Symmetric rank one updating from partial spectrum with an application to out-of-sample extension

Roy Mitz* 1, Nir Sharon† 2, and Yoel Shkolnisky‡ 1

1School of Mathematical sciences, Tel-Aviv University, Tel-Aviv, Israel
2The Program in Applied and Computational Mathematics, Princeton University, Princeton NJ, USA

Abstract

Rank-one updating of the spectrum of a matrix is fundamental in classical perturbation theory. In this paper, we suggest an updating method for a case where only partial spectrum is available. This scenario is ubiquitous in many modern applications that involve big matrices. One particular application is the extension of the top eigenvectors of the graph Laplacian to a new data sample. Each extension is then modeled as a perturbation problem and is solved by updating the leading eigenpairs. We provide a theoretical analysis of the error for such approximation approach, and back it up with empirical results that illustrate the advantages of our method.

Key Words. Out-of-sample extension, perturbation theory, one rank update, secular equation, graph Laplacian, partial spectrum.

1 Introduction

The last few decades have witnessed the emergence of various algorithms that require the calculation of the eigendecomposition of some matrix. A few known examples are Google’s PageRank [12], PCA [15], Laplacian eigenmaps [1], LLE [14], MDS [5] and many more. Since datasets nowadays may contain tens of millions of data points, the calculation of a matrix eigendecomposition efficiently becomes fundamental.

In many scenarios, only part of the eigendecomposition, i.e., only the leading eigenvalues and eigenvectors, can or need to be calculated. While algorithms for eigendecomposition, such as the Lanczos Algorithm and some variants of SVD, are designed especially for this task, they still require a hefty amount of calculation. A natural question that arises in such cases is how to update the eigendecomposition of a matrix given its partial eigendecomposition and some “small” perturbation to it, without doing the entire heavy calculation again.

In this paper, we focus on rank-one update (i.e., updating under a rank-one perturbation matrix) of a symmetric matrix. The classical approach for this kind of an update is by updating first the eigenvalues using the roots of the secular equation, see e.g., [6]. However, several other approaches for updating
the eigenvalues and eigenvectors after a perturbation have been suggested. The popular ones are quite
general and include recalculating from scratch [9], restarting the power method (which actually has
similar convergence rate as calculated from scratch [9]) and perturbation methods [16]. Some methods
that utilize the structure of a specific problem were suggested, with Google’s page rank being the most
popular application [10, Chapter 10]. Another important method is defined using a geometric embedding
of the available data, see [4]. This approach becomes computationally attractive when one updates a
low-rank matrix.

Many of the methods mentioned above might be very hard to adapt or alternatively provide very poor
estimation for cases where we do not have access for the complete eigendecomposition. Some assume the
updated matrix is low-rank, which is not always the right model for real-world data. At last, almost all
approaches do not provide an error analysis. In our method, we provide a rank-update algorithm that
does not require the full eigendecomposition of the matrix and does not assume that it is low rank. We
also introduce a full analysis to the error that is induced by the partial knowledge. In particular, our
analysis shows that the error of our algorithm is independent of the number of unknown eigenpairs, which
is confirmed by both synthetic and real-world data examples.

The paper is organized as follows. In Section 2, we develop the algorithm for the symmetric rank one
update based on partial eigendecomposition and analyze its error. Section 3 describes the application of
the algorithm for the extension of the graph Laplacian. In Section 4, we illustrate numerically some of
our theoretical results in sections 2 and 3 for both synthetic and real data.

2 Rank one updating with limited spectral access

Rank one update of the spectrum of a linear operator is a classical task in perturbation theory, e.g.,
[2, Chapter 7]. Given the original eigenvalues and eigenvectors, the rank one perturbation defines the
new spectrum by the secular equation. However, for a case of having access only to a few leading
eigenvalues and their associated eigenvectors, the classical approach requires further adaptation. Inspired
by [3], we propose a solution for the “partially knowledge” rank-one update problem, where we aim to
estimate the leading eigenpairs of a matrix after rank-one perturbation, having only partially known
eigendecomposition of the original matrix. We describe in detail the construction of our methods and
provide error bounds and complexity analysis.

2.1 Notation, classical setting, and problem formulation

Let $A_{n \times n}$ be a symmetric matrix with the real (not necessary distinct) eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ and
their associated, orthogonal eigenvectors $q_1, \ldots, q_n$. We denote this eigendecomposition by $A = Q\Lambda Q^T$,
with $Q = [q_1 q_2 \cdots q_n]$ and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$. We focus on the problem of (symmetric) rank-one
updating where we wish to find the spectral decomposition of

$$A + \rho vv^T, \quad \rho \in \mathbb{R}, \quad v \in \mathbb{R}^n, \quad \|v\| = 1.$$ 

We denote the updated eigenvalues by $t_1 \geq t_2 \geq \cdots \geq t_n$ and their associated, orthogonal eigenvectors by
$p_1, \ldots, p_n$, to form the decomposition $A + \rho vv^T = P\Sigma P^T$. The relation between the two decompositions is
well-studied, e.g., [6, 8]. One important result is the eigenvalue interlacing property, which for completion we provide in Appendix A.1. The naive upper bound for the updated eigenvalues is denoted by $\lambda_0 = \lambda_1 + \rho$.

In addition, the approximated objects (whether scalars or vectors) constructed in this section are simply denoted by an over tilde, for example the approximation for $x$ is denoted by $\tilde{x}$. Finally, $X = [x_1 \cdots x_n]$ is a matrix expressed by its column vectors, and $X^{(m)} = [x_1 \cdots x_m]$ is its truncated version consists of only $m$ first columns, $m < n$.

Classical perturbation theory assumes we are given the eigendecomposition $A = Q \Lambda Q^T$. Then, the updated eigenvalues of $A + \rho vv^T$ can be calculated by finding the $n$ roots of the secular equation (SE),

$$w(t) = 1 + \rho \sum_{i=1}^{n} \frac{z_i^2}{\lambda_i - t}, \quad z = Q^T v.$$  \hspace{1cm} (1)

Then, the corresponding eigenvector for the $k$-th root (eigenvalue) $t_k$ is given by the explicit formula

$$p_k = \frac{Q \Delta_k^{-1} z}{\|Q \Delta_k^{-1} z\|}, \quad z = Q^T v, \quad \Delta_k = \Lambda - t_k I.$$  \hspace{1cm} (2)

One important assumption in the above is the knowledge of the full eigendecomposition of the matrix $A$. This is not always feasible in modern problems due to high computational and storage costs. Therefore, a natural question is what can one do in such cases, where only part of the eigenspectrum is known beforehand. We formulate such a scenario in the following problem statement.

**Problem 1** (symmetric rank-one perturbation with partial spectrum knowledge). Let $A_{n \times n}$ be a symmetric matrix and let $1 \leq m < n$. Assume having only the first $m$ leading eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$ and their associated eigenvectors $q_1, \ldots, q_m$. Find an estimation to the first $m$ leading eigenpairs of $A + \rho vv^T$ with $\rho \in \mathbb{R}$ and $\|v\| = 1$.

We start by considering the first part of the problem — the eigenvalues.

### 2.2 Truncating the secular equation

The classical perturbation method solves the eigenvalues by finding the roots of the secular equation (1). We introduce two variations of the secular equation, adapted to the new setting of Problem 1.

Using the notation of the classical solution (2), we have from orthogonality of $Q$ that

$$\|z\|^2 = \|Q^T v\|^2 = \|v\|^2 = 1.$$

Therefore, $\sum_{j=m+1}^{n} z_j^2 = 1 - \sum_{i=1}^{m} z_i^2$. Since the last $n - m$ eigenvectors of $A$ are unknown, we denote by $\mu \leq \lambda_{m+1}$ a fixed scalar, ideally represents an estimation of the magnitude of $\lambda_j$, $j = m + 1, \ldots, n$, to form a first first order truncated secular equation,

$$w_1(t; \mu) = 1 + \rho \sum_{i=1}^{m} \frac{z_i^2}{\lambda_i - t} + \rho \frac{1 - \sum_{i=1}^{m} z_i^2}{\mu - t}.$$  \hspace{1cm} (3)

Here, $z = (Q^{(m)})^T v$ is a vector of length $m$ (first $m$ entries of $Q^T v$) with $Q^{(m)}$ consists of the $m$ columns.
of orthogonal eigenvectors associated with the leading, known eigenvalues. We will discuss the choosing of $\mu$ further on.

As a first observation, we bound the error obtained from the new formula. Namely, we show that the first $m$ largest roots of the truncated secular equation (3), which are our estimation to the eigenvalues $t_1, \ldots, t_m$, are at the order of magnitude of $\max_{m+1 \leq j \leq n} |\lambda_j - \mu|$ away from the roots of the original secular equation.

**Proposition 2.1.** Using the notation of Section 2.1, let $\rho > 0$. Then, there exist $m$ roots $\tilde{t}_1, \ldots, \tilde{t}_m$ of $w_1(t; \mu)$ of (3), correspond to the first $m$ roots $t_1, \ldots, t_m$ of the SE of (1), such that

$$|t_k - \tilde{t}_k| \leq C_k \max_{m+1 \leq j \leq n} |\lambda_j - \mu|, \quad k = 1, \ldots, m. \quad (4)$$

Here, $C_k$ is a constant bounded by $(\lambda_{m+1} - \lambda_m)^2 \max\{(\lambda_k - \lambda_1)^2, (\lambda_{k-1} - \lambda_n)^2\}^{-1}$.

**Proof.** We start by showing the existence of the roots. Indeed, since $\lim_{t \to \lambda_i^+} w_1(t; \mu) = \pm\infty$ for $i = 1, \ldots, m$, there exists a root in any segment $[\lambda_j, \lambda_{j-1}]$, $2 \leq j \leq m$. Also, since $w_1(\lambda_1 + \rho; \mu) \geq 0$ it means a root in $[\lambda_1, \lambda_1 + \rho]$. Note that together with an additional root in $(-\infty, \mu]$, these are the exactly the $m + 1$ roots of $w_1$ as seen by the $m + 1$ degree polynomial $w_1(t; \mu)(\mu - t) \prod_{i=1}^{m} (\lambda_i - t)$.

