(Nearly) Efficient Algorithms for the Graph Matching Problem

Tselil Schramm
(Harvard/MIT)

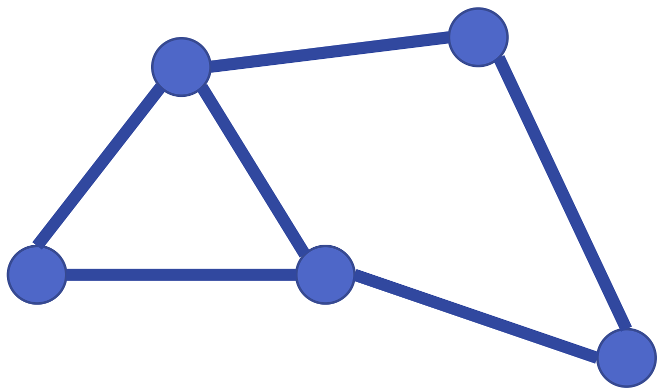
with Boaz Barak, Chi-Ning Chou, Zhixian Lei & Yueqi Sheng (Harvard)

graph matching problem (approximate graph isomorphism)

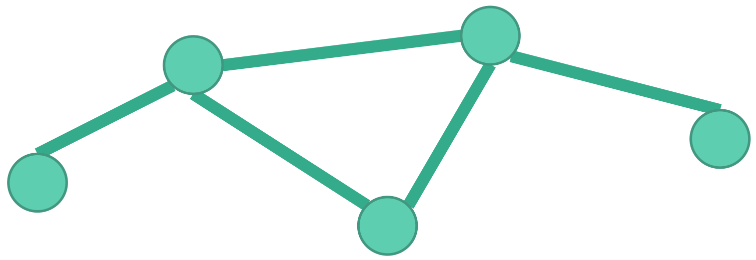
input: two graphs on n vertices

goal: find permutation of vertices that maximizes # shared edges

$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle$$



G_0



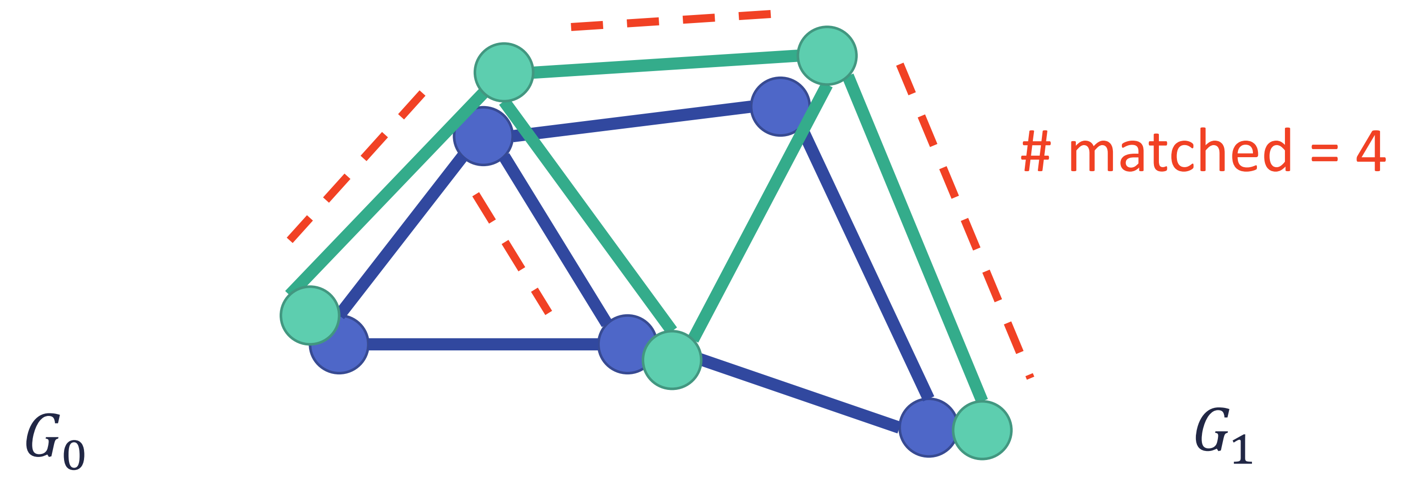
G_1

graph matching problem (approximate graph isomorphism)

input: two graphs on n vertices

goal: find permutation of vertices that maximizes # shared edges

$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle$$

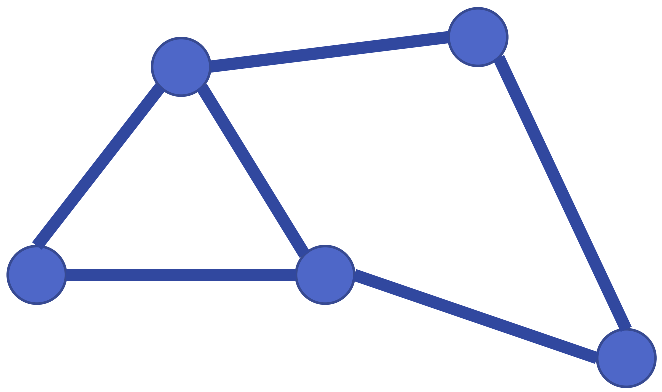


graph matching problem (approximate graph isomorphism)

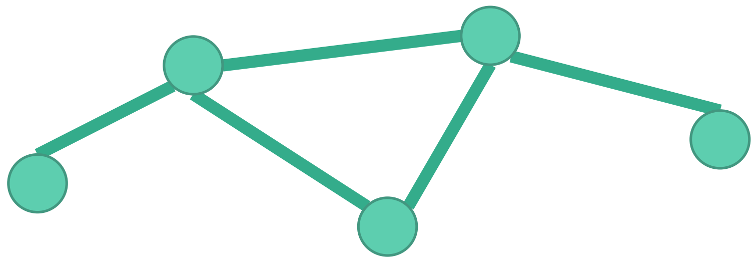
input: two graphs on n vertices

goal: find permutation of vertices that maximizes # shared edges

$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle$$



G_0



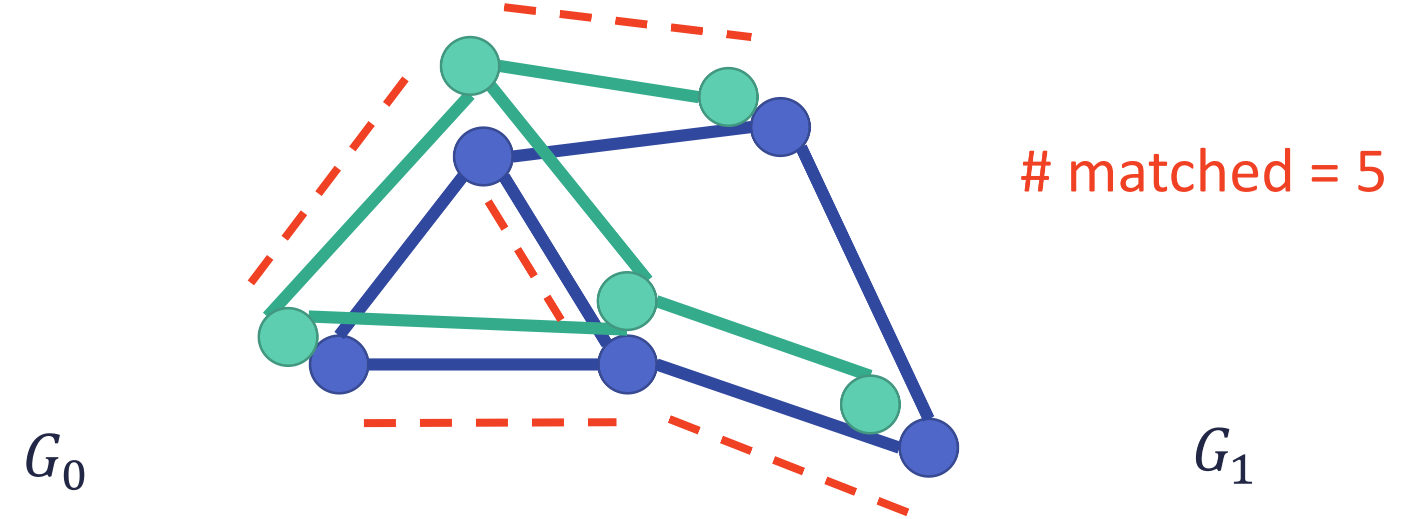
G_1

graph matching problem (approximate graph isomorphism)

input: two graphs on n vertices

goal: find permutation of vertices that maximizes # shared edges

$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle$$



computationally hard (of course)

NP-hard: reduction from quadratic assignment problem (non-simple graphs).

[Lawler'63]

also: reduction from sparse random 3-SAT to approximate version

[O'Donnell-Wright-Wu-Zhou'14]

practitioners: undeterred

- computational biology [e.g. Singh-Xu-Berger'08]
- de-anonymization [e.g. Narayanan-Shmatikov'09]
- social networks [e.g. Korula-Lattanzi'14]
- image alignment [e.g. Cho-Lee'12]
- machine learning [e.g. Cour-Srinivasan-Shi'07]
- pattern recognition, e.g.

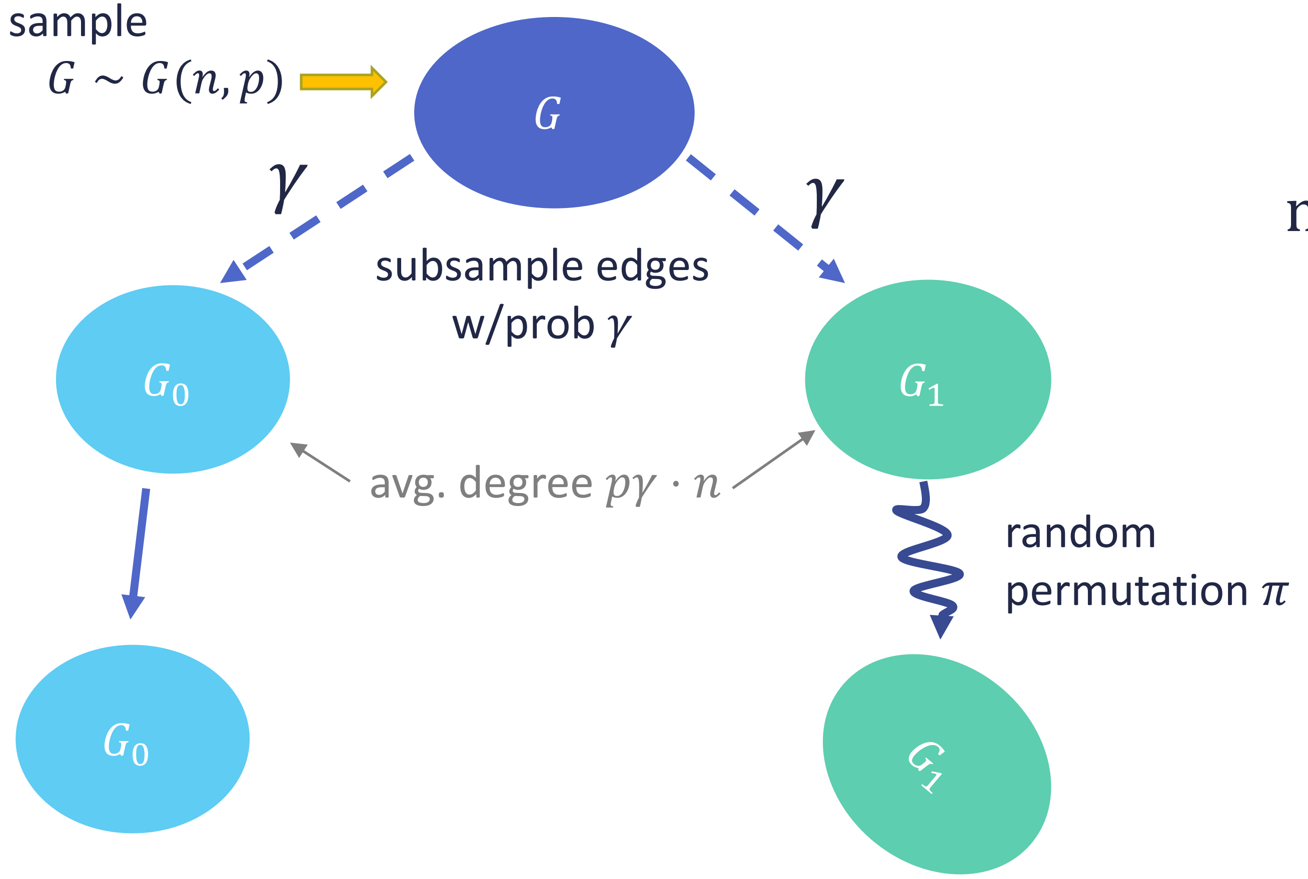
  "thirty years of graph matching in pattern recognition"

