Federated Myopic Community Detection with One-shot Communication

Chuyang Ke
Department of Computer Science
Purdue University
cke@purdue.edu

Jean Honorio
Department of Computer Science
Purdue University
jhonorio@purdue.edu

Abstract

In this paper, we study the problem of recovering the community structure of a network under federated myopic learning. Under this paradigm, we have several clients, each of them having a myopic view, i.e., observing a small subgraph of the network. Each client sends a censored evidence graph to a central server. We provide an efficient algorithm, which computes a consensus signed weighted graph from clients evidence, and recovers the underlying network structure in the central server. We analyze the topological structure conditions of the network, as well as the signal and noise levels of the clients that allow for recovery of the network structure. Our analysis shows that exact recovery is possible and can be achieved in polynomial time. In addition, our experiments show that in an extremely sparse network with 10 000 nodes, our method can achieve exact recovery of the community structure even if every client has access to only 20 nodes. We also provide information-theoretic limits for the central server to recover the network structure from any single client evidence. Finally, as a byproduct of our analysis, we provide a novel Cheeger-type inequality for general signed weighted graphs.

1 INTRODUCTION

Modern social networks have underlying community structures (Javed et al., 2018). Take Twitter as an example: one may assume that Twitter users with similar political views are more likely to interact with each other. *Community detection* is the task of identifying underlying group structures in a network given observation of node interactions (Kelley et al., 2012; Ke and Honorio, 2018). At the same time, modern social networks exhibit two opposite defining properties: big volume, small neighborhoods. In the Twitter example, more than 1.3 billion Twitter accounts have been created as of the year 2019, yet the average number of followers is 707 (Smith, 2020). Thus when studying modern social networks, it is impractical and inefficient to collect all data. Furthermore, due to growing privacy concerns and regulations, it is unfavorable for a central server to collect all users' information.

As an emerging technique in the machine learning community, *federated learning* tries to address the issues discussed above. The idea of federated learning is not limited to any particular learning algorithm; rather, it is a learning paradigm, under which a central server trains a high-quality learning model with the coordination of a federation of participating clients (Konečnỳ et al., 2016). In a typical federated learning setting, it is often assumed that the participating clients are large in number. At the same time, each client has a very small and non-i.i.d. (independent and identically distributed) dataset, very limited computational power, and very restricted communication capabilities.

In this paper we study the problem of community detection under a federated learning paradigm. We focus on a myopic setting, in which every client has a limited and non-identical access to the network. Each client observes a small subgraph of the whole network, and sends a censored evidence graph to a central server. The server then computes a consensus graph from the evidence sent by the clients, and recovers the underlying community structure of the full network. To model community interaction we adopt a generative approach similar to the one in the stochastic block model (SBM). In other words, we assume nodes from the same group are more likely to be connected than those from differ-

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s).

ent groups. We try to answer the following questions:

- Does there exist an efficient central server algorithm that takes the censored clients' local observation as the input, and recovers the underlying community structure of the *full* network?
- Under what topological and statistical conditions, the central algorithm will work correctly?

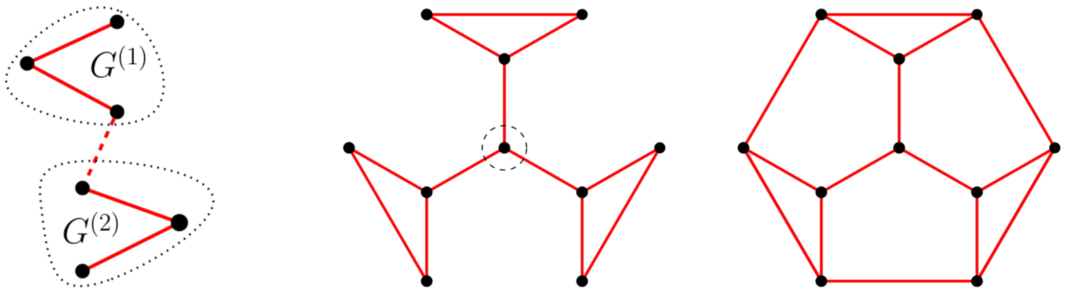
In community detection tasks, one of the most important properties is the edge expansion. In general, for undirected unweighted graphs, the edge expansion measures how connected every component of the network is. It is well-known that the Cheeger constant can be used to measure the edge expansion property of a graph. The edge expansion is important for successful community detection. For example, if the subgraphs observed by all clients do not have any intersection at all, arguably it is not possible to correctly recover the full community structure, as there is no observed community interaction between nodes in different subgraphs (see Figure 1a for illustration). Similarly, if the union of the subgraphs is a chain graph or a star graph, it has a "bottleneck". In this case, removing very few edges will disconnect the graph (see Figure 1b and 1c for illustration). We find that the correctness of community detection heavily depends on the bottleneck edges. In our analysis we propose a novel Cheeger-type constant, which characterizes the edge expansion property of general signed weighted graphs (that is, weighted graphs with potentially negative weights). We show that the Cheeger-type constant of the server consensus graph is critical for correct recovery of the underlying community structure, along with the signal and noise level parameters.

As a rapidly growing research area, federated learning was formally defined in Konečnỳ et al. (2015, 2016) as a general machine learning setting. Recently, both Li et al. (2020) and Yang et al. (2019) provided an overview of federated learning, and interestingly, both overviews highlighted that one main challenge in federated learning is the statistical heterogeneity of clients' local data. The framework of federated learning has been applied to various problems such as deep networks (McMahan et al., 2017), principal component analysis (Grammenos et al., 2020) and sparse linear regression (Barik and Honorio, 2020), to mention a few. On the other hand, federated learning for community detection tasks has not been studied yet. Mercado et al. (2018) studied a tangentially related problem called multilayer graph clustering, however the topological structure in that case is much simpler, and the weights are assumed to be nonnegative. To the best of our knowledge, we are providing the first community detection model under a federated myopic learning setting, where the clients observe a non-identical small subgraph of the full network. Many federated learning algorithms only have some convergence guarantee, i.e., the output gets close to the true solution. Our federated community detection algorithm, arguably more challenging, aims at exact recovery of the underlying community structure. In other words, our algorithm does not require rounding of the solution. Our theorem states that if the statistical conditions are met, the proposed algorithm will always return the exact true community structure.

Related Works. There are several different generalizations of signed graphs and the Laplacian (Bronski and DeVille, 2014). One variation studied in Kunegis et al. (2010); Cucuringu et al. (2019); Knyazev (2017) is the so-called signed degree matrix. The idea is that instead of summing all edge weights directly, they sum over the absolute value of the edge weights, and their signed Laplacian matrix can be defined in a similar way. Readers should be aware that this is different from our approach, and more importantly, only our approach fulfills the Karush–Kuhn–Tucker (KKT) conditions required by our algorithm. There are also some works on the application of signed networks, for example, Doreian and Mrvar (2009) proposed a heuristic algorithm to partition signed social networks, and Giotis and Guruswami (2006) proposed an approximation algorithm to maximize agreements in a graph. Our theorems focus on the spectral analysis of the signed weighted graph, and provide provable theoretical guarantees for efficient exact recovery.

Summary of Our Contribution. Our work is mostly theoretical. We provide a series of novel results in this paper:

- We introduce a highly-general federated myopic learning framework with one-shot communication for community detection tasks. Under this paradigm, every client observes a small subgraph of the full network and sends a censored evidence graph to a central server. The central server computes a consensus signed weighted graph, and recovers the underlying network structure.
- We provide provable theoretical guarantees for efficient exact recovery of the community structure. We analyze the topological structure conditions of the consensus graph, as well as the signal and noise levels of the clients that allow for recovery of the community structure. We establish the regime in which exact recovery is possible and can be achieved in polynomial time. We also provide information-theoretic limits for any algorithm to recover the local community structure from any single client evidence.
- We propose a novel Cheeger-type inequality for



(a) Chain graph (smallest edge expansion)

(b) Star graph (median edge expansion)

(c) Regular graph (largest edge expansion)

Figure 1: Graph topology examples. Figure 1a shows a chain graph, where two clients observe subgraph $G^{(1)}$ and $G^{(2)}$, respectively. In this case it is impossible for any algorithm to recover the full network structure, since there is no observed interaction between the two subgraphs. Figure 1b shows a star graph, in which every edge connecting the highlighted node in the center is a "bottleneck," i.e., removing any of the three edges disconnects the graph. Figure 1c shows a 3-regular graph, in which every component is connected with no "bottleneck."

general signed weighted graphs with potentially negative weights, from a graph theoretical point of view. The inequality relates the eigenvalue gap of a signed weighted graph to a Cheeger-type constant.

- Our method is computationally efficient on large scale networks. In particular, our experiments show that in an extremely sparse network with 10 000 nodes, our method can achieve exact recovery of the community strucutre even if every client has access to only 20 nodes.

2 PRELIMINARIES

In this section, we present an overview of federated myopic learning. We formally define the community detection task under this paradigm, and present an central server algorithm that solves the problem by using the censored subgraph information reported by the clients. We also introduce the notations that will be used later in the paper.

Without specification we use lowercase letters (e.g., a, b, u, v) for scalars and vectors, and uppercase letters (e.g., A, B, C) for matrices and sets. For any natural number n, we use $[n]$ to denote the set $\{1, \ldots, n\}$. For clarity when dealing with a sequence of objects, we use the superscript (i) to denote the i-th object in the sequence, and subscript j to denote the j-th entry. For example, for a sequence of vectors $\{x^{(i)}\}_{i \in [n]}$, $x_2^{(1)}$ represents the second entry of vector $x^{(1)}$. We use $\mathbb{R}$ to denote the set of real numbers. We use $\mathbf{1}$ to denote the all-one vector. For any matrix A, we use $\lambda_m(A)$ to denote its mth-smallest eigenvalue. For any vector u, we use $\operatorname{diag}(u)$ to denote the diagonal matrix with u in the diagonal. For every graph mentioned in this paper, without further discussion we always assume there exists no self loops.

