SIAM J. MATRIX ANAL. APPL.
Vol. 40, No. 3, pp. 973–997

SYMMETRIC RANK-ONE UPDATES FROM PARTIAL SPECTRUM WITH AN APPLICATION TO OUT-OF-SAMPLE EXTENSION*

ROY MITZ†, NIR SHARON†, AND YOEL SHKOLNISKY†

Abstract. Rank-one update of the spectrum of a matrix is a fundamental problem in classical perturbation theory. In this paper, we consider its variant where only part of the spectrum is known. We address this variant using an efficient scheme for updating the known eigenpairs with guaranteed error bounds. Then, we apply our scheme to the extension of the top eigenvectors of the graph Laplacian to a new data sample. In particular, we model this extension as a perturbation problem and show how to solve it using our rank-one updating scheme. We provide a theoretical analysis of this extension method and back it up with numerical results that illustrate its advantages.

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

AMS subject classifications. 15A18, 47J10, 65D99

DOI. 10.1137/18M1172120

1. Introduction.

The last few decades have witnessed the emergence of various algorithms that require the calculation of the eigendecomposition of matrices. A few known examples are Google's PageRank [18], PCA [21], Laplacian eigenmaps [1], LLE [20], and MDS [5]. Since datasets nowadays may contain tens of millions of data points, an efficient calculation of the eigendecomposition 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 calculations. 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 repeating the entire decomposition again.

In this paper, we focus on rank-one updates of symmetric matrices. The classical approach for such an update is updating the eigenvalues using the roots of the secular equation (see, e.g., [6]). However, several other approaches for updating the eigenvalues and eigenvectors of a perturbed matrix have been suggested. The popular ones are quite general and include recalculating from scratch or restarting the power method [14] and perturbation methods [23]. Some methods that utilize the structure of a specific problem were suggested, with Google's page rank being the most popular application [15, Chapter 10]. Another important method is based on a geometric embedding of the available data [4]. This approach becomes computationally attractive when one updates a low-rank matrix.

*Received by the editors February 21, 2018; accepted for publication (in revised form) by Z. Drmac June 3, 2019; published electronically August 13, 2019.
https://doi.org/10.1137/18M1172120
Funding: The work of the authors was supported by the Israel Science Foundation grant 578/14, by the NIGMS award R01GM090200, and by the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme grant 723991–CRYOMATH. The work of the second author was partially supported by the Moore Foundation Data-Driven Discovery Investigator Award.
†School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel (roymitz@mail.tau.ac.il, nsharon@tauex.tau.ac.il, yoelsh@tauex.tau.ac.il).

Many of the methods mentioned above are inapplicable or provide very poor guarantees in cases where we do not have access to the complete eigendecomposition. Some methods assume that the updated matrix is low-rank, which is not always the right model for real-world data. Finally, almost none of the existing approaches is equipped with error analysis. In our method, we provide a rank-one update algorithm that does not require the full eigendecomposition of the matrix and does not assume that it is low rank. We demonstrate that the structure of the problem enables us to use the unknown tail of the eigenvalues in order to improve the accuracy of the update. Additionally, the complexity of our algorithm is linear in the number of rows of the matrix. We also analyze the accuracy of our method, showing that it is independent of the number of unknown eigenpairs, but rather only depends on their "behavior." This observation is confirmed by both synthetic and real-world examples.

The eigenvalues and eigenvectors of the graph Laplacian have been of special interest recently, as evident by its various applications in machine learning, dimensionality reduction [1, 10], clustering [17, 24], graph theory [9], and image processing [11]. The problem of out-of-sample extension of the graph Laplacian, which will be described in detail later, is essentially updating the eigendecomposition of the graph Laplacian after the insertion of a new vertex to the graph. This problem is often addressed by the Nyström method [3]. We propose a different approach to the extension, based on the observation that under mild assumptions, the insertion of a new vertex to a graph translates to an almost rank-one update of the corresponding graph Laplacian matrix. Then, we apply our rank-one update algorithm to estimate the extension with high accuracy.

The paper is organized as follows. In section 2, we derive our algorithm for the symmetric rank-one update based on partial eigendecomposition and analyze its error. In section 3, we describe the application of the algorithm to the extension of the graph Laplacian. In section 4, we illustrate numerically some of our theoretical results from sections 2 and 3 for both synthetic and real data. We give some concluding remarks in section 5.

2. Rank-one update with partial spectral information. Computing the spectrum of a matrix following its rank-one update is a classical task in perturbation theory, e.g., [2, Chapter 7]. Given a rank-one perturbation to a matrix, the spectrum of the perturbed matrix is related to that of the original matrix via the secular equation, which involves the entire spectrum of the original matrix. However, if only the few leading eigenpairs of the original matrix are known, the classical approach requires further adaptation. Inspired by [6], we propose a solution to the "partial knowledge" rank-one update problem, where we aim to estimate the leading eigenpairs of a matrix after a rank-one perturbation, having only part of the eigendecomposition of the original matrix. We describe in detail the derivation of our method and provide error bounds and complexity analysis.

2.1. Notation, classical setting, and problem formulation. We denote by $X = [x_1 x_2 \cdots x_n]$ a matrix expressed by its column vectors, and by $X^{(m)} = [x_1 \cdots x_m]$ its truncated version consisting only of its first m columns, $m < n$. Let A be an $n \times n$ symmetric real matrix with real (not necessarily 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 $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. We focus on the problem of (symmetric) rank-one update, where we wish to find the eigendecomposition of

$$A + \rho v v^T, \quad \rho \in \mathbb{R}, \quad v \in \mathbb{R}^n, \quad \|v\| = 1. \tag{2.1}$$

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 $A + \rho vv^T = PTP^T$ with $P = [p_1 p_2 \cdots p_n]$ and $T = \operatorname{diag}(t_1, \ldots, t_n)$. Approximated objects (whether scalars or vectors) constructed in this section are denoted by an over tilde. For example, an approximation for x is denoted by $\widetilde{x}$.

The relation between the decompositions before and after the rank-one update is well-studied, e.g., [6, 12]. Without loss of generality, we further assume that $\lambda_1 > \lambda_2 > \cdots > \lambda_n$ and that for $z = Q^T v$ we have $z_j \neq 0$ for all $1 \leq j \leq n$. The deflation process in [6] reduces any update (2.1) to this form. Given the eigendecomposition $A = Q\Lambda Q^T$, the updated eigenvalues $t_1, \ldots, t_n$ of $A + \rho vv^T$ are given by the n roots of the secular equation

$$w(t) = 1 + \rho \sum_{i=1}^{n} \frac{z_i^2}{\lambda_i - t}, \quad z = Q^T v. \tag{2.2}$$

The corresponding eigenvector for the kth root (eigenvalue) t_k is given by the explicit formula

$$p_k = \frac{Q\Delta_k^{-1} z}{\left\| Q\Delta_k^{-1} z \right\|}, \quad z = Q^T v, \quad \Delta_k = \Lambda - t_k I. \tag{2.3}$$

An 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 one can do in cases where only part of the spectrum is known. Thus, we are interested in the following problem. Let A be an $n \times n$ real symmetric matrix and let $1 \leq m < n$. Assume we have only the first m leading eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_m$ of A and their associated eigenvectors $q_1, \ldots, q_m$. Find an estimate to the first m leading eigenpairs of $A + \rho vv^T$ with $\rho \in \mathbb{R}$ and $\|v\| = 1$.

2.2. Truncating the secular equation. We start by considering the first part of the above problem—the eigenvalues. The classical perturbation method solves for the eigenvalues of $A + \rho vv^T$ by finding the roots of the secular equation (2.2). We introduce two modifications of the secular equation, adapted to our new setting.

Using the notation of (2.2), we have from the orthogonality of Q that

$$\|z\|^2 = \|Q^T v\|^2 = \|v\|^2 = 1. \tag{2.4}$$

Therefore, $\sum_{j=m+1}^{n} z_j^2 = 1 - \sum_{i=1}^{m} z_i^2$. Since the last $n - m$ eigenvalues of A are unknown, we denote by $\mu < \lambda_m$ a fixed scalar, whose purpose is to approximate λ_j, $j = m+1, \ldots, n$. Choosing μ will be discussed below. We then define the first order truncated secular equation by

$$w_1(t; \mu) = 1 + \rho \left(\sum_{i=1}^{m} \frac{z_i^2}{\lambda_i - t} + \frac{1 - \sum_{i=1}^{m} z_i^2}{\mu - t} \right), \tag{2.5}$$

where $z = (Q^{(m)})^T v$ is a vector of length m (the first m entries of $Q^T v$), with the columns of the matrix $Q^{(m)}$ consisting of the (orthogonal) eigenvectors corresponding to the (known) leading eigenvalues of A.

As a first observation, we bound the error obtained from the new formula. Namely, we show that the deviation of the m largest roots of the truncated secular equation (2.5) from the roots $t_1, \ldots, t_m$ of (2.2) is of the order of $\max_{m+1 \leq j \leq n} |\lambda_j - \mu|$.

PROPOSITION 2.1. *Let $\rho > 0$. Then, there exist m roots $\widetilde{t}_1, \ldots, \widetilde{t}_m$ of $w_1(t;\mu)$ of (2.5), such that*

$$\left| t_k - \widetilde{t}_k \right| \leq C_k \max_{m+1 \leq j \leq n} \left| \lambda_j - \mu \right|, \quad k = 1, \ldots, m, \tag{2.6}$$

where $t_1, \ldots, t_m$ are the largest m roots of (2.2), and C_k is a constant bounded by

$$(\lambda_k - \mu)^{-1} (\lambda_m - \lambda_{m+1})^{-1} \max\{(\lambda_k - \lambda_1)^2, (\lambda_{k-1} - \lambda_n)^2\}.$$

Proof. We start by showing the existence of the roots of (2.5). Indeed, since $w_1(t;\mu)$ is monotone and since $\lim_{t \to \lambda_i^{\pm}} w_1(t;\mu) = \mp\infty$ for $i = 1, \ldots, m$, there exists a single root in any segment $[\lambda_j, \lambda_{j-1}]$, $2 \leq j \leq m$. Additionally, since $\lim_{t \to \infty} w_1(t;\mu) = 1$ and by classical perturbations bounds for the eigenvalues of symmetric matrices [13, Corollary 8.1.6], there exits a single root in the interval $[\lambda_1, \lambda_1 + \rho]$. These are the m roots of $w_1(t,\mu)$ in the interval $[\lambda_m, \lambda_1 + \rho]$.