  [Conte-Foggia-Sansone-Vento'04]

"robust average-case graph isomorphism"

average case: correlated random graphs

structured model



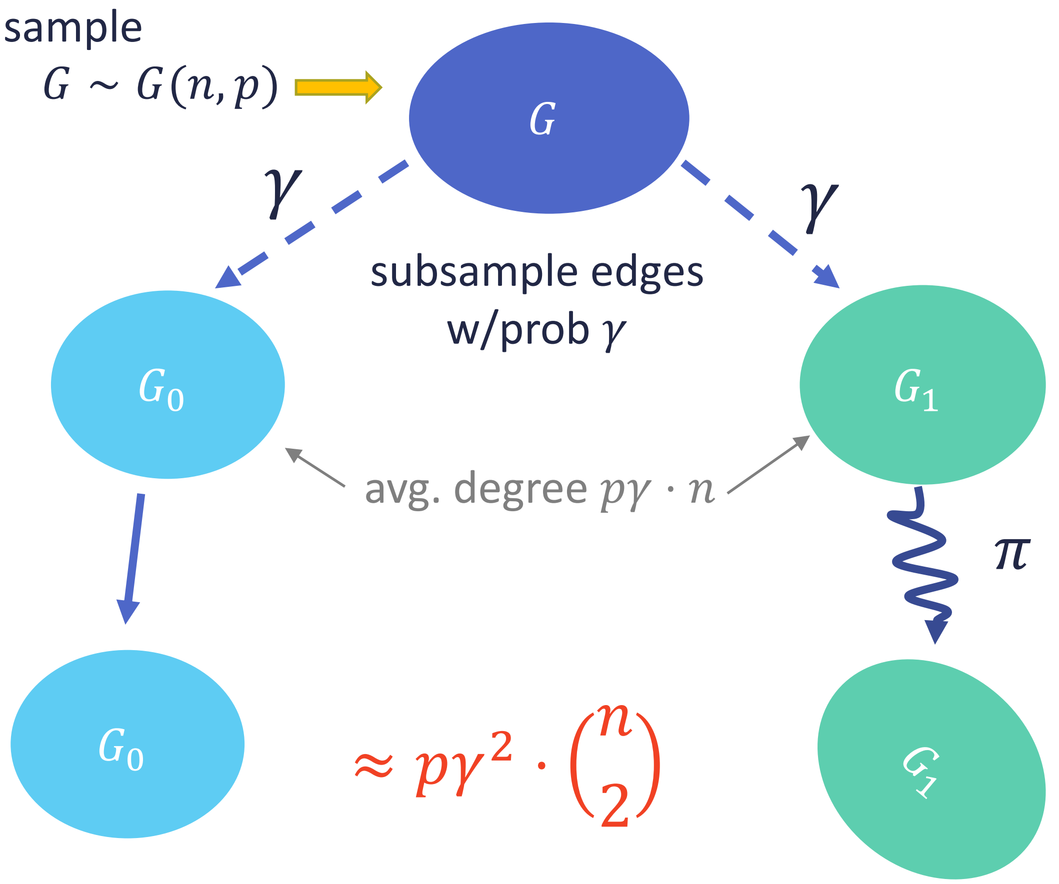
$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle \approx p\gamma^2 \cdot \binom{n}{2}$$

[e.g. Pedarsani-Grossglauser'11,
Lyzinski-Fishkind-Priebe'14,
Korula-Lattanzi'14]

“robust average-case graph isomorphism”

average case: correlated random graphs

structured model

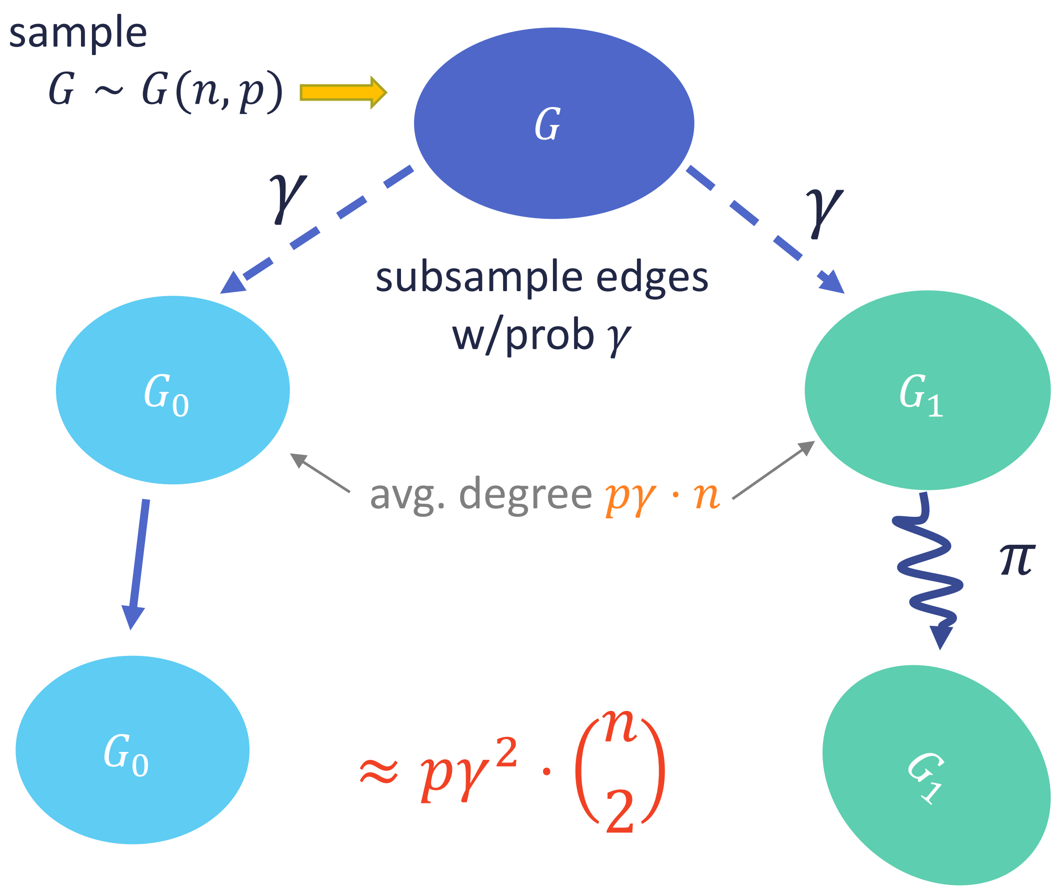


“null” model

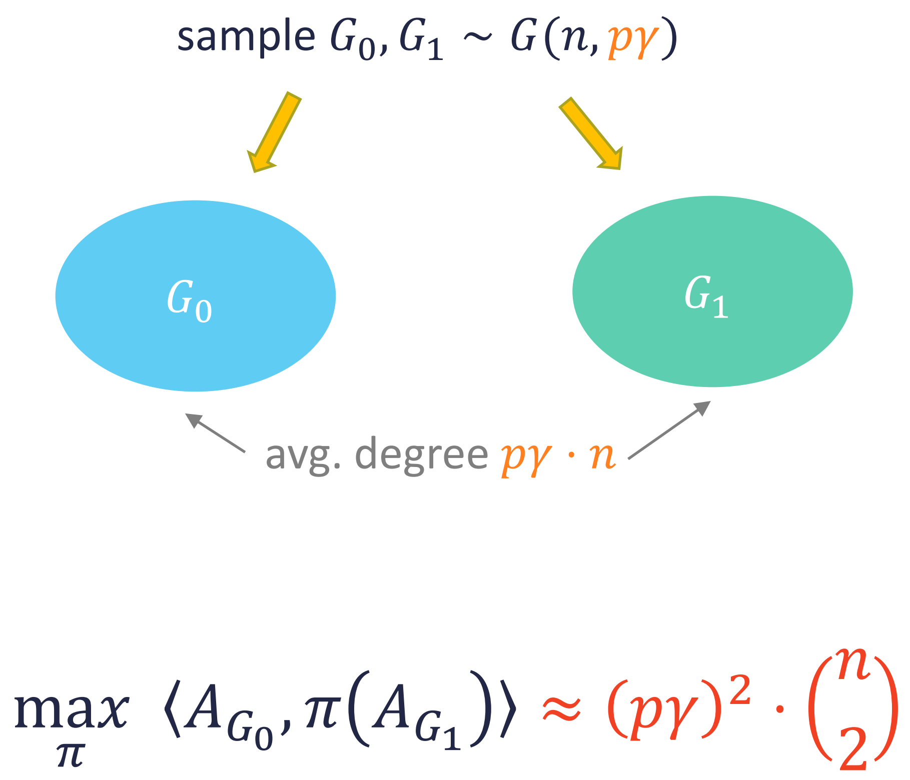
"robust average-case graph isomorphism"

average case: correlated random graphs

structured model

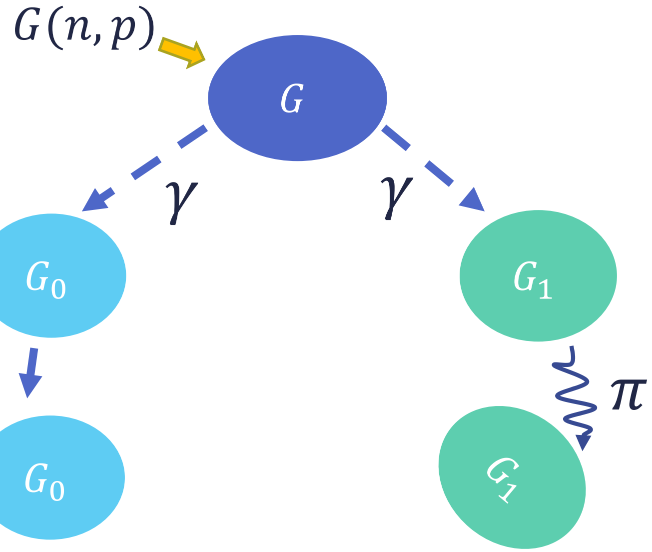


"null" model



information theoretic limit

for which p, γ can we recover π?



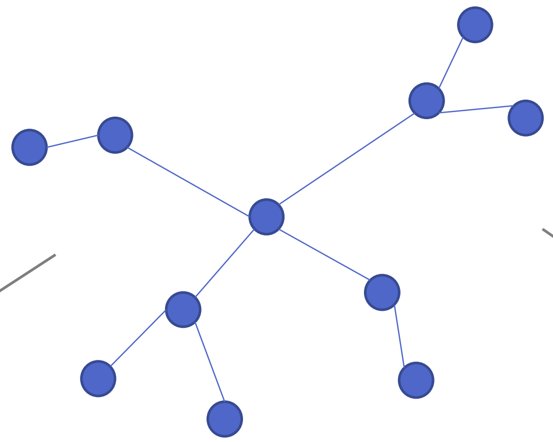
Theorem [Cullina-Kivayash’16&17]

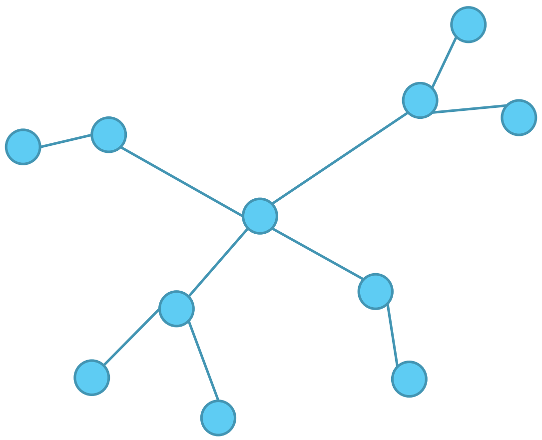
Iff $p\gamma^2 > \frac{\log n}{n}$, with high probability π is the unique maximizing permutation.

algorithms for robust average case?

