Lecture 3:

- Estimate average degree
    - recap
    - 2-approximation
    - $1+\varepsilon$ - approximation

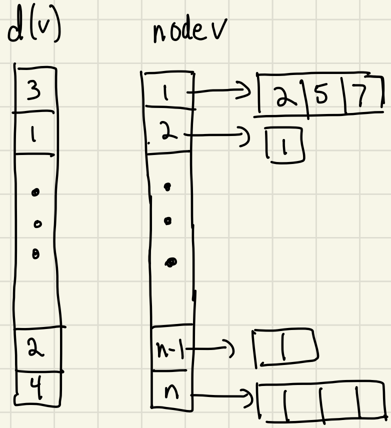
Estimating the average degree of a graph

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

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?

$d(1)$	$d(2)$	...				$d(n)$
0	0	0	$n-1$	0	0	0

need $\Omega(n)$ samples to find "needle in haystack"

not a possible degree sequence!!

$n-1$	1	1	1	1	1	1

is possible

Some lower bounds:

"ultrasparse" case:

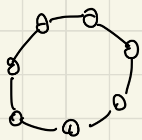
0 edges vs. 1 edge

need $\Omega(n)$ queries to distinguish

$\Rightarrow$ multiplicative approx needs $\Omega(n)$

ave deg ≥ 2:

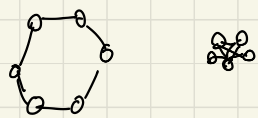
n - cycle $\bar{d} = 2$



vs.

note: c can be any const e.g. 1000

$\left\{ \begin{array}{l} n - c\cdot\sqrt{n} \text{ cycle} \\ + \ c\sqrt{n}\text{ - clique} \end{array} \right.$ $\bar{d} \approx 2 + c^2$



need $\Omega(n^{1/2})$ queries to find clique node

$$\boxed{\Rightarrow \text{need } \Omega(n^{1/2}) \text{ queries for constant mult approx !!!}}$$

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

$\Rightarrow$ r needs to be

$\Omega\left(\frac{1}{p\delta^2} \right)$ so p very small is not good!

let's assume δ is a constant

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

so if $X_i \leftarrow \frac{\deg(i)}{n}$

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

$\Rightarrow$ r needs to be $\Omega(1/p) = \Omega(n)$

but if $b \leq \deg(i) \leq (1+\varepsilon)b$

can set $X_i \leftarrow \frac{\deg(i)}{(1+\varepsilon)b}$

p is at least a constant

$\Rightarrow$ then $p \geq \frac{1}{1+\varepsilon}$

$\Rightarrow$ r needs to be only $\Omega(1)$. Much better!!!

\+ each group has bnded variance
− doesn't work for arbitrary #'s
why here?

Bcketing:

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

$t = O(\log n / \varepsilon)$ #buckets

$(1+\beta)^t > n$ when $t \geq \frac{\log n}{\log(1+\beta)}$

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

for $i \in \{0, \dots, (t-1)\}$

(can add bucket for deg 0 nodes
or
* assume none)

note: total degree of nodes in B_i

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

total degree of graph:

$$\sum_{i=1}^{t} (1+\beta)^{i-1} |B_i| \leq d_{total} \leq \sum_{i=1}^{t} (1+\beta)^{i} |B_i|$$

First idea for algorithm:

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

- Take sample S of nodes how many?
- $S_i \leftarrow S \cap B_i$ ← use degree queries to partition

define $\sigma_j^{(i)} = \begin{cases} 1 & \text{if sample } j \text{ falls in bucket } i \\ 0 & \text{o.w.} \end{cases}$

- estimate $|B_i|$:

$$\rho_i \leftarrow \frac{|S_i|}{|S|} \leftarrow E[\rho_i] = E\left[\frac{|S_i|}{|S|}\right] = \frac{E\left[\sum_{j=1}^{|S|} \sigma_j^{(i)}\right]}{|S|}$$

$$= \frac{\cancel{|S|} \cdot \frac{|B_i|}{n}}{\cancel{|S|}} = \frac{|B_i|}{n}$$

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

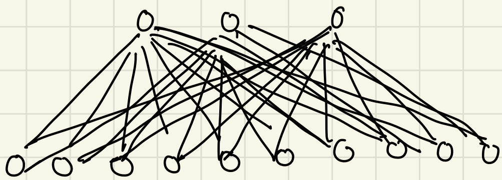
undercount by defn.

