Supplementary materials for this article are available online. Please click the JASA link at http://pubs.amstat.org.

Spectral Connectivity Analysis

Ann B. LEE and Larry WASSERMAN

Spectral kernel methods are techniques for mapping data into a coordinate system that efficiently reveals the geometric structure—in particular, the "connectivity"—of the data. These methods depend on tuning parameters. We analyze the dependence of the method on these tuning parameters. We focus on one particular technique—diffusion maps—but our analysis can be used for other spectral methods as well. We identify the key population quantities, we define an appropriate risk function for analyzing the estimators, and we explain how these methods relate to classical kernel smoothing. We also show that, in some cases, fast rates of convergence are possible even in high dimensions. The Appendix of the article is available online as supplementary materials.

KEY WORDS: Diffusion maps; Graph Laplacian; Kernels; Manifold learning; Smoothing; Spectral clustering.

1. INTRODUCTION

There has been growing interest in spectral kernel methods such as spectral clustering (Ng, Jordan, and Weiss 2001, von Luxburg 2007), Laplacian maps (Belkin and Niyogi 2003), Hessian maps (Donoho and Grimes 2003), and locally linear embeddings (Roweis and Saul 2000). The main idea behind these methods is that the geometry of a dataset can be analyzed using certain operators and their corresponding eigenfunctions. These eigenfunctions provide a new coordinate system that show the data more clearly. In Figure 1, for example, we have applied spectral kernel methods to text data. In the new coordinate system, different words from Science News articles are roughly organized according to their semantic meaning.

Figure 2 shows an application to astronomy data. To the left, we see an example of a galaxy spectrum (a function that measures photon flux at more than 3000 different wavelengths) from the Sloan Digital Sky Survey. To the right, we have computed a low-dimensional embedding of a sample of 2793 such spectra. The results indicate that by analyzing only a few dominant eigenfunctions of this highly complex dataset, one can capture the main variability in redshift (a quantity related to the distance of a galaxy from the observer), although redshift was not taken into account in the construction of the embedding.

More generally, the central goal of spectral kernel methods can be described as follows:

> Find a transformation $Z = \Psi(X)$ such that the structure of the distribution P_Z is simpler than the structure of the distribution P_X while preserving key geometric properties of P_X.

"Simpler" can mean lower dimensional but can be interpreted much more broadly as we shall see.

These new methods of data transformation are more flexible than traditional methods such as principal component analysis, clustering, and kernel smoothing. Applications of these methods include: manifold learning (Levina and Bickel 2005), fast internet web searches (Page et al. 1998), semi-supervised learning for regression and classification (Belkin and Niyogi 2005a), inference of arbitrarily shaped clusters, etc. The added flexibility, however, comes at a price: there are tuning parameters, such as a kernel bandwidth ε, and the dimension q of the embedding that need to be chosen and these parameters often interact in a complicated way. The first step in understanding these tuning parameters is to identify the population quantity these methods are effectively estimating, then define an appropriate loss function.

We restrict our discussion to Laplacian-based methods, though the analysis generalizes to other spectral kernel methods. Several authors, including Coifman and Lafon (2006), Belkin and Niyogi (2005b), Hein, Audibert, and von Luxburg (2005b), Singer (2006), and Giné and Koltchinskii (2006) have studied the convergence of the empirical graph Laplacian to the Laplace–Beltrami operator of a smooth manifold as the sample size $n \to \infty$ and the kernel bandwidth $\varepsilon \to 0$. In all these studies, the data are assumed to lie exactly on a Riemannian submanifold in the ambient space $\mathbb{R}^d$.

Here we drop the submanifold assumption and instead consider data that are drawn from some general underlying distribution P. For the dimension reduction methods to be useful, it is implicitly assumed that the measure P places most of its probability on a subset of $\mathbb{R}^d$ of small Lebesgue measure. Hence, our analysis includes but is not restricted to low-dimensional Riemannian submanifolds and sets of clusters. Recently, von Luxburg, Belkin, and Bousquet (2008) has taken a similar approach when studying the consistency of spectral clustering. For a fixed kernel bandwidth ε and in the limit of the sample size $n \to \infty$, the authors show that the eigenvectors of the graph Laplacian converge to the eigenvectors of certain limit operators. In this paper, we allow ε to go to 0.

The goals of the paper are to:

1. explain how spectral kernel methods relate to classical kernel smoothing methods
2. identify the key population quantities ($\mathbf{A}_t$ and $\mathbf{D}_t$ in Sections 3 and 4) in Laplacian-based spectral methods and describe their relation to the Laplace–Beltrami operator commonly discussed in other work
3. find the appropriate risk in estimation of these quantities and discuss the problem of choosing the tuning parameters (Section 5).

We analyze a metric formulation of Laplacian-based spectral methods, called *diffusion maps*. Diffusion maps capture the

Ann B. Lee is Associate Professor, Department of Statistics (E-mail: *annlee@stat.cmu.edu*) and Larry Wasserman is Professor, Department of Statistics, Carnegie Mellon University, Pittsburgh, PA 15213. We are grateful to R. R. Coifman and S. Lafon for discussions that preceded this work. We also thank the editors and two anonymous reviewers for helpful comments on the paper. This work was supported in part by ONR grant N0014-08-1-0673.

Journal of the American Statistical Association
September 2010, Vol. 105, No. 491, Theory and Methods
DOI: 10.1198/jasa.2010.tm09754

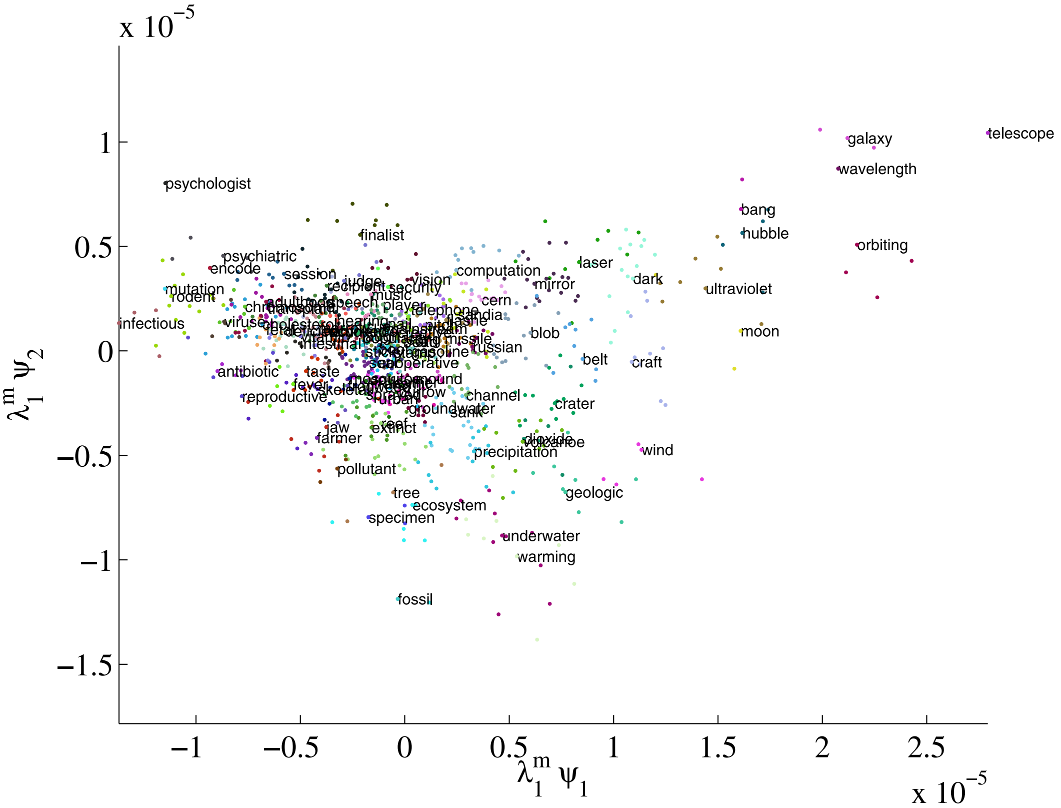

Figure 1. Low-dimensional embedding and grouping of words from Science News articles. The labels denote the representative centers of each group of words. See Section 6.2 for details. The online version of this figure is in color.

multiscale structure of the data by propagating local neighborhood information through a Markov process. Spectral geometry and higher-order connectivity are two new concepts in data analysis. In the paper, we show how these ideas can be incorporated into a traditional statistical framework (see, e.g., Interpretation in Section 3), and how this connection extends classical techniques to a whole range of new applications. We refer to the family of approaches based on spectral analysis and higher-order connectivity as *Spectral Connectivity Analysis (SCA)*.

2. REVIEW OF SPECTRAL DIMENSION REDUCTION METHODS

The goal of dimensionality reduction is to find a function Ψ that maps our data X from a space $\mathcal{X}$ to a new space $\mathcal{Z}$ where

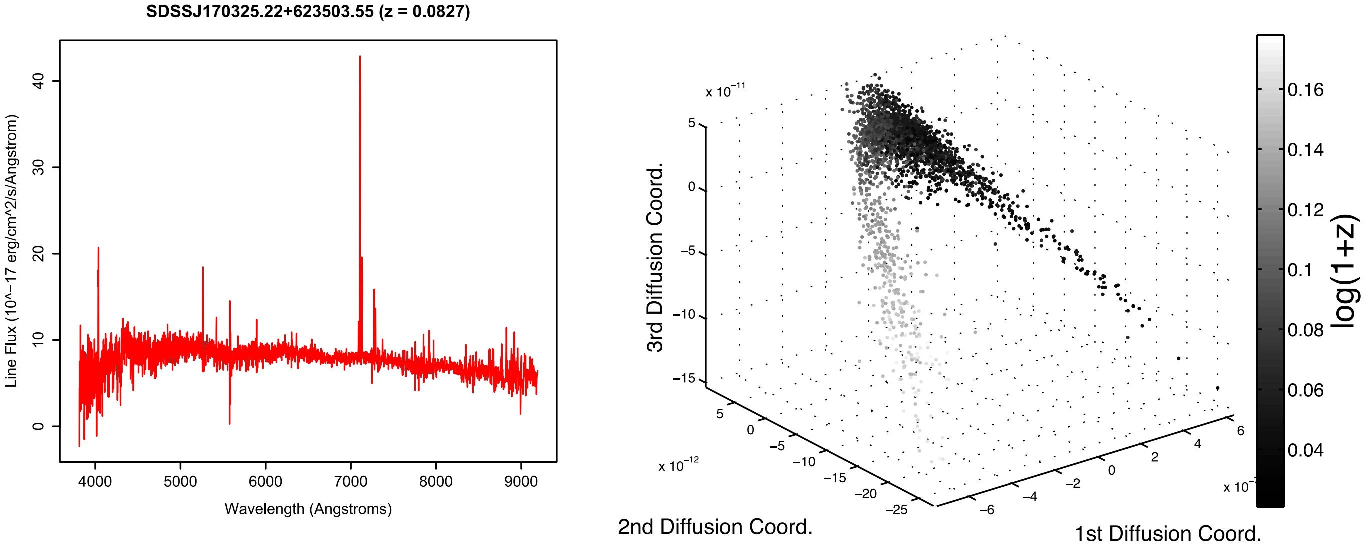

Figure 2. Left: Flux versus wavelength for a typical Sloan Digital Sky Survey (SDSS) galaxy spectrum. Right: Embedding of a sample of 2793 SDSS galaxy spectra using the first 3 diffusion map coordinates. The gray scale codes for redshift. (Reproduced from Richards et al. 2009b).

their description is considered to be simpler. Some of the methods naturally lead to an eigen-problem. Below we give some examples.

2.1 Principal Component Analysis and Multidimensional Scaling

Principal component mapping is a simple and popular method for data reduction. In principal component analysis (PCA), one attempts to fit a globally linear model to the data. If S is a set, define the projection risk $R(S) = \mathbb{E}\|X - \pi_S X\|^2$ where $\pi_S X$ is the projection of X onto S. Finding $\arg\min_{S\in\mathcal{C}} R(S)$, where $\mathcal{C}$ is the set of all q-dimensional planes, gives a solution that corresponds to the first q eigenvectors of the covariance matrix of X.

In "principal coordinate analysis" (PCO), the projections $\pi_S x = (z_1, \dots, z_q)$ on these eigenvectors are used as coordinates of the data. This method of data transformation is also known as classical or metric multidimensional scaling (MDS). The goal here is to find a lower-dimensional embedding of the data that best preserves pairwise Euclidean distances. Assume that X and Y are covariates in $\mathbb{R}^p$. One way to measure the discrepancy between the original configuration and its embedding is to compute

$$\begin{aligned} R(\Psi) &= \mathbb{E}\big(d(X,Y)^2 - \|\Psi(X) - \Psi(Y)\|^2\big) \\ &= \int \big(d(x,y)^2 - \|\Psi(x) - \Psi(y)\|^2\big)\, dP(x)\, dP(y), \end{aligned}$$

where $d(x,y)^2 = \|x-y\|^2$. One can show that amongst all linear projections $\Psi = \pi_S$ onto q-dimensional subspaces of $\mathbb{R}^p$, this quantity is minimized when the data are projected onto their first q principal components (Mardia, Kent, and Bibby 1980).