2.1 Federated Myopic Community Detection Paradigm

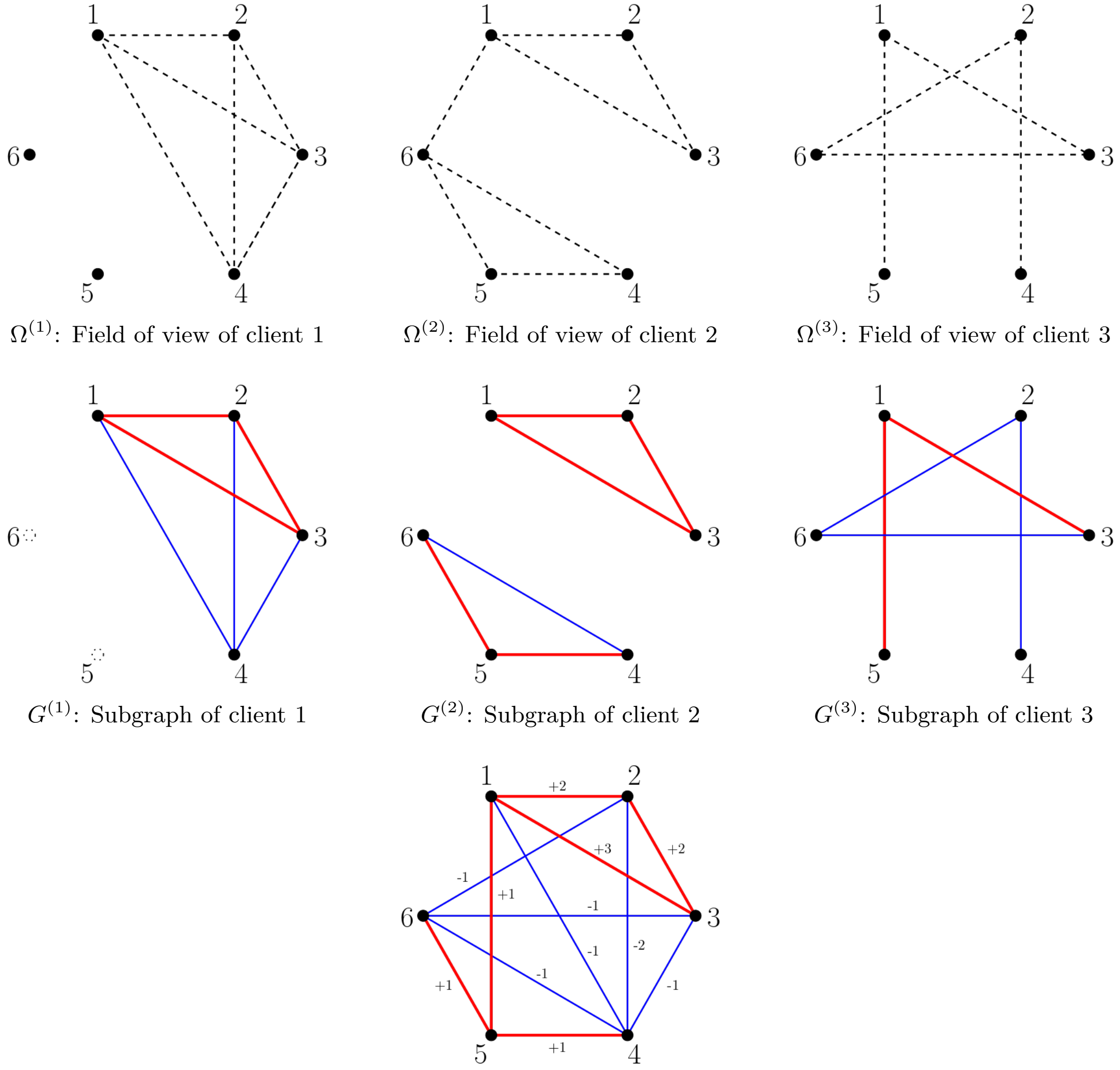
A federated myopic learning model $\mathcal{M}(n, K, \{\Omega^{(k)}, p^{(k)}, q^{(k)}, r^{(k)}\}_{k=1}^{K} \mid y^*)$ consists of n nodes and K clients, and every client is equipped with a tuple $\{\Omega^{(k)}, p^{(k)}, q^{(k)}, r^{(k)}\}$. In this model $y^* \in \{+1, -1\}^n$ is the node label vector of the network, indicating the underlying community structure. We use $V = [n]$ to denote the set of nodes.

For every single client $k \in [K]$, $\Omega^{(k)} \subset |V| \times |V|$ is the *field of view (FOV)* of client k, $p^{(k)}, q^{(k)}$ are the local *signal and noise level* parameters, and $r^{(k)}$ is the local *censorship* parameter. We say client k is *myopic*, if its field of view is not equal to the complete graph spanned by V, i.e., $\Omega^{(k)} \neq |V| \times |V|$.

Now, nature generates a local subgraph $G^{(k)} = (V^{(k)}, E^{(k)})$ for every client k using the following rule: for every viewable pair $(i, j) \in \Omega^{(k)}$ with $i \neq j$, connect (i, j) with probability $p^{(k)}$ if the labels are equal, i.e., $y_i^* = y_j^*$; otherwise connect (i, j) with probability $q^{(k)}$. After that, remove all isolated nodes from $V^{(k)}$, and denote $n^{(k)} := \left|V^{(k)}\right|$. We call $G^{(k)} = (V^{(k)}, E^{(k)})$ the local subgraph observed by client k.

Next, client k constructs the corresponding *censored evidence graph* $\tilde{G}^{(k)} = (V^{(k)}, \tilde{E}^{(k)})$ from the observed subgraph $G^{(k)}$ as follows: the edge set $\tilde{E}^{(k)}$ starts empty. For every viewable pair $(i, j) \in \Omega^{(k)}$ with $i \neq j$, if (i, j) is an edge in $E^{(k)}$, add (i, j) to $\tilde{E}^{(k)}$ with probability $1 - r^{(k)}$; if (i, j) is not an edge in $E^{(k)}$, add it to $\tilde{E}^{(k)}$ with probability $r^{(k)}$. Each client sends its censored evidence graph to a central server.

We now summarize the federated myopic learning task in this paper. It is worth mentioning, that one can only recover the node label vector y^* up to permutation of



$\Omega^{(1)}$: Field of view of client 1

$\Omega^{(2)}$: Field of view of client 2

$\Omega^{(3)}$: Field of view of client 3

$G^{(1)}$: Subgraph of client 1

$G^{(2)}$: Subgraph of client 2

$G^{(3)}$: Subgraph of client 3

G: Consensus signed weighted graph

Figure 2: Examples of client fields of view ($\Omega^{(1)}, \Omega^{(2)}, \Omega^{(3)}$), client-observed subgraphs ($G^{(1)}, G^{(2)}, G^{(3)}$), and server consensus signed weighted graph. In FOVs, each viewable edge is represented by a dashed line. In subgraphs, each edge is colored red, and each non-edge is colored blue. The consensus graph is the weighted summation of all subgraphs, such that each edge is counted as $+1$, and each non-edge is counted as -1. In the particular examples above, it is assumed that clients do not censor evidence graphs, i.e., $\tilde{G}^{(k)} = G^{(k)}$, for simplicity of visualization. In the rest of the paper, we assume that clients censor evidence graphs.

the groups without prior knowledge. We define the community structure matrix $Y^* := y^* {y^*}^\top \in \{+1, -1\}^{n \times n}$. For any pair of nodes i and j, if they are in the same community then $Y^*_{ij} = 1$; if they are not in the same community then $Y^*_{ij} = -1$. Note that the recovery of Y^* is equivalent to the recovery of y^*, up to permutation of the groups.

Definition 1 (Federated Myopic Community Detection)**.** **Unknown**: Community structure matrix $Y^* = y^* {y^*}^\top$ indicating the underlying network community structure. **Observation**: Censored evidence graphs $\{\tilde{G}^{(k)}\}_{k=1}^K$ sent by the clients, which are generated from the local subgraphs $\{G^{(k)}\}_{k=1}^K$ observed by the clients; fields of view $\{\Omega^{(k)}\}_{k=1}^K$ of the clients. **Problem**: Recover the hidden community structure matrix Y^* from the censored evidence graphs $\{\tilde{G}^{(k)}\}_{k=1}^K$.

3 SIGNED WEIGHTED CHEEGER-TYPE INEQUALITY

In this section, we provide a novel Cheeger-type inequality, which relates the spectral gap of a signed weighted graph Laplacian to the *signed weighted edge expansion* of the graph. Readers should be aware that the graph results in this section is general, and not limited to our model.

Assume $G = (V, E, w)$ with $V = [n]$ is a general weighted graph, and $w : E \to \mathbb{R}$ is a general weight function. We use W to denote the corresponding weighted adjacency matrix. For any edge $(i, j) \in E$, we have $W_{ij} := w(i, j)$; otherwise $W_{ij} := 0$. We define the signed weight matrix W^+, W^- as follows: for every entry (i, j), $W^+_{ij} = \max(W_{ij}, 0)$, and $W^-_{ij} = \min(W_{ij}, 0)$. We now introduce the essential graph definitions.

Definition 2 (Boundary of a Set)**.** For any set $S \subset V$, denote its boundary as $\partial S = \{(i, j) \mid i \in S, j \notin S\}$.

Definition 3 (Boundary Weight)**.** For any set $S \subset V$, let $w^+(\partial S)$ and $w^-(\partial S)$ denote its signed boundary weights, and let $w(\partial S)$ denote its boundary weight, formally defined as $w^+(\partial S) = \sum_{i\in S, j\notin S} W^+_{ij}$, $w^-(\partial S) = \sum_{i\in S, j\notin S} W^-_{ij}$, $w(\partial S) = \sum_{i\in S, j\notin S} W_{ij}$.

Definition 4 (Node Degree and Set Degree)**.** For any node $i \in V$, let $d^+(i)$ and $d^-(i)$ denote its signed node degree, and let $d(i)$ denote its node degree, formally defined as $d^+(i) = \sum_{j\neq i} W^+_{ij}$, $d^-(i) = \sum_{j\neq i} W^-_{ij}$, $d(i) = \sum_{j\neq i} W_{ij}$. Similarly for any set $S \subset V$, let $d^+(S)$ and $d^-(S)$ denote its signed set degree, and let $d(S)$ denote its set degree, formally defined as $d^+(S) = \sum_{i\in S} d^+(i)$, $d^-(S) = \sum_{i\in S} d^-(i)$, $d(S) = \sum_{i\in S} d(i)$.

We use the shorthand notation $d^+_{\min} := \min_{i\in V} d^+(i)$ to denote the minimum positive node degree. We can similarly define the maximum degree $d_{\max} := \max_{i\in V} d(i)$, and the minimum degree $d_{\min} := \min_{i\in V} d(i)$. We now define the signed weighted edge expansion, an important Cheeger-type constant that will be used in our analysis.

Definition 5 (Signed Weighted Edge Expansion)**.** Given a graph $G = (V, E, w)$, for any non-empty set $S \subset V$, let ϕ^+_S and ϕ^-_S denote the signed weighted edge expansion of S, and let ϕ_G denote the signed weighted edge expansion of graph G, formally defined as $\phi^+_S = \frac{w^+(\partial S)}{d^+(S)}$, $\phi^-_S = w^-(\partial S)$, $\phi_G = \frac{1}{2} d^+_{\min} \min_{S\subset V, d^+(S)\leq d^+(V)/2} (\phi^+_S)^2 + 2 \min_{S\subset V} \phi^-_S$.

