Learning Eigenfunctions Links Spectral Embedding and Kernel PCA

Yoshua Bengio
bengioy@iro.umontreal.ca
Olivier Delalleau
delallea@iro.umontreal.ca
Nicolas Le Roux
lerouxni@iro.umontreal.ca
Jean-François Paiement
paiemeje@iro.umontreal.ca
Pascal Vincent
vincentp@iro.umontreal.ca
Marie Ouimet
ouimema@iro.umontreal.ca
Département d'Informatique et Recherche Opérationnelle, Centre de Recherches Mathématiques, Université de Montréal, Montréal, Québec, Canada, H3C 3J7

In this letter, we show a direct relation between spectral embedding methods and kernel principal components analysis and how both are special cases of a more general learning problem: learning the principal eigenfunctions of an operator defined from a kernel and the unknown data-generating density. Whereas spectral embedding methods provided only coordinates for the training points, the analysis justifies a simple extension to out-of-sample examples (the Nyström formula) for multidimensional scaling (MDS), spectral clustering, Laplacian eigenmaps, locally linear embedding (LLE), and Isomap. The analysis provides, for all such spectral embedding methods, the definition of a loss function, whose empirical average is minimized by the traditional algorithms. The asymptotic expected value of that loss defines a generalization performance and clarifies what these algorithms are trying to learn. Experiments with LLE, Isomap, spectral clustering, and MDS show that this out-of-sample embedding formula generalizes well, with a level of error comparable to the effect of small perturbations of the training set on the embedding.

1 Introduction

In the past few years, many unsupervised learning algorithms have been proposed that share the use of an eigendecomposition for obtaining a lower-dimensional embedding of the data that characterize a nonlinear manifold near which the data would lie: locally linear embedding (LLE) (Roweis &

Saul, 2000), Isomap (Tenenbaum, de Silva, & Langford, 2000), and Laplacian eigenmaps (Belkin & Niyogi, 2003). There are also many variants of spectral clustering (Weiss, 1999; Ng, Jordan, & Weiss, 2002) in which such an embedding is an intermediate step before obtaining a clustering of the data that can capture flat, elongated, and even curved clusters. The two tasks (manifold learning and clustering) are linked because the clusters that spectral clustering manages to capture can be arbitrary curved manifolds (as long as there are enough data to locally capture the curvature of the manifold). Clustering and manifold learning are intimately related: both clusters and manifold are zones of high density. All of these unsupervised learning methods are thus capturing salient features of the data distribution. As shown here, spectral clustering is in fact working in a way that is very similar to manifold learning algorithms.

The end result of most inductive machine learning algorithms is a function that minimizes the empirical average of a loss criterion (possibly plus regularization). The function can be applied on new points, and for such learning algorithms, it is clear that the ideal solution is a function that minimizes the expected loss under the unknown true distribution from which the data were sampled, also known as the generalization error. However, such a characterization was missing for spectral embedding algorithms such as metric multidimensional scaling (MDS) (Cox & Cox, 1994), spectral clustering (see Weiss, 1999, for a review), Laplacian eigenmaps, LLE, and Isomap, which are used for either dimensionality reduction or clustering. They do not provide a function that can be applied to new points, and the notion of generalization error is not clearly defined.

This article seeks to provide an answer to these questions. A loss criterion for spectral embedding algorithms can be defined. It is a reconstruction error that depends on pairs of examples. Minimizing its average value yields the eigenvectors that provide the classical output of these algorithms, that is, the embeddings. Minimizing its expected value over the true underlying distribution yields the eigenfunctions of a linear operator that is defined by a kernel (which is not necessarily positive semidefinite) and the data generating density. When the kernel is positive semidefinite and we work with the empirical density, there is a direct correspondence between these algorithms and kernel principal components analysis (PCA) (Schölkopf, Smola, & Müller, 1998). Our work is therefore a direct continuation of previous work (Williams & Seeger, 2000) noting that the Nyström formula and the kernel PCA projection (which are equivalent) represent an approximation of the eigenfunctions of the above linear operator (called G here). Previous analysis of the convergence of generalization error of kernel PCA (Shawe-Taylor, Cristianini, & Kandola, 2002; Shawe-Taylor & Williams, 2003) also helps to justify the view that these methods are estimating the convergent limit of some eigenvectors (at least when the kernel is positive semidefinite). The eigenvectors can thus be turned into estimators of eigenfunctions, which can therefore be applied to new points, turning the spectral embedding algorithms into function induction algorithms. The Nyström formula obtained this way is well known (Baker, 1977) and will be given in equation 1.1. This formula has been used previously for estimating extensions of eigenvectors in gaussian process regression (Williams & Seeger, 2001), and Williams and Seeger (2000) noted that it corresponds to the projection of a test point computed with kernel PCA.

In order to extend spectral embedding algorithms such as LLE and Isomap to out-of-sample examples, this article defines for these spectral embedding algorithms data-dependent kernels K_n that can be applied outside the training set. See also the independent work of Ham, Lee, Mika, and Schölkopf (2003) for a kernel view of LLE and Isomap, but where the kernels are only applied on the training set.

Additional contributions of this article include a characterization of the empirically estimated eigenfunctions in terms of eigenvectors in the case where the kernel is not positive semidefinite (often the case for MDS and Isomap), a convergence theorem linking the Nyström formula to the eigenfunctions of G, as well as experiments on MDS, Isomap, LLE and spectral clustering and Laplacian eigenmaps showing that the Nyström formula for out-of-sample examples is accurate.

All of the algorithms described in this article start from a data set $D = (x_1, \ldots, x_n)$ with $x_i \in \mathbb{R}^d$ sampled independently and identically distributed (i.i.d.) from an unknown distribution with density p. Below we will use the notation

$$E_x[f(x)] = \int f(x)p(x)dx$$

for averaging over $p(x)$ and

$$\hat{E}_x[f(x)] = \frac{1}{n}\sum_{i=1}^{n} f(x_i)$$

for averaging over the data in D, that is, over the empirical distribution denoted $\hat{p}(x)$. We will denote kernels with $K_n(x, y)$ or $\tilde{K}(x, y)$, symmetric functions, not always positive semidefinite, that may depend not only on x and y but also on the data D. The spectral embedding algorithms construct an affinity matrix M either explicitly through $M_{ij} = K_n(x_i, x_j)$ or implicitly through a procedure that takes the data D, and computes M. We denote by v_{ik} the ith coordinate of the kth eigenvector of M, associated with the eigenvalue ℓ_k. With these notations, the Nyström formula discussed above can be written

$$f_{k,n}(x) = \frac{\sqrt{n}}{\ell_k}\sum_{i=1}^{n} v_{ik}K_n(x, x_i), \tag{1.1}$$

where $f_{k,n}$ is the kth Nyström estimator with n samples. We will show in section 4 that it estimates the kth eigenfunction of a linear operator.

