
Empirical Error Estimates for Graph Sparsification

Siyao Wang
University of California, Davis

Miles E. Lopes
University of California, Davis

Abstract

Graph sparsification is a well-established technique for accelerating graph-based learning algorithms, which uses edge sampling to approximate dense graphs with sparse ones. Because the sparsification error is random and unknown, users must contend with uncertainty about the reliability of downstream computations. Although it is possible for users to obtain conceptual guidance from theoretical error bounds in the literature, such results are typically impractical at a numerical level. Taking an alternative approach, we propose to address these issues from a data-driven perspective by computing *empirical error estimates*. The proposed error estimates are highly versatile, and we demonstrate this in four use cases: Laplacian matrix approximation, graph cut queries, graph-structured regression, and spectral clustering. Moreover, we provide two theoretical guarantees for the error estimates, and explain why the cost of computing them is manageable in comparison to the overall cost of a typical graph sparsification workflow.

1 INTRODUCTION

The scalability of graph-based algorithms in machine learning is often limited in applications that involve dense graphs with very large numbers of edges. For this reason, *graph sparsification* has become a well-established acceleration technique, which speeds up computations by replacing dense graphs with sparse approximations (Benczúr and Karger, 1996; Spielman and Teng, 2011). Furthermore, there are myriad applications that illustrate the popularity and flexibility of

graph sparsification, such as graph partitioning (Keller et al., 2014; Chen et al., 2022), clustering (Chen et al., 2016; Agarwal et al., 2022), solving linear systems (Spielman and Teng, 2004; Jambulapati and Sidford, 2021), graph-structured regression (Sadhanala et al., 2016; Calandriello et al., 2018), and deep learning (Hamilton et al., 2017; Zeng et al., 2020; Zheng et al., 2020).

Graph sparsification is commonly implemented in a randomized manner via edge sampling, which confronts the user with substantial uncertainty: The error produced by the sampling is both random and unknown, which raises doubts about the accuracy of results that rely on the sparsified graph. This uncertainty can also lead to less efficient computation, as users are inclined to “hedge their bets” with conservatively large sample sizes—undermining the benefit of sparsification.

To deal with these issues, it is necessary to estimate the error created by sparsification. Indeed, error estimates not only provide a gauge for the reliability of computations, but also help to avoid the inefficiency of excessive sampling. For example, error estimates can enable *incremental refinement*, which involves estimating the error of an inexpensive preliminary sparsified graph, and then sampling extra edges as needed until the estimated error falls below a target threshold. Hence, such an approach can help users to sample just enough edges to suit their purpose.

Up to now, the literature has generally addressed sparsification error from a theoretical standpoint. For instance, this is often done by deriving theoretical bounds on the runtimes of sparsified graph algorithms as a function of the error. However, such results tend to be inherently conservative, as they are often designed to hold uniformly over a large class of possible inputs. Making matters worse, such results typically involve unknown parameters or unspecified constants. Consequently, it can be infeasible to use theoretical error bounds in a way that is practical on a problem-specific basis.

Based on the issues just discussed, we propose to

address error estimation from a more data-driven perspective—by using bootstrap methods to compute *empirical error estimates* that only rely on the information acquired in the edge sampling process. As a result, this approach delivers error estimates that are adapted to the particular inputs at hand, avoiding the drawbacks of worst-case error analysis.

Our main contributions are summarized as follows: **(1)** To the best of our knowledge, this paper is the first to systematically develop empirical error estimates for graph sparsification. **(2)** We illustrate the flexibility of our error estimates in four use cases, including *Laplacian matrix approximation*, *graph cut queries*, *graph-structured regression*, and *spectral clustering*. All of these examples are supported by numerical experiments under a variety of conditions. **(3)** In two different contexts, we prove that the error estimates perform correctly in the limit of large problem sizes. Because we allow the number of graph vertices and edges to diverge simultaneously with the number of sampled edges, our theoretical results require in-depth analyses based on *high-dimensional central limit theorems*.

Preliminaries. We consider weighted undirected graphs $G = (V, E, w)$, with vertex set $V = \{1, \dots, n\}$, edge set $E \subset \{\{i, j\} | i, j \in V, i \neq j\}$, and weight function $w : E \rightarrow [0, \infty)$. The Laplacian matrix $L \in \mathbb{R}^{n \times n}$ of the graph G is defined by $L_{ii} = \sum_{\{i, l\} \in E} w(i, l)$, and $L_{ij} = -w(i, j)$ if $i \neq j$. Equivalently, if $\Delta_e \in \mathbb{R}^{n \times n}$ denotes the symmetric rank-1 matrix associated to an edge $e = \{i, j\}$ such that $x^\top \Delta_e x = (x_i - x_j)^2$ for all $x \in \mathbb{R}^n$, then L can be represented as

$$L = \sum_{e \in E} w(e) \Delta_e. \quad (1)$$

With regard to graph sparsification, we focus on settings where the sparsified graph $\hat{G} = (V, \hat{E}, \hat{w})$ is obtained by sampling N edges from G in an i.i.d. manner. On each sample, an edge e appears with a probability denoted by $p(e)$, and the sampled edge is incorporated into \hat{G} with weight $w(e)/(Np(e))$. (If an edge is sampled more than once, then the weights are added.) There are many choices of interest for the sampling probabilities, such as edge-weight sampling with $p(e) \propto w(e)$, and effective-resistance sampling with $p(e) \propto w(e) \text{tr}(L^+ \Delta_e)$, where L^+ is the Moore-Penrose inverse of L (Spielman and Srivastava, 2011). Importantly, *our proposed algorithms can be applied in practice without restricting the user's choice of sampling probabilities for generating the sparsified graph*.

The Laplacian matrix associated with \hat{G} is denoted by \hat{L} , and is referred to as a sparsified Laplacian. It should be emphasized that \hat{L} is a random matrix that can be interpreted as a sample average in the following way: If we define the collection of rank-1 matrices $\mathcal{Q} =$

$\{(w(e)/p(e))\Delta_e | e \in E\}$, and let $Q_1, \dots, Q_N \in \mathbb{R}^{n \times n}$ be i.i.d. samples from \mathcal{Q} such that $(w(e)/p(e))\Delta_e$ appears on each draw with probability $p(e)$, then \hat{L} can be represented as

$$\hat{L} = \frac{1}{N} \sum_{i=1}^N Q_i. \quad (2)$$

Furthermore, it can be checked that $\mathbf{E}(Q_1) = L$, ensuring unbiasedness, $\mathbf{E}(\hat{L}) = L$.

Problem setting. Graph sparsification is often intended for settings where G is so large or dense that accessing it incurs high communication costs, and only \hat{G} or \hat{L} can be stored in fast memory. For this reason, our error estimates will only rely on the sampled matrices Q_1, \dots, Q_N , and will not require access to G or L . Likewise, we view quantities depending on G and L as fixed unknown parameters.

Error functionals. To measure how well \hat{L} approximates L , we will consider a variety of scalar-valued error functionals, denoted $\psi(\hat{L}, L)$. For example, ψ could correspond to error in the Frobenius norm $\psi(\hat{L}, L) = \|\hat{L} - L\|_F$ or operator (spectral) norm $\psi(\hat{L}, L) = \|\hat{L} - L\|_{\text{op}}$. More generally, users can select ψ to suit their preferred notion of error in specific applications, as illustrated in Section 2.1.

For a given choice of ψ , our goal is to estimate the tightest possible upper bound on the *unobserved* random variable $\psi(\hat{L}, L)$ that holds with a prescribed probability, say $1 - \alpha$ with $\alpha \in (0, 1)$. Although this optimal bound is unknown, it can be defined precisely as the $(1 - \alpha)$ -quantile of $\psi(\hat{L}, L)$, denoted

$$q_{1-\alpha} = \inf \left\{ t \in \mathbb{R} \mid \mathbf{P}(\psi(\hat{L}, L) \leq t) \geq 1 - \alpha \right\}. \quad (3)$$

Accordingly, we aim to develop algorithms that can compute estimates $\hat{q}_{1-\alpha}$ of $q_{1-\alpha}$. Furthermore, these estimates are intended to perform well in three respects: (I) They should be flexible enough to handle many choices of ψ . (II) They should nearly match $q_{1-\alpha}$, so that the event $\{\psi(\hat{L}, L) \leq \hat{q}_{1-\alpha}\}$ holds with probability close to $1 - \alpha$. (III) They should be affordable to compute, so that the extra step of error estimation only modestly increases the overall cost of the user's workflow. In Sections 2-4, we demonstrate that all three desiderata are achieved by our proposed estimates.

Simultaneous confidence intervals. In addition to measuring error through various choices of $\psi(\hat{L}, L)$, it is natural in many applications to develop simultaneous confidence intervals (CIs) for unknown quantities depending on L . Denoting these quantities as $\theta_1(L), \dots, \theta_k(L)$, some important examples include graph cut values, and eigenvalues of L that are relevant in spectral clustering. (See Section 2.2 and Appendix A.) In such contexts, our approach can be

extended to construct CIs that simultaneously cover $\theta_1(L), \dots, \theta_k(L)$, while enjoying properties analogous to (I)-(III) above.

Related work and novelty. Over the last 15 years, randomized approximation algorithms have been widely adopted in many applications of machine learning and large-scale computing (Cormode et al., 2011; Mahoney et al., 2011; Woodruff, 2014; Martinsson and Tropp, 2020; Buluc et al., 2021). However, the research on empirical error estimation for these algorithms is still at a relatively early stage, and it has only just begun to accelerate within the last few years. A notable theme in this recent work is that statistical resampling techniques—such as the bootstrap, jackknife, and subsampling—have proven to be key ingredients in estimating the errors of many types of randomized algorithms. Examples of randomized algorithms for which statistical error estimation methods have been developed include low-rank approximation (Epperly and Tropp, 2024), regression (Lopes et al., 2018, 2020b; Zhang et al., 2023), matrix multiplication (Lopes et al., 2019, 2023), trace estimation (Martinsson and Tropp, 2020), Fourier features (Yao et al., 2023), and PCA (Lunde et al., 2021; Lopes et al., 2020a; Wang et al., 2024).

Within this growing line of research, the current paper is novel in several ways. Most importantly, our work is the first to specifically target graph sparsification, *which demands methodology and theory that are both new*. At a more technical level, our work is also differentiated in the way that we adapt resampling methods to our setting. In particular, for certain applications, we leverage a specialized type of resampling known as a “double bootstrap” (Chernick, 2011; Hall, 2013). In many classical statistical problems, it is known that a double bootstrap can substantially improve upon more basic bootstrap methods, but up to now, *its advantages have not been considered in the contemporary line of work on error estimation for randomized algorithms*. Our choice to use this approach in Section 2.1 is based on practical necessity, as we found that simpler resampling techniques led to unsatisfactory error estimates. Lastly, it is worth clarifying that although the enhancements provided by double bootstrapping do require a more technical implementation, the computational cost is not an obstacle in modern computing environments that are relevant to graph sparsification, as explained in Section 2.3.

Notation and terminology. If A is a finite set of real numbers and $\alpha \in (0, 1)$, then the *empirical* $(1 - \alpha)$ -quantile of A is denoted as $\text{quantile}(A; 1 - \alpha)$, which is the smallest $a_0 \in A$ such that $|\{a \in A : a \leq a_0\}|/|A| \geq 1 - \alpha$, where $|\cdot|$ refers to cardinality. If $q \geq 1$, then the ℓ_q norm of $v \in \mathbb{R}^d$ is $\|v\|_q = (\sum_{j=1}^d |v_j|^q)^{1/q}$, and

$\|v\|_\infty = \max_{1 \leq j \leq d} |v_j|$. If M is a symmetric real matrix, then $\lambda_1(M) \leq \lambda_2(M) \leq \dots$ refer to the sorted eigenvalues. To refer to the multinomial distribution based on tossing N balls into N bins with probabilities p_1, \dots, p_N , we write $\text{Mult.}(N; p_1, \dots, p_N)$.

2 METHODS

In this section, we present two algorithms and explain how they quantify the errors that arise from \hat{L} in several tasks. Section 2.1 focuses on quantile estimates for error functionals $\psi(\hat{L}, L)$, which can be used in Laplacian matrix approximation and graph-structured regression. Section 2.2 develops simultaneous CIs for the values of graph cuts and eigenvalues of L .

2.1 Error functionals

Recall that \hat{L} can be represented as $\hat{L} = \frac{1}{N} \sum_{i=1}^N Q_i$, where Q_1, \dots, Q_N are i.i.d. random matrices such that $\mathbf{E}(\hat{L}) = \mathbf{E}(Q_1) = L$. Letting $\psi(\hat{L}, L)$ denote a generic error functional, and letting $q_{1-\alpha}$ denote its $(1 - \alpha)$ -quantile, *our goal is to compute an estimate $\hat{q}_{1-\alpha}$ using only knowledge of Q_1, \dots, Q_N* . To develop the estimate, a bootstrap approach relies on a mechanism for generating approximate samples of the random variable $\psi(\hat{L}, L)$, so that $\hat{q}_{1-\alpha}$ can be constructed as the empirical $(1 - \alpha)$ -quantile of those approximate samples. But it turns out that even for simple choices of ψ , this approach can sometimes produce poor estimates of $q_{1-\alpha}$. In such situations, it is known in the bootstrap literature that better performance can often be achieved by using approximate samples of a suitably standardized version of $\psi(\hat{L}, L)$ (Hall, 2013, Ch.3). For this reason, we aim to generate approximate samples of the random variable $\zeta = (\psi(\hat{L}, L) - \hat{\mu})/\hat{\sigma}$, where $\hat{\mu}$ and $\hat{\sigma}^2$ denote estimates of $\mu = \mathbf{E}(\psi(\hat{L}, L))$ and $\sigma^2 = \text{var}(\psi(\hat{L}, L))$ that will be defined later. Specifically, if the approximate samples are denoted $\zeta_1^*, \dots, \zeta_B^*$, then they can be used to define the estimate $\hat{q}_{1-\alpha} = \text{quantile}(\hat{\mu} + \hat{\sigma}\zeta_1^*, \dots, \hat{\mu} + \hat{\sigma}\zeta_B^*; 1 - \alpha)$.

The main ideas for generating approximate samples of ζ are as follows. Since ζ can be viewed as a function of Q_1, \dots, Q_N , denoted $\zeta = \varphi(Q_1, \dots, Q_N)$, the standard bootstrap approach would be to randomly sample matrices Q_1^*, \dots, Q_N^* with replacement from Q_1, \dots, Q_N , and then define $\zeta^* = \varphi(Q_1^*, \dots, Q_N^*)$ as an approximate sample of ζ . However, this is not directly applicable in our context with a generic choice of ψ , because there are generally no explicit formulas for computing $\hat{\mu}$ and $\hat{\sigma}$ in terms of Q_1, \dots, Q_N , and hence, there are generally no explicit formulas for computing $\varphi(Q_1, \dots, Q_N)$. Nevertheless, an approximation $\hat{\varphi}$ to the function φ can also be developed via bootstrap sampling, and approximate samples of ζ can be defined

as $\zeta^* = \hat{\varphi}(Q_1^*, \dots, Q_N^*)$.

From an algorithmic standpoint, this way of defining ζ^* is more intricate than it might appear at first sight. A particularly important point is that computing ζ^* actually involves a “second level” of bootstrap sampling. This is because the quantity $\hat{\varphi}(Q_1^*, \dots, Q_N^*)$ will be computed by sampling from the (already re-sampled) matrices Q_1^*, \dots, Q_N^* with replacement. Another consideration is that even though it is natural to think about the proposed method in terms of sampling from sets of matrices with replacement, it is possible to implement this more efficiently by reweighting matrices with coefficients drawn from certain multinomial distributions, as shown in Algorithm 1 below.

Algorithm 1 (Quantile estimate for error functionals)

Input: Number of bootstrap samples $B \geq 1$, a number $\alpha \in (0, 1)$, and the matrices \hat{L}, Q_1, \dots, Q_N .

for $b = 1, \dots, B$ **in parallel do:**

- Generate $(W_1^*, \dots, W_N^*) \sim \text{Mult.}(N; \frac{1}{N}, \dots, \frac{1}{N})$.
 - Compute $\varepsilon_b^* = \psi(\hat{L}^*, \hat{L})$, where $\hat{L}^* = \frac{1}{N} \sum_{i=1}^N W_i^* Q_i$.
- for** $b' = 1, \dots, B$ **in parallel do:**
- Generate $(W_1^{**}, \dots, W_N^{**}) \sim \text{Mult.}(N; \frac{W_1^*}{N}, \dots, \frac{W_N^*}{N})$.
 - Compute $\varepsilon_{b'}^{**} = \psi(\hat{L}^{**}, \hat{L}^*)$, where $\hat{L}^{**} = \frac{1}{N} \sum_{i=1}^N W_i^{**} Q_i$.

end for

- Compute $\hat{\mu}_b^* = \frac{1}{B} \sum_{b'=1}^B \varepsilon_{b'}^{**}$ as well as

$$\hat{\sigma}_b^* = \sqrt{\frac{1}{B} \sum_{b'=1}^B (\varepsilon_{b'}^{**} - \hat{\mu}_b^*)^2},$$

$$\zeta_b^* = \frac{1}{\hat{\sigma}_b^*} (\varepsilon_b^* - \hat{\mu}_b^*). \quad (\text{If } \hat{\sigma}_b^* = 0, \text{ put } \zeta_b^* = 0.)$$

end for

Compute $\hat{\mu} = \frac{1}{B} \sum_{b=1}^B \varepsilon_b^*$ and $\hat{\sigma} = \sqrt{\frac{1}{B} \sum_{b=1}^B (\varepsilon_b^* - \hat{\mu})^2}$.

Output: $\hat{q}_{1-\alpha} = \text{quantile}(\hat{\mu} + \hat{\sigma} \zeta_1^*, \dots, \hat{\mu} + \hat{\sigma} \zeta_B^*; 1 - \alpha)$.

Graph-structured regression. To illustrate other choices of error functionals beyond norms such as $\psi(\hat{L}, L) = \|\hat{L} - L\|_F$ or $\psi(\hat{L}, L) = \|\hat{L} - L\|_{\text{op}}$, we now discuss an application to graph-structured regression (Sadhanala et al., 2016; Calandriello et al., 2018). In this context, the user has a vector of observations $y = (y_1, \dots, y_n)$ associated with the n vertices of G , and the unsparified version of the task is to use y and L to estimate a vector of unknown parameters $\beta^\circ \in \mathbb{R}^n$. Ordinarily, the estimate $r(L)$ for β° is computed as a solution to an optimization problem of the form $r(L) = \arg\min_{\beta \in \mathbb{R}^n} \{\ell(y, \beta) + \tau \beta^\top L \beta\}$. Here, $\ell(y, \beta)$ measures the goodness of fit between y and a candidate

vector β , and $\tau \beta^\top L \beta$ penalizes vectors β that do not respect the structure of G , with $\tau \geq 0$ being a tuning parameter. In situations where L is very large or dense, the previously cited works have proposed approximating $r(L)$ with $r(\hat{L}) = \arg\min_{\beta \in \mathbb{R}^n} \{\ell(y, \beta) + \tau \beta^\top \hat{L} \beta\}$. However, the accuracy of $r(\hat{L})$ as an approximation to $r(L)$ is unknown. To address this issue, Algorithm 1 can be applied with an error functional such as $\psi(\hat{L}, L) = \|r(\hat{L}) - r(L)\|_2$, and we illustrate this empirically in Section 4.

2.2 Simultaneous confidence intervals

The second aspect of our proposed methodology deals with CIs for various quantities associated with L . We discuss this first in the context of graph cut values, and then explain how the same approach can be extended to the eigenvalues of L .

Background on graph cuts. By definition, a *cut* in a graph $G = (V, E, w)$ is a partition of the vertex set V into two disjoint subsets, and the *value* of the cut is the sum of the weights of the edges that connect vertices in the two subsets. Recalling the notation $V = \{1, \dots, n\}$, every cut can be identified with a binary vector $x \in \{0, 1\}^n$, where the two subsets of vertices are $\{i \in V | x_i = 0\}$ and $\{i \in V | x_i = 1\}$. This representation of a cut allows its value, denoted $C(x)$, to be computed as $C(x) = x^\top L x = \sum_{\{i,j\} \in E} w(i,j)(x_i - x_j)^2$.

Because many fundamental characteristics of graphs can be computed in terms of cut values, it is common for algorithms to be formulated in terms of a collection of “cut query” vectors $\mathcal{C} \subset \{0, 1\}^n$, which is specific to the user’s task. Moreover, there is a well-established line of research on using edge sampling to efficiently approximate the cut values of large or dense graphs (Benczúr and Karger, 1996, 2015; Andoni et al., 2016; Arora and Upadhyay, 2019). Hence, this amounts to approximating $\{C(x) | x \in \mathcal{C}\}$ using the cut values of a sparsified graph $\{\hat{C}(x) | x \in \mathcal{C}\}$, where we define $\hat{C}(x) = x^\top \hat{L} x$. To quantify the approximation error, we propose an algorithm that uses $\{\hat{C}(x) | x \in \mathcal{C}\}$ to build simultaneous CIs for $\{C(x) | x \in \mathcal{C}\}$.

Simultaneous CIs for graph cut values. The starting point for our approach is to consider the $(1 - \alpha)$ -quantile $q_{1-\alpha}$ of the unobserved random variable

$$\xi = \max_{x \in \mathcal{C}} \frac{|\hat{C}(x) - C(x)|}{\hat{\sigma}(x)},$$

where $\hat{\sigma}^2(x)$ is an estimate of $\text{var}(\hat{C}(x))$ to be detailed shortly. It is straightforward to check that if $q_{1-\alpha}$ were known, then the (theoretical) CIs defined by $\mathcal{I}_{1-\alpha}(x) = [\hat{C}(x) \pm \hat{\sigma}(x) q_{1-\alpha}]$ would have a simultaneous coverage probability $\mathbf{P}(\cap_{x \in \mathcal{C}} \{C(x) \in \mathcal{I}_{1-\alpha}(x)\})$

that is at least $1 - \alpha$. The crux of the problem is to construct a quantile estimate $\hat{q}_{1-\alpha}$, which will allow us to use practical intervals defined by $\hat{\mathcal{I}}_{1-\alpha}(x) = [\hat{C}(x) \pm \hat{\sigma}(x)\hat{q}_{1-\alpha}]$. Despite the seeming simplicity of this definition, the theoretical problem of demonstrating that these intervals have a simultaneous coverage probability close to $1 - \alpha$ is quite involved when the number of queries $|\mathcal{C}|$ is large. Nevertheless, we will show in Theorem 1 that the intervals can succeed even when $|\mathcal{C}|$ is allowed to diverge asymptotically.

