ABSTRACT OF DISSERTATION

Weifeng Zhi

The Graduate School
University of Kentucky
2012

SPECTRAL ANALYSIS OF NONLINEAR DIMENSIONALITY REDUCTION METHODS

ABSTRACT OF DISSERTATION

A dissertation submitted in partial
fulfillment of the requirements for
the degree of Doctor of Philosophy
in the College of Arts and Sciences
at the University of Kentucky

By
Weifeng Zhi
Lexington, Kentucky

Director: Dr. Qiang Ye, Professor of Mathematics
Lexington, Kentucky 2012

ABSTRACT OF DISSERTATION

SPECTRAL ANALYSIS OF NONLINEAR DIMENSIONALITY REDUCTION METHODS

Nonlinear dimensionality reduction problems arise in a wide range of applications in information science and technology. Enormous amount of high-dimensional data sets are often obtained from various kinds of sensing systems or measurements. Processing high dimensional data set is a challenging problem in data classification and regression. Typically, these high-dimensional data sets admit low-dimensional parametric representations. It is then important to construct low-dimensional representations of high-dimensional data sets, called dimensionality reduction problem. Several competitive methods have been proposed in recent years for this task.

In this thesis, we present theoretical analysis of several recently developed dimensionality reduction methods. We generalize the previous analysis of the local tangent space alignment (LTSA) algorithm to include the case of alignments of sections of manifolds of different dimensions. We show that, under certain conditions, the alignment algorithm can successfully recover global coordinates even when local sections have different dimensions. We also present a spectral analysis for the alignment matrix to include this more general situation. Moreover, we present a theoretical analysis for the numerical procedure of Hessian Eigenmaps method. We formulate a discrete Hessian Eigenmaps method and show when it will recover the global coordinates. Our results provide a theoretical understanding of the Local Tangent Space Alignment method and Hessian Eigenmaps method for nonlinear dimensionality reduction.

KEYWORDS: nonlinear, dimensionality reduction, spectral analysis, Local Tangent Space Alignment, Hessian Eigenmaps

Author's signature:______________________

Date:______________________

SPECTRAL ANALYSIS OF NONLINEAR DIMENSIONALITY REDUCTION METHODS

By
Weifeng Zhi

Director of Dissertation:______________________

Director of Graduate Studies:______________________

Date:______________________

DISSERTATION

Weifeng Zhi

The Graduate School
University of Kentucky
2012

SPECTRAL ANALYSIS OF NONLINEAR DIMENSIONALITY REDUCTION METHODS

DISSERTATION

A dissertation submitted in partial
fulfillment of the requirements for
the degree of Doctor of Philosophy
in the College of Arts and Sciences
at the University of Kentucky

By
Weifeng Zhi
Lexington, Kentucky

Director: Dr. Qiang Ye, Professor of Mathematics
Lexington, Kentucky 2012

ACKNOWLEDGMENTS

I am most grateful to my advisor Dr. Qiang Ye for his great support and guidance during my time at University of Kentucky. Dr. Ye led me into the field on nonlinear dimensionality reduction and gave me numerous insightful suggestions and instructions. This thesis would not have been possible without his help. My thanks also go to the members of my dissertation committee: Dr. Russell Brown, Dr. Simon Bonner, Dr. Cidambi Srinivasan and Dr. Changyou Wang for their valuable advice on my dissertation.

I would also like to thank Dr. Li Gao, Dr. Ren-Cang Li and Dr. Shu-Fang Xu. I appreciate their great help during my academic experience.

I am also thankful to all the professors that have taught me and all the colleagues that I have been working with at University of Kentucky.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1 Introduction

Nonlinear dimensionality reduction problems arise in a wide range of applications in information science and technology. Enormous amount of high-dimensional data sets are often obtained from various kinds of sensing systems and measurements, such as systems of digital cameras, video surveillance, text document processing and sound analysis [9, 7, 19, 20, 35, 28]. Processing high dimensional data sets is a challenging problem in data classification and regression even though there have been tremendous progresses in the areas of data communication, storage and computation. Here, the computational work increases exponentially with the dimension of the data set. This is a phenomenon called "curse of dimensionality" [2].

Fortunately, in many cases, there are some low dimensional structures underlying these high dimensional data sets. For example, consider taking many pictures of a person by a camera with different pan angles and tilt angles. Assume the pictures that we obtain are 64×64 gray-scale digital images. Each picture consists of 3096 pixel values. We consider each picture as a 3096-dimensional vector, i.e. each pixel value is an entry of this high dimensional vector. Then there is a set of high dimensional vectors. However, we notice that this data set has only two underlying degrees of freedom, i.e. the pan angle and the tilt angle of the camera. If we change pan and/or tilt angles of the camera, the values of those high dimensional vectors change. These changes are highly related to each other. If we can explore the underlying correlation of the data points, we may perform further analysis of the high dimensional data set, such as data classification, regression and visualization.

In the last ten years or so, it has become clear that the high dimensional data governed by a few degrees of freedom can be modeled as points lying close to a low dimensional manifold in a high-dimensional space with/without noise [22, 27]. The problem of extracting the underlying low dimensional structure of the high dimensional data set turns into the problem of finding the low dimensional parametrization of the data points on the manifold, also called dimensionality reduction.

This manifold-based nonlinear dimensionality reduction has attracted significant interest in recent years. Mathematically, it can be described as follows. Consider a d-dimensional parameterized manifold $\mathcal{M}$ embedded in $\mathbb{R}^m$ $(d < m)$ characterized by a, possibly nonlinear, map,

$$\psi : \mathcal{C} \subset \mathbb{R}^d \to \mathbb{R}^m,$$

where $\mathcal{C}$ is a compact and connected subset of $\mathbb{R}^d$. Here $\mathbb{R}^m$ represents the high-dimensional data space, and $\mathbb{R}^d$ represents the low-dimensional parameter space. Given a set of data points $x_1, \cdots, x_N \in \mathbb{R}^m$ with

$$x_i = \psi(\tau_i), \quad i = 1, \ldots, N, \tag{1.1}$$

where $\tau_i \in \mathcal{C}$, the problem of dimensionality reduction is to recover low dimensional coordinates (parametrization) τ_i's from the x_i's.

Traditionally, the linear dimensionality reduction problem has been considered where the data set lies close to an affine subspace, i.e. ψ is a linear map. Such a problem can be solved by the Principal Component Analysis or Multidimensional Scaling method. However, many problems do not admit a linear structure. A more interesting problem concerns a nonlinear structure underlying the high dimensional data points, i.e. when ψ is a nonlinear map. In 2000, two algorithms, called Locally Linear Embedding (LLE) [22] and Isometric Mapping (Isomap) [27], were developed for this problem. Since then, several competitive algorithms have been proposed for nonlinear dimensionality reduction, which include Laplacian Eigenmap [1], Hessian

Eigenmaps [8], and Local Tangent Space Alignment (LTSA) [34] among many others; see [23] for a thorough review.

One idea underlying several of these methods is to reconstruct global coordinates τ_i from their local relations as defined by data points in a small neighborhood. For example, the LTSA method [34] recovers global coordinates through first constructing local coordinate systems for local neighborhoods and then aligning them into a global coordinate system by constructing an alignment matrix and computing its null space. The first theoretical analysis for this kind of alignment methods was obtained in [30]. It is shown in [30] that LTSA is able to recover the low dimensional representation of the high dimensional data up to a rigid motion under a certain condition on local neighborhoods called fully overlap, provided coordinates for points lying in local neighborhoods are constructed correctly.

One common assumption of the global construction methods based on local relations is that the underlying manifolds for the local neighborhoods (or the sets of local points) all have the same dimension d. (Here, we say a set of data points (1.1) is of dimension p if the corresponding set of coordinates τ_i, after being centered, spans a p-dimensional space.) However, there are many situations where such an assumption may not hold. For example, the data points may lie on several manifolds of different dimensions or they may be sampled from a d-dimensional manifold with lower dimensional branches/sections. Then the ability of dimensionality reduction algorithms to detect and work with change of dimension in local data points is very important. For the first part of this thesis, we derive a thorough analysis for the alignment of manifold sections of different dimensions. We show that the alignment algorithm can work with manifold sections of different dimensions under certain conditions on local neighborhoods. To demonstrate the application of this analysis, we consider a semi-supervised manifold learning method to allow alignment of manifolds with different dimensions.

Another important issue for the alignment methods is concerned with the computation of the null space of the alignment matrix. To computationally separate the null space, it is important to have a sufficient gap between the smallest positive eigenvalue and the zero eigenvalue. This is the subject of [16, 32] and indeed a full characterization of the eigenvalues of the alignment matrix was obtained in [16]. We generalize the results of [16] to include the case of alignment of manifold sections of different dimensions. Specifically, we present the characterization of the eigenvalues of the alignment matrix and a lower bound on the smallest positive eigenvalue.

We finally consider the Hessian Eigenmaps method, which is probably mathematically most sophisticated among the nonlinear dimensionality methods that have been developed. By introducing a Hessian operator and a quadratic form called $\mathcal{H}$-functional defined for a function $f : \mathcal{M} \to \mathbb{R}$, it is proved in [8] that the $\mathcal{H}$-functional has a $(d+1)$-dimensional null space consisting of the constant functions and a d-dimensional space of functions spanned by the original isometric coordinates. Hence, the locally isometric coordinates can be obtained up to a linear transformation by computing the $\mathcal{H}$-functional and its null space. This procedure, called the Hessian Eigenmap method, is set on the continuous framework and is a theoretical method only, however. To derive a practical method in the discrete setting, various approximations and necessary modifications have to be introduced and a resulting procedure is called Hessian LLE [8]. However, it is not clear exactly whether the modified discrete procedure still recovers the isometric coordinates as in the theoretical analysis for the continuous case. Here, we present a discrete Hessian Eigenmap method that is based on the numerical procedure developed in [8]. By defining a discrete Hessian operator and a generalized $\mathcal{H}$-functional that we call Hessian alignment matrix, we show that the null space of the Hessian alignment matrix recovers the locally isometric coordinates, provided local neighborhoods are sufficiently "overlapped".

The rest of this thesis is organized as follows.

In Chapter 2, we present a review of some dimensionality reduction methods that have been developed. In Section 2.1, we review two classical linear dimensionality reduction methods, i.e. Principle Component Analysis and Multidimensional Scaling. In Section 2.2, we present several nonlinear dimensionality reduction methods.

In Chapter 3, we consider an alignment algorithm for reconstructing global coordinates from local coordinates constructed for sections of manifolds. We show that, under certain conditions, the alignment algorithm can successfully recover global coordinates even when local sections have different dimensions. In Section 3.1, we review the alignment algorithm. We present an analysis of the alignments of sections of different dimensions in Section 3.2. We discuss a semi-supervised learning problem and put forth an algorithm for this problem in Section 3.3. We present several examples to illustrate our results in Section 3.4.

Chapter 4 provides a spectral analysis for the alignment matrix that arises in the alignment algorithms. In Section 4.1, we set up the framework and introduce notations by introducing the LTSA method and a more general alignment matrix. As in [16], Section 4.2 first presents our main results for the case of two submatrices and then discusses how to obtain a bound recursively in the general case of more than two subsections. We present some numerical examples to illustrate our bounds in Section 4.3.

In Chapter 5, we present and analyze a discrete Hessian Eigenmaps method that is based on the numerical procedure developed in [8]. In Section 5.1, we introduce the discrete Hessian Eigenmaps method including the construction of Hessian alignment matrix from local coordinates. In Section 5.2, we provide the analysis for the discrete Hessian Eigenmaps. In particular, we derive the condition under which the locally isometric coordinates can be recovered by Hessian alignment matrix constructed from local subsets.

In Chapter 6, we present some concluding remarks and discuss future work.

Notation. Throughout, I_n denotes the identity matrix of dimension n and e denotes a column vector of all ones the dimension of which should be clear from the context. $\mathsf{null}(\cdot)$ is the null space of the argument matrix, and $\mathsf{span}(\cdot)$ denotes the subspace spanned by all the columns of argument matrix. We also use the following MATLAB notation. For an index set $\boldsymbol{I} = [i_1, \ldots, i_k]$, $A(:, \boldsymbol{I})$ denotes the submatrix of A consisting of columns of A with indices $\boldsymbol{I}$.

For a matrix $A = [a_1, \ldots, a_n] \in \mathbb{R}^{m\times n}$, $\mathsf{vec}(A)$ represents the mn-dimension column vector formed by stacking the columns of A on top of each other as

$$\mathsf{vec}(A) = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \in \mathbb{R}^{mn}.$$

For two matrices $A = (a_{ij})_{m\times n} \in \mathbb{R}^{m\times n}$ and $B = (b_{ij})_{p\times q} \in \mathbb{R}^{p\times q}$, $A \bigotimes B \in \mathbb{R}^{mp\times nq}$ represents the Kronecker product of A and B, i.e.

$$A \bigotimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{1m}B & \cdots & a_{mn}B \end{pmatrix}.$$

For two row vectors $a = [a_1, \ldots, a_n] \in \mathbb{R}^{1\times n}$ and $b = [b_1, \ldots, b_n] \in \mathbb{R}^{1\times n}$, we also define $a \bigodot b = [c_1, \ldots, c_{n(n-1)/2}] \in \mathbb{R}^{n(n-1)/2}$ where

$$c_{\frac{k(k-1)}{2}+\ell} = a_k b_\ell \quad \text{for } \ell \le k. \tag{1.2}$$

Chapter 2 Preliminaries

In this section, we review several methods for dimensionality reduction and manifold learning. We limit ourselves to 'unsupervised' methods. The unsupervised methods extract features from the unlabeled data sets without specification of the task needing to be done from the low dimensional representation. The methods described here are divided into two groups:

- Linear Dimensionality Reduction methods. Traditional dimensionality reduction methods such as Principal Component Analysis and Multidimensional S-caling, extract the linear features out of the data sets.
- Nonlinear Dimensionality Reduction methods. Nonlinear dimensionality reduction methods such as LLE and Isomap explore the nonlinear map between the high dimensional data set and its low dimensional representation.

This chapter is organized as follows. In Section 2.1, we introduce the classical linear dimensionality methods. The nonlinear dimensionality reduction methods, including Locally Linear Embedding algorithm (LLE), Isometric Mapping algorithm (Isomap), Laplacian Eigemaps algorithm, Local Tangent Space Alignment algorithm (LTSA) and Hessian Eigenmaps (HLLE), are presented in Section 2.2.

2.1 Linear Dimensionality Reduction

In this section, we present two classical linear dimensionality reduction methods. These methods attempt to extract the linear features out of the high dimensional data. They are very useful in neutral networks [18, 15], Meteorology [12], Oceanography [21], data compression and computing [14].

2.1.1 Principle Component Analysis

Principle component analysis (PCA) is one of the traditional linear dimensionality reduction methods. There are several ways to describe PCA method. One of the popular ways is considering PCA as a method to discover a low rank projection of the high dimensional data set by maximizing the variance matrix [13]. The other idea is to find the linear low dimensional parametrization of the data set by minimizing the distance between the original data and its extracted low dimensional parametrization [11, 34]. The latter idea is used in the derivation of LTSA, and we present the details of this idea as follows.

Given a collection of data points $\{x_i\}$ sampled from a m-dimensional linear subspace, assume $x_i \in \mathbb{R}^m$ and the low dimensional parametrization $\tau_i \in \mathbb{R}^d$, $i = 1, \ldots, N$ such that

$$x_i = c + U\tau_i + \epsilon_i, \quad i = 1, \ldots, N,$$

where $c \in \mathbb{R}^m$, $U \in \mathbb{R}^{m\times d}$ and $\epsilon_i \in \mathbb{R}^m$ is the term of noise. The column vectors of matrix U is a set of orthogonal basis for the linear subspace. We rewrite this model in the matrix form below:

$$X = ce^T + UT + E,$$

where $X = [x_1, x_2, \ldots, x_N] \in \mathbb{R}^{m\times N}$, $T = [\tau_1, \tau_2, \ldots, \tau_N] \in \mathbb{R}^{d\times N}$ and $E = [\epsilon_1, \epsilon_2, \ldots, \epsilon_N] \in \mathbb{R}^{m\times N}$. The problem of dimensionality reduction in the linear case can be considered as minimizing the distance between the original data points and the low dimensional parametrization of the data points by seeking c, U and T, i.e.,

$$\min \|E\|_F = \min_{c,U,T} \|X - (ce^T + UT)\|_F,$$

where $\|\cdot\|_F$ is the Frobenius norm of a matrix and E is the difference between X and $ce^T + UT$.

The problem can be solved by singular value decomposition (SVD) based on the following two observations:

(1) The norm of the error matrix E can be reduced by centering the columns of E, and hence we can assume the optimal E is such that $Ee = 0$. Assume $Te = 0$, i.e. T is centered. This requirement can be fulfilled if c is chosen as $c = Xe/N \equiv \bar{x}$.

(2) The matrix UT is the rank-d approximation to the matrix $X - \bar{x}e^T$. Let

$$X - \bar{x}e^T = Q\Sigma V^T,$$

be the SVD for $X - \bar{x}e^T$, where $Q \in \mathbb{R}^{m\times m}$, $\Sigma \in \mathbb{R}^{m\times N}$ and $V \in \mathbb{R}^{N\times N}$. We have $UT = Q_d\Sigma_d V_d^T$, where $\Sigma_d = \text{diag}(\sigma_1, \ldots, \sigma_d)$ with d largest singular values $\sigma_1 \geq \cdots \geq \sigma_d$, Q_d and V_d consist of the corresponding left and right singular vectors, respectively. The Q_d is the optimal matrix U.

For PCA, the low dimensional parametrization is given by

$$T = Q_d^T(X - \bar{x}e^T) = \text{diag}(\sigma_1, \ldots, \sigma_d)V_d^T.$$

Ideally, the dimension d of the dimensionality reduction model should be chosen such that $\sigma_{d+1} \ll \sigma_d$.

2.1.2 Multidimensional Scaling

The other classical linear dimensionality reduction method is Multidimensional Scaling (MDS). The idea of MDS is to find the low dimensional projection of the data set that preserves the pairwise Euclidean distance between data points [5, 33]. We present MDS as follows.

Given a set of data points $\{x_i\}$, $i = 1, \ldots, N$, $x_i \in \mathbb{R}^m$ with $N > m$, assume $\tau_i \in \mathbb{R}^d$, $i = 1, \ldots, N$, is the low dimensional representation for x_i, $i = 1, \ldots, N$. We construct the set of pairwise Euclidean distances

$$d(x_i, x_j) = \|x_i - x_j\|_2.$$

We are asked to reconstruct the low dimensional parametrization τ_i's from the above set of pairwise distances. We can proceed as follows: without loss of generality, assume that the x_i's are centered

$$\sum_{i=1}^{N} x_i = 0.$$

Notice that the squared pairwise distance

$$d^2(x_i, x_j) = \|x_i - x_j\|_2^2. \tag{2.1}$$

Let $D = (D_{i,j})_{N\times N}$ be the squared-distance matrix, where $D_{i,j} = d^2(x_i, x_j)$. Let e be the N-dimensional vector of all ones and $X = [x_1, \dots, x_N]$. Let $J = I - ee^T/N$. Define

$$H \equiv -JDJ/2 = X^T X.$$

Let $T = [\tau_1, \dots, \tau_N]$. To recover the low dimensional representation T, let the eigen-decomposition of H be

$$H = U\mathrm{diag}(\lambda_1, \dots, \lambda_m)U^T,$$

where $U \in \mathbb{R}^{N\times m}$ is an orthogonal matrix and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m \geq 0$. Then we set

$$T = \mathrm{diag}(\lambda_1^{\frac{1}{2}}, \dots, \lambda_d^{\frac{1}{2}})U_d^T,$$

where U_d is the matrix consisting of the column vectors corresponding to the eigenvalues $\lambda_1, \dots, \lambda_d$. The column vectors of T are the low dimensional parametrization of the data points.

In applications, we can substitute the Euclidean distance (2.1) by any other proper dissimilarity criteria depending on the problem. See [5] for a thorough review.

Moreover, we notice that the procedure to find a d-dimensional representation of the data set by MDS is equivalent to projecting the data on the d-dimensional principle components. Thus MDS is equivalent to PCA. We can consider MDS as an alternative way for PCA in dimensionality reduction if we have the distance between each two data points rather than the coordinates of the data points [29].

PCA and MDS work well if the data points lie close to an affine subspace (manifold), whereas they fail to detect the nonlinear structure underlying the data points.

2.2 Nonlinear Dimensionality Reduction

Since the publications of LLE [22] and Isomap [27], several competitive methods have been proposed for the nonlinear dimensionality reduction, which include Laplacian Eigenmap [1], Hessian Eigenmap [8], and LTSA (Local Tangent Space Alignment)[34] among many others. In this section, we introduce several nonlinear dimensionality reduction methods, which can explore the nonlinear structure among the high dimensional data points.

2.2.1 Isometric Mapping

The Isomap method can be considered an extension of MDS for the nonlinear dimensionality reduction. Assume that we have a set of high dimensional data points lying closely to a nonlinear low dimensional manifold. We try to extract the low dimensional structure from the data set. The idea of Isomap is that the pairwise Euclidean distance between low dimensional parameters preserves the pairwise geodesic distance between high dimensional data points [27].

Given a set of data points $\{x_i\}$, $1, \ldots, N$, $x_i \in \mathbb{R}^m$, assume $\tau_i \in \mathbb{R}^d$, $i = 1, \ldots, N$, are the low dimensional parametrization for x_i, $i = 1, \ldots, N$. The main idea of ISOMAP is to compute pairwise geodesic distance among pairs of data points, which are the same as the Euclidean distance among τ_i's. Then τ_i can be constructed from its distance matrix using MDS.

First of all, Isomap finds the neighborhood with k nearest points of each data point x_i. We construct a neighborhood graph for all the data points as follows. We consider each point as a vertex of the neighborhood graph. If x_j is in the neighborhood of x_i, there is an edge between x_i and x_j. The weight of the edge between x_i and x_j is the Euclidean distance between vertex x_i and vertex x_j defined as $d_x(i,j)$. If the radius of the neighborhood is very small, the Euclidean distance between the two points in a small neighborhood approximates to the geodesic distance between those two points.

From the neighborhood graph constructed in the first step, we estimate the geodesic distance between any two points in the data set. For any two vertices in the graph, we compute the shortest path through the graph as an estimation for the geodesic distance between those two vertices. To compute the shortest distance among all vertices in the graph, we can use either Dijkstra's or Floyd's algorithm.

- Initialize

$$d_G(i,j) = \begin{cases} d_x(i,j), & \text{if } x_i \text{ and } x_j \text{ are linked by an edge;} \\ \infty, & \text{otherwise;} \end{cases}$$

- Compute

$$d_G(i,j) = \min\{d_G(i,j), d_G(i,k) + d_G(k,j)\}$$

  for $k = 1, \ldots, N$.

Given the geodesic distances between any two points, we construct a matrix with the squares of the estimated geodesic distances as follows.

$$D_G = \{d_G^2(i,j)\}.$$

Then we can apply MDS to discover the low dimensional parametrization for the data set. Set $J = I - \frac{1}{N}ee^T$. Let the eigendecomposition of matrix $H \equiv -\frac{1}{2}JDJ$ be

$$H = U\mathrm{diag}(\lambda_1, \ldots, \lambda_m)U^T,$$

where $U \in \mathbb{R}^{N\times m}$ is an orthonormal matrix and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m \geq 0$. Then we set

$$T = \mathrm{diag}(\lambda_1^{\frac{1}{2}}, \ldots, \lambda_d^{\frac{1}{2}}) U_d^T,$$

where U_d is the matrix consisting of the column vectors of U corresponding to the eigenvalues $\lambda_1, \ldots, \lambda_d$. The column vectors of T are the low dimensional parametrization of the data points. One disadvantage for Isomap is that it is not computationally efficient since it needs to solve an eigenvector problem of a dense matrix instead of a sparse matrix.

2.2.2 Locally Linear Embedding

Assume we have a set of high dimensional data points that lies closely on a low dimensional manifold. We consider that there is a locally linear mapping between the high dimensional data points and its low dimensional parametrization in a small neighborhood. The idea of the locally linear embedding method (LLE) is to minimize a convex error function constructed from the neighborhood of each data point [22]. We present the main procedure of LLE as follows.

Given a set of data points $\{x_i\}$, $x_i \in \mathbb{R}^m$, $1, \ldots, N$, the first step of LLE is to find a neighborhood $\mathcal{N}_i$ for each x_i by choosing the k nearest neighbors of x_i. Then we construct the weight $W = (W_{ij})_{N\times N}$ by minimizing the following error

$$e(W) = \sum_i \|x_i - \sum_j W_{ij} x_j\|_2^2.$$

LLE considers reconstructing x_i from its neighbors, so they enforce $W_{ij} = 0$ when x_j is not in the neighborhood of x_i. To get rid of the scaling freedom, it further requires that$\sum_j W_{ij} = 1$. If there are more neighbors than the dimension of the data sets, the optimal W_{ij} is not unique.

For the second step, it finds the low dimensional parametrization $\tau_i \in \mathbb{R}^d$, $1, \ldots, N$, using the weights from the first step. The weights constructed from the high dimensional space provide a good reconstruction of the coordinates on low dimensional manifold. Then it can set up a cost function by minimizing the reconstruction error for the low dimensional parametrization τ_i using the weights W_{ij} constructed from the high dimensional data below.

$$\Phi(T) = \sum_{i=1}^{N} \|\tau_i - \sum_j W_{ij}\tau_j\|_2^2. \tag{2.2}$$

To remove the rotational freedom in the final embedding, it further requires

$$\sum_{i=1}^{N} \tau_i = 0, \quad \tau_i^T \tau_j = 0 \text{ for } (i \neq j) \quad \text{and} \quad \|\tau_i\|_2 = 1. \tag{2.3}$$

We notice that the optimization problem is similar to the first step whereas the weights are fixed here and we try to find the coordinates τ_i, $i = 1, \ldots, N$.

We rewrite the problem of (2.2) and (2.3) as

$$\min_{TT^T=I} \Phi(T) = \min_{TT^T=I} \text{Trace}\left(T(I-W)(I-W)^T T^T\right)$$

where $T = [\tau_1, \ldots, \tau_N]$. Then the optimal d-dimensional parametrization is found by computing the eigenvectors corresponding to the smallest $d+1$ eigenvalues of the matrix $(I-W)(I-W)^T$. We notice that vector e is always the eigenvector corresponding to eigenvalue 0 of the matrix $(I-W)(I-W)^T$. Then the eigenvectors corresponding to the 2nd to d+1-st smallest eigenvalues are the matrix T^T. Therefore, we find the d-dimensional parametrization $T = [\tau_1, \ldots, \tau_N]$ for the data set.

2.2.3 Laplacian Eigenmaps

Laplacian Eigenmaps, like Isomap, is another nonlinear dimensionality reduction method based on a weighted graph for neighborhoods of high dimensional data points.

The idea of Laplacian Eigenmaps is preserving the geometric properties of high dimensional data from low dimensional parametrization.

Given a set of data points $\{x_i\}$, $x_i \in \mathbb{R}^m$, $1, \ldots, N$, the first step of Laplacian Eigenmaps is constructing the adjacency graph. If any two points are in the neighborhood of each other, there is an edge between those two points. As LLE and Isomap, we can define the neighborhoods as ϵ-neighborhoods or k-neighborhoods, which are both decided by the Euclidean distance between two points. For the ϵ-neighborhood of x_i, it consists of $x_j \in \mathcal{N}_i$, i.e., the neighborhood of x_i if $\|x_i - x_j\|_2 < \epsilon$. For the k-neighborhoods of x_i, it considers the k nearest points in the neighborhood of x_i, including x_i. Next, we need to choose the weights for the edges. For Laplacian Eigenmaps, there are two ways to assign the weights W_{ij} between two connected vertices x_i and x_j. We present them as follows.

- Heat Kernel.

$$W_{ij} = \begin{cases} e^{-\frac{\|x_i - x_j\|^2}{4t}}, & \text{if } x_i \text{ and } x_j \text{ are connected by an edge;} \\ 0, & \text{otherwise.} \end{cases}$$

  Here $t > 0$ is the parameter and does not affect the eigenvectors of the discrete Laplacian [1].

- Simple-minded.

$$W_{ij} = \begin{cases} 1, & \text{if } x_i \text{ and } x_j \text{ are connected by an edge;} \\ 0, & \text{otherwise.} \end{cases}$$

The third step of Laplacian Eigenmaps is to compute the low dimensional parametrization by solving a generalized eigenvector problem. We set up an objective function to minimize the sums of weighted squared distance between the low dimensional parametrization of any two points. The objective function is

$$\sum_{ij} \|\tau_i - \tau_j\|_2^2 W_{ij},$$

where τ_i is the low dimensional parametrization for x_i. Let D be a diagonal matrix, whose entries are the column sums of weight matrix $W = (W_{ij})_{N\times N}$ of last step. Let $D = (D_{ij})_{N\times N}$, where

$$D_{ij} \overset{\text{def}}{=} \begin{cases} \sum_k W_{kj}, & \text{if } i = j; \\ 0, & \text{if } i \neq j. \end{cases}$$

The Laplacian matrix is defined as $L = D - W$. Then the minimization problem can be rewrite in the matrix form as below.

$$\arg \min_{TDT^T = I} \text{Trace}(TLT^T),$$

where $T = [\tau_1, \ldots, \tau_N]$. We impose the restriction $TDT^T = I$ since this objective function is invariant to linear transformation and a scale of the parametrization has to be fixed here. Then the low dimensional parametrization can be achieved by computing the eigenvectors of the following generalized eigenvalue problem

$$LY = \lambda DY,$$

where $Y = T^T$. Noticing vector e being always the eigenvector corresponding to the eigenvalue zero, we take the eigenvectors corresponding to the 2nd to $(d+1)$-st eigenvalues as T^T. Then the column vectors of T are the low dimensional parametrization of the data set.