2 Kernel Principal Component Analysis

Kernel PCA is an unsupervised manifold learning technique that maps data points to a generally lower-dimensional space. It generalizes the principal component analysis approach to nonlinear transformations using the kernel trick (Schölkopf, Smola, & Müller, 1996; Schölkopf et al., 1998; Schölkopf, Burges, & Smola, 1999). The algorithm implicitly finds the leading eigenvectors and eigenvalues of the covariance of the projection $\phi(x)$ of the data in feature space, where $\phi(x)$ is such that the kernel $K_n(x, y) = \phi(x) \cdot \phi(y)$ (i.e., K_n must not have negative eigenvalues). If the data are centered in feature space ($\hat{E}_x[\phi(x)] = 0$), the feature space covariance matrix is

$$C = \hat{E}_x[\phi(x)\phi(x)'], \tag{2.1}$$

with eigenvectors w_k and eigenvalues λ_k. To obtain a centered feature space, a kernel $\tilde{K}$ (e.g., the gaussian kernel) is first normalized into a data-dependent kernel K_n via

$$K_n(x, y) = \tilde{K}(x, y) - \hat{E}_{x'}[\tilde{K}(x', y)] - \hat{E}_{y'}[\tilde{K}(x, y')] + \hat{E}_{x'}[\hat{E}_{y'}[\tilde{K}(x', y')]]. \tag{2.2}$$

The eigendecomposition of the corresponding Gram matrix M is performed, solving $Mv_k = \ell_k v_k$, as with the other spectral embedding methods (Laplacian eigenmaps, LLE, Isomap, MDS). However, in this case, one can obtain a test point projection $P_k(x)$ that is the inner product of $\phi(x)$ with the eigenvector w_k of C, and using the kernel trick, it is written as the expansion

$$P_k(x) = w_k \cdot \phi(x) = \frac{1}{\sqrt{\ell_k}} \sum_{i=1}^{n} v_{ik} K_n(x_i, x). \tag{2.3}$$

Note that the eigenvectors of C are related to the eigenvectors of M through $\lambda_k = \ell_k/n$ and

$$w_k = \frac{1}{\sqrt{\ell_k}} \sum_{i=1}^{n} v_{ik} \phi(x_i),$$

as shown in Schölkopf et al. (1998).

Ng et al. (2002) already noted the link between kernel PCA and spectral clustering. Here we take advantage of that link to propose and analyze out-of-sample extensions for spectral clustering and other spectral embedding algorithms. Recently Ham et al. (2003) showed how Isomap, LLE, and Laplacian eigenmaps can be interpreted as performing a form of kernel PCA, although they do not propose to use that link in order to perform function induction (i.e., obtain an embedding for out-of-sample points).

Recent work has shown convergence properties of kernel PCA that are particularly important here. Shawe-Taylor et al. (2002) & Shawe-Taylor and

Williams (2003) give bounds on the kernel PCA convergence error (in the sense of the projection error with respect to the subspace spanned by the eigenvectors), using concentration inequalities.

3 Data-Dependent Kernels for Spectral Embedding Algorithms

The spectral embedding algorithms can be seen to build an $n \times n$ similarity matrix M, compute its principal eigenvectors $v_k = (v_{1k}, \ldots, v_{nk})'$ with eigenvalues ℓ_k, and associate with the ith training example the embedding with coordinates $(v_{i1}, v_{i2}, \ldots)$ (for Laplacian eigenmaps and LLE)[1] or $(\sqrt{\ell_1}v_{i1}, \sqrt{\ell_2}v_{i2}, \ldots)$ (for Isomap and MDS). In general, we will see that M_{ij} depends not only on (x_i, x_j) but also on the other training examples. Nonetheless, as we show below, it can always be written $M_{ij} = K_n(x_i, x_j)$ where K_n is a data-dependent kernel. In many algorithms, a matrix $\tilde{M}$ is first formed from a simpler, often data-independent kernel (such as the gaussian kernel) and then transformed into M. By defining a kernel K_n for each of these methods, we will be able to generalize the embedding to new points x outside the training set via the Nyström formula. This will only require computations of the form $K_n(x, x_i)$ with x_i a training point.

3.1 Multidimensional Scaling. Metric MDS (Cox & Cox, 1994) starts from a notion of distance $d(x, y)$ that is computed between each pair of training examples to fill a matrix $\tilde{M}_{ij} = d^2(x_i, x_j)$. These distances are then converted to equivalent dot products using the double-centering formula, which makes M_{ij} depend not only on (x_i, x_j) but also on all the other examples:

$$M_{ij} = -\frac{1}{2}\left(\tilde{M}_{ij} - \frac{1}{n}S_i - \frac{1}{n}S_j + \frac{1}{n^2}\sum_k S_k\right), \tag{3.1}$$

where $S_i = \sum_{j=1}^n \tilde{M}_{ij}$. The embedding of the example x_i is given by $\sqrt{\ell_k}v_{ik}$ where $v_{\cdot k}$ is the kth eigenvector of M.

To generalize MDS, we define a corresponding data-dependent kernel that generates the matrix M,

$$K_n(a, b) = -\frac{1}{2}(d^2(a, b) - \hat{E}_x[d^2(x, b)] - \hat{E}_{x'}[d^2(a, x')] + \hat{E}_{x,x'}[d^2(x, x')]), \tag{3.2}$$

where the expectations are taken on the training set D. An extension of metric MDS to new points has already been proposed in Gower (1968), in which

[1] For Laplacian eigenmaps and LLE, the matrix M discussed here is not the one defined in the original algorithms, but a transformation of it to reverse the order of eigenvalues, as we see below.

one solves exactly for the coordinates of the new point that are consistent with its distances to the training points, which in general requires adding a new dimension. Note also that Williams (2001) makes a connection between kernel PCA and metric MDS, remarking that kernel PCA is a form of MDS when the kernel is isotropic. Here we pursue this connection in order to obtain out-of-sample embeddings.

3.2 Spectral Clustering. Several variants of spectral clustering have been proposed (Weiss, 1999). They can yield impressively good results where traditional clustering looking for "round blobs" in the data, such as K-means, would fail miserably (see Figure 1). It is based on two main steps: first embedding the data points in a space in which clusters are more "obvious" (using the eigenvectors of a Gram matrix) and then applying a classical clustering algorithm such as K-means, as in Ng et al. (2002). To construct the the spectral clustering affinity matrix M, we first apply a data-independent kernel $\tilde{K}$ such as the gaussian kernel to each pair of examples: $\tilde{M}_{ij} = \tilde{K}(x_i, x_j)$. The matrix $\tilde{M}$ is then normalized, for example, using divisive normalization (Weiss, 1999; Ng et al., 2002):[2]

$$M_{ij} = \frac{\tilde{M}_{ij}}{\sqrt{S_i S_j}}. \tag{3.3}$$

To obtain m clusters, the first m principal eigenvectors of M are computed, and K-means is applied on the unit-norm coordinates, obtained from the embedding v_{ik} of each training point x_i.

To generalize spectral clustering to out-of-sample points, we define a kernel that could have generated that matrix:

$$K_n(a, b) = \frac{1}{n} \frac{\tilde{K}(a, b)}{\sqrt{\hat{E}_x[\tilde{K}(a, x)]\hat{E}_{x'}[\tilde{K}(x', b)]}}. \tag{3.4}$$

This normalization comes out of the justification of spectral clustering as a relaxed statement of the min-cut problem (Chung, 1997; Spielman & Teng, 1996) (to divide the examples into two groups such as to minimize the sum of the "similarities" between pairs of points straddling the two groups). The additive normalization performed with kernel PCA (see equation 2.2) makes sense geometrically as a centering in feature space. Both normalization procedures make use of a kernel row and column average. It would be interesting to find a similarly pleasing geometric interpretation to the divisive normalization.

