Lecture 2:

- Estimate MST weight
- Estimate average degree

Recall from last time:

- Can estimate # connected components of degree $\leq d$ graph to within $\pm \varepsilon n$ in time $\underbrace{O(d/\varepsilon^4)}_{\text{no dependence on } n}$ with prob $\geq 3/4$

- HW0 + above $\Rightarrow$ $\pm \varepsilon n$ estimate with prob $\geq 1-\beta$ in time $O\left(d/\text{poly}(\varepsilon) \cdot \log\frac{1}{\beta}\right)$

Approximate Min Spanning Tree (MST)

Input (1) $G = (V, E)$ adjacency list representation

$n = |V|$

max degree d

each edge has weigt

$w_{uv} \in \{1, .., w\} \cup \{\infty\}$

i.e. $w_{uv} \notin E$

Connected

(2) ε

Output let $M = \min\limits_{T \text{ spans } G} \{w(T)\}$

tree touches every node

$w(T) = \sum_{(i,j) \in T} w_{ij}$

output $\hat{M}$ s.t. $(1-\varepsilon) \cdot M \le \hat{M} \le (1+\varepsilon) \cdot M$

assumption on wts $\Rightarrow n-1 \le w(T) \le w(n-1)$

A different characterization of MST:

$$E^{(i)} = \{(u,v) \mid w_{uv} \in \{1..i\}\}$$

$$G^{(i)} = (V, E^{(i)})$$

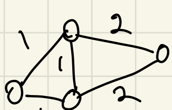
$$C^{(i)} = \# \text{ conn comp of } G^{(i)}$$

Some examples:

1) $w=1$ all edges have wt 1

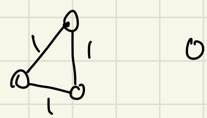
$M = n-1$

2) $w=2$ $w_{ij} \in \{1,2\}$ (or ∞)



G

$C = 1$



$G^{(1)}$

$C^{(1)} = 2$

Kruskal's Alg:
- sort edges, consider them from smallest to largest
- add next edge if doesn't create cycle

$E^{(i)} = \{(u,v) \mid w_{u,v} \in \{1..i\}\}$

$G^{(i)} = (V, E^{(i)})$

$C^{(i)} = \#$ conn comp of $G^{(i)}$

Kruskal's idea:

Use as many wt 1 edges as you can

then only need wt 2 edges to connect components

$\Rightarrow$ need $C^{(1)} - 1$ wt 2 edges

Total MST wt:

$$M = 1 \cdot (\# \text{wt } 1 \text{ edges}) + 2 \cdot (\# \text{wt } 2 \text{ edges})$$
$$= (n-1) + \# \text{ wt } 2 \text{ edges}$$
$$= (n-1) + C^{(1)} - 1 = n - 2 + C^{(1)}$$

Claim $M = n - w + \sum_{i=1}^{w-1} C^{(i)}$

Pf $\alpha_i \equiv \#$ edges of wt i in any MST of G

Kruskal's alg $\Rightarrow$ all MST's have same values of α_i

$$\sum_{i \geq \ell} \alpha_i = C^{(\ell)} - 1$$ # conn comp of $G^{(\ell)}$ -1

where $C^{(0)} = n$ since no edges in $G^{(0)}$

$E^{(i)} = \{ (u,v) \mid w_{u,v} \in \{1..i\} \}$

$G^{(i)} = (V, E^{(i)})$

$C^{(i)}$ = # conn comp of $G^{(i)}$

$\underline{\text{Claim}}$ $M = n - w + \sum_{i=1}^{w-1} C^{(i)}$

$$M = \sum_{i=1}^{w} i \cdot \alpha_i$$

$$= \sum_{i=1}^{w} \alpha_i + \sum_{i=2}^{w} \alpha_i + \sum_{i=3}^{w} \alpha_i + \dots + \sum_{i=w}^{w} \alpha_i$$

$$= (n-1) + (C^{(1)} - 1) + (C^{(2)} - 1) + \dots + (C^{(w-1)} - 1)$$

$$= n - w + \sum_{i=1}^{w-1} C^{(i)}$$

$\blacksquare$

$E^{(i)} = \{(u,v) \mid w_{u,v} \in \{1..i\}\}$

$G^{(i)} = (V, E^{(i)})$

$C^{(i)} = \#$ conn comp of $G^{(i)}$

Approximation algorithm:

Claim $M = n - w + \sum_{i=1}^{w-1} C^{(i)}$

For $i = 1$ to $w-1$

$\hat{C}^{(i)} \leftarrow$ approx # conn comp of $G^{(i)}$ to within $\frac{\varepsilon}{2w} \cdot n$ additive error & confidence parameter β

let $\varepsilon' = \frac{\varepsilon}{2w}$

Output $\hat{M} = n - w + \sum_{i=1}^{w-1} \hat{C}^{(i)}$

Can use G + $\boxed{\text{ignore}}$ edges of wt $> i$

Runtime:

$\tilde{O}\left(\frac{d}{(\varepsilon')^4}\right) = \tilde{O}\left(\frac{d\, w^4}{\varepsilon^4}\right) \cdot \log\frac{1}{\beta}$ for each call to approx conn comp

Total $\tilde{O}\left(\frac{d\, w^5}{\varepsilon^4} \cdot \log\frac{1}{\beta}\right)$ $\beta \sim \Theta\left(\frac{1}{w}\right)$

Can improve to $O\left(\frac{dw}{\varepsilon^2} \log\frac{dw}{\varepsilon}\right)$

need $\Omega(dw/\varepsilon^2)$

no dependence on n

$E^{(i)} = \{ (u,v) \mid w_{u,v} \in \{1..i\} \}$

$G^{(i)} = (V, E^{(i)})$

$C^{(i)} = \#$ conn comp of $G^{(i)}$

Approximation algorithm:

Claim $M = n - w + \sum_{i=1}^{w-1} C^{(i)}$

For $i = 1$ to $w-1$

$\hat{C}^{(i)} \leftarrow$ approx # conn comp of $G^{(i)}$ to within

$\frac{\varepsilon}{2w} \cdot n$ additive error

let $\varepsilon' = \frac{\varepsilon}{2w}$

Output $\hat{M} = n - w + \sum_{i=1}^{w-1} \hat{C}^{(i)}$

Approximation guarantee:

- Call approx # conn comp with failure prob $\beta < \frac{1}{4w}$

  runtime additional $\log \frac{1}{\beta}$ mult factor (HW 0) $(\times \log 4w)$

- Pr [all calls to approx #C.C. give output that is ε' additive approx] $\geq 1 - \frac{w}{4w} \geq 3/4$

- If $\Uparrow$ happens: $|M - \hat{M}| \leq \cancel{w} \cdot \frac{\varepsilon n}{2\cancel{w}} = \frac{\varepsilon n}{2}$ additive error

since $M \geq n-1 \geq \frac{n}{2}$ $\quad |M - \hat{M}| \leq \varepsilon M$. mult error

Which edges are on MST?

What if w is big?

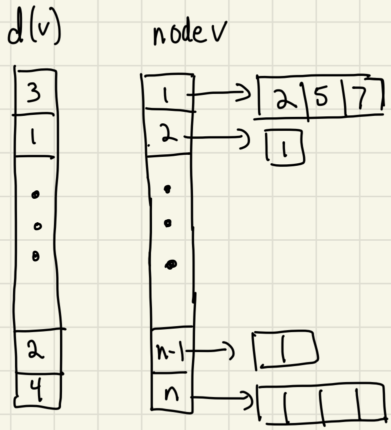
Estimating the average degree of a graph

def Average degree $\bar{d} = \dfrac{\sum_{u \in V} d(u)}{n}$

Assume: G simple (no parallel edges, self-loops)

$\Omega(n)$ edges (not "ultra-sparse")

Representation via adj list + degrees:



- degree queries: on v return $d(v)$
- neighbor queries: on (v,j) return j^{th} nbr of v

Naive sampling:

Pick $O(??)$ sample nodes $V_1 \cdots V_s$

output ave degree of sample:

$$\frac{1}{s} \sum_i d(v_i)$$

Straightforward Chernoff/Hoeffding needs $\Omega(n)$ samples

lower bound?