2.2.4 Local Tangent Space Alignment

Local Tangent Space Alignment method (LTSA) is another nonlinear dimensionality reduction algorithm. The idea of the LTSA method [34] is to construct global coordinates through first constructing the local coordinates on the local tangent space and then aligning those local coordinates to form a global coordinates. We preset the LTSA method as follows.

Given a set of data points $\{x_i\}$ sampled with noise from an underlying nonlinear manifold, $x_i \in \mathbb{R}^m$, $1, \dots, N$, set the matrix $X = [x_1, \dots, x_N]$. For each x_i, let $X_i = [x_{i_1}, \dots, x_{i_k}]$ be a matrix consisting of its k nearest neighbors including x_i in the Euclidean distance. Consider computing the best d-dimensional linear approximation for the data points in X_i,

$$\min_{x,\Theta,Q} \sum_{j=1}^{k} \|x_{i_j} - (x + Q\theta_j)\|_2^2 = \min_{x,\Theta,Q} \|X_i - (xe^T + Q\Theta)\|_2^2,$$

where Q is of d columns and is orthonormal, and $\Theta = [\theta_1, \dots, \theta_k]$. This is the same problem as PCA. The optimal solution is as follows.

$$x = \bar{x}_i, \quad \Theta_i = Q_i^T \left(I - \frac{1}{k} ee^T \right) = [\theta_1^{(i)}, \dots, \theta_k^{(i)}], \quad \theta_j^{(i)} = Q_i^T (x_{i_j} - \bar{x}_i),$$

where $\bar{x}_i = \frac{1}{k}\sum_{j=1}^{k} x_{i_j}$, $\theta_j^{(i)}$ is the local coordinate of x_{i_j} in the neighborhood of x_i and Q_i is the matrix consisting of d left singular vectors of matrix $X_i(I - \frac{1}{k}ee^T)$ corresponding to its d largest singular values. Next, we construct the global coordinate τ_i, $i = 1, \dots, N$, in the low dimensional feature space based on the local coordinates $\theta_j^{(i)}$. Assume that in each neighborhood, the corresponding global coordinates differ from the local coordinates by a local affine transformation.

$$\tau_{i_j} = \bar{\tau}_i + L_i \theta_j^{(i)} + \epsilon_j^{(i)}, \quad j = 1, \dots, k, \quad i = 1, \dots, N,$$

where $\bar{\tau}_i$ is the mean of τ_{i_j}'s, L_i is a local affine transformation matrix that needs to be determined, and $\epsilon_j^{(i)}$ is the local reconstruction error. Denoting $T_i = [\tau_{i_1}, \dots, \tau_{i_k}]$, we have the local reconstruction error

$$\sum_{j=1}^{k} \|\tau_{i_j} - (\bar{\tau}_i + L_i \theta_j^{(i)})\|_F^2 = \|T_i - (\bar{\tau}_i e^T + L_i \Theta_i)\|_F^2.$$

To preserve as much of the local geometry in the low-dimensional feature space, we seek to find τ_i and L_i to minimize the reconstruction errors $\epsilon_j^{(i)}$, i.e.,

$$\min \sum_i \|T_i \left(I - \frac{1}{k} ee^T - L_i \Theta_i \right) \|_F^2$$

Obviously, the optimal alignment matrix L_i is given by

$$L_i = T_i \left(I - \frac{1}{k} e e^T \right) \Theta_i^\dagger.$$

Then we have

$$T_i \left(I - \frac{1}{k} e e^T - L_i \Theta_i \right) = T_i \left(I - \frac{1}{k} e e^T \right) (I - \Theta_i^\dagger \Theta_i),$$

where $\Theta_i^\dagger$ is the Moore-Penrose generalized inverse of Θ_i.

Let $T = [\tau_1, \ldots, \tau_N]$ and E_i be the 0-1 selection matrix such that $TE_i = T_i$. We need to find T to minimize the overall construction error

$$\sum_i \|T_i \left(I - \frac{1}{k} e e^T \right) (I - \Theta_i^\dagger \Theta_i)\|_F^2 = \|TEW\|_F^2,$$

where $E = [E_1, \ldots, E_N]$ and $W = \text{diag}(W_1, \ldots, W_N)$ with

$$W_i = \left(I - \frac{1}{k} e e^T \right) (I - \Theta_i^\dagger \Theta_i).$$

Imposing the constraints $TT^T = I$ and $Te = 0$, we can determine T uniquely. Set

$$\Phi \equiv EWW^T E^T.$$

The optimal T^T is given by the d eigenvectors of Φ corresponding to the 2nd to $(d+1)$-st smallest eigenvalues. e corresponding to the eigenvalue 0, which is determined by the structure of Φ. In the subsection below, we shall state a full LTSA algorithm.

2.2.4.1 Analysis of Alignment Matrix of LTSA

The LTSA method provides a numerical procedure to compute the low dimensional representation for the high dimensional data set and the representation preserves the local geometry of the data points. However, what we obtained globally is not clear. In [30], Ye and etc. presents the insightful theoretical analysis of the global alignment procedure for LTSA. We introduce the analysis as follows.

Consider the data set (1.1). We are interested in computing a set of low dimensional parametrization τ_i's from the high dimensional data x_i's. First, the data set is partitioned into several subsets called local patch (neighborhood) of the manifold. On each of the patches, a local coordinate system is computed, from which we construct an alignment matrix and recover the global parametrization. Next, we introduce the alignment matrix in detail.

The analysis begins with the construction of the alignment matrix based on τ_i's. Consider $\boldsymbol{S}_i = \{\tau_1, \dots, \tau_N\}$ and let $\{\boldsymbol{S}_i, 1 \le i \le s\}$ be a collection of s subsets of $\boldsymbol{S}$. Write

$$\boldsymbol{S}_i = \{\tau_{i_1}, \dots, \tau_{i_{k_i}}\}, \quad i_1 < i_2 < \cdots < i_{k_i}, \tag{2.4}$$

and set

$$T = [\tau_1, \cdots, \tau_N] \in \mathbb{R}^{d\times N}, \ \ T_i = [\tau_{i_1}, \cdots, \tau_{i_{k_i}}]. \tag{2.5}$$

Let P_i be orthogonal projection onto the orthogonal complement of span$([e, T_i^T])$, i.e., $\mathsf{null}(P_i) = \mathsf{span}([e, T_i^T])$. Let

$$E_i = [e_{i_1}, \dots, e_{i_{k_i}}] \in \mathbb{R}^{N\times k_i}, \tag{2.6}$$

where $e_i \in \mathbb{R}^N$ is the i-th column of I_N (the $N \times N$ identity matrix). Define

$$\Phi_i = E_i P_i E_i^T; \tag{2.7}$$

and

$$\Phi = \sum_{i=1}^{s} \Phi_i. \tag{2.8}$$

Φ is the alignment matrix of the collection $\{T_i\}$.

In practice, only an approximate isometric coordinates $\{\theta_1^{(i)}, \dots, \theta_{k_i}^{(i)}\}$ can be computed from each local patch instead of $\{\tau_{i_1}, \dots, \tau_{i_{k_i}}\}$. $\{\theta_1^{(i)}, \dots, \theta_{k_i}^{(i)}\}$ are the local coordinates of $\boldsymbol{S}_i$. It has been shown [30, 32] that the global coordinates τ_i's can be constructed from the local coordinates through an alignment process as follows. Set

$$\Theta_i = \left[\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\right]. \tag{2.9}$$

and define Q_i to be the orthogonal projection onto the orthogonal complement of $\mathsf{span}\{\left[e, \Theta_i^T\right]\}$ in $\mathbb{R}^{k_i}$. Let

$$\Psi_i = E_i Q_i E_i^T, \quad \Psi = \sum_{i=1}^{s} \Psi_i. \tag{2.10}$$

Note that Ψ_i is the embedding of Q_i into an $N \times N$ matrix such that the (i_p, i_q)th element of Ψ_i is the (p, q)th element of Q_i. Ψ is called the alignment matrix for $\{\boldsymbol{S}_i, 1 \le i \le s\}$. We can recover T from the null space of Φ under certain condition called fully overlap. For the ease of references, we state the alignment process as follows.

Algorithm 2.1. Alignment Algorithm:
Given $\boldsymbol{X} = \{x_1, \cdots, x_N\} \subset \mathbb{R}^m$.

1. Construct a fully overlapped covering $\{\boldsymbol{X}_i, i = 1, \ldots, s\}$ with $\boldsymbol{X}_i = \{x_{i_1}, \ldots, x_{i_{k_i}}\}$.
2. For each $\boldsymbol{X}_i$, construct its local coordinates $\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}$.
3. Construct Ψ from $\Theta_i = [\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}]$ as in (2.10)
4. compute $[e/\sqrt{N}, Q^T]$ as an orthonormal basis of the spectral subspace of Ψ corresponding to the smallest $d+1$ eigenvalues, where $Q^T \in \mathbb{R}^{N \times d}$.
5. Recover T as $T = WQ$, where $W = \Theta_p \hat{Z}_p^\dagger$ and $\hat{Z}_p = Z E_i (I - \frac{1}{k_i} e e^T)$.

To introduce the main theorem of this section, we present some definitions and preliminary results at first.

Definition 2.1. Let $\boldsymbol{S}_x = \{x_1, \ldots, x_m\}$ and $\boldsymbol{S}_y = \{y_1, \ldots, y_n\}$ be two subsets of $\mathbb{R}^d$. Denote by $\boldsymbol{S}_z = \boldsymbol{S}_x \bigcap \boldsymbol{S}_y = \{z_1, \ldots, z_k\}$ the set of column vectors that are in the intersection of $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$. We say the two sets $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$ are fully overlapped if

$$\mathsf{span}(z_1 - \bar{z}, z_2 - \bar{z}, \ldots, z_k - \bar{z}) = \mathbb{R}^d, \tag{2.11}$$

where $\bar{z} = (\sum_{i=1}^{k} z_i)/k$.

We notice that condition (2.11) is equivalent to

$$[e, Z^T] \quad \text{having full column rank.} \tag{2.12}$$

Definition 2.2. This definition is recursive. Let $\boldsymbol{S}_i$, $1 \leq i \leq s$, be s subsets of $\mathbb{R}^d$. The collection $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ is fully overlapped if it can be partitioned into two nonempty disjoint collections, say, $\{\boldsymbol{S}_i, i = 1, \ldots, p\}$ and $\{\boldsymbol{S}_i, i = p+1, \ldots, s\}$, each which is a fully overlapped collection, and if the union sets of the two collections $\hat{\boldsymbol{S}}_1 \equiv \bigcup_{i=1}^{p} \boldsymbol{S}_i$ and $\hat{\boldsymbol{S}}_2 \equiv \bigcup_{i=p+1}^{s} \boldsymbol{S}_i$ are fully overlapped.

Next, we present the definition for the covering of a set $\boldsymbol{S}$.

Definition 2.3. The collection $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ is a covering of $\boldsymbol{S}$ if $\bigcup_{i=1}^{s} \boldsymbol{S}_i = \boldsymbol{S}$, and a fully overlapped covering if the collection is a covering and fully overlapped.

Under the condition of fully overlap for the covering $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$, two important theorems are shown in [30].

Theorem 2.1. Let Φ_i and Φ be defined as in (2.7) and (2.8), respectively, and let $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$ be a covering of $\boldsymbol{S}$. If it is fully overlapped, then $\mathsf{null}(\Phi) = \mathsf{span}([e, T^T])$.

However, in practice, τ_i is not available, but we have the local coordinate Θ_i for the points the neighborhood of x_i. The following theorem shows that the same alignment matrix is constructed with the local coordinates and the global coordinates. Then the global coordinates T is recovered from the null space of the alignment matrix.

Theorem 2.2. Let $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$, given by (2.4), be a covering of $\boldsymbol{S}$, and let $\{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\} \in \mathbb{R}^d$ which is isometric in the Euclidean distance to $\{\tau_{i_1}, \cdots, \tau_{i_{k_i}}\}$. Then

$$\Psi = \Phi,$$

where Ψ is defined according to (2.10) and Φ is the alignment matrix for $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ defined according to (2.8). In particular, $\mathsf{span}\{[e, T^T]\} \subset \mathsf{null}(\Psi)$ and, if $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$ is fully overlapped, then $\mathsf{null}\{\Psi\} = \mathsf{span}\{[e, T^T]\}$.

Hence, the global coordinates τ_i's can be obtained from computing $\mathsf{null}(\Psi)$, up to an orthogonal transformation (a rigid motion).

2.2.4.2 Eigenvalues of Alignment Matrix of LTSA

From the theoretical analysis [30] discussed above, we note that recovering the low dimensional representation of the high dimensional data set is equivalent to finding the null space of the alignment matrix constructed from the local coordinates. To computationally separate the null space, it is important to have a sufficient gap between the smallest positive eigenvalue and the zero eigenvalues. The spectral properties of alignment matrix for LTSA have been characterized in [16]. In Chapter 4, we generalize these results and derive a new lower bound for the smallest positive eigenvalue of alignment matrix when there are large local errors due to different dimensions of manifold sections.

[30] considers the alignment matrix in a more general setting. Given a matrix $Z \in \mathbb{R}^{N\times(d+1)}$ and s submatrices $Z_j \in \mathbb{R}^{k_j\times(d+1)}$ (for $1 \leq j \leq s$) consisting of certain rows of Z, let

$$T = \left[\begin{array}{c} \tau_1, \cdots, \tau_N \end{array}\right], \quad Z = [e, T^T]$$

and

$$T_j = \left[\begin{array}{c} \tau_{j_1}, \cdots, \tau_{j_{k_j}} \end{array}\right] \quad \text{and} \quad Z_j = [e, T_j^T].$$

We can study the eigenstructure of the alignment matrix Φ defined on $\{Z_j\}$ and Z.

In this analysis, we first define the alignment matrix in a general setting that the first column of Z does not have to be all ones. Given an $N \times \ell$ matrix Z, let

$Z_j \in \mathbb{R}^{k_j \times \ell}$ (for $1 \le j \le s$) be s submatrices consisting of certain rows of Z, i.e.

$$Z_j = E_j^T Z \tag{2.13}$$

for some $E_j = [e_{j_1}, \ldots, e_{j_{k_j}}] \in \mathbb{R}^{N \times k_j}$ (with $1 \le j_1 < j_2 < \cdots < j_{k_j} \le N$), where $e_i \in \mathbb{R}^N$ is the i-th column of I_N (the $N \times N$ identity matrix). Let $P_{Z_j} = Z_j Z_j^\dagger$ be the orthogonal projector in $\mathbb{R}^{k_j}$ onto the column space of Z_j, and let $P_{Z_j}^\perp = I - P_{Z_j}$. Embed $P_{Z_j}^\perp$ into $\mathbb{R}^{N \times N}$ according to the position of the rows of Z_j in Z and denote the resulting $N \times N$ matrix by Φ_j, i.e. $\Phi_j = E_j P_{Z_j}^\perp E_j^T$. Then the matrix

$$\Phi := \sum_{j=1}^{s} \Phi_j. \tag{2.14}$$

is called the *alignment matrix* for $\{Z_j : 1 \le j \le s\}$, see [16, 30]. It is easily seen from $P_{Z_j}^\perp Z_j = 0$ that $\Phi_j Z = 0$ and hence

$$\Phi Z = 0 \quad \text{or} \quad \mathsf{span}(Z) \subset \mathsf{null}(\Phi) \tag{2.15}$$

where $\mathsf{span}(Z)$ is the span of the columns of Z. In the context of dimensionality reduction, the rows of Z are the low dimensional coordinates (parametrization) of the data points that we wish to find. We present the definition of fully overlap for the general settings.

Definition 2.4. This definition is recursive.

1. Z_j always fully overlaps itself regardless of its rank;

2. Z_i and Z_j for $i \ne j$ are *fully overlapped*, if $Z_{(\boldsymbol{I}_i \bigcap \boldsymbol{I}_j,:)}$ has full column rank;

3. The collection $\boldsymbol{Z} = \{Z_j, 1 \le j \le s\}$ for $s \ge 3$ is *fully overlapped*, if it can be partitioned into two nonempty disjoint subsets $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ each of which is a fully overlapped collection and that $Z_{(\widetilde{\boldsymbol{I}}_1,:)}$ and $Z_{(\widetilde{\boldsymbol{I}}_2,:)}$ are fully overlapped, where

$$\widetilde{\boldsymbol{I}}_i = \bigcup_{Z_j \in \boldsymbol{Z}_i} \boldsymbol{I}_j. \tag{2.16}$$

Theorem 2.3. If $\{Z_j, 1 \le j \le s\}$ is fully overlapped, then $\mathsf{null}(\Phi) = \mathsf{span}(Z)$.

Theorem 2.3 is an extension of Theorem 2.1.

Corollary 2.1. Under the conditions of Theorem 2.3,

$$\lambda_{\min}^{+}(\Phi)\, P_Z^{\perp} \le \Phi \le \lambda_{\max}(\Phi)\, P_Z^{\perp},$$

where $\lambda_{\min}^{+}(\Phi)$ is the smallest nonzero eigenvalue of Φ, and $\lambda_{\max}(\Phi)$ is the largest eigenvalue of Φ.

Corollary 2.1 is used in Chapter 4. In the case of two submatrices for Z, we define

$$Z_1 = \begin{array}{c} \\ {\scriptstyle m_{11}} \\ {\scriptstyle m_{12}} \end{array} \!\!\begin{array}{c} {\scriptstyle \ell} \\ \left(\begin{array}{c} Z_{11} \\ Z_{12} \end{array}\right) \end{array}, \quad Z_2 = \begin{array}{c} \\ {\scriptstyle m_{21}} \\ {\scriptstyle m_{22}} \end{array} \!\!\begin{array}{c} {\scriptstyle \ell} \\ \left(\begin{array}{c} Z_{21} \\ Z_{22} \end{array}\right) \end{array}, \tag{2.17}$$

where $Z_{12} = Z_{21}$ is the common part in Z_1 and Z_2, $m_{12} = m_{21}$. Then

$$\Phi = \begin{array}{c|cc} & {\scriptstyle m_{11}+m_{12}} & {\scriptstyle m_{22}} \\ \hline {\scriptstyle m_{11}+m_{12}} & P_{Z_1}^{\perp} & 0 \\ {\scriptstyle m_{22}} & 0 & 0 \end{array} + \begin{array}{c|cc} & {\scriptstyle m_{11}} & {\scriptstyle m_{12}+m_{22}} \\ \hline {\scriptstyle m_{11}} & 0 & 0 \\ {\scriptstyle m_{12}+m_{22}} & 0 & P_{Z_2}^{\perp} \end{array}. \tag{2.18}$$

For this general settings, we have the following theorems from [16].

Theorem 2.4. Assume $m_{12} \ge 1$, $m_{11} \ge 1$, and $m_{22} \ge 1$. Z_{11}, $Z_{12} = Z_{21}$ and Z_{22} admit the following decompositions

$$\begin{aligned}
Z_{11} &= U_2 \times \begin{array}{c|ccc} & {\scriptstyle r_1} & {\scriptstyle r_2} & {\scriptstyle \ell-r_1-r_2} \\ \hline {\scriptstyle r_2} & \widetilde{M}_1 & \Sigma_2 & 0 \\ {\scriptstyle m_{11}-r_2} & M_1 & 0 & 0 \end{array} \times \begin{array}{cc} {\scriptstyle r_1} & {\scriptstyle \ell-r_1} \\ \hline I & 0 \\ 0 & V_2^* \end{array} V_1^*, \\
Z_{12} = Z_{21} &= U_1 \times \begin{array}{c|cc} & {\scriptstyle r_1} & {\scriptstyle \ell-r_1} \\ \hline {\scriptstyle r_1} & \Sigma_1 & 0 \\ {\scriptstyle m_{12}-r_1} & 0 & 0 \end{array} V_1^*, \\
Z_{22} &= U_3 \times \begin{array}{c|ccc} & {\scriptstyle r_1} & {\scriptstyle r_3} & {\scriptstyle \ell-r_1-r_3} \\ \hline {\scriptstyle r_3} & \widetilde{M}_2 & \Sigma_3 & 0 \\ {\scriptstyle m_{22}-r_3} & M_2 & 0 & 0 \end{array} \times \begin{array}{cc} {\scriptstyle r_1} & {\scriptstyle \ell-r_1} \\ \hline I & 0 \\ 0 & V_3^* \end{array} V_1^*,
\end{aligned}$$

where $U_1(m_{12}\times m_{12})$, $U_2(m_{11}\times m_{11})$, $U_3(m_{22}\times m_{22})$, V_1 $(\ell\times\ell)$, and V_2 and V_3 (both $(\ell-r_1)\times(\ell-r_1)$) are unitary, Σ_1 and Σ_2 are diagonal with positive diagonal entries. In particular

$$r_1=\mathsf{rank}(Z_{12}),\ r_2=\mathsf{rank}((Z_{11}V_1)_{(:,r_1+1:\ell)}),\ r_3=\mathsf{rank}((Z_{22}V_1)_{(:,r_1+1:\ell)}). \tag{2.19}$$

Theorem 2.5. Let all symbols keep their assignments as Theorem 2.4. Then

1. $\mathsf{dim}\,(\mathsf{null}(\Phi))=r_1+r_2+r_3$;
2. Suppose Z_1 and Z_2 have full column rank. Then $\mathsf{null}(\Phi)=\mathsf{span}(Z)$ if and only if Z_1 and Z_2 are fully overlapped.

Theorem 2.6. The nonzero eigenvalues of Φ is no smaller than $1-\tau$ where

$$\tau\equiv\frac{\|Z_{11}Z_{12}^{\dagger}\|_2}{\sqrt{1+\|Z_{11}Z_{12}^{\dagger}\|_2^2}}\,\frac{\|Z_{22}Z_{12}^{\dagger}\|_2}{\sqrt{1+\|Z_{22}Z_{12}^{\dagger}\|_2^2}}. \tag{2.20}$$

Its largest eigenvalue is no greater than $1+\tau$ if $m_{12}=r_1$ and it is 2 if $m_{12}>r_1$, where $r_1=\mathsf{rank}(Z_{12})$.

Theorem 2.7. Let τ be defined by (2.20). If Z_1 and Z_2 are fully overlapped, then

$$(1-\tau)P_Z^{\perp}\le\Phi\le\left\{\begin{array}{ll}(1+\tau) & \text{if } m_{12}=\ell,\\ 2 & \text{if } m_{12}>\ell.\end{array}\right\}P_Z^{\perp}. \tag{2.21}$$

Furthermore,

$$\lambda_{\min}^{+}(\Phi)\ge\frac{1}{2}\left(\frac{\sigma_{\min}^2(Z_{12})}{\sigma_{\max}^2(Z_{11})}+\frac{\sigma_{\min}^2(Z_{12})}{\sigma_{\max}^2(Z_{22})}\right)\Big/\left(1+\frac{\sigma_{\min}^2(Z_{12})}{\sigma_{\max}^2(Z_{11})}+\frac{\sigma_{\min}^2(Z_{12})}{\sigma_{\max}^2(Z_{22})}\right), \tag{2.22}$$

where $\sigma_{\min}$ and $\sigma_{\max}$ denote the smallest and the largest singular value respectively.

For the case of more than two submatrices for Z, we can bound the smallest positive eigenvalue $\lambda_{\min}^{+}(\Phi)$ recursively from below. Given

$$\widetilde{Z}_i=Z_{(\widetilde{\boldsymbol{I}}_i,:)},\quad i=1,2,$$

define a function τ

$$\tau(\widetilde{Z}_1, \widetilde{Z}_2) \overset{\text{def}}{=} \frac{t_1}{\sqrt{1+t_1^2}} \frac{t_2}{\sqrt{1+t_2^2}}, \quad t_i = \|Z_{(\boldsymbol{J}_i,:)} Z^{\dagger}_{(\widetilde{\boldsymbol{I}}_1 \cap \widetilde{\boldsymbol{I}}_2,:)}\|_2, \tag{2.23}$$

where $\boldsymbol{J}_i$ is the complement set of $\widetilde{\boldsymbol{I}}_1 \cap \widetilde{\boldsymbol{I}}_2$ in $\widetilde{\boldsymbol{I}}_i$.

Assume the collection $\boldsymbol{Z}$ is fully overlapped and partitioned into two disjoint subsets $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ each of which is a fully overlapped collection. The following procedure recursively computes $\alpha(\boldsymbol{Z})$ that satisfies $\alpha(\boldsymbol{Z})P_Z^{\perp} \leq \Phi$:

$$\alpha(\{Z_i\}) \;=\; 1, \tag{2.24}$$

$$\alpha(\{Z_i, Z_j\}) \;=\; 1 - \tau(Z_i, Z_j), \tag{2.25}$$

$$\alpha(\boldsymbol{Z}) \;=\; \left[1 - \tau(\widetilde{Z}_1, \widetilde{Z}_2)\right] \min\{\alpha(\boldsymbol{Z}_1), \alpha(\boldsymbol{Z}_2)\}. \tag{2.26}$$

The smallest positive eigenvalue $\lambda^{+}_{\min}(\Phi)$ is then no smaller than $\alpha(\boldsymbol{Z})$.

Theorem 2.8. Suppose $\boldsymbol{Z} = \{Z_1, Z_2, \ldots, Z_s\}$ is a fully overlapped collection, where Z_j are submatrices of $Z \in \mathbb{C}^{N\times\ell}$ as defined by (2.13). Let $\alpha(\boldsymbol{Z})$ be computed recursively by (2.24) – (2.26). Then $\alpha(\boldsymbol{Z})P_Z^{\perp} \leq \Phi$, where alignment matrix Φ is defined by (2.14).

2.2.5 Hessian Eigenmaps

In this section, we introduce the Hessian Eigenmaps. We first describe the Hessian Eigenmaps method of Donoho and Griems [8] in the continuous setting. Given that the map ψ defined in (1.1) is a local isometric embedding, the map $\phi = \psi^{-1} : \mathcal{M} \subset \mathbb{R}^m \to \mathbb{R}^d$ provides a (locally) isometric coordinate system for $\mathcal{M}$. The local isometry means that in a small neighborhood of each point x, geodesic distances to nearby points $x' \in \mathcal{M}$ in $\mathcal{M}$ are identical to Euclidean distances between the corresponding parameter points τ and τ', i.e. $\|\tau - \tau'\|_2 = d_{\mathcal{M}}(x, x')$, where $d_{\mathcal{M}}(\cdot,\cdot)$ is the geodesic distance along $\mathcal{M}$. The parameters τ and τ' are called the locally isometric coordinates for x' and x, respectively. Each component of ϕ is a function defined on $\mathcal{M}$ that provides one coordinate. The main idea of the Hessian Eigenmaps is to introduce a Hessian operator and a functional called the $\mathcal{H}$-functional defined for functions on $\mathcal{M}$, for which the null space consists of the d coordinate functions and the constant function.

Let $f : \mathcal{M} \mapsto \mathbb{R}$ be a function defined on $\mathcal{M}$ and let x_0 be an interior point of manifold $\mathcal{M}$. We can define a function $h : \mathcal{C} \mapsto \mathbb{R}$ as $h(\tau) = f(\psi(\tau))$, where $\mathcal{C} = \phi(\mathcal{M}) \subset \mathbb{R}^d$ and $\tau = [t_1, \ldots, t_d]^T \in \mathcal{C}$. Let $\tau_0 = \phi(x_0)$. We call the Hessian matrix of h at τ_0 the Hessian matrix of function f at x_0 in the isometric coordinate and we denote it by $H_f^{iso}(x_0)$. Then

$$(H_f^{iso})_{i,j}(x_0) = \frac{\partial^2 h(\tau_0)}{\partial t_i \partial t_j} \tag{2.27}$$

From the Hessian matrix, we define a $\mathcal{H}$-functional of f in isometric coordinates, denoted by $\mathcal{H}^{iso}(f)$, as

$$\mathcal{H}^{iso}(f) = \int_{\mathcal{M}} \|H_f^{iso}(x)\|_F^2 dx,$$

where dx is a probability measure on $\mathcal{M}$ which has strictly positive density everywhere on the interior of $\mathcal{M}$. It is clear from this definition that the $\mathcal{H}$-functionals of the d components of ϕ in isometric coordinates are zero. Indeed, $\mathcal{H}^{iso}(\cdot)$ has a $d+1$-dimensional null space, consisting of the span of the constant functions and the d component functions of ϕ; see [8, Corollary 4].

The Hessian matrix and the $\mathcal{H}$-functional in isometric coordinates introduced above are unfortunately not computable without knowing the isometric coordinate system ϕ first. To obtain a functional with the same property but independent of the isometric coordinate system ϕ, a Hessian matrix and the $\mathcal{H}$-functional in local tangent coordinates systems are introduced in [8]. We describe it now.