For the bound in (2.6), we expand

$$\frac{z_i^2}{\lambda_i - t} = \frac{z_i^2}{\mu - t} + \frac{z_i^2 (\mu - \lambda_i)}{(\mu - t)(\lambda_i - t)}. \tag{2.7}$$

Therefore, by splitting the sum in (2.2) and using (2.7) we have

$$w(t) = \underbrace{1 + \rho \sum_{i=1}^{m} \frac{z_i^2}{\lambda_i - t} + \rho \sum_{i=m+1}^{n} \frac{z_i^2}{\mu - t}}_{w_1(t;\mu)} + \underbrace{\rho \sum_{i=m+1}^{n} \frac{z_i^2 (\mu - \lambda_i)}{(\mu - t)(\lambda_i - t)}}_{e(t;\mu)}. \tag{2.8}$$

By Taylor expansion of (2.2),

$$0 = w(\widetilde{t}_k - (\widetilde{t}_k - t_k)) = w(\widetilde{t}_k) - (\widetilde{t}_k - t_k) \frac{d}{dt} w(\xi) = w_1(\widetilde{t}_k;\mu) + e(\widetilde{t}_k;\mu) - (\widetilde{t}_k - t_k) \frac{d}{dt} w(\xi), \tag{2.9}$$

where $\widetilde{t}_k, \xi \in (\lambda_k, \lambda_{k-1})$.

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

$$(\widetilde{t}_k - t_k) = \frac{e(\widetilde{t}_k;\mu)}{\frac{d}{dt} w(\xi)}. \tag{2.10}$$

For the error term $e(\widetilde{t}_k;\mu)$ we have

$$\begin{aligned} \left| e(\widetilde{t}_k;\mu) \right| &\leq \frac{\rho}{\left| \mu - \widetilde{t}_k \right| \left| \lambda_{m+1} - \widetilde{t}_k \right|} \sum_{i=m+1}^{n} z_i^2 |\lambda_i - \mu| \\ &\leq \frac{\rho}{\left| \mu - \widetilde{t}_k \right| \left| \lambda_{m+1} - \widetilde{t}_k \right|} \max_{m+1 \leq i \leq n} |\lambda_i - \mu|, \end{aligned} \tag{2.11}$$

where the last inequality follows since $\|z\|_2 = 1$. In addition, $\widetilde{t}_k \geq \lambda_m$ and $\mu < \lambda_m$ so the denominator in (2.11) is bounded from below by $|\mu - \lambda_k||\lambda_m - \lambda_{m+1}|$. Back to (2.10), the derivative of the secular equation is

$$\frac{d}{dt} w(t) = \rho \sum_{i=1}^{n} \frac{z_i^2}{(\lambda_i - t)^2}, \tag{2.12}$$

and thus

(2.13)
$$\left|\frac{d}{dt}w(t)\right| \geq \rho\sum_{i=1}^{n}\frac{z_i^2}{(\lambda_i-t)^2} \geq \rho\min_{1\leq i\leq n}\left\{\frac{1}{(\lambda_i-t)^2}\right\}\sum_{i=1}^{n}z_i^2 = \rho\left\{\max_{1\leq i\leq n}(\lambda_i-t)^2\right\}^{-1}.$$

Therefore,

$$\left|\frac{1}{\frac{d}{dt}w(t)}\right| \leq \frac{\max\{(\lambda_k-\lambda_1)^2,(\lambda_{k-1}-\lambda_n)^2\}}{\rho}, \quad t\in(\lambda_k,\lambda_{k-1}). \tag{2.14}$$ ∎

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 $\widetilde{t}_m$ is merely guaranteed to lie in the segment $[\mu, \lambda_m]$. Consequently, the constant C_m cannot be bounded with the same arguments.

We now address the issue of choosing μ. A common assumption in many real world applications is that the matrix A is low-rank, and thus the unknown eigenvalues are zero, implying the choice $\mu = 0$. This is indeed the case for several important kernel matrices, as we will see in the next section. For matrices that are not low rank, the error term of Proposition 2.1 using $\mu = 0$ would be $O(|\lambda_{m+1}|)$ and we have no reason to believe that this will result in a good approximation.

A better method for choosing μ would be to minimize the sum in the middle term of (2.11). However, an analytic minimizer is not attainable in this case since both λ_i and z_i^2, $i = m+1,\ldots,n$, are unknown. Shortly, we will devise an approximation of the secular equation for which an analytic minimizer can be calculated. Nevertheless, assuming the trace of A is available, an intuitive choice for μ that works well in practice and is also fast to compute is the mean of the unknown eigenvalues, which is accessible since

$$\mu_{mean} = \frac{\sum_{i=m+1}^{n}\lambda_i}{n-m} = \frac{\mathrm{tr}(A)-\sum_{i=1}^{m}\lambda_i}{n-m}. \tag{2.15}$$

Following the proof of Proposition 2.1, we are encouraged to try to improve the approximation to the eigenvalues of (2.1) by using a higher order approximation for (2.7), namely,

$$\frac{z_i^2}{\lambda_i-t} = \frac{z_i^2}{\mu-t} - \frac{z_i^2(\lambda_i-\mu)}{(\mu-t)^2} + \frac{z_i^2(\lambda_i-\mu)^2}{(\mu-t)^2(\lambda_i-t)}. \tag{2.16}$$

Since $Aq_i = \lambda_i q_i$, we have

(2.17)
$$\sum_{i=m+1}^{n}z_i^2\lambda_i = \sum_{i=m+1}^{n}(q_i^Tv)(q_i^Tv)\lambda_i = \sum_{i=m+1}^{n}(v^T\lambda_i q_i)(q_i^Tv) = \sum_{i=m+1}^{n}(v^TAq_i)(q_i^Tv),$$

and thus

$$\sum_{i=m+1}^{n}z_i^2\lambda_i = v^TA\left(I-Q^{(m)}(Q^{(m)})^T\right)v \triangleq s, \tag{2.18}$$

which is a known quantity. This analysis gives rise to the second order approximation of the secular equation

$$w_2(t;\mu) = 1+\rho\left(\sum_{i=1}^{m}\frac{z_i^2}{\lambda_i-t} + \frac{1-\sum_{i=1}^{m}z_i^2}{\mu-t} - \frac{s-\mu(1-\sum_{i=1}^{m}z_i^2)}{(\mu-t)^2}\right). \tag{2.19}$$

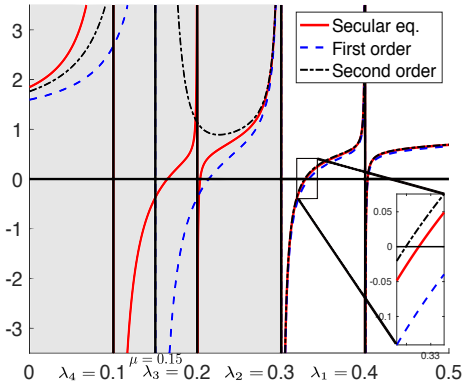


FIG. 1. *The secular equation* (2.2) *and its two approximations: the first order* w_1 *of* (2.5) *and the second order* w_2 *of* (2.19). *The original matrix has four eigenvalues at* $0.1, 0.2, 0.3$, *and* 0.4 *and a rank-one update with* $\rho > 0$. *The approximations use* $m = 2$, *and* $\mu = \mu_{mean} = 0.15$. *In the lower right corner, we zoom in to a small neighborhood of the second root.*

In this case, the roots of $w_2(t;\mu)$ of (2.19) are at most $\max_{m+1\le i\le n}(\lambda_i - \mu)^2$ away from the roots of the original secular equation (2.2). This is concluded in the next result, which is analogous to Proposition 2.1.

PROPOSITION 2.2. *Let* $\rho > 0$. *Then, there are* m *roots* $\widetilde{t}_1, \ldots, \widetilde{t}_m$ *of* $w_2(t;\mu)$ *of* (2.19), *such that*

$$\left| t_k - \widetilde{t}_k \right| \le C_k \max_{m+1\le j\le n} (\lambda_j - \mu)^2, \quad k = 1, \ldots, m, \tag{2.20}$$

where $t_1, \ldots, t_m$ *are the largest* m *roots of* (2.2), *and* C_k *is a constant bounded by*

$$(\lambda_k - \mu)^{-2}(\lambda_m - \lambda_{m+1})^{-1} \max\{(\lambda_k - \lambda_1)^2, (\lambda_{k-1} - \lambda_n)^2\}.$$

Proof. This proof is similar to the proof of Proposition 2.1. Here, we have $w(t) = w_2(t;\mu) + e(t;\mu)$ with

$$e(t;\mu) = \rho \sum_{i=m+1}^{n} \frac{z_i^2(\lambda_i - \mu)^2}{(\mu - t)^2(\lambda_i - t)}. \tag{2.21}$$

Then, due to the additional $\mu - t$ in the denominator of the error term (2.21) compared to the error term of (2.8), the bound for (2.10) has an additional factor of $(\lambda_k - \mu)^{-1}$ in the constant C_k. ∎

To conclude the above discussion on the two approximations of the secular equation, we present Figure 1. In this figure, we construct a matrix of size $n = 4$ with eigenvalues $0.1, 0.2, 0.3, 0.4$ and random orthogonal eigenvectors. To form the truncated equations, we use $m = 2$ and $\mu = \mu_{mean}$ of (2.15), which in this case satisfies $\mu = 0.15$. The figure depicts the two truncated secular equations w_1 of (2.5) and w_2 of (2.19), for a rank-one update with $\rho > 0$, alongside the original secular equation of (2.2). The two roots that are approximated are on the white part of the figure. 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. The other two roots (that are not approximated)

are on the grey part of the figure, where the asymptotic behavior around μ of the two approximations is demonstrated.

