A Nearly-Linear Time Algorithm for Exact Community Recovery in Stochastic Block Model

Peng Wang[1], Zirui Zhou[2], Anthony Man-Cho So[1]

[1]Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong

[2]Department of Mathematics, Hong Kong Baptist University

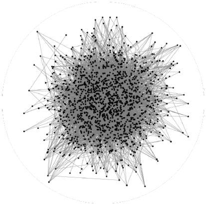
June 14, 2020

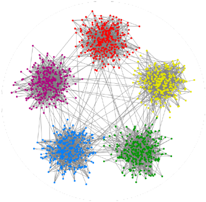
Table of Contents

Table of Contents

Community Detection

- Community detection refers to the problem of inferring similarity classes of vertices (i.e., communities) in a network by observing their local interactions (Abbe 2017); see the below graphs.
- Broad applications in machine learning, biology, social science and many areas.
- **Exact recovery** requires to identify the entire partition correctly.





Overview

- **Problem**: exactly recover the communities in the binary symmetric stochastic block model (SBM), where n vertices are partitioned into two equal-sized communities and the vertices are connected with probability $p = \alpha \log(n)/n$ within communities and $q = \beta \log(n)/n$ across communities.
- **Goal**: propose an efficient algorithm that achieves exact recovery at the information-theoretic limit, i.e., $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$.
- **Proposed Method**: a two-stage iterative algorithm:

  (i) 1st-stage: power method, coarse estimate,

  (ii) 2nd-stage: generalized power method, refinement.
- **Theoretic Results**: the proposed method can achieve exact recovery at the information-theoretic limit within $\tilde{O}(n)$ time complexity.

Table of Contents

Stochastic Block Model

Given n nodes in two equal-sized clusters, we denote by $\boldsymbol{x}^*$ its true community structures, e.g., for every $i \in [n]$, $x_i^* = 1$ if the node i belongs to the first cluster and $x_i^* = -1$ if it belongs to the second one.

Model 1 (Binary symmetric SBM)

The elements $\{a_{ij} : 1 \leq i \leq j \leq n\}$ of $\boldsymbol{A}$ are generated independently by

$$a_{ij} \sim \begin{cases} \textbf{Bern}(p), & \text{if } x_i^* x_j^* = 1, \\ \textbf{Bern}(q), & \text{if } x_i^* x_j^* = -1, \end{cases}$$

where

$$p = \frac{\alpha \log n}{n} \quad \text{and} \quad q = \frac{\beta \log n}{n}$$

for some constants $\alpha > \beta > 0$. Besides, we have $a_{ij} = a_{ji}$ for all $1 \leq j < i \leq n$.

The problem of achieving exact recovery is to develop efficient methods that can find $\boldsymbol{x}^*$ or $-\boldsymbol{x}^*$ with high probability given the adjacency matrix $\boldsymbol{A}$.

Phase Transition

The maximum likelihood (ML) estimator of $\boldsymbol{x}^*$ in the binary symmetric SBM is the solution of the following problem:

$$\max\left\{\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}:\ \mathbf{1}_n^T\boldsymbol{x}=0,\ x_i=\pm 1,\ i=1,\ldots,n\right\}. \tag{1}$$

Theorem 1 (Abbe et al. (2016), Mossel et al. (2014))

In the binary symmetric SBM, exact recovery is impossible if $\sqrt{\alpha}-\sqrt{\beta}<\sqrt{2}$, while it is possible and can be achieved by the ML estimator if $\sqrt{\alpha}-\sqrt{\beta}>\sqrt{2}$.

In literature, $\sqrt{\alpha}-\sqrt{\beta}>\sqrt{2}$ is called the information-theoretic limit.

Question: Is it possible to develop efficient methods for achieving exact recovery at the information-theoretic limit?

Related Works

Table: Methods above the information-theoretic limit

Authors	Methods	Time complexity	Recovery bounds
Boppana, 1987	spectral algo.	polynomial time	$(\alpha-\beta)^2/(\alpha+\beta) > 72$
McSherry, 2001	spectral algo.	polynomial time	$(\alpha-\beta)^2/(\alpha+\beta) > 64$
Abbe et al., 2016	SDP	polynomial time	$3(\alpha-\beta)^2 > 24(\alpha+\beta)+ 8(\alpha-\beta)$
Bandeira et al., 2016	manifold opti.	polynomial time	$(p-q)/\sqrt{p+q} \geq cn^{-1/6}$

Table: Methods at the information-theoretic limit

Authors	Methods	Time complexity	Recovery bounds
Hajek et al., 2016	SDP	polynomial time	$\sqrt{\alpha}-\sqrt{\beta} > \sqrt{2}$
Abbe et al., 2017	spectral algo.	polynomial time	$\sqrt{\alpha}-\sqrt{\beta} > \sqrt{2}$
Gao et al., 2017	two-stage algo.	polynomial time	$\sqrt{\alpha}-\sqrt{\beta} > \sqrt{2}$
Our paper	two-stage algo.	nearly-linear time	$\sqrt{\alpha}-\sqrt{\beta} > \sqrt{2}$