Problem: i s.t. $|S_i|$ is small

$\Rightarrow \rho_i$ is a bad approx

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

example:



3 nodes each deg $n-3$

$n-3$ nodes each deg 3

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

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

$\forall c \ne a, b \quad |B_c| = 0$

$|B_a| = n-3$

$|B_b| = 3$

both contribute $(n-3)\cdot 3$ edges

B_a contributes $(n-3)\cdot 3$ edges

B_b contributes $3\cdot(n-3)$ edges

not seen in sample of size $o(n)$

Next idea: use "0" for small buckets

(helps "variance")

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

$t = \#$ buckets

Old algorithm:

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

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

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

New algorithm:

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

for all i

if $|S_i| \geq \sqrt{\frac{\varepsilon}{n}} \cdot \frac{|S|}{C \cdot t}$ — i "big"

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

else $\rho_i \leftarrow 0$ — i "small"

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

- how big is S?
- why $\sqrt{\frac{\varepsilon}{n}} \cdot \frac{|S|}{C \cdot t}$?

one of these — comes from $t = O\left(\frac{\log n}{\varepsilon}\right)$

let $|S| = \Theta\left(\sqrt{n} \cdot \text{poly} \log n \cdot \text{poly} \frac{1}{\varepsilon}\right)$

$\Rightarrow$ big $|S_i| \geq \sqrt{\frac{\varepsilon}{n}} \cdot \frac{|S|}{C \cdot t} \geq \Omega\left(\text{polylog}\, n \times \text{poly}\, 1/\varepsilon\right)$

$\Rightarrow$ by union bnd + Chernoff bnd

assume this

$$\forall i \text{ big} \quad (1-\gamma) \frac{|B_i|}{n} \leq \rho_i \leq (1+\gamma) \frac{|B_i|}{n}$$

Why these settings of S? (ignore dependence on ε for now)

* each bucket that has at least $\approx \frac{1}{\sqrt{n}}$ fraction of <u>nodes</u> should have enough samples to be able to estimate the fraction.

* why $\approx \frac{1}{\sqrt{n}}$?
  - we will want to argue that "small" buckets represent a very small fraction of the <u>edges</u> so it is ok to zero them out
  - remember the clique lower bound example? if we set the "small" threshold to bigger than $\frac{1}{\sqrt{n}}$ we might miss lots of edges (e.g. a clique on $\sqrt{n}$ nodes will have $\Theta(n)$ edges & shouldn't be missed, but represents only $\frac{\sqrt{n}}{n} = \frac{1}{\sqrt{n}}$ fraction of nodes)

- why is $\frac{1}{\sqrt{n}}$ small enough?
  See later!

* what is "enough" samples for each bucket?

- we will need to argue that we are getting good estimates of $\frac{|B_i|}{n}$ for each big bucket

Chernoff bound

union bound over $\log n$ buckets

so need prob of having bad estimate "δ" set to $\leq < \frac{1}{\log n}$ per bucket

Chernoff will also depend on accuracy parameter $\beta = \frac{\varepsilon}{c}$

So if we set $S \approx \sqrt{n} \cdot \text{poly}\left(\frac{1}{\varepsilon}\right) \text{poly}(\log n)$

we should be more than ok

to get buckets with $\frac{1}{\sqrt{n}}$ fraction of nodes

this comes in everywhere

to satisfy Chernoff + union bnds

Analysis:

1) Output not too large:

for all i

if $|S_i| \ge \sqrt{\frac{\varepsilon}{n}} \cdot \frac{|S|}{c \cdot t}$

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

else $\rho_i \leftarrow 0$ $\quad$ i "small"

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

idealistic case $\quad$ Suppose $\forall i$ $\rho_i = \frac{|B_i|}{n}$,

then $\sum_i \rho_i (1+\beta)^{i-1} = \sum_i \frac{|B_i|}{n} \underbrace{(1+\beta)^{i-1}}_{\le \text{ deg of any node in } B_i}$