We address once again the choice of μ. Setting $\mu = 0$ implies that the eigenvalues λ_j, $j = m+1, \ldots, n$ are assumed to be small. Then, we get according to Proposition 2.2 an improved error of $O(\lambda_{m+1}^2)$. Nevertheless, in this case of a second order approximation to the secular equation (2.19), an improved method for choosing μ is possible by minimizing an upper bound on (2.21). For simplicity, we further assume that $\mu \leq \lambda_{m+1}$. Since the denominator in (2.21) depends on t (specifically, on the yet to be calculated approximated eigenvalues), we bound it by some constant that will apply to all eigenvalues simultaneously. One such bound is $e(t;\mu) \leq \rho(\lambda_m - \lambda_{m+1})^{-3} \sum_{i=m+1}^{n} z_i^2(\lambda_i - \mu)^2$ which holds for all $t \in (\lambda_m, \lambda_1 + \rho)$. We would then like to minimize $\sum_{i=m+1}^{n} z_i^2(\lambda_i - \mu)^2$. By standard methods we get the minimizer

$$\mu_* = \frac{\sum_{i=m+1}^{n} z_i^2 \lambda_i}{\sum_{i=m+1}^{n} z_i^2} = \frac{s}{1 - \sum_{i=1}^{m} z_i^2}, \tag{2.22}$$

where s is defined in (2.18). The minimizer μ_* is essentially a weighted mean of the unknown eigenvalues (and thus obeys the assumption $\mu \leq \lambda_{m+1}$). Unlike μ_{mean}, this variant does not require the knowledge of $\mathrm{tr}(A)$ but rather a few matrix-vector evaluations to calculate s of (2.18). Interestingly, note that when using $\mu = \mu_*$ we have

$$w_2(t;\mu_*) = w_1(t;\mu_*), \tag{2.23}$$

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

Next, we address the problem of eigenvectors estimation.

2.3. Truncated formulas for the eigenvectors. In this section, we introduce two approximations to the eigenvectors formula (2.3). These are analogous to the approximations to 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 (2.3) is by calculating

$$\widetilde{p_i} = Q^{(m)}(\Delta_i^{(m)})^{-1}(Q^{(m)})^T v, \quad \Delta_i^{(m)} = \mathrm{diag}\left(\lambda_1 - t_i, \ldots, \lambda_m - t_i\right), \quad i = 1, \ldots, m, \tag{2.24}$$

followed by a normalization, 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, namely (see (2.3)),

$$\begin{aligned} p_i &= Q(\Lambda - t_i I)^{-1} Q^T v \\ &= \begin{bmatrix} q_1 & \ldots & q_n \end{bmatrix} \begin{bmatrix} \frac{\langle q_1, v\rangle}{\lambda_1 - t_i} \\ \cdots \\ \frac{\langle q_n, v\rangle}{\lambda_n - t_i} \end{bmatrix} = \sum_{k=1}^{n} \frac{\langle q_k, v\rangle}{\lambda_k - t_i} q_k \\ &= \underbrace{\sum_{k=1}^{m} \frac{\langle q_k, v\rangle}{\lambda_k - t_i} q_k}_{\text{known}} + \underbrace{\sum_{k=m+1}^{n} \frac{\langle q_k, v\rangle}{\lambda_k - t_i} q_k}_{\text{unknown}}. \end{aligned} \tag{2.25}$$

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

$$\sum_{k=m+1}^{n} \langle q_k, v\rangle q_k = \sum_{k=1}^{n} \langle q_k, v\rangle q_k - \sum_{k=1}^{m} \langle q_k, v\rangle q_k = v - \sum_{k=1}^{m} q_k q_k^T v = v - Q^{(m)}(Q^{(m)})^T v. \tag{2.26}$$

Again, we denote by μ a fixed parameter whose purpose is to approximate the unknown eigenvalues. Having the m leading eigenvectors in $Q^{(m)}$, and recalling that $\Delta_i^{(m)} = \operatorname{diag}(\lambda_1 - t_i, \ldots, \lambda_n - t_i)$, we define the first order truncated eigenvectors formula for $1 \le i \le m$ as

$$\widetilde{p_i} = Q^{(m)}(\Delta_i^{(m)})^{-1}(Q^{(m)})^T v + \frac{1}{\mu - t_i} r, \quad r = v - Q^{(m)}(Q^{(m)})^T v. \tag{2.27}$$

The second order truncated eigenvectors formula, which is the eigenvectors' analogue of (2.19), is given by

$$\begin{aligned} \widetilde{p_i} &= Q^{(m)}(\Delta_i^{(m)})^{-1}(Q^{(m)})^T v + \left(\frac{1}{\mu - t_i} + \frac{\mu}{(\mu - t_i)^2}\right) r - \frac{1}{(\mu - t_i)^2} Ar, \\ r &= v - Q^{(m)}(Q^{(m)})^T v. \end{aligned} \tag{2.28}$$

Note that r and Ar are constant vectors and can be computed once for all $1 \le i \le m$.

The error bounds of both formulas (2.27) and (2.28) are summarized in the following theorem.

THEOREM 2.3. *Let $A = Q\Lambda Q^T$ be an $n \times n$ real symmetric matrix with m known leading eigenvalues $\lambda_1 > \cdots > \lambda_m$ and known corresponding eigenvectors $q_1, \ldots, q_m$. The m leading eigenvectors $p_1, \ldots, p_m$ of the rank-one update $A + \rho v v^T$ with $\|v\| = 1$ and $\rho \in \mathbb{R}$ can be approximated by (2.27) or (2.28), given their associated leading eigenvalues $t_1, \ldots, t_m$ and a fixed scalar μ, such that the approximations satisfy the following:*

1. *For $\widetilde{p_i}$ of (2.27),*

$$\|p_i - \widetilde{p_i}\| \le C_i \max_{m+1 \le i \le n} |\lambda_i - \mu|, \quad C_i \le |\mu - t_i|^{-1} |\lambda_{m+1} - t_i|^{-1}. \tag{2.29}$$

2. *For $\widetilde{p_i}$ of (2.28),*

$$\|p_i - \widetilde{p_i}\| \le C_i \max_{m+1 \le i \le n} |\lambda_i - \mu|^2, \quad C_i \le |\mu - t_i|^{-2} |\lambda_{m+1} - t_i|^{-1}. \tag{2.30}$$

Proof. For (2.29), by (2.25) and (2.27) we have

$$e_i = \|p_i - \widetilde{p_i}\| = \left\| \sum_{k=m+1}^{n} \frac{\langle q_k, v \rangle}{\lambda_k - t_i} q_k + \frac{1}{\mu - t_i} r \right\|.$$

Similarly to (2.7), $\frac{1}{\lambda_k - t_i} = \frac{1}{\mu - t_i} + \frac{(\mu - \lambda_k)}{(\mu - t_i)(\lambda_k - t_i)}$, and using (2.26) and the orthogonality of q_i, we have

$$\begin{aligned} e_i^2 &= \left\| \sum_{k=m+1}^{n} \frac{(\mu - \lambda_k)\langle q_k, v \rangle}{(\mu - t_i)(\lambda_k - t_i)} q_k \right\|^2 = \sum_{k=m+1}^{n} \left(\frac{(\mu - \lambda_k)\langle q_k, v \rangle}{(\mu - t_i)(\lambda_k - t_i)} \right)^2 \\ &\le \frac{1}{(\mu - t_i)^2 (\lambda_{m+1} - t_i)^2} \sum_{k=m+1}^{n} (\mu - \lambda_k)^2 \langle q_k, v \rangle^2 \\ &= \frac{1}{(\mu - t_i)^2 (\lambda_{m+1} - t_i)^2} \sum_{k=m+1}^{n} (\mu - \lambda_k)^2 z_k^2. \end{aligned} \tag{2.31}$$

Since $\|Q^T v\| = 1$, taking the square root of (2.31) gives us the bound.

For (2.30), we use (2.16) 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 (2.28) with exact eigenvalues,

$$\|p_i - \widetilde{p}_i\|^2 = \left\| \sum_{k=m+1}^{n} \frac{(\mu - \lambda_k)^2 \langle q_k, v \rangle}{(\mu - t_i)^2 (\lambda_k - t_i)} q_k \right\|^2 = \sum_{k=m+1}^{n} \left(\frac{(\mu - \lambda_k)^2 \langle q_k, v \rangle}{(\mu - t_i)^2 (\lambda_k - t_i)} \right)^2, \tag{2.32}$$

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 (2.27) and (2.28), respectively. The value of μ which minimizes the bound on the last term in (2.31) is μ_* of (2.22), and is thus expected to provide a better approximation than $\mu = 0$. Experimental results have shown that the choice $\mu = \mu_{mean}$ of (2.15) is competitive with μ_* while being slightly faster to compute. Note that the approximate formulas (2.27) and (2.28) will generally not produce an orthogonal set of vectors. In that case, a reorthogonalization procedure may be used. This issue is discussed in Appendix C.

2.4. Algorithm summary. Given a parameter μ, we have provided first and second order truncated approximations to the secular equation and corresponding formulas for the eigenvectors. As for μ, we suggested three choices. If the matrix is low-rank, choose $\mu = 0$. Otherwise, choose either μ_* which minimizes the error term, or μ_{mean} which is faster to compute. We summarize the previous subsections in Algorithm 2.1, which computes the symmetric rank-one update with partial spectrum.

Algorithm 2.1 Rank-one update with partial spectrum.

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

Output: An approximation $\{(\widetilde{t}_i, \widetilde{p}_i)\}_{i=1}^m$ of the eigenpairs of $A + \rho v v^T$

1: Choose a parameter μ (i.e., $\mu = 0$, (2.15) or (2.22)).
2: Calculate the m largest roots $\{(\widetilde{t}_i\}_{i=1}^m$ of a truncated secular equation (either (2.5) or (2.19))
3: **for all** $\{q_i\}_{i=1}^m$ **do**
4: find $\widetilde{p}_i$ by a truncated eigenvectors formula (either (2.27) or (2.28))
5: **end for**

A complexity analysis of Algorithm 2.1 is provided in Appendix B.1.

3. Updating the graph Laplacian for out-of-sample extension. In this section, we introduce an application of the rank-one update scheme of section 2, to the problem of out-of-sample extension of the graph Laplacian. We start by formulating the problem, and then justify the use of a rank-one update by proving that a single point extension of the graph Laplacian is close to a rank-one perturbation. We conclude the section with a few algorithms, which are demonstrated numerically in section 4.

3.1. Preliminaries and problem formulation. We begin by introducing the notation and the model for the extension problem. Given a set of discrete points

$\mathcal{X} = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$, we define a weighted graph whose vertices are the given points. An edge is added to the graph if its two vertices are "similar." The common ways of defining "similar" include the following:

1. k-nearest neighbors (kNN)—Vertex i and j are connected iff i is within the kNN of j or vice versa.
2. δ-neighborhood—Vertex i and j are connected iff $\|x_i - x_j\| < \delta$ for some $\delta > 0$.