2.2 Nonlinear Methods

For complex data, a linear approximation may not be adequate. There are a large number of nonlinear data reduction methods; some of these are direct generalizations of the PCA projection method. For example, local PCA (Kambhatla and Leen 1997) partitions the data space into different regions and fits a hyperplane to the data in each partition. In principal curves (Hastie and Stuetzle 1989), the goal is to minimize a risk of the same form as the projection risk $R(S)$ but with S representing some class of smooth curves or surfaces.

Among nonlinear extensions of PCA and MDS, we also have kernel PCA (Schölkopf, Smola, and Müller 1998) which applies PCA to data $\Phi(X)$ in a higher (possibly infinite) dimensional "feature space." The kernel PCA method never explicitly computes the map Φ, but instead expresses all calculations in terms of inner products $k(x,y) = \langle\Phi(x), \Phi(y)\rangle$ where the "kernel" k is a symmetric and positive semi-definite function. Common choices include the Gaussian kernel $k(x,y) = \exp(-\frac{\|x-y\|^2}{4\varepsilon})$ and the polynomial kernel $k(x,y) = \langle x,y\rangle^r$, where $r = 1$ corresponds to the linear case in Section 2.1. As shown in Bengio et al. (2004), the low-dimensional embeddings $\Psi(x)$ used by the eigenmap and spectral clustering methods in Section 2.2.1 are equivalent to the projections [of $\Phi(x)$ on the principal axes in feature space] computed by the kernel PCA method.

In this paper, we study diffusion maps, a particular spectral embedding technique. Because of the close connection between MDS, kernel PCA, and eigenmap techniques, our analysis can be used for other methods a well. Below we start by providing some background on spectral dimension reduction methods from a more traditional graph-theoretic perspective. In Section 3 we begin our main analysis.

2.2.1 Laplacian Eigenmaps and Other Locality-Preserving Spectral Methods. The usual strategy in spectral methods is to construct an adjacency graph on a given dataset and then find the optimal clustering or encoding of the data that minimizes some empirical locality-preserving objective function on the graph. We define a graph $G = (V, E)$, where the vertex set $V = \{1, \dots, n\}$ denotes the observations, and the edge set E represents connections between pairs of observations. Typically, the graph is associated with a weight matrix $\mathbb{K}$ that reflects the "edge masses" or strengths of the edge connections. A common starting point is the Gaussian kernel: Let, for example, $\mathbb{K}(u,v) = \exp(-\frac{\|x_u - x_v\|^2}{4\varepsilon})$ for all data pairs (x_u, x_v) with $(u,v) \in E$, and only include cases where the weights $\mathbb{K}(u,v)$ are above some threshold δ in the definition of the edge set E.

Consider now a one-dimensional map $f: V \to \mathbb{R}$ that assigns a real value to each vertex; we will later generalize to the multidimensional case. Many spectral embedding techniques are locality preserving; for example, locally linear embedding, Laplacian eigenmaps, Hessian eigenmaps, local tangent space alignment, etc. These methods are special cases of kernel PCA, and all aim at minimizing distortions of the form $Q(f) = \sum_{v\in V} Q_v(f)$ under the constraints that $Q_M(f) = 1$. Typically, $Q_v(f)$ is a symmetric positive semi-definite quadratic form that measures local variations of f around vertex v, and $Q_M(f)$ is a quadratic form that acts as a normalization for f. For Laplacian eigenmaps, for example, the neighborhood structure of G is described in terms of the graph Laplacian matrix $\mathbb{L} = \mathbb{M} - \mathbb{K}$, where $\mathbb{M} = \operatorname{diag}(\rho_1, \dots, \rho_n)$ is a diagonal matrix with $\rho_u = \sum_v \mathbb{K}(u,v)$ for the "node mass" or degree of vertex u. The goal is to find the map f that minimizes the weighted local distortion

$$Q(f) = f^T \mathbb{L} f = \sum_{(u,v)\in E} \mathbb{K}(u,v)(f(u) - f(v))^2 \geq 0, \qquad (1)$$

under the constraints that $Q_M(f) = f^t \mathbb{M} f = \sum_{v\in V} \rho_v f(v)^2 = 1$ and (to avoid the trivial solution of a constant function) $f^T \mathbb{M} 1 = 0$. Minimizing the distortion in (1) forces $f(u)$ and $f(v)$ to be close if $\mathbb{K}(u,v)$ is large. From standard linear algebra it follows that the optimal embedding is given by the eigenvector of the generalized eigenvalue problem

$$\mathbb{L} f = \mu \mathbb{M} f \qquad (2)$$

with the smallest nonzero eigenvalue.

We can easily extend the discussion to higher dimensions. Let $f_1, \dots, f_q$ be the q first nontrivial eigenvectors of (2), normalized so that $f_i^T M f_j = \delta_{ij}$, where δ_{ij} is Kronecker's delta function. The map $f: V \to \mathbb{R}^q$, where $f = (f_1, \dots, f_q)$ is the Laplacian eigenmap (Belkin and Niyogi 2003) of G in q dimensions. It is optimal in the sense that it provides the q-dimensional embedding that minimizes

$$\sum_{(u,v)\in E} \mathbb{K}(u,v)\|f(u) - f(v)\|^2 = \sum_{i=1}^{q} f_i^T \mathbb{L} f_i \qquad (3)$$

in the subspace orthogonal to $\mathbb{M}1$, under the constraints that $f_i^T \mathbb{M} f_j = \delta_{ij}$ for $i, j = 1, \ldots, q$.

If the data points x_u lie on a Riemannian manifold $\mathcal{M}$, and $f : \mathcal{M} \to \mathbb{R}$ is a twice differentiable function, then the expression in Eq. (1) is the discrete analogue on graphs of $\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f\|^2 = -\int_{\mathcal{M}} (\triangle_{\mathcal{M}} f) f$, where $\nabla_{\mathcal{M}}$ and $\triangle_{\mathcal{M}}$, respectively, are the gradient and Laplace–Beltrami operators on the manifold. The solution of $\arg\min_{\|f\|=1} \int_{\mathcal{M}} \|\nabla_{\mathcal{M}} f\|^2$ is given by the eigenvectors of the Laplace–Beltrami operator $\triangle_{\mathcal{M}}$. To give a theoretical justification for Laplacian-based spectral methods, several authors have derived results for the convergence of the graph Laplacian of a point cloud to the Laplace–Beltrami operator under the manifold assumption; see Belkin and Niyogi (2005b); Coifman and Lafon (2006); Singer (2006); Giné and Koltchinskii (2006).

2.2.2 Laplacian-Based Methods With an Explicit Metric. Diffusion mapping is an MDS technique that belongs to the family of Laplacian-based spectral methods. The original scheme was introduced in the thesis work by Lafon (2004) and in Coifman et al. (2005a, 2005b). See also independent work by Fouss, Pirotte, and Saerens (2005) for a similar technique called Euclidean commute time (ECT) maps. In this paper, we will describe a slightly modified version of diffusion maps that appeared in (Coifman and Lafon 2006; Lafon and Lee 2006) (see *http://www.stat.cmu.edu/~annlee/software.htm* for example code in Matlab and R).

The starting point of the diffusion framework is to introduce a distance metric that reflects the higher-order connectivity of the data. This is done by defining a diffusion process or random walk on the data. In a graph approach, nodes of the graph represent the observations in the data set. Assuming nonnegative weights $\mathbb{K}$ and a degree matrix $\mathbb{M}$, we define a row-stochastic matrix $\mathbb{A} = \mathbb{M}^{-1}\mathbb{K}$. We then imagine a random walk on the graph $G = (V, E)$ where $\mathbb{A}$ is the transition matrix, and element $\mathbb{A}(u, v)$ corresponds to the probability of reaching node v from u in one step. Now if $\mathbb{A}^m$ is the mth matrix power of $\mathbb{A}$, then element $\mathbb{A}^m(u, v)$ can be interpreted as the probability of transition from u to v in m steps. By increasing m, we are running the Markov chain forward in time, thereby describing larger scale structures in the data set. Under certain conditions on $\mathbb{K}$, the Markov chain has a unique stationary distribution $s(v) = \rho_v / \sum_{u \in V} \rho_u$.

As in multidimensional scaling, the ultimate goal is to find an embedding of the data where Euclidean distances reflect similarities between points. In classical MDS, one attempts to preserve the original Euclidean distances $d^2(u, v) = \|x_u - x_v\|^2$ between points. In diffusion maps, the goal is to approximate diffusion distances defined by

$$D_m^2(u, v) = \sum_{k \in V} \frac{(\mathbb{A}^m(u, k) - \mathbb{A}^m(v, k))^2}{s(k)}.$$

This quantity captures the higher-order connectivity of the data at a scale m and is very robust to noise since it integrates multiple-step, multiple-path connections between points. The distance $D_m^2(u, v)$ is small when $\mathbb{A}^m(u, v)$ is large, or when there are many paths between nodes u and v in the graph. Furthermore, one can show (see Appendix A.1) that the optimal embedding in q dimensions is given by the eigenvectors of the Markov matrix $\mathbb{A}$. In fact, assuming the kernel matrix $\mathbb{K}$ is positive semi-definite, we have the "diffusion map"

$$v \in V \quad \mapsto$$
$$\Psi_m(v) = (\lambda_1^m \psi_1(v), \lambda_2^m \psi_2(v), \ldots, \lambda_q^m \psi_q(v)) \in \mathbb{R}^q,$$

where $\{\psi_\ell\}_{\ell \ge 0}$ are the principal eigenvectors of $\mathbb{A}$ and the eigenvalues $\lambda_0 = 1 \ge \lambda_1 \ge \cdots \ge 0$. This solution is, up to a rescaling of eigenvectors, the same as the solution of Laplacian eigenmaps and spectral clustering, since $\mathbb{L}\psi = \mu \mathbb{M} \psi$ if and only if $\mathbb{A}\psi = \lambda \psi$ for $\lambda = 1 - \mu$ and $\mathbb{L} = \mathbb{M} - \mathbb{K}$. The diffusion framework provides a link between Laplacian-based spectral methods, MDS and kernel PCA. It can be generalized to multiscale geometries (Coifman and Maggioni 2006), and other locality-preserving methods (Coifman and Lafon 2006).

3. DIFFUSION MAPS

Here we study the diffusion map under the assumption that the data are drawn from a general underlying distribution. By introducing a Markov chain, the method creates a distribution-sensitive data transformation.

3.1 A Discrete-Time Markov Chain

Definitions. Suppose that the data $X_1, \ldots, X_n$ are drawn from a distribution P with compact support $\mathcal{X} \subset \mathbb{R}^d$. We assume P has a density p with respect to Lebesgue measure μ. Let $k_\varepsilon(x, y) = \frac{1}{(4\pi\varepsilon)^{d/2}} \exp(-\frac{\|x-y\|^2}{4\varepsilon})$ denote the Gaussian kernel (Other kernels can be used. For simplicity, we will focus on the Gaussian kernel which is also the Green's function of the heat equation in $\mathbb{R}^d$.) with bandwidth $h = \sqrt{2\varepsilon}$. We write the bandwidth in terms of ε instead of h because ε is more natural for our purposes. Consider the Markov chain with transition kernel $\Omega_\varepsilon(x, \cdot)$ defined by

$$\Omega_\varepsilon(x, A) = \mathbb{P}(x \to A) = \frac{\int_A k_\varepsilon(x, y)\, dP(y)}{p_\varepsilon(x)}, \tag{4}$$

where $p_\varepsilon(x) = \int k_\varepsilon(x, y)\, dP(y)$.

Starting at x, this chain moves to points y close to x, giving preference to points with high density $p(y)$. In a sense, this chain measures the connectivity of the sample space relative to p. The stationary distribution S_ε is given by

$$S_\varepsilon(A) = \frac{\int_A p_\varepsilon(x)\, dP(x)}{\int p_\varepsilon(x)\, dP(x)}$$

and $S_\varepsilon(A) \to \frac{\int_A p(x)\, dP(x)}{\int p(x)\, dP(x)}$ as $\varepsilon \to 0$.

Define the densities $\omega_\varepsilon(x, y) = \frac{d\Omega_\varepsilon}{d\mu}(x, y) = \frac{k_\varepsilon(x,y) p(y)}{p_\varepsilon(x)}$ and

$$a_\varepsilon(x, y) = \frac{d\Omega_\varepsilon}{dP}(x, y) = \frac{k_\varepsilon(x, y)}{p_\epsilon(x)}.$$

The *diffusion operator* A_ε—which maps a function f to a new function $A_\varepsilon f$—is defined by

$$A_\varepsilon f(x) = \int a_\varepsilon(x, y) f(y)\, dP(y) = \frac{\int k_\varepsilon(x, y) f(y)\, dP(y)}{\int k_\varepsilon(x, y)\, dP(y)}. \tag{5}$$

We normalize the eigenfunctions $\{\psi_{\varepsilon,0}, \psi_{\varepsilon,1}, \ldots\}$ of A_ε by $\int \psi_{\varepsilon,\ell}^2(x) s_\varepsilon(x)\, dP(x) = 1$, where $s_\varepsilon(x) = \frac{p_\varepsilon(x)}{\int p_\varepsilon(y)\, dP(y)}$ is the density of the stationary distribution with respect to P. The first eigenfunction of the operator A_ε is $\psi_{\varepsilon,0}(x) = 1$ with eigenvalue $\lambda_{\varepsilon,0} = 1$. In general, the eigenfunctions have the following interpretation: $\psi_{\varepsilon,j}$ is the smoothest function relative to p, subject to being orthogonal to $\psi_{\varepsilon,i}$, $i < j$. The eigenfunctions form an efficient basis for expressing smoothness, relative to p. If a distribution has a few well-defined clusters then the first few eigenfunctions tend to behave like indicator functions (or combinations of indicator functions) for those clusters. The rest of the eigenfunctions provide smooth basis functions within each cluster. These smooth functions are Fourier-like. Indeed, the uniform distribution on the circle yields the usual Fourier basis. Figure 3 shows a density which is a mixture of two Gaussians. Also shown are the eigenvalues and the first 4 eigenfunctions which illustrate these features.

