Community Detection in Networks: understanding the fundamental limits of polynomial-time algorithms

Jiaming Xu [1]

Joint work with Yudong Chen[2], Bruce Hajek[3], Yihong Wu[3]

[1]Wharton Statistics, University of Pennsylvania

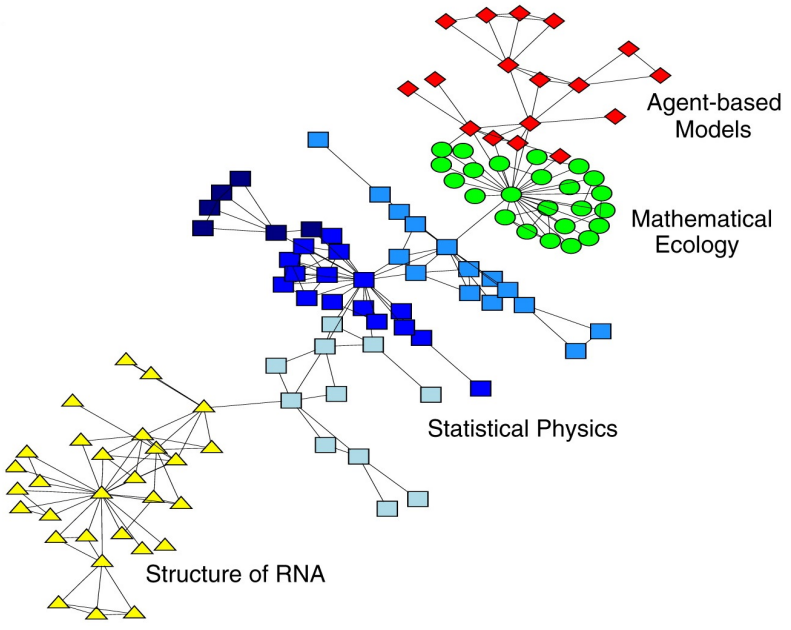
[2]EECS, University of California, Berkeley

[3]ECE, University of Illinois at Urbana-Champaign

April 29, 2015

Community detection in networks

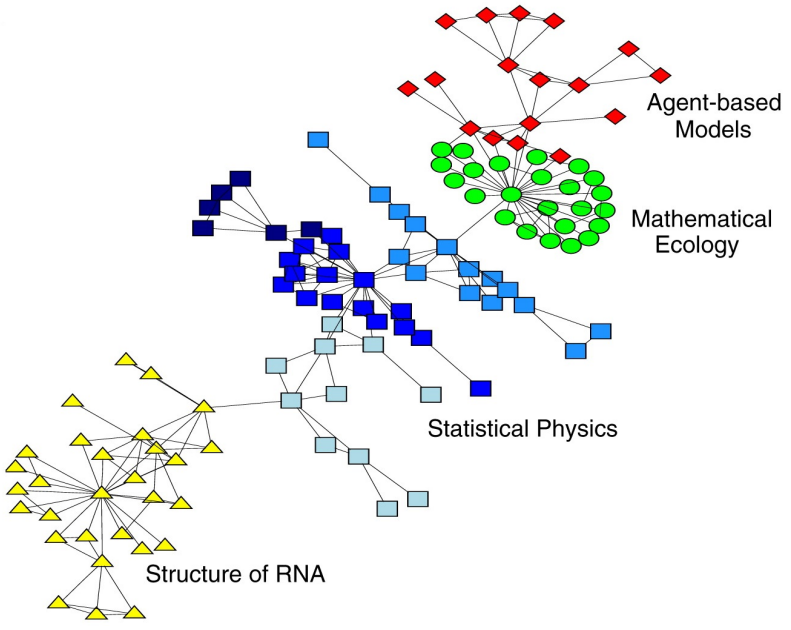
- Networks with community structures arise in many applications



Collaboration network: 118 scientists [Girvan-Newman ’02]

Community detection in networks

- Networks with community structures arise in many applications

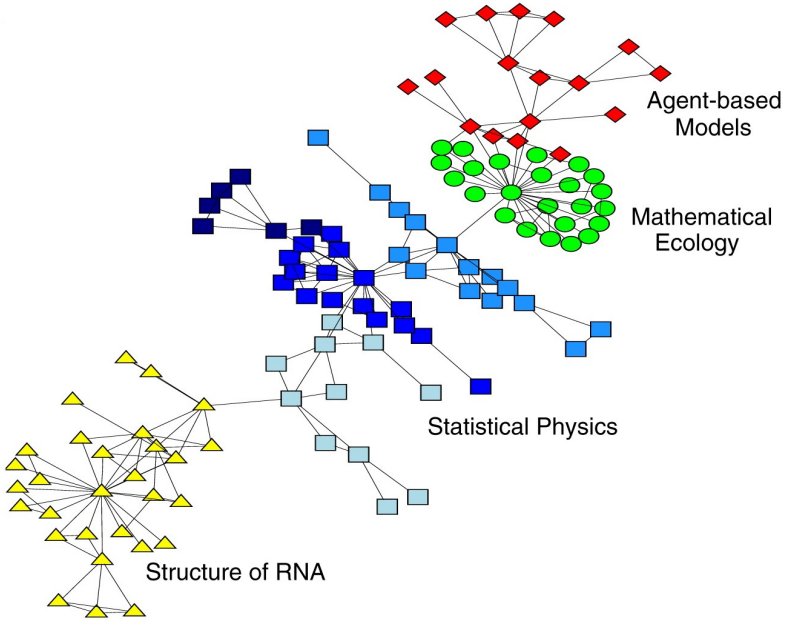


Collaboration network: 118 scientists [Girvan-Newman '02]

- Task: Find underlying communities based on the network topology

Community detection in networks

- Networks with community structures arise in many applications



Collaboration network: 118 scientists [Girvan-Newman '02]

- Task: Find underlying communities based on the network topology
- Applications: Friend or movie recommendation in online social networks

Statistical and computational challenges

- From a statistical perspective
  - A large number of (small) communities
  - The observed network is sparse

Statistical and computational challenges

- From a statistical perspective
  - A large number of (small) communities
  - The observed network is sparse
- From a computational perspective
  - Large solution space

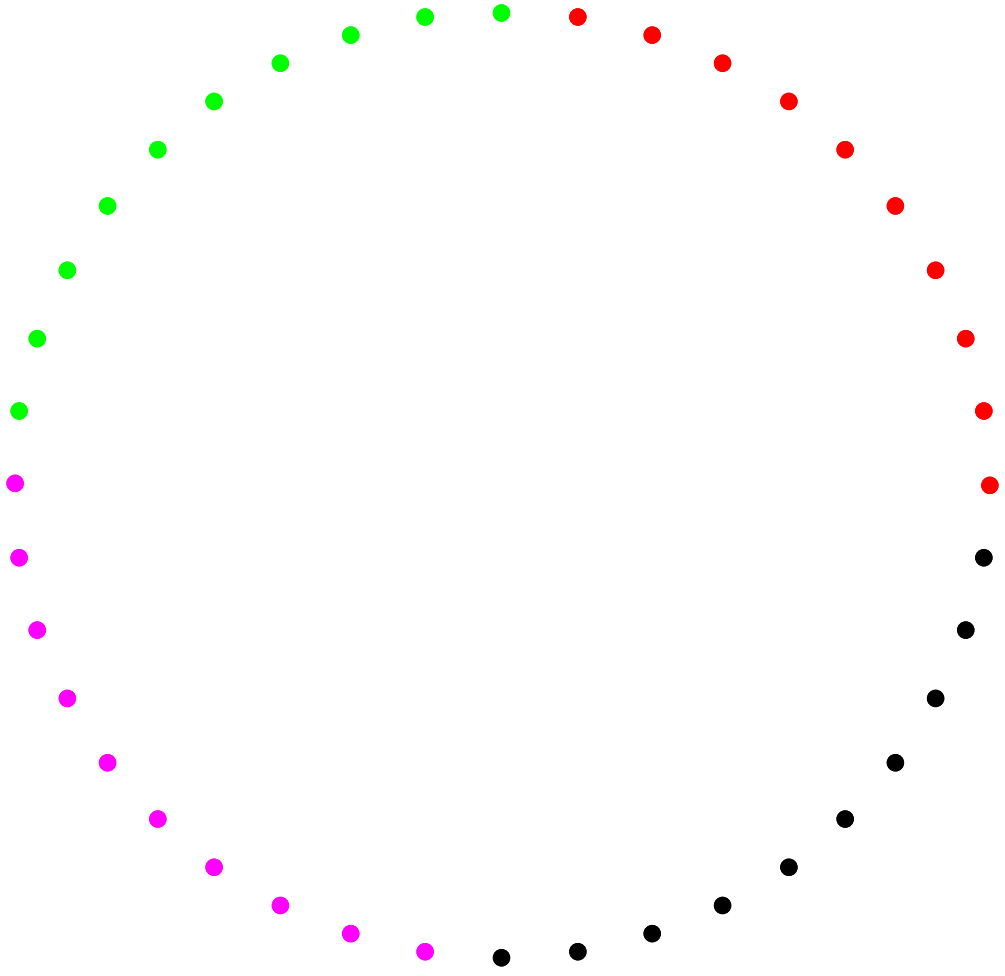
Statistical and computational challenges

- From a statistical perspective
  - A large number of (small) communities
  - The observed network is sparse
- From a computational perspective
  - Large solution space

Question

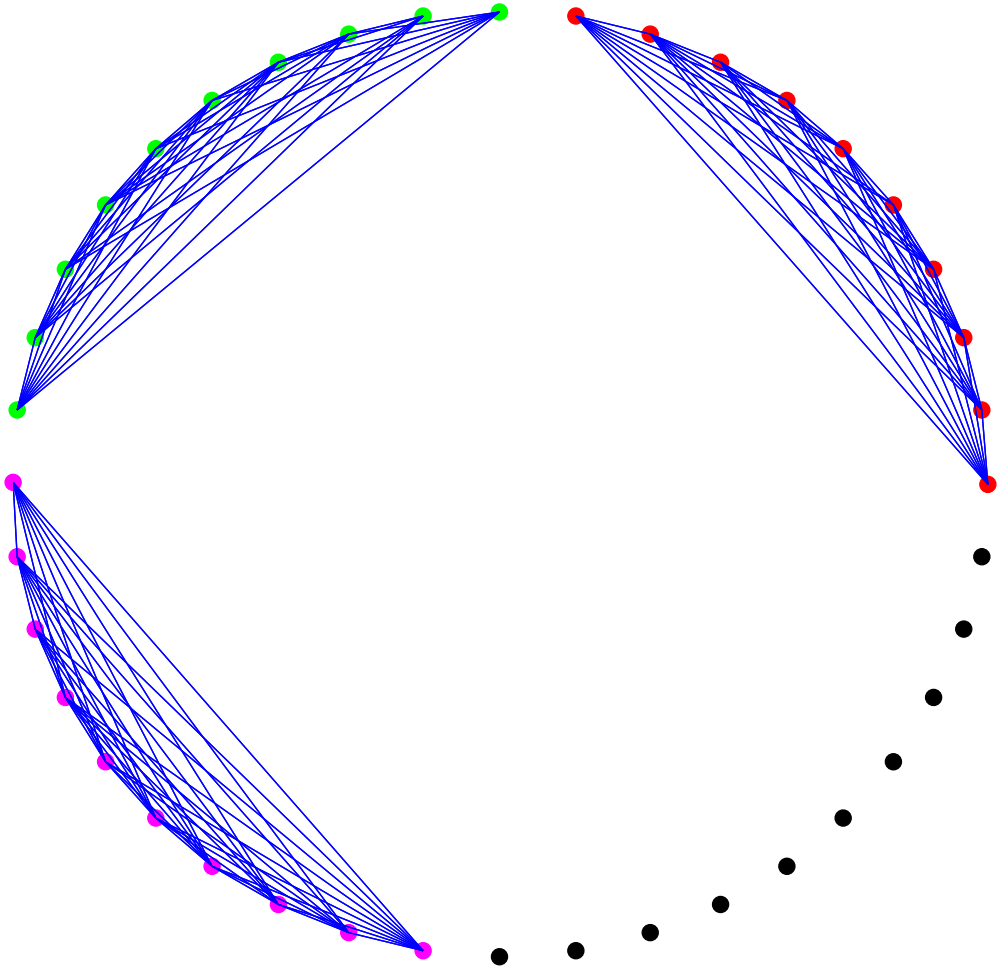
- Is there a computationally efficient and statistically optimal community detection algorithm?

