Spherelets

Stat 185 Term Paper

Caleb Ren

December 14, 2019

Contents

1 Introduction

Data that exists in a high-dimensional ambient space may instead be considered to lie near a lower dimensional manifold (e.g. a circle in $\mathbb{R}^2$ that lies ambiently in $\mathbb{R}^3$ space). Many techniques are focused on reducing the ambient space to one closer to the intrinsic space such as clustering (Duda, Stork, and Hart 2000), data compression (Hurtado 2012), and predictive modelling (Lee and Verleysen 2008). Many of these techniques that attempt to approximate manifolds embedded in a lower dimensional space are locally linear, use multiscale dictionaries, and as such struggle with areas with high Gaussian curvature. Additionally, many techniques that approximate local manifolds do not perform well with out-of-sample error as they do not reveal much information on the traits of the lower dimension manifold.

A simple alternative that is able to handle curvature as well as out-of-sample error is to use pieces of (hyper)spheres to locally approximate the unknown subspace (Didong Li and Dunson 2019). This method was first proposed by Li and Dunson, who termed the technique spherical PCA (SPCA) (Li and Dunson 2017), or *spherelets* for short. Whereas principal component analysis (PCA) is an eigenvalue/eigenvector problem from an inherently *linear* dimension reduction problem, spherelets rely on projection mappings to the surfaces of a hypersphere as an underlying mechanism. As such, in simple cases of SPCA, closed form analytic solutions exist and can be seen as a generalization of PCA that is able to incorporate degrees of curvature. Spherical PCA in general does not have a closed-form solution due to the myriad ways in which a particular dataset can be partitioned and fit with sub-manifolds. Various algorithms exist to best partition the original dataset such as cover trees (Beygelzimer, Kakade, and Langford 2006), METIS (Karypis and Kumar 1998), and iterated PCA (Szlam 2009). The type of partitioning done is context-dependent and multiple types of subsetting can be done to cross-validated to reduce test MSE. As examples, we will examine two types of partitioning: naively—by splitting the dataset into equal-sized clusters—and via partioning by iterated PCA (Didong Li and Dunson 2019).

Data that is represented in higher dimension $\mathbb{R}^D$ can be projected down to a manifold in $\mathbb{R}^d$ consisting of a set of spherelets in $\mathbb{R}^d$. Ultimately, the algorithm partitions the dataset into k subsets, each of which is fit with a submanifold $M_k \in \mathbb{R}^d$ with a corresponding projection map $\Psi_k : \mathbb{R}^D \to \mathbb{R}^d$. Given a dataset $\mathbf{X} \in \mathbb{R}^D$, the spherical PCA algorithm returns manifold M and the projection map Ψ, which are the manifold and mapping from the data to the approximate manifold in $\mathbb{R}^d$, respectively.

Manifold approximation for model-building is one useful application of spherelets but the technique can be used for other purposes. Spherical PCA can also be used to denoise manifolds and generate visualizations of data in higher dimensions. Spherelets have been shown to outperform competiting denoising algorithms like Gaussian Blurring Mean Shift (GBMS), Manifold Blurring Mean Shift (MBMS), and Local Tangent Projection (LTP). Visualization techniques that preserve local geometry like Isomap, Locally Linear Embedding (LLE), and t-distributed Stochastic Neighborhood Embedding (t-SNE) can also be modified to incorporate spherical geometry using the same projection to spherical affine subspace, as in the original spherelets paper (Didong Li and Dunson 2019).

In spherical PCA, a point $\vec{y}_i$ is projected down to a sphere $S_V(c, r)$, where c is the center of the sphere, r is the radius, and V specifies the subspace on which the sphere lies. The optimal estimates for the sphere $S_{\hat{V}}(\hat{c}, \hat{r})$ are given by:

$$\hat{V} = (\vec{v}_1, \ldots, \vec{v}_{d+1})$$
$$\hat{c} = -\frac{\vec{\eta}^\star}{2}$$
$$\hat{r} = \frac{1}{N}\sum_{k=1}^{N} \|\vec{z}_i - \hat{c}\|$$

Where $\vec{v}_i$ is the ith eigenvector of the covariance matrix times $\mathbf{\Sigma}$, $\vec{z}_i$ is the the projected data down to the subspace and $\vec{\eta}^\star$ is a vector based on the entire projected dataset $\mathbf{Z}$. After an introduction into notation of spherical PCA, I will introduce relevant theory, including the Main Theorem that undergirds spherical

PCA. Assumptions made in spherical PCA are covered in section _________ and I will provide numerical examples demonstrating areas in which spherelets perform well and poorly based on the intrinsic structure of the manifold in Section 4.

