Approximating the Average degree

def Average degree \[ d = \frac{\sum_{u \in V} d(u)}{|V|} \]

\( G \): simple (no parallel edges, self-loops)
\( \Omega(n) \) edges (not "ultra-sparse")

<table>
<thead>
<tr>
<th>Node ( v )</th>
<th>( d(v) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>\ldots</td>
<td></td>
</tr>
<tr>
<td>( n )</td>
<td>2</td>
</tr>
</tbody>
</table>

Adjacency list + degrees:
- degree queries: on \( v \) return \( d(v) \)
- neighbor queries: for \((v, j)\) return \( j \)th nbr of \( v \)
Naive sampling:

Pick **?? random nodes** \( V_1 \ldots V_5 \)

output \( \frac{1}{5} \sum_i d(v_i) \) (are degree of sample)

straight-forward use of Chernoff/Hoeffding needs \( \Omega(\frac{1}{n}) \) samples

Degree sequences are special?

\((n-1, 0, 0, \ldots, 0)\) not possible

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

Some lower bounds for approximation:

"Ultra-sparse" case:

need linear time to get any multiplicative approx.

graph with 0 edges vs. graph with 1 edge

are deg = 0

are deg = \( \frac{1}{n} \)

\( \Omega(n) \) queries to distinguish
Ave deg $\geq 2$ case:

- $n$-cycle $\bar{d} = 2$ [diagram of an $n$-cycle]
- $n - cn^{1/2}$ cycle + $n^{1/2}$-clique $d \approx 2 + c^2$

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

Algorithm

**idea**: group nodes of similar degrees and estimate average within each group.

- $\exists$ doesn't work for estimating arbitrary numbers, why should it work here?

buckets:

- Set $\beta = \varepsilon / c$
- $t = O\left(\frac{\log n}{\varepsilon}\right)$ # buckets

$$B_i = \left\{ v \mid (1+\beta)^{d(v)} < d(v) \leq (1+\beta)^{d(v)} \right\}$$ for $i \leq 0 \ldots t-1$

Note that total degree of nodes in $B_i$

$$(1+\beta)^{d(v)} |B_i| \leq d_{B_i} \leq (1+\beta)^{d(v)} |B_i|$$

+ Total degree of graph $\geq \sum \frac{1}{n} (1+\beta)^{d(v)} |B_i| = \sum \leq (1+\beta)^{d(v)} |B_i|$. 

\[\varepsilon \leq (1+\beta)^{d(v)} |B_i|\]
First idea:

- Take sample $S$
- $S_i \in S \cap B_i$ (samples that fell in $i^{th}$ bucket)
- Estimate average degree of $B_i$
  
  using $S_i$
  
  i.e. $\rho_i \leq \frac{|S_i|}{3}$

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

Problem: i. st. $|S_i|$ is small

("
  " $|B_i|$ " "
) $\forall$ for these, estimate of $\rho_i$ will be "off"
Next idea: use "0" for small buckets

Algorithm:
1. Sample $S$
2. $S_x = S \cap B_x$
3. For all $i$
   - if $S_x = \frac{e}{\sqrt{n}} \cdot \frac{|S_x|}{c \cdot t}$
     - use $p_x = \frac{|S_x|}{|S_x|}$
   - else $p_x = 0$
4. Output $\sum_x p_x (1+\beta)^{i-1}$

Analysis:
Output not too large:

Ideal (unrealistic) case
Suppose $\forall i \ p_x = \frac{|B|}{n} \Rightarrow \sum_x p_x (1+\beta)^{i-1} = \sum_x \frac{|B|}{n} (1+\beta)^{i-1} \leq d$

Realistic case
Suppose $\forall i \ p_x \leq \frac{|B|}{n} (1+\beta)$

$\Rightarrow \sum_x p_x (1+\beta)^{i-1} \leq d (1+\beta)$
for small $i$ $p_x \leq \frac{|B_x|}{n} (1+\beta)$ by def
large $i$ $p_x \leq \frac{|B_x|}{n} (1+\beta)$ whp by sampling

So output not too large!
But are we undercounting by a lot?

