Multi-View Stochastic Block Models

Vincent Cohen-Addad [*1] **Tommaso d'Orsi** [*1 2] **Silvio Lattanzi** [*1] **Rajai Nasser** [*1]

Abstract

Graph clustering is a central topic in unsupervised learning with a multitude of practical applications. In recent years, multi-view graph clustering has gained a lot of attention for its applicability to real-world instances where one has access to multiple data sources. In this paper we formalize a new family of models, called *multi-view stochastic block models* that captures this setting. For this model, we first study efficient algorithms that naively work on the union of multiple graphs. Then, we introduce a new efficient algorithm that provably outperforms previous approaches by analyzing the structure of each graph separately. Furthermore, we complement our results with an information-theoretic lower bound studying the limits of what can be done in this model. Finally, we corroborate our results with experimental evaluations.

1. Introduction

Clustering graphs is a fundamental topic in unsupervised learning. It is used in a variety of fields, including data mining, social sciences, statistics, and more. The goal of graph clustering is to partition the vertices of the graph into disjoint sets so that similar vertices are grouped together and dissimilar vertices lie in different clusters. In this context, several notions of similarity between vertices have been studied throughout the years resulting in different clustering objectives and clustering algorithms (Von Luxburg, 2007; Ng et al., 2001; Bansal et al., 2004; Goldberg, 1984; Dasgupta, 2016).

Despite the rich literature, most of the algorithmic results in graph clustering only focus on the setting where a single graph is presented in input. This is in contrast with the increasing practical importance of multimodality and with the growing attention in applied fields to multi-view or multi-layer clustering (Paul & Chen, 2016; Corneli et al., 2016; Han et al., 2015; De Bacco et al., 2017; Khan & Maji, 2019; Zhong & Pun, 2021; Hu et al., 2019; Abavisani & Patel, 2018; Kim et al., 2016; Gujral et al., 2020; Ni et al., 2016; De Santiago et al., 2023; Papalexakis et al., 2013; Gujral & Papalexakis, 2018; Gorovits et al., 2018). In practice it is in fact observed that while a single data source only offers a specific characterization of the underlying objects, leading to a coarse partition of the data, a careful combination of multiple views often allows a semantically richer network structure to emerge (Fu et al., 2020; Fang et al., 2023). For a practical example, consider the task of clustering users of a social network platform like Facebook, Instagram or X. To solve such task one could simply cluster the friendship graph, or one could cluster such graph by looking together at the friendship graph, the co-like graph (a graph where two users are connected if they like the same picture/video), the co-comment graph (a graph where two users are connected if they comment on the same post), the co-repost graph (a graph where two users are connected if they repost the same post) and so on and so forth. In practice, one would expect the second approach to work better in many settings because it provides a more fine-grained description of the behaviors of the users.

Despite the large number of basic applications, very little is known on the theoretical aspect of the problem. Several works (Paul & Chen, 2016; Corneli et al., 2016; Han et al., 2015; De Bacco et al., 2017) consider the multi-layer stochastic block models where the goal is to identify k communities given several instances (i.e., layer or view) of the stochastic block model, each with k communities. In this paper, we would like to work in a more general and more realistic setup, where there are k communities but each instance only provides *partial information* about these k communities. Very recently and concurrently to us, (De Santiago et al., 2023) introduced the *multi-view stochastic block model*. In this model, one is given in input multiple graphs, each coming from a stochastic block model, and the goal is to leverage the information contained in the graphs to recover the underlying clustering structure. More precisely, given a vector of labels[1] $\mathbf{z}$ where the labels capture the clustering assignment and are in $[k]$, and

[*]Equal contribution [1]Google Research [2]BIDSA, Bocconi. Correspondence to: Tommaso d'Orsi <tommaso.dorsi@unibocconi.it>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

[1]We write random variables in boldface.

t graphs $\mathbf{G}_1, \dots, \mathbf{G}_t$, where each graph $\mathbf{G}_\ell$ is drawn independently from a stochastic block model with 2 labels and possibly distinct parameters, we are interested in designing an algorithm to weakly recover the underlying vector $\mathbf{z}$. One important aspect of the model is that none of the input graphs $\mathbf{G}_1, \dots, \mathbf{G}_t$ may contain enough information to recover the full clustering structure (for example because $2 < k$), nevertheless one can show that if enough graphs are observed it is possible to recover the clustering structure of the underlying instance.

Armed with this new model we study different approaches to cluster the input graphs $\mathbf{G}_1, \dots, \mathbf{G}_t$. First, we note that the natural approach (sometimes used in practice) of merging the graphs and then clustering the union of the graphs, called *early fusion*, leads to suboptimal results. Then we design a more careful *late fusion* clustering algorithm that first clusters all the graphs separately and then carefully merges their results. This shows the superiority of late over early fusion. Finally, we complement our results with an information-theoretic lower bound studying the limits of what can be done in this model. The bounds obtained are a drastic improvement over the ones obtained by (De Santiago et al., 2023).

Model Before formally introducing our model, we recall the classic definition of the stochastic block model.

The k community symmetric stochastic block model (see (Abbe, 2017) for a survey) denotes the following joint distribution $(\mathbf{x}, \mathbf{G}) \sim \mathsf{SBM}_{n,k,d,\varepsilon}$ over a vector of n labels in $[k]$ and a graph on n vertices:

- draw $\mathbf{x}$ from $[k]^n$ uniformly at random;
- for each distinct $i, j \in [n]$, independently create an edge ij in $\mathbf{G}$ with probability $(1 + (1 - \frac{1}{k})\varepsilon)\frac{d}{n}$ if $\mathbf{x}_i = \mathbf{x}_j$ and probability $(1 - \frac{\varepsilon}{k})\frac{d}{n}$ otherwise.

We denote the conditional distribution of $\mathbf{G}$ given $\mathbf{x} = x$ as $\mathsf{SBM}_{k,d,\varepsilon}(x)$. Given a graph $\mathbf{G}$ sampled according to this model, the goal is to recover the (unknown) underlying vector of labels as well as possible.

Most of the statistical and computational phenomena at play can already be observed in the simplest settings with two communities, so we will often focus on those. For $k = 2$, we denote the distribution by $\mathsf{SBM}_{n,2,d,\varepsilon}$, i.e., we explicitly replace the subscript k. It will also be convenient to use $\{+1, -1\}$ for the community labels instead of $[2]$, so we will sometimes do this. The labeling convention should be clear from the context.

One of the most widely studied natural objective in the context of stochastic block models is that of *weak recovery* –asking to approximately recover the communities. Specifically, we say that an algorithm achieves weak recovery for $\{\mathsf{SBM}_{n,k,d,\varepsilon}\}_{n\in\mathbb{N}}$ if the correlation of the algorithm's output $\hat{\mathbf{x}}(\mathbf{G}) \in [k]^n$ and the underlying vector $\mathbf{x}$ of labels is better than random as n grows,[2]

$$\mathbb{P}\left(R(\hat{\mathbf{x}}(\mathbf{G}_\ell), \mathbf{x}_\ell) \geqslant \frac{1}{k} + \Omega_{d,\varepsilon/k}(1)\right) \geqslant 1 - o(1)\,, \quad (1)$$

where $R(\hat{x}, x)$ is the agreement between $\hat{x}$ and x, defined as[3]

$$R(\hat{x}, x) = \max_{\pi\in P_k} \frac{1}{n} \sum_{i\in[n]} [\![\hat{x}_i = \pi(x_i)]\!]\,, \quad (2)$$

and P_k is the permutation group of $[k]$.

A sequence of works (Decelle et al., 2011; Massoulié, 2014; Mossel et al., 2014; 2015b; Abbe & Sandon, 2016a; Mossel et al., 2018; Montanari & Sen, 2016), have studied the statistical and computational landscapes of this objective, with great success. The emerging picture (Bordenave et al., 2015; Abbe & Sandon, 2016a; Montanari & Sen, 2016) shows that it is possible to achieve weak recovery in polynomial time whenever $d\varepsilon^2/k^2 > 1$, this value is called the Kesten-Stigum threshold. Further evidence (Hopkins & Steurer, 2017) suggests that this threshold is optimal for polynomial time algorithms. In particular, for the special case of weak recovery with 2 communities (Mossel et al., 2015b) showed that the problem is solvable (also computationally efficiently) *if and only if* $d\varepsilon^2/4 > 1$. For larger values of k a gap between information-theoretic results and efficient algorithms exists (Abbe & Sandon, 2016b; Banks et al., 2016).

In the context of multimodality, we define the following multi-view model.

Model 1.1 (Multi-View stochastic block model)**.** Let $k \geqslant 1$ and let $\mathcal{T}$ be a sequence of t tuples $(d_\ell, \varepsilon_\ell)$ where $d_\ell \geqslant 0$ and $\varepsilon_\ell \in (0, 1)$. We refer to the following joint distribution $(\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t)) \sim (\mathcal{T}$,k,t)-MV-$\mathsf{SBM}_n$ as the $(\mathcal{T}, k, t)$-multi-view stochastic block model:

1. for each $\ell \in [t]$, independently draw a mapping $\mathbf{f}_\ell : [k] \to \{\pm 1\}$ uniformly at random;
2. independently draw a vector $\mathbf{z}$ from $[k]^n$ uniformly at random;
3. for each $\ell \in [t]$, independently draw a graph $\mathbf{G}_\ell \sim \mathsf{SBM}_{n,2,d_\ell,\varepsilon_\ell}(\mathbf{f}_\ell(\mathbf{z}))$, where $\mathbf{f}_\ell(\mathbf{z})$ is the n-dimensional vector with entries $\mathbf{f}_\ell(\mathbf{z}_1), \dots, \mathbf{f}_\ell(\mathbf{z}_n)$.

Given $\mathbf{G}_1, \dots, \mathbf{G}_t$, the goal is to approximately recover the unknown vector $\mathbf{z}$ of labels.

[2] We use $o(1)$ to denote a function f such that $\lim_{n\to\infty} f(n)/1 = 0$.

[3] We use Iverson's brackets to denote the indicator function.

When $\mathcal{T} = \{(d_\ell, \varepsilon_\ell)\}_{\ell\in[t]}$ is such that $(d_\ell, \varepsilon_\ell) = (d, \varepsilon)$ for some d, ε, we denote the model simply by (d,ε,k,t)-MV-SBM$_n$.

Although Model 1.1 captures the algorithmic phenomena of multi-view models used in practice, more general versions of Model 1.1 could be defined, we discuss them in Section 6. Similarly to the vanilla stochastic block model, weak recovery can also be defined for Model 1.1. We say that an algorithm achieves weak recovery for ($\mathcal{T}$,k,t)-MV-SBM$_n$ with t observations, if it outputs a vector $\hat{\mathbf{z}}(\mathbf{G}_1, \dots \mathbf{G}_t)$ satisfying:

$$\mathbb{P}\left(R(\hat{\mathbf{z}}(\mathbf{G}_1, \dots \mathbf{G}_t), \mathbf{z}) \geqslant \frac{1}{k} + \Omega(1)\right) \geqslant 1 - o_t(1)\,. \tag{3}$$

Differently from the vanilla stochastic block model, the complexity of Model 1.1 is governed both by the SBM parameters in $\mathcal{T}$ *and* by the number of observations t. A good algorithm should then achieve weak recovery with the best possible multiway tradeoff between the edge-densities of the graphs, the biases and the number of observations at hand. That is, extract as much information as possible so to require as few observations as possible. This novel interplay of parameters immediately raises two natural questions, which are the main focus of this work.

How many observations are needed? The problem gets easier the larger the number of observations one has access to (see Appendix A for a formal proof). It is also easy to see that for $t = o(\log k)$, it is information theoretically *impossible* to approximately recover the communities (since $\log_2 k$ bits are needed to encode k labels). Furthermore, as we will see, stronger lower bounds can also be obtained.

How many observations suffice? To understand how many observations suffice to recover the communities, it is instructive to consider the union graph $\mathbf{G}^* = \bigcup_{\ell\in[t]} \mathbf{G}_\ell$ of an instance from (d,ε,k,t)-MV-SBM$_n$, which turns out to follow a k-communities stochastic block model with parameters $d^* = \Theta(dt)\,, \varepsilon^* = \Theta(\varepsilon)$ (see Appendix A). Building on the aforementioned results, this implies that at least $t \geqslant \Omega(k^2/d\varepsilon^2)$ observations are needed for efficient weak recovery of the communities from $\mathbf{G}^*$! However, as we show later, exponentially better algorithms can bridge this gap.

1.1. Results

Weak recovery Our main algorithmic result shows that weak recovery for ($\mathcal{T}$,k,t)-MV-SBM$_n$ can be achieved in polynomial time with only $O(\log k)$ many observations.

Theorem 1.2 (Weak recovery for multi-view models)**.** *Let $n, k > 0$. Let $(\mathbf{z}, (\mathbf{f_1}, \mathbf{G_1}), \dots (\mathbf{f_t}, \mathbf{G}_t)) \sim$ ($\mathcal{T}$,k,t)-MV-SBM$_n$ for a sequence of tuples $\mathcal{T} = \{(d_\ell, \varepsilon_\ell)\}_{\ell=1}^t$, each satisfying $d_\ell \cdot \varepsilon_\ell^2/4 > 1$. Then there exists a constant $C_\mathcal{T} > 0$ depending only on $\mathcal{T}$, such that if $t \geqslant \Omega\left(\frac{\log k}{C_\mathcal{T}^2}\right)$, weak recovery of $\mathbf{z}$ in the sense of (3) is possible. Moreover, the underlying algorithm runs in polynomial time.*

Theorem 1.2 implies that whenever the algorithm has access to $\Theta(\log k)$ observations, each above the relative Kesten-Stigum threshold, the guarantees of the underlying algorithm match the aforementioned trivial lower bound, up to constant factors. Moreover, as we will see in Section 4, the underlying algorithm turns out to be surprisingly simple and efficient.

The algorithm in Theorem 1.2 applies a specialized weak-recovery algorithm on each view $\mathbf{G}_\ell$ to obtain a matrix $\hat{\mathbf{X}}_\ell$ estimating $\mathbf{f}_\ell(\mathbf{z})\mathbf{f}_\ell(\mathbf{z})^\mathsf{T}$ and achieving the correlation

$$C_\ell \leqslant \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{f}_\ell(\mathbf{z})_i = \mathbf{f}_\ell(\mathbf{z})_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{f}_\ell(\mathbf{z}) \neq \mathbf{f}_\ell(\mathbf{z})_j\right]\,, \tag{4}$$

where C_ℓ depends only on d_ℓ, ε_ℓ. The algorithm then proceeds into processing the outputs $\hat{\mathbf{X}}_\ell$ *in a blackbox fashion* to produce an estimate $\hat{\mathbf{z}}$ of $\mathbf{z}$.

The constant $C_\mathcal{T}$ in Theorem 1.2 is the average of the correlations $(C_\ell)_{\ell\in[t]}$. It is natural to wonder whether the dependency of the number of observations on $C_\mathcal{T}$ is needed. Moreover, as for canonical stochastic block models, it is natural to ask what the exact phase transition of Model 1.1 is. While we leave this latter fascinating question open, our next result shows that if we want an algorithm that processes estimates in a blackbox fashion, then some dependency on $C_\mathcal{T}$ is indeed needed.

Theorem 1.3 (Lower bound for multi-view models - Informal)**.** *Let $n, k > 0$. Let $(\mathbf{z}, (\mathbf{f_1}, \mathbf{G_1}), \dots (\mathbf{f_t}, \mathbf{G}_t)) \sim$ ($\mathcal{T}$,k,t)-MV-SBM$_n$ for a sequence of tuples $\mathcal{T} = \{(d_\ell, \varepsilon_\ell)\}_{\ell=1}^t$, each satisfying $d_\ell \cdot \varepsilon_\ell^2/4 > 1$. Assume that for every $\ell \in [t]$ we have an estimate[4] $\hat{\mathbf{X}}_\ell$ of $\mathbf{f}_\ell(\mathbf{z})\mathbf{f}_\ell(\mathbf{z})^\mathsf{T}$ satisfying a pair-wise correlation (as in (4)) of at least $C_\ell > 0$, and let $C_\mathcal{T}$ be the average correlation.*

If $t = o\left(\frac{\log k}{C_\mathcal{T}}\right)$, then by only using the estimates $\hat{\mathbf{X}}_1, \dots, \hat{\mathbf{X}}_t$, it is information-theoretically impossible to return a vector $\hat{\mathbf{z}}$ satisfying

$$\mathbb{P}\left(R(\hat{\mathbf{z}}, \mathbf{z}) \geqslant \frac{1}{k} + \Omega(1)\right) \geqslant 0.99\,. \tag{5}$$

Exact recovery Another widely studied objective for stochastic block models is that of *exact recovery*, where the goal is to correctly classify all vertices in the graph

[4]Such estimates might be obtained by applying a blackbox weak-recovery algorithm for SBM$_{n,2,d,\varepsilon}$ on each of $\mathbf{G}_1, \dots, \mathbf{G}_t$, and which has the mentioned correlation guarantee.

(see (Abbe et al., 2015; Mossel et al., 2015a; Abbe, 2017) and references therein). In the context of Model 1.1 this objective becomes

$$\mathbb{P}\left(R(\hat{\mathbf{z}}, \mathbf{z}) = 1\right) \geqslant 1 - o(1)\,. \tag{6}$$

As a corollary we show that, when given access to more views, the algorithm behind Theorem 1.2 can achieve exact recovery.

Corollary 1.4 (Exact recovery for multi-view models)**.** *Consider the settings of Theorem 1.2, if* $t \geqslant \Omega\left(\frac{\log n}{C_\tau^2}\right)$ *then exact recovery of* $\mathbf{z}$ *in the sense of Equation* (6) *is possible. Moreover, the underlying algorithm runs in polynomial time.*

Experiments Theorem 1.2 show hows, for Model 1.1, *late fusion* algorithms can provide better guarantees –by requiring an exponentially smaller number of observations to achieve the same error in a large parameters regime– than early fusion algorithms. We further corroborated these findings with experiments on synthetic data in Section 5.

