Regularized Laplacian Estimation and Fast Eigenvector Approximation

Patrick O. Perry
Information, Operations, and Management Sciences
NYU Stern School of Business
New York, NY 10012
pperry@stern.nyu.edu

Michael W. Mahoney
Department of Mathematics
Stanford University
Stanford, CA 94305
mmahoney@cs.stanford.edu

Abstract

Recently, Mahoney and Orecchia demonstrated that popular diffusion-based procedures to compute a quick *approximation* to the first nontrivial eigenvector of a data graph Laplacian *exactly* solve certain regularized Semi-Definite Programs (SDPs). In this paper, we extend that result by providing a statistical interpretation of their approximation procedure. Our interpretation will be analogous to the manner in which ℓ_2-regularized or ℓ_1-regularized ℓ_2-regression (often called Ridge regression and Lasso regression, respectively) can be interpreted in terms of a Gaussian prior or a Laplace prior, respectively, on the coefficient vector of the regression problem. Our framework will imply that the solutions to the Mahoney-Orecchia regularized SDP can be interpreted as regularized estimates of the pseudoinverse of the graph Laplacian. Conversely, it will imply that the solution to this regularized estimation problem can be computed very quickly by running, *e.g.*, the fast diffusion-based PageRank procedure for computing an approximation to the first nontrivial eigenvector of the graph Laplacian. Empirical results are also provided to illustrate the manner in which approximate eigenvector computation *implicitly* performs statistical regularization, relative to running the corresponding exact algorithm.

1 Introduction

Approximation algorithms and heuristic approximations are commonly used to speed up the running time of algorithms in machine learning and data analysis. In some cases, the outputs of these approximate procedures are "better" than the output of the more expensive exact algorithms, in the sense that they lead to more robust results or more useful results for the downstream practitioner. Recently, Mahoney and Orecchia formalized these ideas in the context of computing the first nontrivial eigenvector of a graph Laplacian [1]. Recall that, given a graph G on n nodes or equivalently its $n \times n$ Laplacian matrix L, the top nontrivial eigenvector of the Laplacian *exactly* optimizes the Rayleigh quotient, subject to the usual constraints. This optimization problem can equivalently be expressed as a vector optimization program with the objective function $f(x) = x^T Lx$, where x is an n-dimensional vector, or as a Semi-Definite Program (SDP) with objective function $F(X) = \mathrm{Tr}(LX)$, where X is an $n \times n$ symmetric positive semi-definite matrix. This first nontrivial vector is, of course, of widespread interest in applications due to its usefulness for graph partitioning, image segmentation, data clustering, semi-supervised learning, etc. [2, 3, 4, 5, 6, 7].

In this context, Mahoney and Orecchia asked the question: do popular diffusion-based procedures—such as running the Heat Kernel or performing a Lazy Random Walk or computing the PageRank function—to compute a quick *approximation* to the first nontrivial eigenvector of L solve some other regularized version of the Rayleigh quotient objective function *exactly*? Understanding this algorithmic-statistical tradeoff is clearly of interest if one is interested in very large-scale applications, where performing statistical analysis to derive an objective and then calling a black box solver to optimize that objective exactly might be too expensive. Mahoney and Orecchia answered the above question in the affirmative, with the interesting twist that the regularization is on the SDP formulation rather than the usual vector optimization problem. That is, these three diffusion-based procedures exactly optimize a regularized SDP with objective function $F(X) + \frac{1}{\eta}G(X)$, for some regularization function $G(\cdot)$ to be described below, subject to the usual constraints.

In this paper, we extend the Mahoney-Orecchia result by providing a statistical interpretation of their approximation procedure. Our interpretation will be analogous to the manner in which ℓ_2-regularized or ℓ_1-regularized ℓ_2-regression (often called Ridge regression and Lasso regression, respectively) can be interpreted in terms of a Gaussian prior or a Laplace prior, respectively, on the coefficient vector of the regression problem. In more detail, we will set up a sampling model, whereby the graph Laplacian is interpreted as an observation from a random process; we will posit the existence of a "population Laplacian" driving the random process; and we will then define an estimation problem: find the inverse of the population Laplacian. We will show that the maximum a posteriori probability (MAP) estimate of the inverse of the population Laplacian leads to a regularized SDP, where the objective function $F(X) = \mathrm{Tr}(LX)$ and where the role of the penalty function $G(\cdot)$ is to encode prior assumptions about the population Laplacian. In addition, we will show that when $G(\cdot)$ is the log-determinant function then the MAP estimate leads to the Mahoney-Orecchia regularized SDP corresponding to running the PageRank heuristic. Said another way, the solutions to the Mahoney-Orecchia regularized SDP can be interpreted as regularized estimates of the pseudoinverse of the graph Laplacian. Moreover, by Mahoney and Orecchia's main result, the solution to this regularized SDP can be computed very quickly—rather than solving the SDP with a black-box solver and rather computing explicitly the pseudoinverse of the Laplacian, one can simply run the fast diffusion-based PageRank heuristic for computing an approximation to the first nontrivial eigenvector of the Laplacian L.

