2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)
Allerton Park and Retreat Center
Monticello, IL, USA, October 2-5, 2018

On the Joint Recovery of Community Structure and Community Features

Jisang Yoon
R&D Center
Kakaocorp
Sungnam, Korea
jason.yoon@kakaocorp.com

Kangwook Lee
School of EE
KAIST
Daejeon, Korea
kw1jjang@kaist.ac.kr

Changho Suh
School of EE
KAIST
Daejeon, Korea
chsuh@kaist.ac.kr

***Abstract*—We study the problem of recovering both K communities and their features from a labeled graph observation. We assume that the edges of an observed graph are generated as per the symmetric Stochastic Block Model (SBM), and that the label of each node is a noisy and partially-observed version of the corresponding community feature. We characterize the information-theoretic limit of this problem, and then propose a computationally efficient algorithm that achieves the information-theoretic limit.**

***Index Terms*—Community recovery, Stochastic Block Model, Matrix completion, Recommendation algorithms, Signal processing**

I. Introduction

Community recovery has been extensively studied in various fields, including mathematics, computer science, machine learning, and biology [1]. Most of the works focus on the problem of recovering hidden communities when the graph structure (nodes and edges) and additional side-information (node and edge values) is given. However, in many practical applications, one may want to recover not only hidden communities but also *their features* [2]. Motivated by this, we study the problem of recovering both hidden communities and their features simultaneously.

As a concrete application of this problem, consider social recommendation systems [3]. In a social recommendation system, one is given with 1) rating vector of each user and 2) a social graph of users. Since it is known that the users of the same community share similar preferences, one may assume that each community has its own representative rating vector (i.e. community feature) and that the rating vectors of its members are noisy observations of the community feature. For this case, one may want to jointly recover the hidden communities and their features, with which one can provide more reliable recommendations to users.

We now briefly describe the problem. We first assume that n nodes are partitioned into K hidden communities, and each hidden community is associated with a community feature. In this work, we assume that community features are m-dimensional binary vectors. That is, each cluster, say cluster k, is associated with a m-dimensional binary feature vector $u_k \in \{-1, 1\}^m$. The goal is to recover both the hidden partition as well as the feature vector of each community from the observed graph.

The structure of the observed graph G is assumed to follow the *stochastic block model (SBM)* [4]. SBM has been proved to fit a variety of real datasets [5]. In the SBM, edges are generated independently of the others, and the edge probability between community i and j is constant, which we denote by $Q_{i,j}$. Denoting the community assignment by $C : [n] \to [K]$, the edge probability between node i and j can be written as $Q_{C(i),C(j)}$. Especially, in this paper, we only consider the *symmetric* SBM, i.e.,

$$Q_{i,j} = \begin{cases} \alpha, & \text{if } i = j, \\ \beta, & \text{else.} \end{cases} \tag{1}$$

In addition to the graph structure, we assume that node features, which are correlated with the corresponding community feature, are available. Specifically, the observed feature of node i, denoted by $N^{\Omega}(i, :)$, is a noisy and partially-observed version of $u_{C(i)}$, i.e.,

$$N^{\Omega}(i,j) := \begin{cases} u_{C(i)}(j) & \text{w.p. } p \cdot (1-\theta) \\ -u_{C(i)}(j) & \text{w.p. } p \cdot \theta \\ 0 & \text{w.p. } (1-p). \end{cases} \tag{2}$$

Here, each case represents correctly observed, incorrectly observed, and unobserved cases in order.

The goal is to *exactly* recover both the hidden community structure and the community features. That is, one wants to recover C (up to permutation) and u's for K communities.

In this work, we characterize the information-theoretic limit of exact recovery, and then develop a computationally efficient algorithm, and show that it achieves the aforementioned limit.

The rest of this paper is organized as follows. In the rest of this section, we briefly overview related works and define useful notations. In Sec. II, we present the problem formulation. Our main theoretical results are presented in Sec. III and Sec. IV. Finally, we give experimental results with synthetic data in Sec. V and conclude the paper in Sec. VI.

A. Related works

Community recovery has been extensively studied in the literature, and we refer the readers to a recent survey of Abbe [6]. Some recent works have taken into account side-information in addition to the graph structure. For instance, edge weights [7]

and node values [2], [8] are shown to help when finding hidden communities. However, these problems still aim at recovering hidden communities only, not community features.

In a recent work [9], the authors firstly study the joint recovery of community structure and community features. While they focus on the case of 2 equal-sized communities, we consider a more general case of K communities of possibly different sizes. We remark that these extensions are non-trivial: It requires us to concretely define target space in sequence level and its topology. For instance, even for extension to K equal-sized communities, the original framework cannot be applied since one cannot regulate $\|u_i^X - u_j^X\|_0$ using the technique of [9]. Moreover, we have constructed a non-parametric algorithm, which is guaranteed to succeed exact recovery *a.s.*, even without knowing ground truth parameters $(\alpha, \beta, p, \theta)$.

B. Notations

The following notations will be used throughout the paper.

- For any $t \in \mathbb{N}$, $[t] := \{0, 1, 2, \ldots, t-1\}$.
- Target information, which we want to recover, is denoted by $X := (C^X, u^X)$; It is a pair of communities (C^X) and their binary feature vectors (u^X).
- Community assignment function is denoted as $C^X : [n] \to [K]$, and $C_k^X := (C^X)^{-1}(\{k\})$ (inverse image).
- $u_k^X \in \{-1, 1\}^m$ is a binary feature vector of the k^{th} community.
- G is an observed graph, drawn from symmetric SBM(α, β).
- $F^X \in \{-1, 1\}^{n \times m}$ is a ground truth binary feature matrix, i.e., $F^X(i, j) := u^X_{C^X(i)}(j)$.
- For given n, m, $\mathcal{O}_{n,m}$ is the set of all possible observation with n nodes and m features.
- For given n, m, $\mathcal{X}_{n,m}$ is the set of all possible targets with n nodes and m features.
- Bold symbols represent sequences, e.g., $\boldsymbol{x} := (x_i)_{i \in \mathbb{N}}$
- Parameters are denoted by $\Gamma := (\alpha, \beta, p, \theta, n, m)$.

II. Problem Formulation

We now formally define the featured community recovery problem. We first define valid parameter sequences. Here, asymptotic notation is w.r.t. i, e.g., $O \Rightarrow O_i$ and $w \Rightarrow w_i$.

Definition 1 (valid parameter (non-target))**.** A parameter sequence $\boldsymbol{\Gamma} := (\Gamma_i)_{i \in \mathbb{N}} := (\alpha_i, \beta_i, p_i, \theta_i, n_i, m_i)_{i \in \mathbb{N}}$ is *valid* if

$$
\begin{aligned}
&n_i, m_i = w(1),\\
&\log(n_i) = O(m_i),\ \ \log(m_i) = O(n_i),\\
&m_i \cdot \log(m_i) = O(n_i \cdot \log(n_i)),\\
&\alpha_i = w\left(\frac{1}{n}\right),\ \alpha_i = O\left(\frac{\log(n_i)}{n_i}\right),\ \alpha_i > \beta_i,\ \beta_i = \Omega(\alpha_i),\\
&p_i = \Theta\left(\frac{\log(n_i)}{m_i} + \frac{\log(m_i)}{n_i}\right),\ \theta_i = \Omega(1).
\end{aligned}
$$

Note that this definition implies $\alpha, \beta, p = o(1)$.

We now define the pseudo-metric topology of target space.

