Sublinear Algorithms

LECTURE 13

Last time

- Graph property testing (for dense graphs)
- Testing bipartiteness
- Started approximate Max-Cut

Today

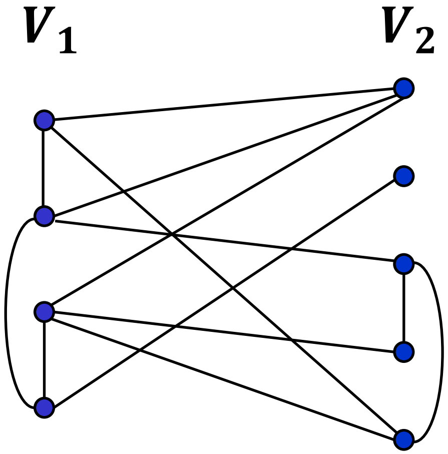
- Finish approximate Max-Cut
  [Goldreich Goldwasser Ron 98]

Project progress reports due Thursday after spring break
Sign up for project meetings

Max Cut in Dense Graphs

- Let $G = (V, E)$ be an undirected n-node graph.

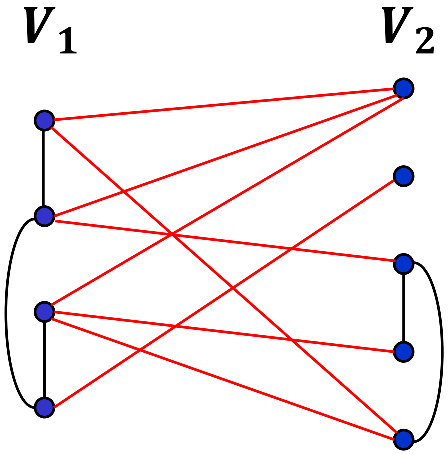
- Let (V_1, V_2) be a partition of V.
  $e(V_1, V_2)$ = set of edges crossing the cut



Max Cut in Dense Graphs

- Let $G = (V, E)$ be an undirected n-node graph.

- Let (V_1, V_2) be a partition of V.
  $e(V_1, V_2)$ = set of edges crossing the cut



- The edge density of the cut, denoted $\mu(V_1, V_2)$, is $\frac{|e(V_1,V_2)|}{n^2}$.

- The edge density of the largest cut in G is
$$\mu(G) = \max_{(V_1,V_2)} \mu(V_1, V_2)$$

Approximate Max-Cut Problem

[Goldreich Goldwasser Ron 98]

Input: parameter ε, access to an undirected graph $G = (V, E)$ represented by $n \times n$ adjacency matrix.

Goal 1: Output an estimate $\hat{\mu}$ such that:

$$\Pr[|\hat{\mu} - \mu(G)| \leq \varepsilon] \geq 2/3$$

- [GGR98]: poly$\left(\frac{1}{\varepsilon}\right)$ queries and $O(2^{poly\left(\frac{1}{\varepsilon}\right)})$ time

Goal 2: Output a partition (V_1, V_2) with edge density

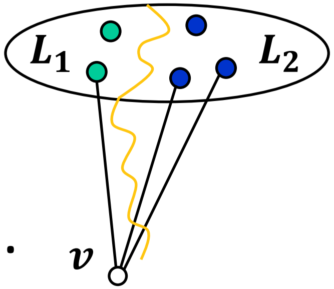
$$\mu(V_1, V_2) \geq \mu(G) - \varepsilon$$

with probability at least 2/3.

- [GGR98]: $O\left(2^{poly\left(\frac{1}{\varepsilon}\right)} + n \cdot poly\left(\frac{1}{\varepsilon}\right)\right)$ time

Greedy Partitioning

- Suppose we have a partition (L_1, L_2) of $L \subset V$.
- In which part should we place a new node v to maximize edge density?



- Let $\Gamma(v, U)$ be the number of neighbors of v in U.