The next section describes some background. Section 3 then describes a statistical framework for graph estimation; and Section 4 describes prior assumptions that can be made on the population Laplacian. These two sections will shed light on the computational implications associated with these prior assumptions; but more importantly they will shed light on the implicit prior assumptions associated with making certain decisions to speed up computations. Then, Section 5 will provide an empirical evaluation, and Section 6 will provide a brief conclusion. Additional discussion is available in the Appendix of the technical report version of this paper [8].

2 Background on Laplacians and diffusion-based procedures

A weighted symmetric graph G is defined by a vertex set $V = \{1, \dots, n\}$, an edge set $E \subset V \times V$, and a weight function $w : E \to \mathbb{R}_+$, where w is assumed to be symmetric (*i.e.*, $w(u, v) = w(v, u)$). In this case, one can construct a matrix, $L_0 \in \mathbb{R}^{V \times V}$, called the combinatorial Laplacian of G:

$$L_0(u, v) = \begin{cases} -w(u, v) & \text{when } u \neq v, \\ d(u) - w(u, u) & \text{otherwise,} \end{cases}$$

where $d(u) = \sum_v w(u, v)$ is called the degree of u. By construction, L_0 is positive semidefinite. Note that the all-ones vector, often denoted 1, is an eigenvector of L_0 with eigenvalue zero, *i.e.*, $L1 = 0$. For this reason, 1 is often called trivial eigenvector of L_0. Letting D be a diagonal matrix with $D(u, u) = d(u)$, one can also define a normalized version of the Laplacian: $L = D^{-1/2} L_0 D^{-1/2}$. Unless explicitly stated otherwise, when we refer to the Laplacian of a graph, we will mean the normalized Laplacian.

In many situations, *e.g.*, to perform spectral graph partitioning, one is interested in computing the first *nontrivial* eigenvector of a Laplacian. Typically, this vector is computed "exactly" by calling a black-box solver; but it could also be approximated with an iteration-based method (such as the Power Method or Lanczos Method) or by running a random walk-based or diffusion-based method to the asymptotic state. These random walk-based or diffusion-based methods assign positive and negative "charge" to the nodes, and then they let the distribution of charge evolve according to dynamics derived from the graph structure. Three canonical evolution dynamics are the following:

Heat Kernel. Here, the charge evolves according to the heat equation $\frac{\partial H_t}{\partial t} = -LH_t$. Thus, the vector of charges evolves as $H_t = \exp(-tL) = \sum_{k=0}^{\infty} \frac{(-t)^k}{k!} L^k$, where $t \geq 0$ is a time parameter, times an input seed distribution vector.

PageRank. Here, the charge at a node evolves by either moving to a neighbor of the current node or teleporting to a random node. More formally, the vector of charges evolves as

$$R_\gamma = \gamma \left(I - (1 - \gamma) M \right)^{-1}, \tag{1}$$

where M is the natural random walk transition matrix associated with the graph and where $\gamma \in (0, 1)$ is the so-called teleportation parameter, times an input seed vector.

Lazy Random Walk. Here, the charge either stays at the current node or moves to a neighbor. Thus, if M is the natural random walk transition matrix associated with the graph, then the vector of charges evolves as some power of $W_\alpha = \alpha I + (1 - \alpha)M$, where $\alpha \in (0, 1)$ represents the "holding probability," times an input seed vector.

In each of these cases, there is a parameter (t, γ, and the number of steps of the Lazy Random Walk) that controls the "aggressiveness" of the dynamics and thus how quickly the diffusive process equilibrates; and there is an input "seed" distribution vector. Thus, *e.g.*, if one is interested in global spectral graph partitioning, then this seed vector could be a vector with entries drawn from $\{-1, +1\}$ uniformly at random, while if one is interested in local spectral graph partitioning [9, 10, 11, 12], then this vector could be the indicator vector of a small "seed set" of nodes. See Appendix A of [8] for a brief discussion of local and global spectral partitioning in this context.

Mahoney and Orecchia showed that these three dynamics arise as solutions to SDPs of the form

$$\begin{array}{ll} \underset{X}{\text{minimize}} & \mathrm{Tr}(LX) + \frac{1}{\eta} G(X) \\ \text{subject to} & X \succeq 0, \\ & \mathrm{Tr}(X) = 1, \\ & X D^{1/2} 1 = 0, \end{array} \tag{2}$$

where G is a penalty function (shown to be the generalized entropy, the log-determinant, and a certain matrix-p-norm, respectively [1]) and where η is a parameter related to the aggressiveness of the diffusive process [1]. Conversely, solutions to the regularized SDP of (2) for appropriate values of η can be computed *exactly* by running one of the above three diffusion-based procedures. Notably, when $G = 0$, the solution to the SDP of (2) is uu', where u is the smallest nontrivial eigenvector of L. More generally and in this precise sense, the Heat Kernel, PageRank, and Lazy Random Walk dynamics can be seen as "regularized" versions of spectral clustering and Laplacian eigenvector computation. Intuitively, the function $G(\cdot)$ is acting as a penalty function, in a manner analogous to the ℓ_2 or ℓ_1 penalty in Ridge regression or Lasso regression, and by running one of these three dynamics one is *implicitly* making assumptions about the form of $G(\cdot)$. In this paper, we provide a statistical framework to make that intuition precise.