[2] Better embeddings are usually obtained if we define $S_i = \sum_{j \neq i} \tilde{M}_{ij}$. This alternative normalization can also be cast into the general framework developed here, with a slightly different kernel.

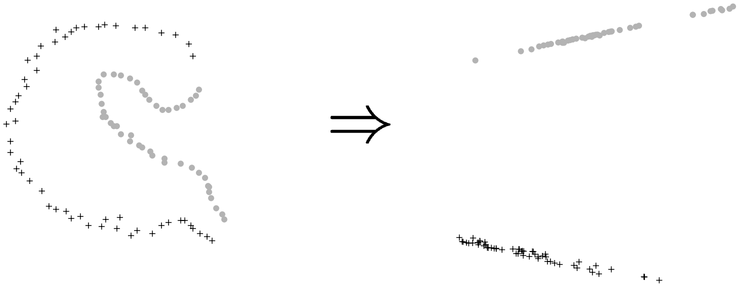

Figure 1: Example of the transformation learned as part of spectral clustering. Input data are on the left and transformed data on the right. Gray level and cross and circle drawing are used only to show which points get mapped where: the mapping reveals both the clusters and the internal structure of the two manifolds.

3.3 Laplacian Eigenmaps. The Laplacian eigenmaps method is a recently proposed dimensionality-reduction procedure (Belkin & Niyogi, 2003) that was found to be very successful for semisupervised learning. The authors use an approximation of the Laplacian operator such as the gaussian kernel or the k-nearest-neighbor graph: the symmetric matrix whose element (i, j) is 1 if x_i and x_j are k-nearest-neighbors (x_i is among the k nearest neighbors of x_j or vice versa) and 0 otherwise. Instead of solving an ordinary eigenproblem, the following generalized eigenproblem is solved:

$$(S - \tilde{M})y_k = \sigma_k S y_k, \tag{3.5}$$

with eigenvalues σ_k, eigenvectors y_k, and S the diagonal matrix with elements S_i previously defined (row sums of $\tilde{M}$). The smallest eigenvalue is left out, and the eigenvectors corresponding to the other small eigenvalues are used for the embedding. This is actually the same embedding that is computed with the spectral clustering algorithm from (Shi & Malik, 1997). As noted in Weiss (1999) (normalization lemma 1), an equivalent result (up to a component-wise scaling of the embedding) can be obtained by considering the principal eigenvectors v_k of the normalized matrix M defined in equation 3.3. To fit the common framework for spectral embedding in this article, we have used the latter formulation. Therefore, the same data-dependent kernel can be defined as for spectral clustering, equation 3.4, to generate the matrix M; that is, spectral clustering adds a clustering step after a Laplacian eigenmap dimensionality reduction.

3.4 Isomap. Isomap (Tenenbaum et al., 2000) generalizes MDS to nonlinear manifolds. It is based on replacing the Euclidean distance by an empirical approximation of the geodesic distance on the manifold. We define the geodesic distance $\mathcal{D}(\cdot,\cdot)$ with respect to a data set D, a distance $d(\cdot,\cdot)$, and a neighborhood k as follows:

$$\mathcal{D}(a,b) = \min_{\pi} \sum_{i=1}^{|\pi|} d(\pi_i, \pi_{i+1}), \tag{3.6}$$

where π is a sequence of points of length $|\pi| = l \geq 2$ with $\pi_1 = a$, $\pi_l = b$, $\pi_i \in D\ \forall i \in \{2,\ldots,l-1\}$ and (π_i,π_{i+1}) are k-nearest-neighbors of each other. The length $|\pi| = l$ is free in the minimization. The Isomap algorithm obtains the normalized matrix M from which the embedding is derived by transforming the raw pairwise distances matrix as follows: (1) compute the matrix $\tilde{M}_{ij} = \mathcal{D}^2(x_i, x_j)$ of squared geodesic distances with respect to the data D and (2) apply to this matrix the distance-to-dot-product transformation, equation 3.1, as for MDS. As in MDS, the embedding of x_i is $\sqrt{\ell_k} v_{ik}$ rather than v_{ik}. There are several ways to define a kernel that generates M and also generalizes out-of-sample. The solution we have chosen simply computes the geodesic distances without involving the out-of-sample point(s) along the geodesic distance sequence (except possibly at the extreme). This is automatically achieved with the above definition of geodesic distance $\mathcal{D}$, which uses only the training points to find the shortest path between a and b. The double-centering kernel transformation of equation 3.2 can then be applied, using the geodesic distance $\mathcal{D}$ instead of the MDS distance d.

A formula has been proposed (de Silva & Tenenbaum, 2003) to approximate Isomap using only a subset of the examples (the "landmark" points) to compute the eigenvectors. Using the notation of this article, that formula is

$$e_k(x) = \frac{1}{2\sqrt{\ell_k}} \sum_i v_{ik} (\hat{E}_{x'}[\mathcal{D}^2(x', x_i)] - \mathcal{D}^2(x_i, x)). \tag{3.7}$$

The formula is applied to obtain an embedding for the nonlandmark examples. One can show (Bengio et al., 2004) that $e_k(x)$ is the Nyström formula when $K_n(x,y)$ is defined as above. Landmark Isomap is thus equivalent to performing Isomap on the landmark points only and then predicting the embedding of the other points using the Nyström formula. It is interesting to note a recent descendant of Isomap and LLE, Hessian Eigenmaps (Donoho & Grimes, 2003), which considers the limit case of the continuum of the manifold and replaces the Laplacian in Laplacian eigenmaps by a Hessian.

3.5 Locally Linear Embedding. The LLE algorithm (Roweis & Saul, 2000) looks for an embedding that preserves the local geometry in the neighborhood of each data point. First, a sparse matrix of local predictive weights W_{ij} is computed, such that $\sum_j W_{ij} = 1$, $W_{ij} = 0$ if x_j is not a k-nearest-neighbor of x_i and $||(\sum_j W_{ij} x_j) - x_i||^2$ is minimized. Then the matrix $\tilde{M} = (I - W)'(I - W)$ is formed. The embedding is obtained from the lowest eigenvectors of $\tilde{M}$, except for the eigenvector with the smallest eigenvalue, which is uninteresting because it is proportional to $(1, 1, \ldots, 1)$ (and its eigenvalue is 0). To select the principal eigenvectors, we define our normalized matrix here by $M = cI - \tilde{M}$ and ignore the top eigenvector (although one could apply an additive normalization to remove the components along the $(1, 1, \ldots, 1)$ direction). The LLE embedding for x_i is thus given by the v_{ik}, starting at the second eigenvector (since the principal one is constant). If one insists on having a positive semidefinite matrix M, one can take for c the largest eigenvalue of $\tilde{M}$ (note that c changes the eigenvalues only additively and has no influence on the embedding of the training set).

