Learning Topological Structures and Vector Fields on Manifolds with (Higher-order) Discrete Laplacians

Yu-Chia Chen

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2021

Reading Committee:

Marina Meilă, Chair

Les Atlas

Yen-Chi Chen

Program Authorized to Offer Degree:

Electrical Engineering

University of Washington

Abstract

Learning Topological Structures and Vector Fields on Manifolds with (Higher-order) Discrete Laplacians

Yu-Chia Chen

Chair of the Supervisory Committee:
Professor Marina Meilă
Department of Statistics

Unsupervised learning algorithms, which extract geometric information without labels, are pivotal in analyzing high-dimensional observational data in complex physical and social systems. Prior accomplishments of scientific discoveries with these methods include applying (i) non-linear dimensionality reduction, also called manifold learning (ML), algorithms in revealing hidden structures of quantum chemistry and astronomy datasets. Additionally, (ii) clustering analysis techniques are critical for categorizing stages in cellular differentiation or analyzing community structures in social networks. Finally, (iii) topological data analysis (TDA) methods are essential for investigating the cyclic/periodic structures in the neuroscientific, galactic, and human action systems.

Despite these early successes in the scientific community, the vast majority of the unsupervised learning methodologies are highly unexplored. For instance, how can we learn from a dataset equipped with temporal information? On the other hand, how can we tell/test whether the obtained topological structures are signals instead of noises or algorithmic defects? Lastly, can we extend the current unsupervised learning framework to deal with the higher-order (e.g., triplet-wise) relations? If so, what potentials does it open up?

In this thesis, we will answer some of these questions under the lens of differential geometry, topology, and machine learning. This thesis centers around the estimators for the

Laplace-Beltrami operator Δ_0 of a manifold $\mathcal{M}$ and its higher-order counterparts Δ_k (called the k-Laplacian). In particular, we are interested in the spectral (i.e., the eigenvalues and the eigenvectors) properties of these estimators.

First, we analyze a known deficiency in the outputs of the standard embedding algorithms when the aspect ratio of the manifold is large. This deficiency, called the Independent Eigencoordinate Search (IES) problem, arises due to the functional dependencies in the eigenfunctions of Δ_0. We address the IES problem by proposing a bicriterial algorithm that has a low computational overhead and has an analyzable asymptotic limit.

Second, the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$ (a first-order extension of the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ to the edge space) is introduced to enrich the manifold learning methodology. We provide a theoretical analysis of the large sample limit of $\boldsymbol{\mathcal{L}}_1$ and show its connection to the manifold Helmholtzian (1-Laplacian) Δ_1, an operator that acts on vector fields on the manifold. The proposed Helmholtzian estimator $\boldsymbol{\mathcal{L}}_1$ made it possible to distill higher-order topological structures, such as the first homology vector space $\mathcal{H}_1$ encoding the cyclic information.

Third, we explore the possibility of utilizing the vector field basis defined from the eigenflows of $\boldsymbol{\mathcal{L}}_1$; specifically, we study the extensions of the learning algorithms that are based on $\boldsymbol{\mathcal{L}}_0$ to vector fields smoothing, vector field interpolation, and inferring underlying vector fields from sparsely observed trajectories.

Lastly, we study the decomposition of the k-th homology vector space $\mathcal{H}_k$ (null space of the k-Laplacian $\boldsymbol{\mathcal{L}}_k$) of the sparsely connected manifolds; under this condition, we show that the homology embedding can be roughly factorized. Our analysis is conducted by viewing the connected sum (gluing) of manifolds as a perturbation to the matrix $\boldsymbol{\mathcal{L}}_k$. We exemplify the efficacy of the proposed framework by applying it to the *shortest homologous loop detection* problem, a problem known to be NP-hard in general.

We support our claims in each section with an extensive set of experiments on synthetic manifolds along with real datasets from chemistry, biology, medical imaging, and astronomy.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACKNOWLEDGMENTS

I would like to thank my advisor, Professor Marina Meilă. Her passion for research and her depth of knowledge has inspired me throughout my graduate studies. Numerous ideas leading to the core parts of this thesis would not have materialized without her guidance.

Without Professor Ioannis G. Kevrekidis' insights, the vector field analysis framework would not have come to light. I am grateful for his assistance and collaboration. I would also like to thank Professor Hugh Hillhouse. It was a pleasure to collaborate with him, and he has taught me so much about interdisciplinary scientific research. Additionally, I am grateful to Professor Les Atlas, Professor Yen-Chi Chen, and Professor David A. C. Beck. This thesis would not have taken shape without their astute comments and feedback.

Furthermore, I would like to thank all my friends: Pearl Woo, Yen-Ju Chen, Samson Koelle, Dr. Anupum Pant, Tanner Fiez, Dr. Avleen S. Bijral, Dr. Jimmy Kuo, Chun-Hao Chang, Dr. Yueyang Chen, Yu Tse Heng, Dr. Daniel Ting, Zhenman Yuan, Dr. Wiley Dunlap-Shohl, Dr. Timothy D. Siegler, Preetham P. Sunkari, and many others. Everything that they taught me, the support that they provided, and the fun times that we have spent together will all be cherished memories of my time in graduate school.

Above all, I would like to express my ultimate gratitude to my parents, brother, and grandmother. They have always said that they are proud of me for pursuing my Ph.D.; however, I am prouder of them for being a supportive family. They are always there when I feel like I am failing or helpless, and words cannot describe how grateful I am to them.

Lastly, I would like to thank my mentors, my friends, and my family once more for their continuous love and support. With them, I was able to navigate through the hard, lonely, and boring life of the COVID-19 pandemic.

DEDICATION

to my faimly

Chapter 1

INTRODUCTION

Recent advances in computer science fuel the collection of large-scale and high-dimensional data (such as point clouds, trajectories, time series, etc.) sampled from complex biological, physical, and social systems. Being able to analyze these complex datasets with theoretically justified methods can bypass the *curse of dimensionality* and expedite interdisciplinary scientific discoveries. *Unsupervised learning algorithms* are widely applied to extract underlying structures in these datasets. They include manifold learning (ML) algorithms [10, 32, 118, 120] that aim at obtaining a low-dimensional embedding from the data sampled from a manifold, clustering methods [47, 65, 79, 87] that partition data into groups, density estimation procedures [25, 32, 110] for estimating underlying probability density function, topological data analysis algorithms [21, 22, 92, 128] that extract the number of high-dimensional *holes* (e.g., loops and cavities), etc.

Notwithstanding the advances of these methodologies in the scientific community, there are several limitations of the unsupervised learning algorithms that hinder their applicability. For instance, (i) many of these algorithms have no clearly defined success criterion, unlike the supervised learning tasks. Therefore, solid knowledge in *both* the algorithms and the domains of interest are oftentimes needed for practitioners to determine whether the estimates are indeed signals or merely errors amplified from noise in datasets or algorithmic artefacts (see, e.g., discussions in Alquicira-Hernandez et al. [3] and Novembre and Stephens [88]). Furthermore, (ii) the outputs of these algorithms are sometimes hard to interpret; extra efforts are needed to connect these outputs to descriptors that have scientific meaning [36, 80]. Lastly, (iii) data with temporal information, such as displacement or velocity fields, has recently been comprehensively collected; however, systematic ways to process, analyze, and distill information from these datasets are currently lacking.

In this thesis, we partially address the aforementioned limitations under the lens of differential geometry, topology, matrix perturbation theory, and manifold learning. The core instruments extensively used in this thesis are the k-Hodge Laplacian Δ_k (that encodes geometric and topological information) and its discrete estimator. For $k = 0$ and $k = 1$, the corresponding (continuous) Laplacians are called the Laplace-Beltrami operator Δ_0 (acting

on a scalar function on a manifold) and the manifold Helmholtzian Δ_1 (operating on a vector field on a manifold), respectively. We are especially interested in the spectral properties of the discrete k-Laplacian $\boldsymbol{\mathcal{L}}_k$ [58]; specifically, our goal is to turn the learning of geometric and topological information into solving an eigenproblem or a linear system involving $\boldsymbol{\mathcal{L}}_k$. Chapter 2 covers the tools, techniques, and background for constructing $\boldsymbol{\mathcal{L}}_k$ from the high-dimensional point clouds; most of them are built upon the differential geometry, Hodge theory, and (higher-order) graph theory.

1.1 The Independent Eigen-coordinate Selection (IES) problem (Chapter 3)

In this chapter, we study a well-documented deficiency [51] (the IES problem) of a subset of manifold learning algorithms (called the *local, spectral methods* [119]) when the data manifold has a large aspect ratio, such as a long and thin strip. This defect arises since the eigen-coordinates of the limiting operators (e.g., the Laplace-Beltrami operator Δ_0 for the Diffusion Maps (DM) algorithm [32]) are not functionally independent even though they are mutually orthogonal to each other. The hardness of the IES problem lies in the fact that it is pervasive and more complex than previously recognized, i.e., new topological structures and neighborhood relationships can be created or destroyed when the algorithm fails.

In this chapter, we study the IES problem under the DM algorithm; however, it can be easily generalized to other local, spectral methods such as LTSA [131] or Hessian Eigenmaps (HLLE) [42]. Our contributions are to (i) formulate the success and failure of the algorithm in terms of finding a smooth and full-rank embedding that maps the original high-dimensional points to a low-dimensional subspace. Mathematically speaking, (ii) success is possible under broad conditions, provided that embedding is obtained by carefully selected eigenfunctions of the Laplace-Beltrami operator Δ_0 [7]. (iii) We then propose a bicriterial IES algorithm that selects smooth embeddings with few eigenvectors. These criteria are based on the projected volume onto a subset of coordinates. (iv) Additionally, we analyze the population version of the criteria and show that they are reminiscent of a Kullbach-Leibler (KL) divergence between unnormalized measures on the subset of coordinates. Lastly, (v) we show how to

drastically reduce the computational burden per subset of the coordinates for our algorithm.

1.2 The manifold Helmholtzian estimator (Chapter 4)

The manifold Helmholtzian (1-Laplacian) operator Δ_1 elegantly generalizes the Laplace-Beltrami operator Δ_0 to mapping vector fields on a manifold $\mathcal{M}$ to their second-order derivatives. Similar to Δ_0, the Helmholtzian contains topological and geometric information, with part of it not accessible in the space of scalar functions. This information encodes the non-trivial closed loops (called *1-dimensional holes*) of $\mathcal{M}$ and can be revealed by the null vector space of Δ_1 (called the first homology vector space $\mathcal{H}_1$). Furthermore, the *eigenforms* (vector fields) of Δ_1, which are obtained by solving the eigenproblem $\Delta_1\zeta = \lambda\zeta$ with proper boundary conditions, construct a basis for any vector field on the manifold by the well-known Helmholtz-Hodge Decomposition (HHD) [16, 73].

In this chapter, we initiate the estimation of the manifold Helmholtzian from high-dimensional point clouds and explore the possibility for topological feature discovery with the proposed estimator. The key mathematical ingredient in this chapter is the weighted (discrete) 1-Laplacian $\boldsymbol{\mathcal{L}}_1$ [58, 105], a matrix operating on the edges of a simplicial complex (a generalization of a graph). While higher-order Laplacians were introduced over seven decades ago [46], this work is the first to present a discrete Helmholtzian constructed from a simplicial complex as an estimator for the continuous operator in a non-parametric setting. This Helmholtzian estimator opens up new avenues in topological feature discovery (Chapters 4 and 6) and vector field learning (Chapter 5).

The challenge of proposing a proper Helmholtzian estimator lies in its asymptotic limit; specifically, since the points are randomly sampled from a continuous manifold $\mathcal{M}$, $\boldsymbol{\mathcal{L}}_1$ itself is a collection of random variables. Therefore, one has to make sure the corresponding limit (when the sample size n tends to infinity) is intuitive and analyzable. It is not guaranteed for an arbitrary chosen (or heuristic) estimator to have a limit; for instance, the exponential random graph model in network science is not always guaranteed to converge [107, 108].

The main contributions are to (i) propose a weighting function on triangles of the simpli-

cial complex constructed from a point cloud such that the corresponding *up* Helmholtzian has an asymptotic limit in the form of Δ_1^{up}. Additionally, (ii) the spectral consistency of the *down* Helmholtzian can be derived using the connection of its spectrum to those of the corresponding graph Laplacian ($\boldsymbol{\mathcal{L}}_0$). We support our theoretical claims by (iii) experiments on matching the spectra of synthetic manifolds and (iv) discovering topological structures in the molecular dynamics datasets of small molecules.

1.3 Smoothing and learning vector fields with the Helmholtzian (Chapter 5)

Vector fields are used to represent displacements and to model time-dependent phenomena across many scientific disciplines. The celebrated Hodge theory and HHD establish a mathematical framework for the analysis of vector fields in arbitrary dimensions; they define concepts such as sources and sinks (where fluxes are generated or disappear), cyclicity (*harmonics*), and turbulence (categorized by *curl*) in the systems. Central to this decomposition is the manifold Helmholtzian Δ_1, whose eigenforms construct a basis of the vector field supported on the manifold. However, analyzing complex vector fields from different datasets might be challenging under the current framework due to the vast domain-specific modeling techniques required.

Hence, proposing *non-parametric* methodologies for vector fields, which requires little prior knowledge of the systems and domains of interest, is crucial for expediting interdisciplinary scientific discovery. Non-parametric learning algorithms on manifolds [10, 90, 134], such as signal processing or semi-supervised learning (SSL) tasks for scalar fields on a manifold, have advanced with the introduction of the discrete Laplacians $\boldsymbol{\mathcal{L}}_0$ [32]. Now, we would like to extend these algorithms to the vector field situatin with the framework proposed in the previous chapter.

From the proposed discrete Helmholtzian estimator $\boldsymbol{\mathcal{L}}_1$, the basis vector fields are estimated from the eigenflows of $\boldsymbol{\mathcal{L}}_1$. Containing the geometric and topological information about $\mathcal{M}$, these basis vector fields and the corresponding subspaces obtained from HHD of $\boldsymbol{\mathcal{L}}_1$ are powerful tools for analyzing flows and vector fields on $\mathcal{M}$. Our contributions are to

expand the signal processing or SSL algorithms from scalar to vector fields on a manifold with the aid of the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$. Namely, we design (i) a vector field smoothing algorithm by projecting a noisy vector field onto the low-frequency (small eigenvalues) subspaces. Additionally, since the large-scale structure of a vector field is constrained by the underlying manifold, (ii) a handful of low-frequency flows can be used to inform the flows that are not previously observed, also called the edge flow semi-supervised learning (SSL). Lastly, a direct application of the edge flow SSL is to (iii) infer the underlying vector field from partially observed trajectories. We demonstrate these possibilities on datasets from quantum chemistry, oceanography, and cell biology.

1.4 Decomposing the homology embedding of the k-Laplacian (Chapter 6)

In this chapter, we turn our focus on the k-Laplacian $\boldsymbol{\mathcal{L}}_k$, a higher-order generalization of the graph Laplacian and discrete Helmholtzian. Specifically, we investigate the null space of $\boldsymbol{\mathcal{L}}_k$ (called the *k-th homology vector space* $\mathcal{H}_k$), which encodes the *k-dimensional holes* of a manifold or a network. For instance, the zeroth homology space $\mathcal{H}_0$ identifies clusters, $\mathcal{H}_1$ reveals the independent loops in the manifold, and the cavity information can be extracted from the second homology vector space $\mathcal{H}_2$.

Understanding the structure of the homology vector space $\mathcal{H}_k$ can disclose geometric or topological information from the data. The study of the embedding of $\mathcal{H}_0$ (constructed from the null space of the graph Laplacian $\boldsymbol{\mathcal{L}}_0$) has spurred new research and applications. For example, spectral clustering and community detection algorithms with theoretical guarantees; their theoretical analyses are based on either matrix perturbation [79, 87] or under a mixture model setting [106]. However, the higher-order (for $k \geq 1$) counterparts have rarely been attempted due to their combinatorial and intractable natures.

We investigate the decomposition of the k-th homology embedding (of $\mathcal{H}_k$) and focus on cases reminiscent of spectral clustering. Namely, we analyze the *connected sum* (gluing) [71] of manifolds as a perturbation to the direct sum (concatenation) of their homology embeddings. We show that, under the lens of the Davis-Kahan [130], (i) if the manifold can

be expressed as a series of connected sum, then the homology embedding of the combined manifold can be roughly factorized into disjoint subspaces with each corresponding to a single homology space of the disjoint manifold. Based on the analysis above, (ii) the factorization of the homology embedding into subspaces becomes a *blind source separation* problem and can (iii) be obtained efficiently using independent component analysis [11].

The proposed framework can (iv) support numerous applications such as localization of the harmonic edge-flows, higher-order clustering [45], and the *shortest homologous loop detection* problem [36]. We emphasize (v) the applicability of this framework by proposing a spectral homologous loop detection algorithm and show its efficacy on diverse data such as high-dimensional point clouds and images.

Chapter 2

BACKGROUND

2.1 Problem formulations

Suppose we observe data $\mathbf{X} \in \mathbb{R}^{n \times D}$ (called a *point cloud*), with points denoted by $\mathbf{x}_i \in \mathbb{R}^D$ for $i = 1, \cdots, n$, from a high dimensional Euclidean space $\mathbb{R}^D$ (i.e., large D). Furthermore, these points $\mathbf{x}_i$'s are sampled with density $\psi(\mathbf{x})$ from a low ($d \ll D$) d-dimensional (non-linear) subspace of $\mathbb{R}^D$, such as a plane, a sphere, or a torus, called a *manifold*.

In the classical manifold task, we are interested in obtaining an *embedding* ϕ that maps points $\mathbf{x}_i \in \mathbb{R}^D$ to $\phi(\mathbf{x}_i) \in \mathbb{R}^s$ with $s \geq d$ such that "some" geometric and topological properties are "preserved". This task is our main objective in Chapter 3.

In addition to point cloud data $\mathbf{X}$, we might also observe temporal or displacement information such as a sparse vector field $\mathbf{v}_i \in \mathbb{R}^D$ at each point $\mathbf{x}_i$ or trajectories (an edge flow) that traverse through these high-dimensional points. We will develop the machinery for this problem in Chapter 4 and present the applications in Chapter 5.

Finally, the topological information, such as clusters, loops, cavities, etc., of the manifold $\mathcal{M}$ can be identified using the vector subspace of higher-order simplices (k-polytope) of points $\mathbf{x}_i$. We will discuss this task in Chapter 6 using the tools developed in Chapter 4.

2.2 Differential geometry and topology

In this section, we cover the mathematical background, terminologies, definitions, and theorems necessary for this thesis in the field of differential geometry and topology; readers are encouraged to refer to Lee [71] for a complete overview of differential geometry. Additionally, Armstrong [4] and Hatcher [54] provide comprehensive reviews of topology.

2.2.1 Differential geometry

In this section, we will formalize the notions of *manifold*, *embedding*, as well as the meaning of "preserving geometry" using the idea of *isometry* and *Riemannian metric*. We conclude this section by introducing the Laplace-Beltrami operator, which can be used to recover the geometric information of the manifold $\mathcal{M}$. Concepts discussed in this section will be used

in Chapter 3. We start with the main building block of the manifold learning algorithm: *smooth manifolds*.

Definition 2.1 (Smooth manifold)**.** A d-dimensional *topological manifold* $\mathcal{M}$ is a Hausdorff (topological) space such that: (i) there exists a countable basis for the topology of $\mathcal{M}$, and (ii) every point in $\mathcal{M}$ has a neighborhood that is homeomorphic to an open subset of the d-dimensional Euclidean space $\mathbb{R}^d$.

A *coordinate chart* (U, φ) of a manifold $\mathcal{M}$ is a tuple of an open set U of $\mathcal{M}$ and a homeomorphism $\varphi : U \to V$ to an open subset $V \subseteq \mathbb{R}^d$. Define I by an index set such that $\mathcal{M} = \cup_{i \in I} U_i$. If the transition map

$$\varphi_j \circ \varphi_i^{-1} : \varphi_i(U_i \cap U_j) \to \mathbb{R}^d$$

is smooth (infinitely differentiable) for any $i, j \in \mathcal{I}$, then a C^∞-*Atlas* $\mathcal{A}$ is a collection of coordinate charts, i.e.,

$$\mathcal{A} \coloneqq \bigcup_{i \in I} (U_i, \varphi_i).$$

Lastly, a *smooth manifold* is a tuple $(\mathcal{M}, \mathcal{A})$ where $\mathcal{M}$ is a topological manifold and $\mathcal{A}$ is a smooth C^∞-Atlas.

As discussed earlier, we are interested in obtaining a low dimensional representation, called an *embedding*, of the original dataset $\mathbf{X}$ (sampled from a manifold) for classical manifold learning tasks. We formalize the concept of embedding as follows.

Definition 2.2 (Embedding)**.** If $\mathcal{M}$ and $\mathcal{N}$ are smooth manifolds, an *embedding* of $\mathcal{M}$ into $\mathcal{N}$ is a smooth immersion $\phi : \mathcal{M} \to \mathcal{N}$ with rank being the dimension of $\mathcal{M}$ (i.e., $\text{rank}(\phi) = \dim(\mathcal{M})$) that is also homeomorphism onto its image $\phi(\mathcal{M})$.

The embedding is not restricted to points sampled from a manifold only. For instance, in Chapter 6, we investigate the higher-order k-simplex embedding, which is the embedding on higher-order simplices (k-dimensional polytope).

Next, we study the preservation of geometric quantities under an embedding ϕ; we first introduce the notion of *Riemannian metric* and *Riemannian manifold*.

Definition 2.3 (Riemannian metric and Riemannian manifold)**.** A *Riemannian metric* g is a symmetric and positive definite tensor field that defines an inner product $\langle , \rangle_g$ on the tangent space $\mathcal{T}_p\mathcal{M}$ for every point $p \in \mathcal{M}$.

A *Riemannian manifold* is a tuple $(\mathcal{M}, g)$ of a smooth manifold $\mathcal{M}$ equipped with a Riemannian metric g defined on every point $p \in \mathcal{M}$.

