Statistica Sinica Preprint No: SS-2023-0287	
Title	Consistent Community Detection in Multi-Layer Networks with Heterogeneous Differential Privacy
Manuscript ID	SS-2023-0287
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202023.0287
Complete List of Authors	Yaoming Zhen, Shirong Xu and Junhui Wang
Corresponding Authors	Yaoming Zhen
E-mails	yzhen8-c@my.cityu.edu.hk
Notice: Accepted author version.	

CONSISTENT COMMUNITY DETECTION IN MULTI-LAYER NETWORKS WITH HETEROGENEOUS DIFFERENTIAL PRIVACY

Yaoming Zhen[a], Shirong Xu[b], and Junhui Wang[c]

Department of Statistical Sciences, University of Toronto[a]

Department of Statistics and Data Science, University of California, Los Angeles[b]

Department of Statistics, The Chinese University of Hong Kong[c]

Abstract: As network becomes increasingly prevalent, significant attention has been devoted to addressing privacy issues in publishing network data. One of the critical challenges for data publishers is to preserve the topological structures of the original network while protecting sensitive information. In this paper, we investigate the utility of community detection in multi-layer networks under a personalized edge-flipping mechanism. This mechanism enables data publishers to protect edge information based on each node's privacy preferences. Within this framework, the community structure under the multi-layer degree-corrected stochastic block model remains invariant after appropriate debiasing, making consistent community detection in privatized multi-layer networks achievable. Theoretically, we establish the consistency of community detection in the privatized multi-layer network, demonstrating the fundamental privacy-utility tradeoff

in differentially private community detection in multi-layer networks under the proposed mechanism. Moreover, the proposed method is further supported by extensive numerical experience on synthetic and real-life multi-layer networks.

Key words and phrases: Community detection, degree heterogeneity, personalized privacy, stochastic block model, tensor decomposition.

1. Introduction

Network data has arisen as one of the most popular data formats in the past decades, providing an efficient way to represent complex systems involving various entities and their interactions. Among its wide spectrum of applications, the most notable examples reside in social networks (Du et al., 2007; Leskovec et al., 2010; Abawajy et al., 2016), which have been frequently collected by social network sites including Facebook, Twitter, LinkedIn, and Sina Weibo, and then published to third party consumers for academic research (Granovetter, 2005; Li and Das, 2013), advertisement (Klerks, 2004; Gregurec et al., 2011), crime analysis (Carrington, 2011; Ji et al., 2014), and other possible purposes. However, social network data usually conveys sensitive information related to users' privacy, and releasing them to public will inevitably lead to privacy breach, which may be abused for spam or fraudulent behaviors (Thomas and Nicol, 2010). Therefore, it is imperative

to obfuscate network data to avoid privacy breach without compromising the intrinsic topological structures of the network data.

To protect privacy of data, differential privacy has emerged as a standard framework for measuring the capacity of a randomized algorithm in terms of privacy protection. Its applications to network data are mainly concentrated on two scenarios, node differential privacy (Kasiviswanathan et al., 2013; Day et al., 2016; Ullman and Sealfon, 2019) and edge differential privacy (Karwa and Slavković, 2016; Hehir et al., 2022; Yan, 2021, 2025). The former aims to protect the privacy of all edges of some nodes while the latter mainly focuses on limiting the disclosure of edges in networks. A critical challenge in privacy-preserving network data analysis lies in understanding the effect of privacy guarantee on the subsequent data analyses, such as community detection (Hehir et al., 2022), degree inference (Yan, 2021), and link prediction (Xu et al., 2018; Epasto et al., 2022).

In this paper, we investigate a scenario where a multi-layer network is shared with third parties for community detection while preserving edge privacy. Although numerous methods have been proposed for community detection in multi-layer networks (Lei et al., 2020; Chen et al., 2022; Xu et al., 2023; Ma and Nandy, 2023), the privacy implications in this context remain largely unexplored in the literature. Moreover, existing network

data analyses predominantly consider providing uniform privacy protection for edges within single-layer networks, disregarding the heterogeneous privacy preferences of users in practical scenarios. These approaches not only diminish the service quality for users willing to give up their privacy to some great extend but also offer inadequate protection for those who are more concerned about their privacy. To address this challenge, we introduce a personalized edge-flipping mechanism designed to accommodate the diverse privacy preferences of individual users. It empowers users to specify the level of connectivity behavior they are comfortable sharing within a social network. Thus, our approach enables the release of networks with varying degrees of privacy protection on edges. Notably, we find that the community structure of the privatized network remains consistent through appropriate debiasing procedure under the degree-corrected multi-layer stochastic block model (DC-MSBM), preserving the utility of the original network for community detection. Correspondingly, we develop a community detection method tailored for privatized multi-layer networks and establish its theoretical guarantees for community detection consistency. Our theoretical findings are reinforced through experimentation on synthetic networks and the FriendFeed network.

The rest of the paper is structured as follows. Section 2 introduces

the notations of tensors and the background of DC-MSBM. Section 3 introduces the application of differential privacy in network data. In Section 4, we propose the personalized edge-flipping mechanism and show that the community structure of DC-MSBM stays invariant under this mechanism, for which we develop an algorithm for community detection on privatized networks. Section 5 establishes the consistency of community detection of the proposed method. Section 6 conducts various simulations to validate the theoretical results and apply the proposed method to a FriendFeed network. Section 7 concludes the paper, and all technical proofs and necessary lemmas are deferred to the Appendix.

2. Preliminaries

2.1 Background of Multi-layer Networks

We first introduce some notations and the DC-MSBM (Paul and Chen, 2021). Throughout the paper, we denote $[n] = \{1, ..., n\}$ for any positive integer n, and denote tensors by bold Euler script letters. For a tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, denote $\boldsymbol{\mathcal{A}}_{i_1,:,:} \in \mathbb{R}^{I_2 \times I_3}$, $\boldsymbol{\mathcal{A}}_{:,i_2,:} \in \mathbb{R}^{I_1 \times I_3}$ and $\boldsymbol{\mathcal{A}}_{:,:,i_3} \in \mathbb{R}^{I_1 \times I_2}$ as the i_1-th horizontal, i_2-th lateral, and i_3-th frontal slide of $\boldsymbol{\mathcal{A}}$, respectively. In addition, denote $\boldsymbol{\mathcal{A}}_{:,i_2,i_3} \in \mathbb{R}^{I_1}$, $\boldsymbol{\mathcal{A}}_{i_1,:,i_3} \in \mathbb{R}^{I_2}$, and $\boldsymbol{\mathcal{A}}_{i_1,i_2,:} \in \mathbb{R}^{I_3}$ as the (i_2, i_3)-th mode-1, (i_1, i_3)-th mode-2 and (i_1, i_2)-th mode-3 fibers of $\boldsymbol{\mathcal{A}}$, respectively.

For $j \in [3]$, let $\boldsymbol{\mathcal{M}}_j(\boldsymbol{\mathcal{A}})$ be the mode-j major matricization of $\boldsymbol{\mathcal{A}}$ (Kolda and Bader, 2009). Specifically, $\boldsymbol{\mathcal{M}}_j(\boldsymbol{\mathcal{A}})$ is a matrix in $\mathbb{R}^{I_j \times \prod_{i \neq j} I_i}$ such that

$$\boldsymbol{\mathcal{A}}_{i_1,i_2,i_3} = \big(\boldsymbol{\mathcal{M}}_j(\boldsymbol{\mathcal{A}})\big)_{i_j,m}, \text{ with } m = 1 + \sum_{\substack{l=1 \\ l \neq j}}^{3} (i_l - 1) \prod_{\substack{i=1 \\ i \neq j}}^{l-1} I_i.$$

For any matrices $\boldsymbol{M}^{(1)} \in \mathbb{R}^{J_1 \times I_1}$, $\boldsymbol{M}^{(2)} \in \mathbb{R}^{J_2 \times I_2}$, $\boldsymbol{M}^{(3)} \in \mathbb{R}^{J_3 \times I_3}$, the mode-1 product between $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{M}^{(1)}$ is a $J_1 \times I_2 \times I_3$ tensor, defined as $(\boldsymbol{\mathcal{A}} \times_1 \boldsymbol{M}^{(1)})_{j_1,i_2,i_3} = \sum_{i_1=1}^{I_1} \boldsymbol{\mathcal{A}}_{i_1,i_2,i_3} \boldsymbol{M}^{(1)}_{j_1,i_1}$, for $j_1 \in [J_1]$, $i_2 \in [I_2]$, and $i_3 \in [I_3]$. The mode-2 product $\boldsymbol{\mathcal{A}} \times_2 \boldsymbol{M}^{(2)} \in \mathbb{R}^{I_1 \times J_2 \times I_3}$ and mode-3 product $\boldsymbol{\mathcal{A}} \times_3 \boldsymbol{M}^{(3)} \in \mathbb{R}^{I_1 \times I_2 \times J_3}$ are defined similarly. The Tucker rank, also known as multi-linear rank, of $\boldsymbol{\mathcal{A}}$ is defined as (r_1, r_2, r_3), where $r_1 = \text{rank}(\boldsymbol{\mathcal{M}}_1(\boldsymbol{\mathcal{A}}))$, $r_2 = \text{rank}(\boldsymbol{\mathcal{M}}_2(\boldsymbol{\mathcal{A}}))$ and $r_3 = \text{rank}(\boldsymbol{\mathcal{M}}_3(\boldsymbol{\mathcal{A}}))$. Further, if $\boldsymbol{\mathcal{A}}$ has Tucker rank (r_1, r_2, r_3), it admits the following Tucker decomposition,

$$\boldsymbol{\mathcal{A}} = \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{U} \times_2 \boldsymbol{V} \times_3 \boldsymbol{W},$$

where $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$ is a core tensor and $\boldsymbol{U} \in \mathbb{R}^{I_1 \times r_1}$, $\boldsymbol{V} \in \mathbb{R}^{I_2 \times r_2}$ and $\boldsymbol{W} \in \mathbb{R}^{I_3 \times r_3}$ have orthonormal columns.