Organization

The rest of the paper is organized as follows. We introduce the main ideas in Section 2. In Section 3 we introduce our specialized weak recovery algorithm for the standard stochastic block model. This is then used in the design of the algorithm behind Theorem 1.2 in Section 4. Experiments are presented in Section 5. Future directions and conclusions are discussed in Section 6. In Appendix A we show the limits of algorithms using the union graph. Appendix B contains a proof of (the formal version of) Theorem 1.3. Deferred proofs are presented in Appendix C.

Notation

We denote random variables in **boldface**. We hide constant factors using the notation $O(\cdot), \Omega(\cdot), \Theta(\cdot)$. We write $O_\delta(\cdot), \Omega_\delta(\cdot)$ to specify that the hidden constant may depend on the parameter δ. Similarly, we sometimes write C_δ to denote a constant depending only on δ. We further denote the indicator function with Iverson's brackets $[\![\cdot]\!]$. Given functions $f, g : \mathbb{R} \to \mathbb{R}$, we say $f \in o_n(g)$ if $\lim_{n\to\infty} f(n)/g(n) = 0$. Similarly we write $f \in \omega_n(g)$ if $g \in o_n(f)$. With a slight abuse of notation we often write $o_n(g)$ to denote a function in $o_n(g)$. When the context is clear we drop the subscript. In particular, we often write $o(1)$ to denote functions that tends to zero as n grows. We say that an event happens with high probability if this probability is at least $1 - o(1)$. For a set $S \subseteq [n]$, we write $\mathbf{i} \overset{u.a.r.}{\sim} S$ to denote an element drawn uniformly at random. For a given probability distribution and a measurable event $\mathcal{E}$, we denote the probability that the event occurs by $\mathbb{P}(\mathcal{E})$. We denote the complement event by $\mathcal{E}^c$.

For a vector $v \in \mathbb{R}^n$, we write $\|v\|$ for its Euclidean norm. For a matrix $M \in \mathbb{R}^{n\times n}$, we denote by $\|M\|$ its spectral norm and by $\|M\|_{\mathrm{F}}$ its Frobenius norm. We also let $\|M\|_1 := \sum_{ij} |M_{ij}|$. We denote the i-th row of M by M_i. For a graph G with n vertices, we denote by $A(G)$ its adjacency matrix. When the context is clear we simply write A. If the graph is directed, row A_i contains the outgoing edges of vertex i.

For a given vector of labels $z \in [k]^n$, we denote by $c_1(z) \ldots, c_k(z)$ the n-dimensional indicator vectors of the k communities defined by z. In the interest of simplicity, we often denote instances $(\mathbf{z}, (\mathbf{f_1}, \mathbf{G_1}), \ldots (\mathbf{f_t}, \mathbf{G}_t))$ drawn from ($\mathcal{T}$,k,t)-MV-SBM$_n$ simply by $\boldsymbol{\mathcal{I}}$. For $z \in [k]^n$, we also denote by ($\mathcal{T}$,k,t)-MV-SBM$_n(z)$ the distribution of $(\mathbf{z}, (\mathbf{f_1}, \mathbf{G_1}), \ldots (\mathbf{f_t}, \mathbf{G}_t)) \sim$ ($\mathcal{T}$,k,t)-MV-SBM$_n$ conditioned on the event $\mathbf{z} = z$. We often call $z \in [k]^n$ a "community vector".

We say that an algorithm runs in time T, if in the worst case it performs at most T elementary operations.

2. Techniques

We outline here the main ideas behind Theorem 1.2 and Theorem 1.3. In the interest of clarity, we limit our discussion to (d,ε,k,t)-MV-SBM$_n$.

Behavior of the union graph The algorithm behind Theorem 1.2 is remarkably simple and leverages known algorithms for weak recovery of stochastic block models (particularly related to the robust algorithms of (Ding et al., 2022; 2023)). As a first step, to gain intuition, it is instructive to understand why the union graph instead requires $t \geqslant \Omega(k^2)$ observations (see Theorem A.1). An instance $\boldsymbol{\mathcal{I}}$ of (d,ε,k,t)-MV-SBM$_n$ is given by a vector $\mathbf{z} \in [k]^n$ and a collection of t independent pairs $(\mathbf{f}_1, \mathbf{G}_1), \ldots, (\mathbf{f}_t, \mathbf{G}_t)$, where each $\mathbf{G}_\ell$ is sampled from $\mathsf{SBM}_{2,d,\varepsilon}(\mathbf{f}_\ell(\mathbf{z}))$. Since, by definition, each edge $\{i, j\}$ appears in $\mathbf{G}_\ell$ with probability $\left(1 + \frac{\varepsilon}{2} \cdot \mathbf{f}_\ell(\mathbf{z})_i \cdot \mathbf{f}_\ell(\mathbf{z})_j\right) \cdot \frac{d}{n}$, in the union graph $\bigcup_{\ell\in[t]} \mathbf{G}_\ell$ the same edge appears roughly with probability

$$\frac{d}{n} \cdot \sum_{\ell\in[t]} \left(1 + \frac{\varepsilon}{2} \cdot \mathbf{f}_\ell(\mathbf{z})_i \cdot \mathbf{f}_\ell(\mathbf{z})_j\right)\,.$$

The intuition here is that if $\mathbf{z}_i \neq \mathbf{z}_j$ the second term in the sum would be close to 0, while for $\mathbf{z}_i = \mathbf{z}_j$ it would be $\frac{\varepsilon t}{2}$. Hence, we will see this edge in the union graph roughly with probability

$$\frac{dt}{n} \cdot \left(1 + O(\varepsilon) \cdot [\![\mathbf{z}_i = \mathbf{z}_j]\!]\right).$$

That is, the union graph behaves similarly to a vanilla stochastic block model with k communities, bias $O(\varepsilon)$ and

expected degree dt . As stated in the introduction, existing efficient algorithms can achieve weak recovery for that distribution whenever $(dt)\varepsilon^2/k^2 > \Omega(1)$, implying $t \geqslant \Omega(k^2)$ in the regime $4 < d\varepsilon^2 \leqslant O(1)$ where weak recovery for each observation is possible.

Amplifying the signal-to-noise ratio via black-box estimators The above approach of taking the union graph and then running community detection on it yields sub-optimal guarantees because the graph does not keep all information regarding the instance $\boldsymbol{\mathcal{I}}$. Our strategy to overcome this issue is to proceed in the reverse order: *first* extract as much information as possible from each graph, and *then* combine the data. Concretely, in the context of (d,ε,k,t)-MV-SBM$_n$, our plan is to accurately estimate the matrix $\mathbf{f}_\ell(\mathbf{z})\mathbf{f}_\ell(\mathbf{z})^\mathsf{T}$ for each graph $\mathbf{G}_\ell$. Indeed, the polynomial $\sum_{\ell\in[t]} \mathbf{f}_\ell(\mathbf{z})_i\mathbf{f}_\ell(\mathbf{z})_j$ strongly correlates with $[\![\mathbf{z}_i = \mathbf{z}_j]\!]$ in the sense that

$$\begin{aligned} t &= \mathbb{E}\left[\sum_{\ell\in[t]} \mathbf{f}_\ell(\mathbf{z})_i\mathbf{f}_\ell(\mathbf{z})_j \,\middle|\, \mathbf{z}_i = \mathbf{z}_j\right] \\ &> \mathbb{E}\left[\sum_{\ell\in[t]} \mathbf{f}_\ell(\mathbf{z})_i\mathbf{f}_\ell(\mathbf{z})_j \,\middle|\, \mathbf{z}_i \neq \mathbf{z}_j\right] = 0\,. \end{aligned}$$

In other words, we can accurately estimate whether $\mathbf{z}_i = \mathbf{z}_j$ or not by accurately estimating the products $\sum_{\ell\in[t]} \mathbf{f}_\ell(\mathbf{z})_i\mathbf{f}_\ell(\mathbf{z})_j$. Now, we do not have access to the functions $\mathbf{f}_\ell(\mathbf{z})$ but we can hope (see the subsequent paragraphs) that existing weak recovery algorithms can provide a close enough estimate in the sense

$$\begin{aligned} t\cdot C_{d,\varepsilon} &= \mathbb{E}\left[\sum_{\ell\in[t]} \hat{\mathbf{x}}(\mathbf{G}_\ell)_i\hat{\mathbf{x}}(\mathbf{G}_\ell)_j \,\middle|\, \mathbf{z}_i = \mathbf{z}_j\right] \qquad (7)\\ &> \mathbb{E}\left[\sum_{\ell\in[t]} \hat{\mathbf{x}}(\mathbf{G}_\ell)_i\hat{\mathbf{x}}(\mathbf{G}_\ell)_j \,\middle|\, \mathbf{z}_i \neq \mathbf{z}_j\right] = 0\,. \end{aligned}$$

If so, we may simply decide whether i, j should be clustered together based on how large $\sum_{\ell\in[t]} \hat{\mathbf{x}}(\mathbf{G}_\ell)_i\hat{\mathbf{x}}(\mathbf{G}_\ell)_j$ is. By independence of the observations, standard concentration of measure results tell us that $\Omega(\log(n)/C_{d,\varepsilon}^2)$ observations[5] would suffice to *exactly* predict all the n^2 pairs (and hence achieve exact recovery with this number of observations).

Improvements via neighborhoods intersection Continuing with the above line of thinking, one can further improve the dependency on t to $t = \Theta(\log k)$ as promised in Theorem 1.2. For a label $p \in [k]$, let $c_p(z) \in \{0,1\}^n$ be the indicator vector of the corresponding community. The improvement comes from observing that for $\ell \neq \ell'$ and for any *typical* labelling z (i.e. a labelling that is approximately balanced), we have large separation between the community indicator vectors $\|c_p(z) - c_{p'}(z)\|^2 \geqslant \Omega(n/k)$. The crucial consequence is that for a fixed index $i \in [n]$, we do not need to guess correctly $[\![z_i = z_j]\!]$ for all j and we may misclassify some pairs. Indeed if $A_i \in \{0,1\}^n$ is a vector with entries $(A_i)_j$ that accurately predicts $[\![z_i = z_j]\!]$ up to a $\rho < n/k$ misclassification error, then we can deduce whether i, j come from the same community by verifying if A_i and A_j agree on the majority of their entries. Concretely, by the reverse triangle inequality it holds that

$$\begin{aligned} &|\|A_i - A_j\| - \|c_p(z) - c_{p'}(z)\|| \\ &\quad\leqslant \|c_p(z) - A_i\| + \|c_{p'}(z) - A_j\| \leqslant O(n/k)\,. \end{aligned}$$

That is, we are still able to exactly deduce whether i, j sit in the same community! The improvement over t then comes as $O(\log(k)/C_\delta^2)$ observations suffice to bound the misclassification error by n/k times a tiny constant. Finally, we remark that we are bound to misclassify some vertices as for some vertices i the estimator vector $\mathbf{A}_i$ will not accurately represent its community when $t \leqslant O(\log(k)/C_\delta^2)$ (this is due to the well-known gap between weak recovery and exact recovery in the vanilla stochastic block model (Abbe, 2017)).

The pair-wise weak recovery estimator So far, we glossed over the fact that we do not have an algorithm returning an accurate estimate $\hat{\mathbf{x}}(\mathbf{G}_\ell)\hat{\mathbf{x}}(\mathbf{G}_\ell)^\mathsf{T}$ of $\mathbf{f}_\ell(\mathbf{z})\mathbf{f}_\ell(\mathbf{z})^\mathsf{T}$ given the graph $\mathbf{G}_\ell$. Notice that for a pair $(\mathbf{x}, \mathbf{G}) \sim \mathsf{SBM}_{n,2,d,\varepsilon}$, an algorithm achieving weak recovery returns a vector $\hat{\mathbf{x}}(\mathbf{G}) \in \{\pm 1\}^n$ such that

$$\begin{aligned} \Omega(n^2) &\leqslant \mathbb{E}\left[\langle \hat{\mathbf{x}}(\mathbf{G}), \mathbf{x}\rangle^2\right] \\ &\leqslant \mathbb{E}\left[\langle \hat{\mathbf{x}}(\mathbf{G})\hat{\mathbf{x}}(\mathbf{G})^\mathsf{T}, \mathbf{x}\mathbf{x}^\mathsf{T}\rangle\right] \end{aligned}$$

which is enough to obtain the separation required in Equation (7). Indeed, the above implies that on average

$$\begin{aligned} \mathbb{E}\left[\hat{\mathbf{x}}(\mathbf{G})_\mathbf{i}\hat{\mathbf{x}}(\mathbf{G})_\mathbf{j} \mid \mathbf{x_i} = \mathbf{x_j}\right] - \mathbb{E}\left[\hat{\mathbf{x}}(\mathbf{G})_\mathbf{i}\hat{\mathbf{x}}(\mathbf{G})_\mathbf{j} \mid \mathbf{x_i} \neq \mathbf{x_j}\right] \\ \geqslant \Omega_{d,\varepsilon}(1)\,, \end{aligned}$$

which is enough to carry out the strategy outlined in the previous paragraphs. Notice that, a priori, it is not clear whether these estimators should work for (d,ε,k,t)-MV-SBM$_n$ as the model introduces some subtle difficulties compared to SBM$_{n,2,d,\varepsilon}$. Most importantly, the labels in $\mathbf{f}_\ell(\mathbf{z})$ –and hence the edges in $\mathbf{G}_\ell$– are *not* pair-wise independent.

We bypass this obstacle carrying out the analysis *after* conditioning on the choice of $\mathbf{f}_\ell$, so that, even though the communities may be unbalanced, the edges are again independent. Now, for highly unbalanced communities the expected degree of each vertex is an accurate predictor for its community, hence weak recovery is easy to achieve. On the other hand, one can treat slightly unbalanced communities as *perturbed* balanced communities and apply the node-robust algorithm of (Ding et al., 2023).

[5] This is better than $\Omega(k^2)$ as long as $k \geqslant \Omega(\sqrt{\log n})$.

Remark 2.1 (Connection with (Liu et al., 2022)). Liu, Moitra and Raghavendra studied a joint distribution model over hypergraphs with independence edges. In the special case of graphs, their model can be seen as a simpler version of Model 1.1 in which every $\mathbf{f}_\ell(\cdot)$ is *known*, and each d_ℓ is a constant. For this model, (Liu et al., 2022) beats random guessing: They produce a unit vector that correlates with the community vector better than a random vector guess would. However, they provide no rounding strategy. The algorithmic techniques in (Liu et al., 2022) are very different from ours and do not imply a result of the form of Theorem 1.2 for Model 1.1.

Information theoretic lower bounds for black-box algorithms The proof of Theorem 1.3 consists of two main steps. In the first step, we bound how much information an estimate $\hat{\mathbf{X}}_\ell$ with pair-wise correlation C_ℓ can reveal about $\mathbf{z}$. We show that this can be bounded (in terms of mutual information) as $I(\mathbf{z}; \hat{\mathbf{X}}_\ell) \leqslant O(C_\ell \cdot n)$. In the second step we use an adapted version of Fano's inequality to show that if $\hat{\mathbf{z}}$ is an estimate of $\mathbf{z}$ achieving (5), then $\hat{\mathbf{z}}$ must have at least $\Omega(n \cdot \log k)$ bits of information about $\mathbf{z}$, i.e., $I(\mathbf{z}; \hat{\mathbf{z}}) \geqslant \Omega(n \cdot \log k)$.

Now if $\hat{\mathbf{z}}$ is obtained by only processing $(\hat{\mathbf{X}}_\ell)_{\ell \in [t]}$, then by using the data processing inequality, we can show that

$$\begin{aligned}\Omega(n \cdot \log k) &\leqslant \sum_{t \in [t]} I(\mathbf{z}; \hat{\mathbf{X}}_\ell) \\ &\leqslant \sum_{t \in [t]} O(C_\ell \cdot n) \leqslant O(C_{\mathcal{T}} \cdot t \cdot n)\,,\end{aligned}$$

where $C_{\mathcal{T}}$ is the average of the correlations $(C_\ell)_{\ell \in [t]}$. From this we deduce that we must have $t \geqslant \Omega\left(\frac{\log k}{C_{\mathcal{T}}}\right)$.

3. Specialized weak recovery for vanilla SBMs

General weak recovery results (Abbe, 2017) for stochastic block models turn out to be too weak for our objective. We rely instead on the following stronger statement, which we obtain by exploiting the robust algorithms of (Ding et al., 2022), (Ding et al., 2023), and which also works down to the Kesten-Stigum threshold. This specialized estimator will be used in the main algorithm behind Theorem 1.2.

Theorem 3.1 (Pair-wise weak recovery for unbalanced 2 communities stochastic block model)**.** *Let $n, d, \varepsilon > 0$ be satisfying $d\varepsilon^2/4 - 1 > 0$. Let $\mathbf{x} = (\mathbf{x}_i)_{i \in [n]} \in \{\pm 1\}^n$ be a sequence of i.i.d. binary random variables with $\mathbb{P}(\mathbf{x}_i = +1) = p$. There exists a polynomial time algorithm such that, on input $\mathbf{G} \sim \mathsf{SBM}_{n,2,d,\varepsilon}(\mathbf{x})$, returns with probability $1 - o(1)$ a matrix $\hat{\mathbf{X}}(\mathbf{G}) \in [-1, +1]^{n \times n}$ satisfying $\forall i, j \in [n]$*

$$C_{d,\varepsilon} \leqslant \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G})_{ij} \,\middle|\, \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G})_{ij} \,\middle|\, \mathbf{x}_i \neq \mathbf{x}_j\right]$$

for some constant $C_{d,\varepsilon} > 0$.

Notice that the algorithm only receives $\mathbf{G}, d, \varepsilon$ and hence it *does not* know p. Note also that in the above, if one of the events $\mathbf{x}_i = \mathbf{x}_j$ or $\mathbf{x}_i \neq \mathbf{x}_j$ has probability 0 (i.e., if $p = 0$ or $p = 1$), then we adopt the convention that the corresponding conditional expectation is 0.

We defer the proof of Theorem 3.1 to Appendix C.