We use $L := D - W$ to denote the graph Laplacian G, where $D := \operatorname{diag}(d(1), \dots, d(n))$ is the degree matrix. Note that $\mathbf{1}$ is always an eigenvector of any graph Laplacian L, with the corresponding eigenvalue being 0. We use $\lambda_{\mathbf{1}^\perp}(L)$ to denote the smallest eigenvalue of L, with the corresponding eigenvector being orthogonal to $\mathbf{1}$. As a side note, for unweighted graphs and positively weighted graphs, $\lambda_{\mathbf{1}^\perp}(L)$ is exactly equivalent to $\lambda_2(L)$, the second smallest eigenvalue of the graph Laplacian. This quantity is also called the algebraic connectivity of graph G. We now present our Cheeger-type inequality for signed weighted graphs.

Theorem 1 (Cheeger-type Inequality for Signed Weighted Graphs)**.** *For any general signed weighted graph $G = (V, E, w)$ with graph Laplacian L, we have*

$$\lambda_{\mathbf{1}^\perp}(L) \geq \phi_G .$$

4 FEDERATED MYOPIC EXACT RECOVERY

In this section, we present a central server algorithm, which recovers the hidden community structure of a network by computing a *consensus graph* and solving a semidefinite program (SDP).

Algorithm 1 Central Server Community Detection

```
Input: Evidence graphs $\{\tilde{G}^{(k)}\}_{k=1}^K$, client fields of view $\{\Omega^{(k)}\}_{k=1}^K$
Output: Estimated community structure matrix $\hat{Y}$
Initialize G as a weighted complete graph spanned by $V$
Set all edge weights in $G$ to 0
for $k \in [K]$ do
    for $(i, j) \in \Omega^{(k)}$ do
        if $(i, j) \in \tilde{G}^{(k)}$ then
            $w(i, j) \leftarrow w(i, j) + 1$
        else
            $w(i, j) \leftarrow w(i, j) - 1$
        end if
    end for
end for
$W \leftarrow$ weighted adjacency matrix of $G$
Solve the following semidefinite program
```

$$\hat{Y} = \underset{Y}{\arg\max} \quad \langle W, Y \rangle \qquad \text{subject to} \quad Y_{ii} = 1 \,, Y \succeq 0 \,. \tag{1}$$

Algorithm 1 computes a consensus graph $G = (V, E, w)$, where $w : E \to \mathbb{R}$ is a weight function for edges. For every client k and every pair (i, j) in the field of view of client k, if it is an edge in the evidence graph, the server increments the corresponding weight $w(i, j)$ in the consensus graph by 1; otherwise, the server decrements the weight by 1. We use matrix $W \in \mathbb{R}^{n\times n}$ to denote the weighted adjacency matrix of the consensus graph, where $W_{ij} := w(i, j)$. The server then solves program (1) for $\hat{Y}$, the estimated community structure matrix.

Claim 1. *Efficient exact recovery of the community structure is possible. Under certain topological and statistical conditions, Algorithm 1 recovers the true* *community structure matrix Y^* perfectly and efficiently.*

Figure 2 illustrates an example federated myopic learning model with $n = 6$ nodes and $K = 3$ clients. We show fields of view of the clients, the local subgraphs, and the consensus weighted graph from top to bottom.

Here we list and discuss the assumptions that will be used in our analysis. For simplicity of analysis, it is assumed that the groups are balanced, i.e., $\mathbf{1}^\top y^* = 0$. If the groups are unbalanced, one can solve the semidefinite program in Algorithm 1 by adding an extra constraint. For example, suppose $\mathbf{1}^\top y^* = n_0$, where $n_0 \neq 0$; then one can solve the SDP with the help of an extra constraint $\left\langle Y, \mathbf{1}\mathbf{1}^\top \right\rangle = n_0^2$.

We also assume that for each client, the signal and noise level parameters fulfill $0 < q^{(k)} < p^{(k)} < 1$. The motivation is that nodes from the same group are more likely to be connected than those from different groups. Similarly, we assume that the censorship parameter $r^{(k)}$ is in the range $(0, 0.5)$. If $r^{(k)} = 0$, there is no censorship at all, and if $r^{(k)} = 0.5$, the censored evidence graph is pure noise and provides no information.

Finally, our analysis focuses on the scenario, where every client is myopic. Our model reduces to a weighted version of the stochastic block model (SBM), if clients are not myopic and observe the whole network. Recovery in the SBM has been studied extensively in prior literature (Abbe, 2017). In this paper we are interested in the signal and noise level parameters, *as well as* the network FOV topology of the clients (i.e., $\{\Omega^{(k)}\}_{k=1}^K$). Thus we focus on the myopic regime.

4.1 Exact Recovery Conditions

In this section, we investigate the conditions for efficient exact recovery of the community structure in a federated myopic learning model. We say an algorithm achieves *exact recovery*, if the estimated community structure matrix $\hat{Y}$ is identical to the true matrix Y^*. Our analysis provides provable guarantee of efficient exact recovery through Algorithm 1.

Before we proceed, we first introduce the definition of the *signed consensus graph.*

Definition 6 (Signed Consensus Graph)**.** For a consensus graph G with the weighted adjacency matrix W as in Algorithm 1, and the community structure matrix Y^* as in Definition 1, we use $\bar{G}$ to denote the graph generated from the adjacency matrix $\mathbb{E}\left[W\right] \circ Y^*$, where $\circ$ denotes the Hadamard product. We call $\bar{G}$ the signed consensus graph of G.

Naturally, $\phi_{\bar{G}}$ is defined as the edge expansion of the signed consensus graph $\bar{G}$ as in Definition 5. For the readers' convenience, here we restate the optimization problem in Algorithm 1:

$$\begin{aligned} \underset{Y}{\text{maximize}} \quad & \langle W, Y \rangle \\ \text{subject to} \quad & Y_{ii} = 1\,, Y \succeq 0\,. \end{aligned} \tag{2}$$

It is clear that in Algorithm 1, the computation of the weighted adjacency matrix W and the signed consensus graph $\bar{G}$ can be done in polynomial time in terms of n. Also note that Problem (2) is a semidefinite program. It is known that semidefinite programs are convex and can be solved efficiently in polynomial time using existing solvers (Boyd et al., 2004). Thus the whole server algorithm can be run in polynomial time efficiently.

It remains to prove the correctness of Algorithm 1. In other words, we want to know under what conditions, Problem (2) returns the correct community structure matrix Y^* from input W, the weighted adjacency matrix of the consensus graph. For simplicity of presentation, we define a signal coefficient $s_{ij}^{(k)}$ for every node pair $i, j \in V$ and every client $k \in [K]$ as follows. If pair $(i, j) \notin \Omega^{(k)}$, we define $s_{ij}^{(k)} := 0$. Otherwise, we define $s_{ij}^{(k)} := p^{(k)} + r^{(k)} - 2p^{(k)}r^{(k)}$ if $y_i^* = y_j^*$, and $s_{ij}^{(k)} := q^{(k)} + r^{(k)} - 2q^{(k)}r^{(k)}$ if $y_i^* \neq y_j^*$. We now present our main theorem.

Theorem 2 (Exact Recovery of Community Structure)**.** *For a federated myopic learning model $\mathcal{M}(n, K, \{\Omega^{(k)}, p^{(k)}, q^{(k)}, r^{(k)}\}_{k=1}^K \mid y^*)$ with the signed consensus graph $\bar{G}$, if $\phi_{\bar{G}} > 0$ and*

$$\max_{i \in V} \left| \sum_{j \in V, k \in [K]} s_{ij}^{(k)} (1 - s_{ij}^{(k)}) \right| = O\left(\frac{\phi_{\bar{G}}^2 - \phi_{\bar{G}} \log n}{\log n} \right), \tag{3}$$

then Problem (2) *achieves exact recovery of the community structure, i.e., $\hat{Y} = Y^* = y^* y^{*\top}$, in polynomial time with probability tending to* 1*.*

5 IMPOSSIBILITY OF CLIENT LOCAL RECOVERY

In this section, we investigate the information-theoretic lower bounds for any algorithm to recover the local community structure from a single client's evidence graph $\tilde{G}^{(k)} = (V^{(k)}, \tilde{E}^{(k)})$. One question may rise from Algorithm 1 is whether it is possible for the clients to run the semidefinite program (or any other algorithm) *locally*, and obtain correct local community structure, *without* sending the information to the server. This is often unwanted from a federated learning point of view, as the community structure information should be kept confidential. As a result, we are interested in an *impossibility* guarantee for any client to recover its local community structure without the assistance of a server. Equivalently, this provides an impossibility guarantee for the server to recover the local community structure from a single client's input.

Consider client $k \in [K]$. Intuitively, for any node that is not in $V^{(k)}$, there is no way to recover its label. We define a subgraph community structure matrix $Y^{(k)*} \in \{+1, -1\}^{n^{(k)} \times n^{(k)}}$, as the matrix obtained by removing all rows and columns that are not in $V^{(k)}$. As a side note, it is worth mentioning that recovery the true edge set $E^{(k)}$ from the censored edge set $\tilde{E}^{(k)}$ is very difficult, as the probability of recovering any single edge is $1 - r$.

For simplicity, we introduce two client signal coefficients $s_+^{(k)}, s_-^{(k)}$, where $s_+^{(k)} := p^{(k)} + r^{(k)} - 2p^{(k)}r^{(k)}$, and $s_-^{(k)} := q^{(k)} + r^{(k)} - 2q^{(k)}r^{(k)}$. We now present the impossibility theorem.

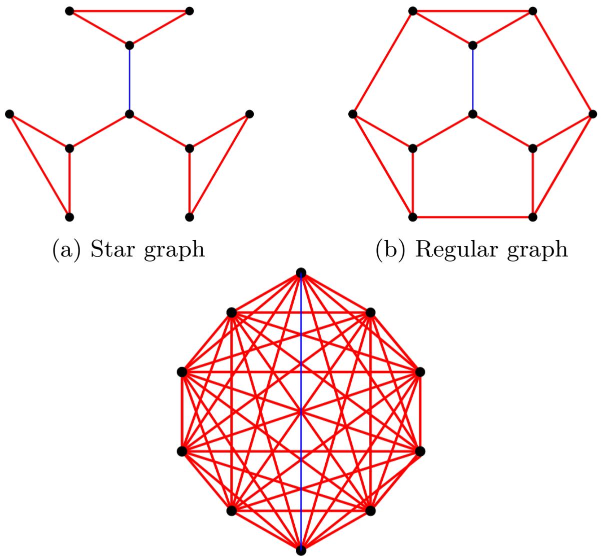
Theorem 3. *For each client $k \in [K]$, any algorithm a learner could use to guess the local community structure $Y^{(k)*}$ will fail with probability at least $1/2$, if*

$$\frac{(1 - 2r^{(k)})^2 (p^{(k)} - q^{(k)})^2}{\min(s_+^{(k)}(1 - s_+^{(k)}), s_-^{(k)}(1 - s_+^{(k)}))} = O\left(\frac{n^{(k)}}{\left|\Omega^{(k)}\right|}\right). \tag{4}$$

6 ILLUSTRATIVE EXAMPLES

We first discuss the relationship between the proposed signed weighted edge expansion ϕ_G, the regular Cheeger constant $h(G)$, and the Laplacian matrix L, for any general graph G. The case-by-case discussion in Appendix A is broken down into three parts: 1) G as an unweighted graph; 2) G as a positively weighted graph; and 3) G as a signed weighted graph.