Definition 2 (pseudo-metric topology of target space)**.** For two targets $X = (C^X, u^X)$, $Y = (C^Y, u^Y)$ and $\forall k \in [K]$, Let $\mathcal{P}$ be the set of all permutations of $[K]$, and Φ_X^E be the set of all possible sequence $\boldsymbol{X} := (X_i)_{i \in \mathbb{N}}$ For $\boldsymbol{X}, \boldsymbol{Y} \in \Phi_X^E$, distance function $\boldsymbol{d}$ can be defined as following:

$$d_C^P(X, Y) := \max_{k \in [K]} |C_k^X \setminus C_{P(k)}^Y|, \forall P \in \mathcal{P}, \tag{3}$$

$$d_u^P(X, Y) := \max_{k \in [K]} \|u_k^X - u_{P(k)}^Y\|_0, \forall P \in \mathcal{P}, \tag{4}$$

$$d(X, Y) := \min_{P \in \mathcal{P}} \max\left(\frac{d_C^P(X, Y)}{n}, \frac{d_u^P(X, Y)}{m}\right), \tag{5}$$

$$\boldsymbol{d}(\boldsymbol{X}, \boldsymbol{Y}) := \limsup_{i \to \infty} d(X_i, Y_i), \tag{6}$$

where $\|\cdot\|_0$ denotes the hamming distance. One can trivially check that $\boldsymbol{d}$ satisfies non-negativity, symmetry, subadditivity, and $\boldsymbol{d}(\boldsymbol{X}, \boldsymbol{X}) = 0$, but note that it does not satisfy identity of indiscernibles, which means $\boldsymbol{d}$ is *pseudo-metric*. However, it can still define topology.

From now, all statements concerning topology are with respect to the topology defined in Def. 2. Using this pseudo-metric, we now can define a ball: $\boldsymbol{B}_\delta(\boldsymbol{X}) := \{\boldsymbol{Y} \in \Phi_X^E : \boldsymbol{d}(\boldsymbol{X}, \boldsymbol{Y}) < \delta\}$. Similarly, one can also define $B_\delta(X)$.

We now define valid target sequences.

Definition 3 (valid target)**.** Φ_X^E be the set of all possible sequences $\boldsymbol{X} := (X_i)_{i \in \mathbb{N}}$. Denote $c_k^X := |C_k^X|, d_{i,j}^X := \|u_i^X - u_j^X\|_0$, then $\boldsymbol{X} \in \Phi_X^E$ is valid, if it satisfies:

$$c_k^X = \Omega(n),\ \forall k \in [K],\ \ d_{j,k}^X = \Omega(m),\ \forall j \neq k \in [K]. \tag{7}$$

Hence, we can assume that

$$c_k^X \geq c^{\boldsymbol{X}} \cdot n, \forall k \in [K], \tag{8}$$

$$d_{j,k}^X \geq d^{\boldsymbol{X}} \cdot m, \forall j \neq k \in [K] \tag{9}$$

hold for all sufficiently large i. Further, we can define the set of valid targets $\Phi_X := \{\boldsymbol{X} \in \Phi_X^E : \boldsymbol{X} \text{ is valid}\}$, and show that it is open: See [10] for the proof. We will assume that the constraint space is an open sub-space of Φ_X.

We now define the error event.

Definition 4 (error events)**.** Consider fixed, and valid non-target parameter $\boldsymbol{\Gamma}$. Sequence of functions $\boldsymbol{\psi}$, where $\psi_i : \mathcal{O}_{n_i, m_i} \to \mathcal{X}_{n_i, m_i}$ is called *estimator*. Ψ is the set of all possible estimators. The *worst-case probability of error under open constraint space* $\Phi'_X \subset \Phi_X$, $\boldsymbol{P}_{\boldsymbol{\Gamma}}^e(\cdot|\Phi'_X) : \Psi \to [0, 1]$ is defined as following:

$$P_\Gamma^e(\psi|X) := \Pr(\psi(G, N^\Omega) \neq X; (G, N^\Omega) \sim \mathcal{D}(\cdot|\Gamma, X)),$$

$$\boldsymbol{P}_{\boldsymbol{\Gamma}}^e(\boldsymbol{\psi}|\Phi'_X) := \sup_{\boldsymbol{X} \in \Phi'_X} \limsup_{i \to \infty} P_{\Gamma_i}^e(\psi_i|X_i),$$

where $\mathcal{D}$ is the distribution of observed variable (G, N^Ω) given Γ and X. Note that the distribution of observed variables are fully specified by Γ (edge probabilities, noise parameters) and X (hidden communities, binary feature vectors).

With this definition, one can view the exact recovery problem under constraint Φ'_X given $\boldsymbol{\Gamma}$ as the problem of finding an estimator $\boldsymbol{\psi}$ such that $\boldsymbol{P}_{\boldsymbol{\Gamma}}^e(\boldsymbol{\psi}|\Phi'_X) = 0$.

Definition 5. Open constraint space $\Phi'_X \subset \Phi_X$ is $\mathbf{\Gamma}$-*solvable* if there exists estimator $\boldsymbol{\psi}$ such that $\boldsymbol{P}^e_{\mathbf{\Gamma}}(\boldsymbol{\psi}|\Phi'_X) = 0$.

Finally, our goal is to find the sharp condition of constraint Φ'_X that is solvable, and construct the estimator (in other word, algorithm) to solve the constrained problem when it is solvable.

III. Achievability and Information Limit

In this section, we characterize the sharp phase transition of the exact recovery. Here, we outline the proof, deferring the full proof to the supplemental material [10]. We first define the followings before stating the phase transition.

Definition 6. For given fixed, valid non-target parameter Γ, we define the following:

$$I^{\Gamma}_r := p \cdot (\sqrt{1-\theta} - \sqrt{\theta})^2, \tag{10}$$

$$I^{\Gamma}_s := (\sqrt{\alpha} - \sqrt{\beta})^2. \tag{11}$$

Further, function $\mathbf{\Lambda_\Gamma}(\cdot) : \Phi_X \to \mathbb{R}$ is defined as follows:

$$\Lambda^{i,j}_{\Gamma}(X) := \frac{c^X_{i,j} \cdot I^{\Gamma}_s + d^X_{i,j} \cdot I^{\Gamma}_r}{\log(n)} \ \forall i \neq j \in [K], \tag{12}$$

$$\Lambda^{k}_{\Gamma}(X) := \frac{c^X_k \cdot I^{\Gamma}_r}{\log(m)}, \ \forall k \in [K], \tag{13}$$

$$\Lambda_{\Gamma}(X) := \min(\{\Lambda^{i,j}_{\Gamma}(X) : i \neq j \in [K]\} \tag{14}$$

$$\cup \{\Lambda^{k}_{\Gamma}(X) : k \in [K]\}), \tag{15}$$

$$\mathbf{\Lambda_\Gamma}(\boldsymbol{X}) := \liminf_{i\to\infty} \Lambda_{\Gamma_i}(X_i), \tag{16}$$

where $c^X_{i,j} := \frac{c^X_i + c^X_j}{2}$. Using the above definition, one can define positive, and negative spaces given valid $\mathbf{\Gamma}$ as following:

$$\Phi^{\mathbf{\Gamma},+}_X := \{\boldsymbol{X} \in \Phi_X : \mathbf{\Lambda_\Gamma}(X) > 1\} \tag{17}$$

$$\Phi^{\mathbf{\Gamma},-}_X := \{\boldsymbol{X} \in \Phi_X : \mathbf{\Lambda_\Gamma}(X) < 1\} \tag{18}$$

Remark 1. Note that by validity, one can check that $\frac{I^{\Gamma}_s \cdot n}{I^{\Gamma}_r \cdot m}$ is bounded, this fact will be used to show Lemma 1.