4. The algorithm

In this section we prove Theorem 1.2. We start by describing the underlying algorithm. Throughout the section, for a given t, we assume $\mathcal{T} = \{(d_\ell, \varepsilon_\ell)\}_{\ell=1}^t$ to be a sequence of t tuples $(d_\ell, \varepsilon_\ell)$ each satisfying $d_\ell \cdot \varepsilon_\ell^2/4 - 1 > 0$. For each $\ell \in [t]$, let C_ℓ be the weak-recovery constant that is achievable for $\mathsf{SBM}_{n,d_\ell,\varepsilon_\ell}$ in the sense of Theorem 3.1 (recall this constant depends only on d_ℓ, ε_ℓ) and let

$$\overline{C} = \frac{1}{t} \sum_{\ell \in [t]} C_\ell > 0\,.$$

We also assume $k \leqslant n^{1-\Omega(1)}$.

Algorithm 1 Community detection for multi-view stochastic block models

Input: $k, G_1, \ldots, G_t$.
Output: Community vector $\hat{\mathbf{z}}$ in $[k]^n$.
For each G_ℓ with $\ell \in [t]$, run the community detection algorithm of Theorem 3.1.
Construct the directed graph $\mathbf{F}$ on the vertex set $[n]$ as follows.
for $i = 1$ **to** n **do**
  Add an outgoing edge to the n/k vertices $j \in [n]$ with largest $\sum_{\ell \in [t]} \hat{\mathbf{X}}(G_\ell)_{ij}$.
end for
Run Algorithm 2 on the adjacency matrix $\mathbf{A}$ of $\mathbf{F}$ and return the resulting vector.

Remark 4.1 (Running time). By Theorem 3.1, the first step of the algorithm takes time $O\left(t \cdot n^{O(1)}\right)$. Computing the values of $\sum_{\ell \in [t]} \hat{\mathbf{X}}(G_\ell)_{ij}$ for all pairs takes time $O(t \cdot n^2)$. Drawing the edges takes time $O(n^2 \log n)$. Hence the algorithm runs in time $O\left((tn)^{O(1)} + T\right)$ where T is the running time of Algorithm 2.

Graph structure on balanced instances Algorithm 1 will work on typical instances from ($\mathcal{T}$,k,t)-MV-SBM_n. In particular, it will work on instances that are approximately balanced, as described below.

Definition 4.2 (Balanced vector)**.** Let $z \in [k]^n$. We say that z is balanced if for all $p \in [k]$, it holds

$$\left(1 - n^{-\Omega(1)}\right) \frac{n}{k} \leqslant \|c_p(z)\|^2 \leqslant \left(1 + n^{-\Omega(1)}\right) \frac{n}{k}\,.$$

If z is balanced we also say that $\mathcal{I} \sim (\mathcal{T}$,k,t)-MV-SBM$_n(z)$ is balanced.

It is immediate to see that a random instance $(\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t)) \sim (\mathcal{T}$,k,t)-MV-SBM$_n$ is balanced with high probability.

Fact 4.3 (Probability of balanced instance). Let $\mathcal{I} := (\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t)) \sim (\mathcal{T}$,k,t)-MV-SBM$_n$,. Then, with probability $1 - n^{-10}$, $\mathcal{I}$ is balanced.

The proof of Fact 4.3 is straightforward and we defer it to Appendix C.

On balanced instances with sufficiently many observations, the adjacency matrix $\mathbf{A}$ of $\mathbf{F}$ becomes a good approximation of the true community matrix $\sum_{i\in[k]} c_i(z)c_i(z)^\mathsf{T}$.

Lemma 4.4 (Graphs structure from good instances)**.** *Let $n, k, t > 0$. Let $\mathcal{I} := (\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t))$. Then Algorithm 1 constructs an adjacency matrix $\mathbf{A}$ such that $\forall p \in [k], \forall i \in [n]$*

$$\mathbb{E}\left[\left\|\mathbf{A}_i - c_p(\mathbf{z})\right\|^2 \,\middle|\, \mathbf{z}_i = p\right] \leqslant O(n) \cdot \left(n^{-\Omega(1)} + e^{-\Omega(\bar{C}^2 \cdot t)}\right) .$$

To prove Lemma 4.4 we require an intermediate step.

Fact 4.5. Consider the setting of Lemma 4.4. There exists a constant $C^* \in [-t, t]$ such that, for $i, j \in [n]$,

$$\mathbb{P}\left(\sum_{\ell\in[t]} \hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} < C^* - \frac{\overline{C}\cdot t}{3} \,\middle|\, \mathbf{z}_i = \mathbf{z}_j\right) \leqslant e^{-\Omega(\overline{C}^2\cdot t)} ,$$

$$\mathbb{P}\left(\sum_{\ell\in[t]} \hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \geqslant C^* - \frac{\overline{C}\cdot t}{3} \,\middle|\, \mathbf{z}_i \neq \mathbf{z}_j\right) \leqslant e^{-\Omega(\overline{C}^2\cdot t)} .$$

We defer the proof of Fact 4.5 to Appendix C. We are now ready to prove Lemma 4.4.

Proof of Lemma 4.4. Fix $i \in [n]$. We can limit our analysis after conditioning on the event $\mathcal{E}(\mathbf{z})$ that $\mathbf{z}$ is balanced. Indeed, we have

$$\mathbb{E}\left[\left\|\mathbf{A}_i - c_p(\mathbf{z})\right\|^2 \,\middle|\, \mathcal{E}(\mathbf{z})^c, \mathbf{z}_i = p\right] \cdot \mathbb{P}\left(\mathcal{E}(\mathbf{z})\right) \leqslant O(n) \cdot \mathbb{P}\left(\mathcal{E}(\mathbf{z})\right) \leqslant n^{-\Omega(1)} .$$

Let C^* be the constant of Fact 4.5. Define

$$\mathbf{B}_{ij} = \sum_{\ell\in[t]} \hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} ,$$

and let $\mathbf{A}'_i = (\mathbf{A}'_{ij})_{j\in[n]}$ be the binary vector defined as

$$\mathbf{A}'_{ij} = \left[\!\left[\mathbf{B}_{ij} \geqslant C^* - \frac{\bar{C}\cdot t}{3}\right]\!\right] .$$

Notice that $\|\mathbf{A}'_i\|_1$ is the number of indices j with $\mathbf{B}_{ij} \geqslant C^* - \frac{\bar{C}\cdot t}{3}$. Now from the definition of the binary vector $\mathbf{A}_i$, we know that $\mathbf{A}_{ij} = 1$ if and only if $\mathbf{B}_{ij}$ is among the top n/k values in $\{\mathbf{B}_{ij'} : j' \in [n]\}$. From this it is not hard to see that that $\|\mathbf{A}_i - \mathbf{A}'_i\|_1$ can be bounded from above by how much $\|\mathbf{A}'_i\|_1$ deviates from n/k , that is

$$\|\mathbf{A}_i - \mathbf{A}'_i\|_1 \leqslant \left|\|\mathbf{A}'_i\|_1 - \frac{n}{k}\right| .$$

Now assuming that $\mathcal{E}(\mathbf{z})$ holds, we have

$$\|c_p(z)\|_1 = \|c_p(z)\|^2 = \frac{n}{k} \pm n^{1-\Omega(1)} ,$$

hence

$$\begin{aligned}\|\mathbf{A}_i - \mathbf{A}'_i\|_1 &\leqslant \left|\|\mathbf{A}'_i\|_1 - \|c_p(z)\|_1\right| + n^{1-\Omega(1)} \\ &\leqslant \|\mathbf{A}'_i - c_p(z)\|_1 + n^{1-\Omega(1)} .\end{aligned}$$

Since $c_p(\mathbf{z})$ and $\mathbf{A}_i$ are binary vectors, we have

$$\begin{aligned}\|\mathbf{A}_i - c_p(z)\|^2 &= \|\mathbf{A}_i - c_p(z)\|_1 \\ &\leqslant \|\mathbf{A}_i - \mathbf{A}'_i\|_1 + \|\mathbf{A}'_i - c_p(z)\|_1 ,\end{aligned}$$

and hence, given $\mathcal{E}(\mathbf{z})$, we have

$$\|\mathbf{A}_i - c_p(z)\|^2 \leqslant 2\|\mathbf{A}'_i - c_p(z)\|_1 + n^{1-\Omega(1)} . \tag{8}$$

Now notice that

$$\begin{aligned}&|(\mathbf{A}'_i)_j - c_p(\mathbf{z})_j| \\ &= [\![(\mathbf{A}'_i)_j = 1]\!][\![c_p(\mathbf{z})_j = 0]\!] + [\![(\mathbf{A}'_i)_j = 0]\!][\![c_p(\mathbf{z})_j = 1]\!] \\ &= [\![(\mathbf{A}'_i)_j = 1]\!][\![\mathbf{z}_j \neq p]\!] + [\![(\mathbf{A}'_i)_j = 0]\!][\![\mathbf{z}_j = p]\!] .\end{aligned}$$

Now using Fact 4.5 and leveraging the fact that $\mathbb{P}\left(\mathcal{E}(\mathbf{z})^c \mid \mathbf{z}_i = p\right) \leqslant O\left(n^{-10}\right)$ we get

$$\begin{aligned}&\mathbb{E}\left[|(\mathbf{A}'_i)_j - c_p(\mathbf{z})_j| \mid \mathcal{E}(\mathbf{z}), \mathbf{z}_i = p\right] \\ &\qquad \leqslant e^{-\Omega(\overline{C}^2\cdot t)} + O\left(n^{-10}\right) .\end{aligned}$$

Combining this with (8) we conclude that

$$\begin{aligned}&\mathbb{E}\left[\left\|\mathbf{A}_i - c_p(\mathbf{z})\right\|^2 \,\middle|\, \mathcal{E}(\mathbf{z}), \mathbf{z}_i = p\right] \\ &\quad \leqslant O(n) \cdot \left(n^{-\Omega(1)} + e^{-\Omega(\bar{C}^2\cdot t)}\right) .\end{aligned}$$

□

Rounding Thanks to Lemma 4.4, an application of the following rounding scheme –sometimes called *second moment rounding*, as one may see $\mathbf{A}$ as an estimate of $\sum_p c_p(z)c_p(z)^\mathsf{T}$– suffices to compute the true communities.

Remark 4.6 (Running time). Each step in the outer loop takes time $O(n^2)$. Overall Algorithm 2 runs in $O(k \cdot n^2)$.

Algorithm 2 Second moment rounding

Input: A matrix $A \in \{0,1\}^{n\times n}$.
Output: Community vector $\hat{\mathbf{z}}$ in $[n]^k$.
Let $S = \emptyset$.
for $p = 1$ **to** k (stop earlier if $S = [n]$) **do**
  Pick uniformly at random $\mathbf{i} \in [n] \setminus \mathbf{S}$. Set $\mathbf{S}_p = \{\mathbf{i}\}$.
  for $j \in [n] \setminus \mathbf{S}, j \neq \mathbf{i}$ **do**
    Set $j \in \mathbf{S}_p$ if $\|A_{\mathbf{i}} - A_j\|^2 \leqslant \frac{n}{k}$.
  end for
  Add $\mathbf{S}_p$ to $\mathbf{S}$.
end for
Assign each $i \in [n] \setminus \mathbf{S}$ to a set $\mathbf{S}_{\mathbf{p}}$, where $\mathbf{p}$ is chosen uniformly at random.
Return: the vector $\hat{\mathbf{z}}$ with $\hat{\mathbf{z}}_i = p$ if and only if $i \in \mathbf{S}_p$.

As first step of the proof, we introduce a new definition.

Definition 4.7 (Representative row)**.** Let $z \in [k]^n$, let $c_1(z), \dots, c_k(z)$ be the indicator vectors of the labels in z and let $A^*(z) = \sum_p c_p(z)c_p(z)^\mathsf{T}$. For a matrix $A \in \{0,1\}^{n\times n}$, we say A_i is q-representative if

$$\|A_i - A^*(z)_i\|^2 \leqslant n \cdot e^{-q\cdot\overline{C}^2\cdot t}. \tag{9}$$

We denote by $\mathcal{R}_q$ the set of q representatives of A.

The use of representative rows is convenient because, for sufficiently many observations, it is easy to see that rows which are representative of the same community must be close together, while rows that are representative of different communities must be far from each other.

Lemma 4.8. *Let $n, k, t > 0$, and let $q > 0$ be the hidden constant in Lemma 4.4. Let $t \geqslant C\frac{\log k}{\overline{C}^2}$ for a large enough universal constant $C > 0$. Let $z \in [k]^n$ be balanced. Suppose that at each iteration of the outer loop, Algorithm 2 picks some $A_{\mathbf{i}}$ that is a q-representative. Then $\max_{\pi\in P_k} \sum_{i\in\mathcal{R}_q(z)} [\![\hat{\mathbf{z}}_i = \pi(z_i)]\!] = |\mathcal{R}_q|$, where P_k is the permutation group over $[k]$.*

We defer the proof of Lemma 4.8 to Appendix C. Now, to prove Theorem 1.2, it remains to argue that, with high probability, *first* there are few non-representatives in $\mathbf{A}$, and *second* Algorithm 2 picks q-representatives at each iteration of the outer loop.

Lemma 4.9. *Let $n, k, t > 0$, and let $q > 0$ be the hidden constant in Lemma 4.4. Let $t = C\frac{\log k}{\overline{C}^2}$ for a large enough universal constant $C > 0$. Let $\mathcal{I} := (\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t)) \sim (\mathcal{T}$,k,t)-MV-SBM$_n$. Let $\mathbf{A}$ be the matrix constructed by Algorithm 1. On input $\mathbf{A}$, with probability at least $1 - k^{-\Omega(1)}$, the following holds:*

- *there are at least $n \cdot (1 - k^{-\Omega(1)})$ rows in $\mathbf{A}$ which are $\Omega(q)$-representatives,*
- *step 1.(a) of Algorithm 2 only picks $\Omega(q)$-representative vectors.*

Proof. By Fact 4.3 we may assume $\mathbf{z}$ is balanced. By Lemma 4.4 and Markov's inequality, with probability $1 - e^{-\Omega(q\cdot\overline{C}^2\cdot t)}$ there are at most $O(n) \cdot e^{-\Omega(q\cdot\overline{C}^2\cdot t)}$ indices $\mathbf{i} \in [n]$ such that $\mathbf{A}_{\mathbf{i}}$ is not a $\Omega(q)$-representative. Note that if C is large enough, then $e^{-\Omega(q\cdot\overline{C}^2\cdot t)} = k^{-\Omega(1)}$. At every iteration of the loop, if we pick a representative vector we remove at most $(1 + o(1))\frac{n}{k}$ indices, since $\mathbf{z}$ is balanced. Hence at each iteration there are at least $\frac{n}{k}(1 - o(1))$ indices that are $\Omega(q)$-representative and which have not yet been picked. It follows that the probability we never pick an index that is not $\Omega(q)$-representative is at least $\left(1 - \frac{n\cdot k^{-\Omega(1)}}{n/k}\right)^k \geqslant 1 - k^{-\Omega(1)}$ as desired (by choosing C to be large enough). □

Theorem 1.2 now immediately follows combining Fact 4.3, Lemma 4.4, Lemma 4.8 and Lemma 4.9.

Exact recovery Lemma 4.9 also implicitly yield exact recovery for sufficiently many observations.

Corollary 4.10. *Let $n, k, t > 0$, and let $q > 0$ be the hidden constant in Lemma 4.4. Let $t \geqslant C\frac{\log n}{\overline{C}^2}$ for a large enough universal constant $C > 0$. Let $\mathcal{I} := (\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots (\mathbf{f}_t, \mathbf{G}_t)) \sim (\mathcal{T}$,k,t)-MV-SBM$_n$. Let $\mathbf{A}$ be the matrix constructed by Algorithm 1. On input $\mathbf{A}$, with probability at least $1 - n^{-\Omega(1)}$, all rows in $\mathbf{A}$ are $\Omega(q)$-representatives.*

Proof. By Lemma 4.4 and Markov's inequality, with probability $1 - e^{-\Omega(q\cdot\overline{C}^2\cdot t)}$ there are at most $O(n) \cdot e^{-\Omega(q\cdot\overline{C}^2\cdot t)}$ indices $\mathbf{i} \in [n]$ such that $\mathbf{A}_{\mathbf{i}}$ is not a $\Omega(q)$-representative. Therefore, for $t \geqslant \frac{C}{\overline{C}^2}\log n$ no such index exists. □

Since by Lemma 4.8 only indices corresponding to rows that are not $\Omega(q)$-representative can be misclassified, by Fact 4.3, Lemma 4.4, Lemma 4.9 and Corollary 4.10 we obtain Corollary 1.4.

5. Experiments

We show here experiments on synthetic data sampled from (d,ε,k,t)-MV-SBM$_n$. The estimator in Theorem 3.1 is complex and relies on a high order sum-of-squares program, making it hard to implement in practice. Nevertheless, it is reasonable to believe that: *(i)* the guarantees of Theorem 3.1 are only sufficient, but not necessary, to obtain Theorem 1.2; *(ii)* other estimators provide the guarantees of Theorem 3.1. For this reason, it makes sense to test Algorithm 1 with other community detection algorithms.

The next figures compares the results on $(\mathbf{z}, (\mathbf{f}_1, \mathbf{G}_1), \dots, (\mathbf{f}_t, \mathbf{G}_t)) \sim$ (d,ε,k,t)-MV-SBM$_n$ (for a wide range of parameters) of the following algorithms:

A.1 Louvain's algorithm (Blondel et al., 2008) on the union graph $\bigcup_{i\in[t]} \mathbf{G}_t$.

A.2 Algorithm 1 with Louvain's algorithm applied in place of the estimator of Theorem 3.1 .

The y-axis measures agreement as defined in Equation (2). Results are averaged over 20 simulations.

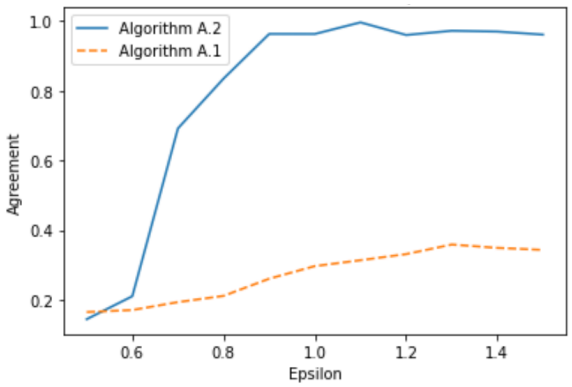

Figure 1. Fixing $t = 10$, $n = 1000$, $k = 10$, $d = 50$ and varying ε in $[0.5, 1.5]$.