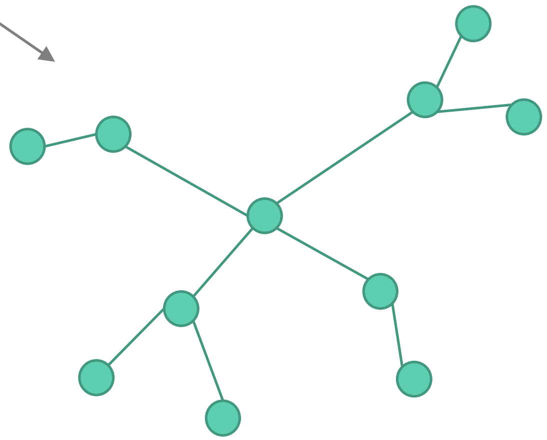
average-case graph isomorphism algorithms fail.

e.g. matching local neighborhoods





match radius-k neighborhoods?



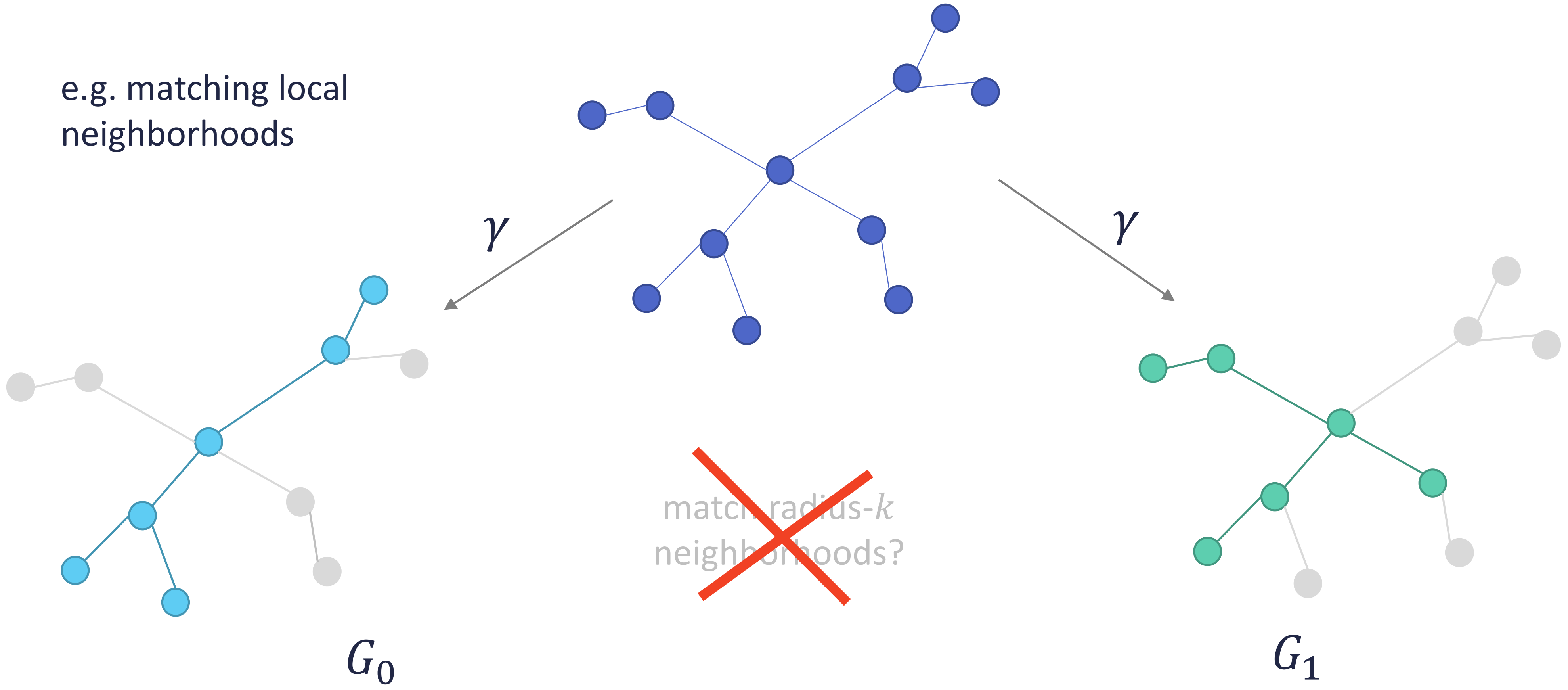
G_0

G_1

algorithms for robust average case?

average-case graph isomorphism algorithms fail.

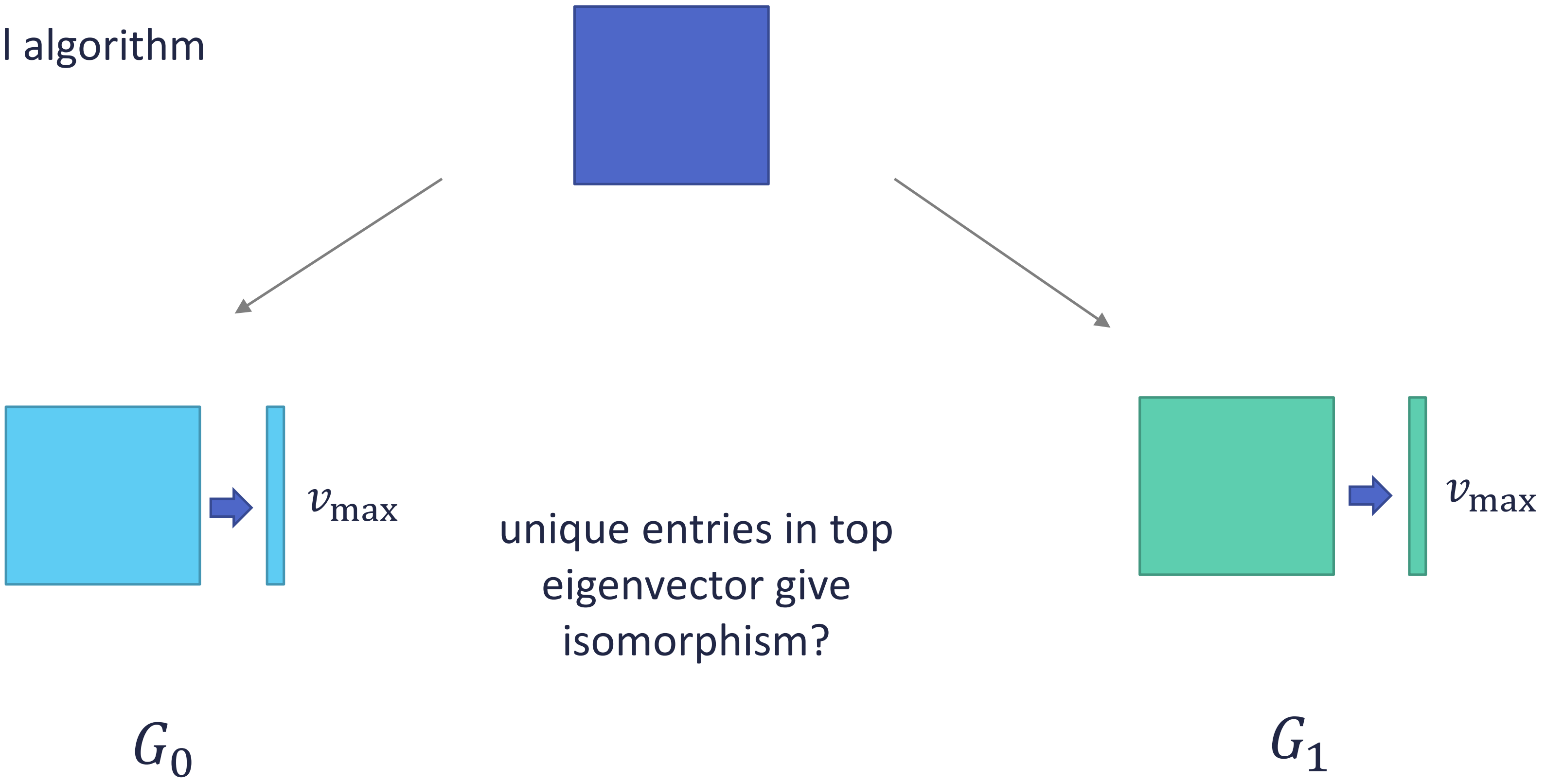
e.g. matching local neighborhoods



algorithms for robust average case?

average-case graph isomorphism algorithms fail.

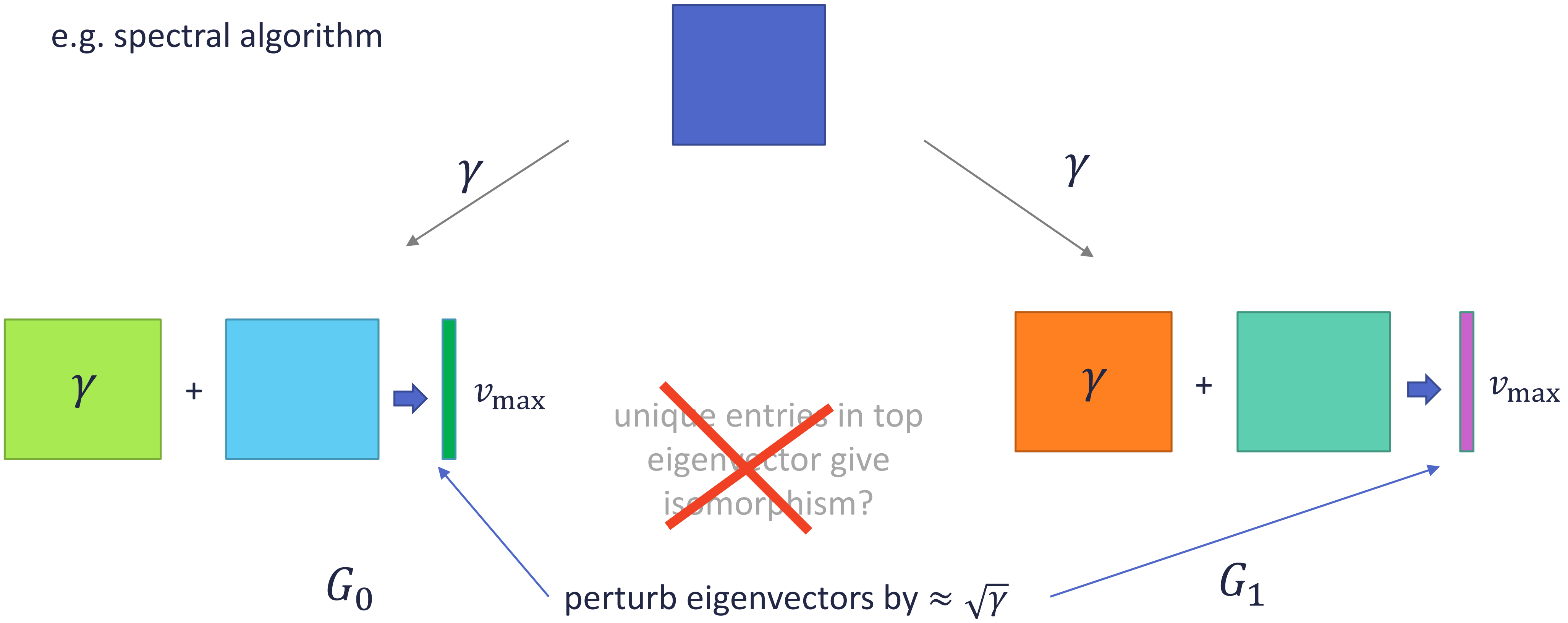
e.g. spectral algorithm



algorithms for robust average case?