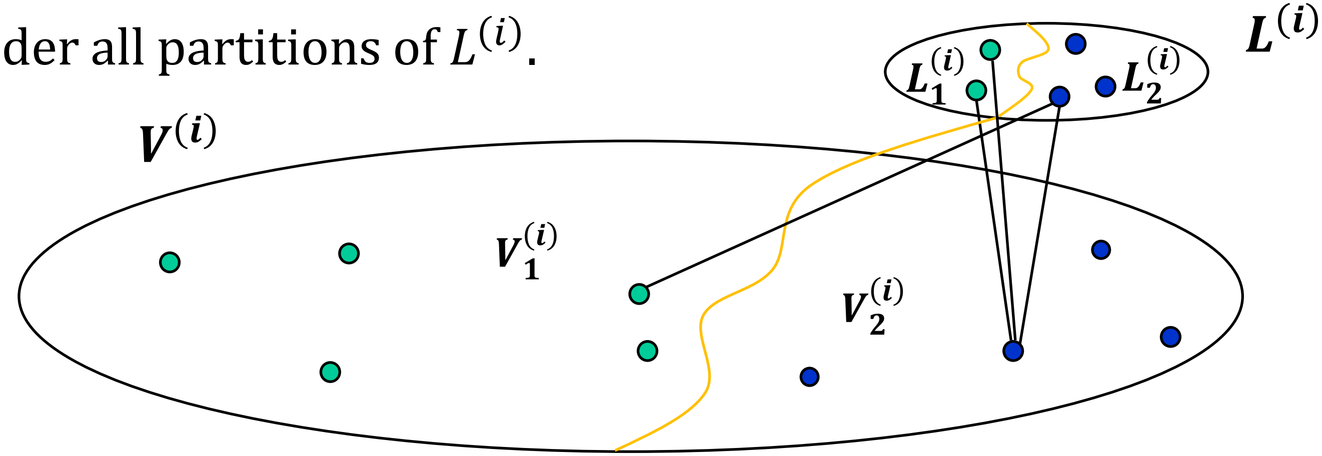
- Greedy: If $\Gamma(v, L_1) \leq \Gamma(v, L_2)$, place v in L_1; otherwise, place it in L_2.

Main Idea

- Partition V into sets $V^{(i)}$ of (almost) equal size.

Assume they are of equal size.

- For each set $V^{(i)}$, sample a learning set $L^{(i)}$ from the vertices not in $V^{(i)}$.
- Consider all partitions of $L^{(i)}$.



A partition of $L^{(i)}$ induces a partition of $V^{(i)}$ via the greedy rule

A partition sequence $\pi(L) = \left(\left(L_1^{(1)}, L_2^{(1)}\right), \ldots, \left(L_1^{(t)}, L_2^{(t)}\right)\right)$ induces a partition of V

- Consider all such partitions of V and pick the best.

Preliminary Max-Cut Approximation Algorithm

```
Algorithm (Input: $\varepsilon, n$; query access to adjacency matrix of $G=(V,E)$)
1. Partition $V$ into $t = \frac{4}{\varepsilon}$ sets $V^{(1)}, V^{(2)}, \dots, V^{(t)}$ of (almost) equal size.
2. For each $i \in [t]$, select a set $L^{(i)}$ of size $\ell = \frac{320}{\varepsilon^2} \cdot \log\frac{1}{\varepsilon}$ u.i.r. from $V \backslash V^{(i)}$.
   Let $L = \left(L^{(1)}, L^{(2)}, \dots, L^{(t)}\right)$.
3. For each partition sequence $\pi(L) = \left(\left(L_1^{(1)}, L_2^{(1)}\right), \dots, \left(L_1^{(t)}, L_2^{(t)}\right)\right)$
4.     For each $i \in [t]$
5.         Partition $V^{(i)}$ into $\left(V_1^{(i)}, V_2^{(i)}\right)$ using the greedy rule:
            place $v$ in $V_1^{(i)}$ iff $\Gamma\left(v, L_1^{(i)}\right) \leq \Gamma\left(v, L_2^{(i)}\right)$.
6.     Let $V_1^{\pi} = \cup_i V_1^{(i)}$ and $V_2^{\pi} = \cup_i V_2^{(i)}$; calculate $\mu(V_1^{\pi}, V_2^{\pi})$.
7. Output the cut $(V_1^{\pi}, V_2^{\pi})$ with the largest density.
```