Analogously to Algorithm 1, the main idea for constructing $\hat{q}_{1-\alpha}$ here is to generate approximate samples ξ_1^*, \dots, ξ_B^* of ξ , and then define $\hat{q}_{1-\alpha} = \text{quantile}(\xi_1^*, \dots, \xi_B^*; 1 - \alpha)$. For this purpose, it is natural to define the quantities $\hat{C}_i(x) = x^\top Q_i x$ so that we have $\hat{C}(x) = \frac{1}{N} \sum_{i=1}^N \hat{C}_i(x)$, and we may estimate $\text{var}(\hat{C}(x))$ using $\hat{\sigma}^2(x) = \frac{1}{N} \sum_{i=1}^N (\hat{C}_i(x) - \hat{C}(x))^2$. In this notation, Algorithm 2 generates approximate samples having the form

$$\xi^* = \max_{x \in \mathcal{C}} \frac{|\hat{C}^*(x) - \hat{C}(x)|}{\hat{\sigma}(x)},$$

where $\hat{C}^*(x) = \frac{1}{N} \sum_{i=1}^N \hat{C}_i^*(x)$ and $\hat{C}_1^*(x), \dots, \hat{C}_N^*(x)$ are drawn with replacement from $\hat{C}_1(x), \dots, \hat{C}_N(x)$. (The exceptional case that $\hat{\sigma}(x) = 0$ for some $x \in \mathcal{C}$ is handled by treating $|\hat{C}^*(x) - \hat{C}(x)|/\hat{\sigma}(x)$ as 0, because in this case we must have $\hat{C}^*(x) = \hat{C}(x)$.)

Algorithm 2 (Simultaneous CIs for graph cut values)

Input: Number of bootstrap samples $B \geq 1$, a number $\alpha \in (0, 1)$, and the set $\{\hat{C}_i(x) | x \in \mathcal{C}, 1 \leq i \leq N\}$.

Compute the estimates $\hat{C}(x) = \frac{1}{N} \sum_{i=1}^N \hat{C}_i(x)$ and $\hat{\sigma}^2(x) = \frac{1}{N} \sum_{i=1}^N (\hat{C}_i(x) - \hat{C}(x))^2$ for each $x \in \mathcal{C}$.

for $b = 1, \dots, B$ **in parallel do:**

- Generate $(W_1^*, \dots, W_N^*) \sim \text{Mult.}(N; \frac{1}{N}, \dots, \frac{1}{N})$.
- Compute $\xi_b^* = \max_{x \in \mathcal{C}} \frac{1}{\hat{\sigma}(x)} |\frac{1}{N} \sum_{i=1}^N (W_i^* - 1)\hat{C}_i(x)|$.

end for

Compute $\hat{q}_{1-\alpha} = \text{quantile}(\xi_1^*, \dots, \xi_B^*; 1 - \alpha)$.

Output: The collection of CIs $\{\hat{\mathcal{I}}_{1-\alpha}(x) | x \in \mathcal{C}\}$ defined by $\hat{\mathcal{I}}_{1-\alpha}(x) = [\hat{C}(x) \pm \hat{\sigma}(x)\hat{q}_{1-\alpha}]$.

Remarks. Notably, this algorithm does not require a second level of bootstrap sampling, which is an important contrast with Algorithm 1. The main reason for this simplification is that we can estimate $\mathbf{E}(\hat{C}(x))$ and $\text{var}(\hat{C}(x))$ using explicit functions of $\hat{C}_1(x), \dots, \hat{C}_N(x)$, whereas it was not possible to estimate $\mathbf{E}(\psi(\hat{L}, L))$ and $\text{var}(\psi(\hat{L}, L))$ in the same manner for a general choice of ψ . One more significant point is that $\hat{q}_{1-\alpha}$ in Algorithm 2 can be used to extract

information about the maximal cut query value $C_{\max} = \max_{x \in \mathcal{C}} C(x)$ and minimal cut query value $C_{\min} = \min_{x \in \mathcal{C}} C(x)$, which are of interest in many graph partitioning problems. Specifically, C_{\max} is covered by $[\max_{x \in \mathcal{C}} \{\hat{C}(x) - \hat{\sigma}(x)\hat{q}_{1-\alpha}\}, \max_{x \in \mathcal{C}} \{\hat{C}(x) + \hat{\sigma}(x)\hat{q}_{1-\alpha}\}]$ with a probability at least as large as the simultaneous coverage probability of $\{C(x) \in \hat{\mathcal{I}}_{1-\alpha}(x) | x \in \mathcal{C}\}$. The same holds, mutatis mutandis, for C_{\min} .

Simultaneous CIs in spectral clustering. One of the most well known machine learning tasks involving graph Laplacians is spectral clustering (von Luxburg, 2007), which uses Laplacian eigenvectors to construct low-dimensional representations of data that allow clusters to be distinguished more effectively. Because the Laplacians in spectral clustering tend to be dense, sparsification has been advocated as a way to improve computational efficiency (Chakeri et al., 2016; Chen et al., 2016; Sun and Zanetti, 2019). On the other hand, sparsification can also distort the clustering results.

As an illustration of how Algorithm 2 can be adapted to address this issue, we focus on one of the most pivotal steps in clustering: the selection of the number of clusters. Often, this choice is made by searching for a prominent gap among the bottom eigenvalues of a Laplacian, and then choosing the number of clusters to be the number of eigenvalues that fall below that gap (von Luxburg, 2007). However, when a sparsified Laplacian is used, this selection technique becomes more nuanced, because if the gaps between eigenvalues are too sensitive to the chance variation from sparsification, then they may be unreliable indicators for the correct number of clusters.

To quantify the uncertainty, it is possible to construct simultaneous CIs, say $\hat{\mathcal{I}}_1, \dots, \hat{\mathcal{I}}_r$, for the eigenvalues $\lambda_1(L) \leq \dots \leq \lambda_r(L)$, where $r \geq 2$ is a number that the user believes is safely above the correct number of clusters. If there is an index $j \in \{1, \dots, r\}$ such that a clear gap exists between the upper endpoint of $\hat{\mathcal{I}}_j$ and the lower endpoint of $\hat{\mathcal{I}}_{j+1}$, then this gives more credible evidence that j clusters are present, because in this case, the gap cannot be easily explained away by the sparsification error.

The intervals $\hat{\mathcal{I}}_1, \dots, \hat{\mathcal{I}}_r$ are constructed as follows. First, we put $\hat{\mathcal{I}}_1 = \{0\}$, since $\lambda_1(L)$ is always 0. Next, the definition of ξ_b^* in Algorithm 2 can simply be replaced by $\xi_b^* = \max_{2 \leq j \leq r} |\lambda_j(\hat{L}^*)/\lambda_j(\hat{L}) - 1|$, where the j th quantity in the max is set to 0 when $\lambda_j(\hat{L}) = 0$, because this implies $\lambda_j(\hat{L}^*) = 0$. In turn, $\hat{q}_{1-\alpha}$ in Algorithm 2 can be used to define the CIs $\hat{\mathcal{I}}_j = [\lambda_j(\hat{L})/(1 + \hat{q}_{1-\alpha}), \lambda_j(\hat{L})/(1 - \hat{q}_{1-\alpha})]$ for $j \in \{2, \dots, r\}$, with the upper endpoint interpreted as ∞ in the unlikely case $\hat{q}_{1-\alpha} \geq 1$. Lastly, in Appendix A, we present

several empirical examples showing that these CIs provide effective guidance in selecting the number of clusters.

2.3 Computational efficiency

We now address the computational efficiency of the proposed algorithms. Given that Algorithm 1 uses a double bootstrap, it is important to begin by providing some historical context. Because double bootstrapping was first developed in the 1980s (Efron, 1983; Beran, 1988), the computing environments of that time were ill-suited to its structure, and it acquired a long-held reputation of being computationally intensive. However, due to major technological shifts, this perception is becoming increasingly outdated. In particular, there are three aspects of our algorithms that make them affordable in modern computing environments: (1) low communication cost, (2) high parallelism, and (3) incremental refinement.

Low communication cost. As was discussed in the introduction, graph sparsification is often intended for settings where G is too large or dense to be stored in fast memory. In these situations, the communication cost of accessing G in order to generate \hat{L} is often of greater concern than the flop count of subsequent computations on \hat{L} (Martinsson and Tropp, 2020, §16.2). (This is sometimes also referred to as an instance of the “memory wall” problem (Gholami et al., 2024).) Meanwhile, it is crucial to recognize that Algorithms 1 and 2 *do not require any additional access to G* , since they only rely on the samples used to produce \hat{L} . Hence, when the communication cost to access G is high, it is less likely that error estimation will be a bottleneck.

High parallelism. Another factor that counts in favor of Algorithms 1 and 2 is that bootstrap sampling is “embarrassingly parallel”, which is to say that all of the samples within a given loop can be computed independently. Moreover, this is especially favorable as cloud and GPU computing are becoming ubiquitous. In fact, the Python Package Index now includes a GPU-compatible package that is specifically designed to perform bootstrap sampling (Nowotny, 2024).

With regard to Algorithm 1, some additional attention should be given to the fact that its two loops are nested—which might appear to restrict the benefit of parallelism. However, the nested structure is manageable for two reasons. First, in many settings, it is sufficient to take only $B \sim 50$ bootstrap samples in each loop, and this is demonstrated empirically in Section 4. Second, there are established techniques in GPU computing for parallelizing nested loops.

Processing cost and incremental refinement.

Whereas the communication cost of Algorithms 1 and 2 is likely to be much less than that of the overall graph sparsification workflow, a comparison of processing cost (e.g. flop count) involves more considerations. Due to the high parallelism of Algorithms 1 and 2, the main driver of their runtimes will be the processing cost of one iteration of each loop. Often, this cost will be similar to that of the main task involving \hat{L} . For example, in graph-structured regression, where the main task is to compute $r(\hat{L})$, the cost of computing $\psi(\hat{L}^*, \hat{L}) = \|r(\hat{L}^*) - r(\hat{L})\|_2$ and $\psi(\hat{L}^{**}, \hat{L}^*) = \|r(\hat{L}^{**}) - r(\hat{L}^*)\|_2$ in Algorithm 1 will be dominated by the cost of computing $r(\hat{L}^{**})$ and $r(\hat{L}^*)$, which is proportional to the cost of computing $r(\hat{L})$. Similarly, for cut queries, if the user’s main task with \hat{L} is to compute the maximal approximate cut value $\max_{x \in \mathcal{C}} \hat{C}(x)$, then this will be similar to the cost of computing ξ_b^* in Algorithm 2.

Based on the reasoning above, the runtimes of Algorithms 1 and 2 are expected to be similar to the runtime of the main task involving \hat{L} , which in turn, is expected to be less than the communication time of accessing G . So, from this standpoint, error estimation is not expected to substantially increase the overall cost of the workflow. But as it turns out, there is one further technique that can be used to make the cost of error estimation even lower—which is *incremental refinement*. The first step of this technique is to generate a “rough” preliminary instance of \hat{L} based on a small sample size, say N_0 . If we let $q_{1-\alpha}(N)$ denote the $(1-\alpha)$ -quantile of $\psi(\hat{L}, L)$ based on a generic sample size N , then the key idea is that an estimate $\hat{q}_{1-\alpha}(N_0)$ can be obtained inexpensively, and then it can be used to “forecast” what larger sample size $N_1 \gg N_0$ is needed to *refine* the sparsified Laplacian so that $q_{1-\alpha}(N_1)$ is below a target threshold. In other words, the error estimation is accelerated because it is faster to run Algorithms 1 and 2 when there are N_0 sampled edges, rather than N_1 . This process of “forecasting” N_1 is based on an easily implemented type of extrapolation that is well established in the bootstrap literature (Bickel and Yahav, 1988), and is detailed in Appendix B. In particular, we show empirically that the rule is effective when N_0 is 10 times smaller than N_1 , enabling substantial speedups.

3 THEORETICAL RESULTS

In this section, we present two results that establish the theoretical validity of Algorithms 1 and 2 in the limit of large graphs with a diverging number of vertices, $n \rightarrow \infty$. The first result shows that Algorithm 2 produces CIs that simultaneously cover the exact cut values $\{C(x) | x \in \mathcal{C}\}$ with a probability that is asymptotically correct. Likewise, the second result shows

that when $\psi(\hat{L}, L) = \|\hat{L} - L\|_F^2$, Algorithm 1 produces a quantile estimate that upper bounds $\|\hat{L} - L\|_F^2$ with a probability converging to the correct value. The proofs of both results require extensive theoretical analysis based on *high-dimensional central limit theorems*, and are deferred to Appendices E and F.

Setting for theoretical results. Our theoretical results are framed in terms of a sequence of weighted undirected graphs $G_n = (V_n, E_n, w_n)$ indexed by the number of vertices $n = 1, 2, \dots$, such that V_n , E_n , and w_n are allowed to vary as functions of n . For each n , we assume that the sparsified Laplacian \hat{L}_n is obtained by drawing N_n edges from G_n in an i.i.d manner via edge-weight sampling. Lastly, the number of bootstrap samples B_n and the set of cut queries \mathcal{C}_n may also vary with n .

Simultaneous CIs for cut values. Some notation is needed for our first result. Let $w_n(E_n) = \sum_{e \in E_n} w_n(e)$ denote the total weight of G_n , and for any binary cut vector x , let $\underline{C}(x) = x^\top L_n x / w_n(E_n)$ be its standardized value, which satisfies $0 \leq \underline{C}(x) \leq 1$. Lastly, for a set $\mathcal{C}_n \subset \{0, 1\}^n$, define the theoretical quantity $\eta(\mathcal{C}_n) = \min_{x \in \mathcal{C}_n} \{\underline{C}(x)(1 - \underline{C}(x))\}$.

Theorem 1. *As $n \rightarrow \infty$, suppose that $N_n \rightarrow \infty$ and $B_n \rightarrow \infty$, as well as $\log(N_n|\mathcal{C}_n|)^5 = o(\sqrt{N_n}\eta(\mathcal{C}_n))$. Then, for any fixed $\alpha \in (0, 1)$, the confidence intervals $\{\hat{\mathcal{I}}_{1-\alpha}(x) | x \in \mathcal{C}_n\}$ produced by Algorithm 2 have a simultaneous coverage probability that satisfies the following limit as $n \rightarrow \infty$,*

$$\mathbf{P}\left(\bigcap_{x \in \mathcal{C}_n} \{C(x) \in \hat{\mathcal{I}}_{1-\alpha}(x)\}\right) \rightarrow 1 - \alpha. \quad (4)$$

Remarks. A valuable feature of this result is that it can handle situations where \mathcal{C}_n is a large set, since the cardinality $|\mathcal{C}_n|$ is only constrained through a polylogarithmic function, $\log(N_n|\mathcal{C}_n|)^5 = o(\sqrt{N_n}\eta(\mathcal{C}_n))$. This means that Algorithm 2 can succeed in high-dimensional inference problems, because $|\mathcal{C}_n|$ (i.e. the number of unknown parameters) may diverge.

With regard to the role of $\eta(\mathcal{C}_n)$, a notable point is that its value is allowed to approach 0 as $n \rightarrow \infty$, as long as $\eta(\mathcal{C}_n)$ is of larger order than $\log(N_n|\mathcal{C}_n|)^5 / \sqrt{N_n}$. In essence, values of $\eta(\mathcal{C}_n)$ near 0 occur when \mathcal{C}_n contains a cut x whose value is negligible compared to $w_n(E_n)$, or when the two graph components induced by x have negligible weight compared to $w_n(E_n)$. The reason that such cuts need to be excluded is technical, because if $\underline{C}(x)$ is close to 0 or 1, then the random variable $x^\top \hat{L}_n x$ is nearly degenerate—which interferes with establishing limiting distributions for statistics that depend on $x^\top \hat{L}_n x$. To briefly mention some explicit examples that are covered by Theorem 1, it is known that for Erdős-Renyi graphs with average de-

gree γ , our assumption involving $\eta(\mathcal{C}_n)$ holds with high probability as $n \rightarrow \infty$, provided that all $x \in \mathcal{C}_n$ are bi-sections (i.e. $\|x\|_1 = n/2$), γ is sufficiently large, and $\log(|\mathcal{C}_n|)^5 = o(\sqrt{N_n})$ (Dembo et al., 2017). We will also show empirically that Algorithm 2 can work well for natural graphs when all cuts in \mathcal{C}_n are drawn uniformly at random and $|\mathcal{C}_n| = n$.

Error estimates for the Frobenius norm. Our next result provides a guarantee on the performance of Algorithm 1 when $\psi(\hat{L}_n, L_n) = \|\hat{L}_n - L_n\|_F^2$. Here, the choice to use $\|\cdot\|_F^2$ rather than $\|\cdot\|_F$ is essentially a matter of mathematical convenience, because if $\hat{q}_{1-\alpha}$ is an estimated quantile for $\|\hat{L}_n - L_n\|_F^2$, then $\sqrt{\hat{q}_{1-\alpha}}$ has equivalent performance for $\|\hat{L}_n - L_n\|_F$.

Theorem 2. *As $n \rightarrow \infty$, suppose that $N_n \rightarrow \infty$, $B_n \rightarrow \infty$, $n/N_n \rightarrow 0$, and $\|\mathbf{d}_n\|_\infty / \|\mathbf{d}_n\|_2 \rightarrow 0$ hold, where $\mathbf{d}_n \in \mathbb{R}^n$ contains the diagonal entries of L_n . Then, for any fixed $\alpha \in (0, 1)$, the quantile estimate $\hat{q}_{1-\alpha}$ produced by Algorithm 1 satisfies the following limit as $n \rightarrow \infty$,*

$$\mathbf{P}(\|\hat{L}_n - L_n\|_F^2 \leq \hat{q}_{1-\alpha}) \rightarrow 1 - \alpha. \quad (5)$$

Remarks. The condition that the sample size N_n be of larger order than the number of vertices n is typical in the analysis of graph sparsification algorithms. As for the vector of degrees \mathbf{d}_n , the condition $\|\mathbf{d}_n\|_\infty / \|\mathbf{d}_n\|_2 \rightarrow 0$ has the interpretation that no single vertex dominates the entire graph with respect to degrees.

4 EMPIRICAL RESULTS

This section investigates the empirical performance of our proposed error estimation methods in three applications: graph cut queries, Laplacian matrix approximation, and graph-based regression. A fourth application to spectral clustering is covered in Appendix A.

Graphs. The experiments were based on five graphs: **Citations**, **DIMACS**, **Genes**, **Howard**, and **M14**, which are detailed in Appendix C, along with information about the computing resources used in our experiments. **Citations** represents the co-citation graph of scientific papers from a section of arXiv (Rossi and Ahmed, 2015). **DIMACS** is a benchmarking graph from the DIMACS Implementation Challenge (Bader et al., 2011). **Genes** is a human gene regulatory network (Davis and Hu, 2011). **Howard** is a student social network (Traud et al., 2012; Rossi and Ahmed, 2015). **M14** is the Mycielskian14 graph, which is part of a test suite for benchmarking graph algorithms (Davis and Hu, 2011).

From each of the graphs mentioned above, we constructed a corresponding graph G by randomly sam-

Table 1: Results for Algorithms 1 and 2 in several error estimation tasks. Under the heading of ‘graph cuts’, we report the observed value of the simultaneous coverage probability $\mathbf{P}(\cap_{x \in \mathcal{C}} \{C(x) \in \hat{\mathcal{I}}_{1-\alpha}(x)\})$, with $1 - \alpha$ being 90% or 95%. In the columns to the right of ‘graph cuts’, we report the observed value of $\mathbf{P}(\psi(\hat{L}, L) \leq \hat{q}_{1-\alpha})$ for the three choices of ψ in a similar manner.

G	$ E $	sampling	graph cuts		$\ \hat{L} - L\ _F$		$\ \hat{L} - L\ _{\text{op}}$		$\ r(\hat{L}) - r(L)\ _2$	
			90%	95%	90%	95%	90%	95%	90%	95%
Citations	218,835	EW	88.3	93.2	89.6	94.1	89.4	93.3	90.1	94.3
		ER	87.8	92.8	92.0	95.7	91.6	95.7	93.3	96.8
		AER	88.1	93.4	90.1	94.6	89.5	94.7	92.4	96.0
DIMACS	1,799,532	EW	90.1	94.9	88.6	94.9	89.4	93.9	89.3	94.8
		ER	90.4	94.6	89.0	93.3	87.9	93.7	90.3	94.4
		AER	89.3	93.5	88.8	93.5	88.5	93.0	90.0	94.7
Genes	743,712	EW	87.7	93.4	87.1	93.2	88.2	93.5	88.2	94.7
		ER	87.7	93.7	90.4	94.3	89.7	94.4	87.0	92.6
		AER	87.5	93.4	88.8	94.5	88.9	93.3	89.8	95.8
Howard	107,264	EW	88.7	94.6	89.8	94.2	90.0	94.7	88.4	94.5
		ER	87.3	92.2	90.9	94.5	93.1	96.5	87.7	93.9
		AER	89.7	94.5	90.7	94.5	91.5	94.8	89.3	93.6
M14	172,195	EW	90.3	94.7	91.2	96.1	90.8	94.6	87.3	93.7
		ER	89.2	94.0	90.9	95.2	92.5	95.9	88.4	93.5
		AER	89.6	94.0	89.0	94.2	92.0	95.3	89.4	94.2

pling $n = 2,000$ vertices according to their degrees (without replacement) and retaining all edges among the sampled vertices. The resulting number of edges $|E|$ for each graph G is reported in Table 1, where we use the same name to refer to G and the original graph it was drawn from.

Experiment settings. We considered three sampling schemes for sparsifying each G : edge-weight sampling (EW), effective-resistance sampling (ER), and approximate effective-resistance sampling (AER) (Spielman and Srivastava, 2011; Lebron, 2025). Whereas EW and ER were defined in the introduction, the definition of AER is more involved and is discussed in Appendix C. For the five choices of G and three choices of edge sampling scheme, we generated 1,000 sparsified Laplacians \hat{L} , yielding 15,000 in total. The number of sampled edges N for constructing \hat{L} was chosen to be 10% of the total number of edges, $N = |E|/10$.