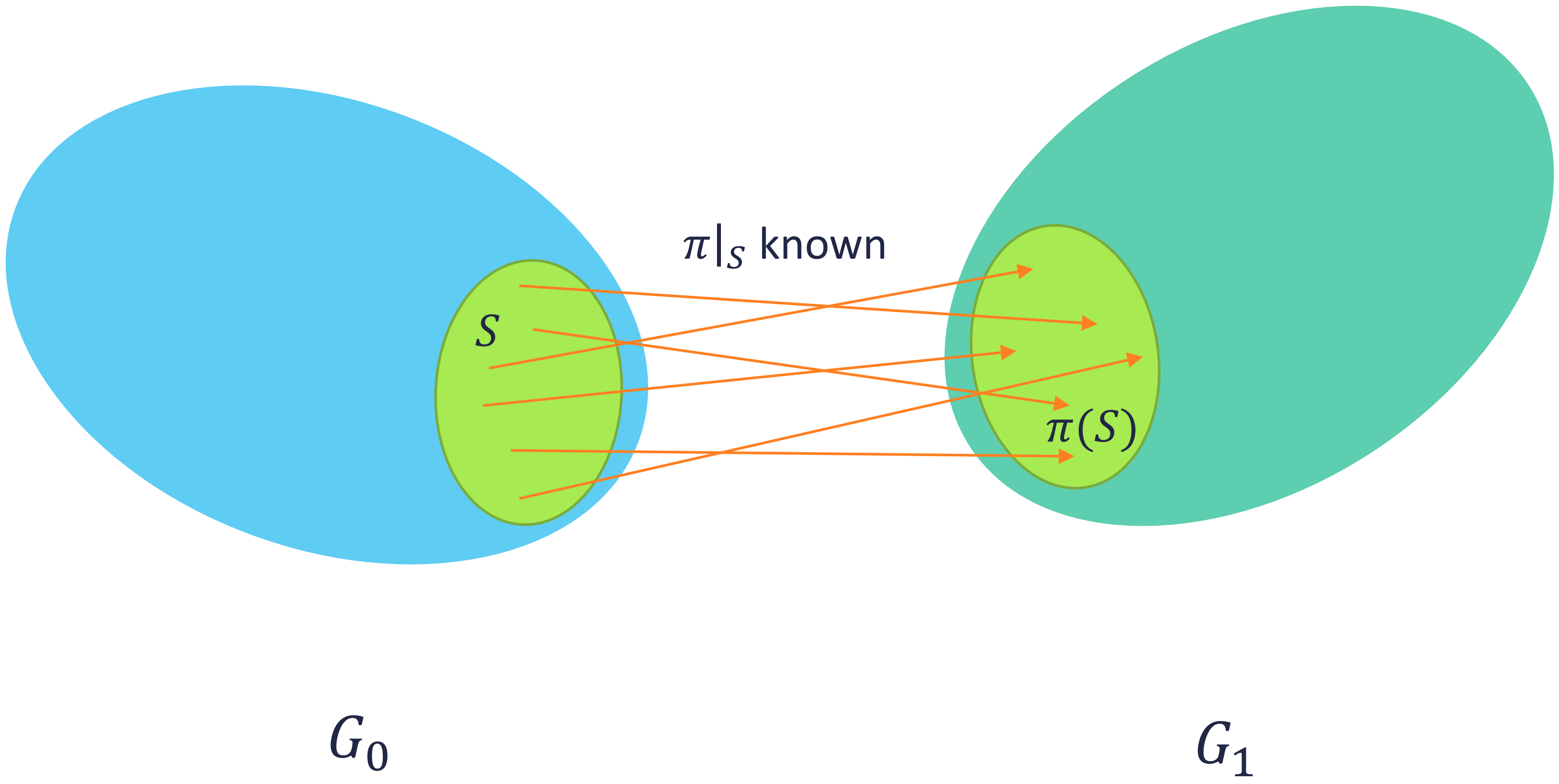
average-case graph isomorphism algorithms fail.

e.g. spectral algorithm



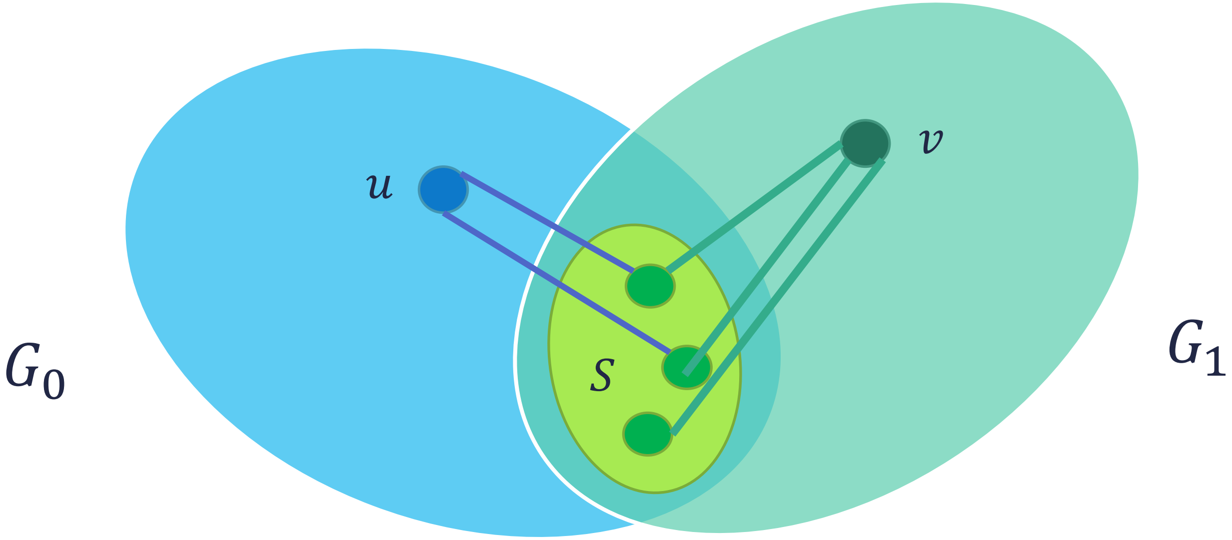
actual algorithms for robust average case?

starting from a seed



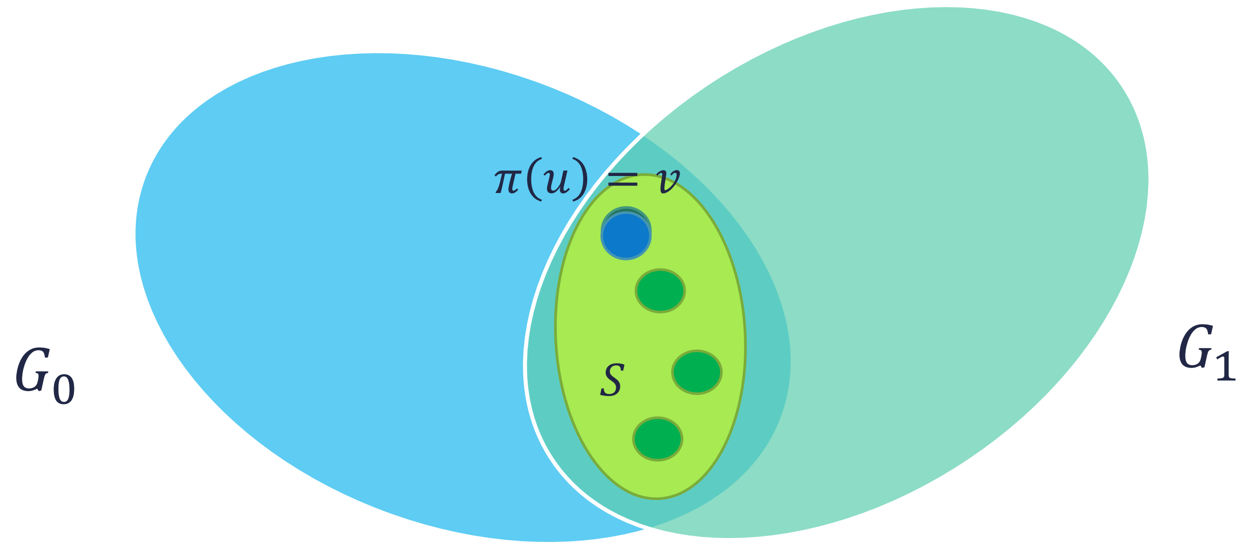
starting from a seed

match vertices with similar adjacency into S



starting from a seed

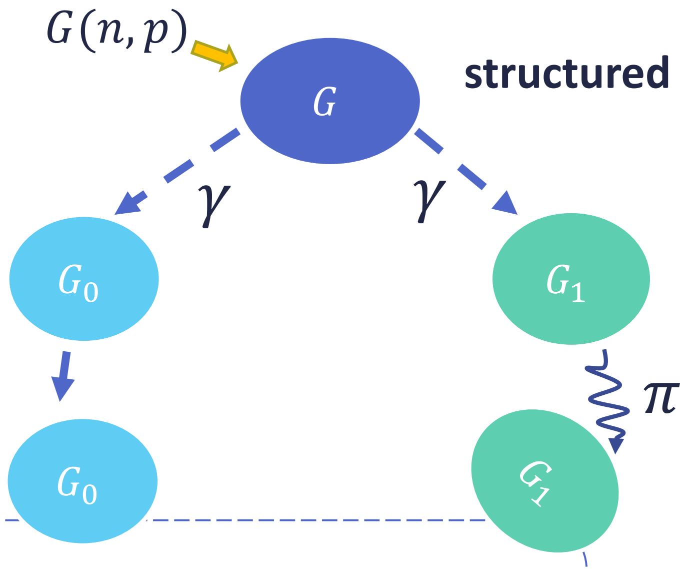
match vertices with similar adjacency into S



iff seed $\geq \Omega(n^{\epsilon})$, the seeded algorithm approximately recovers π. [Yartseva-Grossglauser'13]

need $2^{\tilde{O}(n^{\epsilon})}$ time to guess a seed.

our results

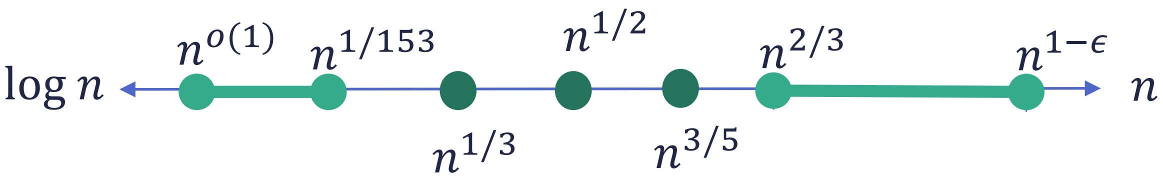


Theorem

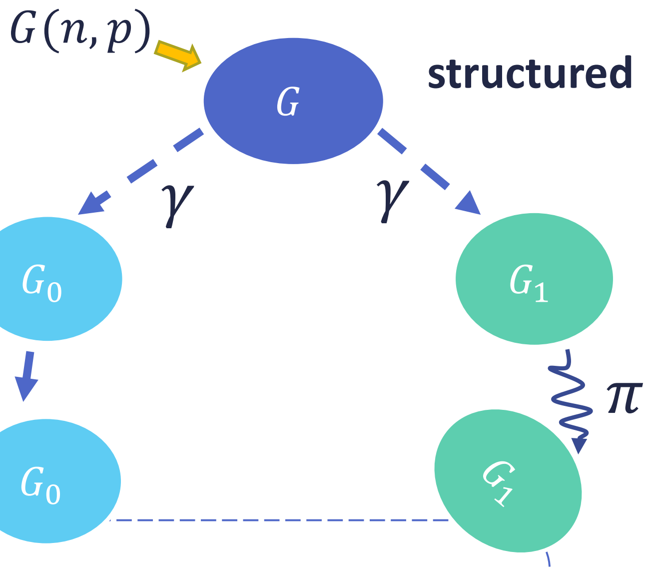
For any $\epsilon > 0$, if $p \in \left[\frac{n^{o(1)}}{n}, \frac{n^{\frac{1}{153}}}{n}\right] \cup \left[\frac{n^{\frac{2}{3}}}{n}, \frac{n^{1-\epsilon}}{n}\right]$ and $\gamma = \Omega(1)$,* there is a $n^{O(\log n)}$ time algorithm that recovers π on $n - o(n)$ of the vertices w/prob ≥ 0.99.

