Differential Privacy Preserving Spectral Graph Analysis

Yue Wang, Xintao Wu, and Leting Wu

University of North Carolina at Charlotte
{ywang91,xwu,lwu8}@uncc.edu

Abstract. In this paper, we focus on differential privacy preserving spectral graph analysis. Spectral graph analysis deals with the analysis of the spectra (eigenvalues and eigenvector components) of the graph's adjacency matrix or its variants. We develop two approaches to computing the ϵ-differential eigen decomposition of the graph's adjacency matrix. The first approach, denoted as *LNPP*, is based on the Laplace Mechanism that calibrates Laplace noise on the eigenvalues and every entry of the eigenvectors based on their sensitivities. We derive the global sensitivities of both eigenvalues and eigenvectors based on the matrix perturbation theory. Because the output eigenvectors after perturbation are no longer orthogonormal, we postprocess the output eigenvectors by using the state-of-the-art vector orthogonalization technique. The second approach, denoted as *SBMF*, is based on the exponential mechanism and the properties of the matrix Bingham-von Mises-Fisher density for network data spectral analysis. We prove that the sampling procedure achieves differential privacy. We conduct empirical evaluation on a real social network data and compare the two approaches in terms of utility preservation (the accuracy of spectra and the accuracy of low rank approximation) under the same differential privacy threshold. Our empirical evaluation results show that *LNPP* generally incurs smaller utility loss.

Keywords: differential privacy, spectral graph analysis, privacy preservation.

1 Introduction

There have been attempts [1–3] to formalize notions of differential privacy in releasing aggregate information about a statistical database and the mechanism to providing privacy protection to participants of the databases. Differential privacy [1] is a paradigm of post-processing the output of queries such that the inclusion or exclusion of a single individual from the data set make no statistical difference to the results found. Differential privacy is usually achieved by directly adding calibrated laplace noise on the output of the computation f. The calibrating process of this approach includes the calculation of the global sensitivity of the computation f that bounds the possible change in the computation output over any two neighboring databases. The added noise is generated

J. Pei et al. (Eds.): PAKDD 2013, Part II, LNAI 7819, pp. 329–340, 2013.

from a Laplace distribution with the scale parameter determined by the global sensitivity of f and the user-specified privacy threshold ϵ. This approach works well for traditional aggregate functions (often with low sensitivity values) over tabular data. In [4], McSherry and Talwar introduced a general mechanism with differential privacy that comes with guarantees about the quality of the output, even for functions that are not robust to additive noise. The idea is to sample from the distribution specified by the exponential mechanism distribution. This mechanism skews a base measure to the largest degree possible while ensuring differential privacy, focusing probability on the outputs of highest value.

In this paper, we focus on differential privacy preserving spectral graph analysis. Spectral graph analysis deals with the analysis of the spectra (eigenvalues and eigenvector components) of the graph's adjacency matrix or its variants. We develop two approaches to computing the ϵ-differential private spectra, the first k eigenvalues and the corresponding eigenvectors, from the input graph G. The first approach, denoted as *LNPP*, is based on the Laplace Mechanism [1] that calibrates Laplace noise on the eigenvalues and every entry of the eigenvectors based on their sensitivities. We derive the global sensitivities of both eigenvalues and eigenvectors based on the matrix perturbation theory [5]. Because the output eigenvectors after perturbation are no longer orthogonormal, we postprocess the output eigenvectors by using the state-of-the-art vector orthogonalization technique [6]. The second approach, denoted as *SBMF*, is based on the exponential mechanism [4] and the properties of the matrix Bingham-von Mises-Fisher density for network data spectral analysis [7]. We prove that the Gibbs sampling procedure [7] achieves differential privacy. We conduct empirical evaluation on a real social network data and compare the two approaches in terms of utility preservation (the accuracy of spectra and the accuracy of low rank approximation) under the same differential privacy threshold. Our empirical evaluation results show that *L*NPP generally incurs smaller utility loss.

2 Preliminaries

2.1 Differential Privacy

We revisit the formal definition and the mechanism of differential privacy. For differential privacy, a database is treated as a collection of *rows*, with each row corresponding to an individual record. Here we focus on how to compute graph statistics (eigen-pairs) from private network topology described as its adjacency matrix. We aim to ensure that the inclusion or exclusion of a link between two individuals from the graph make no statistical difference to the results found.

Definition 1. *(Differential Privacy [1]) A graph analyzing algorithm Ψ that takes as input a graph G, and outputs $\Psi(G)$, preserves ϵ-differential edge privacy if for all closed subsets S of the output space, and all pairs of neighboring graphs G and G' from $\Gamma(G)$,*

$$Pr[\Psi(G) \in S] \leq e^{\varepsilon} \cdot Pr[\Psi(G') \in S], \tag{1}$$

where $\Gamma(G) = \{G'(V, E') | \exists!(u, v) \in G \text{ but } (u, v) \notin G'\}$.