For the bound, since $t > \lambda_i$, $i = m + 1, \ldots, n$,

$$\frac{z_i^2}{\lambda_i - t} = \frac{z_i^2}{\mu - t} + \frac{z_i^2(\mu - \lambda_i)}{(\mu - t)(\lambda_i - t)}. \quad (5)$$

Therefore, split the sum in the SE (1) and using (5) we have

$$w(t) = 1 + \rho \sum_{i=1}^{m} \frac{z_i^2}{\lambda_i - t} + \rho \sum_{i=m+1}^{n} \frac{z_i^2}{\mu - t} + \rho \sum_{i=m+1}^{n} \frac{z_i^2(\mu - \lambda_i)}{(\mu - t)(\lambda_i - t)}.$$

According to Taylor, there exists $\xi$ in the segment $\lambda_k, \lambda_{k-1}$ so

$$0 = w(\tilde{t}_k - (\tilde{t}_k - t_k)) = w(\tilde{t}_k) - (\tilde{t}_k - t_k) \frac{d}{dt} w(\xi) = w_1(\tilde{t}_k; \mu) + e(\tilde{t}_k; \mu) - (\tilde{t}_k - t_k) \frac{d}{dt} w(\xi).$$

By definition, $w_1(\tilde{t}_k; \mu) = 0$. In addition, the derivative of the secular equation does not have a real root, meaning that

$$e(\tilde{t}_k; \mu) = \frac{d}{dt} w(\xi). \quad (6)$$

For the error term $e(t; \mu)$ we have

$$|e(t; \mu)| \leq \frac{|\rho|}{|\mu - t| |\lambda_{m+1} - t|} \sum_{i=m+1}^{n} z_i^2 |\lambda_i - \mu| \leq \frac{|\rho|}{|\mu - t| |\lambda_{m+1} - t|} \max_{m+1 \leq i \leq n} |\lambda_i - \mu|, \quad (7)$$

where the last inequality is since $\|z\|_2 = 1$. In addition, $\tilde{t}_i \geq \lambda_m$ and $\lambda_{m+1} \geq \mu$ so the denominator is
bounded from below by \((\lambda_m - \lambda_{m+1})^2\). Back to (6), the derivative of the secular equation is

\[
\frac{d}{dt} w(t) = \rho \sum_{i=1}^{n} \frac{z_i^2}{(\lambda_i - t)^2},
\]

and thus

\[
\left| \frac{d}{dt} w(t) \right| \geq |\rho| \sum_{i=1}^{n} \frac{z_i^2}{(\lambda_i - t)^2} \geq |\rho| \min_{1 \leq i \leq n} \left\{ \frac{1}{(\lambda_i - t)^2} \right\} \sum_{i=1}^{n} z_i^2 = |\rho| \max_{1 \leq i \leq n} (\lambda_i - t)^2.
\]

Therefore,

\[
\left| \frac{d}{dt} w(t) \right| \leq \frac{1}{|\rho| \max\{ (\lambda_k - \lambda_1)^2, (\lambda_k - 1 - \lambda_n)^2 \}}, \quad t \in [\lambda_k, \lambda_{k-1}].
\]

\[\square\]

**Remark 2.1.** Proposition 2.1 describes the case of \(\rho > 0\). The case of \(\rho < 0\) is analogous with one exception – the last root \(\tilde{t}_k\) is merely guaranteed to lie in the segment \([\mu, \lambda_m]\). Consequently, the constant \(C_m\) cannot be bounded with the same arguments. The same is also applied for the next result of this section.

We now address the problem of choosing \(\mu\). A common assumption in many real world applications is that the matrix \(A\) is low-rank, thus the unknown eigenvalues tend to zero resulting in possible choosing \(\mu = 0\). This is indeed the case for several important kernel matrices, as we will see in the next section. Generally, however, the error term in such case would be \(O(|\lambda_{m+1}|)\) and we have no reason to believe that this will result in a good approximation.

A more intelligent method for choosing \(\mu\) would be to minimize the sum in (7). However, an analytic minimizer is not attainable in this case due to the fact that both \(\lambda_i\)’s and \(z_i^2\)’s are unknown. Nevertheless, assume the trace of \(A\) is available, one choice for \(\mu\) that works well in practice is the mean of the unknown eigenvalues, which is accessible since

\[
\mu_{\text{mean}} = \frac{\sum_{i=m+1}^{n} \lambda_i}{n - m} = \frac{\text{tr}(A) - \sum_{i=1}^{m} \lambda_i}{n - m}.
\]

(8)

Shortly however, we will devise a better approximation of which an analytic minimizer can be calculated accurately.

Following the proof of Proposition 2.1, an improved accuracy can be potentially obtained by using higher order approximation for (5). Namely,

\[
\frac{z_i^2}{\lambda_i - t} = \frac{z_i^2}{\mu - t} - \frac{z_i^2(\lambda_i - \mu)}{(\mu - t)^2} + \frac{z_i^2(\lambda_i - \mu)^2}{(\mu - t)^2(\lambda_i - t)}.
\]

(9)

We evaluate the first order term by noticing that \(Aq_i = \lambda_i q_i\) and thus,

\[
\sum_{i=m+1}^{n} z_i^2 \lambda_i = \sum_{i=m+1}^{n} (q_i^T v)(q_i^T v) \lambda_i = \sum_{i=m+1}^{n} (v^T \lambda_i q_i)(q_i^T v) = \sum_{i=m+1}^{n} (v^T Aq_i)(q_i^T v).
\]

(10)
Finally,
\[ \sum_{i=m+1}^{n} z_i^2 \lambda_i = v^T A \left( I - Q^{(m)}(Q^{(m)})^T \right) v = s, \] (11)

which is a known quantity that (as marked above) is denoted by \( s \). This analysis gives rise to the second order approximation of the secular equation,
\[ w_2(t; \mu) = 1 + \rho \left( \sum_{i=1}^{m} \frac{z_i^2}{\lambda_i - t} + \frac{1 - \sum_{i=1}^{m} z_i^2}{\mu - t} - \frac{s - \mu (1 - \sum_{i=1}^{m} z_i^2)}{(\mu - t)^2} \right). \] (12)

In this case, the roots of \( w_2(t; \mu) \) of (12) are at magnitude of \( \max_{m+1 \leq i \leq n} (\lambda_i - \mu)^2 \) away from the roots of the original secular equation (1). This is concluded in the next result, which is analogous to Proposition 2.1.

**Proposition 2.2.** Using the notation of Section 2.1, let \( \rho > 0 \). Then, there are \( m \) roots \( \tilde{t}_1, \ldots, \tilde{t}_m \) of \( w_2(t; \mu) \) (12), correspond to the first \( m \) roots \( t_1, \ldots, t_m \) of (1), such that
\[ |t_k - \tilde{t}_k| \leq C_k \max_{m+1 \leq j \leq n} (\lambda_j - \mu)^2, \quad k = 1, \ldots, m. \] (13)

Here, \( C_k \) is a constant bounded by \( (\lambda_{m+1} - \lambda_m)^3 \max\{(\lambda_k - \lambda_1)^2, (\lambda_{k-1} - \lambda_n)^2\})^{-1} \), with \( \lambda_0 = \lambda_1 + \rho \).

**Proof.** This proof is done similarly to the proof of Proposition 2.1. Here, we have \( w(t) = w_2(t; \mu) + e(t; \mu) \) with
\[ e(t; \mu) = \rho \sum_{i=m+1}^{n} \frac{z_i^2(\lambda_i - \mu)^2}{(\mu - t)^2(\lambda_i - t)}. \] (14)

Then, the bound for (6) uses an additional factor of \( \lambda_m - \lambda_{m+1} \) to the constant \( C \) (due to an additional \( \mu - t \) in denominator).

To conclude the above discussion on the two approximation of the secular equation, we present Figure 1. In this figure, we set a matrix of size \( n = 4 \) with eigenvalues at 0.1, 0.2, 0.3, 0.4 and randomly rotate the eigenvectors. To form the truncated equations, we use \( m = 2 \), and \( \mu = \mu_{\text{mean}} \) of (8) which is in this case \( \mu = 0.15 \). The figure depicts the two truncated secular equations \( w_1 \) of (8) and \( w_2 \) of (12), for a rank one update with \( \rho > 0 \), along side with the original SE of (1). The analytic properties of the three functions, including their asymptotic behavior around the roots, are clearly seen. We zoom in on a neighborhood of the second root of the secular equation \( t_2 \), to observe how the second order approximation has a closer root than the root of the first order approximation, as theory suggests.

We address once again the choosing of \( \mu \). Under the low rank assumption \( (\mu = 0) \), where \( \lambda_j, j = m+1, \ldots, n \) are assumed to be small, we get according to Proposition 2.2 an improved error of \( O(\lambda_{m+1}^2) \). Nevertheless, in this case of the second order approximation to the secular equation (12), an analytic optimizer for the error (14) is attainable. Specifically, we would like to minimize \( \sum_{i=m+1}^{n} z_i^2(\lambda_i - \mu)^2 \). By standard methods we have the minimizer
\[ \mu_* = \frac{\sum_{i=m+1}^{n} z_i^2 \lambda_i}{\sum_{i=m+1}^{n} z_i^2} = \frac{s}{1 - \sum_{i=1}^{m} z_i^2}. \] (15)
Figure 1: The secular equation (1) and its two approximations: the first order $w_1$ of (3) and the second order $w_2$ of (12). The original matrix has four eigenvalues at 0.1, 0.2, 0.3 and 0.4 and a rank one update with $\rho > 0$. The approximations use $m = 2$, and $\mu = \mu_{\text{mean}} = 0.15$. In the lower right corner, we zoom-in to a small neighborhood of the second root.

which is essentially a weighted mean of the unknown eigenvalues. Unlike $\mu_{\text{mean}}$, this variant does not require the knowledge of $\text{tr}(A)$ but a few matrix-vector evaluations to calculate $s$ of (11). Interestingly enough, note that when using $\mu = \mu_*$ we have

$$w_2(t; \mu_*) = w_1(t; \mu_*),$$

meaning that we have a second order approximation in both formulas.