2 Method

2.1 Notation

Assume we have the $N \times D$ data matrix $\mathbf{X}$, with N observations and D variable where $x_{i,j}$ represents the ith observation of the jth variable:

$$\mathbf{X} = \begin{bmatrix} x_{1,1} & \dots & x_{1,D} \\ \vdots & \ddots & \vdots \\ x_{i,1} & \dots & x_{i,D} \\ \vdots & \ddots & \vdots \\ x_{N,1} & \dots & x_{N,D} \end{bmatrix}$$

As in linear PCA, a more succinct way to represent this matrix is to write:

$$\mathbf{X} = \begin{bmatrix} \vec{x}_1^T \\ \vdots \\ \vec{x}_i^T \\ \vdots \\ \vec{x}_N^T \end{bmatrix}$$

We will once again treat $\vec{x}_1, \dots, \vec{x}_N$ as i.i.d. samples of a random vector in $\mathbb{R}^D$ from the same underlying distribution F for which $E_{x\sim F}\|x\|^2 < \infty$ to assure that the mean and covariance of the data matrix are well-deefined. We use:

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} \vec{x}_i$$

to denote the sample mean of $\vec{x}$. To denote the covariance operator, we use:

$$\mathbf{\Sigma}_x = \frac{1}{N}\sum_{i=1}^{N}(\vec{x}_i - \overline{x})(\vec{x}_i - \overline{x})^T = \frac{1}{N}\left(\sum_{i=1}^{N}\vec{x}_i\vec{x}_i^T\right) - \overline{x}\overline{x}^T = \frac{1}{N}\mathbf{X}^T\mathbf{X} - \overline{x}\overline{x}^T$$

This covariance operator will come into use later on. Additional notation includes:

- 1_N is the column vector of all ones with length N (i.e. $1_N \in \mathbb{R}^N$)
- $\|\vec{z}\|$ denotes the Euclidean norm (i.e. for $\vec{z} \in \mathbb{R}^d$, $\|\vec{z}\| = \left(\sum_{i=1}^{d} z_i^2\right)^{1/2} = \sqrt{\vec{z}^T\vec{z}}$)
- $\Psi : \mathbb{R}^D \to \mathbb{R}^{d+1}$ is a projection map from space $\mathbb{R}^D$ to $\mathbb{R}^{d+1}$. Note: Ψ maps from $\mathbb{R}^D$ to $\mathbb{R}^{d+1}$ rather than $\mathbb{R}^d$ because the projected points lie on an affine subspace with one fewer degree of freedom than the manifold. One example is fitting a curve in $\mathbf{R}^2$ with sections of 2-dimensional circles; while circles are 2D, the projected points only live along the edge of the circle. As such, we include on extra dimension in our projection map to account for the extra degree of freedom.

2.2 Main Theorem

The Main Theorem underlying spherical PCA is

2.3 Spherical PCA

Given a set of data $\vec{x}_1, \ldots, \vec{x}_N \in \mathbb{R}^D$, we find the best approximating sphere $S_V(c, r)$, where c is the center, r is the radius, and $V \in \mathbb{R}^{(d+1)\times(d+1)}$ is the $(d+1)$th dimensional affine subspace the sphere lives on. For any point in the dataset $\vec{x}_i$, the closest point $\vec{y}_i$ lying on the sphere $S_V(c, r)$ is the point that minimizes Euclidean distance $\|x, y\|^2$ between x and y. The optimal subspace V is given by $\hat{V} = (\vec{v}_1, \ldots, \vec{v}_{d+1})$, where $\vec{v}_i, i \in \{1, \ldots, d+1\}$ is the ith eigenvector ranked in descending order of $(\mathbf{X} - 1_N\bar{\mathbf{X}})^T(\mathbf{X} - 1_N\bar{\mathbf{X}})$.