It is worth highlighting that the regular Cheeger constant $h(G)$ is only defined for unweighted graphs. The Laplacian matrix of any positively weighted graph is always positive semidefinite, which can be proved by invoking Gershgorin circle theorem and the diagonal dominance property. Algebraically, diagonal dominance requires that $|d(i)| = \left|\sum_{j \neq i} W_{ij}\right| \geq \sum_{j \neq i} |W_{ij}|$ for every node i. The direction above, however, does not hold in the case of signed weighted graphs. If there exists some W_{ij} that is less than 0, The right-hand side will be strictly greater than the left-hand side, thus breaking diagonal dominance of the Laplacian. To this end, the proposed signed weighted edge expansion ϕ_G and Theorem 1 provides a one-way guarantee for positive semidefiniteness of the Laplacian: if ϕ_G is nonnegative, then the related graph Laplacian is positive semidefinite.



(a) Star graph

(b) Regular graph

(c) Complete graph

Figure 3: Signed weighted graph with one edge perturbation. In each graph, a red edge has a positive weight of 1. The blue edge is the the perturbed edge: its weight goes from 1 to -1. The signed weighted edge expansion ϕ_G of every graph is recorded in Table 1.

Consider the following example graphs in Figure 3. Every graph is a signed weighted graph with $n = 10$ nodes. A red edge is assigned a weight of 1 and remains constant. The blue edge is a variable edge: its weight is assigned to be $1, 0.5, 0, -0.5$ and 1 in each iteration respectively. We check the signed weighted edge expansion of each graph and report the values in Table 1. Note that in terms of the signed weighted edge expansion ϕ_G, the complete graph is the most robust one, and the star graph is the least. We also test if the graph Laplacians are positive semidefinite and mark the results in italic font. The test shows that the Laplacian of a star graph is more prone to the perturbation of a negatively weighted edge without breaking PSD, while the complete graph is less prone.

Table 1: Signed weighted edge expansion ϕ_G of every graph in Figure 3 with different weights assigned to the blue edge. Graphs breaking positive semidefiniteness are marked in italic font.

	Perturbed Edge Weight				
	1	0.5	0	-0.5	-1
Star	0.02	0.01	0.00	*-1.00*	*-2.00*
Regular	0.17	0.11	0.06	-0.94	*-1.94*
Complete	1.39	1.29	1.19	0.19	-0.81

6.1 Simulation Results

In this section, we validate the proposed Algorithm 1 and Theorem 2 through synthetic experiments.

Experiment 1: We test the proposed algorithm on a small graph using CVX. We fix the parameters p, q and r to be the same across all clients. In particular, we set $p = 0.9, q = 0.1$. The number of nodes is fixed to be $n = 30$. We control the field of view of each client as follows: every client randomly samples M nodes, and the FOV of the client is the complete subgraph spanned by the sampled nodes. Thus, the parameters are the size of the field of view M, and the number of clients K. These are the x-axis and y-axis in Figure 4, respectively. Furthermore, we are interested in comparing two regimes. One is the *multiview regime*, in which there are few clients (maximum of 20), but every client sends an evidence graph with a high signal-to-noise ratio ($r = 0.1$). The other one is the *federated regime*, in which there are many clients (maximum of 200), but every client sends a noisy evidence graph ($r = 0.4$).

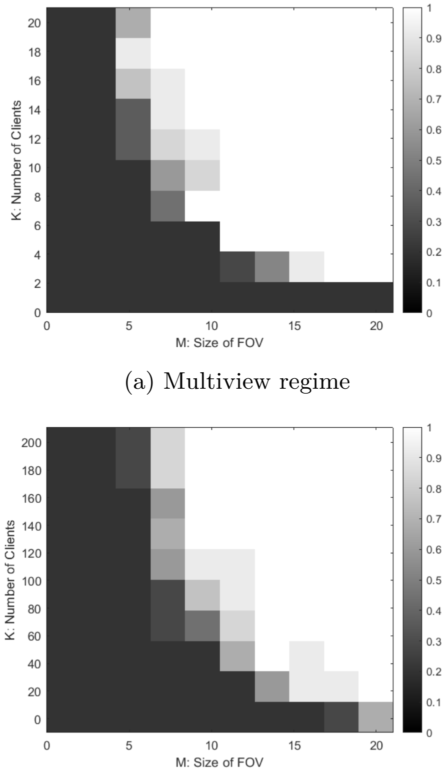
We run Algorithm 1 and check agreement between the recovered label vector $\hat{y}$, and the ground truth vector y^*. We solve the semidefinite program in Algorithm 1 using CVX (Grant and Boyd, 2014, 2008). For each pair of parameters, we calculate the empirical probability that the recovered label vector is identical to the ground truth. The experiment shows that exact recovery of the community structure can be achieved in both regimes, with a tradeoff between the signal-to-noise ratio and the number of clients.

Experiment 2: We test the proposed algorithm on a large scale graph. In this experiment we fix the number of nodes n to be $10\,000$. It is known that CVX can be inefficient to solve large scale semidefinite programs due to the large memory cost. To test the proposed algorithm with more nodes, we implement a gradient method SDP solver Yurtsever et al. (2021).

We fix the parameters p, q and r to be the same across all clients. In particular, we set $p = 0.9, q = 0.1$, and $r = 0.1$. The number of clients K iterates from 1000 to $10\,000$ with an interval of 1000, and every client randomly observes a subgraph of only $M = 20$ nodes. We run the experiment for 10 trials.

We report the average number of correctly recovered labels, and the empirical probability of exact recovery, in Figure 5. Our result suggests that even in the highly sparse, large scale case ($10\,000$ nodes in total, each client observes only 0.2% of the graph), exact recovery of the community structure can be achieved efficiently through the proposed federated learning algorithm.

Experiment 3: We test the runtime scalability of the gradient method SDP solver on a large scale graph. We iterate n from 1000 to $10\,000$ with an interval of 1000. We set $p = 0.9, q = 0.1$, $r = 0.1$, $M = 20$, and $K = n$. We report the runtime of the gradient method SDP solver in Figure 6. The runtime can be fitted almost perfectly by a third order polynomial, suggesting the efficiency of the method.



(a) Multiview regime

(b) Federated regime

Figure 4: Validating Algorithm 1 using CVX. The x-axis is the size of FOV of each client, and the y-axis is the number of clients in the federation. Even in the highly noisy case (federated regime), with a large number of clients, the central server is able to recover the community structure of the network perfectly. There is tradeoff between the amount of signal sent by each client (decided by the size of FOV, as well as the signal and noise levels), and the number of clients in the federation.

As a comparison, when n is set to 1000, CVX SDPT3 solver takes 36.26 seconds, and our gradient method solver takes only 2.86. When n is set to 2000, CVX SDPT3 solver runs out of 16GB memory, and our gradient method solver succeeds and takes only 16.66 seconds.

Experiment 4.1: We validate Theorem 2 through simulation. In this experiment we fix the number of nodes n to be 1000, and set $K = 500$, $M = 20$. In the

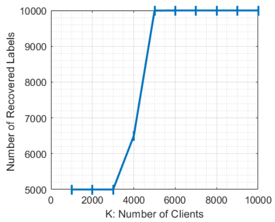


Figure 5: Validating Algorithm 1 using gradient method SDP solver (Yurtsever et al., 2021). The x-axis is the number of clients in the federation. In a large scale network with $n = 10\,000$, even each client has a limited observation of $M = 20$, with a large number of clients, the central server is able to recover the community structure of the network perfectly.

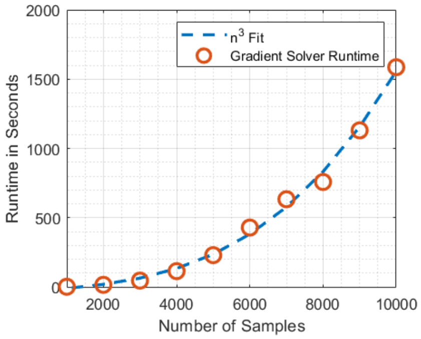


Figure 6: Runtime of the gradient method SDP solver versus the number of nodes. The runtime can be fitted almost perfectly by a third order polynomial, suggesting the efficiency of the method.

experiment the signal parameters p and q iterate over the range $(0, 1)$, with an interval of 0.05. We run 20 trials for each pair of p and q. We test cases with r being 0.1, 0.25, and 0.4, respectively.

We set $C := \frac{\phi_{\bar{G}}^2 - \phi_{\bar{G}} \log n}{\log n \max_{i \in V} \left| \sum_{j \in V, k \in [K]} s_{ij}^{(k)} (1 - s_{ij}^{(k)}) \right|}$ in the x-axis, which is equal to the right-hand side of (3) divided by its left-hand side. We report the average number of correctly recovered labels, and the empirical probability of exact recovery, in Figure 7. Our result suggests that if C is greater than a threshold, the proposed algorithm performs well and recovers the underlying community structure with high probability. This matches our theoretic findings in Theorem 2.

Experiment 4.2: We run the same experiment as in 4.1 but with different sizes of subgraphs. In this case we fix $r = 0.1$. We test cases with M being 20, 40, and 80, respectively. We report the results in Figure 8. Again, our result suggests that as long as C is greater than a constant threshold, the proposed algorithm performs well and recovers the underlying community structure with high probability. This again matches our theoretic findings in Theorem 2.

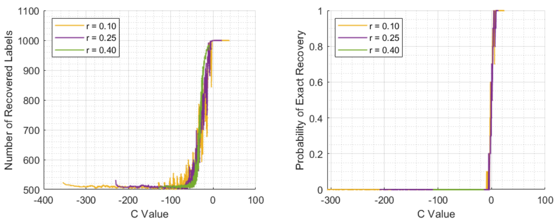


Figure 7: Simulations using different values of signal parameters p and q, and censorship parameter r. The x-axis is set by C, and the y-axis is the number of recovered labels and the empirical probability of exact recovery, respectively. This matches our theoretic findings in Theorem 2.