One can show that

Proposition 1. $\mathbf{\Lambda_\Gamma} : \Phi_X \to \mathbb{R}$ *is a continuous map.*

Proof. We deferred the proof to [10] □

Corollary 1. $\Phi^{\mathbf{\Gamma},+}_X, \Phi^{\mathbf{\Gamma},-}_X$ *are both open subspace.*

We now are ready to state our main theorem, which sharply characterizes the phase transition threshold.

Theorem 1. *For given valid* $\mathbf{\Gamma}$, $\Phi'_X \subset \Phi_X$ *is* $\mathbf{\Gamma}$*-solvable if and only if* $\Phi'_X \cap \Phi^{\mathbf{\Gamma},-}_X = \emptyset$

Proof: We first outline the proof of the achievability part and then overview that of the converse part.

Achievability (⇐): The following lemma holds due to the the property of topology [10].

Lemma 1. *if* $\Phi'_X \subset \Phi_X$*: open, and* $\Phi'_X \cap \Phi^{\mathbf{\Gamma},-}_X = \emptyset$*, then* $\Phi'_X \subset \Phi^{\mathbf{\Gamma},+}_X$

Note that the above lemma holds not because $\mathbf{\Lambda_\Gamma}$ is continuous. Intuitively, if the constraint space *touches* the boundary ($\{\mathbf{\Lambda_\Gamma} = 1\}$), then it would have *crossed* the boundary.

It is enough to show that $\forall \boldsymbol{X} \in \Phi'_X \subset \Phi^{\mathbf{\Gamma},+}_X$,

$$\limsup_{i\to\infty} P^e_{\Gamma_i}(\psi^{ML}_i|X_i) = 0 \tag{19}$$

where ψ^{ML}_i is a maximum likelihood estimator. By definition of $\Phi^{\mathbf{\Gamma},+}_X$, $\exists \epsilon > 0$, for every sufficiently large i,

$$\Lambda_{\Gamma_i}(X_i) > 1 + \epsilon \tag{20}$$

Maximizing likelihood is equivalent to minimizing negative log-likelihood function, which we denoted by $L(\cdot)$.

$$\begin{aligned} P^e_{\Gamma}(\psi^{ML}|X) &:= \Pr(\psi^{ML}(N^{\Omega}, G) \neq X) \\ &= \Pr(\exists Y \neq X : L(Y) \leq L(X); N^{\Omega}, G) \\ &= \Pr(\bigcup_{Y\in\mathcal{Y}} L(Y) \leq L(X); N^{\Omega}, G) \\ &\leq \sum_{Y\in\mathcal{Y}} \Pr(L(Y) \leq L(X); N^{\Omega}, G) \end{aligned} \tag{21}$$

where $\mathcal{Y} := \mathcal{X}_{n_i,m_i} \setminus \{X\}$.

It is enough to show that RHS of Eq. (21) goes to zero as $i \to \infty$. Now fix i, thus omit it from subscript. We will split $\mathcal{Y}$ into 3 sets, and show that summation over each set goes to zero. Define

$$C^Y_{i,j} := C^X_i \cap C^Y_j \tag{22}$$

then it is K^2-partition of the entire nodes. Next, define

$$D(Y) := \|F^X - F^Y\|_0 \tag{23}$$

and it can be represented as

$$D(Y) = \sum_{i,j} |C^Y_{i,j}| \cdot \|u^X_i - u^Y_j\|_0 \tag{24}$$

With these tools, we can partition $\mathcal{Y}$ as following (δ is some positive constant that we will take for the proof):

$$\begin{aligned} \mathcal{Y}_1 := &\{Y \in \mathcal{Y} | \exists i, j \neq k \in [K] : |C^Y_{i,j}|, |C^Y_{i,k}| \geq \delta \cdot n\} \\ &\cup \{Y \in \mathcal{Y} | \exists i, j \neq k \in [K] : |C^Y_{j,i}|, |C^Y_{j,k}| \geq \delta \cdot n\} \end{aligned} \tag{25}$$

$$\mathcal{Y}_2 := \{Y \in \mathcal{Y} \setminus \mathcal{Y}_1 | \exists i \in [K] : \|u^X_i - u^Y_i\|_0 \geq \delta \cdot n\} \tag{26}$$

$$\mathcal{Y}_3 := \mathcal{Y} \setminus \mathcal{Y}_1 \setminus \mathcal{Y}_2 \tag{27}$$

The following lemma is useful for showing the claim behind.

Lemma 2. *For* $N_1, N_2, L \in \mathbb{Z}^+$*, let* $\{A_i\}_{i\in[N_1]} \overset{i.i.d}{\sim}$ *Bern*(α), $\{B_i\}_{i\in[N_2]} \overset{i.i.d}{\sim}$ *Bern*(β), $\{P_i\}_{i\in[L]} \overset{i.i.d}{\sim}$ *Bern*(p)*, and* $\{\Theta\}_{i\in[L]} \overset{i.i.d}{\sim}$ *Bern*(θ)*. Assume that* $\alpha, \beta, p = o(1)$*. Then, for any* $l > 0$*,*

$$\begin{aligned} &\Pr(S_1 + S_2 + S_3 \geq -l) \\ &\leq \exp\left(\frac{1}{2}l - (1+o(1))\frac{N_1+N_2}{2} I^{\Gamma}_s - (1+o(1)) \cdot L \cdot I^{\Gamma}_r\right) \end{aligned}$$

where

$$S_1 := c_1^{\Gamma} \cdot (N_2 - N_1),\ S_2 := c_2^{\Gamma} \left(\sum_{i \in [N_2]} B_i - \sum_{i \in [N_1]} A_i \right)$$
$$S_3 := c_3^{\Gamma} \cdot \sum_{i \in [L]} P_i (2\Theta_i - 1)$$

and

$$c_1^{\Gamma} := \log\left(\frac{1-\alpha}{1-\beta}\right),\ c_2^{\Gamma} := \log\left(\frac{\alpha \cdot (1-\beta)}{\beta \cdot (1-\alpha)}\right),$$
$$c_3^{\Gamma} := \log\left(\frac{1-\theta}{\theta}\right).$$

The proof of this Chernoff-based concentration inequality is deferred to [10], [11].

Remark 2. One can view S_1 as the term balancing with each community size. This term is one of the key differences from the previous work [9], and it allows us to handle multiple communities with different sizes.

Remark 3. In company with this lemma, we can take the upper bound of probability which other target has higher likelihood than original target. You can find the full proof in the supplemental material.

Claim 1. *Following holds:*

$$\sum_{Y \in \mathcal{Y}_1} \Pr(L(Y) \le L(X); N^{\Omega}, G) = o(1) \tag{28}$$
$$\sum_{Y \in \mathcal{Y}_2} \Pr(L(Y) \le L(X); N^{\Omega}, G) = o(1) \tag{29}$$
$$\sum_{Y \in \mathcal{Y}_3} \Pr(L(Y) \le L(X); N^{\Omega}, G) = o(1) \tag{30}$$

By Claim 1, and Eq. (25), exact recovery can be achieved by $\boldsymbol{\psi}^{ML}$.