$\le \bar{d}$

realistic case $\quad$ Suppose $\forall i$ $\rho_i \le \frac{|B_i|}{n}(1+\gamma)$

$\Rightarrow \sum_i \rho_i (1+\beta)^{i-1} \le \bar{d}(1+\gamma)$

(note that we are assuming this for all big i & for all small i we set $\rho_i = 0$)

2) Can output be too small?

if $\forall i$ $\rho_i = \frac{|B_i|}{n}$ then $\sum_i \rho_i (1+\beta)^{i-1} = \sum_i \frac{|B_i|}{n}(1+\beta)^{i-1}$

$\ge (1-\beta)\sum_i \frac{|B_i|}{n} \underbrace{(1+\beta)^{i}}_{\ge \text{ deg of nodes in } B_i}$

since $(1+\beta)(1-\beta) < 1$

$\ge (1-\beta)\bar{d}$

by sampling, for big i, $p_i \geq \frac{|B_i|}{n}(1-\gamma)$

for small i ????

$t \approx O(\log n/\varepsilon)$

big $\frac{|S_i|}{|S|} \geq \gamma\sqrt{\frac{\varepsilon}{n}} \cdot \frac{1}{Ct}$

o.w. small

How much undercounting?

divide edges into 3 types

1) big-big: both endpts in big buckets — counted twice

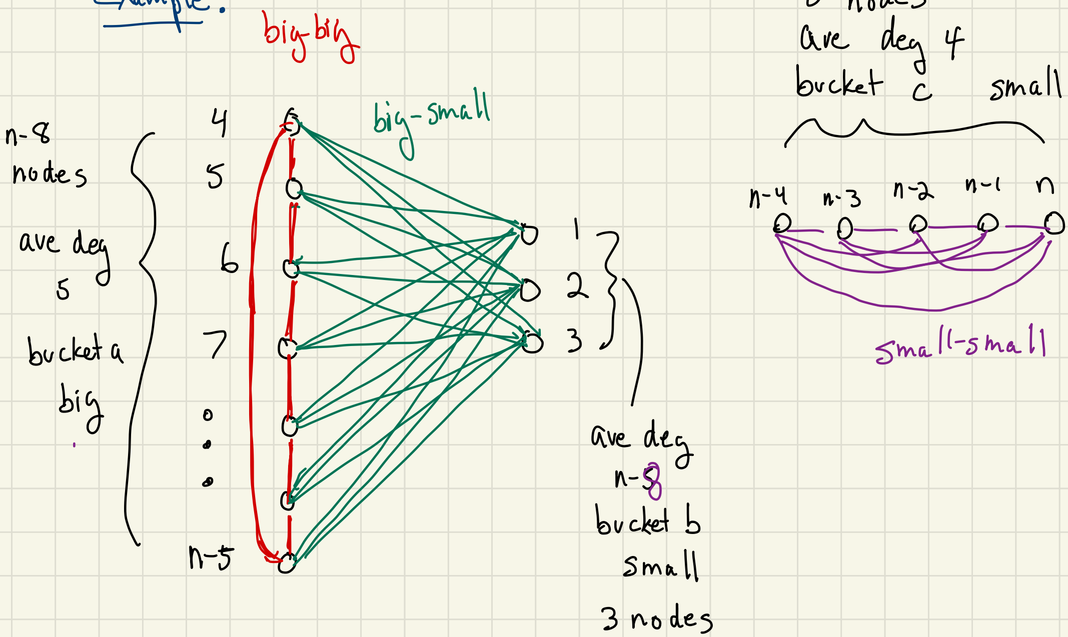
2) big-small: one endpt in big bucket
" " " small " — counted once

3) small-small: both endpts in small buckets — never counted

big-big ok
big-small undercounted by $\frac{1}{2}$
small-small not counted at all

} hope for factor 2 approx

Example:

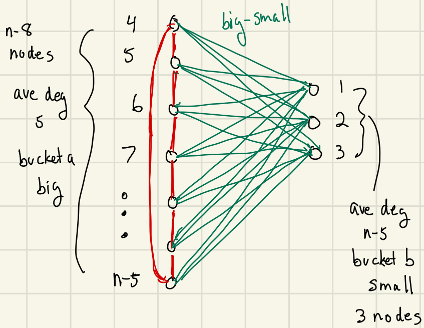


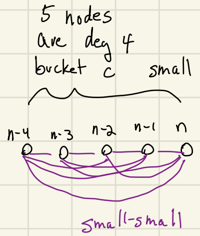
Total degree $5\cdot(n-8) + (n-8)\cdot 3 + 5\cdot 4 = 8(n-8)+20$

ave degree $\approx 8n$

algorithm will likely output ≈ 5

Example:





New algorithm:

- Take sample S (how big?)
- $S_i \leftarrow S \cap B_i$
- estimate $|B_i|$:

  for all i

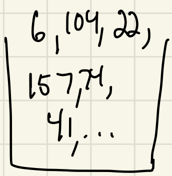
  if $|S_i| \geq \sqrt{\frac{\varepsilon}{n}} \cdot \frac{|S|}{c \cdot t}$ i "big"

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

  else $\rho_i \leftarrow 0$ i "small"

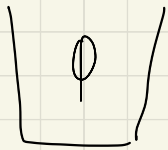
- Output $\sum_i \rho_i (1+\beta)^{i-1}$

Samples:



bucket a

bucket b



bucket c

most nodes in sample

$\Rightarrow$ whp bucket a big

$\rho_a \leftarrow 1$

output $= 5$

unlikely to see <u>any</u> sample

whp $\rho_b = \rho_c = 0$

Good news:

Small buckets can't have many nodes

$\Rightarrow$ bound on total # small-small edges

if $|B_i| > \frac{2\sqrt{\varepsilon n}}{ct}$ then expected size of S_i is

$$\geq |S| \cdot \frac{|B_i|}{n}$$

$$\geq |S| \cdot 2 \cdot \sqrt{\frac{\varepsilon}{n}} \cdot \frac{1}{ct}$$

= twice the threshold for being big

this is why we set threshold for big as we did

so very likely that algorithm will decide i is big

(sampling Chernoff + union bnd)

Assume for all i "small" that $|B_i| \leq \frac{2\sqrt{\varepsilon n}}{ct}$

then total # small-small edges is:

$$\leq \left(\frac{2\sqrt{\varepsilon n}}{c\cancel{t}} \cdot \cancel{t}\right)^2 = O(\varepsilon n)$$

if ignore small-small edges,

they affect approx of $\bar{d}$

recall we assumed $\bar{d} \geq 1$

by $\leq \varepsilon n$ additive factor

$\leq (1+\varepsilon)$ multiplicative factor

First Claim:

big-small error

small small eror

Algorithm gives factor $(2+\varepsilon)$-mult approx

so far, all we have used are degree queries!

Improving further:

need to improve on "big-small" edges

Main idea:

double count from the big side!

New queries:

random neighbor query (v):

given v, return random nbr of v

Implementation:

1st use of nbr queries!! →

1. degree query to v.
2. pick random $i \in [1 .. \deg(v)]$
3. neighbor query (v, i)

pick (almost) random edge in (big) bucket i:

sample nodes until fall into bucket i

random nbr query from 1st node that falls in i

Estimate fraction big-small in B_i (big):

repeat $O(1/\delta)$ times:

pick random node $u \in B_i$

$e \leftarrow$ random nbr of u ← deg queries

set a_j to be $\begin{cases} 1 & \text{if } e \text{ "big-small"} \\ 0 & \text{o.w. (e "big-big")} \end{cases}$