3 A statistical framework for regularized graph estimation

Here, we will lay out a simple Bayesian framework for estimating a graph Laplacian. Importantly, this framework will allow for regularization by incorporating prior information.

3.1 Analogy with regularized linear regression

It will be helpful to keep in mind the Bayesian interpretation of regularized linear regression. In that context, we observe n predictor-response pairs in $\mathbb{R}^p \times \mathbb{R}$, denoted $(x_1, y_1), \ldots, (x_n, y_n)$; the goal is to find a vector β such that $\beta' x_i \approx y_i$. Typically, we choose β by minimizing the residual sum of squares, *i.e.*, $F(\beta) = \mathrm{RSS}(\beta) = \sum_i \|y_i - \beta' x_i\|_2^2$, or a penalized version of it. For Ridge regression, we minimize $F(\beta) + \lambda \|\beta\|_2^2$; while for Lasso regression, we minimize $F(\beta) + \lambda \|\beta\|_1$.

The additional terms in the optimization criteria (*i.e.*, $\lambda \|\beta\|_2^2$ and $\lambda \|\beta\|_1$) are called penalty functions; and adding a penalty function to the optimization criterion can often be interpreted as incorporating prior information about β. For example, we can model $y_1, \ldots, y_n$ as independent random observations with distributions dependent on β. Specifically, we can suppose y_i is a Gaussian random variable with mean $\beta' x_i$ and known variance σ^2. This induces a conditional density for the vector $y = (y_1, \ldots, y_n)$:

$$p(y \mid \beta) \propto \exp\{-\tfrac{1}{2\sigma^2} F(\beta)\}, \tag{3}$$

where the constant of proportionality depends only on y and σ. Next, we can assume that β itself is random, drawn from a distribution with density $p(\beta)$. This distribution is called a prior, since it encodes prior knowledge about β. Without loss of generality, the prior density can be assumed to take the form

$$p(\beta) \propto \exp\{-U(\beta)\}. \tag{4}$$

Since the two random variables are dependent, upon observing y, we have information about β. This information is encoded in the posterior density, $p(\beta \mid y)$, computed via Bayes' rule as

$$p(\beta \mid y) \propto p(y \mid \beta)\, p(\beta) \propto \exp\{-\tfrac{1}{2\sigma^2} F(\beta) - U(\beta)\}. \tag{5}$$

The MAP estimate of β is the value that maximizes $p(\beta \mid y)$; equivalently, it is the value of β that minimizes $-\log p(\beta \mid y)$. In this framework, we can recover the solution to Ridge regression or Lasso regression by setting $U(\beta) = \frac{\lambda}{2\sigma^2}\|\beta\|_2^2$ or $U(\beta) = \frac{\lambda}{2\sigma^2}\|\beta\|_1$, respectively. Thus, Ridge regression can be interpreted as imposing a Gaussian prior on β, and Lasso regression can be interpreted as imposing a double-exponential prior on β.

3.2 Bayesian inference for the population Laplacian

For our problem, suppose that we have a connected graph with n nodes; or, equivalently, that we have L, the normalized Laplacian of that graph. We will view this observed graph Laplacian, L, as a "sample" Laplacian, *i.e.*, as random object whose distribution depends on a true "population" Laplacian, $\mathcal{L}$. As with the linear regression example, this induces a conditional density for L, to be denoted $p(L \mid \mathcal{L})$. Next, we can assume prior information about the population Laplacian in the form of a prior density, $p(\mathcal{L})$; and, given the observed Laplacian, we can estimate the population Laplacian by maximizing its posterior density, $p(\mathcal{L} \mid L)$.

Thus, to apply the Bayesian formalism, we need to specify the conditional density of L given $\mathcal{L}$. In the context of linear regression, we assumed that the observations followed a Gaussian distribution. A graph Laplacian is not just a single observation—it is a positive semidefinite matrix with a very specific structure. Thus, we will take L to be a random object with expectation $\mathcal{L}$, where $\mathcal{L}$ is another normalized graph Laplacian. Although, in general, $\mathcal{L}$ can be distinct from L, we will require that the nodes in the population and sample graphs have the same degrees. That is, if $d = \big(d(1), \ldots, d(n)\big)$ denotes the "degree vector" of the graph, and $D = \operatorname{diag}\big(d(1), \ldots, d(n)\big)$, then we can define