Denote the m-step transition measure by $\Omega_{\varepsilon,m}(x,\cdot)$. Let $A_{\varepsilon,m}$ be the corresponding diffusion operator which can be written as $A_{\varepsilon,m}f(x) = \int a_{\varepsilon,m}(x,y)f(y)\,dP(y)$ where $a_{\varepsilon,m}(x,y) = d\Omega_{\varepsilon,m}/dP$.

Define the empirical operator $\widehat{A}_\varepsilon$ by

$$\widehat{A}_\varepsilon f(x) = \frac{\sum_{i=1}^n k_\varepsilon(x,X_i)f(X_i)}{\sum_{i=1}^n k_\varepsilon(x,X_i)} = \int \widehat{a}_\varepsilon(x,y)f(y)\,d\widehat{P}_n(y), \tag{6}$$

where $\widehat{P}_n$ denotes the empirical distribution, $\widehat{a}_\varepsilon(x,y) = k_\varepsilon(x,y)/\widehat{p}_\varepsilon(x)$ and

$$\widehat{p}_\varepsilon(x) = \int k_\varepsilon(x,y)\,d\widehat{P}_n(y) = \frac{1}{n}\sum_{i=1}^n k_\varepsilon(x,X_i) \tag{7}$$

is the kernel density estimator. Let $\widehat{A}_{\varepsilon,m}$ be the corresponding m-step operator. Let $\widehat{\psi}_{\varepsilon,\ell}$ denote the eigenvectors of the matrix $\mathbb{A}_\varepsilon$ where $\mathbb{A}_\varepsilon(j,k) = k_\varepsilon(X_j,X_k)/\widehat{p}_\varepsilon(X_j)$. These eigenvectors are estimates of ψ_ℓ at the observed values $X_1,\ldots,X_n$. The function $\psi_\ell(x)$ can be estimated at values of x not corresponding to one of the X_i's by kernel smoothing as follows. The eigenfunction-eigenvalue equation $\lambda_{\varepsilon,\ell}\psi_{\varepsilon,\ell} = A_\varepsilon\psi_{\varepsilon,\ell}$ can be rearranged as

$$\psi_{\varepsilon,\ell}(x) = \frac{A_\varepsilon\psi_{\varepsilon,\ell}}{\lambda_{\varepsilon,\ell}} = \frac{\int k_\varepsilon(x,y)\psi_{\varepsilon,\ell}(y)\,dP(y)}{\lambda_{\varepsilon,\ell}\int k_\varepsilon(x,y)\,dP(y)} \tag{8}$$

suggesting the estimate

$$\widehat{\psi}_{\varepsilon,\ell}(x) = \frac{\sum_i k_\varepsilon(x,X_i)\widehat{\psi}_{\varepsilon,\ell}(X_i)}{\widehat{\lambda}_{\varepsilon,\ell}\sum_i k_\varepsilon(x,X_i)} \tag{9}$$

which is known in the applied mathematics literature as the Nyström approximation.

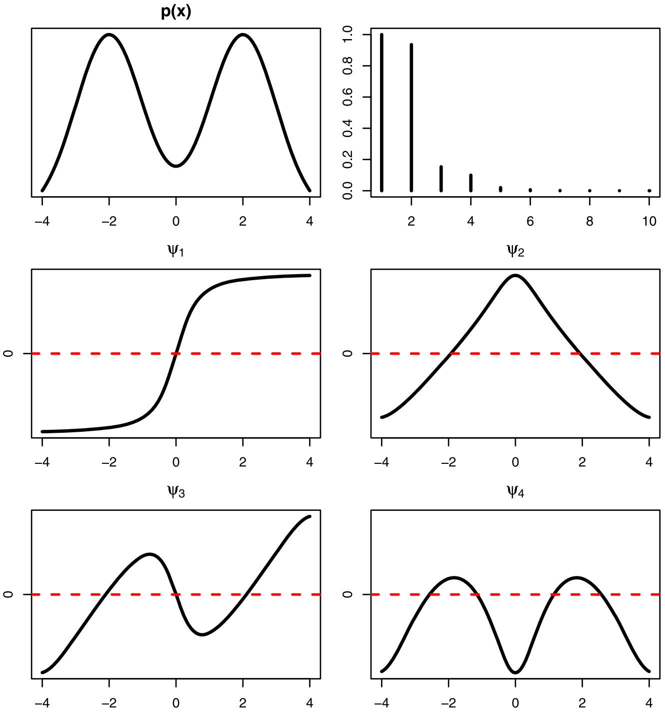

Figure 3. A mixture of two Gaussians. Density, eigenvalues, and first four eigenfunctions. The online version of this figure is in color.

Interpretation. The diffusion operators are averaging operators. Equation (5) arises in nonparametric regression. If we are given regression data $Y_i = f(X_i) + \epsilon_i$, $i = 1,\ldots,n$, then the kernel regression estimator of f is

$$\widehat{f}(x) = \frac{(1/n)\sum_{i=1}^n Y_i k_\varepsilon(x,X_i)}{(1/n)\sum_{i=1}^n k_\varepsilon(x,X_i)}. \tag{10}$$

Replacing the sample averages in (10) with their population averages yields (5). One may then wonder: in what way is spectral smoothing different from traditional nonparametric smoothing? There are at least three differences:

1. Estimating A_ε is an unsupervised problem, that is, there are no responses Y_i.
2. In spectral smoothing, we are interested in $\widehat{A}_{\varepsilon,m}$ for $m \geq 1$. The value $m = 1$ leads to a local analysis of the nearest-neighbor structure—this part is equivalent to classical smoothing. Powers $m > 1$, however, take *higher-order* structure into account. See Section 3.3 for the difference between smoothing by diffusion and smoothing by ε.
3. In spectral methods, smoothing is often not the end goal. The eigenvalues and eigenvectors of $\widehat{A}_\varepsilon$ provide information on the intrinsic geometry of the data and can be used to define a new coordinate system for the data.

The concept of connectivity is new in nonparametric statistics and is perhaps best explained in terms of stochastic processes. Introduce the forward Markov operator $M_\varepsilon g(x) = \int_{\mathcal{X}} a_\varepsilon(y,x)g(y)\,dP(y)$ and its m-step version $M_{\varepsilon,m}$. The first eigenfunction of M_ε is $\varphi_{\varepsilon,0}(x) = s_\varepsilon(x)$, the density of the stationary distribution. In general, $\varphi_{\varepsilon,\ell} = s_\varepsilon(x)\psi_{\varepsilon,\ell}(x)$. The averaging operator A_ε and the Markov operator M_ε and (and hence also the iterates $A_{\varepsilon,m}$ and $M_{\varepsilon,m}$) are adjoint under the inner product $\langle f,g\rangle = \int_{\mathcal{X}} f(x)g(x)\,dP(x)$, that is, $\langle A_\varepsilon f,g\rangle = \langle f, M_\varepsilon g\rangle$. By comparing the Gaussian kernel and the heat kernel of a continuous-time diffusion process [see equation (3.28) in Grigor'yan 2006], we identify the time step of the discrete system as $t = m\varepsilon$ for small ε.

The Markov operator $M_\varepsilon = A_\varepsilon^*$ maps measures into measures. That is, let $L_P^1(\mathcal{X}) = \{g : g(y) \geq 0, \int g(y)\,dP(y) = 1\}$. Then $g \in L_P^1(\mathcal{X})$ implies that $M_{\varepsilon,m}g \in L_P^1(\mathcal{X})$. In particular, if φ is the probability density at time $t = 0$, then $M_{\varepsilon,m}\varphi$ is the probability density after m steps. The averaging operator A_ε maps observables into observables. Its action is to compute conditional expectations. If $f \in L_P^\infty(\mathcal{X})$ is the test function (observable) at $t = 0$, then $A_{\varepsilon,m}f \in L_P^\infty(\mathcal{X})$ is the average of the function after m steps, that is, at a time comparable to $t = m\varepsilon$ for a continuous time system.

3.2 Continuous Time

Under appropriate regularity conditions, the eigenfunctions $\{\psi_{\varepsilon,\ell}\}$ converge to a set of functions $\{\psi_\ell\}$ as $\varepsilon \to 0$. These limiting eigenfunctions correspond to some operator. In this section we identify this operator. The key is to consider the Markov chain with infinitesimal transitions. In physics, local infinitesimal transitions of a system lead to global macroscopic descriptions by integration. Here we use the same tools (infinitesimal operators, generators, exponential maps, etc.) to extend short-time transitions to larger times.

Define the operator

$$G_\varepsilon f(x) = \frac{1}{\varepsilon}\left(\int_{\mathcal{X}} a_\varepsilon(x,y) f(y)\,dP(y) - f(x)\right). \tag{11}$$

Assume that the limit

$$\mathbf{G}f = \lim_{\varepsilon\to 0} G_\varepsilon f = \lim_{\varepsilon\to 0}\frac{A_\varepsilon f - f}{\varepsilon} \tag{12}$$

exists for all functions f in some appropriately defined space of functions $\mathcal{F}$. The operator $\mathbf{G}$ is known as the *infinitesimal generator*. A Taylor expansion shows that

$$\mathbf{G} = -\triangle + \frac{\nabla p}{p} \tag{13}$$

for smooth functions where $\triangle$ is the Laplacian and ∇ is the gradient. Indeed, $G_\varepsilon f = -\triangle f + \frac{\nabla p}{p} + O(\varepsilon)$ which is precisely the bias for kernel regression.

Remark 1. In kernel regression smoothing, the term $\nabla p/p$ is considered an undesirable extra bias, called design bias (Fan 1993). In regression it is removed by using local linear smoothing which is asymptotically equivalent to replacing the Gaussian kernel k_ε with a bias-reducing kernel k_ε^*. In this case, $\mathbf{G} = -\triangle$.

For $\ell > 0$ define $\nu_{\varepsilon,\ell}^2 = \frac{1-\lambda_{\varepsilon,\ell}}{\varepsilon}$ and $\nu_\ell^2 = \lim_{\varepsilon\to 0}\nu_{\varepsilon,\ell}^2$. The eigenvalues and eigenvectors of G_ε are $-\nu_{\varepsilon,\ell}^2$ and $\psi_{\varepsilon,\ell}$ while the eigenvalues and eigenvectors of the generator $\mathbf{G}$ are $-\nu_\ell^2$ and ψ_ℓ.

Let $\mathbf{A}_t = \lim_{\varepsilon\to 0} A_{\varepsilon,t/\varepsilon}$. From (11) and (12), it follows that

$$\begin{aligned}\mathbf{A}_t &\equiv \lim_{\varepsilon\to 0} A_{\varepsilon,t/\varepsilon} = \lim_{\varepsilon\to 0}(I + \varepsilon G_\varepsilon)^{t/\varepsilon}\\ &= \lim_{\varepsilon\to 0}(I + \varepsilon\mathbf{G})^{t/\varepsilon} = e^{\mathbf{G}t}.\end{aligned} \tag{14}$$

The family $\{\mathbf{A}_t\}_{t\ge 0}$ defines a continuous semigroup of operators (Lasota and Mackey 1994). The notation is summarized in Table 1.

Table 1. Summary of notation

Operator	Eigenfunctions	Eigenvalues
$A_\varepsilon f(\cdot) = \frac{\int k_\varepsilon(\cdot,y) f(y)\,dP(y)}{\int k_\varepsilon(\cdot,y)\,dP(y)}$	$\psi_{\varepsilon,\ell}$	$\lambda_{\varepsilon,\ell}$
$\mathbf{G} = \lim_{\varepsilon\to 0}\frac{A_\varepsilon - I}{\varepsilon}$	ψ_ℓ	$-\nu_\ell^2 = \lim_{\varepsilon\to 0}\frac{\lambda_{\varepsilon,\ell} - 1}{\varepsilon}$
$\mathbf{A}_t = e^{t\mathbf{G}} = \sum_{\ell=0}^{\infty} e^{-\nu_\ell^2 t}\Pi_\ell$ $= \lim_{\varepsilon\to 0} A_{\epsilon,t/\varepsilon}$	ψ_ℓ	$e^{-t\nu_\ell^2} = \lim_{\varepsilon\to 0}\lambda_{\varepsilon,\ell}^{t/\varepsilon}$

One of our goals is to find the bandwidth ε so that $\widehat{A}_{\varepsilon,t/\varepsilon}$ is a good estimate of $\mathbf{A}_t$. We show that this is a well-defined problem. Related work on manifold learning, on the other hand, only discusses the convergence properties of the graph Laplacian to the Laplace–Beltrami operator, that is, the generators of the diffusion. Estimating the generator $\mathbf{G}$, however, does not answer questions regarding the the number of eigenvectors, the number of groups in spectral clustering, etc.