Output α_i = average a_j

Analysis:

easy case: all nodes in B_i have same degree d

$T_i \leftarrow$ # big-small edges in B_i

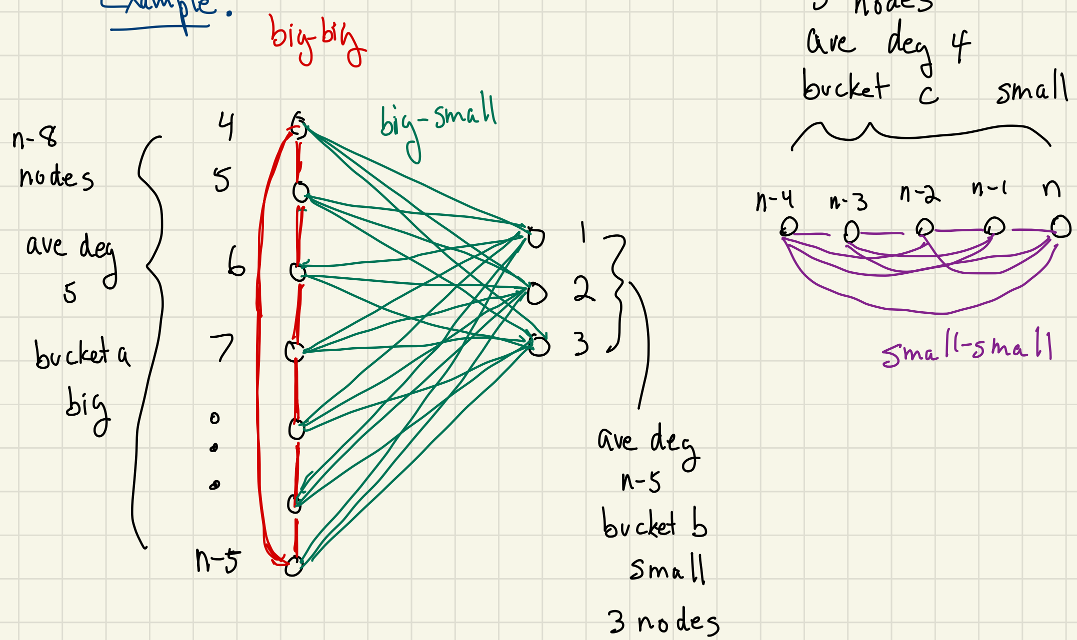
$$\Pr[\text{specific "big-small" edge } \underset{(u,v)}{\overset{\shortparallel}{e}} \text{ in } B_i \text{ chosen}] = \frac{1}{|B_i|} \cdot \frac{1}{d}$$

$$\Pr[a_j = 1] = E[a_j] = \frac{T_i}{|B_i| \cdot d}$$

general case: all nodes in B_i have degrees within $(1+\beta)$ factor of each other

$$\frac{T_i}{|B_i|(1+\beta)^i} \leq E[a_j] \leq \frac{T_i}{|B_i|(1+\beta)^{i-1}}$$

Example:



Total degree: $5\cdot(n-8) + (n-8)\cdot 3 + 4\cdot 5 = 8(n-8)+20$

ave degree $\approx$ 8

algorithm will likely output ≈ 5

$\#$ big-small edges slots: $3\cdot(n-8)$

Fraction of big-small over total: $\approx \frac{3(n-8)}{5(n-8)} = \frac{3}{5}$

$E[a_j] = \frac{3}{5}$

Output $1\cdot(1+\frac{3}{5})\underbrace{(1+\beta)^a}_{\approx 5} \approx 8$

Final Algorithm:

- sample $\theta\left(\frac{\sqrt{n}}{\varepsilon}t\right)$ nodes + place in S
- $S_i \leftarrow S \cap B_i$
- For all i

  if $|S_i| \geq \sqrt{\frac{\varepsilon}{n}} \frac{|S|}{ct}$

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

  for all $v \in S_i$

  - Pick random nbr u of v
  - $\chi(v) \leftarrow \begin{cases} 1 & \text{if } u \text{ is small} \\ 0 & \text{o.w.} \end{cases}$

  est fraction of big-small edges hanging off bucket i

  $$\alpha_i \leftarrow \frac{|\{v \in S_i \mid \chi(v) = 1\}|}{|S_i|}$$

  else use $\rho_i \leftarrow 0$

- Output $\sum_{\text{large } i} \rho_i (1 + \alpha_i)(1+\beta)^{i-1}$

big-big + one side of big-small

Correction to get other side of big-small

Where do errors come from?

estimate p_i's, α_i's $\Big\}$ mult $1+\varepsilon$

small-small edges $\}$ additive