For every realization of \hat{L} , we applied Algorithms 1 and 2 in four error estimation tasks: simultaneous CIs for graph cut values, as well as quantile estimation for $\psi(\hat{L}, L) = \|\hat{L} - L\|_F$, $\psi(\hat{L}, L) = \|\hat{L} - L\|_{\text{op}}$, and $\psi(\hat{L}, L) = \|r(\hat{L}) - r(L)\|_2$. With regard to the number of bootstrap samples in Algorithm 1, we used $B = 50$ for the outer loop and $B = 30$ for the inner loop. For Algorithm 2, we used $B = 50$. Under the heading of ‘graph cuts’ in Table 1, we report the observed value of the simultaneous coverage probability $\mathbf{P}(\cap_{x \in \mathcal{C}} \{C(x) \in \hat{\mathcal{I}}_{1-\alpha}(x)\})$, and in a similar manner, we report the observed value of $\mathbf{P}(\psi(\hat{L}, L) \leq \hat{q}_{1-\alpha})$ for the three choices of ψ , where the desired confidence

level $1 - \alpha$ is either 90% or 95%. The observed probabilities were computed by averaging over the 1,000 trials in each setting.

There are a few more details to mention about cut queries and graph-structured regression. The set of cuts $\mathcal{C} \subset \{0, 1\}^n$ was selected by independently generating 2,000 random vectors whose entries were i.i.d. Bernoulli(1/2) random variables. Next, for the graph-structured regression task on page 4, we adopted the following setting considered in Sadhanala et al. (2016): The vector of observations $y \in \mathbb{R}^n$ was generated from the Gaussian distribution $N(\beta^\circ, \zeta^2 I)$, where the mean $\beta^\circ \in \mathbb{R}^n$ was obtained by averaging 20 (unit norm) eigenvectors of L corresponding to the smallest 20 eigenvalues, and the scalar variance parameter was $\zeta^2 = \frac{1}{n} \sum_{i=1}^n (\beta_i^\circ - \beta^\circ)^2$, with $\beta^\circ = \frac{1}{n} \sum_{i=1}^n \beta_i^\circ$. Also, the loss function was taken as $\ell(y, \beta) = \|y - \beta\|_2^2$, and the tuning parameter was set to $\tau = 0.01$.

Discussion of empirical results. Table 1 captures the performance of our proposed algorithms in 120 distinct settings—corresponding to five choices of G , three choices of edge sampling, four choices of task, and two choices of confidence level. Thus, both the quality and consistency of the empirical results are excellent, as the observed probabilities match the desired confidence level $1 - \alpha$ to within about 2% in all but a few settings. We also show in Appendix A that simultaneous CIs for the eigenvalues of L exhibit similar performance in the context of spectral clustering.

Computational efficiency. The sizes of the five graphs used in the previous experiments were lim-

ited by a number of factors, such as the need to perform thousands of Monte Carlo trials, and compute ground truth errors involving unsparisified Laplacians. To assess the computational efficiency of error estimation, it is of interest to consider a runtime experiment involving a much larger graph. For this purpose, we used the M20 (Mycielskian20) graph (Davis and Hu, 2011), which is part of the same benchmarking suite as M14, and contains $n = 786,431$ nodes and $|E| = 2,710,370,560$ edges. Storing this graph in a 3-column CSV file with $|E|$ rows requires more than 42 GB, where an edge connecting nodes i and j with weight $w(i, j)$ is saved as the row vector $(i, j, w(i, j))$. In particular, this graph is too large to be stored in the RAM of a typical laptop, and presents a situation where graph sparsification is a practical option for dealing with limited memory.

Taking the approach of incremental refinement described in Section 2.3, we generated an initial sparsified Laplacian with $N_0 \approx 0.02|E|$ sampled edges. (See Appendix B for additional experiments demonstrating the effectiveness of incremental refinement with such a choice of N_0 .) Due to the large size of M20, we used an approximate form of EW sampling that takes advantage of the fact that all the edge weights of M20 are equal. Under exact EW sampling, the counts for the sampled edges would be a random vector drawn from a Multinomial distribution, corresponding to tossing N_0 balls into $|E|$ bins, each with probability $1/|E|$. Since the entries of such a random vector are approximately independent Poisson(0.02) random variables, it is computationally simpler to divide the full graph into a series of small “blocks” that can fit into RAM, and independently sample the edge counts as independent Poisson(0.02) random variables within each block.

The blockwise edge sampling was performed on a laptop with 16 GB of RAM by referring to the full graph as a `tabularTextDatastore` object in MATLAB, which is a type of object that allows for the blocking to be automated. After this was done, the initial sparsified Laplacian \hat{L} , and the matrices Q_1, \dots, Q_{N_0} were stored implicitly using sampled edge counts, so that memory need not be allocated for $n \times n$ matrices. Next, we applied Algorithm 1 to estimate the 90% quantile of $\psi(\hat{L}, L) = \|\hat{L} - L\|_F$, with $B = 30$ iterations for the inner loop and $B = 50$ iterations for the outer loop. Without using any parallelization for these loops, the overall runtime to obtain the quantile estimate was approximately 7 hours.

To place this runtime into context, we proceeded to the second stage of the incremental refinement approach, which involved generating a “refined” sparsified Laplacian based on $N_1 \approx 0.1|E|$ sampled edges. The edge sampling in this stage was performed in the same man-

ner as in the previous stage and took approximately 25 hours. We did not perform any additional tasks with this refined sparsified Laplacian, but if we did, it would have clearly increased the overall runtime of the workflow beyond 25 hours. This shows that the error estimation process increased the runtime of the workflow by at most $7/25 = 28\%$. Moreover, this does not reflect the straightforward speedup that could be obtained by running either of the loops in Algorithm 1 in parallel. For instance, if the outer loop were distributed across 8 processors with the inner loop still being run sequentially, the error estimation process would only increase the runtime of the workflow by at most $(7/8)/25 = 3.5\%$.

Code. The code for Algorithms 1 and 2 is available at the repository <https://github.com/sy-www/Error-Estimates-Graph-Sparsification>.

5 CONCLUSION

Due to the fact that graph sparsification has had far-reaching impact in machine learning and large-scale computing, our work has the potential to enhance many applications by providing users with practical error estimates. Indeed, considering that this is the first paper to develop a systematic way to estimate graph sparsification error, there is a substantial opportunity to adapt our approach to applications beyond the four that we have already presented here. Furthermore, the possibility of such extensions is underscored by our empirical results, which show that the error estimates perform reliably across a substantial range of conditions, corresponding to different graphs, edge sampling schemes, and error metrics. Lastly, we have also provided two theoretical performance guarantees that hold in a high-dimensional asymptotic setting where n , $|E|$, and N diverge simultaneously.

Acknowledgements

The authors gratefully acknowledge partial support from DOE grant DE-SC0023490.

References

- Agarwal, A., Khanna, S., Li, H., and Patil, P. (2022). Sublinear algorithms for hierarchical clustering. *Advances in Neural Information Processing Systems*, 35:3417–3430.
- Andoni, A., Chen, J., Krauthgamer, R., Qin, B., Woodruff, D. P., and Zhang, Q. (2016). On sketching quadratic forms. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pages 311–319.

- Arcones, M. A. (1995). A Bernstein-type inequality for U-statistics and U-processes. *Statistics & Probability Letters*, 22(3):239–247.
- Arora, R. and Upadhyay, J. (2019). On differentially private graph sparsification and applications. *Advances in Neural Information Processing Systems*.
- Bader, D. A., Meyerhenke, H., Sanders, P., and Wagner, D. (2011). 10th DIMACS implementation challenge-graph partitioning and graph clustering.
- Benczúr, A. A. and Karger, D. R. (1996). Approximating s-t minimum cuts in $\tilde{O}(n^2)$ time. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, pages 47–55.
- Benczúr, A. A. and Karger, D. R. (2015). Randomized approximation schemes for cuts and flows in capacitated graphs. *SIAM Journal on Computing*, 44(2):290–319.
- Beran, R. (1988). Prepivoting test statistics: a bootstrap view of asymptotic refinements. *Journal of the American Statistical Association*, 83(403):687–697.
- Bickel, P. J. and Doksum, K. A. (2015). *Mathematical Statistics: Basic Ideas and Selected Topics, volume I*. Chapman and Hall/CRC.
- Bickel, P. J. and Yahav, J. A. (1988). Richardson extrapolation and the bootstrap. *Journal of the American Statistical Association*, 83(402):387–393.
- Buluc, A., Kolda, T., Wild, S., Anitescu, M., Degenaro, A., Jakeman, J., Kamath, Ramakrishnan. K., K., Lopes, M. E., Martinsson, P.-G., Myers, K., Nelson, J., Restrepo, J., Seshadri, C., Vrabie, D., Wohlberg, B., Wright, S., Yang, C., and Zwart, P. (2021). Randomized algorithms for scientific computing (RASC). *U. S. Department of Energy, Office of Scientific and Technical Information (OSTI)*.
- Calandriello, D., Lazaric, A., Koutis, I., and Valko, M. (2018). Improved large-scale graph learning through ridge spectral sparsification. In *International Conference on Machine Learning*, pages 688–697.
- Chakeri, A., Farhidzadeh, H., and Hall, L. O. (2016). Spectral sparsification in spectral clustering. In *23rd International Conference on Pattern Recognition*, pages 2301–2306.
- Chen, J., Sun, H., Woodruff, D., and Zhang, Q. (2016). Communication-optimal distributed clustering. *Advances in Neural Information Processing Systems*.
- Chen, L., Kyng, R., Liu, Y. P., Peng, R., Gutenberg, M. P., and Sachdeva, S. (2022). Maximum flow and minimum-cost flow in almost-linear time. In *IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–623.
- Chernick, M. R. (2011). *Bootstrap Methods: A Guide for Practitioners and Researchers*. John Wiley & Sons.
- Chernozhuokov, V., Chetverikov, D., Kato, K., and Koike, Y. (2022). Improved central limit theorem and bootstrap approximations in high dimensions. *Annals of Statistics*, 50(5):2562–2586.
- Cormode, G., Garofalakis, M., Haas, P. J., Jermaine, C., et al. (2011). Synopses for massive data: Samples, histograms, wavelets, sketches. *Foundations and Trends® in Databases*, 4(1–3):1–294.
- Davis, T. A. and Hu, Y. (2011). The university of Florida sparse matrix collection. <https://sparse.tamu.edu/Mycielski/mycielskian14>
<https://sparse.tamu.edu/Mycielski/mycielskian20>
https://sparse.tamu.edu/Belcastro/human/_gene2.
- Dembo, A., Montanari, A., and Sen, S. (2017). Extremal cuts of sparse random graphs. *Annals of Probability*, 45(2):1190 – 1217.
- Efron, B. (1983). Estimating the error rate of a prediction rule: improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331.
- Epperly, E. N. and Tropp, J. A. (2024). Efficient error and variance estimation for randomized matrix computations. *SIAM Journal on Scientific Computing*, 46(1):A508–A528.
- Gholami, A., Yao, Z., Kim, S., Hooper, C., Mahoney, M. W., and Keutzer, K. (2024). AI and memory wall. *IEEE Micro*, pages 1–5.
- Hall, P. (2013). *The Bootstrap and Edgeworth Expansion*. Springer.
- Hamilton, W., Ying, Z., and Leskovec, J. (2017). Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*.
- Huang, K. H., Liu, X., Duncan, A., and Gandy, A. (2023). A high-dimensional convergence theorem for U-statistics with applications to kernel-based testing. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 3827–3918.
- Jambulapati, A. and Sidford, A. (2021). Ultrasparse ultrasparsifiers and faster Laplacian system solvers. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms*, pages 540–559.
- Kelly, M., Longjohn, R., and Nottingham, K. (2025). The UCI Machine Learning Repository. <https://archive.ics.uci.edu>.

- Kelner, J. A., Lee, Y. T., Orecchia, L., and Sidford, A. (2014). An almost-linear-time algorithm for approximate max flow in undirected graphs, and its multicommodity generalizations. In *Proceedings of the 2014 Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 217–226.
- Kunegis, J. (2013). KONECT – The Koblenz Network Collection.
<http://konect.cc/networks/ca-cit-HepTh/>.
- Lebron, R. G. (2025). Fast Effective Resistances.
<https://www.cs.cmu.edu/~jkoutis/SpectralAlgorithms.htm>.
- Lopes, M., Wang, S., and Mahoney, M. (2018). Error estimation for randomized least-squares algorithms via the bootstrap. In *International Conference on Machine Learning*, pages 3217–3226.
- Lopes, M. E. (2022). Central limit theorem and bootstrap approximation in high dimensions: Near $1/\sqrt{n}$ rates via implicit smoothing. *Annals of Statistics*, 50(5):2492–2513.
- Lopes, M. E., Erichson, N. B., and Mahoney, M. (2020a). Error estimation for sketched SVD via the bootstrap. In *International Conference on Machine Learning*, pages 6382–6392.
- Lopes, M. E., Erichson, N. B., and Mahoney, M. W. (2023). Bootstrapping the operator norm in high dimensions: Error estimation for covariance matrices and sketching. *Bernoulli*, 29(1):428–450.
- Lopes, M. E., Wang, S., and Mahoney, M. W. (2019). A bootstrap method for error estimation in randomized matrix multiplication. *Journal of Machine Learning Research*, 20(39):1–40.
- Lopes, M. E., Wu, S., and Lee, T. C. (2020b). Measuring the algorithmic convergence of randomized ensembles: The regression setting. *SIAM Journal on Mathematics of Data Science*, 2(4):921–943.
- Lunde, R., Sarkar, P., and Ward, R. (2021). Bootstrapping the error of Oja’s algorithm. *Advances in Neural Information Processing Systems*, 34:6240–6252.
- Mahoney, M. W. et al. (2011). Randomized algorithms for matrices and data. *Foundations and Trends® in Machine Learning*, 3(2):123–224.
- Martinsson, P.-G. and Tropp, J. A. (2020). Randomized numerical linear algebra: Foundations and algorithms. *Acta Numerica*, 29:403–572.
- Mycielski, J. (1955). Sur le coloriage des graphs. In *Colloquium Mathematicae*, volume 3, pages 161–162.
- Nazarov, F. (2003). On the maximal perimeter of a convex set in \mathbb{R}^n with respect to a Gaussian measure. In *Geometric Aspects of Functional Analysis: Israel Seminar 2001-2002*, pages 169–187.
- Nowotny, M. (2024). Recombinator - Statistical Resampling in Python.
<https://pypi.org/project/recombinator/>.
- Red, V., Kelsic, E. D., Mucha, P. J., and Porter, M. A. (2011). Comparing community structure to characteristics in online collegiate social networks. *SIAM review*, 53(3):526–543.
- Rossi, R. and Ahmed, N. (2015). The network data repository with interactive graph analytics and visualization. <https://networkrepository.com/socfb-Howard90.php>
<https://networkrepository.com/C2000-9.php>
https://networkrepository.com/ca_cit_HepTh.php.
- Sadhanala, V., Wang, Y.-X., and Tibshirani, R. (2016). Graph sparsification approaches for Laplacian smoothing. In *International Conference on Artificial Intelligence and Statistics*, pages 1250–1259.
- Spielman, D. A. and Srivastava, N. (2011). Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926.
- Spielman, D. A. and Teng, S.-H. (2004). Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the Thirty-Sixth Annual ACM Symposium on Theory of Computing*, pages 81–90.
- Spielman, D. A. and Teng, S.-H. (2011). Spectral sparsification of graphs. *SIAM Journal on Computing*, 40(4):981–1025.
- Sun, H. and Zanetti, L. (2019). Distributed graph clustering and sparsification. *ACM Transactions on Parallel Computing*, 6(3):1–23.
- Traud, A. L., Mucha, P. J., and Porter, M. A. (2012). Social structure of Facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180.
- van der Vaart, A. W. (2000). *Asymptotic Statistics*. Cambridge.
- von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing*, 17:395–416.
- Wainwright, M. J. (2019). *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge.
- Wang, L., Zhang, Z., and Dobriban, E. (2024). Inference in randomized least squares and PCA via normality of quadratic forms. *arXiv:2404.00912*.
- Woodruff, D. P. (2014). Sketching as a tool for numerical linear algebra. *Foundations and Trends® in Theoretical Computer Science*, 10(1–2):1–157.

- Yao, J., Erichson, N. B., and Lopes, M. E. (2023). Error estimation for random Fourier features. In *International Conference on Artificial Intelligence and Statistics*, pages 2348–2364.
- Zeng, H., Zhou, H., Srivastava, A., Kannan, R., and Prasanna, V. (2020). GraphSAINT: Graph sampling based inductive learning method. In *International Conference on Learning Representations*.
- Zhang, Z., Lee, S., and Dobriban, E. (2023). A framework for statistical inference via randomized algorithms. *arXiv:2307.11255*.
- Zheng, C., Zong, B., Cheng, W., Song, D., Ni, J., Yu, W., Chen, H., and Wang, W. (2020). Robust graph representation learning via neural sparsification. In *International Conference on Machine Learning*, pages 11458–11468.

Checklist

1. For all models and algorithms presented, check if you include:
 - (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes (Sections 1 and 2)
 - (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Yes (Section 2.3)
 - (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes
2. For any theoretical claim, check if you include:
 - (a) Statements of the full set of assumptions of all theoretical results. Yes (Section 3)
 - (b) Complete proofs of all theoretical results. Yes (Appendices D-F)
 - (c) Clear explanations of any assumptions. Yes (Section 3)
3. For all figures and tables that present empirical results, check if you include:
 - (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes
 - (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes (Section 4 and Appendices A, B, C)
 - (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes. The paper provides appropriate statistical information about the experiments. (Section 4 and Appendices A-C)
- (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes (Appendix C)
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
 - (a) Citations of the creator If your work uses existing assets. Yes (References)
 - (b) The license information of the assets, if applicable. Yes (Appendix C)
 - (c) New assets either in the supplemental material or as a URL, if applicable. Not Applicable
 - (d) Information about consent from data providers/curators. Not Applicable
 - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
 - (a) The full text of instructions given to participants and screenshots. Not Applicable
 - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable
 - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

Supplementary material

The appendices are organized as follows: Appendix A covers an application to spectral clustering. Appendix B presents experiments illustrating the performance of incremental refinement. Appendix C provides additional details about the design of the experiments and computing resources. Appendix D presents the notation necessary for the proofs. Appendices E and F contain the proofs of Theorems 1 and 2 respectively. Appendix G provides background results used in the proofs.

A Empirical results on spectral clustering

In this section, we examine the performance of the simultaneous CIs for the eigenvalues of L in spectral clustering.

Data. The results are based on a synthetic dataset labeled as **Mixture** and two natural datasets labeled as **Beans** and **Images**. **Mixture** was constructed from 1000 total samples, with 200 being drawn from each of five Gaussian distributions in \mathbb{R}^6 whose covariance matrices were all equal to the identity matrix, and whose mean vectors were $(0, 0, 0, 0, 0, 0)$, $(5, 5, 5, 0, 0, 0)$, $(0, 5, 5, 5, 0, 0)$, $(0, 0, 5, 5, 5, 0)$ and $(0, 0, 0, 5, 5, 5)$. Regarding the natural datasets, **Beans** and **Images** correspond to the Dry Bean and Image Segmentation datasets from the UCI Machine Learning Repository (Kelly et al., 2025). **Beans** is derived from a set of beans (observations) in 7 categories, with all beans having 16 associated features. We extracted the observations from 3 categories, “Bombay”, “Dermason” and “Seker”, and uniformly sampled 500 observations from each of these categories. Lastly, **Images** is based on a collection of outdoor images in 7 categories, with each image having 19 features. We extracted all the observations from 4 categories, “brickface”, “foliage”, “path” and “sky” images, with all of these categories having the same number of 330 observations. For each dataset, we applied the `rescale()` function in MATLAB with the default settings. Lastly, we calculated the two top eigenvectors from the sample covariance matrix of each dataset, and plotted the observations in 2 dimensions based on their coordinates with respect to these eigenvectors, as shown in Figure 1.

Graphs. We adopted a commonly used approach to spectral clustering that involves assigning a vertex to each observation, and assigning a weighted edge to each pair of observations x and x' , where the weight value is given by the Gaussian kernel $\exp(-\frac{1}{2\delta^2}\|x - x'\|_2^2)$. (The bandwidth parameter δ was set to 0.2 for **Mixture** and 0.3 for both **Beans** and **Images**.) In this way, each of the three datasets above induces a fully connected graph G with associated Laplacian L . In particular, the pairs of values $(n, |E|)$ for the numbers of vertices and edges are $(1000, 499500)$ for **Mixture**, $(1500, 1124250)$ for **Beans**, and $(1320, 870540)$ for **Images**.

Experiment design. Since ER sampling and AER sampling are specifically designed to preserve spectral properties of L (Spielman and Srivastava, 2011), whereas EW sampling is not, we focused on ER sampling and AER sampling. (See Appendix C for background on AER.)

For each of the graphs associated with **Mixture**, **Beans**, and **Images**, we generated 1000 realizations of \hat{L} using both ER and AER sampling, and employed the method outlined in Section 2.2 to construct simultaneous CIs for $\lambda_1(L) \leq \dots \leq \lambda_{15}(L)$. Table 2 shows the observed simultaneous coverage probabilities $\mathbf{P}(\cap_{j=1}^{15} \{\lambda_j(L) \in \hat{\mathcal{I}}_j\})$ based on desired confidence levels of 90% and 95%, which were computed by averaging over the 1000 trials in each setting. A display of the CIs and eigenvalues of L are given in Figure 1. For clarity of presentation, we only plotted a single representative CI at each index, corresponding to one whose center was nearly equal to the median of the centers of all the 1000 intervals. Also, for clarity, Figure 1 only displays the intervals corresponding to the 7 bottom Laplacian eigenvalues and a confidence level of 95%.