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote a multi-layer network with $\mathcal{V} = [n]$ being the set of n nodes and $\mathcal{E} = \{\boldsymbol{E}^{(l)}\}_{l=1}^{L}$ being the edge sets for all L layers, where $(i, j) \in \boldsymbol{E}^{(l)}$ if there exists an edge between nodes i and j in the l-th network. Generally, $\mathcal{G}$ can be equivalently represented by an order-3 adjacency tensor $\boldsymbol{\mathcal{A}} \in \{0, 1\}^{n \times n \times L}$ with $\boldsymbol{\mathcal{A}}_{i,j,l} = \boldsymbol{\mathcal{A}}_{j,i,l} = 1$ if $(i, j) \in \boldsymbol{E}^{(l)}$ and 0 otherwise.

Moreover, we assume the edges are independent Bernoulli random variables with $P(\boldsymbol{\mathcal{A}}_{i,j,l} = \boldsymbol{\mathcal{A}}_{j,i,l} = 1) = \boldsymbol{\mathcal{P}}_{i,j,l}$, for any $i \leq j \in [n]$ and $l \in [L]$, where $\boldsymbol{\mathcal{P}} \in \mathbb{R}^{n\times n\times L}$ is the underlying probability tensor. The degree-corrected multi-layer stochstic block model assumes that

$$\boldsymbol{\mathcal{P}}_{i,j,l} = d_i d_j \boldsymbol{\mathcal{B}}_{c_i,c_j,l}, \text{ for } i,j \in [n], l \in [L],$$

where c_i and d_i denote the community membership assignment and degree heterogeneity parameter of node i across all network layers, and $\boldsymbol{\mathcal{B}}_{c_i,c_j,l}$ is the linking probability between community c_i and c_j in the l-th layer. Note that we assume the community memberships of the nodes are homogeneous across all network layers. This allows us to define a community membership matrix. Specifically, let $\boldsymbol{Z} \in \{0,1\}^{n\times K}$ be the community membership matrix of K communities such that $\boldsymbol{Z}_{i,c_i} = 1$ and $\boldsymbol{Z}_{i,k} = 0$ for $k \neq c_i$. The probability tensor of the DC-MSBM can thus be written as

$$\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{B}} \times_1 \boldsymbol{DZ} \times_2 \boldsymbol{DZ}, \tag{2.1}$$

where $\boldsymbol{D} = diag\{d_1, \ldots, d_n\}$ is a diagonal matrix.

Furthermore, for two sequences f_n and g_n, we denote $f_n = O(g_n)$ if $\lim_{n\to+\infty} \sup |f_n|/g_n < +\infty$, $f_n = o(g_n)$ if $\lim_{n\to+\infty} |f_n|/g_n = 0$, $f_n = \Omega(g_n)$ if $\lim_{n\to+\infty} \sup |f_n|/g_n > 0$, $f_n \gg g_n$ if $\lim_{n\to+\infty} |f_n|/g_n = +\infty$, and $f_n \asymp g_n$

if $f_n = O(g_n)$ and $f_n = \Omega(g_n)$. Let $\|\cdot\|$ denote the l_2-norm of a vector or the spectral norm of a matrix, $\|\cdot\|_\infty$ denote the l_∞-norm of the vectorization of the input matrix or tensor, and $\|\cdot\|_F$ denote the Frobenius norm of a matrix or tensor, and the $l_{2,1}$-norm of a matrix $\boldsymbol{M} \in \mathbb{R}^{r\times c}$ is defined as $\|\boldsymbol{M}\|_{2,1} = \sum_{i=1}^{r} \|\boldsymbol{M}_{i,:}\|$, where $\boldsymbol{M}_{i,:}$ is the i-th row of $\boldsymbol{M}$.

2.2 Differential Privacy

Differential privacy (DP; Dwork et al. 2006) has emerged as a standard statistical framework for protecting personal data during data sharing processes. The formal definition of ϵ-DP is given as follows.

Definition 1 (ϵ-DP)**.** A randomized mechanism $\mathcal{M}$ satisfies ϵ-differential privacy if for any two neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$ differing in only one record, it holds that

$$\sup_{S\in\mathcal{S}} \frac{P(\mathcal{M}(\mathcal{D}) = S)}{P(\mathcal{M}(\mathcal{D}') = S)} \leq \exp(\epsilon),$$

where $\mathcal{S}$ denotes the output space of $\mathcal{M}$.

Another variant of differential privacy is known as local differential privacy (LDP), wherein each individual data point undergoes perturbation with noise prior to data collection procedure. The formal definition of noninteractive ϵ-LDP is provided as follows.

Definition 2 (ϵ-LDP)**.** For any $\epsilon > 0$, the randomized mechanism $\mathcal{M}$ satisfies ϵ-local differential privacy for an individual data point $X \in \mathcal{D}$ if

$$\sup_{\widetilde{x}\in\widetilde{\mathcal{X}}} \sup_{x,x'} \frac{P\big(\mathcal{M}(X)=\widetilde{x}|X=x\big)}{P\big(\mathcal{M}(X)=\widetilde{x}|X=x'\big)} \leq \exp(\epsilon),$$

where $\widetilde{\mathcal{X}}$ denotes the output space of $\mathcal{M}$.

It's worth noting that privacy protection under ϵ-LDP can be analyzed within the framework of classic ϵ-DP in specific scenarios. Particularly, if $\mathcal{M}$ satisfies ϵ-local differential privacy and is independently applied to datasets of independent samples, for any neighboring datasets $\mathcal{D}$ and $\mathcal{D}'$ differing only in the i-th record, we have

$$\sup_{\widetilde{\mathcal{D}}\in\mathcal{X}^n} \frac{P\big(\mathcal{M}(\mathcal{D})=\widetilde{\mathcal{D}}\big)}{P\big(\mathcal{M}(\mathcal{D}')=\widetilde{\mathcal{D}}\big)} = \sup_{\widetilde{x}\in\mathcal{X}} \frac{P\big(\mathcal{M}(X_i)=\widetilde{x}|X_i=x\big)}{P\big(\mathcal{M}(X_i)=\widetilde{x}|X_i=x'\big)} \leq \exp(\epsilon),$$

for $i \in [n]$. Thus, ϵ-LDP achieves the classic ϵ-DP if we consider the output space $\mathcal{S} = \mathcal{X}^n$. In addition, as suggested by a referee, when $\mathcal{D}$ is a multi-layer network, Definition 2 has already given a nature definition of ϵ-LDP on multi-layer network. In the next section, we will provide a simplified definition thanks to the independence among the network edges.

3. Edge Differential Privacy in Network Data

In the realm of network data, two primary variants of differential privacy emerge: node differential privacy (Kasiviswanathan et al., 2013; Day et al.,

2016) and edge differential privacy (Karwa and Slavković, 2016; Wang et al., 2022; Yan, 2025). The former considers the protection of all information associated with a node in network data, while the latter on the edges. This paper delves into the privacy protection of edges in multi-layer networks. The formal definition of ϵ-edge differential privacy is given as follows.

Definition 3. (ϵ-edge DP) A randomized mechanism $\mathcal{M}$ is ϵ-edge differentially private if

$$\sup_{S\in\mathcal{S}} \sup_{\delta(\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{A}}')=1} \frac{P(\mathcal{M}(\boldsymbol{\mathcal{A}})=S|\boldsymbol{\mathcal{A}})}{P(\mathcal{M}(\boldsymbol{\mathcal{A}}')=S|\boldsymbol{\mathcal{A}}')} \leq \exp(\epsilon),$$

where $\delta(\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{A}}')$ counts the number of difference entries between $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{A}}'$ and $\mathcal{S}$ denotes the output space of $\mathcal{M}(\cdot)$.

