Community Detection via Random and Adaptive Sampling

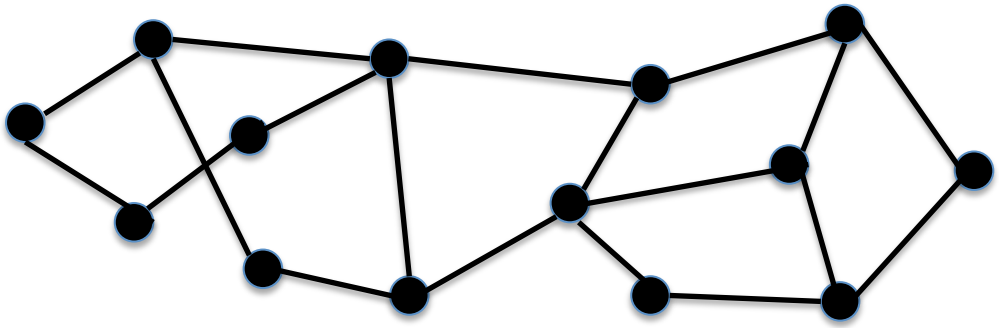
Se-Young Yun (MSR-Inria)
Alexandre Proutiere (KTH)

Community detection in networks

Objective: Extract *K* communities in a network of *n* nodes from random *observations*

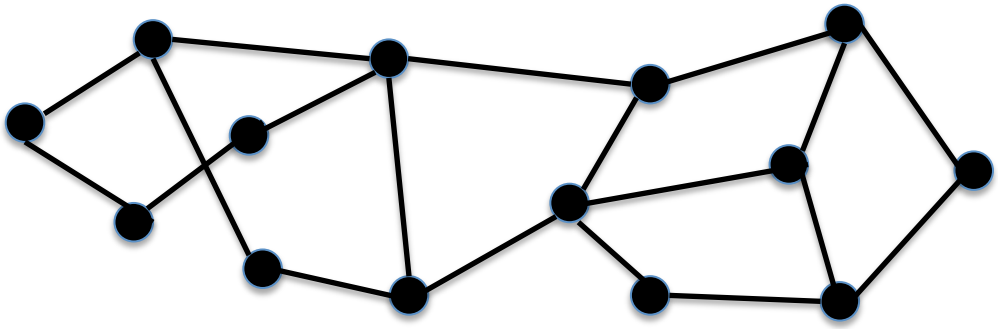
Observations

1. A graph (e.g. the stochastic block model)



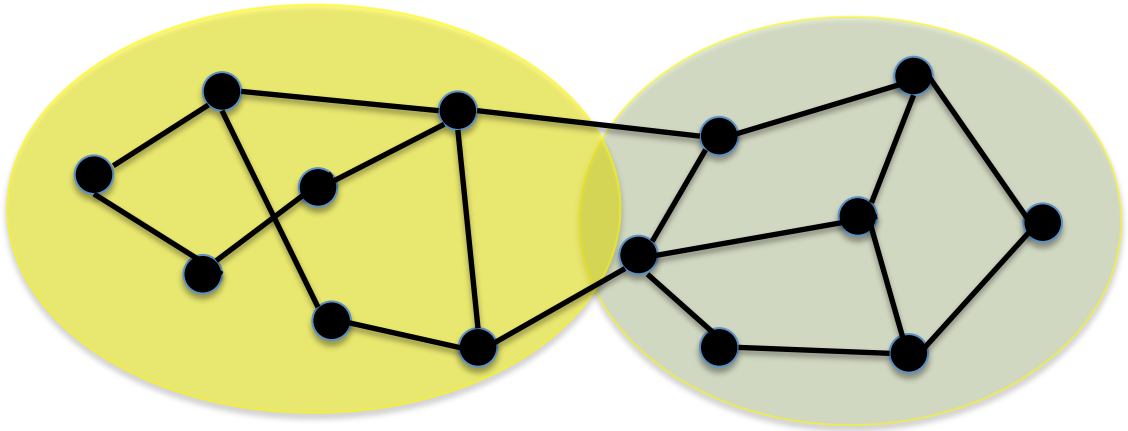
2. This talk: a more general sampling framework

1. Community detection in graphs



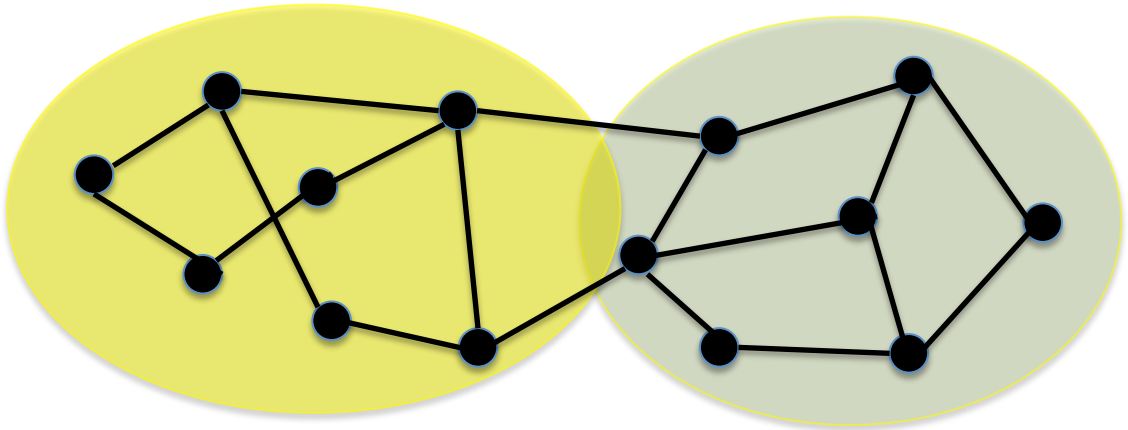
- Find communities in a graph

1. Community detection in graphs



- Find communities in a graph
- Applications: biology (the role of proteins), social networks (targeted ads), distributed computing (balanced partitions), …
- Large graphs: e.g. web: > 1 billion pages

1. Community detection in graphs

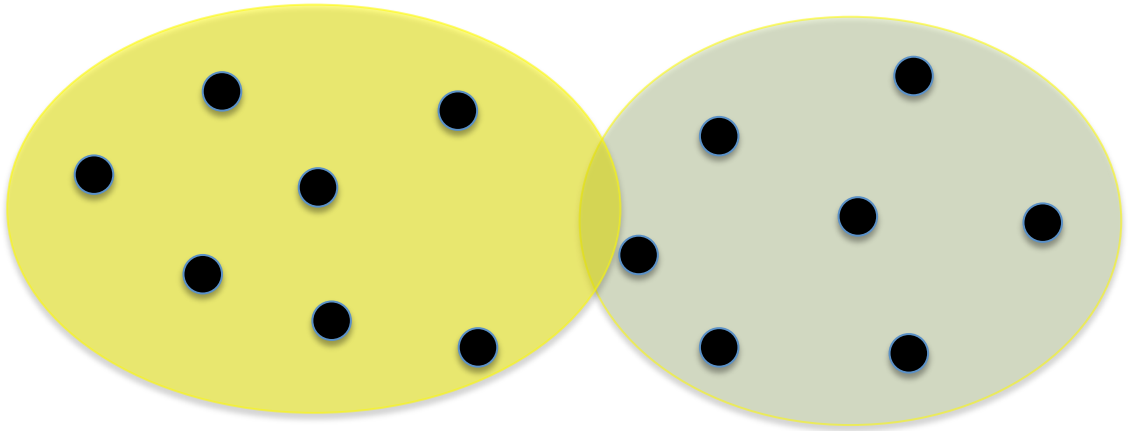


- Objective: find conditions on the graph under which communities can be efficiently detected using low complexity algorithms

Related work

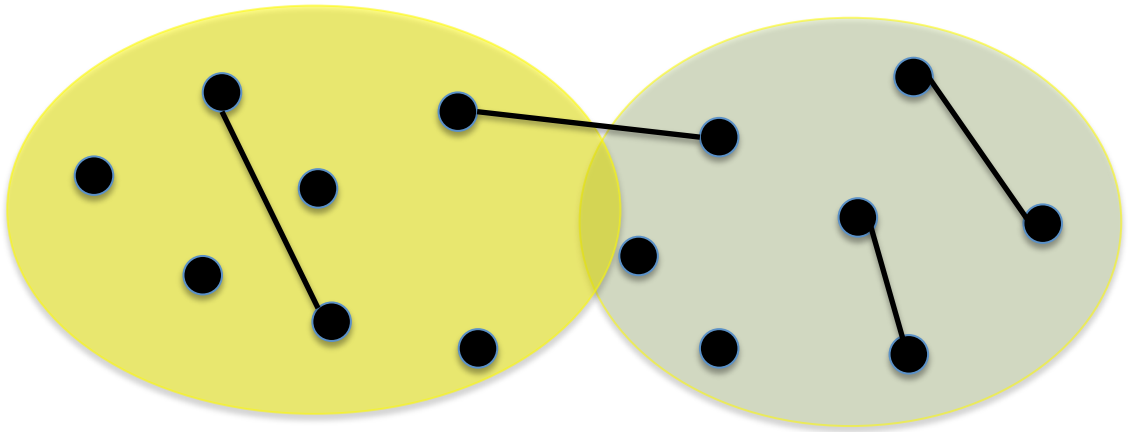
Arora, Rao, Newman, Coja-Oghlan, Jerrum, Chen, Frieze, McSherry, Dyer, Sorkin, Kannan, Vempala, Vetta, Fortunato, Decelle, Krzakala, Karp, Condon, Reichart, Sanghavi, Nadakuditi, Girvan, Mosel, Sly, Rohe, Chatterjee, Yu, Massoulie, Lelarge, Vazirani, Karger, Feld, Fischer, Kleinberg, Gibson, Raghavan, Hopcroft, Khan, Kulis, Santo, Wellman, Hogan, Berg, White, Boorman, Kelley, Xie, Kumar, Mathieu, Schudy, Alon, Krivelevich, Sudakov, Xu, Achlioptas, Kahale, Feige, Zdeborova, Carson, Giesen, Mitshe, Shamir, Tsur, Hassibi, Oymak, Ames, Parrilo, Holland, Laskley, Pothen, Simon, Liou, Girvan, Chauhan, Leone, Ball, Karrer, …