Table of Contents

Algorithm

Algorithm 1 A Two-Stage Algorithm for Exact Recovery

Input: adjacency matrix $\boldsymbol{A}$, positive integer N
set $\rho \leftarrow \mathbf{1}_n^T \boldsymbol{A} \mathbf{1}_n / n^2$ and $\boldsymbol{B} \leftarrow \boldsymbol{A} - \rho \boldsymbol{E}_n$
choose $\boldsymbol{y}^0$ randomly with uniform distribution over the unit sphere
for $k = 1, 2, \ldots, N$ **do**
  set $\boldsymbol{y}^k \leftarrow \boldsymbol{B}\boldsymbol{y}^{k-1} / \|\boldsymbol{B}\boldsymbol{y}^{k-1}\|_2$
end for

power method (PM): coarse estimate

set $\boldsymbol{x}^0 \leftarrow \sqrt{n}\boldsymbol{y}^N$
for $k = 1, 2, \ldots$ **do**
  set $\boldsymbol{x}^k \leftarrow \boldsymbol{B}\boldsymbol{x}^{k-1} / |\boldsymbol{B}\boldsymbol{x}^{k-1}|$
  if $\boldsymbol{x}^k = \boldsymbol{x}^{k-1}$ **then**
    terminate and return $\boldsymbol{x}^k$
  end if

stopping criteria

end for

generalized power method (GPM): refinement

For any $\boldsymbol{v} \in \mathbb{R}^n$, $\boldsymbol{v}/|\boldsymbol{v}|$ denotes the vector of $\mathbb{R}^n$ defined as

$$\left(\frac{\boldsymbol{v}}{|\boldsymbol{v}|}\right)_i = \begin{cases} 1, & \text{if } v_i \geq 0, \\ -1, & \text{otherwise,} \end{cases} \quad i = 1, \ldots, n.$$

Main Theorem

Theorem 2 (Iteration Complexity for Exact Recovery)

Let $\boldsymbol{A}$ be randomly generated by Model 1. If $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$, then the following statement holds with probability at least $1 - n^{-\Omega(1)}$: Algorithm 1 finds $\boldsymbol{x}^*$ or $-\boldsymbol{x}^*$ in $O(\log n / \log\log n)$ power iterations and $O(\log n / \log\log n)$ generalized power iterations.

Consequences:

- Algorithm 1 achieves exact recovery at the information-theoretic limit.
- Explicit iteration complexity bound for Algorithm 1 to achieve exact recovery.

The number of non-zero entries in $\boldsymbol{A}$ is, with high probability, in the order of $n \log n$.

Corollary 3 (Time Complexity for Exact Recovery)

Let $\boldsymbol{A}$ be randomly generated by Model 1. If $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$, then with probability at least $1 - n^{-\Omega(1)}$, Algorithm 1 finds $\boldsymbol{x}^*$ or $-\boldsymbol{x}^*$ in $O(n \log^2 n)$ time complexity.

Analysis of Power Method

Proposition 1 (Convergence Rate of Power Method)

Let $\{\boldsymbol{y}^k\}_{k\geq 0}$ be the sequence generated in the first-stage of Algorithm 1. Then, it holds with probability at least $1 - n^{-\Omega(1)}$ that

$$\min_{s\in\{\pm 1\}} \|\boldsymbol{y}^k - s\boldsymbol{u}_1\|_2 \lesssim n/(\log n)^{k/2}, \ \forall\ k \geq 0, \tag{2}$$

where $\boldsymbol{u}_1$ is an eigenvector of $\boldsymbol{B}$ associated with the largest eigenvalue.

- $\{\boldsymbol{y}^k\}_{k\geq 0}$ with high probability converges at least linearly to $\boldsymbol{u}_1$.
- Equation (2) shows that the ratio in the linear rate of convergence tends to 0 as $n \to \infty$.

Lemma 4 (Distance from Leading Eigenvalue of B to Ground Truth)

It holds with probability at least $1 - n^{-\Omega(1)}$ that

$$\min_{s\in\{\pm 1\}} \left\|\sqrt{n}\boldsymbol{u}_1 - s\boldsymbol{x}^*\right\|_2 \lesssim \sqrt{n/\log n}. \tag{3}$$

- It suffices to compute $\boldsymbol{y}^{N_p}$ such that $\min_{s\in\{\pm 1\}} \|\boldsymbol{y}^{N_p} - s\boldsymbol{u}_1\|_2 \lesssim 1/\sqrt{\log n}$. By (2), we have $N_p = O(\log n/\log\log n)$.