Stochastic block model [Holland et al. ’83]



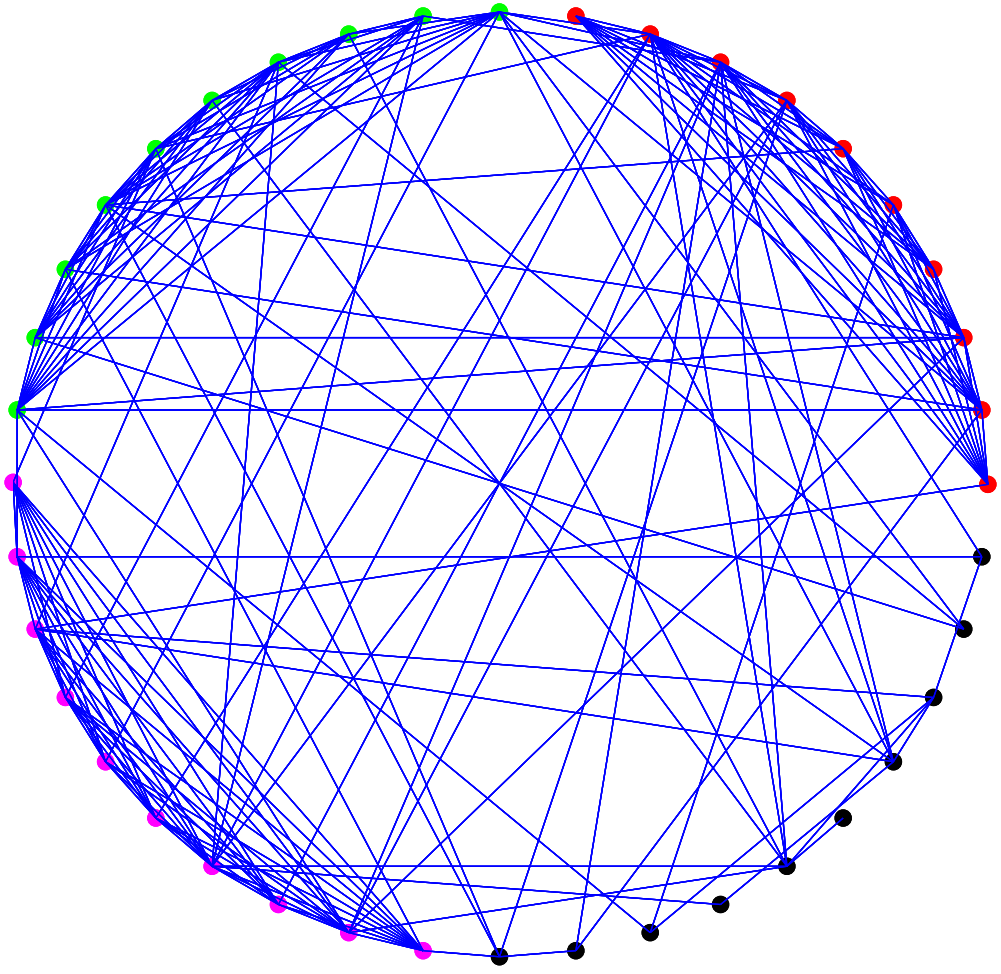
$n = 40$, $K = 10$, $r = 3$

Stochastic block model [Holland et al. ’83]



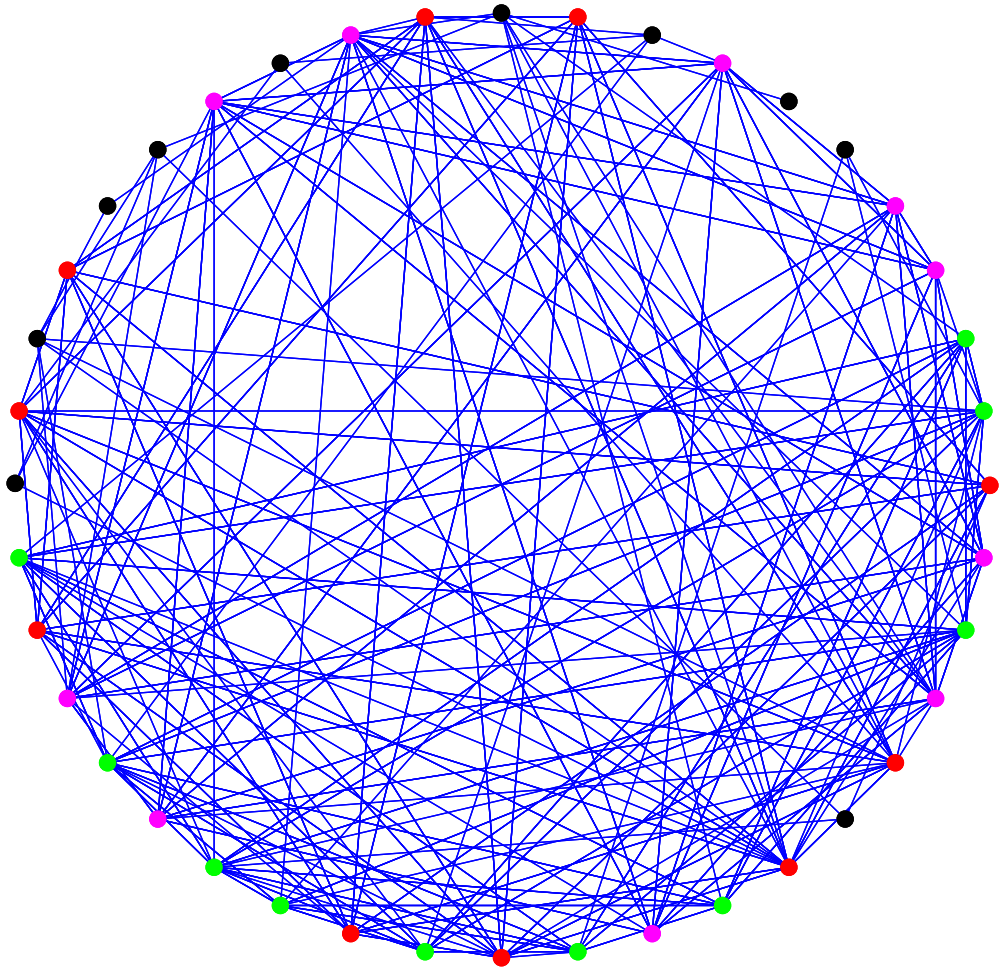
$p = 0.9$

Stochastic block model [Holland et al. ’83]



$p = 0.9 \quad q = 0.1$

Stochastic block model [Holland et al. ’83]



$p = 0.9 \quad q = 0.1$

Exact recovery

- True cluster structure: C^*
- Estimated cluster structure: $\widehat{C}$
- Exact recovery if as $n \to \infty$,

$$\mathbb{P}\left\{\widehat{C} = C^*\right\} \to 1.$$

Notice: Model parameters K, r, p, q can scale with n

Exact recovery

- True cluster structure: C^*
- Estimated cluster structure: $\widehat{C}$
- Exact recovery if as $n \to \infty$,

$$\mathbb{P}\left\{\widehat{C} = C^*\right\} \to 1.$$

Notice: Model parameters K, r, p, q can scale with n

- Correlated recovery: [Decelle et al. '11][Mossel-Neeman-Sly '12]
- Partial recovery: [Abbe-Sandon '15] [Montanari '15]...

Objectives of this talk

Objectives:

- Information limit: In which regime of n, K, r, p, q, is exact recovery possible (impossible)?
- Computational limit: In which regime of n, K, r, p, q, is exact recovery computationally easy (hard)?

Disclaimers:

- Will emphasize intuition
- Many things will be left out: e.g., phase transitions in correlated recovery: [Decelle et al. '11][Mossel-Neeman-Sly '12, '13] [Massoulié '13]

Outline of the remainder

1. Impossible-hard-easy transitions in exact recovery
2. Sharp recovery via semidefinite programming
3. Computational lower bounds for detecting a community

Information limit for cluster recovery

Theorem [Chen-Xu '14]

Exact cluster recovery is possible if and only if

$$K \cdot D(q\|p) \gtrsim \log(rK) \quad \text{and} \quad K \cdot D(p\|q) \gtrsim \log n$$

Information limit for cluster recovery

Theorem [Chen-Xu '14]

Exact cluster recovery is possible if and only if

$$K \cdot D(q\|p) \gtrsim \log(rK) \quad \text{and} \quad K \cdot D(p\|q) \gtrsim \log n$$

- $q \asymp p$: simplifies to $K(p-q)^2 \gtrsim q(1-q)\log n$ (independent of r)

Information limit for cluster recovery

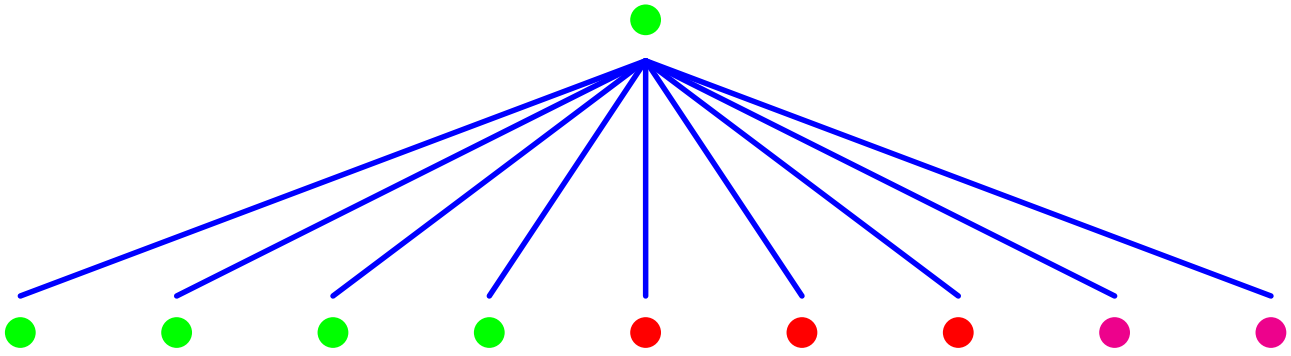
Theorem [Chen-Xu '14]

Exact cluster recovery is possible if and only if

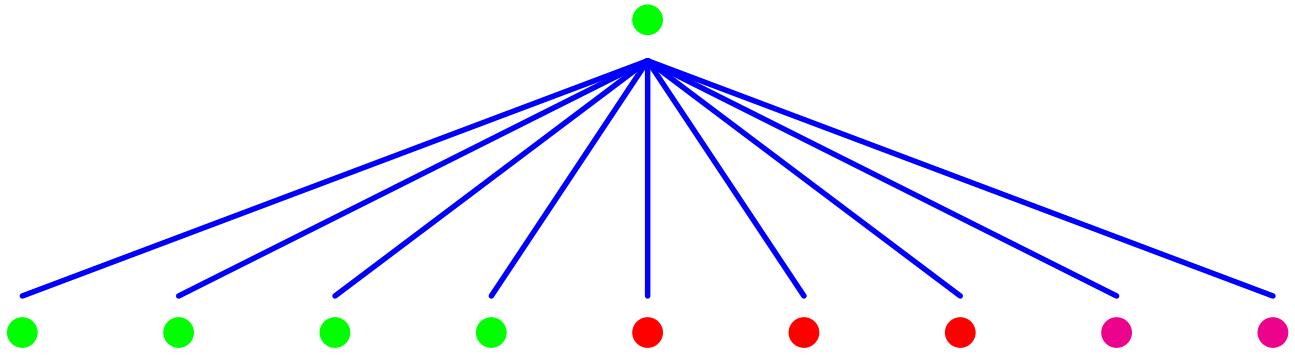
$$K \cdot D(q\|p) \gtrsim \log(rK) \quad \text{and} \quad K \cdot D(p\|q) \gtrsim \log n$$

- $q \asymp p$: simplifies to $K(p-q)^2 \gtrsim q(1-q)\log n$ (independent of r)
- Converse: Fano's inequality
- Achievability: Maximum likelihood estimator (computationally intractable)

Interpretation of information limit: Local testing condition

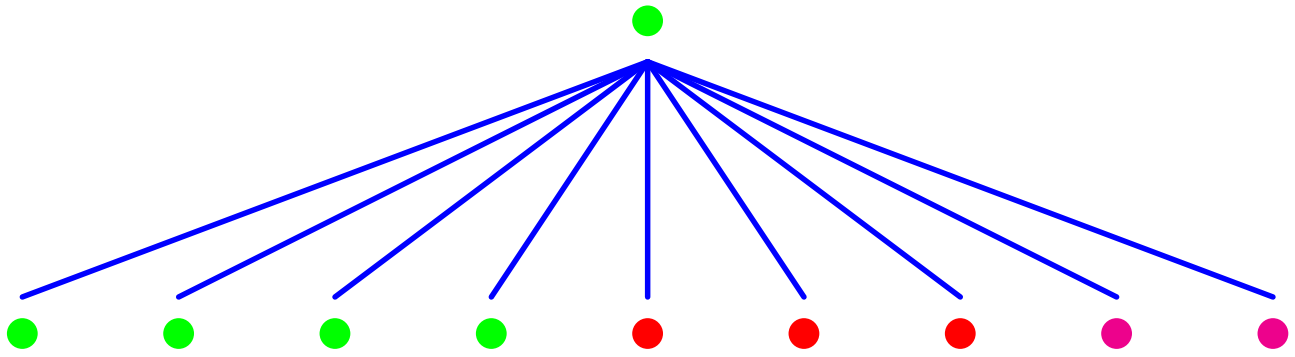


Interpretation of information limit: Local testing condition



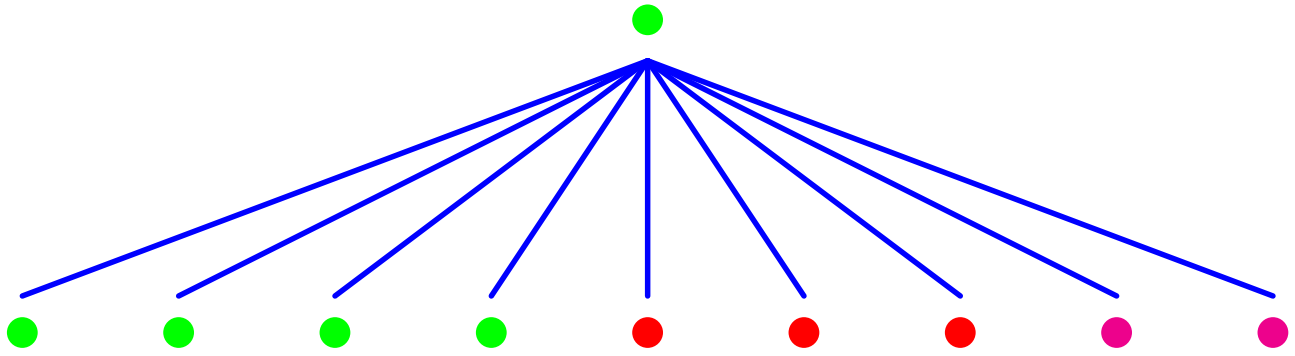
- $\mathbb{P}\{S < T_1\} \lesssim ?$

Interpretation of information limit: Local testing condition



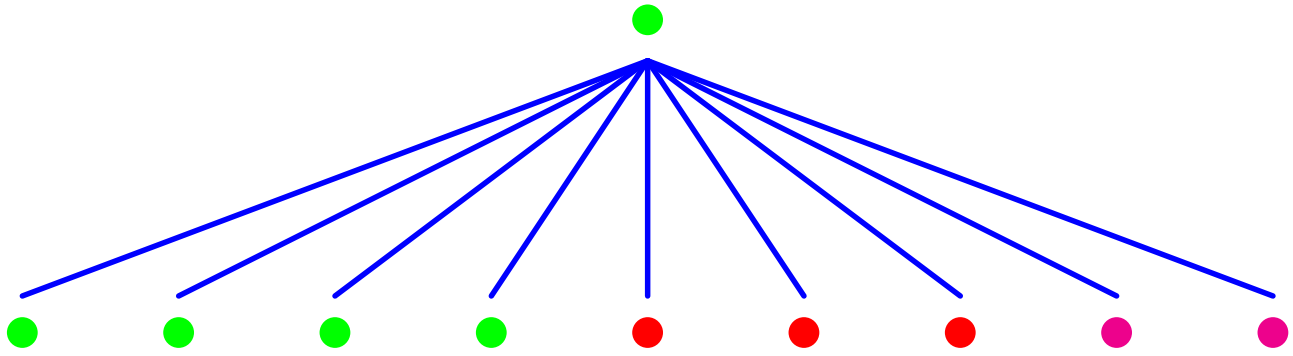
- $\mathbb{P}\{S < T_1\} \lesssim e^{-K \min\{D(q\|p), D(p\|q)\}}$