Stochastic Block (SB) model



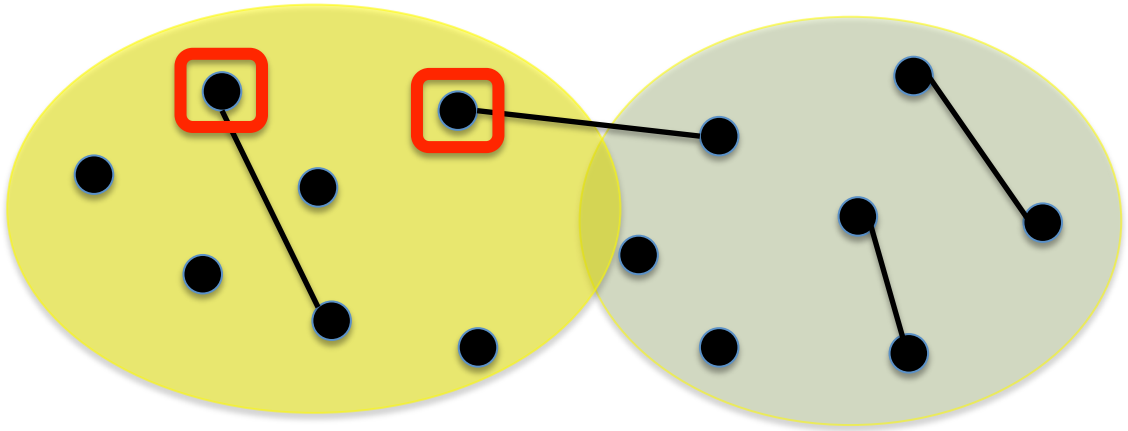
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



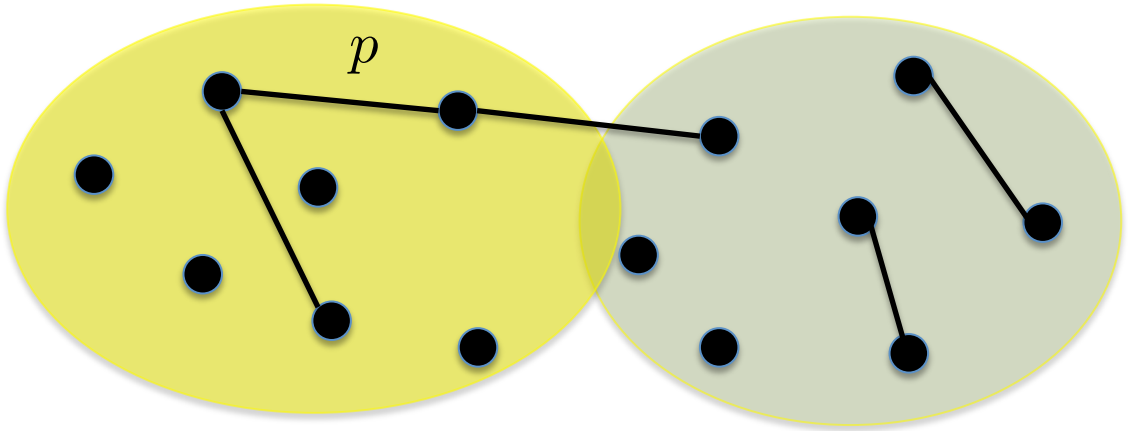
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



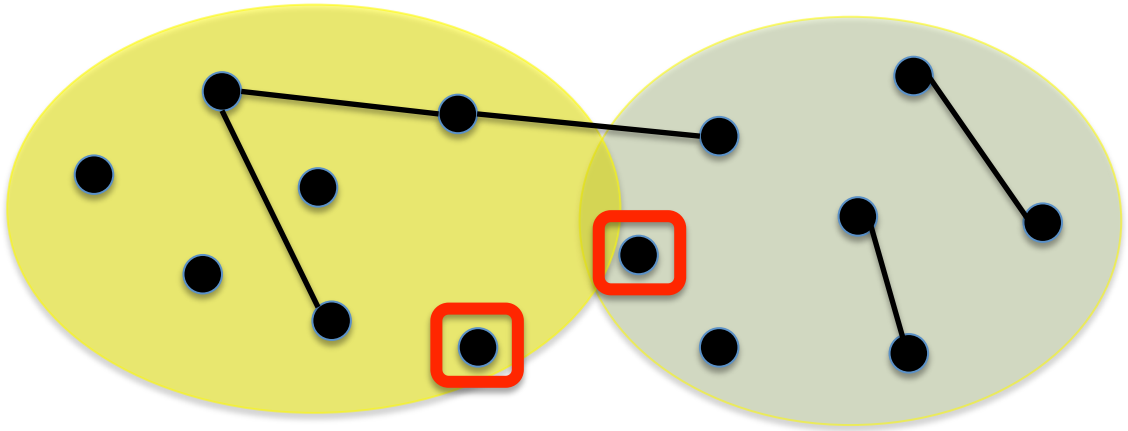
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



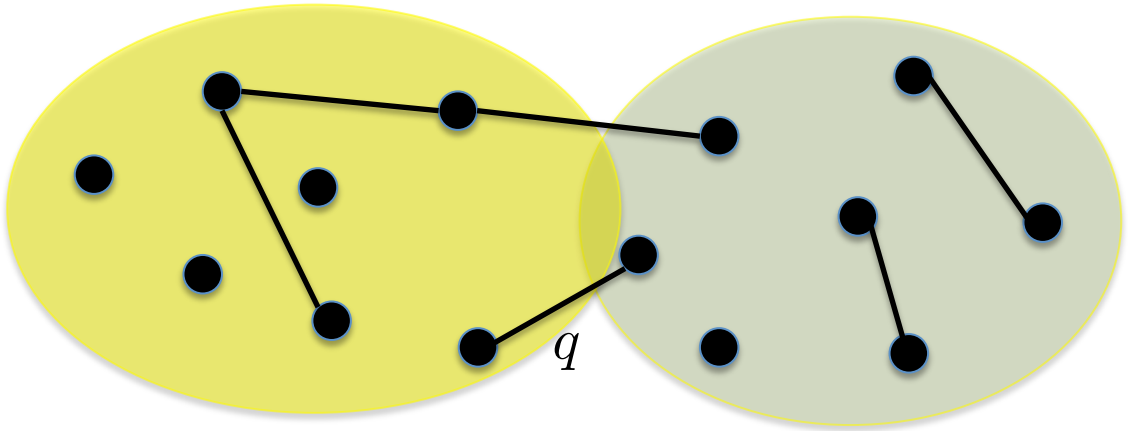
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



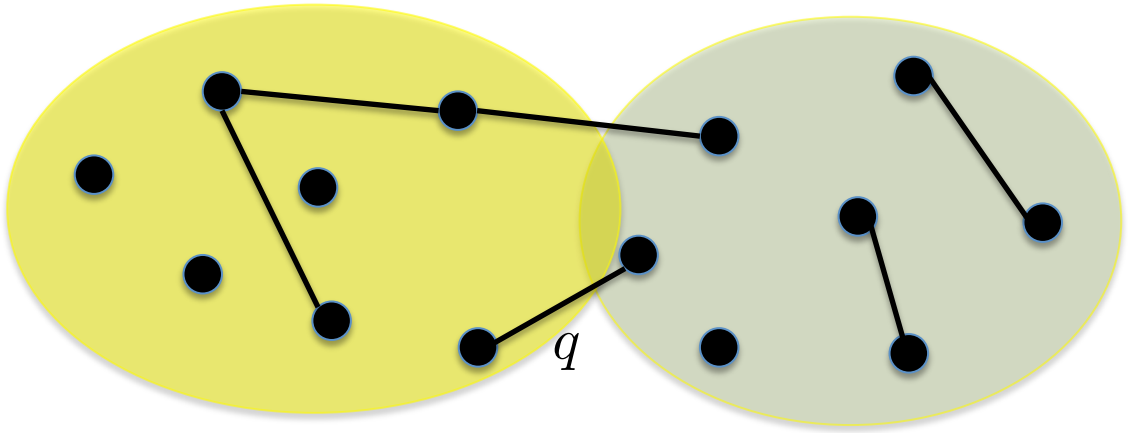
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