- Number of partition sequences: $\left(2^{\ell}\right)^t = 2^{poly\left(\frac{1}{\varepsilon}\right)}$
- Running time: $n^2 \cdot 2^{poly\left(\frac{1}{\varepsilon}\right)}$

$O(n^2)$ time for calculating each density

Correctness of Max-Cut Approximation

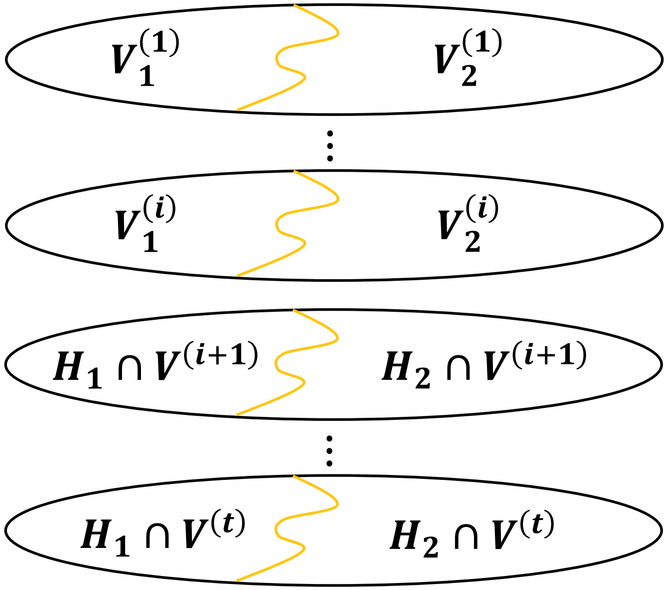
Correctness Theorem

Let (H_1, H_2) be a partition of V.
W. p. $\geq 5/6$ over the choice of L, some partition sequence $\pi(L)$ induces a partition (V_1^π, V_2^π) of V s. t. $\mu(V_1^\pi, V_2^\pi) \geq \mu(H_1, H_2) - 3\varepsilon/4$

Think: (H_1, H_2) is a max-cut

Main Proof Idea: Use a hybrid argument.

- $\left(H_1^{(0)}, H_2^{(0)}\right) = (H_1, H_2)$
- For all $i \in [t]$, partition $\left(H_1^{(i)}, H_2^{(i)}\right)$ is obtained from $\left(H_1^{(i-1)}, H_2^{(i-1)}\right)$ by repartitioning $V^{(i)}$ into into $\left(V_1^{(i)}, V_2^{(i)}\right)$, the best out of all partitions induced by a partition of $L^{(i)}$.
- We will show that when we go from one hybrid to the next, the density does not drop too much (w.h.p.)



i-th hybrid partition $\left(H_1^{(i)}, H_2^{(i)}\right)$

Correctness of Max-Cut Approximation

Correctness Theorem

Let (H_1, H_2) be a partition of V.

W. p. $\geq 5/6$ over the choice of L, some partition sequence $\pi(L)$ induces a partition (V_1^π, V_2^π) of V s. t. $\mu(V_1^\pi, V_2^\pi) \geq \mu(H_1, H_2) - 3\varepsilon/4$

Proof: Consider $i \in [t]$ and fix learning sets $L^{(1)}, \ldots, L^{(i-1)}$.

- Let B_i be the event that $\mu\left(H_1^{(i)}, H_2^{(i)}\right) < \mu\left(H_1^{(i-1)}, H_2^{(i-1)}\right) - \frac{3\varepsilon}{4t}$

Main Lemma

$\Pr[B_i] \leq \frac{1}{6t}$, where the probability is taken over the choice of $L^{(i)}$.