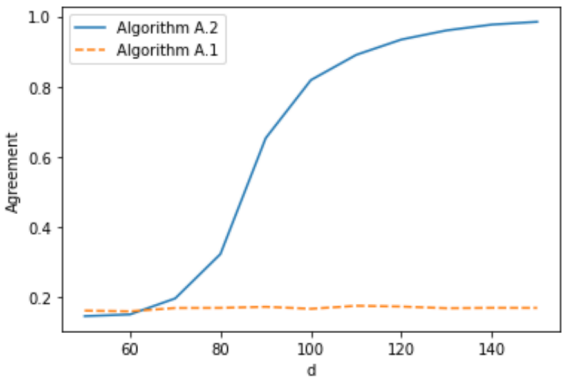

Figure 2. Fixing $t = 10$, $n = 1000$, $k = 10$, $\varepsilon = 0.5$ and varying d in $[50, 150]$.

6. Conclusions and future directions

The introduction of Model 1.1 raises several natural questions, for which we only provide initial answers.

On the phase transition threshold One of the most interesting question concerns the phase transition of the model. Concretely, one may expect a rich interplay between the signal-to-noise ratio of each of the observed graphs (possibly below the relative KS threshold) and the number of observations required to weakly recover the hidden vector $\mathbf{z}$. We leave the characterization of this trade-off beyond Theorem 1.2 and Theorem 1.3 as a fascinating open question.

From 2 communities to k communities in the multi-view model Another natural question concern the generalization to a model in which each view may have $2 \leqslant k_\ell \leqslant k$ communities. The ideas outlined above translate in principle to these settings but the correctness appears difficult to prove. Concretely, any estimator achieving guarantees comparable to Theorem 3.1 but for more than 2 communities can immediately be plugged-in Algorithm 1 to achieve weak recovery in these more general settings.[6] However, to the best of our knowledge existing weak-recovery algorithms for SBM$_{n,k,d,\varepsilon}$ do not lead to estimators of this form.

A first obstacle appears to be the fact that for each distribution the labels $\mathbf{f}_\ell(\mathbf{z})_1 \dots, \mathbf{f}_\ell(\mathbf{z})_n$ are *not* independent. Existing estimators for stochastic block models instead crucially relies on the independence of the labels. Notice that independence holds when conditioning on $\mathbf{f}_\ell$ but due to the high variance of random $[k] \to [k']$ mappings, an argument along these lines would require an analysis for communities whose distribution is not known a priori. This makes each observation $\mathbf{G}_\ell$ more akin to a stochastic block model with *unknown* parameters. These models have been studied in (Abbe & Sandon, 2015) in the context of *partial recovery* and exact recovery but are not known to achieve the notion of weak recovery we require.

A second obstacle arising from the current proof structure is that existing robust algorithm for weak recovery –which are used in Theorem 3.1– are only known to work for $k = 2$. But a generalization of these algorithms to $k > 2$ appears to be difficult to analyze (the current proofs is already more than 200 pages long! (Ding et al., 2022)). Hence, it remains open how to provide guarantees for more general models.

Acknowledgments

We thank David Steurer for insightful discussions in the early stages of this work. Tommaso d'Orsi is partially supported by the project MUR FARE2020 PAReCoDi.

Impact Statement

This paper presents work whose goal is to provide a theoretical foundation to machine learning heuristics often used in practice. We do not feel that any particular societal consequences of our work should be highlighted here.

[6] Without changing the proof of correctness of the algorithm!

References

Abavisani, M. and Patel, V. M. Deep multimodal subspace clustering networks. *IEEE Journal of Selected Topics in Signal Processing*, 12(6):1601–1614, 2018.

Abbe, E. Community detection and stochastic block models: recent developments. *The Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

Abbe, E. and Sandon, C. Recovering communities in the general stochastic block model without knowing the parameters. *Advances in neural information processing systems*, 28, 2015.

Abbe, E. and Sandon, C. Achieving the ks threshold in the general stochastic block model with linearized acyclic belief propagation. *Advances in Neural Information Processing Systems*, 29, 2016a.

Abbe, E. and Sandon, C. Crossing the ks threshold in the stochastic block model with information theory. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 840–844. IEEE, 2016b.

Abbe, E., Bandeira, A. S., and Hall, G. Exact recovery in the stochastic block model. *IEEE Transactions on information theory*, 62(1):471–487, 2015.

Banks, J., Moore, C., Neeman, J., and Netrapalli, P. Information-theoretic thresholds for community detection in sparse networks. In *Conference on Learning Theory*, pp. 383–416. PMLR, 2016.

Bansal, N., Blum, A., and Chawla, S. Correlation clustering. *Machine learning*, 56:89–113, 2004.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

Bordenave, C., Lelarge, M., and Massoulié, L. Non-backtracking spectrum of random graphs: community detection and non-regular ramanujan graphs. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pp. 1347–1357. IEEE, 2015.

Corneli, M., Latouche, P., and Rossi, F. Exact icl maximization in a non-stationary temporal extension of the stochastic block model for dynamic networks. *Neurocomputing*, 192:81–91, 2016.

Dasgupta, S. A cost function for similarity-based hierarchical clustering. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 118–127, 2016.

De Bacco, C., Power, E. A., Larremore, D. B., and Moore, C. Community detection, link prediction, and layer interdependence in multilayer networks. *Phys. Rev. E*, 95:042317, Apr 2017. doi: 10.1103/PhysRevE.95.042317. URL https://link.aps.org/doi/10.1103/PhysRevE.95.042317.

De Santiago, K., Szafranski, M., and Ambroise, C. Mixture of stochastic block models for multiview clustering. In *ESANN 2023-European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 151–156, 2023.

Decelle, A., Krzakala, F., Moore, C., and Zdeborová, L. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011.

Ding, J., d'Orsi, T., Nasser, R., and Steurer, D. Robust recovery for stochastic block models. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 387–394. IEEE, 2022.

Ding, J., d'Orsi, T., Hua, Y., and Steurer, D. Reaching kesten-stigum threshold in the stochastic block model under node corruptions. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 4044–4071. PMLR, 2023.

Fang, U., Li, M., Li, J., Gao, L., Jia, T., and Zhang, Y. A comprehensive survey on multi-view clustering. *IEEE Transactions on Knowledge and Data Engineering*, 2023.

Fu, L., Lin, P., Vasilakos, A. V., and Wang, S. An overview of recent multi-view clustering. *Neurocomputing*, 402: 148–161, 2020.

Goldberg, A. V. Finding a maximum density subgraph. 1984.

Gorovits, A., Gujral, E., Papalexakis, E. E., and Bogdanov, P. Larc: Learning activity-regularized overlapping communities across time. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 1465–1474, 2018.

Gujral, E. and Papalexakis, E. E. Smacd: Semi-supervised multi-aspect community detection. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pp. 702–710. SIAM, 2018.

Gujral, E., Pasricha, R., and Papalexakis, E. Beyond rank-1: Discovering rich community structure in multi-aspect graphs. In *Proceedings of The Web Conference 2020*, pp. 452–462, 2020.

Han, Q., Xu, K., and Airoldi, E. Consistent estimation of dynamic and multi-layer block models. In Bach, F. and Blei, D. (eds.), *Proceedings of the 32nd International*

Conference on Machine Learning, volume 37 of *Proceedings of Machine Learning Research*, pp. 1511–1520, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/hanb15.html.

Hopkins, S. B. and Steurer, D. Efficient bayesian estimation from few samples: community detection and related problems. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 379–390. IEEE, 2017.

Hu, D., Nie, F., and Li, X. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9248–9257, 2019.

Khan, A. and Maji, P. Approximate graph laplacians for multimodal data clustering. *IEEE transactions on pattern analysis and machine intelligence*, 43(3):798–813, 2019.

Kim, M., Han, D. K., and Ko, H. Joint patch clustering-based dictionary learning for multimodal image fusion. *Information Fusion*, 27:198–214, 2016.

Liu, S., Mohanty, S., and Raghavendra, P. On statistical inference when fixed points of belief propagation are unstable. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 395–405. IEEE, 2022.

Massoulié, L. Community detection thresholds and the weak ramanujan property. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pp. 694–703, 2014.

Montanari, A. and Sen, S. Semidefinite programs on sparse random graphs and their application to community detection. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 814–827, 2016.

Mossel, E., Neeman, J., and Sly, A. Belief propagation, robust reconstruction and optimal recovery of block models. In *Conference on Learning Theory*, pp. 356–370. PMLR, 2014.

Mossel, E., Neeman, J., and Sly, A. Consistency thresholds for the planted bisection model. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 69–75, 2015a.

Mossel, E., Neeman, J., and Sly, A. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 162:431–461, 2015b.

Mossel, E., Neeman, J., and Sly, A. A proof of the block model threshold conjecture. *Combinatorica*, 38(3):665–708, 2018.

Ng, A., Jordan, M., and Weiss, Y. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14, 2001.

Ni, J., Cheng, W., Fan, W., and Zhang, X. Self-grouping multi-network clustering. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 1119–1124. IEEE, 2016.

Papalexakis, E. E., Akoglu, L., and Ience, D. Do more views of a graph help? community detection and clustering in multi-graphs. In *Proceedings of the 16th International Conference on Information Fusion*, pp. 899–905. IEEE, 2013.

Paul, S. and Chen, Y. Consistent community detection in multi-relational data through restricted multi-layer stochastic blockmodel. 2016.

Von Luxburg, U. A tutorial on spectral clustering. *Statistics and computing*, 17:395–416, 2007.

Zhong, G. and Pun, C.-M. Latent low-rank graph learning for multimodal clustering. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pp. 492–503. IEEE, 2021.

A. Failure of community detection on the union graph

In this section we provide rigorous evidence that efficient algorithm cannot achieve comparable guarantees to Theorem 1.2 by only considering the union graph $\bigcup_{i\in[t]}\mathbf{G}_i$ for $(\mathbf{z},(\mathbf{f_1},\mathbf{G_1}),\dots(\mathbf{f}_t,\mathbf{G}_t))\sim$ (d,ε,k,t)-MV-SBM$_n$. Concretely, we prove the following theorem.

Theorem A.1 (Limits of weak recovery from the union graph)**.** *Let $n,k,d,\varepsilon>0$, assume that $t\geqslant 100\cdot(\log k)^2$, and let $\boldsymbol{\mathcal{I}}:=(\mathbf{z},(\mathbf{f_1},\mathbf{G_1}),\dots(\mathbf{f_t},\mathbf{G}_t))\sim$ (d,ε,k,t)-MV-SBM$_n$. With probability at least $1-k^{-\Omega(1)}$ over the draws of $\mathbf{f}_1,\dots,\mathbf{f}_t$, the conditional distribution over $(\mathbf{z}\,,\mathbf{G}_1,\dots,\mathbf{G}_t)$ satisfies:*

- $\mathbf{z}$ *is drawn uniformly at random from* $[k]^n$*;*
- *each edge ij appears (independently) in $\bigcup_{i\in[t]}\mathbf{G}_i$ with probability at most $\frac{d^*}{n}(1+(1-\frac{1}{k})\varepsilon^*)$ if $\mathbf{z}_i=\mathbf{z}_j$ and probability at least $\frac{d^*}{n}(1-\frac{\varepsilon}{k})$ otherwise, for some d^*,ε^* such that*

$$dt\cdot\frac{1+\frac{\varepsilon}{2}}{1+\left(1-\frac{1}{k}\right)\varepsilon^*}\leqslant d^*\leqslant dt\cdot\frac{1+\frac{\varepsilon^*}{1-\varepsilon^*/k}}{1+\left(1-\frac{1}{k}\right)\varepsilon^*}-o(1)\,.$$

In words, Theorem A.1 shows that the union graph $\bigcup_{i\in[t]}\mathbf{G}_i$ is essentially a k-community stochastic block model with parameters $d^*=\Theta(dt)\,,\varepsilon^*=\Theta(\varepsilon)$. As discussed in Section 1, it is conjecturally hard to achieve weak recovery in polynomial time for $d^*(\varepsilon^*/k)^2\leqslant 1$. In the context of Theorem A.1, this implies that the parameters of the distribution of $\bigcup_{i\in[t]}\mathbf{G}_i$ are above the Kesten-Stigum threshold *only* for $d^*(\varepsilon^*/k)^2\geqslant\Omega(1)$. That is, at least $t\geqslant\Omega(k^2)$ observations are required!

Next we prove the theorem.

Proof of Theorem A.1. Let $q,q'\in[k]$ be distinct. By Chernoff's bound and choice of t we have[7]

$$\mathbb{P}\left(\frac{t}{2}(1-o(1))\leqslant\sum_{\ell\in[t]}[\![\mathbf{f}_\ell(q)=\mathbf{f}_\ell(q')]\!]\leqslant\frac{t}{2}(1+o(1))\right)\geqslant 1-k^{-5}\,. \tag{10}$$

Hence we may take a union over all such pairs $q,q'\in[k]$ as the corresponding event $\mathcal{E}$ will hold with probability at least $1-k^{-O(1)}$. So let $f_1\dots,f_t:[k]\to\pm1$ be fixed functions verifying the event $\mathcal{E}$ of (10). We condition the rest of the analysis on $\mathbf{f}_1=f_1,\dots,\mathbf{f}_t=f_t$. In these settings each edge appears in $\mathbf{G}^*:=\bigcup_{i\in[t]}\mathbf{G}_i$ independently of others. Moreover, by union bound

$$\mathbb{P}\left(ij\in\mathbf{G}^*\mid\mathbf{z}_i=\mathbf{z}_j\,,\mathbf{f}_1=f_1,\dots,\mathbf{f}_t=f_t\right)\leqslant\left(1+\frac{\varepsilon}{2}\right)\frac{dt}{n}\,,$$

and

$$\begin{aligned}
&\mathbb{P}\left(ij\in\mathbf{G}^*\mid\mathbf{z}_i\neq\mathbf{z}_j\,,\mathbf{f}_1=f_1,\dots,\mathbf{f}_t=f_t\right)\\
&\quad\geqslant 1-\left(1-\left(1+\frac{\varepsilon}{2}\right)\frac{d}{n}\right)^{\frac{t}{2}(1-o(1))}\left(1-\left(1-\frac{\varepsilon}{2}\right)\frac{d}{n}\right)^{\frac{t}{2}(1+o(1))}\\
&\quad\geqslant 1-\left(1-\left(1+\frac{\varepsilon}{2}\right)\frac{dt}{2n}(1-o(1))\right)\left(1-\left(1-\frac{\varepsilon}{2}\right)\frac{dt}{2n}(1+o(1))\right)\\
&\quad\geqslant(1-o(1))\frac{dt}{n}\,,
\end{aligned}$$

where we used the inequality $(1+s)^r\geqslant 1+rs$, for $r>1\,,s>-1$. It remains to compute d^* and ε^* so that

$$\left(1+\frac{\varepsilon}{2}\right)\frac{dt}{n}\leqslant\frac{d^*}{n}\left(1+\left(1-\frac{1}{k}\right)\varepsilon^*\right)\,,$$

[7] We can take $o(1)$ to be $t^{-1/4}$ and by Hoeffding's inequality we can bound the probability of the event $\mathcal{E}$ in (10) not happening by $2\exp(-\sqrt{t})\leqslant 2\exp(-10\log k)\leqslant k^{-5}$.

$$(1-o(1))\frac{dt}{n} \geqslant \frac{d^*}{n}\left(1-\frac{\varepsilon^*}{k}\right).$$

Rearranging the inequalities,

$$d^* \geqslant dt \cdot \frac{1+\frac{\varepsilon}{2}}{1+\left(1-\frac{1}{k}\right)\varepsilon^*},$$

$$d^* \leqslant dt \cdot \frac{1}{1-\frac{\varepsilon^*}{k}} - o(1) = dt \cdot \frac{1+\frac{\varepsilon^* k}{k-\varepsilon^*}}{1+\left(1-\frac{1}{k}\right)\varepsilon^*} - o(1)$$

as desired. □

Remark A.2 (On the weighted union graph). A natural question to ask is whether the weighted union graph – the graph over $[n]$, in which edge ij has weight $\sum_{\ell\in[t]}[\![ij \in \mathbf{G}_\ell]\!]$– could provide better guarantees. In the sparse settings $d \leqslant n^{o(1)}, t \leqslant n^{o(1)}$ only a $n^{o(1)-1}$ fraction of the edges have weight larger than 1 and thus one may expect that this additional information does not simplify the problem.

B. Information theoretic lower bound for blackbox algorithms

In this section we would like to study, in the context of ($\mathcal{T}$,k,t)-MV-SBM$_n$, the information-theoretic limitations for having an algorithm that *(i)* runs the procedure in Theorem 3.1 on each observation and *(ii)* uses the resulting matrices $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ to reconstruct the original k communities. Essentially, we would like to prove a formal version of Theorem 1.3. In order to do this, we first need to introduce some useful notation and terminology.

Throughout the section let $\mathbf{z} \in [k]^n$ be the vector of communities, and let $(\mathbf{f}_\ell)_{\ell\in[t]}$ be t independent and uniformly distributed random mappings $[k] \to \{+1,-1\}$. For every $\ell \in [t]$ let $\mathbf{x}_\ell = \mathbf{f}_\ell(\mathbf{z})$ and let $\mathbf{G}_\ell \sim \mathsf{SBM}_{n,2,d_\ell,\varepsilon_\ell}(\mathbf{f}_\ell(\mathbf{z}))$. We introduce a quantitative version of weak-recovery.