If $\vec{z}_i = \bar{\mathbf{X}} + \hat{V}\hat{V}^T(\vec{x}_i - \bar{\mathbf{X}})$ are a change of basis to affine subspace V, then it can be shown that the minimizing pair $(\vec{\eta}^\star, \vec{\xi}^\star)$ of loss function $g(\vec{\eta}, \vec{\xi}) = \sum_{k=1}^N (\vec{z}_i^T\vec{z}_i + \vec{\eta}^T\vec{x}_i + \vec{\xi})^2$ is:

$$\vec{\eta} = -H^{-1}\omega$$

$$\vec{\xi} = -\frac{1}{N}\sum_{k=1}^{N}(\vec{z}_i^T\vec{z}_i + \vec{\eta}^T\vec{z}_i)$$

where H and ω are defined as:

$$H = \sum_{k=1}^{N}(\vec{z}_i - \overline{z})(\vec{z}_i - \vec{z})^T$$

$$\vec{\omega} = \sum_{k=1}^{N}\left(\|\vec{z}_i^T\vec{z}_i\| - \frac{1}{N}\sum_{j=1}^{N}\|\vec{z}_j^T\vec{z}_j\|\right)(\vec{z}_i - \overline{z})$$

The optimal parametrization $(\hat{V}, \hat{c}, \hat{r})$ of the projection of $\mathbf{X} \in \mathbb{R}^{N\times D}$ onto the sphere $S_V(c, r)$ is:

$$\hat{V} = (\vec{v}_1, \ldots, \vec{v}_{d+1})$$

$$\hat{c} = -\frac{\vec{\eta}^\star}{2}$$

$$\hat{r} = \frac{1}{N}\sum_{k=1}^{N}\|\vec{z}_i - \hat{c}\|$$

The projection map $\hat{\Psi}$ of data matrix $\mathbf{X}$ onto sphere $S_{\hat{V}}(\hat{c}, \hat{r})$ is the projection map onto affine subspace $\hat{c} + \hat{V}$, given by:

$$\hat{\Psi}(\vec{x}_i) = \hat{c} + \frac{\hat{r}}{\|\hat{V}\hat{V}^T(\vec{x}_i - \hat{c})\|}\hat{V}\hat{V}^T(\vec{x}_i - \hat{c})$$

2.4 Local SPCA

We have now defined spherical PCA (SPCA) to project the data $\mathbf{X}$ down to single sphere S_V. However, this single sphere will typically not be a sufficient approximation for the inherent manifold M. Instead, we partition the space $\mathbb{R}^D$ into k disjoint subsets $C_1, \ldots, C_k$. For the kth disjoint subset, we can define a data matrix $\mathbf{X}_k = \{X_i : X_i \in C_k\}$ that is a partition of the original data that lies within C_k. After applying SPCA to $\mathbf{X}_k$, we obtain spherical volume, center, and radius $(\hat{V}_k, \hat{c}_k, \hat{r}_k)$ alongside projection map Φ_k as a map from $x \in C_k$ to $y \in S_{\hat{V}_k}(\hat{c}_k, \hat{r}_k)$. A spherelets estimation $\hat{M}$ of the manifold M can be obtained by setting $\hat{M} = \bigcup_{k=1}^{K} \hat{M}_k$, where $\hat{M}_k$ is the local SPCA in the kth region and $\hat{M}_k = S_{\hat{V}_k}(\hat{c}_k, \hat{r}_k) \cap C_k$

2.5 Assumptions

There are two main

2.6 Method

The algorithm is as follows:

Algorithm 1 Spherelets

Input: Data matrix $\mathbf{X}$; intrinsic dimension d; partition $\{C_k\}_{k=1}^{K}$

Output: Local estimated manifolds $\hat{M}_k$ and projection map $\hat{\Psi}_k, k \in \{1, \ldots, K\}$; global estimated manifold $\hat{M}$ of intrinsic manifold M and projection map $\hat{\Psi}$

for (k = 1 : K) **do**
  Define $\mathbf{X}_{[k]} = \mathbf{X} \cap C_k$
  Calculate $\hat{V}_k, \hat{c}_k, \hat{r}_k$
  Calculate $\hat{\Psi}_k(x) = \hat{c}_k + \frac{\hat{r}_k}{\|\hat{V}_k \hat{V}_k^T (x - \hat{c}_k)\|}(x - \hat{c}_k)$
  Calculate $\hat{M}_k = S_{\hat{V}_k}(\hat{c}_k, \hat{r}_k) \cap C_k$
end for
Calculate $\hat{\Psi}(x) = \sum_{k=1}^{K} \mathbf{1}_{\{x \in C_k\}} \hat{\Psi}_k(x)$, and $\hat{M} = \bigcup_{k=1}^{K} \hat{M}_k$.