Converse ($\Rightarrow$): Suppose that $\boldsymbol{X}^0 \in \Phi'_X \cap \Phi_X^{\boldsymbol{\Gamma},-}$, and let

$$\boldsymbol{\Lambda}_{\boldsymbol{\Gamma}}(\boldsymbol{X}^0) = 1 - 3 \cdot \epsilon \tag{31}$$

Since Φ'_X is open, and $\boldsymbol{\Lambda}_{\boldsymbol{\Gamma}}$ is a continuous map (by Prop. 1), we can take δ small enough to make the following holds:

$$\boldsymbol{B}_{\delta}(\boldsymbol{X}^0) \subset \Phi'_X \tag{32}$$
$$\forall \boldsymbol{X} \in \boldsymbol{B}_{\delta}(\boldsymbol{X}^0), \boldsymbol{\Lambda}_{\boldsymbol{\Gamma}}(\boldsymbol{X}) < 1 - 2 \cdot \epsilon \tag{33}$$

Furthermore, by definition of $\boldsymbol{\Lambda}_{\boldsymbol{\Gamma}}$, there is a subsequence S such that

$$\forall \boldsymbol{X} \in \boldsymbol{B}_{\delta}(\boldsymbol{X}^0), i \in S, \Lambda_{\Gamma_i}(X_i) < 1 - \epsilon \tag{34}$$

Now, we only consider about above subsequence, and omit i from the subscript.

Define $\mathcal{B}_{\delta} := \{\boldsymbol{X} \in \Phi_X^E : X_i \in B_{\delta}(X_i^0), \forall i\}$, then $\mathcal{B}_{\delta} \subset \boldsymbol{B}_{\delta}(\boldsymbol{X}^0) \subset \Phi'_X$. For convenience, denote $B_{\delta}(X_i^0)$ as $\mathcal{B}_i$.

Suppose that $\exists \boldsymbol{\psi} \in \Psi : \boldsymbol{P}^e_{\boldsymbol{\Gamma}}(\boldsymbol{\psi}|\Phi'_X) = 0$.

$$\begin{aligned} \boldsymbol{P}^e_{\boldsymbol{\Gamma}}(\boldsymbol{\psi}|\Phi'_X) &:= \sup_{\boldsymbol{X} \in \Phi'_X} \limsup_{i \to \infty} P^e_{\Gamma_i}(\psi_i|X_i) \\ &\ge \sup_{\boldsymbol{X} \in \mathcal{B}_{\delta}} \limsup_{i \to \infty} P^e_{\Gamma_i}(\psi_i|X_i) \end{aligned} \tag{35}$$

and we can also show that

$$\sup_{\boldsymbol{X} \in \mathcal{B}_{\delta}} \limsup_{i \to \infty} P^e_{\Gamma_i}(\psi_i|X_i) \ge \limsup_{i \to \infty} \sup_{\boldsymbol{X} \in \mathcal{B}_{\delta}} P^e_{\Gamma_i}(\psi_i|X_i) \tag{36}$$

Eq. (36) can be shown by constructing $\boldsymbol{X}$ such that $X_i := \text{argmax}_{X_i \in \mathcal{B}_i} P^e_{\Gamma_i}(\psi_i|X_i)$, note that $\mathcal{B}_i$ is just a finite set. Using Eq. (35), (36),

$$\begin{aligned} \boldsymbol{P}^e_{\boldsymbol{\Gamma}}(\boldsymbol{\psi}|\Phi'_X) &\ge \limsup_{i \to \infty} \sup_{X_i \in \mathcal{B}_i} P^e_{\Gamma_i}(\psi_i|X_i) \\ &\overset{(a)}{\ge} \limsup_{i \to \infty} \mathbb{E}_{X_i \sim \text{Unif}(\mathcal{B}_i)}[P^e_{\Gamma_i}(\psi_i|X_i)] \\ &\overset{(b)}{\ge} \limsup_{i \to \infty} \mathbb{E}_{X_i \sim \text{Unif}(\mathcal{B}_i)}[P^e_{\Gamma_i}(\psi_i^{ML}|_{\mathcal{B}_i}|X_i)] \end{aligned} \tag{37}$$

where $\psi_i^{ML}|_{\mathcal{B}_i}$ is maximum a posteriori (MAP) estimator, with prior $\text{Unif}(\mathcal{B}_i)$. (a) is simply by max $\ge$ mean arguement, and (b) can be checked by the definition of MAP (Lemma 3).

Lemma 3. *$X \in \mathcal{X}$ is a finite random variable, p is the pmf of X. $Y \in \mathcal{Y}$ is also a finite random variable, whose pmf is $q(\cdot|X)$ given X. $\psi^{MAP} : \mathcal{Y} \to \mathcal{X}$ is a MAP estimator of X given Y. Then for any estimator $\psi : \mathcal{Y} \to \mathcal{X}$,*

$$\begin{aligned} &\mathbb{E}_{X \sim p}[\Pr(\psi(Y) \ne X; Y \sim q(\cdot|X))] \\ \ge &\mathbb{E}_{X \sim p}[\Pr(\psi^{MAP}(Y) \ne X; Y \sim q(\cdot|X))] \end{aligned} \tag{38}$$

Thus, it is sufficient to show that $\boldsymbol{\psi}^{ML}|_{\mathcal{B}}$ cannot exactly recover communities and their features. For convenience, we will omit the subscript i from now.

We first write:

$$\mathbb{E}_{X \in \text{Unif}(\mathcal{B})}[P^e_{\Gamma}(\psi^{ML}|_{\mathcal{B}}|X)] \ge \min_{X \in \mathcal{B}} P^e_{\Gamma}(\psi^{ML}|_{\mathcal{B}}|X) \tag{39}$$

We will now show that

$$\min_{X \in \mathcal{B}} P^e_{\Gamma}(\psi^{ML}|_{\mathcal{B}}|X) = 1 - o(1). \tag{40}$$

Note that we are actually proving a sufficient condition of the actual converse statement: When $\Phi'_X \subset \Phi_X^{\boldsymbol{\Gamma},-} \ne \emptyset$, for any estimator $\boldsymbol{\psi} \in \Psi$, $\exists \boldsymbol{X} \in \Phi'_X$ such that if $\boldsymbol{X}$ is the original target, then $\boldsymbol{\psi}$ may give wrong answer *a.s.*.

We now present a useful lemma, which can be used to lower bound the above error probability.

Lemma 4. *For $K, L \in \mathbb{Z}^+$, let $\{A_i\}_{i \in \mathbb{N}} \overset{i.i.d}{\sim}$ Bern(α), $\{B_i\}_{i \in \mathbb{N}} \overset{i.i.d}{\sim}$ Bern(β), $\{P_i\}_{i \in \mathbb{N}} \overset{i.i.d}{\sim}$ Bern(p), and $\{\Theta\}_{i \in [L]} \overset{i.i.d}{\sim}$ Bern(θ). Assume that $\alpha, \beta, p = o(1)$, and*

$$\max\left(\sqrt{\alpha \cdot \beta} \cdot K, p \cdot L\right) = w(1) \tag{41}$$

Then, for any $l > 0$

$$\begin{aligned} &\Pr(S_2 + S_3 \ge 0)) \\ \ge &\frac{1}{4} \cdot \exp(-(1 + o(1)) \cdot K \cdot I_s^{\Gamma} - (1 + o(1)) \cdot L \cdot I_r^{\Gamma}) \end{aligned} \tag{42}$$

where

$$S_2 := c_2^{\Gamma} \cdot \left(\sum_{i \in [K]} B_i - \sum_{i \in [K]} A_i \right),\ S_3 := c_3^{\Gamma} \cdot \sum_{i \in [L]} P_i (2\Theta_i - 1).$$

Note that all the conditions of the above lemma are satisfied by valid parameters.