Next we address the corresponding problem of eigenvectors estimation.

2.3 Truncated formulas for the eigenvectors

In this section we introduce two truncated variations of the eigenvector formula (2). These are analogous to the ones of the secular equation from the previous section. The two approximations are designed to use only the $m$ leading eigenvalues and their eigenvectors and differ in accuracy and time complexity.

A naive way to truncate the eigenvectors formula (2) is by calculating

$$\tilde{p}_i = Q^{(m)} (\Delta_i^{(m)})^{-1} (Q^{(m)})^T v, \quad i = 1, \ldots, m,$$

followed by a normalization. Here $\Delta_i^{(m)} = \text{diag} (\lambda_1 - t_i, \ldots, \lambda_m - t_i)$, where $t_i$ are the roots of the secular equation (the updated eigenvalues) in descending order. We now ignore for a while the normalization and
focus on the unnormalized vectors as,

\[ p_i = Q(\Lambda - t_i I)^{-1}Q^T v \]

\[ = \begin{bmatrix} q_1 & \ldots & q_n \end{bmatrix} \begin{bmatrix} \langle q_i, v \rangle \\ \lambda_i - t_i \\ \vdots \\ \langle q_n, v \rangle \\ \lambda_n - t_n \end{bmatrix} = \sum_{k=1}^{n} \frac{\langle q_k, v \rangle}{\lambda_k - t_i} q_k. \]  

(17)

In other words, the updated eigenvectors can be written as the weighted sum of projections on the eigenspaces

\[ p_i = \sum_{k=1}^{m} \frac{\langle q_k, v \rangle}{\lambda_k - t_i} q_k + \sum_{k=m+1}^{n} \frac{\langle q_k, v \rangle}{\lambda_k - t_i} q_k. \]  

(18)

Note that the sum of unknown projections, without weights, is accessible as

\[ \sum_{k=m+1}^{n} \langle q_k, v \rangle q_k = \sum_{k=1}^{n} \langle q_k, v \rangle q_k - \sum_{k=1}^{m} \langle q_k, v \rangle q_k = v - \sum_{k=1}^{m} q_k q_k^T v = v - Q^{(m)}(Q^{(m)})^T v \]  

(19)

Again we denote by \( \mu \leq \lambda_{m+1} \) a fixed parameter in the magnitude of the unknown eigenvalues. Having the \( m \) leading eigenvectors in \( Q^{(m)} \), and recall that \( \Delta_i^{(m)} = \text{diag} (\lambda_1 - t_i, \ldots, \lambda_n - t_i) \), we define the first order truncated eigenvectors formula for any \( 1 \leq i \leq m \) as

\[ \tilde{p}_i = Q^{(m)}(\Delta_i^{(m)})^{-1}(Q^{(m)})^T v + \frac{1}{\mu - t_i} r, \quad r = v - Q^{(m)}(Q^{(m)})^T v. \]  

(20)

The second order truncated eigenvectors formula, which is the eigenvectors analogue for (12), is given by

\[ \tilde{p}_i = Q^{(m)}(\Delta_i^{(m)})^{-1}(Q^{(m)})^T v + \left( \frac{1}{\mu - t_i} + \frac{\mu}{(\mu - t_i)^2} \right) r - \frac{1}{(\mu - t_i)^2} Ar, \quad r = v - Q^{(m)}(Q^{(m)})^T v. \]  

(21)

Note that \( r \) and \( Ar \) are constant vectors and can be computed only once for all \( 1 \leq i \leq m \).

For the simplicity of analysis, at this point, we assume that the updated eigenvalues are given accurately. The error bounds of both formulas (20) and (21) are summarized in the following theorem.

**Theorem 2.3.** Let \( A_{n \times n} = Q\Lambda Q^T \) be a symmetric matrix with \( m \) known leading eigenvectors \( \lambda_1 \geq \cdots \geq \lambda_m \) and known corresponding eigenvectors \( q_1, \ldots, q_m \). The leading \( m \) eigenvectors \( p_1, \ldots, p_m \) of the rank one update \( A + \rho vv^T \), with a unit \( v \in \mathbb{R}^n \) and \( \rho \in \mathbb{R} \) can be approximated by (20) or (21), given their associated leading eigenvalues \( t_1, \ldots, t_m \) and a fixed \( \mu < \lambda_m \) with the following bounds,

1. For \( \tilde{p}_i \) of (20),

\[ \|p_i - \tilde{p}_i\| \leq C \max_{m+1 \leq i \leq n} |\lambda_i - \mu|, \quad C \leq |\lambda_{m+1} - t_i|^{-2}. \]

2. For \( \tilde{p}_i \) of (21),

\[ \|p_i - \tilde{p}_i\| \leq C \max_{m+1 \leq i \leq n} |\lambda_i - \mu|^2, \quad C \leq |\lambda_{m+1} - t_i|^{-3}. \]

**Proof.** By (18), we have

\[ e_i = \|p_i - \tilde{p}_i\| = \left\| \sum_{k=m+1}^{n} \frac{\langle q_k, v \rangle}{\lambda_k - t_i} q_k + \frac{1}{\mu - t_i} r \right\|. \]
Similar to (5), \( \frac{1}{\lambda_k - t_i} = \frac{1}{\mu - t_i} + \frac{(\mu - \lambda_k)}{(\mu - t_i)(\lambda_k - t_i)} \) and using (19) and the orthogonality of \( q_i \), we have

\[
e_i^2 = \left\| \sum_{k=m+1}^{n} \frac{(\mu - \lambda_k)\langle q_k, v \rangle}{(\mu - t_i)(\lambda_k - t_i)} q_k \right\|^2 = \sum_{k=m+1}^{n} \left( \frac{(\mu - \lambda_k)\langle q_k, v \rangle}{(\mu - t_i)(\lambda_k - t_i)} \right)^2 \leq \frac{1}{(\lambda_{m+1} - t_i)^4} \sum_{k=m+1}^{n} (\mu - \lambda_k)^2 \langle q_k, v \rangle^2 = \frac{1}{(\lambda_{m+1} - t_i)^4} \sum_{k=m+1}^{n} (\mu - \lambda_k)^2 z_k^2.
\]

Since \( \|Q^T v\| = 1 \), taking the square root gives us the bound.

For the second claim we use the longer expansion, as done in (9), and recall that \( Aq_k = \lambda_k q_k \) which in this case means

\[
\frac{\mu}{(\mu - t_i)^2} R - \frac{1}{(\mu - t_i)^2} Ar = \frac{1}{(\mu - t_i)^2} \sum_{k=m+1}^{n} \langle q_k, v \rangle (\mu - \lambda_k) q_k.
\]

Thus, we have for the corrected formula (21) with exact eigenvalues,

\[
\|p_i - \tilde{p}_i\|^2 = \left\| \sum_{k=m+1}^{n} \frac{(\mu - \lambda_k)^2 \langle q_k, v \rangle}{(\mu - t_i)^2(\lambda_k - t_i)} q_k \right\|^2 = \sum_{k=m+1}^{n} \left( \frac{(\mu - \lambda_k)^2 \langle q_k, v \rangle}{(\mu - t_i)^2(\lambda_k - t_i)} \right)^2,
\]

and the second claim follows as before.

As with the eigenvalues, under the low rank assumption \( \mu = 0 \) Theorem 2.3 guarantees errors of \( O(|\lambda_{m+1}|) \) and \( O(\lambda_{m+1}^2) \) for (20) and (21), respectively. The choice of \( \mu \) to minimize the sum in (22) is \( \mu_* \) of (15), and is thus expected to provide better estimation than \( \mu = 0 \). Experimental results (see Section 4) have showed that the choice \( \mu = \mu_{\text{mean}} \) (8) is competitive with \( \mu_* \) while being slightly faster to compute.

### 2.4 An algorithmic summary

Given a parameter \( \mu \), we have provided first and second order truncated approximations to the secular equation and corresponding eigenvectors formula. As for \( \mu \), we suggest three main options. If the matrix is low-rank, choose \( \mu = 0 \). Otherwise, \( \mu_* \) would help minimize the error term, while \( \mu_{\text{mean}} \) is easier to compute. We summarize the previous subsections with Algorithm 1, which addresses Problem 1.

A complexity analysis is provided as complementary in Appendix B.
Algorithm 1 An algorithm for the updating problem with partial spectrum

Input: m leading eigenpairs \( \{(\lambda_i, q_i)\}_{i=1}^m \) of a symmetric matrix \( A \), an update direction \( v \in \mathbb{R}^n \) with \( \|v\| = 1 \) and a scalar \( \rho > 0 \)

Output: An approximation \( \{ (\tilde{\lambda}_i, \tilde{p}_i) \}_{i=1}^m \) of the eigenpairs of \( A + \rho vv^T \)

1: Choose a parameter \( \mu \) (e.g., \( \mu = 0 \), \( 8 \) or \( 15 \)).
2: Calculate \( m \) largest roots \( \{(t_i)_{i=1}^m\} \) of a truncated secular equation (either (3), (12))
3: for all \( \{q_i\}_{i=1}^m \) do
4: find \( \tilde{p}_i \) by a truncated eigenvectors formula (either (20), (21))
5: end for

3 Updating the graph Laplacian for out-of-sample extension

In this section we introduce the application of the rank-one updating of Section 2 to the problem of out-of-sample extension of the graph Laplacian. We start by defining the problem. Then, we justify the use of rank-one modification by proving that each extension of the graph Laplacian is close to a rank-one perturbation. The section is concluded with a few algorithms, which are demonstrated numerically in the following Section 4.

