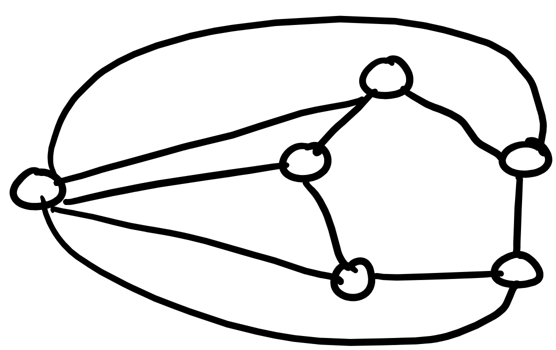
Q: How much sleep did Slota get last night?

A: Not nearly enough.

Hamiltonian paths → spanning paths

Recall: graph join between G and H ($G \vee H$) is a construction where $\forall u \in V(G), \forall v \in V(H)$ we add edge (u,v)

join K_1 to C_5

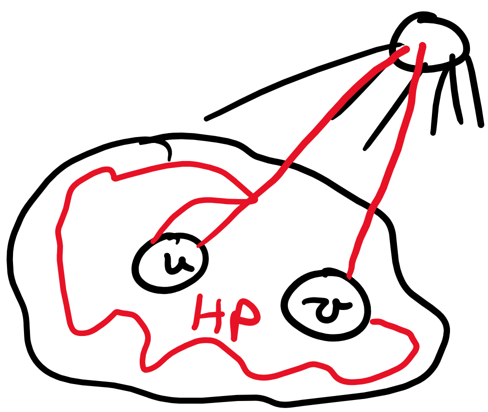


G has a Hamiltonian path

iff

$G \vee K_1$ has a Hamiltonian cycle

(=>)



→ we can trivially construct a HC given that HP on G

(<=) Necessarily, exactly 2 edges from the HC are incident on K_1 → removing them leaves a HP with start/end at the other endpoints of those edges

Let's modify Chvátal's condition to get a sufficient condition for the existence of a HP

If G has degrees:

$$d_1 \le d_2 \le \dots \le d_n$$

then if

$$i < \frac{n+1}{2} \text{ implies } d_i \ge i$$
$$\text{or } d_{n+1-i} \ge n-i$$

$\Rightarrow$ G has a Hamiltonian path

Note: also <u>sufficient</u> but not necessary

RANDOM GRAPHS

GRAPHS

Q: What are random graphs?

A: A graph with randomly configured edges in "some way"

(sp?)

→ Probabilistic model

→ Random generativecpro ess

why: Mirror properties of real graphs for analytical study

Also: random graphs can be used as <u>null models</u>

How do we define a random graph model?

→ Classic model

Erdös-Rényi

O.G.: $G(n, m)$ $\qquad \langle k \rangle = 2\frac{m}{n}$

n: # verts $\quad$ m: # edges $\quad$ $\langle k \rangle$: avg. degree

→ all m edges have randomly selected endpoints, chosen from n vertices with replacement

Newer: $G(n, p)$

n: # verts $\quad$ p: attachment probability

↳ between any $u, v \in V(G)$, $(u, v) \in E(G)$ exists with probability p

Evaluating all u, v pairs for edge generation gives us a Bernoulli process

1

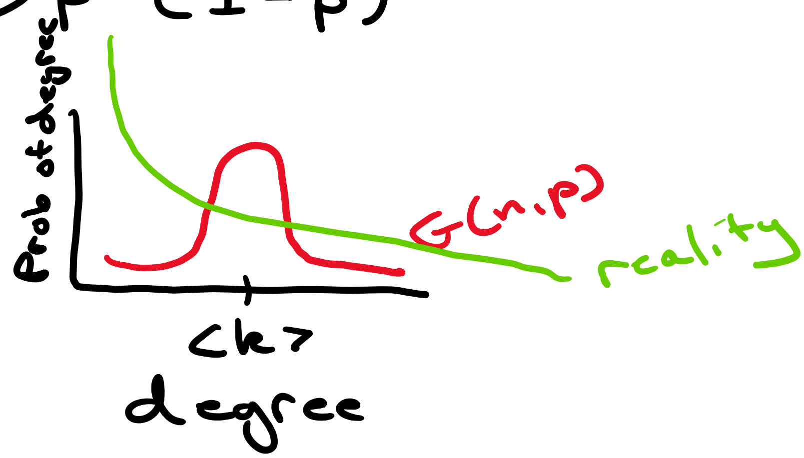
↓

can give us explicit edges or
an analytic <u>degree distribution</u>

→ How many degree-1,
degree-2, degree-k
vertices?