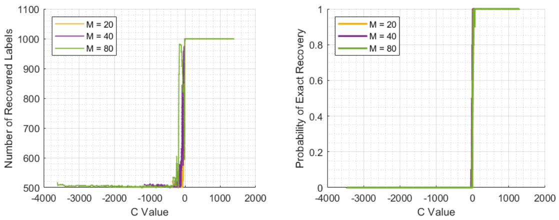


Figure 8: Simulations using different values of signal parameters p and q, and size of subgraphs M. The x-axis is set by C, and the y-axis is the number of recovered labels and the empirical probability of exact recovery, respectively. This matches our theoretic findings in Theorem 2.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 1716609-IIS.

References

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

A. Barik and J. Honorio. Exact support recovery in federated regression with one-shot communication. *arXiv preprint arXiv:2006.12583*, 2020.

S. Boyd, S. P. Boyd, and L. Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

J. C. Bronski and L. DeVille. Spectral theory for dynamics on graphs containing attractive and repulsive interactions. *SIAM Journal on Applied Mathematics*, 74(1):83–105, 2014.

T. M. Cover. *Elements of information theory*. John Wiley & Sons, 1999.

M. Cucuringu, P. Davies, A. Glielmo, and H. Tyagi. Sponge: A generalized eigenproblem for clustering signed networks. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1088–1098. PMLR, 2019.

P. Doreian and A. Mrvar. Partitioning signed social networks. *Social Networks*, 31(1):1–11, 2009.

I. Giotis and V. Guruswami. Correlation clustering with a fixed number of clusters. *Theory OF Computing*, 2:249–266, 2006.

A. Grammenos, R. Mendoza-Smith, J. Crowcroft, and C. Mascolo. Federated principal component analysis. *Neural Information Processing Systems*, 33, 2020.

M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited, 2008. `http://stanford.edu/~boyd/graph_dcp.html`.

M. Grant and S. Boyd. CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`, Mar. 2014.

M. A. Javed, M. S. Younis, S. Latif, J. Qadir, and A. Baig. Community detection in networks: A multidisciplinary review. *Journal of Network and Computer Applications*, 108:87–111, 2018.

C. Ke and J. Honorio. Information-theoretic limits for community detection in network models. *Advances in Neural Information Processing Systems*, 31, 2018.

S. Kelley, M. Goldberg, M. Magdon-Ismail, K. Mertsalov, and A. Wallace. Defining and discovering communities in social networks. In *Handbook of Optimization in Complex Networks*, pages 139–168. Springer, 2012.

A. V. Knyazev. Signed laplacian for spectral clustering revisited. *arXiv preprint arXiv:1701.01394*, 1, 2017.

J. Konečnỳ, M. Brendan, and D. Ramage. Federated optimization: Distributed optimization beyond the datacenter. *Neural Information Processing Systems, Workshop on Optimization for Machine Learning*, 2015.

J. Konečnỳ, M. Brendan, F. Yu, P. Richtárik, A. Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. *Neural Information Processing Systems, Workshop on Private Multi-Party Machine Learning*, 2016.

J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. De Luca, and S. Albayrak. Spectral analysis of signed graphs for clustering, prediction and visualization. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 559–570. SIAM, 2010.

T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3): 50–60, 2020.

B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

P. Mercado, A. Gautier, F. Tudisco, and M. Hein. The power mean laplacian for multilayer graph clustering. In *International Conference on Artificial Intelligence and Statistics*, pages 1828–1838. PMLR, 2018.

K. Smith. 60 incredible and interesting twitter stats and statistics, Jan 2020. URL `https://www.brandwatch.com/blog/twitter-stats-and-statistics/`.

J. A. Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 12(4):389–434, 2012.

Q. Yang, Y. Liu, T. Chen, and Y. Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

B. Yu. Assouad, fano, and le cam. In *Festschrift for Lucien Le Cam*, pages 423–435. Springer, 1997.

A. Yurtsever, J. A. Tropp, O. Fercoq, M. Udell, and V. Cevher. Scalable semidefinite programming. *SIAM Journal on Mathematics of Data Science*, 3 (1):171–200, 2021.

Supplementary Material: Federated Myopic Community Detection with One-shot Communication

A ADDITIONAL DISCUSSION OF ILLUSTRATIVE EXAMPLES

Here we continue the discussion in Section 6.

Unweighted Graphs. Suppose $G = (V, E)$ is an unweighted graph. Recall that the definition of the regular Cheeger constant $h(S)$ of a non-empty set $S \subset V$ and the graph G is defined as $h(S) = \frac{|\partial S|}{|S|}$, and $h(G) = \min_{S \subset V, |S| \leq n/2} h(S)$. The classical Cheeger inequality states that the spectral gap $\lambda_2(G)$ fulfills

$$\lambda_2(G) \geq \frac{h(G)^2}{2d_{\max}} \;= \min_{S \subset V, |S| \leq n/2} \left\{ \frac{|\partial S|^2}{2d_{\max} |S|^2} \right\} .$$

Without loss of generality, an unweighted graph can be reduced to a general weighted graph, by setting $W_{ij} = 1$ if there exists an edge between i and j. As a result, our signed weighted edge expansion ϕ_G of a unweighted graph simplifies to

$$\lambda_2(G) \geq \phi_G = \min_{S \subset V, |S| \leq |E|} \left\{ \frac{d_{\min} |\partial S|^2}{2 |S|^2} \right\} .$$

Here we would like to compare the lower bound provided by the classical Cheeger inequality and our novel result. Intuitively, the term $\frac{|\partial S|^2}{|S|^2}$ provides a tighter bound in the classical Cheeger inequality than in ours (because of the $|S| \leq n/2$ constraint instead of $|S| \leq |E|$), but the term $\frac{1}{d_{\max}}$ can be much worse than the term $d_{\min}$ in our bound, which is at least 1 if the graph is connected.

One may criticize that in our bound $d_{\min}$ could be 0 in some cases, making the whole bound useless. We want to clarify that it is not true. If $d_{\min} = 0$, there must exist some isolated node i and the graph is disconnected. In that case the edge expansion in both versions will be 0 by picking $S = \{i\}$, thus even the classical Cheeger inequality will not provide any insight about the spectral gap.

Positively Weighted Graphs. We now assume $G = (V, E, w)$ is a positively weighted graph, with edge weight $W_{ij} \geq 0$ for every $i \neq j$. We also assume all non-zero edge weights in G are bounded between α and β with $\alpha \geq \beta > 0$. Although the regular Cheeger constant $h(G)$ is only defined for unweighted graphs, comparison with ϕ_G is possible by introducing the following unweighted indicator graph $G' = (V, E')$, where $E' = \{(i,j) \mid W_{ij} > 0\}$. In other words, an edge in G' indicates a positively weighted edge in the original graph G. Then we have $\frac{\beta^2}{\alpha} \phi_{G'} \leq \phi_G \leq \frac{\alpha^2}{\beta} \phi_{G'}$ for any positively weighted graph G. Thus, the edge expansion of a positively weighted graph is bounded by the edge expansion of the corresponding unweighted indicator graph multiplied by a constant factor (decided by the extreme weights of the graph).

Signed Weighted Graphs. It would be hard to make general claims about the edge expansion if the graph edge weights are allowed to be negative. Intuitively, with more negatively weighted edges, the Laplacian is more likely to break positive semidefiniteness. Thus, the negatively weighted edges can be interpreted as perturbation in the Laplacian matrix, i.e., making non-diagonal entries positive. This gives rise to a question: what network topology is more robust to the perturbation of negatively weighted edges, without breaking positive semidefiniteness of the Laplacian? The question is also related to the federated myopic learning model, because without positive definiteness it would be impossible to recover the network structure using the SDP approach.

B PROOF OF THEOREM 1

In this section, we provide the proof of the Cheeger-type inequality for signed weighted graphs as stated in Theorem 1. Recall that $G = (V, E, w)$ is a general weighted graph, W is the weighted adjacency matrix, D is the degree matrix, and $L = D - W$ is the graph Laplacian.

First we introduce the following definitions and lemmas that will be used later in the proof.

Definition 7 (Rayleigh Quotient)**.** Let $L = D - W$ be an $n \times n$ graph Laplacian matrix, where D is the degree matrix, and W is the weighted adjacency matrix. The *Rayleigh Quotient* of a vector $v \in R^n$ with respect to L is defined as

$$R_L(v) := \frac{v^\top L v}{v^\top v} = \frac{\sum_{i<j} W_{ij}(v_i - v_j)^2}{v^\top v}\,.$$

Similarly, the signed Rayleigh Quotients are defined as

$$R_L^+(v) := \frac{\sum_{i<j} W_{ij}^+(v_i - v_j)^2}{v^\top v}\,, \qquad R_L^-(v) := \frac{\sum_{i<j} W_{ij}^-(v_i - v_j)^2}{v^\top v}\,.$$

Note that $R_L(v) = R_L^+(v) + R_L^-(v)$.

Using the variational characterization of eigenvalues, it follows that $\lambda_{\mathbf{1}^\perp}(L) = \min_{v \perp \mathbf{1}} R_L(v)$.

Lemma 1. *For any non-zero $\alpha \in \mathbb{R}$, it follows that*

$$R_L(v) = R_L(\alpha v)\,.$$

Proof. Note that $R_L(\alpha v) = \frac{(\alpha v)^\top L(\alpha v)}{(\alpha v)^\top(\alpha v)} = R_L(v)$. □

Lemma 2. *For any $\delta \in \mathbb{R}$ and $v \in \mathbb{R}^n, v \perp \mathbf{1}$, it follows that*

$$R_L^+(v) \geq R_L^+(v + \delta\mathbf{1})\,.$$

Proof. Starting from the right-hand side, we have

$$\begin{aligned}
R_L^+(v + \delta\mathbf{1}) &= \frac{\sum_{i<j} W_{ij}^+\left((v_i + \delta) - (v_j + \delta)\right)^2}{\sum_i \left(v_i + \delta\right)^2} \\
&= \frac{\sum_{i<j} W_{ij}^+\left(v_i - v_j\right)^2}{\sum_i \left(v_i + \delta\right)^2} \\
&= \frac{\sum_{i<j} W_{ij}^+\left(v_i - v_j\right)^2}{\sum_i \left(v_i^2 + \delta^2 + 2\delta v_i\right)} \\
&= \frac{\sum_{i<j} W_{ij}^+\left(v_i - v_j\right)^2}{\sum_i v_i^2 + n\delta^2 + 2\delta \sum_i v_i} \\
&= \frac{\sum_{i<j} W_{ij}^+\left(v_i - v_j\right)^2}{\sum_i v_i^2 + n\delta^2} \\
&\leq \frac{\sum_{i<j} W_{ij}^+\left(v_i - v_j\right)^2}{\sum_i v_i^2} \\
&= R_L^+(v)\,.
\end{aligned}$$

□

We now proceed to prove the main theorem. Our proof takes a constructive approach: if we can construct any witness set $S, S' \subset V$ with $d^+(S) \leq d^+(V)/2$ fulfilling $\lambda_{\mathbf{1}^\perp}(L) \geq \frac{1}{2} d^+_{\min}(\phi_S^+)^2 + 2\phi_{S'}^-$, then it follows from Definition 5 that $\lambda_{\mathbf{1}^\perp}(L) \geq \phi_G = \frac{1}{2} d^+_{\min} \min_{S \subset V, d^+(S) \leq d^+(V)/2} (\phi_S^+)^2 + 2 \min_{S \subset V} \phi_S^-$.