In order to find our kernel K_n, we first denote by $w(x, x_i)$ the weight of x_i in the reconstruction of any point $x \in \mathbb{R}^d$ by its k nearest neighbors in the training set (if $x = x_j \in D$, $w(x, x_i) = \delta_{ij}$). Let us first define a kernel K'_n by $K'_n(x_i, x) = K'_n(x, x_i) = w(x, x_i)$ and $K'_n(x, y) = 0$ when neither x nor y is in the training set D. Let K''_n be such that $K''_n(x_i, x_j) = W_{ij} + W_{ji} - \sum_k W_{ki} W_{kj}$ and $K''_n(x, y) = 0$ when either x or y is not in D. It is then easy to verify that the kernel $K_n = (c-1)K'_n + K''_n$ is such that $K_n(x_i, x_j) = M_{ij}$ (again, there could be other ways to obtain a data-dependent kernel for LLE that can be applied out-of-sample). Something interesting about this kernel is that when $c \to \infty$, the embedding obtained for a new point x converges to the extension of LLE proposed in Saul and Roweis (2002), as shown in Bengio et al. (2004) (this is the kernel we actually used in the experiments reported here).

As noted independently in Ham et al. (2003), LLE can be seen as performing kernel PCA with a particular kernel matrix. The identification becomes more accurate when one notes that getting rid of the constant eigenvector (principal eigenvector of M) is equivalent to the centering operation in feature space required for kernel PCA (Ham et al., 2003).

4 Similarity Kernel Eigenfunctions

As noted in Williams and Seeger (2000), the kernel PCA projection formula (equation 2.3) is proportional to the so-called Nyström formula (Baker, 1977; Williams & Seeger, 2000), equation 1.1, which has been used successfully to "predict" the value of an eigenvector on a new data point, in order to speed up kernel methods computations by focusing the heavier computations (the eigendecomposition) on a subset of examples (Williams & Seeger, 2001). The use of this formula can be justified by considering the convergence of eigenvectors and eigenvalues, as the number of examples increases (Baker, 1977; Koltchinskii & Giné, 2000; Williams & Seeger, 2000; Shawe-Taylor & Williams, 2003). Here we elaborate on this relation in order to better understand what all these spectral embedding algorithms are actually estimating.

If we start from a data set D, obtain an embedding for its elements, and add more and more data, the embedding for the points in D converges (for eigenvalues that are unique): Shawe-Taylor and Williams (2003) give bounds on the convergence error (in the case of kernel PCA). Based on these kernel PCA convergence results, we conjecture that in the limit, each eigenvector would converge to an eigenfunction for the linear operator defined below, in the sense that the ith element of the kth eigenvector converges to the application of the kth eigenfunction to x_i.

In the following, we assume that the (possibly data-dependent) kernel K_n is bounded (i.e., $|K_n(x, y)| < c$ for all x, y in $\mathbb{R}$) and has a discrete spectrum; it can be written as a discrete expansion:

$$K_n(x, y) = \sum_{k=1}^{\infty} \alpha_k \phi_k(x) \phi_k(y).$$

Consider the space $\mathcal{H}_p$ of continuous functions f that are square integrable as follows:

$$\int f^2(x) p(x) dx < \infty,$$

with the data-generating density function $p(x)$. One must note that we actually work not on functions but on equivalence classes: we say two functions f and g belong to the same equivalence class (with respect to p) if and only if $\int (f(x) - g(x))^2 p(x) dx = 0$ (if p is strictly positive, then each equivalence class contains only one function).

We will assume that K_n converges uniformly in its arguments and in probability to its limit K as $n \to \infty$. This means that for all $\epsilon > 0$,

$$\lim_{n \to \infty} P(\sup_{x, y \in \mathbb{R}^d} |K_n(x, y) - K(x, y)| \geq \epsilon) = 0.$$

We will associate with each K_n a linear operator G_n and with K a linear operator G, such that for any $f \in \mathcal{H}_p$,

$$G_n f = \frac{1}{n} \sum_{i=1}^{n} K_n(\cdot, x_i) f(x_i) \tag{4.1}$$

and

$$G f = \int K(\cdot, y) f(y) p(y) dy, \tag{4.2}$$

which makes sense because we work in a space of functions defined everywhere. Furthermore, as $K_n(\cdot, y)$ and $K(\cdot, y)$ are square integrable in the sense defined above, for each f and each n, $G_n(f)$ and $G(f)$ are square integrable in the sense defined above. We will show that the Nyström formula, equation 1.1, gives the eigenfunctions of G_n (see proposition 1), that their value on the training examples corresponds to the spectral embedding, and that they converge to the eigenfunctions of G (see proposition 2) if they converge at all. These results will hold even if K_n has negative eigenvalues.

The eigensystems of interest are thus the following:

$$Gf_k = \lambda_k f_k \tag{4.3}$$

and

$$G_n f_{k,n} = \lambda_{k,n} f_{k,n}, \tag{4.4}$$

where (λ_k, f_k) and $(\lambda_{k,n}, f_{k,n})$ are the corresponding eigenvalues and eigenfunctions. Note that when equation 4.4 is evaluated only at the $x_i \in D$, the set of equations reduces to the eigensystem

$$Mv_k = n\lambda_{k,n} v_k.$$

The following proposition gives a more complete characterization of the eigenfunctions of G_n, even in the case where eigenvalues may be negative. The next two propositions formalize the link already made in Williams and Seeger (2000) between the Nyström formula and eigenfunctions of G.

Proposition 1. *G_n has in its image $m \leq n$ eigenfunctions of the form:*

$$f_{k,n}(x) = \frac{\sqrt{n}}{\ell_k} \sum_{i=1}^{n} v_{ik} K_n(x, x_i) \tag{4.5}$$

with corresponding nonzero eigenvalues $\lambda_{k,n} = \frac{\ell_k}{n}$, where $v_k = (v_{1k}, \ldots, v_{nk})'$ is the kth eigenvector of the Gram matrix M, associated with the eigenvalue ℓ_k. For $x_i \in D$ these functions coincide with the corresponding eigenvectors, in the sense that $f_{k,n}(x_i) = \sqrt{n} v_{ik}$.

Proof. First, we show that the $f_{k,n}$ coincide with the eigenvectors of M at $x_i \in D$. For $f_{k,n}$ associated with nonzero eigenvalues,

$$\begin{aligned} f_{k,n}(x_i) &= \frac{\sqrt{n}}{\ell_k} \sum_{j=1}^{n} v_{jk} K_n(x_i, x_j) \\ &= \frac{\sqrt{n}}{\ell_k} \ell_k v_{ik} \\ &= \sqrt{n} v_{ik}. \end{aligned} \tag{4.6}$$

The v_k being orthonormal, the $f_{k,n}$ (for different values of k) are therefore different from each other. Then for any x,

$$\begin{aligned} (G_n f_{k,n})(x) &= \frac{1}{n} \sum_{i=1}^{n} K_n(x, x_i) f_{k,n}(x_i) \\ &= \frac{1}{\sqrt{n}} \sum_{i=1}^{n} K_n(x, x_i) v_{ik} = \frac{\ell_k}{n} f_{k,n}(x), \end{aligned} \tag{4.7}$$

which shows that $f_{k,n}$ is an eigenfunction of G_n with eigenvalue $\lambda_{k,n} = \ell_k / n$.