Interpretation of information limit: Local testing condition



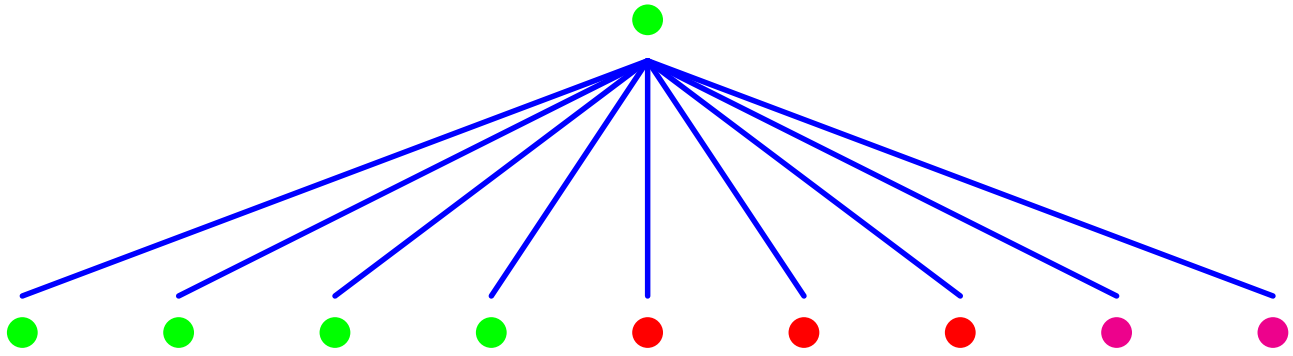
- $\mathbb{P}\{S < T_1\} \lesssim e^{-K \min\{D(q\|p), D(p\|q)\}}$
- $\mathbb{P}\{S < \max\{T_1, \ldots, T_{r-1}\}\} \lesssim r \cdot e^{-K \min\{D(q\|p), D(p\|q)\}}$

Interpretation of information limit: Local testing condition



- $\mathbb{P}\{S < T_1\} \lesssim e^{-K \min\{D(q\|p), D(p\|q)\}}$
- $\mathbb{P}\{S < \max\{T_1, \ldots, T_{r-1}\}\} \lesssim r \cdot e^{-K \min\{D(q\|p), D(p\|q)\}}$
- $\mathbb{P}\{S < \max\{T_1, \ldots, T_{r-1}\} \text{ for all nodes}\} \lesssim nr \cdot e^{-K \min\{D(q\|p), D(p\|q)\}}$

Interpretation of information limit: Local testing condition



- $\mathbb{P}\{S < T_1\} \lesssim e^{-K \min\{D(q\|p), D(p\|q)\}}$
- $\mathbb{P}\{S < \max\{T_1, \ldots, T_{r-1}\}\} \lesssim r \cdot e^{-K \min\{D(q\|p), D(p\|q)\}}$
- $\mathbb{P}\{S < \max\{T_1, \ldots, T_{r-1}\} \text{ for all nodes}\} \lesssim nr \cdot e^{-K \min\{D(q\|p), D(p\|q)\}}$
- If $K \min\{D(q\|p), D(p\|q)\} \gtrsim \log n$, then for every node, its color is the same as the most representative color among its neighbors

Polynomial-time recovery: semidefinite programming

- Maximum likelihood estimator:

$$\max_{Y} \quad \sum_{i,j} A_{ij} Y_{ij}$$

s.t. Y is a binary matrix representing r clusters of size K

Polynomial-time recovery: semidefinite programming

- Maximum likelihood estimator:

$$\max_{Y} \quad \sum_{i,j} A_{ij} Y_{ij}$$

$$\text{s.t.} \quad Y \text{ is a binary matrix representing } r \text{ clusters of size } K$$

- A semidefinite programming (SDP) relaxation of MLE:

$$\max_{Y} \quad \sum_{i,j} A_{ij} Y_{ij}$$

$$\text{s.t.} \quad Y \succeq 0,\ \mathrm{Tr}(Y) \leq rK,\ Y_{ij} \in [0,1]$$

$$\sum_{j} Y_{ij} \leq K,\ \sum_{i,j} Y_{ij} \leq rK^2$$

Performance guarantee of semidefinite programming

Theorem [Chen-Xu '14]

Exact cluster recovery is achieved via semidefinite programming if

$$K^2(p-q)^2 \gtrsim p(1-q)K\log n + q(1-q)n.$$

Performance guarantee of semidefinite programming

Theorem [Chen-Xu '14]

Exact cluster recovery is achieved via semidefinite programming if

$$K^2(p-q)^2 \gtrsim p(1-q)K\log n + q(1-q)n.$$

- Information limit ($q \asymp p$): $K(p-q)^2 \gtrsim q(1-q)\log n$

Performance guarantee of semidefinite programming

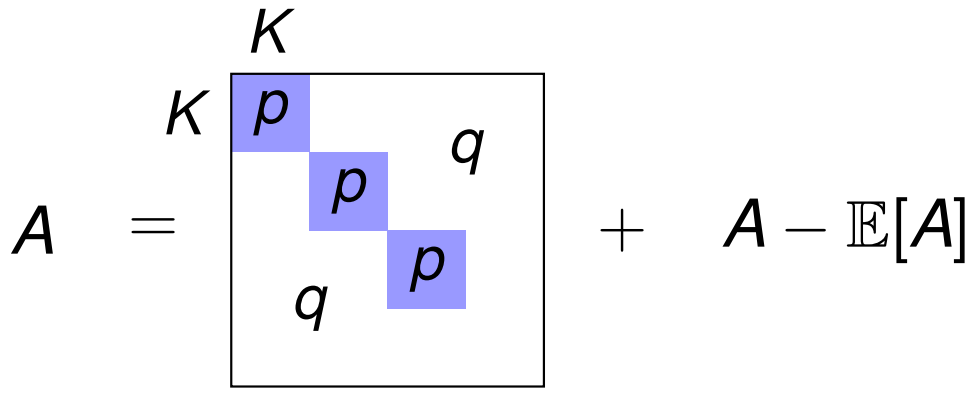
Theorem [Chen-Xu '14]

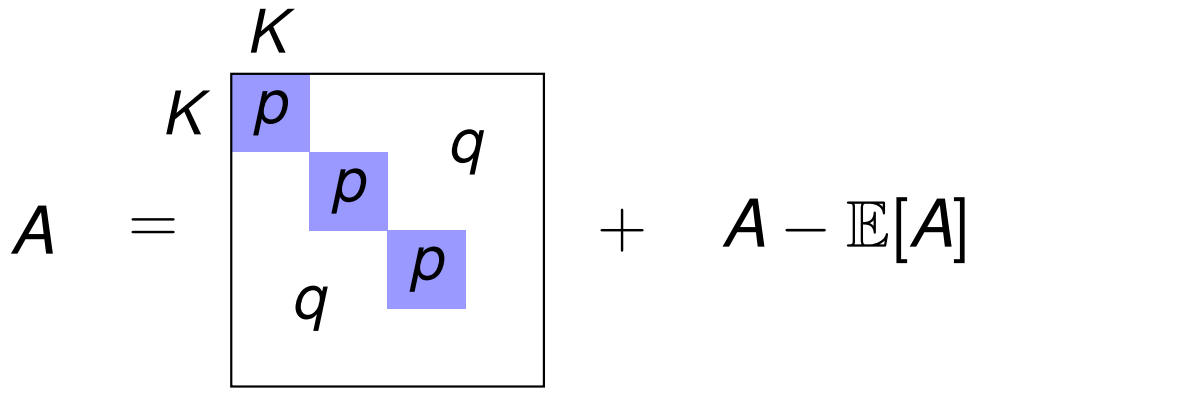
Exact cluster recovery is achieved via semidefinite programming if

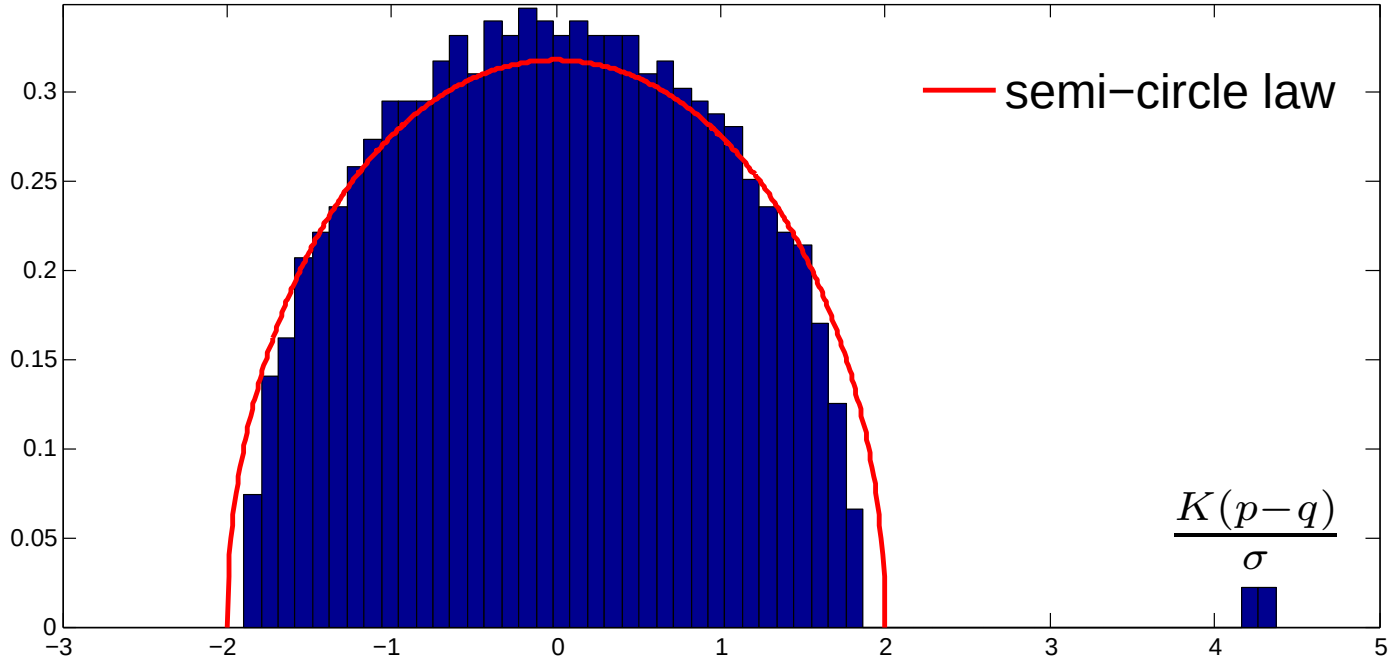
$$K^2(p-q)^2 \gtrsim p(1-q)K\log n + q(1-q)n.$$

- Information limit ($q \asymp p$): $K(p-q)^2 \gtrsim q(1-q)\log n$
- Spectral condition

$$K(p-q) \gtrsim \sqrt{q(1-q)n}$$

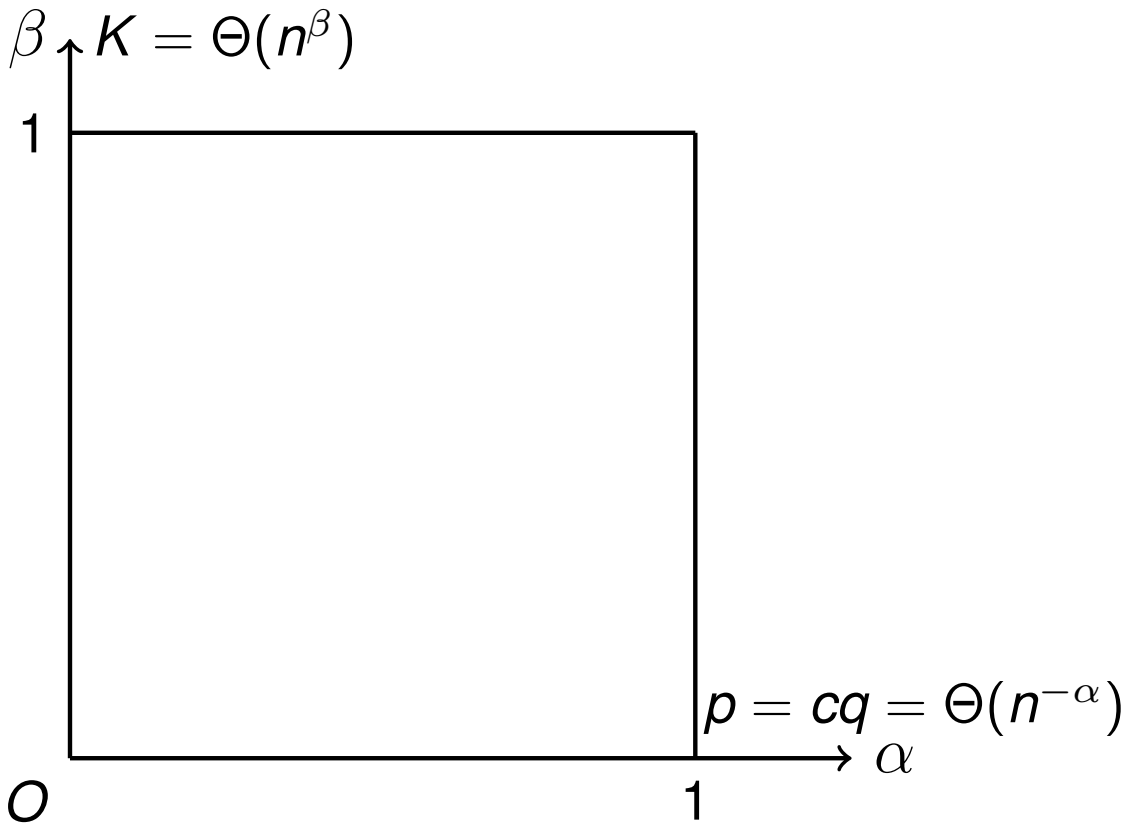

K
K
p
q
p
p
q
A = + A − 𝔼[A]