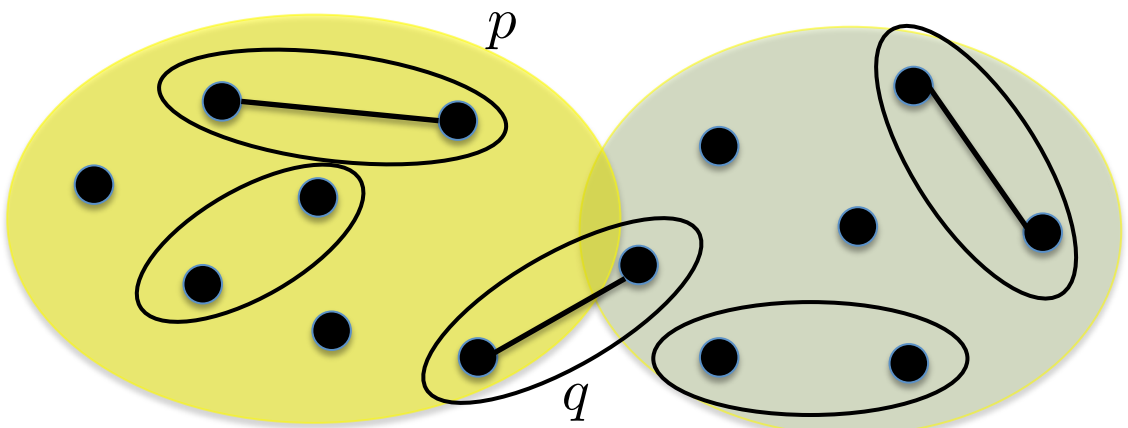
- The graph is built by considering each pair of nodes once
  - If in the same community: put an edge with probability p
  - Else: put an edge with probability $q < p$

Stochastic Block (SB) model



- Network size: *n* nodes, *n* tends to ∞
- Sparse interaction: *p, q* = o(1)
- Dense interaction: *p, q* = O(1)

2. General Sampling Framework



- The interaction of two nodes can be sampled repeatedly
- Sample for a given node pair: Bernoulli with mean *p* if nodes are in the same cluster, with mean *q* otherwise
- Sample budget: *T*

Sampling Strategies

- **Non-adaptive Random Strategies**
  - The pair of nodes sampled in round *t* does not depend on past observations, and is chosen uniformly at random
  - S1: sampling with replacement
  - S2: sampling without replacement

- **Adaptive Strategies**
  - The pair of nodes sampled in round *t* depends on past observations

- Stochastic Block Model: random sampling without replacement, and $T = n(n-1)/2$

Performance metrics

Proportion of misclassified nodes under π: $\varepsilon^{\pi}(n, T)$

1. Asymptotic detection: an algorithm *detects* the clusters if it does better than the algorithm that randomly assigns nodes to clusters

2. Accurate asymptotic detection: a joint sampling and clustering algorithm π is asymptotically accurate if

$$\lim_{n \to \infty} \mathbb{E}[\varepsilon^{\pi}(n, T)] = 0.$$

Objectives of this work

1. Identify necessary and sufficient conditions on p, q, n, T for the existence of asymptotically accurate algorithms, and for both random and adaptive sampling strategies

2. Design asymptotically optimal algorithms for both random and adaptive sampling strategies

Outline

1. The Stochastic Block Model

2. Fundamental limits: Conditions for asymptotically accurate detection in the general sampling framework

3. Optimal algorithms for random and adaptive sampling strategies

1. The Stochastic Block Model

A few results

- Dyer-Frieze 1989: fixed *p* and *q*, min bisection in expected O(n^3) time.
- Jerrum-Sorkin 1998: $p - q \geq n^{-\frac{1}{6}+\epsilon}$, expected running time O($n^{2+\varepsilon}$), further improvement by Condon-Karp.
- McSherry 2001: $p - q \geq \Omega(\sqrt{q \log(n)/n})$, degree $\log^6(n)$
- …
- Sparse graphs are more difficult

Sparse graphs, 2 communities

- SB model: sparse regime $p = \frac{a}{n}, \quad q = \frac{b}{n}$

Theorem (Mossel-Neeman-Sly 2012)

If $a - b < \sqrt{2(a+b)}$, then clusters are not detectable.

Conjectured by Decelle-Krzakala-Moore-Zdeborova 2012

Theorem (Massoulie 2013)

If $a - b > \sqrt{2(a+b)}$, then there exists an algorithm leading to clusters that are positively correlated with the true clusters.

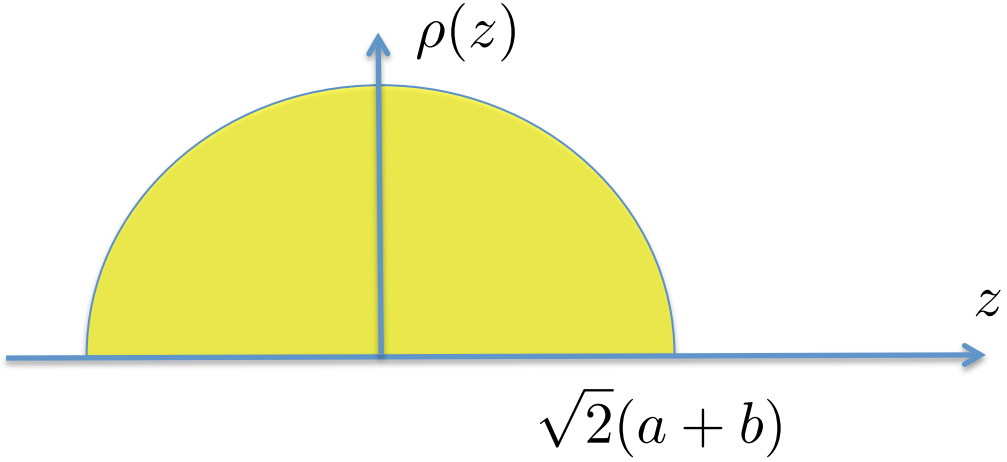
Non-rigorous Spectral Analysis

- Average adjacency matrix

$$\mathbb{E}[A] = \frac{1}{2}(a+b)11^T + \frac{1}{2}(a-b)uu^T$$

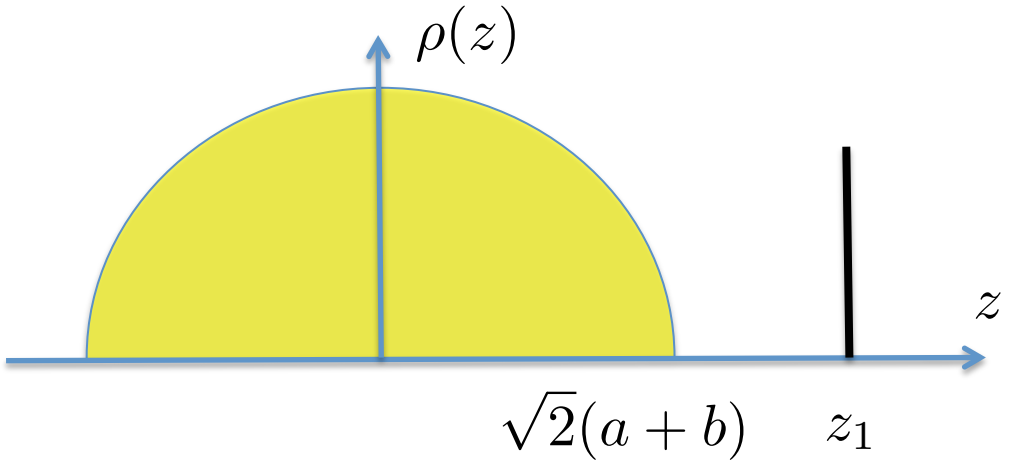
$$1 = \frac{1}{\sqrt{n}}(1,\ldots,1)^T, \quad u = \frac{1}{\sqrt{n}}(1,\ldots,1,-1,\ldots,-1)^T$$

- Noisy observation: $A = \mathbb{E}[A] + X$
- Spectral density of noise matrix X (Wigner semicircle law)



Non-rigorous Spectral Analysis

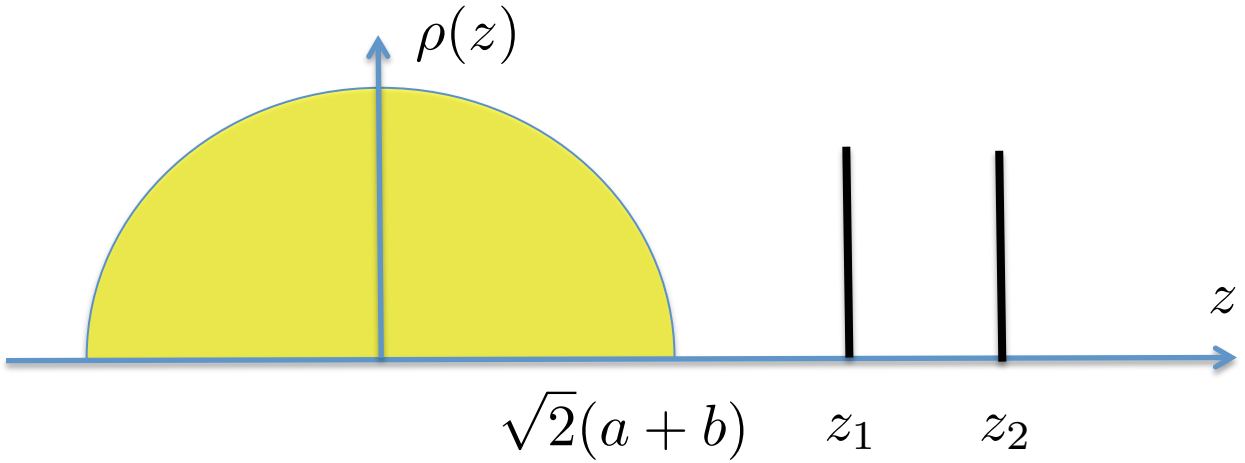
- Spectral density of the modularity matrix: $\frac{1}{2}(a-b)uu^T + X$



$$z_1 = \frac{1}{2}(a-b) + \frac{a+b}{a-b}$$

Non-rigorous Spectral Analysis

- Spectral density of the observed matrix:



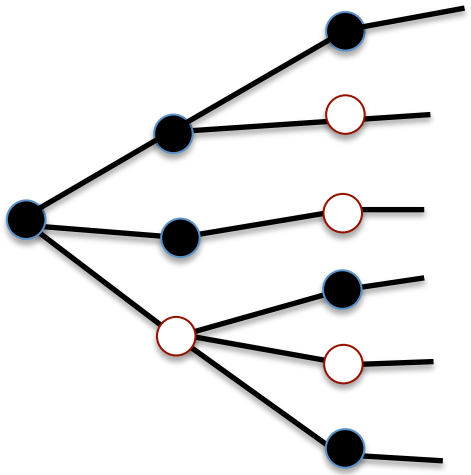
- Communities are detectable if $z_1 > \sqrt{2}(a+b)$
- Method: find z_1 and the corresponding eigen vector u

High-degree nodes

- Lack of rigor: some nodes have high degree, up to $\frac{\log(n)}{\log\log(n)}$
- It perturbs the spectral density!
- The problem vanishes when $p, q \gg 1/n$

Rigorous proof (Mossel et al.)

- Analogy with the tree reconstruction problem
- Each node (labeled with its type) gives birth to Poi(a) nodes of the same type, and Poi(b) of node of different types



Can we recover the type of the root by observing the types of nodes at large level r?

- Impossible if $a - b < \sqrt{2(a+b)}$ (Evans-Kenyon-Peres-Schulman 2000)

Examples of algorithms

- Maximum Likelihood
  - Belief Propagation
  - Compressed sensing (convex relaxation)

- Spectral approaches
  - Provide a K-rank approximation of the adjacency matrix

  + Trimming + Post-processing

Maximum Likelihood

- Observed adjacency matrix *A*
- Output: a symmetric binary matrix *Y*, rank 2
  - nodes *i* and *j* are in the same community iff $y_{ij} = 1$
- Likelihood:

$$\log \mathbb{P}[A|Y] = \log \prod_{(i,j):y_{ij}=1} p^{a_{ij}}(1-p)^{1-a_{ij}} \prod_{(i,j):y_{ij}=0} \prod q^{a_{ij}}(1-q)^{1-a_{ij}}$$

$$\log \mathbb{P}[A|Y] = \log(\frac{p}{q}) \sum_{(i,j):a_{ij}=1} y_{ij} - \log(\frac{1-q}{1-p}) \sum_{(i,j):a_{ij}=0} y_{ij} + C$$

- Max likelihood: exact solution, Decelle et al. via Belief Propagation algorithm (no analysis)

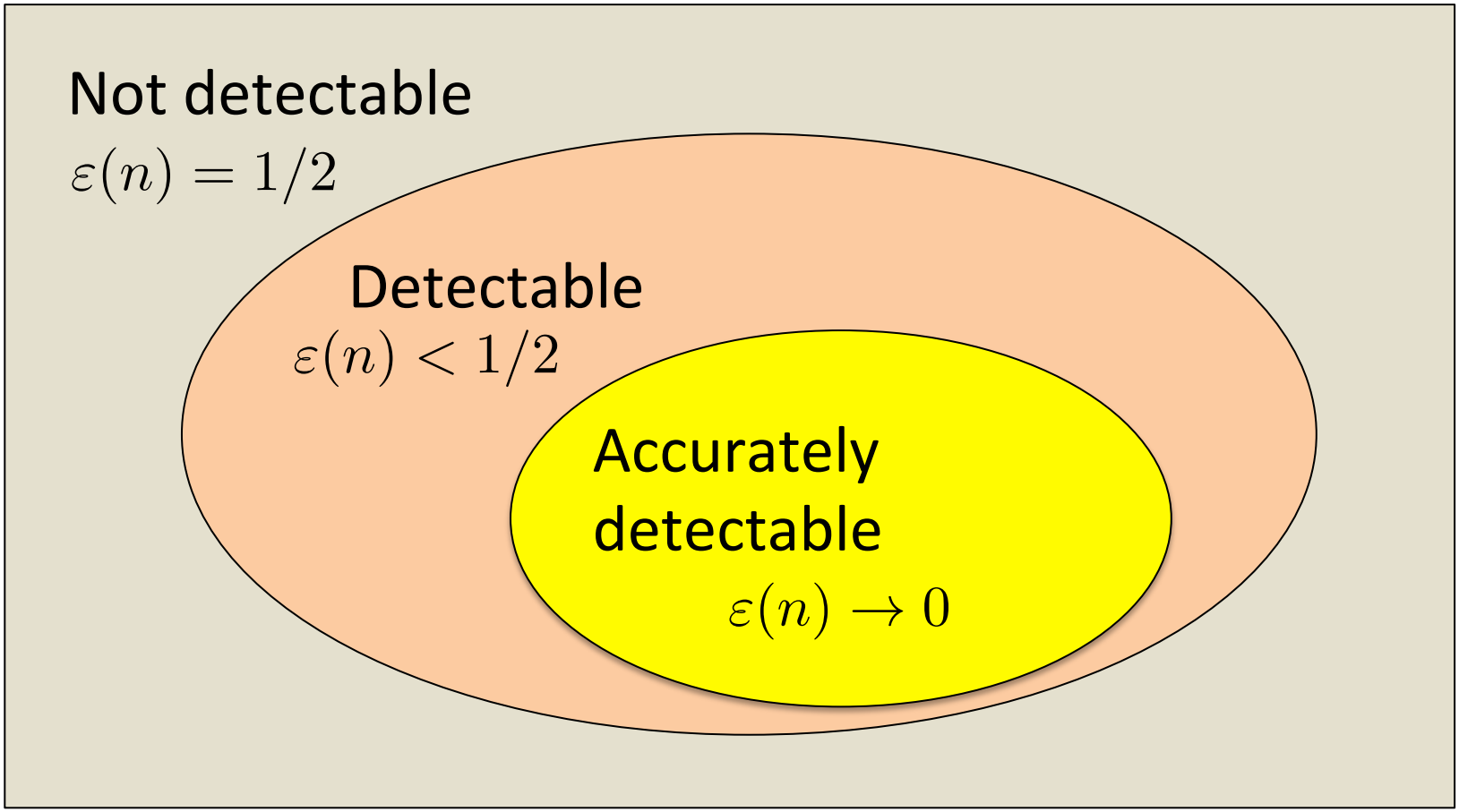
Maximum Likelihood

- Relaxation: compressed sensing approach, Chen et al.
  - First relaxation: $y_{ij} \in [0,1]$
  - Second relaxation: nuclear norm of *Y* (instead of low rank)
- Maximize: (convex program)

$$\log(\frac{p}{q}) \sum_{(i,j):a_{ij}=1} y_{ij} - \log(\frac{1-q}{1-p}) \sum_{(i,j):a_{ij}=0} y_{ij} - K\sqrt{n}\|Y\|_*$$

- Performance guarantee:
  - we get the right solution w.h.p. provided that $\frac{p-q}{\sqrt{p(1-q)}} \geq c_1 \frac{\log(n)^2}{\sqrt{n}}$
  - Example: does not work if $p = \frac{a\log(n)}{n}, q = \frac{b\log(n)}{n}$