For a smooth manifold $\mathcal{M}$ and an interior point $x_0 \in \mathcal{M}$, let $T_{x_0}(\mathcal{M})$ denote the tangent space at x_0. Let $\mathcal{N}_{x_0}$ be the set of points $x \in \mathcal{M}$ in a small neighborhood of x_0. Consider the tangent space as a plane at x_0 (or a linear subspace $\mathbb{R}^n$ with the origin at x_0). There is an orthonormal basis $\{v_i, 1 \leq i \leq d\}$ for $T_{x_0}(\mathcal{M})$, where $v_i \in \mathbb{R}^m$. If $\mathcal{N}_{x_0}$ is a sufficiently small neighborhood, then for any point $x \in \mathcal{N}_{x_0}$, there is a unique point $v(x) \in T_{x_0}(\mathcal{M})$ that is closest to x. For x_0, the closest point in $T_{x_0}(\mathcal{M})$ is x_0 itself. We can write $v(x)$ in the basis $\{v_i\}$ as

$$v(x) = \theta_1^{(tan,x_0)}(x)v_1 + \cdots + \theta_d^{(tan,\theta_0)}(x)v_d.$$

In this way, each $x \in \mathcal{N}_{x_0}$ is uniquely defined by

$$\theta^{(tan,x_0)}(x) = [\theta_1^{(tan,x_0)}(x), \ldots, \theta_d^{(tan,x_0)}(x)]^T \in \mathbb{R}^d. \tag{2.28}$$

which we call a local tangent coordinate (parametrization) of $x \in \mathcal{N}_{x_0}$.

Now, let $f \in C^2(\mathcal{M}) : \mathcal{M} \mapsto \mathbb{R}$. It induces a function $g(\theta) : \theta \in U_0 \to \mathbb{R}$ defined as

$$g(\theta) = f(x), \tag{2.29}$$

where $\theta = \theta^{(tan,x_0)}(x) \in \mathbb{R}^d$ for $x \in \mathcal{N}_{x_0}$ and $U_0 \subset \mathbb{R}^d$ is a small neighborhood of $0 \in \mathbb{R}^d$ such that there is a one-to-one correspondence between $\theta \in U_0$ and $x \in \mathcal{N}_{x_0}$. From this, we define the Hessian matrix of f at x_0 in the local tangent coordinates as the ordinary Hessian matrix of $g(\theta)$ at $0 \in \mathbb{R}^d$ and denote it by $H_f^{tan}(x_0) = \big((H_f^{tan})_{i,j}(x_0)\big)_{d\times d}$. Then,

$$(H_f^{tan})_{i,j}(x_0) = \frac{\partial^2 g}{\partial\theta_i \partial\theta_j}(\theta)|_{\theta=0} \quad i,j = 1, \ldots, d. \tag{2.30}$$

While the definition of the Hessian above is dependent of the coordinate systems and the basis chosen for the tangent space, it is easy to see that the Hessians defined under different coordinate systems are orthogonally similar. Thus, up to an orthogonally similarity transformation, the Hessian matrix in the local tangent coordinates is uniquely defined.

We now define a $\mathcal{H}$-functional on a function $f \in C^2(\mathcal{M})$ as

$$\mathcal{H}(f) = \int_{\mathcal{M}} \|H_f^{tan}(x)\|_F^2 dx,$$

where dx is a probability measure on $\mathcal{M}$ which has strictly positive density everywhere on the interior of $\mathcal{M}$. It can be shown that $\mathcal{H}(f) = \mathcal{H}^{iso}(f).$, from which the following main theorem of [8] follows. Then we have the following theorem for $\mathcal{H}(f)$ defined on the tangent coordinates.

Theorem 2.9. (Donoho and Grimes [8]) Suppose $\mathcal{M} = \psi(\mathcal{C})$ where $\mathcal{C}$ is an open connected subset of $\mathbb{R}^d$, and ψ is a locally isometric embedding of $\mathcal{C}$ into $\mathbb{R}^n$. Then $\mathcal{H}(f)$ has a $d+1$ dimensional null space consisting of the constant function and a d-dimensional space of functions spanned by the original isometric coordinates (i.e the component functions of ϕ).

This theorem shows that we can recover the isometric coordinates of the manifold $\mathcal{M}$ from the null space of the $\mathcal{H}$-functional $\mathcal{H}(f)$. The original isometric coordinates can be recovered up to a rigid motion, by identifying a suitable basis for the null space of $\mathcal{H}(f)$.

In the discrete setting where we are given N high dimensional data points $\boldsymbol{X} = \{x_1, \cdots, x_N\} \subset \mathbb{R}^n$, the following numerical procedure called Hessian LLE is introduced to implement Theorem 2.9.

Step 1 **Identify Neighbors.** For every x_i, we identify a neighborhood $\mathcal{N}_{x_i}$ with k_i nearest point Let $N_{x_i} = \{x_{i_1}, \ldots, x_{i_{k_i}}\}$, $X_i = [x_{i_1}, \ldots, x_{i_{k_i}}]^T$ and $\bar{X}_i = (I - \frac{1}{k_i}ee^T)X_i = [x_{i_1} - \bar{x}_i, \ldots, x_{i_{k_i}} - \bar{x}_i]^T$, where $\bar{x}_i = \frac{1}{k_i}\sum_{j=1}^{k_i} x_{i_j}$.

Step 2 **Obtain Tangent Coordinates.** Let the singular value decomposition of $\bar{M}_i$ be

$$\bar{X}_i = U^{(i)} \Sigma^{(i)} {V^{(i)}}^T, \tag{2.31}$$

where $U^{(i)} = [u_1^{(i)}, \ldots, u_{k_i}^{(i)}] \in \mathbb{R}^{k_i \times k_i}$, $\Sigma^{(i)} = \mathrm{diag}(\sigma_1, \ldots, \sigma_{k_i}) \in \mathbb{R}^{k_i \times m}$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{k_i}$ and $V^{(i)} \in \mathbb{R}^{m \times m}$. The first d columns of $V^{(i)}$ span approximately the tangent space at x_i and the tangent coordinates of points in $\mathcal{N}_{x_i}$ are

$$[\theta_{i_1}^{(i)}, \ldots, \theta_{i_{k_i}}^{(i)}] := \mathrm{diag}(\sigma_1, \ldots, \sigma_d)[u_1^{(i)}, \ldots, u_d^{(i)}]. \tag{2.32}$$

Step 3 **Develop Hessian Estimator.** For the points in $\mathcal{N}_{x_i}$, we construct a matrix

$$X^{(i)} = \begin{array}{c} \begin{array}{ccc} \scriptstyle 1 & \scriptstyle d & \scriptstyle p \end{array} \\ \begin{pmatrix} 1 & {\theta_{i_1}^{(i)}}^T & {\theta_{i_1}^{(i)}}^T \bigodot {\theta_{i_1}^{(i)}}^T \\ 1 & {\theta_{i_2}^{(i)}}^T & {\theta_{i_2}^{(i)}}^T \bigodot {\theta_{i_2}^{(i)}}^T \\ \vdots & \vdots & \\ 1 & {\theta_{i_{k_i}}^{(i)}}^T & {\theta_{i_{k_i}}^{(i)}}^T \bigodot {\theta_{i_{k_i}}^{(i)}}^T \end{pmatrix} \end{array}, \tag{2.33}$$

where $p = d(d+1)/2$ and the operation $\bigodot$ is defined according to (1.2). We perform the Gram-Schmidt orthonormalization for $X^{(i)}$, yielding a matrix $\widetilde{X}_i \in \mathbb{R}^{k_i \times q}$ consisting of the orthonormal basis of $X^{(i)}$, where

$$q = (d+2)(d+1)/2.$$

Taking the $(d+2)$-th column to the $\mathsf{rank}(X^{(i)})$-th column of $\widetilde{X}_i$ as $G_i^T \in \mathbb{R}^{k_i \times p_i}$ with $p_i = \left[\mathsf{rank}(X^{(i)}) - (d+1)\right]$, we say G_i is a Hessian estimator for the neighborhood $\mathcal{N}_{x_i}$.

Step 4 **Develop Quadratic Form.** From the Hessian estimator G_i of the neighborhood $\mathcal{N}_{x_i}$, we define the matrix Ψ that corresponds to the H-functional as follows . Let $\widetilde{G}_i$ be the embedding G_i into $\mathbb{R}^{p_i \times N}$, i.e.

$$\widetilde{G}_i = G_i E_i^T, \tag{2.34}$$

where $E_i = [e_{i_1}, \ldots, e_{i_{k_i}}] \in \mathbb{R}^{N\times k_i}$ and where $e_i \in \mathbb{R}^N$ is the i-th column of I_N. Then

$$\Psi = \Sigma_{i=1}^{N} \widetilde{G}_i^T \widetilde{G}_i.$$

Step 5 **Find Approximate Null Space.** Compute the $d+1$ dimensional subspace corresponding to the $d+1$ smallest eigenvalues of Ψ. The vector e is the eigenvector of Ψ corresponding to the eigenvalue 0. Let T^T be the matrix consisting of the d eigenvectors corresponding to the 2nd to $(d+1)$-st eigenvalues, where $T \in \mathbb{R}^{d\times N}$. Then columns of T consist the isometric coordinates that we look for.

Chapter 3 Alignments of Manifold Sections of Different Dimensions

In this chapter, we consider the alignment algorithm for reconstructing global coordinates from local coordinates as derived in the LTSA method. We show that, under certain conditions, the alignment algorithm can successfully recover global coordinates from local neighborhood sections of different dimensions. Our main results generalize the analysis of [30] to allow alignment of sections of different dimensions. We shall also consider an application to a semi-supervised learning problem [10] where one wishes to find full association of two data sets that are partially associated.

3.1 Alignment Algorithms

Consider the data set (1.1). Let $\boldsymbol{X} = \{x_1, \cdots, x_N\}$ and let $\{\boldsymbol{X}_i, i = 1, \ldots, s\}$ be a collection of subsets of $\boldsymbol{X}$ with $\boldsymbol{X}_i = \{x_{i_1}, \ldots, x_{i_{k_i}}\}$ $(i_1 < i_2 < \cdots < i_{k_i})$. Assume that $\cup_i \boldsymbol{X}_i = \boldsymbol{X}$, in which case we say $\{\boldsymbol{X}_i, i = 1, \ldots, s\}$ is a covering of $\boldsymbol{X}$. In the context of LTSA, each $\boldsymbol{X}_i$ is a small local neighborhood so that a coordinate system on the local tangent space can be approximately constructed. In general, we assume that $\boldsymbol{X}_i$ is any subset such that an isometric coordinate $\{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\} \subset \mathbb{R}^d$ can be constructed, i.e. $\|\theta_p^{(i)} - \theta_q^{(i)}\|_2 = d_{\mathcal{M}}(x_{i_p}, x_{i_q})$ (for any $1 \le p, q \le k_i$) where $d_{\mathcal{M}}(\,\cdot\,,\,\cdot\,)$ is the geodesic distance along $\mathcal{M}$. In practice, only an approximate isometric coordinate can be computed.

It has been shown [32, 30] that the global coordinates τ_i's can be constructed from the local coordinates through an alignment process as follows. Set

$$\Theta_i = \left[\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\right]. \tag{3.1}$$

and define Q_i to be the orthogonal projection onto the orthogonal complement of

$\mathsf{span}\{\big[e, \Theta_i^T\big]\}$ in $\mathbb{R}^{k_i}$. Let $E_i = [e_{i_1}, \ldots, e_{i_{k_i}}] \in \mathbb{R}^{N\times k_i}$, where $e_i \in \mathbb{R}^N$ is the i-th column of I_N (the $N \times N$ identity matrix). Let

$$\Psi_i = E_i Q_i E_i^T, \quad \Psi = \sum_{i=1}^{s} \Psi_i. \tag{3.2}$$

Note that Ψ_i is the embedding of Q_i into an $N \times N$ matrix such that the (i_p, i_q)th element of Ψ_i is the (p, q)th element of Q_i. Ψ is called the alignment matrix for $\{\Theta_i\}$.

Under a condition called fully overlap for the covering $\{\boldsymbol{X}_i\}$, it is shown in Theorem 2.7 of [30] that

$$\mathsf{null}\{\Psi\} = \mathsf{span}\{\big[e, T^T\big]\}$$

where $T = [\tau_1, \cdots, \tau_N]$. Hence, the global coordinates τ_i's can be obtained from computing $\mathsf{null}\{\Psi\}$, up to an orthogonal transformation (a rigid motion).

The fully overlapped condition guarantees sufficient intersection (overlap) among $\boldsymbol{X}_i$'s to allow alignments. In the case of two subsets $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, it requires that the intersection $\boldsymbol{X}_1 \bigcap \boldsymbol{X}_2$ is of dimension d (see Definition 3.1 below or [30] for details). This immediately requires that all subsets $\boldsymbol{X}_i$ to have the same dimension d. However, the structure of the data set $\boldsymbol{X}$ may contain lower dimensional branches. In the next section, we generalize the analysis of [30] to include such cases. Interestingly, the alignment algorithm still works as long as a generalized fully overlapped condition holds.

3.2 Alignment of Sections of Different Dimensions

First we define the dimension of a data set or its coordinate set.

Definition 3.1. A data set $\boldsymbol{X}_0 = \{x_{i_1}, \ldots, x_{i_k}\}$ and the corresponding coordinate set $\boldsymbol{S}_0 = \{\tau_{i_1}, \ldots, \tau_{i_k}\}$ are said to be of dimension p if

$$\mathsf{rank}[\tau_{i_1} - \bar{\tau}, \tau_{i_2} - \bar{\tau}, \ldots, \tau_{i_k} - \bar{\tau}] = p \tag{3.3}$$

where $\bar{\tau} = (\Sigma_{j=1}^{k} \tau_{i_j})/k$. We write $\mathsf{dim}(\boldsymbol{X}_0) = \mathsf{dim}(\boldsymbol{S}_0) = p$.

The following lemma is shown in [30].

Lemma 3.1. $\mathsf{dim}(\boldsymbol{S}_0) = p$ if and only if

$$\mathsf{rank}(\big[e, T_0^T\big]) = 1 + p,$$

where $T_0 = [\tau_{i_1}, \ldots, \tau_{i_k}]$; see [30].

We construct the alignment matrix based on τ_i's first. Let $\boldsymbol{S} = \{\tau_1, \tau_2, \cdots, \tau_N\} \subset \mathbb{R}^d$ and let $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ be the collection corresponding to $\{\boldsymbol{X}_i, 1 \leq i \leq s\}$, where $\boldsymbol{S}_i$, T and T_i are defined according to (2.4) and (2.5). Let P_i be orthogonal projection onto the orthogonal complement of span($[e, T_i^T]$), i.e., $\mathsf{null}(P_i) = \mathsf{span}([e, T_i^T])$. Define Φ_i and Φ according to (2.7) and(2.8). Then Φ is the alignment matrix of the collection $\{T_i\}$. If $\boldsymbol{\Theta}_i$ is isometric to $\boldsymbol{X}_i$ (and hence to $\boldsymbol{S}_i$), then it can be shown that $\Psi = \Phi$, see [30].

First, we extend the definition of fully overlap to sets with different dimensions.

Definition 3.2. Let $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ be two subsets of $\boldsymbol{S} \subset \mathbb{R}^d$. We say $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are fully overlapped if

$$\min\{\mathsf{dim}(\boldsymbol{S}_1), \mathsf{dim}(\boldsymbol{S}_2)\} = \mathsf{dim}(\boldsymbol{S}_1 \cap \boldsymbol{S}_2).$$

Definition 3.3. This definition is recursive. Let $\boldsymbol{S}_i, 1 \leq i \leq s$, be s subsets of $\mathbb{R}^d$. The collection $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ is fully overlapped if it can be partitioned into two nonempty disjoint collections, say, $\{\boldsymbol{S}_i, i = 1, \ldots, p\}$ and $\{\boldsymbol{S}_i, i = p+1, \ldots, s\}$ each of which is a fully overlapped collection, and if the union sets of the two collections $\hat{\boldsymbol{S}}_1 := \cup_{i=1}^{p} \boldsymbol{S}_i$ and $\hat{\boldsymbol{S}}_2 := \cup_{i=p+1}^{s} \boldsymbol{S}_i$ are fully overlapped.

Definition 3.4. The collection $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ is a covering of $\boldsymbol{S}$ if $\cup_{i=1}^{s} \boldsymbol{S}_i = \boldsymbol{S}$, and a fully overlapped covering if the collection is a covering and fully overlapped.

We now show that this fully overlapped condition is sufficient to guarantee reconstruction of T from Φ or Ψ. First, the following is a lemma from [30].

Lemma 3.2. Let $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ be a covering of $\boldsymbol{S}$, and let Φ_i and Φ be defined as in (2.7) and (2.8). Then

$$\begin{aligned} \mathsf{null}(\Phi_i) &= \{x | E_i^T x \in \mathsf{span}([e, T_i^T])\} \\ \mathsf{null}(\Phi) &= \bigcap_{i=1}^{s} \mathsf{null}(\Phi_i). \end{aligned}$$

Theorem 3.1. Let $\{\boldsymbol{S}_1, \boldsymbol{S}_2\}$ be a fully overlapped covering of $\boldsymbol{S}$ and let Φ_i and Φ be defined as in (2.7) and (2.8). We have $\mathsf{null}\{\Phi\} = \mathsf{span}[e, T^T]$.

Proof: Without loss of generality, we assume that $d_2 := \mathsf{dim}(\boldsymbol{S}_2) < d_1 := \mathsf{dim}(\boldsymbol{S}_1)$. Then $\mathrm{rank}([e, T_2^T]) = d_2 + 1$. There is a nonsingular matrix U, such that $[e, T_1^T]U = [e, \widetilde{T}_1^T]$ and $[e, T_2^T]U = [e, \widetilde{T}_2^T]$ with the last $d - d_2$ columns of $\widetilde{T}_2^T$ being all zero. Suppose there are k vectors in $\boldsymbol{S}_1 \cap \boldsymbol{S}_2$. Without loss of generality, we assume that the last k columns of T_1 and the first k columns of T_2 are the vectors in $\boldsymbol{S}_1 \cap \boldsymbol{S}_2$. Then we write

$$\widetilde{T}_1 = \begin{array}{c} {\scriptstyle d_2} \\ {\scriptstyle d-d_2} \end{array} \begin{pmatrix} \widetilde{T}_{11}^{(1)} & \widetilde{T}_{12} \\ \widetilde{T}_{11}^{(2)} & 0 \end{pmatrix}; \; \widetilde{T}_2 = \begin{array}{c} {\scriptstyle d_2} \\ {\scriptstyle d-d_2} \end{array} \begin{pmatrix} \widetilde{T}_{21} & \widetilde{T}_{22} \\ 0 & 0 \end{pmatrix}$$

where $\widetilde{T}_{12} = \widetilde{T}_{21}$. Next, let the columns of Q form a basis of $\mathsf{null}(\Phi)$. We have $\mathsf{span}(Q) \subset \{x | E_i^T x \in \mathsf{span}([e, \widetilde{T}_i^T])\}$ for each i. Then we can find a matrix $W_i \in R^{(d+1)\times m}$, where $m = \mathsf{dim}(\mathsf{null}(\Phi))$, such that $E_i^T Q = [e, \widetilde{T}_i^T]W_i$. Let

$$W_i = \begin{array}{c} {\scriptstyle d_2+1} \\ {\scriptstyle d-d_2} \end{array} \begin{pmatrix} W_i^{(1)} \\ W_i^{(2)} \end{pmatrix}.$$

Comparing the common rows of $E_1^T Q$ and $E_2^T Q$, we have

$$[e, \begin{pmatrix} \widetilde{T}_{12}^T & 0 \end{pmatrix}]W_1 = [e, \begin{pmatrix} \widetilde{T}_{21}^T & 0 \end{pmatrix}]W_2.$$

From the first d_2+1 columns of last equation, we obtain that

$$[e, \widetilde{T}_{12}^T]W_1^{(1)} = [e, \widetilde{T}_{21}^T]W_2^{(1)}.$$

Since $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ are fully overlapped, we have that $[e, \widetilde{T}_{12}^T]$ has full column rank. From

$$[e, \widetilde{T}_{12}^T](W_1^{(1)} - W_2^{(1)}) = 0,$$

it follows

$$W_1^{(1)} - W_2^{(1)} = 0.$$

Noting that

$$\begin{aligned}
[e, \widetilde{T}_2^T]W_2 &= \left[e, \begin{pmatrix} \widetilde{T}_{21}^T & 0 \\ \widetilde{T}_{22}^T & 0 \end{pmatrix}\right] \begin{pmatrix} W_2^{(1)} \\ W_2^{(2)} \end{pmatrix} \\
&= \left[e, \begin{pmatrix} \widetilde{T}_{21}^T & 0 \\ \widetilde{T}_{22}^T & 0 \end{pmatrix}\right] \begin{pmatrix} W_1^{(1)} \\ W_1^{(2)} \end{pmatrix} \\
&= [e, \widetilde{T}_2^T]W_1,
\end{aligned}$$

we have

$$E_i^T Q = [e, T_i^T]UW_1.$$

So we can write Q as

$$Q = [e, T^T]UW_1.$$

Thus

$$\mathsf{null}\{\Phi\} = \mathsf{span}[e, T^T].$$

Theorem 3.2. Let $\{\boldsymbol{S}_i, i = 1, \ldots s\}$ be a fully overlapped covering of $\boldsymbol{S}$ and let Φ_i and Φ be defined as in (2.7) and (2.8). Then $\mathsf{null}\{\Phi\} = \mathsf{span}[e, T^T]$.

Proof: This is proved by virtually the same induction as in the proof of Theorem 2.6 [30] using Theorem 3.1 and Definition 3.3. We omit the details.

In practice, when we have a neighborhood consisting of points lying on a lower dimensional branches, their coordinates are likely computed with large errors in the components that are supposed to be zero. Amazingly, with a slightly extra condition, this does not affect the result of the alignment process. Before we present an analysis, we first illustrate with an example.

Example 3.1. Let $a, b, c, d, f, g, u, v, w, x$ be distinct numbers, such $(c-b)(w-u) - (d-b)(v-u) \neq 0$, $b \neq 0$ and $g \neq 0$, and

$$\boldsymbol{S} = \left\{ \begin{bmatrix} a \\ g \end{bmatrix}, \begin{bmatrix} b \\ 0 \end{bmatrix}, \begin{bmatrix} c \\ 0 \end{bmatrix}, \begin{bmatrix} d \\ 0 \end{bmatrix}, \begin{bmatrix} f \\ 0 \end{bmatrix} \right\}.$$

Assume that we have two subsets

$$\boldsymbol{S}_1 = \left\{ \begin{bmatrix} a \\ g \end{bmatrix}, \begin{bmatrix} b \\ 0 \end{bmatrix}, \begin{bmatrix} c \\ 0 \end{bmatrix}, \begin{bmatrix} d \\ 0 \end{bmatrix} \right\}$$

and

$$\boldsymbol{S}_2 = \left\{ \begin{bmatrix} b \\ 0 \end{bmatrix}, \begin{bmatrix} c \\ 0 \end{bmatrix}, \begin{bmatrix} d \\ 0 \end{bmatrix}, \begin{bmatrix} f \\ 0 \end{bmatrix} \right\}.$$

Then $\mathsf{dim}(\boldsymbol{S}_1) = 2$ and $\mathsf{dim}(\boldsymbol{S}_2) = \mathsf{dim}(\boldsymbol{S}_1 \cap \boldsymbol{S}_2) = 1$. By Theorem 3.1, we can recover T from Φ as constructed from T_1 and T_2.

In practice, however, we can only compute two coordinate sets $\boldsymbol{\Theta}_1$ and $\boldsymbol{\Theta}_2$ that are (approximately) isometric to $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$. However, large errors could be present in the second components of $\boldsymbol{S}_2$. For example, when computing $\boldsymbol{S}_2^T$ from local first order approximation [34], they are computed as 2 singular vectors and the second components derived from a singular vector corresponding to a tiny singular value may effectively be random. $\boldsymbol{S}_1$ can be computed accurately, however. Suppose the computed coordinates for the two sections are

$$\Theta_1 = \begin{bmatrix} a & b & c & d \\ g & 0 & 0 & 0 \end{bmatrix}; \; \Theta_2 = \begin{bmatrix} b & c & d & f \\ u & v & w & x \end{bmatrix}.$$

Now, constructing Ψ from Θ_i as in (3.2). Using Maple, we can compute Ψ and verify that $\mathsf{null}\{\Psi\} = \mathsf{span}\{[e, T^T]\}$. Hence, even when the second components in Θ_2 are computed completely wrong, original T can still be recovered from Ψ!

The phenomenon explained in the example above is true in general as shown in the following theorem.

Theorem 3.3. Let $\{\boldsymbol{S}_1, \boldsymbol{S}_2\}$ be a fully overlapped covering of $\boldsymbol{S} \subset \mathbb{R}^d$ with $\mathsf{dim}(\boldsymbol{S}_1) = d_1$ and $\mathsf{dim}(\boldsymbol{S}_2) = \mathsf{dim}(\boldsymbol{S}_1 \cap \boldsymbol{S}_2) = d_0 < d_1$. Assume that the vectors in $\boldsymbol{S}_2$ have vanishing last $d - d_0$ components. Let $\boldsymbol{\Theta}_1 = \boldsymbol{S}_1$ and

$$\boldsymbol{\Theta}_2 = \{ \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} \hat{\tau}_i \\ \hat{\rho}_i \end{pmatrix} : \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} \hat{\tau}_i \\ 0 \end{pmatrix} \in \boldsymbol{S}_2 \}.$$

Let Ψ_i and Ψ be defined from Θ_i as in (3.2). If the points of $\boldsymbol{\Theta}_2$ that correspond to $\boldsymbol{S}_1 \cap \boldsymbol{S}_2$ form a d-dimensional set, i.e.

$$\mathsf{dim}\left(\left\{\begin{pmatrix} \hat{\tau}_i \\ \hat{\rho}_i \end{pmatrix} \in \boldsymbol{\Theta}_2 : \begin{pmatrix} \hat{\tau}_i \\ 0 \end{pmatrix} \in \boldsymbol{S}_1 \cap \boldsymbol{S}_2\right\}\right) = d \qquad (3.4)$$

then we have $\mathsf{null}\{\Psi\} = \mathsf{span}[e, T^T]$.

Proof: Without loss of generality, we assume that the last k columns of T_1 and the first k columns of T_2 are the vectors in $\boldsymbol{S}_1 \cap \boldsymbol{S}_2$. Write

$$T_1 = \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} T_{11}^{(1)} & T_{12}^{(1)} \\ T_{11}^{(2)} & 0 \end{pmatrix} \quad \text{and} \quad T_2 = \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} T_{21}^{(1)} & T_{22}^{(1)} \\ 0 & 0 \end{pmatrix},$$

where $T_{12}^{(1)} = T_{21}^{(1)}$. Let Θ_1 and Θ_2 be the matrices whose columns are the vectors in $\boldsymbol{\Theta}_1$ and $\boldsymbol{\Theta}_2$ respectively, i.e. we write

$$\Theta_1 = T_1 = \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} T_{11}^{(1)} & T_{12}^{(1)} \\ T_{11}^{(2)} & 0 \end{pmatrix} \quad \text{and} \quad \Theta_2 = \begin{array}{c} {\scriptstyle d_0} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} T_{21}^{(1)} & T_{22}^{(1)} \\ T_{21}^{(2)} & T_{22}^{(2)} \end{pmatrix}.$$

Let Q be such that its columns form a basis for $\mathsf{null}(\Psi)$. We have $\mathsf{span}(Q) \subset \{x | E_i^T x \in \mathsf{span}([e, \Theta_i^T])\}$. Then we can find a matrix $W_i \in \mathbb{R}^{(d+1)\times m}$, where $m = \mathsf{dim}(\mathsf{null}(\Phi))$, such that $E_i^T Q = [e, \Theta_i^T] W_i$. Let

$$W_i = \begin{array}{c} {\scriptstyle d_0+1} \\ {\scriptstyle d-d_0} \end{array} \begin{pmatrix} W_i^{(1)} \\ W_i^{(2)} \end{pmatrix}.$$

Then we have

$$\left[e, \begin{pmatrix} T_{12}^{(1)^T} & 0 \end{pmatrix}\right] W_1 = \left[e, \begin{pmatrix} T_{21}^{(1)^T} & T_{21}^{(2)^T} \end{pmatrix}\right] W_2.$$

Equivalently,

$$[e, T_{12}^{(1)^T}] W_1^{(1)} = [e, T_{21}^{(1)^T}] W_2^{(1)} + T_{21}^{(2)^T} W_2^{(2)}.$$

Noting $T_{12}^{(1)^T} = T_{21}^{(1)^T}$, we have

$$[e, T_{21}^{(1)^T}](W_1^{(1)} - W_2^{(1)}) = T_{21}^{(2)^T} W_2^{(2)}.$$

Using (3.4), we see that $[e, T_{21}^{(1)^T}, T_{21}^{(2)^T}]$ has full column rank. It follows from

$$[e, T_{21}^{(1)^T}](W_1^{(1)} - W_2^{(1)}) - T_{21}^{(2)^T} W_2^{(2)} = 0$$

that

$$[e, T_{21}^{(1)^T}](W_1^{(1)} - W_2^{(1)}) = 0, \quad T_{21}^{(2)^T} W_2^{(2)} = 0.$$

This further implies that

$$W_1^{(1)} - W_2^{(1)} = 0, \quad W_2^{(2)} = 0.$$

Thus $W_1^{(1)} = W_2^{(1)}$. Then we can write

$$E_2^T Q = [e, S_2^T] W_2 = [e, T_2^T] W_1.$$

This together with $E_1^T Q = [e, T_1^T] W_1$ implies that

$$Q = [e, T^T] W_1.$$

Therefore,

$$\mathsf{null}\{\Psi\} = \mathsf{span}[e, T^T].$$

Remark 3.1. For the last example, we see $\mathsf{dim}(\boldsymbol{S}_2) = 1 < \mathsf{dim}(\boldsymbol{S}_1) = 2$ and

$$\left\{ \begin{pmatrix} \hat{\tau}_i \\ \hat{\rho}_i \end{pmatrix} \in \boldsymbol{\Theta}_2 : \begin{pmatrix} \hat{\tau}_i \\ 0 \end{pmatrix} \in \boldsymbol{S}_1 \cap \boldsymbol{S}_2 \right\} = \mathsf{span}\left(\begin{bmatrix} b & c & d \\ u & v & w \end{bmatrix} \right).$$

We notice $\mathsf{dim}\left(\mathsf{span}\left(\begin{bmatrix} b & c & d \\ u & v & w \end{bmatrix}\right)\right) = 2$. Then Θ_1 and Θ_2 satisfy the condition of the theorem and we have $\mathsf{null}\{\Psi\} = \mathsf{span}[e, T^T]$, which verifies the theorem.