3.1 Preliminaries and Problem formulation

We begin by introducing the notations and the model for the updating problem. Given a set of discrete points \( X = \{x_i\}_{i=1}^n \subset \mathbb{R}^d \), we define a weighted graph whose vertices are the given points. An edge is inserted to the graph if its two vertices are ”close”. Some common variants are,

1. k-nearest neighbors (kNN). Nodes \( i \) and \( j \) are connected iff \( i \) is within the kNN of \( j \) or vice versa.

   The neighborhood is typically chosen according to some distance function on the data.

2. \( \delta \)-neighborhood. Nodes \( i \) and \( j \) are connected iff \( \|x_i - x_j\| < \delta \) for some \( \delta > 0 \).

Each edge on the graph is assigned with a weight, usually determined by a kernel function. A kernel function \( K \) is a symmetric function \( K: \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R} \) that defines a weight on the edge between \( i \) and \( j \) as \( w_{ij} = K(x_i, x_j) \). A kernel is said to be radial if

\[
K(x, y) = g(\|x - y\|), \quad x, y \in \mathbb{R}^d, \tag{23}
\]

for a non-negative real function \( g \). One common choice of such is the heat kernel (also known as Gaussian kernel) that induces the weights

\[
w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{\varepsilon}\right), \tag{24}
\]

with a width parameter \( \varepsilon > 0 \).

Given a weight matrix \( W = \{w_{ij}\} \) and the induced (diagonal) degree matrix \( D_{ii} = \sum_{j=1}^n W_{ij} \), one could define the Graph Laplacian of the graph. Two popular variants of the graph Laplacian are the random walk graph Laplacian, \( L = D^{-1}W \), and the symmetric normalized graph Laplacian,

\[
L = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}, \tag{25}
\]
(a) The graph corresponding to $L_0$, with $x_0$ (empty dot) disconnected
(b) The graph corresponding to $L_1$, with $x_0$ (empty dot) connected

Figure 2: Illustration of connecting a node, representing a new sample point $x_0$, to the graph

see e.g., [1]. Note that $W$ and $L$ are $n \times n$ matrices where $n$ is the number of samples in $\mathcal{X}$. Naturally, we consider only the case of symmetric normalized graph Laplacian, however, similar results can be achieved for the random walk graph Laplacian as it maintains a similarity relation to the symmetric version. Henceforth, unless otherwise stated, by referring to “graph Laplacian” we mean the symmetric normalized graph Laplacian (25). We now formulate the updating problem for the graph Laplacian.

**Problem 2** (Out-of-sample extension of the graph Laplacian). Let $\mathcal{X} = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$ and let $x_0 \notin \mathcal{X}$ be a new sample point in $\mathbb{R}^d$. Denote by $L_0$ the graph Laplacian constructed from $\mathcal{X}$ with a given kernel. Assume the top $m$ eigenvalues and eigenvectors of $L_0$ are known ($m < n$). Find the top $m$ eigenpairs of $L_1$, the graph Laplacian of $\mathcal{X} \cup \{x_0\}$.

The above Problem 2 is reduced to Problem 1 by observing that the difference matrix $\Delta L = L_1 - L_0$ is very close to be rank one (will be formulated and proven next). In other words, by looking at the leading eigenvalue of $\Delta L$, $\rho = \lambda_1(\Delta L)$, and its associated eigenvector $v = q_1(\Delta L)$, we estimate the leading eigenpairs of $L_1$ using the proxy,

$$ \tilde{L}_1 = L_0 + \rho vv^T. $$

Note that since $L_0$ and $L_1$ are of a different size, we add a row and a column to $L_0$ consisting of 1 at the first index and 0 everywhere, that is, considering $x_0$ as an isolated node. An illustration of Problem 2 is given in Figure 2.

### 3.2 Updating the graph Laplacian is almost rank one perturbation

To formulate our observation about the extension of the graph Laplacian, we equip with a few settings and notation. As described in the previous Section 3.1, the weights on the edges of the graph are determined by a kernel. For our further claims, we will require that our kernel is radial with $g(0) > 0$ and that in some
neighborhood of 0 the derivative is bounded, that is \( \left| \frac{d}{dx} g \right| < M \) for some \( M > 0 \). These requirements are not too restrictive as they are met by most common kernels used, such as the heat kernel.

In our analysis, we refer to the construction of graphs according to \( \delta \)-neighborhoods. As we will see next, the analogue for kNN is straightforward. Therefore, we require the parameter \( \delta \) to be “small enough”, and more specifically, to satisfy \( \delta < \frac{g(0)}{2M} \). The justification for this is that we wish to keep the neighborhoods small in order for them to capture the local structure of the graph.

We denote by \( k \) the minimum number of neighbors of a vertex. In addition, we denote by \( c_1 \geq 1 \) a constant such that \( c_1 \cdot k \) is the maximum number of neighbors of a vertex (we assume \( c_1 \) is independent of \( k \)). We now present the main theoretical result of this section. As denoted previously, \( \lambda_j(X) \) stands for the \( j \)-th eigenvalue of \( X \).

**Theorem 3.1.** Under the requirements and notation just described, let \( L_0 \) and \( L_1 \) be two graph Laplacians as defined in Problem \([2]\). Then, there is a constant \( \beta \), independent of \( k \) and \( n \), such that

\[
\lambda_1(L_1 - L_0) = 1 - \beta \frac{k}{\delta} \quad \text{and} \quad \lambda_i(L_1 - L_0) = \beta \frac{k}{\delta}, \quad i \geq 2.
\]

Theorem [3.1] shows that for large enough \( k \), \( \lambda_1(L_1 - L_0) \approx 1 \) and \( \lambda_i(L_1 - L_0) \approx 0, \quad i \geq 2 \). In other words, \( \Delta L = L_1 - L_0 \) is indeed close to be rank one.

### 3.3 Proof of Theorem 3.1

The proof is divided into a few steps. First, we adapt an important result, also known as Weyl’s Theorem, to our setting for an initial bound on the singular values. Then, we use our assumptions to derive, based on the specific structure of the graph Laplacian, the right constants and bounds to use in the main body of the proof.

From classic perturbation theory, we have the following result. We quote this general result with respect to the singular values of a matrix, which we denote by \( \sigma_i(\cdot), \quad i = 1, \ldots, n \).

**Theorem 3.2** (Weyl’s Theorem). Let \( S, E \in \mathbb{R}^{n \times n} \) then for all \( 1 \leq i \leq n \) we have,

\[
\left| \sigma_i(S + E) - \sigma_i(S) \right| \leq \|E\|_2.
\]

As it turns out, for the special case where \( S \) is diagonal, we can further improve the above estimation.

**Theorem 3.3.** Let \( S \in \mathbb{R}^{n \times n} \) a diagonal matrix and let \( E \in \mathbb{R}^{n \times n} \). Denote \( E = (e_{ij}) \). Let \( \rho > 0 \) such that \( \|E\|_2 < \rho \). Then for small enough \( \rho \), there exists \( c_H = c_H(S, \rho) > 0 \) independent of \( E \) so that for all \( 1 \leq i \leq n, \)

\[
\left| \sigma_i(S + E) - \sigma_i(S) - e_{ii} \right| \leq c_H \|E\|_F^2.
\]

The proof of Theorem 3.3 is given in Appendix A.2.

Recall that our aim is to bound the eigenvalues of \( \Delta L = L_1 - L_0 \), that is the difference matrix between the graph Laplacians before and after the insertion of a new node. To apply Theorem 3.3, we denote \( S = \text{diag}(\Delta L) \) and \( E = \Delta L - S \). Note that in our specific case, the diagonal entries of \( E \) are in fact zero and the matrix \( E \) is symmetric positive definite so the singular values are identical to the eigenvalues.
Therefore, there exists $c' > 0$ so that

$$|\lambda_i(\Delta L) - \lambda_i(S)| \leq c'\|E\|_F^2.$$  \hspace{1cm} (26)

It is clear now that estimating $\|E\|_F$ will provide us the wanted relation between the eigenvalues of $\Delta L$ and $S$.

We start by examining $\Delta L$; the only nonzero elements are the ones affected by the introduction of the new node. There are at most $c_1k$ such rows, each consists of at most $c_1k$ nonzero elements by assumption. Thus, the total number of elements changed in these rows is at most $c_1^2k^2$. Due to symmetry, the same goes for the columns. Finally, we have at most $c_2^2k^2 + c_2^2k^2 = 2c_2^2k^2$ changed entries. In other words, using the convention that $\text{nnz}(X)$ is the number of nonzero elements of a matrix $X$, we have that

$$\text{nnz}(\Delta L) \leq (2c_2^2)k^2.$$  \hspace{1cm} (27)

An element-wise estimation of the entries of the graph Laplacian, stating they are of order $\frac{1}{k}$, is given next.