$$P(k) = \binom{n-1}{k} p^k (1-p)^{(n-1)-k}$$

prob. of
degree k



mean value: $\langle k \rangle = \sum_{k=0}^{n-1} k\,p(k) = p(n-1)$

or $= pn$

We can use the above to
study graph properties

- Consider vertex v
- v is expected to have a degree of $\langle k \rangle \rightarrow |N(v)| = \langle k \rangle$

  approximate

- each of $u \in N(v)$ is also expected to have a degree of $\langle k \rangle$

$\rightarrow$ 2-hop neighborhood of v

$$|N_2(v)| = \langle k \rangle \langle k \rangle = \langle k \rangle^2$$

Assume n is suitably large

Assume $\langle k \rangle \approx \langle k \rangle - 1$

(as v is a neighbor of its neighbor

Note: we generally need to make assumptions

In general:

In general:

$$|N_d(v)| = \langle k \rangle + \langle k \rangle^2 + \dots + \langle k \rangle^d$$

d-hop neighborhood

$$|N_d(v)| = \frac{\langle k \rangle^{d+1} - 1}{\langle k \rangle - 1}$$

Consider as $d \to n$

Note: we can take $|N_d(v)| = n$
to determine for what d
we have the entire graph
(aka we can get the expected
graph diameter)

$$|N_d(v)| = n = \frac{\langle k \rangle^{d+1} - \cancel{1}}{\langle k \rangle - \cancel{1}}$$

$$n = \langle k \rangle^d$$

$$d = \frac{\ln(n)}{\ln(k)}$$

much less than

Consider that $\langle k \rangle \lll\lll n$

consider that $\langle k \rangle \lll\lll n$

$$\boxed{d \approx \ln(n)}$$

$\Rightarrow$ expected diameter grows logarithmically with $|V(G)|$

Note: this generally does mirror reality

BIG issue: degree distribution is not realistic

Introducing:

the <u>configuration model</u>

→ this will give us a random graph with some given degree distribution degree distribution

Basic idea:

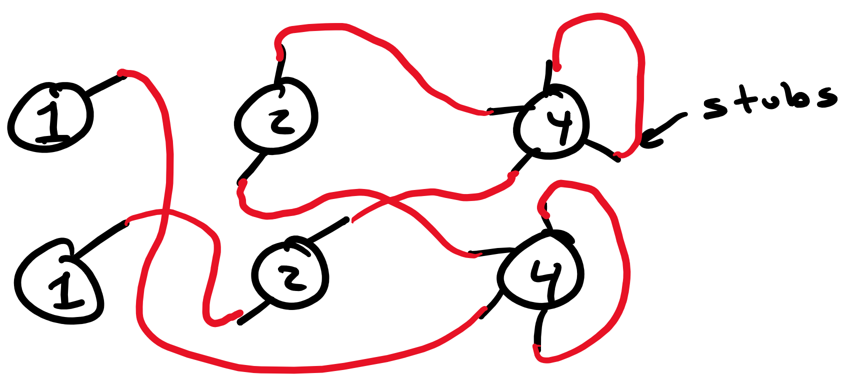
We map degrees (input) to a vertex set

→ v has specified $d(v)$

→ v get $d(v)$ <u>stubs</u>

→ we randomly <u>wire</u> these stubs together

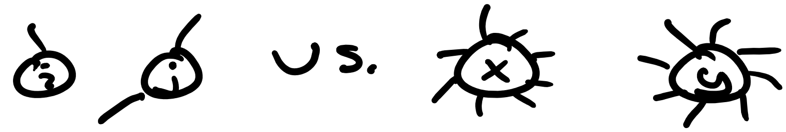
$S = \{4\ 4\ 2\ 2\ 1\ 1\}$



Note: we have a loopy multigraph
(random graph models are usually specific to some graph class)

What about attachment probs?

consider



→ more likely to select a stub from a higher degree vertex

⇒ high degree vertices are pairwise more likely to be attached than low degree vertices

Our attachment probs:

for some $i, j, d(i), d(j), m$

$$m = \frac{1}{2}\sum_{i}^{n} d(i)$$

Prob of edge (i, j):

= (prob. of selecting i's stub)

$*$

$$\frac{d(i)}{2m}$$

$*$ (prob. of selecting j's stub)

$*$

2 ← can select (i,j) or (j,i)

$*$

m ← we make m selections

$$P_{i,j} = \frac{d(i)}{2m}\,\frac{d(j)}{\cancel{2m}}\,\cancel{2m}$$

$$\boxed{P_{i,j} = \frac{d(i)\,d(j)}{2m}}$$

Configuration model

↳ attachment probs.