3.3 Semi-supervised Alignment of Manifolds

The results in the previous section show that the alignment algorithm is capable of dealing with sections of different dimensions. Other than reconstruction of global coordinates, this result has application in other contexts. Here we consider a problem in semi-supervised learning of manifolds that has been discussed in [10] and [28].

Assume that there are two data sets that admit a pairwise correspondence, some of which are known. The objective is to generate full association (correspondence) of the data sets from the partial association of samples. One approach to this problem is to first generate a common low-dimensional embedding for those two data sets. From the common embedding, we can associate samples between the two data sets.

This semi-supervised learning problem has many applications, including the image comparison [10], cross-lingual information retrieval, bioinformatics [28], and speech analysis [20]. For example, in the image comparison, we have several sets of pictures of different objects taken by a camera from various positions and angles and we wish to match images taken from the same or similar positions/angles, provided matching of the samples is available.

Consider two data sets $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ that are sampled from two d-dimensional manifolds and are represented by vectors in two matrices X_1 and X_2. Let $X_1 = [X_1^\ell, X_1^u]$ and $X_2 = [X_2^\ell, X_2^u]$ and assume that X_1^ℓ and X_2^ℓ are already known to be in pairwise association. We want to determine any possible pairwise associations between X_1^u and X_2^u. This can be done by finding a joint low dimensional parametrization (embedding) for $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, in which the corresponding vectors in X_1^ℓ and X_2^ℓ are mapped to the same coordinates.

First, assume that a low dimensional parametrization for each of $\boldsymbol{X}_i$ is available and let $Z_1 = [Z_1^\ell, Z_1^u]$ and $Z_2 = [Z_2^\ell, Z_2^u]$ be the parameterizations. To find a joint parametrization, we find $T_1 = [T_1^\ell, T_1^u]$ and $T_2 = [T_2^\ell, T_2^u]$ that are respectively affine transformations of Z_1 and Z_2 such that $T_1^\ell = T_2^\ell$. Then, $T = [T^\ell, T_1^u, T_2^u]$ defines a joint parametrization for $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$, where $T^\ell = T_1^\ell = T_2^\ell$. Hence any association between vectors in X_1^u and X_2^u can be derived from that between T_1^u and T_2^u. Clearly, this is an alignment problem and T can be computed from the null space of

$$\Psi = E_1 P_1 E_1^T + E_2 P_2 E_2^T,$$

where P_i is the orthogonal projection onto the orthogonal complement of $\mathsf{span}\{[e, Z_i^T]\}$, and E_i is the selection matrix, such that $TE_i = T_i$.

Note that as in the derivation of LTSA [34], the alignment algorithm can also be regarded as solving

$$\min_{c_i, L_i, T} \sum_{i=1,2} \|[T^\ell, T_i^u] - (c_i e^T + L_i Z_i)\|_F^2,$$

which is reduced to $\min_{TT^T=I, Te=0} \|\Psi T\|_F^2$, or computing the null space of Ψ.

In practice, if we do not have a low dimensional parametrization Z_i available, we can compute them first using the LTSA algorithm, but this can be combined with the alignment process as follows.

We partition $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ into several small neighborhoods $\{\boldsymbol{X}_i^{(1)}, i = 1, \ldots, s_1\}$ and $\{\boldsymbol{X}_i^{(2)}, i = 1, \ldots, s_2\}$. For each small neighborhood, we compute a local parametrization and let $S_i^{(j)}$ be the matrix of local coordinate vectors for $\boldsymbol{X}_i^{(j)}$. Then a joint parametrization $T = [T^\ell, T_1^u, T_2^u]$ can be found by aligning all $\{S_i^{(1)}, 1 \le i \le s_1\}$ and $\{S_i^{(2)}, 1 \le i \le s_2\}$ together using the alignment algorithm. Specifically, let $P_i^{(j)}$ be orthogonal projection onto the orthogonal complement of span$([e, S_i^{(j)}])$ and let

$$\Psi_i^{(j)} = E_i^{(j)} P_i^{(j)} E_i^{(j)T} \tag{3.5}$$

where $E_i^{(j)}$ is the selection matrix, such that $TE_i^{(j)} = T_i^{(j)}$ with the columns of $T_i^{(j)}$ corresponding to the vectors in $\boldsymbol{X}_i^{(j)}$. Then, T is computed from the null space (or the $d+1$ smallest eigenvectors) of

$$\Psi = \sum_{i=1}^{s_1} \Psi_i^{(1)} + \sum_{i=1}^{s_2} \Psi_i^{(2)}. \tag{3.6}$$

Since we have shown in Section 3 that the alignment algorithm works for neighborhoods of different dimension, the above alignment process for semi-supervised learning is applicable to the manifolds or neighborhoods of different dimension. Furthermore, the idea can be easily generalized to matching of n data sets. We state it as the following algorithm.

Algorithm 3.1. Semi-supervised Alignment Algorithm for n Manifolds
Given $\boldsymbol{X}_j \subset \mathbb{R}^m$, $j = 1, \ldots, n$.

1. Construct a fully overlapped covering $\{\boldsymbol{X}_i^{(j)}, i = 1, \ldots, s_i\}$ for $\boldsymbol{X}_j$, j=1, . . . , n.
2. For each $\boldsymbol{X}_i^{(j)}$, construct its local coordinates $S_i^{(j)}$.
3. Construct $\Psi_i^{(j)}$ from $\Theta_i^{(j)}$, $i = 1, \ldots, s_i$ as in (3.5).
4. Construct the matrix Ψ as in (3.6).
5. compute $[e, T^T]$ as an orthogonal basis of the spectral subspace of Ψ corresponding to the smallest $d+1$ eigenvalues.

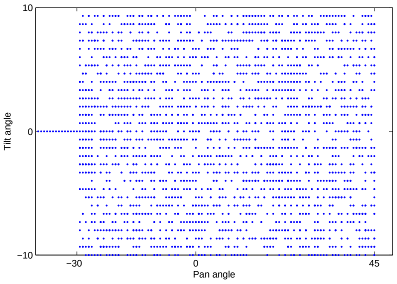


Figure 3.1: The generating coordinates of the data set.

Remark 3.2. Several algorithms have been proposed for this semi-supervised learning problem, such as [10] and [28]. However, they are not capable of working with manifolds of different dimensions, as shown by examples in Section 5. The alignment algorithm that we propose has some other advantages as well. For example, it appears the method of [28] can work with two data sets only. It is also more computationally intensive since it requires computing SVD for the data points already in pairwise correspondence.

3.4 Numerical Examples

In this section, we present two examples to show the alignment algorithm works well with sections having different dimensions. The first example is a manifolds learning problem. We have a set of face images generated from a 3D face model depending on two parameters. We try to find the low-dimensional parametrization for this image set.

Example 3.2. Consider that the data set consists of $N = 2715$ face images generated based on the 3D face model in [3]. The set contains 64×64 images of the same face, which are obtained by varying pan and tilt angles for the observer. For 2700 images, they vary from -30 to 45 degrees of pan angles and -10 to 10 degrees for tilt angles. For 15 images, they vary from -45 to -30 degrees of pan angles and have 0 degree

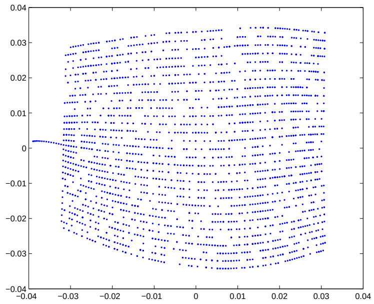


Figure 3.2: The reconstructed coordinates of the data set by LTSA with $k_i = 15$ and $d = 2$.

of tilt angles. We are interested in finding the low-dimensional parametrization for these face images. The original coordinates of all those 2715 pictures are showed in Figure 3.1, where the x-axis is the pan angle and the y-axis is the tilt angle. From Figure 3.1, we can see that those 2700 images lie on a manifold with dimensionality $d = 2$, whereas the other 15 images with the same tilt angles lie on a branch with dimensionality $d = 1$.

We implement LTSA algorithm with fifteen neighbors of each x_i ($k_i = 15$) and dimension two ($d = 2$) to recover the parameters of the images. We notice that for those 15 images with the same tilt angles, their local coordinates should have intrinsic dimensionality one with this example, but our algorithm will treat it as if it were dimension two, having the second components derived from a singular vector corresponding to a tiny singular value. The reconstructed coordinates of all these 2715 images after LTSA are showed in Figure 3.2. Though one set of these data points is of intrinsic dimensionality one and the other set of data points is of intrinsic dimensionality two, LTSA recover the parametrization correctly. The lower dimensional branch is clearly identified in Figure 3.2.

Our second example is a semi-supervised learning problem concerning two sets of face images generated from different face models. We are interested in matching the face images shoot from the same tilt and pan angles.

Example 3.3. We have two sets of pictures generated from two different 3D face models in [3]. The pictures of two different persons are taken from different pan and tilt angle. We are interested in matching the images with the same pan and tilt angles from different image sets. This problem can be solved by computing a joint low dimensional parametrization (Algorithm 3.1) that we discussed in the previous section.

The first data set $\boldsymbol{X}$ consists of 100 pictures coming from face model A and all these pictures have the same tilt angle of 0 degree and pan angles varying from -45 to 45 degrees. The second data set $\boldsymbol{Y}$ contains 2720 pictures generated from face model B. These pictures have pan angles varying from -45 to 45 degrees and tilt angle varying from -10 to 10 degrees . The goal is to match the images with the same tilt angle and the same pan angle. First, 20 matching pairs of pictures in $\boldsymbol{X}$ and $\boldsymbol{Y}$ are manually chosen so that each pair of images are shoot from the same tilt angle and pan angle. These 20 pictures are labeled samples. We implement the alignment algorithms in [10], [28] and Algorithm 4. Each of these algorithms requires setting the number of points in neighborhoods. For our test, we have used ten or twelve points ($k_i = 10$ or $k_i = 12$) and the dimension is set to 2.

For the data set $\boldsymbol{X}$, we notice that the intrinsic dimensionality should be one, whereas we treat them as having dimension two which is necessary in order to carry out alignment. In the left three plots of Figure 3.3, we show the computed joint parametrization of data sets $\boldsymbol{X}$ and $\boldsymbol{Y}$ with 10 neighbors ($k_i = 10$) and dimensions two ($d = 2$) by the algorithms in [10] (top), in [28] (center) and Algorithm 3.1 (bottom). The right three plots of Figure 3.3 show the corresponding results when we use 12 neighbors ($k_i = 12$) and dimensions two ($d = 2$). The red circle line are the joint parametrization of the images in $\boldsymbol{X}$. The blue points are the joint parametrization of the images in $\boldsymbol{Y}$. Our semi-supervised alignment algorithm works well with data sets of different dimensions.

Given one unlabeled sample picture x_i from the data set $\boldsymbol{X}$ as input, which has a parameter z_{x_i} from joint parametrization, we find an image $y_j \in \boldsymbol{Y}$ that associates to x_i by solving

$$y_j = \arg_j \min \|z_{x_i} - z_{y_j}\|_2,$$

where z_{y_j} is the parameter for y_j. For this test, we take five unlabeled sample pictures from $\boldsymbol{X}$ data as the input, which are shown in Figure 3.4(a). The best matching data for $\boldsymbol{Y}$ found by algorithms in [10], in [28] and Algorithm 3.1 are shown in Figure 3.4(b), Figure 3.4(c) and Figure 3.4(d). There is a clear match in the pan and tilt angle for the pairs by our algorithm, while the other two methods obviously have at least some of the pictures mismatched.

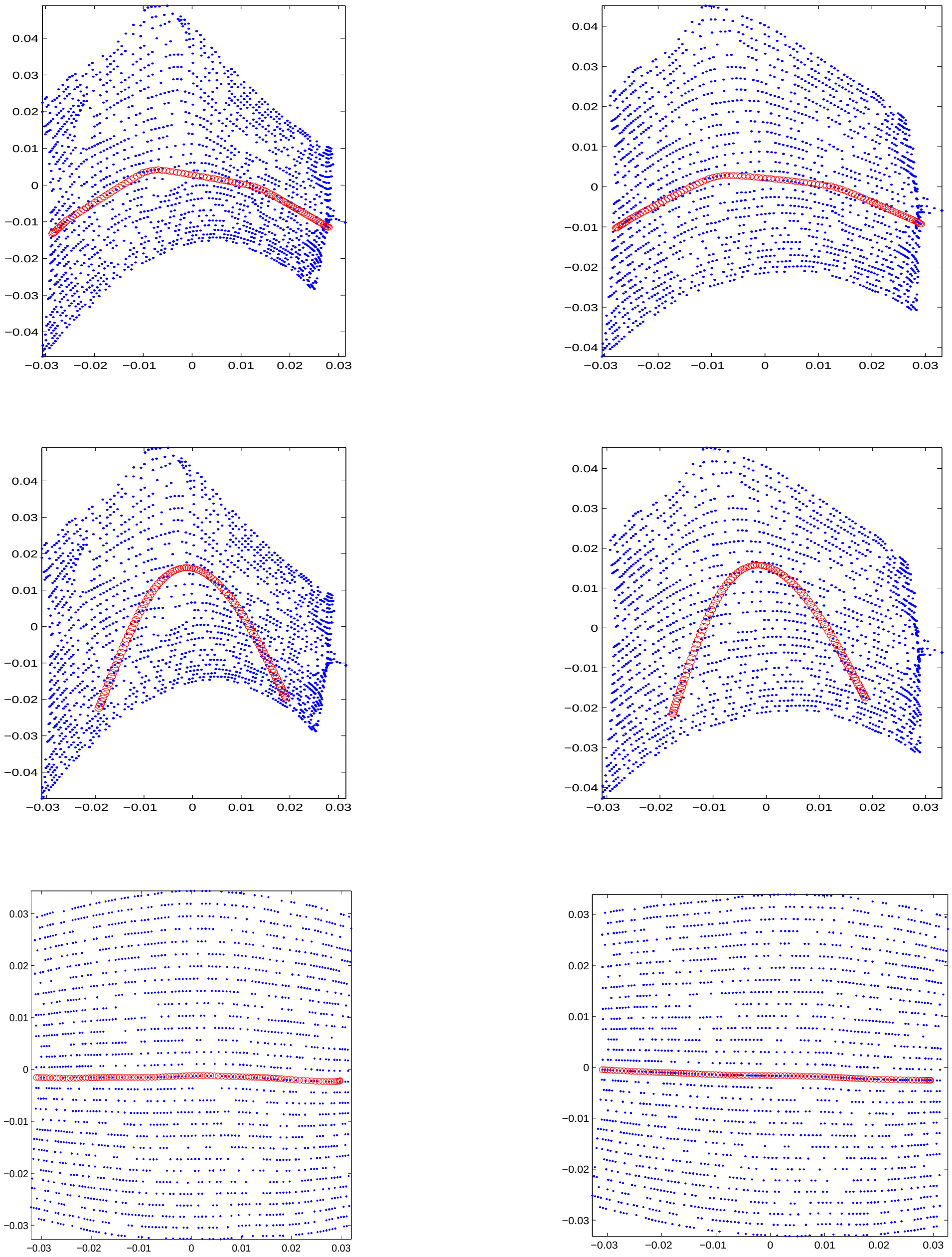
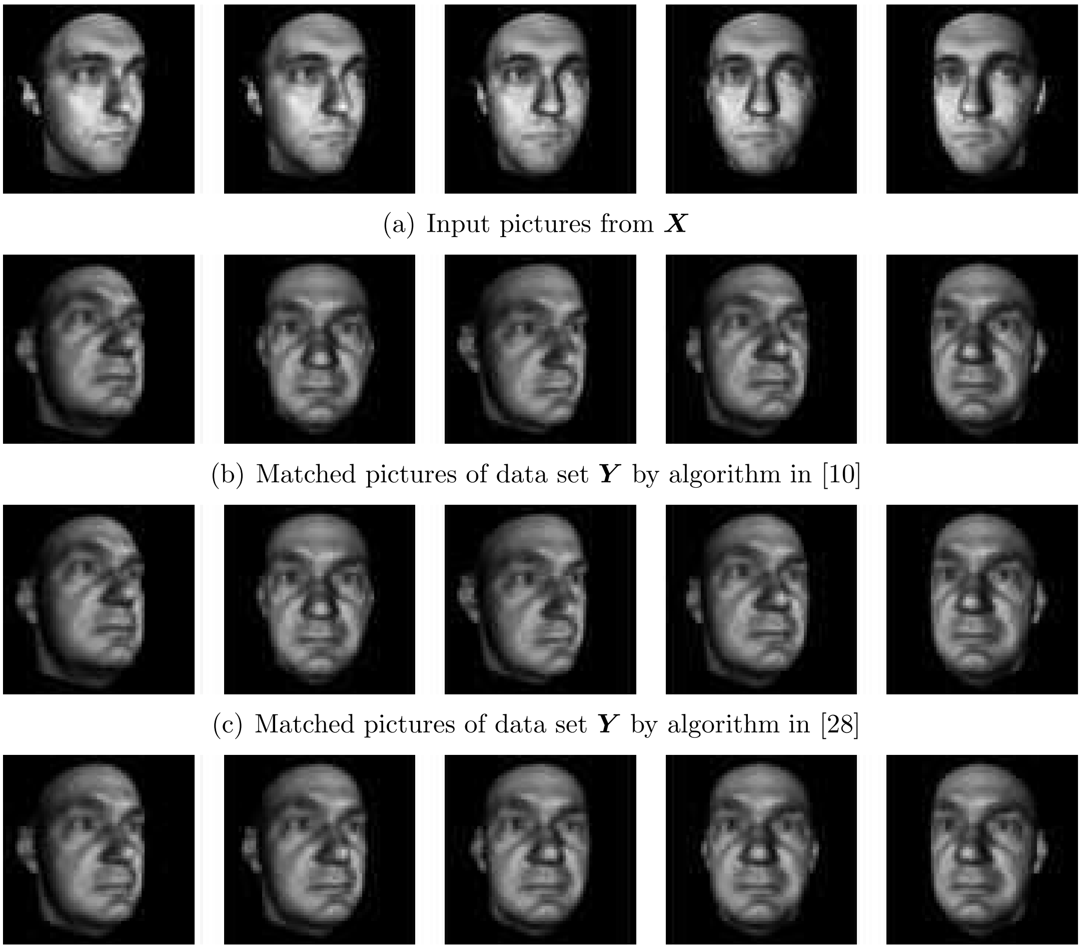


Figure 3.3: Comparison of three algorithms with $k_i = 10$ (left) and $k_i = 12$ (right).



(a) Input pictures from $\boldsymbol{X}$

(b) Matched pictures of data set $\boldsymbol{Y}$ by algorithm in [10]

(c) Matched pictures of data set $\boldsymbol{Y}$ by algorithm in [28]

(d) Matched pictures of data set $\boldsymbol{Y}$ by Algorithm 3.1

Figure 3.4: Original pictures and matched pictures from three algorithms.

Chapter 4 Eigenvalue Bounds for an Alignment Matrix in Manifold Learning

In this chapter, we generalize the results of Section 2.2.4.2 to a general case where some Z_j may be rank deficient and the computed local coordinates (an approximation of P_{Z_j}) may contain large errors in its vanishing components. In this case, the alignment matrix Ψ constructed from the perturbed local coordinates may be significantly different from Φ. We present some characterizations of the eigenvalues of Ψ, from which the null space is determined and a lower bound on the smallest positive eigenvalue is derived. Our results show that Ψ has spectral properties similar to Φ.

4.1 Alignment Matrix for Dimensionality Reduction

In this section, we describe how the alignment matrix arises in the manifold learning problem. We also set the related notation.

Consider the data set (1.1). The LTSA (Local Tangent Space Alignment) method [34] is based on partitioning the points into small local neighborhoods, then constructing coordinates for points in each local neighborhood through linear approximation, and finally aligning the locally constructed coordinates into (global) coordinates for all points. This last step is accomplished through the alignment matrix as follows.

Let $\boldsymbol{X} = \{x_1, \cdots, x_N\}$ and let $\{\boldsymbol{X}_j, j = 1, \ldots, s\}$ be a collection of subsets of $\boldsymbol{X}$ with $\boldsymbol{X}_j = \{x_{j_1}, \ldots, x_{j_{k_j}}\}$ $(j_1 < j_2 < \cdots < j_{k_j})$. Assume that $\cup_j \boldsymbol{X}_j = \boldsymbol{X}$. Each $\boldsymbol{X}_j$ consists of points in a small neighborhood so that a local coordinate system can be approximately constructed through the projection on the local tangent space. More generally, we only need to assume that $\boldsymbol{X}_j$ is any subset for which a local isometric coordinate $\{\theta_1^{(j)}, \ldots, \theta_{k_j}^{(j)}\} \subset \mathbb{R}^d$ can be constructed, i.e. $\|\theta_p^{(j)} - \theta_q^{(j)}\|_2 = \|\tau_{j_p} - \tau_{j_q}\|_2$

(for any $1 \le p, q \le k_j$). From this isometric property and provided that $[\tau_{j_1}, \cdots, \tau_{j_{k_j}}]$ has full row rank, it can be shown [30] that the two coordinate sets differ by a rigid motion, i.e. there is an orthogonal matrix V_j and some $c_j \in \mathbb{R}^d$ such that

$$\Theta_j = V_j T_j + c_j e^T. \tag{4.1}$$

where

$$\Theta_j = \left[\begin{array}{ccc} \theta_1^{(j)}, & \dots, & \theta_{k_j}^{(j)} \end{array}\right], \quad T_j = \left[\begin{array}{ccc} \tau_{j_1}, & \cdots, & \tau_{j_{k_j}} \end{array}\right]. \tag{4.2}$$

In practice, Θ_j is constructed by computing a basis for a local tangent space through the singular value decomposition(SVD) of $[x_{j_1} - \bar{x}, \cdots, x_{j_{k_j}} - \bar{x}]$ where $\bar{x} = (x_{j_1} + \cdots + x_{j_{k_j}})/k_j$. The local coordinate $\theta_p^{(j)}$ is the projection of $x_{j_p} - \bar{x}$ in this basis; see [34].

Let

$$T = \left[\begin{array}{c} \tau_1, \cdots, \tau_N \end{array}\right], \quad Z = [e, T^T], \tag{4.3}$$

and

$$Y_j = [e, \Theta_j^T] \quad \text{and} \quad Z_j = [e, T_j^T]. \tag{4.4}$$

We have defined the alignment matrix for $\{Z_j\}$ in the introduction. The alignment matrix for $\{Y_j\}$ can be defined similarly. Namely, let $P_{Y_j} = Y_j Y_j^\dagger$ be the orthogonal projector in $\mathbb{R}^{k_j}$ onto $\mathsf{span}(Y_j)$ and let $P_{Y_j}^\perp = I - P_{Y_j}$. Let Ψ_j be the embedding of $P_{Y_j}^\perp$ into $\mathbb{R}^{N \times N}$ i.e. $\Psi_j = E_j P_{Y_j}^\perp E_j^T \in \mathbb{R}^{N \times N}$. Note that E_j is defined through the positions of Z_j in Z as in (2.6). Then

$$\Psi = \sum_{j=1}^{s} \Psi_j. \tag{4.5}$$

is called the alignment matrix for $\{Y_j\}$.

It follows from (4.1) that $P_{Y_j} = P_{Z_j}$. Therefore $\Phi = \Psi$ and

$$\mathsf{null}(\Psi) = \mathsf{span}(Z) \tag{4.6}$$

provided that $\{Z_j, 1 \leq j \leq s\}$ satisfies a condition called fully overlapped [30]. In the case of two submatrices, the condition can be easily stated as follows (the case of s submatrices is defined recursively, see Definition 2.4)

Definition 4.1. Two submatrices Z_1 and Z_2 of Z are said to be fully overlapped if the matrix consisting of their common rows has full column rank.

This definition necessarily implies that each Z_j has full column rank. This is equivalent to that the data points in a local neighborhood has dimension ℓ (or their projections into the local tangent space span a full tangent space). Here, we are interested in a practical situation where the local neighborhood may consist of points lying (or nearly lying) in a lower dimensional branch of the manifold. In that case, Z_j will be (or nearly) column rank deficient. Furthermore, the computed Θ_j may not satisfy (4.1) but may contain some large errors in the vanishing components of τ_j. In that case, Ψ_j need not be the same as Φ_j. In a recent paper [31], we have considered some manifold learning problems giving rise to this situation and we have shown that with a proper generalization of the fully overlapped condition, the main result (4.6) on the null space remains intact.

In the next section, we study the spectral properties of Ψ, generalizing the result of Section 2.2.4.2 for Φ to allow the cases that Z_j may be rank deficient or the computed local coordinates may have large error components. As pointed out in the introduction, it is important to bound the smallest positive eigenvalue away from 0 for the computation of the null space (4.6).

4.2 Eigenvalues of Alignment Matrix

In this chapter, we consider a more general fully overlapped condition defined below. Recall that $Z \in \mathbb{R}^{N\times\ell}$ with $N > \ell$ and $Z_j \in \mathbb{R}^{k_j\times\ell}$ (for $1 \leq j \leq s$) are submatrices of

Z. Let

$$\boldsymbol{I}_j = \{j_1, j_2, \ldots, j_{k_j}\} \tag{4.7}$$

be the index set for the rows of Z_j, i.e.,

$$Z_j = Z_{(\boldsymbol{I}_j,:)} = (I_N)_{(\boldsymbol{I}_j,:)} \times Z \in \mathbb{R}^{k_j \times \ell}. \tag{4.8}$$

Assume throughout this chapter that $\bigcup_{j=1}^{s} \boldsymbol{I}_j = \{1, 2, \ldots, N\}$.

Definition 4.2. This definition is recursive.

1. Z_i always fully overlaps itself regardless of its rank;

2. Z_i and Z_j for $i \neq j$ are *fully overlapped*, if one of them has full column rank and $\min\{\mathsf{rank}(Z_i), \mathsf{rank}(Z_j)\} = \mathsf{rank}(Z_{(\boldsymbol{I}_i \bigcap \boldsymbol{I}_j,:)})$;

3. The collection $\boldsymbol{Z} = \{Z_j, 1 \leq j \leq s\}$ for $s \geq 3$ is *fully overlapped*, if it can be partitioned into two nonempty disjoint subsets $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ each of which is a fully overlapped collection and that $Z_{(\widetilde{\boldsymbol{I}}_1,:)}$ and $Z_{(\widetilde{\boldsymbol{I}}_2,:)}$ are fully overlapped, where $\widetilde{\boldsymbol{I}}_i = \bigcup_{Z_j \in \boldsymbol{Z}_i} \boldsymbol{I}_j$.

We shall first present our main results for the case of two submatrices and then discuss how to obtain a bound recursively in the general case in three subsections in Section 4.2. We present some numerical examples to illustrate our bounds in Section 4.3.

4.2.1 Case of Two Submatrices for Φ

As in Section 2.2.4.2, we first analyze Φ as constructed from two submatrices Z_1 and Z_2, but here we only assume the more general definition of the fully overlapped condition (Definition 4.2). Most results follow directly by adapting those of Section

2.2.4.2 to this more general case; we therefore only present the result concerning the null space as an illustration.