The previous result shows that the Nyström formula generalizes the spectral embedding outside the training set. However, there could be many possible generalizations. To justify the use of this particular generalization, the following theorem helps in understanding the convergence of these functions as n increases. We would like the out-of-sample embedding predictions obtained with the Nyström formula to be somehow close to the asymptotic embedding (the embedding one would obtain as $n \to \infty$). Note also that the convergence of eigenvectors to eigenfunctions shown in Baker (1977) applies to data x_i, which are deterministically chosen to span a domain, whereas here the x_i form a random sample from an unknown distribution.

Proposition 2. *If the data-dependent bounded kernel K_n ($|K_n(x, y)| \leq c$) converges uniformly in its arguments and in probability, with the eigendecomposition of the Gram matrix converging, and if the eigenfunctions $f_{k,n}$ of G_n associated with nonzero eigenvalues converge uniformly in probability, then their limits are the corresponding eigenfunctions of G.*

Proof. Denote $f_{k,\infty}$ the nonrandom function such that

$$\sup_x |f_{k,n}(x) - f_{k,\infty}(x)| \to 0 \tag{4.8}$$

in probability. Similarly, let K the nonrandom kernel such that

$$\sup_{x,y} |K_n(x, y) - K(x, y)| \to 0 \tag{4.9}$$

in probability.

Let us start from the Nyström formula and insert $f_{k,\infty}$, taking advantage of Koltchinskii and Giné (2000), theorem 3.1, which shows that $\lambda_{k,n} \to \lambda_k$ almost surely, where λ_k are the eigenvalues of G:

$$f_{k,n}(x) = \frac{1}{n\lambda_{k,n}} \sum_{i=1}^{n} f_{k,n}(x_i) K_n(x, x_i) \tag{4.10}$$

$$\begin{aligned} &= \frac{1}{n\lambda_k} \sum_{i=1}^{n} f_{k,\infty}(x_i) K(x, x_i) \\ &\quad + \frac{\lambda_k - \lambda_{k,n}}{n\lambda_{k,n}\lambda_k} \sum_{i=1}^{n} f_{k,\infty}(x_i) K(x, x_i) \\ &\quad + \frac{1}{n\lambda_{k,n}} \sum_{i=1}^{n} f_{k,\infty}(x_i) [K_n(x, x_i) - K(x, x_i)] \\ &\quad + \frac{1}{n\lambda_{k,n}} \sum_{i=1}^{n} K_n(x, x_i) [f_{k,n}(x_i) - f_{k,\infty}(x_i)]. \end{aligned} \tag{4.11}$$

Below, we will need to have shown that $f_{k,\infty}(x)$ is bounded. For this, we use the assumption that K_n is bounded, independently of n: $|K_n(x,y)| < c$. Then

$$\begin{aligned}
|f_{k,n}(x)| = \left|\frac{1}{n\lambda_{k,n}}\sum_{i=1}^{n} f_{k,n}(x_i)K_n(x,x_i)\right| &\leq \frac{1}{n|\lambda_{k,n}|}\sum_{i=1}^{n}|f_{k,n}(x_i)||K_n(x,x_i)| \\
&\leq \frac{c}{n|\lambda_{k,n}|}\sum_{i=1}^{n}|f_{k,n}(x_i)| \\
&\leq \frac{c}{n|\lambda_{k,n}|}\sum_{i=1}^{n}\sqrt{n}|v_{ik}| \\
&\leq \frac{c}{\sqrt{n}|\lambda_{k,n}|}\sum_{i=1}^{n}\frac{1}{\sqrt{n}} \\
&\leq \frac{c}{|\lambda_{k,n}|},
\end{aligned}$$

where in the second line, we used the bound on K_n, on the third equation 4.6, and on the fourth, the fact that the maximum of $\sum_{i=1}^{n} a_i$ s.t. $a_i \geq 0$ and $\sum_{i=1}^{n} a_i^2 = 1$ is achieved when $a_i = \frac{1}{\sqrt{n}}$. Finally, using equation 4.8 and the convergence of $\lambda_{k,n}$, we can deduce that $|f_{k,\infty}| \leq c/|\lambda_k|$, thus is bounded.

We now insert $\frac{1}{\lambda_k}\int f_{k,\infty}(y)K(x,y)p(y)\,dy$ in equation 4.11 and obtain the following inequality:

$$\begin{aligned}
&\left|f_{k,n}(x) - \frac{1}{\lambda_k}\int f_{k,\infty}(y)K(x,y)p(y)\,dy\right| \\
&\quad\leq \left|\frac{1}{n\lambda_k}\sum_{i=1}^{n} f_{k,\infty}(x_i)K(x,x_i) - \frac{1}{\lambda_k}\int f_{k,\infty}(y)K(x,y)p(y)\,dy\right| \\
&\quad\quad + \left|\frac{\lambda_k - \lambda_{k,n}}{n\lambda_{k,n}\lambda_k}\sum_{i=1}^{n} f_{k,\infty}(x_i)K(x,x_i)\right| \\
&\quad\quad + \left|\frac{1}{n\lambda_{k,n}}\sum_{i=1}^{n} f_{k,\infty}(x_i)[K_n(x,x_i) - K(x,x_i)]\right| \\
&\quad\quad + \left|\frac{1}{n\lambda_{k,n}}\sum_{i=1}^{n} K_n(x,x_i)[f_{k,n}(x_i) - f_{k,\infty}(x_i)]\right| \\
&\quad\leq A_n + B_n + C_n + D_n. \qquad (4.12)
\end{aligned}$$

From our convergence hypothesis (equations 4.8 and 4.9), the convergence of $\lambda_{k,n}$ to λ_k almost surely, and the fact that $f_{k,\infty}$, K, and K_n are bounded, it is clear that the last three terms B_n, C_n, and D_n converge to 0 in probability. In addition, applying the law of large numbers, the first term A_n also converges to 0 in probability. Therefore,

$$f_{k,n}(x) \rightarrow \frac{1}{\lambda_k} \int f_{k,\infty}(y) K(x, y) p(y)\, dy$$

in probability for all x. Since we also have $f_{k,n}(x) \rightarrow f_{k,\infty}(x)$, we obtain

$$\lambda_k f_{k,\infty}(x) = \int f_{k,\infty}(y) K(x, y) p(y)\, dy,$$

which shows that $f_{k,\infty}$ is an eigenfunction of G, with eigenvalue λ_k; therefore $f_{k,\infty} = f_k$: the limit of the Nyström function, if it exists, is an eigenfunction of G.

Kernel PCA has already been shown to be a stable and convergent algorithm (Shawe-Taylor et al., 2002; Shawe-Taylor & Williams, 2003). These articles characterize the rate of convergence of the projection error on the subspace spanned by the first m eigenvectors of the feature space covariance matrix. When we perform the PCA or kernel PCA projection on an out-of-sample point, we are taking advantage of the above convergence and stability properties in order to trust that a principal eigenvector of the empirical covariance matrix estimates well a corresponding eigenvector of the true covariance matrix. Another justification for applying the Nyström formula outside the training examples is therefore, as already noted earlier and in Williams and Seeger (2000), in the case where K_n is positive semidefinite, that it corresponds to the kernel PCA projection (on a corresponding eigenvector of the feature space correlation matrix C).

