_1, \ldots, g_n) : \text{std. gaussian} \]

Then:

\[
\begin{pmatrix}
\langle z_{u1} g \rangle \\
\langle z_{v1} g \rangle
\end{pmatrix}
\sim
\begin{pmatrix}
g_1 \\
(1+n)g_1 + \sqrt{2\pi-n^2}g_2
\end{pmatrix}
\]
$$Pr[ X_u \neq X_v ]$$

$$= Pr[ \text{Sign}(g_i) \neq \text{Sign}(E_1 + \eta)g_i + \sqrt{2\eta - \eta^2}g_2 ]$$

$$= \frac{2\Theta_{uv}}{2\pi} = \frac{\Theta_{uv}}{\pi}$$
Thus,
\[ P_r \left[ x_u \neq x_v \right] = \frac{\Theta_{UV}}{\pi} \]
\[ = \frac{\arccos(X_{UV})}{\pi} \]

Parameterize \( X_{UV} = -(1 - \eta) \cdot \arccos(C_{Cterr}) = \pi - \arccos(1 - \eta) \).

\[ \arccos(1 - \eta) = \sqrt{2\eta + \left( \frac{2\eta^{3/2}}{24} \right)} + O(\eta^2) \]

Plugging in:
\[ P_r \left[ x_u \neq x_v \right] = 1 - \frac{\sqrt{2\eta}}{\pi} + O(\eta^{3/2}) \]
Use calculus/mathematica to minimize
\[
\frac{\Theta_{uv}}{\pi \cos(\Theta_{uv})}
\]
minimizing \( \Theta_{uv} : -0.69 \)
min val value: 0.878.

Can we improve GW?

Yes!
1) For bounded degree graphs, can beat 0.878
2) There is a rounding that does better than GW in some regimes.

Can get a better \((C,s)\)-approx. curve. [O’Donnell-Wu].
Even better?

Next time: limitations of GW algo

Future: UGC and "optimality" of the above alg