*we allow $\gamma = \Omega\left(\frac{1}{\log\log n}\right)$

$G(n,p)$ **average degrees:**



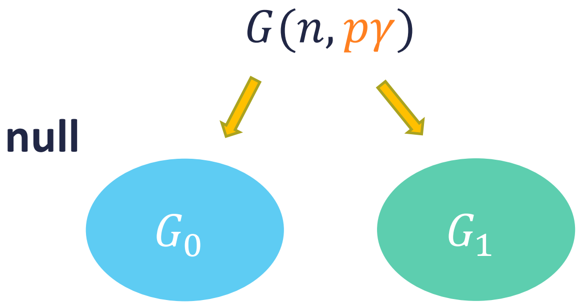
our results



Theorem

For any $\epsilon > 0$, if $p \in \left[\frac{n^{o(1)}}{n}, \frac{n^{\frac{1}{153}}}{n}\right] \cup \left[\frac{n^{\frac{2}{3}}}{n}, \frac{n^{1-\epsilon}}{n}\right]$ and $\gamma = \Omega(1)$,* there is a $n^{O(\log n)}$ time algorithm that recovers π on $n - o(n)$ of the vertices w/prob ≥ 0.99.

*we allow $\gamma \geq \frac{1}{\log^{o(1)} n}$



Theorem

hypothesis testing

If p, γ are as above then there is a $\mathrm{poly}(n)$ time *distinguishing* algorithm for the **structured** vs **null** distributions.

our approach: small subgraphs

hypothesis testing: correlation of subgraph counts

recovery: match rare subgraphs

seedless algorithms!

outline

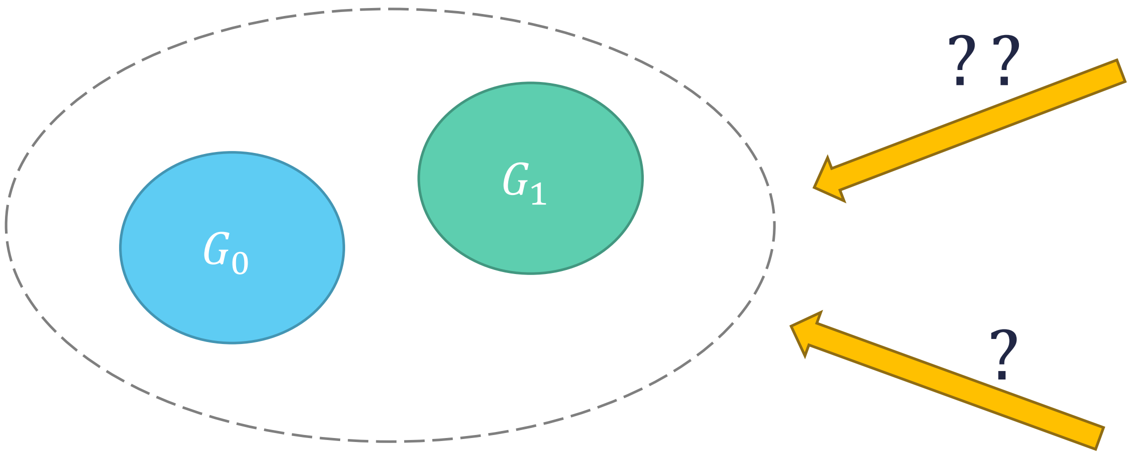
- distinguishing/hypothesis testing
- recovery
- concluding

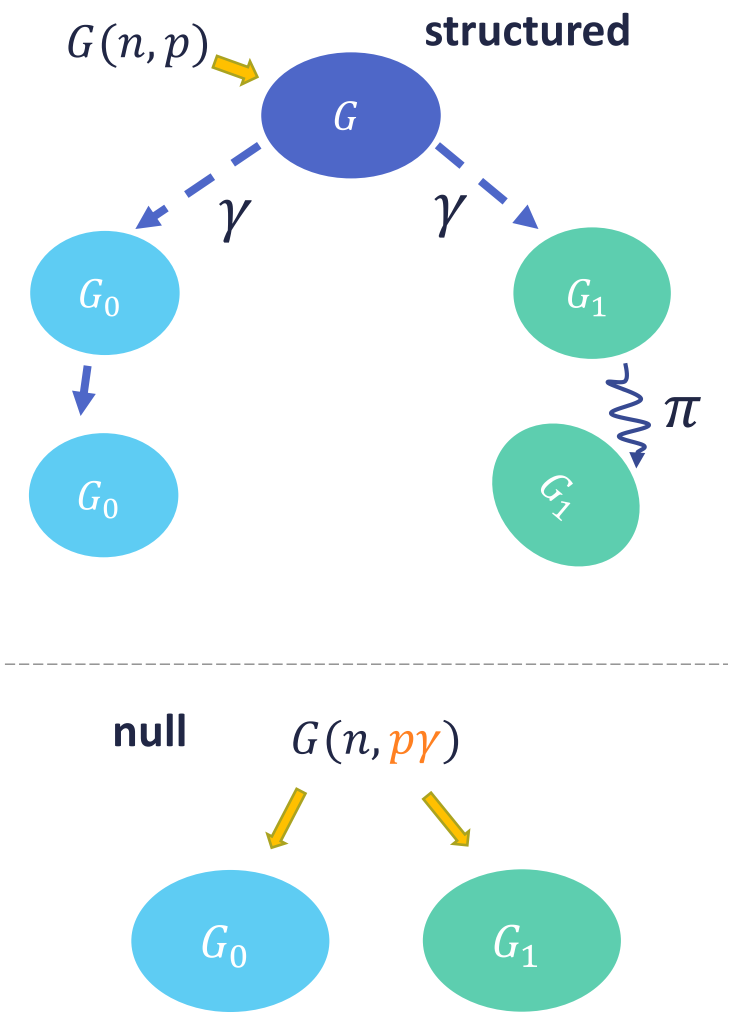
outline

- distinguishing/hypothesis testing
- recovery
- concluding

distinguishing/hypothesis testing

Given G_0, G_1 sampled equally likely from **structured** or **null**, decide w/prob $1 - o(1)$ from which.

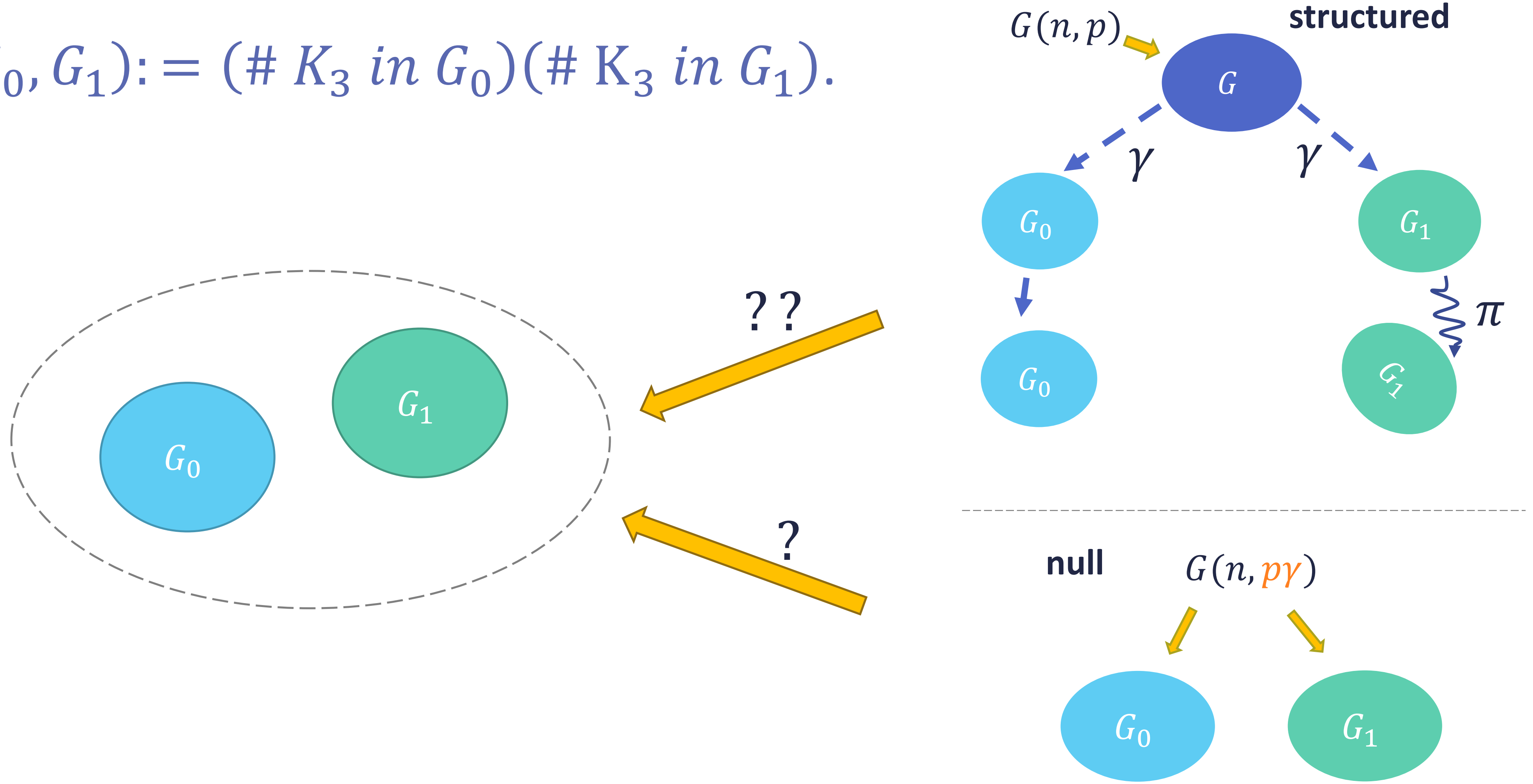




brute force: is there a π with $\geq p\gamma^2 n^2$ matched edges?

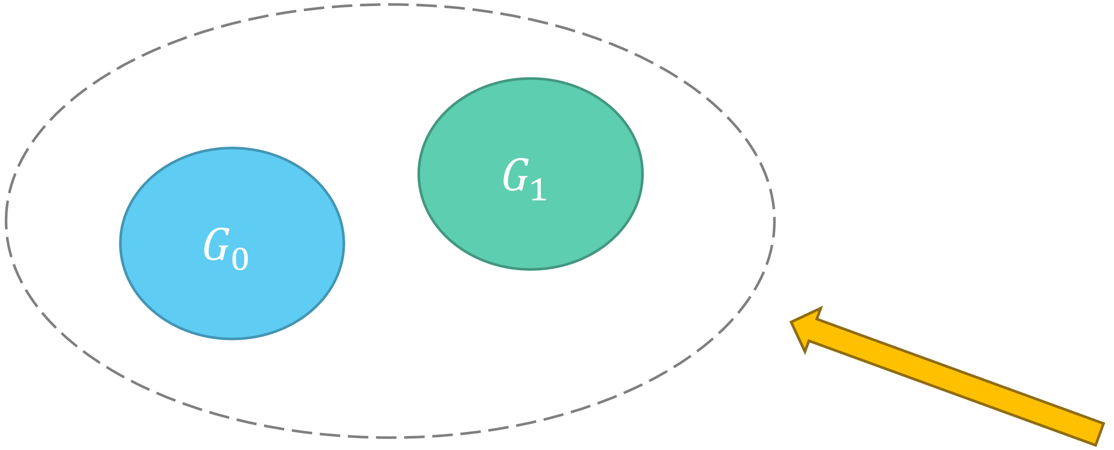
…counting triangles?