**Lemma 3.4.** Let $L = (\ell_{i,j})$ be a graph Laplacian, calculated for $\delta$-neighborhoods, using radial kernel with a bounded derivative $\left|\frac{d}{dx}g\right| < M$ and $g(0) > 0$. Then,

$$\frac{1}{c_1c} \cdot \frac{1}{k} < \ell_{ij} < \frac{c}{k}, \quad 1 \leq i, j \leq n, \quad c = 1 + \frac{g(0)}{M\delta}.$$

**Proof.** Let $\alpha_{ij} = \|x_i - x_j\|$. Then, using Lagrange remainder theorem, there exists $\xi_{ij}$ for any $ij$-th entry of the weight matrix $W$, such that

$$w_{ij} = g(\|x_i - x_j\|) = g(0) + \frac{d}{dx}g(\alpha_{ij})\xi_{ij}.$$

It follows that the $ij$-th entry of the graph Laplacian is

$$\ell_{ij} = \frac{g(0) + \frac{d}{dx}g(\xi_{ij})\alpha_{ij}}{\sqrt{\sum_p g(0) + \frac{d}{dx}g(\xi_{pj})\alpha_{pj}} \cdot \sqrt{\sum_p g(0) + \frac{d}{dx}g(\xi_{ip})\alpha_{ip}}},$$

where the two sums are taken on all the neighbors of the $i$-th and $j$-th vertices. Since $\alpha_{ij} < \delta$ and $\left|\frac{d}{dx}g\right| < M$, we have that an upper bound on the numerator is $g(0) + M\delta$. On the other hand, $g(0) > 2M\delta$ and the number of neighbors is at least $k$ so

$$\sum_p g(0) + \frac{d}{dx}g(\xi_{pj})\alpha_{pj} \geq kM\delta.$$

Therefore,

$$\ell_{ij} < \frac{g(0) + M\delta}{\sqrt{kM\delta} \cdot \sqrt{kM\delta}} = \frac{g(0) + M\delta}{M\delta k}.$$
Similarly, using the upper bound \( c_1 k \) on the number of neighbors we get
\[
\ell_{ij} > \frac{M\delta}{c_1 (g(0) + M\delta) k}.
\]

An immediate conclusion from Lemma 3.4 is that the entries of \( \Delta L \) are of order \( O(\frac{1}{k}) \) except the first entry which is \( 1 - O(\frac{1}{k}) \). Namely, \( \Delta L \) is dominated by its first entry, and so it is somewhat unsurprising that it is close to be rank one. A sharper element-wise bound is given in the following lemma.

**Lemma 3.5.** The entries of \( \Delta L \) that are not on the first row/column are smaller than \( c_2^2 k^2 \), \( c_2 = \sqrt{g(0) M\delta} \).

The proof of Lemma 3.5 is given in Appendix A.3.

The nonzero elements of \( E \) are either on the first row/columns or not. The elements that are on the first row/columns are at most \( c_1 k \) in size, and there are at most \( c_1 k \) of them. Within the elements that are not on the first row/column, based on (27), there are at most \( 2 c_1 k^2 \) nonzero elements and by Lemma 3.5, they are at most \( c_2^2 k^2 \) in size. Therefore, we can bound the Frobenius norm of \( E \),
\[
\|E\|_F^2 \leq c_1 k \left( \frac{c}{k} \right)^2 + 2 c_1 k^2 \left( \frac{c^2}{2k^2} \right)^2 = \frac{c^2 c_1}{k} + \frac{c^4 c_1}{2k^2} < \frac{c^4 c_1}{k} + \frac{c^4 c_1}{k} = \frac{2 c^4 c_1}{k}.
\]

Namely,
\[
\|E\|_F \leq \sqrt{2 c_1 c^2 \frac{1}{\sqrt{k}}}, \tag{28}
\]

We finally resume to prove our main theorem.

**Proof of Theorem 3.1.** By (26),
\[
|\lambda_i(\Delta L) - \lambda_i(S)| < c' \|E\|_F^2 < c' \left( \frac{\sqrt{2 c_1 c^2}}{\sqrt{k}} \right)^2 = \frac{2 c_1 c^4 c'}{k} = \frac{\overline{c}}{k}, \quad 1 \leq i \leq n.
\]
The first singular value of \( S \) is its largest entry, and thus by Lemma 3.4
\[
\lambda_1(\Delta L) < \lambda_1(S) + \frac{\overline{c}}{k} < 1 - \frac{2}{c_1 c k} + \frac{\overline{c}}{k} = 1 - \frac{2 + c_1 c}{c_1 c k},
\]

and
\[
\lambda_1(\Delta L) > \lambda_1(S) - \frac{\overline{c}}{k} > 1 - \frac{c}{k} - \frac{\overline{c}}{k} = 1 - \frac{c + \overline{c}}{k}.
\]

Namely, \( \lambda_1(\Delta L) \) is of order \( 1 - \frac{1}{k} \). The other singular values of \( S \) are the other diagonal entries, which are at most \( \frac{c}{k} \), by Lemma 3.4. Thus, we have
\[
\lambda_i(\Delta L) < \lambda_i(S) + \frac{\overline{c}}{k} < \frac{c}{k} + \frac{\overline{c}}{k} = \frac{\overline{c} + c}{k}, \quad i \geq 2,
\]
which shows \( \lambda_i(\Delta L) \) is of order \( \frac{1}{k} \) as required. \( \square \)
3.4 Exploiting the rank one updating

We discuss the required adjustment for applying Algorithm 1 of rank-one updating to address Problem 2 of out-of-sample updating of the graph Laplacian. Such application requires recovering largest eigenpair of \( \Delta L = L_1 - L_0 \), in order to define the rank-one update of \( L_0 \). As a result, such an algorithm introduces two forms of error: the error induced by using truncation on the perturbation equations, which was discussed in Section 2, and the error induced by the rank-one approximation, which we examine now.

To further improve our estimation, we use two classical results from matrix perturbation theory. These results, Lemma 3.6 and Lemma 3.7, are given without proof, the interested reader is referred to [7, Chapter 4]. As before, we denote by \( q_i(X) \) a normalized eigenvector that is associated with the \( i \)-th largest eigenvalue of \( X \).

**Lemma 3.6.** Let \( A, B \in \mathbb{R}^{n \times n} \) be symmetric matrices. Denote by \( B_k = \sum_{i=1}^{k} \lambda_i(B)u_iu_i^T \) the best rank-\( k \) approximation of \( B \). Then, the following holds for any \( 1 \leq i \leq k \),

1. \( \lambda_i(A + B) - \lambda_i(A + B_k) = O(\lambda_{k+1}) \).
2. \( \|q_i(A + B) - q_i(A + B_k)\| = O(\lambda_{k+1}) \).

With Lemma 3.6, let

\[
A = L_0, \quad A + B = L_1, \quad \text{and} \quad B_1 = \lambda_1(B)q_1(B)q_1(B)^T. \tag{29}
\]

Then, the rank-one update induces an error of order \( O(\lambda_2(B)) = O(\lambda_2(L_1 - L_0)) \), and by Theorem 3.1 we conclude that this error is of order

\[
\lambda_2(L_1 - L_0) = O\left(\frac{1}{k}\right).
\]

Similarly to Section 2, we can have higher order approximation with a further correction. We use the second classical result.

**Lemma 3.7.** Under the notation of Lemma 3.6, define \( C_k = B - B_k \). Then, the following holds for any \( 1 \leq i \leq k \), for any \( 1 \leq i \leq k \),

1. \( \lambda_i(A + B) - \left[ \lambda_i(A + B_k) + q_i^T(A)C_kq_i(A) \right] = O(\lambda_{k+1}^2) \).
2. \( \|q_i(A + B) - \left[ q_i(A + B_k) + \sum_{j \neq i} \frac{q_j(A)^T C_k q_j(A)}{\lambda_i - \lambda_j} q_j(A) \right]\| = O(\lambda_{k+1}^2) \).

Lemma 3.7 gives rise to an improvement due to the extra term of perturbation correction. Using (29), and by Theorem 3.1 we have now that

\[
\lambda_2(L_1 - L_0) = O\left(\frac{1}{k^2}\right).
\]

Note that for this correction to be tractable in large scale problems, the correction matrix \( C_1 \) should be sparse. This matter and other complexity questions are discussed in detail on Appendix B. The perturbation correction is embedded in our method as described fully in Algorithm 2.
Algorithm 2 Laplacian out-of-sample extension

Input: The original Graph Laplacian $L_0$ and its top $m$ eigenpairs $\{(\lambda_i, q_i)\}_{i=1}^m$.

A new sample point $x_0$.

Output: approximate top eigenpairs $\{((\hat{\lambda}_i, \hat{p}_i))\}_{i=1}^m$.

1: $L_1 \leftarrow \text{the graph Laplacian of } \mathcal{X} \cup \{x_0\}$
2: $\Delta L \leftarrow L_1 - L_0$
3: $\rho \leftarrow \lambda_1(\Delta L)$
4: $v \leftarrow q_1(\Delta L)$
5: $\{((\tilde{\lambda}_i, \tilde{p}_i))\}_{i=1}^m \leftarrow \text{Algorithm 1}\left(\{((\lambda_i, q_i))\}_{i=1}^m, \rho, v\right)$
6: $C \leftarrow L_1 - (L_0 + \rho vv^T)$
7: for all $i = 1...m$ do // Perturbation Correction
8: $\hat{\lambda}_i \leftarrow \tilde{\lambda}_i + \tilde{p}_i^T C \tilde{p}_i$
9: $\hat{p}_i \leftarrow \tilde{p}_i + \sum_{j \neq i} \tilde{p}_j^T C \tilde{p}_i \tilde{p}_j - \tilde{p}_i^T C \tilde{p}_i$
10: end for

4 Numerical examples

In this section, we provide various numerical examples to demonstrate empirically the theory that we developed in previous sections. We use both synthetic datasets and real-world datasets. We begin by providing several numerical examples for the rank-one updating formulae of Section 2. These examples demonstrate the high accuracy of the methods, as well as their runtime efficiency. We continue by providing numerical examples for Section 3 showing numerically that insertion of a new node to the graph is almost a rank one update to the graph Laplacian matrices. We proceed by applying our algorithm for updating the eigenvalues and eigenvectors of the graph Laplacian under real-world data and measure the accuracy of our approach comparing to other methods.