A differentially private algorithm provides an assurance that the probability of a particular output is almost the same no matter whether any particular edge is included or not. A general method for computing an approximation to any function while preserving ϵ-differential privacy is given in [1]. This mechanism for achieving differential privacy computes the sum of the true answer and random noise generated from a Laplace distribution. The magnitude of the noise distribution is determined by the sensitivity of the computation and the privacy parameter specified by the data owner. The sensitivity of a computation bounds the possible change in the computation output over any two neighboring graphs (differing at most one link).

Definition 2. *(Global Sensitivity [1]) The global sensitivity of a function* $f : D \to \boldsymbol{R}^d$ *(*$G \in D$*),in the analysis of a graph* G*, is*

$$GS_f(G) := \max_{G,G' s.t. G' \in \Gamma(G)} \|f(G) - f(G')\|_1 \tag{2}$$

Theorem 1. *(The Laplace Mechanism [1]) An algorithm* A *takes as input a graph* G*, and some* $\varepsilon > 0$*, a query* Q *with computing function* $f : D^n \to \boldsymbol{R}^d$*, and outputs*

$$\boldsymbol{A}(G) = f(G) + (Y_1, ..., Y_d) \tag{3}$$

where the Y_i *are drawn i.i.d from* $Lap(GS_f(G)/\varepsilon)$*. The Algorithm satisfies* ϵ*-differential privacy.*

Another exponential mechanism was proposed to achieve differential privacy for diverse functions especially those with large sensitivities [4]. The exponential mechanism is driven by a score function q that maps a pair of input(G) and output(r) from $D^n \times \boldsymbol{R}^d$ to a real valued score($q(G, r)$) which indicates the probability associated with the output. Given an input graph G, an output $r \in \boldsymbol{R}^d$ is returned such that $q(G, r)$ is approximately maximized while guaranteeing differential privacy.

Theorem 2. *(The General Exponential Mechanism [4]) For any function* q*:* $(D^n \times \boldsymbol{R}^d) \to \mathbb{R}$*, based on a query* Q *with computing function* $f : D^n \to \boldsymbol{R}^d$*, and base measure* μ *over* $\boldsymbol{R}^d$*, the algorithm* Υ *which takes as input a graph* G *and some* $\alpha > 0$ *and outputs some* $r \in \boldsymbol{R}^d$ *is defined as*

$$\Upsilon_q^\alpha(G) := \textit{Choosing } r \textit{ with probability proportional to } exp(\alpha q(G, r)) \times \mu(r).$$

$\Upsilon_q^\alpha(G)$ *gives* $(2\alpha\Delta q)$*-differential privacy, where* Δq *is the largest possible difference in* q *when applied to two input graphs that differ only one link, for all* r*.*

Theorem 3. *(Composition Theorem [2]) If we have* n *numbers of* ϵ*-differentially private mechanisms* $M_1, \cdots, M_n$*, computed using graph* G*, then any composition of these mechanisms that yields a new mechanism* M *is* $n\varepsilon$*-differentially private.*

Differential privacy can extend to group privacy as well: changing a group of k edges in the data set induces a change of at most a multiplicative $e^{k\epsilon}$ in the corresponding output distribution. In this paper, we focus on the edge privacy. We can extend the algorithm to achieve the node privacy by using the composition theorem [2].

2.2 Spectral Analysis of Network Topologies

A graph G can be represented as a symmetric adjacent matrix $A_{n\times n}$ with $A_{i,j} = 1$ if there is an edge between nodes i and j, and $A_{i,j} = 0$ otherwise. We denote the i-th largest eigenvalue of A by λ_i and the corresponding eigenvector by $\boldsymbol{u}_i$. The eigenvector $\boldsymbol{u}_i$ is a $n \times 1$ column vector of length 1. The matrix A can be decomposed as

$$A = \sum_{i=1}^{n} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^T. \tag{4}$$

One major application of the spectral decomposition is to approximate the graph data A by a low dimension subspace A_k that captures the main information of the data, i.e., minimizes $\|A - A_k\|_F$. Given the top-k eigenvalues and corresponding eigenvectors, we have a rank-k approximation to A as

$$A_k = \sum_{i=1}^{k} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^T = U_k \Lambda_k {U_k}^T, \tag{5}$$

where Λ_k is a diagonal matrix with $\Lambda_{ii} = \lambda_i$ and $U_k = (\boldsymbol{u}_1, .., \boldsymbol{u}_k)$.

U_k belongs to the Stiefel manifold. Denoted as $\nu_{k,n}$, the Stiefel manifold is defined as the set of rank-k $k \times n$ orthonormal matrices. One of the commonly used probability distributions on the Stiefel manifold $\nu_{k,n}$ is called the matrix Bingham-von Mises-Fisher density (Definition 3).