Let Z_1 and Z_2 be two fully overlapped submatrices and, without loss of generality, we assume $\mathsf{rank}(Z_1) = \ell \geq \mathsf{rank}(Z_2) = \mathsf{rank}(Z_{(\boldsymbol{I}_1 \bigcap \boldsymbol{I}_{2},:)}) = d_1$. Furthermore, upon permuting rows of Z, we may write

$$Z_1 = \begin{array}{c c} & \ell \\ m_{11} & \multirow{2}{*}{} \\ m_{12} & \end{array} \!\! \begin{array}{cc} & \ell \\ m_{11} & Z_1^{(1)} \\ m_{12} & Z_2^{(1)} \end{array}, \quad Z_2 = \begin{array}{cc} & \ell \\ m_{21} & Z_1^{(2)} \\ m_{22} & Z_2^{(2)} \end{array}, \tag{4.9}$$

where $Z_2^{(1)} = Z_1^{(2)}$ is the common rows in Z_1 and Z_2 and $m_{12} = m_{21}$.

We shall always consider the nontrivial case that $m_{12} \geq 1$, $m_{11} \geq 1$, and $m_{22} \geq 1$. We first show that, using Theorem 2.5, the original theorem on the null space of Φ extends to our more general definition of the fully overlapped condition (Definition 4.2).

Theorem 4.1. Let Z_1 and Z_2 be two fully overlapped submatrices of Z. We have $\mathsf{null}(\Phi) = \mathsf{span}(Z)$.

Proof: First, applying Theorem 2.4, we have the following decompositions

$$Z_1^{(1)} = U_2 \times \begin{array}{c} \\ r_2 \\ m_{11}-r_2 \end{array} \!\! \begin{array}{ccc} d_1 & r_2 & \ell-d_1-r_2 \\ \left(\begin{array}{c} \widetilde{M}_1 \\ M_1 \end{array}\right. & \begin{array}{c} \Sigma_2 \\ 0 \end{array} & \left.\begin{array}{c} 0 \\ 0 \end{array}\right) \end{array} \times \begin{array}{cc} d_1 & \ell-d_1 \\ \left(\begin{array}{c} I \\ 0 \end{array}\right. & \left.\begin{array}{c} 0 \\ V_2^* \end{array}\right) \end{array} V_1^*, \tag{4.10}$$

$$Z_1^{(2)} = Z_2^{(1)} = U_1 \times \begin{array}{c} \\ d_1 \\ m_{12}-d_1 \end{array} \!\! \begin{array}{cc} d_1 & \ell-d_1 \\ \left(\begin{array}{c} \Sigma_1 \\ 0 \end{array}\right. & \left.\begin{array}{c} 0 \\ 0 \end{array}\right) \end{array} V_1^*, \tag{4.11}$$

$$Z_2^{(2)} = U_3 \times \begin{array}{c} \\ r_3 \\ m_{22}-r_3 \end{array} \!\! \begin{array}{ccc} d_1 & r_3 & \ell-d_1-r_3 \\ \left(\begin{array}{c} \widetilde{M}_2 \\ M_2 \end{array}\right. & \begin{array}{c} \Sigma_3 \\ 0 \end{array} & \left.\begin{array}{c} 0 \\ 0 \end{array}\right) \end{array} \times \begin{array}{cc} d_1 & \ell-d_1 \\ \left(\begin{array}{c} I \\ 0 \end{array}\right. & \left.\begin{array}{c} 0 \\ V_3^* \end{array}\right) \end{array} V_1^*, \tag{4.12}$$

where $U_1 \in \mathbb{R}^{(m_{12}\times m_{12})}$, $U_2 \in \mathbb{R}^{(m_{11}\times m_{11})}$, $U_3 \in \mathbb{R}^{(m_{22}\times m_{22})}$, $V_1 \in \mathbb{R}^{(\ell\times\ell)}$, and $V_2 \in \mathbb{R}^{(\ell-d_1)\times(\ell-d_1)}$ and $V_3 \in \mathbb{R}^{(\ell-d_1)\times(\ell-d_1)}$ are orthogonal matrices, Σ_1, Σ_2 and Σ_3 are diagonal matrices with positive diagonal entries. In particular

$$d_1 = \mathsf{rank}(Z_1^{(2)}),\ r_2 = \mathsf{rank}((Z_1^{(1)}V_1)_{(:,d_1+1:\ell)}),\ r_3 = \mathsf{rank}((Z_2^{(2)}V_1)_{(:,d_1+1:\ell)}). \tag{4.13}$$

It follows from the fully overlapped condition that $\mathsf{rank}(Z_2^{(1)}) = \mathsf{rank}(Z_2)$ and hence $r_3 = 0$. Now, using Theorem 2.5, we have $\mathsf{dimnull}(\Phi) = d_1 + r_2$. Since $\mathsf{span}(Z) \subset \mathsf{null}(\Phi)$ and $\mathsf{rank}(Z) = d_1 + r_2$, we obtain $\mathsf{null}(\Phi) = \mathsf{span}(Z)$. □

4.2.2 Case of Two Submatrices for Ψ

In the application of dimensionality reduction, we do not have Z_j's (and T_j's). Instead, we construct Θ_j's (and hence Y_j's) through singular value decompositions (see Section 4.1). If Z_j has full column rank, $Y_j = Z_jH_j$ for some H_j. However, if Z_j is rank deficient, say, $\mathsf{rank}(Z_j) = d_1 < d$, then the corresponding data points lie on a d_1-dimensional branch. When constructing local coordinates for these points, we can only compute the first d_1 coordinates correctly while the remaining coordinates computed are essentially noise. For example, in a typical situation like this, Z_j has the last $\ell - d_1$ columns zero, but Y_j, as computed from the corresponding data, may have random nonzero vectors in its last $\ell - d_1$ columns that result from errors, i.e.

$$Y_j = \begin{array}{c c} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{21} \\ m_{22} \end{array} & \begin{pmatrix} Z_{11}^{(j)} & Y_{12}^{(j)} \\ Z_{21}^{(j)} & Y_{22}^{(j)} \end{pmatrix} \end{array}, \quad \text{if} \quad Z_j = \begin{array}{c c} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{21} \\ m_{22} \end{array} & \begin{pmatrix} Z_{11}^{(j)} & 0 \\ Z_{21}^{(j)} & 0 \end{pmatrix} \end{array}, \tag{4.14}$$

where $Y_{12}^{(j)}, Y_{22}^{(j)}$ are error components. Therefore, when Z_j is rank deficient, it is no longer reasonable to assume and indeed we do not have in practice that $Y_j = Z_jR_j$. What we have is that $\mathsf{span}(Y_j)$ contains $\mathsf{span}(Z_j)$, but it may also contain some other components. We shall therefore assume that

$$\mathsf{span}(Z_j) \subset \mathsf{span}(Y_j),$$

which is equivalent to (4.14) after a column transformation as shown in the lemma below.

For two fully overlapped submatrices Z_1 and Z_2 in the form (4.9), we note that, with a column transformation Z_1R and Z_2R for some nonsingular matrix R, we can write Z_1 and Z_2 as

$$Z_1 = \begin{array}{cc} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{11} \\ m_{12} \end{array} & \begin{pmatrix} Z_{11}^{(1)} & Z_{12}^{(1)} \\ Z_{21}^{(1)} & 0 \end{pmatrix} \end{array}, \quad Z_2 = \begin{array}{cc} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{21} \\ m_{22} \end{array} & \begin{pmatrix} Z_{11}^{(2)} & 0 \\ Z_{21}^{(2)} & 0 \end{pmatrix} \end{array}, \tag{4.15}$$

where $Z_{21}^{(1)} = Z_{11}^{(2)}$ and $d_1 = \mathsf{rank}(Z_2)$.

Lemma 4.1. Let $Z_1 \in \mathbb{R}^{k_1\times\ell}$ and $Z_2 \in \mathbb{R}^{k_2\times\ell}$ be two submatrices of Z in the form (4.15) with $\mathsf{rank}(Z_2) = d_1 \le \mathsf{rank}(Z_1) = l$. Let $Y_1 \in \mathbb{R}^{k_1\times\ell}$ and $Y_2 \in \mathbb{R}^{k_2\times\ell}$ be such that $\mathsf{span}(Z_j) \subset \mathsf{span}(Y_j)$. Then, there are some nonsingular matrices H_1 and H_2 such that

$$Y_1 = Z_1H_1, \quad Y_2 = \begin{array}{cc} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{21} \\ m_{22} \end{array} & \begin{pmatrix} Z_{11}^{(2)} & Y_{12}^{(2)} \\ Z_{21}^{(2)} & Y_{22}^{(2)} \end{pmatrix} \end{array} H_2. \tag{4.16}$$

Proof: Since $\mathsf{rank}(Z_1) = \ell$, we have $\mathsf{span}(Z_1) = \mathsf{span}(Y_1)$ and then $Y_1 = Z_1H_1$ for some nonsingular $H_1 \in \mathbb{R}^{\ell\times\ell}$. For Y_2, let $\hat{Y}_2 \in \mathbb{R}^{k_2\times p}$ with $p \le \ell - d_1$ be such that the nonzero columns of Z_2 and the columns of $\hat{Y}_2$ form a basis for $\mathsf{span}(Y_2)$. Then there is some full row rank matrix $\hat{H}_2$ such that

$$Y_2 = \begin{array}{cc} & \begin{array}{cc} d_1 & \ell-d_1 \end{array} \\ \begin{array}{c} m_{21} \\ m_{22} \end{array} & \begin{pmatrix} Z_{11}^{(2)} & Y_{12}^{(2)} \\ Z_{21}^{(2)} & Y_{22}^{(2)} \end{pmatrix} \end{array} \hat{H}_2 \quad \text{where} \quad \hat{Y}_2 = \begin{pmatrix} Y_{12}^{(2)} \\ Y_{22}^{(2)} \end{pmatrix}.$$

Now, if $p = \ell - d_1$ the lemma is proved with $H_2 = \hat{H}_2$. If $p < \ell - d_1$, we can append $\ell - d_1 - p$ zero columns to $\hat{Y}_2$ and correspondingly some rows to $\hat{H}_2$ to obtain the equation (4.16) for Y_2 with a nonsingular H_2. The lemma is proved. □

We now proceed to calculate $\mathsf{span}(Y_j)$ and hence Ψ_j through a sequence of simplifications of Y_j using column transformations. First, the following lemma follows directly from the QR factorization with column pivoting.

Lemma 4.2. Let $W \in \mathbb{R}^{p\times q}$ and $\mathsf{rank}(W) = r$. Then there exist $Q \in \mathbb{R}^{p\times r}$ with orthonormal columns and a nonsingular matrix $R \in \mathbb{R}^{q\times q}$ such that

$$WR = \begin{array}{cc} r & n-r \\ \left[\, Q, \right. & \left. 0 \,\right] \end{array}. \tag{4.17}$$

Lemma 4.3. Under the assumptions of Lemma 4.1, there are two nonsingular matrices R_1 and R_2 such that

$$Y_1R_1 = W_1 := \left(\begin{array}{ccc} & \widetilde{d}_1 & \ell-\widetilde{d}_1 \\ m_{11} & W_{11}^{(1)} & W_{12}^{(1)} \\ m_{12} & W_{21}^{(1)} & 0 \end{array}\right),$$

and

$$Y_2R_2 = W_2 := \left(\begin{array}{ccc} & \widetilde{d}_1 & \ell-\widetilde{d}_1 \\ m_{21} & W_{11}^{(2)} & W_{12}^{(2)} \\ m_{22} & W_{21}^{(2)} & W_{22}^{(2)} \end{array}\right),$$

where $W_{21}^{(1)} = W_{11}^{(2)}$, $W_{12}^{(2)} = \begin{array}{cc} \widetilde{d}_2-\widetilde{d}_1 & \ell-\widetilde{d}_2 \\ \left[\, \widetilde{W}_{12}^{(2)}, \right. & \left. 0 \,\right] \end{array}$ with $\left[W_{11}^{(2)},\ \widetilde{W}_{12}^{(2)}\right]$ having orthonormal columns, $\widetilde{d}_1 = \mathsf{rank}(Z_{11}^{(2)})$, and $\widetilde{d}_2 = \mathsf{rank}([Z_{11}^{(2)}, Y_{12}^{(2)}])$.

Proof: With Lemma 4.1, we can assume without loss of generality that

$$Y_1 = \left(\begin{array}{ccc} & d_1 & \ell-d_1 \\ m_{11} & Z_{11}^{(1)} & Z_{12}^{(1)} \\ m_{12} & Z_{21}^{(1)} & 0 \end{array}\right), \quad Y_2 = \left(\begin{array}{ccc} & d_1 & \ell-d_1 \\ m_{21} & Z_{11}^{(2)} & Y_{12}^{(2)} \\ m_{22} & Z_{21}^{(2)} & Y_{22}^{(2)} \end{array}\right). \tag{4.18}$$

From Lemma 4.2, there is a nonsingular matrix $\widetilde{R}_1$ such that

$$Z_{11}^{(2)}\widetilde{R}_1 = \begin{array}{cc} \widetilde{d}_1 & d_1-\widetilde{d}_1 \\ \left[\, W_{11}^{(2)}, \right. & \left. 0 \,\right] \end{array},$$

where $W_{11}^{(2)}$ has orthonormal columns. Furthermore, we also have a nonsingular matrix R_2 such that

$$\left[\begin{array}{cc} Z_{11}^{(2)}, & Y_{12}^{(2)} \end{array}\right] R_2 = \begin{array}{c} \begin{array}{ccc} \scriptstyle \widetilde{d}_1 & \scriptstyle \widetilde{d}_2-\widetilde{d}_1 & \scriptstyle \ell-\widetilde{d}_2 \end{array} \\ \left[\begin{array}{ccc} W_{11}^{(2)}, & \widetilde{W}_{12}^{(2)}, & 0 \end{array}\right] \end{array},$$

with $[W_{11}^{(2)}, \widetilde{W}_{12}^{(2)}]$ having orthonormal columns. Then the lemma follows with

$$R_1 = \begin{array}{c c} & \begin{array}{cc} \scriptstyle d_1 & \scriptstyle \ell-d_1 \end{array} \\ \begin{array}{c} \scriptstyle d_1 \\ \scriptstyle \ell-d_1 \end{array} & \begin{pmatrix} \widetilde{R}_1 & 0 \\ 0 & I \end{pmatrix} \end{array}.$$

□

Lemma 4.4. Under the assumptions and notation of Lemma 4.3, let $W_1^{(1)} = \left[\begin{array}{cc} W_{11}^{(1)}, & W_{12}^{(1)} \end{array}\right]$, $W_2^{(1)} = \left[\begin{array}{cc} W_{21}^{(1)}, & 0 \end{array}\right]$, $W_1^{(2)} = \left[\begin{array}{cc} W_{11}^{(2)}, & W_{12}^{(2)} \end{array}\right]$ and $W_2^{(2)} = \left[\begin{array}{cc} W_{21}^{(2)}, & W_{22}^{(2)} \end{array}\right]$. Then, we have the following decompositions

$$W_2^{(1)} = U_1 \times \begin{array}{c c} & \begin{array}{cc} \scriptstyle \widetilde{d}_1 & \scriptstyle \ell-\widetilde{d}_1 \end{array} \\ \begin{array}{c} \scriptstyle \widetilde{d}_1 \\ \scriptstyle m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \tag{4.19}$$

$$W_1^{(2)} = U_1 \times \begin{array}{c c} & \begin{array}{cc} \scriptstyle \widetilde{d}_2 & \scriptstyle \ell-\widetilde{d}_2 \end{array} \\ \begin{array}{c} \scriptstyle \widetilde{d}_2 \\ \scriptstyle m_{12}-\widetilde{d}_2 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \tag{4.20}$$

$$W_1^{(1)} = U_2 \times \begin{array}{c c} & \begin{array}{ccc} \scriptstyle \widetilde{d}_1 & \scriptstyle r_1 & \scriptstyle \ell-\widetilde{d}_1-r_1 \end{array} \\ \begin{array}{c} \scriptstyle r_1 \\ \scriptstyle m_{11}-r_1 \end{array} & \begin{pmatrix} \widetilde{M}_1 & \Sigma_1 & 0 \\ M_1 & 0 & 0 \end{pmatrix} \end{array} \times \begin{array}{c} \begin{array}{cc} \scriptstyle \widetilde{d}_1 & \scriptstyle \ell-\widetilde{d}_1 \end{array} \\ \begin{pmatrix} I & 0 \\ 0 & V_2^* \end{pmatrix} \end{array} \tag{4.21}$$

$$W_2^{(2)} = U_3 \times \begin{array}{c c} & \begin{array}{ccc} \scriptstyle \widetilde{d}_2 & \scriptstyle r_2 & \scriptstyle \ell-\widetilde{d}_2-r_2 \end{array} \\ \begin{array}{c} \scriptstyle r_2 \\ \scriptstyle m_{22}-r_2 \end{array} & \begin{pmatrix} \widetilde{M}_2 & \Sigma_2 & 0 \\ M_2 & 0 & 0 \end{pmatrix} \end{array} \times \begin{array}{c} \begin{array}{cc} \scriptstyle \widetilde{d}_2 & \scriptstyle \ell-\widetilde{d}_2 \end{array} \\ \begin{pmatrix} I & 0 \\ 0 & V_3^* \end{pmatrix} \end{array} \tag{4.22}$$

where $U_1 \in \mathbb{R}^{(m_{12} \times m_{12})}$, $U_2 \in \mathbb{R}^{(m_{11} \times m_{11})}$, $U_3 \in \mathbb{R}^{(m_{22} \times m_{22})}$, $V_2 \in \mathbb{R}^{(\ell-\widetilde{d}_1)\times(\ell-\widetilde{d}_1)}$ and $V_3 \in \mathbb{R}^{(\ell-\widetilde{d}_2)\times(\ell-\widetilde{d}_2)}$ are orthogonal matrices, Σ_1 and Σ_2 are diagonal matrices with positive diagonal entries. In particular,

$$r_1 = \mathsf{rank}\left((W_1^{(1)})_{(:,\widetilde{d}_1+1:\ell)}\right),\; r_2 = \mathsf{rank}\left((W_2^{(2)})_{(:,\widetilde{d}_2+1:\ell)}\right). \tag{4.23}$$

Proof: Let Q be such that $U_1 := \left[\; W_{11}^{(2)},\;\; \widetilde{W}_{12}^{(2)},\;\; Q \;\right] \in \mathbb{R}^{m_{21}\times m_{21}}$ is a square orthogonal matrix. This immediately leads to (4.19) and (4.20). Let

$$W_{12}^{(1)} = U_2 \times \begin{array}{cc} & \begin{array}{cc} r_1 & \ell-\widetilde{d}_1-r_1 \end{array} \\ \begin{array}{c} r_1 \\ m_{11}-r_1 \end{array} & \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} \end{array} V_2^* \tag{4.24}$$

be the SVD of $W_{12}^{(1)}$. Now noting that $U_2^* W_1^{(1)} = \left[U_2^* W_{11}^{(1)},\; U_2^* W_{12}^{(1)}\right]$ and using (4.24), we have (4.21) with $\widetilde{M}_1$ and M_1 being the top r_1 rows and the bottom $m_{11}-r_1$ rows of $U_2^* W_{11}^{(1)}$, respectively. Similarly let the SVD of the submatrix consisting of the last $\ell-\widetilde{d}_2$ columns of $W_2^{(2)}$ be

$$(W_2^{(2)})_{(:,\widetilde{d}_2+1:\ell)} = U_3 \times \begin{array}{cc} & \begin{array}{cc} r_2 & \ell-\widetilde{d}_2-r_2 \end{array} \\ \begin{array}{c} r_2 \\ m_{22}-r_2 \end{array} & \begin{pmatrix} \Sigma_2 & 0 \\ 0 & 0 \end{pmatrix} \end{array} V_3^* \tag{4.25}$$

This leads to (4.22) with $\widetilde{M}_2$ and M_2 being the top r_2 rows and the bottom $m_{22}-r_2$ rows of $U_3^*(W_2^{(2)})_{(:,1:\widetilde{d}_2)}$, respectively. □

We now calculate the spaces spanned by W_1 and W_2. Note that $\mathsf{span}(Y_j) = \mathsf{span}(W_j)$ by Lemma 4.3. Below, recall that we use $Y \overset{\text{cols}}{\Leftrightarrow} Z$ to denote $\mathsf{span}(Y) = \mathsf{span}(Z)$.

First, from Lemma 4.4, (4.11) and the fact that

$$\begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & \ell-\widetilde{d}_1 \end{array} \\ \begin{array}{c} \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \times \begin{array}{c} \begin{array}{cc} \widetilde{d}_1 & \ell-\widetilde{d}_1 \end{array} \\ \begin{pmatrix} I & 0 \\ 0 & V_2^* \end{pmatrix} \end{array} = \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & \ell-\widetilde{d}_1 \end{array} \\ \begin{array}{c} \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array},$$

we have

$$W_2^{(1)} = U_1 \times \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & \ell-\widetilde{d}_1 \end{array} \\ \begin{array}{c} \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \times \begin{array}{c} \begin{array}{cc} \widetilde{d}_1 & \ell-\widetilde{d}_1 \end{array} \\ \begin{pmatrix} I & 0 \\ 0 & V_2^* \end{pmatrix} \end{array}. \tag{4.26}$$

Then we obtain

$$\begin{pmatrix} U_2^* & \\ & U_1^* \end{pmatrix}\begin{pmatrix} W_1^{(1)} \\ W_2^{(1)} \end{pmatrix} \overset{\text{cols}}{\Leftrightarrow} \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & r_1 \end{array} \\ \begin{array}{c} r_1 \\ m_{11}-r_1 \\ \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} \widetilde{M_1} & \Sigma_1 \\ M_1 & 0 \\ I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \overset{\text{cols}}{\Leftrightarrow} \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & r_1 \end{array} \\ \begin{array}{c} r_1 \\ m_{11}-r_1 \\ \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} \widetilde{M_1} & I \\ M_1 & 0 \\ I & 0 \\ 0 & 0 \end{pmatrix} \end{array}$$

$$\overset{\text{cols}}{\Leftrightarrow} \quad F_1 := \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_1 & r_1 \end{array} \\ \begin{array}{c} r_1 \\ m_{11}-r_1 \\ \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} 0 & I \\ M_1 & 0 \\ I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \begin{pmatrix} E_1 & \\ & I \end{pmatrix}, \tag{4.27}$$

where

$$E_1 = (I + M_1^* M_1)^{-1/2}. \tag{4.28}$$

It is easy to check that

$$F_1^{\perp} := \begin{array}{cc} & \begin{array}{cc} m_{11}-r_1 & m_{12}-\widetilde{d}_1 \end{array} \\ \begin{array}{c} r_1 \\ m_{11}-r_1 \\ \widetilde{d}_1 \\ m_{12}-\widetilde{d}_1 \end{array} & \begin{pmatrix} 0 & 0 \\ I & 0 \\ -M_1^* & 0 \\ 0 & I \end{pmatrix} \end{array} \begin{pmatrix} D_1 & \\ & I \end{pmatrix} \quad \text{with } D_1 = (I + M_1 M_1^*)^{-1/2}$$

has orthonormal columns and spans the orthogonal complement of $\mathsf{span}(F_1)$. Then $\begin{pmatrix} U_2 & \\ & U_1 \end{pmatrix} F_1^{\perp}$ spans the orthogonal complement of $\mathsf{span}(W_1)$.

Similarly, for W_2, we have

$$W_1^{(2)} = U_1 \times \begin{array}{cc} & \begin{array}{cc} \widetilde{d}_2 & \ell-\widetilde{d}_2 \end{array} \\ \begin{array}{c} \widetilde{d}_2 \\ m_{12}-\widetilde{d}_2 \end{array} & \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \end{array} \times \begin{array}{c} \begin{array}{cc} \widetilde{d}_2 & \ell-\widetilde{d}_2 \end{array} \\ \begin{pmatrix} I & 0 \\ 0 & V_3^* \end{pmatrix} \end{array}. \tag{4.29}$$

By (4.22) and (4.29),

$$\begin{pmatrix} U_1^* & \\ & U_3^* \end{pmatrix} \begin{pmatrix} W_1^{(2)} \\ W_2^{(2)} \end{pmatrix} \overset{\text{cols}}{\Leftrightarrow} \begin{array}{c|cc} & \tilde{d}_2 & r_2 \\ \hline \tilde{d}_2 & I & 0 \\ m_{12}-\tilde{d}_2 & 0 & 0 \\ r_2 & \widetilde{M_2} & \Sigma_2 \\ m_{22}-r_2 & M_2 & 0 \end{array} \overset{\text{cols}}{\Leftrightarrow} F_2 := \begin{array}{c|cc} & \tilde{d}_2 & r_2 \\ \hline \tilde{d}_2 & I & 0 \\ m_{12}-\tilde{d}_2 & 0 & 0 \\ r_2 & 0 & I \\ m_{22}-r_2 & M_2 & 0 \end{array} \begin{pmatrix} E_2 & \\ & I \end{pmatrix}, \tag{4.30}$$

where

$$E_2 = (I + M_2^* M_2)^{-1/2}. \tag{4.31}$$

Let

$$F_2^{\perp} := \begin{array}{c|cc} & m_{22}-r_2 & m_{12}-\tilde{d}_2 \\ \hline \tilde{d}_2 & -M_2^* & 0 \\ m_{12}-\tilde{d}_2 & 0 & I \\ r_2 & 0 & 0 \\ m_{22}-r_2 & I & 0 \end{array} \begin{pmatrix} D_2 & \\ & I \end{pmatrix} \quad \text{with } D_2 = (1 + M_2 M_2^*)^{-1/2}.$$

Then, $F_2^{\perp}$ has orthonormal columns and spans the orthogonal complement of $\mathsf{span}(F_2)$. It follows that $\begin{pmatrix} U_1 & \\ & U_3 \end{pmatrix} F_1^{\perp}$ spans the orthogonal complement of $\mathsf{span}(W_2)$. Let

$$G_1 = \begin{array}{c|c} m_{11}+m_{12} & F_1^{\perp} \\ m_{22} & 0 \end{array}, \quad G_2 = \begin{array}{c|c} m_{11} & 0 \\ m_{12}+m_{22} & F_2^{\perp} \end{array},$$

and let

$$G = (G_1 \quad G_2) = \begin{array}{c|cccc} & m_{11}-r_1+\tilde{d}_2-\tilde{d}_1 & m_{12}-\tilde{d}_2 & m_{22}-r_2 & m_{12}-\tilde{d}_2 \\ \hline r_1 & 0 & 0 & 0 & 0 \\ m_{11}-r_1 & D & 0 & 0 & 0 \\ \tilde{d}_2 & M & 0 & -M_2^* D_2 & 0 \\ m_{12}-\tilde{d}_2 & 0 & I & 0 & I \\ r_2 & 0 & 0 & 0 & 0 \\ m_{22}-r_2 & 0 & 0 & D_2 & 0 \end{array},$$

where

$$D = \begin{array}{c c} & \begin{array}{cc} {\scriptstyle m_{11}-r_1} & {\scriptstyle \tilde{d}_2-\tilde{d}_1} \end{array} \\ {\scriptstyle m_{11}-r_1} & \left(\begin{array}{cc} D_1 & 0 \end{array}\right) \end{array}$$

and

$$M = \begin{array}{c c} & \begin{array}{cc} {\scriptstyle m_{11}-r_1} & {\scriptstyle \tilde{d}_2-\tilde{d}_1} \end{array} \\ \begin{array}{c} {\scriptstyle \tilde{d}_1} \\ {\scriptstyle \tilde{d}_2-\tilde{d}_1} \end{array} & \left(\begin{array}{cc} -M_1^* D_1 & 0 \\ 0 & I \end{array}\right) \end{array}.$$

Set

$$Q := \begin{array}{c c} & \begin{array}{ccc} {\scriptstyle m_{11}} & {\scriptstyle m_{12}} & {\scriptstyle m_{22}} \end{array} \\ \begin{array}{c} {\scriptstyle m_{11}} \\ {\scriptstyle m_{12}} \\ {\scriptstyle m_{22}} \end{array} & \left(\begin{array}{ccc} U_2 & & \\ & U_1 & \\ & & U_3 \end{array}\right) \end{array}. \tag{4.32}$$

Then Q is an orthogonal matrix and it can be checked $QG_iG_i^*Q^* = \Psi_i$. Let

$$\widetilde{\Psi} := Q^*\Psi Q = Q^*\Psi_1 Q + Q^*\Psi_2 Q = G_1G_1^* + G_2G_2^* = GG^*. \tag{4.33}$$

Then the null space of $\widetilde{\Psi}$ is the same as the null space of G^*, which is the same as the orthogonal complement of the column space of G.

We construct the orthogonal complement of $\mathsf{span}(G)$. Let

$$G_3 = \begin{array}{c c} & \begin{array}{ccc} {\scriptstyle r_1} & {\scriptstyle \tilde{d}_1} & {\scriptstyle r_2} \end{array} \\ \begin{array}{c} {\scriptstyle r_1} \\ {\scriptstyle m_{11}-r_1} \\ {\scriptstyle \tilde{d}_1} \\ {\scriptstyle m_{12}-\tilde{d}_1} \\ {\scriptstyle r_2} \\ {\scriptstyle m_{22}-r_2} \end{array} & \left(\begin{array}{ccc} I & 0 & 0 \\ 0 & M_1 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I \\ 0 & (M_2)_{(:,1:\widetilde{d}_1)} & 0 \end{array}\right) \end{array}.$$

Note that in G, the 4th block column is the same as the 2nd one, and the first 3 block columns are linearly independent. Therefore $\mathsf{rank}(G) = m_{11}+m_{12}+m_{22}-(r_1+r_2+\widetilde{d}_1)$ which implies $\mathsf{dimnull}(G^*) = r_1+r_2+\widetilde{d}_1$. Evidently, $\mathsf{rank}(G_3) = r_1+r_2+\widetilde{d}_1$. Therefore

$\mathsf{null}(\widetilde{\Psi}) = \mathsf{null}(G^*) = \mathsf{span}(G_3)$ because $G^*G_3 = 0$. We have proved the first two parts of the following theorem on the null space of Ψ.