4.1 Truncated formulas for rank-one update (Section 2)

We start with a synthetic example to demonstrate empirically the use of the truncated secular equation and eigenvectors formula for rank one update eigenvalue problem. We generate a random symmetric matrix $A$ of $n = 1000$ with $m = 10$ known leading eigenvalues of order $O(1)$ and their corresponding eigenvectors. The rest of the eigenvalues are unknown for the algorithm and were drawn from a normal distribution with mean $\mu$ and standard deviation of $\sigma = 0.0001$. For the update, we use a random perturbation vector $v$. The goal is to recover the $m$ top eigenpairs of $A + vv^T$ for various values of $\mu$. As a rough estimator for the unknown eigenvalues, our parameters $\mu$ is chosen to be either $\mu = 0$ or $\mu = \mu^*$. According to the theory, for the above setting, we expect the case $\mu = 0$ to yield errors of magnitude $O(\mu)$ for the first order approximations, and an error of magnitude $O(\mu^2)$ for the second order approxi-
Table 1: Absolute errors for synthetic example of $n = 1000$, $m = 10$, with unknown eigenvalues distributed around various values of mean $\hat{\mu}$ and a standard deviation of $\sigma = 0.0001$

| $\hat{\mu}$ | Eigenvalues |  | Eigenvalues |  |
|--------------|-------------|---|-------------|---|
|               | first order $\mu = 0$ | second order $\mu = 0$ | first order $\mu = \mu^*$ | second order $\mu = \mu^*$ |
| 1e-00 | 8.79e-02 | 3.82e-02 | 9.22e-10 | 1.79e-01 | 1.70e-01 | 3.45e-05 | 5.25e-08 |
| 1e-01 | 4.20e-03 | 4.24e-04 | 4.42e-10 | 1.26e-02 | 7.90e-03 | 9.68e-06 | 8.27e-09 |
| 1e-02 | 3.08e-04 | 2.77e-06 | 2.72e-10 | 7.83e-04 | 9.72e-05 | 8.28e-06 | 9.61e-09 |
| 1e-03 | 3.00e-05 | 2.68e-08 | 2.61e-10 | 7.66e-05 | 1.00e-06 | 8.20e-06 | 9.88e-09 |
| 1e-04 | 3.12e-06 | 5.83e-10 | 2.95e-10 | 1.17e-05 | 2.21e-08 | 8.72e-06 | 1.12e-08 |

Figure 3: Plot of $\log_2$-absolute error as a function of $\log_2 \hat{\mu}$ for a synthetic example of $n = 1000$, $m = 10$, with unknown eigenvalues distributed around various values of mean $\hat{\mu}$ and a standard deviation of $\sigma = 0.0001$. We can notice the three main trends: the error of $\mu^*$ is independent on $\hat{\mu}$, a linear error decay for $\mu = 0$ and first order approximation, and a quadratic error decay for second order approximation.
Table 2: Time measurements in seconds

| n      | MATLAB first order $\mu = 0$ | second order $\mu = \mu_*$ | MATLAB second order $\mu = 0$ | second order $\mu = \mu_*$ |
|--------|-------------------------------|-----------------------------|-------------------------------|-----------------------------|
| 2000   | 0.47 ± 0.05                   | 0.85 ± 0.02                 | 0.45 ± 0.01                   | 0.51 ± 0.01                 |
| 4000   | 1.71 ± 0.10                   | 0.92 ± 0.02                 | 0.95 ± 0.01                   | 1.08 ± 0.01                 |
| 8000   | 5.34 ± 0.05                   | 0.88 ± 0.02                 | 1.98 ± 0.04                   | 2.21 ± 0.04                 |
| 16000  | 17.1 ± 0.12                   | 0.97 ± 0.02                 | 4.74 ± 0.20                   | 5.08 ± 0.42                 |
| 32000  | 50.6 ± 0.57                   | 1.11 ± 0.02                 | 7.14 ± 0.20                   | 28.5 ± 0.42                 |
| 64000  | 154 ± 1.01                    | 1.36 ± 0.01                 | 10.9 ± 0.20                   | 11.7 ± 0.42                 |

(a) A varying $n$ with a fixed number of eigenpairs, $m = 10$

| m      | MATLAB first order $\mu = 0$ | second order $\mu = \mu_*$ | MATLAB second order $\mu = 0$ | second order $\mu = \mu_*$ |
|--------|-------------------------------|-----------------------------|-------------------------------|-----------------------------|
| 50     | 12.4 ± 0.13                   | 0.45 ± 0.01                 | 0.45 ± 0.01                   | 0.51 ± 0.01                 |
| 100    | 26.5 ± 0.34                   | 0.95 ± 0.01                 | 0.95 ± 0.01                   | 1.08 ± 0.01                 |
| 200    | 60.5 ± 0.33                   | 1.98 ± 0.04                 | 1.98 ± 0.04                   | 2.21 ± 0.04                 |
| 400    | 155 ± 4.05                    | 4.74 ± 0.20                 | 4.74 ± 0.20                   | 5.08 ± 0.42                 |
| 600    | 345 ± 4.05                    | 7.14 ± 0.20                 | 7.14 ± 0.20                   | 28.5 ± 0.42                 |
| 800    | 542 ± 4.05                    | 10.9 ± 0.20                 | 10.9 ± 0.20                   | 11.7 ± 0.42                 |

(b) A varying number of eigenpairs $m$ with a fixed matrix size $n = 20,000$

Figure 4: Plot of log-runtime as a function of log($n$), for matrices of size $n$. We can see that the runtime difference between our approximation and MATLAB’s calculation increases considerably with larger $n$ values, as witnessed in Figure 4. Additionally, the runtime difference between our variants is negligible.

4.2 Updating the graph Laplacian (Section 3)

We provide several examples from three real-world datasets to demonstrate different aspects of the problem of updating the symmetric graph Laplacian, that is Problem 2. The datasets description is given in Table 3.

In the first illustration, we aim to demonstrate that inserting a new node to the graph Laplacian is almost rank-one, as suggested by Theorem 3.1.

In this example, for any given dataset, we randomly select a subset of it. We calculate the symmetric graph Laplacian $L_0$ of the selected subset leaving the first node out. Then, we connect this node to the graph, resulting in a new graph Laplacian $L_1$, for which we compute the first and second eigenvalues of $\Delta L = L_1 - L_0$. The results consist of the mean of 10 independent experiments, each one with a different random subset of the data. These results are shown in Table 4 for various datasets and values of $k$. Clearly, one can observe, as expected by the theory, that the first eigenvalue is very close to 1, while the second singular value is close to 0.

Next, we demonstrate empirically the theoretical dependency of the eigenvalues on $k$. Specifically,
| Name          | Samples | Attributes | Description                                           |
|--------------|---------|------------|-------------------------------------------------------|
| MNIST        | 60,000  | 784        | grey scale images of a handwritten digit between 0 and 9 |
| Poker Hand   | 25,000  | 10         | Each record is an example of a hand consisting of five playing cards drawn from a standard deck of 52 |
| Yeast        | 1484    | 8          | Information about a set of Yeast cells               |

Table 3: Real-world datasets in used

| dataset             | $k = 100$ | $k = 200$ | $k = 300$ |
|---------------------|-----------|-----------|-----------|
|                      | $\lambda_1$ | $\lambda_2$ | $\lambda_1$ | $\lambda_2$ | $\lambda_1$ | $\lambda_2$ |
| MNIST (5K samples)  | 0.973     | 0.004     | 0.978     | 0.003     | 0.979     | 0.002     |
| poker (10K samples) | 0.946     | 0.006     | 0.959     | 0.003     | 0.963     | 0.002     |
| yeast (1.5K samples)| 0.996     | 0.004     | 0.997     | 0.003     | 0.998     | 0.002     |

Table 4: Two largest eigenvalues of $\Delta L$ for real-world datasets and three different values of $k$. As theory suggests, there is a two orders of magnitude difference between the first and second eigenvalues, indicating that indeed $\Delta L$ is close to be rank one.

Theorem 3.1 indicates that up to a constant

$$\log(\lambda_1(\Delta L) - 1) = \log(\lambda_i(\Delta L)) = -k, \quad 2 \leq i \leq n.$$ 

Thus the log-dependency of the eigenvalues and $k$ is expected to be linear with slope that equals 1. This is demonstrated for the MNIST dataset in Figure 5. Similar results occur for the other datasets as well.

As our main comparison example, we use as a benchmark the Nyström method \[3\], which is a widely used method for eigenvectors extensions. In addition, we use the naive approach of merely having the old eigenvalues and eigenvectors as approximations to the new ones, and also for the case of $m = n$, the standard untruncated eigenvalues and eigenvectors formulae, \[1\] and \[2\], respectively. In this manner, we would like to understand how much we could potentially gain if we have the entire spectrum available to estimate the top new $m$ eigenpairs. In other words, it shows how good is the truncation formulae in this scenario. We then apply the perturbation correction algorithm for each of the methods. The results are the mean absolute error of the $m$ eigenpairs in 10 independent experiments. The eigenvalues calculated were of order $O(0.1)$, the eigenvectors error is measured in norm. The full comparison is given in Table 5 where for each dataset we also mention the parameter $\varepsilon$ of the width of the Gaussian (see \[24\]) that we used for constructing the graph Laplacian.