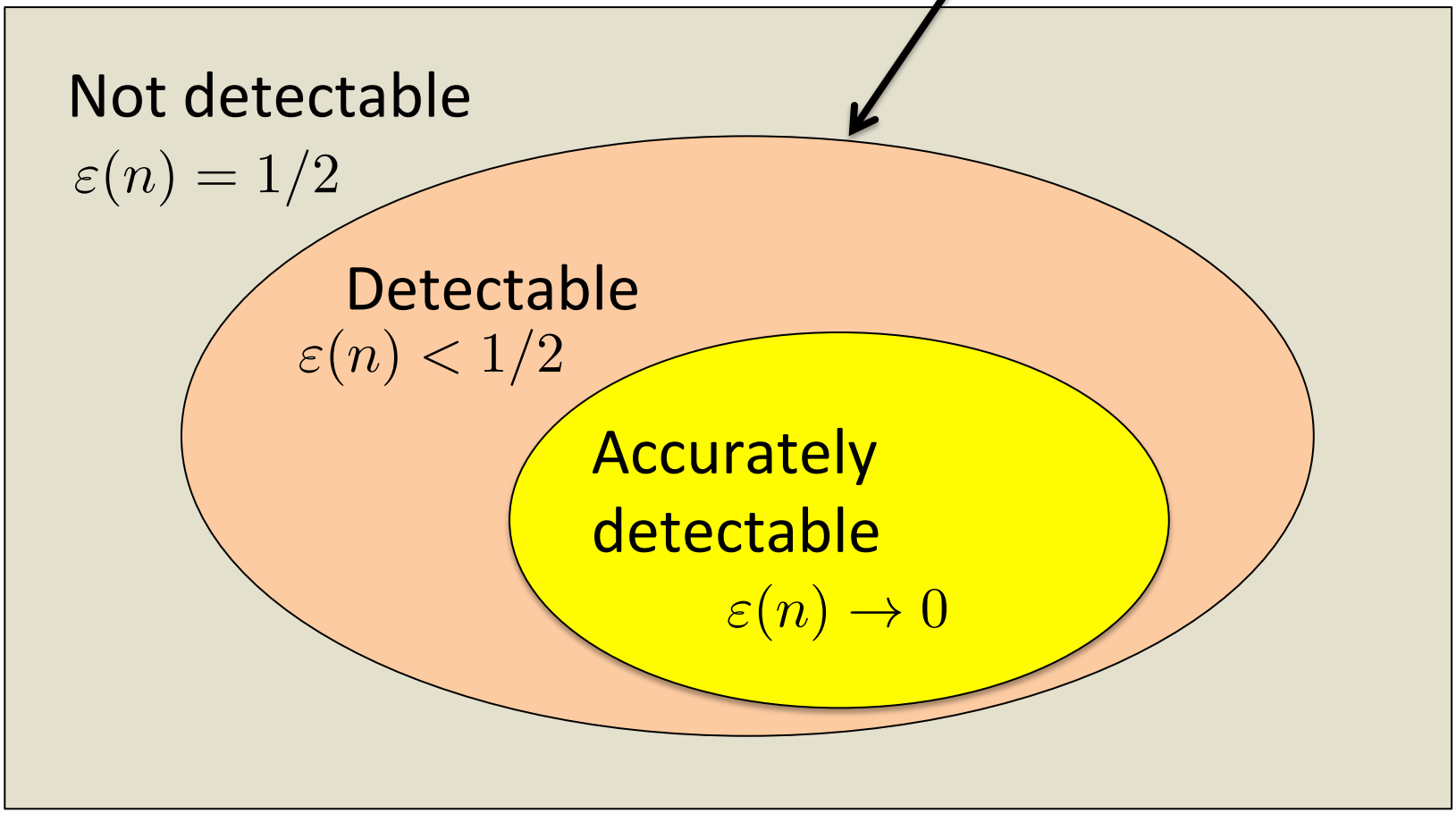
SBM $p = \frac{a}{n}, \quad q = \frac{b}{n}$