As discussed earlier, a manifold $\mathcal{M}$ is locally d-dimensional Euclidean space; equipped with the Riemannian metric g, one can define the

1. inner product of $u, v \in \mathcal{T}_p\mathcal{M}$ for all $p \in \mathcal{M}$:

$$\langle u, v \rangle_{g(p)} = \sum_{i=1}^{d} \sum_{j=1}^{d} g_{ij}(p) u_i v_j;$$

2. path length of a piece-wise smooth parametric curve $\gamma : [a, b] \to \mathcal{M}$:

$$\int_a^b \sqrt{\langle \gamma'(t), \gamma'(t) \rangle_{g(\gamma(t))}} \mathsf{d}t; \text{ and}$$

3. volume of the manifold $\mathcal{M}$ (with $|g(p)| \coloneqq |\det(g(p))|$):

$$\mathrm{Vol}(\mathcal{M}) = \int_{p \in \mathcal{M}} \sqrt{|g(p)|} \mathsf{d}s_1 \cdots \mathsf{d}s_d.$$

A smooth embedding $\phi : \mathcal{M} \to \phi(\mathcal{M})$ does not typically preserve geometric quantities, such as distances along curves in $\mathcal{M}$. One can always associate with $\phi(\mathcal{M})$ a Riemannian metric $g_{*\phi}$, called the *pushforward Riemannian metric* [71], which preserves the geometry of $(\mathcal{M}, g)$. Before introducing the *pushforward Riemannian metric*, we visit the notion of *isometry*, which defines the quantity that is "preserved" under a smooth embedding $\phi : \mathcal{M} \to \mathcal{N}$.

Definition 2.4 (Isometry)**.** Let $\phi : \mathcal{M} \to \mathcal{N}$ be a diffeomorphism from a Riemannian manifold $(\mathcal{M}, g_{\mathcal{M}})$ to another Riemannian manifold $(\mathcal{N}, g_{\mathcal{N}})$. ϕ is called an isometry if and only if for all $p \in \mathcal{M}$ and $u, v \in \mathcal{T}_p(\mathcal{M})$, the following relationship holds:

$$\langle u, v \rangle_{g_{\mathcal{M}}(p)} = \langle \mathfrak{J}\phi(p)u, \mathfrak{J}\phi(p)v \rangle_{g_{\mathcal{N}}(\phi(p))},$$

where $\mathfrak{J}\phi(p)$ is the Jacobian of ϕ at point p.

The Riemannian metric g is preserved under isometries, indicating that geometric quantities, such as path lengths, volumes, inner products, etc., are preserved. We can then introduce the *pushforward metric*, under the lens of isometry, to ensure the preservation of geometric quantities under a smooth embedding ϕ.

Definition 2.5 (Pushforward Riemannian metric)**.** Let $\phi : \mathcal{M} \to \mathcal{N}$ be a smooth embedding from the Riemannian manifold $(\mathcal{M}, g)$ to another manifold $\mathcal{N} = \phi(\mathcal{M})$ and $\varphi \coloneqq \phi^{-1}$ be the inverse of ϕ. Then the pushforward metric $g_{*\phi}$ is defined by

$$\begin{gathered} \langle u, v \rangle_{g_{*\phi}(\phi(p))} = \langle \mathfrak{J}\varphi(p)u, \mathfrak{J}\varphi(p)v \rangle_{g(p)}, \\ \text{for all } u, v \in \mathcal{T}_{\phi(p)}\mathcal{N}. \end{gathered} \tag{2.1}$$

With the notion of the Riemannian metric, we can finally introduce the Laplace-Beltrami operator Δ_0. To gain intuition, we first visit the Laplace operator $\Delta_0^{\mathbb{R}_d}$ on a d-dimensional Euclidean space.

Definition 2.6 (Laplace operator)**.** The Laplace operator $\Delta_0^{\mathbb{R}^d}$ in a d-dimensional Euclidean space $\mathbb{R}^d$ (with the local coordinate being $(s_1, \cdots, s_d)$) is a second-order differential operator $\Delta_0^{\mathbb{R}^d} : C^k(\mathbb{R}^d) \to C^{k-2}(\mathbb{R}^d)$ acting on an at least twice-differentiable and real-valued scalar field $f \in C^2$; it is defined as the divergence of the gradient of f, i.e.,

$$\Delta_0^{\mathbb{R}^d} f = \nabla \cdot \nabla f = \sum_{i=1}^{d} \frac{\partial^2 f}{\partial s_i^2}.$$

The Laplace-Beltrami operator is the generalization of the Laplace operator to manifolds; it can be defined on the local coordinate chart using the Riemannian metric tensor g.

Definition 2.7 (Laplace-Beltrami operator)**.** The Laplace-Beltrami operator is the generalization of the Laplace operator defined on the local coordinate chart of $\mathcal{M}$; specifically, the Laplace-Beltrami [100] operator $\Delta_0^{\mathcal{M}} : C^k(\mathcal{M}) \to C^{k-2}(\mathcal{M})$ for a scalar field $f \in C^2(M)$ on the manifold is

$$\Delta_0^{\mathcal{M}} f = \nabla \cdot \nabla f = \frac{1}{\sqrt{|g(p)|}} \sum_{i=1}^{d} \sum_{j=1}^{d} \frac{\partial}{\partial s_i} \left(\sqrt{|g(p)|} g_{ij}(p) \frac{\partial f}{\partial s_j} \right).$$

When the metric tensor is an identity metric, i.e., $g_{ij} = \delta_{ij}$, we have

$$\Delta_0^{\mathcal{M}} f = \sum_{i=1}^{d} \frac{\partial^2 f}{\partial s_i^2}$$

on the local coordinate $(s_1, \cdots, s_d)$. We oftentimes drop the superscript $\mathcal{M}$ of the Laplace-Beltrami operator, i.e., writing it as Δ_0, for simplicity.

Lastly, the ij-th entry of the *dual metric* $h = g^{-1}(p)$, the inverse of the Riemannian metric at point p, can be obtained by

$$h_{ij} = \left[\frac{1}{2} \Delta_0 (s_i - s_i(p))(s_j - s_j(p)) \right]_{s_i = s_i(p); s_j = s_j(p)}. \tag{2.2}$$

(2.2) will be discussed again in Section 2.3.4 for estimating the discrete dual metric of the pushforward Riemannian metric [95].

2.2.2 The higher-order geometry by exterior algebra

The higher-order k-Laplacian operator Δ_k elegantly generalizes the Laplace-Beltrami operator Δ_0 from the scalar field to higher-order relations (such as a vector field) on a manifold

$\mathcal{M}$. Besides, the null space of this operator, which is isomorphic to the k-th homology group, has a strong connection to the topological structures in the manifold.

To formally define the k-Laplacian Δ_k and the k-th homology group, we will introduce the concepts of k-covector, wedge product, differential form, Hodge star, exterior derivative, codifferential, and finally k-Laplacian. Notions introduced in this section are the main building blocks in Chapters 4–6.

Definition 2.8 ((Alternating) covariant k-tensors)**.** Given a real vector space V with its dual space being V^*, a *covariant k-tensor* on V is an element in the k-fold tensor product space $T^k(V^*) = V^* \otimes \cdots \otimes V^*$.

A covariant k-tensor η is *alternating* if the following condition holds:

$$\eta(v_1, \cdots, v_i, \cdots, v_j, \cdots, v_k) = -\eta(v_1, \cdots, v_j, \cdots, v_i, \cdots, v_k)$$

for all vectors $v_1, \cdots, v_k \in V$ and all $i \neq j$.

Alternating covariant k-tensors are also called k-covectors; the subspace spanned by all k-covectors is denoted $\bigwedge^k(V^*) \subseteq T^k(V^*)$. By definition, a k-covector $\eta \in \bigwedge^k(V^*)$ is anti-symmetric with respect to swapping any two arguments.

Because $\bigwedge^k(V^*) \subseteq T^k(V^*)$, one can define a mapping, called *alternation*, to project any covariant k-tensor to the space of alternating covariant k-tensors.

Definition 2.9 (Alternation)**.** An *alternation* $\mathrm{Alt} : T^k(V^*) \to \bigwedge^k(V^*)$ is a projection of the space of covariant k-tensors to the space of k-covectors; it is defined as follows.

$$\mathrm{Alt}(\eta)(v_1, \cdots, v_k) = \frac{1}{k!} \sum_{\sigma \in S_k} \mathrm{sign}(\sigma) \eta(v_{\sigma_1}, \cdots, v_{\sigma_k}).$$

Definition 2.10 (Wedge product)**.** Given a k-covector $\eta_k \in \bigwedge^k(V^*)$ and an ℓ-covector $\eta_\ell \in \bigwedge^\ell(V^*)$, the *wedge product* (or *exterior product*) of η_k and η_ℓ is defined to be the

following $(k+\ell)$-covector

$$\eta_1 \wedge \eta_2 = \frac{(k+\ell)!}{k!\ell!}\mathrm{Alt}(\eta_1 \otimes \eta_2).$$

Remark. One can also define the k-covector space $\bigwedge^k$ using the wedge product. Specifically, the *k-th exterior power* of a vector space V^* (denoted by $\bigwedge^k(V^*)$) is a space spanned by elements with the expression:

$$v_1 \wedge \cdots \wedge v_k,$$
$$\text{with } v_i \in V^* \text{ for all } i = 1, \cdots, k.$$

With the notion of alternating covariant k-tensor, wedge product, and exterior power, one can define the k-differential form on the local coordinate chart of the manifold.

Definition 2.11 (k-differential forms)**.** For every point $p \in \mathcal{M}$, let $\mathcal{T}_p^*\mathcal{M}$ be the cotangent space (the dual space of $\mathcal{T}_p\mathcal{M}$), a k-form ζ defines an element in the space of alternating covariant k-tensors on $\mathcal{T}_p^*\mathcal{M}$, i.e.,

$$\zeta \in \bigwedge^k(\mathcal{T}_p^*\mathcal{M}) = \Omega^k(\mathcal{M}).$$

We denote the space of k-differential forms by $\Omega^k(\mathcal{M})$.

On the local chart with coordinate being $(s_1, \cdots, s_d)$ and i-th basis vector being $\mathsf{d}s_i$, a k-form can be expressed as the wedge product of k distinct $\mathsf{d}s_i$'s, i.e,

$$\zeta_k = \sum_{I \in \binom{[d]}{k}} f_I \mathsf{d}s_{i_1} \wedge \cdots \wedge \mathsf{d}s_{i_k}, \tag{2.3}$$

where $\binom{[d]}{k}$ represents the collection of all possible subsets of $[d] = \{1, \cdots, d\}$ with cardinality being k, i.e., $\binom{[d]}{k} = \{S \subseteq [d] : |S| = k\}$.

A 0-form is a scalar field on $\mathcal{M}$; a 1-form $\zeta_1 \in \Omega^1(\mathcal{M})$ can be written as $\zeta = \sum_{i=1}^d f_i \mathsf{d}s_i$ with (2.3). Since $\mathsf{d}s_i$ is a basis vector on local chart, one can view ζ_1 as a vector field on the

manifold $\mathcal{M}$[1].

Next, we are interested in finding the "complement" of a k-form. Specifically, one can define an operator, called the *Hodge star*, that maps a k-form to a $(d-k)$-form on $\mathcal{M}$.

Definition 2.12 (Hodge star)**.** Let $I = \{i_1, \cdots, i_k\}$ be an increasing set of indices (i.e., $i_1 < i_2 < \cdots < i_k$), $J = \{j_1, \cdots, j_{d-k}\}$ be the (ordered) complement of I, and ϵ_{IJ} be the Levi-Civita symbol of the permutation $IJ = \{i_1, \cdots, i_k, j_1, \cdots, j_{d-k}\}$. Then the *Hodge star* operator $\star$ of a k-differential form described by I on the local coordinate chart is defined as follows:

$$\star(\mathsf{d}s_{i_1} \wedge \cdots \wedge \mathsf{d}s_{i_k}) = \epsilon_{IJ}\mathsf{d}s_{j_1} \wedge \cdots \wedge \mathsf{d}s_{j_{d-k}}.$$

For example, for in 7-dimensional space, we can map a 2-form ζ_2 described by $I = \{1, 5\}$ to a 5-form $\star\zeta_2$ described by $J = \{2, 3, 4, 6, 7\}$; the corresponding 5-form is called the *Hodge dual* of ζ_2. Note that the Hodge star is a one-to-one mapping; the ranks of these two spaces $\Omega^k(\mathcal{M})$ and $\Omega^{d-k}(\mathcal{M})$ are the same, i.e., $\dim(\Omega^k(\mathcal{M})) = \dim(\Omega^{d-k}(\mathcal{M})) = \binom{d}{k}$.

Below we are interested in generalizing the "gradient" operator, which maps a scalar field (0-form) to a vector field (1-form), to higher-order scenarios; specifically, we define *exterior derivative*, which maps a k-form to $(k+1)$-form. The inverse of this operator, called the *co-differential*, can also be defined; the core component of this operator is the Hodge star $\star$. Namely, the Hodge dual of an increased order (e.g., $(d-k-1) \to (d-k)$) operation is essentially a decrease in the order of the differential form (e.g., $(k+1) \to k$).

Definition 2.13 (Exterior derivative and co-differential)**.** The *exterior derivative* $\mathsf{d}_k : \Omega^k(\mathcal{M}) \to \Omega^{k+1}(\mathcal{M})$ operator maps a k-form $\zeta_k = f ds_{i_1} \wedge \cdots \wedge ds_{i_k}$ to a $(k+1)$-form by

$$\mathsf{d}_k\zeta_k = \sum_{j=1}^{d} \frac{\partial f}{\partial s_i}\mathsf{d}s_j \wedge (\mathsf{d}s_{i_1} \wedge \cdots \wedge \mathsf{d}s_{i_k}).$$

---

[1] A 1-form (covector) lies on the space of $\Omega^1(\mathcal{M})$; by contrast, a vector field lives in the dual space of the 1-form space. In this thesis, we use 1-form and vector fields interchangeably despite their slight differences in their definitions. Readers are encouraged to refer to Lee [71] for a detailed discussion.

The *co-differential operator* $\delta_k : \Omega^k(\mathcal{M}) \to \Omega^{k-1}(\mathcal{M})$, which maps a k-form to a $(k-1)$-form, is defined to be

$$\delta_k = (-1)^{d(k-1)+1} \star \mathsf{d}_{d-k} \star .$$

Finally, we can define the k-Hodge Laplacian using the exterior derivative d and the co-differential δ operators.

Definition 2.14 (k-Hodge Laplacian)**.** The k-Hodge Laplacian $\Delta_k : \Omega^k(\mathcal{M}) \to \Omega^k(\mathcal{M})$, which maps a k-differential form on the cotangent bundle $\mathcal{T}^*\mathcal{M}$ to another k-form, is defined to be

$$\Delta_k = \underbrace{\mathsf{d}_{k-1}\delta_k}_{\Delta_k^{\text{down}}} + \underbrace{\delta_{k+1}\mathsf{d}_k}_{\Delta_k^{\text{up}}}.$$

The first term Δ_k^{down} is the *down* k-Laplacian, while the second term Δ_k^{up} represents the *up* k-Laplacian. For $k = 0$, the *down* Laplacian term disappears; additionally, d_0 is the *gradient* ∇ and δ_1 is the *divergence* $\nabla\cdot$. Therefore, one can verify that $\Delta_0 = \delta_1\mathsf{d}_0 = \nabla \cdot \nabla$ is indeed the *Laplace-Beltrami* operator as defined in Definition 2.7. In three-dimensional Euclidean space $\mathbb{R}^3$, the *curl* operator is $\nabla\times = \star\mathsf{d}_1$ and $\delta_2 = - \star \mathsf{d}_1\star$; hence, $\Delta_1 = \nabla\nabla\cdot - (\star\mathsf{d}_1)(\star\mathsf{d}_1) = \nabla\nabla\cdot - \nabla\times\nabla\times$ is the well-known *vector Laplacian*. Finally, as we will show in Corollary 2.23, if the metric tensor g is an identity metric on $p \in \mathcal{M}$, then the 1-Laplacian, also called the *Helmholtzian*, applied on a 1-form $\zeta_1 = \sum_{i=1}^d f_i \mathsf{d}s_i$ is the coordinate-wise 0-Laplacian, i.e., $\Delta_1\zeta_1 = -\sum_{i=1}^d \left(\sum_{j=1}^d \frac{\partial^2 f_i}{\partial s_j^2}\right) \mathsf{d}s_i$.

Definition 2.15 (Harmonic vector space)**.** The harmonic vector space $\mathcal{H}_k \subseteq \Omega^k(\mathcal{M})$ is a subspace of the k-differential form defined as the null of the k-Laplacian, i.e.,

$$\mathcal{H}_k = \{\zeta \in \Omega^k(\mathcal{M}) : \Delta_k\zeta = 0\}.$$

The dimension of this subspace is called the k-th Betti number $\beta_k = \dim(\mathcal{H}_k)$.

The k-th Betti number counts the number of k-dimensional *holes* in the manifold $\mathcal{M}$; for instance, the 0-, 1-, and 2-dimensional *holes* are connected components (clusters), loops, and cavities, respectively. The harmonic space $\mathcal{H}_k$ is isomorphic to the homology group[2] $H_k(\mathcal{M}, \mathbb{R}) \coloneqq \ker(\mathsf{d}_k)/\mathrm{im}(\mathsf{d}_{k-1})$ by $\mathcal{H}_k \cong H_k(\mathcal{M}, \mathbb{R})$; namely, each homology class on a compact Riemannian manifold $\mathcal{M}$ contains a unique harmonic representative [127]. With a slight abuse of notation, we use $\mathcal{H}_k$ for denoting both the harmonic vector space and the homology group.

With the harmonic vector space defined, we can finally introduce the core motivator of Chapters 4 and 5: the Helmholtz-Hodge Decomposition (HHD).

Theorem 2.16 (Helmholtz-Hodge Decomposition [16])**.** *The space of the k-differential form $\Omega^k(\mathcal{M})$ on $\mathcal{T}^*\mathcal{M}$ can be expressed as the directed sum of three different subspaces: the image of Δ_k^{down}, the image of Δ_k^{up}, and the kernel of both, i.e.,*

$$\Omega^k(\mathcal{M}) = \mathrm{im}(\mathsf{d}_{k-1}) \oplus \mathrm{im}(\delta_{k+1}) \oplus \mathcal{H}_k. \tag{2.4}$$

Definition 2.17 ((Co-)closed and (co-)exact differential form)**.** A k-differential form ζ is called

- *closed* if $\mathsf{d}\zeta = 0$;
- *exact* if $\zeta = \mathsf{d}\eta$ for some $\eta \in \Omega^{k-1}(\mathcal{M})$;
- *co-closed* if $\delta\zeta = 0$; and
- *co-exact* if $\zeta = \delta\eta$ for some $\eta \in \Omega^{k+1}(\mathcal{M})$.

[2]More precisely, it is the *de Rham cohomology group*. In this thesis, we use homology and cohomology interchangeably. Readers are encouraged to refer to Armstrong [4] and Lee [70] for more rigorous definitions for these two terms.

Elements in the first term $\mathrm{im}(\mathsf{d}_{k-1})$ of (2.4) is called the *closed* and *exact* k-forms, while the second subspace $\mathrm{im}(\delta_{k+1})$ contains the *co-closed* and *co-exact* k-forms. Differential forms in the last subspace $\mathcal{H}_k$ are the *closed* but *not exact* (or the *co-closed* but *not co-exact*) differential forms; they are also called the harmonic forms for short. For $k = 1$, the first three terms of (2.4) are called respectively the *gradient* (conservative), the *curl*, and the *harmonic* vector fields.

2.3 The discrete Laplacians, manifold learning, and the estimation of Riemannian metric

In this section, we cover the estimation of the continuous operators discussed in Section 2.2. Specifically, we will introduce the discrete estimators of the Laplace-Beltrami operator Δ_0 and the dual pushforward Riemannian metric.

2.3.1 Manifold learning (ML) algorithms

Suppose we observe data $\mathbf{X} \in \mathbb{R}^{n\times D}$ (called a *point cloud*), with points denoted by $\mathbf{x}_i \in \mathbb{R}^D$ for $i = 1, \cdots, n$ that are sampled from a *smooth* d-dimensional manifold $\mathcal{M} \subseteq \mathbb{R}^D$ with density $p(\mathbf{x})$. The classical Manifold Learning (ML) algorithms map $\mathbf{x}_i$ for all i to $\mathbf{y}_i = \phi(\mathbf{x}_i) \in \mathbb{R}^s$, where $d \leq s \ll D$, thus reducing the dimension of the data $\mathbf{X}$ while preserving (some of) its properties.

ML algorithms, which aim at estimating the non-linear subspace (manifold) from the dataset, is a generalization of the linear dimensionality reductions method such as *principal component analysis* (PCA) and *multi-dimensional scaling* (MDS) [19]. The linear subspace of PCA is obtained by the eigenvector of the covariance matrix $\mathbf{X}^\top\mathbf{X}$; dimensionality is reduced by discarding the subspaces (called components) having small variance. MDS, on the other hand, is performed by solving the eigenproblem of the (doubled) centered squared-Euclidean

distance matrix; it is equivalent to minimize the stress defined as

$$\mathbf{Y} = \underset{\mathbf{y}'_1,\cdots,\mathbf{y}'_n \in \mathbb{R}^s}{\operatorname{argmin}} \operatorname{Stress}(\mathbf{Y}';\mathbf{X}) = \underset{\mathbf{y}'_1,\cdots,\mathbf{y}'_n \in \mathbb{R}^s}{\operatorname{argmin}} \left(\sum_{i \neq j} \left(\|\mathbf{x}_i - \mathbf{x}_j\| - \|\mathbf{y}'_i - \mathbf{y}'_j\| \right)^2 \right)^{1/2}.$$

A non-exhaustive list of the classical ML algorithms is summarized as follows.

- Isomap [118]: this algorithm is a generalization of the linear MDS. Specifically, it is done by first estimating the geodesic distances of each pair (i, j) of points with $i \neq j$ estimated by the k-nearest neighbor graph (introduced below) using Dijkstra. The embedding of points is obtained by MDS using the estimated geodesic distances.

- Laplacian Eigenmaps (LE) [9] and Diffusion Maps (DM) [32]: these two algorithms aim to estimate the Laplace-Beltrami operator Δ_0 using the *random-walk Laplacian* and the *Diffusion Maps Laplacian* constructed from the δ-radius graph, respectively; these concepts will be introduced in this section later. For both methods, the s-dimensional embedding is computed from the second to the $(s+1)$-th smallest eigenvalues of the corresponding Laplacian. LE is well suited for points sampled uniformly from a manifold, while DM is an extension of LE that can remove the bias caused by a non-uniform sampling density.

- Local Tangent Space Alignment (LTSA) [131]: as its name suggests, LTSA finds the local embedding by aligning the tangent space estimated locally at each point. Namely, the local tangent space of point $p \in \mathcal{M}$ is estimated by s principal components of the k-nearest neighbor of p. It then computes the alignment matrix from the local tangent subspace of each point. Finally, the s-dimensional embedding is obtained from the s eigenvectors corresponding to second to the $(s+1)$-th smallest eigenvalues of the alignment matrix.

- Hessian Locally Linear Embedding (HLLE) [42]: similar to LTSA, LE, and DM, this algorithm starts with a k-nearest neighbor graph to estimate the Hessian locally. The

embedding is obtained from a symmetric matrix computed by the quadratic form of the local Hessian matrices. Loosely speaking, one can view the HLLE as performing a local quadratic regression in p's neighborhood for each $p \in \mathcal{M}$.