We can express the diffusion in terms of its eigenfunctions. Mercer's theorem gives the biorthogonal decomposition $a_\varepsilon(x,y) = \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}\psi_{\varepsilon,\ell}(x)\varphi_{\varepsilon,\ell}(y)$ and

$$a_{\varepsilon,t/\varepsilon}(x,y) = \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}^{t/\varepsilon}\psi_{\varepsilon,\ell}(x)\varphi_{\varepsilon,\ell}(y), \tag{15}$$

where $\psi_{\varepsilon,\ell}$ are the eigenvectors of A_ε, and $\varphi_{\varepsilon,\ell}$ are the eigenvectors of its adjoint M_ε. The details are given in Appendix A.1. Note that $\{\psi_\ell\}$ is an orthonormal basis with respect to the inner product $\langle f, g\rangle_\varepsilon$, while $\{\varphi_\ell\}$ is an orthonormal basis with respect to $\langle f, g\rangle_{1/\varepsilon} = \int f(x)g(x)/s_\varepsilon(x)\,dP(x)$.

From (11), it follows that the eigenvalues $\lambda_{\varepsilon,\ell} = 1 - \varepsilon\nu_{\varepsilon,\ell}^2$. The averaging operator A_ε and its generator G_ε have the same eigenvectors. Inserting (15) into (5) and recalling that $\varphi_{\varepsilon,\ell}(x) = s_\varepsilon(x)\psi_{\varepsilon,\ell}(x)$, gives

$$\begin{aligned}A_\varepsilon f(x) &= \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}\psi_{\varepsilon,\ell}(x)\int_{\mathcal{X}}\varphi_{\varepsilon,\ell}(y) f(y)\,dP(y)\\ &= \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}\psi_{\varepsilon,\ell}(x)\langle\psi_{\varepsilon,\ell}, f\rangle_\varepsilon = \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}\Pi_{\varepsilon,\ell} f(x),\end{aligned}$$

where $\langle f, g\rangle_\varepsilon \equiv \int_{\mathcal{X}} f(y)g(y)s_\varepsilon(y)\,dP(y)$ and $\Pi_{\varepsilon,\ell}$ is the weighted orthogonal projector on the eigenspace spanned by $\psi_{\varepsilon,\ell}$. Thus, $A_{\varepsilon,t/\varepsilon} = \sum_{\ell\ge 0}\lambda_{\varepsilon,\ell}^{t/\varepsilon}\Pi_{\varepsilon,\ell}$. Similarly, assuming the limit in (14) exists, $\mathbf{A}_t = \sum_{\ell\ge 0} e^{-\nu_\ell^2 t}\Pi_\ell$ where Π_ℓ is the weighted orthogonal projector on the eigenspace corresponding to the eigenfunction ψ_ℓ of $\mathbf{G}$. Weyl's theorem (Stewart 1991) gives

$$\sup_\ell\left|e^{-\nu_\ell^2 t} - \lambda_{\varepsilon,\ell}^{t/\varepsilon}\right| \le \left\|A_{\varepsilon,t/\varepsilon} - e^{\mathbf{G}t}\right\| = t\varepsilon + O(\varepsilon^2),$$

$$\lim_{\varepsilon\to 0}\lambda_{\varepsilon,\ell}^{t/\varepsilon} = e^{-\nu_\ell^2 t},\quad \lim_{\varepsilon\to 0}\Pi_{\varepsilon,\ell} = \Pi_\ell. \tag{16}$$

To estimate the action of the limiting operator $\mathbf{A}_t$ at a given time $t > 0$, we need the dominant eigenvalues and eigenvectors of the generator $\mathbf{G}$. Finally, we also define the limiting transition density $\mathbf{a}_t(x,y) = \lim_{\varepsilon\to 0} a_{\varepsilon,t/\varepsilon}(x,y)$. As $t \to 0$, $\mathbf{a}_t(x,y)$ converges to a point mass at x; as $t \to \infty$, $\mathbf{a}_t(x,y)$ converges to $s(y)$.

Remark 2. There is an important difference between estimating $\mathbf{A}_t$ and $\mathbf{G}$: the diffusion operator $\mathbf{A}_t$ is a compact operator, while the generator $\mathbf{G}$ is not even a bounded operator.

We will consider some examples in Section 6 but let us first illustrate the definitions for a one-dimensional distribution with multiscale structure.

Example 1. Suppose that P is a mixture of three Gaussians. Figure 4 shows the density p. The left column of Figure 5 shows $\mathbf{A}_t$ for increasing t. The right column shows $\mathbf{a}_t(x,\cdot)$ for a fixed x indicated by the horizontal line. The density $\mathbf{a}_t(x,\cdot)$ starts out concentrated near x. As t increases, it begins to spread

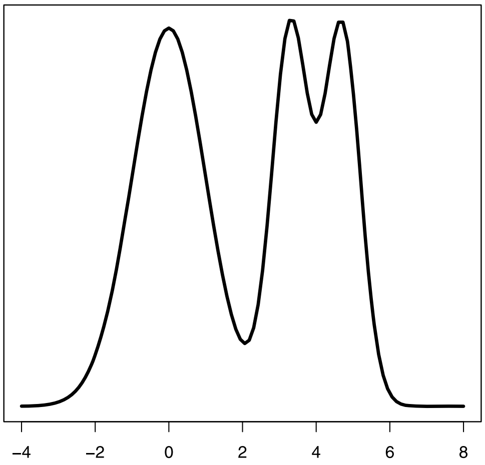

Figure 4. The density p for Example 1.

out. It becomes bimodal at $t = 1$ indicating that the two closer clusters have merged. Eventually, the density has three modes (indicating a single cluster) at $t = 10$, and then resembles p when $t = 1000$ since $\mathbf{a}_t(x, \cdot) \to p(\cdot)/\int p^2(u)\,du$ as $t \to \infty$.

3.3 Comparing ε and t

The parameters t and ε are both related to smoothing but they are quite different. The parameter t is part of the population quantity being estimated and controls the scale of the analysis. Hence, the choice of t is often determined by the problem at hand. The parameter ε is a smoothing parameter for estimating the population quantity from data. As $n \to \infty$, we let $\varepsilon_n \to 0$ for more accurate estimates. The following example illustrates the differences of smoothing in data when using ε versus t.

Example 2. Assume that the distribution is supported along two parallel lines of length π at $v = 0$ and $v = 1$, respectively, in a (u, v)-plane. The probability measure is $P = \frac{1}{2}U_0 + \frac{1}{2}U_1$ where U_0 is uniform on $\{(u, 0) : 0 \le u \le \pi\}$ and U_1 is uniform on $\{(u, 1) : 0 \le u \le \pi\}$. Consider a fixed test function f and an arbitrary point x. We have that

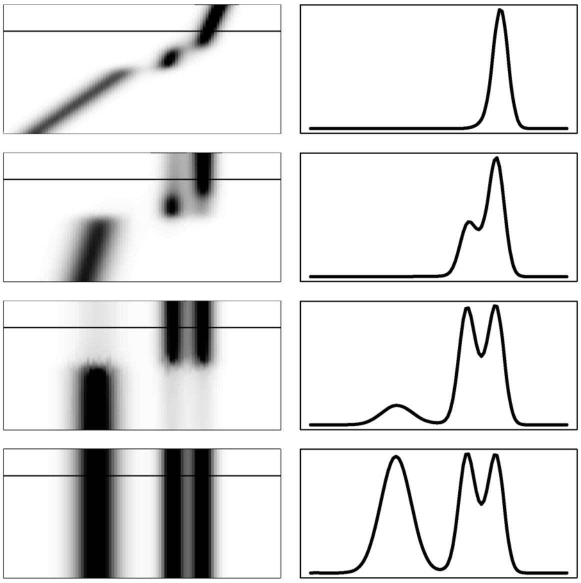

Figure 5. Example 1. Left column: $\mathbf{a}_t(x, y)$ for $t = 0.1, 1, 10, 1000$. Right column, $\mathbf{a}_t(x, y)$ for a fixed x.

$$A_\varepsilon f(x) = \int \omega_\varepsilon(x, y) f(y)\, dy,$$

$$\mathbf{A}_t f(x) = \int \omega_t(x, y) f(y)\, dy,$$

where the smoothing kernel $\omega_\varepsilon(x, y) = \frac{k_\varepsilon(x,y)p(y)}{p_\varepsilon(x)}$ and $\omega_t(x, y) = \mathbf{a}_t(x, y)p(y) = \lim_{\varepsilon \to 0} a_{\varepsilon, t/\varepsilon}(x, y)p(y)$.

Let $x = (0, 0)$ and $y = (u, v)$. Figure 6(a) shows how $\omega_\varepsilon(x, y)$ changes with the parameter ε. When ε is small, $A_\varepsilon f(x)$ only involves values of f close to the origin along the line at $v = 0$. However, with increasing ε, smoothing will also depend on function values further from the origin, including values along the parallel line at $v = 1$, as indicated by the dashed curves in the figure.

In contrast, for $x = (0, 0)$, $\mathbf{A}_t f(x)$ only involves values of f in the same connected set as x, that is, function values along the line at $v = 0$, regardless of t. Figure 6(b) illustrates how $\omega_t(x, y)$ changes as the parameter t increases. Smoothing by t reflects the *connectivity* of the data. In particular, there is no mixing of values of f from disconnected sets.

4. DIFFUSION DISTANCE

The diffusion distance is another key quantity that captures the underlying geometry of the data distribution. Unlike the geodesic distance, it is extremely robust to noise.

4.1 Definition

For an m-step Markov chain, the diffusion distances are defined by

$$D^2_{\varepsilon,m}(x, z) = \int \frac{(a_{\varepsilon,m}(x, u) - a_{\varepsilon,m}(z, u))^2}{s_\varepsilon(u)}\, dP(u)$$

for $m = 1, 2, \ldots$. It can be shown (see Appendix A.1) that $D^2_{\varepsilon,m}(x, z) = \sum_{\ell \ge 0} \lambda^{2m}_{\varepsilon,\ell}(\psi_{\varepsilon,\ell}(x) - \psi_{\varepsilon,\ell}(z))^2$. Following the same arguments as for $\widehat{A}_{\varepsilon,m}$ and $\mathbf{A}_t$, we deduce that the corresponding population quantity is $\mathbf{D}^2_t(x, z) = \sum_{\ell \ge 0} e^{-2\nu^2_\ell t} \times (\psi_\ell(x) - \psi_\ell(z))^2$.

4.2 Comparison to Geodesic Distance

The geodesic distance, or the shortest path, is an intuitive way of measuring the distance between two points in a set but, as shown here, it has several shortcomings for noisy data. Some manifold learning algorithms, such as Isomap (Tenenbaum, de Silva, and Langford 2000), rely on being able to estimate the geodesic distance on a manifold given data in $\mathbb{R}^p$. The idea is to construct a graph G on pairs of points at a distance less than a given threshold δ, and define a graph distance

$$d_G(A, B) = \min_\pi(\|A - x_1\| + \|x_1 - x_2\| + \cdots + \|x_m - B\|),$$

where $\pi = (A, x_1, x_2, \ldots, x_m, B)$ varies over all paths along the edges of G connecting A and B. Multidimensional scaling is then used to find a low-dimensional embedding of the data that best preserves these distances.

Under the assumption that the data lie exactly on a smooth manifold $\mathcal{M}$, Bernstein et al. (2000) have shown that the graph

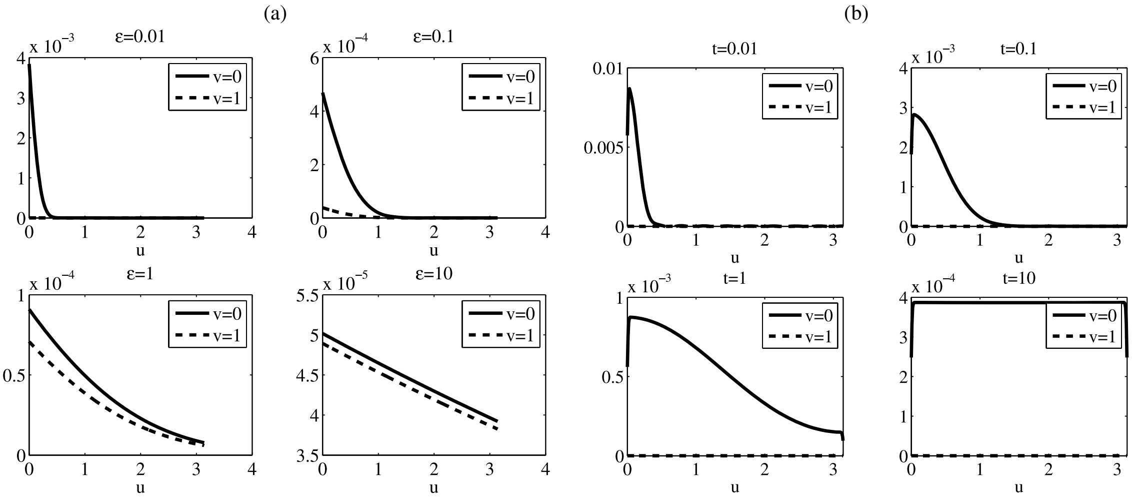

Figure 6. Smoothing for data along two parallel lines $v = 0$ (solid) and $v = 1$ (dashed). (a) $\omega_\varepsilon(x, y)$ for $x = (0, 0)$, $y = (u, v)$ and $\varepsilon = 0.01, 0.1, 1, 10$. (b) $\omega_t(x, y)$ for $x = (0, 0)$, $y = (u, v)$ and $t = 0.01, 0.1, 1, 10$.

distance $d_G(A, B)$ converges to the geodesic manifold metric $d_{\mathcal{M}}(A, B) = \inf\{\text{length}(\gamma)\}$, where γ varies over the set of smooth arcs connecting A and B in $\mathcal{M}$. Beyond this ideal situation, little is known about the statistical properties of the graph distance. Here we compare the graph distance and the diffusion metric for a data set where the support of the distribution is not exactly on a manifold. More specifically, consider a one-dimensional spiral in a plane:

$$\begin{cases} x = t^a \cos(bt) \\ y = t^a \sin(bt), \end{cases}$$

where $a = 0.8$ and $b = 10$. The geodesic manifold distance $d_{\mathcal{M}}(A, B)$ between two reference points A and B with $t = \pi/2b$ and $t = 5\pi/2b$, respectively, is 3.46. The corresponding Euclidean distance is 0.60.