Clearly, we have with the Nyström formula a method to generalize spectral embedding algorithms to out-of-sample examples, whereas the original spectral embedding methods provide only the transformed coordinates of training points (i.e., an embedding of the training points). The experiments described below show empirically the good generalization of this out-of-sample embedding.

An interesting justification for estimating the eigenfunctions of G has been shown in Williams and Seeger (2000). When an unknown function f is to be estimated with an approximation g that is a finite linear combination of basis functions, if f is assumed to come from a zero-mean gaussian process prior with covariance $E_f[f(x)f(y)] = K(x, y)$, then the best choices of basis functions, in terms of expected squared error, are (up to rotation/scaling) the leading eigenfunctions of the linear operator G as defined above.

5 Learning Criterion for the Leading Eigenfunctions

Using an expansion into orthonormal bases (e.g., generalized Fourier decomposition in the case where p is continuous), the best approximation of $K(x, y)$ (in the sense of minimizing expected squared error) using only m terms is the expansion that uses the first m eigenfunctions of G (in the order of decreasing eigenvalues):

$$\sum_{k=1}^{m} \lambda_k f_{k,n}(x) f_{k,n}(y) \approx K_n(x, y).$$

This simple observation allows us to define a loss criterion for spectral embedding algorithms, something that was lacking up to now for such algorithms. The limit of this loss converges toward an expected loss whose minimization gives rise to the eigenfunctions of G. One could thus conceivably estimate this generalization error using the average of the loss on a test set. That criterion is simply the reconstruction error:

$$L(x_i, x_j) = \left(K_n(x_i, x_j) - \sum_{k=1}^{m} \lambda_{k,n} f_{k,n}(x_i) f_{k,n}(x_j) \right)^2 .$$

Proposition 3. *Asymptotically, the spectral embedding for a continous kernel K with discrete spectrum is the solution of a sequential minimization problem, iteratively minimizing the expected value of the loss criterion $L(x_i, x_j)$. First, with $\{(f_k, \lambda_k)\}_{k=1}^{m-1}$ already obtained, one can recursively obtain (λ_m, f_m) by minimizing*

$$\begin{aligned} J_m(\lambda', f') = \int \Bigg(& K(x, y) - \lambda' f'(x) f'(y) \\ & - \sum_{k=1}^{m-1} \lambda_k f_k(x) f_k(y) \Bigg)^2 p(x) p(y) dx dy, \end{aligned} \tag{5.1}$$

where by convention we scale f' such that $\int f'(x)^2 p(x) = 1$ (any other scaling can be transferred into λ'). Second, if the K_n are bounded (independently of n) and the $f_{k,n}$ converge uniformly in probability, with the eigendecomposition of the Gram matrix converging, the Monte Carlo average of this criterion,

$$\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K_n(x_i, x_j) - \sum_{k=1}^{m} \lambda_{k,n} f_{k,n}(x_i) f_{k,n}(x_j) \right)^2 ,$$

converges in probability to the above asymptotic expectation.

Proof. We prove the first part of the proposition concerning the sequential minimization of the loss criterion, which follows from classical linear algebra (Strang, 1980; Kreyszig, 1990). We proceed by induction, assuming that we have already obtained $f_1, \ldots, f_{m-1}$ orthogonal eigenfunctions in order of decreasing absolute value of λ_i. We want to prove that (λ', f') that minimizes J_m is (λ_m, f_m).

Setting $\frac{\partial J_m}{\partial \lambda'} = 0$ yields

$$\lambda' = \langle f', Kf' \rangle - \sum_{i=1}^{m-1} \int \lambda_i f_i(x) f_i(y) . f'(x) f'(y) p(x) p(y) dx dy. \tag{5.2}$$

Thus, we have

$$\begin{aligned} J_m = J_{m-1} & \\ & - 2 \int \lambda' f'(x) f'(y) (K(x,y) - \sum_{i=1}^{m-1} \lambda_i f_i(x) f_i(y)) p(x) p(y) dx dy \\ & + \int (\lambda' f'(x) f'(y))^2 p(x) p(y) dx dy, \end{aligned}$$

which gives $J_m = J_{m-1} - \lambda'^2$, so that λ'^2 should be maximized in order to minimize J_m. Take the derivative of J_m with regard to the value of f' at a particular point z (under some regularity conditions to bring the derivative inside the integral), and set it equal to zero, which yields the equation

$$\begin{aligned} \int K(z,y) f'(y) p(y) dy = & \int \lambda' f'(z) f'(y)^2 p(y) dy \\ & + \sum_{i=1}^{m-1} \int \lambda_i f_i(z) f_i(y) f'(y) p(y) dy. \end{aligned}$$

Using the constraint $||f'||^2 = \langle f', f' \rangle = \int f'(y)^2 p(y) dy = 1$, we obtain:

$$(Kf')(z) = \lambda' f'(z) + \sum_{i=1}^{m-1} \int \lambda_i f_i(z) f_i(y) f'(y) p(y) dy, \tag{5.3}$$

which rewrites into $Kf' = \lambda' f' + \sum_{i=1}^{m-1} \lambda_i f_i \langle f', f_i \rangle$. Writing Kf' in the basis of all the eigenfunctions, $Kf' = \sum_{i=1}^{\infty} \lambda_i f_i \langle f', f_i \rangle$, we obtain

$$\lambda' f' = \lambda_m f_m \langle f', f_m \rangle + \sum_{i=m+1}^{\infty} \lambda_i f_i \langle f', f_i \rangle.$$

Since the f_i are orthogonal, take the norm and apply Parseval's theorem:

$$\lambda'^2 = {\lambda_m}^2 \langle f', f_m \rangle^2 + \sum_{i=m+1}^{\infty} {\lambda_i}^2 \langle f', f_i \rangle^2.$$

If the eigenvalues are distinct, we have $\lambda_m > \lambda_i$ for $i > m$, and the last expression is maximized when $\langle f', f_m \rangle = 1$ and $\langle f', f_i \rangle = 0$ for $i > m$, which proves that $f' = f_m$ is in fact the mth eigenfunction of the kernel K and thereby $\lambda' = \lambda_m$.

If the eigenvalues are not distinct, then the result can be generalized in the sense that the choice of eigenfunctions is not unique anymore, and the eigenfunctions sharing the same eigenvalue form an orthogonal basis for a subspace.

This concludes the proof of the first statement.

To prove the second part (convergence statement), we want to show that the difference between the average cost and the expected asymptotic cost tends toward 0. If we write $\widehat{K}_n(x, y) = \sum_{k=1}^{m} \lambda_{k,n} f_{k,n}(x) f_{k,n}(y)$ and $\widehat{K}(x, y) = \sum_{k=1}^{m} \lambda_k f_k(x) f_k(y)$, that difference is

$$\begin{aligned}
&\left| \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K_n(x_i, x_j) - \widehat{K}_n(x_i, x_j) \right)^2 - E_{x,y}\left[\left(K(x, y) - \widehat{K}(x, y) \right)^2 \right] \right| \\
&\quad \leq \left| \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K(x_i, x_j) - \widehat{K}(x_i, x_j) \right)^2 - E_{x,y}\left[\left(K(x, y) - \widehat{K}(x, y) \right)^2 \right] \right| \\
&\quad + \left| \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K_n(x_i, x_j) - \widehat{K}_n(x_i, x_j) - K(x_i, x_j) + \widehat{K}(x_i, x_j) \right) \right. \\
&\quad \left. \left(K_n(x_i, x_j) - \widehat{K}_n(x_i, x_j) + K(x_i, x_j) - \widehat{K}(x_i, x_j) \right) \right|.
\end{aligned}$$

The eigenfunctions and the kernel being bounded, the second factor in the product (in the second term of the inequality) is bounded by a constant B with probability 1 (because of the $\lambda_{k,n}$ converging almost surely).