$$\mathcal{X} = \{X : X \succeq 0,\; X D^{1/2} 1 = 0,\; \operatorname{rank}(X) = n-1\}, \tag{6}$$

in which case the population Laplacian and the sample Laplacian will both be members of $\mathcal{X}$. To model L, we will choose a distribution for positive semi-definite matrices analogous to the Gaussian distribution: a scaled Wishart matrix with expectation $\mathcal{L}$. Note that, although it captures the trait that L is positive semi-definite, this distribution does not accurately model every feature of L. For example, a scaled Wishart matrix does not necessarily have ones along its diagonal. However, the mode of the density is at $\mathcal{L}$, a Laplacian; and for large values of the scale parameter, most of the mass will be on matrices close to $\mathcal{L}$. Appendix B of [8] provides a more detailed heuristic justification for the use of the Wishart distribution.

To be more precise, let $m \geq n-1$ be a scale parameter, and suppose that L is distributed over $\mathcal{X}$ as a $\frac{1}{m}\operatorname{Wishart}(\mathcal{L}, m)$ random variable. Then, $\mathbb{E}[L \mid \mathcal{L}] = \mathcal{L}$, and L has conditional density

$$p(L \mid \mathcal{L}) \propto \frac{\exp\{-\frac{m}{2}\operatorname{Tr}(L\mathcal{L}^+)\}}{|\mathcal{L}|^{m/2}}, \tag{7}$$

where $|\cdot|$ denotes pseudodeterminant (product of nonzero eigenvalues). The constant of proportionality depends only on L, d, m, and n; and we emphasize that the density is supported on $\mathcal{X}$. Eqn. (7) is analogous to Eqn. (3) in the linear regression context, with $1/m$, the inverse of the sample size parameter, playing the role of the variance parameter σ^2. Next, suppose we have know that $\mathcal{L}$ is a random object drawn from a prior density $p(\mathcal{L})$. Without loss of generality,

$$p(\mathcal{L}) \propto \exp\{-U(\mathcal{L})\}, \tag{8}$$

for some function U, supported on a subset $\bar{\mathcal{X}} \subseteq \mathcal{X}$. Eqn. (8) is analogous to Eqn. (4) from the linear regression example. Upon observing L, the posterior distribution for $\mathcal{L}$ is

$$p(\mathcal{L} \mid L) \propto p(L \mid \mathcal{L})\, p(\mathcal{L}) \propto \exp\{-\tfrac{m}{2}\operatorname{Tr}(L\mathcal{L}^+) + \tfrac{m}{2}\log|\mathcal{L}^+| - U(\mathcal{L})\}, \tag{9}$$

with support determined by $\bar{\mathcal{X}}$. Eqn. (9) is analogous to Eqn. (5) from the linear regression example. If we denote by $\hat{\mathcal{L}}$ the MAP estimate of $\mathcal{L}$, then it follows that $\hat{\mathcal{L}}^+$ is the solution to the program

$$\begin{aligned} \underset{X}{\text{minimize}} \quad & \operatorname{Tr}(LX) + \tfrac{2}{m}U(X^+) - \log|X| \\ \text{subject to} \quad & X \in \bar{\mathcal{X}} \subseteq \mathcal{X}. \end{aligned} \tag{10}$$

Note the similarity with Mahoney-Orecchia regularized SDP of (2). In particular, if $\bar{\mathcal{X}} = \{X : \mathrm{Tr}(X) = 1\} \cap \mathcal{X}$, then the two programs are identical except for the factor of $\log|X|$ in the optimization criterion.

4 A prior related to the PageRank procedure

Here, we will present a prior distribution for the population Laplacian that will allow us to leverage the estimation framework of Section 3; and we will show that the MAP estimate of $\mathcal{L}$ for this prior is related to the PageRank procedure via the Mahoney-Orecchia regularized SDP. Appendix C of [8] presents priors that lead to the Heat Kernel and Lazy Random Walk in an analogous way; in both of these cases, however, the priors are data-dependent in the strong sense that they explicitly depend on the number of data points.

4.1 Prior density

The prior we will present will be based on neutrality and invariance conditions; and it will be supported on $\mathcal{X}$, *i.e.*, on the subset of positive-semidefinite matrices that was the support set for the conditional density defined in Eqn. (7). In particular, recall that, in addition to being positive semidefinite, every matrix in the support set has rank $n-1$ and satisfies $XD^{1/2}1 = 0$. Note that because the prior depends on the data (via the orthogonality constraint induced by D), this is not a prior in the fully Bayesian sense; instead, the prior can be considered as part of an empirical or pseudo-Bayes estimation procedure.