Proof of Theorem 1. Suppose v is the eigenvector associated with the eigenvalue $\lambda_{\mathbf{1}^\perp}(L)$. By definition, it follows that $v \perp \mathbf{1}$. By Lemma 1, without loss of generality we assume that the eigenvector v has unit norm, i.e., $\|v\| = 1$. We also assume that v is sorted in ascending order, i.e., $v_1 \leq \cdots \leq v_n$.

Let m be the smallest integer, such that $\sum_{i=1}^{m+1} d^+(i) > \frac{1}{2} d^+(V)$. Then $v - v_m \mathbf{1}$ is centered at m. We can find $\alpha, \alpha' > 0$, such that

$$\alpha^2 (v_1 - v_m)^2 + \alpha^2 (v_n - v_m)^2 = 1\,, \qquad \alpha'^2 v_1^2 + \alpha'^2 v_n^2 = 1\,.$$

Denote $u = \alpha(v - v_m \mathbf{1})$, and $u' = \alpha' v$. By Lemma 1 and 2, it follows that $R_L^+(u) \leq R_L^+(v)$, and $R_L^-(u') = R_L^-(v)$. Our goal is to prove

$$\lambda_{\mathbf{1}^\perp}(L) = R_L(v) = R_L^+(v) + R_L^-(v) \geq R_L^+(u) + R_L^-(u') \geq \frac{1}{2} d^+_{\min} (\phi_S^+)^2 + 2\phi_{S'}^-\,. \tag{5}$$

for some $S, S' \subset V$ with $d^+(S) \leq d^+(V)/2$.

We first prove the positive part $R_L^+(u) \geq \frac{1}{2} d^+_{\min} (\phi_S^+)^2$ of (5). The proof of the negative part will be similar.

We define a random variable t on the support $[u_1, u_n]$, with probability density function $f(t) = 2\,|t|$. One can verify that $\int_{t=u_1}^{u_n} 2\,|t| = 1$ because $u_1^2 + u_n^2 = 1$, thus $f(t)$ is a valid probability density function. Then, for any interval $[a, b]$, it follows that the probability of t falling in the interval is

$$\mathbb{P}\left\{a \leq t \leq b\right\} = \int_{t=a}^{b} 2\,|t| = -a^2 \operatorname{sgn}(a) + b^2 \operatorname{sgn}(b)\,.$$

It can be verified that

$$(a-b)^2/2 \leq \mathbb{P}\left\{a \leq t \leq b\right\} \leq |a-b|\,(|a| + |b|)\,, \tag{6}$$

in the range $[-1, 1]$. We construct a random set $S_t := \{i \mid u_i \leq t\}$. Note that we have $\sum_{i=1}^{m} d^+(i) \geq \frac{1}{2} d^+(V)$, and $u_m = 0$. As a result, if $t \leq 0$, we have $\min(d^+(S_t), d^+(V \setminus S_t)) = d^+(S_t)$, otherwise $\min(d^+(S_t), d^+(V \setminus S_t)) = d^+(V \setminus S_t)$. By Definition 4, we obtain

$$\mathbb{E}\left[d^+(S_t)\right] = \sum_i \mathbb{P}\left\{u_i \leq t\right\} d^+(i)\,, \qquad \mathbb{E}\left[d^+(V \setminus S_t)\right] = \sum_i \mathbb{P}\left\{u_i \geq t\right\} d^+(i)\,.$$

It follows that

$$\begin{aligned}\mathbb{E}\left[\min(d^+(S_t), d^+(V \setminus S_t))\right] &= \sum_{i \leq m} \mathbb{P}\left\{u_i \leq t \leq 0\right\} d^+(i) + \sum_{i > m} \mathbb{P}\left\{u_i \geq t > 0\right\} d^+(i) \\ &= \sum_i u_i^2 d^+(i)\,.\end{aligned}$$

We now analyze the positive boundary weight of S_t. It follows that

$$\begin{aligned}\mathbb{E}\left[w^+(\partial S_t)\right] &= \mathbb{E}\left[\sum_{i \in S_t, j \notin S_t} W_{ij}^+\right] \\ &= \sum_{i<j} \mathbb{P}\left\{u_i \leq t \leq u_j\right\} W_{ij}^+ \\ &\leq \sum_{i<j} |u_i - u_j|\,(|u_i| + |u_j|) W_{ij}^+ \\ &\overset{(a)}{\leq} \sqrt{\sum_{i<j} (u_i - u_j)^2 W_{ij}^+} \sqrt{\sum_{i<j} (|u_i| + |u_j|)^2 W_{ij}^+} \\ &= \sqrt{R_L^+(u) \sum_i u_i^2} \sqrt{\sum_{i<j} (|u_i| + |u_j|)^2 W_{ij}^+} \\ &\overset{(b)}{\leq} \sqrt{\frac{R_L^+(u)}{d^+_{\min}} \sum_i u_i^2 d^+(i)} \sqrt{2 \sum_i u_i^2 d^+(i)}\end{aligned}$$

$$= \sqrt{\frac{2R_L^+(u)}{d_{\min}^+}} \cdot \mathbb{E}\left[\min(d^+(S_t), d^+(V \setminus S_t))\right] ,$$

where (a) follows from the Cauchy-Schwarz inequality, and (b) follows from the definition of node degree and picking the minimum positive node degree. Rearranging the inequality above gives us

$$\mathbb{E}\left[\sqrt{\frac{2R_L^+(u)}{d_{\min}^+}} \min(d^+(S_t), d^+(V \setminus S_t)) - w^+(\partial S_t)\right] \geq 0 .$$

Note that inside the expectation is a function of t. Thus, there exists some t fulfilling

$$\sqrt{\frac{2R_L^+(u)}{d_{\min}^+}} \min(d^+(S_t), d^+(V \setminus S_t)) - w^+(\partial S_t) \geq 0 ,$$

which leads to

$$R_L^+(u) \geq \frac{1}{2} d_{\min}^+ (\phi_{S_t}^+)^2 = \frac{1}{2} d_{\min}^+ \left(\frac{w^+(\partial S_t)}{d^+(S_t)}\right)^2 , \tag{7}$$

under the condition of $d^+(S_t) \leq d^+(V)/2$.

The proof of the negative part of (5) is similar. Instead of using the random variable t on the support $[u_1, u_n]$, we define a new random variable t' on the support $[u'_1, u'_n]$. We use the same probability density function $f(t') = 2\,|t'|$, so (6) still holds for t'. For the negative boundary weight, we have

$$\begin{aligned}
\mathbb{E}\left[w^-(\partial S_{t'})\right] &= \mathbb{E}\left[\sum_{i \in S_{t'}, j \notin S_{t'}} W_{ij}^-\right] \\
&= \sum_{i<j} \mathbb{P}\left\{u'_i \leq t' \leq u'_j\right\} W_{ij}^- \\
&\overset{(a)}{\leq} \frac{1}{2} \sum_{i<j} (u'_i - u'_j)^2 W_{ij}^- \\
&= \frac{1}{2} R_L^-(u') \sum_i {u'}_i^2 \\
&\overset{(b)}{\leq} \frac{1}{2} R_L^-(u') .
\end{aligned}$$

where (a) follows from the left-hand side of (6), and (b) follows from the fact that $\sum_i {u'}_i^2 \geq {u'}_1^2 + {u'}_n^2 = 1$. Thus, there exists some t' fulfilling

$$R_L^-(u') \geq 2\phi_{S_{t'}}^- = 2w^-(\partial S_{t'}) . \tag{8}$$

Combining (7) and (8) completes our proof of (5). □

C PROOF OF THEOREM 2

Before going into the proof details, we first discuss the connection between the edge expansion $\phi_{\bar{G}}$, the exact recovery task (as in Theorem 2), and the types of graphs (as in Section 6), in the context of our federated myopic model.

Recall that in our federated myopic learning model, W is the weighted adjacency matrix computed by the server, as in Algorithm 1. Intuitively, for every pair i and j, $\mathbb{E}\left[W_{ij}\right]$ represents the average connection strength between these two nodes in the network. In other words, if $\mathbb{E}\left[W_{ij}\right] > 0$, node i and j are more likely to be in the same community; if $\mathbb{E}\left[W_{ij}\right] < 0$, they are more likely to be in different communities. In the proof we will show that $\sum_{k \in [K]} s_{ij}^{(k)} = \mathbb{E}\left[W_{ij}\right]$. We call (i, j) a "good" edge if $\sum_{k \in [K]} s_{ij}^{(k)} y_i^* y_j^* = \mathbb{E}\left[W_{ij}\right] y_i^* y_j^* > 0$, and a "bad" edge if $\sum_{k \in [K]} s_{ij}^{(k)} y_i^* y_j^* = \mathbb{E}\left[W_{ij}\right] y_i^* y_j^* < 0$. By Definition 6, the adjacency matrix of $\bar{G}$ is $\mathbb{E}\left[W\right] \circ Y^*$. As a consequence, the signed consensus graph $\bar{G}$ and the corresponding edge expansion $\phi_{\bar{G}}$ are heavily related to the chance of successful community detection. In the proof we use L^* to denote the graph Laplacian of $\bar{G}$. We show that successful exact recovery requires that the second smallest eigenvalue of L^* is strictly positive. By Theorem 1, this is fulfilled as long as the edge expansion $\phi_{\bar{G}}$ is strictly positive.

Now we discuss two types of graphs characterized by the signed consensus graph $\bar{G}$ and the signed weight matrix $\mathbb{E}[W] \circ Y^*$.

Positively Weighted Graphs. In this case, every entry in $\mathbb{E}[W] \circ Y^*$ is greater than or equal to 0. This implies that every edge in the graph is a "good" edge. Community detection is easy in this case, as long as the graph is connected. In fact, exact recovery in this case can be achieved through a greedy algorithm: the algorithm adds the first node to either community, and then adds the neighboring nodes to the same community if they are connected by a positive edge $W_{ij} > 0$, or adds them to the other community if connected by a negative edge $W_{ij} < 0$. The algorithm repeats until all nodes are assigned to a community. The edge expansion $\phi_{\bar{G}}$ is strictly positive in this case.

