(Intro to) Average-Case Reductions for Statistics Problems

Guy Bresler

FODSI Summer School

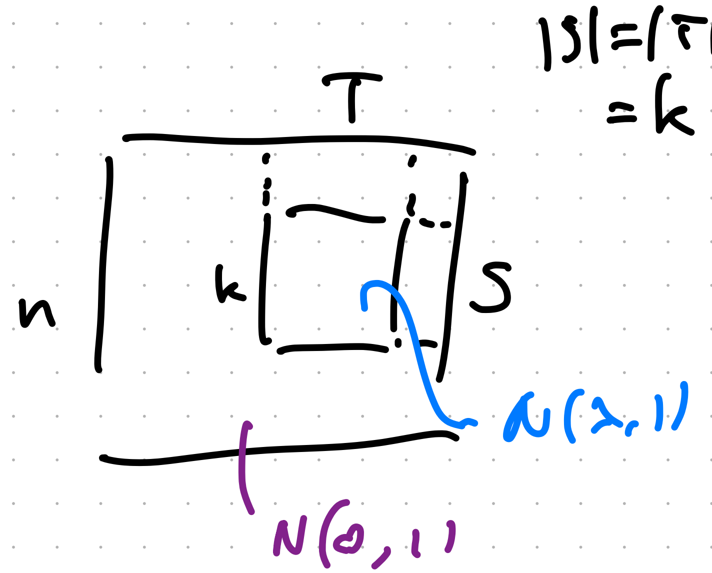
Data problem example. Biclustering. [MaWu14]

$BC(n,k,\lambda)$. Observe: $X \in \mathbb{R}^{n \times n}$. Decide between

H_0: $X_{ij} \overset{iid}{\sim} N(0,1)$

H_1: sample $S, T \sim \mathrm{Unif}\binom{[n]}{k}$

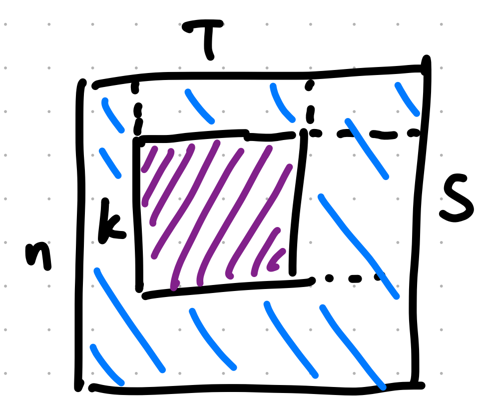
$$X_{ij} \sim \begin{cases} N(\lambda,1) & \text{if } i \in S,\ j \in T \\ N(0,1) & \text{otherwise} \end{cases}$$



Data problem example. Biclustering. [MaWu14]

$BC(n,k,\lambda)$. Observe: $X \in \mathbb{R}^{n \times n}$. Decide between

H_0: $X_{ij} \overset{iid}{\sim} N(0,1)$



H_1: sample $S,T \sim \mathrm{Unif}\binom{[n]}{k}$

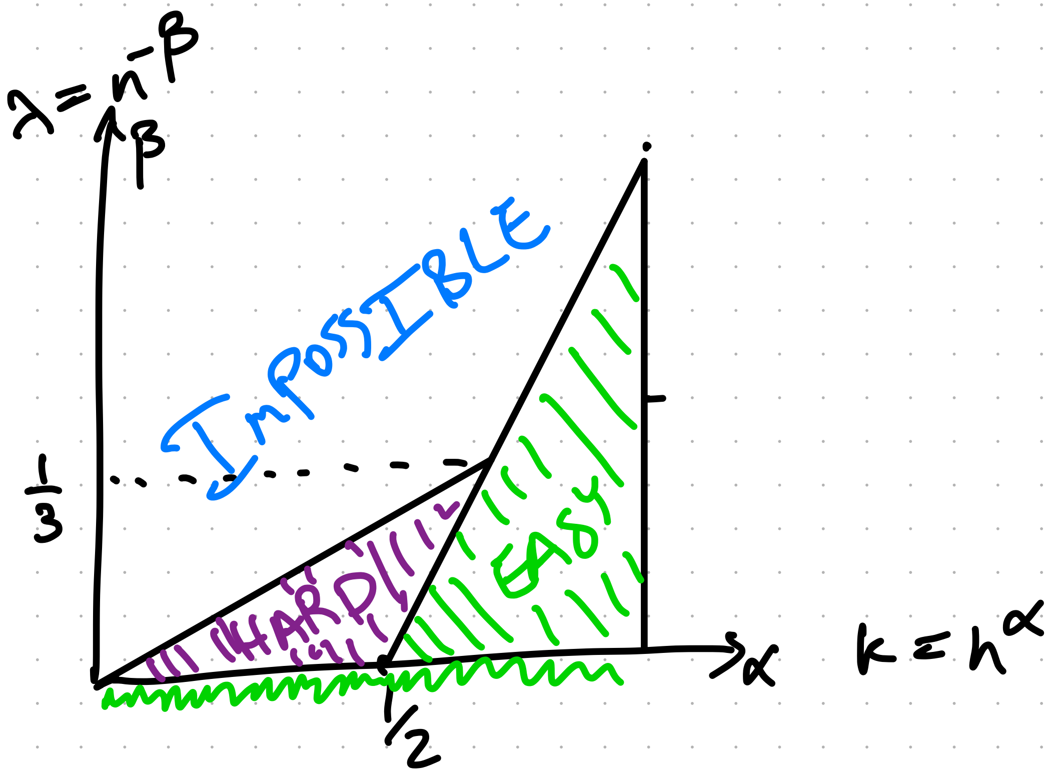
$$X_{ij} \sim \begin{cases} N(\lambda,1) & i\in S,\ j\in T \\ N(0,1) & \text{otherwise} \end{cases}$$

Task: Find eff. comp. $\Phi : \mathbb{R}^{n\times n} \to \{0,1\}$ s.t.