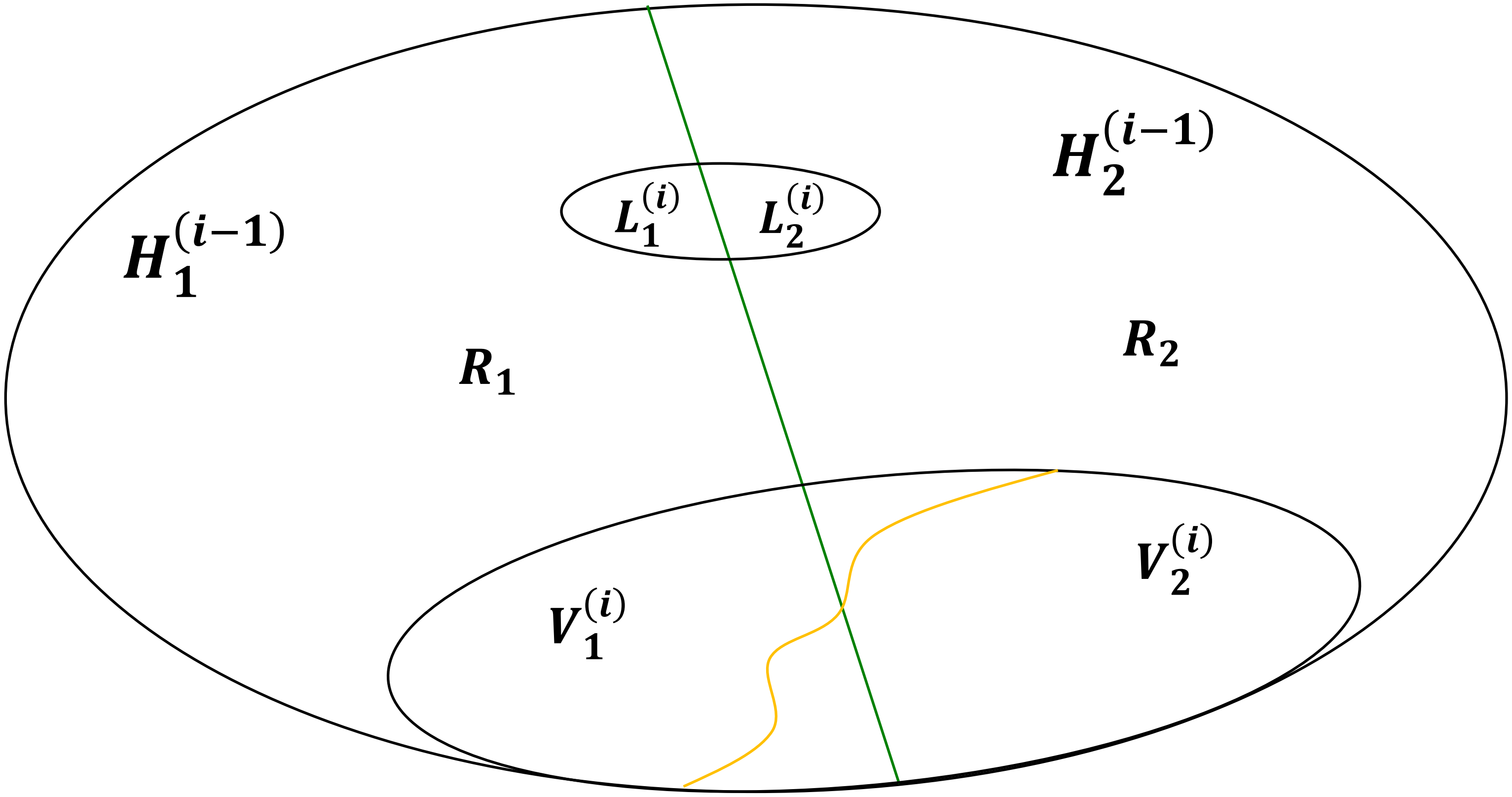
- Then, by a union bound,

$$\Pr\left[\bigcup_i B_i\right] \leq t \cdot \frac{1}{6t} = \frac{1}{6}$$

Big Picture

When we go from hybrid $i-1$ to hybrid i, only nodes in $V^{(i)}$ get repartitioned.