Discussion of empirical results. An intended feature of the experiments is that the clustering problems corresponding to **Mixture**, **Beans**, and **Images** have increasing levels of difficulty (as can be seen in Figure 1), which allows us to see the performance of the CIs in a range of conditions. Table 2 shows that the observed simultaneous coverage probabilities agree well with the desired confidence levels in all three problems. Also, the largest gaps among the CIs coincide with the correct number of clusters in all three problems—which demonstrates that the intervals can provide practical guidance to users in selecting the number of clusters. An especially good illustration of this occurs in the case of **Beans**, where there are large gaps between the *centers* of the 5th, 6th, and 7th CIs, but the gaps between their relevant *endpoints* are much smaller. In other words, this is a case where a user might be tempted to conclude that 5 or 6 clusters are present based only on the eigenvalues of \hat{L} (i.e. when error estimation is not used), whereas the CIs guard against these incorrect conclusions.

Table 2: Observed simultaneous coverage probabilities $\mathbf{P}(\cap_{j=1}^{15} \{\lambda_j(L) \in \hat{\mathcal{I}}_j\})$.

Dataset	ER		AER	
	90th	95th	90th	95th
Mixture	90.3	95.7	91.6	95.0
Beans	90.2	95.0	93.3	97.2
Images	90.0	94.2	89.2	94.3

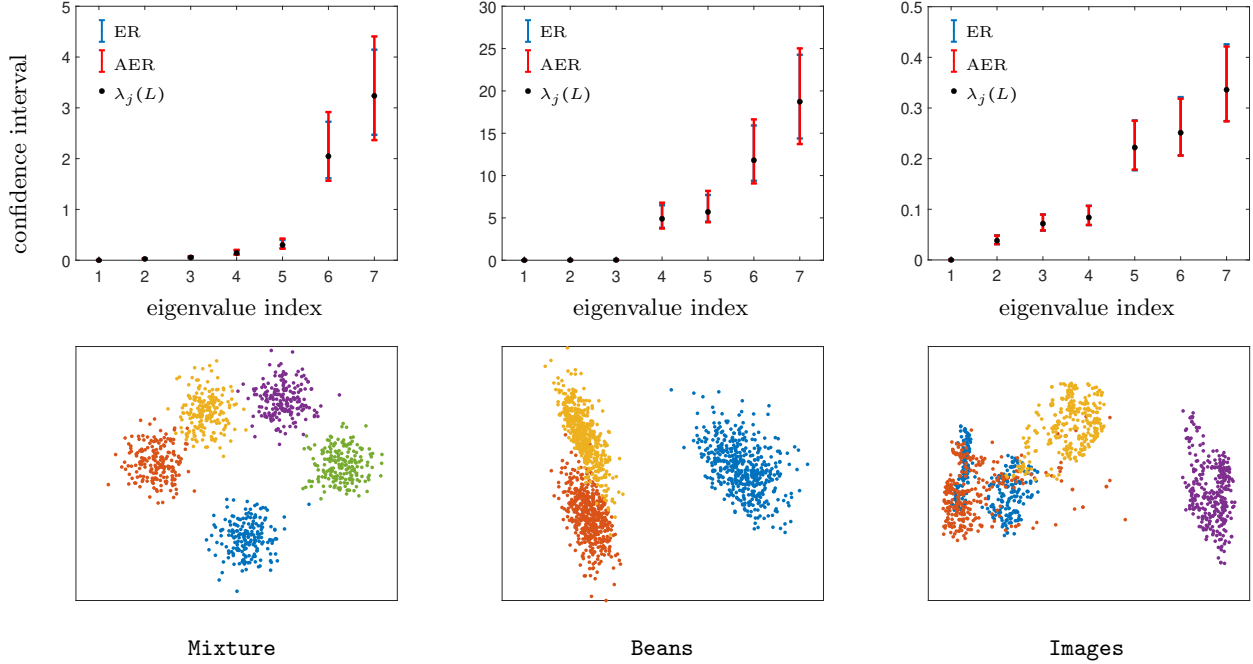


Figure 1: Scatter plots and simultaneous CIs.

B Empirical results on incremental refinement

In this section, we illustrate the performance of the incremental refinement technique discussed in Section 2.3. The experiments are based on the graphs **Genes**, **M14**, **Howard**, **Citations**, and **DIMACS**, as well as the same choices of ψ and sampling schemes covered in Section 4. To reduce the number of plots, only the confidence level $1 - \alpha = 95\%$ was considered.

Experiment design. Here, we follow the notation introduced in Section 2.3. For each graph and sampling scheme, we generated 1000 sparsified Laplacians \hat{L} based on $N_0 = 0.02|E|$ sampled edges, and applied Algorithm 1 to obtain 1000 corresponding quantile estimates $\hat{q}_{1-\alpha}(N_0)$. To construct estimates $\hat{q}_{1-\alpha}(N)$ for all $N \geq N_0$ by extrapolating from $\hat{q}_{1-\alpha}(N_0)$, we used the rule defined by $\hat{q}_{1-\alpha}(N) = \sqrt{N_0/N} \hat{q}_{1-\alpha}(N_0)$, which is based on the intuition that fluctuations of the entries of \hat{L} should have a $1/\sqrt{N}$ scaling with respect to N . We refer to (Bickel and Yahav, 1988) for further background on the use of extrapolation rules to reduce the cost of bootstrapping.

In all cases, the average of $\hat{q}_{1-\alpha}(N)$ over all 1000 trials is plotted in Figures 2-6 as a function of N using a solid line, where N ranges between $0.02|E|$ and $0.2|E|$. The variability $\hat{q}_{1-\alpha}(N)$ is indicated by dashed lines, which are plotted 1 standard deviation above and below the solid curve. (Note that for the ER and AER sampling schemes, the curves tend to overlap in many cases, making only the curves for AER visible.) Also, all the plots were put on a common scale by dividing all curves in a given plot by the value of the highest curve at $N_0 = 0.02|E|$. Lastly, as a substitute for the ground truth value of $q_{1-\alpha}(N)$, we computed the empirical 95% quantile of the 1000 values of $\psi(\hat{L}, L)$ at $N \in \{0.05|E|, 0.1|E|, 0.2|E|\}$, and these values are marked with large dots.

Discussion of empirical results. The accuracy of the extrapolated estimates $\hat{q}_{1-\alpha}(N)$ is judged by how well

the curves agree with the large dots of the same color. Overall, Figures 2-6 show that the estimates perform well, considering that in most cases the dots are within about one standard deviation of the corresponding solid curve. The stability of the estimates is also notable, as the standard deviation is generally small in proportion to the height of the solid curve. Lastly, and perhaps most importantly, the curves remain accurate up to $N = 0.2|E|$ even though they were extrapolated from a sample size $N_0 = 0.02|E|$ that is *10 times smaller*. This indicates that the incremental refinement technique has the potential to substantially improve computational efficiency, because error estimation can be performed more quickly when the number of sampled edges is small.

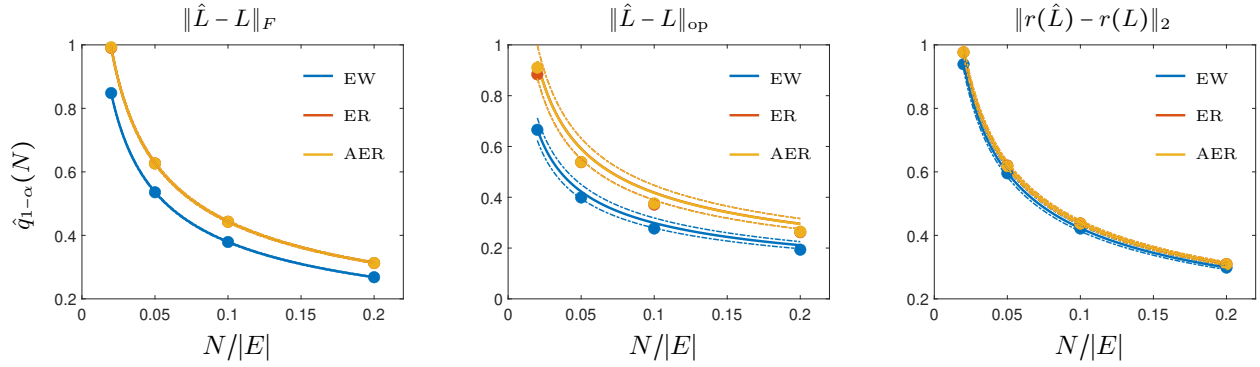


Figure 2: Results on incremental refinement for Genes.

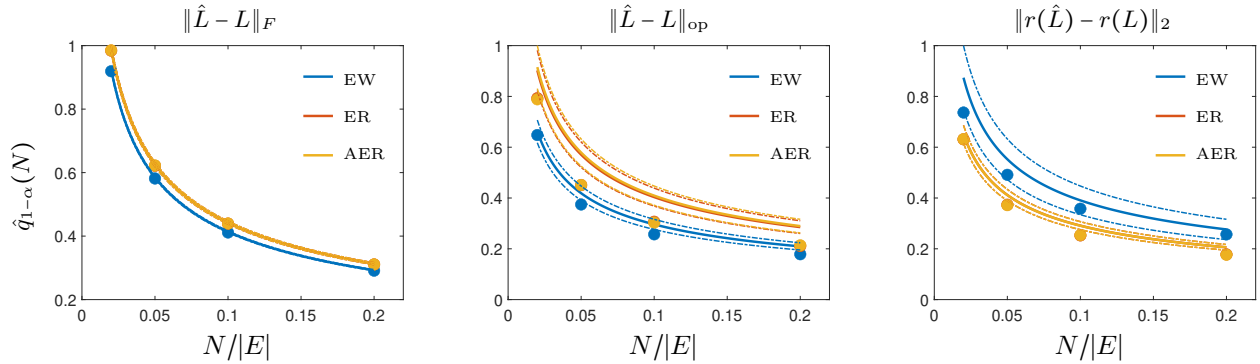


Figure 3: Results on incremental refinement for M14.

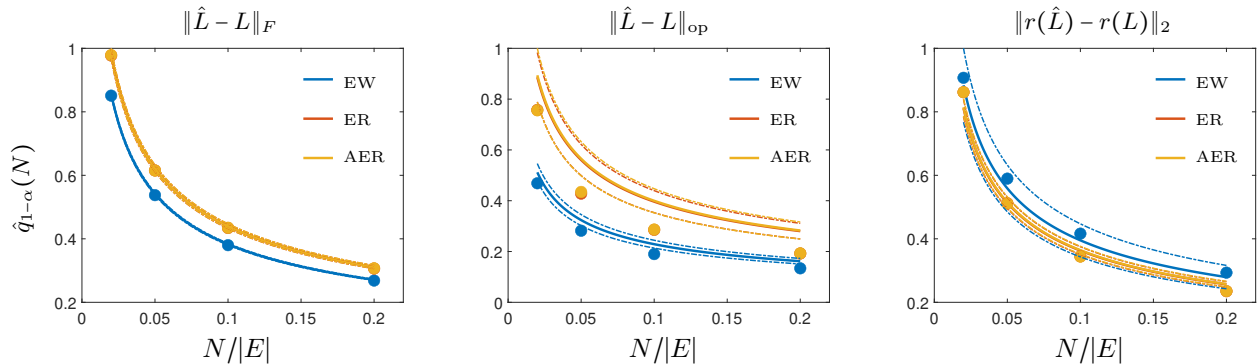


Figure 4: Results on incremental refinement for Howard.

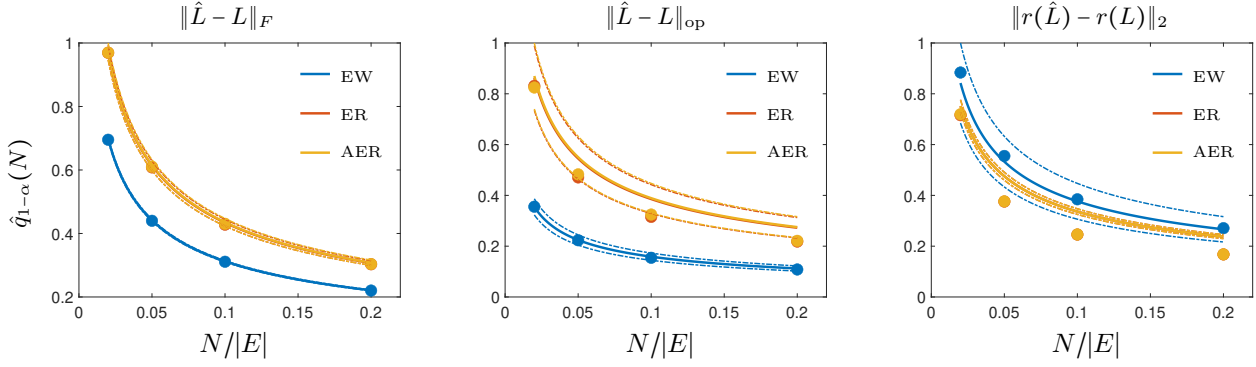


Figure 5: Results on incremental refinement for Citation.

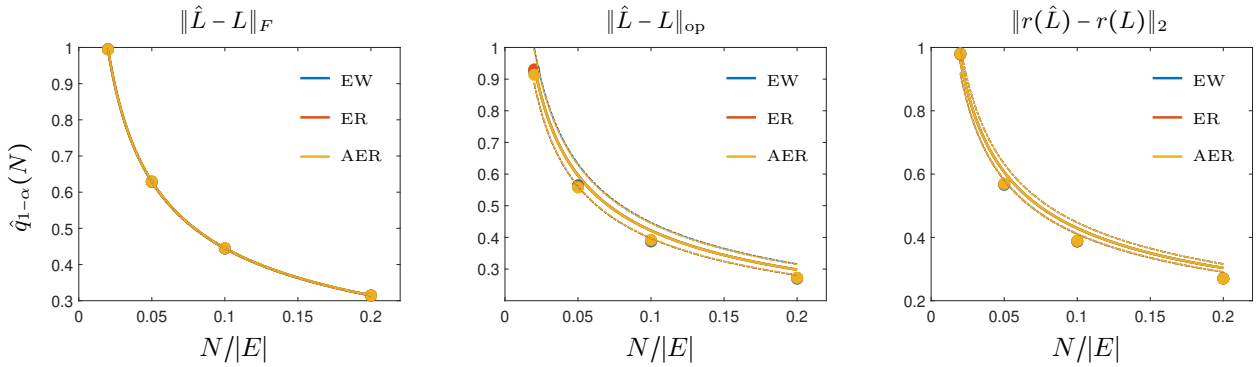


Figure 6: Results on incremental refinement for DIMACS.

C Additional details on experiments

Details of the graphs used in Section 4. The edges of **Genes** and **Citations** have varying weights, whereas the edges of the other graphs all have equal weights.

- **ca-cit-HepTh (Citations)** (Rossi and Ahmed, 2015): This graph represents the co-citation of scientific papers from arXiv’s high energy physics-theory (HEP-TH) section, involving 22,908 vertices and 2,444,798 edges. An edge between two papers means that both papers have been cited by a common third paper (Kunegis, 2013).
- **C2000-9 (DIMACS)** (Rossi and Ahmed, 2015): This graph is from the DIMACS Implementation Challenge (Bader et al., 2011), consisting of 2,000 nodes and 1,799,532 edges.
- **human-gene2 (Genes)** (Davis and Hu, 2011): This graph is a human gene regulatory network, with 14,340 vertices and 9,041,364 edges.
- **FB-Howard90 (Howard)** (Rossi and Ahmed, 2015): This graph is a social network graph constructed based on Howard University Facebook data (Red et al., 2011; Traud et al., 2012). All friendships are represented as undirected links. The graph contains 4,047 vertices and 204,850 edges.
- **Mycielskian14 (M14)** (Davis and Hu, 2011): This graph is part of a test suite for benchmarking graph algorithms. It is triangle free with a chromatic number of 14, and contains 12,287 vertices and 1,847,756 edges (Mycielski, 1955).
- **Mycielskian20 (M20)** (Davis and Hu, 2011): This graph is part of a test suite for benchmarking graph algorithms. It is triangle free with a chromatic number of 20, and contains 786,431 vertices and 2,710,370,560 edges (Mycielski, 1955).

Discussion of AER sampling. Recall that effective-resistance sampling is based on edge probabilities of the form $p(e) \propto w(e)\text{tr}(L^+\Delta_e)$, where L^+ is the Moore-Penrose inverse of L . Because it is often costly or infeasible to compute L^+ , there has been substantial research interest in developing efficient ways to approximate

these probabilities. Here, we discuss one approach that was proposed in Spielman and Srivastava (2011) and implemented in Lebron (2025). In a nutshell, the approximate effective-resistance sampling probabilities are of the form $p(e) \propto w(e)\text{tr}(S^\top S \Delta_e)$, where S is a $k \times n$ random matrix and k is of order $\log(n)/\epsilon^2$ for some accuracy parameter $\epsilon > 0$. The computation of the matrix S combines random projections with repeated use of the Spielman and Teng solver (Spielman and Teng, 2004), but the precise details are beyond the scope of our work here. In Section 4 we set $\epsilon = 0.01$, and in Appendix A we set $\epsilon = 1$. The reason for using $\epsilon = 1$ in the second case is that it was the smallest choice for which some difference in the results for ER and AER could be observed.

License information. The University of Florida sparse matrix collection (Davis and Hu, 2011) is under the CC BY 4.0 License, and the graphs from Rossi and Ahmed (2015) are under a CC BY-SA License. **Beans** and **Images** from the UCI machine learning repository are both available under the CC BY 4.0 License.

Computing resources. The results presented in Table 1, Appendix A, and Appendix B were obtained using MATLAB on servers equipped with 32 CPUs and 216 GB of RAM. All of the experiments together consumed roughly 200 hours of computing time. The results in Section 4 in the discussion of computational efficiency were obtained using a laptop with approximately 16 GB of RAM, 8 physical cores, and 16 logical cores.

D Notation and conventions in proofs

The L^q norm of a scalar random variable V is denoted as $\|V\|_{L^q} = (\mathbf{E}(|V|^q))^{1/q}$. For any random object V , we use $\mathcal{L}(V)$ to refer to its distribution, while $\mathcal{L}(\cdot|Q)$, $\mathbf{P}(\cdot|Q)$ and $\mathbf{E}(\cdot|Q)$ refer to conditional distributions, probabilities and expectations given the random matrices Q_1, \dots, Q_N . Convergence in probability and convergence in distribution are respectively denoted by $\xrightarrow{\mathbf{P}}$ and $\xrightarrow{\mathcal{L}}$. The Kolmogorov metric is defined as $d_K(\mathcal{L}(V), \mathcal{L}(W)) = \sup_{t \in \mathbb{R}} |\mathbf{P}(V \leq t) - \mathbf{P}(W \leq t)|$. In connection with this metric, we will sometimes use Pólya’s theorem (Bickel and Doksum, 2015, Theorem B.7.7), which implies that if $\{V_n\}$ is a sequence of random variables satisfying $V_n \xrightarrow{\mathcal{L}} Z$ as $n \rightarrow \infty$ for a standard normal random variable Z , then $d_K(\mathcal{L}(V_n), \mathcal{L}(Z)) \rightarrow 0$ as $n \rightarrow \infty$.

For matrices $A, B \in \mathbb{R}^{n \times n}$, let $\langle A, B \rangle = \text{tr}(A^\top B)$. For $A \in \mathbb{R}^{n \times n}$, define $\|A\|_\infty = \max_{1 \leq i, j \leq n} |A_{ij}|$. For two sequences of non-negative real numbers a_n and b_n , we write $a_n \lesssim b_n$ if there exists a constant $C > 0$, independent of n , such that $a_n \leq C b_n$ holds for all large n . If both $a_n \lesssim b_n$ hold and $b_n \lesssim a_n$ hold, then we write $a_n \asymp b_n$. The relation $a_n = o(b_n)$ means $a_n/b_n \rightarrow 0$ as $n \rightarrow \infty$, while $a_n = \mathcal{O}(b_n)$ is equivalent to $a_n \lesssim b_n$. For two sequences of random variables $\{U_n\}$ and $\{V_n\}$, the relation $U_n = o_{\mathbf{P}}(V_n)$ means that $U_n/V_n \xrightarrow{\mathbf{P}} 0$, and the relation $U_n = \mathcal{O}_{\mathbf{P}}(V_n)$ means that for every $\epsilon > 0$, there exists a positive constant C not depending on n such that the inequality $\mathbf{P}(|U_n|/|V_n| \geq C) \leq \epsilon$ holds for all large n . The indicator function for a condition \dots is represented as $1\{\dots\}$.

Because the probabilities $\mathbf{P}(\cap_{x \in \mathcal{C}} \{C(x) \in \hat{\mathcal{T}}_{1-\alpha}(x)\})$ and $\mathbf{P}(\|\hat{L} - L\|_F^2 \leq \hat{q}_{1-\alpha})$ in Theorems 1 and 2 are both invariant to rescaling L by a positive constant, we may always assume without loss of generality that all the edge weights sum to 1,

$$\sum_{e \in E} w(e) = 1.$$

In this case, the edge weights $w(e)$ and sampling probabilities $p(e)$ are the same, and so the collection $\mathcal{Q} = \{(w(e)/p(e))\Delta_e | e \in E\}$ is the same as $\{\Delta_e | e \in E\}$. Furthermore, this means that the i.i.d. random matrices Q_1, \dots, Q_N satisfy $\mathbf{P}(Q_1 = \Delta_e) = p(e)$ for all $e \in E$. As another simplification, the proofs will generally omit the subscript n that was used in the statements of the theorems.