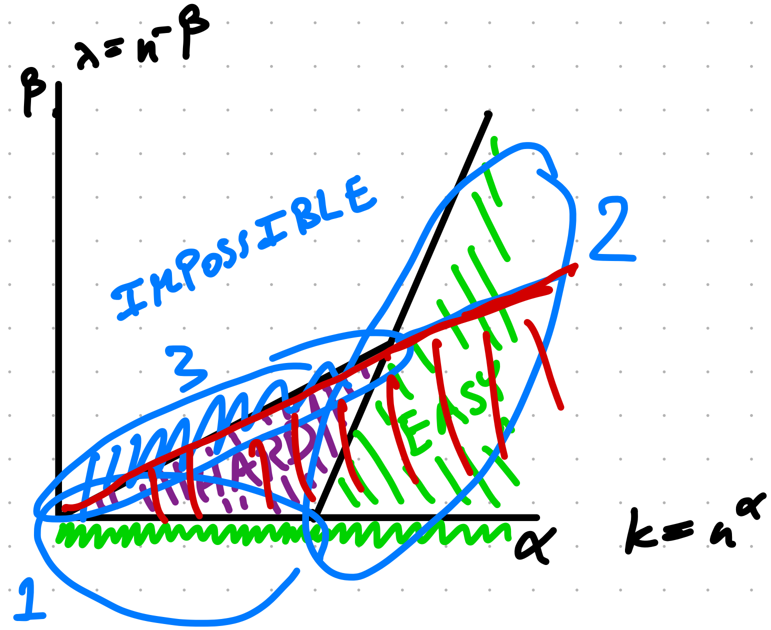
$$\mathbb{P}_{H_0}(\Phi(X)=1) + \mathbb{P}_{H_1}(\Phi(X)=0) \to 0.$$

Note: different problem for each (n, k, λ)

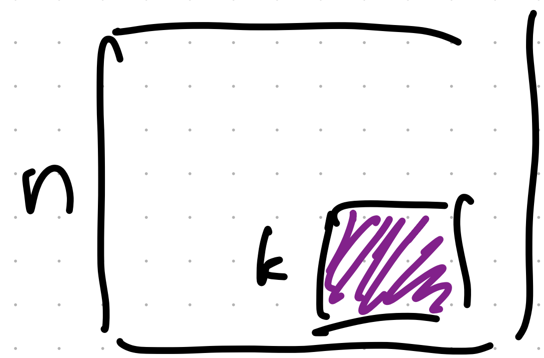
For what (n, k, λ) is problem feasible?



Efficient Algorithms?



1. $\beta < 0$ $\quad \lambda = n^{-\beta} = \mathrm{poly}(n)$. Entries are <u>large</u>.



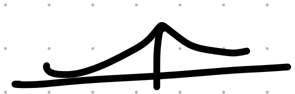
$$T_{max} := \max_{ij} X_{ij}$$

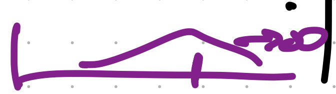
<u>Lemma</u>: (H_0) If $X_{ij} \sim N(0,1)$ then $T_{max} \leq \sqrt{4 \log n}$ w.h.p.

<u>Lemma</u>: (H_1) If $X_{ij} \sim N(\lambda, 1)$ $i \in S, j \in T$ then $T_{max} \geq \lambda$ w.h.p. $= \mathrm{poly}(n)$

2. Many slightly large entries... $\quad T_{avg} = \frac{1}{n} \sum_{ij} X_{ij}$

<u>Lemma</u> (H_0) $T_{avg} \sim N(0,1)$



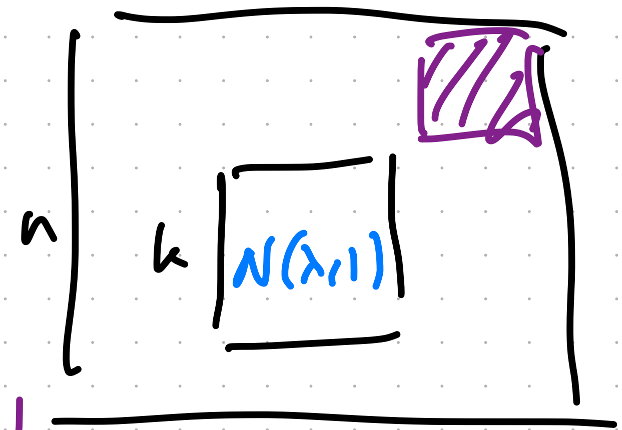


<u>Lemma</u> (H_1): $T_{avg} \sim N\left(\frac{k^2 \lambda}{n}, 1\right)$

$\Rightarrow$ If $\beta < 2\alpha - 1$ then $\frac{k^2 \lambda}{n} \to \infty$.

Algorithm achieving info limit (inefficiently)

3. $$T_{search} = \max_{|S|=|T|=k} \underbrace{\frac{1}{k} \sum_{\substack{i \in S \\ j \in T}} X_{ij}}$$



Lemma (H_0): Under H_0

max of $\approx \binom{n}{k}^2$ $N(0,1)$ random variables.

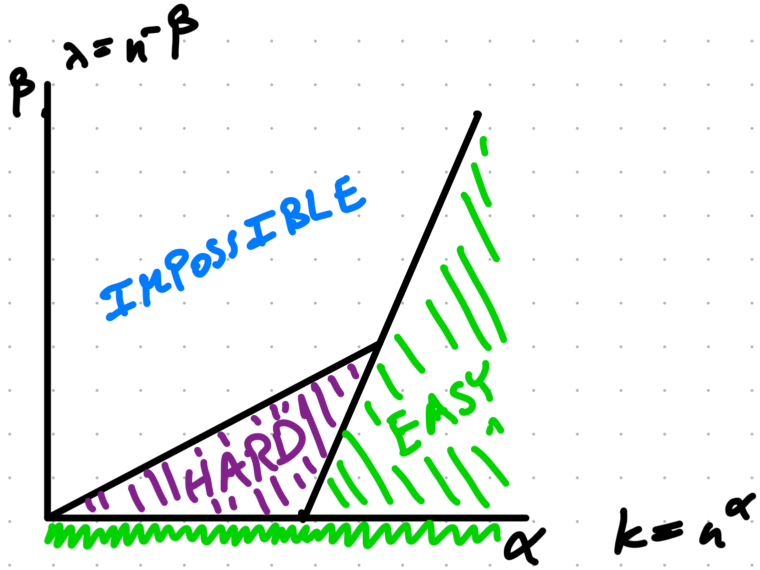
$$T = O\left(\sqrt{k \log n}\right)$$

Lemma (H_1) $\mathbb{E}\, T \geq k \cdot \lambda$

$$T \geq \frac{k \cdot \lambda}{\sqrt{\log n}} \quad \text{w.h.p.}$$

$\Rightarrow$ If $\lambda \gg \frac{\log n}{\sqrt{k}}$ then can solve.