The prior we will specify depends only on the eigenvalues of the normalized Laplacian, or equivalently on the eigenvalues of the pseudoinverse of the Laplacian. Let $\mathcal{L}^+ = \tau\, O \Lambda O'$ be the spectral decomposition of the pseudoinverse of the normalized Laplacian $\mathcal{L}$, where $\tau \geq 0$ is a scale factor, $O \in \mathbb{R}^{n \times n-1}$ is an orthogonal matrix, and $\Lambda = \mathrm{diag}\big(\lambda(1), \dots, \lambda(n-1)\big)$, where $\sum_v \lambda(v) = 1$. Note that the values $\lambda(1), \dots, \lambda(n-1)$ are unordered and that the vector $\lambda = \big(\lambda(1), \dots, \lambda(n-1)\big)$ lies in the unit simplex. If we require that the distribution for λ be exchangeable (invariant under permutations) and neutral ($\lambda(v)$ independent of the vector $\big(\lambda(u)/(1-\lambda(v)) : u \neq v\big)$, for all v), then the only non-degenerate possibility is that λ is Dirichlet-distributed with parameter vector $(\alpha, \dots, \alpha)$ [13]. The parameter α, to which we refer as the "shape" parameter, must satisfy $\alpha > 0$ for the density to be defined. In this case,

$$p(\mathcal{L}) \propto p(\tau) \prod_{v=1}^{n-1} \lambda(v)^{\alpha-1}, \tag{11}$$

where $p(\tau)$ is a prior for τ. Thus, the prior weight on $\mathcal{L}$ only depends on τ and Λ. One implication is that the prior is "nearly" rotationally invariant, in the sense that $p(P'\mathcal{L}P) = p(\mathcal{L})$ for any rank-$(n-1)$ projection matrix P satisfying $PD^{1/2}1 = 0$.

4.2 Posterior estimation and connection to PageRank

To analyze the MAP estimate associated with the prior of Eqn. (11) and to explain its connection with the PageRank dynamics, the following proposition is crucial.

Proposition 4.1. *Suppose the conditional likelihood for L given $\mathcal{L}$ is as defined in* (7) *and the prior density for $\mathcal{L}$ is as defined in* (11)*. Define $\hat{\mathcal{L}}$ to be the MAP estimate of $\mathcal{L}$. Then,* $[\mathrm{Tr}(\hat{\mathcal{L}}^+)]^{-1}\hat{\mathcal{L}}^+$ *solves the Mahoney-Orecchia regularized SDP* (2)*, with $G(X) = -\log|X|$ and η as given in Eqn. (12) below.*

Proof. For $\mathcal{L}$ in the support set of the posterior, define $\tau = \mathrm{Tr}(\mathcal{L}^+)$ and $\Theta = \tau^{-1}\mathcal{L}^+$, so that $\mathrm{Tr}(\Theta) = 1$. Further, $\mathrm{rank}(\Theta) = n-1$. Express the prior in the form of Eqn. (8) with function U given by

$$U(\mathcal{L}) = -\log\{p(\tau)\,|\Theta|^{\alpha-1}\} = -(\alpha-1)\log|\Theta| - \log p(\tau),$$

where, as before, $|\cdot|$ denotes pseudodeterminant. Using (9) and the relation $|\mathcal{L}^+| = \tau^{n-1}|\Theta|$, the posterior density for $\mathcal{L}$ given L is

$$p(\mathcal{L} \mid L) \propto \exp\Big\{ -\tfrac{m\tau}{2}\,\mathrm{Tr}(L\Theta) + \tfrac{m+2(\alpha-1)}{2}\log|\Theta| + g(\tau)\Big\},$$

where $g(\tau) = \frac{m(n-1)}{2} \log \tau + \log p(\tau)$. Suppose $\hat{\mathcal{L}}$ maximizes the posterior likelihood. Define $\hat{\tau} = \mathrm{Tr}(\hat{\mathcal{L}}^+)$ and $\hat{\Theta} = [\hat{\tau}]^{-1} \hat{\mathcal{L}}^+$. In this case, $\hat{\Theta}$ must minimize the quantity $\mathrm{Tr}(L\hat{\Theta}) - \frac{1}{\eta} \log |\hat{\Theta}|$, where

$$\eta = \frac{m\hat{\tau}}{m + 2(\alpha - 1)}. \tag{12}$$

Thus $\hat{\Theta}$ solves the regularized SDP (2) with $G(X) = -\log |X|$. $\square$

Mahoney and Orecchia showed that the solution to (2) with $G(X) = -\log |X|$ is closely related to the PageRank matrix, R_γ, defined in Eqn. (1). By combining Proposition 4.1 with their result, we get that the MAP estimate of $\mathcal{L}$ satisfies $\hat{\mathcal{L}}^+ \propto D^{-1/2} R_\gamma D^{1/2}$; conversely, $R_\gamma \propto D^{1/2} \hat{\mathcal{L}}^+ D^{-1/2}$. Thus, the PageRank operator of Eqn. (1) can be viewed as a degree-scaled regularized estimate of the pseudoinverse of the Laplacian. Moreover, prior assumptions about the spectrum of the graph Laplacian have direct implications on the optimal teleportation parameter. Specifically Mahoney and Orecchia's Lemma 2 shows how η is related to the teleportation parameter γ, and Eqn. (12) shows how the optimal η is related to prior assumptions about the Laplacian.