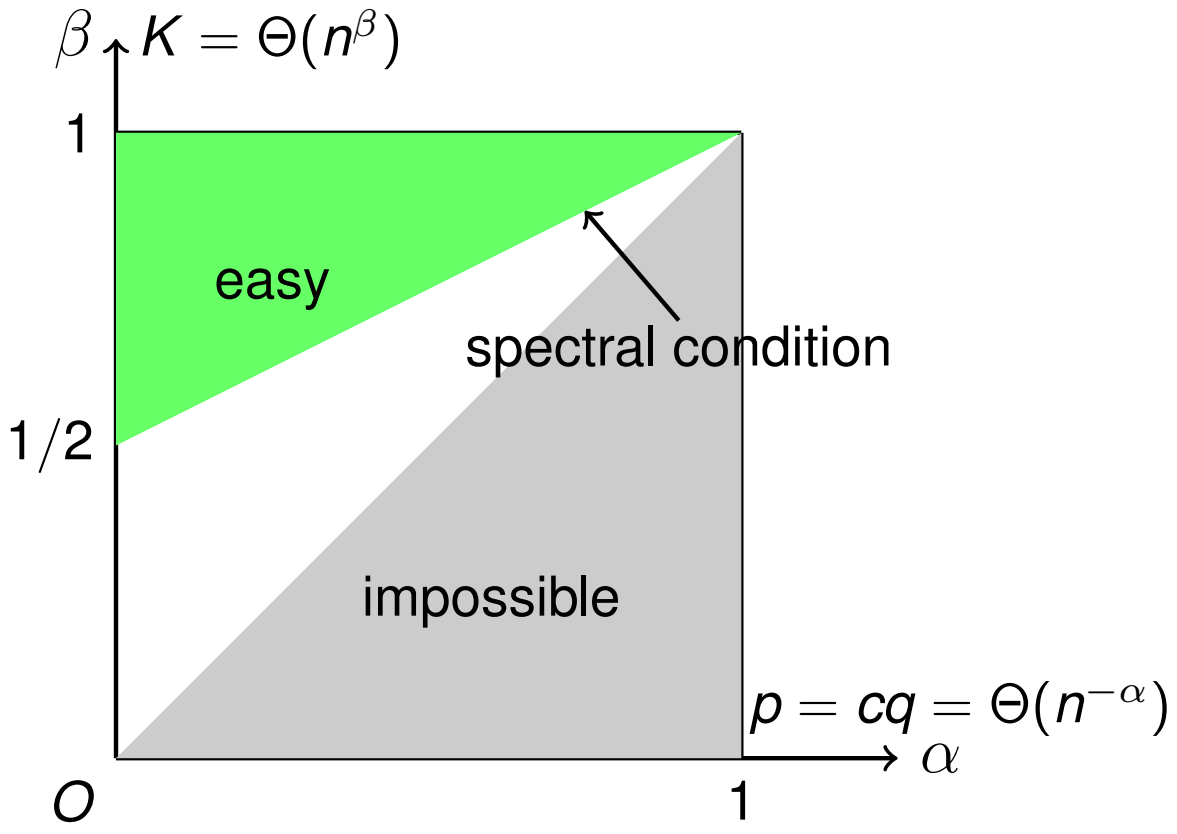


Eigenvalue distribution of $\frac{A-q\mathbf{1}\mathbf{1}^\top}{\sigma}$ for $\sigma = \sqrt{q(1-q)n}$

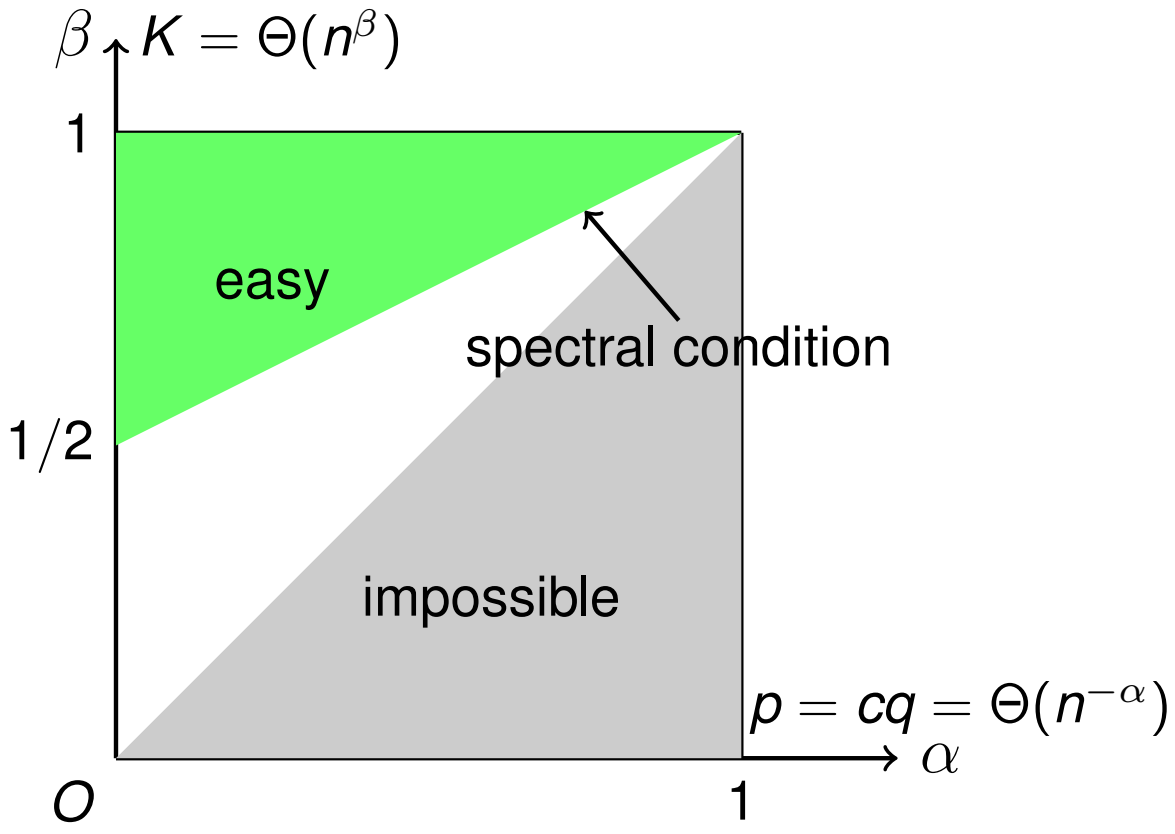
Impossible-easy-hard phase transitions



Impossible-easy-hard phase transitions



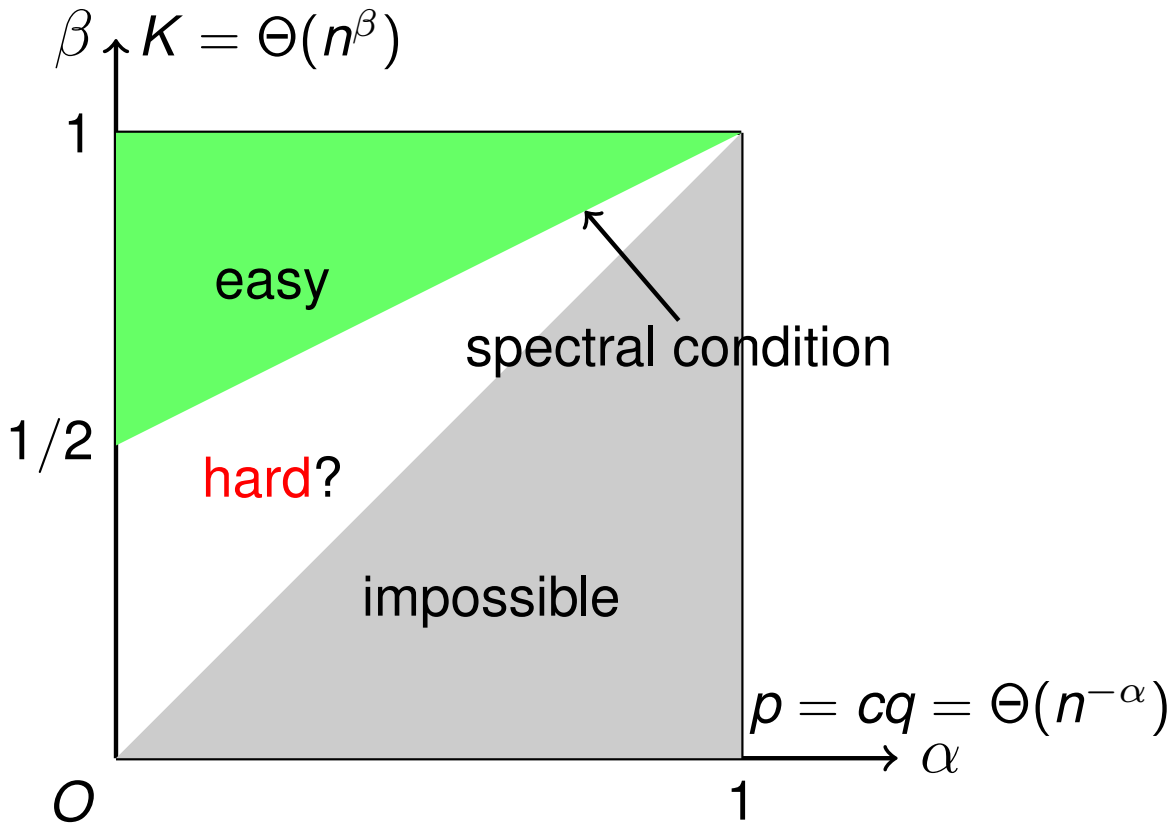
Impossible-easy-hard phase transitions



Two main questions:

1. $\beta = 1$: Can SDP achieve the sharp recovery threshold?

Impossible-easy-hard phase transitions



Two main questions:

1. $\beta = 1$: Can SDP achieve the sharp recovery threshold?
2. $\beta < 1$: Is the gap fundamental?

Outline of the remainder

1. Impossible-hard-easy transitions in exact recovery
2. Optimal recovery via semidefinite programming
3. Computational lower bounds for detecting a community

Optimal recovery via SDP

Theorem (Abbe et al. '14, Mossel et al. '14)

Suppose $r = 2$, $K = n/2$, $p = a \log n/n$, and $q = b \log n/n$.

- *If $(\sqrt{a} - \sqrt{b})^2 > 2$, recovery is achievable in polynomial-time.*
- *If $(\sqrt{a} - \sqrt{b})^2 < 2$, recovery is impossible.*

Conjecture (Abbe et al. '14)

SDP achieves the optimal recovery threshold $(\sqrt{a} - \sqrt{b})^2 > 2$.

Optimal recovery via SDP

Theorem (Abbe et al. '14, Mossel et al. '14)

Suppose $r = 2$, $K = n/2$, $p = a \log n/n$, and $q = b \log n/n$.

- *If $(\sqrt{a} - \sqrt{b})^2 > 2$, recovery is achievable in polynomial-time.*
- *If $(\sqrt{a} - \sqrt{b})^2 < 2$, recovery is impossible.*

Conjecture (Abbe et al. '14)

SDP achieves the optimal recovery threshold $(\sqrt{a} - \sqrt{b})^2 > 2$.

- Conjectured established in [Hajek-Wu-Xu '14]; also proved by [Bandeira '15] independently

Why is there no computational gap?

- Spectral condition $K(p-q) \gtrsim \sqrt{q(1-q)n}$ is satisfied

Why is there no computational gap?

- Spectral condition $K(p-q) \gtrsim \sqrt{q(1-q)n}$ is satisfied
- Local testing condition [Abbe et al. '14]: Follows from Chernoff bound

$$\mathbb{P}\left\{\mathrm{Binom}\left(\frac{n}{2}, \frac{a\log n}{n}\right) < \mathrm{Binom}\left(\frac{n}{2}, \frac{b\log n}{n}\right)\right\} = n^{-(\sqrt{a}-\sqrt{b})^2/2+o(1)}$$

Why is there no computational gap?

- Spectral condition $K(p-q) \gtrsim \sqrt{q(1-q)n}$ is satisfied
- Local testing condition [Abbe et al. '14]: Follows from Chernoff bound

$$\mathbb{P}\left\{\mathrm{Binom}\left(\frac{n}{2}, \frac{a\log n}{n}\right) < \mathrm{Binom}\left(\frac{n}{2}, \frac{b\log n}{n}\right)\right\} = n^{-(\sqrt{a}-\sqrt{b})^2/2+o(1)}$$

- Suggest: The following two-step procedure is optimal:
  1. Correctly cluster all but $o(n)$ vertices via spectral method
  2. Clean-up via local neighborhood test

Why is the SDP optimal?

Seeking ± 1 cluster matrix $Y^* = \sigma^*(\sigma^*)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \\ & Y_{ii} = 1 \\ & \langle \mathbf{J}, Y \rangle = 0 \end{aligned}$$

Why is the SDP optimal?