Definition 3. *(The matrix Bingham-von Mises-Fisher density [7]) The probability density of the matrix Bingham-von Mises-Fisher distribution is given by*

$$\mathbb{P}_{\mathrm{BMF}}(X|C_1, C_2, C_3) \propto etr\{C_3^T X + C_2 X^T C_1 X\}, \tag{6}$$

where C_1 and C_2 are assumed to be symmetric and diagonal matrices, repectively.

The matrix Bingham-von Mises-Fisher density arises as a posterior distribution in latent factor models for multivariate and relational data. Recently, a Gibbs sampling scheme was developed for sampling from the matrix Bingham-von Mises-Fisher density with application of network spectral analysis [7] based on the latent factor model(Definition 4).

Definition 4. *(The latent factor model for network data [7]) The network data is represented with a binary matrix A so that $A_{i,j}$ is the 0-1 indicator of a link between nodes i and j. The latent factor model with a probit link for such network data is defined as:*

$$\begin{aligned} A_{i,j} &= \delta_{(c,\infty)}\left(Z_{i,j}\right) \\ Z_{i,j} &= \boldsymbol{u}_i^T \Lambda \boldsymbol{u}_j + e_{i,j} \\ Z &= U \Lambda U^T + E \end{aligned}$$

where E is modeled as a symmetric matrix of independent normal noise, Λ is a diagonal matrix and U is an element of $\nu_{k,n}$, with k generally much smaller than n. Given a uniform prior distribution for U, we have

$$\mathbb{P}(U|Z,\Lambda) \propto etr(Z^T U \Lambda U^T/2) = etr(\Lambda U^T Z U/2),$$

which is a Bingham distribution with parameters $C_1 = Z/2$, $C_2 = \Lambda$ and $C_3 = 0$.

Lemma 1. *[7]A uniform prior distribution on eigenvectors U and independent normal(0, τ^2) prior distributions for the eigenvalues Λ give*

$$\mathbb{P}(\Lambda|Z,U) = \Pi_{i=1}^{k} normal(\tau^2 \boldsymbol{u}_i^T Z \boldsymbol{u}_i/(2+\tau^2), 2\tau^2/(2+\tau^2))$$
$$\mathbb{P}(U|Z,\Lambda) \propto etr(Z^T U \Lambda U^T/2) = etr(\Lambda U^T Z U/2),$$

where 'normal(u, σ^2)' denotes the normal density with mean u and variance σ^2.

The sampling scheme by Hoff [7] ensures Lemma 1 to approximate inferences for U and Λ for a given graph topology. As suggested in [7], the prior parameter τ^2 is usually chosen as the number of nodes n since this is roughly the variance of the eigenvalues of an $n \times n$ matrix of independent standard normal noise.

3 Mechanism for Spectral Differential Privacy

In this section, we present two approaches to computing the ϵ-differential private spectra: *LNPP*, which is based on the Laplace Mechanism (Theorem1), and *SBMF*, which is based on the exponential mechanism [4] and the properties of the matrix Bingham-von Mises-Fisher density for network data spectral analysis [7].

3.1 LNPP: Laplace Noise Perturbation with Postprocessing

In this approach, we output the first k eigenvalues, $\boldsymbol{\lambda}^{(k)} = (\lambda_1, \lambda_2, ..., \lambda_k)$, and the corresponding eigenvectors, $U_k = (\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_k)\big)$, under ϵ-differential privacy with the given graph G and parameters k, ϵ. We first derive the sensitivities for the eigenvalues and eigenvectors in Results 1, 2. We then follow Theorem 1 to calibrate Laplace noise to the eigenvalues and eigenvectors based on the derived sensitivities and privacy parameter. Because the perturbed eigenvectors will no longer be orthogonalized to each other, we finally do a postprocess to normalize and orthogonalize the perturbed eigenvectors following Theorem 4.

Result 1. *Given a graph G with its adjacent matrix A, the global sensitivity of each eigenvalue is $GS_{\lambda_i}(G) = 1$,($i \in [1, n]$); the global sensitivity of the first $k(k > 1)$ eigenvalues as a vector, $\boldsymbol{\lambda}^{(k)} = (\lambda_1, \lambda_2, ..., \lambda_k)$, is $GS_{\boldsymbol{\lambda}^{(k)}}(G) = \sqrt{2k}$.*