The definition of ϵ-edge DP bears a resemblance to classic ϵ-DP, as it requires the output distribution of the randomized mechanism $\mathcal{M}$ to remain robust against alterations to any single edge in the network. It is thus difficult for attackers to infer any single edge based on the released network information S. In the literature, ϵ-edge DP finds widespread usages in releasing various network information privately, such as node degrees (Karwa and Slavković, 2016; Fan et al., 2020), shortest path length (Chen et al., 2014), and community structure (Mohamed et al., 2022). Under the framework of DC-MSBM, we consider a specific variant of ϵ-LDP for edges

in multilayer network data.

Definition 4. (Weak ϵ-edge DP) Let $\boldsymbol{\mathcal{A}}$ denote the adjacency tensor of a multi-layer network with n nodes. We say a randomized mechanism $\mathcal{M}$ satisfies weak ϵ-edge differential privacy if

$$\sup_{\widetilde{x}\in\widetilde{\mathcal{X}}}\sup_{x,x'\in\mathcal{X}}\frac{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l})=\widetilde{x}|\boldsymbol{\mathcal{A}}_{i,j,l}=x\big)}{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l})=\widetilde{x}|\boldsymbol{\mathcal{A}}_{i,j,l}=x'\big)}\leq\exp(\epsilon), \tag{3.2}$$

for any $i,j\in[n]$ and $l\in[L]$, where $\widetilde{\mathcal{X}}$ denotes the range of edges.

Definition 4 is a weak one because the left hand side of (3.2) is the supremum of a single probability ratio instead of the product of all probability ratios corresponding to all edges. The latter is usually larger and leads to stronger definition. However, if $\mathcal{M}$ satisfies weak ϵ-edge DP and is independently applied to $\boldsymbol{\mathcal{A}}$ entrywisely.

Given the independence of $\boldsymbol{\mathcal{A}}_{i,j,l}$'s, we have $P\big(\mathcal{M}(\boldsymbol{\mathcal{A}})|\boldsymbol{\mathcal{A}}\big)=\Pi_{i\leq j\in[n],l\in[L]}$ $P\left(M(\boldsymbol{\mathcal{A}}_{i,j,l})|\boldsymbol{\mathcal{A}}_{i,j,l}\right)$, leading to

$$\begin{aligned}&\sup_{\widetilde{\boldsymbol{\mathcal{A}}}}\sup_{\delta(\boldsymbol{\mathcal{A}},\boldsymbol{\mathcal{A}}')=1}\frac{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}})=\widetilde{\boldsymbol{\mathcal{A}}}|\boldsymbol{\mathcal{A}}\big)}{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}}')=\widetilde{\boldsymbol{\mathcal{A}}}|\boldsymbol{\mathcal{A}}'\big)}\\ =&\sup_{i,j,l}\sup_{\widetilde{x}\in\widetilde{\mathcal{X}}}\sup_{x,x'\in\mathcal{X}}\frac{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l})=\widetilde{x}|\boldsymbol{\mathcal{A}}_{i,j,l}=x\big)}{P\big(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l})=\widetilde{x}|\boldsymbol{\mathcal{A}}_{i,j,l}=x'\big)}\leq\exp(\epsilon).\end{aligned} \tag{3.3}$$

It is evident from (3.3) that privacy protection through weak ϵ-edge DP is equivalent to achieving ϵ-edge DP, provided the independence of edges. Furthermore, a similar correlation can be established between ϵ-edge LDP

and the (k, ϵ)-edge DP framework, as explored in prior works such as Hay et al. (2009) and Yan (2025).

In this paper, we mainly consider multi-layer networks with binary edges, i.e., $\mathcal{X} = \{0, 1\}$. To achieve weak ϵ-edge DP, one popular choice of $\mathcal{M}$ is the edge-flipping mechanism of $\boldsymbol{\mathcal{A}}$ with a uniform flipping probability (Nayak and Adeshiyan, 2009; Wang et al., 2016; Hehir et al., 2022). Specifically, denote the flipped multi-layer network as $\mathcal{M}_\theta(\boldsymbol{\mathcal{A}})$ with a flipping probability $1-\theta$, for some $\theta \geq 1/2$, then the (i, j, l)-th entry of $\mathcal{M}(\boldsymbol{\mathcal{A}})$ is given by

$$\mathcal{M}_\theta(\boldsymbol{\mathcal{A}}_{i,j,l}) = \begin{cases} \boldsymbol{\mathcal{A}}_{i,j,l}, & \text{with probability } \theta, \\ 1 - \boldsymbol{\mathcal{A}}_{i,j,l}, & \text{with probability } 1-\theta. \end{cases}$$

It then follows that $P\big(\mathcal{M}_\theta(\boldsymbol{\mathcal{A}})_{i,j,l} = 1\big) = \theta \boldsymbol{\mathcal{P}}_{i,j,l} + (1-\theta)(1 - \boldsymbol{\mathcal{P}}_{i,j,l})$.

Lemma 1. *The edge-flipping mechanism $\mathcal{M}_\theta$ satisfies weak ϵ-edge differential privacy when $\theta = \frac{1}{1+e^{-\epsilon}}$.*

Lemma 1 characterizes the capacity of the edge-flipping mechanism in protecting privacy under the framework of weak ϵ-edge DP. It should be noted that privacy of $\boldsymbol{\mathcal{A}}$ is completely protected when $\theta = 1/2$ or $\epsilon = 0$, in the sense that there exists no algorithm capable of inferring $\boldsymbol{\mathcal{A}}_{i,j,l}$ based on $\mathcal{M}_{1/2}(\boldsymbol{\mathcal{A}}_{i,j,l})$ more effectively than random guessing. Yet, a key disad-

vantage of the uniform flipping mechanism is its inability to accommodate different privacy preferences among edges.

We further emphasize that the data we release is the privacy-preserving network after random flipping, presented as a unified tensor, despite its composition of $O(n^2L)$ edges. In contrast to mechanisms that solely disclose summary statistics of the network, our approach enables the release of a complete data tensor with the same expected expectation as the original network after a debiasing step. However, it is important to note that some structures of the original network cannot be recovered directly due to privacy protection. Instead, estimations, such as the precise count of triangles in the original network, are still obtainable. In essence, the publication of the privacy-preserving network allows for releasing more data about networks.

4. Proposed Method

4.1 Personalized Edge-flipping

In this section, we propose a personalized edge-flipping mechanism whose flipping probabilities are governed by node-wise privacy preferences. Specifically, let $\Theta = (\theta_{i,j})_{n\times n}$ with $\theta_{i,j}$ denoting the flipping probability of the potential edge between nodes i and j across all network layers, and $\mathcal{M}_\Theta(\boldsymbol{\mathcal{A}}) =$

$\left(\mathcal{M}_{\theta_{i,j}}(\boldsymbol{\mathcal{A}}_{i,j,l})\right)_{n\times n\times L}$ with

$$\mathcal{M}_{\theta_{i,j}}(\boldsymbol{\mathcal{A}}_{i,j,l}) = \begin{cases} \boldsymbol{\mathcal{A}}_{i,j,l}, & \text{with probability } \theta_{i,j}, \\ 1-\boldsymbol{\mathcal{A}}_{i,j,l}, & \text{with probability } 1-\theta_{i,j}, \end{cases} \tag{4.4}$$

for $i \le j$ and $l \in [L]$. Also, we set $\mathcal{M}_{\theta_{i,j}}(\boldsymbol{\mathcal{A}}_{i,j,l}) = \mathcal{M}_{\theta_{j,i}}(\boldsymbol{\mathcal{A}}_{j,i,l})$ to preserve the semi-symmetry in $\mathcal{M}_\Theta(\boldsymbol{\mathcal{A}})$ with respect to the first two modes, for $i > j$.

Definition 5. (Heterogenous ϵ-edge LDP) Let $\mathcal{M}$ denote a randomized mechanism. We say $\mathcal{M}$ satisfies heterogenous $\boldsymbol{\epsilon}$-edge LDP if for any $1 \le i \le j \le n$ and $l \in [L]$ with $\boldsymbol{\epsilon} = (\epsilon_{i,j})_{i,j=1}^n$, we have

$$\sup_{\widetilde{x}\in\mathcal{X}} \sup_{x,x'\in\mathcal{X}} \frac{P(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l}) = \widetilde{x} | \boldsymbol{\mathcal{A}}_{i,j,l} = x)}{P(\mathcal{M}(\boldsymbol{\mathcal{A}}_{i,j,l}) = \widetilde{x} | \boldsymbol{\mathcal{A}}_{i,j,l} = x')} \le \exp(\epsilon_{i,j}),$$

where $\epsilon_{i,j}$ is a privacy parameter depending on nodes i and j.

Clearly, the proposed heterogenous ϵ-edge LDP allows for the variation of the privacy parameter $\epsilon_{i,j}$ from edge to edge. Particularly, heterogenous $\boldsymbol{\epsilon}$-edge LDP is equivalent to weak ϵ-edge DP when $\epsilon = \max_{i,j} \epsilon_{i,j}$. The developed concept bears resemblance to heterogeneous differential privacy (Alaggan et al., 2015) in nature, wherein individual points in a dataset are provided different privacy guarantees. The motivation behind heterogenous $\boldsymbol{\epsilon}$-edge LDP is to cater to the diverse preferences among users in the network. While some users may prioritize better service over privacy, others may prioritize keeping their social interactions as private as possible.