Seeking ± 1 cluster matrix $Y^* = \sigma^*(\sigma^*)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & \langle \mathbf{J}, Y \rangle = 0 \quad \lambda \end{aligned}$$

- $d_i = e(i, C_1^*) - e(i, C_2^*)$ for $i \in C_1$; similarly for $i \in C_2$

Why is the SDP optimal?

Seeking ± 1 cluster matrix $Y^* = \sigma^*(\sigma^*)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & \langle \mathbf{J}, Y \rangle = 0 \quad \lambda \end{aligned}$$

- $d_i = e(i, C_1^*) - e(i, C_2^*)$ for $i \in C_1$; similarly for $i \in C_2$
- $S = D - A + \lambda \mathbf{J} \succeq 0$ if $\min d_i \geq \|A - \mathbb{E}[A]\|$ and $\lambda \geq (p+q)/2$

Why is the SDP optimal?

Seeking ± 1 cluster matrix $Y^* = \sigma^*(\sigma^*)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & \langle \mathbf{J}, Y \rangle = 0 \quad \lambda \end{aligned}$$

- $d_i = e(i, C_1^*) - e(i, C_2^*)$ for $i \in C_1$; similarly for $i \in C_2$
- $S = D - A + \lambda \mathbf{J} \succeq 0$ if $\min d_i \geq \|A - \mathbb{E}[A]\|$ and $\lambda \geq (p+q)/2$
- $\|A - \mathbb{E}[A]\| \sim \sqrt{\log n}$ and $\min d_i \geq \frac{\log n}{\log \log n}$ if $\sqrt{a} - \sqrt{b} > \sqrt{2}$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \\ & Y_{ii} = 1 \\ & Y_{ij} \geq 0 \\ & \sum_{j} (Y_{ij} + Y_{ji}) = 2K \end{aligned}$$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$
\begin{aligned}
\max_{Y} \ & \langle A, Y \rangle \\
\text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\
& Y_{ii} = 1 \quad d_i \\
& Y_{ij} \geq 0 \quad B \geq 0 \\
& \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i
\end{aligned}
$$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & Y_{ij} \geq 0 \quad B \geq 0 \\ & \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i \end{aligned}$$

- $S\,\xi_k = 0 \Rightarrow d_i = e(i, C_k^*) - \lambda_i K - \sum_{j \in C_k} \lambda_j$ for $i \in C_k$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & Y_{ij} \geq 0 \quad B \geq 0 \\ & \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i \end{aligned}$$

- $S\,\xi_k = 0 \Rightarrow d_i = e(i, C_k^*) - \lambda_i K - \sum_{j \in C_k} \lambda_j$ for $i \in C_k$
- $\lambda_i = \max_{k' \neq k} e(i, C_{k'})/K - q/2 + \sqrt{\log n}/(2K)$ for $i \in C_k$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & Y_{ij} \geq 0 \quad B \geq 0 \\ & \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i \end{aligned}$$

- $S\,\xi_k = 0 \Rightarrow d_i = e(i, C_k^*) - \lambda_i K - \sum_{j \in C_k} \lambda_j$ for $i \in C_k$
- $\lambda_i = \max_{k' \neq k} e(i, C_{k'})/K - q/2 + \sqrt{\log n}/(2K)$ for $i \in C_k$
- $B_{C_k \times C_{k'}} = y_{kk'} \mathbf{1}^\top + \mathbf{1} z_{kk'}^\top$ rank 2 !

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & Y_{ij} \geq 0 \quad B \geq 0 \\ & \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i \end{aligned}$$

- $S\,\xi_k = 0 \Rightarrow d_i = e(i, C_k^*) - \lambda_i K - \sum_{j \in C_k} \lambda_j$ for $i \in C_k$
- $\lambda_i = \max_{k' \neq k} e(i, C_{k'})/K - q/2 + \sqrt{\log n}/(2K)$ for $i \in C_k$
- $B_{C_k \times C_{k'}} = y_{kk'} \mathbf{1}^\top + \mathbf{1} z_{kk'}^\top$ rank 2 !
- $S = D - A - B + \lambda \mathbf{1}^\top + \mathbf{1} \lambda^\top \succeq 0$ if $\min d_i \geq \|A - \mathbb{E}[A]\|$

Is the SDP still optimal with r equal-sized clusters?

[Hajek-Wu-Xu '15] Seeking $\{0,1\}$ cluster matrix $Y^* = \sum_{k=1}^{r} \xi_k (\xi_k)^\top$:

$$\begin{aligned} \max_{Y} \ & \langle A, Y \rangle \\ \text{s.t.} \ & Y \succeq 0 \quad S \succeq 0 \\ & Y_{ii} = 1 \quad d_i \\ & Y_{ij} \geq 0 \quad B \geq 0 \\ & \sum_j (Y_{ij} + Y_{ji}) = 2K \quad \lambda_i \end{aligned}$$

- $S\,\xi_k = 0 \Rightarrow d_i = e(i, C_k^*) - \lambda_i K - \sum_{j \in C_k} \lambda_j$ for $i \in C_k$
- $\lambda_i = \max_{k' \neq k} e(i, C_{k'})/K - q/2 + \sqrt{\log n}/(2K)$ for $i \in C_k$
- $B_{C_k \times C_{k'}} = y_{kk'} \mathbf{1}^\top + \mathbf{1} z_{kk'}^\top$ rank 2 !
- $S = D - A - B + \lambda \mathbf{1}^\top + \mathbf{1} \lambda^\top \succeq 0$ if $\min d_i \geq \|A - \mathbb{E}[A]\|$
- $\|A - \mathbb{E}[A]\| \sim \sqrt{\log n}$ and $\min d_i \geq \frac{\log n}{\log \log n}$ if $\sqrt{a} - \sqrt{b} > \sqrt{r}$

Is the SDP optimal more generally?

[Hajek-Wu-Xu '14, '15]

- Two unequal-sized clusters $(\rho n, (1-\rho)n)$
- One cluster of size ρn plus outliers
- Two clusters with no prior knowledge of cluster sizes
- Binary censored block model: $a\,(\sqrt{1-\epsilon}-\sqrt{\epsilon})^2 > 1$, closing gaps in [ABBS '14]

Is the SDP optimal more generally?

[Hajek-Wu-Xu '14, '15]

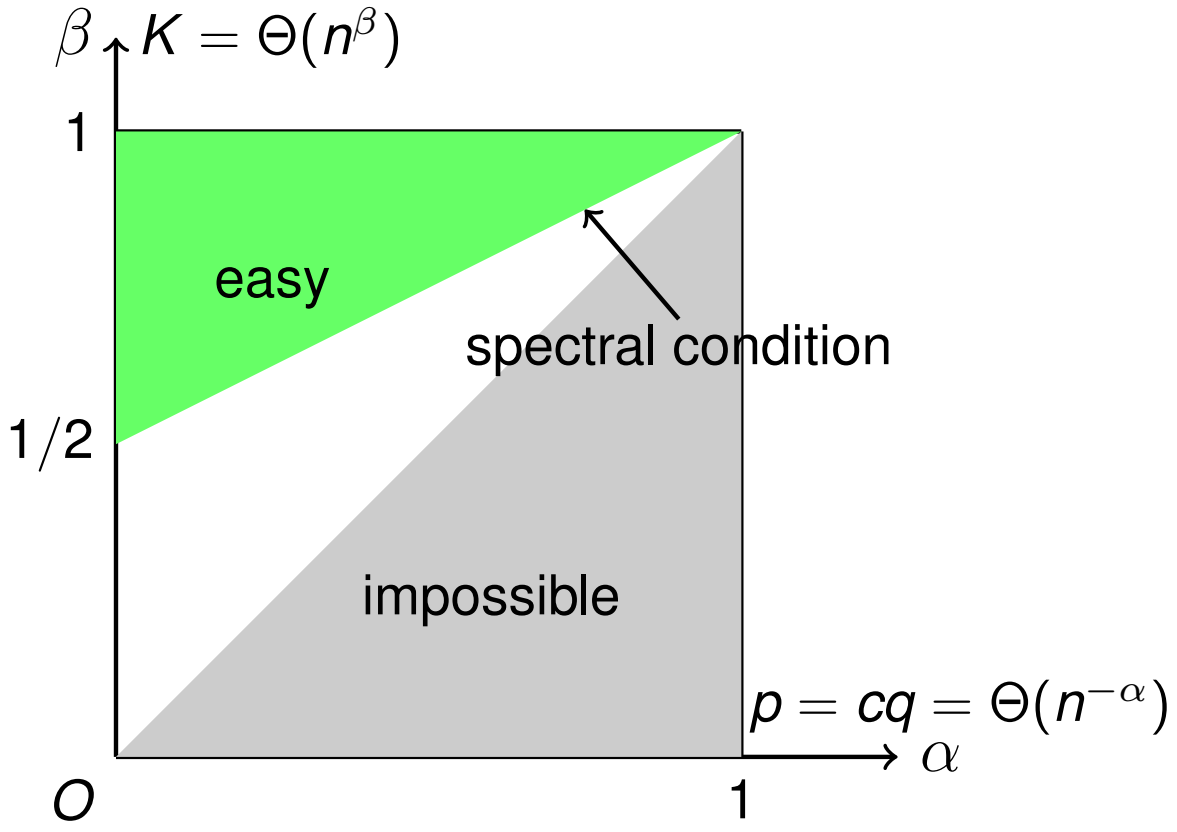
- Two unequal-sized clusters $(\rho n, (1-\rho)n)$
- One cluster of size ρn plus outliers
- Two clusters with no prior knowledge of cluster sizes
- Binary censored block model: $a\,(\sqrt{1-\epsilon}-\sqrt{\epsilon})^2 > 1$, closing gaps in [ABBS '14]

[Yun-Proutiere '14]: The two-step procedure is optimal in r equal-sized cluster case

[Abbe-Sandon '15]: The two-step procedure is optimal in more general cases

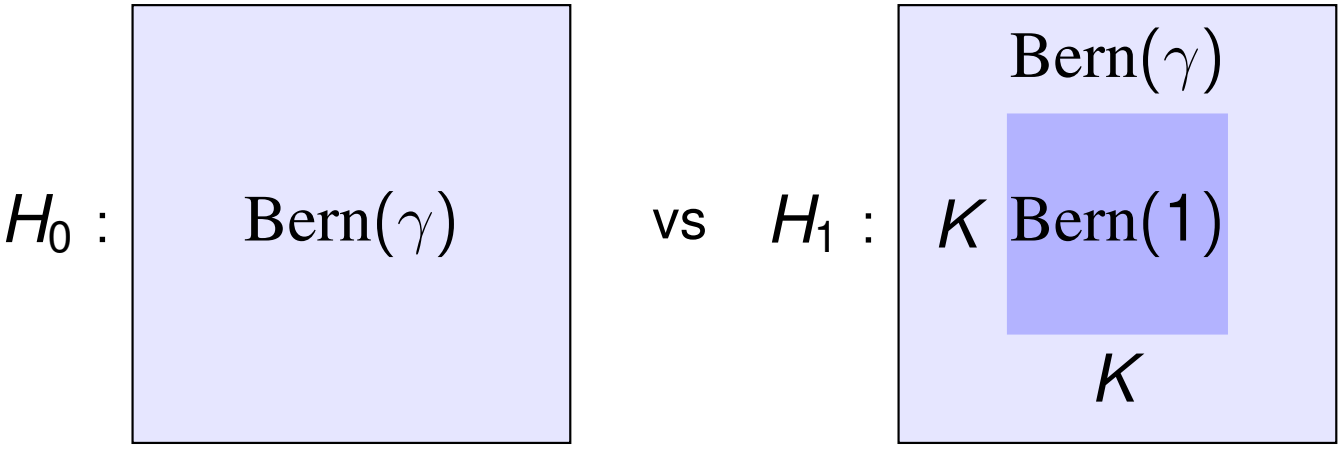
Outline

Conjecture on computational limit



Conjecture [Chen-Xu '14]: no polynomial-time algorithm succeeds significantly beyond the spectral condition.