References

[1] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.
| Dataset   | Eigenvalues   | Eigenvalues (corrected) | Eigenvectors   | Eigenvectors (corrected) |
|-----------|---------------|-------------------------|----------------|-------------------------|
| MNIST     | 1.05e-05      | 1.45e-05                | 5.41e-02       | 2.94e-02                |
|           | -             | -                       | 5.01e-02       | 2.30e-02                |
|           | 8.96e-06      | 4.88e-07                | 2.40e-02       | 3.00e-03                |
|           | First order ($\mu = 0$) | 8.94e-06       | 7.02e-07       | 2.46e-02                | 4.00e-03                |
|           | First order ($\mu = \mu_*$) | 8.94e-06       | 6.31e-07       | 2.45e-02                | 3.80e-03                |
|           | Second order ($\mu = 0$) | 8.93e-06       | 6.30e-07       | 2.45e-02                | 3.80e-03                |
|           | Second order ($\mu = \mu_*$) | 8.94e-06       | 6.25e-07       | 2.45e-02                | 3.70e-03                |

| Poker     | 1.13e-04      | 1.57e-04                | 8.07e-02       | 3.79e-02                |
|           | -             | -                       | 7.82e-02       | 3.26e-02                |
|           | 8.35e-05      | 3.97e-06                | 3.76e-02       | 3.30e-03                |
|           | First order ($\mu = 0$) | 7.14e-05       | 6.45e-05       | 1.02e-01                | 2.16e-02                |
|           | First order ($\mu = \mu_*$) | 8.51e-05       | 4.55e-06       | 3.83e-02                | 4.30e-03                |
|           | Second order ($\mu = 0$) | 8.51e-05       | 4.75e-06       | 3.83e-02                | 4.20e-03                |
|           | Second order ($\mu = \mu_*$) | 8.51e-05       | 4.67e-06       | 3.82e-02                | 4.10e-03                |

| Yeast     | 2.03e-04      | 4.52e-04                | 1.60e-01       | 7.78e-02                |
|           | -             | -                       | 1.60e-01       | 7.16e-02                |
|           | 1.69e-04      | 1.08e-05                | 5.02e-02       | 5.30e-03                |
|           | First order ($\mu = 0$) | 1.68e-04       | 2.42e-05       | 5.49e-02                | 9.30e-03                |
|           | First order ($\mu = \mu_*$) | 1.68e-04       | 2.13e-05       | 5.41e-02                | 8.70e-03                |
|           | Second order ($\mu = 0$) | 1.68e-04       | 2.19e-05       | 5.40e-02                | 8.50e-03                |
|           | Second order ($\mu = \mu_*$) | 1.68e-04       | 2.16e-05       | 5.38e-02                | 8.30e-03                |

Table 5: Absolute error comparison for three real datasets: MNIST ($n = 3000, m = 50, k = 50, \varepsilon = 100$), poker ($n = 3000, m = 50, k = 100, \varepsilon = 100$) and yeast ($n = 1400, m = 100, k = 50, \varepsilon = 100$). We can see that our methods present superior performance.
(a) $1 - \lambda_1 (\Delta L) = O\left(\frac{1}{k}\right)$. The log dependency of $1 - \lambda_1$ with $k$ is linear with slope $(-1)$, as evidenced by the linear regression plot.

(b) $\lambda_i (\Delta L) = O\left(\frac{1}{k}\right)$ for $i = 2, 3, 4$. The log dependency of $\lambda_i$ with $k$ is linear with slope $(-1)$, as evidenced by the linear regression plot.

Figure 5: Demonstration of Theorem 3.1 for the poker dataset with $n = 10,000$

[2] Carl M Bender and Steven A Orszag. *Advanced mathematical methods for scientists and engineers I: Asymptotic methods and perturbation theory*. Springer Science & Business Media, 2013.

[3] Yoshua Bengio, Jean-françois Paiement, Pascal Vincent, Olivier Delalleau, Nicolas L Roux, and Marie Ouimet. Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering. In *Advances in neural information processing systems*, pages 177–184, 2004.

[4] Matthew Brand. Fast low-rank modifications of the thin singular value decomposition. *Linear algebra and its applications*, 415(1):20–30, 2006.

[5] Andreas Buja, Deborah F Swayne, Michael L Littman, Nathaniel Dean, Heike Hofmann, and Lisha Chen. Data visualization with multidimensional scaling. *Journal of Computational and Graphical Statistics*, 17(2):444–472, 2008.

[6] James R Bunch, Christopher P Nielsen, and Danny C Sorensen. Rank-one modification of the symmetric eigenproblem. *Numerische Mathematik*, 31(1):31–48, 1978.

[7] Frederick W Byron and Robert W Fuller. *Mathematics of classical and quantum physics*. Courier Corporation, 2012.

[8] Jiu Ding and Aihui Zhou. Eigenvalues of rank-one updated matrices with some applications. *Applied Mathematics Letters*, 20(12):1223–1226, 2007.

[9] Amy N Langville and Carl D Meyer. Updating Markov chains with an eye on google’s PageRank. *SIAM Journal on Matrix Analysis and Applications*, 27(4):968–987, 2006.

[10] Amy N Langville and Carl D Meyer. *Google’s PageRank and beyond: The science of search engine rankings*. Princeton University Press, 2011.

[11] Jan R Magnus. On differentiating eigenvalues and eigenvectors. *Econometric Theory*, 1(02):179–191, 1985.
Appendix A Complementary materials

A.1 Interlacing eigenvalues

The interlacing theorem is given as complementary. For more details, see e.g., [6].

**Theorem A.1.** Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ be the eigenvalues of $A$ and $t_1 \geq t_2 \geq \cdots \geq t_n$ the eigenvalues of $A + \rho vv^T$. Then, if $\rho > 0$,

$$t_1 \geq \lambda_1 \geq t_2 \geq \lambda_2 \geq \cdots \geq t_n \geq \lambda_n,$$

else if $\rho < 0$

$$\lambda_1 \geq t_1 \geq \lambda_2 \geq t_2 \geq \cdots \geq \lambda_n \geq t_n.$$

A.2 Proof of Theorem 3.3

We use the following lemma.

**Lemma A.2.** Let $f : \mathbb{R}^n \to \mathbb{R}$ twice continuously differentiable function. Let $u, v$ be some unit vectors. If $|D_u D_v f| < T$ for some $T > 0$ then $\|\nabla^2 f\|_2 < T$. In other words, if the directional second derivative is bounded, then so is the norm of the Hessian matrix.

**Proof.** Since $\nabla^2 f$ is symmetric, by definition $\|\nabla^2 f\| = \max_{\|u\|=\|v\|=1} |u^T \nabla^2 f v|$. Let $u_0, v_0$ be the maximizes. Then,

$$\|\nabla^2 f\| = |u_0^T \nabla^2 f v_0| = |D_{u_0} D_{u_1} f| < T.$$

We now turn to prove the theorem.

**Proof of Theorem 3.3.** WLOG, we assume the diagonal entries of $S$ are different, and thus the singular values of $S$ are different. If not, add random small values to the diagonal and subtract them from the matrix $E$, resulting in $S + E$ unchanged. When the singular values of a matrix are different, they are an analytic function of the matrix entries and can thus be expanded with a Taylor expansion [11]. We use
the elegant result in [13], indicating that if \( A = UDV^T \) is the SVD of \( A \), then \( \frac{\partial \sigma_k(A)}{\partial v_{ij}} = u_{ik}v_{jk} \). Denote by \( \text{vec}(E) \) and \( \text{vec}(S) \) the matrices \( E, S \) as elements in \( \mathbb{R}^{n^2} \). So, expanding the singular value function to a first degree Taylor polynomial with a remainder yields

\[
\sigma_i(S + E) = \sigma_i(S) + \nabla \sigma_i(S) \cdot \text{vec}(E) + R
\]

\[
= \sigma_i(S) + \sum_{n,m} \frac{\partial \sigma_i(S)}{\partial s_{nm}} \cdot e_{nm} + R = \sigma_i(S) + \sum_{n,m} u_{ni}v_{mi} \cdot e_{nm} + R
\]

with \( u_i \) and \( v_i \) being the left and right singular vectors of \( S \) respectively. But \( S = ISIT \) is the SVD of \( S \) and thus the above reduces to

\[
\sigma_i(S + E) = \sigma_i(S) + u_{ii}v_{ii} e_{ii} + R = \sigma_i(S) + e_{ii} + R.
\]

The remainder \( R \) has the form \( R = \frac{1}{2} \text{vec}(E)^T \nabla^2 \sigma_i(Z) \text{vec}(E) \) with \( \nabla^2 \sigma_i(Z) \) being the Hessian matrix evaluated for a matrix that lies between \( S \) and \( S + E \), that is \( Z = S + cE \) for \( c \in (0, 1) \). Therefore, by Cauchy-Schwartz inequality we have,

\[
|R| = \frac{1}{2} |\text{vec}(E)^T \nabla^2 \sigma_i(Z) \text{vec}(E)| \leq \frac{1}{2} \|\text{vec}(E)\| \|\nabla^2 \sigma_i(Z)\| \|\text{vec}(E)\|.
\]

Now since \( \sigma_i \) is an analytic function of the matrix entries, its second derivative is locally bounded around \( S \) by some \( c_H > 0 \). By Lemma A.2 we have \( \|\nabla^2 \sigma_i(S^*)\|_2 < c_H \) for a neighborhood \( S^* \) of \( S \). We take \( \rho \) so that \( Z = S + cE \in S^* \). We then have

\[
|R| \leq \frac{1}{2} c_H \|E\|_F^2.
\]

\[\square\]

### A.3 Proof of Lemma 3.5

**Proof.** Let the node \( i \) be connected to the newly inserted node. Denote \( w_{i1} = w_{1i} = \gamma \), and so when normalizing row \( i \), a term of \( \gamma \) is added. Lets examine what happens to the entry of row \( \Delta L_{ij} \). We show the case that node \( j \) is connected to the newly inserted node, as the analysis for the other case is similar.