Analysis of Generalized Power Method

Proposition 2 (Convergence Rate of Generalized Power Method)

Let $\alpha > \beta > 0$ be fixed such that $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$. Suppose that the $\mathbf{x}^0$ in Algorithm 1 satisfies $\|\mathbf{x}^0\|_2 = \sqrt{n}$ and $\|\mathbf{x}^0 - \mathbf{x}^*\|_2 \lesssim \sqrt{n/\log n}$. Then, it holds with probability at least $1 - n^{-\Omega(1)}$ that

$$\|\mathbf{x}^k - \mathbf{x}^*\|_2 \leq \|\mathbf{x}^0 - \mathbf{x}^*\|_2/(\log n)^{k/2}. \tag{4}$$

- Note that $\|\mathbf{x}^0 - \mathbf{x}^*\|_2 \leq \|\mathbf{x}^0 - \sqrt{n}\mathbf{u}_1\|_2 + \|\sqrt{n}\mathbf{u}_1 - \mathbf{x}^*\|_2 \lesssim \sqrt{n/\log n}$.

Lemma 5 (One-step Convergence of Generalized Power Iterations)

For any fixed $\alpha > \beta > 0$ such that $\sqrt{\alpha} - \sqrt{\beta} > \sqrt{2}$, the following event happens with probability at least $1 - n^{-\Omega(1)}$: for all $\mathbf{x} \in \{\pm 1\}^n$ such that $\|\mathbf{x} - \mathbf{x}^*\|_2 \leq 2$, it holds that

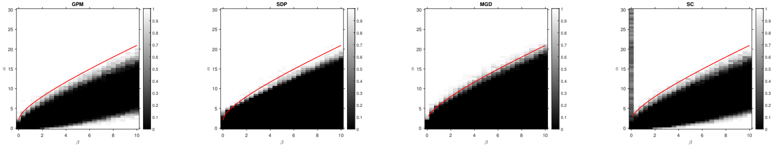
$$\mathbf{B}\mathbf{x}/|\mathbf{B}\mathbf{x}| = \mathbf{x}^*. \tag{5}$$

- This lemma indicates that the GPM exhibits finite termination.
- If $\|\mathbf{x}^0 - \mathbf{x}^*\|_2/(\log n)^{N_g/2} \leq 2$, by (4), we have $\|\mathbf{x}^{N_g} - \mathbf{x}^*\|_2 \leq 2$. Then, $\mathbf{x}^{N_g+1} = \mathbf{x}^*$. One can verify $N_g = O(\log n/\log\log n)$.

Table of Contents

Phase Transition and Computation Efficiency

- Benchmark methods:
  - SDP-based approach in Amini et al. (2018) solved by ADMM.
  - Manifold optimization (MFO) based approach in Bandeira et al. (2016) solved by manifold gradient descent (MGD) method.
  - Spectral clustering approach in Abbe et al. (2017) solved by Matlab function *eigs*.
- Parameters setting:
  - $n = 300$; α and β vary from 0 to 30 and 0 to 10, with increments 0.5 and 0.4, respectively.
  - For fixed (α, β), we generate 40 instances and calculate the ratio of exact recovery.



running time: 25 s | running time: 9313 s | running time: 1064 s | running time: 118 s

Figure: Phase transition: the x-axis is β, the y-axis is α, and darker pixels represent lower empirical probability of success. The red curve is $\sqrt{\alpha} - \sqrt{\beta} = \sqrt{2}$.

Convergence Performance

- Parameters setting:
  - $\alpha = 10,\ \beta = 2.$
  - $n = 1000,\ 5000,\ 10000.$

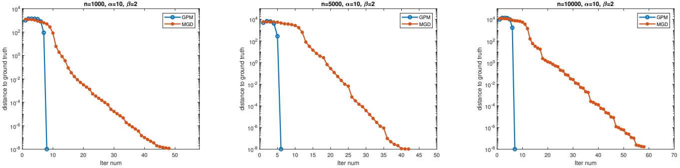


Figure: Convergence performance: the x-axis is number of iterations, the y-axis for GPM is $\|\boldsymbol{x}^k {\boldsymbol{x}^k}^T - \boldsymbol{x}^* {\boldsymbol{x}^*}^T\|_F$, and the y-axis for MGD is $\|\boldsymbol{Q}^k {\boldsymbol{Q}^k}^T - \boldsymbol{x}^* {\boldsymbol{x}^*}^T\|_F$, where $\boldsymbol{x}^k$ and $\boldsymbol{Q}^k$ are the iterates generated in the k-th iteration of GPM and MGD, respectively.

Table of Contents

Conclusions

1. We propose a two-stage iterative algorithm to solve the problem of exact community recovery in the binary symmetric SBM:

   (i) 1st-stage: power method,

   (ii) 2nd-stage: generalized power method.

2. We show that the proposed method can achieve exact recovery at the information-theoretic limit within $\tilde{O}(n)$ time complexity.

3. Numerical experiments demonstrate that the proposed approach has strong recovery performance and is highly efficient.