Example 3 (Sensitivity to noise). We first generate 1000 instances of the spiral without noise (i.e., the data fall exactly on the spiral) and then 1000 instances of the spiral with exponential noise with mean parameter $\beta = 0.09$ added to both x and y. For each realization of the spiral, we construct a graph by connecting all pairs of points at a distance less than a threshold τ. The associated adjacency matrix has only zeros or ones, corresponding to the absence or presence of an edge, respectively.

Figure 7(a) shows histograms of the relative change in the geodesic graph distance (top) and the diffusion distance (bottom) when the data are perturbed. (The value 0 corresponds to no change from the average distance in the noiseless cases.) The sample size $n = 800$ and the neighborhood size $\tau = 0.15$. For the geodesic distance, we have a bimodal distribution with a large variance. The mode near -0.15 corresponds to cases where the shortest path between A and B approximately follows the branch of the spiral; see Figure 8 (left) for an example. The second mode around -0.75 occurs because some realizations of the noise give rise to shortcuts, which can dramatically reduce the length of the shortest path; see Figure 8 (right) for an example. The diffusion distance, on the other hand, is not sensitive to

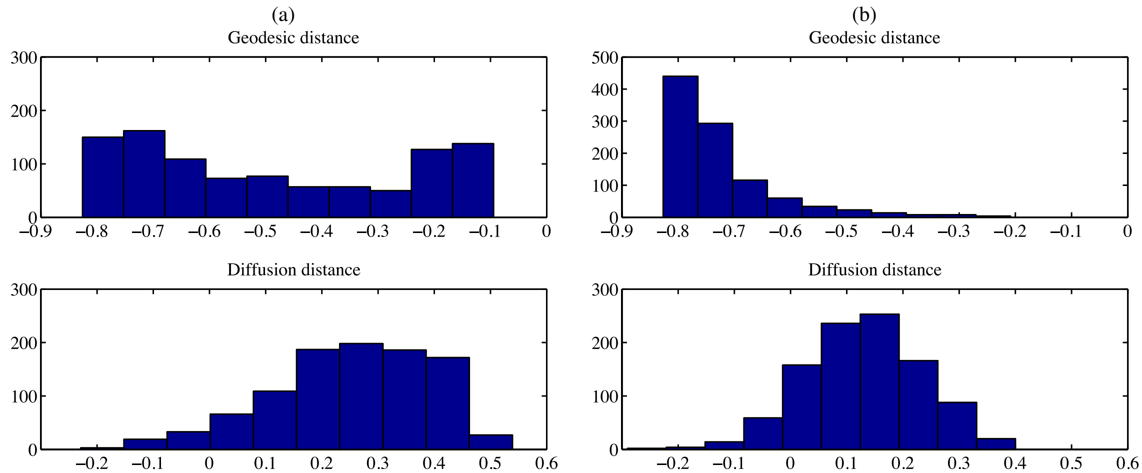

Figure 7. Sensitivity to noise. (a) Distribution of the geodesic, top, and diffusion distances, bottom, for a noisy spiral for $n = 800$ and $\tau = 0.15$. (b) Results for $n = 1600$ and $\tau = 0.15$. Each histogram has been shifted and rescaled so as to show the relative change from the noiseless case. The online version of this figure is in color.

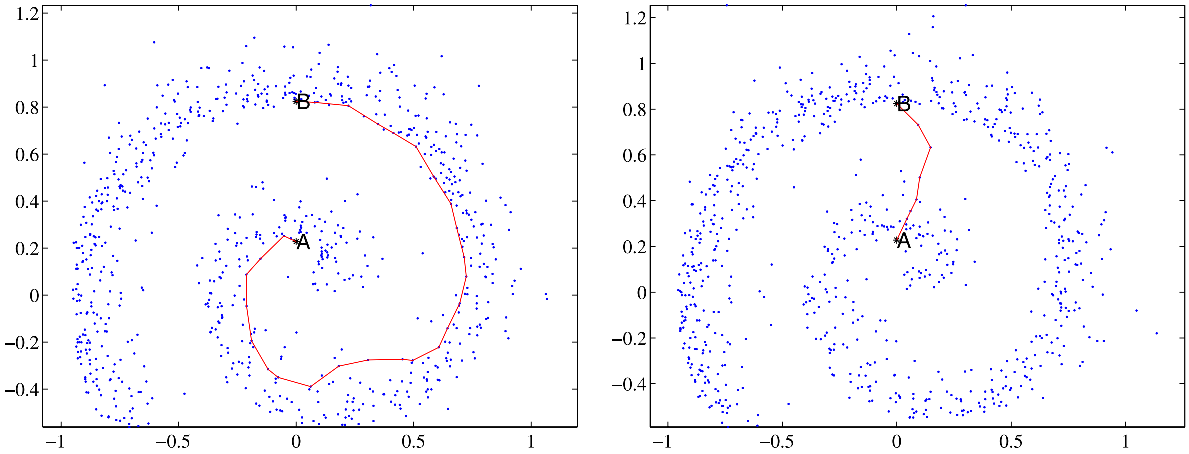

Figure 8. Two realizations of a noisy spiral. The solid line represents the shortest path between two reference points A and B in a graph constructed on the data.

small random perturbations of the data, because unlike the geodesic distance, it represents an average quantity. Shortcuts due to noise have little weight in the computation, as the number of such paths is much smaller than the number of paths following the shape of the spiral. This is also what our experiment confirms: Figure 7(a) (bottom) shows a unimodal distribution with about half the variance as for the geodesic distance.

Example 4 (Increasing sample size). A larger sample size increases the chance of shortcuts in the presence of noise for the geodesic distance. The diffusion distance, on the other hand, becomes more robust to noise with increasing sample sizes. This is illustrated in Figure 7(b) where $n = 1600$ but the other parameters (including the neighborhood size τ) are the same as in Example 3 and Figure 7(a). Choosing the right tuning parameters for a given dataset is in general a hard problem. The lack of robustness of the geodesic distance can lead to inconsistent results while this is of a lesser problem for the diffusion metric.

5. ESTIMATION

In Section 3, we identified $\mathbf{A}_t$ as the key population quantity in Laplacian-based spectral methods. Now we study the properties of $\widehat{A}_{\varepsilon,t/\varepsilon}$ as an estimator of $\mathbf{A}_t$. Let $\Pi_{\varepsilon,\ell}$ be the orthogonal projector onto the subspace spanned by $\psi_{\varepsilon,\ell}$ and let Π_ℓ be the projector onto the subspace spanned by ψ_ℓ. Consider the following operators:

$$A_t(\varepsilon, P) \equiv A_{\varepsilon,t/\varepsilon} = \sum_{\ell=0}^{\infty} \lambda_{\varepsilon,\ell}^{t/\varepsilon} \Pi_{\varepsilon,\ell},$$

$$A_t(\varepsilon, q, P) = \sum_{\ell=0}^{q} \lambda_{\varepsilon,\ell}^{t/\varepsilon} \Pi_{\varepsilon,\ell}, \qquad A_t(\varepsilon, q, \widehat{P}_n) = \sum_{\ell=0}^{q} \widehat{\lambda}_{\varepsilon,\ell}^{t/\varepsilon} \widehat{\Pi}_{\varepsilon,\ell},$$

$$\mathbf{A}_t = \sum_{\ell \geq 0} e^{-\nu_\ell^2 t} \Pi_\ell,$$

where $\psi_{\varepsilon,\ell}$ and $\lambda_{\varepsilon,\ell}$ denote the eigenfunctions and eigenvalues of A_ε, and $\widehat{\psi}_{\varepsilon,\ell}$ and $\widehat{\lambda}_{\varepsilon,\ell}$ are the eigenfunctions and eigenvalues of the data-based operator $\widehat{A}_\varepsilon$. Two estimators of $\mathbf{A}_t$ are the truncated estimator $A_t(\varepsilon, q, \widehat{P}_n)$ and the nontruncated estimator $A_t(\varepsilon, \widehat{P}_n) \equiv e^{t(\widehat{A}_\varepsilon - I)/\varepsilon}$. In practice, truncation is important since it corresponds to choosing a dimension for the transformed data.

5.1 Estimating the Diffusion Operator $\mathbf{A}_t$

Given data with a sample size n, we estimate $\mathbf{A}_t$ using a finite number q of eigenfunctions and a kernel bandwidth $\varepsilon > 0$. We define the loss function as

$$L_n(\varepsilon_n, q, t) = \|\mathbf{A}_t - A_t(\varepsilon_n, q, \widehat{P}_n)\|, \tag{17}$$

where $\|B\| = \sup_{f \in \mathcal{F}} \|Bf\|_2 / \|f\|_2$ and $\|f\|_2 = \sqrt{\int_{\mathcal{X}} f^2(x)\, dP(x)}$ where $\mathcal{F}$ is the set of uniformly bounded, three times differentiable functions with uniformly bounded derivatives whose gradients vanish at the boundary. By decomposing L_n into a bias-like and variance-like term, we derive the following result for the estimate based on truncation. Define $\rho(t) = \sum_{\ell=1}^{\infty} e^{-\nu_\ell^2 t}$.

Theorem 1. Suppose that P has compact support, and has bounded density p such that $\inf_x p(x) > 0$ and $\sup_x p(x) < \infty$. Let $\varepsilon_n \to 0$ and $n\varepsilon_n^{d/2} / \log(1/\varepsilon_n) \to \infty$. Then

$$L_n(\varepsilon_n, q, t) = \rho(t)(O_P(\gamma_n) + O(\varepsilon_n)) + C_n \sum_{q+1}^{\infty} e^{-\nu_\ell^2 t}, \tag{18}$$

where $C_n = O(1)$ and $\gamma_n = \sqrt{\frac{\log(1/\varepsilon_n)}{n\varepsilon_n^{(d+4)/2}}}$.

The optimal choice of ε_n is $\varepsilon_n \asymp (\log n/n)^{2/(d+8)}$ in which case

$$L_n(\varepsilon_n, q, t) = \rho(t) \cdot O_P\left(\frac{\log n}{n}\right)^{2/(d+8)} + C_n \sum_{q+1}^{\infty} e^{-\nu_\ell^2 t}. \tag{19}$$

We also have the following result which does not use truncation.

Theorem 2. Define $A_t(\varepsilon_n, \widehat{P}_n) = e^{t(\widehat{A}_{\varepsilon_n} - I)/\varepsilon_n}$. Then, under the same assumptions on P as in Theorem 1,

$$\|\mathbf{A}_t - A_t(\varepsilon_n, \widehat{P}_n)\| = (O_P(\gamma_n) + O(\varepsilon_n)) \cdot \rho(t). \tag{20}$$

The optimal ε_n is $\varepsilon_n \asymp (\log n/n)^{2/(d+8)}$. With this choice,

$$\|\mathbf{A}_t - A_t(\varepsilon_n, \widehat{P}_n)\| = O_P\left(\frac{\log n}{n}\right)^{2/(d+8)} \cdot \rho(t).$$

See Appendix A.2 for proofs.

Let us now make some remarks on the interpretation of these results.

1. The terms $O(\varepsilon_n)$ and $\sum_{q+1}^{\infty} e^{-\nu_\ell^2 t}$ correspond to bias. The term $O_P(\gamma_n)$ corresponds to the square root of the variance.
2. The rate $n^{-2/(d+8)}$ is slow. Indeed, the variance term $1/(n\varepsilon_n^{(d+4)/2})$ is the usual rate for estimating the second derivative of a regression function which is a notoriously difficult problem. This suggests that estimating $\mathbf{A}_t$ accurately is quite difficult.
3. We also have that $\|G_\varepsilon - \mathbf{G}\| = O_P(\gamma_n) + O(\varepsilon_n)$, and the first term is slower than the rate $1/\sqrt{n\varepsilon_n^{(d+2)/2}}$in Giné and Koltchinskii (2006) and Singer (2006) where d, in their case, is the intrinsic dimension of $\mathcal{X}$. The difference in rates is because they assume a uniform distribution. The slower rate comes from the term $p_\varepsilon(x) - \widehat{p}_\varepsilon(x)$ which cannot be ignored when p is unknown.
4. If the distribution is supported on a manifold of dimension $r < d$ then $\varepsilon^{(d+4)/2}$ becomes $\varepsilon^{(r+4)/2}$. In between full support and manifold support, one can create distributions that are concentrated near manifolds. That is, one first draws X_i from a distribution supported on a lower-dimensional manifold, then adds noise to X_i. This corresponds to full support again unless one lets the variance of the noise decrease with n. In that case, the exponent of ε can be between r and d.
5. Combining the above results with the result from Zwald and Blanchard (2006), we have that
$$\|\psi_\ell - \widehat{\psi}_{\varepsilon_n,\ell}\| = (O_P(\gamma_n) + O(\varepsilon_n)) \cdot \frac{1}{\min_{0\le j\le \ell}(\nu_j^2 - \nu_{j-1}^2)}.$$
6. The function $\rho(t)$ is decreasing in t. Hence for large t, the rate of convergence can be arbitrarily fast, even for large d. In particular, for $t \ge \rho^{-1}(n^{-(d+4)/(2(d+8))})$ the loss has the parametric rate $O_P(\sqrt{\log n/n})$.
7. An estimate of the diffusion distance is
$$\widehat{D}_t^2(x,y) = \sum_{\ell=0}^{q} \widehat{\lambda}_{\varepsilon,\ell}^{2t/\varepsilon}(\widehat{\psi}_{\varepsilon,\ell}(x) - \widehat{\psi}_{\varepsilon,\ell}(y))^2.$$
The approximation properties are similar to those of $\widehat{A}_t$.
8. The parameter $q = q_n$ should be chosen as small as possible for dimension reduction, while keeping the last term in (18) no bigger than the first term so as to minimize the loss function L_n. As illustrated below, the optimal q will depend on the smoothing parameter t, the decay rate of the eigenvalues ν_ℓ and the sample size n.