Denote the following scalars \( A = \sum_p g(0) + \frac{d}{dt} g(\xi_{pj}) \delta_{pj} \) and \( B = \sum_p g(0) + \frac{d}{dt} g(\xi_{ip}) \delta_{ip} \), to have

\[
|\Delta L_{ij}| = |\ell_{ij}^0 - \ell_{ij}^1| = \frac{g(0) + \frac{d}{dt} g(\xi_{ij}) \delta_{ij}}{\sqrt{A} \cdot \sqrt{B}} - \frac{g(0) + \frac{d}{dt} g(\xi_{ij}) \delta_{ij}}{\sqrt{A} \cdot \sqrt{\gamma + B}}
\]

\[
= \frac{g(0) + \frac{d}{dt} g(\xi_{ij}) \delta_{ij}}{\sqrt{A}} \cdot \left( \frac{1}{\sqrt{B}} - \frac{1}{\sqrt{\gamma + B}} \right)
\]

\[
= \frac{g(0) + \frac{d}{dt} g(\xi_{ij}) \delta_{ij}}{\sqrt{A}} \cdot \left( \frac{\sqrt{\gamma + B} - \sqrt{B}}{\sqrt{B} \cdot \sqrt{\gamma + B}} \right)
\]

23
Note that the by multiplying by $\frac{\sqrt{\gamma + B} + \sqrt{B}}{\sqrt{\gamma + B} + \sqrt{B}}$, the last difference can be written as:
\[
\frac{\gamma}{(\gamma + B)\sqrt{B} + B\sqrt{\gamma + B}}.
\]
Now, similarly to Lemma 3.4, we bound
\[
|\langle \Delta L \rangle_{ij}| < g(0) + M\delta \left( \frac{g(0) + M\delta}{(M\delta + kM\delta)\sqrt{kM\delta + kM\delta}} \right)
\]
\[
< \frac{g(0) + M\delta}{\sqrt{kM\delta}} \cdot \left( \frac{g(0) + M\delta}{kM\delta\sqrt{kM\delta + kM\delta}} \right) = \frac{(g(0) + M\delta)^2}{2(M\delta)^2k^2} = \frac{c^2}{2k^2}.
\]

Appendix B: Complexity analysis for the algorithms

B.1 Analysis for Algorithm 1

We start with determining our parameter $\mu$, the rough estimator for the unknown eigenvalues. For $\mu = 0$ no calculation is needed. For $\mu_{\text{mean}}$ we need to sum the matrix diagonal which is $O(n)$. The calculation of $\mu_*$ requires the calculation of $s$ of (11) which is $O(mn + \text{nnz}(A))$. Note that this calculation will be made anyway if we wish to solve the second order approximation of the secular equation.

Solving the variants of the truncated secular equations, namely (3) and (12), is done by standard solvers, i.e., Newton’s method. Indeed, if we consider Newton’s method, each iteration consists of summing up to $O(m)$ expressions, thus resulting in $O(m)$ operations. We also expect this method to converge in $O(1)$ iterations, as Newton method has quadratic convergence rate, giving $O(m)$ operations for one eigenvalue. We do this for all $m$ wanted eigenvalues, and thus we have $O(m^2)$ operations for all eigenvalues. However, for the second order approximation, there is an additional calculation of $s$ which is done only once and costs (as already mentioned above) $O(mn + \text{nnz}(A))$. The calculation of $z = Q_m^T v$ is also done only once so we get a total complexity of $O(m^2 + m \text{nnz}(v))$, with additional $O(mn + \text{nnz}(A))$ for the second order.

For the eigenvectors, we first discuss the evaluation of the naive formula (16). Again, $z = Q_m^T v$ needs to be calculated only once for all the eigenvectors and costs $O(m \text{nnz}(v))$ to compute. The rest of the formula involves multiplication of an $(n \times m)$ full matrix and $(m \times m)$ diagonal matrix which costs $O(mn)$. Finally, a product of an $(n \times m)$ matrix with an $m$ size vector costs $O(nm)$. Normalizing is $O(n)$. Thus, it costs $O(m \text{nnz}(v) + mn)$ to compute one eigenvector, and
\[
O(m \text{nnz}(v) + m^2n),
\]
(32)
to compute all of them.

For the all the variants of the eigenvectors calculation we add the calculation of $r$ which is $O(mn)$ and can be done only once for all eigenvectors. Therefore, asymptotically the naive formula has the exact same complexity of (32). Lastly, for the second order approximations, we add the calculation of $Ar$ which again
can be done only once and costs $O(\text{nnz}(A))$. Thus for one eigenvector we have $O(\text{nnz}(A) + m \text{nnz}(v) + mn)$ and for all of them 

$$O(\text{nnz}(A) + m \text{nnz}(v) + m^2 n).$$

### B.2 Analysis for Algorithm 2

The algorithm starts with a calculation of the new graph Laplacian matrix, that is Line 1 of Algorithm 2. This step involves updating the neighborhood of the new node, which requires $O(n)$ operations and then updating $O(k^2)$ entries, i.e., a total of $O(n + k^2)$.

Calculating the rank one update in lines 3-4 requires applying power iteration to $\Delta L$ which is a sparse $n \times n$ matrix, with large spectral gap (Theorem 3.1). We can thus expect a high convergence rate, with no more than a few iterations. In fact, the update is very “local” in a sense that an immediate upper bound of the size of the perturbation is $O(k^2) \times O(k^2)$, as stated in the next lemma.

**Lemma B.1.** $\text{size}(\Delta L) = O(k^2) \times O(k^2)$ and $\text{nnz}(v) = O(k^2)$

**Proof.** The immediate $O(k^2) \times O(k^2)$ bound can be achieved by noting that in the worst case, non of the $O(k)$ neighbors of the newly connected node are within the $kNN$ of one another. This means that the rows normalization will affect distinct $O(k)$ columns in $O(k)$ rows, a total of $O(k) \times O(k^2)$ rows $\times$ columns. Due to symmetry, the same goes for the columns normalization, giving a total of $O(k^2) \times O(k^2)$ rows $\times$ columns that had been changed. $\square$

We demonstrate Lemma B.1. We calculate a graph Laplacian $L_0$ of a given set of nodes, and then connect a new node resulting in a new graph Laplacian $L_1$. We count only the nonzero rows of $L_1 - L_0$, which is the real size of $\Delta L$ (since the calculations can be done only on that sub-matrix with no loss in accuracy). We repeat this process with several values of $k$ to check the dependency of $k$. We use the notation $\text{size}(X)$ for the product of the number of rows of a matrix $X$ with the number of columns of $X$. Since $\text{size}(\Delta L) = c \cdot k^\alpha$, applying log on both sides will give $\log(\text{size}(\Delta L)) = \alpha \cdot k + c$. Thus plotting $\log(\text{size}(\Delta L))$ vs. $\log(k)$ should result in a linear line with slope equals to $\alpha$. This is nicely demonstrated in Figure 6.

In Figure 6, we present empirical results from real data that shows that in practice the bound is even lower. We will thus assume that

$$\text{size}(\Delta L) = O(k^\alpha) \times O(k^\alpha),$$

with $0 < \alpha \leq 2$ as just proven. The real value of $\alpha$ is dataset dependent. The intuition behind this is that connecting a new node to the graph will affect only its immediate neighbors, thus the rows and columns corresponding to distant vertices would not be affected. We conclude that the complexity of this stage is $O(k^{2\alpha})$, with $\alpha \leq 2$. The fact that $\text{nnz}(v) = O(k^\alpha)$ is a direct conclusion.

We now analyze the perturbation correction. For one eigenvalue, the correction consists of a multiplication of the form $q^T C q = q^T (C q)$, this means $O(\text{nnz}(C))$ operations for $Cq$ and $O(n)$ operations for $q^T (C q)$. Thus the total number of arithmetic operations for one eigenvalues is $O(n + \text{nnz}(C))$ and $O(mn + m \text{nnz}(C))$ for correcting all the eigenvalues. For one eigenvector, we sum up $m$ elements of the form $q^T \frac{C}{\lambda} q$. Based on the analysis before, each element is $O(n + \text{nnz}(C))$ operations to compute, resulting in $O(mn + m \text{nnz}(C))$ for one eigenvector and
Figure 6: Demonstration of size(\(\Delta L\)) = \(O(k^\alpha)\) for several datasets

We can see a clear linear dependency between \(\log \text{size}(\Delta L)\) and \(\log k\) as expected by Lemma B.1, with slope smaller than 2.

| Algorithm               | Complexity      |
|-------------------------|-----------------|
| \(\mu\)                |                 |
| 0                       | \(O(1)\)       |
| \(\mu_*\)               | \(O(mn + \text{nnz}(A))\) |
| \(\mu_{\text{mean}}\)  | \(O(n)\)       |
| Eigenvalues             |                 |
| First order             | \(O(m \text{nnz}(v) + m^2)\) |
| Second order            | \(O(m \text{nnz}(v) + m^2 + mn + \text{nnz}(A))\) |
| Perturbation Correction | \(O(mn + mn \text{nnz}(C))\) |
| Eigenvectors            |                 |
| First order             | \(O(m^2n + \text{nnz}(v))\) |
| Second order            | \(O(m^2n + \text{nnz}(A) + mn \text{nnz}(v))\) |
| Perturbation Correction | \(O(m^2n + m^2 \text{nnz}(C))\) |

Table 6: Summary of complexity analysis

\(O(m^2n + m^2 \text{nnz}(C))\) for all eigenvectors. The number of nonzeros in \(C\) is concluded next.

**Lemma B.2.** \(\text{nnz}(C) = O(k^{2\alpha})\) with \(0 < \alpha \leq 2\)

**Proof.** Recall that \(C = L_1 - (L_0 + \lambda vv^T) = (L_1 - L_0) - \lambda vv^T\). It was already proven that \(\text{nnz}(L_1 - L_0) = O(k^2)\). Also, \(v\) is the first singular vector of an at most \(O(k^\alpha) \times O(k^\alpha)\) matrix, thus \(\text{nnz}(\lambda vv^T) = O(k^{2\alpha})\).

A simple conclusion is that \(\text{nnz}(C) = O(k^{2\alpha})\). □

Based on Lemma B.1 and B.2 we can write

\(\text{nnz}(v) = O(k^\alpha)\) and \(\text{nnz}(C) = O(k^{2\alpha})\).

The entire analysis is summarized in Table 6.