Each edge in the graph is assigned a weight, usually determined by a kernel function. A kernel function K is a symmetric function $K\colon \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$. The weight on the edge between vertices i and j is set to $w_{ij} = K(x_i, x_j)$. A kernel is said to be radial if

$$K(x,y) = g\left(\|x-y\|\right), \quad x, y \in \mathbb{R}^d, \tag{3.1}$$

for a nonnegative real function g. One common choice of a kernel is the heat kernel (also known as the Gaussian kernel) that induces the weights

$$w_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{\varepsilon}\right) \tag{3.2}$$

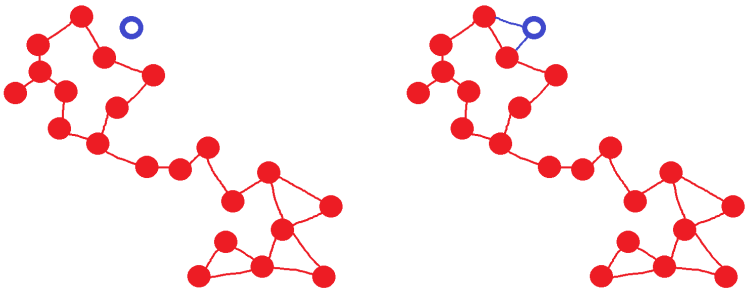
for some fixed width parameter $\varepsilon > 0$.

Given the weight matrix $W = \{w_{ij}\}$ and its corresponding (diagonal) degrees matrix D whose diagonal is $D_{ii} = \sum_{j=1}^n W_{ij}$, the graph Laplacian is typically defined as either $L = D^{-1}W$ (random walk graph Laplacian) or $L = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$ (symmetric normalized graph Laplacian) [1]. Note that most authors define the symmetric normalized graph Laplacian as $L = I - D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$. The latter definition of the graph Laplacian merely applies an affine transformation to the eigenvalues of the graph Laplacian and does not change the corresponding eigenvectors. Since we define our method to act on the largest eigenvalues, we prefer using $L = D^{-\frac{1}{2}}WD^{-\frac{1}{2}}$. Recall that W and L are $n \times n$ matrices where n is the number of samples in $\mathcal{X}$. In the following, we consider only the case of the symmetric normalized graph Laplacian. Nevertheless, similar results can be obtained for the random walk graph Laplacian, as it satisfies a similarity relation with the symmetric graph Laplacian. Henceforth, unless otherwise stated, by referring to the "graph Laplacian" we mean the symmetric normalized graph Laplacian.

We now formulate the 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 point in $\mathbb{R}^d$. Denote by L_0 the graph Laplacian constructed from $\mathcal{X}$ using a given kernel. Assume the top m eigenvalues and eigenvectors of L_0 are known ($m < n$). The out-of-sample extension problem is to find the top m eigenpairs of L_1, the graph Laplacian constructed from $\mathcal{X} \cup \{x_0\}$.

The out-of-sample extension problem is reduced to a symmetric rank-one update as follows. With a slight abuse of notation, we also denote by L_0 the original graph Laplacian to which we added x_0 as an isolated vertex. That is, L_0 is now an $(n+1) \times (n+1)$ matrix whose first row and column correspond to the point x_0: the first row and column have 1 on the diagonal and 0 otherwise. Note that the dimensions of this augmented L_0 are identical to the dimensions of L_1. We will argue that the difference matrix $\Delta L = L_1 - L_0$ is very close to being rank-one (a claim that will be formulated and proved in the next section). In other words, by looking at $\rho = \lambda_1(\Delta L)$ (the leading eigenvalue of ΔL) and its associated eigenvector $v = q_1(\Delta L)$, we estimate the leading eigenpairs of L_1 using the proxy

$$\widetilde{L}_1 = L_0 + \rho v v^T. \tag{3.3}$$



(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.

FIG. 2. *Adding a new sample point x_0 to the graph.*

An illustration of the out-of-sample extension problem is given in Figure 2.

3.2. Updating the graph Laplacian is almost a rank-one perturbation. As described in section 3.1, the weights on the edges of the graph are determined by a kernel. For our subsequent claims, we will require that our kernel is radial with $g(0) > 0$ and that in some neighborhood of 0 its derivative is bounded, that is, $|\frac{d}{dx}g| < M$ 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 the following analysis, we consider graphs constructed using δ-neighborhoods (see section 3.1). As we will see next, the analogue for kNN is straightforward. For δ-neighborhoods, we require the parameter δ to be "small enough," and more specifically, to satisfy $\delta < \frac{g(0)}{2M}$. This assumption is not too restrictive as the purpose of constructing similarity graphs is to model the local neighborhood relationships between the data points [24].

We denote by k the minimal 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 maximal number of neighbors of a vertex. (We assume that c_1 is independent of k.) Denote by $\sigma_i(X)$, $i = 1, \ldots, n$ the singular values of a squared symmetric matrix X (in descending order). We now present the main theoretical result of this section.

THEOREM 3.1. *Under the assumptions and notation described above, let L_0 and L_1 be two graph Laplacians before and after the addition of a new vertex, respectively. Then, there exists a constant β, independent of k, such that*

$$\sigma_1(L_1 - L_0) = 1 - \frac{\beta}{k} \qquad \text{and} \qquad \sigma_i(L_1 - L_0) = \frac{\beta}{k}, \quad i \geq 2.$$

Theorem 3.1 shows that for large enough k, $\sigma_1(L_1 - L_0) \approx 1$ and $\sigma_i(L_1 - L_0) \approx 0$, $i \geq 2$. In other words, $\Delta L = L_1 - L_0$ is indeed close to being rank-one.

3.3. Proof of Theorem 3.1. The proof is divided into a few steps. First, we adapt a classical result from perturbation theory called Weyl's theorem [22] to our setting for an initial bound on the singular values of ΔL. Then, we use our assumptions to derive, based on the specific structure of the graph Laplacian, the required constants and bounds to use in the main body of the proof.

From classical perturbation theory we have the following result regarding the singular values of a matrix.

THEOREM 3.2 (Weyl's theorem). *Let $S, E \in \mathbb{R}^{n\times n}$. Then, for all $1 \le i \le n$ we have*

$$\left|\sigma_i(S+E) - \sigma_i(S)\right| \le \|E\|_2 .$$

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

THEOREM 3.3. *Let $S \in \mathbb{R}^{n\times n}$ be a diagonal matrix whose diagonal entries are different from each other, and let $E \in \mathbb{R}^{n\times n}$. Assume, without loss of generality, that the diagonal entries of S are given in a descending order of magnitude. Denote $E = (e_{ij})$. Let $\eta > 0$ be such that $\|E\|_2 < \eta$. Then, for a small enough η, there exists $c_H = c_H(S,\eta) > 0$ independent of E so that for all $1 \le i \le n$,*

$$\left|\sigma_i(S+E) - \sigma_i(S) - \operatorname{sign}(S_{ii})e_{ii}\right| \le c_H \|E\|_F^2 .$$

The proof of Theorem 3.3 is given in Appendix A.1.

Recall that our aim is to bound the singular values of $\Delta L = L_1 - L_0$, that is, of the difference matrix between the graph Laplacians before and after the insertion of a new vertex. To apply Theorem 3.3, we denote $S = \operatorname{diag}(\Delta L)$ and $E = \Delta L - S$. Assume we permuted the indices of the vertices of the graph such that the diagonal entries of S are in descending order. Note that in our specific case, the diagonal entries of E are in fact zero. Therefore, by Theorem 3.3 there exists $c' > 0$ so that

$$\left|\sigma_i(\Delta L) - \sigma_i(S)\right| \le c' \|E\|_F^2 . \tag{3.4}$$

It is clear now that estimating $\|E\|_F$ will provide us the relation between the singular values of ΔL and S.

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

$$\operatorname{nnz}(\Delta L) \le (2c_1^2)k^2. \tag{3.5}$$

An elementwise 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 using δ-neighborhoods, using a radial kernel g with a bounded derivative $|\frac{d}{dx}g| < M$ such that $g(0) > 2M\delta$. Then,*

$$\frac{1}{c_1 c}\cdot\frac{1}{k} < \ell_{ij} < \frac{c}{k}, \quad 1 \le i,j \le n, \quad c = 1 + \frac{g(0)}{M\delta}. \tag{3.6}$$

Proof. Let $\alpha_{ij} = \|x_i - x_j\|$. Then, using Lagrange's remainder theorem, for each entry w_{ij} of the weight matrix W there exists ξ_{ij} such that

$$w_{ij} = g(\|x_i - x_j\|) = g(0) + \frac{d}{dx}g(\xi_{ij})\alpha_{ij}. \tag{3.7}$$

Since $\alpha_{ij} < \delta$ and $|\frac{d}{dx}g| < M$, we have that an upper bound on w_{ij} is $g(0) + M\delta$. On the other hand, $g(0) > 2M\delta$, so we get the bounds

$$M\delta < w_{ij} < g(0) + M\delta. \tag{3.8}$$

The ijth entry of the graph Laplacian is

$$\ell_{ij} = \frac{w_{ij}}{\sqrt{\sum_p w_{pj}} \cdot \sqrt{\sum_p w_{ip}}}, \tag{3.9}$$

where the two sums are taken over all the neighbors of the ith and jth vertices. The number of neighbors of each vertex is at least k so

$$\sum_p w_{pj} \geq kM\delta. \tag{3.10}$$

Therefore,

$$\ell_{ij} < \frac{g(0) + M\delta}{\sqrt{kM\delta}\sqrt{kM\delta}} = \frac{g(0) + M\delta}{M\delta k}. \tag{3.11}$$

Similarly, using the upper bound $c_1 k$ on the number of neighbors we get

$$\ell_{ij} > \frac{M\delta}{c_1(g(0) + M\delta)k}. \tag{3.12}$$

∎

An immediate conclusion from Lemma 3.4 is the following.

LEMMA 3.5. *The entries of ΔL are of order $O(\frac{1}{k})$ except for the first entry $(\Delta L)_{11}$, which is $-1 + O(\frac{1}{k})$.*

Proof. Denote by l^0_{ij} and l^1_{ij} the (i,j) entry of L_0 and L_1, respectively ($1 \leq i,j \leq n+1$). By Lemma 3.4, for $(i,j) \neq (1,1)$ both l^0_{ij} and l^1_{ij} are $O(\frac{1}{k})$, and thus the entries of ΔL, which are of the form $l^1_{ij} - l^0_{ij}$, are $O(\frac{1}{k})$. In the case $i = j = 1$, by construction $l^0_{11} = 1$ and thus $(\Delta L)_{11} = l^1_{11} - l^0_{11} = -1 + O(\frac{1}{k})$. ∎

It follows that ΔL is dominated by its first entry, and so it is somewhat unsurprising that it is close to being rank-one. A sharper elementwise bound is given in the following lemma.