Theorem 4.2. Under the assumptions of Lemma 4.1, let all symbols keep their assignments so far in this section. Then

1. $\mathsf{dimnull}(\Psi) = \mathsf{dimnull}(\widetilde{\Psi}) = r_1 + r_2 + \widetilde{d}_1$;

2. $\mathsf{null}(\widetilde{\Psi})$ is the column space of G_3 and $\mathsf{null}(\Psi) = Q\,\mathsf{null}(\widetilde{\Psi})$;

3. $\mathsf{null}(\Psi) = \mathsf{span}(Z)$ if and only if $\mathsf{rank}(Y_1^{(2)}) = \mathsf{rank}(Y_2)$ where

$$Y_2 = \begin{array}{c c} & \ell \\ m_{21} & \left(\begin{matrix} Y_1^{(2)} \\ Y_2^{(2)} \end{matrix}\right. \\ m_{22} & \end{array}\!\!\left.\vphantom{\begin{matrix} Y_1^{(2)} \\ Y_2^{(2)} \end{matrix}}\right). \tag{4.34}$$

Proof: We only need to prove part 3. If $\mathsf{rank}(Y_1^{(2)}) = \mathsf{rank}(Y_2)$, then $\mathsf{rank}(W_1^{(2)}) = \mathsf{rank}(W_2) = \tilde{d}_2$. Combining this with Lemma 3.4, we have $r_2 = 0$, which implies $\mathsf{dimnull}(\Psi) = r_1 + \widetilde{d}_1$ by part 1. Since Y_1 has full column rank, $r_1 + \widetilde{d}_1 = \ell$. Now $\mathsf{dimspan}(Z) = \ell$ and $\mathsf{span}(Z) \subset \mathsf{null}(\Psi)$ imply $\mathsf{null}(\Psi) = \mathsf{span}(Z)$. On the other hand, suppose $\mathsf{rank}(Y_1^{(2)}) < \mathsf{rank}(Y_2)$. It implies $r_2 > 0$. Then we have $\mathsf{dimnull}(\Psi) = \widetilde{d}_1 + r_1 + r_2 > \widetilde{d}_1 + r_1$. Thus $\mathsf{null}(\Psi) \neq \mathsf{span}(Z)$. □

Note that Part 3 (the *if* part) has been shown in [31] directly. Next, we discuss the eigenvalues of Ψ or GG^* (see (4.32)), which has the same nonzero eigenvalues as G^*G. We note that

$$G^*G = \begin{array}{c} \\ m_{11}-r_1+\tilde{d}_2-\tilde{d}_1 \\ m_{12}-\tilde{d}_2 \\ m_{22}-r_2 \\ m_{12}-\tilde{d}_2 \end{array} \begin{array}{c} \begin{array}{cccc} m_{11}-r_1+\tilde{d}_2-\tilde{d}_1 & m_{12}-\tilde{d}_2 & m_{22}-r_2 & m_{12}-\tilde{d}_2 \end{array} \\ \left(\begin{array}{cccc} I & 0 & -MM_2^*D_2 & 0 \\ 0 & I & 0 & I \\ -D_2M_2M^* & 0 & I & 0 \\ 0 & I & 0 & I \end{array}\right) \end{array}$$

is permutationally similar to a direct sum of

$$\begin{pmatrix} I & I \\ I & I \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} I & -MM_2^*D_2 \\ -D_2M_2M^* & I \end{pmatrix}.$$

The former matrix has positive eigenvalue 2 with multiplicity $m_{12} - \widetilde{d}_2$; the latter matrix has eigenvalues $1 \pm \sigma_j$ for $\leq j \leq k$, where $\sigma_1, \ldots, \sigma_k$ are the nonzero singular values of $D_2M_2M^*$, and the remaining eigenvalues equal to 1. Thus we have proved the following main theorem of this paper.

Theorem 4.3. Under the assumptions of Lemma 4.1, let all symbols keep their assignments so far in this section. Let the nonzero singular values of $D_2M_2M^*$ be $\sigma_1, \sigma_2, \ldots, \sigma_k$. Then the eigenvalues of Ψ are

$$\begin{array}{lll} 1 \pm \sigma_j & \text{for} \quad 1 \leq j \leq k & \\ 2 & \text{with multiplicity} & m_{12} - \widetilde{d}_2 \\ 1 & \text{with multiplicity} & m_{11} + m_{22} - r_1 - r_2 + \widetilde{d}_2 - \widetilde{d}_1 - 2k \\ 0 & \text{with multiplicity} & r_1 + \widetilde{d}_1 + r_2 \end{array}$$

Using this theorem, we can bound the smallest positive eigenvalue of Ψ as follows. For the singular value σ_j of $D_2M_2M^*$, we have

$$\begin{aligned} \sigma_j &\leq \|D_2M_2M^*\|_2 \\ &\leq \|D_2M_2\|_2 \, \|M\|_2, \\ &\leq \frac{\|M_2\|_2}{\sqrt{1 + \|M_2\|_2^2}}. \end{aligned} \tag{4.35}$$

By Lemma 3.3, $Y_i^{(2)}R_2 = W_i^{(2)}$. If we make additional assumption that $\mathsf{rank}(Y_1^{(2)}) = \mathsf{rank}(Y_2) = \ell$, then $\widetilde{d}_2 = \ell$ and $W_1^{(2)}$ has orthonormal columns. Then $(Y_1^{(2)})^\dagger = R_2(W_1^{(2)})^T$. It follows from (4.20) and (4.22) with $r_2 = 0$ that

$$Y_2^{(2)}(Y_1^{(2)})^\dagger = W_2^{(2)}(W_1^{(2)})^\dagger = U_3M_2U_1^*.$$

It follows that

$$\sigma_j \leq \frac{\|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2}{\sqrt{1 + \|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2^2}}.$$

We have proved the following corollary that bounds the smallest positive eigenvalue in terms of $\|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2$.

Corollary 4.1. Under the assumptions of Lemma 4.1, let all symbols keep their assignments so far in this section. Assume that $\mathsf{rank}(Y_1^{(2)}) = \mathsf{rank}(Y_2) = \ell$. Then the positive eigenvalues of Ψ is no smaller than $1 - \tau$ where

$$\tau \overset{\text{def}}{=} \frac{\|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2}{\sqrt{1 + \|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2^2}}. \tag{4.36}$$

Its largest eigenvalue is no greater than $1 + \tau$ if $m_{12} = \widetilde{d}_2$ and it is 2 if $m_{12} > \widetilde{d}_2$. In particular,

$$\lambda_{\min}^+(\Psi)\, P_Z^\perp \preceq \Psi \preceq \lambda_{\max}(\Psi)\, P_Z^\perp,$$

where $\lambda_{\min}^+(\Psi)$ is the smallest positive eigenvalue of Ψ, and $\lambda_{\max}(\Psi)$ is the largest eigenvalue of Ψ.

Let $t = 1/\|Y_2^{(2)}(Y_1^{(2)})^\dagger\|_2 \le \sigma_{\min}(Y_1^{(2)})/\sigma_{\max}(Y_2^{(2)})$. Then t is a measure of "amount" of overlap. We can write (4.36) as

$$\lambda_{\min}^+(\Psi) \ge 1 - \frac{1}{\sqrt{1+t^2}} \ge \frac{t^2}{2(1+t^2)}.$$

The lower bound implies that if $Y_1^{(2)}$ has full column rank but with nearly linearly dependent columns, $\sigma_{\min}(Y_1^{(2)})$ is small and the smallest positive eigenvalue $\lambda_{\min}^+(\Psi)$ may be nearly zero. In particular, $\lambda_{\min}^+(\Psi)$ may be of order t^2.

4.2.3 Case of s Submatrices

We now generalize the result of Section 4.2.2 to the case of s submatrices $\boldsymbol{Z} = \{Z_j, 1 \le j \le s\}$ where $Z_j = Z_{(\boldsymbol{I}_j,:)} \in \mathbb{R}^{k_j \times \ell}$ is a submatrix of Z; see (4.8). Assume that $\boldsymbol{Z} = \{Z_j, 1 \le j \le s\}$ is a fully overlapped collection. From Definition 4.2, through the recursive process, for each Z_j, there is a subset $\boldsymbol{Z}_j \subset \boldsymbol{Z}$ such that Z_j and $Z_{(\widetilde{\boldsymbol{I}}_j,:)}$ are fully overlapped where

$$\widetilde{\boldsymbol{I}}_j = \bigcup_{Z_i \in \boldsymbol{Z}_j} \boldsymbol{I}_i. \tag{4.37}$$

We also consider s matrices $\{Y_j, 1 \le j \le s\}$ with $Y_j \in \mathbb{R}^{k_j \times \ell}$ such that $\mathsf{span}(Z_j) \subset \mathsf{span}(Y_j)$ for all j. Note that the index set for the rows of Y_j that correspond to the common rows of Z_j and $Z_{(\widetilde{\boldsymbol{I}}_j,:)}$ is

$$\boldsymbol{K}_j = \{k : \; j_k \in \boldsymbol{I}_j \cap \widetilde{\boldsymbol{I}}_j\}, \tag{4.38}$$

where $\boldsymbol{I}_j = \{j_1, j_2, \cdots, j_{k_j}\}$, see (4.7).

Theorem 4.4. Assume that $\{Z_j, 1 \le j \le s\}$ is a fully overlapped collection and $\{Y_j, 1 \le j \le s\}$ is a collection of matrices such that $\mathsf{span}(Z_j) \subset \mathsf{span}(Y_j)$ for all j. For each Z_j that is rank deficient, let $\widetilde{\boldsymbol{I}}_j$ be defined as in (4.37) and assume that $\mathsf{rank}(Y_{j(\boldsymbol{K}_j,:)}) = \mathsf{rank}(Y_j)$. Then we have $\mathsf{null}(\Psi) = \mathsf{null}(\Phi) = \mathsf{span}(Z)$.

Proof: First $\mathsf{null}(\Phi) = \mathsf{span}(Z)$ is proved by virtually the same proof as the one for Theorem 2.1 [30] by using Theorem 4.1 for the more general fully overlapped condition. We omit the details. $\mathsf{null}(\Psi) = \mathsf{span}(Z)$ is also proved similarly by an induction in s as follows.

The case $s = 2$ has already been proved in Theorem 4.2. So suppose the theorem is true for any collection with at most $s-1$ submatrices. We now prove the theorem is true for a fully overlapped collection $\boldsymbol{Z} = \{Z_j, 1 \le j \le s\}$ with $s \ge 3$. From Definition 4.2, we can partition $\boldsymbol{Z}$ into two nonempty disjoint subsets, say, $\widehat{\boldsymbol{Z}}_1 = \{Z_j, i = 1, \ldots, p\}$ and $\widehat{\boldsymbol{Z}}_2 = \{Z_j, j = p+1, \ldots, s\}$, such that both $\widehat{\boldsymbol{Z}}_1$ and $\widehat{\boldsymbol{Z}}_2$ are fully overlapped and $\{Z_{(\hat{\boldsymbol{I}}_1,:)}, Z_{(\hat{\boldsymbol{I}}_2,:)}\}$ are fully overlapped, where

$$\hat{\boldsymbol{I}}_i = \bigcup_{Z_j \in \widehat{\boldsymbol{Z}}_i} \boldsymbol{I}_j. \tag{4.39}$$

Let $\hat{Z}_i = Z_{(\hat{\boldsymbol{I}}_i,:)}$.

We first consider the case that $2 \le p \le s-2$. Then $\widehat{\boldsymbol{Z}}_1$ and $\widehat{\boldsymbol{Z}}_2$ each has less than $s-1$ submatrices and has more than 1 submatrix. Let $\hat{\Psi}_1$ and $\hat{\Psi}_2$ be the alignment matrices for $\{Y_j, j = 1, \dots, p\}$ and $\{Y_j, j = p+1, \dots, s\}$, respectively. Then we have

$$\mathsf{null}(\hat{\Psi}_1) = \mathsf{span}(\hat{Z}_1), \quad \mathsf{null}(\hat{\Psi}_2) = \mathsf{span}(\hat{Z}_2),$$

by the induction assumption. For the collection $\{\hat{Z}_1, \hat{Z}_2\}$, let $P^{\perp}_{\hat{Z}_i}$, $\hat{\Phi}_i$ and $\hat{\Phi}$ be the matrices defined according to (2.14). Then $\hat{\Phi}_i = \left[(I_N)_{(\hat{\boldsymbol{I}}_i,:)}\right]^T \times P^{\perp}_{\hat{Z}_i} \times (I_N)_{(\hat{\boldsymbol{I}}_i,:)}$ and $\hat{\Phi} = \sum_{i=1}^{2} \hat{\Phi}_i$. Since $\{\hat{Z}_1, \hat{Z}_2\}$ being fully overlapped, we obtain

$$\mathsf{null}(\hat{\Phi}) = \mathsf{null}(\hat{\Phi}_1) \bigcap \mathsf{null}(\hat{\Phi}_2) = \mathsf{span}(Z),$$

by Theorem 3.1. Now $\Psi = \sum_{j=1}^{s} \Psi_j$ is the alignment matrix for $\{Y_j, j = 1, \dots, s\}$ with Ψ_j defined according to (4.5). We notice that

$$\sum_{j=1}^{p} \Psi_j = \left[(I_N)_{(\hat{\boldsymbol{I}}_1,:)}\right]^T \times \hat{\Psi}_1 \times (I_N)_{(\hat{\boldsymbol{I}}_1,:)}$$

and

$$\sum_{j=p+1}^{s} \Psi_j = \left[(I_N)_{(\hat{\boldsymbol{I}}_2,:)}\right]^T \times \hat{\Psi}_2 \times (I_N)_{(\hat{\boldsymbol{I}}_2,:)}.$$

Noting that $\mathsf{null}(\hat{\Psi}_i) = \mathsf{span}(\hat{Z}_i)$ and $\mathsf{null}(P^{\perp}_{\hat{Z}_i}) = \mathsf{span}(\hat{Z}_i)$, we see

$$\mathsf{null}\left(\left[(I_N)_{(\hat{\boldsymbol{I}}_i,:)}\right]^T \times \hat{\Psi}_i \times (I_N)_{(\hat{\boldsymbol{I}}_i,:)}\right) = \mathsf{null}\left(\left[(I_N)_{(\hat{\boldsymbol{I}}_i,:)}\right]^T \times P^{\perp}_{\hat{Z}_i} \times (I_N)_{(\hat{\boldsymbol{I}}_i,:)}\right) = \mathsf{null}(\hat{\Phi}_i). \tag{4.40}$$

Then we have

$$\begin{aligned}
\mathsf{null}(\Psi) &= \mathsf{null}(\Psi_1 + \cdots + \Psi_p) \bigcap \mathsf{null}(\Psi_{p+1} + \cdots + \Psi_s) \\
&= \mathsf{null}\left(\left[(I_N)_{(\hat{\boldsymbol{I}}_1,:)}\right]^T \times \hat{\Psi}_1 \times (I_N)_{(\hat{\boldsymbol{I}}_1,:)}\right) \bigcap \mathsf{null}\left(\left[(I_N)_{(\hat{\boldsymbol{I}}_2,:)}\right]^T \times \hat{\Psi}_2 \times (I_N)_{(\hat{\boldsymbol{I}}_2,:)}\right) \\
&= \mathsf{null}(\hat{\Phi}_1) \bigcap \mathsf{null}(\hat{\Phi}_2) \\
&= \mathsf{span}(Z).
\end{aligned}$$

The theorem is proved in this case

We now consider the case that $p = 1$ or $p = s - 1$. Then one of $\{Z_j, j = 1, \ldots, p\}$ and $\{Z_j, j = p + 1, \ldots, s\}$ has only one submatrix. Without loss of generality, we assume $p = 1$, and let $\widetilde{Z}_2 = Z_{(\widetilde{\boldsymbol{I}}_2,:)}$ where

$$\widetilde{\boldsymbol{I}}_2 = \bigcup_{j=2}^{s} \boldsymbol{I}_j. \tag{4.41}$$

Then, $\{Z_1, \widetilde{Z}_2\}$ is fully overlapped. Since $\widetilde{Z}_2$ contains at least two submatrices that are fully overlapped and one of which is of full column rank, then $\widetilde{Z}_2$ has full column rank. Let $\widehat{\Psi}_2$ be the alignment matrices for $\{Y_j, j = 2, \ldots, s\}$. We have $\mathsf{null}(\widehat{\Psi}_2) = \mathsf{span}(\widetilde{Z}_2)$ by the induction assumption. Furthermore,

$$\sum_{j=2}^{s} \Psi_j = \left[(I_N)_{(\hat{\boldsymbol{I}}_2,:)}\right]^T \times \widehat{\Psi}_2 \times (I_N)_{(\hat{\boldsymbol{I}}_2,:)}.$$

Thus

$$\begin{aligned}
\mathsf{null}(\Psi) &= \mathsf{null}(\Psi_1) \bigcap \mathsf{null}(\Psi_2 + \cdots + \Psi_s) \\
&= \mathsf{null}(\Psi_1) \bigcap \mathsf{null}\left(\left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times \widehat{\Psi}_2 \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right) \\
&= \mathsf{null}(\Psi_1) \bigcap \mathsf{null}\left(\left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times P^{\perp}_{\widetilde{Z}_2} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right).
\end{aligned} \tag{4.42}$$

For the collection $\{Z_1, \widetilde{Z}_2\}$, $\{Y_1, \widetilde{Y}_2\}$ with $\widetilde{Y}_2 = \widetilde{Z}_2$ satisfies the assumption of Theorem 4.2. Let $\widetilde{\Psi}_2 = \left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times P^{\perp}_{\widetilde{Y}_2} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}$. Them, by Theorem 4.2, we have $\mathsf{null}(\Psi_1 + \widetilde{\Psi}_2) = \mathsf{span}(Z)$. Thus, it follows from (4.42) $\mathsf{null}(\Psi) = \mathsf{null}(\Psi_1) \bigcap \mathsf{null}(\widetilde{\Psi}_2) = \mathsf{span}(Z)$. The theorem is proved in this case too. □

Corollary 4.2. Under the assumption of Theorem 4.4, we have

$$\lambda^{+}_{\min}(\Psi)\, P^{\perp}_{Z} \preceq \Psi \preceq \lambda_{\max}(\Psi)\, P^{\perp}_{Z},$$

where $\lambda^{+}_{\min}(\Psi)$ is the smallest positive eigenvalue of Ψ, and $\lambda_{\max}(\Psi)$ is the largest eigenvalue of Ψ.

Under the assumption of Theorem 4.4, we now show how to construct a lower bound for $\lambda_{\min}^{+}(\Psi)$ recursively for the case $s > 2$. As in the proof of Theorem 3.4, $\boldsymbol{Z}$ can be partitioned into two nonempty disjoint subsets $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ such that

$$\widetilde{Z}_1 = Z_{(\widetilde{\boldsymbol{I}}_1,:)} \quad \text{and} \quad \widetilde{Z}_2 = Z_{(\widetilde{\boldsymbol{I}}_2,:)} \tag{4.43}$$

are fully overlapped, where $\boldsymbol{Z}_i = \{Z_{i_1}, Z_{i_2}, \ldots, Z_{i_{k_i}}\}$. Let $\boldsymbol{Y}_i = \{Y_{i_1}, Y_{i_2}, \ldots, Y_{i_{k_i}}\}$ and let $\widetilde{\Psi}_i$ be the alignment matrix for $\boldsymbol{Y}_i$ as defined in (4.5). The alignment matrix Ψ for $\{Y_j, 1 \le j \le s\}$ is

$$\Psi = \left[(I_N)_{(\widetilde{\boldsymbol{I}}_1,:)}\right]^T \times \widetilde{\Psi}_1 \times (I_N)_{(\widetilde{\boldsymbol{I}}_1,:)} + \left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times \widetilde{\Psi}_2 \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}. \tag{4.44}$$

Since $\boldsymbol{Z}$ is fully overlapped, we have either one of $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ contains only one submatrix which is rank deficient, say $\boldsymbol{Z}_1 = \{Z_j\}$ for some j, or both $\widetilde{Z}_1$ and $\widetilde{Z}_2$ have full column ranks. In the former case, using (4.44), we have

$$\lambda_{\min}^{+}(\widetilde{\Psi}_2)\left\{\left[(I_N)_{(\widetilde{\boldsymbol{I}}_1,:)}\right]^T \times P_{Y_j}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_1,:)} + \left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times P_{\widetilde{Z}_2}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right\} \preceq \Psi. \tag{4.45}$$

In the latter case, we have

$$\min_{i=1,2}\{\lambda_{\min}^{+}(\widetilde{\Psi}_i)\}\left\{\left[(I_N)_{(\widetilde{\boldsymbol{I}}_1,:)}\right]^T \times P_{\widetilde{Z}_1}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_1,:)} + \left[(I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right]^T \times P_{\widetilde{Z}_2}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}\right\} \preceq \Psi. \tag{4.46}$$

Define

$$\tau(\widetilde{Z}_1, \widetilde{Z}_2) \overset{\text{def}}{=} \begin{cases} \frac{1}{\sqrt{1+t^2}}, & \text{if } \boldsymbol{Z}_1 \text{ or } \boldsymbol{Z}_2 \text{ is } \{Z_j\} \text{ and } \mathsf{rank}(Z_j) < \ell; \\ \frac{1}{\sqrt{1+t_1^2}}\,\frac{1}{\sqrt{1+t_2^2}}, & \text{if } \mathsf{rank}(\widetilde{Z}_1) = \mathsf{rank}(\widetilde{Z}_2) = \ell; \end{cases} \tag{4.47}$$

where $t = \left\|Y_{j(\boldsymbol{J}_j,:)}\, Y_{j(\boldsymbol{K}_j,:)}^{\dagger}\right\|_2^{-1}$, $t_i = \left\|Z_{(\boldsymbol{L}_i,:)}\, Z_{(\widetilde{\boldsymbol{I}}_1 \bigcap \widetilde{\boldsymbol{I}}_2,:)}^{\dagger}\right\|_2^{-1}$, $\boldsymbol{J}_j$ is the complement set of $\boldsymbol{K}_j$ (see (4.38)) in $\{1, 2, \ldots, k_j\}$, and $\boldsymbol{L}_i$ is the complement set of $\widetilde{\boldsymbol{I}}_1 \bigcap \widetilde{\boldsymbol{I}}_2$ in $\widetilde{\boldsymbol{I}}_i$. Here we note that in the first case that $\boldsymbol{Z}_1$ or $\boldsymbol{Z}_2$ is $\{Z_j\}$, $\alpha(\boldsymbol{Z}_1, \boldsymbol{Z}_2)$ is defined implicitly from $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ through Y_j that corresponds to Z_j. Bounding (4.45) using Corollary

4.1 and bounding (4.46) using Theorem 3.6 of [16], we have

$$\min\{\lambda_{\min}^{+}(\widetilde{\Psi}_1), \lambda_{\min}^{+}(\widetilde{\Psi}_2)\} \left[1 - \tau(\widetilde{Z}_1, \widetilde{Z}_2)\right] P_Z^{\perp} \preceq \Psi,$$

where we note that $\lambda_{\min}^{+}(P_{Y_j}^{\perp}) = 1$. Then we can construct the lower bound for $\lambda_{\min}^{+}(\Psi)$ recursively by the method similar to Theorem 2.8. We calculate $\alpha(\boldsymbol{Z})$ that satisfies $\alpha(\boldsymbol{Z})P_Z^{\perp} \preceq \Psi$ by the following three equations recursively:

$$\alpha(\{Z_i\}) = 1, \tag{4.48}$$

$$\alpha(\{Z_i, Z_j\}) = 1 - \tau(Z_i, Z_j), \tag{4.49}$$

$$\alpha(\boldsymbol{Z}) = \left[1 - \tau(\widetilde{Z}_1, \widetilde{Z}_2)\right] \min\{\alpha(\boldsymbol{Z}_1), \alpha(\boldsymbol{Z}_2)\}. \tag{4.50}$$

The smallest positive eigenvalue $\lambda_{\min}^{+}(\Psi)$ is then no smaller than $\alpha(\boldsymbol{Z})$.

Theorem 4.5. Under the assumption of Theorem 4.4, let $\alpha(\boldsymbol{Z})$ be computed recursively by (4.48) – (4.50). Then $\alpha(\boldsymbol{Z})P_Z^{\perp} \preceq \Psi$, where alignment matrix Ψ is defined by (4.5).

We illustrate the recursive computation of the bound with an example.

Example 4.1. Consider a fully overlapped set $\{Z_1, Z_2, Z_3\}$. Suppose Z_3 and $\widetilde{Z}_2 \stackrel{\text{def}}{=} Z_{(\boldsymbol{I}_1 \bigcup \boldsymbol{I}_2,:)}$ are fully overlapped. Then write $\boldsymbol{Z}_1 = \{Z_3\}$ and $\boldsymbol{Z}_2 = \{Z_1, Z_2\}$. Let $\widetilde{\boldsymbol{I}}_2 \stackrel{\text{def}}{=} \boldsymbol{I}_1 \bigcup \boldsymbol{I}_2$. $\mathsf{rank}(Z_3) < \mathsf{rank}(Z_1) = \mathsf{rank}(Z_2)$. We have a lower bound for $\lambda_{\min}^{+}(\Psi)$ by Corollary 4.1.

$$\left[1 - \tau(Z_3, \widetilde{Z}_2)\right] P_Z^{\perp} \preceq (I_N)_{(\boldsymbol{I}_3,:)}^T \times P_{Y_3}^{\perp} \times (I_N)_{(\boldsymbol{I}_3,:)} + (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}^T \times P_{\widetilde{Z}_2}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)},$$

and

$$[1 - \tau(Z_1, Z_2)]\, (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)}^T \times P_{\widetilde{Z}_2}^{\perp} \times (I_N)_{(\widetilde{\boldsymbol{I}}_2,:)} \preceq \sum_{j=1}^{2} (I_N)_{(\boldsymbol{I}_j,:)}^T \times P_{Y_j}^{\perp} \times (I_N)_{(\boldsymbol{I}_j,:)}.$$

Put the two inequalities together to get $\alpha(\boldsymbol{Z})P_Z^{\perp} \preceq \Psi$ with

$$\alpha(\boldsymbol{Z}) = \left[1 - \tau(Z_3, \widetilde{Z}_2)\right] [1 - \tau(Z_1, Z_2)]\,. \tag{4.51}$$

4.3 Numerical Examples

In this section, we present two numerical examples to illustrate the lower bound of the smallest positive eigenvalue $\lambda^+_{\min}(\Psi)$. In particular, we study the dependence of the smallest positive eigenvalue on the number of overlapped rows and the "amount" of overlap t.

Consider the following matrix

$$Z = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & \cdots & N \\ a & b & c & d & e & 0 & \cdots & 0 \end{pmatrix}^T \in R^{N\times 3} \tag{4.52}$$

where a, b, c, d, e are distinct random numbers.

Example 4.2. Let $Z_1 = Z_{(1:N-5,:)}$ be the matrix consisting of the first $N-5$ rows of Z and $Z_2 = Z_{(j:N,:)}$ be the matrix consisting of the last $N-j+1$ rows of Z (for some j with $6 \le j \le N$). Then we have $\mathsf{rank}(Z_1) = 3$ and $\mathsf{rank}(Z_2) = 2$. Z_1 and Z_2 have no overlap if $j > N-5$. Z_1 and Z_2 are fully overlapped if $j < N-6$.

Let $Y_1 = Z_1$ and let Y_2 be the matrix such that its first two columns of Y_2 are the same as Z_2 (i.e. $Y_{2(:,1:2)} = Z_{2(:,1:2)}$) and its third column $Y_{2(:,3)}$ is random numbers uniformly distributed between 0 and 1. Noticing that $Z_{1(j:N-5,:)} = Z_{2(1:N-j-4)}$ is the common part of Z_1 and Z_2, from (4.36), our bound (4.49) on the smallest positive eigenvalue of Ψ as constructed from Y_1 and Y_2 is

$$\lambda^+_{\min}(\Psi) \ge 1 - \frac{1}{\sqrt{1+t^2}}$$

where $t = 1/\|Y_{2(N-j-3:N-j+1,:)}(Y_{2(1:N-j-4,:)})^\dagger\|_2$. We compare this bound with the computed fourth smallest eigenvalue of Ψ for varying values of j, which changes the number of rows in the overlap or the amount of overlap t. For all of our test cases, the first three computed eigenvalues are less than 10^{-13}, confirming that the null space of Ψ is of dimension 3.