- tSNE [124] and UMAP [76]: these two algorithms are loss-based embedding algorithms with their embedding obtained by solving the corresponding loss function using *stochastic gradient descent*, i.e., $\mathbf{Y} = \text{argmin}_{\mathbf{Y}' \in \mathbb{R}^{n \times s}} \text{loss}(\mathbf{Y}'; \mathbf{X})$. The loss function for tSNE is the Kullback-Leibler (KL) divergence between two transition distributions between pairs of points, which are the distribution in the original space $(\mathbf{x}_i, \mathbf{x}_j)$ using a Gaussian distribution and that in the embedding space $(\mathbf{y}'_i, \mathbf{y}'_j)$ using a student-t distribution, respectively. On the other hand, UMAP tries to minimize the cross-entropy between the fuzzy simplicial sets built from the k-nearest graph of $\mathbf{X}$ and that constructed from the embedding $\mathbf{Y}'$. Due to the non-convexity of these two loss functions, the obtained embedding $\mathbf{Y}$ oftentimes depends on the initial guesses. In practice, tSNE is usually initialized with random embedding, while UMAP takes the output of LE/DM as its initialization.

LE, DM, LTSA, and HLLE fall under the umbrella of the *local, spectral* methods [119]; embeddings obtained from these methods are the eigenvectors of some matrices constructed from the local *neighborhood graph*. Asymptotically speaking, each of these algorithms aims to estimate a different second-order differential operator [119], e.g., the DM and LE are discrete estimators for the Laplace-Beltrami operator. Due to the close relationship between DM to Δ_0, we will focus on this algorithm as well as their higher-order extensions in this thesis.

2.3.2 Diffusion Maps and intrinsic geometry

Neighborhood graph. The first two steps of DM [32, 85] algorithms are generic: they are performed by most ML algorithms. First, we encode the neighborhood relations in a *neighborhood graph*, which is an undirected graph $G(V, E)$ with vertex set V being the

collection of all points $i \in [n]$ and edge set E being the collections of tuples $(i,j) \in V^2$ such that i is j's neighbor (and vice versa). Common methods for building neighborhood graphs include the δ-radius graph, the k-nearest neighbor (k-NN) graph, and the continuous k-NN (δ-CkNN) graph [15]. Define $\rho_k(\mathbf{x})$ the Euclidean distance from $\mathbf{x}$ to its k-nearest neighbor, then the edge sets of these graphs are defined as

$$\textit{δ-radius graph:} \quad E = \{(i,j) \in V^2 : \|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \delta\}; \tag{2.5}$$

$$\textit{k-NN graph:} \quad E = \{(i,j) \in V^2 : \|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \max\left(\rho_k(\mathbf{x}_i), \rho_k(\mathbf{x}_j)\right)\}; \text{ and} \tag{2.6}$$

$$\textit{δ-CkNN graph:} \quad E = \left\{(i,j) \in V^2 : \frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{\sqrt{\rho_k(\mathbf{x}_i)\rho_k(\mathbf{x}_j)}} \leq \delta\right\}. \tag{2.7}$$

Readers are encouraged to refer to Berry and Sauer [15], Hein et al. [56], and Ting et al. [120] for details.

Kernel matrix. Closely related to the neighborhood graph is the *kernel matrix* $\mathbf{K} \in \mathbb{R}^{n \times n}$, whose elements are:

$$K_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{\varepsilon^2}\right) & \text{if } (i,j) \in E; \\ 0 & \text{otherwise.} \end{cases}$$

Typically, the radius δ and the *bandwidth* parameter ε are related by $\delta = c\varepsilon$ with c a small constant greater than 1, e.g., $c \in [3, 10]$; this ensures that $\mathbf{K}$ is close to its limit when $\delta \to \infty$ while remaining sparse, with sparsity structure induced by the neighborhood graph. A special case for the kernel matrix is called *adjacency matrix*, which is a binary matrix with $K_{ij} = 1$ if edge $(i,j) \in E$ and 0 otherwise. The adjacency matrix can be constructed from $\mathbf{K}$ by choosing $\varepsilon \to \infty$. Note that each non-zero element of the kernel matrix $\mathbf{K}$ corresponds to an edge $e \in E$, indicating that $\mathbf{K}$ is a weight function on edges E; this notion will be essential when extending the framework to higher-order Laplacians that encode higher-order relations.

Discrete Laplacians on neighborhood graphs. Define $\mathbf{W} = \mathrm{diag}(\mathbf{K}\mathbf{1}_n) \in \mathbb{R}^{n\times n}$ be an $n \times n$ diagonal matrix describing the degree on each vertex; additionally, let $\tilde{\mathbf{K}} = \mathbf{W}^{-1}\mathbf{K}\mathbf{W}^{-1}$ be the renormalized kernel matrix and $\tilde{\mathbf{W}} = \mathrm{diag}(\tilde{\mathbf{K}}\mathbf{1}_n)$ be the corresponding degree. One can construct several Laplacian matrices [30, 87] listed as follows:

$$\text{*Unnormalized Laplacian*:} \qquad \mathbf{L}_0 = \mathbf{W} - \mathbf{K}; \tag{2.8}$$

$$\text{*Random-walk Laplacian*:} \qquad \boldsymbol{\mathcal{L}}_0 = \mathbf{I}_n - \mathbf{W}^{-1}\mathbf{K}; \tag{2.9}$$

$$\text{*Symmetrized Laplacian* [8, 125]:} \qquad \boldsymbol{\mathcal{L}}_0^s = \mathbf{I}_n - \mathbf{W}^{-1/2}\mathbf{K}\mathbf{W}^{1/2}; \text{ and} \tag{2.10}$$

$$\text{*Diffusion Maps Laplacian* [32]:} \qquad \boldsymbol{\mathcal{L}}_0^{\mathrm{DM}} = \mathbf{I}_n - \tilde{\mathbf{W}}^{-1}\tilde{\mathbf{K}} = \mathbf{I}_n - \tilde{\mathbf{W}}^{-1}\mathbf{W}^{-1}\mathbf{K}\mathbf{W}^{-1}. \tag{2.11}$$

One can verify that the eigenvalues of a symmetrized Laplacian in (2.10) are identical to those of a random-walk Laplacian in (2.9). As a side note, $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ is a special kind of the random-walk Laplacian; similar symmetrization exists for $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$. The method of constructing the Diffusion Maps Laplacian $\boldsymbol{\mathcal{L}}^{\mathrm{DM}}$ as described above guarantees that if the data are sampled from a manifold $\mathcal{M}$, $\boldsymbol{\mathcal{L}}^{\mathrm{DM}}$ converges to Δ_0 [56, 120]. A summary of the construction of $\mathbf{L}$ can be found in Algorithm 2.1.

Algorithm 2.1: LAPLACIAN: construct Diffusion Maps Laplacian from $\mathbf{K}$

Input : Symmetric similarity matrix $\mathbf{K}$

1 Calculate the *degree* of node i, $[\mathbf{w}]_i = \sum_{j=1}^n K_{ij}$

2 Set $\mathbf{W} = \mathrm{diag}(\mathbf{w})$

3 Compute the renormalized kernel matrix $\tilde{\mathbf{K}} = \mathbf{W}^{-1}\mathbf{K}\mathbf{W}^{-1}$

4 Calculate the renormalized *degree* $[\tilde{\mathbf{w}}]_i \leftarrow \sum_{j=1}^n \tilde{K}_{ij}$

5 Set $\tilde{\mathbf{W}} = \mathrm{diag}(\tilde{\mathbf{w}})$

6 $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}} = \mathbf{I}_n - \tilde{\mathbf{W}}^{-1}\tilde{\mathbf{K}}$

Return : Diffusion Maps Laplacian $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$

Diffusion Maps embedding. The last step of the LE/DM algorithms embeds the data by solving the minimum eigen-problem of $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$. The matrix $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ is positive semi-definite

(PSD) and satisfies $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}\mathbf{1} = 0$. Hence, the eigenvector ϕ_0 of $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ is always constant and is discarded[3]. The desired m dimensional embedding coordinates are obtained from the second to $(m+1)$-th principal eigenvectors of $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$, with $0 = \lambda_0 < \lambda_1 \leq \ldots \leq \lambda_m$, i.e., $\mathbf{y}_i = (\phi_1(\mathbf{x}_i), \ldots \phi_m(\mathbf{x}_i))$. The whole procedure for obtaining the DM embedding from high-dimensional data $\mathbf{X}$ sampled from a manifold $\mathcal{M}$ is summarized in Algorithm 2.2.

Algorithm 2.2: DIFFUSIONMAPS: Diffusion Maps embedding for points sampled from a manifold

Input : Data matrix $\mathbf{X} \in \mathbb{R}^{n\times D}$, bandwidth ε, embedding dimension m

1 Compute similarity matrix $\mathbf{K}$ with $K_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\varepsilon^2}\right) & \text{if } \|x-y\| \leq 3\varepsilon; \\ 0 & \text{otherwise.} \end{cases}$

▷ `Here we set` $\delta = 3\varepsilon$

2 $\mathbf{L} \leftarrow$ LAPLACIAN($\mathbf{K}$) $\in \mathbb{R}^{n\times n}$ (Algorithm 2.1)

3 Compute $(m+1)$-smallest eigenvecors of $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ $[\boldsymbol{\phi}_0, \cdots, \boldsymbol{\phi}_m] \in \mathbb{R}^{n\times(m+1)}$

4 The *embedding coordinates* of $\mathbf{x}_i$ is $\mathbf{y}_i = [\boldsymbol{\phi}_{1,i}, \cdots, \boldsymbol{\phi}_{m,i}] \in \mathbb{R}^m$

Return : The embedding of $\mathbf{X}$: $\mathbf{Y} = [\mathbf{y}_0, \cdots, \mathbf{y}_n]^\top \in \mathbb{R}^{n\times m}$

To analyze ML algorithms, it is useful to consider the limit of the mapping $\phi : \mathbb{R}^D \to \mathbb{R}^d$, which maps $\mathbf{x}_i$ to $\mathbf{y}_i$, when the data is the entire manifold $\mathcal{M}$. We denote this limit also by ϕ, and its image by $\phi(\mathcal{M}) \in \mathbb{R}^m$. For standard algorithms such as LE/DM, it is known that this limit exists [9, 32, 55, 56, 120]. One of the fundamental requirements of ML is to preserve the neighborhood relations in the original data. In mathematical terms, we require that $\phi : \mathcal{M} \to \phi(\mathcal{M})$ is a *smooth embedding*, i.e., that ϕ is a smooth function (i.e., does not break existing neighborhood relations) whose Jacobian $\mathfrak{J}\phi(\mathbf{x}) \in \mathbb{R}^{d\times D}$ is full rank d at each $\mathbf{x} \in \mathcal{M}$ (i.e., does not create new neighborhood relations).

[3]It is the case when the manifold $\mathcal{M}$ is connected, which is usually true for the scenario of manifold learning; for disconnected manifolds, please refer to the discussions in spectral clustering [79, 87, 125].

2.3.3 Convergences of the Laplacians

Two types of convergence are discussed in this thesis: *pointwise* and *spectral* convergence. We illustrate these two convergence results using the (discrete) random-walk Laplacian $\boldsymbol{\mathcal{L}}_0 = \mathbf{W}^{-1}\mathbf{L}_0$ (in (2.9)) as an *unbiased* estimator of the Laplace-Beltrami operator Δ_0; similar construction works for $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ since it is a special case of *random-walk* Laplacian.

Denote by μ the Riemannian measure corresponding to the metric induced on $\mathcal{M}$. Let $f \in C^3(\mathcal{M})$ be a scalar field on $\mathcal{M}$ with $\mathbf{f} = [f(\mathbf{x}_1), \cdots, f(\mathbf{x}_n)]^\top \in \mathbb{R}^n$ and $\psi(x)$ be the sampling density supported on $\mathcal{M}$. Under the formulation of Berry and Sauer [15], the pointwise and spectral convergences of the (*unbiased*) *random-walk* Laplacian to the Laplace-Beltrami operator Δ_0 are respectively

$$\textit{Pointwise convergence}: \quad \mathbb{E}\left[(\mathbf{W}^{-1}\mathbf{L}_0\mathbf{f})_i\right] \xrightarrow{n\to\infty} c \cdot \Delta_0 f(x_i) + \mathcal{O}(\delta^2); \text{ and} \tag{2.12}$$

$$\textit{Spectral convergence}: \quad \mathbb{E}\left[\frac{\mathbf{f}^\top\mathbf{L}_0\mathbf{f}}{\mathbf{f}^\top\mathbf{W}\mathbf{f}}\right] \xrightarrow{n\to\infty} c \cdot \frac{\int_{p\in\mathcal{M}} f(p)(\Delta_0 f)(p)\psi(p)\mathsf{d}\mu(p)}{\int_{p\in\mathcal{M}} f^2(p)\psi(p)\mathsf{d}\mu(p)} + \mathcal{O}(\delta^2), \tag{2.13}$$

with some constant c. If the estimator is *biased*, the RHS of (2.12)–(2.13) will take the form of a Δ_0 rescaled by some other functions on $\mathcal{M}$; for instance, functions depend on the density $\psi(x)$ or the intrinsic dimension d.

Belkin and Niyogi [8] provided a pointwise convergence of $\boldsymbol{\mathcal{L}}_0$ constructed from a δ-radius graph when the sampling density $\psi(x)$ is constant. Coifman and Lafon [32] studied the pointwise convergence of a δ-radius graph with non-uniform sampling density $\psi(x)$ by proposing a renormalized version of the random-walk Laplacian, i.e., the Diffusion Maps Laplacian $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$. Ting et al. [120] generalized the work of Belkin and Niyogi [8] and Coifman and Lafon [32] to a generalized kernel setting; they show that that the corresponding $\boldsymbol{\mathcal{L}}_0$ of the k-nearest neighbor graph is biased with the sampling density and intrinsic dimension, i.e., the limit is a Δ_0 rescaled with $q^{2/d}(x)$. On the spectral convergence side, Belkin and Niyogi [9] and **?**] studied the convergence of $\boldsymbol{\mathcal{L}}_0$ to Δ_0 under the δ-radius graph setting. Readers

are encouraged to see also Hein et al. [55, 56] for additional discussions in the pointwise consistency of the Laplacians as well as Berry and Harlim [14], Berry and Sauer [15], Trillos and Slepčev [121], Trillos et al. [122] for more details in the spectral convergence of $\boldsymbol{\mathcal{L}}_0$.

2.3.4 Estimation of the pushforward Riemannian metric

Given an embedding $\phi : \mathbb{R}^D \to \mathbb{R}^s$ that maps a d-dimensional manifold $\mathcal{M} \subseteq \mathbb{R}^D$ to $\phi(\mathcal{M}) \subseteq \mathbb{R}^s$. Additionally, we assume that the metric g induced from $\mathbb{R}^D$; specifically, it is an identity metric on the local coordinate chart for each point. For every $\mathbf{y}_i = \phi(\mathbf{x}_i)$, the associated pushforward Riemannian metric $g_{*\phi}$ expressed in the coordinate of $\mathbb{R}^s$ is a symmetric and PSD $s \times s$ matrix $\mathbf{G}(\mathbf{y}_i)$ of rank d. The scalar product $\langle \mathbf{u}, \mathbf{v} \rangle_{g_{*\phi}(\mathbf{y}_i)}$ for $u, v \in \mathcal{T}_{\mathbf{y}} \phi(\mathcal{M})$ takes the form $\mathbf{u}^\top \mathbf{G}(\mathbf{y}_i)\mathbf{v}$. Given an embedding $\mathbf{Y} = \phi(\mathbf{X})$, $\mathbf{G}(\mathbf{y}_i)$ can be estimated by (2.2) [95]; specifically, the estimated dual metric $\mathbf{H}(\mathbf{y}_i)$ with $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$ is

$$[\mathbf{H}(\mathbf{y}_i)]_{k,l} = \frac{1}{2} \cdot \sum_{j \neq i} [\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}]_{ij} (y_{jl} - y_{il})(y_{jk} - y_{ik}).$$

An additional step of reduced rank *singular value decomposition* (SVD) is performed to ensure the rank of $\mathbf{H}$ is $\mathrm{rank}(\mathbf{H}) = d$. The procedure to estimate the dual metric is summarized in Algorithm 2.3. The RMetric algorithm also returns the *pushforward metric* $\mathbf{G}(\mathbf{y}_i)$, which is the pseudo-inverse of the metric $\mathbf{H}(\mathbf{y}_i)$, and its SVD $\boldsymbol{\Sigma}(\mathbf{y}_i), \mathbf{U}(\mathbf{y}_i) \in \mathbb{R}^{m \times d}$. The latter represents an orthogonal basis of $\mathcal{T}_{\phi(\mathbf{x})}(\phi(\mathcal{M}))$. Throughout this thesis, the i in $\mathbf{G}(i)$, $\mathbf{H}(i)$, $\boldsymbol{\Sigma}(i)$, and $\mathbf{U}(i)$ represents $\mathbf{y}_i$ for notational simplicity.

2.4 The discrete k-Laplacian and the Helmholtz-Hodge Decomposition

In this section, we introduce the discrete counterparts of the k-differential form, the exterior derivative, the co-differential, the k-Laplacian Δ_k, and HHD discussed in Section 2.2.

Algorithm 2.3: RMetric: estimating the Riemannian metric from $\mathbf{Y}$ and $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$

Input : Embedding $\mathbf{Y} \in \mathbb{R}^{n\times m}$, Laplacian $\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}$, intrinsic dimension d

for *all* $\mathbf{y}_i \in \mathbf{Y}, k = 1 \to m, l = 1 \to m$ **do**

  $[\tilde{\mathbf{H}}(i)]_{kl} = 1/2 \cdot \sum_{j\neq i}[\boldsymbol{\mathcal{L}}_0^{\mathrm{DM}}]_{ij}(y_{jl} - y_{il})(y_{jk} - y_{ik})$

for $i = 1 \to n$ **do**

  $\mathbf{U}(i), \mathbf{\Sigma}(i) \leftarrow$ ReducedRankSVD$(\tilde{\mathbf{H}}(i), d)$

  $\mathbf{H}(i) = \mathbf{U}(i)\mathbf{\Sigma}(i)\mathbf{U}(i)^\top$

  $\mathbf{G}(i) = \mathbf{U}(i)\mathbf{\Sigma}^{-1}(i)\mathbf{U}(i)^\top$

Return : $\mathbf{G}(i), \mathbf{H}(i) \in \mathbb{R}^{m\times m}$, $\mathbf{U}(i) \in \mathbb{R}^{m\times d}$, $\mathbf{\Sigma}(i) \in \mathbb{R}^{d\times d}$, for $i \in [n]$

2.4.1 The k-Laplacian

Simplicial and cubical complex. An *abstract complex* is a natural extension of a graph designed to capture higher-order relationships between its vertices. A *simplicial k-complex* (used when the data are point clouds or networks) is a tuple $\mathrm{SC}_k = (\Sigma_0, \cdots, \Sigma_k)$, with Σ_ℓ being a set of ℓ dimensional *simplices*, such that every *face* of a simplex $\sigma \in \Sigma_\ell$ is in $\Sigma_{\ell-1}$ for $\ell \leq k$. As a side note, a graph $G = (V, E)$ is an SC_1; and $\mathrm{SC}_2 = (V, E, T)$ commonly used in edge flow learning [28, 105] is obtained by adding a set of 3-cliques (triangles) T of G. This procedure extends to defining Σ_ℓ as the set of all ℓ-cliques of G, with the resulting complex called a *clique complex* of the graph G. This complex is also known as a *Vietoris-Rips (VR) complex* if G is the δ-radius neighborhood graph used in the manifold learning literature [28, 32, 120].

The *cubical k-complex* $\mathrm{CB}_k = (K_0, \cdots, K_k)$ is a complex widely used with image data. The difference between this complex and the SC_k is that a CB_k is a collection of sets of ℓ-cubes, for $\ell < k$. Note that we write $\Sigma_0 = K_0 = V$ the vertex set and $\Sigma_1 = K_1 = E$ the edge set. $\Sigma_2 = T$ and $K_2 = R$ are the triangle and rectangle set, respectively. Additionally, we define $n_\ell = |\Sigma_\ell|$ (or $= |K_\ell|$) to be the cardinality of the ℓ-dimensional cells. For more information about building various complexes on different datasets please refer to Otter et al. [92].

k-cochain. By choosing an orientation to every k-simplex $\sigma_{k,i} \in \Sigma_k$ (or K_k), one can define a finite-dimensional vector space $\mathcal{C}_k$ (*k-cochain space*[4]). An element in the cochain space $\boldsymbol{\omega}_k = \sum_i \boldsymbol{\omega}_k(\sigma_{k,i})\sigma_{k,i} \in \mathcal{C}_k$ is called a *k-cochain*; one can further express $\boldsymbol{\omega}_k$ as $\boldsymbol{\omega}_k = (\omega_{k,1}, \cdots, \omega_{k,n_k})^\top \in \mathbb{R}^{n_k}$ by identifying each $\sigma_{k,i}$ with the standard basis vector $\mathbf{e}_i \in \mathbb{R}^{n_k}$. Functions on nodes and edge flows (functions of edges), for example, are elements of $\mathcal{C}_0$ and $\mathcal{C}_1$, respectively.