For large $i$:

by sampling, \( p_x \approx \frac{|B_x|}{n} (1 - \gamma) \)

so \( \sum \beta_a (1 + \beta)^{i-1} \approx \sum \frac{|B_x|}{n} (1 - \gamma) (1 + \beta)^{i-1} \)

\( \geq (1 - \beta) (1 - \gamma) \sum \frac{d(v)}{n} \)

\( = (1 - \beta) (1 - \gamma) \bar{d} \)

For small $i$ ???

undercounting on small buckets:

3 types of edges:

1) large-large - both endpoints in large buckets
2) large-small - one endpoint in large bucket, one in small
3) small-small - both endpoints in small buckets

how many small-small edges?

good news: small buckets don't have many nodes

Assume for all small buckets, \( |B_x| \leq \frac{\sqrt{E}}{\epsilon} \frac{\sqrt{n}}{C^2} \frac{\sqrt{\lambda n}}{C^2} \)

total # small-small edges:

\( \leq \left( \frac{2 - \sqrt{\lambda n}}{\epsilon^2} \right)^2 = O\left( \frac{E \sqrt{n}}{C^2 \epsilon^2} \right) = O\left( E \sqrt{n} \right) \)

so if we ignore them, they affect approx of

\( d \) by \( \leq (1 + \epsilon) \) multiplicative factor, \( E \) additive.

when graph has degree $\beta n$.
First Claim:

Algorithm gives factor $2$ multi-approx since large-small underestimated by at most $\frac{1}{2}$ factor.

$$\Rightarrow 2+\varepsilon \text{- multiplicative approximation}$$

Improving Further

Need to do better on "large-small"

Idea: estimate fraction of "large-small" & correct for them

How?

Plan: standard sampling?

- Pick random edge
- Or pick "almost" random edge

New Query:

**Random Neighbor Query** ($v)$:

given $v$, return random nbr of $v$

(Implement via 1. degree query to $v$

2. pick random $i \in [1..\text{deg}(v)]$

3. query $(v,i)$)
Pick almost random edge in a bucket:

pick random edge by picking any node that falls in that bucket + random nbr query from that node.

Algorithm to estimate fraction large-small in $B_i$:

repeat $O(1/\epsilon^2)$ times:
pick random node $u \in B_i$
e $\leftarrow$ random nbr of $u$
set $a_i$ to $\{1\}$ if $e$ is "large-small"
set $a_i$ to $\{0\}$ o.w.

Output $a_i = \text{average } a_j$

Analysis

Easy case: if all nodes in $B_i$ have same degree:

Let $T_i$ = number of "large-small" edges in $B_i$

$\Pr [\text{"large small" edge } e \text{ in } B_i \text{ chosen}] = \frac{1}{d |B_i|}$

$E[a_j] = \Pr [\text{any "large small" edge in } B_i \text{ chosen}]$

$= \frac{T_i}{d |B_i|}$

$E[\epsilon] = \text{can only touch } B_i \text{ from one endpoint}$

$B_i$ either "large" or "small" but not both!
general case: all nodes in bucket have degree within $(1 + \beta)$ factor of each other

$$\frac{1}{|B_1| (1+\beta)^{i}} \leq \Pr[\text{"large small" edge in } B_i \text{ chosen}] \leq \frac{1}{|B_1| (1+\beta)^{i-1}}$$

$$\frac{T_i}{|B_1| (1+\beta)^{i}} \leq E[a_{ij}] \leq \frac{T_i}{|B_1| (1+\beta)^{i-1}} \Rightarrow E[a_{ij}] \leq T_i \leq E[a_{ij}] (1+\beta)^{i}$$

algorithm estimates $E[a_{ij}]$ to $(1+\varepsilon)$-multiplicative factor,

giving $(1+\varepsilon)(1+\beta)$ estimate of $T_i$ via $\alpha_i \cdot \rho_i (1+\beta)^{i-1}$

running algorithm for each bucket gives:

**Final Algorithm:**

- Sample $s_i$ s.t. $|s_i| \geq \sqrt{\frac{\log \frac{1}{\varepsilon}}{1}} \cdot \sqrt{n} \geq \varepsilon$
- $s_i \subset s \cap B_i$

  - For all $i$
    - if $s_i \geq \sqrt{\frac{\log \frac{1}{\varepsilon}}{2\varepsilon}} \cdot |s_i|$, use $\rho_i \leftarrow \frac{|s_i|}{|s_i|}$
  
  - For all $v \in s_i$
    - Pick random nbr $u \in v$
    - $X(u) \leftarrow 1$ if $u$ small
    - $X(v) \leftarrow 0$, o.w.
    - $\alpha_i \leftarrow |\{ u \in s_i \mid X(u) = 1 \}|$

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