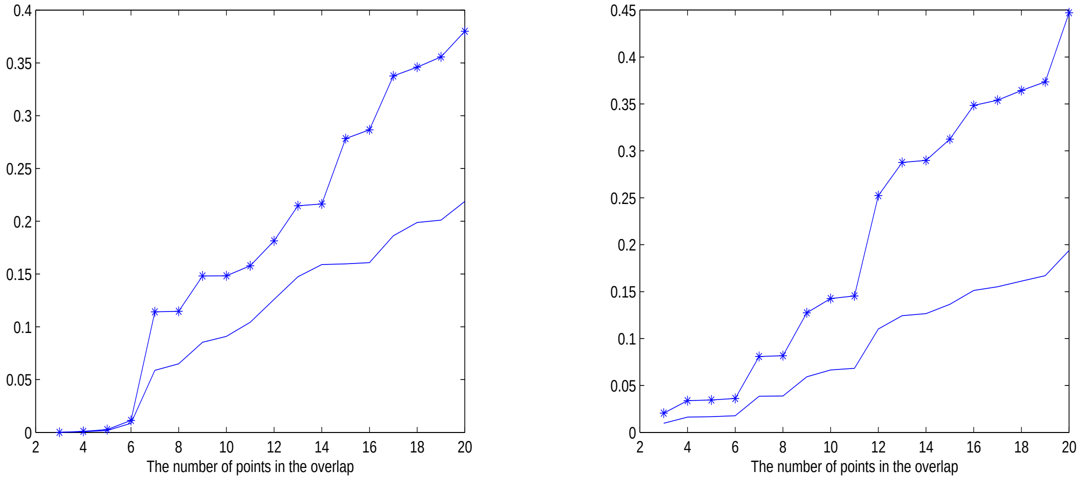


Figure 4.1: $\lambda_{\min}^{+}(\Psi)$ ($*$-line) and its bound (solid line) *vs.* number of rows overlapped

In Figure 4.1, we present the results for two tests (two sets of different Z and Y_2) with $N = 30$, one on the left and the other on the right. Here, we plot $\lambda_{\min}^{+}(\Psi)$ and its lower bound $1 - \frac{1}{\sqrt{1+t^2}}$ against the number of overlapped rows in the overlap. To investigate the dependence of $\lambda_{\min}^{+}(\Psi)$ on the amount of overlap t, we also plot $\lambda_{\min}^{+}(\Psi)$ against t for the same problems in Figure 4.2.

We observe that our lower bound on $\lambda_{\min}^{+}(\Psi)$ is indeed correct. The figures also clearly show that the bound as well as the smallest positive eigenvalue increases as the overlap increases. Although it is a bit pessimistic, it tracks the changing behavior of the eigenvalue very well, namely the point at which the eigenvalue increases significantly (Figure 4.1). Furthermore, Figure 4.2 also demonstrates near quadratic dependence on the amount of overlap t.

Example 4.3. We consider an example with $s = 3$. Let $Z_1 = Z_{(1:5,:)}$ be the matrix consisting of the first 5 rows of Z, $Z_2 = Z_{(3:N-5,:)}$ be the matrix consisting of the 3rd to the $(N-5)$th rows and $Z_3 = Z_{(j:N,:)}$ be the matrix consisting of the last $N - j + 1$ rows of Z (for some j with $6 \le j \le N$), where Z is given in (4.52). Then we have $\{Z_j, 1 \le j \le 3\}$ is a fully overlapped collection if $j < N - 6$. We consider the case

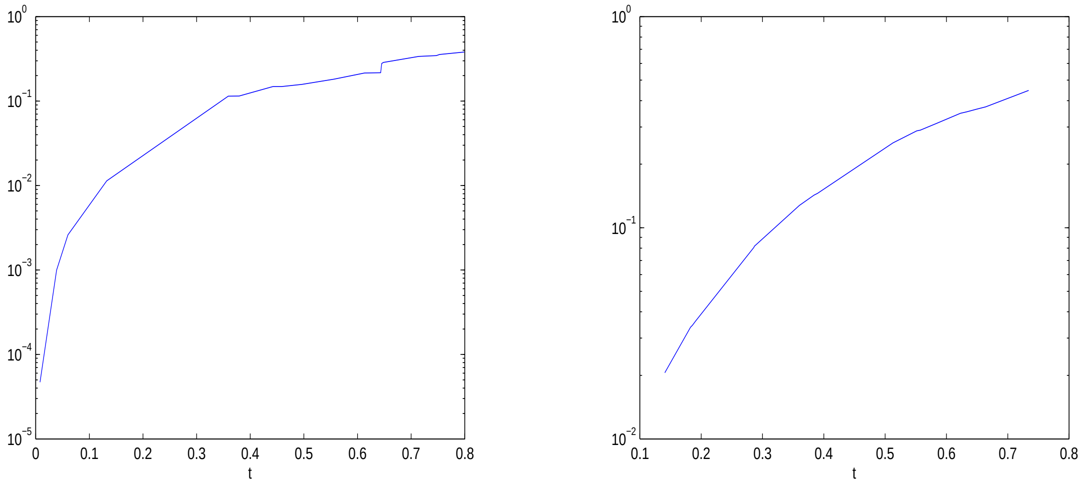


Figure 4.2: $\lambda^{+}_{\min}(\Psi)$ (solid line) *vs.* amount of overlap t

that $Y_1 = Z_1$, $Y_2 = Z_2$ and Y_3 is the matrix such that its first two columns are the same as Z_3 (i.e. $Y_{3(:,1:2)} = Z_{3(:,1:2)}$) and its third column $Y_{3(:,3)}$ is random numbers uniformly distributed between 0 and 1, as in Example 4.1.

To determine a lower bound, we use $\boldsymbol{Z}_1 = \{Z_3\}$ and $\boldsymbol{Z}_2 = \{Z_1, Z_2\}$. Z_3 and $\widetilde{Z}_2 \overset{\text{def}}{=} Z_{(\boldsymbol{I}_1 \bigcup \boldsymbol{I}_2,:)}$ are fully overlapped. We can find the lower bound for the smallest positive eigenvalue of Ψ by (4.51). Noticing that $Z_{1(3:5,:)} = Z_{2(1:3,:)}$ is the common part of Z_1 and Z_2, from (4.47), our bound (4.49) as constructed from Y_1 and Y_2 is

$$1 - \tau(Z_1, Z_2) = 1 - \frac{1}{\sqrt{1+t_1^2}} \frac{1}{\sqrt{1+t_2^2}},$$

where $t_1 = 1/\|Y_{1(1:2,:)}(Y_{1(3:5,:)})^{\dagger}\|_2$ and $t_2 = 1/\|Y_{2(4:N-7,:)}(Y_{2(1:3,:)})^{\dagger}\|_2$. From $Z_{2(j-2:N-7,:)} = Z_{3(1:N-j-4)}$ being the common part of $\widetilde{Z}_2$ and Z_3 and (4.51), our bound (4.50) on the smallest positive eigenvalue of Ψ as constructed from $\{Y_j, 1 \leq j \leq 3\}$ is

$$\lambda^{+}_{\min}(\Psi) \geq \left(1 - \frac{1}{\sqrt{1+t^2}}\right)\left(1 - \frac{1}{\sqrt{1+t_1^2}} \frac{1}{\sqrt{1+t_2^2}}\right), \tag{4.53}$$

where $t = 1/\|Y_{3(N-j-3:N-j+1,:)}(Y_{3(1:N-j-4,:)})^{\dagger}\|_2$. We compare this bound with the computed fourth smallest eigenvalue of Ψ for varying values of j, which changes the number of rows in the overlap or the amount of overlap t. For all of our test cases, the

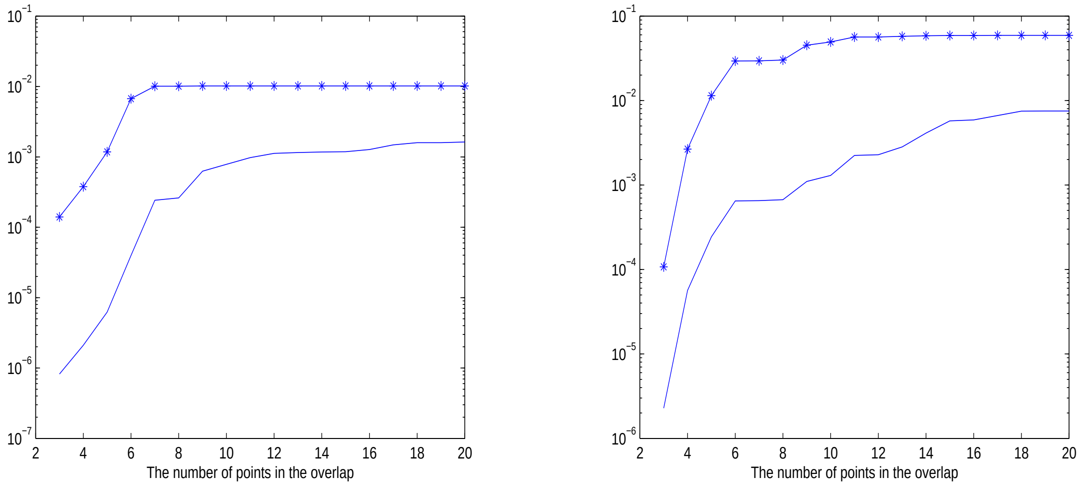


Figure 4.3: $\lambda^+_{\min}(\Psi)$ ($*$-line) and its bound (solid line) *vs.* number of rows overlapped

first three computed eigenvalues are less than 10^{-13}, confirming that the null space of Ψ is of dimension 3.

In Figure 4.3, we present the results for two tests (two sets of different Z and Y_3) with $N = 30$, one on the left and the other on the right. Here, we plot $\lambda^+_{\min}(\Psi)$ and its lower bound (4.53) against the number of overlapped rows in the overlap.

We observe that our lower bound on $\lambda^+_{\min}(\Psi)$ is indeed correct. The figures also clearly show that the bound as well as the smallest positive eigenvalue increases as the overlap increases. The bound is, however, much more pessimistic now. Nevertheless, it tracks the changing behavior of the eigenvalue very well.

Chapter 5 Analysis of Discrete Hessian Eigenmaps Algorithm

In this chapter, we still consider a parameterized manifold of dimension d defined by a mapping ψ: $\mathcal{C} \subset \mathbb{R}^d \to \mathbb{R}^m$, where the embedding space $\mathbb{R}^m$ obeys $d < m$ and $\mathcal{C}$ is open in $\mathbb{R}^d$. The image $\mathcal{M} = \psi(\mathcal{C})$ is the manifold. Here $\mathbb{R}^m$ represents the high-dimensional data space and $\mathbb{R}^d$ represents the low-dimensional parameter space. Suppose we have a set of data points $x_1, \ldots, x_N$ sampled from the manifold $\mathcal{M}$ as (1.1), i.e.,

$$x_i = \psi(\tau_i), \quad i = 1, \ldots, N,$$

for $\tau_i \in \mathcal{C}$. We recover the parameter points $\tau_i's$ from x_i's. We concentrate on the noise-free data (1.1) and follow Donoho and Grimes [8] to assume ψ is a local isometry.

We present a discrete Hessian Eigenmaps method that is based on the numerical procedure developed in [8]; see the Hessian LLE algorithm in Section 2.2.5. By defining a discrete Hessian operator and a generalized $\mathcal{H}$-functional that we call Hessian alignment matrix, we show that the null space of the Hessian alignment matrix recovers the locally isometric coordinates, provided local neighborhoods are sufficiently "overlapped".

5.1 Discrete Hessian Eigenmaps Method

In this section, we present a discrete version of the Hessian Eigenmaps method. Specifically, we introduce discrete Hessian operator and a generalization of the $\mathcal{H}$-functional and prove a generalization of Theorem 2.9. The discrete Hessian Eigenmaps method is essentially the same as the numerical procedure of Hessian LLE described in Chapter 2, but is formulated as a direct generalization of the original

Hessian Eigenmaps method. By establishing a discrete version of Theorem 2.9 with analysis, we directly provide a theoretical basis of the discrete procedure.

We are interested in reconstructing the coordinate set $\{\tau_1, \tau_2, \cdots, \tau_N\}$ for a given data set $\boldsymbol{X} = \{x_1, \ldots, x_N\}$. We partition $\boldsymbol{X}$ into subsets $\{\boldsymbol{X}_i, i = 1, \ldots, s\}$ with $\boldsymbol{X}_i = \{x_{i_1}, \ldots, x_{i_{k_i}}\}$ $(i_1 < i_2 < \cdots < i_{k_i})$ consisting of points is a small neighborhood so that a coordinate system on the local tangent space can be approximately obtained. For theoretical purpose, we assume that $\boldsymbol{X}_i$ is a subset such that it has an isometric coordinates $\boldsymbol{S}_i = \{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\} \subset \mathbb{R}^d$, i.e. $\|\tau_{i_p} - \tau_{i_q}\|_2 = d_{\mathcal{M}}(x_{i_p}, x_{i_q})$ (for any $1 \leq p, q \leq k_i$) where $d_{\mathcal{M}}(\,\cdot\,,\,\cdot\,)$ is the geodesic distance along $\mathcal{M}$. For each $\boldsymbol{S}_i$, we can compute a local isometric coordinate $\boldsymbol{\Theta}_i = \{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\}$ from the pairwise geodesic distance such that $\|\theta_p^{(i)} - \theta_q^{(i)}\|_2 = d_{\mathcal{M}}(x_{i_p}, x_{i_q})$ (for any $1 \leq p, q \leq k_i$) (In the context of Hessian Eigenmaps, the local tangent coordinate is an approximation for the local coordinate defined here). We discuss how to reconstruct τ_i's from the local coordinates based on local Hessian operators.

5.1.1 Hessian Operator

Let $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\} \subset \mathbb{R}^d$ be an ordered set and set

$$T_0 = [\tau_1, \ldots, \tau_k].$$

For our purpose, $\boldsymbol{S}_0$ is a coordinate set and we define its dimension as in Definition 3.3.

Throughout this chapter, we consider ordered subsets of dimension d.

From $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\}$, we define a Hessian operator as follows.

Definition 5.1. Let $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\} \subset \mathbb{R}^d$ be an ordered subset of dimension d and let $T_0 = [\tau_1, \tau_2, \cdots, \tau_k]$. Let

$$Z_0 = Y_0 - [e, T_0^T][e, T_0^T]^\dagger Y_0,$$

where

$$Y_0 = \begin{pmatrix} \tau_1^T \bigodot \tau_1^T \\ \tau_2^T \bigodot \tau_2^T \\ \vdots \\ \tau_k^T \bigodot \tau_k^T \end{pmatrix} \in \mathbb{R}^{k \times q} \text{ with } q = \frac{d(d+1)}{2}. \tag{5.1}$$

and the operation $\bigodot$ is defined in (1.2). We say

$$H_0 = Z_0^\dagger \tag{5.2}$$

is the Hessian operator as defined by $\boldsymbol{S}_0$.

Remark 5.1. The Hessian operator H_0 depends on the order in $\boldsymbol{S}_0$. If we permute the vectors in $\boldsymbol{S}_0$ to get another ordered set $\hat{\boldsymbol{S}}_0$ with a corresponding matrix $\hat{T}_0 = T_0 P$ for some permutation matrix P, it is easy to check that $\hat{H}_0 = H_0 P$ where $\hat{H}_0$ is the Hessian operator defined by $\hat{\boldsymbol{S}}_0$.

Lemma 5.1. Let $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\} \subset \mathbb{R}^d$ be an ordered subset of dimension d and let $T_0 = [\tau_1, \tau_2, \cdots, \tau_k]$. Let H_0 be the Hessian operator for $\boldsymbol{S}_0$. We have $\mathsf{span}([e, T_0^T]) \subset \mathsf{null}(H_0)$.

Proof: From Definition 5.1, we have $[e, T_0^T]^T Z_0 = 0$ where $Z_0 = H_0^\dagger = Y_0 - [e, T_0^T][e, T_0^T]^\dagger Y_0$ and Y_0 is defined in (5.1). Then $Z_0^T[e, T_i^T] = 0$. Hence $Z_0^\dagger[e, T_i^T] = 0$.
□

We note that the columns of $[e, T_0^T]$ consists of the vectors $[f(\tau_i)]_{i=1}^k$ with $f : \mathbb{R}^d \to \mathbb{R}$ being the constant or the d linear functions. Y_0 consists of $[f(\tau_i)]_{i=1}^k$ with $f : \mathbb{R}^d \to \mathbb{R}$ being the quadratic functions. We next present a result that justifies the definition of Hessian operator .

Consider a function $h(\tau) : \mathbb{R}^d \mapsto \mathbb{R}$ and we are interested in an approximation of the Hessian of h at some point τ_0 using the values of $h(\tau_i)$. Performing the Taylor expansion for $h(\tau)$ at τ_0, we have

$$h(\tau) = h(\tau_0) + (\tau - \tau_0)^T \nabla h(\tau_0) + \frac{1}{2}(\tau - \tau_0)^T H_h(\tau_0)(\tau - \tau_0) + R(\tau - \tau_0),$$

where $R(\tau-\tau_0)$ is the remainder term and $H_h(\tau_0)$ is the Hessian matrix of the function $h(\tau)$ at $\tau = \tau_0$. Let

$$\mathfrak{h}_h(\tau) = [a_1(\tau), \ldots, a_p(\tau)]^T \in \mathbb{R}^q, \tag{5.3}$$

where

$$a_{\frac{k(k-1)}{2}+\ell}(\tau) = \begin{cases} \frac{1}{2}\frac{\partial^2 h}{\partial t_k \partial t_\ell}(\tau) & k = \ell; \\ \frac{\partial^2 h}{\partial t_k \partial t_\ell}(\tau) & k > \ell. \end{cases}$$

and $q = d(d+1)/2$. Then $\mathfrak{h}_h(\tau)$ is a vector form of the Hessian matrix containing the entries of the lower triangular part (including diagonal) of the Hessian. Now, considering the Taylor expansion of $h(\tau_i)$ for $i = 1, 2, \cdots, k$ and combining them together, we can write

$$\begin{pmatrix} h(\tau_1) \\ h(\tau_2) \\ \vdots \\ h(\tau_k) \end{pmatrix} = \begin{pmatrix} 1 & (\tau_1-\tau_0)^T & (\tau_1-\tau_0)^T \bigodot (\tau_1-\tau_0)^T \\ 1 & (\tau_2-\tau_0)^T & (\tau_2-\tau_0)^T \bigodot (\tau_2-\tau_0)^T \\ \vdots & \vdots & \vdots \\ 1 & (\tau_k-\tau_0)^T & (\tau_k-\tau_0)^T \bigodot (\tau_k-\tau_0)^T \end{pmatrix} \begin{pmatrix} h(\tau_0) \\ \nabla h(\tau_0) \\ \mathfrak{h}_h(\tau_0) \end{pmatrix} + \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_k \end{pmatrix}. \tag{5.4}$$

where $r_i = R(\tau_i - \tau_0)$. If $h \in C^3(\mathbb{R}^d)$, $|r_i| \leq C\|\tau_i - \tau_0\|^3$ for some constant $C > 0$.

Proposition 5.1.1. Let $h : \mathbb{R}^d \mapsto R$ and H_0 be the Hessian operator as defined by $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\}$. Let $\mathfrak{h}_h(\tau_0)$ be the column form of the Hessian matrix of function $h(\tau)$ at τ_0 defined according to (5.3). If H_0 has full row rank, we have

$$H_0 \begin{pmatrix} h(\tau_1) \\ h(\tau_2) \\ \vdots \\ h(\tau_k) \end{pmatrix} = \mathfrak{h}_h(\tau_0) + H_0 \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_k \end{pmatrix}, \tag{5.5}$$

where $r_i = R(\tau_i - \tau_0)$.

Proof: Set

$$\mathbf{h} = \begin{pmatrix} h(\tau_1) \\ h(\tau_2) \\ \vdots \\ h(\tau_k) \end{pmatrix} \in \mathbb{R}^k, \quad \text{and} \quad \mathbf{r} = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_k \end{pmatrix} \in \mathbb{R}^k.$$

It follows from (5.4) that

$$\mathbf{h} = [e, T_0^T - e\tau_0^T] \begin{pmatrix} h(\tau_0) \\ \nabla h(\tau_0) \end{pmatrix} + (Y_0 + [e, T_0^T]R_0)\mathfrak{h}_h(\tau_0) + \mathbf{r},$$

where $R_0 \in \mathbb{R}^{(d+1)\times p}$. Equivalently,

$$\begin{aligned} \mathbf{h} &= [e, T_0^T - e\tau_0^T] \begin{pmatrix} h(\tau_0) \\ \nabla h(\tau_0) \end{pmatrix} + (I - [e, T_0^T][e, T_0^T]^\dagger)(Y_0 + [e, T_0^T]R_0)\mathfrak{h}_h(\tau_0) \\ &\quad + [e, T_0^T][e, T_0^T]^\dagger(Y_0 + [e, T_0^T]R_0)\mathfrak{h}_h(\tau_0) + \mathbf{r}. \end{aligned} \tag{5.6}$$

Multiplying (5.6) by H_0, we obtain

$$\begin{aligned} H_0\mathbf{h} &= H_0[e, T_0^T - e\tau_0^T] \begin{pmatrix} h(\tau_0) \\ \nabla h(\tau_0) \end{pmatrix} + H_0(I - [e, T_0^T][e, T_0^T]^\dagger)(Y_0 + [e, T_0^T]R_0)\mathfrak{h}_h(\tau_0) \\ &\quad + H_0[e, T_0^T][e, T_0^T]^\dagger(Y_0 + [e, T_0^T]R_0)\mathfrak{h}_h(\tau_0) + H_0\mathbf{r}. \end{aligned} \tag{5.7}$$

Noticing $H_0[e, T_0^T] = 0$ and $(I - [e, T_0^T][e, T_0^T]^\dagger)Y_0 = H_0^\dagger$, we have

$$H_0\mathbf{h} = H_0 H_0^\dagger \mathfrak{h}_h(\tau_0) + H_0\mathbf{r}. \tag{5.8}$$

Since H_0 has full row rank $H_0H_0^\dagger = I$. (5.5) is proved. □

For the rest of this paper, we are only interested in the null space (or the range space) of the discrete Hessian operator. For this purpose then, instead of computing the discrete Hessian operator, we find its range space, which can be easily constructed as follows. Let

$$Y^{(0)} = [e, T_0^T, Y_0] \in \mathbb{R}^{k\times q}, \tag{5.9}$$

where Y_0 is given in (5.1) and perform Gram-Schmidt orthonormalization for $Y^{(0)}$, yielding a matrix $\hat{Y}_0 \in \mathbb{R}^{k\times q}$ such that

$$Y^{(0)} = \hat{Y}_0 R_0 \bar{P}_0^T, \tag{5.10}$$

where $\hat{Y}_0$ is a matrix with the first $\mathsf{rank}(Y^{(0)})$ columns being orthonormal, $R_0 \in \mathbb{R}^{q\times q}$ is an upper triangle matrix and $\bar{P}_0$ is a permutation matrix. Since the first $d+1$ columns of $Y^{(0)}$ are linearly independent by Lemma 3.1, we have

$$\bar{P}_0 = \begin{pmatrix} I_{d+1} & 0 \\ 0 & \widetilde{P}_0 \end{pmatrix}, \tag{5.11}$$

for some permutation matrix $\widetilde{P}_0$. Write

$$\hat{Y}_0 = \begin{array}{ccc} {\scriptstyle d+1} & {\scriptstyle p_0} & {\scriptstyle q_0} \\ \Big(Y_{0,1}, & Q_0, & 0 \Big) \end{array}, \tag{5.12}$$

where

$$p_0 = \big[\mathsf{rank}(Y^{(0)}) - (d+1)\big] \quad \text{and} \quad q_0 = q - \mathsf{rank}(Y^{(0)}). \tag{5.13}$$

The following theorem shows that the columns of Q_0 form an orthonormal basis for $\mathsf{span}(H_0^T)$.

Lemma 5.2. Given an ordered subset $\boldsymbol{S}_0$, let Q_0 be defined according to (5.12). We have $\mathsf{span}(Q_0) = \mathsf{span}(H_0^T)$, i.e. the columns of Q_0 form an orthonormal basis for H_0^T. where $H_0 \in \mathbb{R}^{p\times k}$ is the Hessian operator for $\boldsymbol{S}_0$.

Proof: Let $Y_{0,1}$ be defined according to (5.12). We notice that

$$\mathsf{span}([Y_{0,1}, Q_0]) = \mathsf{span}([e, T_0^T, Y_0]) = \mathsf{span}([e, T_0^T, Z_0]) \tag{5.14}$$

Noticing $Y_{0,1}$ being orthonormal basis for $\mathsf{span}([e, T_0^T])$, we have $\mathsf{span}(Q_0) = \mathsf{span}(Z_0) = \mathsf{span}(H_0^\dagger) = \mathsf{span}(H_0^T)$, we have $\mathsf{span}(Q_0) = \mathsf{span}(H_0^T)$. □

The Hessian operators is defined from a coordinate set but, as in the continuous case, its column space is invariant under a linear transformation on the coordinate set (or change of basis), as is shown in the next lemma.

Lemma 5.3. Let G_0 and H_0 be the Hessian operators for the ordered subsets $\boldsymbol{\Theta}_0 = \{\theta_1, \ldots, \theta_k\}$ and $\boldsymbol{S}_0 = \{\tau_1, \ldots, \tau_k\}$, respectively. Set $\Theta_0 = [\theta_1, \ldots, \theta_k]$ and $T_0 = [\tau_1, \ldots, \tau_k]$. If $\Theta_0 = V_1 T_0 + ce^T$, where $c \in \mathbb{R}^d$ and $V_1 \in \mathbb{R}^{d\times d}$ is a nonsingular matrix, we have $G_0 = V_2 H_0$ for some nonsingular matrix V_2.

Proof: We construct the matrices $X^{(0)}$ and $Y^{(0)}$ for subset $\boldsymbol{\Theta}_0$ and $\boldsymbol{S}_0$ according to (5.9), respectively, i.e.

$$X^{(0)} = [e, \Theta_0^T, X_0] \quad \text{and} \quad Y^{(0)} = [e, T_0^T, Y_0],$$

where

$$X_0 = \begin{pmatrix} {\theta_1}^T \bigodot {\theta_1}^T \\ {\theta_2}^T \bigodot {\theta_2}^T \\ \vdots \\ {\theta_{k_0}}^T \bigodot {\theta_k}^T \end{pmatrix} \text{ and } Y_0 = \begin{pmatrix} {\tau_1}^T \bigodot {\tau_1}^T \\ {\tau_2}^T \bigodot {\tau_2}^T \\ \vdots \\ {\tau_k}^T \bigodot {\tau_k}^T \end{pmatrix}.$$

We notice that

$$\mathsf{span}(X_0) = \mathsf{span}(\widetilde{X}_0) \quad \text{and} \quad \mathsf{span}(Y_0) = \mathsf{span}(\widetilde{Y}_0), \tag{5.15}$$

where

$$\widetilde{X}_0 = \begin{pmatrix} {\theta_1}^T \bigotimes {\theta_1}^T \\ {\theta_2}^T \bigotimes {\theta_2}^T \\ \vdots \\ {\theta_{k_0}}^T \bigotimes {\theta_k}^T \end{pmatrix} \text{ and } \widetilde{Y}_0 = \begin{pmatrix} {\tau_1}^T \bigotimes {\tau_1}^T \\ {\tau_2}^T \bigotimes {\tau_2}^T \\ \vdots \\ {\tau_k}^T \bigotimes {\tau_k}^T \end{pmatrix},$$

since $\widetilde{X}_0$ has the same column vectors as X_0 and $\widetilde{Y}_0$ has the same column vectors as Y_0 excluding the repeated ones. We have $\theta_i = V_1\tau_i + c$ for $i = 1, \dots, k$. Then

$$\begin{aligned} {\theta_i}^T \bigotimes {\theta_i}^T &= (V_1\tau_i + c)^T \bigotimes (V_1\tau_i + c)^T \\ &= (\tau_i^T V_1^T + c^T) \bigotimes (\tau_i^T V_1^T + c^T) \\ &= (\tau_i^T \bigotimes \tau_i^T)(V_1^T \bigotimes V_1^T) + c^T \bigotimes (\tau_i^T V_1^T) + (\tau_i^T V_1^T) \bigotimes c^T + c^T \bigotimes c^T, \end{aligned}$$

for $i = 1, \dots, k$, see [6, Lemma 6.3]. It follows

$$\widetilde{X}_0 = \widetilde{Y}_0(V_1^T \bigotimes V_1^T) + [e, T_0^T]R \tag{5.16}$$

for some matrix $R \in \mathbb{R}^{(d+1)\times d^2}$. Since V_1 is a nonsingular matrix, we have $V_1^T \bigotimes V_1^T$ is a nonsingular matrix by [6, Lemma 6.3]. It also follows from $\Theta_0 = V_1 T_0 + ce^T$ that

$$I - [e, \Theta_0^T][e, \Theta_0^T]^\dagger = I - [e, T_0^T][e, T_0^T]^\dagger. \tag{5.17}$$

Combining (5.16) and (5.17), we have

$$\mathsf{span}\left((I - [e, \Theta_0^T][e, \Theta_0^T]^\dagger)\widetilde{X}_0\right) = \mathsf{span}\left((I - [e, T_0^T][e, T_0^T]^\dagger)\widetilde{Y}_0\right). \tag{5.18}$$

$\mathsf{span}\left((I - [e, \Theta_0^T][e, \Theta_0^T]^\dagger)X_0\right) = \mathsf{span}\left((I - [e, T_0^T][e, T_0^T]^\dagger)Y_0\right)$ from (5.15) and (5.18). Then $\mathsf{span}(G_0^\dagger) = \mathsf{span}(H_0^\dagger)$. It follows $\mathsf{span}(G_0^T) = \mathsf{span}(H_0^T)$. $G_0 = V_2 H_0$ for some nonsingular matrix V_2. □

5.1.2 Hessian Alignment Matrix

We now generalize the definition of the quadratic form of the $\mathcal{H}$-functional to the discrete case and present a generalization of Theorem 2.9. In the discrete setting, the construction of the quadratic form or the corresponding symmetric matrix is more closely related to that of the alignment matrix in the LTSA method [34, 30]. Indeed, they have some similar spectral properties as well. Hence we call it the Hessian alignment matrix.