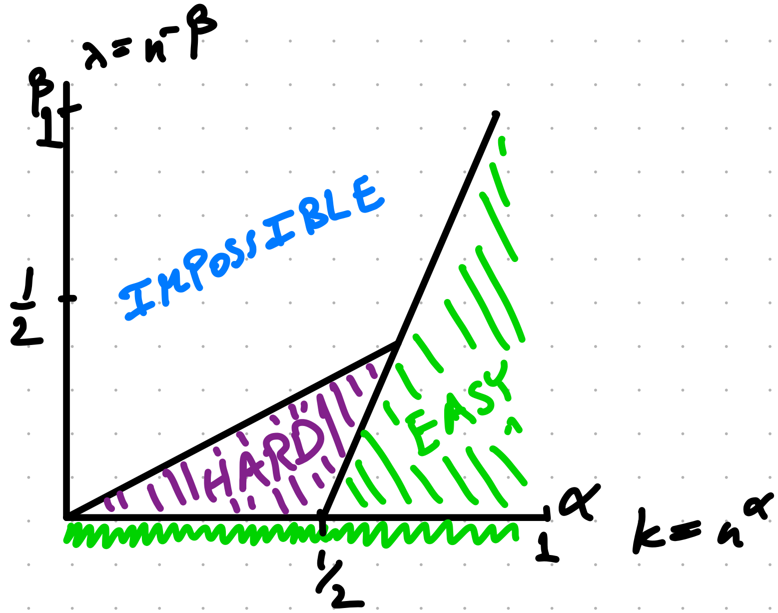
$\beta < \alpha/2$



How to reason about hard region?

Approach: Reductions. E.g. 3-SAT $\longrightarrow$ indep. set.

But: must preserve distn.



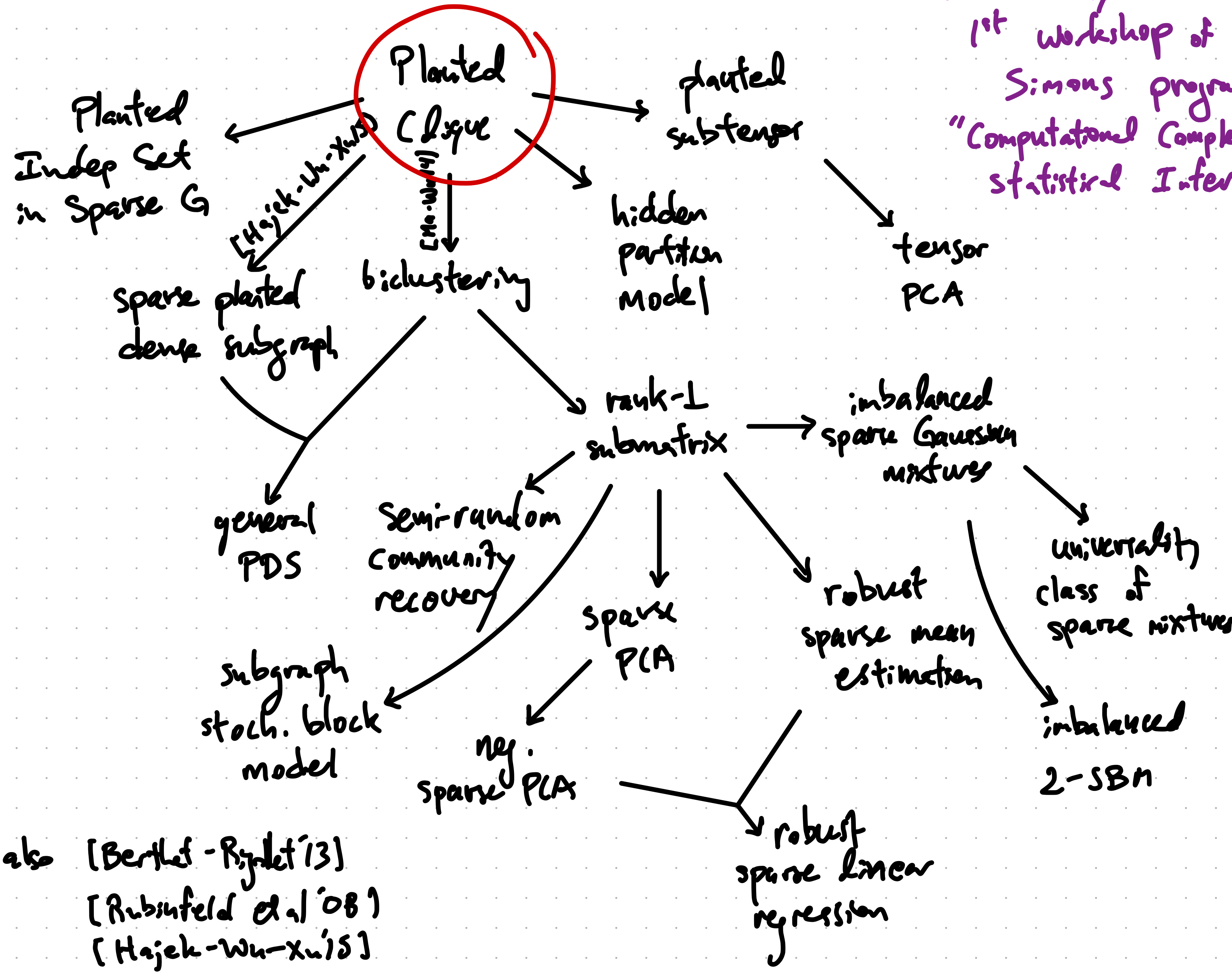
How to reason about hard region?

Approach: reductions... e.g. 3-SAT $\longrightarrow$ INDEP SET

But... we have to preserve distributions.

Web of reductions from joint work with Matthew Brennan

... see my talk in 1st workshop of FA21 Simons program "Computational Complexity of Statistical Inference"



See also [Berthet-Rigollet'13]
[Rubinfeld et al '08]
[Hajek-Wu-Xu'15]
...

What's at the root?

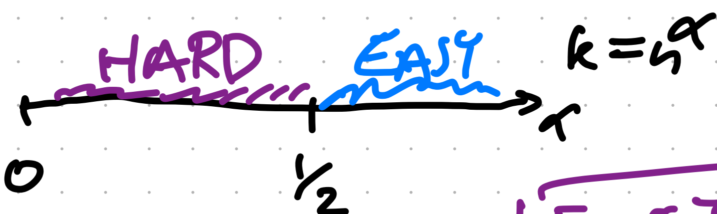
Planted clique. Observe a graph G.

H_0: $G \sim G(n, 1/2)$

H_1: $G \sim PC(n,k)$

Sample $S \sim \mathrm{Unif}\left(\binom{[n]}{k}\right)$

$G \sim G(n, 1/2)$, place a clique on S.



Exercise: IT limit is at $k = 2\log_2 n$

Proposition: If $\alpha > 1/2$, then

$$A(G) = \mathbb{1}\left\{\max \text{degree} \geq \frac{n}{2} + \sqrt{3n\log n}\right\} \text{ succeeds.}$$

Conjecture: No poly-time alg for $\alpha < 1/2$.