Proof. We denote adding/deleting an edge between nodes i and j on the original graph G as a perturbation matrix P added to the original adjacent matrix A. $P_{n\times n}$ is a symmetric matrix where only $P_{i,j}$ and $P_{j,i}$ have value $1/-1$ and all other entries are zeros. We denote λ_i as the eigenvalue of the matrix A and $\tilde{\lambda}_i$

as that of matrix $A + P$. We have the Euclidean norm and Frobenius norm of P respectively as $\|P\|_2 = 1$ and $\|P\|_F = \sqrt{2}$. Based on the matrix perturbation theory [5](Chapter IV, Theorem 4.11), we have

$$GS_{\lambda_i}(G) \leq \max|\tilde{\lambda}_i - \lambda_i| \leq \|P\|_2 = 1$$

and

$$GS_{\boldsymbol{\lambda}^{(k)}}(G) = \sum_{i=1}^{k} |\tilde{\lambda}_i - \lambda_i| \leq \sqrt{k}\sqrt{\sum_{i=1}^{k}(\tilde{\lambda}_i - \lambda_i)^2} \leq \sqrt{k}\|P\|_F = \sqrt{2k}.$$

Result 2. *Given a graph G with its adjacent matrix A, the sensitivity of each eigenvector, $\boldsymbol{u}_i (i > 1)$, is $GS_{\boldsymbol{u}_i}(G) = \frac{\sqrt{n}}{\min\{|\lambda_i - \lambda_{i-1}|, |\lambda_i - \lambda_{i+1}|\}}$, where the denominator is commonly referred as the eigen-gap of λ_i. Specifically, the sensitivities of the first and last eigenvector are respectively $GS_{\boldsymbol{u}_1}(G) = \frac{\sqrt{n}}{\lambda_1 - \lambda_2}$ and $GS_{\boldsymbol{u}_n}(G) = \frac{\sqrt{n}}{\lambda_{n-1} - \lambda_n}$.*

Proof. We define the perturbation matrix P and other terminologies the same as those in the proof of Result 1. We denote eigenvectors of matrix $A, A + P$ respectively as column vectors $\boldsymbol{u}_i$ and $\tilde{\boldsymbol{u}}_i$ $(i \in [1, k])$. Based on the matrix perturbation theory [5](Chapter V, Theorem 2.8), for each eigenvector $\boldsymbol{u}_i (i > 1)$, we have

$$\begin{aligned} GS_{\boldsymbol{u}_i}(G) &\leq \sqrt{n}\|\tilde{\boldsymbol{u}}_i - \boldsymbol{u}_i\|_2 \leq \frac{\sqrt{n}\|P\boldsymbol{u}_i\|_2}{\min\{|\lambda_i - \lambda_{i-1}|, |\lambda_i - \lambda_{i+1}|\}} \\ &\leq \frac{\sqrt{n}}{\min\{|\lambda_i - \lambda_{i-1}|, |\lambda_i - \lambda_{i+1}|\}}. \end{aligned}$$

Specifically for $i = 1$ (similarly for $i = n$),

$$GS_{\boldsymbol{u}_1}(G) \leq \sqrt{n}\|\tilde{\boldsymbol{u}}_1 - \boldsymbol{u}_1\|_2 \leq \frac{\sqrt{n}\|P\|_2}{\lambda_1 - \lambda_2} = \frac{\sqrt{n}}{\lambda_1 - \lambda_2}.$$

Theorem 4. *(Orthogonalization of vectors with minimal adjustment [6]) Given a set of non-orthogononal vectors $\boldsymbol{x}_1, ..., \boldsymbol{x}_k$, we could construct components $\boldsymbol{u}_1, ..., \boldsymbol{u}_k$ such that $\boldsymbol{x}_i$ is close to $\boldsymbol{u}_i$ for each i, and $U^T U$ is an identity matrix where $U = (\boldsymbol{u}_1, ..., \boldsymbol{u}_k)$ following*

$$U = XC,$$

where $X = (\boldsymbol{x}_1, ..., \boldsymbol{x}_k)$ is the set of $n \times 1$vectors and $X^T X$ is non-singular, C is the symmetric square-root of $(X^T X)^{-1}$ and is unique.

Algorithm 1. *LNPP: Laplace noise calibration approach*

Input: Graph adjacent matrix A, privacy parameter ϵ and dimension parameter k
Output: The first k eigenvalues $\widetilde{\boldsymbol{\lambda}}^{(k)} = (\tilde{\lambda}_1, \tilde{\lambda}_2, ..., \tilde{\lambda}_k)$ and corresponding eigenvectors $\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, ..., \tilde{\boldsymbol{u}}_k$, which satisfies ϵ-differential privacy.