We now outline the the proof of Eq. (40). For a complete proof, see the supplementary material [10]. If $\Lambda_\Gamma^k(X) < 1-\epsilon$ for some k, we show that there exists another valid candidate, which can be derived by flipping one entry of the feature vector, but has higher likelihood than the the correct target X a.s. Secondly, when $\Lambda_\Gamma^{i,j}(X) < 1-\epsilon$ for some $i \neq j$, one can find another candidate in which two nodes in community i and j are swapped a.s. ∎

Definition 7. We call the boundary between positive and negative space $\{\mathbf{\Lambda_\Gamma} = 1\}$ *the information-theoretic limit*.

Remark 4 (Comparison with the previous work [9]). Setting $K = 2$, $c_0^X = c_1^X = \frac{n}{2}$, and $d_{0,1}^X = \gamma \cdot m$ exactly recovers the main result of [9].

A. Intuitive explanation

Behind the formula, we now look into how each parameter (both target and non-target) makes exact recovery easier or harder. First, we can see $I_s^\Gamma := (\sqrt{\alpha} - \sqrt{\beta})^2$ may definitely decide difficulty of the problem, because as α, β are more similar, it will be harder to detect community from observed graph. Second, when p decreases, or θ increases (*i.e.* I_r^Γ decreases), gaining meaningful information from feature observation becomes harder.

Next, what is the meaning of constraint space *touching* (equivalently, *crossing*) information limit? In other words, what really happens, when the constraint space overlaps the negative space ($\Phi_X^{\Gamma,-}$)? It means that some candidate targets have too small $\mathbf{\Lambda_\Gamma}$ value. To put it more concretely, one of $\Lambda_\Gamma^k(X)$ or $\Lambda_\Gamma^{i,j}(X)$ is too small.

It may be hard to detect small community, which implies constraint should require c_k^X to be big enough to Λ_Γ^k not being too small. On the other hand, when two communities are too similar, which means their feature vectors are too similar, then we will get into trouble for differentiating communities with feature observation. Therefore, the constraint should also force $\Lambda_\Gamma^{i,j}$ to have some lower bound.

B. Practical aspects

Now, take a look at this results from the view point of practicality. The question is that among these parameters (including non-target), what we can manipulate, and what we cannot? Obviously, we may not be able to change α, β and the constraint space Φ'_X, since they are naturally given. However, p or θ are adjustable parameters in some applications.

θ is noise parameter, observer sometimes "mis"observe node feature, thus we can make this better methodologically. However, in some cases, node feature may be "naturally" different from the community feature, for instance, user preference to items will be partially distinct with its representative community preference, in that case, observer cannot improve such noise.

The other parameter we can manipulate is p, it is related to sample complexity. We may try to collect more samples of node features. For example, most of IT companies are attempting to gather more information about their users, even though they usually have very small portions of the entire features.

Restoring Def. 6, improving p, θ may expand the positive space to cover the constraint space. See Fig. 1. Furthermore, Thm. 1 precisely states how much observer should improve observing conditions.

IV. Algorithm and Performance Proof

In the previous section, we showed that under certain conditions, maximum likelihood estimator will achieve exact recovery. However, the MLE is computationally intractable, and hence in this section, we will provide a polynomial time complexity algorithm which achieves exact recovery *a.s.*

A. Comparision with the previous work

The main difference from the algorithm proposed in [9] is that we precisely found out the proper parameter to guarantee the exact recovery in non-equal community size case. As we mentioned in Remark. 2, c_1^Γ provide the optimal criteria for node to be assigned to the community with proper size.

In particular, when all communities have equal size, then even though algorithm proposed in [9] has no c_1^Γ term, it has been proved to work equivalently, since size differences between communities attained by step 1 are negligible. However, otherwise, the algorithm with $\hat{c}_s, \hat{c}_r$ could not guarantee the recovery, since those parameters does not take community size into account.

Indeed, technicality in the proof is very similar with the earlier study [9]. Nonetheless, we stated more general arguments for extension to the K number of communities, and non-equal sizes.

B. Algorithm Description

See Algorithm 1 where

$$L^\Gamma(i, C, u) := -c_1^\Gamma \cdot |C| - c_2^\Gamma \cdot e(i, C) - c_3^\Gamma \cdot \Pi(i, u), \quad (43)$$

$$\Pi(i, u) := \sum_{j \in [m]} N^\Omega(i, j) \cdot u(j), \quad (44)$$

and

$$\hat{L} := L^{\hat{\Gamma}}, \quad \hat{\Gamma} := (\hat{\alpha}, \hat{\beta}, \hat{p}, \hat{\theta}, n, m), \quad (45)$$

and

$$\hat{\alpha} := \frac{\sum_{k \in [K]} e(C_k^{Y^0}, C_k^{Y^0})}{\sum_{k \in [K]} \binom{|C_k^{Y^0}|}{2}}, \quad (46)$$

$$\hat{\beta} := \frac{\sum_{k_1 \neq k_2 \in [K]} e(C_{k_1}^{Y^0}, C_{k_2}^{Y^0})}{\sum_{k_1 \neq k_2 \in [K]} |C_{k_1}^{Y^0}| \cdot |C_{k_2^{Y^0}}|}, \quad (47)$$

$$\hat{\theta} := \frac{|\{(i,j) : u^Y_{C^{Y^0}(i)}(j) = N^\Omega(i,j)\}|}{|\{(i,j) : N^\Omega(i,j) \neq 0\}|}, \quad (48)$$

and $e(i, C)$ is defined as the number of edges between node i, and the set of nodes C. Note that we do not need to estimate $\hat{p}$, since L^Γ is irrelevant to p.

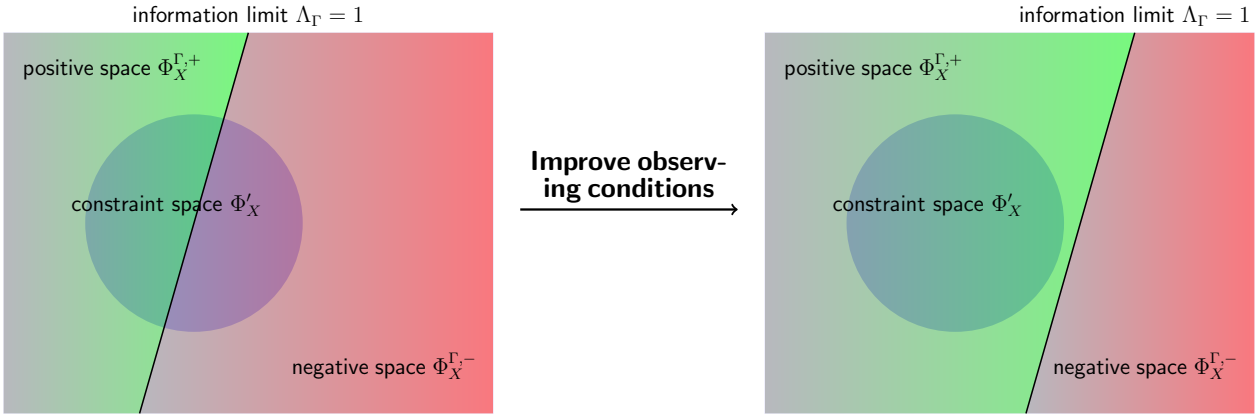


Fig. 1. As improving observing conditions (*e.g.* collect more samples, make observing accuracy better.), positive space expand to cover the constraint, thus it makes exact recovery possible.