To allow for node-specified privacy preferences, we propose to parametrize Θ as

$$\Theta = \frac{1}{2}(\boldsymbol{f}\boldsymbol{f}^\top + \mathbf{1}_n\mathbf{1}_n^T), \tag{4.5}$$

where $\boldsymbol{f} = (f_1, ..., f_n)^\top \in [0, 1)^n$ is a vector consisting of the privacy preferences of all nodes and $\mathbf{1}_n$ is the vector with n ones. In particular, when $f_i = 0$, it signifies that $\theta_{i,j} = 1/2$ for any $j \in [n]$, indicating that the edges associated with node i are protected at the utmost secrecy level. Conversely, when f_i and f_j are close to 1, it indicates that both nodes i and j largely give up their privacy, resulting in $\boldsymbol{\mathcal{A}}_{i,j,l}$ will be truly exposed to the service provider with high probability, for $l \in [L]$. Essentially, the privacy level of an edge between two nodes is solely determined by their respective privacy preferences. Note that $f_i < 1$ ensures every edge could be flipped with a positive probability, which ensures the goal of differential privacy in network releasing.

Lemma 2. *The personalized edge-flipping mechanism $\mathcal{M}_\Theta(\boldsymbol{\mathcal{A}})$ with Θ being parametrized as in* (4.5) *satisfies heterogenous $\boldsymbol{\epsilon}$-edge LDP with $\epsilon_{i,j} = \log\frac{1+f_if_j}{1-f_if_j}$, for $i, j \in [n]$. Moreover,*

$$f_i = \sqrt{\frac{(1 - \frac{2}{1+e^{\epsilon_{i,i'}}})(1 - \frac{2}{1+e^{\epsilon_{i,j}}})}{1 - \frac{2}{1+e^{\epsilon_{i',j}}}}},$$

for any $i' \neq j$, $i' \neq i$, and $j \neq i$.

Lemma 2 shows that, under the personalized edge-flipping mechanism, the privacy guarantee of any single edge is completely determined by the pair of nodes forming that particular edge. Furthermore, it is important to note that the privacy protection provided to edges via $\mathcal{M}_\Theta$ is contingent upon the parameterization specified in (4.5). In other words, the level of privacy protection on edges will vary with the parameterization of Θ.

4.2 Decomposition after Debiasing

A critical challenge in releasing network data is to preserve network structure of interest while protecting privacy of edges. It is interesting to remark that the community structure is still encoded in the flipped network under personalized edge-flipping mechanism, which allows for consistent community detection on the flipped network after some appropriate debiasing.

Lemma 3. *Assume that $\boldsymbol{\mathcal{A}}$ is generated from the DC-MSBM in (2.1) and that the personalized flipping probability matrix satisfies the factorization property in (4.5). Let $\widetilde{\boldsymbol{\mathcal{A}}}_{i,j,l} = \mathcal{M}_{\theta_{i,j}}(\boldsymbol{\mathcal{A}}_{i,j,l}) + \frac{1}{2}(f_i f_j - 1)$. We have*

$$\mathbb{E}\Big(\widetilde{\boldsymbol{\mathcal{A}}}_{i,j,l}\Big) = f_i f_j d_i d_j \boldsymbol{\mathcal{B}}_{c_i,c_j,l}, i,j \in [n], l \in [L], \tag{4.6}$$

Lemma 3 shows that the expectation of the flipped network $\mathcal{M}_{\boldsymbol{\Theta}}(\boldsymbol{\mathcal{A}})$ preserves the same community structure in $\boldsymbol{\mathcal{A}}$ after debiasing, suggesting

that consistent community detection shall be conducted on the debiased network $\widetilde{\boldsymbol{\mathcal{A}}}$.

We remark that various network data analysis tasks remain feasible after an additional debiasing step, including estimating the counts of specific sub-graphs such as k-stars or triangles, as well as inferring the degree sequence. This is achievable because we can obtain a tensor sharing exactly the same expectation as $\boldsymbol{\mathcal{A}}$ by further dividing $\widetilde{\boldsymbol{\mathcal{A}}}_{i,j,l}$ by $f_i f_j$, for any $i, j \in [n]$, and $l \in [L]$. For community detection, this step is not necessary, as $f_i d_i$ can be considered a new degree heterogeneous parameter for node i, which will be normalized in the tensor-based variation of the SCORE method (Jin, 2015; Ke et al., 2019). Estimating and inferring certain network statistics on the differentially private network after debiasing is commonly employed. For example, randomized algorithms in Hay et al. (2009); Karwa and Slavković (2016); Yan (2021, 2025) release a perturbed degree sequence, two perturbed bi-degree sequences, or degree partitions if the order of nodes is not crucial in downstream analysis, by adding discrete Laplacian noise. Subsequently, the parameters in the β-model, with or without covariates, can be estimated using the denoised degree sequences.

By Lemma 3, the expectation of $\widetilde{\boldsymbol{\mathcal{A}}}$ can be decomposed as

$$\mathbb{E}\big(\widetilde{\boldsymbol{\mathcal{A}}}\big) = \boldsymbol{\mathcal{B}} \times_1 \boldsymbol{FDZ} \times_2 \boldsymbol{FDZ},$$

where $\boldsymbol{F} = diag(\boldsymbol{f})$. For ease of notation, we denote $\widetilde{\boldsymbol{\mathcal{P}}} = \mathbb{E}(\widetilde{\boldsymbol{\mathcal{A}}})$, and then

$$\widetilde{\boldsymbol{\mathcal{P}}} = (\boldsymbol{\mathcal{B}} \times_1 \boldsymbol{\Gamma} \times_2 \boldsymbol{\Gamma}) \times_1 \boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1} \times_2 \boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1}, \tag{4.7}$$

where $\boldsymbol{\Gamma} = diag(\sqrt{\gamma_1}, \ldots, \sqrt{\gamma_K})$ and $\gamma_k = \sum_{i=1}^{n} \boldsymbol{Z}_{i,k}(f_i d_i)^2$ is the effective size of the k-th community depending on the nodes' degree heterogeneity coefficients and heterogenous privacy preference parameters. Suppose the Tucker rank of $\boldsymbol{\mathcal{B}} \times_1 \boldsymbol{\Gamma} \times_2 \boldsymbol{\Gamma}$ is (K, K, L_0), and thus $\boldsymbol{\mathcal{B}} \times_1 \boldsymbol{\Gamma} \times_2 \boldsymbol{\Gamma}$ admits the following Tucker decomposition

$$\boldsymbol{\mathcal{B}} \times_1 \boldsymbol{\Gamma} \times_2 \boldsymbol{\Gamma} = \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{O} \times_2 \boldsymbol{O} \times_3 \boldsymbol{V}, \tag{4.8}$$

for a core tensor $\boldsymbol{\mathcal{C}} \in \mathbb{R}^{K \times K \times L_0}$, and the factor matrices $\boldsymbol{O} \in \mathbb{R}^{K \times K}$ and $\boldsymbol{V} \in \mathbb{R}^{L \times L_0}$ whose columns are orthonormal. Note that $\boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1}$ also has orthonormal columns. Plugging (4.8) into (4.7) yields the Tucker decomposition of $\widetilde{\boldsymbol{\mathcal{P}}}$ as

$$\widetilde{\boldsymbol{\mathcal{P}}} = \boldsymbol{\mathcal{C}} \times_1 \boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1}\boldsymbol{O} \times_2 \boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1}\boldsymbol{O} \times_3 \boldsymbol{V}.$$

Denote $\boldsymbol{U} = \boldsymbol{FDZ}\boldsymbol{\Gamma}^{-1}\boldsymbol{O}$ as the mode-1 and mode-2 factor matrix in the Tucker decomposition of $\widetilde{\boldsymbol{\mathcal{P}}}$. Clearly, $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_K$.

Lemma 4. *For any node pair $(i, j) \in [n] \times [n]$, we have $\boldsymbol{U}_{i,:}/\|\boldsymbol{U}_{i,:}\| = \boldsymbol{U}_{j,:}/\|\boldsymbol{U}_{j,:}\|$ if $c_i^* = c_j^*$ and $\left\|\boldsymbol{U}_{i,:}/\|\boldsymbol{U}_{i,:}\| - \boldsymbol{U}_{j,:}/\|\boldsymbol{U}_{j,:}\|\right\| = \sqrt{2}$ otherwise.*

Lemma 4 shows that the spectral embeddings of nodes within the same communities are the same after row-wise normalization. This motivates us to propose Algorithm 1 to estimate the community structure based on the Tucker decomposition of $\widetilde{\boldsymbol{\mathcal{A}}}$.