1: Decomposition A to obtain the first k eigenvalues $\boldsymbol{\lambda}^{(k)} = (\lambda_1, \lambda_2, ..., \lambda_k)$ and the corresponding eigenvectors $\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_k$;
2: Distribute ϵ into $\epsilon_0, ..., \epsilon_k$, s.t.$\epsilon = \sum_{i=0}^{k} \epsilon_i$;
3: Follow Theorem 1 to add Laplace noise to $\boldsymbol{\lambda}^{(k)}$ with ϵ_0 based on $GS_{\boldsymbol{\lambda}^{(k)}}(G)$ derived in Result 1 and obtain $\widetilde{\boldsymbol{\lambda}}^{(k)} = (\tilde{\lambda}_1, ..., \tilde{\lambda}_k)$;
4: For i:=1 to k do
Follow Theorem 1 to add Laplace noise to $\boldsymbol{u}_i$ with ϵ_i based on $GS_{\boldsymbol{u}_i}(G)$ derived in Result 2 and obtain $\tilde{\boldsymbol{x}}_i$;
Endfor
5: Normalize and orthogonalize $\tilde{\boldsymbol{x}}_1, ..., \tilde{\boldsymbol{x}}_k$ to obtain $\tilde{\boldsymbol{u}}_1, ..., \tilde{\boldsymbol{u}}_k$ following Theorem 4.
6: Output $\tilde{\lambda}_1, \tilde{\lambda}_2, ..., \tilde{\lambda}_k$ and $\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, ..., \tilde{\boldsymbol{u}}_k$

Algorithm 1 illustrates our *LNPP* approach. We output the first k eigenvalues, $\widetilde{\boldsymbol{\lambda}}^{(k)} = (\tilde{\lambda}_1, \tilde{\lambda}_2, ..., \tilde{\lambda}_k)$, and the corresponding eigenvectors, $\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, ..., \tilde{\boldsymbol{u}}_k)$, under ϵ-differential privacy with the given graph topology A and parameters k, ϵ. We first compute the real values of eigenvalues $\boldsymbol{\lambda}^{(k)}$ and eigenvectors $\boldsymbol{u}_i (i \in [1, k])$ from the given graph adjacent matrix A (Line 1). Then we distribute the privacy parameter ϵ among $\boldsymbol{\lambda}^{(k)}$ and $\boldsymbol{u}_1, \boldsymbol{u}_2, ..., \boldsymbol{u}_k$ respectively as ϵ_0 and $\epsilon_1, \epsilon_2, ..., \epsilon_k$ where $\epsilon = \sum_{i=0}^{k} \epsilon_i$ (Line 2). With the derived the sensitivities for the eigenvalues $(GS_{\boldsymbol{\lambda}^{(k)}}(G))$ and each of the k eigenvectors $(GS_{\boldsymbol{u}_i}(G), i \in [1, k])$ from Results 1 and 2, next we follow Theorem 1 to calibrate Laplace noise and obtain the private answers $\widetilde{\boldsymbol{\lambda}}^{(k)}$ (Line 3) and $\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, ..., \tilde{\boldsymbol{x}}_k$ (Line 4). Finally we do a postprocess to normalize and orthogonalize $\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, ..., \tilde{\boldsymbol{x}}_k$ into $\tilde{\boldsymbol{u}}_1, \tilde{\boldsymbol{u}}_2, ..., \tilde{\boldsymbol{u}}_k$ following Theorem 4 (Line 5).

3.2 SBMF: Sampling from *BMF* Density

The *SBMF* approach to provide spectral analysis of network data is based on the sampling scheme proposed by Hoff [7] as an application of their recently-proposed technique of sampling from the matrix Bingham-von Mises-Fisher density (Definitions 3, 4). In [8], the authors investigated differentially private approximations to principle component analysis and also developed a method based on the general exponential mechanism [4]. In our work we focus on the eigen-decomposition of the 0-1 adjacency matrix (rather than the second moment matrix of the numerical data) and prove that the sampling scheme from the matrix Bingham-von Mises-Fisher density satisfies differential privacy through the general exponential mechanism (Theorem 2). The sampling scheme proposed by Hoff [7] ensures Lemma 1, with the purpose to build the latent factor model (Definition 4) for network data, i.e, to approximate inferences for U and Λ. We derive the privacy bounds of the output eigenvalues and eigenvectors following the sampling scheme respectively in Claims 1 and 2, based on Lemma 1. Then following the

Composition Theorem (Theorem 3), we come to the conclusion that the *SBMF* approach satisfies ϵ-differential privacy (Theorem 5).

Claim 1. *The sampling scheme which outputs $\boldsymbol{\lambda}^{(k)}$ satisfies ϵ_Λ-differential privacy where $\epsilon_\Lambda = k(\frac{2\tau^2}{2+\tau^2})^{3/2}$.*