Boundary matrix. The k-th *boundary matrix* $\mathbf{B}_k$ [73] maps a k-cochain of k-cells (simplices/cubes) σ_k to the $(k-1)$-cochain of its faces, i.e., $\mathbf{B}_k : \mathcal{C}_k \to \mathcal{C}_{k-1}$. The boundary matrix $\mathbf{B}_k \in \{0, \pm 1\}^{n_{k-1} \times n_k}$ is a sparse binary matrix, with the sign of the non-zero entries σ_{k-1}, σ_k given by the orientation of σ_k w.r.t. its face σ_{k-1}. Hence, different SC or CB will induce different $\mathbf{B}_k$. For $k = 1$ on either the SC or CB, the boundary map is the *graph incidence matrix*, i.e.,

$$(\mathbf{B}_1)_{[x'],[x,y]} = \begin{cases} 1 & \text{if } x' = x; \\ -1 & \text{if } x' = y; \\ 0 & \text{otherwise.} \end{cases}$$

For $k = 2$, each column of $\mathbf{B}_2$ contains the orientation of a triangle/rectangle w.r.t. its edges. Specifically, for an SC,

$$(\mathbf{B}_2)_{[x',y'],[x,y,z]} = \begin{cases} 1 & \text{if } [x', y'] \in \{[x, y], [y, z]\}; \\ -1 & \text{if } [x', y'] = [x, z]; \\ 0 & \text{otherwise;} \end{cases}$$

[4]We use *chain* and *cochain* interchangeably for simplicity, see Lim [73] for the distinction between them.

for a CB,

$$(\mathbf{B}_2)_{[x',y'],[x,y,z,w]} = \begin{cases} 1 & \text{if } [x',y'] \in \{[x,y],[y,z],[z,w]\}; \\ -1 & \text{if } [x',y'] = [x,w]; \\ 0 & \text{otherwise.} \end{cases}$$

Simplex σ_{k+1} is a *coface* of σ_k iff σ_k is a face of σ_{k+1}; let coface(σ_k) be the set of all cofaces of σ_k. The $(k-1)$-th *coboundary matrix* $\mathbf{B}_k^\top$ (adjoint of $\mathbf{B}_k$) maps σ_{k-1}, as a $(k-1)$-cochain, to the k-cochain of coface(σ_{k-1}).

k-Laplacian. Let $\mathbf{W}_\ell$ be a diagonal non-negative *weight matrix* of dimension n_ℓ, with $[\mathbf{W}_\ell]_{\sigma,\sigma}$ representing the weight of the ℓ-simplex/cube σ and $\mathbf{w}_\ell \leftarrow \text{diag}(\mathbf{W}_\ell)$. One can define

$$\textit{Unnormalized } k\textit{-Laplacian } [46]: \qquad \mathbf{L}_k = \underbrace{\mathbf{B}_k^\top \mathbf{B}_k}_{\mathbf{L}_k^{\text{down}}} + \underbrace{\mathbf{B}_{k+1}\mathbf{B}_{k+1}^\top}_{\mathbf{L}_k^{\text{up}}}; \tag{2.14}$$

$$\textit{Random-walk } k\textit{-Laplacian } [58]: \qquad \boldsymbol{\mathcal{L}}_k = \underbrace{\mathbf{B}_k^\top \mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k}_{\boldsymbol{\mathcal{L}}_k^{\text{down}}} + \underbrace{\mathbf{W}_1^{-1}\mathbf{B}_2\mathbf{W}_2\mathbf{B}_2^\top}_{\boldsymbol{\mathcal{L}}_k^{\text{up}}}; \text{ and} \tag{2.15}$$

$$\textit{Symmetrized } k\textit{-Laplacian } [58, 105]: \qquad \boldsymbol{\mathcal{L}}_k^s = \underbrace{\mathbf{A}_k^\top \mathbf{A}_k}_{\boldsymbol{\mathcal{L}}_k^{s,\text{down}}} + \underbrace{\mathbf{A}_{k+1}\mathbf{A}_{k+1}^\top}_{\boldsymbol{\mathcal{L}}_k^{s,\text{up}}}. \tag{2.16}$$

Here $\mathbf{A}_\ell$ for $\ell = k, k+1$ is the *normalized boundary matrix*, with

$$\mathbf{A}_\ell = \mathbf{W}_{\ell-1}^{-1/2}\mathbf{B}_\ell\mathbf{W}_\ell^{1/2}.$$

The weights capture combinatorial or geometric information and must satisfy the consistency relation $\mathbf{w}_\ell(\sigma_\ell) = \sum_{\sigma_{\ell+1}\in\text{coface}(\sigma_\ell)} \mathbf{w}_{\ell+1}(\sigma_{\ell+1})$ (in matrix form: $\mathbf{w}_\ell = |\mathbf{B}_{\ell+1}|\mathbf{w}_{\ell+1}$) for $\ell = k, k-1$. To determine the weight for the $(k+1)$-simplexes, one can select $\mathbf{w}_{k+1}$ to be obtained from the coboundary matrix $\mathbf{B}_{k+1}$, to be constant [105], or based on the pairwise distance kernel [28, 32]. We will discuss the choice of triangle weights for $\boldsymbol{\mathcal{L}}_1$ in Chapter 4

so that the asymptotic limit exists.

The first and second terms of (2.14)–(2.16) are called respectively the *down* and *up* Laplacians. Note that one can verify that the kernel matrix $\mathbf{K}$ is $\mathbf{K} = \mathbf{B}_1\mathbf{W}_1\mathbf{B}_1^\top$; therefore, for $k = 0$, the down components disappear and the resulting Laplacians in (2.14)–(2.16) reduce to those in (2.8)–(2.10), respectively. For $k = 1$, the 1-Laplacian $\boldsymbol{\mathcal{L}}_1$ is also called the Helmholtzian; $\boldsymbol{\mathcal{L}}_1$ will be a critical building block throughout Chapters 4–5.

2.4.2 Hodge decomposition and the basis of vector fields on $\mathcal{M}$

The celebrated Helmholtz-Hodge decomposition (HHD) expresses a vector field as the direct sum of a *gradient*, a *harmonic*, and a *rotational* vector field. The eigenvectors of the discrete graph Helmholtzian, which span the flow space $\mathcal{C}^1$, can be grouped into bases of three different subspaces (the image of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$, the image of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$, and the kernel of both matrices). Let the symbols ker and im denote the null space and image of a matrix, respectively; one can obtain the following decomposition [105] of an edge flow using the Helmholtzian,

$$\mathcal{C}^1 \cong \mathbb{R}^{n_1} = \overbrace{\underbrace{\operatorname{im}\left(\mathbf{W}_1^{1/2}\mathbf{B}_1^\top\right)}_{\text{gradient}} \oplus \overbrace{\ker\left(\boldsymbol{\mathcal{L}}_1\right)}^{\text{harmonic}} \oplus \underbrace{\operatorname{im}\left(\mathbf{W}_1^{-1/2}\mathbf{B}_2\right)}_{\text{curl}}}^{\ker\left(\mathbf{B}_1\mathbf{W}_1^{1/2}\right)=\ker(\text{gradient})}. \qquad \overbrace{\phantom{\underbrace{\operatorname{im}\left(\mathbf{W}_1^{1/2}\mathbf{B}_1^\top\right)}_{\text{gradient}} \oplus \overbrace{\ker\left(\boldsymbol{\mathcal{L}}_1\right)}^{\text{harmonic}} \oplus \underbrace{\operatorname{im}\left(\mathbf{W}_1^{-1/2}\mathbf{B}_2\right)}_{\text{curl}}}}^{\ker\left(\mathbf{B}_2^\top\mathbf{W}_1^{-1/2}\right)=\ker(\text{curl})} \tag{2.17}$$

An edge flow $\boldsymbol{\omega} \in \mathbb{R}^{n_1}$ induces the orthogonal decomposition $\boldsymbol{\omega} = \mathbf{g} \oplus \mathbf{r} \oplus \mathbf{h}$, with $\mathbf{g} = \mathbf{W}_1^{1/2}\mathbf{B}_1^\top\mathbf{p}_0$, $\mathbf{r} = \mathbf{W}_1^{-1/2}\mathbf{B}_2\mathbf{p}_2$, and $\mathbf{h}$ being the *gradient*, the *curl*, and the *harmonic* components of $\boldsymbol{\omega}$, respectively. Here $\mathbf{p}_0$ and $\mathbf{p}_2$ are *potentials* on vertices and triangles, respectively; they are potentials in the sense that the edge flows can be obtained by a "discrete derivative" with $\mathbf{B}_0$ and $\mathbf{B}_1$. Flows $\mathbf{g}$, $\mathbf{r}$, and $\mathbf{h}$ can be estimated by solving two

least-squares problems, i.e.,

$$\hat{\mathbf{p}}_0 = \underset{\mathbf{p}_0 \in \mathbb{R}^{n_0}}{\operatorname{argmin}} \|\mathbf{W}_1^{1/2}\mathbf{B}_1^\top \mathbf{p}_0 - \boldsymbol{\omega}\|^2;$$

$$\hat{\mathbf{p}}_2 = \underset{\mathbf{p}_2 \in \mathbb{R}^{n_2}}{\operatorname{argmin}} \|\mathbf{W}_1^{-1/2}\mathbf{B}_2\mathbf{p}_2 - \boldsymbol{\omega}\|^2;$$

$$\hat{\mathbf{h}} = \boldsymbol{\omega} - \mathbf{W}_1^{1/2}\mathbf{B}_1^\top \hat{\mathbf{p}}_0 - \mathbf{W}_1^{-1/2}\mathbf{B}_2\hat{\mathbf{p}}_2.$$

Note that we exemplify HHD using Helmholtzian $\boldsymbol{\mathcal{L}}_1$, but it can be generalized to other types of Helmholtzian or the cases of $k > 1$.

2.4.3 k-th homology vector space and embedding

The homology vector space $\mathcal{H}_k$ is a subspace of k-cochain space $\mathcal{C}_k$ such that every k-cycle (expressed as a k-cochain) in $\mathcal{H}_k$ is not the boundary of any $(k+1)$-cochain. In mathematical terms, $\mathcal{H}_k \coloneqq \ker(\mathbf{A}_k)/\mathrm{im}(\mathbf{A}_{k+1})$. The rank of the subspace is called the k-th Betti number $\beta_k = \dim(\mathcal{H}_k)$, which counts the number of "loops" (*homology class*) in the SC. $\mathcal{H}_k$ is equivalent to the null space of $\boldsymbol{\mathcal{L}}_k$ [73, 105]; therefore, a basis of $\mathcal{H}_k$ can be obtained by the eigenvectors $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_{\beta_1}] \in \mathbb{R}^{n_k \times \beta_k}$ of $\boldsymbol{\mathcal{L}}_k$ with eigenvalue 0. The *homology embedding* maps a k-simplex σ_k to $\mathbf{Y}_{\sigma_k,:} = [\mathbf{y}_1(\sigma_k), \cdots \mathbf{y}_1(\sigma_k)]^\top \in \mathbb{R}^{\beta_k}$. Note that the null space of $\boldsymbol{\mathcal{L}}_k$ is only identifiable up to a unitary transformation; hence, the homology embedding might change with a different basis $\mathbf{Y}$.

The zeroth homology space $\mathcal{H}_0$, the null space of the graph Laplacian $\boldsymbol{\mathcal{L}}$, identifies the number of connected components in the data $\mathbf{X}$. $\mathcal{H}_0$ has been widely used in spectral clustering and community detection research [79, 87, 106, 125]. Specifically, one can infer the number of clusters by the number of zero eigenvalues; the community information can be extracted from the corresponding homology embedding as proposed by Meilă and Shi [79] and Ng et al. [87]. The first homology space $\mathcal{H}_1$, reveals the number of independent loops parameterizing the non-trivial topology in the manifold $\mathcal{M}$; this space will be one of our major interests in Chapters 4 and 6.

2.4.4 Connections to the continuous operators in Section 2.2.2

The k-cochains are the discrete analogues of *k-forms* [129]. For $k = 1$, the following path integral [129] (along the geodesic $\boldsymbol{\gamma}(t)$ connecting x and y, with $\boldsymbol{\gamma}(0) = \mathbf{x}$ and $\boldsymbol{\gamma}(1) = \mathbf{y}$) relates a 1-cochain $\boldsymbol{\omega}$ to a 1-form $\mathbf{v}$ (vector field):

$$\boldsymbol{\omega}([x, y]) = \int_0^1 \mathbf{v}(\boldsymbol{\gamma}(t))\boldsymbol{\gamma}'(t)\mathsf{d}t. \tag{2.18}$$

To estimate a vector field from $\boldsymbol{\omega}$, one can solve a least-squares problem [28], which is the inverse operation of the path integral (details in Section 2.5).

On the other hand, the *exterior derivative* d_k and the *co-differential* δ_k operators are the continuous counterparts of $\mathbf{B}_{k+1}^\top$ and $\mathbf{B}_k$, respectively. The k-Hodge Laplacian operators $\Delta_k = \mathsf{d}_{k-1}\delta_k + \delta_{k+1}\mathsf{d}_k$ is the continuous version of the k-Laplacians introduced in (2.14)–(2.16). The homology group $H_k(\mathcal{M}, \mathbb{R})$ is the null of Δ_k. Each of its generator corresponds to an unique harmonic k-forms ζ_k in harmonic space $\mathcal{H}_k \subseteq \Omega^k(\mathcal{M})$, computed by solving $\Delta_k\zeta_k = 0$ with proper boundary conditions. The homology generators represent the continuous version of the discrete homology basis $\mathbf{Y}$. A summary of the above connections is in Table 2.1.

Table 2.1: Comparisons of the discrete matrices defined on simplicial/cubical complexes and the continuous operators defined on manifolds.

Discrete		Continuous	
Simplicial/Cubical complex	SC_ℓ (or CB_ℓ)	Manifold	$\mathcal{M}$
k-cochain	$\boldsymbol{\omega}_k$	k-differential form	ζ_k
Boundary matrix	$\mathbf{B}_k$	Codifferential operator	δ_k
Coboundary matrix	$\mathbf{B}_k^\top$	Exterior derivative	d_{k-1}
k-Laplacian	$\boldsymbol{\mathcal{L}}_k$	Laplace-de Rham operator	Δ_k
k-th homology vector space	$\mathcal{H}_k \subset \mathbb{R}^{n_k}$	k-th homology group	$H_k(\mathcal{M}, \mathbb{R})$

2.5 Vector fields and edge flows mapping

2.5.1 Approximate edge flows from vector fields

As we pointed out earlier, a vector field $\mathbf{v}$ on $\mathcal{M}$ induces an edge flow by a path integral relation in (2.18). Given only the vector field $\mathbf{v}(\mathbf{x}_i) \in \mathbb{R}^D$ for all $i \in [n]$, the edge flow $\boldsymbol{\omega}$ on edge $e = (i, j)$ can be obtained by approximating the path integral along $\boldsymbol{\gamma}(t)$, the geodesic connecting nodes i and j. By Lemma 4.7 (will be introduced in Chapter 4), we have the following three approximations: $\boldsymbol{\gamma}(t) \approx \mathbf{x}_i + (\mathbf{x}_j - \mathbf{x}_i)t$, $\boldsymbol{\gamma}'(t) = \mathsf{d}\boldsymbol{\gamma}(t)/\mathsf{d}t \approx (\mathbf{x}_j - \mathbf{x}_i)$, and $\mathbf{v}(\boldsymbol{\gamma}(t)) \approx \mathbf{v}(\mathbf{x}_i) + (\mathbf{v}(\mathbf{x}_j) - \mathbf{v}(\mathbf{x}_i))t$. The edge flow of an edge e can therefore be simplified as

$$\begin{aligned} \omega_e &= \int_0^1 \mathbf{v}^\top(\boldsymbol{\gamma}(t))\boldsymbol{\gamma}'(t)\mathsf{d}t \approx \int_0^1 \left[\mathbf{v}(\mathbf{x}_i) + (\mathbf{v}(\mathbf{x}_j) - \mathbf{v}(\mathbf{x}_i))t\right]^\top (\mathbf{x}_j - \mathbf{x}_i)\mathsf{d}t \\ &= \frac{1}{2}(\mathbf{v}(\mathbf{x}_i) + \mathbf{v}(\mathbf{x}_j))^\top (\mathbf{x}_j - \mathbf{x}_i). \end{aligned} \tag{2.19}$$

This approximation can be rigorously defined using the Witney basis function [129]; it will become equality if $\mathcal{M}$ is a Euclidean space and the vector field $\mathbf{v}$ is a linear function along this path. Note that the approximation above can be written in a more concise form using the boundary operator $\mathbf{B}_1$. Let $\mathbf{V} \in \mathbb{R}^{n \times D}$ with $\mathbf{v}_i = \mathbf{V}_{i,:} = \mathbf{v}(\mathbf{x}_i)$, we have $[|\mathbf{B}_1^\top|\mathbf{V}]_{[i,j]} = \mathbf{v}(\mathbf{x}_i) + \mathbf{v}(\mathbf{x}_j)$. Additionally, we have $[-\mathbf{B}_1^\top \mathbf{X}]_{[i,j]} = \mathbf{x}_j - \mathbf{x}_i$. Therefore,

$$\boldsymbol{\omega} = -\frac{1}{2}\operatorname{diag}(\mathbf{B}_1^\top \mathbf{X}\mathbf{V}^\top |\mathbf{B}_1|).$$

2.5.2 Recovering a smoothed vector field from an edge flow

The procedure for estimating vertex-wise vector field from an edge flow is inspired by (2.19) above. Specifically, the ℓ-th component of the vector field $\mathbf{v}$ on points $\mathbf{x}_i, \mathbf{x}_j$ projected onto $\mathbf{x}_j - \mathbf{x}_i$ is

$$\frac{1}{2}(v_\ell^{\parallel}(\mathbf{x}_i) + v_\ell^{\parallel}(\mathbf{x}_j)) = \frac{(x_{j,\ell} - x_{i,\ell})\omega_{ij}}{\|\mathbf{x}_j - \mathbf{x}_i\|^2} = [\mathbf{Y}_E]_{ij,\ell} \cdot \omega_{ij}.$$

Here $\mathbf{Y}_E \in \mathbb{R}^{n_1 \times D}$ is the matrix $-\mathbf{B}_1^\top \mathbf{X}$ with its rows normalized to length 1. Furthermore,

the LHS of the above equation equals $\frac{1}{2}|\mathbf{B}_1^\top \mathbf{V}^{\|}|$ Given an edge flow $\boldsymbol{\omega}$, the (parallel) vector field $\hat{\mathbf{V}}$ can be estimated by solving the following D least squares problems

$$\hat{\mathbf{v}}_\ell = \underset{\mathbf{v}_\ell \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \left\| |\mathbf{B}_1^\top| \mathbf{v}_\ell - [\operatorname{diag}(\boldsymbol{\omega}) \mathbf{Y}_E]_{:,\ell} \right\|_2^2 \right\} \text{ for } \ell = 1, \cdots, D. \tag{2.20}$$

$\hat{\mathbf{V}}$ is the column concatenation of the D least squares solutions,

$$\hat{\mathbf{V}} = \begin{bmatrix} | & | & & | \\ \hat{\mathbf{v}}_1 & \hat{\mathbf{v}}_2 & \dots & \hat{\mathbf{v}}_D \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times D}.$$

The linear system above is overdetermined (i.e., n_1 is usually greater than n_0), one can obtain a smoother estimated vector field from the edge flow by adding a regularization term. Specifically, we can change the aforementioned loss function to the following,

$$\hat{\mathbf{V}} = \underset{\mathbf{V} \in \mathbb{R}^{n \times D}}{\operatorname{argmin}} \left\{ \left\| |\mathbf{B}_1^\top| \mathbf{V} - \operatorname{diag}(\boldsymbol{\omega}) \mathbf{Y}_E \right\|_F^2 + \lambda \|\mathbf{V}\|_F \right\}. \tag{2.21}$$

Here $\|\cdot\|_F$ represents the Frobenius norm. (2.21) is essentially a multi-output Ridge regression problem. We choose the parameter λ by cross-validation with the Fisher z-transformed Pearson correlation as our scoring function [28].

2.5.3 Mapping a velocity field between representations

Given a set of points $[\mathbf{x}_i]_{i=1}^n$ in $\mathbb{R}^D$ sampled from a manifold $\mathcal{M}$, vectors $\mathbf{v}_i$ in the tangent subspace of $\mathcal{M}$ at each data point, and a mapping $\phi : \mathbb{R}^D \to \mathbb{R}^d$ from the ambient space to another representation (e.g., the DM embedding in Section 2.3.1), we are interested in obtaining the vector field $\mathbf{u}_i \in \mathbb{R}^d$ of each point in the new representation space $\boldsymbol{\phi}_i = \phi(\mathbf{x}_i)$. This problem can be solved by the definition of the velocity field in the space of the mapped

representation $\phi(\mathcal{M})$, i.e.,

$$u_{ij} = \lim_{t \to 0} \frac{\phi_j(\mathbf{x}_i + \mathbf{v}_i t) - \phi_j(\mathbf{x}_i)}{t} = \left(\nabla_{\mathbf{x}} \phi_i(\mathbf{x}_i)\right)^\top \mathbf{v}_i. \tag{2.22}$$

The j-th component of the vector $\mathbf{u}_i$ is essentially the directional derivative of the mapping ϕ along with $\mathbf{v}_i$ in the original space. Let $\mathfrak{J}\phi(\mathbf{x}) \in \mathbb{R}^{d \times D}$ be the Jacobian matrix at $\mathbf{x} \in \mathcal{M}$, one can turn (2.22) to

$$\mathbf{u}_i = \mathfrak{J}\phi(\mathbf{x}_i)\mathbf{v}_i \;\text{ with }\; \mathfrak{J}\phi(\mathbf{x}_i) = \begin{bmatrix} —(\nabla_{\mathbf{x}} \phi_1(\mathbf{x}_i))^\top— \\ —(\nabla_{\mathbf{x}} \phi_2(\mathbf{x}_i))^\top— \\ \vdots \\ —(\nabla_{\mathbf{x}} \phi_d(\mathbf{x}_i))^\top— \end{bmatrix} \mathbf{v}_i. \tag{2.23}$$

The velocity field mapping problem mapping now becomes a gradient estimation problem, which can be solved using any gradient estimation methods, e.g., Luo et al. [74] and Mukherjee and Wu [84]. In this thesis, we use the gradient estimation method by Mukherjee and Wu [84], which aims to solve the (local) weighted linear regression on the local tangent plane. More specifically, the gradient of a function $f : \mathbb{R}^D \to \mathbb{R}$ at point $\mathbf{x}_i$, denoted as $\nabla_{\mathbf{x}} f(\mathbf{x}_i)$, is the minimizer of the following least-squares problem:

$$\nabla_{\mathbf{x}} f(\mathbf{x}_i) = \underset{\mathbf{g} \in \mathbb{R}^D}{\operatorname{argmin}} \sum_{j \sim i} w_{ij} \left\| (f(\mathbf{x}_j) - f(\mathbf{x}_i)) - \mathbf{g}^\top (\mathbf{x}_j - \mathbf{x}_i) \right\|_2 .$$

The w_{ij} can be estimated by the weights used in the Local PCA [24]. Note that if the target embedding is not in Euclidean space, e.g., mapping the small molecule dataset from the ambient space $\mathbf{X}$ to the torsion space as in Figures 4.3e and 4.3j, one has to use the proper boundary condition when calculating $f(\mathbf{x}_j) - f(\mathbf{x}_i)$. That is to say, the angular distance in the torsion space should be used (distance between $\pi/2$ and 2π is $\pi/2$ rather than $3\pi/2$) to get a smooth estimation of the gradients.

2.6 Appendix—additional information for exterior calculus

2.6.1 Rigorous definitions of boundary matrices

First, we define the Levi-Civita notation and permutation parity; this is useful for the definition of boundary operator $\mathbf{B}_k$.

Definition 2.18 (Permutation parity)**.** Given a finite set $\{j_0, j_1, \cdots, j_k\}$ with $k \geq 1$ and $j_\ell < j_m$ if $\ell < m$, the parity of a permutation $\varsigma(\{j_0, \cdots, j_k\}) = \{i_0, i_1, \cdots, i_k\}$ is defined to be

$$\epsilon_{i_0,\cdots,i_k} = -1^{N(\varsigma)}. \tag{2.24}$$

Here $N(\varsigma)$ is the *inversion number* of ς. The inversion number is the cardinality of the inversion set, i.e., $N(\varsigma) = \#\{(\ell, m) : i_\ell > i_m \text{ if } \ell < m\}$. We say ς is an even permutation if $\epsilon_{i_0,\cdots,i_k} = 1$ and is an odd permutation otherwise.