To introduce some further notation that will be used in the proofs, note that the symmetric rank-1 matrix Δ_e associated to an edge $e = \{i, j\}$ with $i < j$ can be written as $\Delta_e = \delta_e \delta_e^\top$, where $\delta_e = u_i - u_j$, and u_i denotes the i th standard basis vector. Also, let $\hat{e}_1, \dots, \hat{e}_N$ be i.i.d samples drawn from E via edge-weight sampling, and define $D_i = \delta_{\hat{e}_i}$ for simplicity. In this case, the random matrix \hat{L} can be represented as

$$\hat{L} = \frac{1}{N} \sum_{i=1}^N D_i D_i^\top.$$

Let $\hat{e}_1^*, \dots, \hat{e}_N^*$ be i.i.d samples uniformly drawn from $\hat{e}_1, \dots, \hat{e}_N$, and $\hat{e}_1^{**}, \dots, \hat{e}_N^{**}$ be i.i.d samples uniformly drawn from $\hat{e}_1^*, \dots, \hat{e}_N^*$. Define

$$D_i^* = \delta_{\hat{e}_i^*} \quad \text{and} \quad D_i^{**} = \delta_{\hat{e}_i^{**}}$$

for $i = 1, \dots, N$, so that the random matrices \hat{L}^* and \hat{L}^{**} can be represented as

$$\hat{L}^* = \frac{1}{N} \sum_{i=1}^N D_i^* (D_i^*)^\top \quad \text{and} \quad \hat{L}^{**} = \frac{1}{N} \sum_{i=1}^N D_i^{**} (D_i^{**})^\top.$$

E Proof of Theorem 1

Let the set of cut vectors be enumerated as $\mathcal{C} = \{x_1, \dots, x_{|\mathcal{C}|}\}$. (Note that in the main text, we used x_i to refer to the i th coordinate of a single vector $x \in \mathcal{C}$, but that earlier usage will no longer be needed for cut vectors.) Also, for $i, j \in \{1, \dots, |\mathcal{C}|\}$, define

$$\begin{aligned} s_{ij} &= \text{cov}(x_i^\top Q_1 x_i, x_j^\top Q_1 x_j) = \sum_{e \in E} w(e) x_i^\top \Delta_e x_i x_j^\top \Delta_e x_j - x_i^\top L x_i x_j^\top L x_j \\ \hat{s}_{ij} &= \text{cov}(x_i^\top Q_1^* x_i, x_j^\top Q_1^* x_j | Q) = \frac{1}{N} \sum_{k=1}^N x_i^\top Q_k x_i x_j^\top Q_k x_j - x_i^\top \hat{L} x_i x_j^\top \hat{L} x_j. \end{aligned} \tag{6}$$

Next, define the random variables

$$\begin{aligned} M &= \sqrt{N} \max_{1 \leq i \leq |\mathcal{C}|} |x_i^\top \hat{L} x_i - x_i^\top L x_i| / \sqrt{s_{ii}}, \\ \hat{M} &= \sqrt{N} \max_{1 \leq i \leq |\mathcal{C}|} |x_i^\top \hat{L} x_i - x_i^\top L x_i| / \sqrt{\hat{s}_{ii}}, \\ M^* &= \sqrt{N} \max_{1 \leq i \leq |\mathcal{C}|} |x_i^\top \hat{L}^* x_i - x_i^\top \hat{L} x_i| / \sqrt{\hat{s}_{ii}}. \end{aligned}$$

Let $G = (G_1, \dots, G_{|\mathcal{C}|})$ be a Gaussian vector drawn from $N(0, R)$, where $R_{ij} = \frac{s_{ij}}{\sqrt{s_{ii}s_{jj}}}$, and define

$$M(G) = \max_{1 \leq i \leq |\mathcal{C}|} |G_i|.$$

Also, let $\hat{G} = (\hat{G}_1, \dots, \hat{G}_{|\mathcal{C}|})$ be a random vector that is drawn $N(0, \hat{R})$ conditionally on Q_1, \dots, Q_N , where $\hat{R}_{ij} = \frac{\hat{s}_{ij}}{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}}$, and define

$$M(\hat{G}) = \max_{1 \leq i \leq |\mathcal{C}|} |\hat{G}_i|.$$

It follows from standard arguments in the bootstrap literature (e.g. the proof of (Lopes, 2022, Lemma 10.4)) that Theorem 1 reduces to showing the following two limits as $n \rightarrow \infty$,

$$\begin{aligned} d_K(\mathcal{L}(\hat{M}), \mathcal{L}(M(G))) &\rightarrow 0 \\ d_K(\mathcal{L}(M^* | Q), \mathcal{L}(M(G))) &\xrightarrow{\mathbf{P}} 0. \end{aligned}$$

These two statements are shown in Lemmas E.1 and E.2 respectively.

Lemma E.1. *If the conditions in Theorem 1 hold, then as $n \rightarrow \infty$,*

$$d_K(\mathcal{L}(\hat{M}), \mathcal{L}(M(G))) \rightarrow 0.$$

Proof. Since the triangle inequality gives

$$d_K(\mathcal{L}(\hat{M}), \mathcal{L}(M(G))) \leq d_K(\mathcal{L}(\hat{M}), \mathcal{L}(M)) + d_K(\mathcal{L}(M), \mathcal{L}(M(G))),$$

we will handle the terms on the right side separately. To handle the second term on the right, we will apply Lemma G.1 to establish a Gaussian approximation for M . Specifically, we will apply this lemma to a set of i.i.d. random vectors $X_1, \dots, X_N \in \mathbb{R}^{|\mathcal{C}|}$, where the j th component of the i th vector is defined by $X_{ij} = (x_j^\top Q_i x_j - x_j^\top L x_j) / \sqrt{s_{jj}}$. Also note that $x_j^\top Q_1 x_j$ is a Bernoulli($x_j^\top L x_j$) random variable for all $j = 1, \dots, |\mathcal{C}|$. In the notation of Lemma G.1, we will put $p = |\mathcal{C}|$, $b_1 = b_2 = 1$ and $c_n = 2/(\log(2)\sqrt{\eta(\mathcal{C})})$. By noting the inequalities $0 \leq x_j^\top Q_i x_j \leq 1$, and $0 \leq x_j^\top L x_j \leq 1$, as well as the fourth central moment formula

$$\mathbf{E}((x_i^\top Q_1 x_i - x_i^\top L x_i)^4) = x_i^\top L x_i (1 - x_i^\top L x_i) (1 - 3x_i^\top L x_i (1 - x_i^\top L x_i)), \tag{7}$$

it is possible to check that the three conditions in Lemma G.1 hold under these choices of b_1 , b_2 , and c_n . Consequently, the lemma gives

$$d_K(\mathcal{L}(M), \mathcal{L}(M(G))) \lesssim \left(\frac{\log(N|\mathcal{C}|)^5}{N\eta(\mathcal{C})} \right)^{1/4}.$$

So, under the conditions in Theorem 1, it follows that as $n \rightarrow \infty$,

$$d_K(\mathcal{L}(M), \mathcal{L}(M(G))) = o(1). \quad (8)$$

To analyze $d_K(\mathcal{L}(M), \mathcal{L}(\hat{M}))$, we will use the basic fact that the Kolmogorov distance between any two random variables U and V can be bounded as

$$d_K(\mathcal{L}(U), \mathcal{L}(V)) \leq \mathbf{P}(|U - V| > \epsilon) + \sup_{r \in \mathbb{R}} \mathbf{P}(|U - r| \leq \epsilon) \quad (9)$$

for any $\epsilon > 0$. Specifically, we will take $U = M$, $V = \hat{M}$, and $\epsilon = \log(N|\mathcal{C}|)^2 / \sqrt{N\eta(\mathcal{C})}$. To handle the second term on the right side of (9), we will use the assumptions in Theorem 1, in conjunction with Lemmas G.3 and G.4 as well as the limit (8) to conclude that

$$\begin{aligned} \sup_{r \in \mathbb{R}} \mathbf{P}\left(|M - r| \leq \frac{\log(N|\mathcal{C}|)^2}{\sqrt{N\eta(\mathcal{C})}}\right) &\leq \sup_{r \in \mathbb{R}} \mathbf{P}\left(|M(G) - r| \leq \frac{\log(N|\mathcal{C}|)^2}{\sqrt{N\eta(\mathcal{C})}}\right) + o(1) \\ &\lesssim \frac{\log(N|\mathcal{C}|)^{5/2}}{\sqrt{N\eta(\mathcal{C})}} + o(1) \\ &= o(1). \end{aligned}$$

For the first term on the right side of (9), note that

$$\begin{aligned} \mathbf{P}\left(|M - \hat{M}| > \frac{\log(N|\mathcal{C}|)^2}{\sqrt{N\eta(\mathcal{C})}}\right) &\leq \mathbf{P}\left(M \max_{1 \leq i \leq |\mathcal{C}|} \left| \frac{\sqrt{s_{ii}}}{\sqrt{\hat{s}_{ii}}} - 1 \right| > \frac{\log(N|\mathcal{C}|)^2}{\sqrt{N\eta(\mathcal{C})}}\right) \\ &\leq \mathbf{P}\left(M > 4\sqrt{\log(N|\mathcal{C}|)}\right) + \mathbf{P}\left(\max_{1 \leq i \leq |\mathcal{C}|} \left| \frac{\sqrt{s_{ii}}}{\sqrt{\hat{s}_{ii}}} - 1 \right| > \frac{\log(N|\mathcal{C}|)^{3/2}}{4\sqrt{N\eta(\mathcal{C})}}\right). \end{aligned} \quad (10)$$

Combining the limit (8) with a union bound, we have

$$\begin{aligned} \mathbf{P}\left(M > 4\sqrt{\log(N|\mathcal{C}|)}\right) &\leq \mathbf{P}\left(M(G) > 4\sqrt{\log(N|\mathcal{C}|)}\right) + o(1) \\ &\leq \sum_{i=1}^{|\mathcal{C}|} \mathbf{P}\left(|G_i| > 4\sqrt{\log(N|\mathcal{C}|)}\right) + o(1). \\ &\lesssim \frac{1}{N} + o(1). \end{aligned} \quad (11)$$

To handle the second term on the right side of (10), note that \hat{s}_{ii} can be represented as

$$\hat{s}_{ii} = \frac{1}{N^2} \sum_{1 \leq k < j \leq N} (x_i^\top Q_k x_i - x_i^\top Q_j x_i)^2$$

and $\mathbf{E}((x_i^\top Q_k x_i - x_i^\top Q_j x_i)^2) = s_{ii} = x_i^\top L x_i (1 - x_i^\top L x_i)$. Since $\text{var}(x^\top Q_1 x (1 - x^\top L x)) \leq 1$ holds for any $x \in \mathcal{C}$, a concentration inequality for U statistics (Arcones, 1995, Theorem 2) can be used to obtain

$$\mathbf{P}\left(\left| \frac{\sum_{1 \leq k < j \leq N} (x_i^\top Q_k x_i - x_i^\top Q_j x_i)^2}{N(N-1)x_i^\top L x_i (1 - x_i^\top L x_i)} - 1 \right| \geq \epsilon\right) \leq 4 \exp\left(-\frac{N\epsilon^2(x_i^\top L x_i (1 - x_i^\top L x_i))^2}{8 + 128x_i^\top L x_i (1 - x_i^\top L x_i)\epsilon}\right).$$

Hence, for any $i = 1, \dots, |\mathcal{C}|$ and $\epsilon \in (0, 1)$, we have

$$\begin{aligned} \mathbf{P}\left(\left| \frac{\sqrt{s_{ii}}}{\sqrt{\hat{s}_{ii}}} - 1 \right| \geq \epsilon\right) &\leq \mathbf{P}\left(\left| \frac{s_{ii}}{\hat{s}_{ii}} - 1 \right| \geq \epsilon\right) \\ &\leq \mathbf{P}\left(\left| \frac{\hat{s}_{ii}}{s_{ii}} - 1 \right| \geq \frac{\epsilon}{2}\right) \\ &\leq \mathbf{P}\left(\left| \frac{\sum_{1 \leq k < j \leq N} (x_i^\top Q_k x_i - x_i^\top Q_j x_i)^2}{N(N-1)x_i^\top L x_i (1 - x_i^\top L x_i)} - 1 \right| \geq \frac{\epsilon}{4}\right) \\ &\leq 4 \exp\left(-\frac{N\epsilon^2(x_i^\top L x_i (1 - x_i^\top L x_i))^2}{128(1 + 4\epsilon)}\right), \end{aligned}$$

and a union bound implies

$$\mathbf{P}\left(\max_{1 \leq i \leq |\mathcal{C}|} \left| \frac{\sqrt{s_{ii}}}{\sqrt{\hat{s}_{ii}}} - 1 \right| \geq \epsilon\right) \leq 4|\mathcal{C}| \exp\left(-\frac{N\eta(\mathcal{C})^2 \epsilon^2}{128(1+4\epsilon)}\right), \quad (12)$$

when N is large. Taking $\epsilon = \frac{\log(N|\mathcal{C}|)^{3/2}}{\sqrt{N}\eta(\mathcal{C})}$, the conditions in Theorem 1 show that as $n \rightarrow \infty$

$$\mathbf{P}\left(\max_{1 \leq i \leq |\mathcal{C}|} \left| \frac{\sqrt{s_{ii}}}{\sqrt{\hat{s}_{ii}}} - 1 \right| \geq \frac{\log(N|\mathcal{C}|)^{3/2}}{4\sqrt{N}\eta(\mathcal{C})}\right) = o(1),$$

which proves that the left side of (10) is $o(1)$, completing the proof. \square

Lemma E.2. *If the conditions in Theorem 1 hold, then as $n \rightarrow \infty$,*

$$d_K(\mathcal{L}(M^*|Q), \mathcal{L}(M(G))) \xrightarrow{\mathbf{P}} 0.$$

Proof. By the triangle inequality, we have

$$d_K(\mathcal{L}(M^*|Q), \mathcal{L}(M(G))) \leq d_K(\mathcal{L}(M^*|Q), \mathcal{L}(M(\hat{G})|Q)) + d_K(\mathcal{L}(M(G)), \mathcal{L}(M(\hat{G})|Q)). \quad (13)$$

With regard to the second term on the right side, Lemma E.3 and the Gaussian comparison inequality in Lemma G.2 imply

$$\begin{aligned} d_K(\mathcal{L}(M(G)), \mathcal{L}(M(\hat{G})|Q)) &\lesssim (\|\hat{R} - R\|_\infty \log(|\mathcal{C}|)^2)^{1/2} \\ &= o_{\mathbf{P}}(1). \end{aligned}$$

To handle the first term on the right side of (13), we will follow the argument used in deriving (8). The calculation in (7) yields

$$\mathbf{E}((x_i^\top Q_1^* x_i - x_i^\top \hat{L} x_i)^4 | Q) = x_i^\top \hat{L} x_i (1 - x_i^\top \hat{L} x_i) (1 - 3x_i^\top \hat{L} x_i (1 - x_i^\top \hat{L} x_i)).$$

We will apply Lemma G.1 (conditionally on Q_1, \dots, Q_N) to a set of random vectors $X_1, \dots, X_N \in \mathbb{R}^{|\mathcal{C}|}$, where the j th component of the i th vector is defined by $X_{ij} = (x_j^\top Q_i^* x_j - x_j^\top \hat{L} x_j) / \sqrt{\hat{s}_{jj}}$. Also, in the notation of that lemma, we will take $b_1 = b_2 = 1$ and $c_n^2 = \frac{4}{\log(2)^2} / \min_{x \in \mathcal{C}} \{x^\top \hat{L} x - (x^\top \hat{L} x)^2\}$, which implies that the following bound holds with probability 1,

$$d_K(\mathcal{L}(M^*|Q), \mathcal{L}(M(\hat{G})|Q)) \lesssim \left(\frac{\log(N|\mathcal{C}|)^5}{N \min_{x \in \mathcal{C}} x^\top \hat{L} x (1 - x^\top \hat{L} x)} \right)^{1/4}. \quad (14)$$

Also, for any numbers $a, b \in [0, 1]$, we have $|a(1-a) - b(1-b)| \leq 2|a-b|$ and so

$$\min_{x \in \mathcal{C}} x^\top \hat{L} x (1 - x^\top \hat{L} x) \geq \eta(\mathcal{C}) - 2 \max_{x \in \mathcal{C}} |x^\top \hat{L} x - x^\top L x|.$$

To demonstrate the right side of (14) is $o_{\mathbf{P}}(1)$, it suffices to show

$$\max_{x \in \mathcal{C}} |x^\top \hat{L} x - x^\top L x| = o_{\mathbf{P}}(\eta(\mathcal{C})),$$

and then combining with the conditions in Theorem 1 will give the right side of (14) is $o_{\mathbf{P}}(1)$. The bound (11) implies $M = \mathcal{O}_{\mathbf{P}}(\sqrt{\log(N|\mathcal{C}|)})$, and so the conditions in Theorem 1 give

$$\begin{aligned} \max_{x \in \mathcal{C}} |x^\top L x - x^\top \hat{L} x| &\leq \frac{M}{\sqrt{\log(N|\mathcal{C}|)}} \sqrt{\frac{\log(N|\mathcal{C}|)}{N}} \\ &= \mathcal{O}_{\mathbf{P}}(1) \cdot o(\eta(\mathcal{C})) \\ &= o_{\mathbf{P}}(\eta(\mathcal{C})). \end{aligned} \quad (15)$$

\square

Lemma E.3. *If the conditions in Theorem 1 hold, then as $n \rightarrow \infty$,*

$$\|\hat{R} - R\|_\infty \log(|\mathcal{C}|)^2 \xrightarrow{\mathbf{P}} 0. \quad (16)$$

Proof. Observe that

$$\|\hat{R} - R\|_\infty \leq \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{|\hat{s}_{ij} - s_{ij}|}{\sqrt{s_{ii}s_{jj}}} + \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{|\hat{s}_{ij}|}{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}} \left| 1 - \frac{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}}{\sqrt{s_{ii}s_{jj}}} \right|. \quad (17)$$

For the second term on the right side, by noting that $\frac{|\hat{s}_{ij}|}{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}} \leq 1$, we can obtain

$$\begin{aligned} \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{|\hat{s}_{ij}|}{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}} \left| 1 - \frac{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}}{\sqrt{s_{ii}s_{jj}}} \right| &\leq \max_{1 \leq i \leq |\mathcal{C}|} \left| 1 - \frac{\sqrt{\hat{s}_{ii}}}{\sqrt{s_{ii}}} \right| + \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{\sqrt{\hat{s}_{ii}}}{\sqrt{s_{ii}}} \left| 1 - \frac{\sqrt{\hat{s}_{jj}}}{\sqrt{s_{jj}}} \right| \\ &\lesssim \max_{1 \leq i \leq |\mathcal{C}|} \left| 1 - \frac{\sqrt{\hat{s}_{ii}}}{\sqrt{s_{ii}}} \right| + \left(\max_{1 \leq i \leq |\mathcal{C}|} \left| 1 - \frac{\sqrt{\hat{s}_{ii}}}{\sqrt{s_{ii}}} \right| \right)^2. \end{aligned}$$

Applying the inequality in (12) with $\epsilon = 1/\log(|\mathcal{C}|)^2$, we have

$$\log(|\mathcal{C}|)^2 \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{|\hat{s}_{ij}|}{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}} \left| 1 - \frac{\sqrt{\hat{s}_{ii}\hat{s}_{jj}}}{\sqrt{s_{ii}s_{jj}}} \right| = o_{\mathbf{P}}(1).$$

For the first term on the right side of (17), combining the definitions of s_{ij} and \hat{s}_{ij} in (6) with the facts that $s_{ii} \geq \sqrt{\eta(\mathcal{C})}$, $x_i^\top \hat{L}x_i \leq 1$, and $x_i^\top Lx_i \leq 1$, we have

$$\begin{aligned} \log(|\mathcal{C}|)^2 \max_{1 \leq i, j \leq |\mathcal{C}|} \frac{|\hat{s}_{ij} - s_{ij}|}{\sqrt{s_{ii}s_{jj}}} &\lesssim \frac{\log(|\mathcal{C}|)^2}{\eta(\mathcal{C})} \max_{1 \leq i, j \leq |\mathcal{C}|} \left| \frac{1}{N} \sum_{k=1}^N x_i^\top Q_k x_i x_j^\top Q_k x_j - \mathbf{E}(x_i^\top Q_1 x_i x_j^\top Q_1 x_j) \right| \\ &\quad + \frac{\log(|\mathcal{C}|)^2}{\sqrt{\eta(\mathcal{C})}} \max_{1 \leq i \leq |\mathcal{C}|} \frac{|x_i^\top \hat{L}x_i - x_i^\top Lx_i|}{\sqrt{s_{ii}}}. \end{aligned} \quad (18)$$

Following a similar argument to (15), the second term on the right side of (18) is $o_{\mathbf{P}}(1)$. It remains to show the first term on the right side of (18) is $o_{\mathbf{P}}(1)$. By noting that $x_i^\top Q_1 x_i x_j^\top Q_1 x_j$ takes values in $\{0, 1\}$, Bernstein's inequality (Wainwright, 2019, Proposition 2.14) gives for any fixed $\epsilon > 0$,

$$\mathbf{P} \left(\frac{\log(|\mathcal{C}|)^2}{\eta(\mathcal{C})} \left| \frac{1}{N} \sum_{k=1}^N x_i^\top Q_k x_i x_j^\top Q_k x_j - \mathbf{E}(x_i^\top Q_1 x_i x_j^\top Q_1 x_j) \right| > \epsilon \right) \leq 2 \exp \left(- \frac{N \eta(\mathcal{C})^2 \epsilon^2}{2 \log(|\mathcal{C}|)^4 (1 + \epsilon)} \right),$$

and so applying a union bound over $1 \leq i, j \leq |\mathcal{C}|$ shows that the first term on the right side of (18) is indeed $o_{\mathbf{P}}(1)$. Combining the above results with the conditions in Theorem 1 completes the proof. \square

F Proof of Theorem 2

Let $\mu = \mathbf{E}(\|\hat{L} - L\|_F^2)$ and $\sigma^2 = \text{var}(\|\hat{L} - L\|_F^2)$. Define the statistics

$$T = \frac{\|\hat{L} - L\|_F^2 - \hat{\mu}}{\hat{\sigma}} \quad \text{and} \quad T^* = \frac{\|\hat{L}^* - \hat{L}\|_F^2 - \hat{\mu}^*}{\hat{\sigma}^*},$$

where $\hat{\mu}, \hat{\mu}^*, \hat{\sigma}, \hat{\sigma}^*$ are defined as in Algorithm 1 with ψ corresponding to $\|\cdot\|_F^2$. In particular, letting $\hat{L}_1^*, \dots, \hat{L}_B^*$ denote conditionally i.i.d. copies of \hat{L}^* given Q_1, \dots, Q_N , the quantities $\hat{\mu}$ and $\hat{\sigma}$ can be represented in distribution as

$$\hat{\mu} = \frac{1}{B} \sum_{b=1}^B \|\hat{L}_b^* - \hat{L}\|_F^2 \quad \text{and} \quad \hat{\sigma}^2 = \frac{1}{B} \sum_{b=1}^B (\|\hat{L}_b^* - \hat{L}\|_F^2 - \hat{\mu})^2.$$

As in the proof of Theorem 1, it is sufficient to show that as $n \rightarrow \infty$,

$$d_K(\mathcal{L}(T), \mathcal{L}(Z)) \rightarrow 0 \quad (19)$$

$$d_K(\mathcal{L}(T^*|Q), \mathcal{L}(Z)) \xrightarrow{\mathbf{P}} 0, \quad (20)$$

where Z denotes a standard Gaussian random variable. Lemma F.2 ensures that $T \xrightarrow{\mathcal{L}} Z$, and consequently, Pólya's theorem (Bickel and Doksum, 2015, Theorem B.7.7) implies the limit (19).