3 Strengths and Weaknesses

3.1 Strengths

- Performs well in areas with high curvature that local PCA can't approximate
- Can perform OOS assessments and returns the underlying manifold

3.2 Weaknesses

- Struggles with areas of non-uniform curvature
- Struggles with non-uniform dimensions
- Must specify inherent dimension d
- Computationally expensive
- Dependent on choice of manifold subsetting

4 Examples

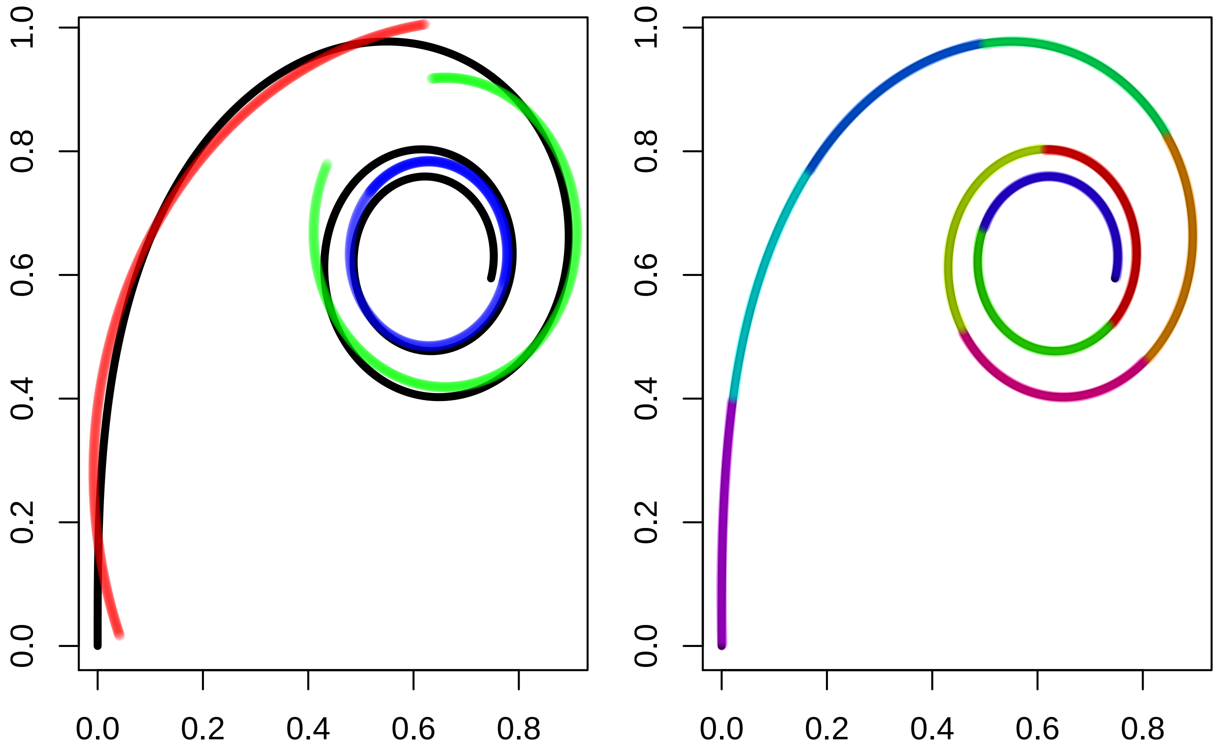
To generate numerical examples, I used the `SPCA` and `SS_calc` functions written by co-author Minerva Mukhopadhyay (mmukhopadhyay 2019). The `SPCA` function takes in a matrix of N observations $\vec{x}_i \in \mathbb{R}^D, i \in 1, \ldots, N$ and returns the error given by spherical and local PCA (`SS` and `SS_new`), as well as the projected values `Y_D`.

4.1 Euler Spiral

The Euler spiral is a curve in $\mathbb{R}^1$ that has a curvature that changes linearly with arc length. In other words, $\kappa(s) = s$. The Euler spiral can be parametrized as follows:

$$\begin{bmatrix} x(s) \\ y(s) \end{bmatrix} = \begin{bmatrix} \int_0^s \cos(t^2)dt \\ \int_0^s \sin(t^2)dt \end{bmatrix} \quad s \in [0, 4]$$

We use the Euler spiral as a demonstration to see how spherical PCA is able to handle regions with curvature.