Signed Weighted Graphs. In this case, every entry in $\mathbb{E}[W] \circ Y^*$ is greater than, less than, or equal to 0. Community detection is harder, since "bad" edges exist in this case. One can see that the simple greedy algorithm described above will not be able to find consistent assignments because of the "bad" edges. Just like the experiments in Figure 3 and Table 1, the chance of successful community detection depends on how many negative entries exist in $\mathbb{E}[W] \circ Y^*$ and how negative the entries are: recovery becomes hard if there are many "bad" edges, or some edges are "really bad." The edge expansion $\phi_{\bar{G}}$ can be positive or negative in this case.

In the proof of Theorem 2, we will characterize the relationship between the chance of exact recovery and the edge expansion $\phi_{\bar{G}}$ in a more rigorous way.

C.1 Main Proof

For every client k with evidence graph $\tilde{G}^{(k)}$, we introduce a weighted matrix $W^{(k)}$ defined as follows. For every pair (i, j), if $(i, j) \notin \Omega^{(k)}$ we assign $W_{ij}^{(k)} = 0$. If $(i, j) \in \Omega^{(k)}$ and (i, j) is an edge in $\tilde{G}^{(k)}$, we assign $W_{ij}^{(k)} = 1$; otherwise we assign $W_{ij}^{(k)} = -1$. From Algorithm 1, one can see that the server weighted adjacency matrix is the summation of the clients' weighted matrices, i.e., $W = \sum_{k \in [K]} W^{(k)}$.

We take a primal-dual witness approach to show that the SDP (2) achieves exact recovery. For the readers' convenience, here we restate the SDP (2):

$$\begin{aligned} \underset{Y}{\text{maximize}} \quad & \langle W, Y \rangle \\ \text{subject to} \quad & Y_{ii} = 1 \,, Y \succeq 0 \,. \end{aligned}$$

Suppose the Lagrangian dual variables are Λ (for $Y_{ii} = 1$) and B (for $Y \succeq 0$), respectively, where Λ is a diagonal matrix of size $\mathbb{R}^{n \times n}$, and $B \succeq 0$ is a PSD matrix of the same size. The Lagrangian dual problem of (2) is

$$\begin{aligned} \underset{\Lambda}{\text{minimize}} \quad & \operatorname{tr}(\Lambda) \\ \text{subject to} \quad & \Lambda \text{ is diagonal} \,, \Lambda - W \succeq 0 \,. \end{aligned} \tag{9}$$

We now list the Karush–Kuhn–Tucker (KKT) conditions for a primal and dual pair (Y, Λ, B) to be optimal.

$$\begin{aligned} \Lambda - W - B &= 0 \,, && \text{(Stationarity)} \\ Y_{ii} = 1 \,, \quad Y &\succeq 0 \,, && \text{(Primal Feasibility)} \\ \Lambda \text{ is diagonal} \,, \quad B &\succeq 0 \,, && \text{(Dual Feasibility)} \\ \langle B, Y \rangle &= 0 \,. && \text{(Complementary Slackness)} \end{aligned}$$

To guarantee $Y^* = y^* y^{*\top}$ is an optimal solution to the SDP (2), all KKT conditions need to be fulfilled. First note that Y^* fulfills (Primal Feasibility) trivially. Next, combining (Stationarity) and (Complementary Slackness), we obtain that an optimal solution must fulfill

$$\langle \Lambda - W, Y^* \rangle = 0 \,. \tag{10}$$

To fulfill (10), we can construct the dual variables Λ^* and B as follows: $\Lambda^*_{ii} := \sum_{j \neq i} W_{ij} y^*_i y^*_j$ for every $i \in [n]$, and $B^* := \Lambda^* - W$. Then it only remains to prove that our construction (Y^*, Λ^*, B^*) fulfills (Dual Feasibility) and (Complementary Slackness), i.e.,

$$B^* = \Lambda^* - W \succeq 0 \,.$$

One can verify that by our constriction, y^* is always an eigenvector of $\Lambda^* - W$ with the corresponding eigenvalue of 0. Thus the inequality above is equivalent to

$$\lambda_2(\Lambda^* - W) \geq 0 \,, \tag{11}$$

The KKT conditions, once fulfilled, guarantee that $Y^* = y^* y^{*\top}$ is an optimal solution to the primal problem. However there could exist other sets of primal and dual variables satisfy all KKT conditions above. To illustrate this, we construct a set of example primal and dual variables $(\tilde{Y}, \tilde{\Lambda}, \tilde{B})$ as follows: $\tilde{Y} := \mathbf{1}\mathbf{1}^\top$ is the all-one matrix, $\tilde{\Lambda}_{ii} := \sum_{j \neq i} W_{ij}$ for every $i \in [n]$, and $\tilde{B} := \tilde{\Lambda} - W$. One can verify that $(\tilde{Y}, \tilde{\Lambda}, \tilde{B})$ fulfill all KKT conditions above, and as a result, $\tilde{Y} = \mathbf{1}\mathbf{1}^\top$ is an optimal solution to the primal SDP.

To ensure that $Y^* = y^* y^{*\top}$ is the *unique* optimal solution to (2) and eliminate all other undesirable solutions, we present the following lemma about uniqueness.

Lemma 3 (Uniqueness Condition)**.** *The SDP* (2) *achieves exact recovery and returns the unique optimal solution* $Y^* = y^* y^{*\top}$*, if*

$$\lambda_2(\Lambda^* - W) > 0 \,,$$

where Λ^* *is a diagonal matrix, such that* $\Lambda^*_{ii} = \sum_{j \neq i} W_{ij} y^*_i y^*_j$ *for every* $i \in [n]$*.*

Proof. First note that (Y^*, Λ^*, B^*) constructed above fulfill all KKT conditions given that $\lambda_2(\Lambda^* - W) \geq 0$. Thus $Y^* = y^* y^{*\top}$ is an optimal solution to the SDP (2).

Now, by enforcing the strict inequality $\lambda_2(\Lambda^* - W) > 0$, we obtain that for every vector v that is not a multiple of y^*, we have $\left\langle \Lambda^* - W, vv^\top \right\rangle > 0$ strictly. Thus to fulfill the optimality condition (10), the solution Y must be a multiple of Y^*. Furthermore, given the constraint $Y_{ii} = 1$ (Primal Feasibility), the only possible optimal solution that fulfills all KKT conditions is Y^*. □

We now begin to prove the main theorem.

Proof of Theorem 2. We first investigate the value of $W^{(k)}_{ij}$. If (i, j) is not observed by client k, the value of $W^{(k)}_{ij}$ is 0. Otherwise, if $Y^*_{ij} = 1$, we obtain that $W^{(k)}_{ij} = 1$ with probability $p^{(k)}(1 - r^{(k)}) + (1 - p^{(k)}) r^{(k)} = s^{(k)}_{ij}$, and $W^{(k)}_{ij} = -1$ with probability $1 - s^{(k)}_{ij}$. Similarly, if $Y^*_{ij} = -1$, we obtain that $W^{(k)}_{ij} = 1$ with probability $q^{(k)}(1 - r^{(k)}) + (1 - q^{(k)}) r^{(k)} = s^{(k)}_{ij}$, and $W^{(k)}_{ij} = -1$ with probability $1 - s^{(k)}_{ij}$.

Our goal is to prove $\lambda_2(\Lambda^* - W) > 0$. Note that

$$\begin{aligned} \lambda_2(\Lambda^* - W) &= \lambda_2(\Lambda^* - W - \mathbb{E}\left[\Lambda^* - W\right] + \mathbb{E}\left[\Lambda^* - W\right]) > 0 \\ &\Leftarrow \lambda_2(\Lambda^* - \mathbb{E}\left[\Lambda^*\right]) + \lambda_2(\mathbb{E}\left[W\right] - W) + \lambda_2(\mathbb{E}\left[\Lambda^* - W\right]) > 0 \\ &\Leftarrow \lambda_1(\Lambda^* - \mathbb{E}\left[\Lambda^*\right]) + \lambda_1(\mathbb{E}\left[W\right] - W) + \lambda_2(\mathbb{E}\left[\Lambda^* - W\right]) > 0 \,. \end{aligned} \tag{12}$$

Thus it is sufficient to prove (12) holds with high probability.

First we bound the third term in (12). We introduce a matrix L^* defined as $L^* := (\Lambda^* - W) \circ Y^*$, where $\circ$ denotes the Hadamard product. It is worth mentioning that $\Lambda^* - W$ and L^* share the same eigenvalues, since $L^* = (\Lambda^* - W) \circ Y^* = \Lambda^* \circ Y^* - W \circ Y^* = \Lambda^* - W \circ Y^*$. Furthermore, L^* satisfies the definition of graph Laplacians, such that $L_{ii} = \sum_{j \neq i} L_{ij} = \sum_{j \neq i} W_{ij} Y^*_{ij}$ for every $i \in V$, and $L_{ij} = -W_{ij} Y^*_{ij}$ for every $i \neq j$. Then by Definition 6, one can see that $\mathbb{E}\left[L^*\right]$ is the Laplacian of graph $\bar{G}$, with the weighted adjacency matrix $\mathbb{E}\left[W\right] \circ Y^*$. By Theorem 1, it follows that as long as $\phi_{\bar{G}} > 0$, we have $\lambda_{\mathbf{1}^\perp}(\mathbb{E}\left[L^*\right]) \geq \phi_{\bar{G}} > 0$. Since $\mathbf{1}$ is always an eigenvector of $\mathbb{E}\left[L^*\right]$ with the eigenvalue 0, this implies that all other eigenvalues of $\mathbb{E}\left[L^*\right]$ are strictly positive, which leads to the conclusion that

$$\lambda_2(\mathbb{E}\left[L^*\right]) = \lambda_{\mathbf{1}^\perp}(\mathbb{E}\left[L^*\right]) \geq \phi_{\bar{G}} > 0 \,.$$

Since $\Lambda^* - W$ and L^* share the same eigenvalues, we obtain

$$\lambda_2(\mathbb{E}\left[\Lambda^* - W\right]) > \phi_{\bar{G}} \,. \tag{13}$$

Next we bound the first term in (12). For every $i, j \in V, k \in [K]$, we construct a matrix $\Phi^{(i,j,k)} \in \mathbb{R}^{n \times n}$ as follows:

$$\Phi_{ii}^{(i,j,k)} := W_{ij}^{(k)} Y_{ij}^* \,, \qquad \Phi_{jj}^{(i,j,k)} := W_{ij}^{(k)} Y_{ij}^* \,,$$

and all other entries are 0. Then it follows that

$$\Lambda^* - \mathbb{E}\left[\Lambda^*\right] = \sum_{i,j \in V, k \in [K]} \Phi^{(i,j,k)} - \mathbb{E}\left[\Phi^{(i,j,k)}\right] .$$