Example 5. Suppose that $\nu_\ell = \ell^\beta$ for some $\beta > 1/2$. Then
$$L_n(\varepsilon_n, q, t) = \frac{C_1}{t^{1/(2\beta)}} O_P\left(\frac{\log n}{n^{2/(d+8)}}\right) + C_2 e^{-tq^{2\beta}}.$$
The smallest q_n such that the last term in (18) does not dominate is
$$q_n \asymp \left(\left(\frac{1}{2\beta}\log t + \frac{2}{d+8}\log n\right)\Big/ t\right)^{1/(2\beta)}$$
and we get
$$L_n(\varepsilon_n, q, t) = O_P\left(\frac{1}{t^{1/(2\beta)}}\frac{\log n}{n^{2/(d+8)}}\right).$$

5.2 Nodal Domains and Low Noise

An eigenfunction ψ_ℓ partitions the sample space into regions where ψ_ℓ has constant sign. This partition is called the *nodal domain* of ψ_ℓ. In some sense, the nodal domain represents the basic structural information in the eigenfuction. In many applications, such as spectral clustering, it is sufficient to estimate the nodal domain rather than ψ_ℓ. We will show that fast rates are sometimes available for estimating the nodal domain even when the eigenfunctions are hard to estimate. This explains why spectral methods can be very successful despite the slow rates of convergence that we and others have obtained.

Formally, the nodal domain of ψ_ℓ is $N_\ell = \{C_1, \dots, C_k\}$ where the sets $C_1, \dots, C_k$ partition the sample space and the sign of ψ_ℓ is constant over each partition element C_j. Thus, estimating the nodal domain corresponds to estimating $H_\ell(x) = \mathrm{sign}(\psi_\ell(x))$. (If ψ is an eigenfunction then so is $-\psi$. We implicitly assume that the sign ambiguity of the eigenfunction has been removed.)

Recently, in the literature on classification, there has been a surge of research on the so-called "low noise" case. If the data have a low probability of being close to the decision boundary, then very fast rates of convergence are possible even in high dimensions. This theory explains the success of classification techniques in high-dimensional problems. In this section we show that a similar phenomema applies to spectral smoothing when estimating the nodal domain.

Inspired by the definition of low noise in Mammen and Tsybakov (1999), Audibert and Tsybakov (2007), and Kohler and Krzyzak (2007), we say that P has noise exponent α if there exists $C > 0$ such that
$$\mathbb{P}\big(|\psi_1(X)| \le \delta\big) \le C\delta^\alpha. \tag{21}$$
We are focusing here on ψ_1 although extensions to other eigenfunctions are immediate. Generally, as two clusters become well separated, ψ_1 behaves like a step function and $\mathbb{P}(0 < |\psi_1(X)| \le \delta)$ puts less and less mass near 0 which corresponds to α being large.

Theorem 3. Let $H(x) = \mathrm{sign}(\psi_1(x))$ and $\widehat{H}(x) = \mathrm{sign}(\widehat{\psi}_1(x))$. Suppose that (21) holds. Set $\varepsilon_n = n^{-2/(4\alpha+d+8)}$. Then,
$$\mathbb{P}(H(X) \ne \widehat{H}(X)) \le n^{-2\alpha/(4\alpha+8+d)}, \tag{22}$$
where $X \sim P$.

See Appendix A.2 for proof. Note that, as $\alpha \to \infty$ the rate tends to the parametric rate $n^{-1/2}$.

5.3 Choosing a Bandwidth

The theory we have developed gives insight into the behavior of the methods. But we are still left with the need for a practical method for choosing the smoothing parameter ε. The most common approach in the machine learning literature is to choose the smallest ε that makes the resulting graph well connected. More specifically, von Luxburg (2007) suggests to

"…choose ε [in an ε-neighborhood graph] as the length of the longest edge in a minimal spanning tree of the fully connected graph on the data points." Other methods for selecting ε include Hein, Audibert, and von Luxburg (2005a), Coifman et al. (2008), and Shi, Belkin, and Yu (2009). Instead of using a single global parameter ε, one can also vary ε over the dataset; see, for example, Zelnik-Manor and Perona (2004). The properties of the above approaches for bandwidth selection are however not fully understood.

Given the similarity between SCA and kernel smoothing, one may wonder if one can use methods for density estimation to choose ε. This turns out to be a nontrivial problem. In density estimation it is common to use the loss function $\int(p(x)-\widehat{p}_\varepsilon(x))^2\,dx$ which is equivalent, up to a constant, to $\mathcal{L}(\varepsilon)=\int\widehat{p}_\varepsilon^2(x)\,dx-2\int\widehat{p}_\varepsilon(x)p(x)\,dx$. A common method to approximate this loss is the cross-validation score $\widehat{\mathcal{L}}(\varepsilon)=\int\widehat{p}_\varepsilon^2(x)\,dx-\frac{2}{n}\sum_{i=1}^n\widehat{p}_{\varepsilon,i}(X_i)$ where $\widehat{p}_{\varepsilon,i}$ is the same as $\widehat{p}_\varepsilon$ except that X_i is omitted. It is well known that $\widehat{\mathcal{L}}(\varepsilon)$ is a nearly unbiased estimate of $\mathcal{L}(\varepsilon)$. One then chooses $\widetilde{\varepsilon}_n$ to minimize $\widehat{\mathcal{L}}(\varepsilon)$. The optimal ε_n^* from our earlier result, however, is (up to log factors) $O(n^{-2/(d+8)})$ while the optimal bandwidth ε_n^0 for minimizing $\mathcal{L}$ is $O(n^{-2/(d+4)})$. Hence, $\varepsilon_n^*/\varepsilon_n^0\asymp n^{8/((d+4)(d+8))}$. This suggests that density cross-validation is not appropriate for our purposes.

Estimating the risk is difficult in most problems that are not prediction problems. As usual in nonparametric inference, the problem is that estimating bias is harder than the original estimation problem. Hence, we take a more modest view of simply trying to find the smallest ε such that the resulting variability is tolerable. In other words, we choose the smallest ε that leads to stable estimates of the eigenstructure (similar to the approach for choosing the number of clusters in Lange et al. 2004). There are several ways to do this. Here are some examples.

Eigen-Stability. Define $\overline{\psi}_{\varepsilon,\ell}(x)=\mathbb{E}(\widehat{\psi}_{\varepsilon,\ell}(x))$. Although $\overline{\psi}_{\varepsilon,\ell}\neq\psi_\ell$, they do have a similar shape. We propose to choose ε by finding the smallest ε for which $\overline{\psi}_{\varepsilon,\ell}$ can be estimated with a tolerable variance. To this end we define

$$\mathrm{SNR}(\varepsilon)=\sqrt{\frac{\|\overline{\psi}_{\varepsilon,\ell}\|_2^2}{\mathbb{E}\|\widehat{\psi}_{\varepsilon,\ell}-\overline{\psi}_{\varepsilon,\ell}\|_2^2}} \tag{23}$$

which we will refer to as the signal-to-noise ratio. When ε is small, the denominator will dominate and $\mathrm{SNR}(\varepsilon)\approx 0$. Conversely, when ε is large, the denominator tends to 0 so that $\mathrm{SNR}(\varepsilon)$ gets very large. We want to find ε_0 such that

$$\varepsilon_0=\inf\{\varepsilon:\mathrm{SNR}(\varepsilon)\geq K_n\}$$

for some $K_n\geq 1$.

We can estimate SNR as follows. We compute B bootstrap replications $\widehat{\psi}_{\varepsilon,\ell}^{(1)},\ldots,\widehat{\psi}_{\varepsilon,\ell}^{(B)}$. We then take

$$\widehat{\mathrm{SNR}}(\varepsilon)=\sqrt{\frac{(\|\overline{\psi}_{\varepsilon,\ell}^*\|_2^2-\xi^2)_+}{\xi^2}}, \tag{24}$$

where $c_+=\max\{c,0\}$, $\xi^2=\frac{1}{B}\sum_{b=1}^B\|\widehat{\psi}_{\varepsilon,\ell}^{(b)}-\overline{\psi}_{\varepsilon,\ell}^*\|_2^2$, and $\overline{\psi}_{\varepsilon,\ell}^*=B^{-1}\sum_{b=1}^B\widehat{\psi}_{\varepsilon,\ell}^{(b)}$. Note that we subtract ξ^2 from the numerator to make the numerator approximately an unbiased estimator of $\|\overline{\psi}_{\varepsilon,\ell}\|^2$. Then we use

$$\widehat{\varepsilon}=\min\{\varepsilon:\widehat{\mathrm{SNR}}(\varepsilon)\geq K_n\}.$$

See the longer technical report (Lee and Wasserman 2008) for illustrations of the method. For $K_n=Cn^{2/(d+8)}$, where C is a constant, the optimal ε is $O(n^{-2/(d+8)})$. To see this, write $\widehat{\psi}_{\varepsilon,\ell}(x)=\psi_\ell(x)+b(x)+\xi(x)$ where $b(x)$ denotes the bias and $\xi(x)=\widehat{\psi}_{\varepsilon,\ell}(x)-\psi_\ell(x)-b(x)$ is the random component. Then $\mathrm{SNR}^2(\varepsilon)=\frac{\|\psi_\ell(x)+b(x)\|^2}{\mathbb{E}\|\xi\|^2}=\frac{O(1)}{O_P(1/(n\varepsilon^{(d+4)/2}))}$. Setting this equal to K_n^2 yields $\varepsilon_0=O(n^{-2/(d+8)})$.

The same bootstrap idea can be applied to estimating the nodal domain. In this case we define

$$\widehat{\mathrm{SNR}}(\varepsilon)=\sqrt{\frac{(\|\overline{H}_{\varepsilon,\ell}^*\|_2^2-\xi^2)_+}{\xi^2}}, \tag{25}$$

where $\xi^2=\frac{1}{B}\sum_{b=1}^B\|\widehat{H}_{\varepsilon,\ell}^{(b)}-\overline{H}_{\varepsilon,\ell}^*\|_2^2$, and $\overline{H}_{\varepsilon,\ell}^*=B^{-1}\times\sum_{b=1}^B\widehat{H}_{\varepsilon,\ell}^{(b)}$.

Neighborhood Size Stability. Another intuitive way to control the variability is to ensure that the number of points involved in the local averages does not get too small. For a given ε let $N=\{N_1,\ldots,N_n\}$ where $N_i=\#\{X_j:\|X_i-X_j\|\leq\sqrt{2\epsilon}\}$. One can informally examine the histogram of N for various ε. A rule for selecting ε is $\widehat{\varepsilon}=\min\{\varepsilon:\mathrm{median}\{N_1,\ldots,N_n\}\geq k\}$.

6. EXAMPLES

6.1 Mixture of Gaussians

We begin with a simple one-dimensional example to illustrate the different errors in the estimation of eigenvectors. Let

$$p(x)=\tfrac{1}{2}\phi(x;0,1)+\tfrac{1}{4}\phi(x;3.3,0.5)+\tfrac{1}{4}\phi(x;4.7,0.5),$$

where $\phi(x;\mu,\sigma)$ denotes a Normal density with mean μ and variance σ^2. Figure 4 shows the density. Figure 9 shows the error $\|\psi_1-\widehat{\psi}_{\varepsilon,1}\|$ as a function of ε for a sample of size $n=1500$. The results are averaged over approximately (we discard simulations where $\widehat{\lambda}_1=\widehat{\lambda}_0=1$ for $\varepsilon=0.02$) 200 independent draws. A minimal error occurs for a range of different values of ε between 0.02 and 0.06. The variance dominates the error in the small ε region ($\varepsilon<0.02$), while the bias dominates in the large ε region ($\varepsilon>0.06$). These results are consistent with Figure 10, which shows the estimated mean and variance of the first eigenvector $\widehat{\psi}_{\varepsilon,1}$ for a few selected values of ε ($\varepsilon=0.01,0.02,0.06,1$), marked with blue circles in Figure 9. Figure 11 shows similar results for the second eigenvector ψ_2. Note that even in cases where the error in the estimates of the eigenvectors is large, the variance around the cross-over points (where the eigenvectors switch signs) can be small. The results also agree with the conclusion in Section 5.2 that estimating the nodal domain of eigenvectors is a simpler problem than estimating the eigenvectors themselves.

6.2 Words

The next example is an application of SCA to text data mining (reproduced from Lafon and Lee 2006). The example shows how one can capture the semantic association of words with diffusion distances, and how one can organize and form representative "meta-words" by eigenanalysis and quantization of the diffusion operator.