To establish $d_K(\mathcal{L}(T^*|Q), \mathcal{L}(Z)) \xrightarrow{\mathbf{P}} 0$, we need to deal with the fact that $\mathcal{L}(T^*|Q)$ is a random probability distribution. It is enough to show that for any subsequence $J \subset \{1, 2, \dots\}$, there is a further subsequence $J' \subset J$ such that the limit $d_K(\mathcal{L}(T^*|Q), \mathcal{L}(Z)) \rightarrow 0$ holds almost surely as $n \rightarrow \infty$ along $n \in J'$. The key ingredient for doing this is to show that if $\hat{\mathbf{d}} \in \mathbb{R}^n$ contains the diagonal entries of \hat{L} , then $\|\hat{\mathbf{d}}\|_\infty / \|\hat{\mathbf{d}}\|_2 = o_{\mathbf{P}}(1)$ holds as $n \rightarrow \infty$, which is established in Lemma F.8. This implies that $\|\hat{\mathbf{d}}\|_\infty / \|\hat{\mathbf{d}}\|_2 \rightarrow 0$ holds almost surely as $n \rightarrow \infty$ along a subsequence of J . Because \hat{L}^* can be viewed as being generated with N edges that are drawn from \hat{G} in an i.i.d. manner with edge-weight sampling, analogues of the original conditions in Theorem 2 hold with respect to \hat{L} (instead of L), almost surely along subsequences. Therefore, the argument for proving (19) can be used in a completely analogous manner to prove the limit (20).

The following lemma provides some basic properties of L that we need at various points in the proof of Theorem 2.

Lemma F.1. *If $\frac{n}{N} \rightarrow 0$ and $\frac{\|\mathbf{d}\|_\infty^2}{\|\mathbf{d}\|_2^2} \rightarrow 0$ hold as $n \rightarrow \infty$, then the following limits also hold as $n \rightarrow \infty$,*

$$\|L\|_\infty = \|\mathbf{d}\|_\infty = o(1), \quad \text{tr}(L^2) = o(1), \quad \text{and} \quad N\text{tr}(L^2) \rightarrow \infty.$$

Proof. Note that the entry of a positive semidefinite matrix with the largest magnitude must always occur along the diagonal, and so $\|L\|_\infty = \|\mathbf{d}\|_\infty$.

To show $\|\mathbf{d}\|_\infty = o(1)$, we write $\|\mathbf{d}\|_\infty = (\|\mathbf{d}\|_\infty / \|\mathbf{d}\|_2) \|\mathbf{d}\|_2$, and so the assumption $\|\mathbf{d}\|_\infty / \|\mathbf{d}\|_2 = o(1)$ implies that it is sufficient to show $\|\mathbf{d}\|_2 \lesssim 1$. For this purpose, first note that $\|\mathbf{d}\|_2^2 \leq \text{tr}(L^2)$. Since $\text{tr}(L) = 2w(E)$ our reduction to the case when $w(E) = 1$ gives $\text{tr}(L) = 2$. Therefore, using the general inequality $1 \leq \text{tr}(A)^2 / \text{tr}(A^2)$ for any non-zero $n \times n$ positive semidefinite matrix A , we have $\text{tr}(L^2) \lesssim 1$, as needed.

To show $\text{tr}(L^2) = o(1)$, observe that Hölder's inequality and our previous steps imply

$$\begin{aligned} \text{tr}(L^2) &\leq \|L\|_\infty \sum_{1 \leq i, j \leq n} |L_{ij}| \\ &= \|L\|_\infty \cdot 4 \sum_{e \in E} w(e) \\ &= \|\mathbf{d}\|_\infty \cdot 4 \\ &= o(1). \end{aligned}$$

Finally, to show that $N\text{tr}(L^2) \rightarrow \infty$ as $n \rightarrow \infty$, note that the inequality $\text{tr}(A)^2 / \text{tr}(A^2) \leq n$ holds for any non-zero $n \times n$ positive semidefinite matrix A , and so $\text{tr}(L^2) \geq \text{tr}(L)^2 / n = 4/n$. So, because our assumption on N and n implies $N/n \rightarrow \infty$, the proof is complete. \square

F.1 Asymptotic normality of $\|\hat{L} - L\|_F^2$

Lemma F.2. *If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$\frac{\|\hat{L} - L\|_F^2 - \hat{\mu}}{\hat{\sigma}} \xrightarrow{\mathcal{L}} N(0, 1).$$

Proof. Lemmas F.6 and F.7 establish

$$\frac{\hat{\mu} - \mu}{\sigma} \xrightarrow{\mathbf{P}} 0 \quad \text{and} \quad \frac{\hat{\sigma}^2}{\sigma^2} \xrightarrow{\mathbf{P}} 1.$$

If we can show

$$\frac{\|\hat{L} - L\|_F^2 - \mu}{\sigma} \xrightarrow{\mathcal{L}} N(0, 1), \quad (21)$$

then the proof is completed by Slutsky's lemma. Recall D_1, \dots, D_N defined in Appendix D are independent and identically distributed random vectors with $\mathbf{E}(D_1 D_1^\top) = L$, and so we have

$$\begin{aligned} \mu &= \frac{1}{N^2} \sum_{i,j=1}^N \mathbf{E}(\langle D_i D_i^\top - L, D_j D_j^\top - L \rangle) \\ &= \frac{1}{N} \mathbf{E}(\|D_1 D_1^\top - L\|_F^2) \\ &= \frac{1}{N} (4 - \text{tr}(L^2)), \end{aligned} \quad (22)$$

where we have used the almost-sure relation $D_1^\top D_1 = 2$ in the last step. Based on this formula for μ , it can be checked by a direct algebraic calculation that $\|\hat{L} - L\|_F^2 - \mu$ may be decomposed according to

$$\|\hat{L} - L\|_F^2 - \mu = \frac{N-1}{N} U - \frac{2}{N} U',$$

where we define the statistics

$$\begin{aligned} U &= \left(\frac{1}{N(N-1)} \sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^2 \right) - \left(\frac{2}{N} \sum_{i=1}^N D_i^\top L D_i \right) + \text{tr}(L^2) \\ U' &= \left(\frac{1}{N} \sum_{i=1}^N D_i^\top L D_i \right) - \text{tr}(L^2). \end{aligned}$$

Lemma F.3 shows that U/σ converges in distribution to $N(0, 1)$ as $n \rightarrow \infty$, and so the desired limit (21) will hold if we show that $U'/(N\sigma)$ is $o_{\mathbf{P}}(1)$. It is simple to check $\mathbf{E}(D_1^\top L D_1) = \text{tr}(L^2)$, and so $\mathbf{E}(U') = 0$. Hence, it is enough to show that the variance of $U'/(N\sigma)$ is $o(1)$. Since Lemma F.9 gives $N^2 \sigma^2 \asymp \text{tr}(L^2)$ and equation (39) in the proof of Lemma F.9 implies $\text{var}(D_1^\top L D_1) = o(\text{tr}(L^2))$, we have

$$\text{var}\left(\frac{U'}{N\sigma}\right) = \frac{1}{N^2 \sigma^2} \cdot \frac{1}{N} \text{var}(D_1^\top L D_1) = o(1).$$

□

Lemma F.3. *If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$d_K\left(\mathcal{L}\left(\frac{U}{\sigma}\right), \mathcal{L}(Z)\right) \rightarrow 0,$$

where Z is a standard Gaussian random variable.

Proof. Observe that

$$U = \frac{1}{N(N-1)} \sum_{1 \leq i \neq j \leq N} h(\hat{e}_i, \hat{e}_j),$$

where

$$\begin{aligned} h(\hat{e}_1, \hat{e}_2) &= \langle D_1 D_1^\top - L, D_2 D_2^\top - L \rangle \\ &= \sum_{1 \leq i \leq n} (D_1 D_1^\top - L)_{ii} (D_2 D_2^\top - L)_{ii} + 2 \sum_{1 \leq i < j \leq n} (D_1 D_1^\top - L)_{ij} (D_2 D_2^\top - L)_{ij}. \end{aligned}$$

Define the collection of ordered pairs $\mathcal{J} = \{(i, i') | 1 \leq i \leq i' \leq n\}$. For each $\mathbf{i} \in \mathcal{J}$, define

$$\begin{aligned} \varphi_{\mathbf{i}}(\hat{e}_1) &= 1\{i = i' \in \hat{e}_1\} - \sqrt{2} \cdot 1\{i < i', \hat{e}_1 = \{i, i'\}\} \\ \phi_{\mathbf{i}}(\hat{e}_1) &= \varphi_{\mathbf{i}}(\hat{e}_1) - \mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)). \end{aligned} \quad (23)$$

It can be checked that

$$\begin{aligned}\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)) &= L_{ii'}(1\{i = i'\} + \sqrt{2} \cdot 1\{i < i'\}) \\ \phi_{\mathbf{i}}(\hat{e}_1) &= (D_1 D_1^\top - L)_{ii'}(1\{i = i'\} + \sqrt{2} \cdot 1\{i < i'\}),\end{aligned}\tag{24}$$

which leads to

$$h(\hat{e}_1, \hat{e}_2) = \sum_{\mathbf{i} \in \mathcal{J}} \phi_{\mathbf{i}}(\hat{e}_1) \phi_{\mathbf{i}}(\hat{e}_2).$$

Let $\varphi(\hat{e}_1)$ and $\phi(\hat{e}_1)$ respectively denote the random vectors in $\mathbb{R}^{n(n+1)/2}$ defined by

$$\begin{aligned}\varphi(\hat{e}_1) &= (\varphi_{\mathbf{i}}(\hat{e}_1))_{\mathbf{i} \in \mathcal{J}} \\ \phi(\hat{e}_1) &= (\phi_{\mathbf{i}}(\hat{e}_1))_{\mathbf{i} \in \mathcal{J}}.\end{aligned}$$

Also, let \mathfrak{S} denote the $\frac{1}{2}n(n+1) \times \frac{1}{2}n(n+1)$ covariance matrix of the random vector $\phi(\hat{e}_1)$, so that

$$\begin{aligned}\mathfrak{S} &= \mathbf{E}(\phi(\hat{e}_1)\phi(\hat{e}_1)^\top) \\ &= \mathbf{E}(\varphi(\hat{e}_1)\varphi(\hat{e}_1)^\top) - \mathbf{E}(\varphi(\hat{e}_1))\mathbf{E}(\varphi(\hat{e}_1)^\top).\end{aligned}\tag{25}$$

Likewise, let $\lambda(\mathfrak{S}) \in \mathbb{R}^{n(n+1)/2}$ denote the vector containing the eigenvalues of \mathfrak{S} , and define the random variable

$$\xi_n = \frac{\sum_{j=1}^{n(n+1)/2} \lambda_j(\mathfrak{S})(Z_j^2 - 1)}{\sqrt{2} \operatorname{var}(h(\hat{e}_1, \hat{e}_2))^{1/2}},$$

where $Z_1, \dots, Z_{\frac{1}{2}n(n+1)}$ are independent standard Gaussian random variables. It can be checked that $\mathbf{E}(h(\hat{e}_1, \hat{e}_2)|\hat{e}_1) = 0$, and applying Lemma G.5 (Huang et al., 2023, Proposition 9) yields

$$d_K\left(\mathcal{L}\left(\frac{\sqrt{N(N-1)}}{\sqrt{2} \operatorname{var}(h(\hat{e}_1, \hat{e}_2))} U\right), \mathcal{L}(\xi_n)\right) \lesssim N^{-1/5} + \left(\frac{\mathbf{E}(|h(\hat{e}_1, \hat{e}_2)|^3)}{\sqrt{N} \operatorname{var}(h(\hat{e}_1, \hat{e}_2))^{3/2}}\right)^{1/7}.\tag{26}$$

Following the calculation in the proof of Lemma F.9, we obtain

$$\begin{aligned}\operatorname{var}(h(\hat{e}_1, \hat{e}_2)) &= \operatorname{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 + o(\operatorname{tr}(L^2)) \\ &\asymp \operatorname{tr}(L^2),\end{aligned}\tag{27}$$

and

$$\frac{2}{N(N-1)\sigma^2} \operatorname{var}(h(\hat{e}_1, \hat{e}_2)) \rightarrow 1.\tag{28}$$

To complete the proof, it remains to establish an upper bound on $\mathbf{E}(|h(\hat{e}_1, \hat{e}_2)|^3)$. As a shorthand, we write $\hat{e}_1 \sim \hat{e}_2$ whenever the edges \hat{e}_1 and \hat{e}_2 share exactly one vertex, $|\hat{e}_1 \cap \hat{e}_2| = 1$. By noting the basic relation

$$|D_1^\top D_2| = 1\{\hat{e}_1 \sim \hat{e}_2\} + 2 \cdot 1\{\hat{e}_1 = \hat{e}_2\},$$

we have

$$h(\hat{e}_1, \hat{e}_2) = 1\{\hat{e}_1 \sim \hat{e}_2\} + 4 \cdot 1\{\hat{e}_1 = \hat{e}_2\} - D_1^\top L D_1 - D_2^\top L D_2 + \operatorname{tr}(L^2).$$

Due to $D_1^\top L D_1 \leq 4\|L\|_\infty = o(1)$ (by Lemma F.1), we have $\mathbf{E}((D_1^\top L D_1)^3) \lesssim \mathbf{E}(D_1^\top L D_1)$ and

$$\begin{aligned}\mathbf{E}(|h(\hat{e}_1, \hat{e}_2)|^3) &\lesssim \mathbf{E}(1\{\hat{e}_1 \sim \hat{e}_2\}) + \mathbf{E}(1\{\hat{e}_1 = \hat{e}_2\}) + \mathbf{E}((D_1^\top L D_1)^3) + \operatorname{tr}(L^2)^3 \\ &\lesssim \sum_{1 \leq i < j \leq n} |L_{ij}|(L_{ii} + L_{jj}) + \sum_{1 \leq i < j \leq n} L_{ij}^2 + \mathbf{E}(D_1^\top L D_1) + \operatorname{tr}(L^2)^3 \\ &\lesssim \operatorname{tr}(L^2),\end{aligned}$$

where the last step is based on Lemma F.1 and the fact that L is diagonally dominant. Applying the above results and Lemma F.1 to (26) implies

$$d_K\left(\mathcal{L}\left(\frac{\sqrt{N(N-1)}}{\sqrt{2\text{var}(h(\hat{e}_1, \hat{e}_2))}}U\right), \mathcal{L}(\xi_n)\right) \lesssim N^{-1/5} + (N\text{tr}(L^2))^{-1/14} \rightarrow 0.$$

Also, Lemma F.4 and (27) imply $\xi_n \xrightarrow{\mathcal{L}} N(0, 1)$. Combining the above results with (28), Slutsky's lemma completes the poof. \square

Lemma F.4. *Let $\{Z_j \mid 1 \leq j \leq n(n+1)/2\}$ be a collection of i.i.d. $N(0, 1)$ random variables. If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$\frac{\sum_{j=1}^{n(n+1)/2} \lambda_j(\mathfrak{S})(Z_j^2 - 1)}{\sqrt{2(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2)}} \xrightarrow{\mathcal{L}} N(0, 1).$$

Proof. It follows from the Lindeberg CLT for triangular arrays given in (van der Vaart, 2000, Prop. 2.27) that if the condition $\frac{\|\lambda(\mathfrak{S})\|_\infty^2}{\|\lambda(\mathfrak{S})\|_2^2} \rightarrow 0$ holds, then as $n \rightarrow \infty$,

$$\frac{1}{\sqrt{2}\|\lambda(\mathfrak{S})\|_2} \sum_{j=1}^{n(n+1)/2} \lambda_j(\mathfrak{S})(Z_j^2 - 1) \xrightarrow{\mathcal{L}} N(0, 1).$$

It remains to show

$$\frac{\|\lambda(\mathfrak{S})\|_2^2}{\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2} \rightarrow 1 \quad \text{and} \quad \frac{\|\lambda(\mathfrak{S})\|_\infty^2}{\|\lambda(\mathfrak{S})\|_2^2} \rightarrow 0. \quad (29)$$

By noting that $\|\lambda(\mathfrak{S})\|_2^2 = \|\mathfrak{S}\|_F^2$, we need to calculate the sum of squares of the following entries

$$\mathfrak{S}_{\mathbf{i}\mathbf{j}} = \mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) - \mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1))\mathbf{E}(\varphi_{\mathbf{j}}(\hat{e}_1))$$

for $\mathbf{i}, \mathbf{j} \in \mathcal{J}$, where $\mathcal{J} = \{(i, i') \mid 1 \leq i \leq i' \leq n\}$. The equalities in (24) give $\sum_{\mathbf{i} \in \mathcal{J}} (\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)))^2 = \text{tr}(L^2)$ and so

$$\|\mathfrak{S}\|_F^2 = \sum_{\mathbf{i}, \mathbf{j} \in \mathcal{J}} (\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)))^2 - 2 \sum_{\mathbf{i}, \mathbf{j} \in \mathcal{J}} \mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1))\mathbf{E}(\varphi_{\mathbf{j}}(\hat{e}_1))\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) + \text{tr}(L^2)^2.$$

Combining with Lemma F.5 yields

$$\begin{aligned} \sum_{\mathbf{i}, \mathbf{j} \in \mathcal{J}} \mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1))^2 &= \sum_{i=1}^n L_{ii}^2 + \sum_{1 \leq i \neq j \leq n} L_{ij}^2 + 4 \sum_{1 \leq i < j \leq n} L_{ij}^2 + 4 \cdot 2 \sum_{1 \leq i < j \leq n} L_{ij}^2 \\ &= \text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2, \end{aligned}$$

and

$$\begin{aligned} \sum_{\mathbf{i}, \mathbf{j} \in \mathcal{J}} |\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1))\mathbf{E}(\varphi_{\mathbf{j}}(\hat{e}_1))\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1))| \\ \lesssim \sum_{i=1}^n L_{ii}^3 + \sum_{1 \leq i \neq j \leq n} |L_{ij}|L_{ii}L_{jj} + \sum_{1 \leq i \neq j \leq n} |L_{ij}|^3 + \sum_{1 \leq i \neq j \leq n} L_{ij}^2 L_{ii} \\ = o(\text{tr}(L^2)), \end{aligned}$$

where the last step is based on $L_{ii} = \sum_{j \neq i} |L_{ij}|$ and $\|L\|_\infty = o(1)$ given in Lemma F.1. By noting that Lemma F.1 shows $\text{tr}(L^2)^2 = o(\text{tr}(L^2))$, we obtain

$$\|\mathfrak{S}\|_F^2 = \text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 + o(\text{tr}(L^2)),$$

which leads to the first limit in (29).