SBM

$$p = \frac{a}{n}, \quad q = \frac{b}{n}$$

$$\frac{n(p-q)^2}{p+q} = 2$$

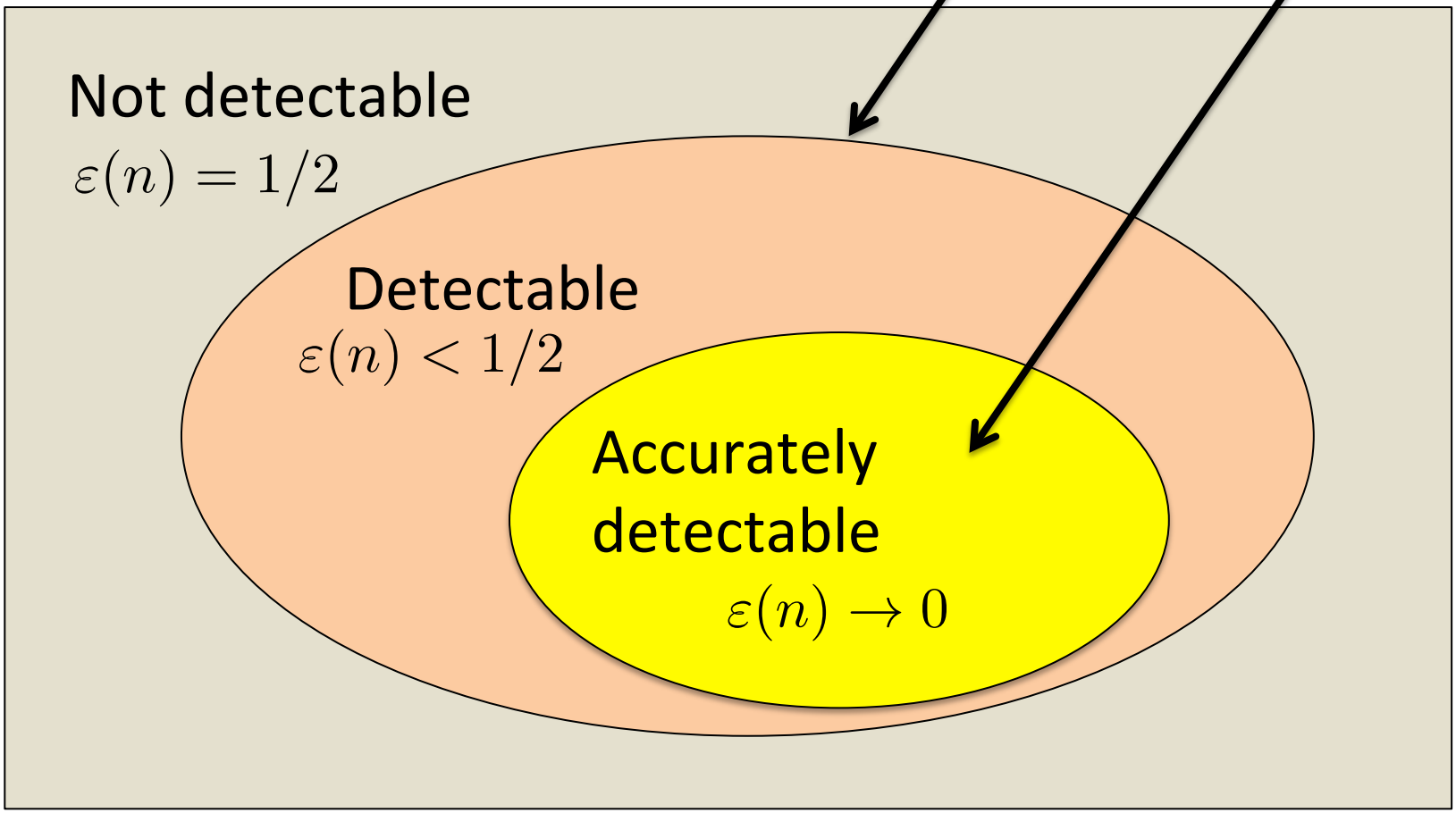


SBM

$p = \frac{a}{n}, \quad q = \frac{b}{n}$

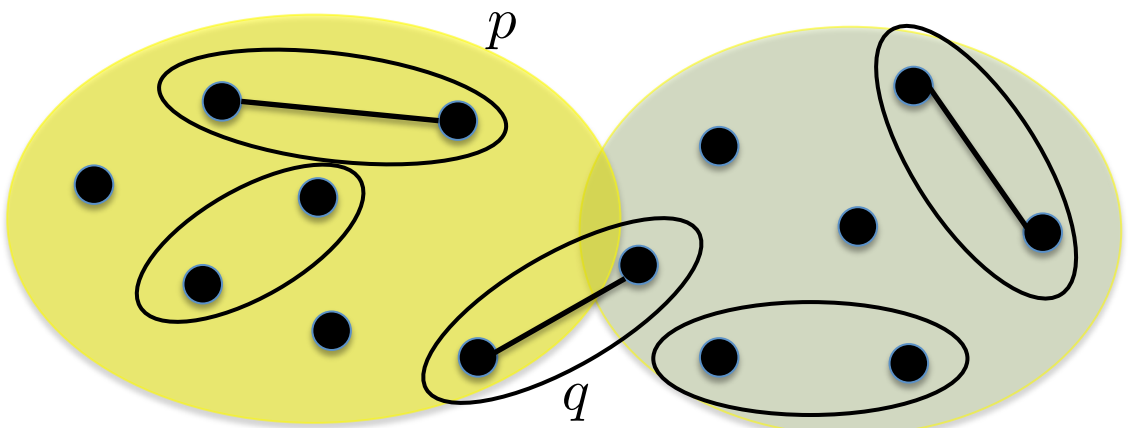
$\frac{n(p-q)^2}{p+q} = 2$

$\emptyset$



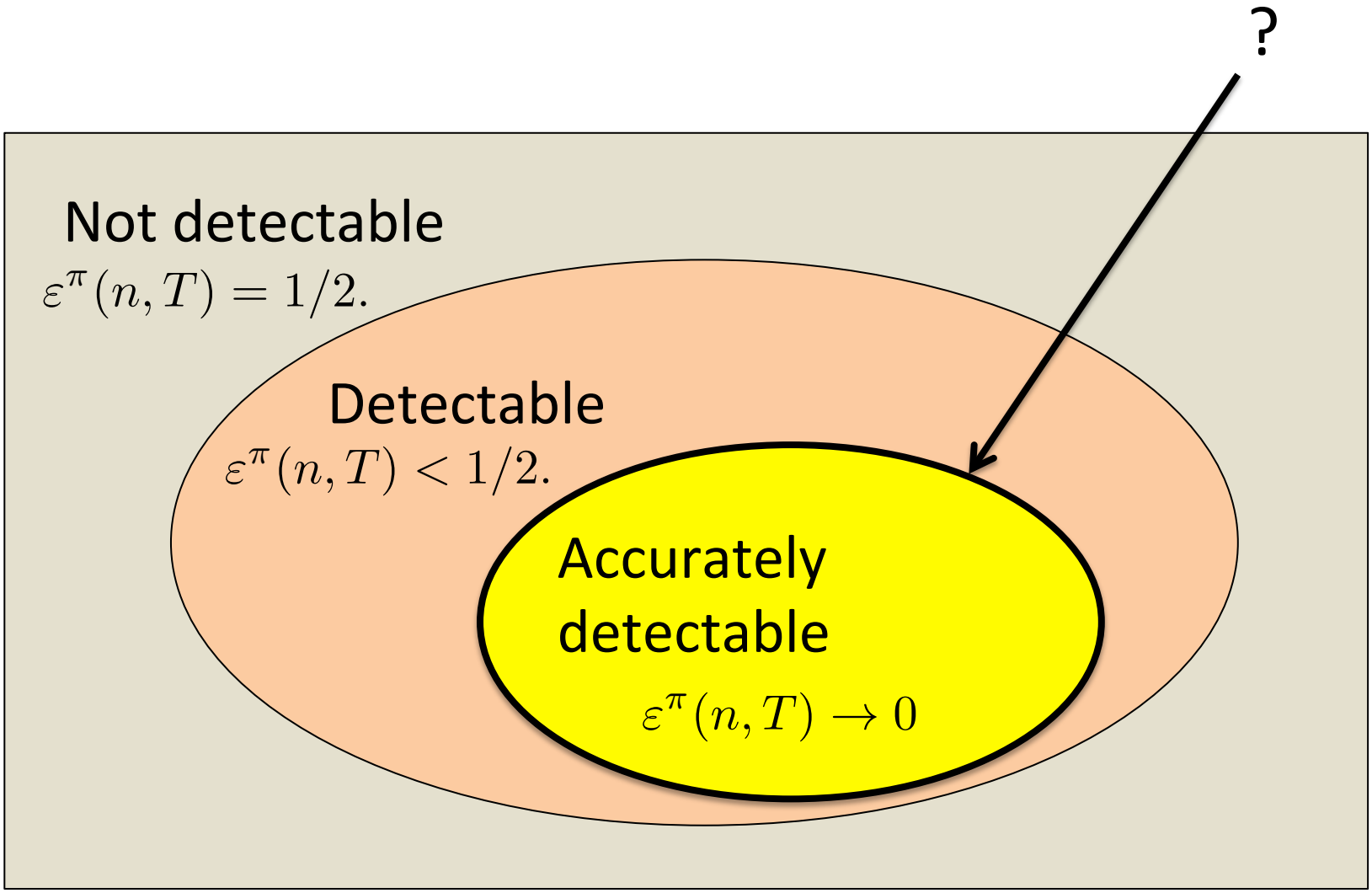
2. Generic Sampling Framework Fundamental limits

General Sampling Framework

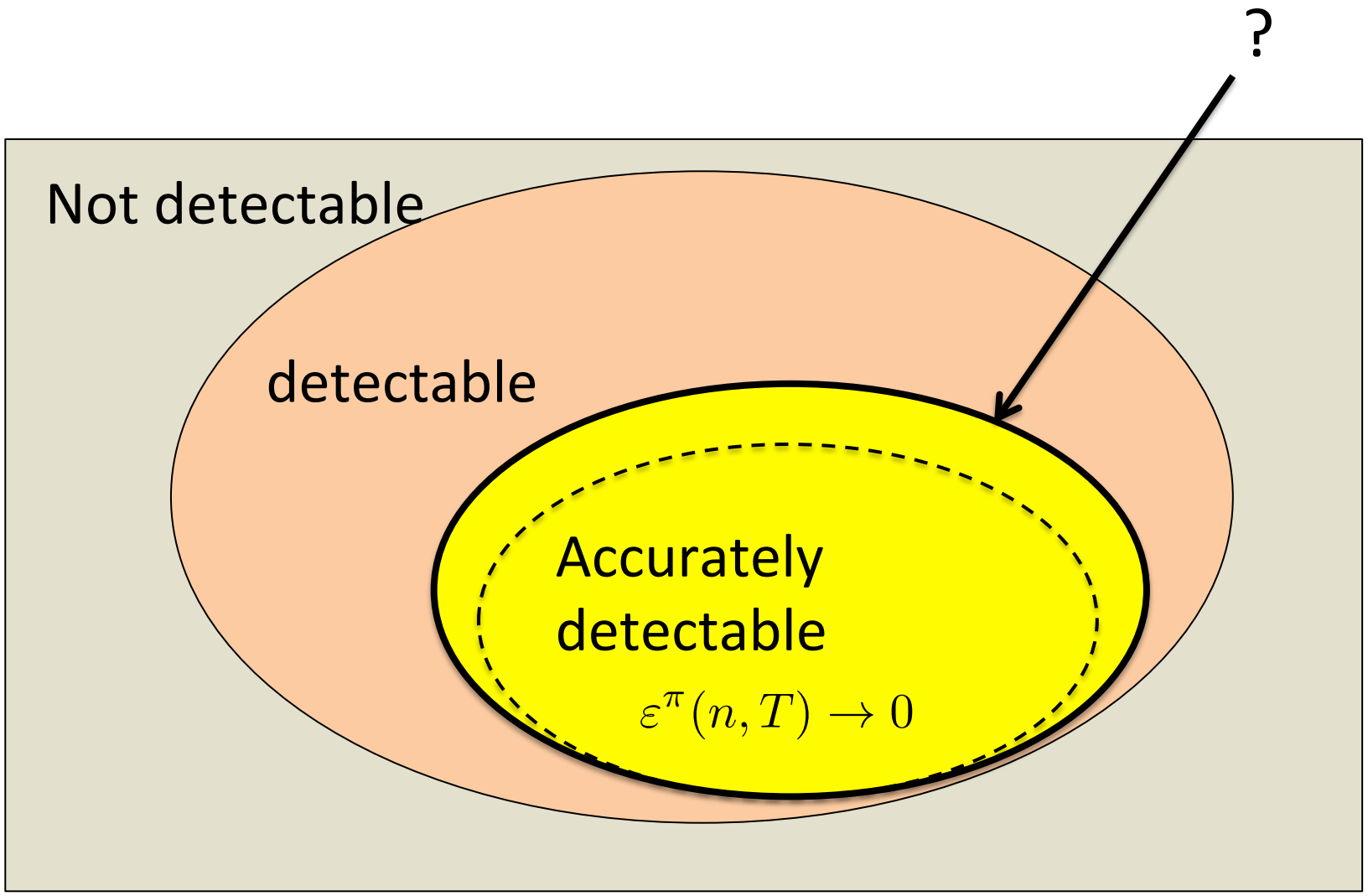


- The interaction of two nodes can be sampled repeatedly
- Sample for a given node pair: Bernoulli with mean *p* if nodes are in the same cluster, with mean *q* otherwise
- Sample budget: *T*

Random Sampling



Adaptive Sampling



Fundamental limits

- Random sampling:

$$\kappa_1(n,T) = T\frac{2(n-2)}{n(n-1)}\min\{KL(q,p), KL(p,q)\}$$

$$+ 2\sqrt{\frac{4T(n-2)}{n(n-1)}\left[\min\{q,1-p\}\left(\log\frac{p(1-q)}{q(1-p)}\right)^2 + \left(\log(\min\{\frac{p}{q},\frac{1-q}{1-p}\})\right)^2\right]}$$

Theorem Under Random sampling strategy S1 or S2, for any clustering algorithm π, we have:

$$\mathbb{E}[\varepsilon^{\pi}(n,T)] \geq \frac{1}{8}\exp(-\kappa_1(n,T)),$$

Fundamental limits

- Non-adaptive random sampling -- necessary conditions for asymptotically accurate detection:

$$\frac{T}{n} = \omega(1), \quad \frac{T}{n}\min(KL(q,p), KL(p,q)) = \omega(1),$$

- Dense interaction: $p, q = \Theta(1)$

$$T(p-q)^2/n = \omega(1)$$

- Sparse interaction: $p, q = o(1)$

$$T(p-q)^2/(pn) = \omega(1)$$

Fundamental limits

- Adaptive sampling:

Theorem For asymptotically accurate detection, we need:

$$\min\{p, 1-q\}\frac{T}{n} = \Omega(1) \quad \text{and} \quad \frac{T}{n}\max(KL(q,p), KL(p,q)) = \omega(1).$$

- Example: $p = \frac{\log n}{n} \qquad q = \frac{\sqrt{\log n}}{n}$
  - Non-adaptive sampling: $\frac{T}{n} = \omega(\frac{n}{\log(n)})$
  - Adaptive sampling: $\frac{T}{n} = \Omega(\frac{n}{\log(n)})$

Fundamental limits

- Adaptive sampling:

Theorem For asymptotically accurate detection, we need:

$$\min\{p, 1-q\}\frac{T}{n} = \Omega(1) \quad \text{and} \quad \frac{T}{n}\max(KL(q,p), KL(p,q)) = \omega(1).$$

- Example: $p = \frac{a\log n}{n} \qquad q = \frac{b\log n}{n}$
  - Non-adaptive sampling: $\frac{T}{n} = \omega(\frac{n}{\log(n)})$
  - Adaptive sampling: $\frac{T}{n} = \omega(\frac{n}{\log(n)})$

2. Generic Sampling Framework Algorithms

Algorithms for non-adaptive sampling

- Spectral algorithms (extension of Coja-Oghlan's algorithm)

1. From random samples, build an observation matrix
2. Trimming (remove nodes with too many interactions)
3. Spectral decomposition (find the largest eigenvalues and corresponding eigenvectors)
4. Greedy improvement (for each node compare the number of interactions with the various clusters)

Performance

Theorem Assume that:

$$\frac{(p-q)^2}{p}\frac{\alpha T}{n} = \omega(1), \quad \frac{(p-q)^2}{p}\frac{\alpha T}{n} \geq \log(p\frac{T}{n}).$$

Then with high probability:

$$\varepsilon^{SP}(n,T) \leq 8\exp\left(-\frac{(p-q)^2}{20p}\frac{\alpha T}{n}\right).$$

- The algorithm is asymptotically accurate under the necessary conditions for accurate detection in the case of random sampling
- The necessary conditions for accurate detection are tight!