Remark. When $k = 1$, the Levi-Civita symbol is

$$\epsilon_{ij} = \begin{cases} +1 & \text{if } (i,j) = (1,2), \\ -1 & \text{if } (i,j) = (2,1). \end{cases}$$

For $k = 2$, the Levi-Civita symbol is

$$\epsilon_{ijk} = \begin{cases} +1 & \text{if } (i,j,k) \in \{(1,2,3),(2,3,1),(3,1,2)\}, \\ -1 & \text{if } (i,j,k) \in \{(3,2,1),(1,3,2),(2,1,3)\}. \end{cases}$$

With this in hand, one can define the boundary map as follows.

Definition 2.19 (Boundary map & boundary matrix)**.** Let $i_0 \cdots \hat{i_j} \cdots i_k \coloneqq i_0, \cdots, i_{j-1}, i_{j+1}, \cdots, i_k$, and $i_0 \cdots \check{i_j} \cdots i_k$ denote i_j insert into $i_0, \cdots, i_k$ with proper order, we define the *boundary*

map (operator) $\mathcal{B}_k : \mathcal{C}_k \to \mathcal{C}_{k-1}$, which maps a simplex to its face, by

$$\mathcal{B}_k([i_0, \cdots, i_k]) = \sum_{j=0}^{k} (-1)^j [i_0 \cdots \hat{i_j} \cdots i_k] = \sum_{j=0}^{k} \epsilon_{i_j, i_0 \cdots \hat{i_j} \cdots i_k} [i_0 \cdots \hat{i_j} \cdots i_k]. \tag{2.25}$$

Here $i_j, i_0 \cdots \hat{i_j} \cdots i_k \coloneqq i_j, i_0, \cdots, i_{j-1}, i_{j+1}, \cdots, i_k$.

The corresponding *boundary matrix* $\mathbf{B}_k \in \{0, \pm 1\}^{n_{k-1} \times n_k}$ can be defined as follows.

$$(\mathbf{B}_k)_{\sigma_{k-1}, \sigma_k} \begin{cases} \epsilon_{i_j, i_0 \cdots \hat{i_j} \cdots i_k} & \text{if } \sigma_k = [i_0, \cdots, i_k],\ \sigma_{k-1} = [i_0 \cdots \hat{i_j} \cdots i_k], \\ 0 & \text{otherwise.} \end{cases} \tag{2.26}$$

$(\mathbf{B}_k)_{\sigma_{k-1}, \sigma_k}$ represents the orientation of σ_{k-1} as a face of σ_k, or equals 0 when the two are not adjacent.

Example. For the simplicial complex SC_2 in Figure 2.1, the corresponding $\mathbf{B}_1$ is in Table 2.3 while $\mathbf{B}_2$ is in Table 2.2.

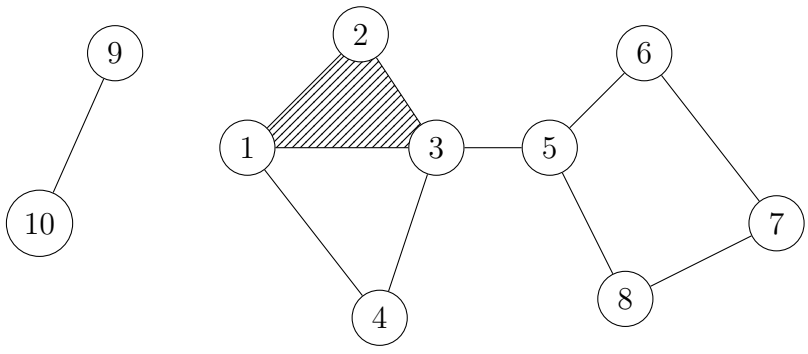

Figure 2.1: Illustration of an $\mathrm{SC}_2 = (\Sigma_0, \Sigma_1, \Sigma_2)$ with the shaded region denoting the triangle $t \in \Sigma_2$.

	(1, 2, 3)
(1, 2)	1
(1, 3)	-1
(1, 4)	0
(2, 3)	1
(3, 4)	0
(3, 5)	0
(5, 6)	0
(5, 8)	0
(6, 7)	0
(7, 8)	0

Table 2.2: $\mathbf{B}_2$ of SC_2 in Figure 2.1.

	(1, 2)	(1, 3)	(1, 4)	(2, 3)	(3, 4)	(3, 5)	(5, 6)	(5, 8)	(6, 7)	(7, 8)	(9, 10)
1	1	1	1	0	0	0	0	0	0	0	0
2	-1	0	0	1	0	0	0	0	0	0	0
3	0	-1	0	-1	1	1	0	0	0	0	0
4	0	0	-1	0	-1	0	0	0	0	0	0
5	0	0	0	0	0	-1	1	1	0	0	0
6	0	0	0	0	0	0	-1	0	1	0	0
7	0	0	0	0	0	0	0	0	-1	1	0
8	0	0	0	0	0	0	0	-1	0	-1	0
9	0	0	0	0	0	0	0	0	0	0	1
10	0	0	0	0	0	0	0	0	0	0	-1

Table 2.3: Boundary matrix $\mathbf{B}_1$ (incident matrix) of SC_2 in Figure 2.1.

In this example, a triangle may not be filled, e.g., $[1,3,4] \notin \Sigma_2$. However, for the clique/VR complex described in Section 2.4.1, every triangle in SC_2 is filled.

2.6.2 Technical lemmas of exterior calculus

In this section, we derive the closed-form of the 1-Laplacian $\Delta_1 = \mathsf{d}\delta + \delta\mathsf{d}$ on the local coordinate system (tangent plane) when the metric tensor at each point is identity. Note that the assumption is sufficient for current work since the Simplicial complex SC_2 is built in the *ambient space*. We start with some useful identities list as follows.

Lemma 2.20 (Identities of permutation parity)**.** *Given the Levi-Civita symbol ϵ, one has the following two identities:*

$$\epsilon_{\{i\}\{-i,-j\}} = \epsilon_{ij}\epsilon_{i,-i}. \tag{2.27}$$

Proof. Consider the case when $j > i$, i.e., $\epsilon_{ij} = 1$. We have $\epsilon_{\{i\}\{-i,-j\}} = \epsilon_{i,-i}$, for the inversion number of $\{i\}\{-i,-j\}$ is $i-1$, which is identical to the inversion number of

permutation $i, -i$. This implies the parity of the permutation is $\epsilon_{i,-i}$. Consider the case $i > j$, i.e., $\epsilon_{ij} = -1$, the inversion number of the permutation $\{i\}\{-i, -j\}$ is $i - 2$ since there is $i - 2$ elements (excluding j) that is smaller than i in $\{-i, -j\}$. This implies that the parity of $\{i\}\{-i, -j\}$ is $-1 \cdot \epsilon_{i,-i}$. This completes the proof. ■

Let $i' < j'$, the second identity states that,

$$\epsilon_{\{i',j'\}\{-i',-j'\}} = -\epsilon_{i',-i'}\epsilon_{j',-j'}. \tag{2.28}$$

Proof. The inversion number of the permutation $\{i', j'\}\{-i', -j'\}$ is $(j' - 2) + (i' - 1) = i' + j' - 3$; the inversion numbers of the permutations $\{j'\}\{-i', -j'\}$ and $i', -i'$ are $j' - 2$ and $i' - 1$, respectively. One can show that $\epsilon_{\{i',j'\}\{-i',-j'\}} = \epsilon_{\{j'\}\{-i',-j'\}} \cdot \epsilon_{i',-i'}$. From (2.27), we have $\epsilon_{\{j'\}\{-i',-j'\}} = \epsilon_{j'i'}\epsilon_{j',-j'} = -\epsilon_{j',-j'}$. This completes the proof. ■

The following lemma presents the codifferential of a 2-form.

Lemma 2.21 (Codifferential of a 2-form)**.** *Let $\zeta_2 = \sum_i \sum_{j\neq i} A_{ij} ds_j \wedge ds_i$ be a 2-form. The codifferential operator δ acting on ζ_2 is*

$$\delta\zeta_2 = \sum_i \sum_{j\neq i} \left(\frac{\partial A_{ji}}{\partial s_j} - \frac{\partial A_{ij}}{\partial s_j} \right) \mathsf{d}s_i. \tag{2.29}$$

Proof.

$$\star\mathsf{d} \star \zeta_2 = \sum_i \sum_{j\neq i} \star\mathsf{d}A_{ij} \star (\mathsf{d}s_j \wedge \mathsf{d}s_i) = \sum_i \sum_{j\neq i} \sum_{k\in\{i,j\}} \frac{\partial A_{ij}}{\partial s_k} \star (\mathsf{d}s_k \wedge \star(\mathsf{d}s_j \wedge \mathsf{d}s_i)) .$$

The last summation is over $k \in \{i, j\}$ otherwise it will produce zero. Next step is to derive

the exact form of $\star(\mathsf{d}s_k \wedge \star(\mathsf{d}s_j \wedge \mathsf{d}s_i))$. Consider the case when $k = i$, we have

$$\begin{aligned}
\star(\mathsf{d}s_i \wedge \star(\mathsf{d}s_j \wedge \mathsf{d}s_i)) &= \epsilon_{ji} \cdot \star(\mathsf{d}s_i \wedge \star(\mathsf{d}s_{i'} \wedge \mathsf{d}s_{j'})) \\
&= \epsilon_{ji}\epsilon_{\{i',j'\}\{-i',-j'\}} \star \left(\mathsf{d}s_i \wedge \bigwedge_{\ell \in \{-i',-j'\}} \mathsf{d}s_\ell \right) \\
&= \epsilon_{ji}\epsilon_{\{i',j'\}\{-i',-j'\}}\epsilon_{\{i\}\{-i,-j\}}\epsilon_{-j,j}\mathsf{d}s_j = \underbrace{\epsilon_{j,-j}\epsilon_{-j,j}}_{=(-1)^{d+1}}\mathsf{d}s_j.
\end{aligned}$$

Last equality holds from Lemma 2.20 and $\epsilon_{j,-j}\epsilon_{-j,j} = (-1)^{d+1}$. Consider the case when $k = j$,

$$\begin{aligned}
\star(\mathsf{d}s_j \wedge \star(\mathsf{d}s_j \wedge \mathsf{d}s_i)) &= \epsilon_{ji} \cdot \star(\mathsf{d}s_j \wedge \star(\mathsf{d}s_{i'} \wedge \mathsf{d}s_{j'})) \\
&= \epsilon_{ji}\epsilon(\{i',j'\}\{-i',-j'\}) \star \left(\mathsf{d}s_j \wedge \bigwedge_{\ell \in \{-i',-j'\}} \mathsf{d}s_\ell \right) \\
&= \epsilon_{ji} \underbrace{\epsilon_{\{i',j'\}\{-i',-j'\}}}_{=-\epsilon_{i,-i}\epsilon_{j,-j}} \underbrace{\epsilon_{\{j\}\{-i,-j\}}}_{=\epsilon_{ji}\epsilon_{j,-j}} \epsilon_{-i,i}\mathsf{d}s_i = (-1)^{d+1} \cdot (-\mathsf{d}s_i).
\end{aligned}$$

Putting things together, we have

$$\begin{aligned}
\delta\zeta_2 &= (-1)^{d+1} \star \mathsf{d} \star \zeta_2 = \sum_i \sum_{j \neq i} \frac{\partial A_{ij}}{\partial s_i}\mathsf{d}s_j - \frac{\partial A_{ij}}{\partial s_j}\mathsf{d}s_i \\
&= \sum_i \sum_{j \neq i} \left(\frac{\partial A_{ji}}{\partial s_j} - \frac{\partial A_{ij}}{\partial s_j} \right) \mathsf{d}s_i.
\end{aligned}$$

Last equality holds by changing the index of summing. ■

With Lemma 2.21, we can obtain the closed-form of Δ_1 in the local coordinate system.

Proposition 2.22 (1-Laplacian in local coordinate system)**.** *Let* $\zeta_1 = \sum_{i=1}^d f_i \mathsf{d}s_i$ *be a 1-form. The* up *Laplacian* $\Delta_1^{\text{down}} = \delta\mathsf{d}$ *operates on* ζ_1 *(in local coordinate system) is*

$$\delta\mathsf{d}\zeta_1 = \sum_i \sum_{j \neq i} \left(\frac{\partial^2 f_j}{\partial s_j \partial s_i} - \frac{\partial^2 f_i}{\partial s_j^2} \right) \mathsf{d}s_i. \tag{2.30}$$

The down *Laplacian* $\Delta_1^{\text{down}} = \mathsf{d}\delta$ *operates on* ζ_1 *(in local coordinate system) is*

$$\mathsf{d}\delta\zeta_1 = -\sum_i \sum_j \frac{\partial^2 f_j}{\partial s_i \partial s_j} \mathsf{d}s_i. \tag{2.31}$$

Proof. We first consider the *up* Laplacian $\delta\mathsf{d}\zeta_1 = \delta\zeta_2$ on 1 form ζ_1 with

$$\zeta_2 = \mathsf{d}\zeta_1 = \mathsf{d}\left(\sum_{i=1}^{d} f_i \mathsf{d}s_i\right) = \sum_i \sum_{j\neq i} \frac{\partial f_i}{\partial s_j} \mathsf{d}s_j \wedge \mathsf{d}s_i.$$

From Lemma 2.21 and let $A_{ij} = \frac{\partial f_i}{\partial s_j}$, we have

$$\delta\mathsf{d}\zeta_1 = \sum_i \sum_{j\neq i} \left(\frac{\partial^2 f_j}{\partial s_j \partial s_i} - \frac{\partial^2 f_i}{\partial s_j^2}\right) \mathsf{d}s_i.$$

Consider the case of the *down* Laplacian and note that $\delta = -\star\mathsf{d}\star$. Since the co-differential is now act on 1 form rather than $k = 2$, we have

$$\begin{aligned} \mathsf{d}\delta\zeta_1 \;&=\; -\mathsf{d}\star\mathsf{d}\star\zeta_1 = -\sum_i \mathsf{d}\star\mathsf{d}\left(f_i \star \mathsf{d}s_i\right) = -\sum_i \mathsf{d}\star\sum_j \frac{\partial f_i}{\partial s_j}\mathsf{d}s_j \wedge \star\mathsf{d}s_i \\ &\overset{\text{(i)}}{=} -\sum_i \mathsf{d}\star\frac{\partial f_i}{\partial s_i}\bigwedge_{\ell=1}^{d}\mathsf{d}s_\ell \overset{\text{(ii)}}{=} -\sum_i\sum_j \frac{\partial^2 f_i}{\partial s_i \partial s_j}\mathsf{d}s_j. \end{aligned}$$

Equality (i) holds since $\mathsf{d}s_j \wedge \star\mathsf{d}s_i = 1$ if $i = j$ and 0 otherwise. Equality (ii) holds for $\star\bigwedge_{\ell=1}^{d}\mathsf{d}s_\ell = 1$ (a 0-form). ■

Remark (Sanity check on 3D)**.** Note that in 3D, curl $= \star\mathsf{d}$, therefore $\delta\mathsf{d} = \star\mathsf{d}\star\mathsf{d} = \text{curl}\,\text{curl}$. For a vector field (1-form) $\zeta_1 = f_1\mathsf{d}s_1 + f_2\mathsf{d}s_2 + f_3\mathsf{d}s_3$, one has

$$\nabla \times \zeta_1 = \left(\frac{\partial f_3}{\partial s_2} - \frac{\partial f_2}{\partial s_3}\right)\mathsf{d}s_1 + \left(\frac{\partial f_1}{\partial s_3} - \frac{\partial f_3}{\partial s_1}\right)\mathsf{d}s_2 + \left(\frac{\partial f_2}{\partial s_1} - \frac{\partial f_1}{\partial s_2}\right)\mathsf{d}s_3.$$

$$\begin{aligned}
\nabla \times (\nabla \times \zeta_1) &= \left(\frac{\partial^2 f_2}{\partial s_1 \partial s_2} - \frac{\partial^2 f_1}{\partial s_2^2} - \frac{\partial^2 f_1}{\partial s_3^2} + \frac{\partial^2 f_3}{\partial s_1 \partial s_3} \right) \mathsf{d}s_1 \\
&= \left(\frac{\partial^2 f_3}{\partial s_2 \partial s_3} - \frac{\partial^2 f_2}{\partial s_3^2} - \frac{\partial^2 f_2}{\partial s_1^2} + \frac{\partial^2 f_1}{\partial s_1 \partial s_2} \right) \mathsf{d}s_2 \\
&= \left(\frac{\partial^2 f_1}{\partial s_1 \partial s_3} - \frac{\partial^2 f_3}{\partial s_1^2} - \frac{\partial^2 f_3}{\partial s_2^2} + \frac{\partial^2 f_2}{\partial s_2 \partial s_3} \right) \mathsf{d}s_3 \\
&= \sum_i \sum_{j \neq i} \left(\frac{\partial^2 f_j}{\partial s_j \partial s_i} - \frac{\partial^2 f_i}{\partial s_j^2} \right) \mathsf{d}s_i.
\end{aligned}$$

Corollary 2.23 (Relation to Laplace-Beltrami)**.** *We have the following,*

$$\Delta_1 f = (\delta \mathsf{d} + \mathsf{d}\delta)\zeta_1 = -\sum_i \sum_j \frac{\partial^2 f_i}{\partial s_j^2} \mathsf{d}s_i = -\sum_i \Delta_0 f_i \mathsf{d}s_i. \tag{2.32}$$

Here $\Delta_0 f_i = \sum_j \frac{\partial^2 f_i}{\partial s_j^2}$ *is the Laplacian on 0-form.*

Proof. Can be obtained by applying the result from Proposition 2.22. ■

Remark (Vector Laplacian in 3D)**.** Note that in 3D case, $\Delta_1 \zeta_1 = -\sum_i \Delta_0 f_i ds_i = -\nabla^2 f$. Here ∇^2 is vector Laplacian. This implies that vector Laplacian in 3D is essentially 1-Laplacian up to a sign change.

Corollary 2.24 (1-Laplacian on pure *curl* & *gradient* vector fields)**.** *If the vector field* ζ_1 *is a pure* curl *or* gradient *vector field, then*

$$\Delta_1 \zeta_1 = (\delta \mathsf{d} + \mathsf{d}\delta) f = -\sum_i \sum_j \frac{\partial^2 f_i}{\partial s_j^2} \mathsf{d}s_i = -\sum_i \Delta_0 f_i \mathsf{d}s_i. \tag{2.33}$$

Proof. Consider the case when ζ_1 is a *curl* flow ($\mathsf{d}\delta\zeta_1 = 0$). This implies

$$\sum_j \frac{\partial^2 f_j}{\partial s_i \partial s_j} = 0 \, \forall \, i \in [d].$$

Hence we have $\sum_{j\neq i}\frac{\partial^2 f_j}{\partial s_i \partial s_j} = -\frac{\partial^2 f_i}{\partial s_i^2}$. Plugging into (2.30), we have

$$\Delta_1\zeta_1 = \delta \mathsf{d}\zeta_1 = -\sum_i \sum_j \frac{\partial^2 f_i}{\partial s_j^2}\mathsf{d}s_i = -\sum_i \Delta_0 f_i \mathsf{d}s_i.$$

Consider the case when ζ_1 is *gradient* flow, which implies $\delta \mathsf{d} f = 0$, or

$$\sum_{j\neq i}\left(\frac{\partial^2 f_j}{\partial s_j \partial s_i} - \frac{\partial^2 f_i}{\partial s_j^2}\right) = 0 \,\forall\, i \in [d].$$

Therefore the identity $\sum_{j\neq i}\frac{\partial^2 f_j}{\partial s_j \partial s_i} = \sum_{j\neq i}\frac{\partial^2 f_i}{\partial s_j^2}$ holds. Plugging into (2.31), one has

$$\Delta_1\zeta_1 = \mathsf{d}\delta\zeta_1 = -\sum_i \left(\left(\sum_{j\neq i}\frac{\partial^2 f_i}{\partial s_j^2}\right) + \frac{\partial^2 f_i}{\partial s_i^2}\right)\mathsf{d}s_i = -\sum_i \Delta_0 f_i \mathsf{d}s_i.$$

This completes the proof. ■

Chapter 3

EIGEN-EMBEDDING OF THE DIFFUSION MAPS LAPLACIAN WITH LARGE ASPECT RATIO

We study a well-documented deficiency of manifold learning algorithms. Namely, as shown in Goldberg et al. [51], algorithms such as Laplacian Eigenmaps (LE) [9], Local Tangent Space Alignment (LTSA) [131], Hessian Eigenmaps (HLLE) [42], and Diffusion Maps (DM) [32] fail spectacularly when the data has a large aspect ratio; that is, it extends much more in one geodesic direction than in others. This problem, illustrated by the strip in Figure 3.1, was studied in Goldberg et al. [51] from a *linear algebraic* perspective; they show that, especially when noise is present, the problem is pervasive.

In this chapter, we revisit the problem from a *differential geometric* perspective. First, we define failure not as distortion, but as drop in the *rank* of the mapping ϕ represented by the embedding algorithm. In other words, the algorithm fails when the map ϕ is not invertible, or, equivalently, when the dimension $\dim \phi(\mathcal{M}) < \dim \mathcal{M} = d$, where $\mathcal{M}$ represents the idealized data manifold, and dim denotes the intrinsic dimension. Figure 3.1 demonstrates that the problem is fixed by choosing the eigenvectors with care. In fact, as we show in Section 3.3, one can *always* find a finite set of m eigenfunctions that provide a smooth d-dimensional map for the DM and LE algorithms [7], under mild geometric conditions. We call this problem the *Independent Eigencoordinate Selection* (IES) problem, formulate this problem, and explain its challenges in Section 3.1.

Our second main contribution is to design a bicriterial method that will select from a set of *coordinate functions* $\phi_1, \ldots \phi_m$, a subset S of small size that provides a smooth full-dimensional embedding of the data. The IES problem requires searching over a combinatorial number of sets. We show (Section 3.2) how to drastically reduce the computational burden per set for our algorithm. Third, we analyze the proposed criterion under asymptotic limit (Section 3.3), showing that it is reminiscent of a Kullbach-Leibler divergence between unnormalized measures. Finally (Section 3.4), we show examples of successful selection on real and synthetic data. The experiments also demonstrate that users of manifold learning for other than toy data *must* be aware of the IES problem and have tools for handling it.