Algorithm 1: Community detection in flipped network

Input : Flipped adjacency tensor $\mathcal{M}_\Theta(\boldsymbol{\mathcal{A}})$, privacy parameter $\boldsymbol{f}$, number of communities K, tolerance τ

Output: Privacy-preserving community memberships $\widehat{\boldsymbol{Z}}$

1 Let $\widetilde{\boldsymbol{\mathcal{A}}} = \mathcal{M}_\Theta(\boldsymbol{\mathcal{A}}) + \frac{1}{2}(\boldsymbol{f}\circ\boldsymbol{f} - \mathbf{1}_n\circ\mathbf{1}_n)\circ\mathbf{1}_L$;

2 Implement Tucker decomposition on $\widetilde{\boldsymbol{\mathcal{A}}}$ with Tucker rank $(K, K, L_0 = \min\{K(K+1)/2, L\})$ as $\widetilde{\boldsymbol{\mathcal{A}}} \approx \widehat{\boldsymbol{\mathcal{C}}} \times_1 \widehat{\boldsymbol{U}} \times_2 \widehat{\boldsymbol{U}} \times_3 \widehat{\boldsymbol{V}}$.

3 Normalized the embedding matrix $\widehat{\widetilde{\boldsymbol{U}}}_{i,:} = \widehat{\boldsymbol{U}}_{i,:}/\|\widehat{\boldsymbol{U}}_{i,:}\|$, for $i \in [n]$.

4 Apply an $(1+\tau)$-optimal K-medians algorithm to $\widehat{\widetilde{\boldsymbol{U}}}$ to obtain a solution $(\widehat{\boldsymbol{Z}}, \widehat{\boldsymbol{W}})$ that satisfies,

$$\|\widehat{\boldsymbol{Z}}\widehat{\boldsymbol{W}} - \widehat{\widetilde{\boldsymbol{U}}}\|_{2,1} \le (1+\tau) \min_{\boldsymbol{Z}\in\boldsymbol{\Delta}, \boldsymbol{W}\in\mathbb{R}^{K\times K}} \|\boldsymbol{Z}\boldsymbol{W} - \widehat{\widetilde{\boldsymbol{U}}}\|_{2,1},$$

where $\boldsymbol{\Delta} \subset \{0,1\}^{n\times K}$ is the set of membership matrices.

In Algorithm 1, we first conduct a debiasing operation on the flipped network $\mathcal{M}_\Theta(\boldsymbol{\mathcal{A}})$ to obtain $\widetilde{\boldsymbol{\mathcal{A}}}$, such that the expectation of $\widetilde{\boldsymbol{\mathcal{A}}}$ admits the same DC-MSBM as in $\boldsymbol{\mathcal{A}}$. Next, a low rank Tucker approximation of $\widetilde{\boldsymbol{\mathcal{A}}}$ is

implemented to estimate the spectral embedding matrix $\widehat{\boldsymbol{U}}$. Finally, a $(1+\tau)$-optimal K-medians algorithm is applied to the normalization version of $\widehat{\boldsymbol{U}}$, which clusters the nodes into K desired communities. Herein, we follow the similar treatment in Lei and Rinaldo (2015) to apply the approximating K-medians algorithm for the normalized nodes' embedding, which appears to be more robust against outliers than the K-means algorithms.

5. Theory

In this section, we establish the asymptotic consistency of community detection on the privatized multi-layer network under the proposed personalized edge-flipping mechanism. Particularly, let $\widehat{\boldsymbol{c}} = (\widehat{c}_1, \ldots, \widehat{c}_n)$ and $\boldsymbol{c}^* = (c_1^*, c_2^*, \ldots, c_n^*)$ denote the estimated community membership vector obtained from Algorithm 1 and the true community membership vector, respectively. We assess the community detection performance with minimum scaled Hamming distance between $\hat{\boldsymbol{c}}$ and $\boldsymbol{c}^*$ under permutation (Jin, 2015; Jing et al., 2021; Zhen and Wang, 2023). Formally, it is defined as

$$\mathrm{Err}(\widehat{\boldsymbol{c}}, \boldsymbol{c}^*) = \min_{\boldsymbol{\pi} \in S_K} \frac{1}{n} \sum_{i=1}^{n} I(c_i^* = \boldsymbol{\pi}(\widehat{c}_i)), \tag{5.9}$$

where S_K is the symmetric group of degree K and $I(\cdot)$ is the indicator function. Clearly, the Hamming error in (5.9) measures the minimum fraction of nodes with inconsistent community assignments between $\widehat{\boldsymbol{c}}$ and $\boldsymbol{c}^*$.

To establish the consistency of community detection, the following technical assumptions are made.

Assumption 1. Let n_k be the cardinality of the k-th true community for $k \in [K]$, and denote $n_{\max} = \max_{k\in[K]} n_k$ and $n_{\min} = \min_{k\in[K]} n_k$, then $n_{\max} = O(n_{\min})$.

Assumption 2. Let $\gamma_{\max} = \max_{k\in[K]} \gamma_k$ and $\gamma_{\min} = \min_{k\in[K]} \gamma_k$. Assume that there exists an absolute constant C_1 such that

$$\gamma_{\max} = O(\gamma_{\min}), \text{ and } f_i^2 d_i^2 \le C_1 \frac{\gamma_{c_i^*}}{n_{c_i^*}}, \text{ for } i \in [n].$$

Assumption 3. Suppose that $\boldsymbol{\mathcal{B}}_{i,j,l} = O(s_n)$ for $i, j \in [n]$ and $l \in [L]$, where s_n is a network sparsity coefficient that may vanish with n and L. Moreover, we require s_n satisfies

$$s_n \gg \frac{1}{\overline{\psi}} \sqrt{\frac{\varphi_n \log n}{nL}},$$

where $\varphi_n = 1 - \min_{i\in[n]} f_i + 4s_n$, and $\overline{\psi} = \frac{1}{n}\sum_{i=1}^n (f_i d_i)^2$.

Assumption 4. Assume that the core tensor $\boldsymbol{\mathcal{B}}$ in the DC-MSBM model satisfies that

$$\sigma_{\min}\Big(\boldsymbol{\mathcal{M}}_3\big(\boldsymbol{\mathcal{B}}\big)\Big) = \Omega(\sqrt{L} s_n),$$

where $\sigma_{\min}(\cdot)$ denotes the smallest non-zero singular value of a matrix.

Assumption 1 ensures all the K true communities in $\mathcal{A}$ are non-degenerate as n diverges (Lei et al., 2020; Zhen and Wang, 2023). Assumption 2 imposes a homogeneity condition on the squared product of the nodes' privacy preference parameters and the degree heterogeneity coefficients. Assumption 3 places a sparsity coefficient on the core probability tensor $\boldsymbol{\mathcal{B}}$ to control the overall network sparsity, which is a common assumption for network modeling (Ghoshdastidar and Dukkipati, 2017; Guo et al., 2023; Zhen and Wang, 2023). If the f_i's are quite close to 1, we have $\varphi_n = O(s_n)$, and $s_n \gg 4\left(\frac{n}{\sum_{i=1}^n d_i^2}\right)^2 \frac{\log n}{nL} = O(\frac{\log n}{nL})$. Clearly, this reduces to the optimal sparsity assumption for consistent community detection in multi-layer network data (Jing et al., 2021). However, if $cs_n \ll 1 - \min_{i\in[n]} f_i$ for some constant c, leading to $\varphi_n \gg s_n$, the proposed network sparsity assumption is stronger than the optimal one in general. Assumption 4 assumes the smallest non-zero singular value of $\boldsymbol{\mathcal{M}}_3(s_n^{-1}\boldsymbol{\mathcal{B}})$ should scale at least at the order of $\sqrt{L}$. This is a mild assumption and can be satisfied if the entries of $s_n^{-1}\boldsymbol{\mathcal{B}}$ are indepedently and identically generated from sub-Gaussian distributions (Rudelson and Vershynin, 2009).

Theorem 1. *Under Assumptions 1-4, the Hamming error of* $\widehat{\boldsymbol{c}}$ *satisfies*

$$\mathrm{Err}(\widehat{\boldsymbol{c}}, \boldsymbol{c}^*) = O_p\left(\left(\sum_{k=1}^K v_k\right)^{1/2} \frac{\sqrt{\varphi_n \log n}}{\sqrt{nL}s_n\overline{\psi}}\right),$$

where $v_k = n_k^{-2} \sum_{c_i^*=k} \gamma_k/(f_i d_i)^2$. *Moreover, in the simplest case that all the* $\epsilon_{i,j}$*'s are the same, denoted as* ϵ*, leading to* $f_i^2 = 1 - \frac{2}{1+\exp\{\epsilon\}}$*, for* $i \in [n]$*, we have*

$$\mathrm{Err}(\hat{\boldsymbol{c}}, \boldsymbol{c}^*) = O_p\left(\sqrt{\frac{\log n}{nLs_n^2\epsilon^2}}\right),$$

when ϵ *is sufficiently small, provided that the degree heterogeneous parameters are asymptotically of the same order.*

Theorem 1 provides a probabilistic upper bound for the community detection error under the personalized edge-flipping mechanism. When $f_i d_i$ close to 1, for $i \in [n]$, Theorem 1 implies that $\varphi_n \asymp s_n$ and $\mathrm{Err}(\hat{\boldsymbol{c}}, \boldsymbol{c}^*) = o(1)$ as long as $s_n \gg \frac{\log n}{nL}$ and $K = O(1)$, which matches with the optimal sparsity requirement for consistent community detection on multi-layer networks (Jing et al., 2021). However, when $\min_{i\in[n]} f_i$ deviates from 1, φ_n will become substantially larger than s_n, leading to deterioration of the convergence rate of the Hamming error. In addition, Corollary 1 discusses the optimal network privacy guarantee of the proposed method in various scenarios.