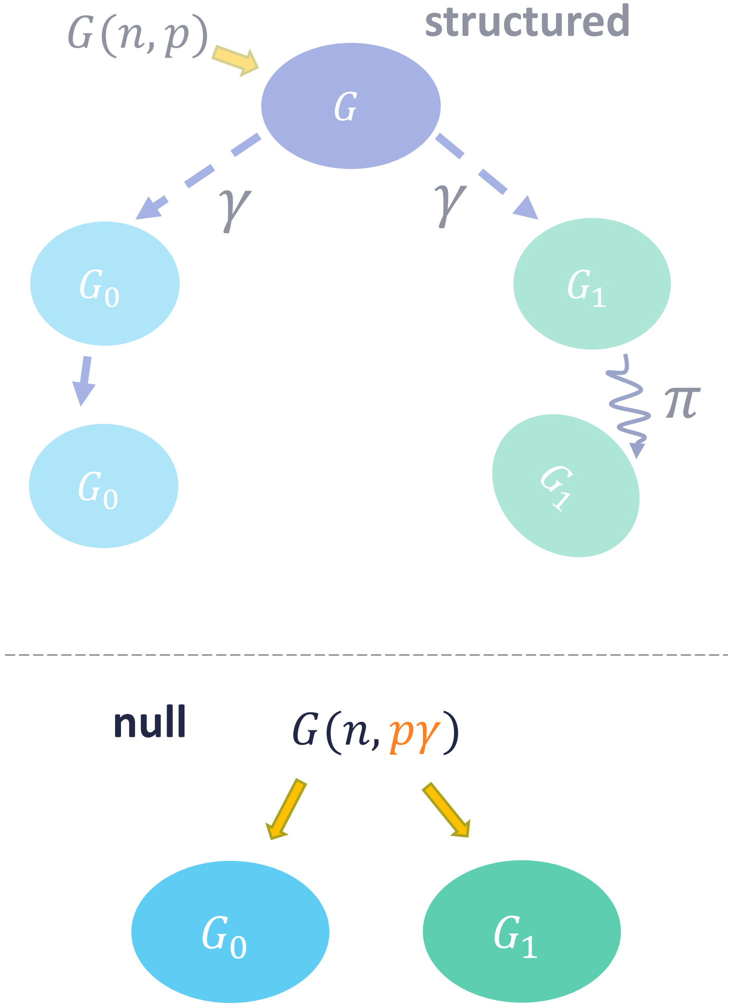
$$cor_{K_3}(G_0, G_1) := (\#\ K_3\ in\ G_0)(\#\ \mathrm{K}_3\ in\ G_1).$$



...counting triangles?

$$cor_{K_3}(G_0, G_1) := (\#\, K_3 \text{ in } G_0)(\#\, \mathrm{K}_3 \text{ in } G_1).$$





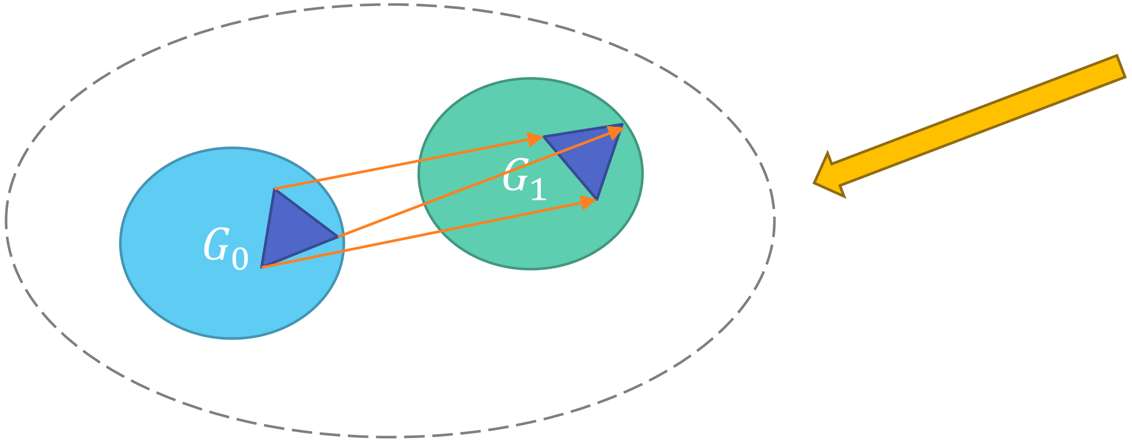
triangle counts in G_0, G_1 are independent

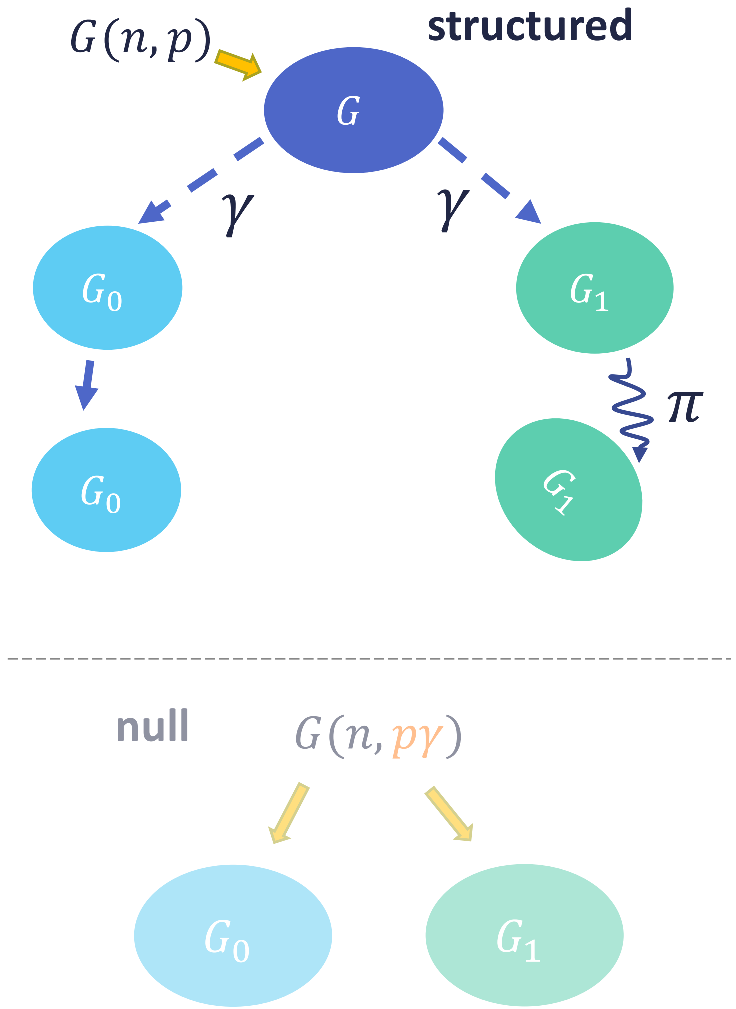
$$\mathbb{E}\left[cor_{k_3}(G_0, G_1)\right] \approx (p\gamma n)^6$$

…counting triangles?

$$\mathbb{E}[cor_{K_3}(G_0, G_1)] \approx (p\gamma n)^6 + (\gamma^2 pn)^3$$

triangle counts in G_0, G_1 are correlated





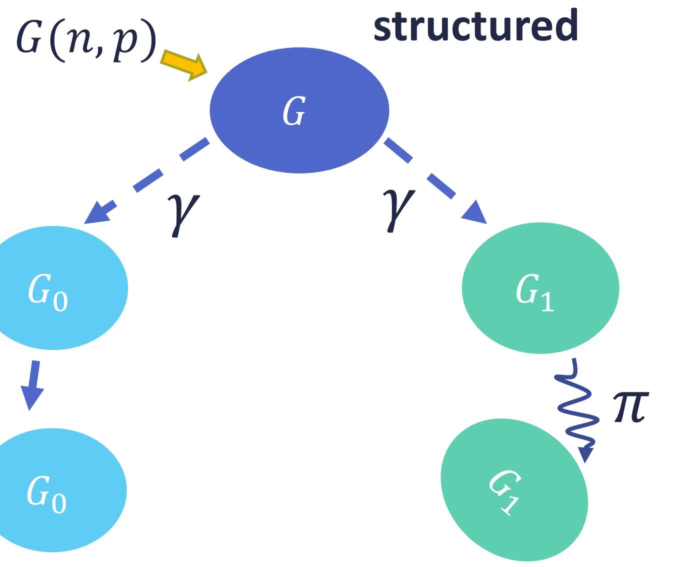
...counting triangles?

structured

$$\mathbb{E}\left[cor_{K_3}(G_0, G_1)\right] \approx (p\gamma n)^6 + (\gamma^2 pn)^3$$

null

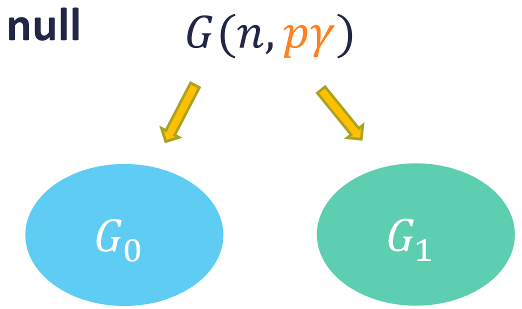
$$\mathbb{E}\left[cor_{K_3}(G_0, G_1)\right] \approx (p\gamma n)^6$$



Variance?

Optimistically, in null case,

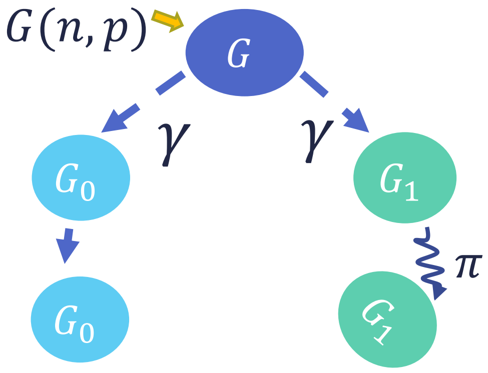
$$\mathbb{V}\left[cor_{K_3}(G_0, G_1)\right]^{1/2} \approx (p\gamma n)^3$$



"independent trials"

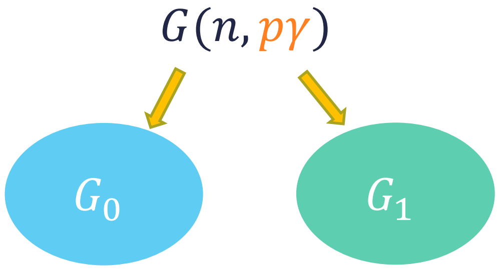
Suppose we had T "independent trials":

$$cor_T(G_0, G_1) = \frac{1}{T}\sum_{i=1}^{T} cor_{K_3}^{(i)}(G_0, G_1)$$



structured

$$\mathbb{E}[cor_T(G_0, G_1)] \approx (p\gamma n)^6 + (\gamma^2 pn)^3$$



null

$$\mathbb{E}[cor_T(G_0, G_1)] \approx (p\gamma n)^6$$

$$\mathbb{V}[cor_T(G_0, G_1)]^{1/2} \approx \frac{1}{\sqrt{T}}(p\gamma n)^3$$

if $T > 1/\gamma^6$,
cor_T is a good test

near-independent subgraphs

~~"independent trials"~~

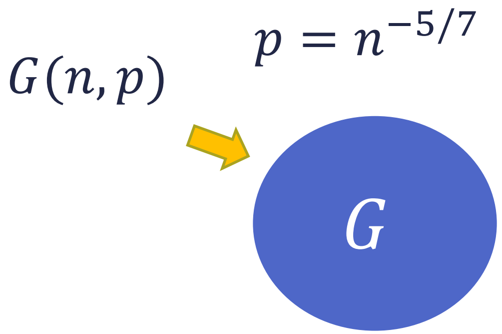
Suppose we had T "independent" *subgraphs*:

$$cor_T(G_0, G_1) = \frac{1}{T}\sum_{i=1}^{T} cor_{H_i}(G_0, G_1)$$

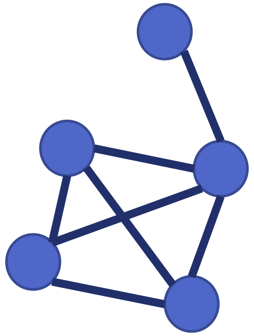
$H_1, \ldots, H_T$

what properties must $H_1, \ldots, H_T$ have to be "independent"?

surprisingly delicate (concentration)



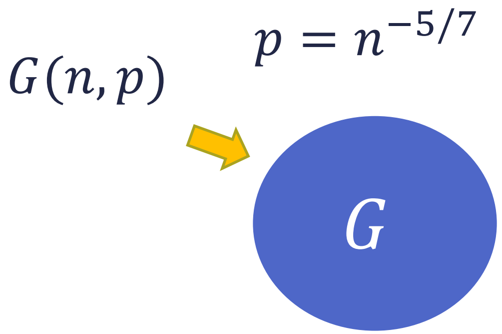
H

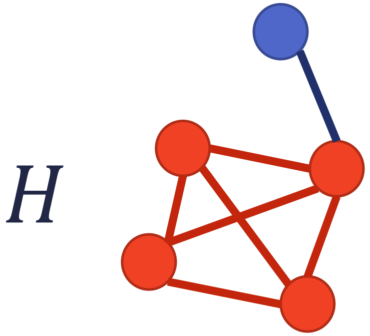


How many labeled copies of H in G?

$$\mathbb{E}[\#_H(G)] = \frac{5!}{|aut(H)|} \cdot \binom{n}{5} \cdot p^7 \approx n^5 p^7 = \Theta(1)$$

surprisingly delicate (concentration)





$\#_H(G)$ does not concentrate!