Definition B.1 (α_ℓ-weak-recovery algorithm)**.** We say that an algorithm[8] $\hat{X}_\ell$ taking $\mathbf{G}_\ell$ as input and producing an estimate $\hat{X}_\ell(\mathbf{G}_\ell) \in [+1,-1]^{n\times n}$ of $\mathbf{x}_\ell\mathbf{x}_\ell^{\mathsf{T}}$ is an α_ℓ-weak-recovery algorithm if we have

$$\mathbb{E}\left[\hat{X}_\ell(\mathbf{G})_{ij} \;\middle|\; (\mathbf{x}_\ell)_i = (\mathbf{x}_\ell)_j\right] - \mathbb{E}\left[\hat{X}(\mathbf{G})_{ij} \;\middle|\; (\mathbf{x}_\ell)_i \neq (\mathbf{x}_\ell)_j\right] \geqslant \alpha_\ell\,, \quad \forall i,j \in [n]\,. \tag{11}$$

Clearly, the algorithm mentioned in Theorem 3.1 is a $C_{d,\varepsilon}$-weak-recovery algorithm.

We are interested in determining the information-theoretic limits for estimating $\mathbf{z}$ based only on the outputs of an α_ℓ-weak-recovery algorithm when applied on the observations $(\mathbf{G}_\ell)_{\ell\in[t]}$. To this end, let us introduce blackbox estimators:

Definition B.2 (Blackbox estimator)**.** A *blackbox estimator* for $\mathbf{z}$ is a mapping

$$\hat{z} : ([+1,-1]^{n\times n})^t \to [k]^n\,.$$

The blackbox estimator is applied as follows: For every $\ell \in [t]$, we first compute $\hat{\mathbf{X}}_\ell = \hat{X}_\ell(\mathbf{G}_\ell)$, for some α-weak-recovery algorithm $\hat{X}_\ell$ for $\mathbf{x}_\ell$ for which we do not know anything about except that it is an α-weak-recovery algorithm, and then compute $\hat{\mathbf{z}} = \hat{z}(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t)$.

We would like to guarantee that the blackbox estimator yields a successful weak-recovery of $\mathbf{z}$ using only the fact that $\hat{\mathbf{X}}_\ell = \hat{X}_\ell(\mathbf{G}_\ell)$ satisfies (11) for every $\ell \in [t]$. In order to formalize this, we will use the notion of α-estimates:

Definition B.3 (α-estimates)**.** Let $\alpha = (\alpha_\ell)_{\ell\in[t]} \in (0,2]^t$ be a sequence of t positive numbers. Let $\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t \in [+1,-1]^{n\times n}$ be t random matrices. We say that $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ are *α-estimates* of $(\mathbf{x}_\ell\mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$ if they satisfy the following three conditions:

(a) Given $(\mathbf{x}_\ell)_{\ell\in[t]}$, the random matrices $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ are conditionally independent from $\mathbf{z}$.

(b) For every $\ell \in [t]$, given $\mathbf{x}_\ell$, the random matrix $\hat{\mathbf{X}}_\ell$ is conditionally independent from $(\mathbf{x}_1,\dots,\mathbf{x}_{\ell-1},\mathbf{x}_{\ell+1},\dots,\mathbf{x}_t,\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_{\ell-1},\hat{\mathbf{X}}_{\ell+1},\dots,\hat{\mathbf{X}}_t)$.

[8] Note that $\hat{X}_\ell$ may be a randomized algorithm.

(c) For every $\ell \in [t]$ and every $i, j \in [n]$ with $i \neq j$ we have

$$\mathbb{E}\left[(\hat{\mathbf{X}}_\ell)_{ij} \;\middle|\; (\mathbf{x}_\ell)_i = (\mathbf{x}_\ell)_j\right] - \mathbb{E}\left[(\hat{\mathbf{X}}_\ell)_{ij} \;\middle|\; (\mathbf{x}_\ell)_i \neq (\mathbf{x}_\ell)_j\right] \geqslant \alpha_\ell\,, \quad \forall i, j \in [n]\,. \tag{12}$$

It is not hard to see that if $\hat{X}_\ell$ is an α_ℓ-weak-recovery algorithms for every $\ell \in [t]$, then $(\hat{X}(\mathbf{G}_\ell))_{\ell\in[t]}$ are α-estimates of $(\mathbf{x}_\ell \mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$.

Now we are ready to formally define what we mean by "guaranteeing that the blackbox estimator yields a successful weak-recovery of $\mathbf{z}$ using only the fact that $\hat{\mathbf{X}}_\ell = \hat{X}(\mathbf{G}_\ell)$ satisfies (11) for every $\ell \in [t]$":

Definition B.4 ((ρ, τ, α, t)-weak-recovery blackbox estimator)**.** Let $\alpha = (\alpha_\ell)_{\ell\in[t]} \in (0, 2]^t$ be a sequence of t positive numbers, and let $\rho > 0$ and $\tau > 0$.

A mapping[9]

$$\hat{z} : ([-1, +1]^{n\times n})^t \to [k]^n$$

is said to be a (ρ, τ, α, t)-*weak-recovery blackbox estimator for* $\mathbf{z}$ *from* α-*estimates of* $(\mathbf{x}_\ell \mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$ if for every t random matrices $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ which are α-*estimates* of $(\mathbf{x}_\ell \mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$, if

$$\mathbf{z} = \hat{z}(\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_t)\,,$$

then with probability at least τ we have

$$\max_{\pi\in P_k} \sum_i \frac{1}{n} [\![\mathbf{z}_i = \pi(\hat{\mathbf{z}}_i)]\!] \geqslant \frac{1}{k} + \rho\,, \tag{13}$$

where P_k is the set of permutations $[k] \to [k]$.

We are now ready to state the main theorem of the section, which implies Theorem 1.3.

Theorem B.5 (Formal statement of Theorem 1.3)**.** *Let $\alpha = (\alpha_\ell)_{\ell\in[t]} \in (0, 2]^t$ be a sequence of t positive numbers and denote*

$$\overline{\alpha} = \frac{1}{t} \sum_{\ell\in[t]} \alpha_\ell\,.$$

Let $\rho > 0$ and $\tau > 0$. Let $\mathbf{z} \in [k]^n$ be the uniformly random vector of communities, and let $(\mathbf{f}_\ell)_{\ell\in[t]}$ be t independent and uniformly distributed random mappings $[k] \to \{+1, -1\}$. For every $\ell \in [t]$ let $\mathbf{x}_\ell = \mathbf{f}_\ell(\mathbf{z})$. If there exists a (ρ, τ, α, t)-weak-recovery blackbox estimator for $\mathbf{z}$ from α-estimates of $(\mathbf{x}_\ell \mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$, and if n is large enoug, then we must have

$$t \geqslant \Omega_\rho\left(\frac{\tau \cdot \log k}{\overline{\alpha}}\right).$$

We prove Theorem B.5 in Appendix B.1, Appendix B.2, and Appendix B.3. We conclude this section showing how Algorithm 1 is indeed a blackbox estimator.

Proof that Algorithm 1 is a blackbox estimator. We can split Algorithm 1 into two steps:

(1) Computing $\hat{\mathbf{X}}_1 = \hat{\mathbf{X}}_1(\mathbf{G}_1), \ldots, \hat{\mathbf{X}}_t = \hat{\mathbf{X}}_t(\mathbf{G}_t)$ by applying the algorithm in Theorem 3.1 to $\mathbf{G}_1, \ldots, \mathbf{G}_t$, respectively.

(2) Computing an estimate $\hat{\mathbf{z}}$ of $\mathbf{z}$ based only on $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$.

In step (1), since each of $\hat{\mathbf{X}}_\ell$ is applied to $\mathbf{G}_\ell$ independently of all other graphs and since $\mathbf{G}_\ell$ depends on $\mathbf{z}$ only through $\mathbf{x}_\ell$, it is not hard to see that $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ satisfy conditions (a) and (b) of Definition B.3. Now if $\alpha_\ell = C_\ell$ is the correlation guaranteed by Theorem 3.1 for $\hat{\mathbf{X}}_\ell$, it follows that $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ are α-estimates of $(\mathbf{x}_\ell \mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$, where $\alpha = (\alpha_\ell)_{\ell\in[t]}$.

Now since step (2) of Algorithm 1 only processes $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$, and since the guarantee on the agreement of $\hat{\mathbf{z}}$ with $\mathbf{z}$ is proved based only on the fact that $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ are α-estimates, we can see that, assuming that t is a large enough multiple of $(\log k)/\overline{\alpha}^2$, step (2) is a $(1 - k^{-\Omega(1)}, 1 - k^{-\Omega(1)}, \alpha, t)$-weak-recovery blackbox estimator. □

[9] Note that $\hat{z}$ may be a randomized function.

B.1. Upper bound on the information revealed by α-estimates

The first step in our proof is to determine how much information about $\mathbf{z}$ the α-estimates can reveal. Let $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ be α-estimates of $(\mathbf{x}_\ell\mathbf{x}_\ell^\mathsf{T})_{\ell\in[t]}$. The mutual information (measured in bits) between $\mathbf{z}$ and $\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t$ can be upper bounded as follows:

$$\begin{aligned} I(\mathbf{z};\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t) &\overset{(*)}{\leqslant} I(\mathbf{x}_1,\dots,\mathbf{x}_t;\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t) \\ &= H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t) - H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t|\mathbf{x}_1,\dots,\mathbf{x}_t)\,, \end{aligned} \tag{14}$$

where $(*)$ follows from the data-processing inequality[10].

The entropy $H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t)$ can be upper bounded as follows:

$$H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t) \leqslant \sum_{\ell\in[t]} H(\hat{\mathbf{X}}_\ell)\,. \tag{15}$$

Now using the chain rule, the conditional entropy $H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t|\mathbf{x}_1,\dots,\mathbf{x}_t)$ can be rewritten as follows:

$$\begin{aligned} H(\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t|\mathbf{x}_1,\dots,\mathbf{x}_t) &= \sum_{\ell\in[t]} H(\hat{\mathbf{X}}_\ell|\mathbf{x}_1,\dots,\mathbf{x}_t,\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_{\ell-1}) \\ &= \sum_{\ell\in[t]} H(\hat{\mathbf{X}}_\ell|\mathbf{x}_\ell)\,, \end{aligned} \tag{16}$$

where the last equality follows from Property (b) of α-estimates.

Combining (14), (15) and (16) we get

$$\begin{aligned} I(\mathbf{z};\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t) &\leqslant \sum_{l\in[t]} \Big(H(\hat{\mathbf{X}}_\ell) - H(\hat{\mathbf{X}}_\ell|\mathbf{x}_\ell)\Big) \\ &= \sum_{l\in[t]} I(\mathbf{x}_\ell;\hat{\mathbf{X}}_\ell) \end{aligned} \tag{17}$$

Now for each $\ell\in[t]$, we will derive an upper bound on $I(\mathbf{x}_\ell;\hat{\mathbf{X}}_\ell)$ (which would then induce an upper bound on $I(\mathbf{z};\hat{\mathbf{X}}_1,\dots,\hat{\mathbf{X}}_t)$). Note that we cannot obtain a non-trivial upper bound on $I(\mathbf{x}_\ell;\hat{\mathbf{X}}_\ell)$ for arbitrary α-estimates because setting $\hat{\mathbf{X}}_\ell = \mathbf{x}_\ell\mathbf{x}_\ell^\mathsf{T}$ would satisfy the definition of α-estimates, and for $\hat{\mathbf{X}}_\ell = \mathbf{x}_\ell\mathbf{x}_\ell^\mathsf{T}$ we have $I(\mathbf{x}_\ell;\hat{\mathbf{X}}_\ell) = n$, which is too large for our purposes. What we will do instead is to show that there exist α-estimates for which we can get the desired upper bound on $I(\mathbf{x}_\ell;\hat{\mathbf{X}}_\ell)$.

The α-estimates that we will consider are of the form $\hat{\mathbf{X}}_\ell = \hat{\mathbf{x}}_\ell\hat{\mathbf{x}}_\ell^\mathsf{T}$ where $\hat{\mathbf{x}}_\ell\in\{+1,1\}^n$ is defined as follows:

$$\mathbb{P}\left(\hat{\mathbf{x}}_\ell = \hat{x}_\ell|\mathbf{x}_\ell = x_\ell\right) = \prod_{i\in[n]} \mathbb{P}\left((\hat{\mathbf{x}}_\ell)_i = (\hat{x}_\ell)_i|(\mathbf{x}_\ell)_i = (x_\ell)_i\right)\,,$$

where

$$\mathbb{P}\left((\hat{\mathbf{x}}_\ell)_i = (\hat{x}_\ell)_i \mid (\mathbf{x}_\ell)_i = (x_\ell)_i\right) = \begin{cases} \frac{1}{2} + \sqrt{\frac{\alpha_\ell}{8}} & \text{if } (\hat{x}_\ell)_i = (x_\ell)_i\,, \\ \frac{1}{2} - \sqrt{\frac{\alpha_\ell}{8}} & \text{if } (\hat{x}_\ell)_i = -(x_\ell)_i\,. \end{cases}$$

In other words, we obtain $\hat{\mathbf{x}}_\ell$ by sending the entries of $\mathbf{x}_\ell$ through a binary symmetric channel with flipping probability $\frac{1}{2} - \sqrt{\frac{\alpha_\ell}{8}}$.

[10] Notice that due to Property (a) of α-estimates, $\mathbf{z} - (\mathbf{x}_\ell)_{\ell\in[t]} - (\hat{\mathbf{x}}_\ell)_{\ell\in[t]}$ is a Markov chain.

Now notice that

$$\begin{aligned}\mathbb{E}\left[(\hat{\mathbf{x}}_\ell)_i \mid (\mathbf{x}_\ell)_i\right] &= (\mathbf{x}_\ell)_i \cdot \mathbb{P}\left((\hat{\mathbf{x}}_\ell)_i = (\mathbf{x}_\ell)_i \mid (\mathbf{x}_\ell)_i\right) - (\mathbf{x}_\ell)_i \cdot \mathbb{P}\left((\hat{\mathbf{x}}_\ell)_i = -(\mathbf{x}_\ell)_i \mid (\mathbf{x}_\ell)_i\right) \\ &= 2\sqrt{\frac{\alpha_\ell}{8}}(\mathbf{x}_\ell)_i = \sqrt{\frac{\alpha_\ell}{2}}(\mathbf{x}_\ell)_i\,.\end{aligned}$$

Hence, for $i \neq j$

$$\begin{aligned}\mathbb{E}\left[(\hat{\mathbf{X}}_\ell)_{i,j} \;\middle|\; (\mathbf{x}_\ell)_i, (\mathbf{x}_\ell)_j\right] &= \mathbb{E}\left[(\hat{\mathbf{x}}_\ell)_i \cdot (\hat{\mathbf{x}}_\ell)_j \mid (\mathbf{x}_\ell)_i, (\mathbf{x}_\ell)_j\right] \\ &= \mathbb{E}\left[(\hat{\mathbf{x}}_\ell)_i \mid (\mathbf{x}_\ell)_i\right]\mathbb{E}\left[(\hat{\mathbf{x}}_\ell)_j \mid (\mathbf{x}_\ell)_j\right] \\ &= \sqrt{\frac{\alpha_\ell}{2}}(\mathbf{x}_\ell)_i \cdot \sqrt{\frac{\alpha_\ell}{2}}(\mathbf{x}_\ell)_j = \frac{\alpha_\ell}{2}(\mathbf{x}_\ell)_i \cdot (\mathbf{x}_\ell)_j\,,\end{aligned}$$

from which it is not hard to see that

$$\mathbb{E}\left[(\hat{\mathbf{X}}_\ell)_{i,j} \;\middle|\; (\mathbf{x}_\ell)_i = (\mathbf{x}_\ell)_j\right] - \mathbb{E}\left[(\hat{\mathbf{X}}_\ell)_{i,j} \;\middle|\; (\mathbf{x}_\ell)_i = -(\mathbf{x}_\ell)_j\right] = \frac{\alpha_\ell}{2} - \left(-\frac{\alpha_\ell}{2}\right) = \alpha_\ell\,.$$

This proves that our choice of $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ indeed yields α-estimates. In the remainder of this subsection we will show that this particular choice of α-estimates is noisy enough to yield a useful upper bound on the mutual information $I(\mathbf{x}_\ell; \hat{\mathbf{X}}_\ell)$.

For every $\ell \in [t]$ we have

$$I(\mathbf{x}_\ell; \hat{\mathbf{X}}_\ell) = I(\mathbf{x}_\ell; \hat{\mathbf{x}}_\ell) = H(\hat{\mathbf{x}}_\ell) - H(\hat{\mathbf{x}}_\ell|\mathbf{x}_\ell) \leqslant n - H(\hat{\mathbf{x}}_\ell|\mathbf{x}_\ell)\,, \tag{18}$$

where the first equality follows from the fact that there is a one-to-one mapping between $\hat{\mathbf{x}}_\ell$ and $\hat{\mathbf{X}}_\ell = \hat{\mathbf{x}}_\ell\hat{\mathbf{x}}_\ell^{\mathsf{T}}$, and the last inequality follows from the fact that $\hat{\mathbf{x}}_\ell \in \{+1, -1\}^n$ is a binary vector of length n .

Now notice that the conditional distribution of $\hat{\mathbf{x}}_\ell$ given $\mathbf{x}_\ell$ can be seen as a sequence of n independent Bernoulli random variables with parameter $\frac{1}{2} \pm \sqrt{\frac{\alpha_\ell}{8}}$. Therefore[11],

$$H(\hat{\mathbf{x}}_\ell|\mathbf{x}_\ell) = n \cdot h_2\left(\frac{1}{2} + \sqrt{\frac{\alpha_\ell}{8}}\right)\,, \tag{19}$$

where

$$h_2(p) = -p\log_2 p - (1-p)\log_2(1-p)$$

is the binary entropy function.