Corollary 1. *Suppose all the conditions of Theorem 1 are met,* $K = O(1)$*,* $d_i = \Omega(1)$*, for* $i \in [n]$*, and* $\frac{\log n}{nLs_n^2} = o(1)$*.*

(1) If $f_i \asymp f_j$ *and* $f_i \gg \left(\frac{\log n}{nLs_n^2}\right)^{1/4}$ *for* $i, j \in [n]$*, we have* $Err(\hat{\boldsymbol{c}}, \boldsymbol{c}^*) = o_p(1)$*.*

(2) Let S denote the set of nodes such that $f_i \asymp \alpha_n$ for any $i \in S$ and $f_i \asymp 1$ otherwise, and assume $|S|/n \asymp \beta_n$. If $\frac{\beta_n}{\alpha_n^2(1-\beta_n)} = o(\frac{nLs_n^2}{\log n})$, we have $Err(\widehat{\boldsymbol{c}}, \boldsymbol{c}^*) = o_p(1)$.

The first scenario of Corollary 1 considers the case that all the personalized preference parameters are asymptotically of the same order. In this case, the proposed method can asymptotically reveal the network community structure as long as the personalized privacy preference parameters f_i vanishes at an order slower than $\left(\frac{\log n}{nLs_n^2}\right)^{1/4}$, which further implies the differential privacy budget parameter $\epsilon_{i,j}$ should vanish at an order slower than $\sqrt{\frac{\log n}{nLs_n^2}}$ by Lemma 2, for $1 \le i \le j \le n$. The second scenario of Corollary 1 considers the case where a small fraction β_n of the nodes are highly concerned about their privacy whose privacy preference parameters are allowed to vanish at a fast order α_n. In order to ensure the consistency of community detection, the condition $\frac{\beta_n}{\alpha_n^2(1-\beta_n)} = o(\frac{nLs_n^2}{\log n})$ is imposed to control the trade-off between α_n and β_n. Furthermore, the asymptotic order of the differential privacy budget parameters are categorized into three cases by Lemma 2; that is, $\epsilon_{i,j} \asymp \alpha_n^2$ if both nodes i and j are in S, $\epsilon_{i,j} \asymp \alpha_n$ if only one node i or j is in S, and $\epsilon_{i,j} \asymp 1$ if neither node i nor j is in S.

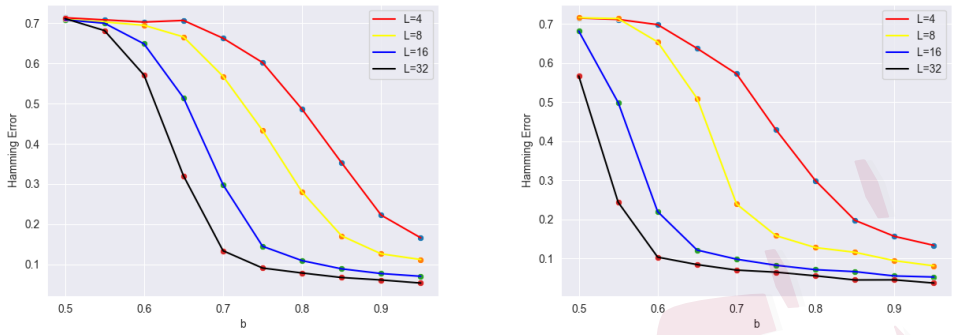
6. Numerical experiment

We now turn to examine the numerical performance of the proposed personalized edge-flipping mechanism in synthetic networks and real applications.

6.1 Synthetic networks

The synthetic multi-layer networks $\boldsymbol{\mathcal{A}} \in \{0,1\}^{n\times n\times L}$ are generated as follows. First, the probability tensor $\boldsymbol{\mathcal{B}} \in [0,1]^{K\times K\times L}$ is generated as $\boldsymbol{\mathcal{B}}_{k_1,k_2,l} = s_n\big(0.5I(k_1=k_2)+b_{k_1,k_2,l}\big)$ with $b_{k_1,k_2,l}\sim \text{Unif}(0,0.5)$, for $k_1,k_2\in[K]$. Second, $\boldsymbol{c}=(c_1,\ldots,c_n)$ are randomly drawn from $[K]$ with equal probabilities, and thus obtain the resultant community assignment matrix $\boldsymbol{Z}$. Third, calculate $\boldsymbol{\mathcal{P}} = \boldsymbol{\mathcal{B}}\times_1 \boldsymbol{DZ}\times_2 \boldsymbol{DZ}$ with $d_i\sim \text{unif}(0.5,1)$ for $i\in[n]$. Finally, each entry of $\boldsymbol{\mathcal{A}}$ is generated independently according to $\boldsymbol{\mathcal{A}}_{i,j,l}\sim \text{Bernoulli}(\boldsymbol{\mathcal{P}}_{i,j,l})$, for $1\le i\le j\le n$ and $l\in[L]$.

Example 1. In this example, we illustrate the interplay between the accuracy of community detection and the distribution of personalized privacy parameters. To mimic the users' privacy preferences, we fix $s_n = 1$ and generate $\boldsymbol{f}$ with $f_i\sim\text{Unif}(0,b)$ and $b\in\{0.5+0.05*i : i=0,1,...,9\}$. As for the size of multi-layer networks, we consider cases that $(n,L)\in\{400,800\}\times\{4,8,16,32\}$. The averaged Hamming errors over 100 replications of all cases are reported in Figure 1.



(a) n=400 (b) n=800

Figure 1: Averaged Hamming errors over 100 replications in Example 1.

In Figure 1, as b increases from 0.5 to 0.95, the Hamming errors for all values of (n, L) decrease simultaneously, indicating that small personalized privacy parameters will deteriorate the community structure in multi-layer networks. In addition, when the distribution of personalized privacy parameters is fixed, the Hamming errors improve as the network size enlarges.

Example 2. In this example, we fix $s_n = 1$, generate $f_i \sim \text{Unif}(0.95, 1)$ for $i \in [n]$, and consider two scenarios with increasing number of nodes or layers. Specifically, for the former scenario, we set the number of layers L and the number of communities K as 8 and 4, respectively, and consider cases $n \in \{100, 150, 200, 250, ..., 500\}$. For the latter one, we set $(n, K) = (200, 4)$ and consider $L \in \{4, 8, 16, 32, 64, 128\}$. The averaged Hamming errors over 100 replications of both scenarios are displayed in Figure 2.

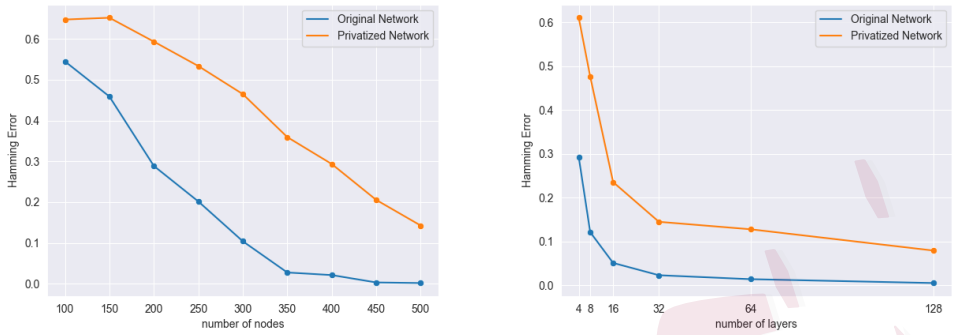


Figure 2: Averaged Hamming errors over 100 replications in Example 2.

Figure 2 shows that the convergence behaviors of the accuracy of community detection over privatized networks shares similar patterns as the original networks, which is consistent with the theory developed in Section 4 that community detection over privatized network maintains the similar order of convergence when personalized privacy parameters are close to 1.

Example 3. In this example, we analyze the convergence behaviors of the Hamming error when the personalized privacy parameters are polarized in that some people give up their privacy completely, whereas some users keep their connectivity behaviors as private as possible. To achieve this, we let n^{α} denote the number of users pursuing privacy with $a \in [0, 1]$ and then we randomly sample $\lfloor 2 * n^{a} \rfloor$ nodes and set their corresponding f_i's as $\sqrt{(nL)^{-1} log(n)}$ while keeping all the other f_i to be 1. Moreover, we set $s_n = 1$, $(K, L) = (4, 4)$, and vary $(n, a) \in \{500, 1000, 1500, 2000, 2500\} \times$

$\{0.1, 0.3, 0.5, 0.7\}$. The averaged Hamming errors over 100 replications of all cases are reported in the left penal of Figure 3.