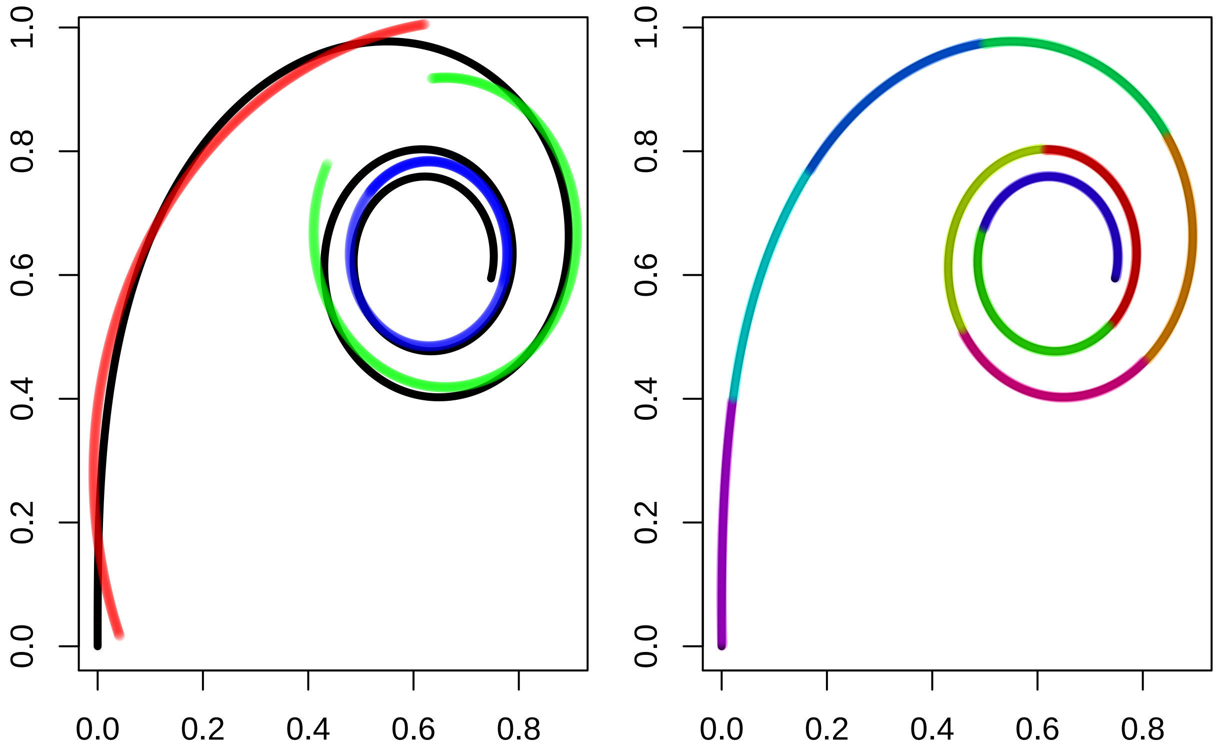


Figure 1: Spherical PCA performed on an Euler spiral with $k = 3, 8$.

4.2 Helix

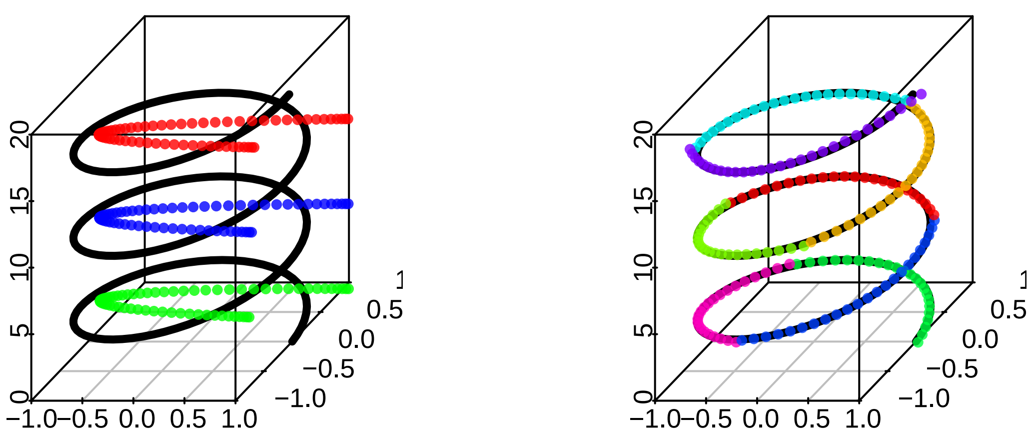


Figure 2: Spherical PCA performed on a helix with $k = 3, 8$.