LEMMA 3.6. *The entries of ΔL that are not on the first row/column are smaller in magnitude than $\frac{c^2}{2k^2}$, $c = 1 + \frac{g(0)}{M\delta}$.*

The proof of Lemma 3.6 is given in Appendix A.2.

According to (3.5), ΔL has at most $2c_1^2k^2$ nonzero elements. At most $2c_1k$ of those are on the first row and column. The magnitude of these elements is at most $\frac{c}{k}$. The rest of the nonzero elements, in light of Lemma 3.6, have a magnitude of at most $\frac{c^2}{2k^2}$.

Consider the nonzero elements of $E = \Delta L - \mathrm{diag}(\Delta L)$. The ones that are on the first row/column have a magnitude of at most $\frac{c}{k}$, and there are at most c_1k of them. Within the elements that are not on the first row/column, based on (3.5), there are at most $2c_1^2k^2$ nonzero elements, and by Lemma 3.6, their magnitude is at most $\frac{c^2}{2k^2}$ in size. Therefore, we can bound the Frobenius norm of E as

$$\|E\|_F^2 \leq 2c_1k \cdot \left(\frac{c}{k}\right)^2 + 2c_1^2k^2\left(\frac{c^2}{2k^2}\right)^2 = \frac{2c^2c_1}{k} + \frac{c^4c_1^2}{2k^2} < \frac{2c^4c_1^2}{k} + \frac{c^4c_1^2}{k} = \frac{3c^4c_1^2}{k}. \tag{3.13}$$

Namely,

$$\|E\|_F \leq \sqrt{3}c^2c_1\frac{1}{\sqrt{k}}. \tag{3.14}$$

We finally prove our main theorem.

Proof of Theorem 3.1. Recall that $S = \mathrm{diag}(\Delta L)$. Denoting $\tilde{c} = 3c^4c_1^2c'$, by (3.4),

$$|\sigma_i(\Delta L) - \sigma_i(S)| < c'\|E\|_F^2 < c' \cdot \left(\frac{\sqrt{3}c^2c_1}{\sqrt{k}}\right)^2 = \frac{3c^4c_1^2c'}{k} = \frac{\widetilde{c}}{k}, \quad 1 \le i \le n. \tag{3.15}$$

The largest singular value of S is the absolute value of its largest entry, and by Lemma 3.5 we have $\sigma_1(S) = |(\Delta L)_{11}|$ for large enough k. By Lemma 3.4 and (3.15),

$$\sigma_1(\Delta L) < \sigma_1(S) + \frac{\widetilde{c}}{k} < 1 - \frac{1}{c_1 ck} + \frac{\widetilde{c}}{k} = 1 - \frac{1 + \widetilde{c}c_1 c}{c_1 ck} \tag{3.16}$$

and

$$\sigma_1(\Delta L) > \sigma_1(S) - \frac{\widetilde{c}}{k} > 1 - \frac{c}{k} - \frac{\widetilde{c}}{k} = 1 - \frac{c + \widetilde{c}}{k}. \tag{3.17}$$

Namely, $\sigma_1(\Delta L)$ 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, by (3.15) we have

$$\sigma_i(\Delta L) < \sigma_i(S) + \frac{\widetilde{c}}{k} < \frac{c}{k} + \frac{\widetilde{c}}{k} = \frac{\widetilde{c} + c}{k}, \quad i \ge 2, \tag{3.18}$$

which shows that $\sigma_i(\Delta L)$ is of order $\frac{1}{k}$ as required. ∎

3.4. Rank-one update and error analysis. We next discuss the required adjustments for applying Algorithm 2.1 of rank-one update to the out-of-sample extension problem. We wish to find the best rank-one approximation of $\Delta L = L_1 - L_0$, which we denote by $(\Delta L)_1$. Such an approximation requires recovering the largest singular value of ΔL, denoted by $\sigma_1(\Delta L)$, and its corresponding left and right singular vectors, denoted by v_L and v_R, respectively.

Denote by (ρ, v) the top eigenpair of ΔL. Since $\Delta L \neq 0$ is symmetric, if its largest eigenvalue λ is positive, then $\sigma_1(\Delta L) = \lambda$ and $v_L = v_R = v$. If $\lambda < 0$, then $\lambda = -\sigma_1(\Delta L)$ and $v_L = -v_R = v$. Thus, in both cases, the best rank-one approximation of ΔL is

$$(\Delta L)_1 = \sigma_1(\Delta L) v_L v_R^T = \rho v v^T. \tag{3.19}$$

The out-of-sample extension algorithm, together with a perturbation correction that will be introduced shortly, is described in Algorithm 3.1. The approximated eigenpairs returned by the algorithm are affected by two types of error: the error induced by truncating the rank-one update equations, which was discussed in section 2, and the error induced by the rank-one approximation of ΔL, which we examine now.

To analyze the error of Algorithm 3.1 and to further improve our approximation for the updated eigenvalues and eigenvectors, we use two classical results from matrix perturbation theory. These results, Lemmas 3.7 and 3.8, are given without proofs, and 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 ith largest eigenvalue of X.

LEMMA 3.7. *Let $A, B \in \mathbb{R}^{n\times n}$ be symmetric matrices. The following holds for all $1 \le i \le n$:*

1. $|\lambda_i(A+B) - \lambda_i(A)| = O(\|B\|)$.
2. $\|q_i(A+B) - q_i(A)\| = O(\|B\|)$.

Using Lemma 3.7, let

$$A + B = L_1 = L_0 + \Delta L, \quad A = L_0 + (\Delta L)_1, \tag{3.20}$$

where $(\Delta L)_1$ is defined in (3.19). Then, the rank-one update (3.3) induces an error of order $\|\Delta L - (\Delta L)_1\| = \sigma_2(\Delta L)$, and by Theorem 3.1 we conclude that this error is of order

$$\sigma_2(\Delta L) = O\left(\frac{1}{k}\right).$$

Similarly to section 2, we can obtain higher order approximation using a further correction, based on the following result.

LEMMA 3.8. *Let $A, B \in \mathbb{R}^{n\times n}$ be symmetric matrices. The following holds for all $1 \le i \le n$:*

1. $|\lambda_i(A+B) - [\lambda_i(A) + q_i^T(A)Bq_i(A)]| = O(\|B\|^2)$.
2. $\|q_i(A+B) - [q_i(A) + \sum_{j\neq i} \frac{q_j(A)^T B q_i(A)}{\lambda_i(A)-\lambda_j(A)} q_j(A)]\| = O(\|B\|^2)$.

Lemma 3.8 gives rise to an improved error bound due to the extra correction term. Using (3.20), the rank-one update (3.3) followed by the correction term obtained by (3.8) induces an error of order $\|\Delta L - (\Delta L)_1\|^2 = \sigma_2^2(\Delta L)$, and by Theorem 3.1, we get that

$$\sigma_2^2(\Delta L) = O\left(\frac{1}{k^2}\right).$$

The perturbation correction is embedded in our method as described in Algorithm 3.1. The complexity of Algorithm 3.1 is discussed in detail on Appendix B.2.

Algorithm 3.1 Out-of-sample extension of the graph Laplacian.

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 $\left\{\left(\widehat{t_i}, \widehat{p_i}\right)\right\}_{i=1}^m$

$L_1 \leftarrow$ the graph Laplacian of $\mathcal{X} \cup \{x_0\}$
$\Delta L \leftarrow L_1 - L_0$
$\rho \leftarrow \lambda_1(\Delta L)$
$v \leftarrow q_1(\Delta L)$
Approximate $\left\{\left(\widetilde{t_i}, \widetilde{p_i}\right)\right\}_{i=1}^m$ using Algorithm 2.1 with input $\left(\{(\lambda_i, q_i)\}_{i=1}^m, \rho, v\right)$
$C \leftarrow L_1 - (L_0 + \rho v v^T)$
for all $i = 1 \dots m$ **do** // perturbation correction
  $\widehat{t_i} \leftarrow \widetilde{t_i} + \widetilde{p_i}^T C \widetilde{p_i}$
  $\widehat{p_i} \leftarrow \widetilde{p_i} + \sum_{j\neq i} \frac{\widetilde{p_j}^T C \widetilde{p_i}}{\widetilde{t_i} - \widetilde{t_j}} \widetilde{p_j}$
end for

4. Numerical examples. In this section, we provide various numerical examples to demonstrate empirically the theory developed in the previous sections. We use both synthetic datasets as well as real-world datasets. We begin by providing several numerical examples for the rank-one update formulas of section 2. These examples demonstrate the high accuracy of the methods, as well as their runtime efficiency. We

TABLE 1
Absolute errors for the synthetic example with $n = 1000, m = 10$, with unknown eigenvalues distributed normally with mean $\hat{\mu}$ and standard deviation $\sigma = 0.0001$.

	Eigenvalues			Eigenvectors			
$\hat{\mu}$	First order $\mu = 0$	Second order $\mu = 0$	First order Second order $\mu = \mu_*$	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

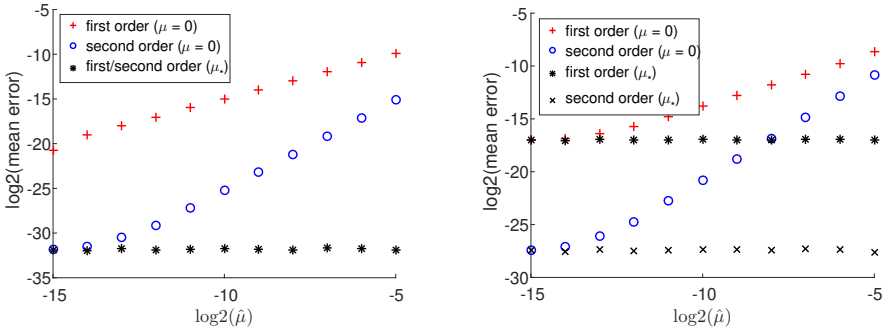
continue by providing numerical examples for section 3, showing numerically that inserting a new vertex to the graph Laplacian is almost a rank-one update to its matrix. We proceed by applying our algorithm for updating the eigenvalues and eigenvectors of the graph Laplacian to real-world data and measure the accuracy of our approach compared to other methods. All experiments were performed on an Intel i7 desktop with 8GB of RAM. All algorithms were implemented in MATLAB. The code to reproduce the examples is available at https://github.com/roymitz/rank-one-update.