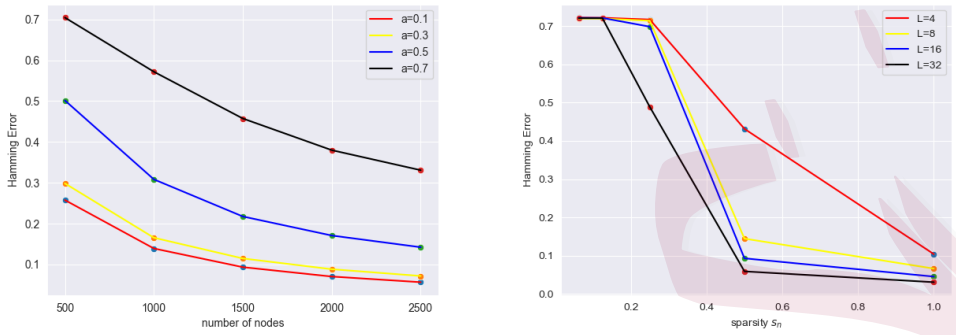


Figure 3: Averaged Hamming errors over 100 replications in Example 3 (left) and Example 4 (right).

It is evident from the left penal of Figure 3 that the Hamming errors still converge when some users chose to keep their connectivity privately, and the convergence rate becomes slower when the size of these users gets larger. It suggests that, under the personalized privacy mechanism, the privacy budget can be allocated according to users' privacy preferences, and hence some users are allowed to pursue better protections of privacy in social networks.

Example 4. In this example, we study the influence of network sparsity. The data generating scheme is exactly the same as in Example 1 except that we fix $n = 800$ and vary $(s_n, L) \in \{0.0625, 0.125, 0.25, 0.5, 1\} \times \{4, 8, 16, 32\}$.

The averaged Hamming errors over 100 replications of all cases are reported in the right penal of Figure 3.

It is clear from the right penal of Figure 3 that the averaged Hamming errors decrease as the network sparsity level s_n increases sine larger s_n yields stronger signal of the networks.

6.2 FriendFeed Multilayer Network

We apply the proposed personalized edge-flipping mechanism to a FriendFeed multi-layer social network, and compare its empirical performance on the privatized network under various personalized privacy preferences. The FriendFeed network consists of a total of 574,600 interactions among 21,006 Italian users during two months' period, which is publicly available at `http://multilayer.it.uu.se/datasets.html`. Furthermore, the users' interactions are treated as undirected edges, and categorized into three aspects, including liking, commenting, and following, which correspond to three network layers. Since the original network layers are relatively sparse and fragile, we collect the nodes in the intersection of the giant connected components of all three network layers, and extracted the corresponding sub-graphs to create a multi-layer sub-network. This pre-processing step leads to a 3-layer network with 2,012 common nodes.

In social network, like the FriendFeed data, some users are not willing to reveal their friendship privacy. For example, someone might not willing to reveal her or his privacy with a famous person or a group leader in a certain community. In this case, user i can choose a smaller f_i to better protect her or his local connectivity pattern. Further, this can even prevent attackers from inferring user i's linking pattern via transitivity. Herein, transitivity refers to the fact that a friend's friend is likely to be a friend. As such, people normally could infer the connectivity behavior between i and j, giving their common friends i''s. It is thus necessary to protect the individual's local neighborhood transitivity privacy personally. Under our randomized network flipping mechanism, the users i's preference is

$$f_i = \sqrt{\frac{(\theta_{i',i} - \frac{1}{2})(\theta_{i,j} - \frac{1}{2})}{(\theta_{i',j} - \frac{1}{2})}},$$

for any $j \neq i$, $i' \neq i$ and $i' \neq j$.

As $\theta_{i',i} > 1/2$ by definition, $\theta_{i',i} - 1/2$ is the excess probability that $\mathcal{A}_{i,i',l}$ maintains unflipped, for $l \in [L]$. Therefore, the larger f_i is, the larger the excess maintaining probability ratio between edge pairs $(\mathcal{A}_{i',i,l}, \mathcal{A}_{i,j,l})$ and edge $\mathcal{A}_{i',j,l}$, and the transitivity pattern is more likely to maintain. If users in the FriendFeed network can choose their own preferences f_i's, their local neighborhood connectivity patterns could be protected.

Before proceeding, we first estimate the number of communities K

following a similar treatment as in Ke et al. (2019). First, let κ be a user-specific upper bound of K, and we perform a Tucker decomposition approximation with Tucker rank (κ, κ, L) on the multi-layer network adjacency tensor $\boldsymbol{\mathcal{A}}$ to obtain mode-1 and mode-2 factor matrix $\bar{\boldsymbol{U}}$ and mode-3 factor matrix $\bar{\boldsymbol{V}}$. Next, we investigate the elbow point of the leading singular values of $\mathcal{M}_1(\boldsymbol{\mathcal{A}} \times_3 \bar{\boldsymbol{V}})$, and estimate K as the number of leading singular values right before the elbow point. In the FriendFeed network, we set $\kappa = 15$, and the first 20 leading singular values of $\mathcal{M}_1(\boldsymbol{\mathcal{A}} \times_3 \bar{\boldsymbol{V}})$ are displayed at Figure 4. It is clear that the elbow point appears at the 3rd leading singular value, and hence we set $K = 2$. As there is no ground truth

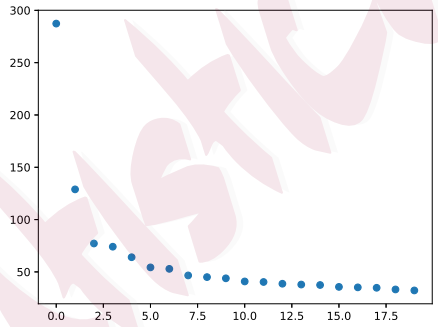


Figure 4: The first 20 leading singular values of $\mathcal{M}_1(\boldsymbol{\mathcal{A}} \times_3 \bar{\boldsymbol{V}})$ in the FriendFeed multi-layer network.

of the community structure in the FriendFeed netowrk, we simply treat the detected communities by the proposed method with $\boldsymbol{f} = \mathbf{1}_{2,012}$ as the truth.

We further select 30 nodes with the largest degrees in each detected community to visualize the 3-layer sub-network with 60 common nodes in the left panel of Figure 5. Clearly, the following layer is much denser than the other two layers, which suggests that a user may follow many other users, but only likes or comments on much fewer users she or he follows.

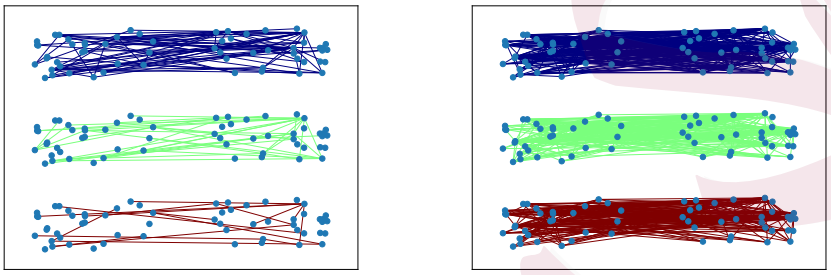


Figure 5: The original 3-layer FriendFeed sub-network with 60 popular nodes (left), and a randomly selected flipped sub-network with $\beta = 10\%$ (right). Both panels consist of the following layer (blue), commenting layer (green), and liking layer (red).

We then evaluate the Hamming error of the proposed method under different distributions of $\boldsymbol{f}$. To generate the personalized privacy preference vector $\boldsymbol{f}$, we randomly selected $\lfloor \beta \times 2,012 \rfloor$ coordinates of $\boldsymbol{f}$ and set these privacy preference parameters as 0.02 while setting other f_i's as 0.98, where $\lfloor x \rfloor$ denotes the largest integer that is small than or equal to x and β

varies in $\{2\%, 4\%, ..., 20\%\}$. Intuitively, as β increases, the expectation of f_i decreases for $i \in [2, 012]$, leading to better privacy protection for the whole network. The corresponding sub-network of a randomly selected flipped network with $\beta = 10\%$ is displayed in the right penal of Figure 5. It is clear that the flipped network becomes relatively denser and substantially deviates form the original network for privacy protection. The averaged Hamming errors of the proposed method over 100 replications on the flipped FriendFeed network with various values of β are reported in Table 1.

Table 1: Hamming errors of the proposed method on the FriendFeed network under different edge-flipping strengths.

β	2%	4%	6%	8%	10%	12%	14%	16%	18%	20%
Err	0.0723	0.0862	0.0969	0.1052	0.1235	0.1251	0.1365	0.1443	0.1501	0.1665

It is evident from Table 1 that the proposed method is able to deliver satisfactory community detection for the flipped multi-layer network the personalized edge flipping mechanism. Its Hamming errors increase with β as expected, as the flipped networks with higher edge-flipping probabilities deviate more from the original one, leading to better privacy protection at the cost of a relatively compromised detection of communities.