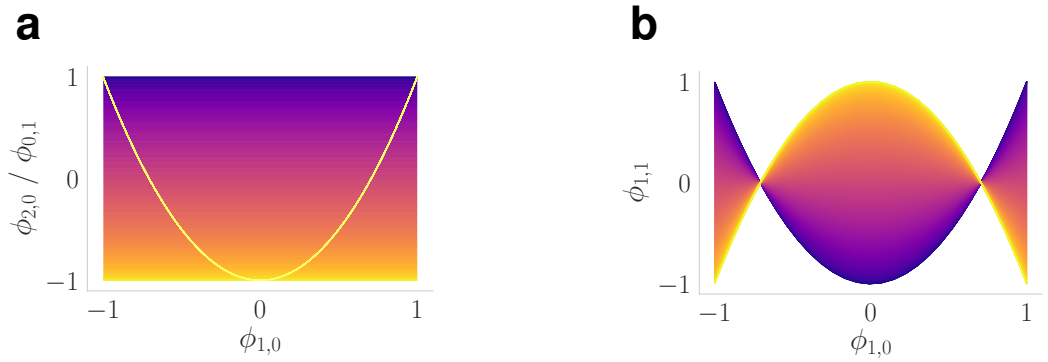

Figure 3.1: (a) Eigenfunction $\phi_{1,0}$ versus $\phi_{2,0}$ (curve) or $\phi_{0,1}$ (two dimensional manifold). (b) Eigenfunction $\phi_{1,0}$ versus $\phi_{1,1}$. All three manifolds are colored by the parameterization h.

3.1 IES problem, related work, and challenges

3.1.1 An example

Consider a continuous two dimensional strip with width W, height H, and *aspect ratio* $W/H \geq 1$, parametrized by coordinates $w \in [0, W], h \in [0, H]$. The eigenvalues and eigenfunctions of the Laplace-Beltrami operator Δ_0 with von Neumann boundary conditions [116] are

$$\lambda_{k_1,k_2} = \left(\frac{k_1\pi}{W}\right)^2 + \left(\frac{k_2\pi}{H}\right)^2,$$
$$\phi_{k_1,k_2}(w,h) = \cos\left(\frac{k_1\pi w}{W}\right)\cos\left(\frac{k_2\pi h}{H}\right).$$

Eigenfunctions $\phi_{1,0}$, $\phi_{0,1}$ are in bijection with the w, h coordinates (and give a full rank embedding), while the mapping by $\phi_{1,0}, \phi_{2,0}$ provides no extra information regarding the second dimension h in the underlying manifold (and is rank 1). Theoretically, one can choose as coordinates eigenfunctions indexed by $(k_1, 0), (0, k_2)$, but, in practice, k_1, and k_2 are usually unknown, as the eigenvalues are index by their rank $0 = \lambda_0 < \lambda_1 \leq \lambda_2 \leq \cdots$. For a two dimensional strip, it is known by Strauss [116] that $\lambda_{1,0}$ always corresponds to λ_1 and $\lambda_{0,1}$ corresponds to $\lambda_{(\lceil W/H \rceil)}$. Therefore, when $W/H > 2$, the mapping of the strip to $\mathbb{R}^2$ by ϕ_1, ϕ_2 is low rank, while the mapping by $\phi_1, \phi_{\lceil W/H \rceil}$ is full rank. Note that other

mappings of rank 2 exist, e.g., $\phi_1, \phi_{\lceil W/H \rceil+2}$ ($k_1 = k_2 = 1$ in Figure 3.1b). These embeddings reflect progressively higher frequencies, as the corresponding eigenvalues grow larger. For a more general manifold $\mathcal{M}$ where the spectral decomposition of Δ_0 has no analytic solution, searching for independent eigenfunctions (the IES problem) is not a trivial task.

3.1.2 Prior work

Goldberg et al. [51] is the first work to give the IES problem a rigurous analysis. Their paper focuses on rectangles, and the failure illustrated in Figure 3.1a is defined as obtaining a mapping $\mathbf{Y} = \phi(\mathbf{X})$ that is not *affinely equivalent* with the original data. They call this the *Price of Normalization* and explain it in terms of the variances along w and h. Dsilva et al. [44] is the first to frame the failure in terms of the rank of $\phi_S = \{\phi_k : k \in S \subseteq [m]\}$, calling it the *repeated eigendirection problem*. They propose a heuristic, LLRCoordSearch, based on the observation that if ϕ_k is a repeated eigendirection of $\phi_1, \cdots, \phi_{k-1}$, one can fit ϕ_k with *local linear regression* on predictors $\phi_{[k-1]}$ with low leave-one-out errors r_k. A sequential algorithm [17] with an unpredictability constraint in the eigenproblem has also been proposed. Under their framework, the k-th coordinate ϕ_k is obtained from the top eigenvector of the modified kernel matrix $\tilde{\mathbf{K}}_k$, which is constructed by the original kernel $\mathbf{K}$ and $\phi_1, \cdots, \phi_{k-1}$.

3.1.3 Existence of solution

Before trying to find an algorithmic solution to the IES problem, we ask the question whether this is even possible, in the smooth manifold setting. Positive answers are given in Portegies [97], which proves that isometric embeddings by DM with finite m are possible, and more recently in Bates [7], which proves that any closed, connected Riemannian manifold $\mathcal{M}$ can be smoothly embedded by its Laplacian eigenfunctions $\phi_{[m]}$ into $\mathbb{R}^m$ for some m, which depends only on the intrinsic dimension d of $\mathcal{M}$, the volume of $\mathcal{M}$, and lower bounds for *injectivity radius* and *Ricci curvature*. The example in Figure 3.1a demonstrates that, typically, not all m eigenfunctions are needed, i.e., there exists a set $S \subset [m]$, so that ϕ_S is also a smooth embedding. We follow Dsilva et al. [44] in calling such a set S *independent*. It is not known

how to find an independent S analytically for a given $\mathcal{M}$, except in special cases such as the strip. In this chapter, we propose a *finite sample* and algorithmic solution, and we support it with asymptotic theoretical analysis.

3.1.4 The IES Problem

We are given data $\mathbf{X}$, and the output of an embedding algorithm (DM for simplicity) $\mathbf{Y} = \phi(\mathbf{X}) = [\phi_1, \cdots, \phi_m] \in \mathbb{R}^{n \times m}$. We assume that $\mathbf{X}$ is sampled from a d-dimensional manifold $\mathcal{M}$, with known d, and that m is sufficiently large so that $\phi(\mathcal{M})$ is a smooth embedding. Further, we assume that there is a set $S \subseteq [m]$, with $|S| = s \leq m$, so that ϕ_S is also a smooth embedding of $\mathcal{M}$. We propose to find such set S so that the rank of ϕ_S is d on $\mathcal{M}$ and ϕ_S varies as slowly as possible.

3.1.5 Challenges

(i) Numerically, and on a finite sample, distiguishing between a full rank mapping and a rank-defective one is imprecise. Therefore, we substitute for rank the volume of a unit parallelogram in $\mathcal{T}_{\phi(\mathbf{x}_i)}\phi(\mathcal{M})$. (ii) Since ϕ is *not* an isometry, we must separate the local distortions introduced by ϕ from the estimated rank of ϕ at $\mathbf{x}$. (iii) Finding the optimal balance between the above desired properties. (iv) In Bates [7] it is strongly suggested that s the number of eigenfunctions needed may exceed the *Whitney embedding dimension* ($\leq 2d$), and that this number may depend on injectivity radius, aspect ratio, and so on. Section 3.5.3 shows an example of a flat 2-manifold, the *strip with cavity*, for which $s > 2$. In this chapter, we assume that s and m are given and focus on selecting S with $|S| = s$; for completeness, in Section 3.5.3 we present a heuristic to select s.

3.1.6 (Global) functional dependencies, knots and crossings

Before we proceed, we describe three different ways a mapping $\phi(\mathcal{M})$ can fail to be invertible. The first, *(global) functional dependency* is the case when $\operatorname{rank} \mathfrak{J}\phi < d$ on an open subset of

$\mathcal{M}$, or on all of $\mathcal{M}$ (yellow curve in Figure 3.1a); this is the case most widely recognized in the literature (e.g., Dsilva et al. [44] and Goldberg et al. [51]). The *knot* is the case when $\operatorname{rank}\mathfrak{J}\phi < d$ at an isolated point (Figure 3.1b). Third, the *crossing* (Figure 3.11 in Section 3.7) is the case when $\phi : \mathcal{M} \to \phi(\mathcal{M})$ is not invertible at $\mathbf{x}$, but $\mathcal{M}$ can be covered with open sets U such that the restriction $\phi : U \to \phi(U)$ has full rank d. Combinations of these three exemplary cases can occur. The criteria and approach we define are based on the (surrogate) rank of ϕ, therefore they will not rule out all crossings. We leave the problem of crossings in manifold embeddings to future work, as we believe that it requires an entirely separate approach (based, e.g., or the injectivity radius or density in the co-tangent bundle rather than differential structure).

3.2 Criteria and algorithm

3.2.1 A geometric criterion

We start with the main idea in evaluating the quality of a subset S of coordinate functions. At each data point i, we consider the orthogonal basis $\mathbf{U}(i) \in \mathbb{R}^{m\times d}$ of the d dimensional tangent subspace $\mathcal{T}_{\phi(\mathbf{x}_i)}\phi(\mathcal{M})$. The projection of the columns of $\mathbf{U}(i)$ onto the subspace $\mathcal{T}_{\phi(\mathbf{x}_i)}\phi_S(\mathcal{M})$ is $\mathbf{U}(i)[S,:] \equiv \mathbf{U}_S(i)$. The following Lemma connects $\mathbf{U}_S(i)$ and the co-metric $\mathbf{H}_S(i)$ defined by ϕ_S, with the *full* $\mathbf{H}(i)$.

Lemma 3.1. *Let* $\mathbf{H}(i) = \mathbf{U}(i)\mathbf{\Sigma}(i)\mathbf{U}(i)^\top$ *be the co-metric defined by embedding* ϕ, $S \subseteq [m]$, $\mathbf{H}_S(i)$ *and* $\mathbf{U}_S(i)$ *defined above. Then*

$$\mathbf{H}_S(i) = \mathbf{U}_S(i)\mathbf{\Sigma}(i)\mathbf{U}_S(i)^\top = \mathbf{H}(i)[S,S].$$

The proof is straightforward and left to the reader. Note that Lemma 3.1 is responsible for the efficiency of the search over sets S, given that the push-forward co-metric $\mathbf{H}_S$ can be readily obtained as a submatrix of $\mathbf{H}$. Denote by $\mathbf{u}_k^S(i)$ the k-th column of $\mathbf{U}_S(i)$. We

further normalize each $\mathbf{u}_k^S$ to length 1 and define the *normalized projected volume*

$$\mathrm{Vol}_{\mathrm{norm}}(S,i) = \frac{\sqrt{\det(\mathbf{U}_S(i)^\top \mathbf{U}_S(i))}}{\prod_{k=1}^d \|\mathbf{u}_k^S(i)\|_2}.$$

Conceptually, $\mathrm{Vol}_{\mathrm{norm}}(S,i)$ is the volume spanned by a (non-orthonormal) "basis" of unit vectors in $\mathcal{T}_{\phi_S(\mathbf{x}_i)}\phi_S(\mathcal{M})$; $\mathrm{Vol}_{\mathrm{norm}}(S,i) = 1$ when $\mathbf{U}_S(i)$ is orthogonal, and it is 0 when $\operatorname{rank}\mathbf{H}_S(i) < d$. In Figure 3.1a, the $\mathrm{Vol}_{\mathrm{norm}}(\{1,2\})$ with $\phi_{\{1,2\}} = \{\phi_{1,0}, \phi_{2,0}\}$ is close to zero, since the projection of the two tangent vectors is parallel to the yellow curve; however $\mathrm{Vol}_{\mathrm{norm}}(\{1, \lceil w/h \rceil\}, i)$ is almost 1, because the projections of the tangent vectors $\mathbf{U}(i)$ will be (approximately) orthogonal. Hence, $\mathrm{Vol}_{\mathrm{norm}}(S,i)$ away from 0 indicates a non-singular ϕ_S at i, and we use the average $\log \mathrm{Vol}_{\mathrm{norm}}(S,i)$, which penalizes values near 0 highly, as the *rank quality* $\mathfrak{R}(S)$ of S.

Higher frequency ϕ_S maps with high $\mathfrak{R}(S)$ may exist, being either smooth, such as the embeddings of the strip mentioned previously, or containing knots involving only small fraction of points, such as $\phi_{\phi_{1,0},\phi_{1,1}}$ in Figure 3.1a. To choose the lowest frequency, slowest varying smooth map, a regularization term consisting of the eigenvalues λ_k, $k \in S$, of the graph Laplacian $\mathbf{L}$ is added, obtaining the criterion

$$\mathfrak{L}(S;\zeta) = \underbrace{\frac{1}{n}\sum_{i=1}^n \log \sqrt{\det\left(\mathbf{U}_S(i)^\top \mathbf{U}_S(i)\right)}}_{\mathfrak{R}_1(S) = \frac{1}{n}\sum_{i=1}^n \mathfrak{R}_1(S;i)} - \underbrace{\frac{1}{n}\sum_{i=1}^n \sum_{k=1}^d \log \|\mathbf{u}_k^S(i)\|_2}_{\mathfrak{R}_2(S) = \frac{1}{n}\sum_{i=1}^n \mathfrak{R}_2(S;i)} - \zeta \sum_{k\in S} \lambda_k \tag{3.1}$$

3.2.2 Search algorithm

With this criterion, the IES problem turns into a subset selection problem parametrized by ζ, with the optimal coordinate subset S_* be the maximizer of (3.1), i.e.,

$$S_*(\zeta) = \operatorname*{argmax}_{S\subseteq[m];|S|=s;1\in S} \mathfrak{L}(S;\zeta). \tag{3.2}$$

Note that we force the first coordinate ϕ_1 to always be chosen, since this coordinate

cannot be functionally dependent on previous ones, and, in the case of DM, it also has lowest frequency. Note also that $\mathfrak{R}_1$ and $\mathfrak{R}_2$ are both submodular set function (proof in Section 3.2.5). For large s and d, algorithms for optimizing over the difference of submodular functions can be used (e.g., see Iyer and Bilmes [61]). For the experiments in this chapter, we have $m = 20$ and $d, s = 2 \sim 4$, which enables us to use exhaustive search to handle (3.2). The exact search algorithm is summarized in Algorithm 3.1 IndEigenSearch. Note that one might be able to search in the continuous space of all s-projections. We conjecture the objective function (3.1) will be a difference of convex function and leave the details as future work.

Algorithm 3.1: IndEigenSearch: IES algorithm in (3.2)

Input : Data $\mathbf{X}$, bandwith ε, intrinsic dimension d, embedding dimension s, regularizer ζ

$\mathbf{Y} \in \mathbb{R}^{n\times m}, \mathbf{L}, \boldsymbol{\lambda} \in \mathbb{R}^m \leftarrow$DiffusionMaps$(\mathbf{X}, \varepsilon)$

$\mathbf{U}(i), \cdots, \mathbf{U}(n) \leftarrow$RMetric$(\mathbf{Y}, \mathbf{L}, d)$

for $S \in \{S' \subseteq [m] : |S'| = s, 1 \in S'\}$ **do**

  $\mathfrak{R}_1(S) \leftarrow 0; \mathfrak{R}_2(S) \leftarrow 0$

  for $i = 1, \cdots, n$ **do**

    $\mathbf{U}_S(i) \leftarrow \mathbf{U}(i)[S, :]$

    $\mathfrak{R}_1(S)$ += $\frac{1}{2n} \cdot \log\det\left(\mathbf{U}_S(i)^\top \mathbf{U}_S(i)\right)$

    $\mathfrak{R}_2(S)$ += $\frac{1}{n} \cdot \sum_{k=1}^{d} \log \|u_k^S(i)\|_2$

  $\mathfrak{L}(S;\zeta) = \mathfrak{R}_1(S) - \mathfrak{R}_2(S) - \zeta \sum_{k\in S} \lambda_k$

$S_* = \text{argmax}_S\, \mathfrak{L}(S;\zeta)$

Return : Independent eigencoordinates set S_*

3.2.3 Regularization path and choosing ζ

According to (3.1), the optimal subset S_* depends on the parameter ζ. The regularization path

$$\ell(\zeta) = \max_{S\subseteq[m];|S|=s;1\in S} \mathfrak{L}(S;\zeta),$$

is the upper envelope of multiple lines (each correspond to a set S) with slopes $-\sum_{k\in S}\lambda_k$ and intercepts $\mathfrak{R}(S)$. The larger ζ is, the more the lower frequency subset penalty prevails, and for sufficiently large ζ the algorithm will output $[s]$. In the supervised learning framework, the regularization parameters are often chosen by cross validation. Here we propose a second criterion, that effectively limits how much $\mathfrak{R}(S)$ may be ignored, or alternatively, bounds ζ by a data dependent quantity. Define the *leave-one-out regret* of point i as follows

$$\begin{aligned}\mathfrak{D}(S,i) &= \mathfrak{R}(S^i_*;[n]\backslash\{i\}) - \mathfrak{R}(S;[n]\backslash\{i\}),\\ \text{with } S^i_* &= \operatorname{argmax}_{S\subseteq[m];|S|=s;1\in S}\mathfrak{R}(S;i).\end{aligned} \tag{3.3}$$

In the above, we denote $\mathfrak{R}(S;T) = \frac{1}{|T|}\sum_{i\in T}\mathfrak{R}_1(S;i) - \mathfrak{R}_2(S;i)$ for some subset $T\subseteq[n]$. The quantity $\mathfrak{D}(S,i)$ in (3.3) measures the gain in $\mathfrak{R}$ if all the other points $[n]\backslash\{i\}$ choose the optimal subset S^i_*. If the regret $\mathfrak{D}(S,i)$ is larger than zero, it indicates that the alternative choice might be better compared to original choice S. Note that the mean value for all i, i.e., $\frac{1}{n}\sum_i \mathfrak{D}(S,i)$ depends also on the variability of the optimal choice of points i, S^i_*. Therefore, it might not favor an S, if S is optimal for every $i\in[n]$. Instead, we propose to inspect the distribution of $\mathfrak{D}(S,i)$, and remove the sets S for which α's percentile are larger than zero, e.g., $\alpha = 75\%$, recursively from $\zeta=\infty$ in decreasing order. Namely, the chosen set is $S_* = S_*(\zeta')$ with $\zeta' = \max_{\zeta\geq 0}\textsc{Percentile}(\{\mathfrak{D}(S_*(\zeta),i)\}_{i=1}^n,\alpha)\leq 0$. The optimal ζ_* value is simply chosen to be the midpoint of all the ζ's that outputs set S_* i.e., $\zeta_* = \frac{1}{2}(\zeta'+\zeta'')$, where $\zeta'' = \min_{\zeta\geq 0} S_*(\zeta) = S_*(\zeta')$. The procedure ReguParamSearch is summarized in Algorithm 3.2.

3.2.4 Greedy search algorithm.

Inspired by the greedy version of submodular maximization [86], a greedy heuristic has been proposed, as in Algorithm 3.3. The algorithm starts from an observation that the optimal value of the $S' = \operatorname{argmax}_{S;d\leq|S|<s}\mathfrak{L}(S;\zeta)$ will often time be a subset of the optimal S_* of (3.2). Since the appropriate cardinality of the set S is unknown, we can simply scan from $|S| = d$

Algorithm 3.2: REGUPARAMSEARCH: searching for the optimal regularization parameter in (3.1)

Input : Threshold parameter α

for $\zeta = \zeta_{\max} \to 0$ **do**
- ▷ $\zeta_{\max}$ `should be sufficiently large such that` $S_*(\zeta_{\max}) = [s]$
- $S \leftarrow S_*(\zeta)$; $S_* \leftarrow$`NULL`; $\zeta'' \leftarrow$ `NULL`
- **for** $i \in [n]$ **do**
  - $\mathfrak{D}(S, i) \leftarrow \mathfrak{R}(S^i_*; [n]\backslash\{i\}) - \mathfrak{R}(S; [n]\backslash\{i\})$ from equation (3.3)
- **if** $\textit{PERCENTILE}(\{\mathfrak{D}(S, i)\}_{i=1}^n, \alpha) \leq 0$ **and** $S_* =$ *NULL* **then**
  - Optimal set $S_* \leftarrow S$
  - $\zeta' \leftarrow \zeta$ ▷ `First found a set that satisfies the criterion.`
- **else if** $S_* \neq$ *NULL* **and** $S_* = S_*(\zeta)$ **then**
  - $\zeta'' \leftarrow \zeta$ ▷ `Searching for` ζ''
- **else if** $S_* \neq$ *NULL* **and** $\zeta'' \neq$ *NULL* **and** $S_* \neq S_*(\zeta)$ **then**
  - $\zeta_* \leftarrow \frac{1}{2}(\zeta' + \zeta'')$
  - **break** ▷ `Leave the loop when found` $\zeta'' = \min_{\zeta \geq 0} S_*(\zeta') = S_*(\zeta)$
- **else**
  - **continue**

Return : Optimal set S_*, optimal regularization parameter ζ_*

to m. The order of the returned elements indicates the significance of the corresponding coordinate.

Algorithm 3.3: GREEDYINDEIGENSEARCH: the greedy version of the IES algorithm in Algorithm 3.1

Input : Orthogonal basis $\{\mathbf{U}(i)\}_{i=1}^n$, eigenvalues $\boldsymbol{\lambda}$, intrinsic dimension d, regularization parameter ζ

Solve $S_* \leftarrow \text{argmax}_{S \subseteq [m]; |S| = d; 1 \in S} \mathfrak{L}(S; \zeta)$.

for $s = d + 1 \to m$ **do**
- $k_* = \text{argmax}_{k \in [m] \backslash S_*} \mathfrak{L}(S_* \cup \{k\}; \zeta)$
- $S_* \leftarrow S_* \cup \{k_*\}$ ▷ `Record order`

Return : Independent coordinates S_*

3.2.5 Submodularity of the utility functions

As discussed earlier, the loss function $\mathfrak{L}$ can be written as a difference of two utility functions $\mathfrak{R}_1$ and $\mathfrak{R}_2$. In this section, we analyze the property of these two utility functions and show that they are both *submodular*. We start with the utility function $\mathfrak{R}_1(S)$.