- Let $R_1 = V\backslash \mathrm{V}^{(i)} \cap H_1^{(i-1)}$ and $R_2 = V\backslash \mathrm{V}^{(i)} \cap H_2^{(i-1)}$
- Let $L_1^{(i)} = L^{(i)} \cap H_1^{(i-1)}$ and $L_2^{(i)} = L^{(i)} \cap H_2^{(i-1)}$



Proof of Main Lemma

- A node $v \in V^{(i)}$ is good w.r.t. $\left(L_1^{(i)}, L_2^{(i)}\right)$ if $\left|\frac{\Gamma\left(v, L_j^{(i)}\right)}{\ell} - \frac{\Gamma(v, R_j)}{n}\right| \le \frac{\varepsilon}{8}\ \forall j \in \{1,2\}$
- Learning set $L^{(i)}$ is good if $\le \frac{\varepsilon}{4}\left|V^{(i)}\right|$ nodes in $V^{(i)}$ are bad w.r.t. $\left(L_1^{(i)}, L_2^{(i)}\right)$

Claim

Fix $i \in [t]$. The probability that $L^{(i)}$ is bad is at most $\frac{1}{6t}$.

- A node $v \in V^{(i)}$ is balanced w.r.t. (R_1, R_2) if $\left|\frac{\Gamma(v, R_1)}{n} - \frac{\Gamma(v, R_2)}{n}\right| \le \frac{\varepsilon}{4}$

Observation

All unbalanced nodes that are good w.r.t. $\left(L_1^{(i)}, L_2^{(i)}\right)$ are placed correctly.

Proof: Suppose w.l.o.g. that $\Gamma(v, R_1) < \Gamma(v, R_2)$ for a good unbalanced node v

$$\frac{\varepsilon}{4} \overset{v \text{ is unbalanced}}{<} \frac{\Gamma(v, R_2)}{n} - \frac{\Gamma(v, R_1)}{n} \overset{v \text{ is good}}{\le} \left(\frac{\Gamma\left(v, L_2^{(i)}\right)}{\ell} + \frac{\varepsilon}{8}\right) - \left(\frac{\Gamma\left(v, L_1^{(i)}\right)}{\ell} - \frac{\varepsilon}{8}\right)$$

So, $\Gamma\left(v, L_1^{(i)}\right) < \Gamma\left(v, L_2^{(i)}\right)$, and v is placed correctly.

Density Loss from Repartitioning $V^{(i)}$

when $\left(L_1^{(i)}, L_2^{(i)}\right)$ is good

Type of cut-edges	Number of edges lost
Incident to good unbalanced nodes	
Incident to bad unbalanced nodes	
Incident to balanced nodes	
Between nodes of $V^{(i)}$	

Total: $\frac{3\varepsilon}{4t} \cdot n^2$

- Recall: B_i is the event that $\mu\left(H_1^{(i)}, H_2^{(i)}\right) < \mu\left(H_1^{(i-1)}, H_2^{(i-1)}\right) - \frac{3\varepsilon}{4t}$
- Event B_i can occur only when $\left(L_1^{(i)}, L_2^{(i)}\right)$ is bad.
- It remains to show that $\left(L_1^{(i)}, L_2^{(i)}\right)$ is bad with probability at most $\frac{1}{6t}$.

Probability of a Bad Learning Set

- A node $v \in V^{(i)}$ is good w.r.t. $\left(L_1^{(i)}, L_2^{(i)}\right)$ if $\left|\frac{\Gamma\left(v,\, L_j^{(i)}\right)}{\ell} - \frac{\Gamma(v,\, R_j)}{n}\right| \le \frac{\varepsilon}{8}\ \forall j \in \{1,2\}$
- Learning set $L^{(i)}$ is good if $\le \frac{\varepsilon}{4}\left|V^{(i)}\right|$ nodes in $V^{(i)}$ are bad w.r.t. $\left(L_1^{(i)}, L_2^{(i)}\right)$

Claim. Fix $i \in [t]$. The probability that $L^{(i)}$ is bad is at most $\frac{1}{6t}$.