Algorithm 1 Exact recovery algorithm

Input G: observe graph, N^Ω partially observed noisy feature matrix, T: number of iterations, $\mathcal{F}$: almost exact recovery algorithm

Output Y: estimator of original community, and feature vector $(X := (C^X, u^X))$

1: **procedure** EXACT RECOVERY($G, N^\Omega, T, \mathcal{F}$)
2: $C^{Y^0} \leftarrow \mathcal{F}(G)$ ▷ step 1
3: **for** $k \in [K]$ **do** ▷ step 2
4: **for** $i \in [m]$ **do**
5: $u_k^Y(i) \leftarrow \text{sign}(\sum_{t \in C_k^{Y_0}} N^\Omega(t,i))$
6: **end for**
7: **end for**
8: **for** $t \in [T]$ **do** ▷ step 3
9: **for** $i \in [n]$ **do**
10: $C^{Y^{t+1}}(i) \leftarrow \text{argmin}_{k\in[K]} \hat{L}(i, C_k^{Y^t}, u_k^Y)$
11: **end for**
12: **end for**
13: **return** $Y := (C^{Y^T}, u^Y)$
14: **end procedure**

The time complexity of Algorithm 1 is $O(\mathcal{T} + n \cdot m \cdot p + \log(n) \cdot n \cdot m \cdot p \cdot K)$

Remark 5. Line 5 can be seen as *majority voting*, which means to pick the value of the feature as the majority of community members have.

C. Performance Proof

Proof for achievabilty of almost exact recovery by some $\mathcal{F}$ can be found in [12], and actually we can deploy any other clustering methods in place of $\mathcal{F}$ such as spectral clustering [13]–[15], non-backtracking matrix based methods [16], semidefinite programming (SDP) [17], and belief propagation (BP) variants [18]. As same as the previous section we will provide the outline of the proof, deferring the full proof to the supplemental materials [10].

Theorem 2. *ψ guided by Algorithm 1 exactly recovers $\boldsymbol{X}$ under constraint $\Phi'_X \subset \Phi_X^{\Gamma,+}$*

Proof: While we go through the proof, following lemma will be very useful:

Lemma 5. *For any $0 < \epsilon < 1$, suppose that $X \sim Bin(\epsilon \cdot n, p)$. Then for any $k \geq 2e$, one has*

$$\Pr\left(X \geq \frac{k \cdot n \cdot p}{\log \frac{1}{\epsilon}}\right) \leq 2 \cdot \exp\left(-\frac{k \cdot n \cdot p}{2}\right) \tag{49}$$

The proof of this lemma is deferred to [10].

Remark 6. Lemma 5 will be helpful in two cases. In step 2, it allows to take the upper bound of the number of "noisy" features. In step 3, it gives the upper bound of the number of edges to "mis"classified nodes.

On account of, $\boldsymbol{X} \in \Phi'_X \subset \Phi_X^{\Gamma,+}$, for some ϵ_0, and sufficiently big sequence index (*i.e.* big n, m), following holds:

$$\Lambda_\Gamma^k(X), \Lambda_\Gamma^{i,j}(X) > 1 + \epsilon_0, \forall i \neq j, k \in [K] \tag{50}$$

Almost exact recovery of community (step 1)

For any $\eta > 0$, we can get C^{Y^0} from $\mathcal{F}$ such that $|C_{P(k)}^{Y^0} \setminus C_k^X| < \eta \cdot n, \forall k \in [K]$ for some permutation P (without loss of generality, we may assume that P is an identity permutation). As we mentioned above, validity of step 1 is proven in the reference papers.

Note that

$$\begin{aligned}\sum_{k\in[K]} |C_k^{Y^0} \setminus C_k^X| &= \sum_{k\in[K]} (|C_k^{Y^0}| - |C_k^{Y^0} \cap C_k^X|) \\ &\overset{(a)}{=} \sum_{k\in[K]} (|C_k^X| - |C_k^{Y^0} \cap C_k^X|) \\ &= \sum_{k\in[K]} |C_k^X \setminus C_k^{Y^0}|\end{aligned} \tag{51}$$

(a) is derived from $\sum_{k\in[K]}|C_k^{Y^0}| = n = \sum_{k\in[K]}|C_k^X|$. Thus for any k, we can derive

$$\begin{aligned}|C_k^{Y^0}\cap C_k^X| =& |C_k^X| - |C_k^X \setminus C_k^{Y^0}| \\ \geq& c_k^X - \eta\cdot n \\ \geq& c_k^X\left(1-\frac{\eta}{c^{\boldsymbol{X}}}\right) = c_k^X(1-O(\eta))\end{aligned} \tag{52}$$

Eq. (52) will be used to prove Claim 3.

Exact recovery of feature vectors (step 2)

What we need to prove is that $\Pr(u^Y \neq u^X)\to 0$.

$$\Pr(u^Y \neq u^X) \leq \sum_{(k,i)\in[K]\times[m]} \Pr(u_k^Y(i)\neq u_k^X(i)) \tag{53}$$

so it is enough to show $\Pr(u_k^Y(i)\neq u_k^X(i)) = o(m^{-1}), \forall k,i$ Without loss of generality, assume that $u_k^X(i)=+1$, then

$$\begin{aligned}\Pr(u_k^Y(i)\neq u_k^X(i)) &= \Pr(u_k^Y(i)=-1) \\ &= \Pr(\sum_{j\in C_k^Y} N^\Omega(j,i) < 0) \\ &\geq \Pr(\sum_{j\in C_k^{Y^0}\cap C_k^X} P_j\cdot(2\cdot\Theta_j-1) \\ &\qquad > -\sum_{j\in C_k^{Y^0}\setminus C_k^X}|P_j\cdot(2\Theta_j-1)|)\end{aligned} \tag{54}$$

hence,

$$\begin{aligned}\text{LHS of Eq. (54)} \leq \Pr\Bigg(&\sum_{j\in C_k^{Y^0}\cap C_k^X} P_j\cdot(2\cdot\Theta_j-1) \\ &> -\sum_{j\in C_k^{Y^0}\setminus C_k^X} P_j\Bigg) \\ \overset{(a)}{=}& o(m^{-1})\end{aligned} \tag{55}$$

For showing (a), since

$$\begin{aligned}&\left\{\sum_{j\in C_k^{Y^0}\cap C_k^X} P_j\cdot(2\cdot\Theta_j-1) > -\sum_{j\in C_k^{Y^0}\setminus C_k^X} P_j\right\} \\ \subset&\left\{\sum_{j\in C_k^{Y^0}\setminus C_k^X} P_j \geq \frac{C\cdot n\cdot p}{\log(1/\eta)}\right\} \\ \cup&\left\{\sum_{j\in C_k^{Y^0}\cap C_k^X} P_j\cdot(2\cdot\Theta_j-1) \geq -\frac{C\cdot n\cdot p}{\log(1/\eta)}\right\}\end{aligned} \tag{56}$$

it is enough to show following two claims, for some $C>0$.

Claim 2. *Following holds:*

$$\Pr\left(\sum_{j\in C_k^{Y^0}\setminus C_k^X} P_j \geq \frac{C\cdot n\cdot p}{\log(1/\eta)}\right) = o(m^{-1}) \tag{57}$$

Claim 3. *Following holds:*

$$\Pr\left(\sum_{j\in C_k^{Y^0}\cap C_k^X} P_j\cdot(2\cdot\Theta_j-1) \geq -\frac{C\cdot n\cdot p}{\log(1/\eta)}\right) = o(m^{-1}) \tag{58}$$

We will use Lemma 5 for showing Claim 2. In the proof of Claim 3, we may use Lemma 2, and note that $\theta=\Omega(1)$, thus $\theta\in[\nu,0.5]$ for some $\nu>0$.

Exact recovery of community (step 3).