Proposition 3.2. *For a rank d tangent space matrix $\mathbf{U} \in \mathbb{R}^{m\times d}$, if any submatrix $\mathbf{U}_S$, with index set $S \subseteq [m]$ and $|S| = s \geq d$, is rank d, we have $\mathfrak{R}_1$ be a submodular set function.*

Proof. W.l.o.g, set $n = 1$, with slightly abuse of notation, let $\mathbf{U} = \mathbf{U}_{T\cup\{i\}} \in \mathbb{R}^{(|T|+1)\times d}$. The matrix can be written in the following form

$$\mathbf{U} = \begin{bmatrix} \mathbf{T} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{S} \\ \mathbf{V} \\ \mathbf{a} \end{bmatrix} \in \mathbb{R}^{(|T|+1)\times d}$$

With $\mathbf{U}_S = \mathbf{S}$, $\mathbf{U}_T = \mathbf{T}$ and $U_{\{i\}} = \mathbf{a}$ for set $S \subseteq T \subseteq [m]$ and $i \in [m]\backslash T$. Here $\mathbf{a} \in \mathbb{R}^{1\times d}$. By the definition of $\mathfrak{R}_1$ in (3.1), one has (ignoring the constants)

$$\begin{aligned}
\mathfrak{R}_1(S) &= \log\det(\mathbf{S}^\top\mathbf{S}), \\
\mathfrak{R}_1(T) &= \log\det\left(\mathbf{T}^\top\mathbf{T}\right), \\
\mathfrak{R}_1(S\cap\{i\}) &= \log\det\left(\begin{bmatrix} \mathbf{S} \\ \mathbf{a} \end{bmatrix}^\top \begin{bmatrix} \mathbf{S} \\ \mathbf{a} \end{bmatrix}\right), \\
\mathfrak{R}_1(T\cap\{i\}) &= \log\det(\mathbf{U}^\top\mathbf{U}).
\end{aligned}$$

Denote $\partial_i f(S) = f(S\cup\{i\}) - f(S)$ for some function f, we have

$$\begin{aligned}
\partial_i\mathfrak{R}_1(S) &= \log\det(\mathbf{S}^\top\mathbf{S} + \mathbf{a}^\top\mathbf{a}) - \log\det(\mathbf{S}^\top\mathbf{S}), \\
\partial_i\mathfrak{R}_1(T) &= \log\det(\mathbf{T}^\top\mathbf{T} + \mathbf{a}^\top\mathbf{a}) - \log\det(\mathbf{T}^\top\mathbf{T}).
\end{aligned}$$

The full rank of any submatrices guarantees the positive definiteness of $\mathbf{S}^\top\mathbf{S}, \mathbf{T}^\top\mathbf{T}$, by matrix determinant lemma [53], we have

$$\det(\mathbf{S}^\top\mathbf{S} + \mathbf{a}^\top\mathbf{a}) = \det(\mathbf{S}^\top\mathbf{S})\left(1 + \mathbf{a}(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{a}^\top\right).$$

Therefore

$$\partial_i\mathfrak{R}_1(S) = 1 + \mathbf{a}(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{a}^\top.$$

Similar equation holds for set T. Therefore,

$$\partial_i\mathfrak{R}_1(S) - \partial_i\mathfrak{R}_1(T) = \log\frac{1 + \mathbf{a}(\mathbf{S}^\top\mathbf{S})^{-1}\mathbf{a}^\top}{1 + \mathbf{a}(\mathbf{T}^\top\mathbf{T})^{-1}\mathbf{a}^\top}.$$

Because $\mathbf{T}^\top\mathbf{T} \succeq \mathbf{S}^\top\mathbf{S}$, we have $(\mathbf{S}^\top\mathbf{S})^{-1} \succeq (\mathbf{T}^\top\mathbf{T})^{-1}$ [59], which implies $\partial_i\mathfrak{R}_1(S) - \partial_i\mathfrak{R}_1(T) \geq 0$ for all $S \subseteq T \subseteq [m]$ and $i \in [m]\backslash T$. This completes the proof. ■

We then turn to the second utility function $\mathfrak{R}_2(S)$.

Proposition 3.3. *$\mathfrak{R}_2$ is a submodular set function.*

Proof. W.L.O.G, set $n, d = 1$. With slightly abuse of notation, let $\mathbf{u} \leftarrow \mathbf{u}_1(i)$ and $\mathbf{u}_S \leftarrow \mathbf{u}_1^S(i)$. For any set $S \subseteq T \subseteq [m]$ and $i \in [m]\backslash T$, we have

$$\partial_i\mathfrak{R}_2(S) = \mathfrak{R}_2\left(S \cap \{i\}\right) - \mathfrak{R}_2(S) = \log\frac{\sum_{k\in S} u_k^2 + u_i^2}{\sum_{k\in S} u_k^2} = \log\frac{\Sigma_S + u_i^2}{\Sigma_S},$$
$$\partial_i\mathfrak{R}_2(T) = \mathfrak{R}_2\left(T \cap \{i\}\right) - \mathfrak{R}_2(T) = \log\frac{\sum_{k\in T} u_k^2 + u_i^2}{\sum_{k\in T} u_k^2} = \log\frac{\Sigma_S + \Sigma_{T\backslash S} + u_i^2}{\Sigma_S + \Sigma_{T\backslash S}}.$$

Where $\Sigma_S = \sum_{k\in S} u_k^2$. By definition, we have $\Sigma_S, \Sigma_{T\backslash S}, u_i^2 \geq 0$. Therefore,

$$\partial_i\mathfrak{R}_2(S) - \partial_i\mathfrak{R}_2(T) = \log\frac{\left(\Sigma_S + u_i^2\right)\cdot\left(\Sigma_S + \Sigma_{T\backslash S}\right)}{\Sigma_S\cdot\left(\Sigma_S + \Sigma_{T\backslash S} + u_i^2\right)}$$
$$= \underbrace{\log\left[\frac{\Sigma_S^2 + \Sigma_S\left(\Sigma_{T\backslash S} + u_i^2\right) + u_i^2\Sigma_{T\backslash S}}{\Sigma_S^2 + \Sigma_S\left(\Sigma_{T\backslash S} + u_i^2\right)}\right]}_{\geq 1} \geq 0.$$

This completes the proof. ■

3.3 $\mathfrak{R}$ as Kullbach-Leibler divergence

In this section we analyze $\mathfrak{R}$ in its population version, and show that it is reminiscent of a Kullbach-Leibler divergence between *unnormalized* measures on $\phi_S(\mathcal{M})$. The population version of the regularization term takes the form of a well-known *smoothness* penalty on the embedding coordinates ϕ_S.

3.3.1 Volume element and the Riemannian metric

Consider a Riemannian manifold $(\mathcal{M}, g)$ mapped by a smooth embedding ϕ_S into $(\phi_S(\mathcal{M}), g_{*\phi_S})$, $\phi_S : \mathcal{M} \to \mathbb{R}^s$, where $g_{*\phi_S}$ is the *push-forward* metric defined in (2.1). A Riemannian metric g induces a *Riemannian measure* on $\mathcal{M}$, with volume element $\sqrt{|g|}$. Denote now by $\mu_{\mathcal{M}}$, respectively $\mu_{\phi_S(\mathcal{M})}$ the Riemannian measures corresponding to the metrics induced on $\mathcal{M}, \phi_S(\mathcal{M})$ by the ambient spaces $\mathbb{R}^D, \mathbb{R}^s$; let g be the former metric.

Lemma 3.4. *Let $S, \phi, \phi_S, \mathbf{H}_S(\mathbf{x}), \mathbf{U}_S(\mathbf{x}), \mathbf{\Sigma}(\mathbf{x})$ be defined as in Section 3.2 and Lemma 3.1. For simplicity, we denote by $\mathbf{H}_S(\mathbf{y}) \coloneqq \mathbf{H}_S(\phi_S^{-1}(\mathbf{y}))$, and similarly for $\mathbf{U}_S(\mathbf{y}), \mathbf{\Sigma}_S(\mathbf{y})$. Assume that ϕ_S is a smooth embedding. Then, for any measurable function $f : \mathcal{M} \to \mathbb{R}$,*

$$\int_{\mathcal{M}} f(\mathbf{x}) \mathsf{d}\mu_{\mathcal{M}}(\mathbf{x}) = \int_{\phi_S(\mathcal{M})} f(\phi_S^{-1}(\mathbf{y})) j_S(y) \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y}),$$
$$\textit{where} \;\; j_S(\mathbf{y}) \;=\; 1/\operatorname{Vol}(\mathbf{U}_S(\mathbf{y}) \mathbf{\Sigma}_S^{1/2}(\mathbf{y})).$$

Proof. Let $\mu^*_{\phi_S(\mathcal{M})}$ denote the Riemannian measure induced by $g_{*\phi_S}$. Since $(\mathcal{M}, g)$ and $(\phi_S(\mathcal{M}), g_{*\phi_S})$ are isometric by definition, $\int_{\mathcal{M}} f(\mathbf{x}) \mathsf{d}\mu_{\mathcal{M}}(\mathbf{x}) = \int_{\phi_S(\mathcal{M})} f(\phi_S^{-1}(\mathbf{y})) \mathsf{d}\mu^*_{\phi_S(\mathcal{M})}(\mathbf{y}) = \int_{\phi_S(\mathcal{M})} f(\phi_S^{-1}(\mathbf{y})) \sqrt{|g_{*\phi_S}(\mathbf{y})|} \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y})$ follows from the change of variable formula. It remains to find the expression of $j_S(\mathbf{y}) = \sqrt{|g_{*\phi_S}(\mathbf{y})|}$. The matrix $\mathbf{U}_S(\mathbf{y})$ (note that $\mathbf{U}_S(\mathbf{y})$ is *not orthogonal*) can be written as

$$\mathbf{U}_S(\mathbf{y}) \;=\; \mathbf{V}\mathbf{Q}_S(\mathbf{y}), \tag{3.4}$$

where $\mathbf{V} \in \mathbb{R}^{s\times d}$ is an orthogonal matrix and $\mathbf{Q}_S(\mathbf{y}) \in \mathbb{R}^{d\times d}$ is upper triangular. Then,

$$\mathbf{H}_S(\mathbf{y}) = \mathbf{U}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})\mathbf{U}_S(y)^\top = \mathbf{V}_S(y)\underbrace{(\mathbf{Q}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})\mathbf{Q}_S(\mathbf{y})^\top)}_{\tilde{\mathbf{H}}_S(\mathbf{y})}\mathbf{V}_S(\mathbf{y})^\top.$$

In the above $\tilde{\mathbf{H}}_S(y)$ is the co-metric expressed in the new coordinate system induced by $\mathbf{V}_S(\mathbf{y})$. Hence, in the same basis, $g_{*\phi_S}$ is expressed by

$$\tilde{\mathbf{G}}_S(y) \;=\; \tilde{\mathbf{H}}_S(y)^{-1} \;=\; (\mathbf{Q}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})\mathbf{Q}_S(\mathbf{y})^\top)^{-1}.$$

The volume element, which is invariant to the chosen coordinate system, is

$$\det\left(\mathbf{Q}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})\mathbf{Q}_S(\mathbf{y})^\top\right)^{-1/2} \;=\; \prod_{k=1}^{d}\sigma_k(\mathbf{y})^{-1/2}q_{S,kk}(\mathbf{y})^{-1}.$$

From (3.4), it follows also that

$$\det\left(\mathbf{Q}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})\mathbf{Q}_S(\mathbf{y})^\top\right)^{-1/2} = 1/\operatorname{Vol}\left(\mathbf{U}_S(\mathbf{y})\mathbf{\Sigma}(\mathbf{y})^{1/2}\right).$$

■

3.3.2 Asymptotic limit of $\mathfrak{R}$

We now study the first term of our criterion in the limit of infinite sample size. We make the following assumptions.

Assumption 3.1. *The manifold* $\mathcal{M}$ *is compact of class* C^3*, and there exists a set* S*, with* $|S| = s$ *so that* ϕ_S *is a smooth embedding of* $\mathcal{M}$ *in* $\mathbb{R}^s$*.*

Assumption 3.2. *The data are sampled from a distribution on* $\mathcal{M}$ *continuous with respect to* $\mu_{\mathcal{M}}$*, whose density is denoted by* p*.*

Assumption 3.3. *The estimate of* $\mathbf{H}_S$ *in Algorithm 2.3 computed w.r.t. the embedding* ϕ_S *is consistent.*

We know from Bates [7] that Assumption 3.1 is satisfied for the DM/LE embedding. The remaining assumptions are minimal requirements ensuring that limits of our quantities exist. Now consider the setting in Section 3.1, in which we have a larger set of eigenfunctions, $\phi_{[m]}$ so that $[m]$ contains the set S of Assumption 3.1. Denote by $\tilde{\jmath}_S$ a new volume element with $\sigma_k = [\mathbf{\Sigma}]_{kk}$ as follows:

$$\tilde{\jmath}_S(\mathbf{y}) = \prod_{k=1}^{d} \left(||\mathbf{u}_k^S(\mathbf{y})|| \sigma_k(\mathbf{y}) \right)^{1/2} \Big)^{-1}.$$

The following theorem is to analyze the limit of the difference of two utility functions $\mathfrak{R} = \mathfrak{R}_1 - \mathfrak{R}_2$.

Theorem 3.5 (Limit of $\mathfrak{R}$)**.** *Under Assumptions 3.1–3.3,*

$$\lim_{n\to\infty} \frac{1}{n} \sum_i \ln \mathfrak{R}(S, \mathbf{x}_i) = \mathfrak{R}(S, \mathcal{M}),$$

and

$$\mathfrak{R}(S, \mathcal{M}) \;=\; -\int_{\phi_S(\mathcal{M})} \ln \frac{j_S(\mathbf{y})}{\tilde{\jmath}_S(\mathbf{y})} j_S(\mathbf{y}) p(\phi_S^{-1}(\mathbf{y})) \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y}) \;\coloneqq\; -\mathrm{KL}(p j_S \| p \tilde{\jmath}_S). \tag{3.5}$$

Proof. Because ϕ_S is a smooth embedding, $j_S(\mathbf{y}) > 0$ on $\phi_S(\mathcal{M})$, and because $\mathcal{M}$ is compact, $\min_{\phi_S(\mathcal{M})} j_S(\mathbf{y}) > 0$. Similarly, noting that $\tilde{\jmath}_S(\mathbf{y}) \geq \prod_{k=1}^{d} \sigma_k^{-1/2}(\mathbf{y})$, we conclude that $\tilde{\jmath}_S(\mathbf{y})$ is also bounded away from 0 on $\mathcal{M}$. Therefore $\ln j_S(\mathbf{y})$ and $\ln \tilde{\jmath}_S(\mathbf{y})$ are bounded, and the integral in the RHS of (3.5) exists and has a finite value. Now,

$$\frac{1}{n} \sum_i \ln \mathfrak{R}(S, \mathbf{x}_i) \;\to\; \int_{\mathcal{M}} \ln \mathfrak{R}(S, \mathbf{x}) p(\mathbf{x}) \mathsf{d}\mu_{\mathcal{M}}(\mathbf{x}) \;=\; \mathfrak{R}(S, \mathcal{M}).$$

The RHS of the equation above is

$$
\begin{aligned}
& \int_{\mathcal{M}} \ln \mathfrak{R}(S,\mathbf{x}) p(\mathbf{x}) \mathsf{d}\mu_{\mathcal{M}}(\mathbf{x}) \\
= & \int_{\phi_S(\mathcal{M})} \ln \mathfrak{R}(\phi_S^{-1}(\mathbf{y})) p(\phi_S^{-1}(\mathbf{y})) j_S(\mathbf{y}) \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y}) \\
= & \int_{\phi_S(\mathcal{M})} \left[\frac{1}{2} \ln \frac{\mathrm{Vol}\left(\mathbf{U}_S^\top(\mathbf{y}) \mathbf{U}_S(\mathbf{y})\right)}{\tilde{j}_S(\mathbf{y})} - \frac{p(\phi_S^{-1}(\mathbf{y}) \prod_{k=1}^d \sigma_k^{1/2}(\mathbf{y})}{p(\phi_S^{-1}(\mathbf{y}) \prod_{k=1}^d \sigma_k^{1/2}(\mathbf{y})} \right] p(\phi_S^{-1}(\mathbf{y})) j_S(\mathbf{y}) \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y}) \\
= & \int_{\phi_S(\mathcal{M})} \ln \frac{j_S(\mathbf{y}) p(\phi_S^{-1}(\mathbf{y})}{\tilde{j}_S(\mathbf{y}) p(\phi_S^{-1}(\mathbf{y})} p(\phi_S^{-1}(\mathbf{y})) j_S(\mathbf{y}) \mathsf{d}\mu_{\phi_S(\mathcal{M})}(\mathbf{y}) \;=\; -\mathrm{KL}(p j_S \| p \tilde{j}_S)
\end{aligned}
$$

■

The expression $\mathrm{KL}(\cdot\|\cdot)$ represents a Kullbach-Leibler divergence. Note that $j_S \geq \tilde{j}_S$, which implies that KL is always positive, and that the measures defined by $pj_S, p\tilde{j}_S$ normalize to different values. By definition, local injectivity is related to the volume element j. Intuitively, pj_S is the *observation* and $p\tilde{j}_S$, where $\tilde{j}_S$ is the minimum attainable for j_S, is the *model*; the objective itself is looking for a view S of the data that agrees with the model.

It is known that λ_k, the k-th eigenvalue of the Laplacian, converges under certain technical conditions [9] to an eigenvalue of the Laplace-Beltrami operator Δ_0 and that

$$
\lambda_k(\Delta_0) = \langle \phi_k, \Delta_0 \phi_k \rangle = \int_{\mathcal{M}} \| \operatorname{grad} \phi_k(\mathbf{x}) \|_2^2 \mathsf{d}\mu(\mathcal{M}). \tag{3.6}
$$

Hence, a smaller value for the regularization term encourages the use of slow varying coordinate functions, as measured by the squared norm of their gradients, as in equation (3.6). Hence, under Assumptions 3.1–3.3, $\mathfrak{L}$ converges to

$$
\mathfrak{L}(S,\mathcal{M}) \;=\; -\mathrm{KL}(p j_S \| p \tilde{j}_S) - \left(\frac{\zeta}{\lambda_1(\mathcal{M})} \right) \sum_{k \in S} \lambda_k(\mathcal{M}). \tag{3.7}
$$

Since eigenvalues scale with the volume of $\mathcal{M}$, the rescaling of ζ in comparison with equation (3.1) makes the ζ above adimensional.

3.4 Experiments

We demonstrate the proposed algorithm on three synthetic datasets, one where the minimum embedding dimension s equals d ($\mathcal{D}_1$ *long strip*), and two ($\mathcal{D}_7$ *high torus* and $\mathcal{D}_{13}$ *three torus*) where $s > d$. The complete list of synthetic manifolds (transformations of 2 dimensional strips, 3 dimensional cubes, two and three tori, etc.) investigated can be found in Appendix 3.7 and Table 3.2. The examples have (i) aspect ratio of at least 4 and (ii) points sampled *non-uniformly* from the underlying manifold $\mathcal{M}$, with (iii) additional Gaussian noise added. The sample size of the synthetic datasets is $n = 10,000$ unless otherwise stated. Additionally, we analyze several real datasets from chemistry and astronomy. All embeddings are computed with the DM algorithm, which outputs $m = 20$ eigenvectors. Hence, we examine 171 sets for $s = 3$ and 969 sets for $s = 4$. No more than 2 to 5 of these sets appear on the regularization path. Detailed experimental results are in Table 3.3. In this section, we show the original dataset $\mathbf{X}$, the embedding ϕ_{S_*}, with S_* selected by IndEigenSearch and ζ_* from ReguParamSearch, and the maximizer sets on the regularization path with box plots of $\mathfrak{D}(S, i)$ as discussed in Section 3.2. The α threshold for ReguParamSearch is set to 75%. The kernel bandwidth ε for synthetic datasets is chosen manually. For real datasets, ε is optimized as in Joncas et al. [64]. All the experiments are replicated for more than 5 times, and the outputs are similar because of the large sample size n.

3.4.1 Synthetic manifolds

The results of synthetic manifolds ($\mathcal{D}_1$, $\mathcal{D}_7$, and $\mathcal{D}_{13}$) are in Figure 3.2.

Manifold with $s = d$. The first synthetic dataset we considered, $\mathcal{D}_1$, is a two-dimensional strip with aspect ratio $W/H = 2\pi$. Left panel of the top row shows the scatter plot of such dataset. From the theoretical analysis in Section 3.1, the coordinate set that corresponds to slowest varying unique eigendirection is $S = \{1, \lceil W/H \rceil\} = \{1, 7\}$. Middle panel, with $S_* = \{1, 7\}$ selected by IndEigenSearch with ζ chosen by ReguParamSearch, confirms this.

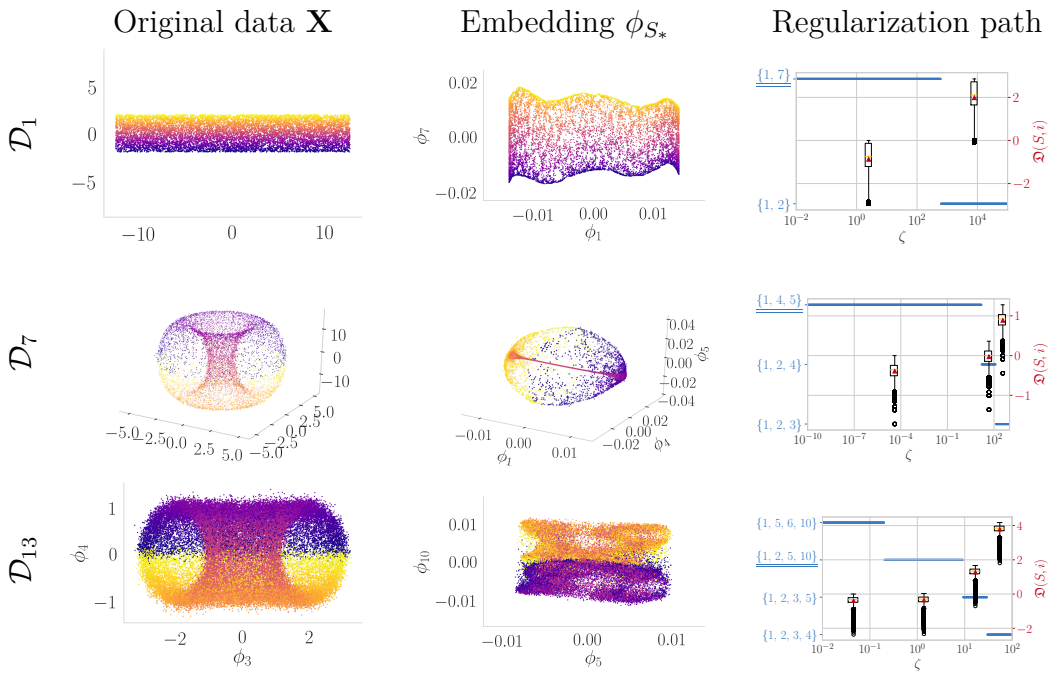

Figure 3.2: Experimental result for synthetic datasets. Rows correspond to different synthetic datasets (please refer to Table 3.2). Optimal subset S_* is selected by IndEigenSearch.