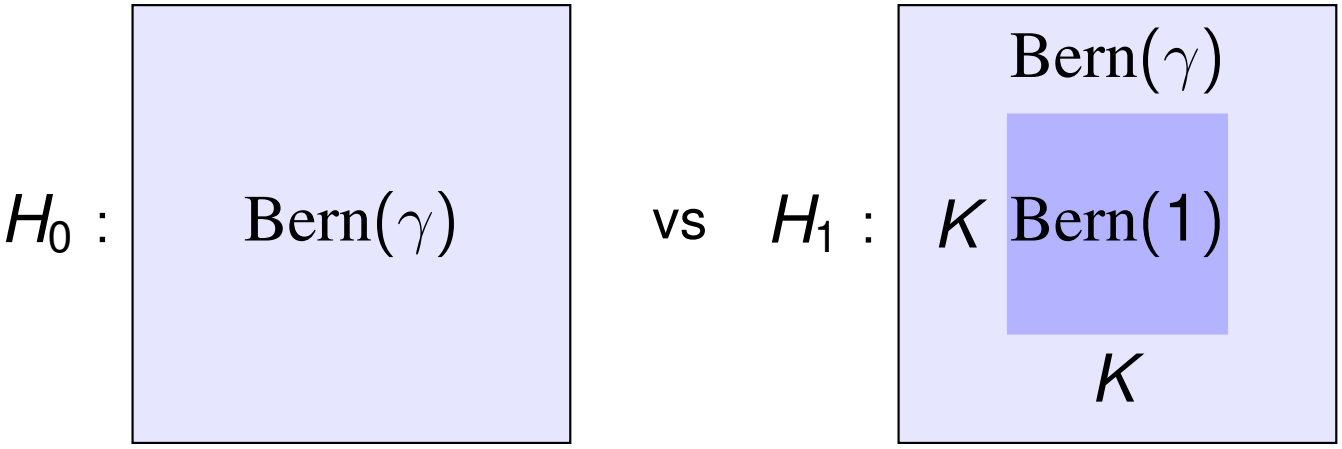
Planted clique hardness hypothesis



Intermediate regime: $\log n \ll K \ll \sqrt{n}$, $\gamma = \Theta(1)$

- detection is possible but believed to have high computational complexity: [Alon et al. '11] [Feldman et al. '13] [Deshpande-Montanari '15] [Meka-Potechin-Wigderson '15]

Planted clique hardness hypothesis

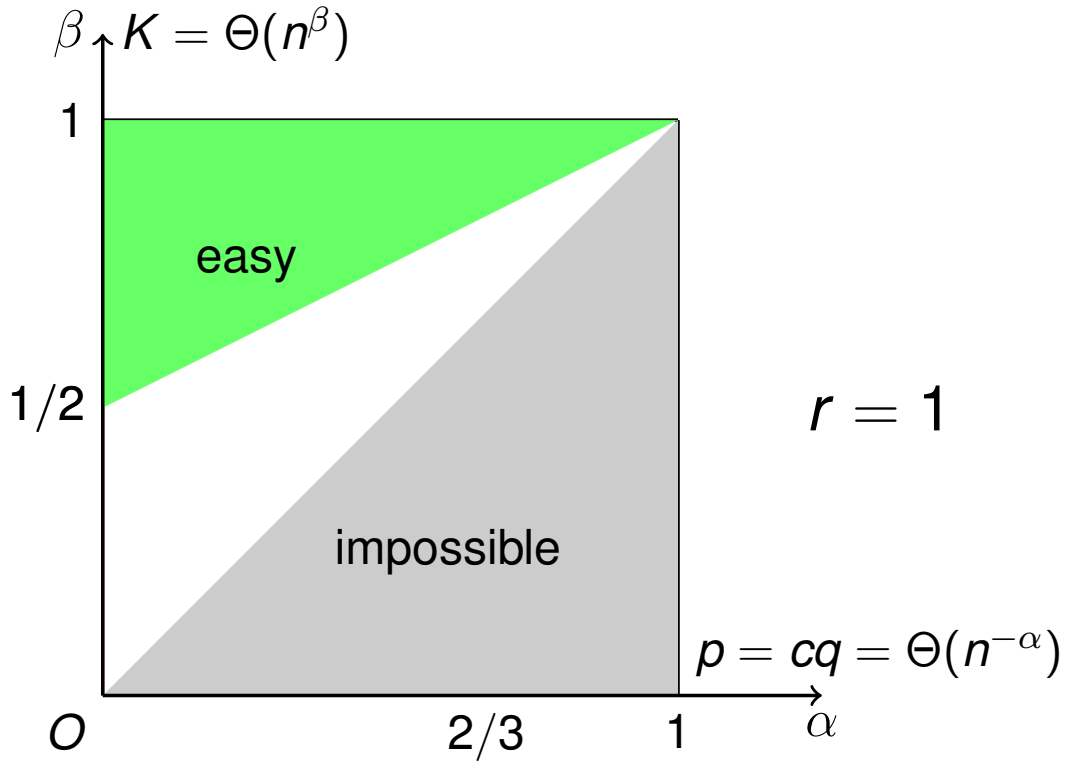


Intermediate regime: $\log n \ll K \ll \sqrt{n}$, $\gamma = \Theta(1)$

- detection is possible but believed to have high computational complexity: [Alon et al. '11] [Feldman et al. '13] [Deshpande-Montanari '15] [Meka-Potechin-Wigderson '15]
- many hardness results assuming Planted Clique hardness with $\gamma = \frac{1}{2}$
  - detecting sparse principal component [Berthet-Rigollet '13]
  - detecting sparse submatrix [Ma-Wu '13] [Cai-Liang-Rakhlin '15]
  - cryptography [Applebaum et al. '10]: $\gamma = 2^{-\log^{0.99} n}$

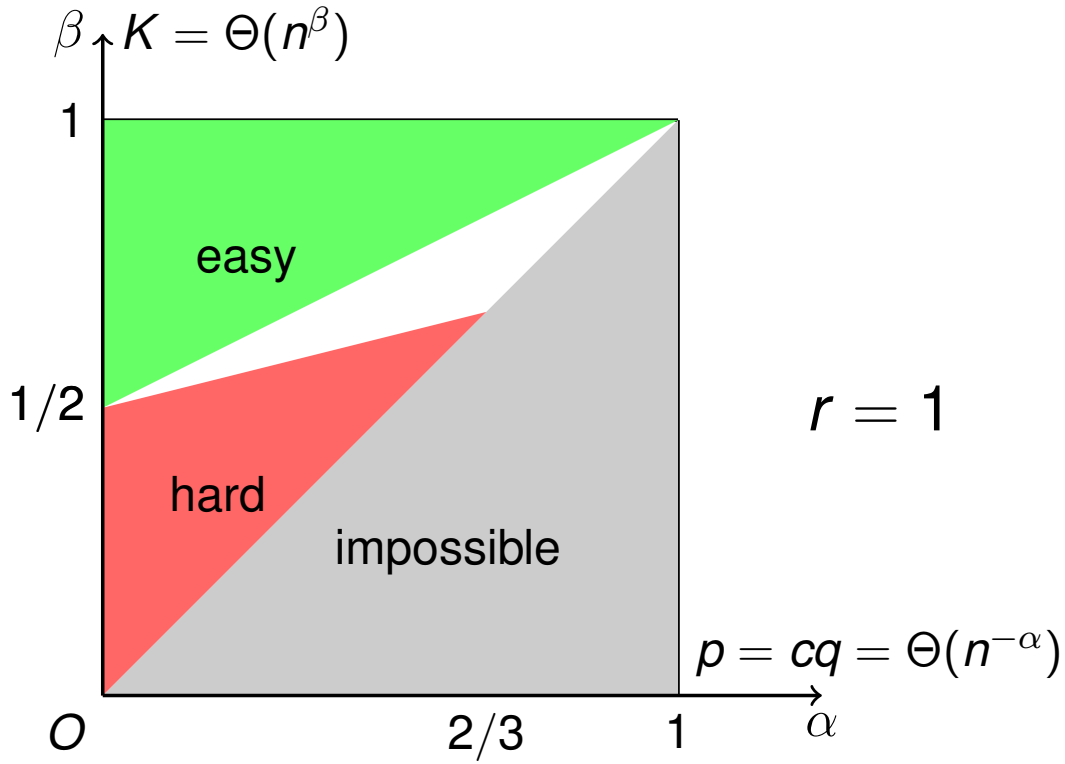
Hard regime for recovering a single cluster

Assuming Planted Clique hardness for any constant $\gamma > 0$



Hard regime for recovering a single cluster

Assuming Planted Clique hardness for any constant $\gamma > 0$



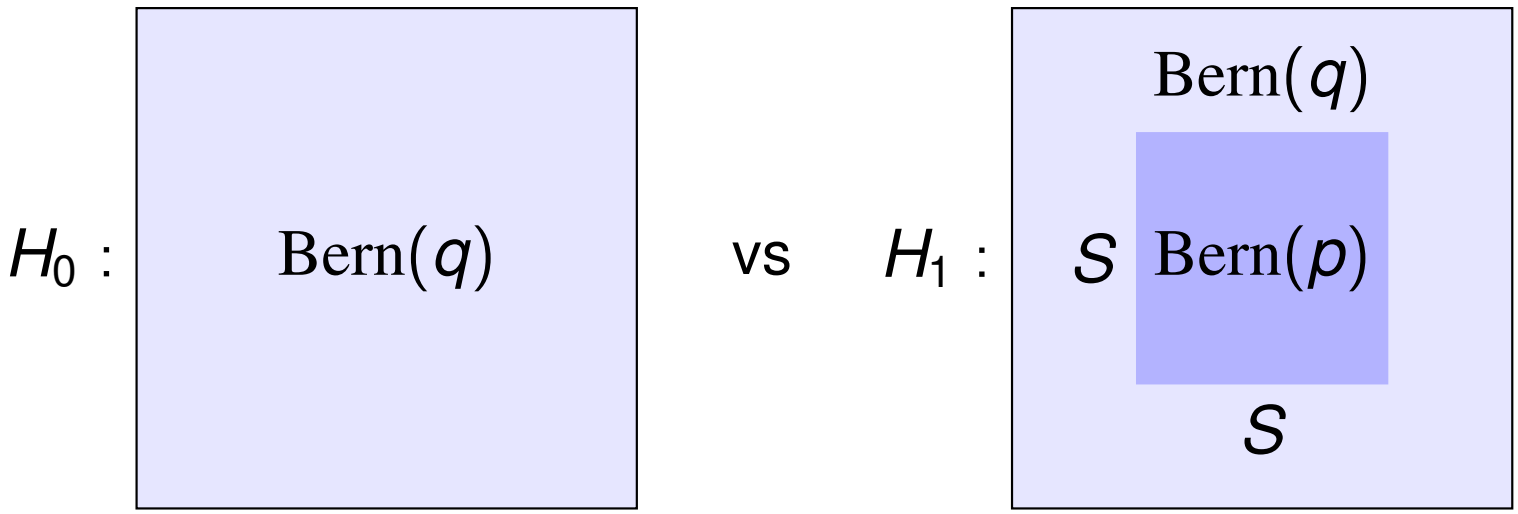
Recovering a single cluster in the red regime is at least as hard as detecting a clique of size $K = o(\sqrt{n})$

Proof step 1: Recovery is harder than *detection*

Recovery versus Detection [Arias-Castro-Verzelen '14] :

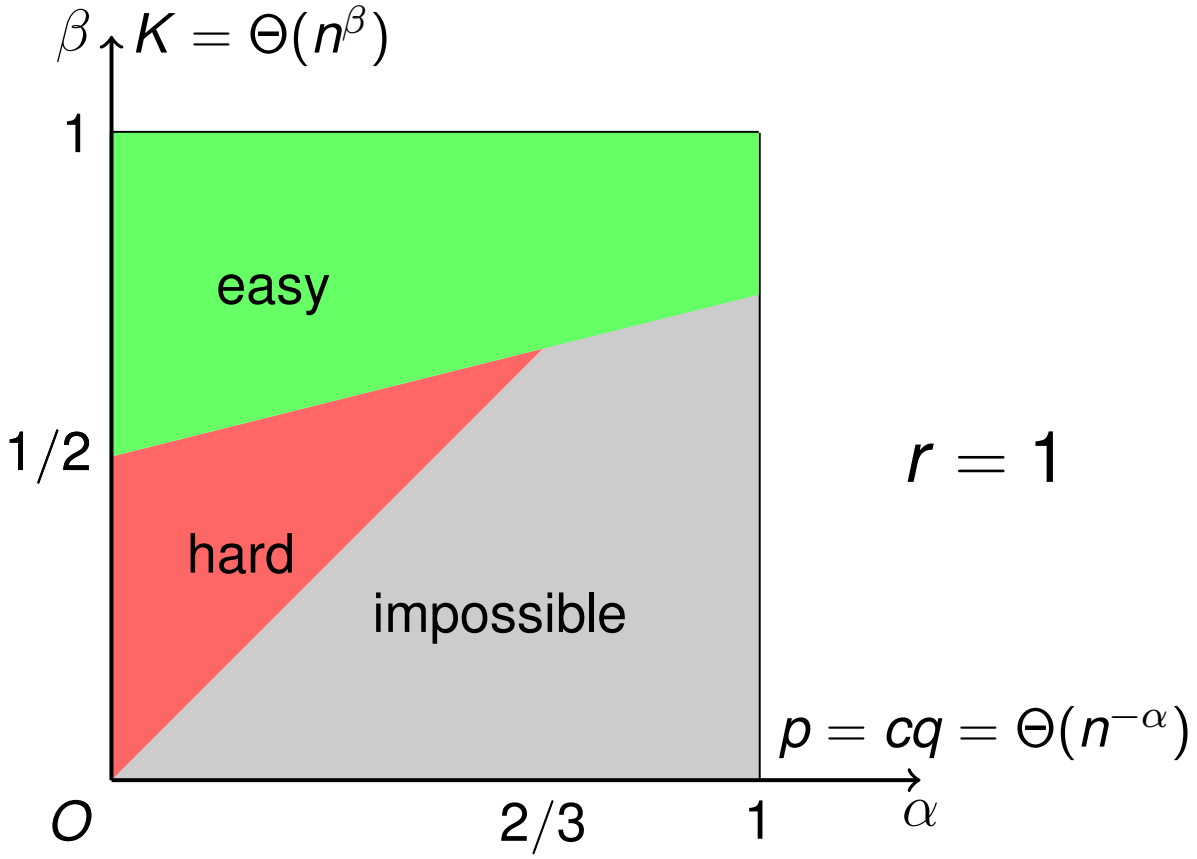
Proof step 1: Recovery is harder than *detection*

Recovery versus Detection [Arias-Castro-Verzelen '14] :