Evidence: SoS algos fail [Barak-Hopkins-et al 15]

Definition: (Total Variation)

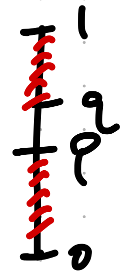
For two probability measures μ and ν (on same space)

$$d_{TV}(\mu,\nu) = \frac{1}{2}\|\mu-\nu\|_1 = \frac{1}{2}\int |f(x)-g(x)|\,dx$$

$$= \sup_{E} \mu(E)-\nu(E)$$

$$= \inf_{\substack{(X,Y)\\ X\sim\mu \\ Y\sim\nu}} \mathbb{P}(X\neq Y)$$

$X\sim \mathrm{Bern}(p)$, $Y\sim \mathrm{Bern}(q)$



Sample $U\sim \mathrm{Unif}(0,1)$

$X = \mathbb{1}_{U\le p}$

$Y = \mathbb{1}_{U\le q}$.

$TV(X,Y) = |p-q|$

Note: Given a sample X, decide between

$H_0: X\sim\mu$

$H_1: X\sim\nu$

$$P_{\text{error}}^{\text{opt}} = \frac{1}{2} - \frac{1}{2}d_{TV}(\mu,\nu)$$

Reductions In TV.

Defn. $X \sim P$, $\mathcal{A}$ an algorithm, write

$$\boxed{P \xrightarrow[\epsilon]{\mathcal{A}} P'} \quad \text{if} \quad d_{TV}(\mathcal{L}(\underline{\mathcal{A}(X)}), P') \le \epsilon$$

$\sim P$

Defn. (Reduction in TV)

Want an efficient algo $\mathcal{A}$

$$P_{H_0} \xrightarrow[\epsilon]{\mathcal{A}} P'_{H_0}$$

$$P_{H_1} \xrightarrow[\epsilon]{\mathcal{A}} P'_{H_1}$$

$\mathcal{A}$ is oblivious to input distn!

Sees $X \in \mathbb{R}^{n \times n}$.

Lemma: If Φ δ-solves P', then $\Phi \circ \mathcal{A}$ $\delta + \epsilon$-solves problem P.

$\Rightarrow$ If P is hard, then P' is hard.

See also [Levin '85]

Useful fact about TV: well-behaved under composition of algorithms.

Suppose

$$\begin{array}{l} P \xrightarrow[\epsilon]{\mathcal{A}} P' \\ \text{ss}_\delta \\ \tilde{P} \end{array} \quad \text{TV} \le \delta \quad \Longrightarrow \quad \tilde{P} \xrightarrow[\epsilon+\delta]{A} P'$$

Lemma: $A = A_2 \circ A_1$

$$P \xrightarrow[\epsilon_1]{A_1} P_1 \xrightarrow[\epsilon_2]{A_2} P_2 \quad \Longrightarrow \quad P \xrightarrow[\epsilon_1+\epsilon_2]{A_2 \circ A_1} P_2$$

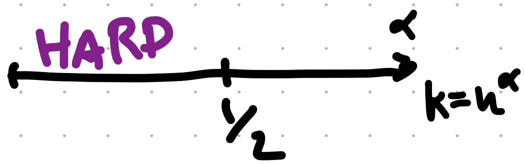
Proof:

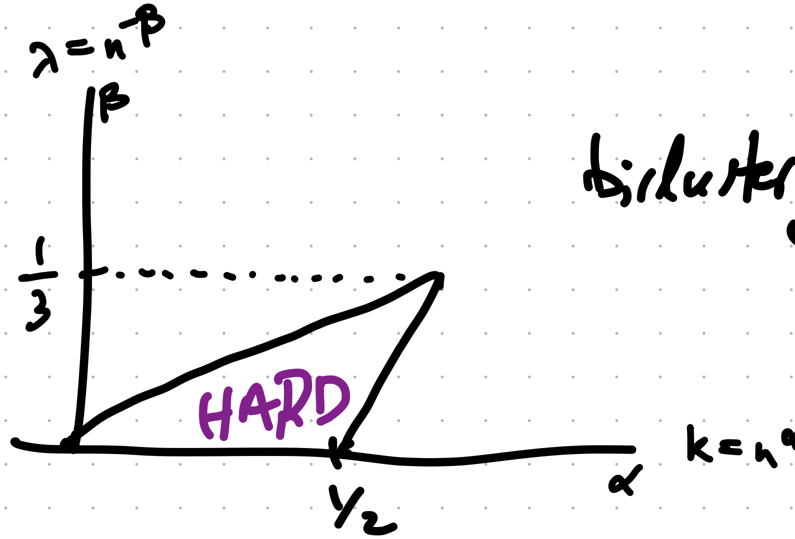
$$\begin{array}{l} P \xrightarrow{A_1} \tilde{P}_1 \overset{A_2 \quad \epsilon_1+\epsilon_2}{\dashrightarrow} P_2 \\ \qquad\quad\;\; \text{ss}_{\epsilon_1} \\ \qquad\quad\;\; P_1 \xrightarrow[A_2]{\epsilon_2} P_2 \end{array}$$

∎

Theorem:

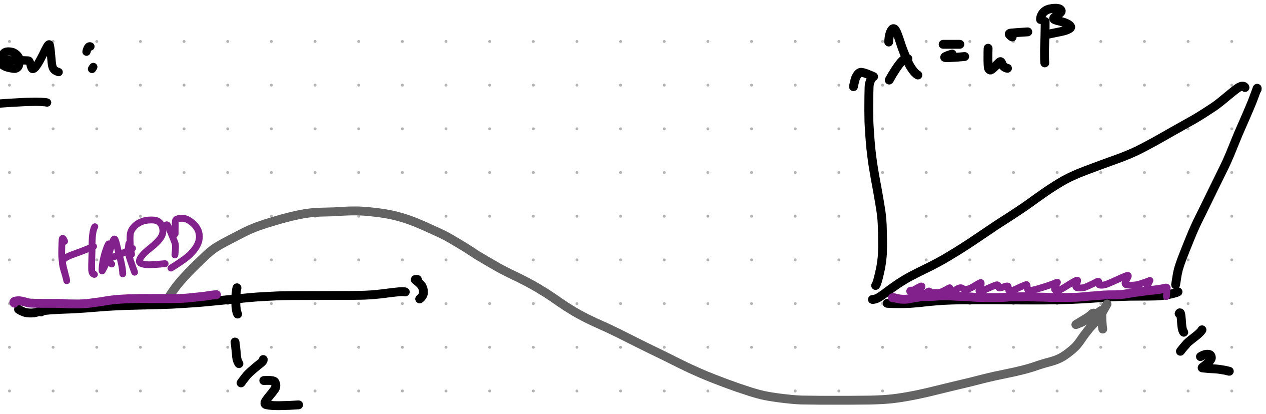
planted clique





biclustering

Proposition:

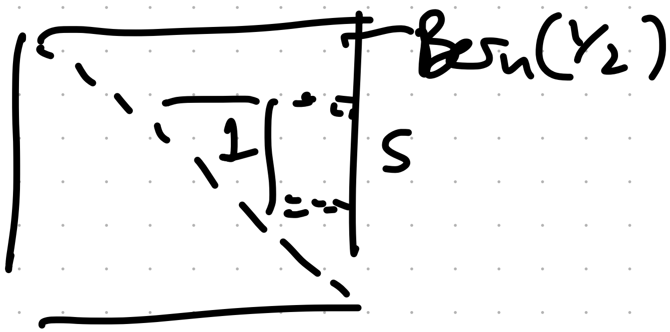


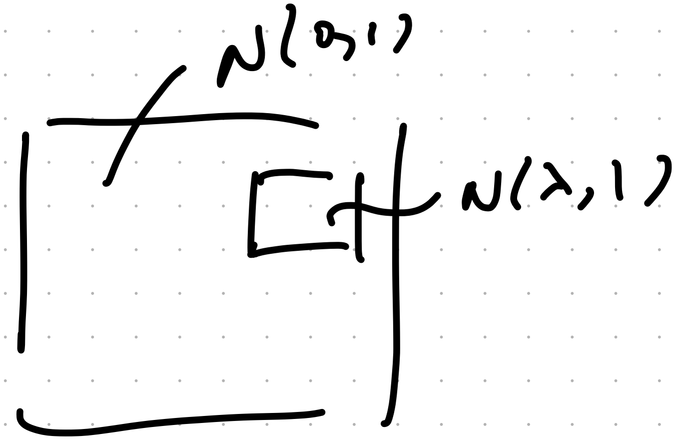
$k = n^{\alpha}$

$\exists\, \mathcal{A}$ s.t. for $0 < \alpha < 1/2$, $k = n^{\alpha}$:

$$PC^{H_i}(n,k) \xrightarrow[o(1)]{\mathcal{A}} BC^{H_i}(n,k,\beta = o(1))$$

$i = 0, 1.$

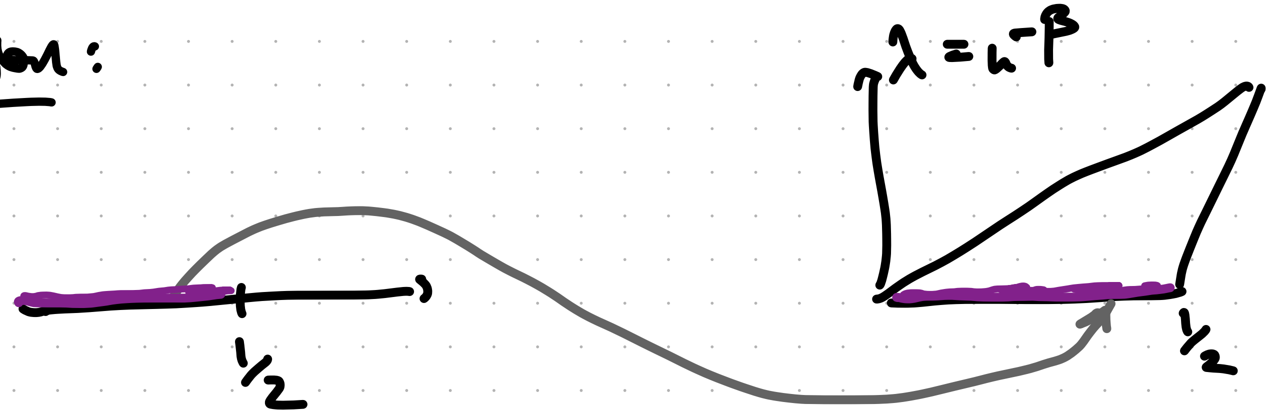




1. Change of distribution
   - $\text{Bern}(1/2) \longrightarrow Q = N(0,1)$
   - $1 \longrightarrow P = N(\lambda, 1)$

2. Add in diagonal and remove symmetry

Proposition:



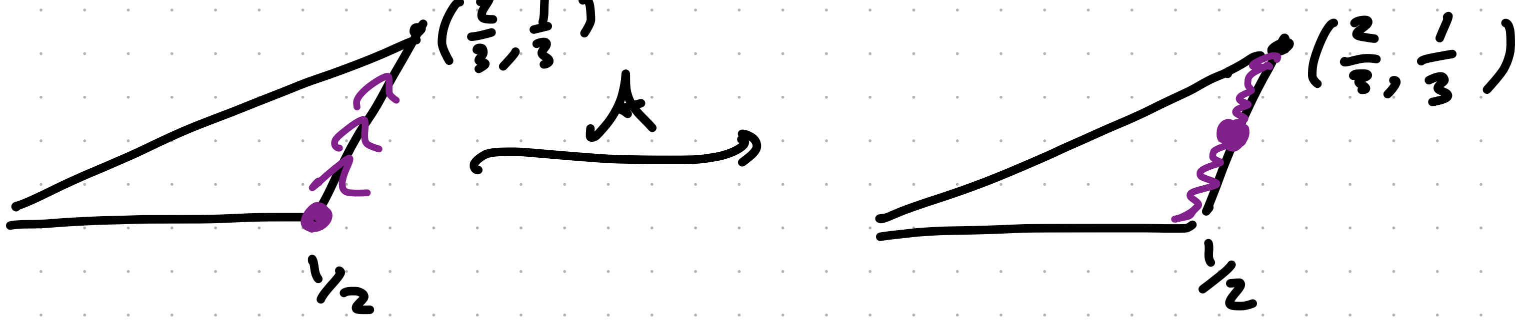
$k = n^{\alpha}$

$\exists\ A$ s.t., for $0<\alpha<1/2$, $k=n^{\alpha}$:

$$PC^{H_i}(n,k) \xrightarrow[o(1)]{A} BC^{H_i}(n,k,\beta=o(1))$$

$i=0,1.$

Proposition: $\exists$ efficient A



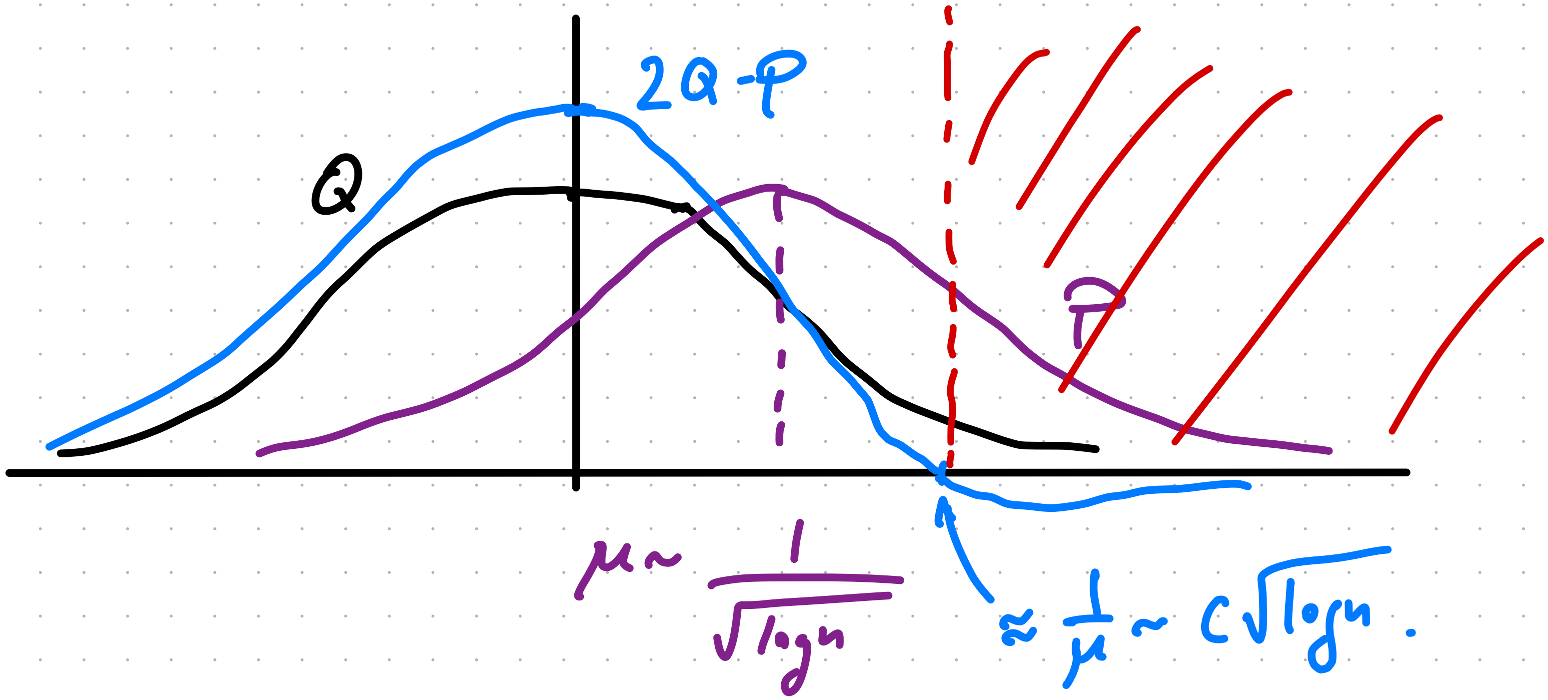
Proof. We'll map

$$0 \mapsto \varphi(0) \sim \nu$$

$$1 \mapsto P$$

$$\mathrm{Bern}(1/2) \longmapsto \begin{cases} \varphi(0) & \text{w.p. } 1/2 \\ \varphi(1) & \text{w.p } 1/2 \end{cases} \sim \frac{1}{2}\nu + \frac{1}{2}P \overset{?}{=} Q$$

$$\Rightarrow \nu = 2Q - P$$



Let $\Omega = \{x : 2Q(x) - P(x) \geq 0\} \subseteq (-\infty, \sqrt{3\log n}\,]$

$$\nu = \frac{2Q-P}{Z} \mathbb{1}_\Omega$$

Lemma: $d_{TV}\Big(\underbrace{\tfrac{1}{2}\nu + \tfrac{1}{2}P}_{\phi(\mathrm{Bern}(1/2))},\ Q\Big) = o(n^{-3})$ if $\mu \leq \frac{1}{\sqrt{6\log n}}$

Key: $Z = 1 + o(n^{-3})$.

Lemma: $X_1 \perp X_2$ and $Y_1 \perp Y_2$ then $d_{TV}((X_1,X_2), (Y_1,Y_2)) \leq d_{TV}(X_1,Y_1) + d_{TV}(X_2,Y_2)$

Removing symmetry?

Lemma (Cloning) There exists an efficient algorithm $\mathcal{C} : \mathbb{R} \to \mathbb{R}^2$ such that

$$N(0,1) \xrightarrow{\mathcal{C}} N(0,1)^{\times 2}$$

$$N(\lambda,1) \xrightarrow{\mathcal{C}} N\Big(\underbrace{\tfrac{\lambda}{\sqrt{2}}}_{},1\Big)^{\times 2}$$

Proof. $\mathcal{C}$:
- Input $X \sim N(\lambda,1) = \lambda + N(0,1)$
- Sample $Z \sim N(0,1)$ indep.
- Output $\frac{1}{\sqrt{2}}\begin{pmatrix} X+Z \\ X-Z \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} X \\ Z \end{pmatrix}$

$$= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}\begin{pmatrix} \lambda \\ 0 \end{pmatrix} + \underbrace{\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}}_{B}\begin{pmatrix} N(0,1) \\ N(0,1) \end{pmatrix} \qquad \Sigma = BB^T = I$$

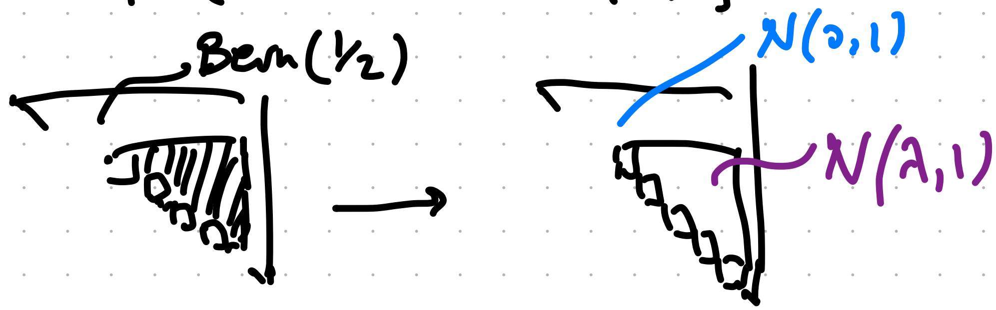
$$= \begin{pmatrix} \lambda/\sqrt{2} \\ \lambda/\sqrt{2} \end{pmatrix} + \begin{pmatrix} N(0,1) \\ N(0,1) \end{pmatrix}$$

$\square$

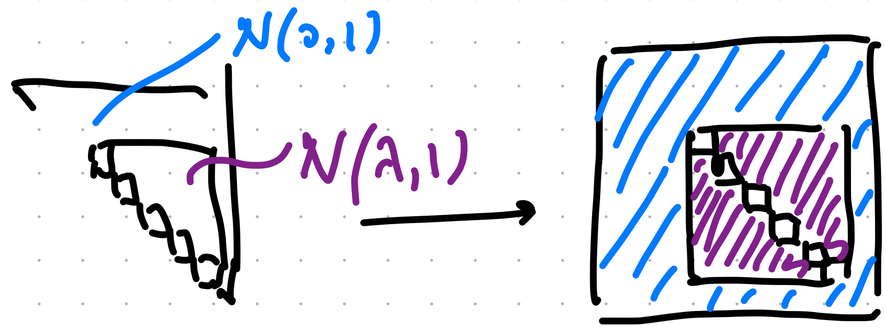
PC to Biclustering.