Thus, we have with probability 1:

$$\begin{aligned}
&\left| \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K_n(x_i, x_j) - \widehat{K}_n(x_i, x_j) \right)^2 - E_{x,y}\left[\left(K(x, y) - \widehat{K}(x, y) \right)^2 \right] \right| \\
&\quad \leq \left| \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(K(x_i, x_j) - \widehat{K}(x_i, x_j) \right)^2 - E_{x,y}\left[\left(K(x, y) - \widehat{K}(x, y) \right)^2 \right] \right|
\end{aligned}$$

$$+\left|\frac{B}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(K_n(x_i,x_j)-K(x_i,x_j)-\widehat{K}_n(x_i,x_j)+\widehat{K}(x_i,x_j)\right)\right|.$$

But then, with our convergence and bounding assumptions, the second term in the inequality converges to 0 in probability. Furthermore, by the law of large numbers, the first term also tends toward 0 (in probability) as n goes to ∞. We have therefore proved the convergence in probability of the average loss to its asymptotic expectation.

Note that the empirical criterion is indifferent to the value of the solutions $f_{k,n}$ outside the training set. Therefore, although the Nyström formula gives a possible solution to the empirical criterion, there may be other solutions. Remember that the task we consider is that of estimating the eigenfunctions of G, that is, approximating a similarity function K where it matters according to the unknown density p. Solutions other than the Nyström formula might also converge to the eigenfunctions of G. For example, one could use a nonparametric estimator (such as a neural network) to estimate the eigenfunctions. Even if such a solution does not yield the exact eigenvectors on the training examples (i.e., does not yield the lowest possible error on the training set), it might still be a good solution in terms of generalization, in the sense of good approximation of the eigenfunctions of G. It would be interesting to investigate whether the Nyström formula achieves the fastest possible rate of convergence to the eigenfunctions of G.

6 Experiments

We want to evaluate whether the precision of the generalizations suggested in the previous sections is comparable to the intrinsic perturbations of the embedding algorithms. The perturbation analysis will be achieved by replacing some examples by others from the same distribution. For this purpose, we consider splits of the data in three sets, $D = F \cup R_1 \cup R_2$, and training with either $F \cup R_1$ or $F \cup R_2$, comparing the embeddings on F. For each algorithm described in section 3, we apply the following procedure:

1. We choose $F \subset D$ with $m = |F|$ samples. The remaining $n - m$ samples in D/F are split into two equal-size subsets, R_1 and R_2. We train (obtain the eigenvectors) over $F \cup R_1$ and $F \cup R_2$, and we calculate the Euclidean distance between the aligned embeddings obtained for each $x_i \in F$. When eigenvalues are close, the estimated eigenvectors are unstable and can rotate in the subspace they span. Thus, we estimate an alignment (by linear regression) between the two embeddings using the points in F.
2. For each sample $x_i \in F$, we also train over $\{F \cup R_1\}/\{x_i\}$. We apply the Nyström formula to out-of-sample points to find the predicted

embedding of x_i and calculate the Euclidean distance between this embedding and the one obtained when training with $F \cup R_1$, that is, with x_i in the training set (in this case, no alignment is done since the influence of adding a single point is very limited).

3. We calculate the mean difference (and its standard error, shown in Figure 2) between the distance obtained in step 1 and the one obtained in step 2 for each sample $x_i \in F$, and we repeat this experiment for various sizes of F.

The results obtained for MDS, Isomap, spectral clustering, and LLE are shown in Figure 2 for different values of $|R_1|/n$ (i.e., the fraction of points exchanged). Experiments are done over a database of 698 synthetic face images described by 4096 components that is available online at http://isomap.stanford.edu. Similar results have been obtained over other databases, such as Ionosphere (http://www.ics.uci.edu/˜mlearn/MLSummary.html) and

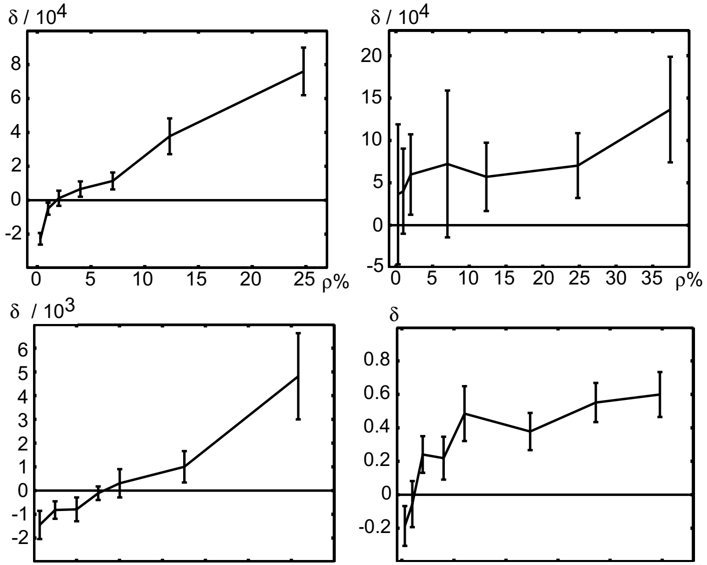

Figure 2: δ (training set variability minus out-of-sample error), with respect to ρ (proportion of substituted training samples) on the Faces data set ($n = 698$), obtained with a two-dimensional embedding. (Top left) MDS. (Top right) Spectral clustering or Laplacian eigenmaps. (Bottom left) Isomap. (Bottom right) LLE. Error bars are 95% confidence intervals. Exchanging about 2% of the training examples has an effect comparable to using the Nyström formula.

swissroll (http://www.cs.toronto.edu/˜roweis/lle/). Each algorithm generates a two-dimensional embedding of the images, following the experiments reported for Isomap. The number of neighbors is 10 for Isomap and LLE, and a gaussian kernel with a standard deviation of 0.01 is used for spectral clustering and Laplacian eigenmaps. Then 95% confidence intervals are drawn beside each mean difference of error on the figure. As expected, the mean difference between the two distances is almost monotonically increasing as the number $|R_1|$ of substituted training samples grows, mostly because the training set embedding variability increases. Furthermore, we find that in most cases, the out-of-sample error is less than or comparable to the training set embedding instability when around 2% of the training examples are substituted randomly.