7. Conclusions

This paper proposes a personalized edge-flipping mechanism to protect nodes' connectivity behaviors in multi-layer network data. On the positive side, the edge flipping probabilities are allocated according to nodes' privacy preferences and demands so that protecting the connectivity behaviors could be vary from one user to the other. However, on the negative side, there might be a risk in leaking the users' privacy preferences. Theoretically, we show that the community structure of the flipped multi-layer network remains invariant under the degree-corrected multi-layer stochastic block model, which makes consistent community detection on the flipped network possible. A simple community detection method is proposed with some appropriate debiasing of the flipped network. Its asymptotic consistency is also established in terms of community detection, which allows a small fraction of nodes to keep their connectivity behaviors as private as possible. The established theoretical results are also supported by numerical experiments on various synthetic networks and a real-life FriendFeed multi-layer network.

Acknowledgements

We thank the Associate Editor and two anonymous referees, whose constructive comments have lead to significant improvement of the paper. This work is supported in part by HK RGC Grants GRF-11301521, GRF-11311022, GRF-14306523, CUHK Startup Grant 4937091, and CUHK Direct Grant 4053588.

References

Abawajy, J. H., M. I. H. Ninggal, and T. Herawan (2016). Privacy preserving social network data publication. *IEEE Communications Surveys & Tutorials 18*(3), 1974–1997.

Alaggan, M., S. Gambs, and A.-M. Kermarrec (2015). Heterogeneous differential privacy. *arXiv preprint arXiv:1504.06998*.

Carrington, P. J. (2011). Crime and social network analysis. *The SAGE Handbook of Social Network Analysis*, 236–255.

Chen, R., B. C. Fung, P. S. Yu, and B. C. Desai (2014). Correlated network data publication via differential privacy. *The VLDB Journal 23*, 653–676.

Chen, S., S. Liu, and Z. Ma (2022). Global and individualized community detection in inhomogeneous multilayer networks. *The Annals of Statistics 50*(5), 2664–2693.

Day, W.-Y., N. Li, and M. Lyu (2016). Publishing graph degree distribution with node differen-

tial privacy. In *Proceedings of the 2016 International Conference on Management of Data*, pp. 123–138.

Du, N., B. Wu, X. Pei, B. Wang, and L. Xu (2007). Community detection in large-scale social networks. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 Workshop on Web Mining and Social Network Analysis*, pp. 16–25.

Dwork, C., F. McSherry, K. Nissim, and A. Smith (2006). Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pp. 265–284. Springer.

Epasto, A., V. Mirrokni, B. Perozzi, A. Tsitsulin, and P. Zhong (2022). Differentially private graph learning via sensitivity-bounded personalized pagerank. *Advances in Neural Information Processing Systems 35*, 22617–22627.

Fan, Y., H. Zhang, and T. Yan (2020). Asymptotic theory for differentially private generalized β-models with parameters increasing. *arXiv preprint arXiv:2002.12733*.

Ghoshdastidar, D. and A. Dukkipati (2017). Uniform hypergraph partitioning: Provable tensor methods and sampling techniques. *The Journal of Machine Learning Research 18*(1), 1638–1678.

Granovetter, M. (2005). The impact of social structure on economic outcomes. *Journal of Economic Perspectives 19*(1), 33–50.

Gregurec, I., T. Vranešević, and D. Dobrinić (2011). The importance of database marketing in social network advertising. *International Journal of Management Cases 13*(4), 165–172.

Guo, X., Y. Qiu, H. Zhang, and X. Chang (2023). Randomized spectral co-clustering for large-scale directed networks. *Journal of Machine Learning Research 24*(380), 1–68.

Hay, M., C. Li, G. Miklau, and D. Jensen (2009). Accurate estimation of the degree distribution of private networks. In *2009 Ninth IEEE International Conference on Data Mining*, pp. 169–178. IEEE.

Hehir, J., A. Slavković, and X. Niu (2022). Consistent spectral clustering of network block models under local differential privacy. *The Journal of privacy and confidentiality 12*(2).

Ji, S., W. Li, M. Srivatsa, J. S. He, and R. Beyah (2014). Structure based data de-anonymization of social networks and mobility traces. In *International Conference on Information Security*, pp. 237–254. Springer.

Jin, J. (2015). Fast community detection by score. *The Annals of Statistics 43*(1), 57–89.

Jing, B.-Y., T. Li, Z. Lyu, and D. Xia (2021). Community detection on mixture multilayer networks via regularized tensor decomposition. *The Annals of Statistics 49*(6), 3181–3205.

Karwa, V. and A. Slavković (2016). Inference using noisy degrees: Differentially private β-model and synthetic graphs. *The Annals of Statistics 44*(1), 87–112.

Kasiviswanathan, S. P., K. Nissim, S. Raskhodnikova, and A. Smith (2013). Analyzing graphs with node differential privacy. In *Theory of Cryptography Conference*, pp. 457–476. Springer.

Ke, Z. T., F. Shi, and D. Xia (2019). Community detection for hypergraph networks via

regularized tensor power iteration. *arXiv preprint arXiv:1909.06503*.

Klerks, N. P. (2004). The network paradigm applied to criminal organisations: theoretical nitpicking or a relevant doctrine for investigators? recent developments in the: Theoretical nitpicking or a relevant doctrine for investigators? recent. In *Transnational Organised Crime*, pp. 111–127. Routledge.

Kolda, T. G. and B. W. Bader (2009). Tensor decompositions and applications. *SIAM review 51*(3), 455–500.

Lei, J., K. Chen, and B. Lynch (2020). Consistent community detection in multi-layer network data. *Biometrika 107*(1), 61–73.

Lei, J. and A. Rinaldo (2015). Consistency of spectral clustering in stochastic block models. *The Annals of Statistics 43*(1), 215–237.

Leskovec, J., K. J. Lang, and M. Mahoney (2010). Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th International Conference on World Wide Web*, pp. 631–640.

Li, N. and S. K. Das (2013). Applications of k-anonymity and l-diversity in publishing online social networks. In *Security and Privacy in Social Networks*, pp. 153–179. Springer.

Ma, Z. and S. Nandy (2023). Community detection with contextual multilayer networks. *IEEE Transactions on Information Theory 69*(5), 3203–3239.

Mohamed, M. S., D. Nguyen, A. Vullikanti, and R. Tandon (2022). Differentially private com-

munity detection for stochastic block models. In *International Conference on Machine Learning*, pp. 15858–15894. PMLR.

Nayak, T. K. and S. A. Adeshiyan (2009). A unified framework for analysis and comparison of randomized response surveys of binary characteristics. *Journal of Statistical Planning and Inference 139*(8), 2757–2766.

Paul, S. and Y. Chen (2021). Null models and community detection in multi-layer networks. *Sankhya A*, 1–55.

Rudelson, M. and R. Vershynin (2009). Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences 62*(12), 1707–1739.

Thomas, K. and D. M. Nicol (2010). The koobface botnet and the rise of social malware. In *2010 5th International Conference on Malicious and Unwanted Software*, pp. 63–70. IEEE.

Ullman, J. and A. Sealfon (2019). Efficiently estimating erdos-renyi graphs with node differential privacy. *Advances in Neural Information Processing Systems 32*, 3770–3780.

Wang, Q., T. Yan, B. Jiang, and C. Leng (2022). Two-mode networks: inference with as many parameters as actors and differential privacy. *Journal of Machine Learning Research 23*(292), 1–38.

Wang, Y., X. Wu, and D. Hu (2016). Using randomized response for differential privacy preserving data collection. In *EDBT/ICDT Workshops*, Volume 1558, pp. 0090–6778.

Xu, D., S. Yuan, X. Wu, and H. Phan (2018). Dpne: Differentially private network embedding. In *Advances in Knowledge Discovery and Data Mining: 22nd Pacific-Asia Conference, PAKDD 2018, Melbourne, VIC, Australia, June 3-6, 2018, Proceedings, Part II 22*, pp. 235–246. Springer.

Xu, S., Y. Zhen, and J. Wang (2023). Covariate-assisted community detection in multi-layer networks. *Journal of Business & Economic Statistics 41*(3), 915–926.

Yan, T. (2021). Directed networks with a differentially private bi-degree sequence. *Statistica Sinica 31*(4), 2031–2050.

Yan, T. (2025). Differentially private analysis of networks with covariates via a generalized β-model. *Science China Mathematics*, 1–32.

Zhen, Y. and J. Wang (2023). Community detection in general hypergraph via graph embedding. *Journal of the American Statistical Association 118*(543), 1620–1629.

Department of Statistical Sciences, University of Toronto

E-mail: yaoming.zhen@utoronto.ca

Department of Statistics and Data Science, University of California, Los Angeles

E-mail: shirong@stat.ucla.edu

Department of Statistics, The Chinese University of Hong Kong

E-mail: junhuiwang@cuhk.edu.hk