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 the rank-one update problem. We generate a random symmetric matrix $A \in \mathbb{R}^{n \times n}$ with $n = 1000$ and $m = 10$ known leading eigenvalues whose magnitude is $O(1)$, together with their corresponding eigenvectors. The rest of the eigenvalues are unknown to the algorithm and are drawn from a normal distribution with mean $\hat{\mu}$ 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 $\hat{\mu}$. As a rough estimate for the unknown eigenvalues, our parameter μ is chosen to be either $\mu = 0$ or $\mu = \mu_*$ (2.22).

The results are shown in Table 1 and Figure 3. For the approximate eigenvalues, we measure the absolute errors of the first order method (2.5) and the second order method (2.19), for the two different choices of μ. Note that for $\mu = \mu_*$, the two estimations are identical and thus appear in the same column of the table. For the eigenvectors, the norm of the approximation error is presented, for the two different methods: first order of (2.27) and second order of (2.28) using the two different choices of μ.

According to section 2, for the above setting we expect the case $\mu = 0$ to yield errors of magnitude $O(\hat{\mu})$ for the first order approximations, and of magnitude $O(\hat{\mu}^2)$ for the second order approximations. For $\mu = \mu_*$ we expect errors independent of $\hat{\mu}$. This may be observed in Table 1, but is even clearer in Figure 3, where we have a line with zero slope for μ_* (error is independent on $\hat{\mu}$), a line with slope equal to one for $\mu = 0$ and first order approximation (linear error decay), and a line with slope equal to two for $\mu = 0$ and second order approximation (quadratic error decay).

The next example demonstrates the mean running time of Algorithm 2.1 for 10 independent runs, compared to the MATLAB function `eigs(L1, m)` for calculating the leading m eigenvalues and eigenvectors. The setting of the example is as follows. A symmetric sparse random matrix with $O(100 \cdot n)$ nonzero entries and a sparse random vector v with $O(100)$ entries were generated. We then used two variants of our algorithm to update the eigenpairs: first order approximation with $\mu = 0$ (fastest



(a) Eigenvalues absolute error (b) Eigenvectors absolute error

FIG. 3. *Plot of* $\log_2$*-absolute error as a function of* $\log_2 \hat{\mu}$ *for the synthetic example with* $n = 1000, m = 10$*, with unknown eigenvalues normally distributed with mean* $\hat{\mu}$ *and standard deviation* $\sigma = 0.0001$*. We can notice the three main trends: the error when using* μ_* *is independent of* $\hat{\mu}$*, the error when using* $\mu = 0$ *and the first order approximation decay linearly, and the error when using* $\mu = 0$ *and the second order approximation decay quadratically.*

TABLE 2
Performance measurements.

n	MATLAB	First order $\mu = 0$	Second order $\mu = \mu_*$
2000	0.47 ± 0.05	0.75 ± 0.02	0.85 ± 0.02
4000	1.71 ± 0.10	0.79 ± 0.02	0.92 ± 0.02
8000	5.34 ± 0.05	0.76 ± 0.03	0.88 ± 0.02
16000	17.1 ± 0.12	0.86 ± 0.03	0.97 ± 0.02
32000	50.6 ± 0.57	0.97 ± 0.03	1.11 ± 0.02
64000	154 ± 1.01	1.23 ± 0.01	1.36 ± 0.01

(a) Running time is seconds for varying n with a fixed number of eigenpairs, $m = 10$.

m	MATLAB	First order $\mu = 0$	Second order $\mu = \mu_*$
50	12.4 ± 0.13	0.45 ± 0.01	0.51 ± 0.01
100	26.5 ± 0.34	0.95 ± 0.01	1.08 ± 0.01
200	60.5 ± 0.33	1.98 ± 0.04	2.21 ± 0.04
400	155 ± 4.05	4.74 ± 0.20	5.08 ± 0.42
600	345 ± 4.05	7.14 ± 0.20	8.25 ± 0.42
800	542 ± 4.05	10.9 ± 0.20	11.7 ± 0.42

(b) Running time is seconds for a varying number of eigenpairs m with a fixed matrix size $n = 20{,}000$.

variant) and second order approximation with $\mu = \mu_*$ (slowest variant). Table 2 demonstrates the dependence of the running time on n and m. While the MATLAB algorithm is accurate and ours is only an approximate, we can see that for relatively small values of n our algorithm is more than an order of magnitude faster. Due to the linear dependence on n, we can expect this difference to be even more dramatic for larger values of n, as witnessed in Figure 4. Additionally, the runtime differences between the two variants of our algorithm are negligible.

4.2. Updating the graph Laplacian (section 3). We provide several examples using three real-world datasets to demonstrate the update of the symmetric graph Laplacian. The datasets are described in Table 3.

In the first example, we demonstrate that inserting a new vertex to the graph Laplacian is almost rank-one, as suggested by Theorem 3.1. In this example, for each dataset, we first randomly select a subset of it and construct the symmetric graph Laplacian L_0 of the selected subset, leaving the first vertex out. Then, we connect this vertex to the graph, which results in a new graph Laplacian L_1. Finally we compute the first and second singular values of $\Delta L = L_1 - L_0$. We repeat this experiment 10 times, each time with a different random subset of the data. The results of the mean magnitude of the singular values are shown in Table 4 for various datasets and values

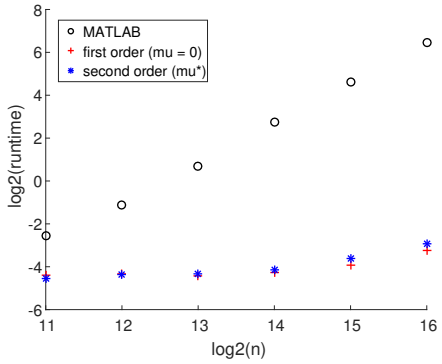


FIG. 4. *Plot of* $\log_2$*-runtime as a function of* $\log_2 n$ *for matrices of size* n*. We can see that the runtime difference between our algorithms and* `eigs` *of MATLAB increases with* n*.*

TABLE 3
Real-world datasets.

Name	Samples	Attributes	Description
MNIST	60,000	784	Grey scale images of handwritten digits between 0 and 9
Poker Hand	25,000	10	Each record is a hand consisting of five playing cards drawn from a standard deck of 52 cards
Yeast	1484	8	Information about a set of yeast cells

TABLE 4
The two largest singular values of ΔL *for real-world datasets and three different values of* k*. As theory predicts, there is a two orders of magnitude difference between the first and second singular values, indicating that indeed* ΔL *is close to being rank-one.*

	$k=5$		$k=10$		$k=20$	
Dataset	σ_1	σ_2	σ_1	σ_2	σ_1	σ_2
MNIST (5K samples)	0.91	0.09	0.97	0.05	0.99	0.02
Poker (10K samples)	0.94	0.09	0.98	0.05	0.98	0.02
Yeast (1.5K samples)	0.94	0.11	0.97	0.05	0.99	0.03

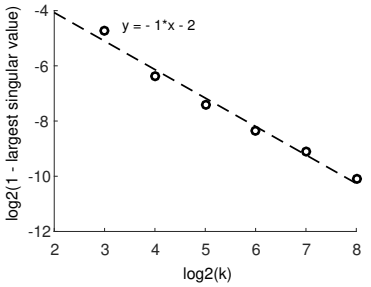
of k. Clearly, one can observe, as predicted by the theory, that the first singular value is very close to 1, while the second singular value is close to 0.

Next, we demonstrate empirically the dependence of the singular values on k. Specifically, Theorem 3.1 implies that up to a constant

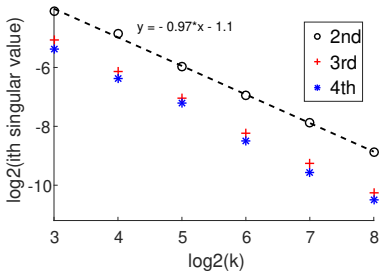
$$\log(\sigma_1(\Delta L) - 1) = \log(\sigma_i(\Delta L)) = -k, \quad 2 \le i \le n.$$

Thus the log of the singular values is expected to be linear in k with a slope that equals to 1. This is demonstrated for the poker dataset in Figure 5. Similar results were obtained for the other datasets as well.

In the second example, we perform out-of-sample extension using several methods and compare their accuracy. As a benchmark for the eigenvectors extension, we use the Nyström method [3], which is a widely used method for this task. Additionally, we use the naive approach of merely having the old eigenvalues and eigenvectors as approximations to the new ones. Regarding our methods, we use both the first order ((2.5), (2.27)) and second order ((2.19), (2.28)) approximations described in section 2. For our methods, we also apply the perturbation correction described in



(a) $1-\sigma_1(\Delta L) = O(\frac{1}{k})$. The dependence of $\log_2(1-\sigma_1)$ on k is linear with slope (-1).



(b) $\sigma_i(\Delta L) = O(\frac{1}{k})$ for $i = 2, 3, 4$. The dependence of $\log_2(\sigma_i)$ on k is linear with slope (-1).

FIG. 5. *Demonstration of Theorem* 3.1 *for the poker dataset with* $n = 10{,}000$.

TABLE 5

Error comparison for three datasets: MNIST $(n = 1000, m = 5, k = 10, \varepsilon = 100)$, *poker* $(n = 3000, m = 5, k = 100, \varepsilon = 100)$, *and yeast* $(n = 1400, m = 5, k = 100, \varepsilon = 100)$. *Best performance is marked in bold.*

MNIST	Eigenvalues	Eigenvalues (after correction)	Eigenvectors	Eigenvectors (after correction)
No update	7.85e-05	-	2.83°	-
Nyström	-	-	1.56°	-
First order ($\mu = 0$)	5.09e-05	7.73e-06	1.00°	0.84°
Second order ($\mu = \mu_*$)	**5.06e-05**	**7.70e-06**	**1.00°**	**0.82°**

Poker	Eigenvalues	Eigenvalues (after correction)	Eigenvectors	Eigenvectors (after correction)
No update	1.97e-05	-	3.04°	-
Nyström	-	-	2.71°	-
First order ($\mu = 0$)	1.44e-05	6.18e-06	1.89°	0.95°
Second order ($\mu = \mu_*$)	**1.43se-05**	**6.15e-06**	**1.88°**	**0.94°**

Yeast	Eigenvalues	Eigenvalues (after correction)	Eigenvectors	Eigenvectors (after correction)
No update	1.58e-04	-	2.40°	-
Nyström	-	-	0.65°	-
First order ($\mu = 0$)	1.11e-04	1.78e-06	0.42°	0.35°
Second order ($\mu = \mu_*$)	**1.11e-04**	**1.78e-06**	**0.41°**	**0.33°**

Algorithm 3.1. To compare the performance of the different algorithms, we measure the angles between the true eigenvectors and their approximations and report the maximal angle out of the m angles calculated. The results reported are the mean error of 10 independent experiments, that is, picking randomly a vertex for the out-of-sample extension in each experiment. The full comparison between the described methods is given in Table 5, where for each dataset we also mention the parameter ε of the width of the Gaussian (see (3.2)) that we used for constructing the graph Laplacian. In the second column of Table 5 is the absolute error of the eigenvalues for each method except the Nyström method which we use only to extend the eigenvectors. In the third column is the absolute error of the eigenvalues after performing the perturbation correction. The fifth and sixth columns present the error of the eigenvectors estimation, before and after perturbation correction, respectively.