Input: Adjacency matrix $X \sim PC(n,k)$. $X \in \{0,1\}^{\binom{n}{2}}$

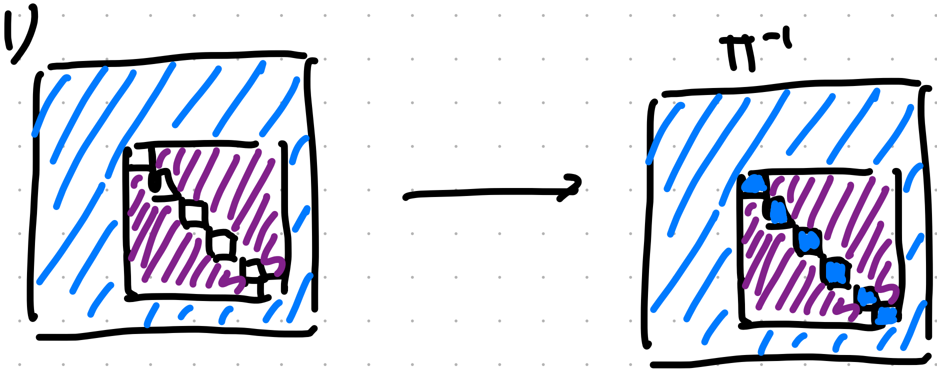
1: map entries using φ



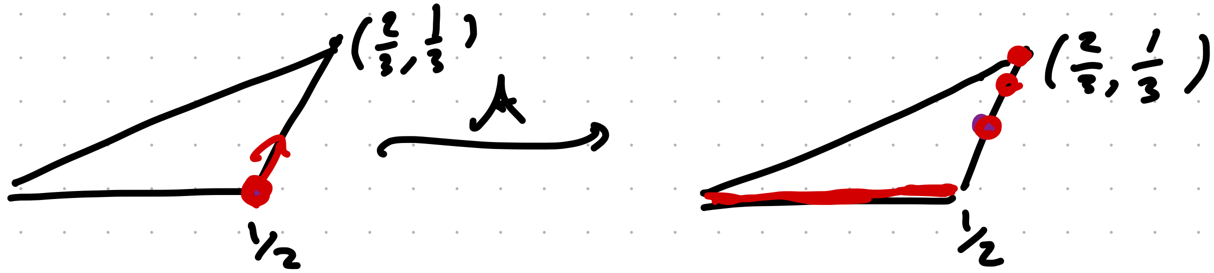
2: clone and symmetrize



3: fill in diagonal with $N(0,1)$
and permute columns



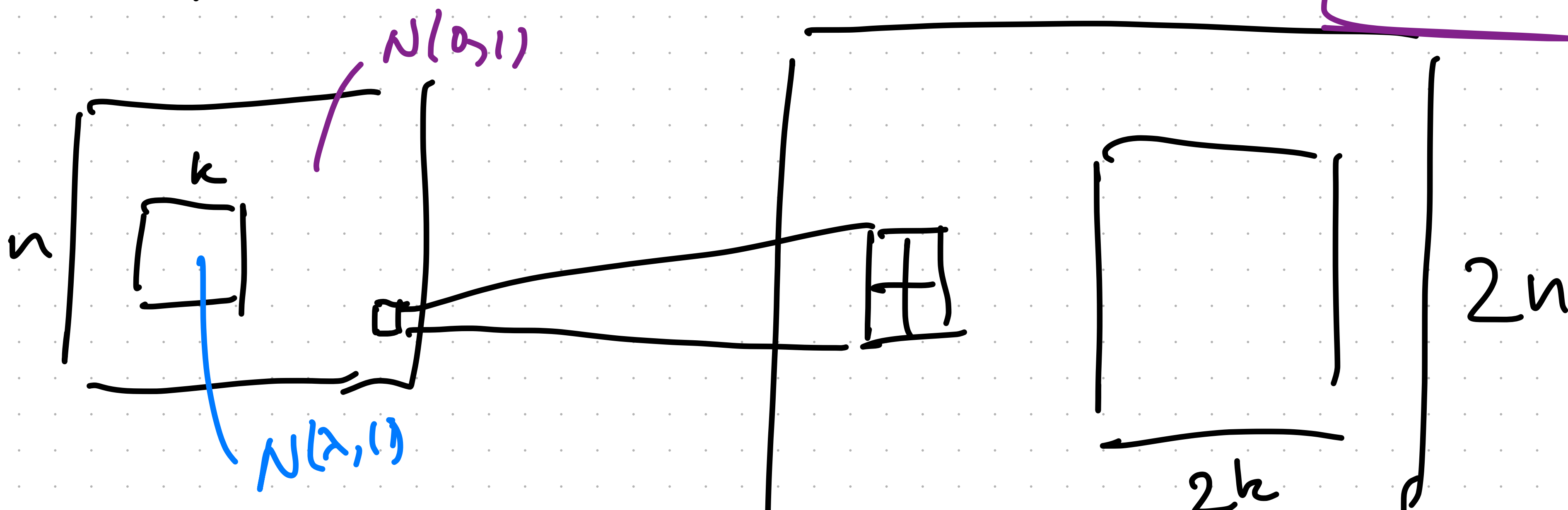
Proposition: $\exists$ efficient A



Grow k and decrease μ (but not too much!)