5 Empirical evaluation

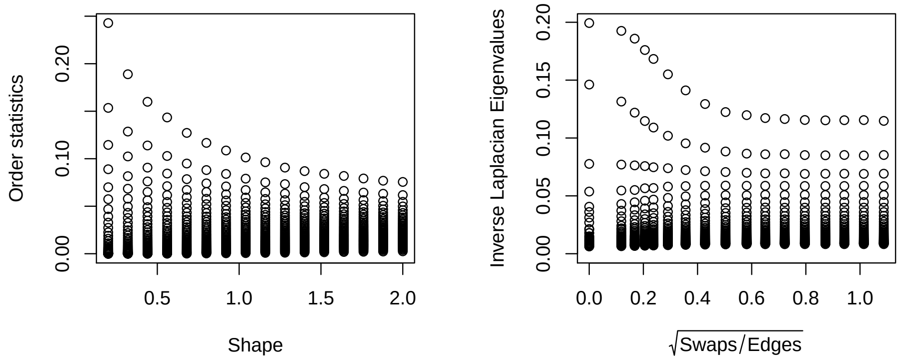
In this section, we provide an empirical evaluation of the performance of the regularized Laplacian estimator, compared with the unregularized estimator. To do this, we need a ground truth population Laplacian $\mathcal{L}$ and a noisily-observed sample Laplacian L. Thus, in Section 5.1, we construct a family of distributions for $\mathcal{L}$; importantly, this family will be able to represent both low-dimensional graphs and expander-like graphs. Interestingly, the prior of Eqn. (11) captures some of the qualitative features of both of these types of graphs (as the shape parameter is varied). Then, in Section 5.2, we describe a sampling procedure for L which, superficially, has no relation to the scaled Wishart conditional density of Eqn. (7). Despite this model misspecification, the regularized estimator $\hat{L}_\eta$ outperforms L for many choices of the regularization parameter η.

5.1 Ground truth generation and prior evaluation

The ground truth graphs we generate are motivated by the Watts-Strogatz "small-world" model [14]. To generate a ground truth population Laplacian, $\mathcal{L}$—equivalently, a population graph—we start with a two-dimensional lattice of width w and height h, and thus $n = wh$ nodes. Points in the lattice are connected to their four nearest neighbors, making adjustments as necessary at the boundary. We then perform s edge-swaps: for each swap, we choose two edges uniformly at random and then we swap the endpoints. For example, if we sample edges $i_1 \sim j_1$ and $i_2 \sim j_2$, then we replace these edges with $i_1 \sim j_2$ and $i_2 \sim j_1$. Thus, when $s = 0$, the graph is the original discretization of a low-dimensional space; and as s increases to infinity, the graph becomes more and more like a uniformly chosen 4-regular graph (which is an expander [15] and which bears similarities with an Erdős-Rényi random graph [16]). Indeed, each edge swap is a step of the Metropolis algorithm toward a uniformly chosen random graph with a fixed degree sequence. For the empirical evaluation presented here, $h = 7$ and $w = 6$; but the results are qualitatively similar for other values.

Figure 1 compares the expected order statistics (sorted values) for the Dirichlet prior of Eqn. (11) with the expected eigenvalues of $\Theta = \mathcal{L}^+ / \mathrm{Tr}(\mathcal{L}^+)$ for the small-world model. In particular, in Figure 1(a), we show the behavior of the order statistics of a Dirichlet distribution on the $(n-1)$-dimensional simplex with scalar shape parameter α, as a function of α. For each value of the shape α, we generated a random $(n-1)$-dimensional Dirichlet vector, λ, with parameter vector $(\alpha, \dots, \alpha)$; we computed the $n-1$ order statistics of λ by sorting its components; and we repeated this procedure for 500 replicates and averaged the values. Figure 1(b) shows a corresponding plot for the ordered eigenvalues of Θ. For each value of s (normalized, here, by the number of edges μ, where $\mu = 2wh - w - h = 71$), we generated the normalized Laplacian, $\mathcal{L}$, corresponding to the random s-edge-swapped grid; we computed the $n-1$ nonzero eigenvalues of Θ; and we performed 1000 replicates of this procedure and averaged the resulting eigenvalues.

Interestingly, the behavior of the spectrum of the small-world model as the edge-swaps increase is qualitatively quite similar to the behavior of the Dirichlet prior order statistics as the shape parameter α increases. In particular, note that for small values of the shape parameter α the first few order-statistics are well-separated from the rest; and that as α increases, the order statistics become



(a) Dirichlet distribution order statistics.

(b) Spectrum of the inverse Laplacian.

Figure 1: Analytical and empirical priors. 1(a) shows the Dirichlet distribution order statistics versus the shape parameter; and 1(b) shows the spectrum of Θ as a function of the rewiring parameter.