Proof: Let $L^{(i)} = \{v_1, \dots, v_\ell\}$. Recall that it is chosen u.i.r. from $V \backslash \mathrm{V}^{(i)}$

- Fix $v \in V^{(i)}$. For all $j \in \{0,1\}$,

$$\text{let} \quad X_j^k = \begin{cases} 1, & \text{if } v_k \text{ is a neigbor of } v \text{ in } R_j \\ 0, & \text{otherwise} \end{cases}$$

Then $X_j = \frac{1}{\ell}\sum_{k\in[\ell]} X_j^k = \frac{\Gamma\left(v,\, L_j^{(i)}\right)}{\ell}$ and $\mathbb{E}[X_j] = \frac{1}{\ell}\sum_{k\in[\ell]} \mathbb{E}[X_j^k] = \frac{\Gamma(v,\, R_j)}{n}$

$$\Pr[v \text{ is bad}] \le \Pr\left[|X_1 - \mathbb{E}[X_1]| > \frac{\varepsilon}{8} \quad \text{or} \quad |X_2 - \mathbb{E}[X_2]| > \frac{\varepsilon}{8}\right]$$

by union bound and symmetry

$$\le 2\Pr\left[|X_1 - \mathbb{E}[X_1]| > \frac{\varepsilon}{8}\right]$$

by Hoeffding

$$\le 2 \cdot 2\exp\left(-2\left(\frac{\varepsilon}{8}\right)^2 \ell\right) = 4\exp\left(-\frac{\varepsilon^2 \ell}{32}\right)$$

$\ell = \frac{320}{\varepsilon^2}\ln\frac{1}{\varepsilon}$

$t = \frac{4}{\varepsilon}$

$$= 4\varepsilon^{10} \le \frac{\varepsilon}{4}\cdot\frac{1}{6t}$$

when ε is sufficently small

by Markov

$$\Pr\left[> \frac{\varepsilon}{4} \text{ fraction of nodes in } V^{(i)} \text{ are bad}\right] \le \frac{1}{6t}$$

Improved Max-Cut Approximation Algorithm

```
Algorithm (Input: $\varepsilon, n$; query access to adjacency matrix of $G=(V,E)$)
1. Partition $V$ into $t = 4/\varepsilon$ sets $V^{(1)}, V^{(2)}, \dots, V^{(t)}$ of (almost) equal size.
2. For each $i \in [t]$, selelect a set $L^{(i)}$ of size $\ell = \frac{320}{\varepsilon^2} \cdot \log\frac{1}{\varepsilon}$ u.i.r. from
   $V \backslash \mathrm{V}^{(i)}$.    Let $L = \left(L^{(1)}, L^{(2)}, \dots, L^{(t)}\right)$.
3. Select u.i.r. $S$ of size $m = \frac{64 \cdot t\ell}{\varepsilon^2}$
4. For each partition sequence $\pi(L) = \left(\left(L_1^{(1)}, L_2^{(1)}\right), \dots, \left(L_1^{(t)}, L_2^{(t)}\right)\right)$
5.     For each $i \in [t]$
6.         Partition $S^{(i)}$ into $\left(S_1^{(i)}, S_2^{(i)}\right)$ using the greedy rule:
             add $v$ to $S_1^{(i)}$ iff $\Gamma(v, L_1) \leq \Gamma(v, L_2)$.
7.     Let $S_1^{\pi} = \cup_i S_1^{(i)}$ and $S_2^{\pi} = \cup_i S_2^{(i)}$; calculate
           $\mu'(S_1^{\pi}, S_2^{\pi}) = \frac{|\{k : \{s_{2k-1}, s_{2k}\} \in e(S_1^{\pi}, S_2^{\pi})\}|}{m/2}$
8. Output $\max_{\pi} \mu'(S_1^{\pi}, S_2^{\pi})$
```

- We can also out put the cut of V induced by π with max μ'