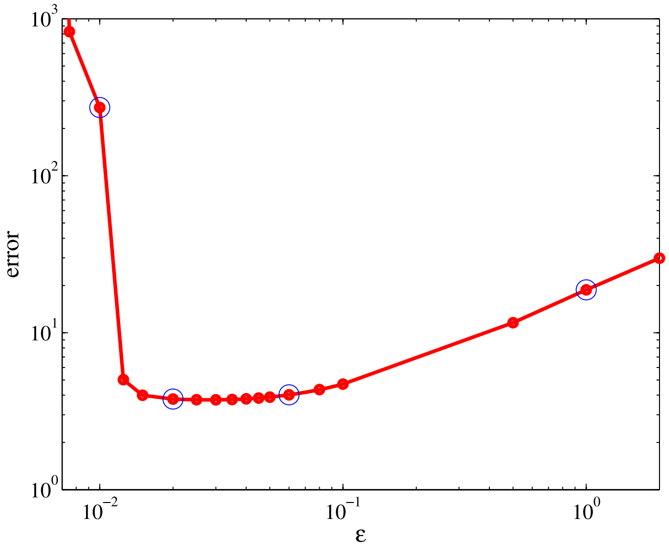

Figure 9. The error $\|\psi_1 - \widehat{\psi}_{\varepsilon,1}\|$ in the estimate of the first eigenvector as a function of ε. For each ε (red dots), an average is taken over approximately 200 independent simulations with $n = 1500$ points from a mixture distribution with two Gaussians. Figure 10 shows the estimated mean and variance of $\widehat{\psi}_{\varepsilon,1}$ for $\varepsilon = 0.01, 0.02, 0.06, 1$ (circles). The online version of this figure is in color.

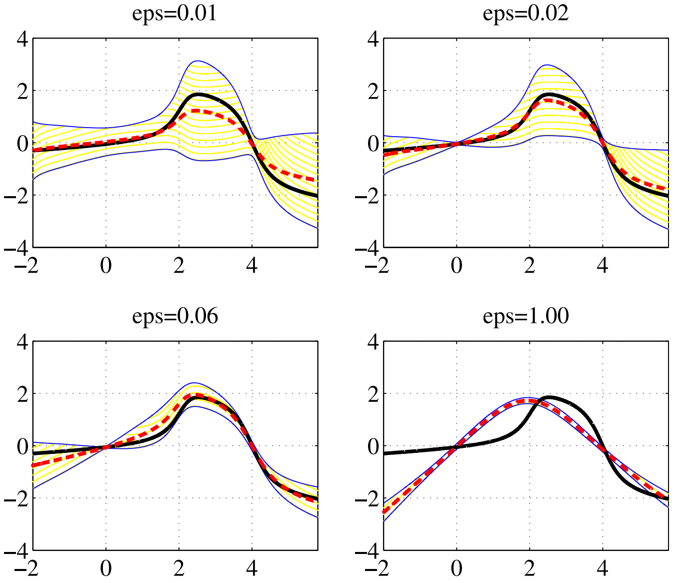

Figure 11. The second eigenvector $\widehat{\psi}_{\varepsilon,2}$ for $\varepsilon = 0.01, 0.02, 0.06, 1$, and $n = 1500$. The red dashed curves with shaded regions indicate the mean value ± two standard deviations for approximately 200 independent simulations. The black solid curves show $\psi_{\varepsilon,2}$ as $\varepsilon \to 0$.

The data consist of $p = 1161$ Science News articles. To encode the text, we extract $n = 1004$ words and form a document-word information matrix. The mutual information between document x and word y is defined as $I_{x,y} = \log(\frac{nc_{x,y}}{\sum_\xi c_{\xi,y} \sum_\eta c_{x,\eta}})$, where $c_{x,y}$ is the number of times word y appears in document x. Let $e_y = [I_{1,y}, I_{2,y}, \ldots, I_{p,y}]$ be a p-dimensional feature vector for word y.

Our goal is to reduce both the dimension p and the number of words n, while preserving the main connectivity structure of the data. In addition, we seek a coordinate system for the words that reflect how similar they are in meaning. Diffusion maps and quantization of the diffusion operator (diffusion coarse-graining) by k-means offer a natural framework for achieving these objectives.

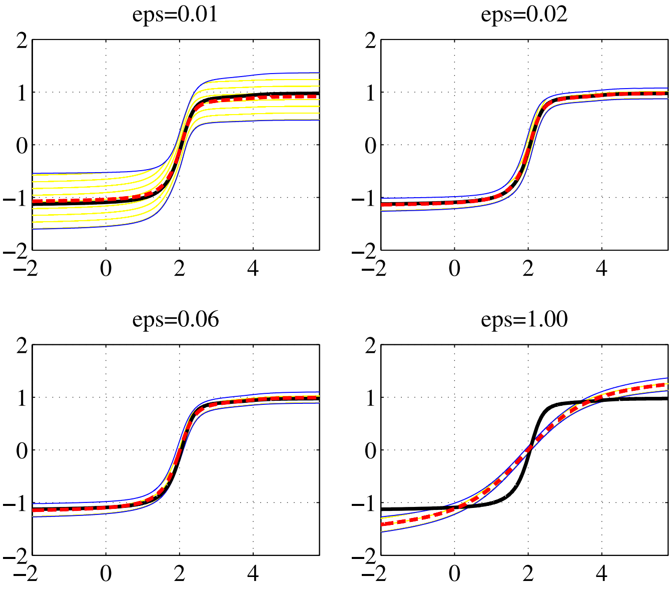

Figure 10. The first eigenvector $\widehat{\psi}_{\varepsilon,1}$ for $\varepsilon = 0.01, 0.02, 0.06, 1$, and $n = 1500$. The red dashed curves with shaded regions indicate the mean value ± two standard deviations for approximately 200 independent simulations. The black solid curves show $\psi_{\varepsilon,1}$ as $\varepsilon \to 0$.

We construct a graph where each node is a word and define the weight matrix by $\mathbb{K}(i,j) = \exp(-\frac{\|e_i - e_j\|^2}{4\varepsilon})$. Let $\mathbb{A}_{\varepsilon,m}$ be the corresponding m-step transition matrix with eigenvalues λ_ℓ^m and eigenvectors ψ_ℓ. Using the bootstrap, we estimate the SNR of ψ_1 as a function of ε. A SNR cut-off at 2, gives the bandwidth $\varepsilon = 150$. For $m = 3$, $q = 12$ and this choice of bandwidth, we have a spectral fall-off $(\lambda_q/\lambda_1)^m < 0.1$, that is, we can obtain a dimensionality reduction of a factor of about $1/100$ by the eigenmap $e_y \in \mathbb{R}^p \mapsto (\lambda_1^m \psi_1(y), \lambda_2^m \psi_2(y), \ldots, \lambda_q^m \psi_q(y)) \in \mathbb{R}^q$ without losing much accuracy. Further, to reduce the size of the dataset, we form a quantized matrix $\widetilde{\mathbb{A}}_{\varepsilon,m}$ for a coarse-grained random walk on a graph with $k < n$ nodes. It can be shown (Lafon and Lee 2006), that the spectral properties of $\mathbb{A}_{\varepsilon,m}$ and $\widetilde{\mathbb{A}}_{\varepsilon,m}$ are similar when the coarse-graining (quantization) corresponds to k-means clustering in diffusion space.

Figure 1 shows the first two diffusion coordinates of the cluster centers (the "meta-words") for $k = 100$. These representative words have roughly been rearranged according to their semantics and can be used as conceptual indices for document representation and text retrieval. Starting to the left, moving counter clockwise, we here have words that express concepts in medicine, biology, earth sciences, physics, astronomy, computer science, and social sciences. Table 2 gives examples of words in a cluster and the corresponding word centers.

7. DISCUSSION

Spectral methods are rapidly gaining popularity. Their ability to reveal nonlinear structure makes them ideal for complex, high-dimensional problems. We have attempted to provide insight into these techniques by identifying the key population

Table 2. Examples of word groupings

Word center	Remaining words in group
Virus	aids, allergy, hiv, vaccine, viral
Reproductive	fruit, male, offspring, reproductive, sex, sperm
Vitamin	calory, drinking, fda, sugar, supplement, vegetable
Fever	epidemic, lethal, outbreak, toxin
Ecosystem	ecologist, fish, forest, marine, river, soil, tropical
Warming	climate, el, nino, forecast, pacific, rain, weather, winter
Geologic	beneath, crust, depth, earthquake, plate, seismic, trapped, volcanic
Laser	atomic, beam, crystal, nanometer, optical, photon, pulse, quantum, semiconductor
Hubble	dust, gravitational, gravity, infrared
Galaxy	cosmic, universe
Finalist	award, competition, intel, prize, scholarship, student, talent, winner

quantities (namely the operators $\mathbf{A}_t$ and $\mathbf{D}_t$), and studying the large sample properties of their estimates.

Our analysis shows that spectral kernel methods in most cases have a convergence rate similar to classical nonparametric smoothing. Laplacian-based kernel methods, for example, use the same smoothing operators as in traditional nonparametric regression. The end goal however is not smoothing, but data transformation and structure definition of data. Spectral methods exploit the fact that the eigenvectors of local smoothing operators provide a coordinate system and information on the underlying geometry and connectivity of the data.

We close by giving examples of how SCA can be a powerful tool in high-dimensional "geometry-based" data analysis. Some of these applications (such as spectral clustering) are well known while others (such as sparse coding and high-dimensional density estimation via SCA) are new. The full details are reported in separate papers.

7.1 Clustering and Sparse Coding

One approach to clustering is spectral clustering (Ng, Jordan, and Weiss 2001; von Luxburg 2007). The idea is to transform the data using the first few nontrivial eigenvectors $\psi_1, \ldots, \psi_m$ and then apply a standard clustering algorithm such as k-means clustering. This approach can be quite effective for finding nonspherical clusters.

On a related note, the output from spectral clustering can be used for encoding of massive datasets. Consider a training set of signals $X_1, \ldots, X_n$ in $\mathbb{R}^p$. In classical sparse coding (Olshausen and Field 1997), one seeks a dictionary, that is, a set of basis vectors, D and vectors β_i to minimize an empirical cost function, typically of the form $R(D, \{\beta_i\}) = \sum_{i=1}^n (\|X_i - D\beta_i\|^2 + \lambda\|\beta_i\|_1)$, where λ is a regularization parameter. For complex data, however, the sparsity constraint on the coefficients β_i's and the assumption that $X_i \approx D\beta_i$ can be overly restrictive. Nonlinear geometries are also not taken into account.

An alternative approach to basis learning is to first transform the data via SCA, and then further quantize the data structure by a weighted k-means algorithm in the embedding space (Lafon and Lee 2006); see Section 6.2 for an application to words. The k centroids $\mathbf{c}_1, \ldots, \mathbf{c}_n$ form "prototypes" of the data. By construction, they capture the underlying geometry of the distribution P_X and form a more efficient covering of the data space. Richards et al. (2009a) use geometric prototyping to construct a basis of simple stellar population (SSP) spectra. For simulated galaxy spectra, such an approach to basis learning leads to more accurate estimation of star formation history than a hand-picked subset of SSP's (Cid Fernandes 2005; Asari et al. 2007) or bases derived from PCA or sparse methods.

7.2 Density Estimation

If Q is a quantization map then the quantized density estimator (Meinicke and Ritter 2002) is $\widehat{p}(x) = (1/n)\sum_{i=1}^n (1/h^d) \times K(\frac{\|x-Q(X_i)\|}{h})$. For highly clustered data, the quantized density estimator can have smaller mean squared error than the usual kernel density estimator. Similarly, we can define the quantized diffusion density estimator as

$$\widehat{p}(x) = \frac{1}{n}\sum_{i=1}^n \frac{1}{h^d} K\left(\frac{\widehat{D}_t(x, X_i)}{h}\right) \tag{26}$$

which can potentially have small mean squared error for appropriately chosen t.

SCA can be a powerful tool for high-dimensional density estimation problems where standard statistical methods are inadequate. Buchman, Lee, and Schafer (2010), for example, use the density estimator in Eq. (26) for modeling and simulation of hurricane tracks. In the analysis, a datum represents an entire hurricane trajectory (see Figure 12); densities are estimated from observed tracks in a lower-dimensional diffusion space; a random sample is drawn from the estimated diffusion density; the sample is finally mapped back into the original track space to simulate as-yet-unobserved tracks.

7.3 Regression

Incorporating data geometry can also radically improve regression and classification (Belkin and Niyogi 2005a; Lafferty and Wasserman 2007; Singh, Nowak, and Zhu 2008). A common method for nonparametric regression is to expand the regression function $m(x) = \mathbb{E}(Y|X = x)$ in a basis and then estimate the coefficients of the expansion from the data. Usually, the basis is chosen beforehand. The diffusion map basis provides a data-adaptive basis for doing nonparametric regression. We expand $m(x) = \mathbb{E}(Y|X = x)$ as $m(x) = \sum_j \beta_j \psi_j(x)$. Let $\widehat{m}(x) = \sum_{j=1}^q \widehat{\beta}_j \widehat{\psi}_{\varepsilon,j}(x)$ where q and ε are chosen by cross-validation. See (Richards et al. 2009b) and Freeman et al. (2009) for applications to redshift prediction of Sloan Digital Sky Survey (SDSS) data, and performance comparisons to PCA and template matching.