Proof $\nearrow$ $\frac{k^2 \lambda}{n}$ is kept constant.

$$\boxed{\frac{(2k)^2 \frac{\lambda}{2}}{2n} = k^2 \lambda / n}$$

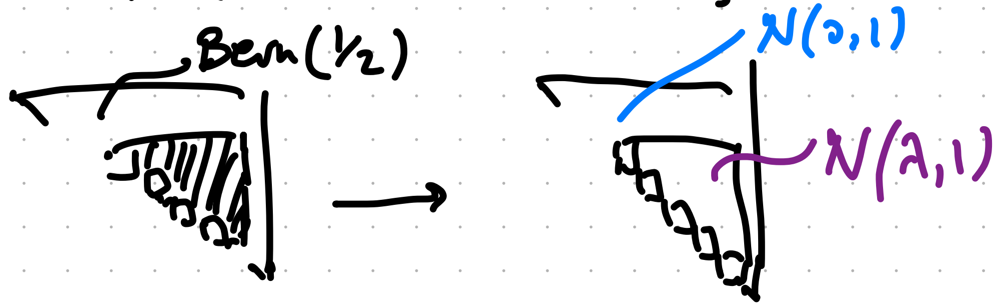


$$\lambda \mapsto \frac{\lambda}{2}$$

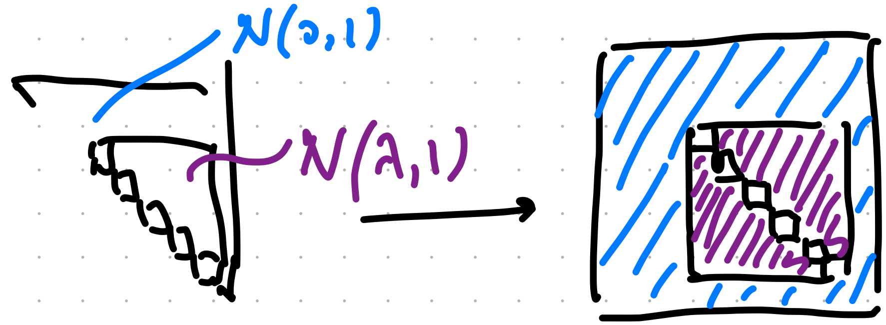
PC TO BICLUSTERING.

Input: Adjacency matrix $X \sim PC(n,k)$. $X \in \{0,1\}^{\binom{n}{2}}$

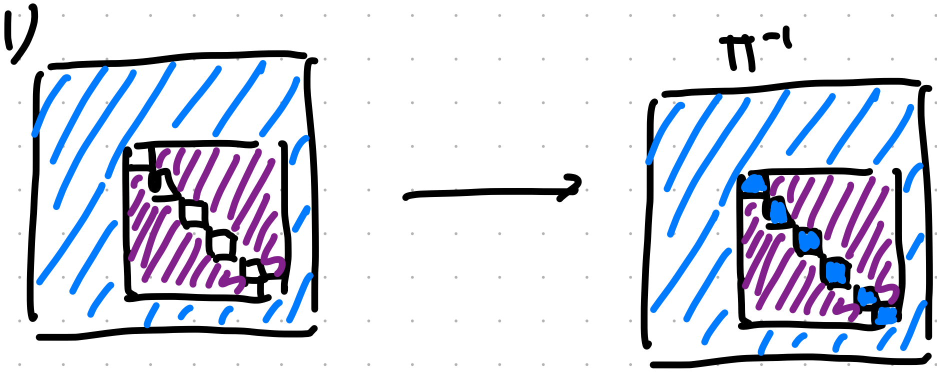
1: map entries using φ



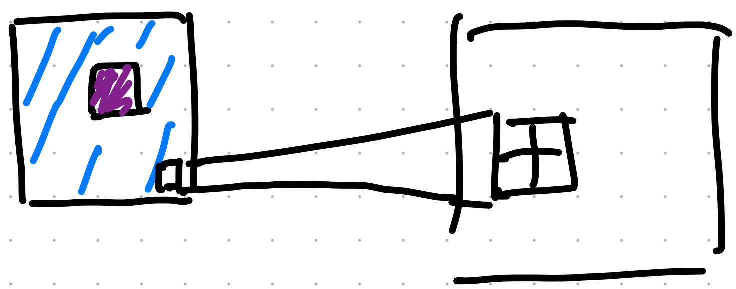
2: clone and symmetrize



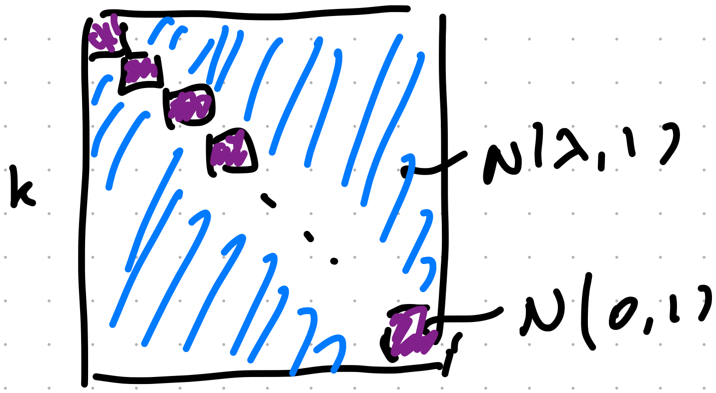
3: fill in diagonal with $N(0,1)$ and permute columns

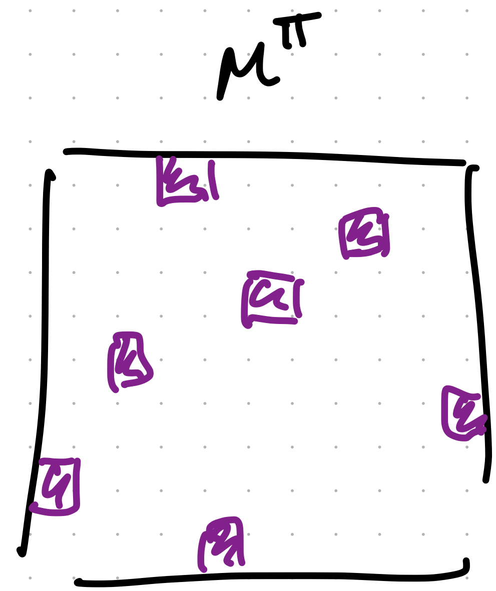


4: clone repeatedly in order to reach hard parameters for $\alpha > 1/2$.



Why filling in diagonal with $N(0,1)$ works under H_1:





<u>Lemma</u>: Let $M \in \mathbb{R}^{m \times m}$ $\quad M_{ij} \sim \begin{cases} Q & i=j \\ P & i \neq j \end{cases}$

and π unif permutation on $[m]$ $\quad M^{\pi}_{ij} = M_{i,\pi(j)}$

Then $d_{TV}(P^{m \times m}, \mathcal{L}(M^{\pi})) = O(\chi^2(P,Q))$

$$= O(\mu^2)$$

$$= O\left(\frac{1}{\log n}\right)$$

"Chi-squared divergence"