Proof. We denote A and A' as the adjacent matrix of any neighboring graph G and G'. The calibrated noise to a function f from the Gaussian distribution $normal(0, \sigma^2)$, similar as that from the Laplace distribution, provides a $2\sigma G S_f$-differential privacy [1]. Based on Lemma 1, we have for each eigenvalue λ_i, the sampling scheme satisfies

$$\begin{aligned}\epsilon_{\lambda_i} &= 2\sigma G S_{\lambda_i} = 2(\frac{2\tau^2}{2+\tau^2})^{1/2}\{\tau^2 \boldsymbol{u}_i^T A \boldsymbol{u}_i/(2+\tau^2) - \tau^2 \boldsymbol{u}_i^T A' \boldsymbol{u}_i/(2+\tau^2)\} \\ &= 2(\frac{2\tau^2}{2+\tau^2})^{1/2}\frac{\tau^2}{2+\tau^2}\boldsymbol{u}_i^T(A-A')\boldsymbol{u}_i \leq (\frac{2\tau^2}{2+\tau^2})^{3/2}\end{aligned}$$

where the proof of $\boldsymbol{u}_i^T(A-A')\boldsymbol{u}_i \leq 1$ is straightforward. With the composition theorem (Theorem 3), $\epsilon_\Lambda = \sum_{i=1}^{k} \epsilon_{\lambda_i} = k(\frac{2\tau^2}{2+\tau^2})^{3/2}$.

Claim 2. *Given the graph G's adjacent matrix A, the sampling scheme which outputs U satisfies ϵ_U-differential privacy where $\epsilon_U = k^2\lambda_1$.*

Proof. The sampling scheme for U can be considered as an instance for the exponential mechanism(Theorem 2) with $\alpha = 1$ and $q(A, U) = tr(\Lambda U^T A U/2)$. We have

$$\begin{aligned}\Delta q(A,U) &= \left|tr(\Lambda U^T A U/2) - tr(\Lambda U^T A' U/2)\right| = \frac{1}{2}\left|tr(\Lambda U^T(A-A')U)\right| \\ &\leq \frac{1}{2}k\lambda_1\left|tr(U^T(A-A')U)\right| \leq \frac{1}{2}k^2\lambda_1.\end{aligned}$$

Following Theorem 2, we have $\epsilon_U = 2\alpha\Delta q(A, U) = k^2\lambda_1$.

Theorem 5. *The* SBMF *approach to computing the spectra, the first k eigenvalues and the corresponding eigenvectors of a given graph topology A satisfies $\epsilon = (\epsilon_\Lambda + \epsilon_U)$-differential privacy, where $\epsilon_\Lambda = k(\frac{2\tau^2}{2+\tau^2})^{3/2}$ and $\epsilon_U = \alpha k^2\lambda_1$.*

In this work, we take the prior parameter τ^2 as n, which is suggested by Hoff [7] since this is roughly the variance of the eigenvalues of an $n \times n$ matrix of independent standard normal noise. We illustrate the *SBMF* approach in Algorithm 2. In the Algorithm, the parameter α is used to change the privacy magnitude by changing ϵ_U (Theorems 2, 5). Given the input graph topology A and dimension parameter k, we acquire the eigenvalues $\widetilde{\Lambda}_k$ and corresponding eigenvectors $\widetilde{U}_k$ from the sampler application provided by Hoff [7] with input matrix αA. The output satisfies $\epsilon = (\epsilon_\Lambda + \epsilon_U)$-differential privacy following Theorem 5.

Algorithm 2. *SBMF: Sampling from BMF density approach*

Input: Graph adjacent matrix $A_{n\times n}$, privacy magnitude α and dimension parameter k

Output: The first k eigenvalues $\widetilde{\Lambda}_k$ and corresponding eigenvectors $\widetilde{U}_k$, which satisfies $\epsilon = (\epsilon_\Lambda + \epsilon_U)$-differential privacy.

1: Set the input matrix $Y = \alpha A$, the parameter $\tau^2 = n$ and the number of iterations t;
2: Acquire $\widetilde{\Lambda}_k$ and $\widetilde{U}_k$ from the sampler provided by Hoff [7] with the input matrix Y, the output satisfies $\epsilon = (\epsilon_\Lambda + \epsilon_U)$-differential privacy(Theorem 5);
3: Output $\widetilde{\Lambda}_k$ and $\widetilde{U}_k$

4 Empirical Evaluation

We conduct experiments to compare the performance of the two approaches, *LNPP* and *SBMF*, in producing the differentially private eigenvalues and eigenvectors. For the *LNPP*, we implement Algorithm 1. For the *SBMF*, we use the R-package provided by Hoff [7]. We use 'Enron' (147 nodes, 869 edges) data set that is derived from an email network [1] collected and prepared by the CALO Project. We take the dimension $k = 5$ since it has been suggested in previous literatures [9] that the first five eigenvalues and eigenvectors are sufficient to capture the main information of this graph. The first two rows in Table 1 show the eigenvalues and their corresponding eigen-gaps (Result 2).