TABLE 6
Accuracy for out-of-sample extension of the MNIST dataset for several methods.

Method	Accuracy
Recalculation (Optimal)	68%
No update	12%
Nyström	58%
Our method	67%

We can see that our methods outperform the other approaches. As expected, the second order approximations using μ_* present the best performance and are marked in bold.

In the last example, we demonstrate a practical rather than a numerical advantage of our method. Starting with 1500 random samples from the MNIST dataset, we split this set into a train set consisting of 1000 samples and a test set consisting of the remaining 500 samples. Each point in the train set is in $\mathbb{R}^{784}$. We then embed the train set samples in $\mathbb{R}^{10}$ using Laplacian eigenmaps [1] with parameters $k = 10$ and $\varepsilon = 100$ for constructing the graph Laplacian. For each sample in the test set, we perform an out-of-sample-extension using four different methods: recalculation of the new embedding (which is the optimal, expensive method), no update where the test points are embedded naively to the origin, Nyström method, and our method. We then train a 15-NN classifier on the embedded vectors of the train set. We use this classifier to label the given test sample and compare it to the true label. Table 6 summarizes the accuracy of each extension method on the test set. One can see that our method performs considerably better than the other approaches, and its performance is very close to the best possible results obtained by the method of recalculating the entire embedding.

5. Conclusions. In this paper, we proposed an approximation algorithm for the rank-one update of a symmetric matrix when only part of its spectrum is known. We provided error bounds for our algorithm and showed that they are independent of the number of unknown eigenvalues. As implied both by theory and numerical examples, our algorithm performs best when the unknown eigenvalues are clustered (i.e., close to each other). On the other hand, numerical evidence shows that when the unknown eigenvalues are not clustered, the results may deteriorate but are still no worse than neglecting the unknown eigenvalues (low rank approximation). As a possible application, we proposed the out-of-sample extension of the graph Laplacian matrix and demonstrated that our method provides superior results.

Appendix A. Complementary materials.

A.1. Proof of Theorem 3.3.

Proof. Denote by $\mathrm{vec}(X)$ the $n \times n$ matrix X rearranged as a vector in $\mathbb{R}^{n^2}$. By assumption, the diagonal entries of S are different from each other, and thus the singular values of S are distinct. When the singular values of a matrix are distinct, they are analytic functions of its entries in some compact η-neighborhood S^* of $\mathrm{vec}(S)$ and can be expanded in a Taylor expansion [16]. Let E be a matrix so that $\mathrm{vec}(S+E) \in S^*$, i.e., $\|\mathrm{vec}(S+E) - \mathrm{vec}(S)\| = \|\mathrm{vec}(E)\| < \eta$. We use the result in [19], stating that if $A = UDV^T$ is the SVD of A, then $\frac{\partial \sigma_k(A)}{\partial a_{ij}} = u_{ik}v_{jk}$. Expanding

the singular value function to a first order Taylor polynomial yields

$$\begin{aligned}\sigma_i(S+E) &= \sigma_i(S) + \nabla\sigma_i(S)\cdot \operatorname{vec}(E) + R \\ &= \sigma_i(S) + \sum_{n,m} \frac{\partial \sigma_i(S)}{\partial s_{nm}} \cdot e_{nm} + R = \sigma_i(S) + \sum_{n,m} u_{ni} v_{mi} \cdot e_{nm} + R\end{aligned} \tag{A.1}$$

with u_i and v_i being the left and right singular vectors of S, respectively. Denote by $|X|$ the matrix whose entries are the absolute values of the entries of the matrix X and by $P_S(X)$ the matrix resulting by flipping the sign of the entries of column i in the matrix X by the sign of S_{ii} for all $1 \le i \le n$. Then, the SVD of the matrix S is $S = I|S|P_S(I)^T$, where I is the identity matrix. Equation (A.1) is then reduced to

$$\sigma_i(S+E) = \sigma_i(S) + u_{ii}v_{ii}e_{ii} + R = \sigma_i(S) + \operatorname{sign}(S_{ii})e_{ii} + R. \tag{A.2}$$

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

$$|R| = \frac{1}{2}|\operatorname{vec}(E)^T H(\sigma_i(Z))\operatorname{vec}(E)| \le \frac{1}{2}\big\|\operatorname{vec}(E)\big\|\big\|H(\sigma_i(Z))\big\|_2\big\|\operatorname{vec}(E)\big\|. \tag{A.3}$$

The entries of $H(\sigma_i(X))$ are the second order directional derivatives of $\sigma_i(X)$. Since $\sigma_i(X)$ is an analytic function in S^*, its second order derivatives at any direction are continuous functions. Since the $\|\cdot\|_2$ norm is a continuous function of the matrix entries, we conclude that $\|H(\sigma_i(X))\|_2$ is a continuous function in S^*. By the compactness of S^* and the boundedness theorem, there exists $c_H > 0$ such that $\|H(\sigma_i(X))\|_2 < c_H$ for all $X \in S^*$. Finally we conclude that

$$|R| \le \frac{1}{2} c_H \|E\|_F^2. \tag{A.4}$$

∎

A.2. Proof of Lemma 3.6.

Proof. Let w_{pq} be the weight on the edge connecting vertices p and q in the graph. We examine the entry $(\Delta L)_{ij}$. If neither of the vertices i nor j were connected to the new vertex, then $\Delta L_{ij} = 0$. Otherwise, assume that the new vertex was connected to vertex i but was not connected to vertex j. Denote by $w_{i1} = w_{1i}$ the weight on the edge connecting the new vertex and vertex i. Let $\alpha = \sum_p w_{pj}$ (the sum of the jth column of L_0) and $\beta = \sum_p w_{ip}$ (the sum of the ith row of L_0). It follows that the ith row of L_1 is normalized by $\sqrt{w_{i1}+\beta}$. Denote by l^0_{ij} and l^1_{ij} the (i,j) entry of L_0 and L_1, respectively. Thus,

$$\begin{aligned}|(\Delta L)_{ij}| = \left|\ell^0_{ij} - \ell^1_{ij}\right| &= \frac{w_{ij}}{\sqrt{\alpha}\cdot\sqrt{\beta}} - \frac{w_{ij}}{\sqrt{\alpha}\cdot\sqrt{w_{i1}+\beta}} \\ &= \frac{w_{ij}}{\sqrt{\alpha}}\cdot\left(\frac{1}{\sqrt{\beta}} - \frac{1}{\sqrt{w_{i1}+\beta}}\right) \\ &= \frac{w_{ij}}{\sqrt{\alpha}}\cdot\left(\frac{\sqrt{w_{i1}+\beta}-\sqrt{\beta}}{\sqrt{\beta}\cdot\sqrt{w_{i1}+\beta}}\right) = \frac{w_{ij}}{\sqrt{\alpha}}\cdot\frac{w_{i1}}{\big(w_{i1}+\beta\big)\sqrt{\beta}+\beta\sqrt{w_{i1}+\beta}},\end{aligned} \tag{A.5}$$

where (A.5) is derived by multiplying by $\frac{\sqrt{w_{i1}+\beta}+\sqrt{\beta}}{\sqrt{w_{i1}+\beta}+\sqrt{\beta}}$. Using the bounds on the graph Laplacian entries in (3.8) and by (3.10), we get that $\alpha, \beta > kM\delta$ and finally

(A.6)

$$|(\Delta L)_{ij}| < \frac{w_{ij}}{\sqrt{kM\delta}} \cdot \left(\frac{w_{i1}}{(M\delta + kM\delta)\sqrt{kM\delta} + kM\delta\sqrt{M\delta + kM\delta}} \right)$$
$$< \frac{g(0) + M\delta}{\sqrt{kM\delta}} \cdot \left(\frac{g(0) + M\delta}{kM\delta\sqrt{kM\delta} + kM\delta\sqrt{kM\delta}} \right) = \frac{(g(0) + M\delta)^2}{2(M\delta)^2 k^2} = \frac{c^2}{2k^2}.$$

The remaining cases, i.e., when both vertex i and vertex j were connected to the new vertex and where only vertex j was connected to the new vertex, are analogous. ∎

Appendix B. Complexity analysis of the algorithms.

B.1. Analysis of Algorithm 2.1. In line 1 of Algorithm 2.1 we determine the parameter μ defined in section 2.2. For $\mu = 0$ no calculation is needed. For μ_{mean}, we need to sum the diagonal elements of an $n \times n$ matrix which costs $O(n)$ operations. The calculation of μ_* requires the calculation of s of (2.18) which costs $O(mn + \text{nnz}(A))$ operations. Note that this calculation is required for the second order approximation of the secular equation (2.19).

Line 2 requires solving a variant of the truncated secular equation, namely, (2.5) or (2.19), and is done by standard solvers such as Newton's method. Indeed, if we consider Newton's method, each iteration consists of summing up to $O(m)$ terms, thus requiring in $O(m)$ operations. We also expect this method to converge in $O(1)$ iterations, as Newton's method has a quadratic convergence rate, resulting in $O(m)$ operations for one eigenvalue. The calculation for all the top m eigenvalues of $A + \rho vv^T$ will thus require $O(m^2)$ operations. For the second order approximation, there is an additional calculation of s which is done only once and costs (as already mentioned above) $O(mn + \text{nnz}(A))$ operations. The calculation of $z = Q_m^T v$ is also done only once so we get a total complexity of $O(m^2 + mn)$ operations for the first order approximation, and an additional $O(mn + \text{nnz}(A))$ operations for the second order one.

Line 4 evaluates the eigenvectors. We first discuss the evaluation of (2.24). Again, $z = Q_m^T v$ needs to be calculated only once for all the eigenvectors and costs $O(mn)$ operations to compute. The rest of the formula involves a multiplication of an $n \times m$ dense matrix by an $m \times m$ diagonal matrix which costs $O(mn)$ operations. Finally, a product of an $n \times m$ matrix with a vector of length m costs $O(nm)$ operations. Normalizing the resulting vector requires $O(n)$ operations. Thus, it costs $O(mn)$ operations to compute one eigenvector and $O(mn + m^2 n)$ operations to compute all m eigenvectors.