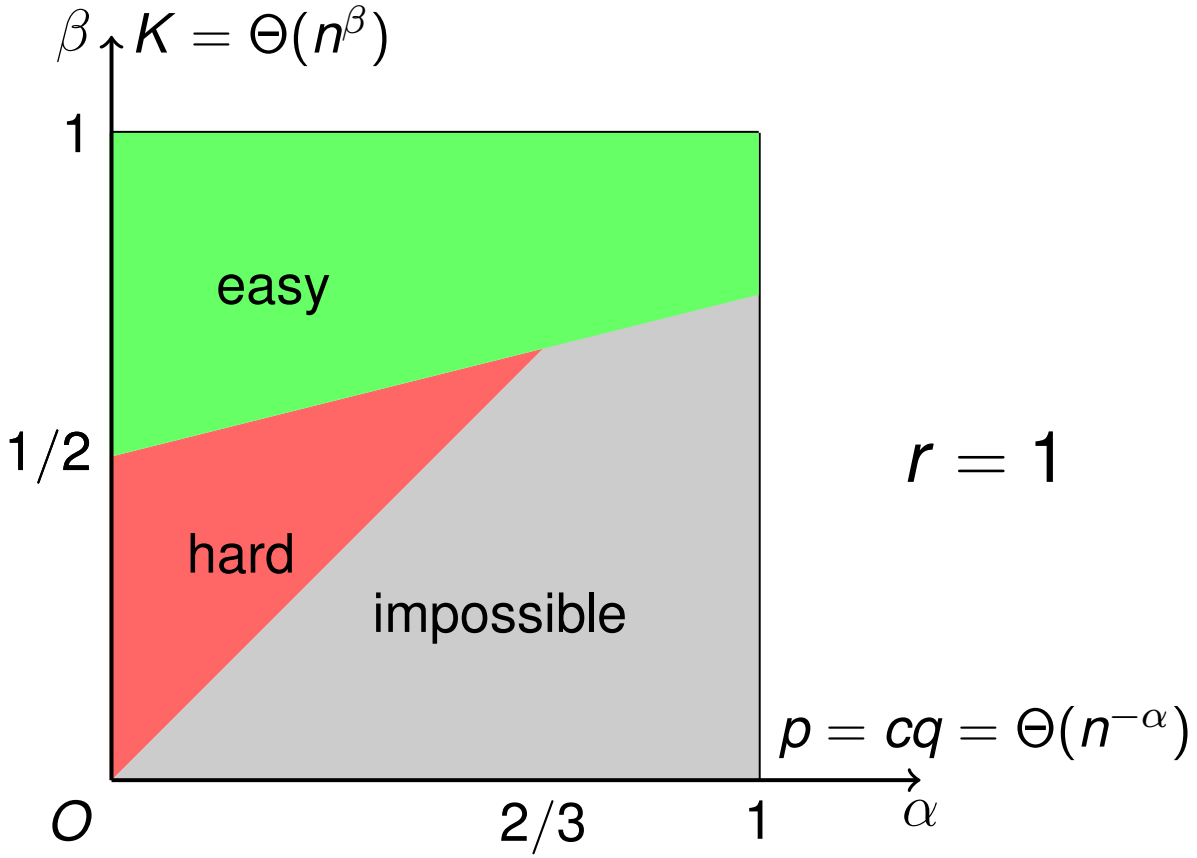
Each node is included in S with probability $\frac{K}{n}$

Proof step 2: Hardness for *detecting* a single cluster

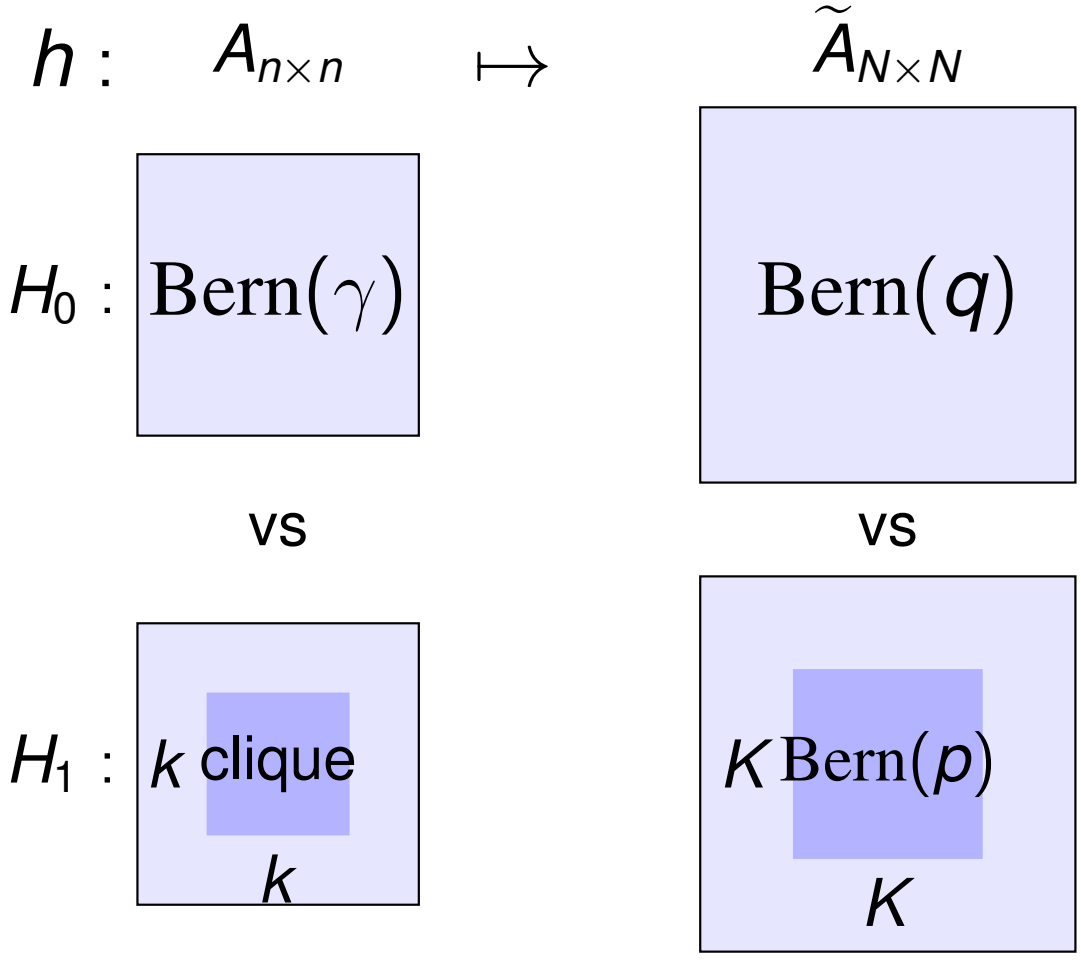


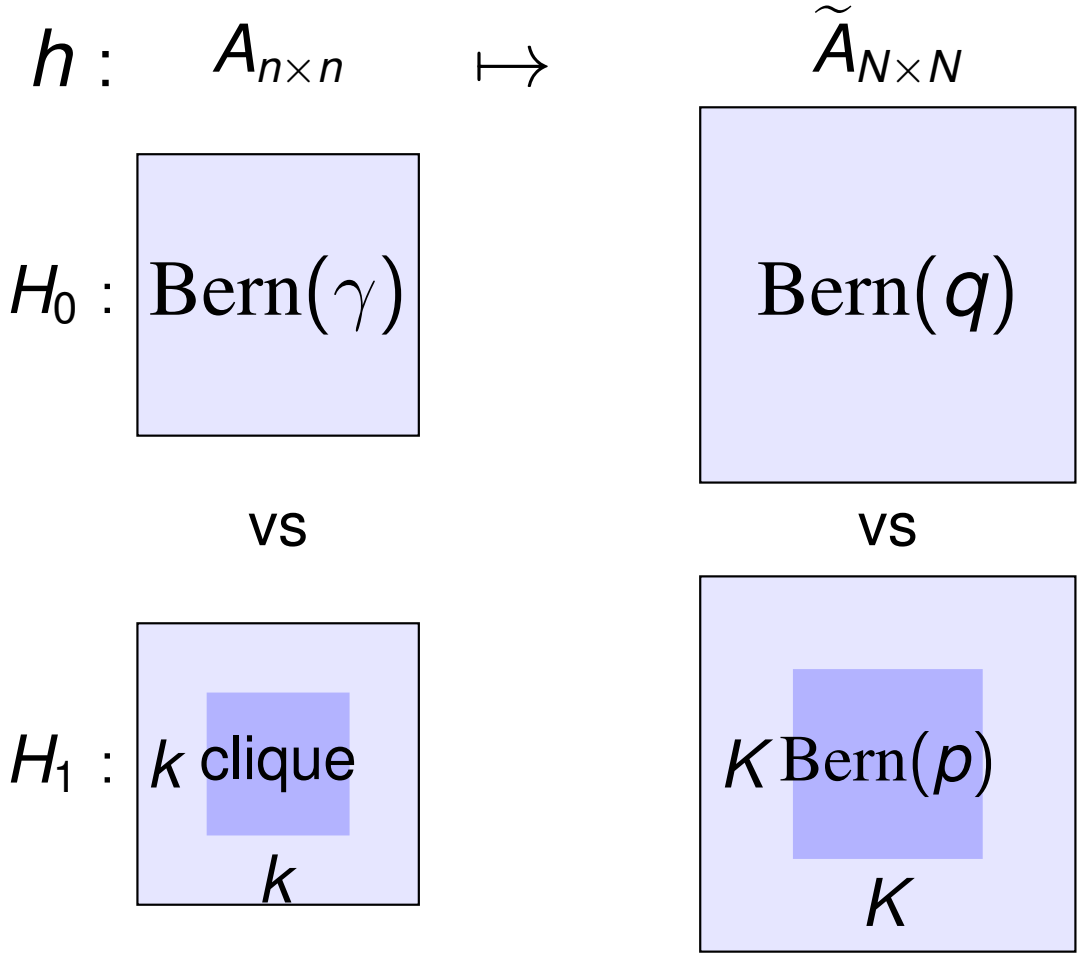
- Detecting a single cluster in the red regime is at least as hard as detecting a clique of size $K = o(\sqrt{n})$

Proof step 2: Hardness for *detecting* a single cluster



- Detecting a single cluster in the red regime is at least as hard as detecting a clique of size $K = o(\sqrt{n})$
- Reduced from Planted Clique detection in polynomial time


h :
$A_{n\times n}$
$\mapsto$
$\widetilde{A}_{N\times N}$
H_0 :
$\mathrm{Bern}(\gamma)$
$\mathrm{Bern}(q)$
vs
vs
H_1 :
k clique
k
K $\mathrm{Bern}(p)$
K

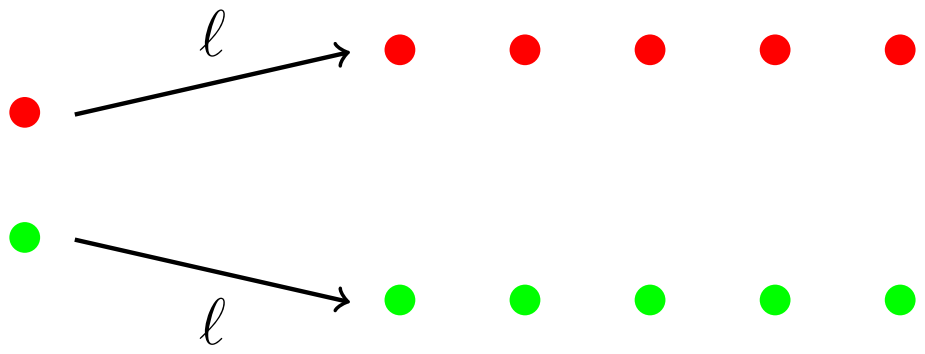


$h : A \mapsto \widetilde{A}$ is agnostic to the clique and can be computed in P-time

Given an integer ℓ, two probability distributions P, Q on $\{0, 1, \ldots, \ell^2\}$

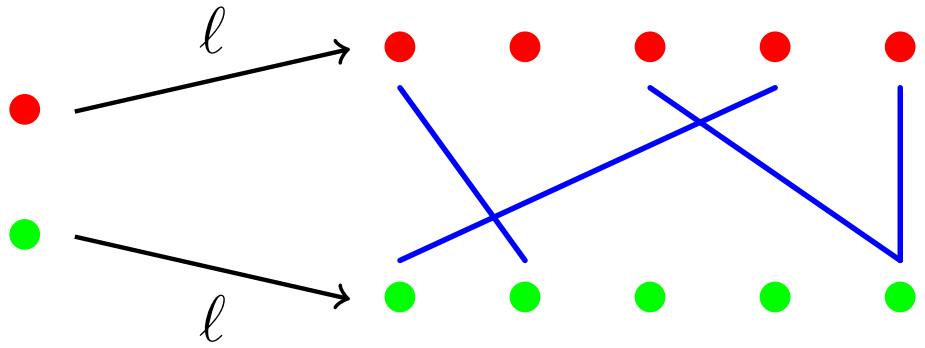
Split each node into ℓ new nodes

$N = n\ell, K = k\ell$



Given an integer ℓ, two probability distributions P, Q on $\{0, 1, \ldots, \ell^2\}$

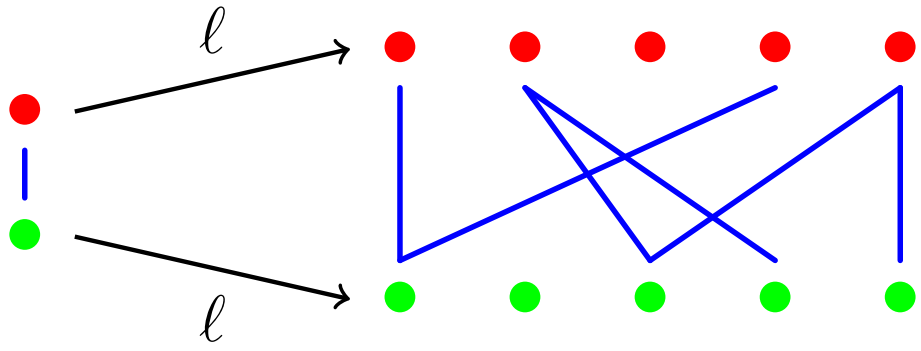
Split each node into ℓ new nodes $N = n\ell, K = k\ell$



Assign edges with distributions P, Q

Given an integer ℓ, two probability distributions P, Q on $\{0, 1, \ldots, \ell^2\}$

Split each node into ℓ new nodes $N = n\ell, K = k\ell$



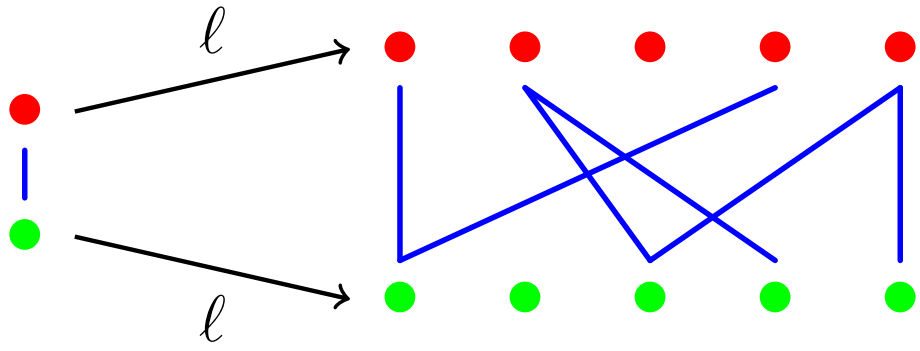
Assign edges with distributions P, Q

$0 \quad \mapsto \quad Q$

$1 \quad \mapsto \quad P$

Given an integer ℓ, two probability distributions P, Q on $\{0, 1, \ldots, \ell^2\}$