To complete the proof, we must show that $\|\lambda(\mathfrak{S})\|_\infty^2 = o(\|\lambda(\mathfrak{S})\|_2^2)$. Note that the previous calculation gives $\|\lambda(\mathfrak{S})\|_2^2 = \|\mathfrak{S}\|_F^2 \asymp \text{tr}(L^2) \geq \|\mathbf{d}\|_2^2$, and so it is sufficient to demonstrate $\|\lambda(\mathfrak{S})\|_\infty \lesssim \|\mathbf{d}\|_\infty$, since $\|\mathbf{d}\|_\infty = o(\|\mathbf{d}\|_2)$ holds under the assumptions of Theorem 2. The definition of \mathfrak{S} in (25) implies that $\mathbf{E}(\varphi(\hat{e}_1)\varphi(\hat{e}_1)^\top) - \mathfrak{S}$ is positive semidefinite, and so $\lambda_{\max}(\mathfrak{S}) \leq \lambda_{\max}(\mathbf{E}(\varphi(\hat{e}_1)\varphi(\hat{e}_1)^\top))$. Since the inequality $\lambda_{\max}(A) \leq \max_{1 \leq i \leq d} \sum_{j=1}^d |A_{ij}|$ holds for any symmetric matrix $A \in \mathbb{R}^{d \times d}$, we have

$$\lambda_{\max}(\mathbf{E}(\varphi(\hat{e}_1)\varphi(\hat{e}_1)^\top)) \leq \max_{\mathbf{i} \in \mathcal{J}} \sum_{\mathbf{j} \in \mathcal{J}} |\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1))|.$$

Letting $\mathbf{i} = (i, i') \in \mathcal{J}$, Lemma F.5 gives for $i = i'$,

$$\begin{aligned} \sum_{\mathbf{j} \in \mathcal{J}} |\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1))| &= L_{ii} + \sum_{j \neq i} |L_{ij}| + \sqrt{2} \sum_{j > i} |L_{ij}| + \sqrt{2} \sum_{j < i} |L_{ij}| \\ &\lesssim \|\mathbf{d}\|_\infty, \end{aligned}$$

and for $i < i'$,

$$\begin{aligned} \sum_{\mathbf{j} \in \mathcal{J}} |\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1))| &= 2|L_{ii'}| + \sqrt{2}|L_{ii'}| + \sqrt{2}|L_{ii'}| \\ &\lesssim \|\mathbf{d}\|_\infty. \end{aligned}$$

Consequently, we conclude $\|\lambda(\mathfrak{S})\|_\infty \lesssim \|\mathbf{d}\|_\infty$. \square

Lemma F.5. *Let $\varphi_i(\hat{e}_1)$ be as defined in (23), and let $\mathbf{i} = (i, i'), \mathbf{j} = (j, j') \in \{(i, i') | 1 \leq i \leq i' \leq n\}$. If the conditions in Theorem 2 hold, then*

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = \begin{cases} L_{ii} & i = i' = j = j' \\ -L_{ij'} & i = i' \neq j = j' \\ -2L_{ij'} & i = j < i' = j' \\ \sqrt{2}L_{i'j'} & i = i' = j < j', j = j' = i < i' \\ \sqrt{2}L_{ij} & i = i' = j' > j, j = j' = i' > i \\ 0 & \text{otherwise.} \end{cases}$$

Proof. Observe that

$$\begin{aligned} \varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1) &= 1\{i = i' \in \hat{e}_1, j = j' \in \hat{e}_1\} + 2 \cdot 1\{i < i', \hat{e}_1 = \{i, i'\}, j < j', \hat{e}_1 = \{j, j'\}\} \\ &\quad - \sqrt{2} \cdot 1\{i = i' \in \hat{e}_1, j < j', \hat{e}_1 = \{j, j'\}\} - \sqrt{2} \cdot 1\{j = j' \in \hat{e}_1, i < i', \hat{e}_1 = \{i, i'\}\}, \end{aligned}$$

and for any \mathbf{i}, \mathbf{j} , only one term on the right side is nonzero at most. To make the first term nonzero, there are two possible cases: $i = i' = j = j'$ and $i = i' \neq j = j'$. When $i = i' = j = j'$, we have

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = \mathbf{E}(1\{i \in \hat{e}_1\}) = L_{ii}.$$

For $i = i' \neq j = j'$, we obtain

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = \mathbf{E}(1\{\hat{e}_1 = \{i, j'\}\}) = -L_{ij'}.$$

To make the second term nonzero, we need $i = j < i' = j'$, and the corresponding expectation is

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = 2\mathbf{E}(1\{\hat{e}_1 = \{i, j'\}\}) = -2L_{ij'}.$$

To make the third term nonzero, there are also two possible ways: $i = i' = j < j'$ and $i = i' = j' > j$. When $i = i' = j < j'$, the corresponding term can be calculated as

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = -\sqrt{2}\mathbf{E}(1\{\hat{e}_1 = \{i', j'\}\}) = \sqrt{2}L_{i'j'}.$$

When $i = i' = j' > j$, the corresponding term is

$$\mathbf{E}(\varphi_{\mathbf{i}}(\hat{e}_1)\varphi_{\mathbf{j}}(\hat{e}_1)) = -\sqrt{2}\mathbf{E}(1\{\hat{e}_1 = \{i, j\}\}) = \sqrt{2}L_{ij}.$$

The fourth term on the right side can be handled in the same way as the third term. For the other cases, all four terms are zero, and thus, the corresponding expectation is zero. \square

F.1.1 Consistency of the mean estimate

Lemma F.6. *If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$\frac{\hat{\mu} - \mu}{\sigma} \xrightarrow{\mathbf{P}} 0.$$

Proof. First recall that Lemma F.9 gives $\sigma^2 \asymp \text{tr}(L^2)/N^2$. In light of the bias-variance decomposition for the mean-squared error of $\hat{\mu}$, it is sufficient to show that $N^2(E(\hat{\mu}) - \mu)^2/\text{tr}(L^2)$ and $N^2 \text{var}(\hat{\mu})/\text{tr}(L^2)$ are both $o(1)$. With regard to the bias, observe that

$$\hat{\mu} - \mu = \frac{1}{B} \sum_{b=1}^B \left(\|\hat{L}_b^* - \hat{L}\|_F^2 - \mathbf{E}(\|\hat{L} - L\|_F^2) \right).$$

Combining the definitions of \hat{L}^* and \hat{L} with the calculation in (22) gives

$$\mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2 | Q) = \frac{4}{N} - \frac{1}{N} \text{tr}(\hat{L}^2). \quad (30)$$

To deal with $\text{tr}(\hat{L}^2)$, the identity $\text{tr}(\hat{L}^2) = -\text{tr}(L^2) + 2\langle \hat{L}, L \rangle + \|\hat{L} - L\|_F^2$ and the calculation in (22) can be used to show that

$$\mathbf{E}(\text{tr}(\hat{L}^2)) = \frac{4}{N} + \frac{N-1}{N} \text{tr}(L^2) \quad (31)$$

and so

$$\mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2) = \frac{N-1}{N^2} (4 - \text{tr}(L^2)).$$

Consequently, we have the following formula for the bias of $\hat{\mu}$,

$$\begin{aligned} |\mathbf{E}(\hat{\mu}) - \mu| &= \left| \frac{\text{tr}(L^2) - 4}{N^2} \right| \\ &= o\left(\frac{\sqrt{\text{tr}(L^2)}}{N}\right), \end{aligned}$$

where the second step is due to Lemma F.1. To analyze the variance of $\hat{\mu}$, equations (44) and (45) in the proof of Lemma F.10 give

$$\begin{aligned} \text{var}(\hat{\mu}) &= \text{var}\left(\frac{1}{B} \sum_{b=1}^B \|\hat{L}_b^* - \hat{L}\|_F^2\right) \\ &= \mathbf{E}\left(\frac{1}{B} \text{var}(\|\hat{L}^* - \hat{L}\|_F^2 | Q)\right) + \text{var}\left(\mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2 | Q)\right) \\ &= o\left(\frac{\text{tr}(L^2)}{N^2}\right), \end{aligned} \quad (32)$$

which completes the proof. \square

F.1.2 Consistency of the variance estimate

Lemma F.7. *If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$\frac{\hat{\sigma}^2}{\sigma^2} \xrightarrow{\mathbf{P}} 1.$$

Proof. Due to the fact that variance is shift invariant, note that the estimate $\hat{\sigma}^2$ can be written as

$$\hat{\sigma}^2 = \frac{1}{B} \sum_{b=1}^B \left(\|\hat{L}_b^* - \hat{L}\|_F^2 - \mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2) \right)^2 - \left(\frac{1}{B} \sum_{b=1}^B \|\hat{L}_b^* - \hat{L}\|_F^2 - \mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2) \right)^2. \quad (33)$$

Lemma F.9 ensures that $\sigma^2 \asymp \text{tr}(L^2)/N^2$, and so it suffices to show that the bias and variance of $\hat{\sigma}^2$ satisfy $|\mathbf{E}(\hat{\sigma}^2) - \sigma^2| = o(\text{tr}(L^2)/N^2)$ and $\text{var}(\hat{\sigma}^2) = o(\text{tr}(L^2)^2/N^4)$.

Towards calculating the bias of $\hat{\sigma}^2$, we first calculate its expectation. Due to the fact that each $\|\hat{L}_b^* - \hat{L}\|_F^2 - \mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2)$ is centered, we have

$$\begin{aligned} \mathbf{E}(\hat{\sigma}^2) &= \text{var}(\|\hat{L}^* - \hat{L}\|_F^2) - \text{var}\left(\frac{1}{B} \sum_{b=1}^B \|\hat{L}_b^* - \hat{L}\|_F^2\right) \\ &= \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right), \end{aligned}$$

where the second step follows from Lemma F.10 and (32). Next, Lemma F.9 shows that

$$\sigma^2 = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right)$$

and so the bias of $\hat{\sigma}^2$ satisfies

$$|\mathbf{E}(\hat{\sigma}^2) - \sigma^2| = o\left(\frac{\text{tr}(L^2)}{N^2}\right),$$

as needed.

Now we turn to the task of bounding the variance of $\hat{\sigma}^2$. For each $b = 1, \dots, B$, define the random variable

$$\xi_b^* = \|\hat{L}_b^* - \hat{L}\|_F^2 - \mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2 | Q). \quad (34)$$

It will be helpful to note that $\hat{\sigma}^2$ can also be expressed as

$$\hat{\sigma}^2 = \frac{1}{B} \sum_{b=1}^B (\xi_b^*)^2 - \left(\frac{1}{B} \sum_{b=1}^B \xi_b^* \right)^2. \quad (35)$$

due shift invariance. Using the bound $\text{var}(X + Y) \leq 2 \text{var}(X) + 2 \text{var}(Y)$ for generic random variables X and Y , we have

$$\text{var}(\hat{\sigma}^2) \lesssim \text{var}\left(\frac{1}{B} \sum_{b=1}^B (\xi_b^*)^2\right) + \text{var}\left(\frac{1}{B^2} \sum_{1 \leq b \neq b' \leq B} \xi_b^* \xi_{b'}^*\right).$$

Using the fact that ξ_1^*, \dots, ξ_B^* are conditionally i.i.d. given Q_1, \dots, Q_N , we apply the law of total variance to each of the terms above, yielding

$$\begin{aligned} \text{var}(\hat{\sigma}^2) &\lesssim \frac{1}{B} \mathbf{E}\left(\text{var}((\xi_1^*)^2 | Q)\right) + \text{var}\left(\mathbf{E}((\xi_1^*)^2 | Q)\right) + \frac{1}{B^2} \mathbf{E}\left(\text{var}(\xi_1^* \xi_2^* | Q)\right) \\ &\lesssim \frac{1}{B} \mathbf{E}\left(\text{var}((\xi_1^*)^2 | Q)\right) + \mathbf{E}\left(\left(\mathbf{E}((\xi_1^*)^2 | Q)\right)^2\right) - \left(\mathbf{E}((\xi_1^*)^2)\right)^2 + \frac{1}{B^2} \mathbf{E}\left(\left(\mathbf{E}((\xi_1^*)^2 | Q)\right)^2\right). \end{aligned}$$

Applying Lemmas F.10, F.11 and F.13 implies $\text{var}(\hat{\sigma}^2) = o\left(\frac{\text{tr}(L^2)^2}{N^4}\right)$. \square

F.2 Conditional asymptotic normality of $\|\hat{L}^* - \hat{L}\|_F^2$

Recall that $\hat{\mathbf{d}} \in \mathbb{R}^n$ is defined to contain the diagonal entries of \hat{L} . As explained on page 22, the proof of the limit (20) reduces to the following lemma.

Lemma F.8. *If the conditions in Theorem 2 hold, then as $n \rightarrow \infty$,*

$$\frac{\|\hat{\mathbf{d}}\|_\infty}{\|\hat{\mathbf{d}}\|_2} \xrightarrow{\mathbf{P}} 0.$$

Proof. By writing $\frac{\|\hat{\mathbf{d}}\|_\infty}{\|\hat{\mathbf{d}}\|_2} = \frac{\|\hat{\mathbf{d}}\|_\infty}{\|\mathbf{d}\|_2} \frac{\|\mathbf{d}\|_2}{\|\hat{\mathbf{d}}\|_2}$, it is enough to establish the limits $\|\hat{\mathbf{d}}\|_\infty / \|\mathbf{d}\|_2 \xrightarrow{\mathbf{P}} 0$ and $\|\hat{\mathbf{d}}\|_2 / \|\mathbf{d}\|_2 \xrightarrow{\mathbf{P}} 1$. With regard to the second limit, note that

$$\left(\|\hat{\mathbf{d}}\|_2 - \|\mathbf{d}\|_2\right)^2 \leq \|\hat{\mathbf{d}} - \mathbf{d}\|_2^2 \leq \|\hat{L} - L\|_F^2,$$

and so equation (22) gives

$$\mathbf{E}\left(\left(\|\hat{\mathbf{d}}\|_2 - \|\mathbf{d}\|_2\right)^2\right) \lesssim \frac{1}{N}.$$

Next, we will show that $\|\mathbf{d}\|_2^2 \gtrsim \text{tr}(L^2)$ so that Lemma F.1 will imply

$$\mathbf{E}\left(\left(\frac{\|\hat{\mathbf{d}}\|_2}{\|\mathbf{d}\|_2} - 1\right)^2\right) \lesssim \frac{1}{N\text{tr}(L^2)} = o(1), \quad (36)$$

yielding limit $\|\hat{\mathbf{d}}\|_2/\|\mathbf{d}\|_2 \xrightarrow{\mathbf{P}} 1$. To show the lower bound $\|\mathbf{d}\|_2^2 \gtrsim \text{tr}(L^2)$, observe that

$$\|\mathbf{d}\|_2^2 = \sum_{i=1}^n \left(\sum_{j \neq i} L_{ij} \right)^2 \geq \sum_{i=1}^n \sum_{j \neq i} L_{ij}^2 = \text{tr}(L^2) - \|\mathbf{d}\|_2^2,$$

and so rearranging implies $\|\mathbf{d}\|_2^2 \geq \frac{1}{2}\text{tr}(L^2)$.

Now we turn to proving the limit $\|\hat{\mathbf{d}}\|_\infty/\|\mathbf{d}\|_2 \xrightarrow{\mathbf{P}} 0$. Consider the basic inequality

$$\frac{\|\hat{\mathbf{d}}\|_\infty}{\|\mathbf{d}\|_2} \leq \frac{\|\mathbf{d}\|_\infty}{\|\mathbf{d}\|_2} + \frac{\|\hat{\mathbf{d}} - \mathbf{d}\|_\infty}{\|\mathbf{d}\|_2}.$$

The conditions of Theorem 2 ensure that the first term on the right is $o(1)$. Meanwhile, the second term is at most $\|\hat{\mathbf{d}} - \mathbf{d}\|_2/\|\mathbf{d}\|_2$, and our earlier work shows that the expectation of this quantity is $\mathcal{O}\left(\sqrt{\mathbf{E}(\|\hat{L} - L\|_F^2)/\|\mathbf{d}\|_2^2}\right) = \mathcal{O}(1/\sqrt{N\text{tr}(L^2)}) = o(1)$. \square

F.3 Moments

Lemma F.9. *If the conditions in Theorem 2 hold, then*

$$\text{var}(\|\hat{L} - L\|_F^2) = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right),$$

and in particular

$$\text{var}(\|\hat{L} - L\|_F^2) \asymp \frac{\text{tr}(L^2)}{N^2}.$$

Proof. Note that

$$\|\hat{L} - L\|_F^2 = \frac{1}{N^2} \sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^2 - \frac{2}{N} \sum_{i=1}^N D_i^\top L D_i + \frac{4}{N} + \text{tr}(L^2).$$

Since the last two terms on the right side are constants, we only need to handle the first two terms. It follows that

$$\begin{aligned} \text{var}(\|\hat{L} - L\|_F^2) &= \frac{4}{N^4} \sum_{1 \leq i \neq j \neq k \leq N} \text{cov}\left((D_i^\top D_j)^2, (D_i^\top D_k)^2\right) + \frac{2}{N^4} \sum_{1 \leq i \neq j \leq N} \text{var}\left((D_i^\top D_j)^2\right) \\ &\quad - \frac{8}{N^3} \sum_{1 \leq i \neq j \leq N} \text{cov}\left((D_i^\top D_j)^2, D_i^\top L D_i\right) + \frac{4}{N^2} \sum_{i=1}^N \text{var}\left(D_i^\top L D_i\right). \end{aligned}$$

When $i \neq j \neq k$, it follows from the independence of D_i , D_j , and D_k that the quantities $\text{cov}\left((D_i^\top D_j)^2, (D_i^\top D_k)^2\right)$, $\text{cov}\left((D_i^\top D_j)^2, D_i^\top L D_i\right)$, and $\text{var}\left(D_i^\top L D_i\right)$ are all equal. Consequently, we can obtain

$$\text{var}(\|\hat{L} - L\|_F^2) = \frac{2(N-1)}{N^3} \text{var}\left((D_1^\top D_2)^2\right) + \frac{4(-N+2)}{N^3} \text{cov}\left((D_1^\top D_2)^2, (D_1^\top D_3)^2\right). \quad (37)$$

It is direct to calculate

$$\begin{aligned}\mathbf{E}\left((D_1^\top D_2)^2\right) &= \text{tr}(L^2), \\ \mathbf{E}\left((D_1^\top D_2)^2(D_1^\top D_3)^2\right) &= \sum_{1 \leq i < j \leq n} |L_{ij}|(L_{ii} + L_{jj} + 2|L_{ij}|)^2, \\ \mathbf{E}\left((D_1^\top D_2)^4\right) &= \text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2.\end{aligned}\tag{38}$$

Based on Lemma F.1, the orders of the above three terms are

$$\begin{aligned}\left(\mathbf{E}\left((D_1^\top D_2)^2\right)\right)^2 &= o(\text{tr}(L^2)), \\ \mathbf{E}\left((D_1^\top D_2)^2(D_1^\top D_3)^2\right) &= o(\text{tr}(L^2)), \\ \mathbf{E}\left((D_1^\top D_2)^4\right) &\asymp \text{tr}(L^2).\end{aligned}\tag{39}$$

Applying the above results to (37) completes the proof of the lemma. \square

Lemma F.10. *Let ξ_1^* be as defined in (34). If the conditions in Theorem 2 hold, then*

$$\text{var}\left(\|\hat{L}^* - \hat{L}\|_F^2\right) = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right)$$

and

$$\mathbf{E}\left((\xi_1^*)^2\right) = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right).$$

Proof. We begin with the law of total variance

$$\text{var}\left(\|\hat{L}^* - \hat{L}\|_F^2\right) = \mathbf{E}\left(\text{var}\left(\|\hat{L}^* - \hat{L}\|_F^2 | Q\right)\right) + \text{var}\left(\mathbf{E}\left(\|\hat{L}^* - \hat{L}\|_F^2 | Q\right)\right).\tag{40}$$

For the first term on the right side, following the proof of Lemma F.9 yields

$$\begin{aligned}\text{var}\left(\|\hat{L}^* - \hat{L}\|_F^2 | Q\right) &= \frac{2(N-1)}{N^3} \left(\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^4 | Q\right) - \left(\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 | Q\right) \right)^2 \right) \\ &\quad + \frac{4(2-N)}{N^3} \left(\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 \langle D_1^*, D_3^* \rangle^2 | Q\right) - \left(\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 | Q\right) \right)^2 \right),\end{aligned}\tag{41}$$

where

$$\begin{aligned}\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 | Q\right) &= \text{tr}(\hat{L}^2) \\ \mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 \langle D_1^*, D_3^* \rangle^2 | Q\right) &= \frac{1}{N^3} \sum_{i,j,k=1}^N (D_i^\top D_j)^2 (D_i^\top D_k)^2 \\ \mathbf{E}\left(\langle D_1^*, D_2^* \rangle^4 | Q\right) &= \frac{1}{N^2} \sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^4 + \frac{16}{N}.\end{aligned}\tag{42}$$

To bound $\mathbf{E}\left(\text{var}\left(\|\hat{L}^* - \hat{L}\|_F^2 | Q\right)\right)$, we will analyze the order of the expectation of the above three terms. Lemmas F.1 and F.14 give $\mathbf{E}\left(\left(\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 | Q\right)\right)^2\right) = o(\text{tr}(L^2))$. Lemma F.1 as well as equations (38) and (39) imply

$$\begin{aligned}\mathbf{E}\left(\langle D_1^*, D_2^* \rangle^2 \langle D_1^*, D_3^* \rangle^2\right) &\lesssim \mathbf{E}\left((D_1^\top D_2)^2 (D_1^\top D_3)^2\right) + \frac{1}{N} \mathbf{E}\left((D_1^\top D_2)^4\right) + \frac{1}{N} \mathbf{E}\left((D_1^\top D_2)^2\right) + \frac{1}{N^2} \\ &= o(\text{tr}(L^2)) \\ \mathbf{E}\left(\langle D_1^*, D_2^* \rangle^4\right) &= \text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 + o(\text{tr}(L^2)).\end{aligned}\tag{43}$$

Therefore, we have

$$\mathbf{E}(\text{var}(\|\hat{L}^* - \hat{L}\|_F^2 | Q)) = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right). \quad (44)$$

For the second term on the right side of (40), Lemmas F.1 and F.14 as well as (30) lead to

$$\begin{aligned} \text{var}(\mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2 | Q)) &= \frac{1}{N^2} \text{var}(\text{tr}(\hat{L}^2)) \\ &= o\left(\frac{\text{tr}(L^2)}{N^2}\right). \end{aligned} \quad (45)$$

Combining the above results yields

$$\text{var}(\|\hat{L}^* - \hat{L}\|_F^2) = \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right)$$

and

$$\begin{aligned} \mathbf{E}((\xi_1^*)^2) &= \text{var}(\|\hat{L}^* - \hat{L}\|_F^2) - \text{var}(\mathbf{E}(\|\hat{L}^* - \hat{L}\|_F^2 | Q)) \\ &= \frac{2}{N^2} \left(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2 \right) + o\left(\frac{\text{tr}(L^2)}{N^2}\right). \end{aligned}$$

□

Lemma F.11. *Let ξ_1^* be as defined in (34). If the conditions in Theorem 2 hold, then*

$$\mathbf{E}(\text{var}((\xi_1^*)^2 | Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}.$$

Proof. Letting $V_i^* = \frac{1}{N}(D_i^*(D_i^*)^\top - \hat{L})$, equation (30) implies the random variable ξ_1^* can be decomposed as

$$\begin{aligned} \xi_1^* &= \sum_{i,j=1}^N \langle V_i^*, V_j^* \rangle - \frac{1}{N}(4 - \text{tr}(\hat{L}^2)) \\ &= \gamma_1^* + \gamma_2^*, \end{aligned}$$

where

$$\gamma_1^* = \sum_{1 \leq i \neq j \leq N} \langle V_i^*, V_j^* \rangle, \quad \gamma_2^* = -\frac{2}{N^2} \sum_{i=1}^N ((D_i^*)^\top \hat{L} D_i^* - \text{tr}(\hat{L}^2)). \quad (46)$$

Note that the following relation holds almost surely,

$$\begin{aligned} \text{var}((\xi_1^*)^2 | Q) &= \text{var}((\gamma_1^*)^2 + (\gamma_2^*)^2 + 2\gamma_1^* \gamma_2^* | Q) \\ &\lesssim \text{var}((\gamma_1^*)^2 | Q) + \text{var}((\gamma_2^*)^2 | Q) + \text{var}(\gamma_1^* \gamma_2^* | Q). \end{aligned} \quad (47)$$

Since Lemma F.12 proves $\mathbf{E}(\text{var}((\gamma_1^*)^2 | Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}$, it remains to show the expectations of the other terms on the right side are $\mathcal{O}(\frac{\text{tr}(L^2)^2}{N^4})$. For $\mathbf{E}(\text{var}((\gamma_2^*)^2 | Q))$, we will bound it through analyzing $\mathbf{E}((\gamma_2^*)^4)$. By noting that $(D_1^*)^\top \hat{L} D_1^* \leq 4$, $\text{tr}(\hat{L}^2) \leq 4$ and $|\gamma_2^*| \leq \frac{16}{N}$, Lemma F.14 gives

$$\begin{aligned} \mathbf{E}((\gamma_2^*)^4) &\lesssim \frac{1}{N^2} \mathbf{E}(\mathbf{E}((\gamma_2^*)^2 | Q)) \\ &\lesssim \frac{1}{N^6} \sum_{i,j=1}^N \mathbf{E}(\mathbf{E}(((D_i^*)^\top \hat{L} D_i^* - \text{tr}(\hat{L}^2))((D_j^*)^\top \hat{L} D_j^* - \text{tr}(\hat{L}^2)) | Q)) \\ &\lesssim \frac{1}{N^5} \mathbf{E}(\mathbf{E}(((D_1^*)^\top \hat{L} D_1^* - \text{tr}(\hat{L}^2))^2 | Q)) \\ &\lesssim \frac{1}{N^5} \mathbf{E}((D_1^*)^\top \hat{L} D_1^* + \text{tr}(\hat{L}^2)) \\ &\lesssim \frac{\text{tr}(L^2)}{N^5}. \end{aligned}$$