To bound the minimum eigenvalue of the expression above, we use the matrix Bernstein inequality (Tropp, 2012). Note that each matrix $\Phi^{(i,j,k)} - \mathbb{E}\left[\Phi^{(i,j,k)}\right]$ is independent, has zero mean, and the maximum eigenvalue is bounded above by 1. Also, since $W_{ij}^{(k)}$ is a $+1/-1$ random variable with parameter $s_{ij}^{(k)}$, we obtain that

$$\left\| \sum_{i,j \in V, k \in [K]} \mathbb{E}\left[\left(\Phi^{(i,j,k)} - \mathbb{E}\left[\Phi^{(i,j,k)}\right]\right)^2\right] \right\| \le 4 \max_{i \in V} \left| \sum_{j \in V, k \in [K]} s_{ij}^{(k)} (1 - s_{ij}^{(k)}) \right| .$$

We denote the right-hand side expression above by σ_Φ^2. Applying the matrix Bernstein inequality, we obtain that for every $t > 0$,

$$\mathbb{P}\left\{\lambda_1(\Lambda^* - \mathbb{E}\left[\Lambda^*\right]) \le -t\right\} \le n \exp\left(\frac{-t^2/2}{\sigma_\Phi^2 + t/3}\right) .$$

By setting $t = \phi_{\bar{G}}/2$ and requiring the probability to be bounded above by n^{-1}, we obtain that

$$\sigma_\Phi^2 \le \frac{3\phi_{\bar{G}}^2 - 8\phi_{\bar{G}} \log n}{48 \log n} \,. \tag{14}$$

Finally we bound the second term in (12). For every $i, j \in V, k \in [K]$, we construct a matrix $\Psi^{(i,j,k)} \in \mathbb{R}^{n \times n}$ as follows:

$$\Psi_{ij}^{(i,j,k)} := W_{ij}^{(k)} Y_{ij}^* \,, \qquad \Psi_{ji}^{(i,j,k)} := W_{ij}^{(k)} Y_{ij}^* \,,$$

and all other entries are 0. Then it follows that

$$\mathbb{E}\left[W\right] - W = \sum_{i,j \in V, k \in [K]} \mathbb{E}\left[\Psi^{(i,j,k)}\right] - \Psi^{(i,j,k)} .$$

To bound the minimum eigenvalue of the expression above, we use the matrix Bernstein inequality again. Similar to the previous case, each matrix $\mathbb{E}\left[\Psi^{(i,j,k)}\right] - \Psi^{(i,j,k)}$ is independent, has zero mean, and the maximum eigenvalue is bounded above by 1. Also, since $W_{ij}^{(k)}$ is a $+1/-1$ random variable with parameter $s_{ij}^{(k)}$, we obtain that

$$\left\| \sum_{i,j \in V, k \in [K]} \mathbb{E}\left[\left(\mathbb{E}\left[\Psi^{(i,j,k)}\right] - \Psi^{(i,j,k)}\right)^2\right] \right\| \le 4 \max_{i \in V} \left| \sum_{j \in V, k \in [K]} s_{ij}^{(k)} (1 - s_{ij}^{(k)}) \right| .$$

Note that the right-hand side expression above is exactly the same as the one in the previous case. Thus by invoking the matrix Bernstein inequality, setting $t = \phi_{\bar{G}}/2$, and requiring the probability to be bounded above by n^{-1}, we obtain the same condition

$$\sigma_\Phi^2 \le \frac{3\phi_{\bar{G}}^2 - 8\phi_{\bar{G}} \log n}{48 \log n} \,. \tag{15}$$

Combining the results of (13), (14), and (15), we obtain that $\lambda_2(\Lambda^* - W) > 0$ holds with probability at least $1 - 2n^{-1}$, as long as

$$4 \max_{i \in V} \left| \sum_{j \in V, k \in [K]} s_{ij}^{(k)} (1 - s_{ij}^{(k)}) \right| \le \frac{3\phi_{\bar{G}}^2 - 8\phi_{\bar{G}} \log n}{48 \log n} \,.$$

This completes our proof. □

D PROOF OF THEOREM 3

Proof of Theorem 3. We use $\mathcal{Y}^{(k)}$ to denote the hypothesis class of the local community structure of client k. One can see that the size of the hypothesis space is $\left|\mathcal{Y}^{(k)}\right| = 2^{n^{(k)}}$. By Fano's inequality (Cover, 1999) for any estimator $\hat{Y}^{(k)}$, it follows that

$$\mathbb{P}\left\{\hat{Y}^{(k)} \neq Y^{(k)*}\right\} \geq 1 - \frac{I(Y^{(k)*}, W^{(k)}) + \log 2}{\log \mathcal{Y}^{(k)}} = 1 - \frac{I(Y^{(k)*}, W^{(k)}) + \log 2}{n^{(k)} \log 2}\,,$$

where $I(Y^{(k)*}, W^{(k)})$ denotes the mutual information between the two matrices. We use $\mathbb{KL}(P_1 \| P_2)$ to denote the Kullback–Leibler divergence between two distributions P_1 and P_2. Using a pairwise KL-based bound (Yu, 1997, p. 428) we obtain

$$\begin{aligned}
I(Y^{(k)*}, W^{(k)}) &\leq \frac{1}{\left|\mathcal{Y}^{(k)}\right|^2} \sum_{Y^{(k)}, Y^{(k)\prime} \in \mathcal{Y}} \mathbb{KL}(P_{W^{(k)}|Y^{(k)}} \| P_{W^{(k)}|Y^{(k)\prime}}) \\
&\leq \max_{Y^{(k)}, Y^{(k)\prime} \in \mathcal{Y}} \mathbb{KL}(P_{W^{(k)}|Y^{(k)}} \| P_{W^{(k)}|Y^{(k)\prime}}) \\
&= \max_{Y^{(k)}, Y^{(k)\prime} \in \mathcal{Y}} \mathbb{KL}(P_{W^{(k)}|Y^{(k)}} \| P_{W^{(k)}|Y^{(k)\prime}}) \\
&= \max_{Y^{(k)}, Y^{(k)\prime} \in \mathcal{Y}} \sum_{W^{(k)}} P_{W^{(k)}|Y^{(k)}} \log \frac{P_{W^{(k)}|Y^{(k)}}}{P_{W^{(k)}|Y^{(k)\prime}}} \\
&\overset{(a)}{\leq} \left|\Omega^{(k)}\right| \cdot \max_{Y_{ij}, Y_{ij}'} \sum_{W_{ij}^{(k)}} P_{W_{ij}^{(k)}|Y_{ij}^{(k)}} \log \frac{P_{W^{(k)}|Y_{ij}^{(k)}}}{P_{W^{(k)}|Y_{ij}^{(k)\prime}}} \\
&\overset{(b)}{\leq} \left|\Omega^{(k)}\right| \cdot \max\left(s_+^{(k)} \log \frac{s_+^{(k)}}{s_-^{(k)}} + (1 - s_+^{(k)}) \log \frac{1 - s_+^{(k)}}{1 - s_-^{(k)}}, s_-^{(k)} \log \frac{s_-^{(k)}}{s_+^{(k)}} + (1 - s_-^{(k)}) \log \frac{1 - s_-^{(k)}}{1 - s_+^{(k)}}\right) \\
&= \left|\Omega^{(k)}\right| \cdot \max\left(\mathbb{KL}(s_+^{(k)} \| s_-^{(k)}), \mathbb{KL}(s_-^{(k)} \| s_+^{(k)})\right)\,,
\end{aligned}$$

where $P_{W^{(k)}|Y^{(k)}}$ denotes the distribution of $W^{(k)}$ given $Y^{(k)}$, and (a) follows from an entrywise decomposition. (b) holds by the fact that given $Y_{ij} = 1$, $W_{ij}^{(k)} = 1$ with probability $p^{(k)}(1-r^{(k)})+(1-p^{(k)})r^{(k)} = s_+^{(k)}$, and $W_{ij}^{(k)} = -1$ with probability $1 - s_+^{(k)}$; similarly, given $Y_{ij}^* = -1$, $W_{ij}^{(k)} = 1$ with probability $q^{(k)}(1 - r^{(k)}) + (1 - q^{(k)})r^{(k)} = s_{ij}^{(k)}$, and $W_{ij}^{(k)} = -1$ with probability $1 - s_{ij}^{(k)}$.

Next we give a upper bound for $\max\left(\mathbb{KL}(s_+^{(k)} \| s_-^{(k)}), \mathbb{KL}(s_-^{(k)} \| s_+^{(k)})\right)$. Note that $\mathbb{KL}(s_+^{(k)} \| s_-^{(k)}) \leq \frac{(s_+^{(k)} - s_-^{(k)})^2}{s_-^{(k)}(1-s_-^{(k)})} = \frac{(1-2r^{(k)})^2(p^{(k)}-q^{(k)})^2}{s_-^{(k)}(1-s_-^{(k)})}$, and $\mathbb{KL}(s_-^{(k)} \| s_+^{(k)}) \leq \frac{(s_-^{(k)} - s_+^{(k)})^2}{s_+^{(k)}(1-s_+^{(k)})} = \frac{(1-2r^{(k)})^2(p^{(k)}-q^{(k)})^2}{s_+^{(k)}(1-s_+^{(k)})}$. This lead to

$$\max\left(\mathbb{KL}(s_+^{(k)} \| s_-^{(k)}), \mathbb{KL}(s_-^{(k)} \| s_+^{(k)})\right) \leq \frac{(1 - 2r^{(k)})^2(p^{(k)} - q^{(k)})^2}{\min\left(s_+^{(k)}(1 - s_+^{(k)}), s_-^{(k)}(1 - s_-^{(k)})\right)}\,.$$

Now going back to the Fano's inequality, we obtain

$$\begin{aligned}
\mathbb{P}\left\{\hat{Y}^{(k)} \neq Y^{(k)*}\right\} &\geq 1 - \frac{I(Y^{(k)*}, W^{(k)}) + \log 2}{n^{(k)} \log 2} \\
&\geq 1 - \frac{\left|\Omega^{(k)}\right| \cdot \frac{(1-2r^{(k)})^2(p^{(k)}-q^{(k)})^2}{\min\left(s_+^{(k)}(1-s_+^{(k)}), s_-^{(k)}(1-s_-^{(k)})\right)} + \log 2}{n^{(k)} \log 2} \\
&\geq \frac{1}{2}\,,
\end{aligned}$$

where last inequality holds given that $\frac{(1-2r^{(k)})^2(p^{(k)}-q^{(k)})^2}{\min\left(s_+^{(k)}(1-s_+^{(k)}), s_-^{(k)}(1-s_-^{(k)})\right)} \leq \frac{n^{(k)}}{2\left|\Omega^{(k)}\right|}$. This completes our proof. □