$d(v) = (0\ 0\ 0\ 0\ \ n-1\ \ 0\ 0\ 0\ 0)$ need $\Omega(n)$ samples to find "needle in haystack"

not possible

$(n-1, 1, 1, 1, \ldots, 1)$ is possible

Some lower bounds:

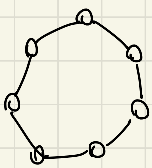
"ultrasparse" case:

graph with 0 edges vs. graph with 1 edge

ave deg = 0 | ave deg $= \frac{2}{n}$

need $\Omega(n)$ to distinguish

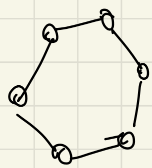
ave deg ≥ 2:



n-cycle

ave deg = 2

vs.



$n - c \cdot n^{1/2}$ cycle

$c \cdot n^{1/2}$ clique

ave deg $= 2 + c^2$

need $\Omega(\sqrt{n})$ queries to distinguish

Algorithm idea:

group nodes of similar degrees

estimate average w/in each group

why does this help?

recall Chernoff:

$X_1 \cdots X_r$ iid $X_i \in [0,1]$

$S = \sum_{i=1}^{r} X_i \quad p = E[X_i] = E[S]/r$

Then $\Pr\left[\left| \frac{S}{r} - p \right| \geq \delta p \right] \leq e^{-\Omega(rp\delta^2)}$

need to pick r to be $\Omega\left(\frac{1}{p\,\delta^2}\right)$

assume δ is $\Theta(1)$

but X_i needs to be in $[0,1]$

what if let $X_i = \frac{\deg(i)}{n}$

then p can be as small as $\frac{1}{n}$

$\Rightarrow \quad r = \Omega(n)$

but if $b \le \deg(u) \le (1+\varepsilon)b$

Can set $x_i \leftarrow \dfrac{\deg(u)}{(1+\varepsilon)b}$

$$\frac{1}{1+\varepsilon} \le x_i \le 1$$

so $\underbrace{p}_{\text{for group}} \ge \dfrac{1}{1+\varepsilon} \Rightarrow r$ can be $O(1/\varepsilon)$

Bucketing:

set parameters $\beta = \dfrac{\varepsilon}{c}$

$t = O\left(\dfrac{\log n}{\varepsilon}\right)$ # buckets

$$B_i = \{ v \mid (1+\beta)^{i-1} \le d(v) \le (1+\beta)^i \}$$

for $i \in \{0 \ldots t-1\}$

total degree of nodes in B_i:

$$(1+\beta)^{i-1} |B_i| \le d_{B_i} \le (1+\beta)^i |B_i|$$

total degree of graph: $\sum_i (1+\beta)^{i-1} |B_i| \le d_{total} \le \sum_i (1+\beta)^i |B_i|$

$$\sum_i (1+\beta)^{i-1} |B_i| \le d_{total} \le \sum_i (1+\beta)^i |B_i|$$

Recall:

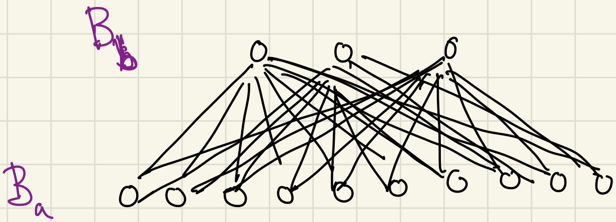
$$B_i = \{ v \mid (1+\beta)^{i-1} < d(v) \le (1+\beta)^i \}$$

First idea for algorithm:

- Take sample S of nodes
- $S_i \leftarrow S \cap B_i$
- estimate $|B_i|$:

$$\rho_i \leftarrow \frac{|S_i|}{|S|}$$

- Output $\sum_i \rho_i (1+\beta)^{i-1}$



$a \leftarrow i$ s.t. $(1+\beta)^{i-1} \le 3 \le (1+\beta)^{i}$

$b \leftarrow i$ s.t. $(1+\beta)^{i-1} \le n-3 \le (1+\beta)^{i}$