In (2.27) and (2.28) we add the calculation of r which requires $O(mn)$ operations and can be done only once for all eigenvectors. Therefore, asymptotically, formula (2.27) has the same complexity as that of (2.24). Last, for the second order approximation (2.28), we add the calculation of Ar which again can be done only once and costs $O(\text{nnz}(A))$ operations. Thus the calculation of one eigenvector requires $O(\text{nnz}(A) + mn)$ operations and the calculation of all m eigenvectors requires $O(\text{nnz}(A) + m^2 n)$ operations.

B.2. Analysis of Algorithm 3.1. Line 1 of Algorithm 3.1 constructs the graph Laplacian L_1. This step involves finding the vertices in the δ-neighborhood of the

TABLE 7
Summary of the complexity analysis.

	Calculation	Complexity
μ	0	$O(1)$
	μ_*	$O(mn + \operatorname{nnz}(A))$
	μ_{mean}	$O(n)$
Eigenvalues	First order	$O(m^2)$
	Second order	$O(mn + m^2 + \operatorname{nnz}(A))$
	Perturbation correction	$O(mn + m\operatorname{nnz}(C))$
Eigenvectors	First order	$O(m^2 n)$
	Second order	$O(m^2 n + \operatorname{nnz}(A)$
	Perturbation correction	$O(m^2 n + m^2\operatorname{nnz}(C))$

new vertex, which requires $O(n)$ operations, followed by updating $O(k^2)$ entries. This gives a total of $O(n + k^2)$ operations for this line, which is independent of the extension scheme.

Calculating the rank-one update in lines 3–4 requires applying the power method [13, Chapter 8] to ΔL, which is a sparse $n \times n$ matrix with a large spectral gap (Theorem 3.1). We can thus expect the power method to converge within a few iterations. The complexity of applying the rank-one update in line 5 depends on the method used and costs at most $O(m^2 n + \operatorname{nnz}(L_0))$. The calculation of C in 6 costs $\max\left\{\operatorname{nnz}(L_0), \operatorname{nnz}(L_1), \operatorname{nnz}(v)^2\right\}$.

We now analyze the perturbation correction in lines 7–10. For one eigenvalue, the correction consists of a multiplication of the form $q^T C q = q^T(Cq)$, which requires $O(\operatorname{nnz}(C))$ operations for Cq and $O(n)$ operations for $q^T(Cq)$. Thus, the total number of arithmetic operations for one eigenvalue is $O(n + \operatorname{nnz}(C))$, and for correcting all the eigenvalues is $O(mn + m\operatorname{nnz}(C))$.

For one eigenvector, we sum up m elements of the form $\frac{q^T C q}{\lambda} q$. Based on the analysis above, each element requires $O(n + \operatorname{nnz}(C))$ operations, resulting in $O(mn + m\operatorname{nnz}(C))$ operations for one eigenvector and $O(m^2 n + m^2\operatorname{nnz}(C))$ operations for all eigenvectors.

The complete complexity analysis is summarized in Table 7.

Appendix C. Loss of orthogonality. The approximate formulas (2.27) and (2.28) will generally not produce an orthogonal set of vectors. Therefore, in its current form, our method may not be applied more than once. In that case, a reorthogonalization procedure may be used. Examples of such procedures are the QR and polar decompositions. We wish to analyze the quality of the approximation in (2.27) and (2.28) after some reorthogonalizing procedure has been applied. Denote $B = A + \rho v v^T$ (see (2.1)), and let $\widetilde{P} = [\widetilde{p}_1 \widetilde{p}_2 \cdots \widetilde{p}_m]$ be the matrix whose columns are the approximated eigenvectors of formulas (2.27) or (2.28) and $\widetilde{T} = \operatorname{diag}(\widetilde{t}_1, \ldots, \widetilde{t}_m)$ the diagonal matrix whose diagonal is the approximated eigenvalues obtained by solving (2.5) or (2.19). As a measure of the quality of the eigenpairs approximations we will use

$$\left\| B\widetilde{P} - \widetilde{P}\widetilde{T} \right\|_F . \tag{C.1}$$

Let $\bar{P} \in \mathbb{R}^{n \times m}$ be the result of orthogonalizing the columns of $\widetilde{P}$ by any method so that

$$\widetilde{P} = \bar{P}(I + E) \tag{C.2}$$

for some $E \in \mathbb{R}^{m\times m}$, and let $G \in \mathbb{R}^{m\times m}$ so that

$$I + G = \widetilde{P}^T \widetilde{P}, \tag{C.3}$$

i.e., G determines how much the approximated eigenvectors deviate from being orthonormal. Then we have

$$I + G = (I + E)^T (I + E). \tag{C.4}$$

By perturbation bound of the Cholesky decomposition [8, Theorem 9] we obtain that if

$$\|G\|_F < \frac{1}{4}, \tag{C.5}$$

then

$$\|E\|_F < 2\|G\|_F. \tag{C.6}$$

Inequality (C.6) enables us to bound the affect of reorthogonalization on the quality of the approximation. By (C.2), we have that

$$B\widetilde{P} = B\bar{P}(I + E) = B\bar{P} + B\bar{P}E \tag{C.7}$$

and

$$\widetilde{P}\widetilde{T} = \bar{P}(I + E)\widetilde{T} = \bar{P}\widetilde{T} + \bar{P}E\widetilde{T}. \tag{C.8}$$

Subtracting (C.8) from (C.7), taking the Frobenius norm, and applying the triangle inequality and the subadditivity of the Frobenius norm, we have that under condition (C.5)

$$\begin{aligned} \left\|B\bar{P} - \bar{P}\widetilde{T}\right\|_F &\le \left\|B\widetilde{P} - \widetilde{P}\widetilde{T}\right\|_F + \|B\|_F \left\|\bar{P}\right\|_F \|E\|_F + \left\|\widetilde{T}\right\|_F \left\|\bar{P}\right\|_F \|E\|_F \\ &\le \left\|B\widetilde{P} - \widetilde{P}\widetilde{T}\right\|_F + 2\|G\|_F \left\|\bar{P}\right\|_F \left(\|B\|_F + \left\|\widetilde{T}\right\|_F\right). \end{aligned} \tag{C.9}$$

Thus the reorthogonalization process may introduce an error bounded by $2\|G\|_F\|\bar{P}\|_F(\|B\|_F + \|\widetilde{T}\|_F)$ to the error introduced by approximation formulas (2.27) and (2.28).

REFERENCES

[1] M. Belkin and P. Niyogi, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Comput., 15 (2003), pp. 1373–1396.

[2] C. M. Bender and S. A. Orszag, *Advanced Mathematical Methods for Scientists and Engineers* I: *Asymptotic Methods and Perturbation Theory*, Springer, New York, 2013.

[3] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N. L. Roux, and M. Ouimet, *Out-of-sample extensions for LLE, Isomap, MDS, eigenmaps, and spectral clustering*, in Advances in Neural Information Processing Systems, MIT Press, Cambridge, MA, 2004, pp. 177–184.

[4] M. Brand, *Fast low-rank modifications of the thin singular value decomposition*, Linear Algebra Appl., 415 (2006), pp. 20–30.

[5] A. Buja, D. F. Swayne, M. L. Littman, N. Dean, H. Hofmann, and L. Chen, *Data visualization with multidimensional scaling*, J. Comput. Graph. Statist., 17 (2008), pp. 444–472.

[6] J. R. Bunch, C. P. Nielsen, and D. C. Sorensen, *Rank-one modification of the symmetric eigenproblem*, Numer. Math., 31 (1978), pp. 31–48.

[7] F. W. Byron and R. W. Fuller, *Mathematics of Classical and Quantum Physics*, Courier Corporation, North Chelmsford, MA, 2012.
[8] X.-W. Chang, C. C. Paige, and G. Stewart, *New perturbation analyses for the Cholesky factorization*, IMA J. Numer. Anal., 16 (1996), pp. 457–484.
[9] F. R. Chung, *Spectral Graph Theory*, CBMS Reg. Conf. Ser. Math. 92, American Mathematical Society, Providence, RI, 1997.
[10] R. R. Coifman and S. Lafon, *Diffusion maps*, Appl. Comput. Harmonic Anal., 21 (2006), pp. 5–30.
[11] R. R. Coifman, Y. Shkolnisky, F. J. Sigworth, and A. Singer, *Graph Laplacian tomography from unknown random projections*, IEEE Trans. Image Process., 17 (2008), pp. 1891–1899.
[12] J. Ding and A. Zhou, *Eigenvalues of rank-one updated matrices with some applications*, Appl. Math. Lett., 20 (2007), pp. 1223–1226.
[13] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Vol. 3, The John Hopkins University Press, Baltimore, MD, 2012.
[14] A. N. Langville and C. D. Meyer, *Updating Markov chains with an eye on Google's PageRank*, SIAM J. Matrix Anal. Appl., 27 (2006), pp. 968–987.
[15] A. N. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press, Princeton, NJ, 2011.
[16] J. R. Magnus, *On differentiating eigenvalues and eigenvectors*, Econometric Theory, 1 (1985), pp. 179–191.
[17] A. Y. Ng, M. I. Jordan, and Y. Weiss, *On spectral clustering: Analysis and an algorithm*, Adv. Neural Inf. Process. Systems, 2 (2002), pp. 849–856.
[18] L. Page, S. Brin, R. Motwani, and T. Winograd, *The PageRank Citation Ranking: Bringing Order to the Web.*, Technical report, Stanford InfoLab, 1999.
[19] T. Papadopoulo and M. I. Lourakis, *Estimating the Jacobian of the singular value decomposition: Theory and applications*, in Proceedings of the European Conference on Computer Vision, Springer, New York, 2000, pp. 554–570.
[20] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science, 290 (2000), pp. 2323–2326.
[21] J. Shlens, *A Tutorial on Principal Component Analysis*, preprint, arXiv:1404.1100, 2014.
[22] G. W. Stewart, *Perturbation Theory for the Singular Value Decomposition*, Technical report, 1998.
[23] G. W. Stewart and J.-g. Sun, *Matrix perturbation theory*, Academic Press, New York, 1990.
[24] U. Von Luxburg, *A tutorial on spectral clustering*, Statist. Comput., 17 (2007), pp. 395–416.