Algorithms for adaptive sampling

- Spatial coupling idea: find reference kernels and build the clusters from these kernels

1. Kernels: select n/log(n) nodes and use T/5 samples to classify these nodes (using the previous spectral algorithm)
2. Select one of remaining nodes. Sample T/3n pairs between the selected nodes to each kernel. Classify the node.
3. Repeat 2. until no remaining node or budget

Performance

Theorem Assume that:

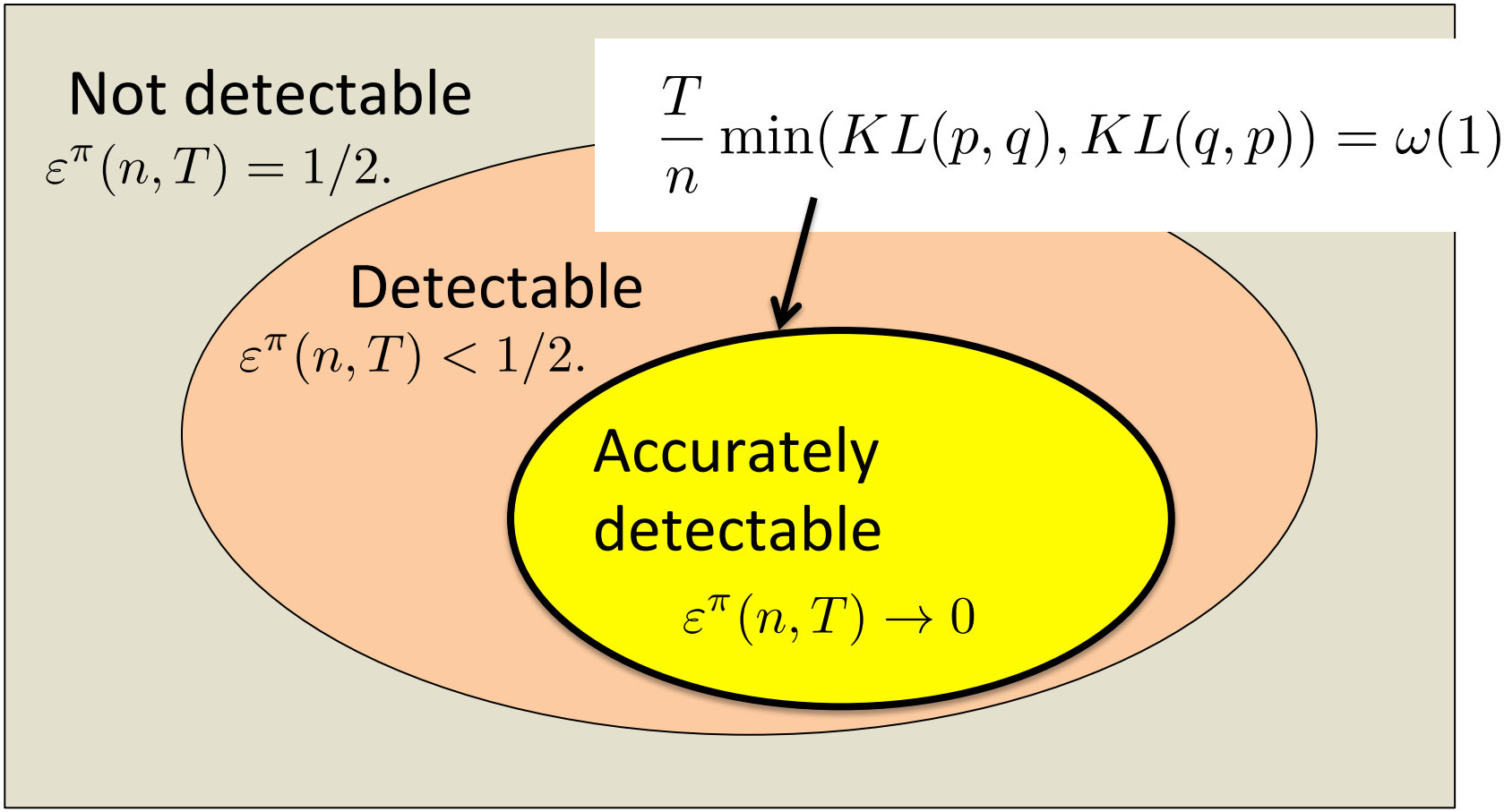
$$\frac{(p-q)^2}{p+q}\frac{T}{n} = \Omega(1), \qquad \frac{T}{n}\max(KL(q,p), KL(p,q)) = \omega(1).$$

Then with high probability:

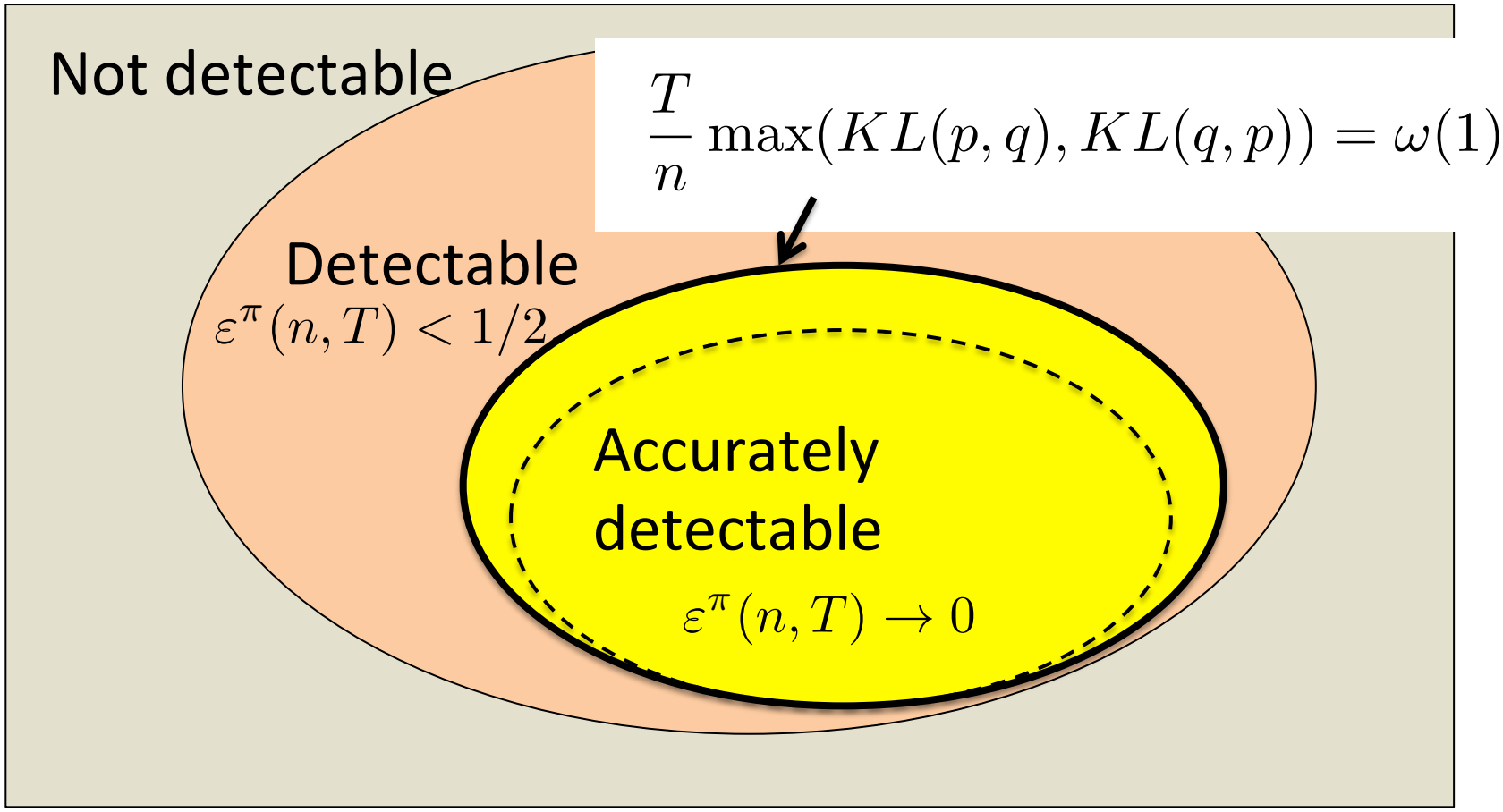
$$\varepsilon^{ASP}(n,T) \leq \exp\left(-\frac{T}{6n}\big(KL(q,p)+KL(p,q)\big)\right).$$

- The algorithm is asymptotically accurate under the necessary conditions for accurate detection in the case of adaptive sampling
- The necessary conditions for accurate detection are tight!