We may assume both Step 1 and Step 2 were successful, which implies $u^{Y^0}=u^X$. Define

$$\mathcal{Z}_\delta := \{C^Y : \bigsqcup_{k\in[K]} C_k^Y = [n], \sum_{k\in[K]}|C_k^Y\Delta C_k^X| < \delta\cdot n\} \tag{59}$$

where $A\Delta B := (A\setminus B)\sqcup(B\setminus A)$, then we will show that for some small $\delta>0$ and $\epsilon>0$, if $C^{Y^t}\in\mathcal{Z}_\delta$, then $C^{Y^{t+1}}\in\mathcal{Z}_{\delta/2}$ with probability $1-O(n^{-\epsilon}) = 1-o(\frac{1}{\log(n)})$. Using this, we can show $C^{Y^T}=C^X$ *a.s.* by taking $T=\frac{\log(\eta\cdot n)}{\log(2)}$, and η small enough.

Suppose that $C^{Y^t}\in\mathcal{Z}_\delta$, then we need to show, for some $\epsilon>0$,

$$\exists k\neq C^X(i) : \hat{L}(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - \hat{L}(i, C_k^{Y^t}, u_k^X) > 0 \tag{60}$$

at most $\frac{\delta\cdot n}{2}$ nodes with probability $1-O(n^{-\epsilon})$. We will consider (LHS) of Eq. (60) as following:

$$\begin{aligned}&\hat{L}(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - \hat{L}(i, C_k^{Y^t}, u_k^X) \\ \leq& L(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - L(i, C_k^{Y^t}, u_k^X) \\ &+ |\hat{L}(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - L(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)})| \\ &+ |\hat{L}(i, C_k^{Y^t}, u_k^X) - L(i, C_k^{Y^t}, u_k^X)|\end{aligned} \tag{61}$$

where $L := L^\Gamma$

Proof will be achieved by following 2 statements, for some $\tau>0$.

Claim 4. *With probability* $1-o\left(\frac{1}{\log(n)}\right)$*, there are at most* $\frac{\delta\cdot n}{2\cdot K}$ *nodes, which satisfy:*

$$\begin{aligned}&\exists k\neq C^X(i) : \\ &L(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - L(i, C_k^{Y^t}, u_k^X) > -\frac{\tau\cdot\log(n)}{2}\end{aligned} \tag{62}$$

Claim 5. *If we take* η *small enough in Step 1, following holds:*

$$\Pr(\forall i,C,u, |\hat{L}(i,C,u) - L(i,C,u)| < \frac{\tau\cdot\log(n)}{8}) = 1-o(1) \tag{63}$$

We may use Lemma 2, and Lemma 5 for showing Claim 4. Considering Claim 5 following lemma will be applied:

Lemma 6. *For any* α,β,θ *satisfying Eq.* (3)*, and taking* η *small enough, following holds:*

$$\left|\frac{\alpha-\hat{\alpha}}{\alpha}\right|, \left|\frac{\beta-\hat{\beta}}{\beta}\right|, \left|\frac{\theta-\hat{\theta}}{\theta}\right| = O(\eta) \tag{64}$$

Suppose that Claim 5 succeeds, and Claim 4 succeeds in every $k \in [K]$, and every $T = \frac{\log(\eta \cdot n)}{\log(2)}$ iterations. By Eq. (60), except at most $\frac{\delta \cdot n}{2}$ nodes, $\forall k \neq C^X(i) \in [K]$,

$$\begin{aligned} &\hat{L}(i, C^{Y^t}_{C^X(i)}, u^X_{C^X(i)}) - \hat{L}(i, C^{Y^t}_k, u^X_k) \\ \leq &- \frac{\tau \cdot \log(n)}{2} + \frac{\tau \cdot \log(n)}{8} + \frac{\tau \cdot \log(n)}{8} \\ = &- \frac{\tau \cdot \log(n)}{4} < 0 \end{aligned} \tag{65}$$

which means the node will be assigned correctly in the iteration of step 3. Then finally, $\sum_{k \in [K]} |C^{Y^T}_k \Delta C^X_k| = \lfloor \frac{\eta \cdot n}{2^T} \rfloor = 0$, thus community may exactly recovers. In other words step 3 succeeds.

Now, we can estimate the probability of failure as following:

$$\begin{aligned} \Pr(\text{step 3 fails}) \leq &\Pr(\text{Claim 5 fails}) \\ &+ \sum_{t \in [T]} \Pr(\text{Claim 4 fails}) \\ = &o(1) + o\left(\frac{1}{\log(n)}\right) \cdot T \\ = &o(1) \quad (\because T = O(\log(n))) \end{aligned} \tag{66}$$

By Eq. (66), step 3 succeeds *a.s.* given step 1, 2 succeeds. To sum up, By step 1, 2, 3, Algorithm 1 exactly recovers $\boldsymbol{X}$ *a.s.* ∎

D. Applications

Recommendation system: We can consider each node as internet service user, and each binary feature as the user's preference to an item. Internet company may have both the information about social network [19], [20], and the information about item preference, through tracking their user activity. On the other hand, user clustering is one of the biggest challenge in the recommendation system [21]. Service provider may split users into several groups, and recommend items for each group. One may think almost exact recovery is enough for that. However, occasionally, "mis"classifying users can be critical even if it is very small number of users (*e.g.* giving bad experiences to the users may affect the reputation of the service entirely).

Signal processing: The major challenge of signal processing is to recover original signal from the observed noisy, and sparsely compressed data [22]. We can consider the community as the original data, from which noisy data come. Suppose that we have relations between those signals, then we can exactly recover their original data, applying this model.

V. Experimental Results

In this section, we present experimental results that corroborate our theoretical findings. Specifically, we run Algorithm 1 on synthetic data, which is generated as follows.

1) Pick n, m, K, θ
2) Set $\alpha, \beta \propto \frac{\log(n)}{n}$, and $p \propto \left(\frac{\log(n)}{m} + \frac{\log(m)}{n}\right)$
3) Partition n nodes into K communities of sizes $c_0 \cdot n \leq c_1 \cdot n \leq \cdots \leq c_{K-1} \cdot n$
4) Randomly draw a graph as per the symmetric SBM(α, β)
5) Randomly draw community features with the minimum distance $d \cdot m$
6) Randomly draw n node features with p, θ

Here, we will simulate only the second and third stages of the proposed algorithm, assuming that the first stage is successfully done by any off-the-shelf community recovery algorithms [12]. We simulate this by feeding a $(1-\eta)$-correct partition to the input of the second stage of the algorithm.

Shown in Fig. 2 are the simulation results. Shown on the left side is the empirical success rate of exact recovery for various values for the minimum community size and the minimum distance between community features, and shown on the right side is the evaluated values of Λ_Γ. For the empirical success rates, we run our algorithm 5 times with randomly generated observations. Corroborating our main theorems, one can see that sharp phase transition happens around the border $\Lambda_\Gamma = 1$.

In Fig. 3, we provide another set of simulation results, where all the other settings remain the same except that p is doubled from the previous setting. This corresponds to the improved observation condition, which is illustrated in Fig. 1. As one can see from the righthand figure, the value of Λ_Γ strictly increases for all pairs of the minimum community size and the minimum feature distance, and hence the phase transition boundary shifts left downward. From the lefthand figure, we can observe that the empirical success rates also increase, exhibiting phase transition around the new (improved) boundary.