4.3 Cylinder

```
## [1] 0.00269493
```

```
## [1] 7.353559e-05
```

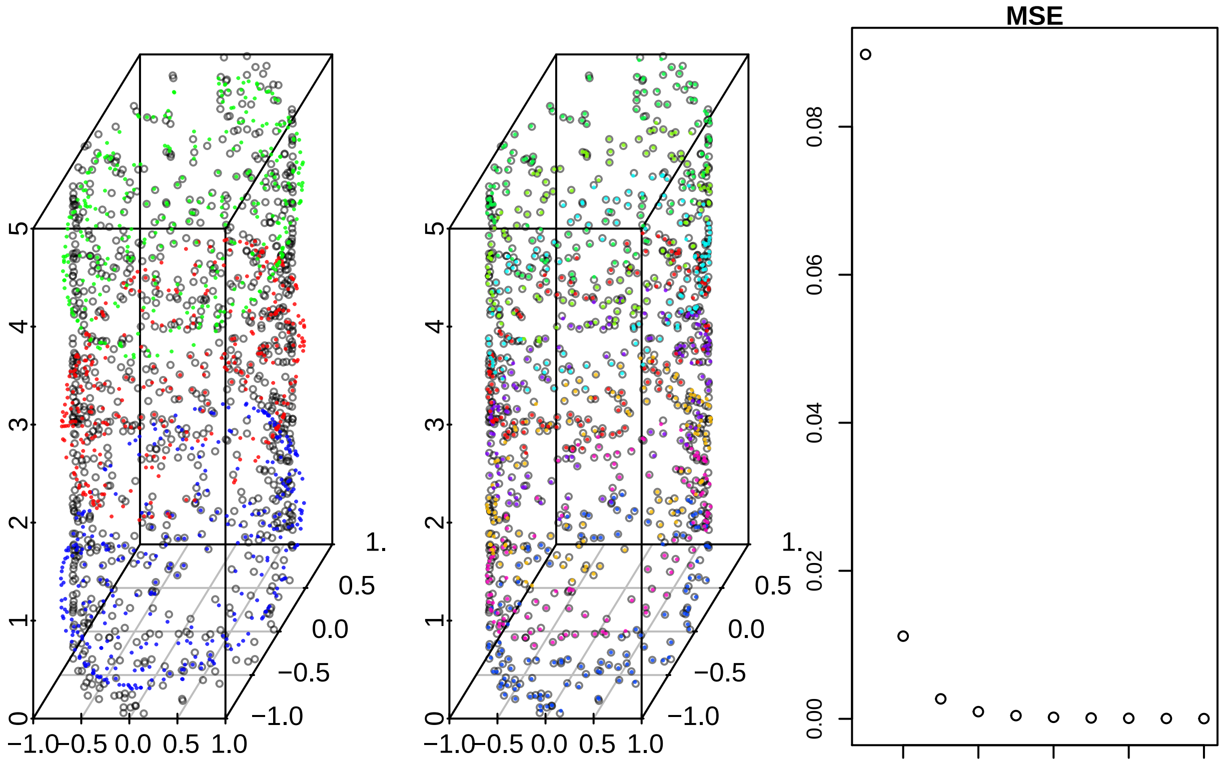


Figure 3: Spherical PCA performed on a cylinder with $k = 3, 8$.

We see that SPCA is not fully capable of handling a cylinder.

5 References

Beygelzimer, Alina, Sham Kakade, and John Langford. 2006. "Cover Trees for Nearest Neighbor." ACM.

Didong Li, Minerva Mukhopadhyay, and David B Dunson. 2019. "Efficient Manifold Approximation with Spherelets." *arXiv*, February.

Duda, Richard O, David G Stork, and Peter E Hart. 2000. *Pattern Classification*. 2nd ed. Wiley.

Hurtado, Jorge E. 2012. *Structural Reliability*. Vol. 17. Springer-Verlag.

Karypis, George, and Vipin Kumar. 1998. "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs." *SIAM Journal on Scientific Computing* 20 (1): 359–92. doi:10.1137/s1064827595287997.

Lee, John A, and Michel Verleysen. 2008. *Nonlinear Dimensionality Reduction*. Springer Science.

Li, Didong, and David B Dunson. 2017. "Efficient Manifold and Subspace Approximations with Spherelets." *arXiv*, June. ResearchGate.

mmukhopadhyay. 2019. "Efficient Manifold Learning Using Spherelets." Github. April 9.

Szlam, Arthur. 2009. "Asymptotic regularity of subdivisions of Euclidean domains by iterated PCA and iterated 2-means." *Applied and Computational Harmonic Analysis*, March. Academic Press. https://www.sciencedirect.com/science/article/pii/S1063520309000190.