Using $N\text{tr}(L^2) \rightarrow \infty$ as $n \rightarrow \infty$ from Lemma F.1, it follows that

$$\mathbf{E}(\text{var}((\gamma_2^*)^2|Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}. \quad (48)$$

It remains to analyze $\mathbf{E}(\text{var}(\gamma_1^*\gamma_2^*|Q))$. By noting that

$$\begin{aligned} \gamma_1^*\gamma_2^* &= -\frac{2}{N^2} \left(\sum_{1 \leq i \neq j \neq k \leq N} \langle V_i^*, V_j^* \rangle ((D_k^*)^\top \hat{L} D_k^* - \text{tr}(\hat{L}^2)) \right. \\ &\quad \left. + 2 \sum_{1 \leq i \neq j \leq N} \langle V_i^*, V_j^* \rangle ((D_i^*)^\top \hat{L} D_i^* - \text{tr}(\hat{L}^2)) \right), \end{aligned}$$

the fact that $(D_1^*)^\top \hat{L} D_1^* \leq 4$ and $\text{tr}(\hat{L}^2) \leq 4$ hold almost surely implies

$$\begin{aligned} \text{var}(\gamma_1^*\gamma_2^*|Q) &\lesssim \frac{1}{N^4} \text{var} \left(\sum_{1 \leq i \neq j \neq k \leq N} \langle V_i^*, V_j^* \rangle ((D_k^*)^\top \hat{L} D_k^* - \text{tr}(\hat{L}^2)) | Q \right) \\ &\quad + \frac{1}{N^4} \text{var} \left(\sum_{1 \leq i \neq j \leq N} \langle V_i^*, V_j^* \rangle ((D_i^*)^\top \hat{L} D_i^* - \text{tr}(\hat{L}^2)) | Q \right) \\ &\lesssim \frac{1}{N} \mathbf{E} \left(\langle V_1^*, V_2^* \rangle^2 \left(((D_3^*)^\top \hat{L} D_3^* - \text{tr}(\hat{L}^2))^2 + ((D_1^*)^\top \hat{L} D_1^* - \text{tr}(\hat{L}^2))^2 \right) | Q \right) \\ &\lesssim \frac{1}{N} \mathbf{E} \left(\langle V_1^*, V_2^* \rangle^2 | Q \right). \end{aligned}$$

Combining with the bound for $\mathbf{E}(\langle V_1^*, V_2^* \rangle^2)$ given in (54) yields

$$\mathbf{E}(\text{var}(\gamma_1^*\gamma_2^*|Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}. \quad (49)$$

Applying equations (48) and (49) to (47) yields the stated result. \square

Lemma F.12. *Let γ_1^* be as defined in (46). If the conditions in Theorem 2 hold, then*

$$\mathbf{E}(\text{var}((\gamma_1^*)^2|Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}.$$

Proof. To analyze $\mathbf{E}(\text{var}((\gamma_1^*)^2|Q))$, note that

$$\begin{aligned} (\gamma_1^*)^2 &= \sum_{1 \leq i \neq j \neq k \neq l \leq N} \langle V_i^*, V_j^* \rangle \langle V_k^*, V_l^* \rangle + 4 \sum_{1 \leq i \neq j \neq k \leq N} \langle V_i^*, V_j^* \rangle \langle V_i^*, V_k^* \rangle + 2 \sum_{1 \leq i \neq j \leq N} \langle V_i^*, V_j^* \rangle^2 \\ &=: \gamma_{11}^* + 4\gamma_{12}^* + 2\gamma_{13}^*, \end{aligned}$$

which implies

$$\mathbf{E}(\text{var}((\gamma_1^*)^2|Q)) \lesssim \mathbf{E}(\text{var}(\gamma_{11}^*|Q)) + \mathbf{E}(\text{var}(\gamma_{12}^*|Q)) + \mathbf{E}(\text{var}(\gamma_{13}^*|Q)). \quad (50)$$

We know

$$\begin{aligned} \text{var}(\gamma_{11}^*|Q) &= \sum_{1 \leq i \neq j \neq k \neq l \leq N} 8 \text{var} \left(\langle V_i^*, V_j^* \rangle \langle V_k^*, V_l^* \rangle | Q \right) \\ &\quad + \sum_{1 \leq i \neq j \neq k \neq l \leq N} 16 \text{cov} \left(\langle V_i^*, V_j^* \rangle \langle V_k^*, V_l^* \rangle, \langle V_i^*, V_k^* \rangle \langle V_j^*, V_l^* \rangle | Q \right) \\ &\leq 24N^4 \mathbf{E} \left(\langle V_1^*, V_2^* \rangle^2 \langle V_3^*, V_4^* \rangle^2 | Q \right) \\ &= 24N^4 \left(\mathbf{E} \left(\langle V_1^*, V_2^* \rangle^2 | Q \right) \right)^2. \end{aligned}$$

To bound $\mathbf{E}(\langle V_1^*, V_2^* \rangle^2|Q)$, note that the following inequalities hold almost surely for all $i, j \in \{1, \dots, N\}$:

$$\begin{aligned} ((D_i^*)^\top D_j^*)^2 &\leq 4, \\ (D_i^*)^\top \hat{L} D_i^* &\leq 4, \\ \text{tr}(\hat{L}^2) &\leq 4. \end{aligned}$$

Hence, for any $i, j \in \{1, \dots, N\}$, we have

$$\begin{aligned} \|\langle V_i^*, V_j^* \rangle\| &\leq \frac{1}{N^2} \left(\langle D_i^*, D_j^* \rangle^2 + (D_i^*)^\top \hat{L} D_i^* + (D_j^*)^\top \hat{L} D_j^* + \text{tr}(\hat{L}^2) \right) \\ &\leq \frac{16}{N^2} \end{aligned} \quad (51)$$

and

$$\mathbf{E}(\|\langle V_1^*, V_2^* \rangle\| | Q) \leq \frac{4}{N^2} \text{tr}(\hat{L}^2) \quad (52)$$

hold almost surely. Combining above results with Lemma F.14 gives

$$\begin{aligned} \mathbf{E}(\text{var}(\gamma_{11}^* | Q)) &\lesssim N^4 \mathbf{E}\left(\frac{\text{tr}(\hat{L}^2)^2}{N^8}\right) \\ &\lesssim \frac{\text{tr}(L^2)^2}{N^4}. \end{aligned} \quad (53)$$

To analyze $\mathbf{E}(\text{var}(\gamma_{12}^* | Q))$, the definition of γ_{12}^* gives

$$\begin{aligned} \text{var}(\gamma_{12}^* | Q) &= \sum_{1 \leq i \neq j \neq k \neq l \leq N} 2 \text{cov}\left(\langle V_i^*, V_j^* \rangle \langle V_i^*, V_k^* \rangle, \langle V_j^*, V_l^* \rangle \langle V_k^*, V_l^* \rangle | Q\right) \\ &\quad + \sum_{1 \leq i \neq j \neq k \leq N} 2 \text{var}\left(\langle V_i^*, V_j^* \rangle \langle V_i^*, V_k^* \rangle | Q\right) \\ &\quad + \sum_{1 \leq i \neq j \neq k \leq N} 4 \text{cov}\left(\langle V_i^*, V_j^* \rangle \langle V_i^*, V_k^* \rangle, \langle V_i^*, V_k^* \rangle \langle V_j^*, V_k^* \rangle | Q\right) \\ &\leq 2N^4 \mathbf{E}\left(\langle V_1^*, V_2^* \rangle \langle V_1^*, V_3^* \rangle \langle V_2^*, V_4^* \rangle \langle V_3^*, V_4^* \rangle | Q\right) \\ &\quad + 6N^3 \mathbf{E}\left(\langle V_1^*, V_2^* \rangle^2 \langle V_1^*, V_3^* \rangle^2 | Q\right). \end{aligned}$$

Lemma F.14, (51) and (52) imply

$$\begin{aligned} N^4 \left| \mathbf{E}\left(\langle V_1^*, V_2^* \rangle \langle V_1^*, V_3^* \rangle \langle V_2^*, V_4^* \rangle \langle V_3^*, V_4^* \rangle\right) \right| &\lesssim \mathbf{E}\left(\|\langle V_1^*, V_2^* \rangle\| | Q\right) \mathbf{E}\left(\|\langle V_3^*, V_4^* \rangle\| | Q\right) \\ &\lesssim \frac{1}{N^4} \mathbf{E}(\text{tr}(\hat{L}^2)^2) \\ &\lesssim \frac{\text{tr}(L^2)^2}{N^4}. \end{aligned}$$

Since Lemmas F.1 and F.14 as well as (43) and (51) imply

$$\begin{aligned} \mathbf{E}\left(\langle V_1^*, V_2^* \rangle^4\right) &\lesssim \frac{1}{N^4} \mathbf{E}\left(\langle V_1^*, V_2^* \rangle^2\right) \\ &\lesssim \frac{1}{N^8} \mathbf{E}\left(\langle D_1^*, D_2^* \rangle^4 + ((D_1^*)^\top \hat{L} D_1^*)^2 + \text{tr}(\hat{L}^2)^2\right) \\ &\lesssim \frac{\text{tr}(L^2)^2}{N^7}, \end{aligned} \quad (54)$$

we have

$$N^3 \mathbf{E}\left(\langle V_1^*, V_2^* \rangle^2 \langle V_1^*, V_3^* \rangle^2\right) \lesssim \frac{\text{tr}(L^2)^2}{N^4}.$$

Combining the above results yields

$$\mathbf{E}(\text{var}(\gamma_{12}^* | Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}. \quad (55)$$

To analyze $\mathbf{E}(\text{var}(\gamma_{13}^* | Q))$, note that

$$\begin{aligned} \text{var}(\gamma_{13}^* | Q) &= \sum_{1 \leq i \neq j \neq k \leq N} 4 \text{cov}\left(\langle V_i^*, V_j^* \rangle^2, \langle V_i^*, V_k^* \rangle^2 | Q\right) + \sum_{1 \leq i \neq j \leq N} 2 \text{var}\left(\langle V_i^*, V_j^* \rangle^2 | Q\right) \\ &\leq 6N^3 \mathbf{E}\left(\langle V_1^*, V_2^* \rangle^4 | Q\right). \end{aligned}$$

Equation (54) gives

$$\mathbf{E}(\text{var}(\gamma_{13}^*|Q)) \lesssim \frac{\text{tr}(L^2)^2}{N^4}. \quad (56)$$

Applying (53), (55) and (56) to (50) completes the proof. \square

Lemma F.13. *Let ξ_1^* be as defined in (34). If the conditions in Theorem 2 hold, then*

$$\begin{aligned} \mathbf{E}\left(\left(\mathbf{E}((\xi_1^*)^2|Q)\right)^2\right) &= \frac{4}{N^4}(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2)^2 + o\left(\frac{\text{tr}(L^2)^2}{N^4}\right) \\ &\asymp \frac{\text{tr}(L^2)^2}{N^4}. \end{aligned}$$

Proof. Note that the definition of ξ_1^* and equation (41) give

$$\begin{aligned} \mathbf{E}((\xi_1^*)^2|Q) &= \text{var}(\|\hat{L}^* - \hat{L}\|_F^2|Q) \\ &= \frac{2(N-1)}{N^5} \sum_{i,j=1}^N (D_i^\top D_j)^4 + \frac{4(2-N)}{N^6} \sum_{i,j,k=1}^N (D_i^\top D_j)^2 (D_i^\top D_k)^2 + \frac{2N-6}{N^3} \text{tr}(\hat{L}^2)^2. \end{aligned}$$

If we can show

$$\begin{aligned} \mathbf{E}\left(\left(\sum_{i,j=1}^N (D_i^\top D_j)^4\right)^2\right) &= N^4(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2)^2 + o(N^4 \text{tr}(L^2)^2) \\ \mathbf{E}\left(\left(\sum_{i,j,k=1}^N (D_i^\top D_j)^2 (D_i^\top D_k)^2\right)^2\right) &= o(N^6 \text{tr}(L^2)^2) \\ \mathbf{E}(\text{tr}(\hat{L}^2)^4) &= o(\text{tr}(L^2)^2), \end{aligned}$$

applying Hölder's inequality completes the proof of the lemma. Since the last statement is implied by Lemmas F.1 and F.14, we will show the first two equalities. Combining the fact $|D_i^\top D_j| \leq 2$ with Lemma F.1, (38) and (39) implies

$$\begin{aligned} \mathbf{E}\left(\left(\sum_{i,j=1}^N (D_i^\top D_j)^4\right)^2\right) &= \mathbf{E}\left(\left(\sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^4\right)^2\right) + 32N \mathbf{E}\left(\sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^4\right) + 216N^2 \\ &= \mathbf{E}\left(\sum_{1 \leq i \neq j \neq k \neq l \leq N} (D_i^\top D_j)^4 (D_k^\top D_l)^4\right) + 2 \mathbf{E}\left(\sum_{1 \leq i \neq j \leq N} (D_i^\top D_j)^8\right) \\ &\quad + 4 \mathbf{E}\left(\sum_{1 \leq i \neq j \neq k \leq N} (D_i^\top D_j)^4 (D_i^\top D_k)^4\right) + o(N^4 \text{tr}(L^2)^2) \\ &= N^4(\text{tr}(L^2) + 12 \sum_{1 \leq i < j \leq n} L_{ij}^2)^2 + o(N^4 \text{tr}(L^2)^2) \end{aligned}$$

and

$$\begin{aligned} \mathbf{E}\left(\left(\sum_{i,j,k=1}^N (D_i^\top D_j)^2 (D_i^\top D_k)^2\right)^2\right) &\leq \mathbf{E}\left(\left(\mathcal{O}(N^2) + \sum_{1 \leq i \neq j \neq k \leq N} (D_i^\top D_j)^2 (D_i^\top D_k)^2\right)^2\right) \\ &\lesssim \sum_{1 \leq i \neq j \neq k \neq l \neq r \neq s \leq N} (D_i^\top D_j)^2 (D_i^\top D_k)^2 (D_s^\top D_l)^2 (D_s^\top D_r)^2 \\ &\quad + N^5 \mathbf{E}\left((D_1^\top D_2)^4 (D_1^\top D_3)^4\right) + N^4 \\ &= o(N^6 \text{tr}(L^2)^2). \end{aligned}$$

\square

Lemma F.14. *If the conditions in Theorem 2 hold, then for any fixed integer $l \geq 1$ not depending on n , we have*

$$\mathbf{E}(\|\hat{L}\|_F^{2l}) \lesssim \|L\|_F^{2l}.$$

Proof. For $l = 1$, Lemma F.1 and (31) imply $\mathbf{E}(\|\hat{L}\|_F^2) \lesssim \|L\|_F^2$. We will use strong induction to complete the proof. Let $l > 1$ and assume $\mathbf{E}(\|\hat{L}\|_F^{2k}) \lesssim \|L\|_F^{2k}$ holds for all integer $0 \leq k < l$. By noting that

$$\mathbf{E}(\|\hat{L}\|_F^{2l}) = \frac{1}{N^{2l}} \sum_{i_1, \dots, i_l, j_1, \dots, j_l=1}^N \mathbf{E}(\Pi_{h=1}^l (D_{i_h}^\top D_{j_h})^2)$$

and $\mathbf{E}(\Pi_{h=1}^l (D_{i_h}^\top D_{j_h})^2) = \|L\|_F^{2l}$ for $1 \leq i_1 \neq \dots \neq i_l \neq j_1 \neq \dots \neq j_l \leq N$, we only need to show that the summation over $i_1, \dots, i_l, j_1, \dots, j_l$ which are not all distinct can be bounded by $N^{2l} \|L\|_F^{2l}$. Due to $|D_i^\top D_j| \leq 2$ for any $i, j = 1, \dots, N$, if there are terms involving a same index, we will only keep one of them and use the bound $|D_i^\top D_j| \leq 2$ for the others. Finally, there are d pairs remaining and the others are changed to be a constant, where $0 \leq d < l$. Without loss of generality, we consider the first d pairs to all be distinct, while the remaining pairs, which involve at least one index with repetition, are changed to 2. Combining with Lemma F.1 gives

$$\begin{aligned} \sum_{|\{i_1, \dots, i_l, j_1, \dots, j_l\}| < 2l} \mathbf{E}(\Pi_{h=1}^l (D_{i_h}^\top D_{j_h})^2) &\lesssim \sum_{d=0}^{l-1} \left(\sum_{1 \leq i_1 \neq \dots \neq j_d \leq N} N^{l-d} \mathbf{E}(\Pi_{h=1}^d (D_{i_h}^\top D_{j_h})^2) \right) \\ &\lesssim \sum_{d=0}^{l-1} N^{l+d} \mathbf{E}(\|\hat{L}\|_F^{2d}) \\ &\lesssim N^{2l} \|L\|_F^{2l}, \end{aligned}$$

which completes the proof. \square

G Background results

Lemma G.1 (Chernozhuokov et al. (2022), Theorem 2.1). *Let X_1, \dots, X_n be independent random vectors in \mathbb{R}^p such that $\mathbf{E}(X_{ij}) = 0$ for all $i = 1, \dots, n$ and $j = 1, \dots, p$. Let (G_1, \dots, G_p) be a Gaussian random vector in \mathbb{R}^p with mean 0 and covariance matrix $\frac{1}{n} \sum_{i=1}^n \mathbf{E}(X_i X_i^\top)$, and define $M(G) = \max_{1 \leq j \leq p} |G_j|$. Let b_1 and b_2 be some strictly positive constants such that $b_1 \leq b_2$ and let $\{c_n\}_{n \geq 1}$ be a sequence of constants such that $c_n \geq 1$. If for all $i = 1, \dots, n$ and $j = 1, \dots, p$, we have*

$$\mathbf{E}\left(\exp\left(\frac{|X_{ij}|}{c_n}\right)\right) \leq 2, \quad b_1^2 \leq \frac{1}{n} \sum_{i=1}^n \mathbf{E}(X_{ij}^2) \quad \text{and} \quad \frac{1}{n} \sum_{i=1}^n \mathbf{E}(X_{ij}^4) \leq b_2^2 c_n^2,$$

then

$$\sup_{t \in \mathbb{R}} \left| \mathbf{P}\left(\max_{1 \leq j \leq p} \left|\frac{1}{\sqrt{n}} \sum_{i=1}^n X_{ij}\right| \leq t\right) - \mathbf{P}(M(G) \leq t) \right| \leq \kappa \left(\frac{c_n^2 \log(2pn)^5}{n} \right)^{1/4},$$

where κ is a constant depending only on b_1 and b_2 .

Lemma G.2 (Chernozhuokov et al. (2022), Proposition 2.1). *Let V and W be mean-zero Gaussian vectors in \mathbb{R}^p with respective covariance matrices Σ and $\tilde{\Sigma}$. Also, assume that $\min_{1 \leq j \leq p} \Sigma_{jj} \geq \varsigma$ for some positive constant ς . Then,*

$$\sup_{t \in \mathbb{R}} \left| \mathbf{P}\left(\max_{1 \leq j \leq p} |V_j| \leq t\right) - \mathbf{P}\left(\max_{1 \leq j \leq p} |W_j| \leq t\right) \right| \leq C \log(2p) \|\Sigma - \tilde{\Sigma}\|_\infty^{1/2}$$

where C is a positive constant depending only on ς .

Lemma G.3 (Nazarov (2003), Nazarov's inequality). *Let $\epsilon > 0$, and let V be a mean-zero Gaussian random vector in \mathbb{R}^p with $\mathbf{E}(V_j^2) = 1$ for all $j = 1, \dots, p$. Then,*

$$\sup_{r \in \mathbb{R}} \mathbf{P}\left(\max_{1 \leq j \leq p} |V_j| - r \leq \epsilon\right) \leq 2\epsilon \left(\sqrt{2 \log(2p)} + 2 \right).$$

Lemma G.4. *If V and W are random variables, then for any $\delta > 0$,*

$$\sup_{t \in \mathbb{R}} \mathbf{P}(|V - t| \leq \delta) \leq \sup_{t \in \mathbb{R}} \mathbf{P}(|W - t| \leq \delta) + 2d_K(\mathcal{L}(V), \mathcal{L}(W)).$$

Lemma G.5 (Huang et al. (2023), Proposition 9). *Let X_1, \dots, X_n be i.i.d. random vectors in \mathbb{R}^p , and let h be a function $h : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$ satisfying $\mathbf{E}(h(X_1, X_2)|X_1) = 0$. Suppose there is a sequence of functions $\phi_1, \dots, \phi_K : \mathbb{R}^p \rightarrow \mathbb{R}$ such that h can be represented as*

$$h(x, x') = \sum_{k=1}^K \phi_k(x) \phi_k(x')$$

for all $x, x' \in \mathbb{R}^p$. Also, let $\phi(X_1) = (\phi_1(X_1), \dots, \phi_K(X_1))$ and let $\Sigma \in \mathbb{R}^{K \times K}$ denote the covariance matrix of $\phi(X_1)$. Lastly, let $\tau^2 = \text{var}(h(X_1, X_2))$, and let Z_1, \dots, Z_K denote independent standard normal random variables. Then,

$$\sup_{t \in \mathbb{R}} \left| \mathbf{P} \left(\sum_{1 \leq i \neq j \leq n} \frac{h(X_i, X_j)}{\sqrt{\tau^2 n(n-1)}} \leq t \right) - \mathbf{P} \left(\sum_{k=1}^K \frac{1}{\tau} \lambda_k(\Sigma) (Z_k^2 - 1) \leq t \right) \right| \lesssim n^{-\frac{1}{5}} + n^{-\frac{1}{14}} \left(\frac{1}{\tau} \|h(X_1, X_2)\|_{L^3} \right)^{\frac{3}{7}}.$$