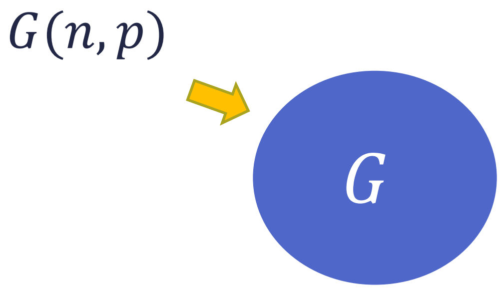
How many labeled copies of H in G?

$$\mathbb{E}[\#_H(G)] = \frac{5!}{|aut(H)|} \cdot \binom{n}{5} \cdot p^7 \approx n^5 p^7 = \Theta(1)$$

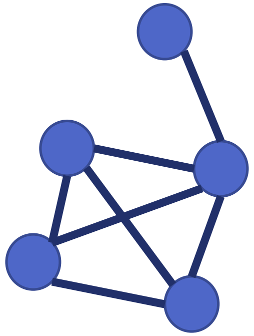
How many labeled copies of K_4 in G?

$$\mathbb{E}[\#_{K_4}(G)] = \frac{4!}{|aut(K_4)|} \cdot \binom{n}{4} \cdot p^6 \approx n^4 p^6 = \Theta(n^{-2/7})$$

variance of subgraph counts



H



Lemma

For a constant-sized subgraph H,

$$\mathbb{V}[\#_H(G)] = \Theta(1) \cdot \frac{\mathbb{E}[\#_H(G)]^2}{\min_{J \subset H} \mathbb{E}[\#_J(G)]}$$

subgraph of H with fewest expected appearances

strict balance

H is *strictly balanced* if all its strict subgraphs have edge density $< \frac{|E(H)|}{|V(H)|}$.

if $\mathbb{E}[\#_H(G)] \approx n^{|V(H)|} p^{|E(H)|} = \Theta(1)$,

then $\mathbb{E}[\#_J(G)] = \omega(1)$ for any $J \subset H$.

Lemma

For a constant-sized subgraph H,

$$\mathbb{V}[\#_H(G)] = \Theta(1) \cdot \frac{\mathbb{E}[\#_H(G)]^2}{\min_{J \subset H} \mathbb{E}[\#_J(G)]} = o(1) \cdot \mathbb{E}[\#_H(G)]$$

concentration AND independence

If $H_1, \dots, H_T$ are non-isomorphic strictly balanced graphs with $\mathbb{E}[\#_{H_i}(G)] = \Theta(1)$,

their counts concentrate

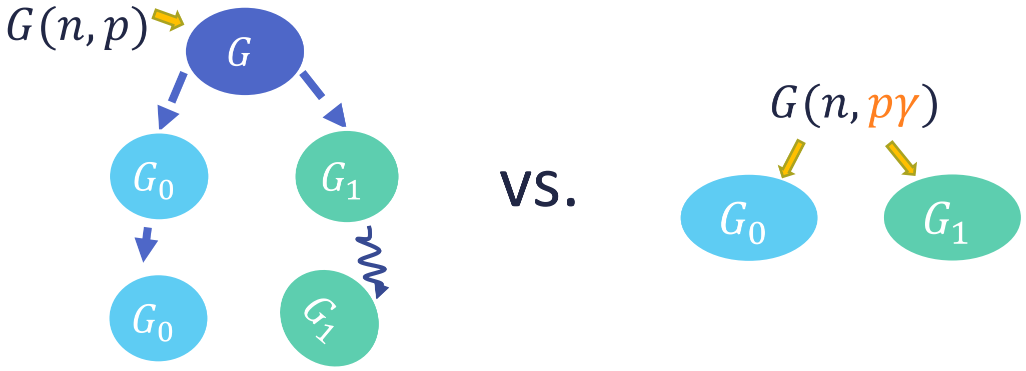
$$\forall i \in [T], \qquad \mathbb{V}[\#_{H_i}(G)] = o(1) \cdot \mathbb{E}[\#_{H_i}(G)]$$

their counts are asymptotically independent

$\forall i \neq j \ \in [T]$,

$$\mathbb{E}\left[\#_{H_i}(G) \cdot \#_{H_j}(G)\right] = (1 + o(1)) \cdot \mathbb{E}[\#_{H_i}(G)] \cdot \mathbb{E}\left[\#_{H_j}(G)\right]$$

distinguishing algorithm

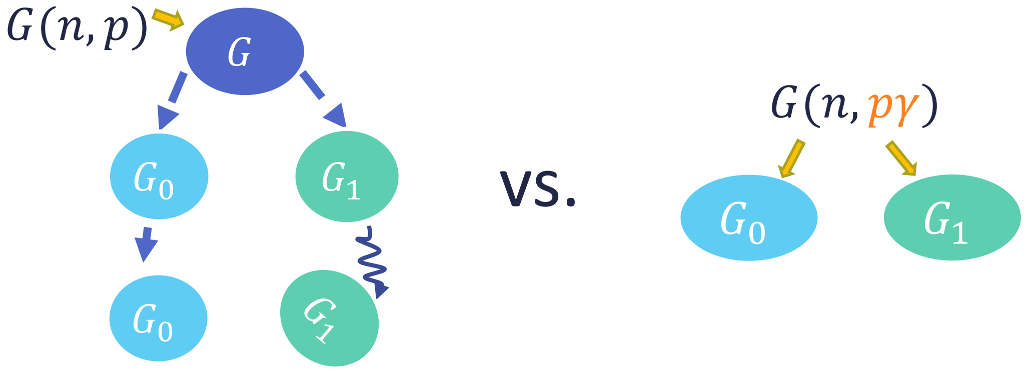


For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a "test set" $H_1, \ldots, H_T$

set $n^v (p\gamma)^e \approx 1$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

distinguishing algorithm



For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a "test set" $H_1, \ldots, H_T$

set $n^v(p\gamma)^e \approx 1$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

compute $\boxed{cor_T(G_0, G_1) = \frac{1}{T}\sum_{i=1}^{T} cor_{H_i}(G_0, G_1)}$

$\geq \theta$ → **structured**

$< \theta$ → **null**

$$\mathbb{E}[cor_T(G_0, G_1)] = \begin{cases} n^{2v}(\gamma p)^{2e} + n^v(\gamma^2 p)^e & \textbf{structured} \\ n^{2v}(\gamma p)^{2e} & \textbf{null} \end{cases}$$

TODO: variance in structured case.

$$\mathbb{V}[cor_T(G_0, G_1)] = \frac{1}{\sqrt{T}} n^v(\gamma p)^e < n^v(\gamma^2 p)^e \quad \textbf{null}$$

outline

- distinguishing/hypothesis testing
- recovery
- concluding

outline

- distinguishing/hypothesis testing
- test graphs
- recovery
- concluding

designing a "test set"

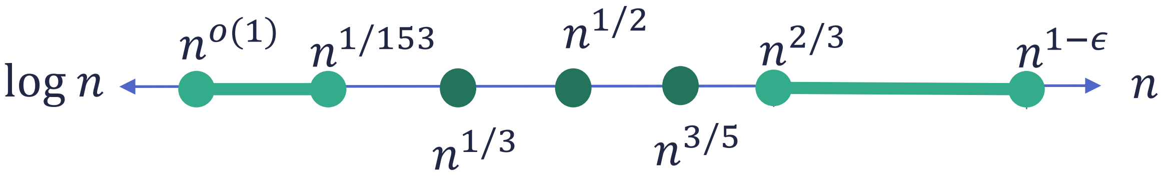
For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a "test set" $H_1, \dots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

remember?

$G(n,p)$ **average degree:**



designing a "test set"

For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a "test set" $H_1, \dots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

designing a “test set”

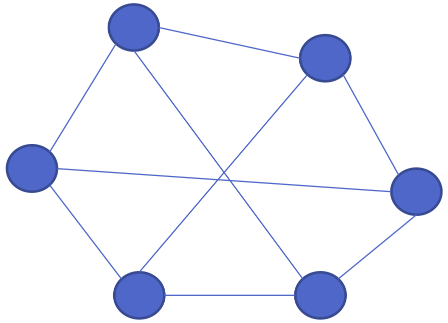
For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a “test set” $H_1, \ldots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

claim: connected d-regular graphs are strictly balanced.

proof: in any strict subgraph, average degree $< d$.



designing a “test set”

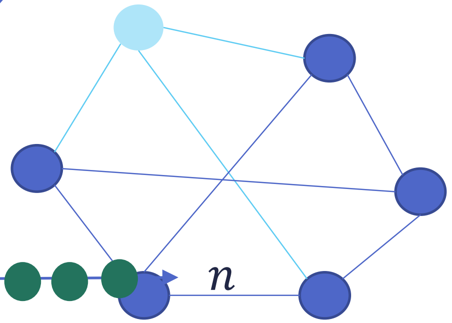
For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a “test set” $H_1, \dots, H_T$

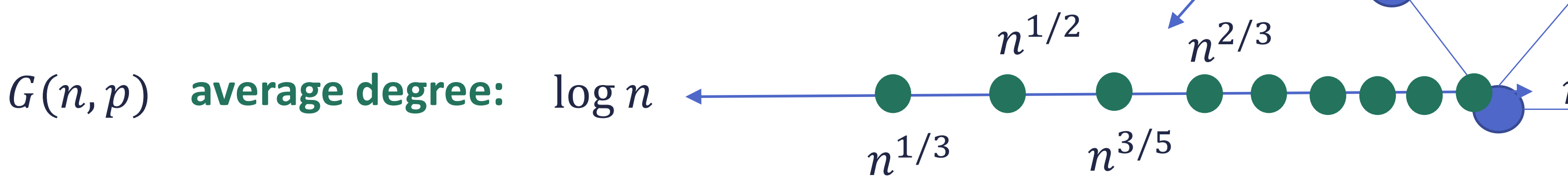
of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

claim: connected d-regular graphs are strictly balanced.

proof: in any strict subgraph, average degree $< d$.





“test set” for non-integer degrees

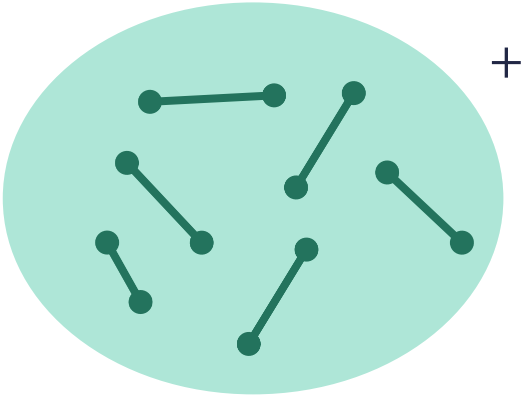
For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a “test set” $H_1, \dots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

what if we want $2 \cdot \frac{e}{v} = \lambda \cdot (d+1) + (1-\lambda) \cdot d$?

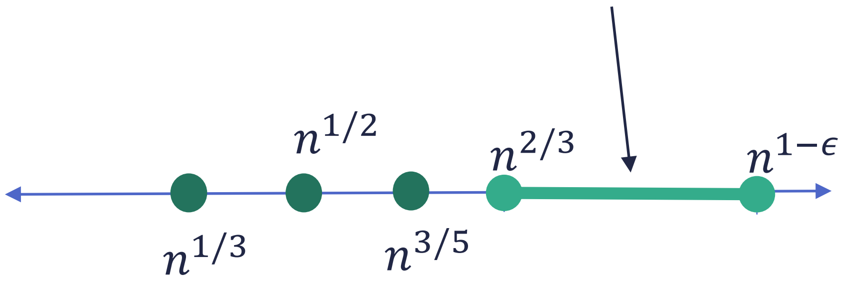
d-regular random graph on v vertices



+ random matching on λv vertices

strict balance? expansion.