4.1 Performance Comparison with $\alpha = 1$

In this section, we compare the performance of the *LNPP* approach with that of the *SBMF* approach in three aspects: the accuracy of eigenvalues, the accuracy of eigenvectors and the accuracy of graph reconstruction with the private eigenpairs. With $\tau^2 = n$ and $\alpha = 1$, we compute that $\epsilon_\lambda = 14$ and $\epsilon_U = 446$ following Claims 1 and 2. Therefore the *SBMF* approach satisfies $\epsilon = 460$ differential privacy following Theorem 5. On the other hand, the same ϵ is taken as the input for the *LNPP* approach. Different strategies have been proposed to address the ϵ distribution problem(Line 2 in Algorithm 1) in previous literatures [10, 11]. In our work, we just take one simple strategy, distributing ϵ as $\epsilon_0 = 10$ to the eigenvalues and $\epsilon_i = 90, (i \in [1, k])$ equally to each eigenvector. Therefore *LNPP* approach also satisfies $\epsilon = 460$ differential privacy.

For eigenvalues, we measure the output accuracy with the absolute error defined as $E_\Lambda = |\widetilde{\boldsymbol{\lambda}}^{(k)} - \boldsymbol{\lambda}^{(k)}|_1 = \sum_{i=1}^{k} |\widetilde{\lambda}_i - \lambda_i|$. The absolute errors E_Λ for *LNPP* and *SBMF* are respectively 0.9555 and 345.2301. One sample o eigenvalues In the third and fourth rows of Table 1, we show the output eigenvalues from the *LNPP* and the *SBMF* approaches. We can see that the *LNPP* outperforms the *SBMF* in more accurately capturing the original eigenvalues.

For eigenvectors, we define the absolute error as $E_U = |\widetilde{U}_k - U_k|_1$. E_U for *LNPP* and *SBMF* approaches are respectively 11.9989 and 13.4224. We also

[1] `http://www.cs.cmu.edu/~enron/`

Table 1. Eigenvalues Comparison

	λ_1	λ_2	λ_3	λ_4	λ_5
eigenvalue	17.8317	12.7264	10.6071	9.7359	9.5528
eigen-gap	5.1053	2.1193	0.8712	0.1832	0.1832
LNPP	18.1978	13.2191	10.6030	9.7311	9.4650
SBMF	107.8450	88.9362	76.1712	76.0596	56.6721

Table 2. Eigenvector Comparison

Approaches	E_U	$cos\langle\widetilde{\boldsymbol{u}}_i, \boldsymbol{u}_i\rangle = \widetilde{\boldsymbol{u}}_i' \cdot \boldsymbol{u}_i$				
		$\boldsymbol{u}_1$	$\boldsymbol{u}_2$	$\boldsymbol{u}_3$	$\boldsymbol{u}_4$	$\boldsymbol{u}_5$
LNPP	**11.9989**	**0.9591**	**0.7925**	0.4786	0.1217	0.1280
SBMF	13.4224	0.6605	0.6995	**0.7336**	**0.2921**	**0.4034**

define the cosine similarity to measure the accuracy of each private eigenvector as $cos\langle\widetilde{\boldsymbol{u}}_i, \boldsymbol{u}_i\rangle = \widetilde{\boldsymbol{u}}_i' \cdot \boldsymbol{u}_i (i \in [1, k])$. We show the detailed values of E_U and the cosine similarities in Table2. Note that the cosine value closer to 1 indicates better utility. We can see that *LNPP* generally outperforms *SBMF* in privately capturing eigenvectors that close to the original ones. Specifically, the *LNPP* approach is sensitive to eigen-gaps (second row in Table 1), i.e., it tends to show better utility when the eigen-gap is large such as for $\boldsymbol{u}_1$ and $\boldsymbol{u}_2$. Thus a better strategy will be distributing privacy parameter ϵ according to magnitudes of eigen-gaps, instead of the equal distribution.

The *SBMF* approach outputs much larger eigenvalues than the original ones. It does not tend to accurately approximate anyone of the original eigenvectors either. The reason is that *SBMF* approach is designed to provide a low rank spectral model for the original graph rather than approximating of the original eigenvalues and eigenvectors.

We consider the application of graph reconstruction using the differentially private first k eigenvalues and the corresponding eigenvectors. $A_k = \sum_{i=1}^{k} \lambda_i \boldsymbol{u}_i \boldsymbol{u}_i^T = U_k \Lambda_k U_k^T$ is commonly used as a rank-k approximation to the original graph topology A when A is not available for privacy reasons or A's rank is too large for analysis. Since A_k is not an 0/1 matrix, We discretize A_k as $\widetilde{A}_k^1$ by choosing the largest $2m$ entries as 1 and all others as 0 (so keeping the number of edges m the same as that of the original graph). We then compare the performance of the two approaches by the absolute reconstruction error defined as $\gamma = \|A - \widetilde{A}_k^1\|_F$.The γ values for *LNPP* and *SBMF* approaches are 47.7912 and 34.1760 respectively. We can see that the result of the *SBMF* approach outperforms the *LNPP*.