Combining (18) and (19), we get

$$I(\mathbf{x}_\ell; \hat{\mathbf{X}}_\ell) \leqslant n \cdot \left(1 - h_2\left(\frac{1}{2} + \sqrt{\frac{\alpha_\ell}{8}}\right)\right)\,.$$

Now note that the function $p \mapsto h_2(p)$ is a strictly concave function achieving its maximum at $p = \frac{1}{2}$ for which we have $h_2(p) = 1$. Therefore, for small α_ℓ , we have

$$1 - h_2\left(\frac{1}{2} + \sqrt{\frac{\alpha_\ell}{8}}\right) = \frac{h_2''(1/2)}{2}\left(\sqrt{\frac{\alpha_\ell}{8}}\right)^2 \pm O\left(\sqrt{\frac{\alpha_\ell}{8}}\right)^3 \leqslant O(\alpha_\ell)\,.$$

We conclude that there exists an absolute constant $C > 0$ such that

$$I(\mathbf{x}_\ell; \hat{\mathbf{X}}_\ell) \leqslant C \cdot \alpha_\ell \cdot n\,.$$

Combining this with (17), we conclude that for some α-estimates $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ of $(\mathbf{x}\mathbf{x}_\ell^{\mathsf{T}})_{\ell\in[t]}$, we have

$$I(\mathbf{z}; \hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_t) \leqslant \sum_{\ell\in[t]} C \cdot \alpha_\ell \cdot n = C \cdot \overline{\alpha} \cdot t \cdot n\,. \tag{20}$$

[11] Notice that $h_2(p) = h_2(1-p)$ and hence $h_2\left(\frac{1}{2} + \sqrt{\frac{\alpha_\ell}{8}}\right) = h_2\left(\frac{1}{2} - \sqrt{\frac{\alpha_\ell}{8}}\right)$.

B.2. Weakly recovering z reduces its entropy

Now let $\hat{\mathbf{z}} \in [k]^n$ be an estimate of $\mathbf{z}$ which satisfies (13) with probability at least τ. We will apply a modified version of the standard Fano inequality in order to upper bound $H(\mathbf{z}|\hat{\mathbf{z}})$. Define the random variable

$$\mathbf{A} = \begin{cases} 1 & \text{if } \mathbf{z} \text{ and } \hat{\mathbf{z}} \text{ satisfy (13)}\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

We have

$$\begin{aligned} H(\mathbf{z}|\hat{\mathbf{z}}) &\leqslant H(\mathbf{A}, \mathbf{z}|\hat{\mathbf{z}}) \\ &= H(\mathbf{A}|\hat{\mathbf{z}}) + H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}) \\ &\leqslant H(\mathbf{A}) + H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=0)\mathbb{P}[\mathbf{A}=0] + H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1)\mathbb{P}[\mathbf{A}=1] \\ &\leqslant 1 + (n\log_2 k)\cdot\mathbb{P}[\mathbf{A}=0] + H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1)\cdot\mathbb{P}[\mathbf{A}=1]\,, \end{aligned}$$

where the last inequality follows from the fact that $\mathbf{A}$ is a binary random variable (and hence its entropy is at most one bit), and the fact that $\mathbf{z} \in [k]^n$, which implies that $H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=0) \leqslant \log_2(k^n) = n\log_2 k$. We conclude that

$$\begin{aligned} H(\mathbf{z}|\hat{\mathbf{z}}) &\leqslant 1 + n\log_2 k - (n\log_2 k - H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1))\cdot\mathbb{P}[\mathbf{A}=1] \\ &\leqslant 1 + n\log_2 k - \tau\cdot(n\log_2 k - H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1))\,, \end{aligned}$$

where the last inequality follows from the fact that $\mathbb{P}[\mathbf{A}=1] \geqslant \tau$ and the fact that $n\log_2 k - H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1) \geqslant 0$ (because $\mathbf{z} \in [k]^n$). Now since $\mathbf{z}$ is a uniform random variable in $[k]^n$, we have $H(\mathbf{z}) = \log_2(k^n) = n\log_2 k$, and hence

$$\begin{aligned} I(\mathbf{z};\hat{\mathbf{z}}) &= H(\mathbf{z}) - H(\mathbf{z}|\hat{\mathbf{z}}) \\ &\geqslant \tau\cdot(n\log_2 k - H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1)) - 1\,. \end{aligned} \tag{21}$$

Now we will focus on upper bounding $H(\mathbf{z}|\hat{\mathbf{z}}, \mathbf{A}=1)$. For every $\hat{z} \in [k]^n$, we have

$$H(\mathbf{z}|\hat{\mathbf{z}} = \hat{z}, \mathbf{A}=1) \leqslant \log_2|Z(\hat{z}, \rho)|\,, \tag{22}$$

where

$$Z(\hat{z}, \rho) = \left\{ z \in [k]^n : \max_{\pi\in P_k} \sum_i \frac{1}{n}[\![z_i = \pi(\hat{z}_i)]\!] \geqslant \frac{1}{k} + \rho \right\} = \bigcup_{\pi\in P_k} Z(\hat{z}, \rho, \pi)\,,$$

and

$$Z(\hat{z}, \rho, \pi) = \left\{ z \in [k]^n : \sum_i \frac{1}{n}[\![z_i = \pi(\hat{z}_i)]\!] \geqslant \frac{1}{k} + \rho \right\}.$$

Hence,

$$|Z(\hat{z}, \rho)| \leqslant \sum_{\pi\in P_k} |Z(\hat{z}, \rho, \pi)|\,.$$

We will further divide $Z(\hat{z}, \rho, \pi)$ as follows:

$$Z(\hat{z}, \rho, \pi) = \bigcup_{\beta n \leqslant m \leqslant n} Z(\hat{z}, \rho, \pi, m)\,,$$

where

$$\beta = \frac{1}{k} + \rho\,,$$

and

$$Z(\hat{z}, \rho, \pi, m) = \left\{ z \in [k]^n : \sum_i [\![z_i = \pi(\hat{z}_i)]\!] = m \right\}.$$

It is not hard to see that

$$|Z(\hat{z},\rho,\pi,m)| = \binom{n}{m}\cdot(k-1)^{n-m}\,.$$

By defining $\beta_m = \frac{m}{n}$ and using Stirling's formula[12], we get:

$$\begin{aligned}\log_2|Z(\hat{z},\rho,\pi,m)| &= n\log_2 n - n\log_2 e\\ &\quad - m\log_2 m + m\log_2 e - (n-m)\log_2(n-m) + (n-m)\log_2 e\\ &\quad \pm O(\log n) + (n-m)\log_2(k-1)\\ &= n\log_2 n - \beta_m n\log_2(\beta_m n) - (1-\beta_m)n\log_2((1-\beta_m)n)\\ &\quad + (1-\beta_m)n\log_2(k-1) \pm O(\log n)\\ &= (h_2(\beta_m) + (1-\beta_m)\cdot\log_2(k-1))\cdot n \pm O(\log n)\,.\end{aligned}$$

By taking derivatives and analyzing the function $g(\beta_m) = h_2(\beta_m) + (1-\beta_m)\cdot\log_2(k-1)$, we can show that g is decreasing after $\beta_m \geqslant \frac{1}{k}$, and hence for $\beta_m \geqslant \beta = \frac{1}{k}+\rho \geqslant \frac{1}{k}$, we have $g(\beta_m) \leqslant g(\beta)$. In particular, for every m satisfying $\beta n \leqslant m \leqslant n$, we have

$$\log_2|Z(\hat{z},\rho,\pi,m)| \leqslant (h_2(\beta) + (1-\beta)\cdot\log_2(k-1))\cdot n + O(\log n)\,.$$

Therefore,

$$\begin{aligned}|Z(\hat{z},\rho)| &\leqslant \sum_{\pi\in P_k}\sum_{\beta n\leqslant m\leqslant n}|Z(\hat{z},\rho,\pi,m)|\\ &\leqslant \sum_{\pi\in P_k}\sum_{\beta n\leqslant m\leqslant n} 2^{(h_2(\beta)+(1-\beta)\cdot\log_2(k-1))\cdot n + O(\log n)}\\ &= k!\cdot n\cdot 2^{(h_2(\beta)+(1-\beta)\cdot\log_2(k-1))\cdot n + O(\log n)}\,,\end{aligned}$$

and hence

$$\log_2|Z(\hat{z},\rho)| \leqslant (h_2(\beta) + (1-\beta)\cdot\log_2(k-1))\cdot n + O(k\log k + \log n)\,.$$

Since this is true for every $\hat{z}\in[k]^n$, we get from (22) that

$$H(\mathbf{z}|\hat{\mathbf{z}},\mathbf{A}=1) \leqslant (h_2(\beta) + (1-\beta)\cdot\log_2(k-1))\cdot n + O(k\log k + \log n)\,.$$

Combining this with (21), we get

$$\begin{aligned}I(\mathbf{z};\hat{\mathbf{z}}) &\geqslant \tau\left(n\log_2 k - (h_2(\beta) + (1-\beta)\cdot\log_2(k-1))\,n - O(\log n + k\log k)\right) - 1\\ &\geqslant \frac{\tau\cdot n}{2}\left(\log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1)\right)\,,\end{aligned} \tag{23}$$

where the last inequality assumes[13] that n is large enough (and in particular $n \gg k\log k$).

B.3. Putting everything together

Proof of Theorem B.5. Assume that there is a (ρ,τ,α,t)-weak-recovery blackbox estimator $\hat{z}$ for $\mathbf{z}$ and assume that n is large enough. Let $(\hat{\mathbf{X}}_\ell)_{\ell\in[t]}$ be α-estimates of $(\mathbf{x}_\ell)_{\ell\in[t]}$ satisfying (20), i.e.,

$$I(\mathbf{z};\hat{\mathbf{X}}_1,\ldots,\hat{\mathbf{X}}_t) \leqslant C\cdot\overline{\alpha}\cdot t\cdot n\,.$$

Let $\hat{\mathbf{z}} = \hat{z}(\hat{\mathbf{X}}_1,\ldots,\hat{\mathbf{X}}_\ell)$. From the data-processing inequality, we have

$$I(\mathbf{z};\hat{\mathbf{z}}) \leqslant I(\mathbf{z};\hat{\mathbf{X}}_1,\ldots,\hat{\mathbf{X}}_t) \leqslant C\cdot\overline{\alpha}\cdot t\cdot n\,.$$

[12] For large n, we have $\log_2(n!) = n\log_2 n - n\log_2 e \pm O(\log_2 n)$.

[13] It is worth noting that if $\beta > 1/k$, then $\log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1) > 0$, as we will show in the next subsection.

On the other hand, since $\hat{z}$ is a (ρ, τ, α, t)-weak-recovery blackbox estimator and since $(\hat{\mathbf{x}}_\ell)_{\ell\in[t]}$ are α-estimates of $(\mathbf{x}_\ell)_{\ell\in[t]}$, it follows that $\hat{\mathbf{z}}$ satisfies (13) with probability $1-\tau$. It follows from (23) that for n large enough, we have

$$I(\mathbf{z};\hat{\mathbf{z}}) \geqslant \frac{\tau\cdot n}{2}\cdot(\log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1))\,.$$

We conclude that

$$\frac{\tau\cdot n}{2}\cdot(\log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1))\cdot n \leqslant C\cdot\overline{\alpha}\cdot t\cdot n\,.$$

Therefore, we must have

$$t \geqslant \tau\cdot\frac{\log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1)}{2C\cdot\overline{\alpha}}\,.$$

Now define

$$\begin{aligned} l(\beta) &= \log_2 k - h_2(\beta) - (1-\beta)\cdot\log_2(k-1)\\ &= \log_2 k + \beta\log_2\beta + (1-\beta)\log_2(1-\beta) - (1-\beta)\cdot\log_2(k-1)\,, \end{aligned}$$

so that

$$t \geqslant \frac{\tau\cdot l(\beta)}{2C\cdot\overline{\alpha}} = \frac{\tau\cdot l(1/k+\rho)}{2C\cdot\overline{\alpha}}\,.$$

Let us analyze the function $l(\beta)$:

- A quick calculation shows that
$$l(1/k) = 0\,.$$
- The derivative of l is
$$\begin{aligned} l'(\beta) &= \log_2\beta + \frac{1}{\ln 2} - \log_2(1-\beta) - \frac{1}{\ln 2} + \log_2(k-1)\\ &= \log_2\beta - \log_2(1-\beta) + \log_2(k-1)\,, \end{aligned}$$
and hence
$$l'(1/k) = 0\,.$$
- The second derivative of l is
$$l''(\beta) = \frac{1}{\beta\ln 2} + \frac{1}{(1-\beta)\ln 2} > 0\,,\quad \forall\beta\in(0,1)\,.$$

Hence, $l'(\beta) > 0$ for $\beta\in(1/k,1)$, and since $l(1/k)=0$ we can see that $l(1/k+\rho)>0$ whenever $\rho>0$. Furthermore, a quick calculation reveals that for fixed ρ we have[14]

$$\lim_{k\to\infty}\frac{l(1/k+\rho)}{\log_2(k)} = \rho > 0\,,$$

which means that

$$\min_{k\geqslant 2}\frac{l(1/k+\rho)}{\log_2(k)} > 0\,,$$

and hence $l(1/k+\rho) = \Omega_\rho(\log_2(k))$. We conclude that

$$t \geqslant \Omega_\rho\left(\frac{\tau\cdot\log k}{\overline{\alpha}}\right)\,.$$

□

[14] Note that $\beta\log_2\beta + (1-\beta)\log_2(1-\beta) = -h_2(\beta)$ and since $0\leqslant h_2(\beta)\leqslant 1$, we can see that $\lim_{k\to\infty}\frac{h_2(\beta)}{\log_2(k)} = 0$. Hence $\lim_{k\to\infty}\frac{l(1/k+\rho)}{\log_2(k)}$ can be simplified as $\lim_{k\to\infty} 1-(1-\rho-\frac{1}{k})\frac{\log_2(k-1)}{\log_2(k)} = \rho$.

C. Deferred proofs

We present here proofs deferred in the main body of the paper.

Deferred proofs of Section 3 .

To obtain Theorem 3.1 we need to introduce results about robust weak recovery.

Definition C.1 (μ-node corrupted, balanced 2 communities SBM)**.** Let $\mu \in [0,1]$. Let $x \in \{\pm 1\}^n$ be a vector satisfying $\sum_i x_i = 0$ and let $\mathbf{G}^0 \sim \mathsf{SBM}_{2,d,\varepsilon}(x)$. An adversary may choose up to $\mu \cdot n$ vertices in $\mathbf{G}^0$ and arbitrarily modify edges (and non-edges) incident to at least one of them to produce the corrupted graph G. We write $G \overset{\mu}{\approx} \mathbf{G}^0$ to denote that G is a μ-node corrupted version of $\mathbf{G}^0$.

In the context of node corrupted graphs, the definition of weak recovery is still with respect to the original vector x as defined in Equation (1). It is known that node robust weak recovery is achievable.

Theorem C.2 (Implicit in (Ding et al., 2023))**.** *Let $n, d, \varepsilon > 0$ be satisfying $d \cdot \varepsilon^2 - 1 =: \delta > 0$. There exist:*

- *constants $0 < \mu_\delta < 1$ and $0 < C_\delta < 1$, and*
- *a (randomized) polynomial time algorithm[15] $\hat{\mathbf{X}}_{\mathrm{r}}$ taking a graph G of n vertices as input and producing a matrix $\hat{\mathbf{X}}_{\mathrm{r}}(G) \in [-1,+1]^{n \times n}$ as output,*

such that $\hat{\mathbf{X}}_{\mathrm{r}}$ is a successful weak-recovery recovery algorithm robust against any μ-node corruption for all $\mu \leqslant \mu_\delta$. More formally, for every $x \in \{\pm 1\}^n$ satisfying $\sum_i x_i = 0$, and every $\mu \leqslant \mu_\delta$, we have

$$\mathop{\mathbb{E}}_{\mathbf{G}^0 \sim \mathsf{SBM}_{2,d,\varepsilon}(x)} \left[\min_{G : G \overset{\mu}{\approx} \mathbf{G}^0} \langle \hat{\mathbf{X}}_{\mathrm{r}}(G), x x^{\mathsf{T}} \rangle \right] \geqslant C_\delta \cdot n^2 \,.$$

We can use Theorem C.2 to obtain Theorem 3.1.

Let $\gamma = \left| p - \frac{1}{2} \right|$ be the unbalancedness in the vector of labels of $\mathbf{x}$, where $p = \mathbb{P}(\mathbf{x}_i = +1)$.

The main idea behind the algorithm in Theorem 3.1 is to first distinguish whether the unbalancedness γ is sufficiently small or not. If it is sufficiently small, then we apply the robust algorithm of Theorem C.2. Otherwise, we can achieve weak-recovery by relying on the degree of a vertex to estimate its community label.

In the following two lemmas, we treat the case where the unbalancedness γ is sufficiently small:

Lemma C.3. *Let $n, d, \varepsilon, p, \mathbf{x} = (\mathbf{x}_i)_{i \in [n]}$ and $\mathbf{G} \sim \mathsf{SBM}_{n,2,d,\varepsilon}(\mathbf{x})$ be as in Theorem 3.1 and let $\delta = \frac{\varepsilon^2 d}{4} - 1 > 0$. Let μ_δ, C_δ and $\hat{\mathbf{X}}_{\mathrm{r}}$ be[16] as in Theorem C.2 and define*

$$\mu'_\delta = \frac{1}{100} \min\{\mu_\delta, C_\delta\} \,.$$

If the unbalancedness $\gamma = \left| \frac{1}{2} - p \right|$ of $\mathbf{x}$ satisfies $\gamma \leqslant \mu'_\delta$, then for n large enough, we have

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle \right] \geqslant \frac{3}{4} C_\delta \cdot n^2 \,.$$

Proof. For the sake of simplicity, we will only treat the case where n is even. Since $\hat{\mathbf{X}}_{\mathrm{r}}$ is robust against node corruptions, it is not hard to see that the proofs can be adapted to the case where n is odd.

[15] The subscript r in $\hat{\mathbf{X}}_{\mathrm{r}}$ stands for "robust".

[16] It is worth noting that since Theorem C.2 assumes that $\sum_i x_i = 0$, then n must be even in Theorem C.2. However, in Lemma C.3 we would like n to be general. Hence, if n is odd, we apply the following procedure: (1) we add a fictitious vertex $n+1$ which is not incident to any vertex in $[n]$ and we call the resulting graph (having $[n+1]$ as its set of vertices) as $\tilde{\mathbf{G}}$, (2) we apply $\hat{\mathbf{X}}_{\mathrm{r}}$ on $\tilde{\mathbf{G}}$, and (3) we take the submatrix of $\hat{\mathbf{X}}_{\mathrm{r}}(\tilde{\mathbf{G}})$ induced by the vertices in $[n]$. We still denote the overall algorithm as $\hat{\mathbf{X}}_{\mathrm{r}}$.