Chung-Lu $\quad p_{i,j} = \dfrac{w_i w_j}{\sum_k w_k}$

generalization to "weights"

We can generate a simple graph by evaluating $p_{i,j}$ for all i,j pairs

Note: we won't be hitting the degree distribution exactly

(in expectation we are)

(not really)

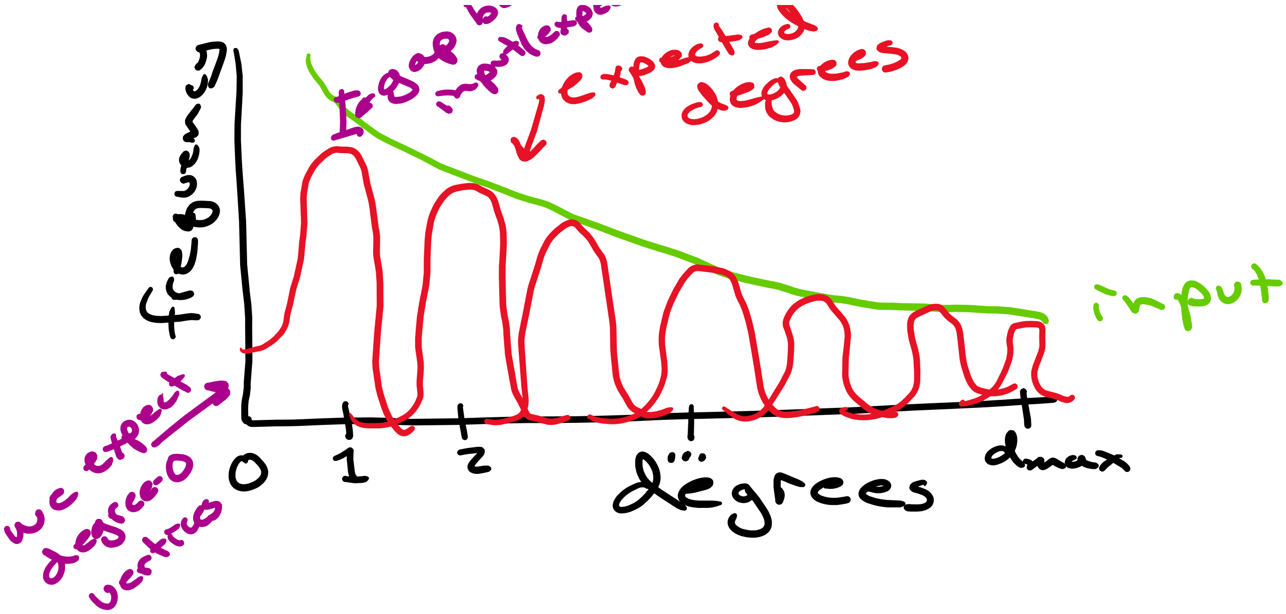
Note x2: we're just layering a bunch of Erdös-Rényi $G(n,p)$ graphs

→ each defined by a unique degree pair

=> a vertex's degree is a sum of expected degrees over all these E-R graphs

gap between input/expected

expected degrees



One more issue: $\dfrac{d(i)\,d(j)}{2m}$

What happens when $d(i)d(j) > 2m$

Can happen when a graph is dense or skewed

multi-graph: $p_{i,j}$ is just the expected number of edges

simple graphs: nonsense

→ random graphs need to be

→ random graphs need to be considered with respect to a specific graph class

---

Null models

→ a graph defined with some properties uniformly randomly selected from all possible graphs with those properties

↳ n, m, DD, etc.

For same n, m: G(n,m) graph

For loopy-multigraphs with same DD: configuration model

For simple graphs with same DD: ??

Chung-Lu probs. are biased

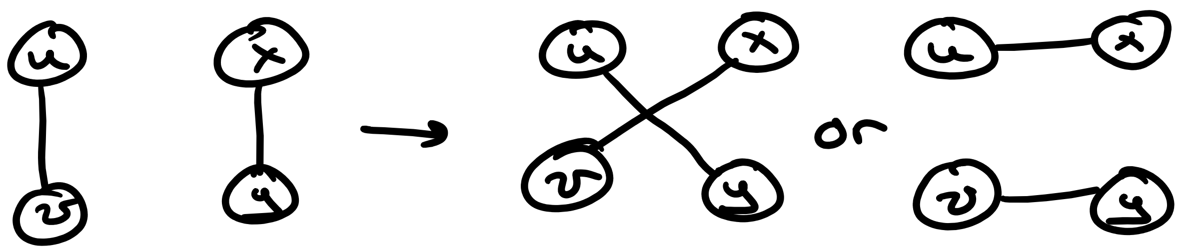
are biased

Q:
How can we get an unbiased sample?

A1: not via attachment probs.

A2: via double-edge swaps
(and Havel-Hakimi)

Double edge swap



Note: degrees are umodified

Note 2: we discard any swaps that add loops or multi-edges

For simple null model generation:

- Generate H-H graph
- Perform "some number" of double-edge swaps

→ this traverses the entire topological space of the class under consideration

Q: How many swaps to get an unbiased sample?

A: Who knows?

↳ open problem: "mixing time"

Note: we can empirically measure "real" attachment probs. by doing this process

Consider a specific subgraph appearing is simply a function of attachment probabilities to realize same topology