4.2 Performance Comparison with Varying α

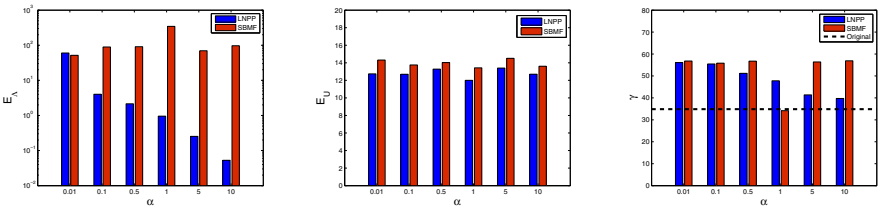
In this section, we change the privacy magnitude to additionally study the performance of the *LNPP* and *SBMF* approaches. α denotes the amplification factor of the privacy parameter ϵ used in section 4.1. We choose the value

Table 3. Comparison of two approaches for varying privacy magnitudes

	α	0.01	0.1	0.5	1	5	10
E_Λ	*LNPP*	60.1586	4.0160	2.1452	0.9555	0.2528	0.0527
	SBMF	51.6551	89.0678	90.9442	345.2301	69.6852	96.8904
E_U	*LNPP*	12.7419	13.2455	13.9874	11.9989	13.3967	12.7033
	SBMF	14.3155	13.7518	14.0238	13.4224	14.5114	13.6087
γ	*LNPP*	56.2139	55.4617	51.1859	47.4912	41.3763	39.7492
	SBMF	56.8155	55.8211	56.7450	34.1760	56.3715	56.9210

of α as $0.01, 0.1, 0.5, 1, 5, 10$ where the corresponding ϵ values are respectively $18.46, 58.6, 237, 460, 2244, 4474$ following Theorem 5.

We show the values of E_Λ, E_U and γ for the *LNPP* and the *SBMF* approaches in Table 3. The accuracy of the *LNPP* approach increases significantly with α for both the eigenvalues(E_Λ) and graph reconstruction (γ). Note that the greater the α, the weaker privacy protection, and hence the more utility preservation. However, the accuracy of eigenvectors measured by E_U is not changed much with α, as shown in Figure 1. This is because of the normalization of eigenvectors in the postprocess step. While the *SBMF* approach cannot accurately capture eigenvalues for any α value; as to graph reconstruction, the case of $\alpha = 1$ shows the best utility.



(a) E_Λ (b) E_U (c) γ

Fig. 1. Utility comparison for varying privacy magnitude

5 Conclusion

In this paper we have presented two approaches to enforcing differential privacy in spectral graph analysis. We apply and evaluate the Laplace Mechanism [1] and the exponential mechanism [4] on the differential privacy preserving eigen decomposition on the graph topology. In our future work, we will investigate how to enforce differential privacy for other spectral graph analysis tasks (e.g., spectral clustering based on graph's Laplacian and normal matrices). Nissim et al. [3] introduced a framework that calibrates the instance-specific noise with smaller magnitude than the worst-case noise based on the global sensitivity. We

will study the use of smooth sensitivity and explore how to better distribute privacy budget in the proposed *LNPP* approach. We will also study how different sampling strategies in the proposed *SBMF* approach may affect the utility preservation.

Acknowledgments. This work was supported in part by U.S. National Science Foundation IIS-0546027, CNS-0831204, CCF-0915059, and CCF-1047621.

References

1. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 265–284. Springer, Heidelberg (2006)
2. Dwork, C., Lei, J.: Differential privacy and robust statistics. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, pp. 371–380. ACM (2009)
3. Nissim, K., Raskhodnikova, S., Smith, A.: Smooth sensitivity and sampling in private data analysis. In: Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing, pp. 75–84. ACM (2007)
4. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science, pp. 94–103. IEEE (2007)
5. Stewart, G., Sun, J.: Matrix perturbation theory. Academic Press, New York (1990)
6. Garthwaite, P., Critchley, F., Anaya-Izquierdo, K., Mubwandarikwa, E.: Orthogonalization of vectors with minimal adjustment. Biometrika (2012)
7. Hoff, P.: Simulation of the Matrix Bingham-von Mises-Fisher Distribution, With Applications to Multivariate and Relational Data. Journal of Computational and Graphical Statistics 18(2), 438–456 (2009)
8. Chaudhuri, K., Sarwate, A., Sinha, K.: Near-optimal algorithms for differentially-private principal components. In: Proceedings of the 26th Annual Conference on Neural Information Processing Systems (2012)
9. Wu, L., Ying, X., Wu, X., Zhou, Z.: Line orthogonality in adjacency eigenspace with application to community partition. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, pp. 2349–2354. AAAI Press (2011)
10. Xiao, X., Bender, G., Hay, M., Gehrke, J.: ireduct: Differential privacy with reduced relative errors. In: Proceedings of the ACM SIGMOD International Conference on Management of Data (2011)
11. Wang, Y., Wu, X., Zhu, J., Xiang, Y.: On learning cluster coefficient of private networks. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (2012)