For every $x \in \{\pm\}^n$, let $n_+(x) = |\{i \in [n] : x_i = +1\}|$ and $n_-(x) = |\{i \in [n] : x_i = -1\}|$. Since $\mathbb{P}(\mathbf{x}_i = +1) = p$, then by the law of large numbers we know that $n_+(\mathbf{x})$ and $n_-(\mathbf{x})$ concentrate around pn and $(1-p)n$, respectively. Furthermore, since $\gamma = \left|\frac{1}{2} - p\right| \leqslant \mu'_\delta$, we can use standard concentration inequalities to show that with probability at least $1 - O(n^{-10})$, the random vector $\mathbf{x}$ satisfies the event

$$\mathcal{E} = \left\{ x \in \{\pm\}^n : \left| \frac{n_+(x)}{n} - \frac{1}{2} \right| \leqslant 2\mu'_\delta \text{ and } \left| \frac{n_-(x)}{n} - \frac{1}{2} \right| \leqslant 2\mu'_\delta \right\} .$$

Now since $\mathbb{P}(\mathbf{x} \in \mathcal{E}) = 1 - O(n^{-10})$ and since $\left|\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle\right| \leqslant n^2$, it is not hard to see that

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle\right] = \mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle \;\middle|\; \mathbf{x} \in \mathcal{E}\right] \pm o(1) , \tag{24}$$

so we can focus on studying $\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle \;\middle|\; \mathbf{x} \in \mathcal{E}\right]$.

Now fix $x \in \mathcal{E}$ and condition on the event that $\mathbf{x} = x$. From the definition of $\mathcal{E}$ it is not hard to see that there is $x' \in \{\pm\}^n$ satisfying $\sum_{i\in[n]} x'_i = 0$ and

$$|D_{x,x'}| \leqslant 4\mu'_\delta n ,$$

where

$$D_{x,x'} = |i \in [n] : x_i \neq x'_i| .$$

Now construct a random graph $\mathbf{G}'$ as follows:

- If $i, j \in [n] \setminus D_{x,x'}$, i.e., if $x_i = x'_i$ and $x_j = x'_j$, then we let $\{i, j\} \in \mathbf{G}'$ if and only if $\{i, j\} \in \mathbf{G}$.
- If either $i \in D_{x,x'}$ or $j \in D_{x,x'}$ then we put the edge $\{i, j\}$ in $\mathbf{G}'$ with probability $\frac{d}{n}\left(1 + \frac{\varepsilon}{2} x'_i \cdot x'_j\right)$.
- The events $(\{i, j\} \in \mathbf{G}')_{i,j\in[n]}$ are mutually independent.

It is not hard to see that:

- $\mathbf{G}' \sim \mathsf{SBM}_{n,2,d,\varepsilon}(x')$, and
- $\mathbf{G} \overset{4\mu'_\delta}{\approx} \mathbf{G}'$, i.e., $\mathbf{G}$ can be obtained from $\mathbf{G}'$ by adding or removing edges incident to at most $4\mu'_\delta n$ vertices.

Since $4\mu'_\delta \leqslant \frac{4}{100}\mu_\delta \leqslant \mu_\delta$, it follows from Theorem C.2 that

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), x'x'^{\mathsf{T}} \rangle \;\middle|\; \mathbf{x} = x\right] \geqslant C_\delta \cdot n^2 . \tag{25}$$

Now notice that

$$\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), xx^{\mathsf{T}} \rangle = \langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), x'x'^{\mathsf{T}} \rangle + \langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), xx^{\mathsf{T}} - x'x'^{\mathsf{T}} \rangle ,$$

and

$$\begin{aligned}\left|\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), xx^{\mathsf{T}} - x'x'^{\mathsf{T}} \rangle\right| &\leqslant \left\|\hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G})\right\|_\infty \cdot \left\|xx^{\mathsf{T}} - x'x'^{\mathsf{T}}\right\|_1 \leqslant 1 \cdot \left\|xx^{\mathsf{T}} - x'x'^{\mathsf{T}}\right\|_1 \\ &\leqslant 2\left\|x - x'\right\|_1 \cdot n = 4|D_{x,x'}| \cdot n \leqslant 16\mu'_\delta \cdot n^2 .\end{aligned}$$

Combining this with (25), we get

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), xx^{\mathsf{T}} \rangle \;\middle|\; \mathbf{x} = x\right] \geqslant (C_\delta - 16\mu'_\delta) \cdot n^2 \geqslant \frac{84}{100} C_\delta \cdot n^2 .$$

Now since this is true for all $x \in \mathcal{E}$, we conclude that

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}}\rangle \;\middle|\; \mathbf{x} \in \mathcal{E}\right] \geqslant (C_\delta - 16\mu'_\delta)\cdot n^2 \geqslant \frac{84}{100} C_\delta \cdot n^2\,,$$

where the last inequality is true because $\mu'_\delta = \frac{1}{100}\min\{\mu_\delta, C_\delta\}$. Combining the above with (24), we can deduce that for n large enough, we have

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}}\rangle\right] \geqslant \frac{3}{4} C_\delta \cdot n^2\,.$$

□

The following lemma takes the algorithm of Lemma C.3 and applies a symmetrization argument in order to get "a positive correlation at the edge level".

Lemma C.4 (Pair-wise weak recovery for sufficiently balanced 2 communities stochastic block mode)**.** *Let $n, d, \varepsilon, p, \mathbf{x} = (\mathbf{x}_i)_{i\in[n]}$ and $\mathbf{G} \sim \mathsf{SBM}_{n,2,d,\varepsilon}(\mathbf{x})$ be as in Theorem 3.1. Let $\delta = \frac{\varepsilon^2 d}{4} - 1 > 0$ and let $\gamma = \left|\frac{1}{2} - p\right|$ be the unbalancedness of $\mathbf{x}$. There exist constants $\mu'_\delta > 0$ and $C'_\delta > 0$ and a randomized polynomial-time algorithm[17] $\hat{\mathbf{X}}_{\mathrm{sb}}$ taking $\mathbf{G}$ as input and producing a matrix $\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G}) \in [-1,+1]^{n\times n}$ such that if*

$$\gamma \leqslant \mu'_\delta\,,$$

then for every $i, j \in [n]$ with $i \neq j$, we have

$$C'_\delta \leqslant \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right]\,.$$

Proof. Let μ_δ, C_δ and $\hat{\mathbf{X}}_{\mathrm{r}}$ be as in Theorem C.2, and let $\mu'_\delta = C'_\delta = \frac{1}{100}\min\{\mu_\delta, C_\delta\}$. Lemma C.3 shows that

$$\mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}}\rangle\right] \geqslant \frac{3}{4} C_\delta \cdot n^2\,.$$

The algorithm $\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})$ can be obtained by "symmetrizing" the algorithm $\hat{\mathbf{X}}_{\mathrm{r}}$ as follows: Let $\boldsymbol{\sigma}$ be a (uniformly) random permutation $[n] \to [n]$ and let $\mathbf{G}_{\boldsymbol{\sigma}}$ be the graph obtained from $\mathbf{G}$ by $\boldsymbol{\sigma}$-permuting its vertices, i.e., we let the edge[18] $\{\boldsymbol{\sigma}_i, \boldsymbol{\sigma}_j\}$ belong to $\mathbf{G}_{\boldsymbol{\sigma}}$ if and only if $\{i, j\} \in \mathbf{G}$. We define the matrix $\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G}) \in [-1,+1]^n$ as follows:

$$\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} = \hat{\mathbf{X}}_r(\mathbf{G}_{\boldsymbol{\sigma}})_{\boldsymbol{\sigma}_i\boldsymbol{\sigma}_j}\,.$$

In other words, we apply the random permutation $\boldsymbol{\sigma}$ to graph $\mathbf{G}$, we apply the algorithm $\hat{\mathbf{X}}_{\mathrm{r}}$, and then we apply the inverse of the permutation on the resulting matrix.

Due to the symmetry of the SBM distribution, it is not hard to see that for every $i, j \in [n]$ with $i \neq j$, we have

$$\begin{aligned}
\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij}\cdot \mathbf{x}_i\mathbf{x}_j\right] &= \sum_{\substack{i',j'\in[n]:\\ i'\neq j'}} \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij}\cdot \mathbf{x}_i\mathbf{x}_j \;\middle|\; \boldsymbol{\sigma}_i = i', \boldsymbol{\sigma}_j = j'\right]\cdot \mathbb{P}\left(\boldsymbol{\sigma}_i = i', \boldsymbol{\sigma}_j = j'\right)\\
&= \sum_{\substack{i',j'\in[n]:\\ i'\neq j'}} \mathbb{E}\left[\hat{\mathbf{X}}_r(\mathbf{G}_{\boldsymbol{\sigma}})_{\boldsymbol{\sigma}_i\boldsymbol{\sigma}_j}\cdot \mathbf{x}_i\mathbf{x}_j \;\middle|\; \boldsymbol{\sigma}_i = i', \boldsymbol{\sigma}_j = j'\right]\cdot \frac{1}{n(n-1)}\\
&\overset{(*)}{=} \frac{1}{n(n-1)} \sum_{\substack{i',j'\in[n]:\\ i'\neq j'}} \mathbb{E}\left[\hat{\mathbf{X}}_r(\mathbf{G})_{\boldsymbol{\sigma}_i\boldsymbol{\sigma}_j}\cdot \mathbf{x}_{\boldsymbol{\sigma}_i}\mathbf{x}_{\boldsymbol{\sigma}_j} \;\middle|\; \boldsymbol{\sigma}_i = i', \boldsymbol{\sigma}_j = j'\right]
\end{aligned}$$

[17] The subscript sb in $\hat{\mathbf{X}}_{\mathrm{sb}}$ stands for "sufficiently balanced".

[18] For simplicity, We denote $\boldsymbol{\sigma}(i)$ as $\boldsymbol{\sigma}_i$.

$$
\begin{aligned}
&= \frac{1}{n(n-1)} \sum_{\substack{i',j' \in [n]: \\ i' \neq j'}} \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G})_{i'j'} \cdot \mathbf{x}_{i'}\mathbf{x}_{j'}\right] \\
&= \frac{1}{n(n-1)} \mathbb{E}\left[\langle \hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G}), \mathbf{x}\mathbf{x}^{\mathsf{T}} \rangle\right] - \frac{1}{n(n-1)} \sum_{i \in [n]} \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{r}}(\mathbf{G})_{ii} \cdot \mathbf{x}_i^2\right] \\
&\geqslant \frac{1}{n(n-1)} \cdot \frac{3}{4} C_\delta \cdot n^2 - \frac{1}{n-1} \\
&\geqslant \frac{3}{4} C_\delta - o(1)\,,
\end{aligned}
$$

where $(*)$ follows from the symmetry of the SBM distribution under the simultaneous permutation of vertices and labels.

Now notice that

$$
\begin{aligned}
\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \cdot \mathbf{x}_i\mathbf{x}_j\right] &= \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] \cdot \mathbb{P}\left(\mathbf{x}_i = \mathbf{x}_j\right) - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] \cdot \mathbb{P}\left(\mathbf{x}_i \neq \mathbf{x}_j\right) \\
&= \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] \cdot \left(p^2 + (1-p)^2\right) - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] \cdot (2p(1-p)) \\
&= \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] \cdot \left(\frac{1}{2} + 2\gamma^2\right) - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] \left(\frac{1}{2} - 2\gamma^2\right) \\
&\leqslant \frac{1}{2} \cdot \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] + 2\gamma^2 - \frac{1}{2} \cdot \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] + 2\gamma^2\,,
\end{aligned}
$$

where in the last inequality we used the fact that $\left|\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij}\right| \leqslant 1$. We can deduce that for n large enough, we have

$$
\begin{aligned}
&\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] \\
&\qquad \geqslant 2\,\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \cdot \mathbf{x}_i\mathbf{x}_j\right] - 8\gamma^2 \geqslant \frac{3}{2} C_\delta - o(1) - 8\gamma^2 \geqslant C_\delta\,,
\end{aligned}
$$

where the last inequality follows from the fact that

$$
8\gamma^2 \leqslant 8\mu_\delta'^2 \leqslant 8\left(\frac{1}{100} C_\delta\right)^2 \leqslant \frac{C_\delta}{100}\,.
$$

By picking $C'_\delta = C_\delta$, the lemma follows. □

Now we turn to show that if $\mathbf{x}$ is sufficiently unbalanced, then there exists an efficient algorithm that achieves pair-wise weak recovery.

Lemma C.5 (Pair-wise weak recovery for sufficiently unbalanced 2 communities stochastic block mode)**.** *Let $n, d, \varepsilon, p, \mathbf{x} = (\mathbf{x}_i)_{i \in [n]}$ and $\mathbf{G} \sim \mathsf{SBM}_{n,2,d,\varepsilon}(\mathbf{x})$ be as in Theorem 3.1. Further assume that $p \in [0,1]$. Let $\delta = \frac{\varepsilon^2 d}{4} - 1 > 0$ and let $\gamma = \left|\frac{1}{2} - p\right|$ be the unbalancedness of $\mathbf{x}$, and let μ'_δ be as in Lemma C.4. There exists a constant $C''_\delta > 0$ and a randomized polynomial-time algorithm [19] $\hat{\mathbf{X}}_{\mathrm{su}}$ taking $\mathbf{G}$ as input and producing a matrix $\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G}) \in [-1,+1]^{n \times n}$ such that if*

$$
\gamma \geqslant \frac{1}{2}\mu'_\delta\,,
$$

then for every $i, j \in [n]$ with $i \neq j$, we have

$$
C''_\delta \leqslant \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right]\,.
$$

Proof. For the sake of simplicity we may assume without loss of generality that $p > \frac{1}{2}$, i.e., $p = \frac{1}{2} + \gamma$ and hence "+1" is the larger community in expectation.

[19] The subscript su in $\hat{\mathbf{X}}_{\mathrm{su}}$ stands for "sufficiently unbalanced".

For every $i, j \in [n]$, let

$$\mathbf{deg}_{\neq j}(i) = |\{v \in [n] \setminus \{i,j\} : \{i,v\} \in \mathbf{G}\}|$$

be the number of vertices in $[n] \setminus \{i,j\}$ which are adjacent to i in $\mathbf{G}$.

Let $C > 0$ be a large enough constant (to be chosen later) and let

$$\hat{x}_i^{(j)}(\mathbf{G}) = \frac{1}{C}\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right) \cdot [\![\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| \leqslant C]\!] \in [-1,1]\,,$$

and define the matrix $\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G}) \in [-1,+1]^{n\times n}$ as:

$$\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} = \hat{x}_i^{(j)}(\mathbf{G}) \cdot \hat{x}_j^{(i)}(\mathbf{G})\,.$$

It is not hard to see that given $(\mathbf{x}_i, \mathbf{x}_j)$, the random variables $\mathbf{deg}_{\neq j}(i)$ and $\mathbf{deg}_{\neq i}(j)$ are conditionally independent. Therefore, $\hat{x}_i^{(j)}(\mathbf{G})$ and $\hat{x}_j^{(i)}(\mathbf{G})$ are conditionally independent given $(\mathbf{x}_i, \mathbf{x}_j)$, hence

$$\begin{aligned}\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i, \mathbf{x}_j\right] &= \mathbb{E}\left[\hat{x}_i^{(j)}(\mathbf{G}) \;\middle|\; \mathbf{x}_i, \mathbf{x}_j\right] \cdot \mathbb{E}\left[\hat{x}_j^{(i)}(\mathbf{G}) \;\middle|\; \mathbf{x}_i, \mathbf{x}_j\right] \\ &= \mathbb{E}\left[\hat{x}_i^{(j)}(\mathbf{G}) \;\middle|\; \mathbf{x}_i\right] \cdot \mathbb{E}\left[\hat{x}_j^{(i)}(\mathbf{G}) \;\middle|\; \mathbf{x}_j\right]\,.\end{aligned}$$

Now, for every $v \in [v] \setminus \{i,j\}$, we have

$$\begin{aligned}\mathbb{P}\left(\{i,v\} \in \mathbf{G} \mid \mathbf{x}_i\right) &= \mathbb{E}\left[\mathbb{P}\left(\{i,v\} \in \mathbf{G} \mid \mathbf{x}_i, \mathbf{x}_v\right) \mid \mathbf{x}_i\right] = \mathbb{E}\left[\frac{d}{n}\left(1 + \frac{1}{2}\varepsilon \cdot \mathbf{x}_i\mathbf{x}_v\right) \;\middle|\; \mathbf{x}_i\right] \\ &= \frac{d}{n}\left(1 + \frac{1}{2}\varepsilon \cdot \mathbf{x}_i\,\mathbb{E}[\mathbf{x}_v]\right) = \frac{d}{n}\left(1 + \frac{1}{2}\varepsilon \cdot (p - (1-p))\right) \\ &= \frac{d}{n}\left(1 + \varepsilon\gamma\mathbf{x}_i\right)\,.\end{aligned}$$

Therefore, the conditional distribution of $\mathbf{deg}_{\neq j}(i)$ given $\mathbf{x}_i$ is $\mathrm{Binomial}\left(n-2, \frac{d}{n}\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)$, hence

$$\mathbb{E}\left[\mathbf{deg}_{\neq j}(i) \;\middle|\; \mathbf{x}_i\right] = (n-2)\cdot\frac{d}{n}\left(1+\varepsilon\gamma\mathbf{x}_i\right)\,,$$

and so

$$\mathbb{E}\left[\frac{1}{C}\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right) \;\middle|\; \mathbf{x}_i\right] = \frac{d(1-2/n)\cdot\varepsilon\gamma\mathbf{x}_i}{C}\,. \tag{26}$$

On the other hand, by the Cauchy-Schwarz inequality, we have

$$\begin{aligned}&\mathbb{E}\left[\left|\hat{x}_i^{(j)}(\mathbf{G}) - \frac{1}{C}\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)\right| \;\middle|\; \mathbf{x}_i\right] \\ &\leqslant \mathbb{E}\left[\frac{1}{C}\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| \cdot [\![\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| > C]\!] \;\middle|\; \mathbf{x}_i\right] \\ &\leqslant \frac{1}{C}\,\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \;\middle|\; \mathbf{x}_i\right]^{1/2} \cdot \mathbb{E}\left[[\![\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| > C]\!]^2 \;\middle|\; \mathbf{x}_i\right]^{1/2} \\ &\leqslant \frac{1}{C}\,\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \;\middle|\; \mathbf{x}_i\right]^{1/2} \cdot \mathbb{P}\left(\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| > C \mid \mathbf{x}_i\right)^{1/2}\,,\end{aligned}$$

and by Chebychev's inequality, we have

$$\mathbb{P}\left(\left|\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right| > C \;\middle|\; \mathbf{x}_i\right) \leqslant \frac{\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \;\middle|\; \mathbf{x}_i\right]}{C^2}\,,$$

hence

$$\mathbb{E}\left[\left|\hat{x}_i^{(j)}(\mathbf{G}) - \frac{1}{C}\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)\right| \,\middle|\, \mathbf{x}_i\right] \leqslant \frac{\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \,\middle|\, \mathbf{x}_i\right]}{C^2}\,. \tag{27}$$

Now notice that

$$\begin{aligned}
\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \,\middle|\, \mathbf{x}_i\right] &= \mathbb{E}\left[\left(\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\left(1+\varepsilon\gamma\mathbf{x}_i\right) + \left(\varepsilon\gamma\mathbf{x}_i\right)d(1-2/n)\right)\right)^2 \,\middle|\, \mathbf{x}_i\right] \\
&\overset{(*)}{\leqslant} \mathbb{E}\left[2\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)^2 + 2\left(\left(\varepsilon\gamma\mathbf{x}_i\right)d(1-2/n)\right)^2 \,\middle|\, \mathbf{x}_i\right] \\
&\leqslant 2\,\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)^2 \,\middle|\, \mathbf{x}_i\right] + 2d^2\varepsilon^2\gamma^2\,,
\end{aligned}$$

where $(*)$ is true because $(a+b)^2 \leqslant 2a^2 + 2b^2$ for all $a, b \in \mathbb{R}$.