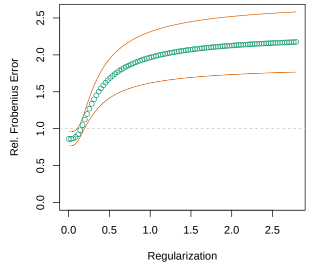
concentrated around $1/(n-1)$. Similarly, when the edge-swap parameter $s = 0$, the top two eigenvalues (corresponding to the width-wise and height-wise coordinates on the grid) are well-separated from the bulk; as s increases, the top eigenvalues quickly merge into the bulk; and eventually, as s goes to infinity, the distribution becomes very close that that of a uniformly chosen 4-regular graph.

5.2 Sampling procedure, estimation performance, and optimal regularization behavior

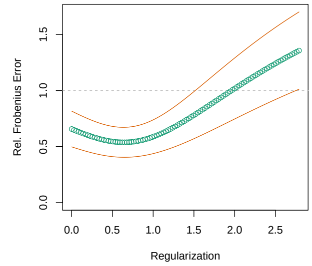
Finally, we evaluate the estimation performance of a regularized estimator of the graph Laplacian and compare it with an unregularized estimate. To do so, we construct the population graph $\mathcal{G}$ and its Laplacian $\mathcal{L}$, for a given value of s, as described in Section 5.1. Let μ be the number of edges in $\mathcal{G}$. The sampling procedure used to generate the observed graph G and its Laplacian L is parameterized by the sample size m. (Note that this parameter is analogous to the Wishart scale parameter in Eqn. (7), but here we are sampling from a different distribution.) We randomly choose m edges with replacement from $\mathcal{G}$; and we define sample graph G and corresponding Laplacian L by setting the weight of $i \sim j$ equal to the number of times we sampled that edge. Note that the sample graph G over-counts some edges in $\mathcal{G}$ and misses others.

We then compute the regularized estimate $\hat{\mathcal{L}}_\eta$, up to a constant of proportionality, by solving (implicitly!) the Mahoney-Orecchia regularized SDP (2) with $G(X) = -\log|X|$. We define the unregularized estimate $\hat{L}$ to be equal to the observed Laplacian, L. Given a population Laplacian $\mathcal{L}$, we define $\tau = \tau(\mathcal{L}) = \mathrm{Tr}(\mathcal{L}^+)$ and $\Theta = \Theta(\mathcal{L}) = \tau^{-1}\mathcal{L}^+$. We define $\hat{\tau}_\eta$, $\hat{\tau}$, $\hat{\Theta}_\eta$, and $\hat{\Theta}$ similarly to the population quantities. Our performance criterion is the relative Frobenius error $\|\Theta - \hat{\Theta}_\eta\|_{\mathrm{F}} / \|\Theta - \hat{\Theta}\|_{\mathrm{F}}$, where $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm ($\|A\|_{\mathrm{F}} = [\mathrm{Tr}(A'A)]^{1/2}$). Appendix D of [8] presents similar results when the performance criterion is the relative spectral norm error.

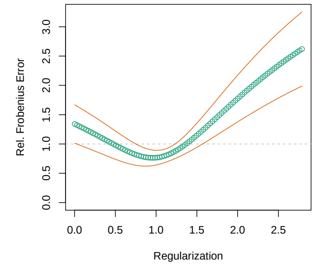
Figures 2(a), 2(b), and 2(c) show the regularization performance when $s = 4$ (an intermediate value) for three different values of m/μ. In each case, the mean error and one standard deviation around it are plotted as a function of $\eta/\bar{\tau}$, as computed from 100 replicates; here, $\bar{\tau}$ is the mean value of τ over all replicates. The implicit regularization clearly improves the performance of the estimator for a large range of η values. (Note that the regularization parameter in the regularized SDP (2) is $1/\eta$, and thus smaller values along the X-axis correspond to stronger regularization.) In particular, when the data are very noisy, *e.g.*, when $m/\mu = 0.2$, as in Figure 2(a), improved results are seen only for very strong regularization; for intermediate levels of noise, *e.g.*, $m/\mu = 1.0$, as in Figure 2(b), (in which case m is chosen such that G and $\mathcal{G}$ have the same number of edges counting multiplicity), improved performance is seen for a wide range of values of η; and for low levels of noise, Figure 2(c) illustrates that improved results are obtained for moderate levels of implicit regularization. Figures 2(d) and 2(e) illustrate similar results for $s = 0$ and $s = 32$.



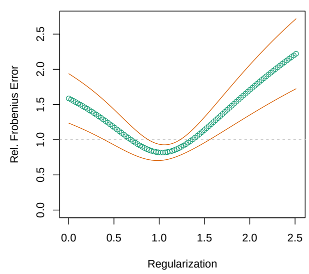
(a) $m/\mu = 0.2$ and $s = 4$.



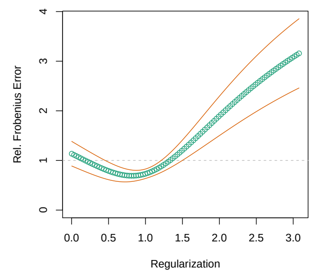
(b) $m/\mu = 1.0$ and $s = 4$.