VI. Conclusion

In this paper, we studied the joint recovery of K communities and their binary feature vectors. The information-theoretic limit is precisely characterized. Further, we developed a computationally efficient recovery algorithm that achieves the limit. In order to generalize the existing result to the case of K communities of different sizes, we defined a topology of target space and provided a sharp analysis using it. We conclude the paper with a list of a few open problems: First, symmetric SBM seems unrealistic, and hence one may want to extend our results to asymmetric cases; Second, binary feature vectors are limited, and a more general feature space needs to be considered.

References

[1] M. Girvan and M. E. J. Newman, "Community structure in social and biological networks," *Proceedings of the National Academy of Sciences*, vol. 99, no. 12, pp. 7821–7826, 2002. [Online]. Available: http://www.pnas.org/content/99/12/7821

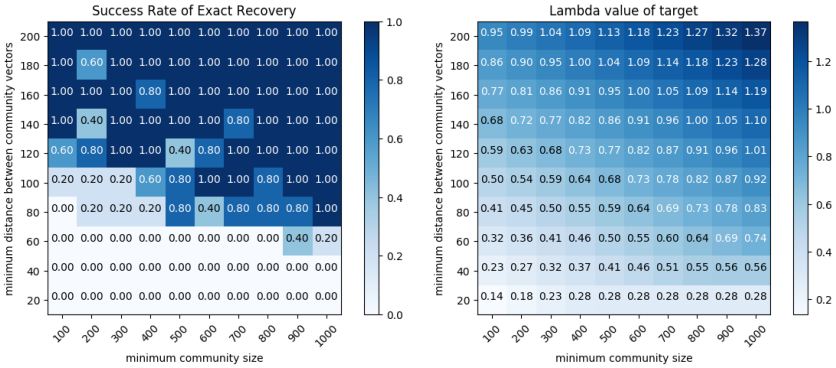
[2] J. Yang, J. McAuley, and J. Leskovec, "Community detection in networks with node attributes," in *Data Mining (ICDM), 2013 IEEE 13th international conference on*. IEEE, 2013, pp. 1151–1156.

[3] J. Tang, X. Hu, and H. Liu, "Social recommendation: a review," *Social Network Analysis and Mining*, vol. 3, no. 4, pp. 1113–1133, 2013.

[4] P. W. Holland, K. B. Laskey, and S. Leinhardt, "Stochastic blockmodels: First steps," *Social networks*, vol. 5, no. 2, pp. 109–137, 1983.

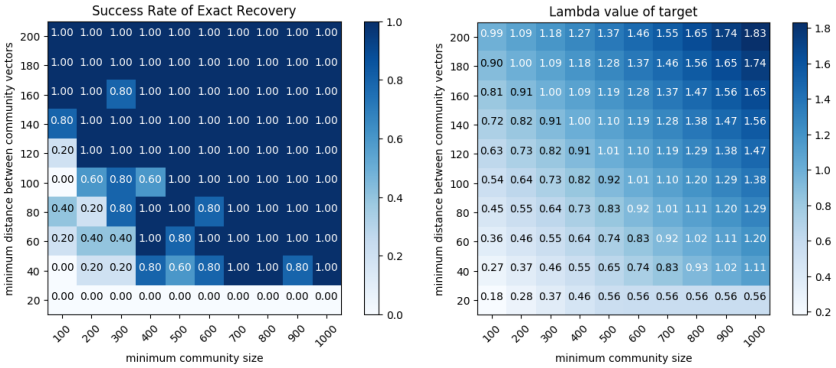
[5] P. K. Gopalan and D. M. Blei, "Efficient discovery of overlapping communities in massive networks," *Proceedings of the National Academy of Sciences*, vol. 110, no. 36, pp. 14 534–14 539, 2013. [Online]. Available: http://www.pnas.org/content/110/36/14534

[6] E. Abbe, "Community detection and stochastic block models: recent developments," *arXiv preprint arXiv:1703.10146*, 2017.



(a) (b)

Fig. 2. $n = 10000, m = 2000, K = 10, \alpha = \frac{\log(n)}{n} \cdot 16, \beta = \frac{\log(n)}{n}, p = \left(\frac{\log(m)}{n} + \frac{\log(n)}{m}\right) \cdot 10, \theta = 0.1, \eta = 0.01$, Unlike Λ_Γ showing smoothness in Fig. 2b, you can see comparatively vivid bisection in Fig. 2a



(a) (b)

Fig. 3. $n = 10000, m = 2000, K = 10, \alpha = \frac{\log(n)}{n} \cdot 16, \beta = \frac{\log(n)}{n}, p = \left(\frac{\log(m)}{n} + \frac{\log(n)}{m}\right) \cdot 20, \theta = 0.1, \eta = 0.01$ As we described in Fig. 1, by collecting more samples (*i.e.* increase p), you can see the expansion of the positive space

[7] C. Aicher, A. Z. Jacobs, and A. Clauset, "Adapting the stochastic block model to edge-weighted networks," *arXiv preprint arXiv:1305.5782*, 2013.

[8] H. Saad and A. Nosratinia, "Community detection with side information: Exact recovery under the stochastic block model," *IEEE Journal of Selected Topics in Signal Processing*, pp. 1–1, 2018.

[9] H. C. C. S. Kwangjun Ahn, Kangwook Lee, "Binary rating estimation with graph side information," in *NIPS*, 2018.

[10] C. S. Jisang Yoon, Kangwook Lee, "Supplemental materials," http://bit.ly/2RvPaBL, 2018.

[11] M. Nathanson, "Review: Elements of information theory. john wiley and sons, inc., hoboken, nj, 2006, xxiv + 748 pp., ISBN 0-471-24195-4, $111.00. by thomas m. cover and joy a. thomas," *The American Mathematical Monthly*, vol. 120, no. 2, pp. 182–187, 2013.

[12] E. Abbe and C. Sandon, "Community detection in general stochastic block models: Fundamental limits and efficient algorithms for recovery," in *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, 2015, pp. 670–688. [Online]. Available: https://doi.org/10.1109/FOCS.2015.47

[13] P. Chin, A. Rao, and V. Vu, "Stochastic block model and community detection in sparse graphs: A spectral algorithm with optimal rate of recovery," in *COLT*, ser. JMLR Workshop and Conference Proceedings, vol. 40. JMLR.org, 2015, pp. 391–423.

[14] J. Lei, A. Rinaldo *et al.*, "Consistency of spectral clustering in stochastic block models," *The Annals of Statistics*, vol. 43, no. 1, pp. 215–237, 2015.

[15] C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou, "Achieving optimal misclassification proportion in stochastic block models," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 1980–2024, 2017.

[16] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborová, and P. Zhang, "Spectral redemption in clustering sparse networks," *Proceedings of the National Academy of Sciences*, vol. 110, no. 52, pp. 20 935–20 940, 2013.

[17] A. Javanmard, A. Montanari, and F. Ricci-Tersenghi, "Phase transitions in semidefinite relaxations," *Proceedings of the National Academy of Sciences*, vol. 113, no. 16, pp. E2218–E2223, 2016.

[18] E. Mossel and J. Xu, "Density evolution in the degree-correlated stochastic block model." in *COLT*, 2016, pp. 1319–1356.

[19] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of facebook networks," *CoRR*, vol. abs/1102.2166, 2011.

[20] L. A. Adamic and N. Glance, "The political blogosphere and the 2004 us election: divided they blog," in *Proceedings of the 3rd international workshop on Link discovery*. ACM, 2005, pp. 36–43.

[21] C. Bouras and V. Tsogkas, "Improving news articles recommendations via user clustering," *Int. J. Machine Learning & Cybernetics*, vol. 8, no. 1, pp. 223–237, 2017.

[22] V. Tuzlukov, *Signal processing noise*. CRC Press, 2002, vol. 8.