Now since the conditional distribution of $\mathbf{deg}_{\neq j}(i)$ given $\mathbf{x}_i$ is $\mathrm{Binomial}\left(n-2, \frac{d}{n}\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)$, its conditional expectation is equal to $d(1-2/n)\left(1+\varepsilon\gamma\mathbf{x}_i\right)$ and its conditional variance is equal to

$$\begin{aligned}
\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)^2 \,\middle|\, \mathbf{x}_i\right] &= (n-2)\left(\frac{d}{n}\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right)\cdot\left(1 - \frac{d}{n}\left(1+\varepsilon\gamma\mathbf{x}_i\right)\right) \\
&\leqslant d(1+\varepsilon\gamma) \leqslant 2d\,.
\end{aligned}$$

Therefore,

$$\mathbb{E}\left[\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)^2 \,\middle|\, \mathbf{x}_i\right] \leqslant 4d + 2d^2\varepsilon^2\gamma^2\,,$$

and hence from (27) we get

$$\mathbb{E}\left[\left|\hat{x}_i^{(j)}(\mathbf{G}) - \frac{1}{C}\left(\mathbf{deg}_{\neq j}(i) - d(1-2/n)\right)\right| \,\middle|\, \mathbf{x}_i\right] \leqslant \frac{4d + 2d^2\varepsilon^2\gamma^2}{C^2}\,.$$

Combining this with (26) we get

$$\begin{aligned}
\mathbb{E}\left[\hat{x}_i^{(j)}(\mathbf{G}) \,\middle|\, \mathbf{x}_i\right] &= \frac{d(1-2/n)\cdot\varepsilon\gamma\mathbf{x}_i}{C} \pm O\left(\frac{d + d^2\varepsilon^2\gamma^2}{C^2}\right) \\
&= \frac{d(1-2/n)}{C}\left(\varepsilon\gamma\mathbf{x}_i \pm O\left(\frac{1 + d\varepsilon^2\gamma^2}{C}\right)\right)\,.
\end{aligned}$$

Let

$$C = \tilde{C}(1-2/n)\left(d\varepsilon + \frac{1}{\varepsilon\mu'_\delta}\right)\,,$$

for some large enough constant $\tilde{C} \geqslant 1$ to be chosen later. We have

$$O\left(\frac{1}{C}\right) \leqslant O\left(\frac{1}{\tilde{C}(1-2/n)/(\varepsilon\mu'_\delta)}\right) = O\left(\frac{\varepsilon\mu'_\delta}{\tilde{C}}\right) \leqslant O\left(\frac{\varepsilon\gamma}{\tilde{C}}\right)\,,$$

where the last inequality follows from $\gamma \geqslant \frac{\mu'_\delta}{2}$. Furthermore,

$$O\left(\frac{d\varepsilon^2\gamma^2}{C}\right) \leqslant O\left(\frac{d\varepsilon^2\gamma^2}{\tilde{C}(1-2/n)\cdot d\varepsilon}\right) \leqslant O\left(\frac{\varepsilon\gamma^2}{\tilde{C}}\right) \leqslant O\left(\frac{\varepsilon\gamma}{\tilde{C}}\right)\,.$$

We conclude that

$$\mathbb{E}\left[\hat{x}_i^{(j)}(\mathbf{G}) \,\middle|\, \mathbf{x}_i\right] = \frac{d(1-2/n)}{C}\left(\varepsilon\gamma\mathbf{x}_i \pm O\left(\frac{\varepsilon\gamma}{\tilde{C}}\right)\right) = \frac{d(1-2/n)\varepsilon\gamma}{C}\left(\mathbf{x}_i \pm O\left(\frac{1}{\tilde{C}}\right)\right)$$

$$= \frac{d\varepsilon\gamma}{\tilde{C}\left(d\varepsilon + \frac{1}{\varepsilon\mu'_\delta}\right)}\left(\mathbf{x}_i \pm O\left(\frac{1}{\tilde{C}}\right)\right) = C''_\delta\gamma\left(\mathbf{x}_i \pm O\left(\frac{1}{\tilde{C}}\right)\right),$$

where

$$C''_\delta = \frac{d\varepsilon^2}{\tilde{C}\left(d\varepsilon^2 + \frac{1}{\mu'_\delta}\right)} = \frac{4(1+\delta)}{\tilde{C}\left(4(1+\delta) + \frac{1}{\mu'_\delta}\right)} = \frac{4(1+\delta)\mu'_\delta}{\tilde{C}\left(4(1+\delta)\mu'_\delta + 1\right)}.$$

Notice how C''_δ depends only on δ.

Finally,

$$\begin{aligned}\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i, \mathbf{x}_j\right] &= \mathbb{E}\left[\hat{x}_i^{(j)}(\mathbf{G}) \;\middle|\; \mathbf{x}_i\right] \cdot \mathbb{E}\left[\hat{x}_j^{(i)}(\mathbf{G}) \;\middle|\; \mathbf{x}_j\right] \\ &= C''_\delta\gamma\left(\mathbf{x}_i \pm O\left(\frac{1}{\tilde{C}}\right)\right) \cdot C''_\delta\gamma\left(\mathbf{x}_j \pm O\left(\frac{1}{\tilde{C}}\right)\right) \\ &= (C''_\delta\gamma)^2\left(\mathbf{x}_i\mathbf{x}_j \pm O\left(\frac{1}{\tilde{C}}\right)\right),\end{aligned}$$

and so

$$\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] = (C''_\delta\gamma)^2\left(2 \pm O\left(\frac{1}{\tilde{C}}\right)\right).$$

By choosing $\tilde{C}$ to be an absolute constant which is large enough, we get

$$\mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i = \mathbf{x}_j\right] - \mathbb{E}\left[\hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G})_{ij} \;\middle|\; \mathbf{x}_i \neq \mathbf{x}_j\right] \geqslant (C''_\delta\gamma)^2 \geqslant \frac{(C''_\delta\mu'_\delta)^2}{4},$$

where the last inequality is true because we assume that $\gamma \geqslant \frac{\mu'_\delta}{2}$. □

Now we can leverage Lemma C.4 and Lemma C.5 in order to prove Theorem 3.1.

Proof of Theorem 3.1. Let $\gamma, \mu'_\delta, C'_\delta, C''_\delta$ be as in Lemma C.4 and Lemma C.5.

We will first distinguish between the sufficiently balanced and sufficiently unbalanced cases by counting the number of edges which are incident to a random sublinear (but sufficiently high) set of vertices. Let $m = \lceil n^{3/4} \rceil$ and let $\mathbf{I}$ be a random subset of $[n]$ of size m.

Let $\deg(\mathbf{I})$ be the number of edges in $\mathbf{G}$ from $\mathbf{I}$ to $[n] \setminus \mathbf{I}$. We have

$$\begin{aligned}\mathbb{E}\left[\deg(\mathbf{I}) \mid \mathbf{I}\right] &= \sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}} \mathbb{E}\left[[\![\{i,j\} \in \mathbf{G}]\!]\right] = \sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}} \mathbb{E}\left[\mathbb{P}\left(\{i,j\} \in \mathbf{G} \mid \mathbf{x}_i, \mathbf{x}_j\right)\right] \\ &= \sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}} \mathbb{E}\left[\frac{d}{n}\left(1 + \frac{1}{2}\varepsilon\mathbf{x}_i\mathbf{x}_j\right)\right] = \frac{d}{n}\sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}}\left(1 + \frac{1}{2}\varepsilon\,\mathbb{E}\left[\mathbf{x}_i\mathbf{x}_j\right]\right) \\ &= \frac{d}{n}\sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}}\left(1 + \frac{1}{2}\varepsilon\left(\mathbb{P}\left(\mathbf{x}_i = \mathbf{x}_j\right) - \mathbb{P}\left(\mathbf{x}_i \neq \mathbf{x}_j\right)\right)\right) \\ &= \frac{d}{n}\sum_{i\in\mathbf{I}, j\in[n]\setminus\mathbf{I}}\left(1 + \frac{1}{2}\varepsilon\left(p^2 + (1-p)^2 - 2p(1-p)\right)\right) \\ &= \frac{dm(n-m)}{n}\left(1 + \frac{1}{2}\varepsilon\left(2p-1\right)^2\right) \\ &= \frac{dm(n-m)}{n}\left(1 + 2\varepsilon\gamma^2\right) = (1 \pm o(1))dn^{3/4}\left(1 + 2\varepsilon\gamma^2\right).\end{aligned}$$

So the algorithm $\hat{\mathbf{X}}$ is defined as follows:

- If $\deg(\mathbf{I}) \geqslant dn^{3/4}\left(1+2\varepsilon\left(\frac{3\mu'_\delta}{4}\right)^2\right)$ we apply the algorithm $\hat{\mathbf{X}}_{\mathrm{su}}$ on the subgraph $\mathbf{G}([n]\setminus\mathbf{I})$ of $\mathbf{G}$ induced on $n\setminus\mathbf{I}$ and define

$$\hat{\mathbf{X}}(\mathbf{G})_{ij} = \begin{cases} \hat{\mathbf{X}}_{\mathrm{su}}(\mathbf{G}([n]\setminus\mathbf{I}))_{ij} & \text{if } i,j\in[n]\setminus\mathbf{I}\,,\\ 0 & \text{otherwise.}\end{cases}$$

- If $\deg(\mathbf{I}) < dn^{3/4}\left(1+2\varepsilon\left(\frac{3\mu'_\delta}{4}\right)^2\right)$ we apply the algorithm $\hat{\mathbf{X}}_{\mathrm{sb}}$ on the subgraph $\mathbf{G}([n]\setminus\mathbf{I})$ of $\mathbf{G}$ induced on $n\setminus\mathbf{I}$ and define

$$\hat{\mathbf{X}}(\mathbf{G})_{ij} = \begin{cases} \hat{\mathbf{X}}_{\mathrm{sb}}(\mathbf{G}([n]\setminus\mathbf{I}))_{ij} & \text{if } i,j\in[n]\setminus\mathbf{I}\,,\\ 0 & \text{otherwise.}\end{cases}$$

By standard concentration inequalities, we can show that:

- If $\gamma < \frac{\mu'_\delta}{2}$, then with probability $1-o(1)$ we have $\deg(\mathbf{I}) < dn^{3/4}\left(1+2\varepsilon\left(\frac{3\mu'_\delta}{4}\right)^2\right)$ and so we apply the algorithm $\hat{\mathbf{X}}_{\mathrm{sb}}$ which will succeed in achieving pair-wise weak recovery according to Lemma C.4, assuming $i,j\in[I]$.
- If $\gamma \geqslant \frac{\mu'_\delta}{2}$, then with probability $1-o(1)$ we have $\deg(\mathbf{I}) \geqslant dn^{3/4}\left(1+2\varepsilon\left(\frac{3\mu'_\delta}{4}\right)^2\right)$ and so we apply the algorithm $\hat{\mathbf{X}}_{\mathrm{ub}}$ which will succeed in achieving pair-wise weak recovery according to Lemma C.5, assuming $i,j\in[I]$.
- If $\frac{\mu'_\delta}{2} < \gamma < \mu'_\delta$, then it follows from Lemma C.4 and Lemma C.5 that it does not matter which algorithm we apply because both of them achieve pair-wise weak recovery, assuming $i,j\in[I]$.

Now for any $i,j\in[n]$, the probability of picking any of them in $\mathbf{I}$ is vanishingly small. We conclude that the algorithm $\hat{\mathbf{X}}$ satisfies the guarantees sought in Theorem 3.1.

□

Deferred proof of Section 4 First we prove Fact 4.3.

Proof of Fact 4.3. Let $p\in[k]$ be fixed. By Chernoff's bound, with probability at least $1-n^{20}$, $\left(1-\sqrt{\frac{400k\log n}{n}}\right)\frac{n}{k} \leqslant \|c_p(\mathbf{z})\|^2 \leqslant \left(1+\sqrt{\frac{400k\log n}{n}}\right)\frac{n}{k}$, hence the property follows by a union bound. □

Next we prove Fact 4.5.

Proof. For each $\ell\in[t]$, define

$$\mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij}\;\middle|\;\mathbf{f}_\ell(\mathbf{z})_i=\mathbf{f}_\ell(\mathbf{z})_j\right] =: C^*_\ell$$

and let

$$C^* = \sum_{\ell\in[t]} C_\ell\,.$$

By independence of the observations, the inequality of Fact 4.5 for the case case $\mathbf{z}_i=\mathbf{z}_j$ follows with an application of Hoeffding's inequality.

The case $\mathbf{z}_i\neq\mathbf{z}_j$ needs a bit more work. By Theorem 3.1, for each $\ell\in[t]$, we have

$$\mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij}\;\middle|\;\mathbf{f}_\ell(\mathbf{z})_i=\mathbf{f}_\ell(\mathbf{z})_j\right] = C^*_\ell\,,$$

and

$$\mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \;\middle|\; \mathbf{f}_\ell(\mathbf{z})_i \neq \mathbf{f}_\ell(\mathbf{z})_j\right] \leqslant C_\ell^* - C_\ell\,.$$

Therefore,

$$\begin{aligned}\mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \;\middle|\; \mathbf{z}_i \neq \mathbf{z}_j\right] =& \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \;\middle|\; \mathbf{f}_\ell(\mathbf{z})_i = \mathbf{f}_\ell(\mathbf{z})_j\right] \cdot \mathbb{P}\left(\mathbf{f}_\ell(\mathbf{z})_i = \mathbf{f}_\ell(\mathbf{z})_j \mid \mathbf{z}_i \neq \mathbf{z}_j\right)\\ &+ \mathbb{E}\left[\hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \;\middle|\; \mathbf{f}_\ell(\mathbf{z})_i \neq \mathbf{f}_\ell(\mathbf{z})_j\right] \cdot \mathbb{P}\left(\mathbf{f}_\ell(\mathbf{z})_i \neq \mathbf{f}_\ell(\mathbf{z})_j \mid \mathbf{z}_i \neq \mathbf{z}_j\right)\\ \leqslant& C_\ell^* - C_\ell \cdot \frac{1}{2}\,,\end{aligned}$$

and so

$$\mathbb{E}\left[\sum_{\ell\in[t]} \hat{\mathbf{X}}(\mathbf{G}_\ell)_{ij} \;\middle|\; \mathbf{z}_i \neq \mathbf{z}_j\right] \leqslant \sum_{\ell\in[t]}\left(C_\ell^* - C_\ell \cdot \frac{1}{2}\right) = C^* - \frac{\overline{C}\cdot t}{2}\,.$$

The inequality of Fact 4.5 for case $\mathbf{z}_i \neq \mathbf{z}_j$ follows with another application of Hoeffding's inequality. □

Now we prove Lemma 4.8.

Proof of Lemma 4.8. Let A_i be a (p,q)-representative and A_j be a (p',q)-representative, for $p\,, p' \in [k]$. It suffices to show that if $p = p'$ then $\|A_i - A_j\|^2 \leqslant n/k$ and otherwise $\|A_i - A_j\|^2 > n/k$. Then no q-representative index remains unassigned at the end of step 1. By the reverse triangle inequality

$$\begin{aligned}&\left|\|A_i - A_j\| - \|c_p(z) - c_{p'}(z)\|\right|\\ &\leqslant \|A_i - A_j - c_p(z) + c_{p'}(z)\|\\ &\leqslant \|A_i - c_p(z)\| + \|A_j - c_{p'}(z)\|\\ &\leqslant 2n\cdot e^{-q\cdot\overline{C}^2\cdot t}\,.\end{aligned}$$

For $p = p'$ we have $\|c_p(z) - c_{p'}(z)\| = 0$ and by choice of t, the first inequality follows. For $p \neq p'$ we have $\|c_p(z) - c_{p'}(z)\| \geqslant \frac{n}{k}(2 - o(1))$ since z is balanced and the second inequality follows again by choice of t. □