Let $\{\boldsymbol{S}_i, 1 \le i \le s\}$ be a collection of ordered subsets of a given ordered set $\boldsymbol{S} = \{\tau_1, \ldots, \tau_N\}$. Write

$$\boldsymbol{S}_i = \{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\}, \quad i_1 < i_2 < \ldots < i_{k_i}, \tag{5.19}$$

and set

$$T = [\tau_1, \cdots, \tau_N] \in \mathbb{R}^{d\times N}, \qquad \text{and} \qquad T_i = [\tau_{i_1}, \cdots, \tau_{i_{k_i}}].$$

We say T_i is a *section* of T. We have $TE_i = T_i$. E_i is called the selection matrix for $\boldsymbol{S}_i$. Let

$$E_i = [e_{i_1}, \ldots, e_{i_{k_i}}] \in \mathbb{R}^{N\times k_i}, \tag{5.20}$$

where $e_i \in \mathbb{R}^N$ is the i-th column of I_N, i.e. the $N \times N$ identity matrix.

In the context of manifold learning, each $\boldsymbol{S}_i$ is a coordinate set for points in a small neighborhood, from which a Hessian operator can be defined. Assembling them together, the following is a generalization of the definition of the $\mathcal{H}$-functional.

Definition 5.2. Given an ordered set $\boldsymbol{S} = \{\tau_1, \ldots, \tau_N\}$ and a collection of ordered subsets $\{\boldsymbol{S}_i, 1 \le i \le s\}$, let

$$\Phi = \sum_{i=1}^{s} E_i H_i^{\dagger} H_i E_i^T, \tag{5.21}$$

where H_i $(1 \leq i \leq s)$ is the Hessian operator for the subset $\boldsymbol{S}_i$ $(1 \leq i \leq s)$ and E_i is the selection matrix for $\boldsymbol{S}_i$ (see (5.20)). We call Φ the Hessian alignment matrix for $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$.

Remark 5.2. We note that the definition of H_i is dependent on the order of the vectors in $\boldsymbol{S}_i$ but $E_i H_i^\dagger H_i E_i^T$ is not. Hence Φ is independent of the order of the vectors in $\boldsymbol{S}_i$. However, if we change the order of the column vectors of $\boldsymbol{S}$ to have another ordered set $\hat{\boldsymbol{S}}$ with corresponding matrix $\hat{T} = TP$ for a permutation matrix P, the Hessian alignment matrix $\hat{\Phi}$ defined on $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ satisfies $\hat{\Phi} = P^T \Phi P$.

By setting

$$P_i = H_i^\dagger H_i, \tag{5.22}$$

we have $\Phi = \sum_{i=1}^s E_i P_i E_i^T$.

Lemma 5.4. Given an ordered set $\boldsymbol{S} = \{\tau_1, \dots, \tau_N\}$ and a collection of subsets $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$ with $\boldsymbol{S} = \bigcup_{i=1}^s \boldsymbol{S}_i$, and $\boldsymbol{S}_i = \{\tau_{i_1}, \dots, \tau_{i_{k_i}}\}$ $(i_1 < i_2 < \cdots < i_{k_i})$, let $T_i = [\tau_{i_1}, \dots, \tau_{i_{k_i}}]$, $T = [\tau_1, \dots, \tau_N]$ and $T_i = TE_i$ with E_i being the selection matrix. Let H_i be the Hessian operator for $\boldsymbol{S}_i$ and Φ be the Hessian alignment matrix for $\{\boldsymbol{S}_i, 1 \leq i \leq s\}$. Then, we have

$$\mathsf{null}(H_i E_i^T) \supset \{x | E_i^T x \in \mathsf{span}([e, T_i^T])\} \tag{5.23}$$

and

$$\mathsf{null}(\Phi) = \bigcap_{i=1}^s \mathsf{null}(H_i E_i^T). \tag{5.24}$$

Moreover, we have $\mathsf{span}([e, T^T]) \subset \mathsf{null}(\Phi)$.

Proof: Let Q_i be the matrix consisting of the orthonormal basis of $\mathsf{span}(H_i^T)$. We have proved that $\Phi_i = Q_i Q_i^T$. Noticing that $\mathsf{span}(Q_i^T) = \mathsf{span}(H_i)$, the rest of the proof of this lemma is the same as the proof of Lemma 2.1 of [30]. We omit it here. □

The main result of this section is to determine under what conditions that $\mathsf{span}([e, T^T]) = \mathsf{null}(\Phi)$. For that, we need to introduce some definitions.

Definition 5.3. Let $\boldsymbol{S}_1$ and $\boldsymbol{S}_2$ be two ordered subsets of $\mathbb{R}^d$. Let T_2 be the matrix with its columns equal to the vectors in $\boldsymbol{S}_2$ in its given order and let H_2 be the Hessian operator for $\boldsymbol{S}_2$. Let $H_{2\backslash 1}$ be the submatrix of H_2 consisting of the columns of H_2 that correspond[1] to the vectors in $\boldsymbol{S}_2 \setminus \boldsymbol{S}_1$ (i.e. $H_{2\backslash 1} = H_2 E_{2\backslash 1}$ where $E_{2\backslash 1}$ is some selection matrix defined in (5.20) such that $T_2 E_{2\backslash 1}$ the submatrix of T_2 constructed from the column vectors that are in $\boldsymbol{S}_2 \setminus \boldsymbol{S}_1$), We say $\boldsymbol{S}_1$ is rigidly overlapped to $\boldsymbol{S}_2$ if $H_{2\backslash 1}$ has full column rank.

Based on the above relation of two subsets, we can associate a directed graph on $\{\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots, \boldsymbol{S}_s\}$.

Definition 5.4. We associate a collection of subsets $\{\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots, \boldsymbol{S}_s\}$ with a directed graph $\mathcal{G}$ constructed as follows: its s vertices represent the s subsets, where there is an edge from vertex i to vertex j if the subset $\boldsymbol{S}_i$ is rigidly overlapped to the subset $\boldsymbol{S}_j$.

Definition 5.5. We say the collection of subsets $\{\boldsymbol{S}_i, 1 \le i \le s\}$ is rooted connected and S_j is the root if its associated directed graph $\mathcal{G}$ is a rooted connected graph with root j, i.e. there is a route from the vertex j to any vertex i (for $1 \le i \le s$).

Definition 5.6. Given an ordered set $\boldsymbol{S} = \{\tau_1, \ldots, \tau_N\}$ and a collection of subsets $\{\boldsymbol{S}_i, 1 \le i \le s\}$ with $\boldsymbol{S} = \bigcup_{i=1}^{s} \boldsymbol{S}_i$, let Φ be the Hessian alignment matrix for $\{\boldsymbol{S}_1, \ldots, \boldsymbol{S}_\ell\}$. We say the collection $\{\boldsymbol{S}_1, \ldots, \boldsymbol{S}_\ell\}$ is a full spanning collection, if $\mathsf{rank}(\Phi) = N - (d+1)$.

[1] Note that each column of the discrete Hessian H_2 corresponds to a column of T_2 or a vector in $\boldsymbol{S}_2$.

Lemma 5.5. Let $\boldsymbol{S} = \{\tau_1, \ldots, \tau_N\}$ be an ordered set and let $\{\boldsymbol{S}_i, 1 \le i \le \ell\}$ be a collection of subsets with $\boldsymbol{S} = \bigcup_{i=1}^{\ell} \boldsymbol{S}_i$. If $\{\boldsymbol{S}_i, i = 1, \ldots, \ell - 1\}$ is a full spanning collection and there is a subset $\boldsymbol{S}_j$ $(1 \le j \le \ell - 1)$ that is rigidly overlapped to $\boldsymbol{S}_\ell$, we have $\{\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots, \boldsymbol{S}_\ell\}$ is a full spanning collection .

Proof: Let $\boldsymbol{S}_i = \{\tau_{i_1}, \ldots \tau_{i_{k_i}}\}$ and write $T_i = [\tau_{i_1}, \ldots \tau_{i_{k_i}}]$ and $T = [\tau_1, \ldots \tau_N]$. Let H_i $(1 \le i \le \ell)$ be the Hessian operator for $\boldsymbol{S}_i$. Let E_i be the selection matrix as in (5.20) such that $T_i = TE_i$ and let $\widetilde{H}_i$ be the embedding of H_i into $\mathbb{R}^{p \times N}$ such that $H_i = \widetilde{H}_i E_i$. Set

$$H = \begin{pmatrix} \widetilde{H}_1 \\ \vdots \\ \widetilde{H}_\ell \end{pmatrix}.$$

We have $\mathsf{null}(H) = \bigcap_{i=1}^{\ell} \mathsf{null}(\widetilde{H}_i)$. It follows $\mathsf{null}(H) = \mathsf{null}(\Phi)$ by Lemma 5.4. Let Φ_i $(1 \le i \le \ell)$ be defined for $\boldsymbol{S}_i$ $(1 \le i \le \ell)$ as (5.22).

Let there be $N - k$ column vectors in $\hat{\boldsymbol{S}}_1 = \bigcup_{i=1}^{\ell-1} \boldsymbol{S}_i$ and there be k $(0 \le k < N)$ column vectors in $\hat{\boldsymbol{S}}_2 = \bigcup_{i=1}^{\ell} \boldsymbol{S}_i \setminus \hat{\boldsymbol{S}}_1$. Without loss of generality, we assume that $\hat{\boldsymbol{S}}_1 = \{\tau_1, \tau_2, \ldots, \tau_{N-k}\}$ and $\hat{\boldsymbol{S}}_2 = \{\tau_{N-k+1}, \ldots, \tau_N\}$. Set $\hat{T}_1 = [\tau_1, \tau_2, \ldots, \tau_{N-k}]$ and $\hat{T}_2 = [\tau_{N-k+1}, \ldots, \tau_N]$. Embedding H_i $(1 \le i \le \ell - 1)$ into $\mathbb{R}^{k \times (N-k)}$ according to the embedding of T_i into $\hat{T}_1$, we have $H_i = \hat{H}_i \hat{E}_i$, where $\hat{E}_i$ is the selection matrix such that $T_i = \hat{T}_1 \hat{E}_i$. Then $\hat{\Phi}_1 = \sum_{i=1}^{\ell-1} \hat{E}_i P_i \hat{E}_i^T$ is the Hessian alignment matrix for the collection $\{\boldsymbol{S}_i, 1 \le i \le \ell - 1\}$, where P_i is defined according to (5.22). Since $\{\boldsymbol{S}_i, 1 \le i \le \ell - 1\}$ being a full spanning collection, we have $\mathsf{null}(\hat{\Phi}_1) = \mathsf{span}([e, \hat{T}_1^T])$. Let

$$\bar{H}_1 = \begin{pmatrix} \hat{H}_1 \\ \vdots \\ \hat{H}_{\ell-1} \end{pmatrix}.$$

We have

$$\mathsf{null}(\bar{H}_1) = \bigcap_{i=1}^{\ell-1} \mathsf{null}(\hat{H}_i) = \mathsf{null}(\hat{\Phi}_1). \tag{5.25}$$

We first consider the case $k = 0$, i.e. $\hat{\boldsymbol{S}}_1 = \{\tau_1, \dots, \tau_N\}$. In this case, we have $\hat{E}_i = E_i$ and $\hat{H}_i = \tilde{H}_i$. Then $\mathsf{null}(\Phi) = \bigcap_{i=1}^{\ell} \mathsf{null}(\tilde{H}_i) \subset \bigcap_{i=1}^{\ell-1} \mathsf{null}(\tilde{H}_i) = \mathsf{null}(\hat{\Phi}_1) = \mathsf{span}([e, T^T])$. Noting also $\mathsf{null}(\Phi) \supset \mathsf{span}([e, T^T])$ by Lemma 5.4, we have $\mathsf{null}(\Phi) = \mathsf{span}([e, T^T])$ and hence $\{\boldsymbol{S}_1, \boldsymbol{S}_2, \dots, \boldsymbol{S}_\ell\}$ is a full spanning collection.

Next, we consider the case $k > 0$. Since $\{\boldsymbol{S}_i, i = 1, \dots, \ell - 1\}$ is a full spanning collection, we have $\mathsf{rank}(\hat{\Phi}_1) = (N - k) - (d + 1)$. It follows

$$\mathsf{rank}(\bar{H}_1) = (N - k) - (d + 1) \tag{5.26}$$

from (5.25). Let

$$\widetilde{H}_\ell = \begin{array}{cc} {\scriptstyle N-k} & {\scriptstyle k} \\ \left(\widetilde{H}_{\ell,1}, \right. & \left. \widetilde{H}_{\ell,2} \right) \end{array}. \tag{5.27}$$

Then we can rewrite H as follows.

$$H = \begin{array}{cc} {\scriptstyle N-k} & {\scriptstyle k} \\ \left(\begin{matrix} \bar{H}_1 \\ \widetilde{H}_{\ell,1} \end{matrix} \right. & \left. \begin{matrix} 0 \\ \widetilde{H}_{\ell,2} \end{matrix} \right) \end{array}.$$

Since one subset $\boldsymbol{S}_j$ $(1 \le j \le \ell - 1)$ is rigidly overlapped to $\boldsymbol{S}_\ell$, the submatrix of $\widetilde{H}_\ell$ consisting of columns corresponding to the vectors in $\boldsymbol{S}_\ell \setminus \boldsymbol{S}_j$, of which $\widetilde{H}_{\ell,2}$ is a submatrix, has full column rank. Therefore $\widetilde{H}_{\ell,2}$ has full column rank. It follows that $\mathsf{rank}(H) \ge N - (d + 1)$. Equivalently, $\mathsf{dimnull}(H) \le d + 1$. Noticing $\mathsf{span}([e, T^T]) \subset \mathsf{null}(H)$ by Lemma 5.4, we have $\mathsf{dimnull}(H) \ge d + 1$. Thus, $\mathsf{null}(H) = \mathsf{null}(\Phi) = \mathsf{span}([e, T^T])$ by Lemma 5.4. The collection $\{\boldsymbol{S}_i, 1 \le i \le \ell\}$ is also a full spanning collection. □

We now proceed to prove our main theorem of this chapter.

Theorem 5.1. Let Φ be the Hessian alignment matrix for a collection of subsets $\{\boldsymbol{S}_1, \boldsymbol{S}_2, \dots, \boldsymbol{S}_s\}$ with $\bigcup_{i=1}^{s} \boldsymbol{S}_i = \{\tau_1, \dots, \tau_N\}$. Assume that there are two nonempty collections, say, $\{\boldsymbol{S}_i, i = 1, \dots, p\}$ and $\{\boldsymbol{S}_i, i = p + 1, \dots, s\}$, such that $\{\boldsymbol{S}_i, i =$

$1, \ldots, p\}$ is a full spanning collection and $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$ is rooted connected with root $\boldsymbol{S}_j$ (for some j with $1 \leq j \leq p$). Then we have $\mathsf{null}(\Phi) = \mathsf{span}([e, T^T])$, where $T = [\tau_1, \ldots, \tau_N]$.

Proof: By Lemma 5.5, we can expand $\{\boldsymbol{S}_i, i = 1, \ldots, p\}$ to obtain a new full spanning collection collection by including every subset that is rigidly overlapped to $\boldsymbol{S}_j$ (for some j with $1 \leq j \leq p$). This can be expanded into the full collection since every $\boldsymbol{S}_i$ (for $i = 1, \ldots, s$) is rooted connected with root $\boldsymbol{S}_j$. □

The previous theorem shows that we can recover T from the Hessian alignment matrix provided the conditions on the the collection of subsets are satisfied. The Hessian alignment matrix Φ here is constructed from the original coordinates τ_i's in each subset and this is a generalization of the $\mathcal{H}$-functional in the isometric coordinate $\mathcal{H}^{iso}$. In the next section, we consider a more general definition of the Hessian alignment matrix constructed from some local coordinates, which generalize the $\mathcal{H}$-functional in the tangent coordinate $\mathcal{H}$.

5.2 Analysis of Discrete Hessian Eigenmaps

We now discuss the problem of how to reconstruct the global coordinates τ_i's for a given data set from their local coordinates as outlined at the beginning of the section using the Hessian alignment matrix. We outline the procedure as the following algorithm that we call discrete Hessian Eigenmaps.

Algorithm 5.1. Discrete Hessian Eigenmaps
Given $\boldsymbol{X} = \{x_1, \cdots, x_N\} \subset \mathbb{R}^m$.

1. Construct $\{\boldsymbol{X}_i, i = 1, \ldots, s\}$ with $\boldsymbol{X}_i = \{x_{i_1}, \ldots, x_{i_{k_i}}\}$ consisting of points in a small neighborhood and $\bigcup_{i=1}^s \boldsymbol{X}_i = \boldsymbol{X}$.

2. For each $\boldsymbol{X}_i$, construct its local coordinates $\boldsymbol{\Theta}_i = \{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\} \in \mathbb{R}^d$. This can be done by the projection onto an approximate local tangent space as in (2.32)

3. Construct $Q_i = G_i^\dagger G_i$ where G_i $(1 \leq i \leq s)$ is the discrete Hessian operator for $\boldsymbol{\Theta}_i$ $(1 \leq i \leq s)$.

4. Construct
$$\Psi = \sum_{i=1}^{s} E_i Q_i E_i^T. \tag{5.28}$$
We call Ψ the Hessian alignment matrix for $\{\mathbf{\Theta}_i\}$.

5. Compute $[e/\sqrt{N}, T^T]$ as an orthonormal basis of the spectral subspace of Φ corresponding to the smallest $d+1$ eigenvalues, where $T^T \in \mathbb{R}^{N\times d}$. The columns of T are used as the coordinate set for $\boldsymbol{X}$.

For the Discrete Hessian Eigenmaps method, we construct a local coordinate system $\mathbf{\Theta}_i = \{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\} \in \mathbb{R}^d$ for points in a local neighborhood $\boldsymbol{X}_i$, from which the Hessian alignment matrix Ψ is constructed. This is similar to how the $\mathcal{H}$-functional in the tangent coordinate $\mathcal{H}$ is defined. As in the continuous case, assuming that the local coordinate system is exact, we now show that the alignemnt matrices as constructed from the local coordinates and the original coordinates are the same and we still have $\mathsf{null}(\Psi) = \mathsf{span}\left([e, T^T]\right)$.

Lemma 5.6. Let G_i and H_i be the Hessian operators for two ordered subsets $\mathbf{\Theta}_i = \{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\}$ and $\boldsymbol{S}_i = \{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\}$, respectively. Set $\Theta_i = [\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}]$ and $T_i = [\tau_{i_1}, \ldots, \tau_{i_{k_i}}]$. If $\Theta_i = VT_i + ce^T$ for some $c \in \mathbb{R}^d$ and some nonsingular matrix V, we have $G_i^\dagger G_i = H_i^\dagger H_i$.

Proof: By Lemma 5.3, we have $\mathsf{span}(G_i^T) = \mathsf{span}(H_i^T)$. Since $G_i^\dagger G_i$ and $H_i^\dagger H_i$ are the orthogonal projections on $\mathsf{span}(G_i^T)$ and $\mathsf{span}(H_i^T)$ respectively, the lemma is proved. □

Theorem 5.2. Let $\mathbf{\Theta}_i$ and Ψ be obtained from the Discrete Hessian Eigenmaps method (Algorithm 5.1). Let $x_i = \psi(\tau_i)$ and $\boldsymbol{S}_i = \{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\}$. Let Φ be the Hessian alignment matrix for the collection of subsets $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$. Assume $\mathbf{\Theta}_i = \{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\} \subset \mathbb{R}^d$, is isometric in the Euclidean distance to $\{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\}$. Then
$$\Psi = \Phi.$$

In particular, if $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$ is a full spanning collection, $\mathsf{null}(\Psi) = \mathsf{span}\left([e, T^T]\right)$, where $T = [\tau_1, \ldots, \tau_N]$.

Proof: Since $\{\theta_1^{(i)}, \ldots, \theta_{k_i}^{(i)}\}$ is isometric to $\{\tau_{i_1}, \ldots, \tau_{i_{k_i}}\}$, it follows from the proof of Theorem 2.4 of [30] that there exist an orthogonal matrix $\hat{V}_i$ and a vector $c \in \mathbb{R}^d$ such that

$$\Theta_i = \hat{V}_i T_i + c e^T$$

Then we have $G_i^\dagger G_i = H_i^\dagger H_i$ by Lemma 5.6. Thus

$$\Psi = \sum_{i=1}^{s} E_i G_i^\dagger G_i E_i^T = \sum_{i=1}^{s} E_i H_i^\dagger H_i E_i^T = \Phi.$$

If $\{\boldsymbol{S}_i, i = 1, \ldots, s\}$ is full spanning collection, we have $\mathsf{null}(\Psi) = \mathsf{span}\left([e, T^T]\right)$. □

Chapter 6 Conclusions and Future Work

In this thesis, we have shown in Chapter 2 that the alignment algorithm can recover global parametrization properly even when local neighborhoods/sections have different intrinsic dimensions. This is a property not known for other manifold learning algorithms and would be an advantage of the alignment algorithm. Our examples confirm our theoretical finding. We have also proposed an application of the alignment algorithm to a semi-supervised learning problem. Our examples have demonstrated that this approach compares very favorably with several other methods that have been proposed.

In Chapter 3, we have presented some characterizations of eigenvalues of the alignment matrix defined on local coordinates, from which we have derived the null space and a lower bound of the smallest positive eigenvalue. The bound suggests a quadratic dependence of the smallest positive eigenvalue on the amount of overlap, which is confirmed in our numerical tests. Our results demonstrate the robustness of using the alignment matrix for reconstructing global coordinates, in the sense that most spectral properties of the alignment matrix are preserved even when the local coordinates are computed with large errors in certain components. This together with earlier analysis on the null space provides a solid theoretical basis for the LTSA method.

In Chapter 4, we have also analyzed the discrete process of Hessian Eigenmaps method by investigating the null space of Hessian operator and Hessian alignment matrix coming from the local coordinates and isometric coordinates. We prove that discrete Hessian Eigenmaps can recover the isometric coordinates of the manifold up to a rigid motion under certain condition. Matrix analysis techniques such as spectral analysis will play an important role in the understanding of the performance of the algorithms of manifold learning and dimensionality reduction.

We finally mention a few possible future works. Firstly, we may apply our theoretical analysis for the current methods such as LTSA and Hessian Eigenmaps to the investigation of other nonlinear dimensionality reduction methods. Secondly, we notice that one of the critical steps in the computation of both LTSA and Hessian Eigenmaps is finding the null space of large sparse matrices. How to improve the speed and accuracy of the null space computation for those large sparse matrices is an important and interesting problem. We hope our understanding in the eigenstructure of the alignment matrix could help build more efficient algorithms for this computational problem.

Bibliography

[1] M. Belkin and P. Niyogi, *Laplacian Eigenmaps for Dimensionality Reduction and Data Representation*, Neural Computation, 15 (2002), pp. 1373–1396.

[2] R. Bellman, *Adaptive control processes: a guided tour*, Princeton University Press, New Jersey, 1961.

[3] V. Blanz and T. Vetter, *A Morphable Model for the Synthesis of 3D Faces*, SIGGRAPH, (1999), pp. 187–194.

[4] M. Brand, *Charting a manifold*, in Advances in Neural Information Processing Systems 15, S. T. S. Becker and K. Obermayer, eds., MIT Press, Cambridge, MA, 2003, pp. 961–968.

[5] F.M. Cox and M.A.A. Cox, *Multidimensional scaling*, Chapman and Hall, 2001.

[6] J. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[7] F. Diaz and D. Metzler *Pseudo-aligned multilin- gual corpora*, The International Joint Conference on Articial Intelligence, (2007), pp. 2727–2732.

[8] D. Donoho and C. Grimes, *Hessian eigenmaps: new tools for nonlinear dimensionality reduction*, Proceedings of National Academy of Science, 100 (2003), pp. 5591–5596.

[9] J. Ham, D Ahn and D. Lee, *Learning a manifoldconstrained map between image sets: applications to matching and pose estimation*, IEEE Computer Vision and Pattern Recognition, (2006) pp. 817–824.

[10] J. Ham and D. Lee, and L. Saul, *Semisupervised alignment of manifolds*, Proceedings of the International Conference on Artificial Intelligence and Statistics, (2005), pp. 120–127.

[11] T. Hastie and W. Stuetzle, *Principal curves*, Journal of American Statistics Association, 84, (1988), pp. 502–516.

[12] J. D. Horel, *Complex principal component analysis: Theory and examples*, Journal of climate and Applied Meteorology, Vol. 23, No. 12, (1984), pp. 1660–1673.

[13] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, 1989.

[14] N. Kambhatla, *Dimension reduction by local principal component analysis*, Neural Computation, Vol. 9, No. 7, (1997), pp. 1493–1516.

[15] S. Y. Kung and K. I. Diamantaras,*A neural network learning algorithm for adaptive principal component extraction*, International Conference on Acoustics, Speech, and Signal Processing, Vol. 2, (1990), pp. 861–864.

[16] C.-K. Li, R.-C. Li and Q. Ye, *Eigenvalues of an alignment matrix in nonlinear manifold learning*, Communications in Mathematical Sciences, 5 (2007), pp. 313–329.

[17] A. E. Naiman, I. M. Babuška, and H. C. Elman, *A note on conjugate gradient convergence*, Numer. Math., 76 (1997), pp. 209–230.

[18] E. Oja, *A principal components, minor components, and linear neural networks*, Neural Networks, vol. 5, no. 6, (1992), pp. 927–936.

[19] N. Patwari and A.O.I. Hero, *Manifold learning algorithms for localization in wireless sensor networks*, Proc. 29th IEEE Int. Conf. Acoustics, Speech, Signal Processing, (2004) pp. 857–860.

[20] , A. Plummer, M. Beckman, M. Belkin, E. Fosler-Lussierand and B. Munson, *Learning speaker normalization using semisupervised manifold alignment*, Proc. Interspeech, (2010), pp. 2918–2921.

[21] R. W. Preisendorfer, *Principal Component Analysis in Meteorology and Oceanography*, Developments in Atmospheric Science, Vol.17. Elsevier, Amsterdam. 1988.

[22] S. T. Roweis and L. K. Saul, *Nonlinear dimensionality reduction by locally linear embedding*, Science, 290 (2000), pp. 2323–2326.

[23] L. Saul, K. Weinberger, J. Ham, F. Sha, and D. Lee. Spectral methods for dimensionality reduction. In O. Chapelle, B. Schoelkopf, and A. Zien, editors, *Semisupervised Learning.* 2006.

[24] G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*, Academic Press, Boston, 1990.

[25] G. Strang and G. Fix, *An Analysis of Finite Element Methods*, Wesley-Cambridge Press, Wesley, MA, 1973.

[26] Y. W. Teh and S. Roweis, *Automatic alignment of local representations*, in Advances in Neural Information Processing Systems 15, S. T. S. Becker and K. Obermayer, eds., MIT Press, Cambridge, MA, 2003, pp. 841–848.

[27] J. B. Tenenbaum, V. de Silva, and J. C. Langford, *A global geometric framework for nonlinear dimensionality reduction*, Science, 290 (2000), pp. 2319–2323.

[28] C. Wang and S. Mahadevan, *Manifold alignment using procrustes analysis*, Proceedings of the 25th International Conference on Machine Learning, (2008), pp. 1120–1127.

[29] C. Williams, *On a connection between kernal PCA and metric multidimensional scaling*, Machine Learning, 46 (2002), pp. 11–19.

[30] Q. Ye, H. Zha, and R.-C. Li, *Analysis of an alignment algorithm for nonlinear dimensionality reduction*, BIT - Numerical Mathematics, 47 (2007), pp. 873–885.

[31] Q. Ye and W. Zhi, *Alignments of manifold sections of different dimensions in manifold learning*, AAAI 2010 Fall Symposium on Manifold Learning and its Applications, (2010), Arlington, VA.

[32] H. Zha and Z. Zhang, *Spectral analysis of alignment in manifold learning*, in Proceedings of Acoustics, Speech, and Signal Processing (ICASSP '05), vol. 5, March 2005, pp. 1069–1072.

[33] H. Zha and Z. Zhang, *Continuum Isomap for manifold learning*, Computational Statistics and Data Analysis, Vol. 52, (2007), pp. 184–200.

[34] Z. Zhang and H. Zha, *Principal manifolds and nonlinear dimension reduction via local tangent space alignment*, SIAM J. Sci. Comput., Vol. 26, (2004), pp. 313–338.

[35] Z. Zhang, H. Zha and M. Zhang, *Spectral methods for semi-supervised manifold learning*, IEEE Conference on Computer Vision and Pattern Recognition, (2008), pp. 1–6.