7 Conclusion

Spectral embedding algorithms such as spectral clustering, Isomap, LLE, metric MDS, and Laplacian eigenmaps are very interesting dimensionality-reduction or clustering methods. However, up to now they lacked a notion of generalization that would allow easily extending the embedding out-of-sample without again solving an eigensystem. This article has shown with various arguments that the well-known Nyström formula can be used for this purpose and that it thus represents the result of a function induction process. These arguments also help us to understand that these methods do essentially the same thing, but with respect to different kernels: they estimate the eigenfunctions of a linear operator associated with a kernel and with the underlying distribution of the data. This analysis also shows that these methods are minimizing an empirical loss and that the solutions toward which they converge are the minimizers of a corresponding expected loss, which thus defines what good generalization should mean, for these methods. It shows that these unsupervised learning algorithms can be extended into function induction algorithms. The Nyström formula is a possible extension, but it does not exclude other extensions that might be better or worse estimators of the eigenfunctions of the asymptotic linear operator G. When the kernels are positive semidefinite, these methods can also be immediately seen as performing kernel PCA. Note that Isomap generally yields a Gram matrix with negative eigenvalues, and users of MDS, spectral clustering, or Laplacian eigenmaps may want to use a kernel that is not guaranteed to be positive semidefinite. The analysis in this article can still be applied in that case, even though the kernel PCA analogy does not hold anymore.

The experiments performed here have shown empirically on several data sets that the predicted out-of-sample embedding is generally not far from the one that would be obtained by including the test point in the training set and that the difference is of the same order as the effect of small perturbations of the training set.

An interesting parallel can be drawn between the spectral embedding algorithms and the view of PCA as finding the principal eigenvectors of a matrix obtained from the data. This article parallels for spectral embedding the view of PCA as an estimator of the principal directions of the covariance matrix of the underlying unknown distribution, thus introducing a convenient notion of generalization, relating to an unknown distribution.

Finally, a better understanding of these methods opens the door to new and potentially much more powerful unsupervised learning algorithms. Several directions remain to be explored:

- Using a smoother distribution than the empirical distribution to define the linear operator G_n. Intuitively, a distribution that is closer to the true underlying distribution would have a greater chance of yielding better generalization, in the sense of better estimating eigenfunctions of G. This relates to putting priors on certain parameters of the density, as in Rosales and Frey (2003).
- All of these methods are capturing salient features of the unknown underlying density. Can one use the representation learned through the estimated eigenfunctions in order to construct a good density estimator? Looking at Figure 1 suggests that modeling the density in the transformed space (right-hand side) should be much easier (e.g., would require much fewer gaussians in a gaussian mixture) than in the original space.
- Learning higher-level abstractions on top of lower-level abstractions by iterating the unsupervised learning process in multiple layers. These transformations discover abstract structures such as clusters and manifolds. It might be possible to learn even more abstract (and less local) structures, starting from these representations.

Acknowledgments

We thank Léon Bottou, Christian Léger, Sam Roweis, Yann Le Cun, and Yves Grandvalet for helpful discussions, the anonymous reviewers for their comments, and the following funding organizations: NSERC, MITACS, IRIS, and the Canada Research Chairs.

References

Baker, C. (1977). *The numerical treatment of integral equations*. Oxford: Clarendon Press.

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

Bengio, Y., Paiement, J., Vincent, P., Delalleau, O., Le Roux, N., & Ouimet, M. (2004). Out-of-sample extensions for LLE, Isomap, Mds, eigenmaps, and spectral clustering. In S. Thrun, L. Saul, & B. Schölkopf (Eds.), *Advances in neural information processing systems, 16*. Cambridge, MA: MIT Press.

Chung, F. (1997). *Spectral graph theory.* Providence, RI: American Mathematical Society.

Cox, T., & Cox, M. (1994). *Multidimensional scaling*. London: Chapman & Hall.

de Silva, V., & Tenenbaum, J. (2003). Global versus local methods in nonlinear dimensionality reduction. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems, 15* (pp. 705–712). Cambridge, MA: MIT Press.

Donoho, D., & Grimes, C. (2003). *Hessian eigenmaps: new locally linear embedding techniques for high-dimensional data* (Tech. Rep. No. 2003-08). Stanford, CA: Department of Statistics, Stanford University.

Gower, J. (1968). Adding a point to vector diagrams in multivariate analysis. *Biometrika, 55*(3), 582–585.

Ham, J., Lee, D., Mika, S., & Schölkopf, B. (2003). *A kernel view of the dimensionality reduction of manifolds* (Tech. Rep. No. TR-110). Tübingen, Germany; Max Planck Institute for Biological Cybernetics.

Koltchinskii, V., & Giné, E. (2000). Random matrix approximation of spectra of integral operators. *Bernoulli, 6*(1), 113–167.

Kreyszig, E. (1990). *Introductory functional analysis with applications*. New York: Wiley.

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2002). On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems, 14*, Cambridge, MA: MIT Press.

Rosales, R., & Frey, B. (2003). Learning generative models of affinity matrices. In *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence* (pp. 485–492). San Francisco: Morgan Kaufman.

Roweis, S., & Saul, L. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science, 290*(5500), 2323–2326.

Saul, L., & Roweis, S. (2002). Think globally, fit locally: Unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research, 4*,119–155.

Schölkopf, B., Burges, C. J. C., & Smola, A. J. (1999). *Advances in kernel methods—Support vector learning*. Cambridge, MA: MIT Press.

Schölkopf, B., Smola, A., & Müller, K.-R. (1996). *Nonlinear component analysis as a kernel eigenvalue problem* (Tech. Rep. No. 44), Tübingen, Germany: Max Planck Institute for Biological Cybernetics.

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10*, 1299–1319.

Shawe-Taylor, J., Cristianini, N., & Kandola, J. (2002). On the concentration of spectral properties. In T. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems, 14*. Cambridge, MA: MIT Press.

Shawe-Taylor, J., & Williams, C. (2003). The stability of kernel principal components analysis and its relation to the process eigenspectrum. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in neural information processing systems, 15*. Cambridge, MA: MIT Press.

Shi, J., & Malik, J. (1997). Normalized cuts and image segmentation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (pp. 731–737). New York: IEEE.

Spielman, D., & Teng, S. (1996). Spectral partitionning works: planar graphs and finite element meshes. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*. New York: IEEE.

Strang, G. (1980). *Linear algebra and its applications*. New York: Academic Press.

Tenenbaum, J., de Silva, V., & Langford, J. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, *290*(5500), 2319–2323.

Weiss, Y. (1999). Segmentation using eigenvectors: A unifying view. In *Proceedings IEEE International Conference on Computer Vision* (pp. 975–982). New York: IEEE.

Williams, C. (2001). On a connection between kernel pca and metric multidimensional scaling. In T. Leen, T. Dietterich, & V. Tresp (Eds.), *Advances in neural information processing systems, 13* (pp. 675–681). Cambridge, MA: MIT Press.

Williams, C., & Seeger, M. (2000). The effect of the input density distribution on kernel-based classifiers. In *Proceedings of the Seventeenth International Conference on Machine Learning*. San Mateo, CA: Morgan Kaufmann.

Williams, C. K. I., & Seeger, M. (2001). Using the Nyström method to speed up kernel machines. In T. Leen, T. Dietterich, & V. Tresp (Eds.), *Advances in neural information processing systems, 13* (pp. 682–688). Cambridge, MA: MIT Press.

Received March 24, 2003; accepted March 28, 2004.