Random Sampling

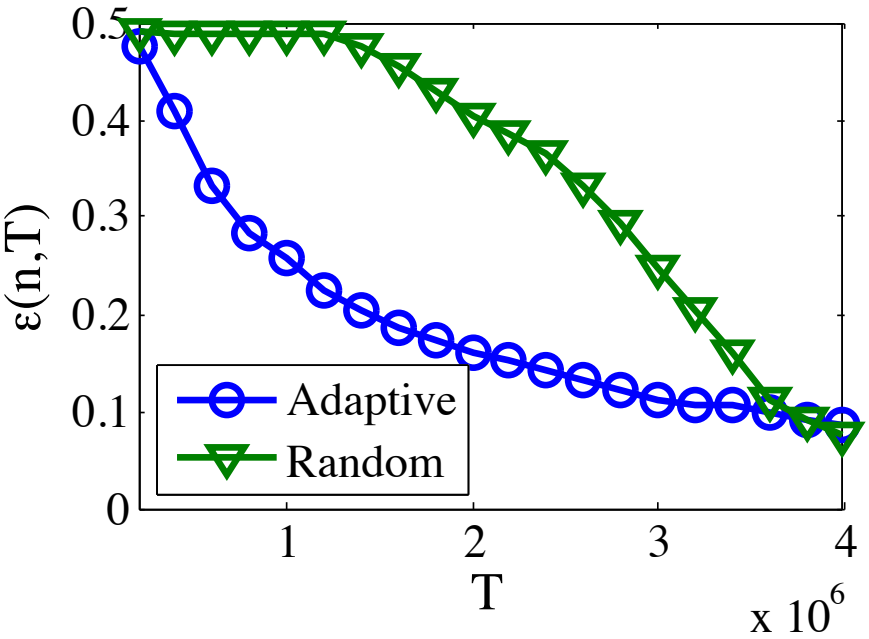


Adaptive Sampling

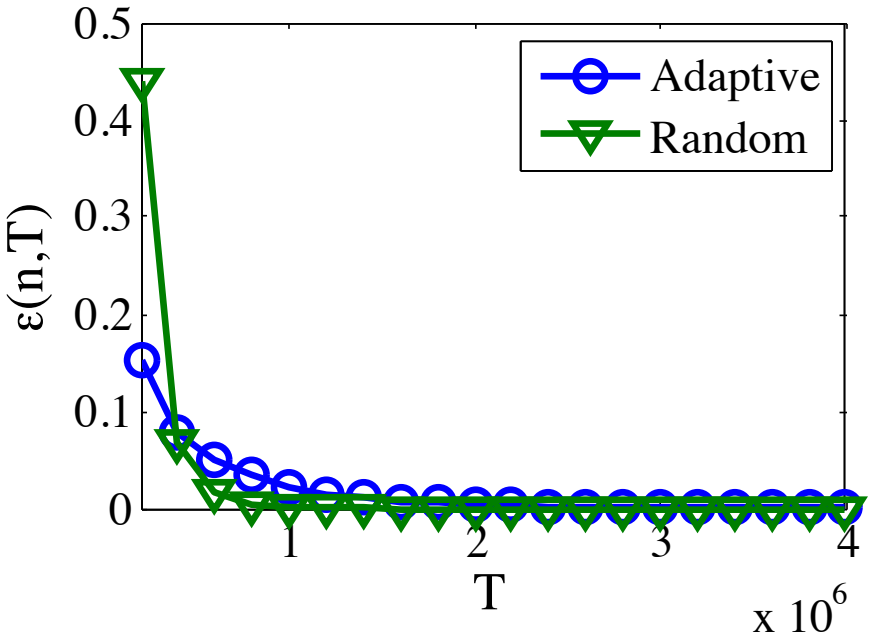


SBM: A Numerical Example

- n = 4000



p = 0.01, q=0.005

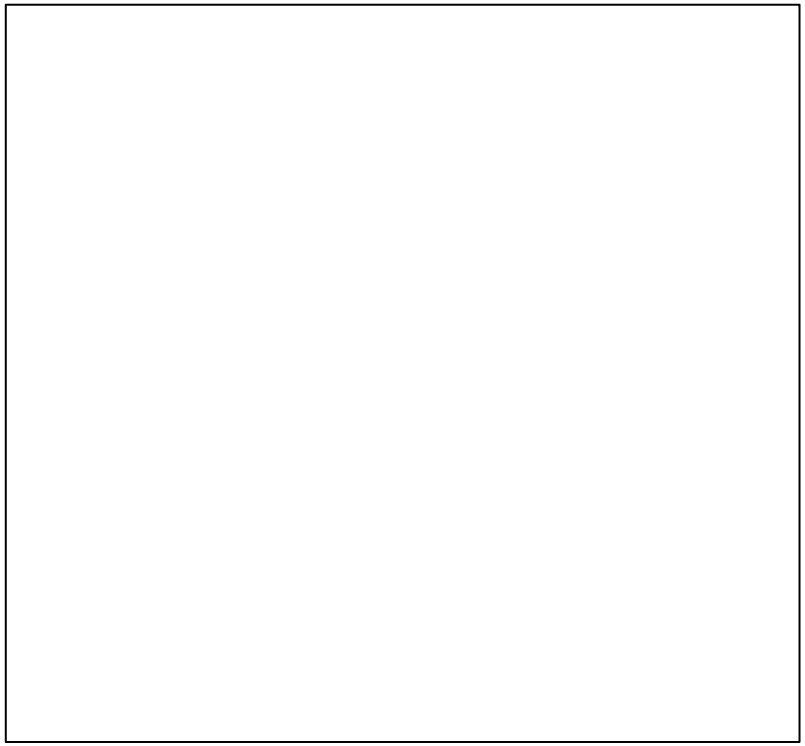


p = 0.1, q=0.05

Detection with limited memory

- Storing the adjacency matrix in RAM could be impossible

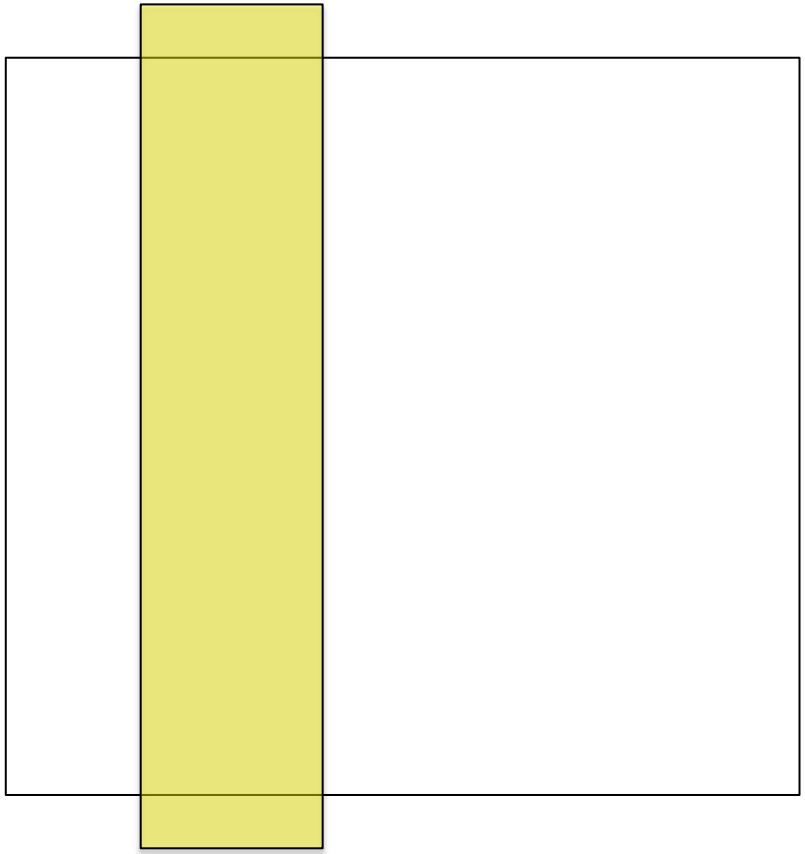
$A =$



Detection with limited memory

- Storing the adjacency matrix in RAM could be impossible

$A =$



Classify nodes in an online streaming way

The spectral method is not optimal here …

A memory scaling linearly with *n* is enough!

Summary

- A generic sampling framework extending the SBM
- Necessary conditions for asymptotically accurate detection valid in *all* regimes (unknown so far for the SMB)
- Asymptotically optimal joint sampling and clustering algorithms
- Arbitrary sample budget:
  - Quantify the impact of lack of information (some pair of nodes not observed)
  - Required budget for detection in very sparse regimes (circumventing the phase transition problem)
- Extensions
  - Beyond the SBM: different *p*'s in different clusters; Overlapping communities; …
  - Limited memory, online classification: No loss of performance …

Thanks!

seyoung.yun@inria.fr

alepro@kth.se