Split each node into ℓ new nodes $N = n\ell, K = k\ell$



Assign edges with distributions P, Q

$$0 \quad \mapsto \quad Q$$

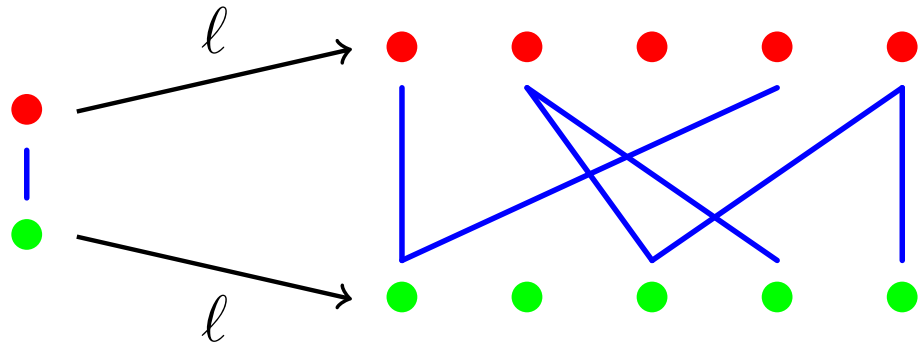
$$1 \quad \mapsto \quad P$$

$$H_0 : \quad \mathrm{Bern}(\gamma) \qquad (1-\gamma)Q + \gamma P$$

$$H_1 : \quad \mathrm{Bern}(1) \text{ (in-clique)} \qquad P \text{ (in-cluster)}$$

Given an integer ℓ, two probability distributions P, Q on $\{0, 1, \ldots, \ell^2\}$

Split each node into ℓ new nodes $N = n\ell, K = k\ell$



Assign edges with distributions P, Q

$$0 \mapsto Q$$

$$1 \mapsto P$$

$$H_0 : \quad \mathrm{Bern}(\gamma) \qquad (1-\gamma)Q + \gamma P$$

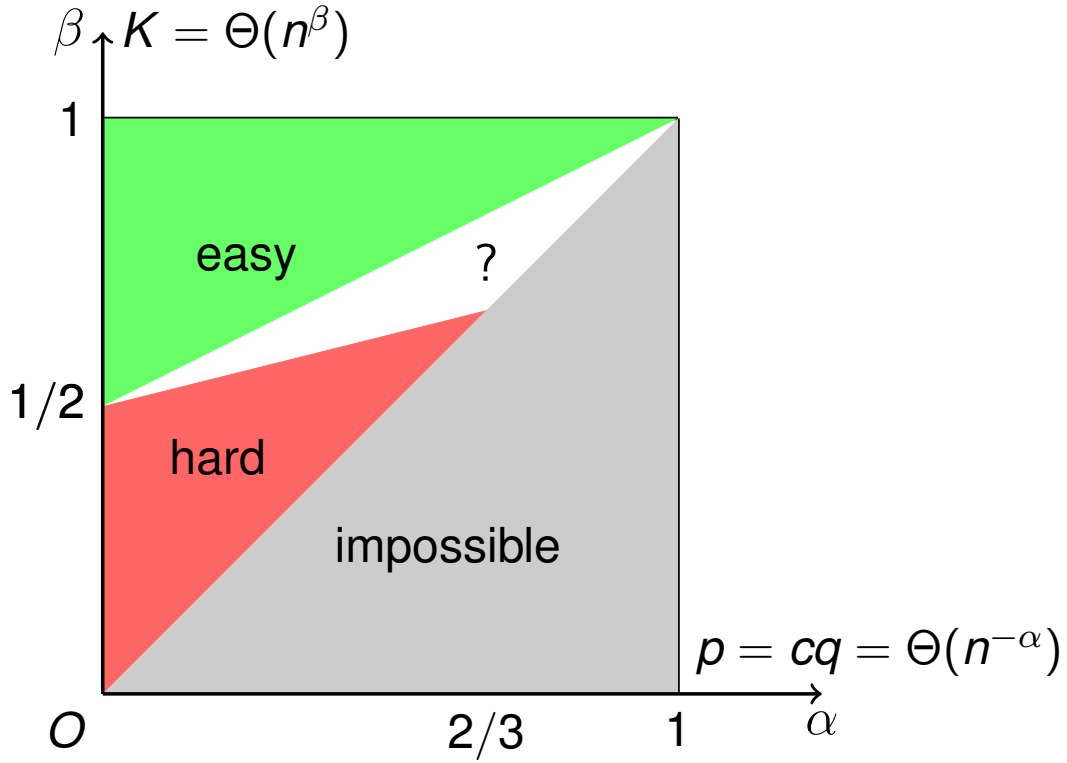
$$H_1 : \quad \mathrm{Bern}(1) \text{ (in-clique)} \qquad P \text{ (in-cluster)}$$

How to choose P, Q?

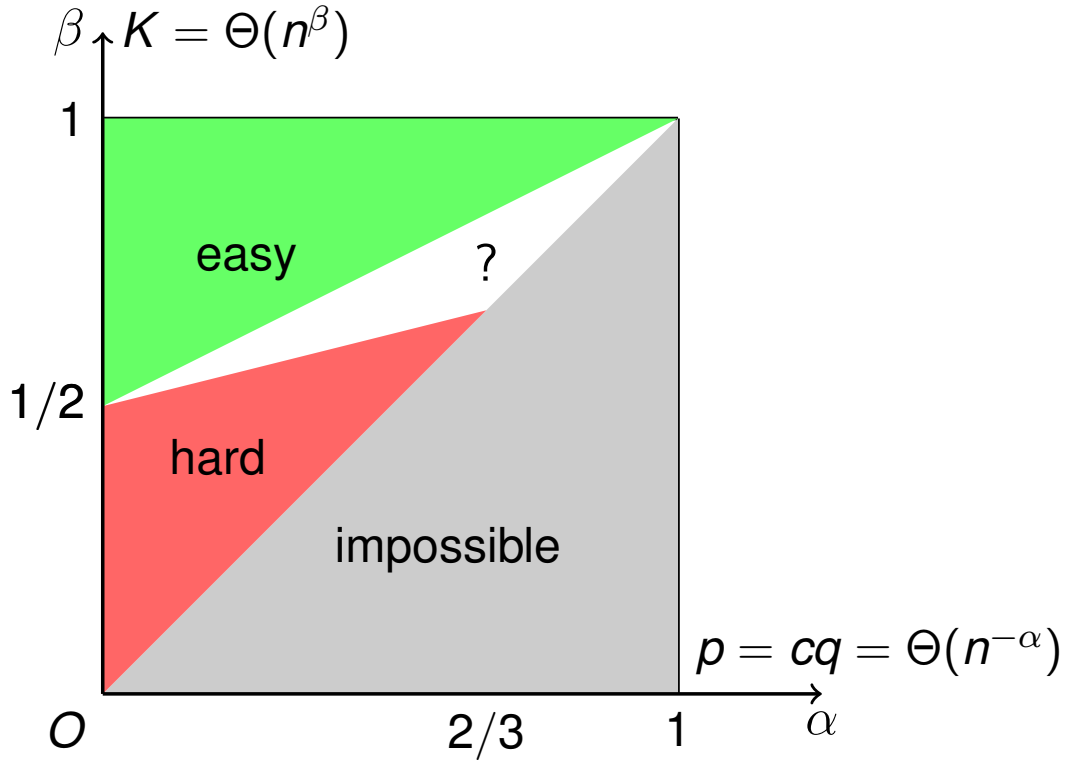
Matching H_0: $(1-\gamma)Q + \gamma P = \mathrm{Binom}(\ell^2, q)$

Matching H_1 approximately: $P \approx \mathrm{Binom}(\ell^2, p)$ in total variation distance

Summary and open problems



Summary and open problems



Open problems

- Optimal recovery via SDP in r unequal-sized clusters

For more information and references see:

1. Y. Chen and J. Xu, "Finding a growing number of planted clusters and submatrices: fundamental limits and statistical-computational tradeoffs," *Submitted to Journal of Machine Learning Research.*, 2014. A preliminary version appeared in Proceedings of 2014 International Conference on Machine Learning.

2. B. Hajek, Y. Wu, and J. Xu, "Achieving exact cluster recovery threshold via semidefinite programming," *Submitted to IEEE Trans. Inf. Theory (On arXiv, Nov. 2014)*, 2014.

3. B. Hajek, Y. Wu, and J. Xu, "Achieving exact cluster recovery threshold via semidefinite programming: Extensions," *arXiv:1502.07738*, 2015.

4. B. Hajek, Y. Wu, and J. Xu, "Computational lower bounds for community detection on random graphs," *arXiv:1406.6625*, 2014.

Formal statement of hardness of detecting a cluster

γ: edge probability in Planted Clique

Theorem

Assume Planted Clique Hypothesis holds for all $0 < \gamma \leq 1/2$*. Let* $\alpha > 0$ *and* $0 < \beta < 1$ *be such that*

$$\alpha < \beta < \frac{1}{2} + \frac{\alpha}{4}.$$

Then there exists a sequence $\{(N_\ell, K_\ell, q_\ell)\}_{\ell \in \mathbb{N}}$ *satisfying* $\lim_{\ell \to \infty} \frac{-\log q_\ell}{\log N_\ell} = \alpha$ *and* $\lim_{\ell \to \infty} \frac{\log K_\ell}{\log N_\ell} = \beta$ *such that for any sequence of randomized polynomial-time tests* ϕ_ℓ *for the* PDS$(N_\ell, K_\ell, 2q_\ell, q_\ell)$ *problem, the Type-I+II error probability is lower bounded by* 1*.*

Proof ideas: Reduce from Planted Clique in polynomial-time

Bound the total variation distance

Lemma

Let $\ell, n \in \mathbb{N}$, $k \in [n]$ *and* $\gamma \in (0, \frac{1}{2}]$. *Let* $N = \ell n$, $K = k\ell$, $p = 2q$ *and* $m_0 = \lfloor \log_2(1/\gamma) \rfloor$. *Assume that* $16q\ell^2 \leq 1$ *and* $k \geq 6e\ell$. *If* $G \sim \mathcal{G}(n, \gamma)$, *then* $\widetilde{G} \sim \mathcal{G}(N, q)$. *If* $G \sim \mathcal{G}(n, k, 1, \gamma)$, *then*

$$d_{\mathrm{TV}}\left(P_{\widetilde{G}}, \mathcal{G}(N, K, p, q)\right) \lesssim e^{-K} + ke^{-\ell} + k^2(q\ell^2)^{m_0+1} + \sqrt{e^{q\ell^2} - 1}$$

Bound the total variation distance

Lemma

Let $\ell, n \in \mathbb{N}$, $k \in [n]$ *and* $\gamma \in (0, \frac{1}{2}]$. *Let* $N = \ell n$, $K = k\ell$, $p = 2q$ *and* $m_0 = \lfloor \log_2(1/\gamma) \rfloor$. *Assume that* $16q\ell^2 \leq 1$ *and* $k \geq 6e\ell$. *If* $G \sim \mathcal{G}(n, \gamma)$, *then* $\widetilde{G} \sim \mathcal{G}(N, q)$. *If* $G \sim \mathcal{G}(n, k, 1, \gamma)$, *then*

$$d_{\mathrm{TV}}\left(P_{\widetilde{G}}, \mathcal{G}(N, K, p, q)\right) \lesssim e^{-K} + ke^{-\ell} + k^2(q\ell^2)^{m_0+1} + \sqrt{e^{q\ell^2} - 1}$$

Proof ideas: $d_{\mathrm{TV}}(P, Q) \leq \frac{1}{2}\sqrt{\chi^2(P, Q)}$ and use negative associations [Dubhashi-Ranjan '98] to get rid of dependency in calculating the χ^2 distance.

Bound the total variation distance

Lemma

Let $\ell, n \in \mathbb{N}$, $k \in [n]$ *and* $\gamma \in (0, \frac{1}{2}]$. *Let* $N = \ell n$, $K = k\ell$, $p = 2q$ *and* $m_0 = \lfloor \log_2(1/\gamma) \rfloor$. *Assume that* $16q\ell^2 \le 1$ *and* $k \ge 6e\ell$. *If* $G \sim \mathcal{G}(n, \gamma)$, *then* $\widetilde{G} \sim \mathcal{G}(N, q)$. *If* $G \sim \mathcal{G}(n, k, 1, \gamma)$, *then*

$$d_{\mathrm{TV}}\left(P_{\widetilde{G}}, \mathcal{G}(N, K, p, q)\right) \lesssim e^{-K} + ke^{-\ell} + k^2(q\ell^2)^{m_0+1} + \sqrt{e^{q\ell^2} - 1}$$

Proof ideas: $d_{\mathrm{TV}}(P, Q) \le \frac{1}{2}\sqrt{\chi^2(P, Q)}$ and use negative associations [Dubhashi-Ranjan '98] to get rid of dependency in calculating the χ^2 distance.

Apply the Lemma by choosing $q = \ell^{-2-\delta}$ so that $q\ell^2 \to 0$: $N = \ell^{\frac{2+\delta}{\alpha}}$, $K = \ell^{\frac{(2+\delta)\beta}{\alpha}}$, $n = \ell^{\frac{2+\delta}{\alpha}-1}$, $k = \ell^{\frac{(2+\delta)\beta}{\alpha}-1}$. Easy to check that

$$\alpha < \beta < \frac{1}{2} - \delta + \frac{\alpha(1+2\delta)}{4+2\delta} \Rightarrow \frac{\log k}{\log n} \le \frac{1}{2} - \delta$$