$G(n,p)$ **average degree:** $\log n$



“test set” for non-integer degrees

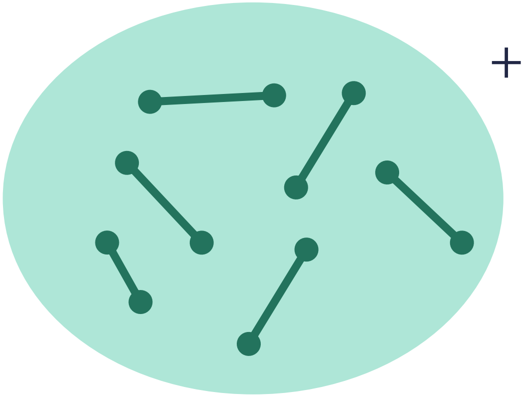
For $v = \frac{1}{\text{poly}(\gamma)}$, design a “test set” $H_1, \ldots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

what if we want $2 \cdot \frac{e}{v} = \lambda \cdot (d+1) + (1-\lambda) \cdot d$?

d-regular random graph on v vertices



+ random matching on λv vertices

strict balance? expansion.

$d < 3$?

2-regular graphs don’t expand.

“test set” for non-integer degrees < 3

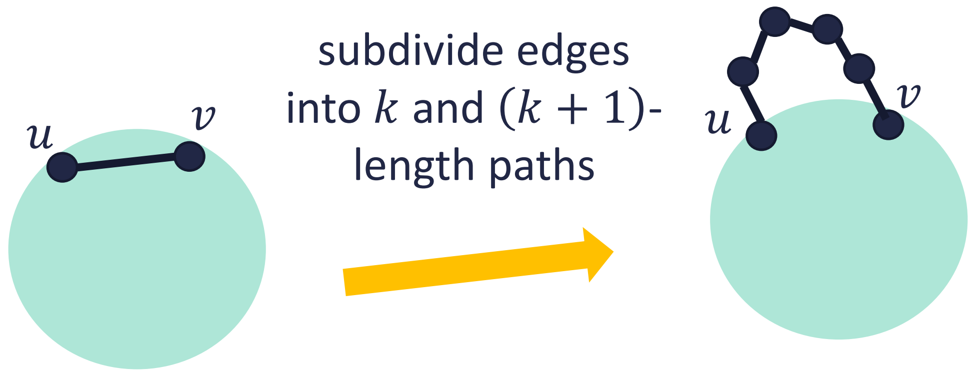
For $v = \frac{1}{\mathrm{poly}(\gamma)}$, design a “test set” $H_1, \ldots, H_T$

of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

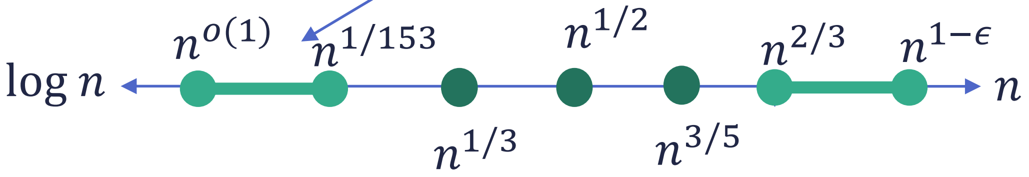
what if we want $2 \cdot \frac{e}{v} = \lambda \cdot 3 + (1 - \lambda) \cdot 2$?

3-regular random graph on λv vertices



strict balance? expansion.

$G(n, p)$ **average degree:**



designing a “test set”

For $v = \frac{1}{\text{poly}(\gamma)}$, design a “test set” $H_1, \ldots, H_T$

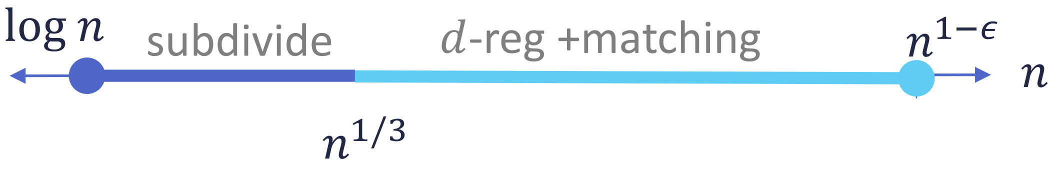
of $T = v^{\Omega(e)}$ strictly balanced graphs w/ v vertices & e edges.

set $n^v (p\gamma)^e \approx 1$

+ more conditions (for recovery)

Conjecture: our construction achieves all $\frac{e}{v}$

$G(n,p)$ **average degree:**



outline

- distinguishing/hypothesis testing
- test graphs
- recovery
- concluding

outline

- distinguishing/hypothesis testing
- test graphs
- recovery
- concluding

distinguishing $\neq$ recovery

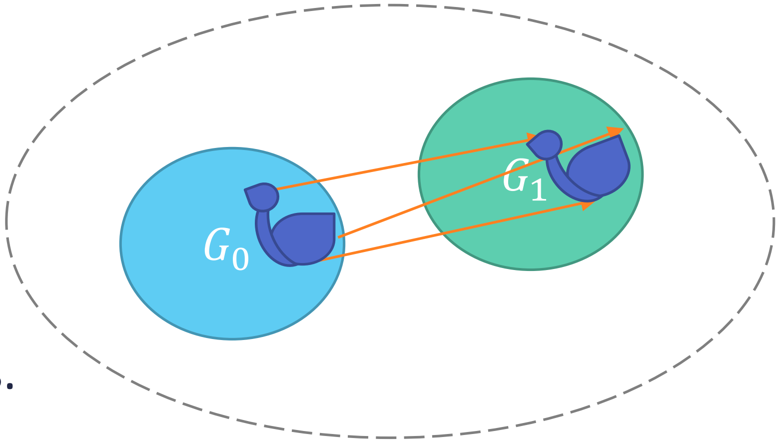
distinguishing: counting subgraphs

⬇ ambiguity in matching; how to conclude $\pi(u) = v$?

distinguishing: subgraphs on $\frac{1}{\mathrm{poly}(\gamma)} = O(1)$ vertices, each appearing $O(1)$ times

⬇ only $O(1)$ vertices participate in subgraphs from our test set.

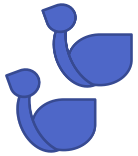
the “black swan” approach



identify rare subgraphs appearing in both graphs, and match vertices.

expected number of  that survive subsampling

choose test set $H_1, \dots, H_T$ so that $(\gamma^2 p)^e n^v \gg (\gamma p)^{2e} n^{2v}$,

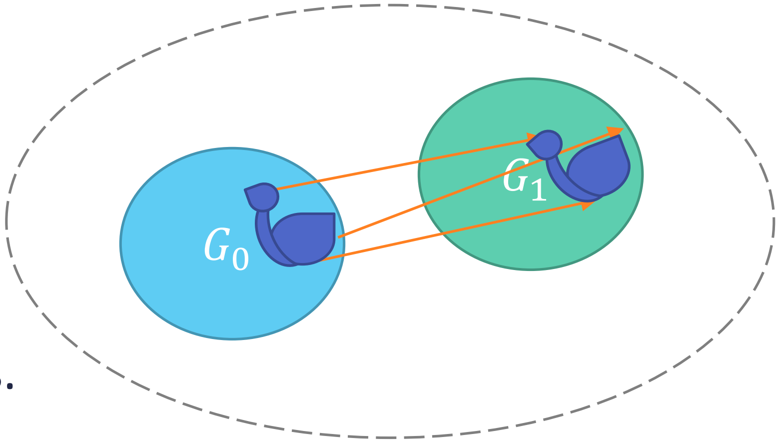
 expected number of unrelated pairs of

if we see H_i in both graphs, it is most likely because of correlation.

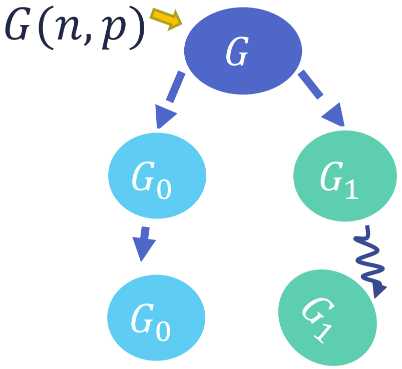
choose large test set $H_1, \dots, H_T$ with $v = O(\log n)$ vertices

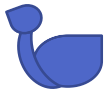
$\Omega(n)$ vertices participate in subgraphs from our test set.

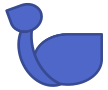
the "black swan" approach



identify rare subgraphs appearing in both graphs, and match vertices.



Claim: there is at most one copy of each  in G with high probability

Claim: $\Omega(n)$ vertices in $G_0 \cap G_1$, appear in a surviving subsampled  with high probability

proofs: second moment method

outline

- distinguishing/hypothesis testing
- test graphs
- recovery
- concluding

outline

- distinguishing/hypothesis testing
- test graphs
- recovery
- concluding

why subgraph counts/statistics?

emerging intuition/conjectures: **SoS $\equiv_{avg}$ low-degree polynomials**

the sum-of-squares (SoS) semidefinite program is at most as powerful as “low-degree” statistics for average-case problems.

known to hold for: planted clique [Barak-Hopkins-Kelner-Kothari-Moitra-Potechin’16]

CSP refutation [Grigoriev’01, Schoenebeck’08, Kothari-Mori-O’Donnell-Witmer’17]

tensor PCA [Hopkins-Kothari-Potechin-Raghavendra-**S**-Steurer’17]

also known: SoS is at most as powerful as “low-degree” spectral algorithms for average-case problems [Hopkins-Kothari-Potechin-Raghavendra-**S**-Steurer’17]

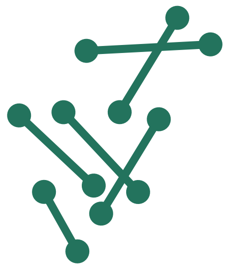
does SoS *know* about the black swans?

does the natural SoS relaxation recover π?

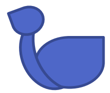
$$\max_{\pi} \langle A_{G_0}, \pi(A_{G_1}) \rangle$$

SoS relaxation

cares about



or



?

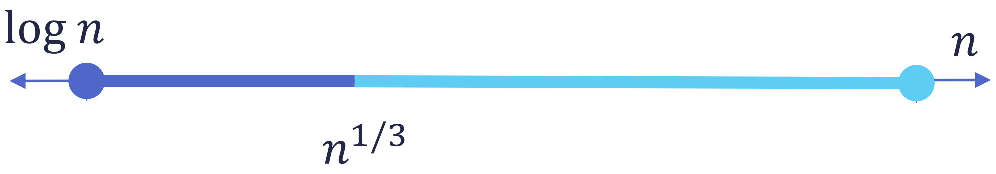
can ask similar questions about other low-degree functions,
e.g. non-backtracking random walk matrix.

more questions

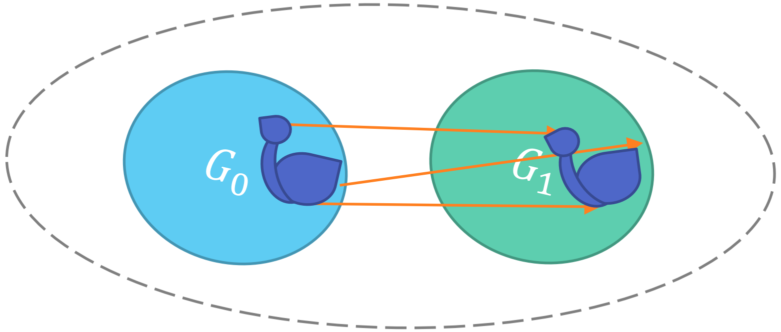
- recovery in polynomial time?

  SoS? or, many variations on our theme are possible.

- all information-theoretically possible $p \in \left[\frac{\log n}{n}, O(1)\right]$?



- practical heuristics?



Thank you!