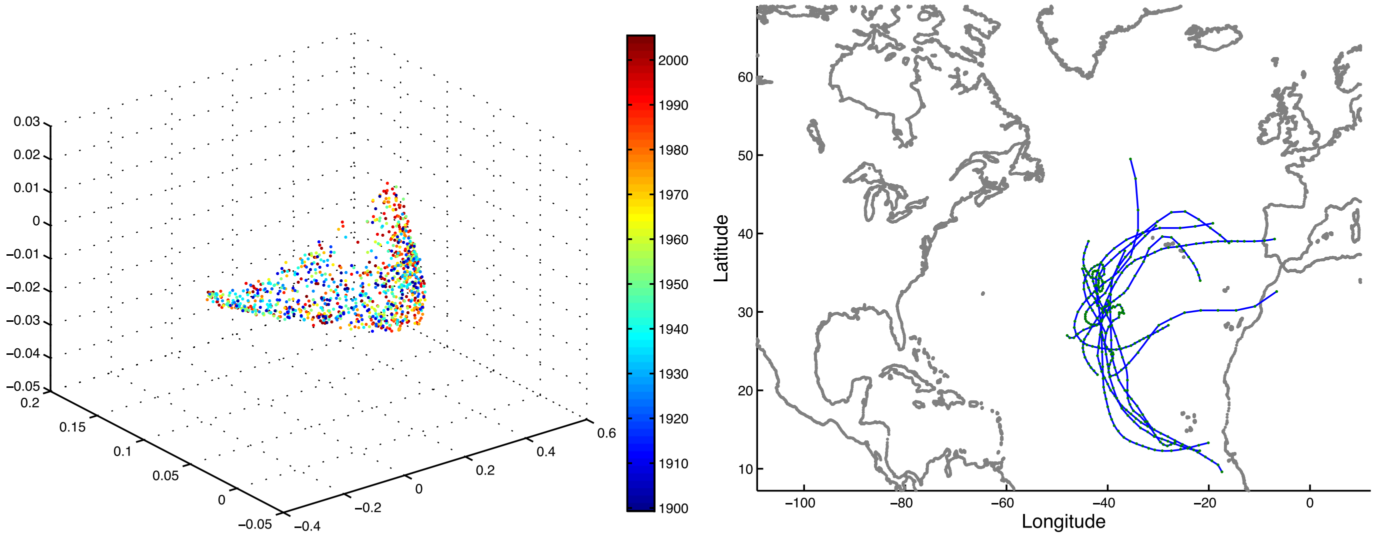

Figure 12. Left: Diffusion map of a set of 1000 Atlantic storm tracks. Right: Tracks of storms which are close to (0.39, 0.086, −0.0098) in diffusion space. Reproduced from Buchman, Lee, and Schafer (2010). The online version of this figure is in color.

SUPPLEMENTAL MATERIALS

Appendix: Two subsections in one pdf containing technical details and proofs. (appendix_rev.pdf)

[Received December 2009. Revised April 2010.]

REFERENCES

Asari, N. V., Cid Fernandes, R., Stasińska, G., Torres-Papaqui, J. P., Mateus, A., Sodré, L., Schoenell, W., and Gomes, J. M. (2007), "The History of Star-Forming Galaxies in the Sloan Digital Sky Survey," *Monthly Notices of the Royal Astronomical Society*, 381, 263–279. [1253]

Audibert, J.-Y., and Tsybakov, A. B. (2007), "Fast Learning Rates for Plug-in Classifiers," *The Annals of Statistics*, 35 (2), 608–633. [1250]

Belkin, M., and Niyogi, P. (2003), "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Computation*, 6 (15), 1373–1396. [1241,1243]

——— (2005a), "Semi-Supervised Learning on Riemannian Manifolds," *Machine Learning*, 56, 209–239. [1241,1253]

——— (2005b), "Towards a Theoretical Foundation for Laplacian-Based Manifold Methods," in *Proceedings of the Conference on Learning Theory*, Vol. 3559, New York: Springer, pp. 486–500. [1241,1244]

Bengio, Y., Delalleau, O., LeRoux, N., Paiement, J.-F., Vincent, P., and Ouimet, M. (2004), "Learning Eigenfunctions Links Spectral Embedding and Kernel PCA," *Neural Computation*, 16 (10), 2197–2219. [1243]

Bernstein, M., de Silva, V., Langford, J. C., and Tenenbaum, J. B. (2000), "Graph Approximations to Geodesics on Embedded Manifolds," technical report, Dept. of Mathematics, Stanford University. [1247]

Buchman, S. M., Lee, A. B., and Schafer, C. M. (2010), "High-Dimensional Density Estimation via SCA: An Example in the Modeling of Hurricane Tracks," *Statistical Methodology*, to appear. [1253,1254]

Cid Fernandes, R., Mateus, A., Sodré, L., Stasińska, G., and Gomes, J. M. (2005), "Semi-Empirical Analysis of Sloan Digital Sky Survey Galaxies—I. Spectral synthesis method," *Monthly Notices of the Royal Astronomical Society*, 358, 363–378. [1253]

Coifman, R., and Lafon, S. (2006), "Diffusion Maps," *Applied and Computational Harmonic Analysis*, 21, 5–30. [1241,1244]

Coifman, R., and Maggioni, M. (2006), "Diffusion Wavelets," *Applied and Computational Harmonic Analysis*, 21, 53–94. [1244]

Coifman, R., Lafon, S., Lee, A., Maggioni, M., Nadler, B., Warner, F., and Zucker, S. (2005a), "Geometric Diffusions as a Tool for Harmonics Analysis and Structure Definition of Data: Diffusion Maps," *Proceedings of the National Academy of Sciences*, 102 (21), 7426–7431. [1244]

——— (2005b), "Geometric Diffusions as a Tool for Harmonics Analysis and Structure Definition of Data: Multiscale Methods," *Proceedings of the National Academy of Sciences*, 102 (21), 7432–7437. [1244]

Coifman, R., Shkolnisky, Y., Sigworth, F. J., and Singer, A. (2008), "Graph Laplacian Tomography From Unknown Random Projections," *IEEE Transactions on Image Processing*, 17 (10), 1891–1899. [1251]

Donoho, D., and Grimes, C. (2003), "Hessian Eigenmaps: New Locally Linear Embedding Techniques for High-Dimensional Data," *Proceedings of the National Academy of Sciences*, 100 (10), 5591–5596. [1241]

Fan, J. (1993), "Local Linear Regression Smoothers and Their Minimax Efficiencies," *The Annals of Statistics*, 21, 196–216. [1246]

Fouss, F., Pirotte, A., and Saerens, M. (2005), "A Novel Way of Computing Similarities Between Nodes of a Graph, With Application to Collaborative Recommendation," in *Proceedings of the 2005 IEEE/WIC/ACM International Joint Conference on Web Intelligence*, Los Alamitos, CA: IEEE Computer Society, pp. 550–556. [1244]

Freeman, P. E., Newman, J., Lee, A. B., Richards, J. W., and Schafer, C. M. (2009), "Photometric Redshift Estimation Using SCA," *Monthly Notices of the Royal Astronomical Society*, 398, 2012–2021. [1253]

Giné, E., and Koltchinskii, V. (2006), "Empirical Graph Laplacian Approximation of Laplace–Beltrami Operators: Large Sample Results," in *High Dimensional Probability: Proceedings of the Fourth International Conference. IMS Lecture Notes*, Beachwood, OH: IMS, pp. 1–22. [1241,1244, 1250]

Grigor'yan, A. (2006), "Heat Kernels on Weighted Manifolds and Applications," *Contemporary Mathematics*, 398, 93–191. [1245]

Hastie, T., and Stuetzle, W. (1989), "Principal Curves," *Journal of the American Statistical Association*, 84, 502–516. [1243]

Hein, M., Audibert, J.-Y., and von Luxburg, U. (2005a), "From Graphs to Manifolds—Weak and Strong Pointwise Consistency of Graph Laplacians," in *Proceedings of the Conference on Learning Theory*, New York: Springer. [1251]

——— (2005b), "Intrinsic Dimensionality Estimation of Submanifolds in R^d," in *Proceedings of the 22nd International Conference on Machine Learning*, New York: ACM. [1241]

Kambhatla, N., and Leen, T. K. (1997), "Dimension Reduction by Local Principal Component Analysis," *Neural Computation*, 9, 1493–1516. [1243]

Kohler, M., and Krzyzak, A. (2007), "On the Rate of Convergence of Local Averaging Plug-in Classification Rules Under a Margin Condition," *IEEE Transactions on Information Theory*, 53, 1735–1742. [1250]

Lafferty, J., and Wasserman, L. (2007), "Statistical Analysis of Semi-Supervised Regression," in *Advances in Neural Information Processing Systems*, Cambridge, MA: MIT Press. [1253]

Lafon, S. (2004), "Diffusion Maps and Geometric Harmonics," Ph.D. thesis, Yale University. [1244]

Lafon, S., and Lee, A. (2006), "Diffusion Maps and Coarse-Graining: A Unified Framework for Dimensionality Reduction, Graph Partitioning, and Data Set Parameterization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28, 1393–1403. [1244,1251-1253]

Lange, T., Roth, V., Braun, M. L., and Buhmann, J. M. (2004), "Stability-Based Validation of Clustering Solutions," *Neural Computation*, 16 (6), 1299–1323. [1251]

Lasota, A., and Mackey, M. C. (1994), *Chaos, Fractals, and Noise: Stochastic Aspects of Dynamics* (2nd ed.), New York: Springer. [1246]

Lee, A. B., and Wasserman, L. (2008), "Spectral Connectivity Analysis," available at *arXiv:0811.0121*. [1251]

Levina, E., and Bickel, P. J. (2005), “Maximum Likelihood Estimation of Instrinsic Dimension,” in *Advances in Neural Information Processing Systems*, Vol. 17. [1241]

Mammen, E., and Tsybakov, A. B. (1999), “Smooth Discrimination Analysis,” *The Annals of Statistics*, 27, 1808–1829. [1250]

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1980), *Multivariate Analysis*, Academic Press. [1243]

Meinicke, P., and Ritter, H. (2002), “Quantizing Density Estimators,” in *Advances in Neural Information Processing Systems*, Vol. 14, Cambridge, MA: MIT Press, pp. 825–832. [1253]

Ng, A. Y., Jordan, M. I., and Weiss, Y. (2001), “On Spectral Clustering: Analysis and an Algorithm,” in *Advances in Neural Information Processing Systems*, Cambridge, MA: MIT Press. [1241,1253]

Olshausen, B. A., and Field, D. J. (1997), “Sparse Coding With an Overcomplete Basis Set: A Strategy Employed by V1?” *Vision Research*, 37, 3311–3325. [1253]

Page, L., Brin, S., Motwani, R., and Winograd, T. (1998), “The Pagerank Citation Ranking: Bringing Order to the Web,” technical report, Stanford University. [1241]

Richards, J. W., Freeman, P. E., Lee, A. B., and Schafer, C. M. (2009a), “Accurate Parameter Estimation for Star Formation History in Galaxies Using SDSS Spectra,” *Monthly Notices of the Royal Astronomical Society*, 399, 1044–1057. [1253]

——— (2009b), “Exploiting Low-Dimensional Structure in Astronomical Spectra,” *Astrophysical Journal*, 691, 32–42. [1242,1253]

Roweis, S., and Saul, L. (2000), “Nonlinear Dimensionality Reduction by Analsly Linear Embedding,” *Science*, 290, 2323–2326. [1241]

Schölkopf, B., Smola, A., and Müller, K.-R. (1998), “Nonlinear Component Analysis as a Kernel Eigenvalue Problem,” *Neural Computation*, 10 (5), 1299–1319. [1243]

Shi, T., Belkin, M., and Yu, B. (2009), “Data Spectroscopy: Eigenspaces of Convolution Operators and Clustering,” *The Annals of Statistics*, 37 (6B), 3960–3984. [1251]

Singer, A. (2006), “From Graph to Manifold Laplacian: The Convergence Rate,” *Applied and Computational Harmonic Analysis*, 21, 128–134. [1241, 1244,1250]

Singh, A., Nowak, R., and Zhu, X. (2008), “Unlabeled Data: Now It Helps, Now It Doesn’t,” in *Advances in Neural Information Processing Systems*, Red Hook, NY: Curran Associates, Inc. [1253]

Stewart, G. (1991), *Perturbation Theory for the Singular Value Decomposition*, *SVD and Signal Processing*, Vol. II, New York: Elsevier Science. [1246]

Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000), “A Global Geometric Framework for Nonlinear Dimensionality Reduction,” *Science*, 290 (5500), 2319–2323. [1247]

von Luxburg, U. (2007), “A Tutorial on Spectral Clustering,” *Statistics and Computing*, 17 (4), 395–416. [1241,1250,1253]

von Luxburg, U., Belkin, M., and Bousquet, O. (2008), “Consistency of Spectral Clustering,” *The Annals of Statistics*, 36 (2), 555–586. [1241]

Zelnik-Manor, L., and Perona, P. (2004), “Self-Tuning Spectral Clustering,” in *Advances in Neural Information Processing Systems*, Vol. 17, Cambridge, MA: MIT Press, pp. 1601–1608. [1251]

Zwald, L., and Blanchard, G. (2006), “On the Convergence of Eigenspaces in Kernel Principal Component Analysis,” in *Advances in Neural Information Processing Systems*, Vol. 18, Cambridge, MA: MIT Press, pp. 1649–1656. [1250]