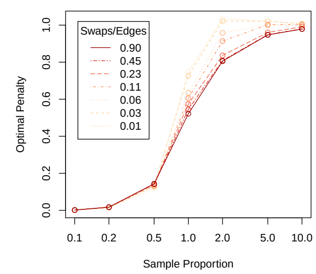
(c) $m/\mu = 2.0$ and $s = 4$.



(d) $m/\mu = 2.0$ and $s = 0$.



(e) $m/\mu = 2.0$ and $s = 32$.



(f) Optimal $\eta^*/\bar{\tau}$.

Figure 2: Regularization performance. 2(a) through 2(e) plot the relative Frobenius norm error, versus the (normalized) regularization parameter $\eta/\bar{\tau}$. Shown are plots for various values of the (normalized) number of edges, m/μ, and the edge-swap parameter, s. Recall that the regularization parameter in the regularized SDP (2) is $1/\eta$, and thus smaller values along the X-axis correspond to stronger regularization. 2(f) plots the optimal regularization parameter $\eta^*/\bar{\tau}$ as a function of sample proportion for different fractions of edge swaps.

As when regularization is implemented explicitly, in all these cases, we observe a "sweet spot" where there is an optimal value for the implicit regularization parameter. Figure 2(f) illustrates how the optimal choice of η depends on parameters defining the population Laplacians and sample Laplacians. In particular, it illustrates how η^*, the optimal value of η (normalized by $\bar{\tau}$), depends on the sampling proportion m/μ and the swaps per edges s/μ. Observe that as the sample size m increases, η^* converges monotonically to $\bar{\tau}$; and, further, that higher values of s (corresponding to more expander-like graphs) correspond to higher values of η^*. Both of these observations are in direct agreement with Eqn. (12).

6 Conclusion

We have provided a statistical interpretation for the observation that popular diffusion-based procedures to compute a quick approximation to the first nontrivial eigenvector of a data graph Laplacian exactly solve a certain regularized version of the problem. One might be tempted to view our results as "unfortunate," in that it is not straightforward to interpret the priors presented in this paper. Instead, our results should be viewed as making explicit the implicit prior assumptions associated with making certain decisions (that are *already* made in practice) to speed up computations.

Several extensions suggest themselves. The most obvious might be to try to obtain Proposition 4.1 with a more natural or empirically-plausible model than the Wishart distribution; to extend the empirical evaluation to much larger and more realistic data sets; to apply our methodology to other widely-used approximation procedures; and to characterize when implicitly regularizing an eigenvector leads to better statistical behavior in downstream applications where that eigenvector is used. More generally, though, we expect that understanding the algorithmic-statistical tradeoffs that we have illustrated will become increasingly important in very large-scale data analysis applications.

References

[1] M. W. Mahoney and L. Orecchia. Implementing regularization implicitly via approximate eigenvector computation. In *Proceedings of the 28th International Conference on Machine Learning*, pages 121–128, 2011.

[2] D.A. Spielman and S.-H. Teng. Spectral partitioning works: Planar graphs and finite element meshes. In *FOCS '96: Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science*, pages 96–107, 1996.

[3] S. Guattery and G.L. Miller. On the quality of spectral separators. *SIAM Journal on Matrix Analysis and Applications*, 19:701–719, 1998.

[4] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transcations of Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[5] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003.

[6] T. Joachims. Transductive learning via spectral graph partitioning. In *Proceedings of the 20th International Conference on Machine Learning*, pages 290–297, 2003.

[7] J. Leskovec, K.J. Lang, A. Dasgupta, and M.W. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009. Also available at: arXiv:0810.1355.

[8] P. O. Perry and M. W. Mahoney. Regularized Laplacian estimation and fast eigenvector approximation. Technical report. Preprint: arXiv:1110.1757 (2011).

[9] D.A. Spielman and S.-H. Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *STOC '04: Proceedings of the 36th annual ACM Symposium on Theory of Computing*, pages 81–90, 2004.

[10] R. Andersen, F.R.K. Chung, and K. Lang. Local graph partitioning using PageRank vectors. In *FOCS '06: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 475–486, 2006.

[11] F.R.K. Chung. The heat kernel as the pagerank of a graph. *Proceedings of the National Academy of Sciences of the United States of America*, 104(50):19735–19740, 2007.

[12] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi. A spectral algorithm for improving graph partitions with applications to exploring data graphs locally. Technical report. Preprint: arXiv:0912.0681 (2009).

[13] J. Fabius. Two characterizations of the Dirichlet distribution. *The Annals of Statistics*, 1(3):583–587, 1973.

[14] D.J. Watts and S.H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.

[15] S. Hoory, N. Linial, and A. Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43:439–561, 2006.

[16] B. Bollobas. *Random Graphs*. Academic Press, London, 1985.