The right panel shows the box plot of $\{\mathfrak{D}(S,i)\}_{i=1}^n$. According to the proposed procedure, we eliminate $S_0 = \{1,2\}$ since $\mathfrak{D}(S_0,i) \geq 0$ for almost all the points.

Manifold with $s > d$. The second data $\mathcal{D}_7$ is displayed in the left panel of the second row. Due to the mechanism we used to generate the data, the resultant torus is non-uniformly distributed along the z axis. Middle panel is the embedding of the optimal coordinate set $S_* = \{1,4,5\}$ selected by IndEigenSearch. Note that the middle region (in red) is indeed a two dimensional narrow tube when zoomed in. The right panel indicates that both $\{1,2,3\}$ and $\{1,2,4\}$ (median is around zero) should be removed. The optimal regularization parameter is $\zeta_* \approx 7$. The result of the third dataset $\mathcal{D}_{13}$, *three torus*, is in the third row of the figure. We displayed only projections of the penultimate and the last coordinate of original data $\mathbf{X}$ and embedding ϕ_{S_*} (which is $\{5,10\}$) colored by α_1 of (3.11) in the left and middle panel to conserve space. A full combinations of coordinates can be found in Figure

3.10. The right panel implies one should eliminate the set $\{1, 2, 3, 4\}$ and $\{1, 2, 3, 5\}$ since both of them have more than 75% of the points such that $\mathfrak{D}(S, i) \geq 0$. The first remaining subset is $\{1, 2, 5, 10\}$, which yields an optimal regularization parameter $\zeta_* \approx 5$.

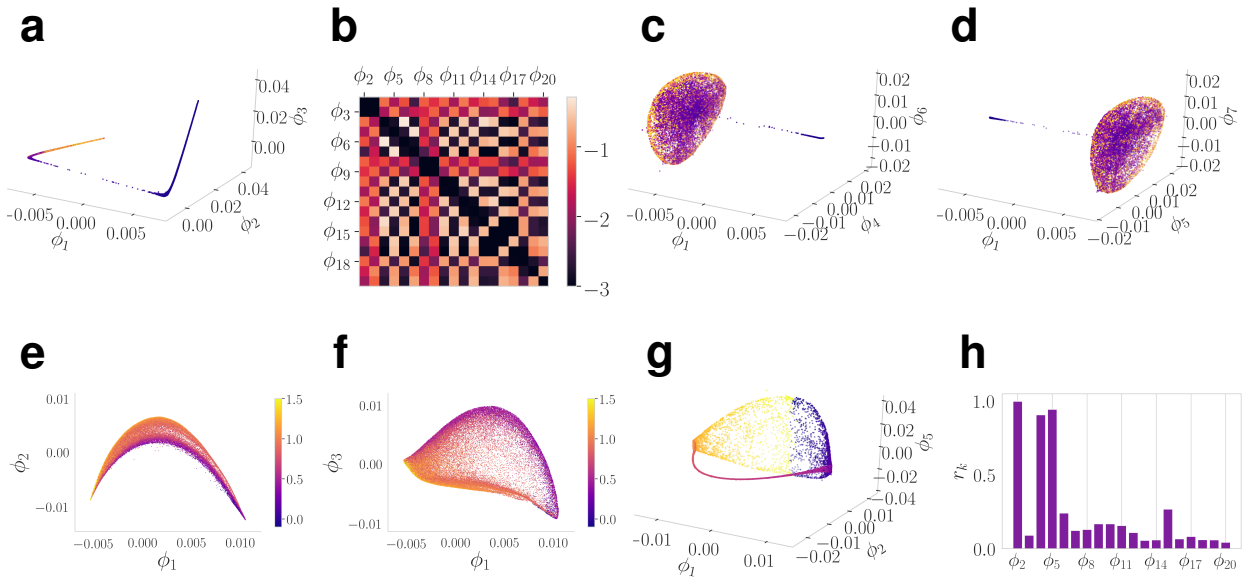

Figure 3.3: Analysis of the IES problem on real datasets. (a–d) Chloromethane dataset. (a) shows the embedding of the first three coordinates $\phi_{[3]}$, with ϕ_2 and ϕ_3 be the harmonic of ϕ_1. (b) is the loss $\mathfrak{L}(\{1, i, j\})$. (c) and (d) are embeddings with top two ranked subsets $S_1 = \{1, 4, 6\}$ and $S_2 = \{1, 5, 7\}$, colored by the distances between C and two different Cl^-, respectively. (e and f) are embeddings of $\boldsymbol{\phi}_{\{1,2\}}$ (suboptimal set) and $\boldsymbol{\phi}_{\{1,3\}}$ (maximizer of $\mathfrak{L}$) for the SDSS datasets, respectively. The loss values are $\mathfrak{L}(\{1, 2\}) = -1.24$ (for (e)) and $\mathfrak{L}(\{1, 3\}) = -0.39$ (for (f)). (g) Embedding with suboptimal choice of subset $S = \{1, 2, 5\}$ selected by LLRCoordSearch. (h) Leave one out error r_k versus coordinates ϕ_k.

3.4.2 Molecular dynamics dataset [48]

`SN2` is a molecular dynamics (MD) trajectory of the chloromethane SN2 reaction with different amounts of conformational degrees of freedom. In this reaction, two chloride substitute with each other in different configurations as described in the following chemical equation $CH_3Cl + Cl^- \longleftrightarrow CH_3Cl + Cl^-$. Due to the substitution of the chloride, we expect there are two clusters in this dataset.

A point in the dataset corresponds to a molecular configuration, which is recorded in the `xyz` format. More specifically, if the molecule has N atoms, then a configuration can be specified by an $N \times 3$ matrix representing the Euclidean coordinate of such configuration. To generate a point cloud $\mathbf{X}$ from a trajectory of configurations, we first preprocess the data by calculating two angles of every triplet of atoms (so that the translational and rotational symmetry is removed). Secondly, we remove the linear relation by keeping the top *principal components* (PCs) such that the unexplained variance ratio is less than 10^{-4}.

`SN2` has size $n \approx 30$k and ambient dimension $D = 40$, with the intrinsic dimension estimate being $\hat{d} = 2$. The embedding with coordinate set $S = [3]$ is shown in Figure 3.3a. The first three eigenvectors parameterize the same directions, which yields a one dimensional manifold in the figure. Top view ($S = [2]$) of the figure is a u-shaped structure similar to the yellow curve in Figure 3.1a. The heat map of $\mathfrak{L}(\{1, i, j\})$ for different combinations of coordinates in Figure 3.3b confirms that $\mathfrak{L}$ for $S = [3]$ is low and that ϕ_1, ϕ_2 and ϕ_3 give a low rank mapping. The heat map also shows high $\mathfrak{L}$ values for $S_1 = \{1, 4, 6\}$ or $S_2 = \{1, 5, 7\}$, which correspond to the top two ranked subsets. The embeddings with S_1, S_2 are in Figures 3.3c and 3.3d, respectively. In this case, we obtain two optimal S sets due to the data symmetry.

3.4.3 Galaxy spectra from the Sloan Digital Sky survey (SDSS) [1]

`GALAXY` is a dataset of galaxy spectra with the number of galaxies being $n' \approx 650$k. Each measurement represents the spectra of the visible light (350 nm–830 nm) of one galaxy; this results in the ambient dimension of each galaxy being $D = 3,750$. The first $n \approx 300$k points correspond to the spectra of the closer galaxies; we keep only these galaxies in this thesis to reduce the noise caused by the far-apart galaxies. Readers are encouraged to refer to McQueen et al. [77] for details regarding how to obtain, preprocess, and benchmark this dataset.

We display a random sample of $n = 50,000$ points from the first 300k closer galaxies. Figures 3.3e and 3.3f show that the first two coordinates are almost dependent; the embedding

with $S_* = \{1, 3\}$ is selected by IndEigenSearch with $d = 2$. Both plots are colored by the blue spectrum magnitude, which is correlated to the number of young stars in the galaxy, showing that this galaxy property varies smoothly and non-linearly with ϕ_1, ϕ_3, but is not smooth w.r.t. ϕ_1, ϕ_2.

3.4.4 Comparison with Dsilva et al. [44]

The LLRCoordSearch method outputs similar candidate coordinates as our proposed algorithm most of the time (see Table 3.3). However, the results differ for *high torus* as in Figure 3.3. Figure 3.3h is the leave one out (LOO) error r_k versus coordinates. The coordinates chosen by LLRCoordSearch was $S = \{1, 2, 5\}$, as in Figure 3.3g. The embedding is clearly shown to be suboptimal, for it failed to capture the cavity within the torus. This is because the algorithm searches in a sequential fashion; the noise eigenvector ϕ_2 in this example appears before the signal eigenvectors e.g., ϕ_4 and ϕ_5.

3.4.5 Additional experiments with real data

The results on additional real datasets are shown in Table 3.1. Not surprisingly, for most real data sets we examined, the independent coordinates are not the first s. They also show that the algorithm scales well and is robust to the noise present in real data.

Table 3.1: IES experiments on additional real datasets. Columns from left to right are sample size n, ambient dimension of data D, average degree of neighbor graph $\deg_{\text{avg}}$, (s, d) and runtime for IES, and the chosen set S^*, respectively. The last three datasets are from Chmiela et al. [29].

	n	D	$\deg_{\text{avg}}$	(s, d)	t (sec)	S_*
SDSS (full)	298,511	3750	144.91	(2, 2)	106.05	(1, 3)
Aspirin	211,762	244	101.03	(4, 3)	85.11	(1, 2, 3, 7)
Ethanol	555,092	102	107.27	(3, 2)	233.16	(1, 2, 4)
Malondialdehyde	993,237	96	106.51	(3, 2)	459.53	(1, 2, 3)

The asymptotic runtime of LLRCoordSearch has quadratic dependency on n, while for our algorithm is linear in n. Details of runtime analysis are in Section 3.5.1. LLRCoordSearch was too slow to be tested on the four larger datasets (see also Figure 3.4).

3.5 Discussions

3.5.1 Computational complexity analysis

For computation complexity analysis, we assume the embedding has already been obtained. Therefore, the computational complexity for building the neighbor graph and solving the eigenproblem of graph Laplacian can be omitted; this is also the case for LLRCoordSearch.

Co-metrics and orthogonal basis. According to Perraul-Joncas and Meila [95], the time complexity for computing $\mathbf{H}(i) \in \mathbb{R}^{m\times m}\ \forall\ i \in [n]$ is $\mathcal{O}(nm^2\nu)$, with ν be the average number of neighbors for each node in the neighbor graph $G(V, E)$. In manifold learning, the graph will be sparse therefore $\nu \ll n$. The complexity for obtaining principal space $\mathbf{U}(i)$ of point i via SVD will be $\mathcal{O}(m^3)$. The total time complexity will be $\mathcal{O}(nm^2\nu + nm^3)$.

Exact search. Evaluating the utility $\mathfrak{L}$ for each point i takes $\mathcal{O}(sd^2)$ in computing $\mathbf{U}_S(i)^\top\mathbf{U}_S(i)$ and $\mathcal{O}(d^3)$ in evaluating the determinant of a $d \times d$ matrix. The normalization step ($\mathfrak{R}_2$ term) takes $\mathcal{O}(ds)$. An exhaustive search over all the subset with cardinality s takes $\mathcal{O}\left(\binom{m}{s}\right)$. The total computational complexity will thus be $\mathcal{O}(nm^s(d^3+d^2s)+nm^2\nu+nm^3) = \mathcal{O}(nm^{s+3} + nm^2\nu)$.

Greedy algorithm. First step of the greedy algorithm includes solving $\text{argmax}_{S\subseteq[m];|S|=d}\,\mathfrak{L}(S, d)$, which takes $\mathcal{O}(nm^d d^3) = \mathcal{O}(nm^{d+3})$. Starting from $s = d+1 \to m$, each step includes exhaustively search over $m - s$ candidates, with the time complexity of evaluating $\mathfrak{L}$ be $n(d^3 + d^2 s)$.

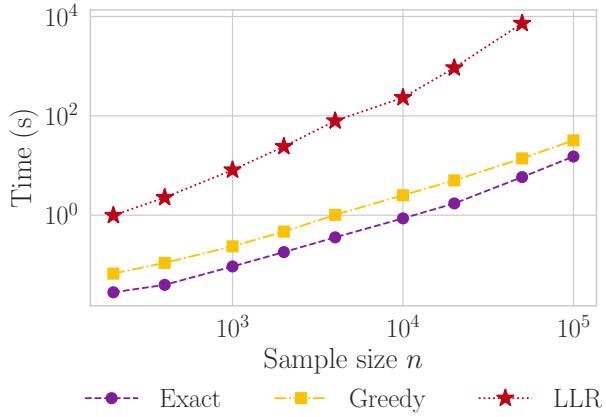

Figure 3.4: Runtimes of different IES algorithms on two dimensional long strip. Purple, yellow and red curves correspond to IndEigenSearch, GreedyIndEigenSearch and LLRCoordSearch algorithm, respectively.

Putting things together, one has the second part of the greedy algorithm be

$$\sum_{s=d}^{m} n(m-s)(d^3+d^2s) = \mathcal{O}(nm^5).$$

Therefore, the total computational complexity will be $\mathcal{O}(n(m^{d+3}+m^5+m^2\nu))$.

The algorithm by Dsilva et al. [44]. The Algorithm LLRCoordSearch is summarized in Algorithm 3.4. For searching over a fixed coordinate s, the algorithm first builds a kernel for local linear regression by constructing a neighborhood graph, which takes $\mathcal{O}(n\log(n)s)$[1] using approximate nearest neighbor search. The s dependency comes from the dimension of the feature. For each point i, a ordinary least square (OLS) problem is solved, which results in $\mathcal{O}(n^2s^2+ns^3)$ time complexity. Searching from $s=2\to m$ will make the total time complexity be

$$\sum_{s=2}^{m} n^2s^2+ns^3+ns\log n = \mathcal{O}(n^2m^3+nm^4).$$

[1]This is a simplified lower bound, see Dasgupta and Sinha [35] for details.

Algorithm 3.4: LLRCoordSearch: parsimonious embedding by Dsilva et al. [44]

Input : Embedding $\mathbf{Y} = [\phi_1, \cdots, \phi_m] \in \mathbb{R}^{n \times m}$

Set the leave-one-out validation error $\mathbf{r} = [1, \cdots, 1] \in \mathbb{R}^m$

for $s = 2 \to m$ **do**

  Bandwidth of LLR: $h \leftarrow \frac{1}{3} \cdot \text{Median}(\text{PairwiseDist}(\phi_{[s-1]}))$

  $\hat{\phi}_s \leftarrow \text{LocalLinearRegression}(\phi_s, \phi_{[s-1]}, h)$

  $r_s = \sqrt{\frac{\|\hat{\phi}_s - \phi_s\|^2}{\|\phi_s\|^2}}$

$S_* \leftarrow \text{ArgSort}(\mathbf{r})$

▷ `Sort in descending order.`

Return : Sorted independent coordinates S_*

Empirical runtimes. For a sparse graph, the overheads of the IndEigenSearch and GreedyIndEigenSearch algorithms come from the enumeration of the subset S. Because of the linear dependency on the sample size n, the algorithm is tractable for small s and d. However, LLRCoordSearch has a quadratic dependency on the sample size n, which is more computationally intensive if n is large. For large s and d, one can use the techniques in the difference between submodular function optimization (e.g. Iyer and Bilmes [61]) because $\mathfrak{R}_1$ and $\mathfrak{R}_2$ are submodular set function (Theorems 3.2 and 3.3). An empirical runtime plot for different algorithms can be found in Figure 3.4. The runtime is evaluated on two dimensional long strip with $s = d = 2$ and is performed on a single desktop computer running Linux with 32GB RAM and a 8-Core 4.20GHz Intel® Core™ i7-7700K CPU.

3.5.2 A discussion on UMAP

UMAP [76] is a commonly used data visualization alternative of t-SNE. The authors proposed to use the spectral embedding of the graph Laplacian as an initialization to the algorithm for faster convergence (compared to random initialization). In this section, we showed empirically that (i) given reasonable computing resources, the IES problem also appears in the UMAP embedding; additionally, (ii) by initializing with spectral embedding with carefully

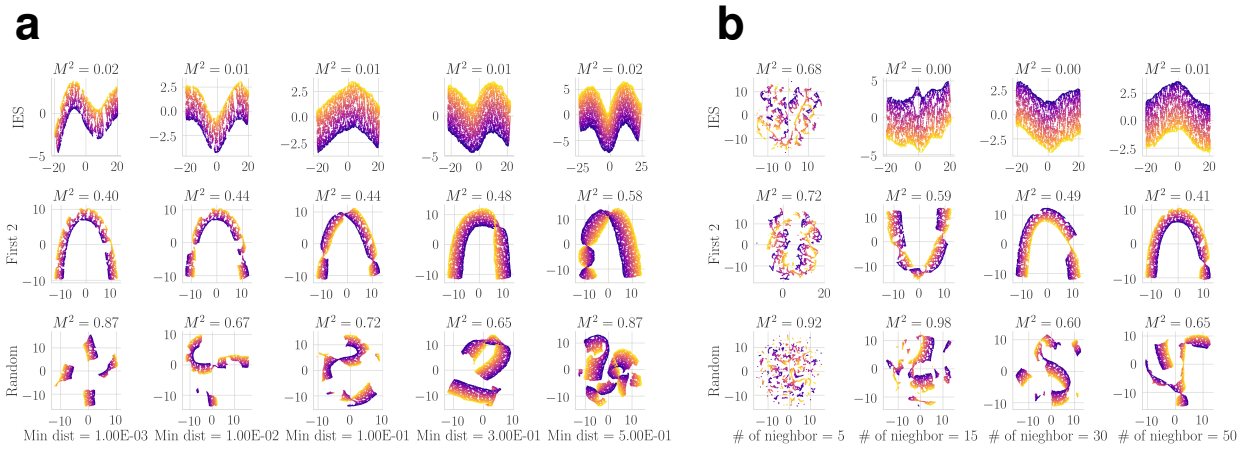

Figure 3.5: UMAP embeddings of 2D long stripe with different initializations and choices of hyper-parameters. Rows from top to bottom correspond to UMAP embedding initialized with DM which coordinates chosen by IndEigenSearch, naïve DM and random initialization, respectively. Columns represent different choices of (a) points separation and (b) number of neighbors.

selected coordinate set chosen by IndEigenSearch, one can obtain a faster convergence and a globally interpretable embedding. Figure 3.5 is the UMAP embedding of 2D long stripe dataset $\mathcal{D}_1$ with different choices of hyper-parameters (points separation in Figure 3.5a and number of neighbors in Figure 3.5b), with total number of epochs be 500. The first row of both plots are the embedding initialized with IndEigenSearch, the second row corresponds to those initialized with naïve DM. The embeddings in the third row are initialized randomly. As shown in the results, algorithmic/random artifacts can be easily seen in the embeddings with Naïve DM or random initialization (2nd and 3rd rows). More precisely, unwanted patterns which reduce the interpretability of the embeddings, e.g., the "knots" in the second row or disconnected components in the second/third rows, are generated. The sum of square procrustes error M^2 between ground truth dataset and the embedding shown on each subplots also confirm our statement. Note that it is possible to unroll the algorithmic artifact with more epochs in the sampling steps of UMAP. (Three to five times more iterations are needed in this example.) However, due to the efficiency of performing IndEigenSearch, it

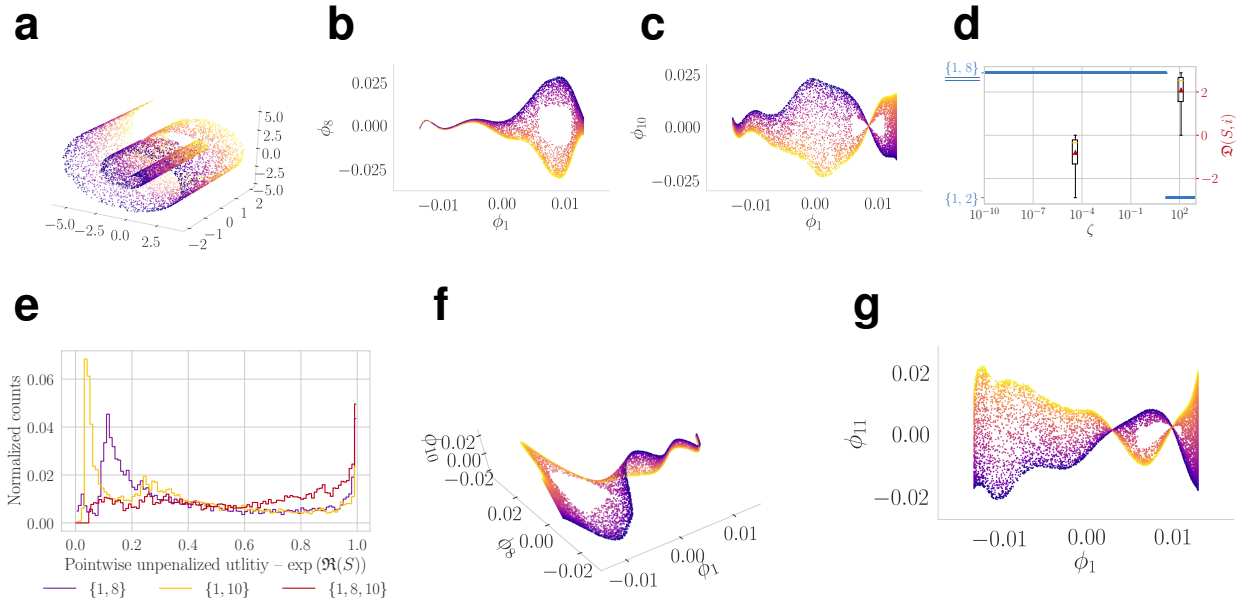

Figure 3.6: A heuristic to determine whether s is sufficiently large. (a) Original data of $\mathcal{D}_4$, *swiss roll with hole* dataset. Embeddings with coordinate subset to be (b) $S = \{1, 8\}$, (c) $S = \{1, 10\}$, (f) $S = \{1, 8, 10\}$ and (g) $S = \{1, 11\}$ on $\mathcal{D}_4$. (e) Histogram of point-wise normalized projected volume on $\mathcal{D}_4$ for top two ranking of subsets (purple and yellow) and the union of two sets (red) obtain from IndEigenSearch algorithm.

is beneficial to initialize with the embedding selected by IndEigenSearch.

3.5.3 A heuristic to determine whether s is sufficiently large

In this section, we propose a heuristic method to determine whether the given s is large enough. Our method is based on the histogram of $\text{Vol}_{\text{norm}}(S, i) = \exp(\mathfrak{R}(S, i))$, the *normalized projected volume* of each point i. Recall that this volume is bounded between 0 and 1. Ideally, a perfect choice of cardinality $|S|$ will result in a concentration of mass in larger Vol_{norm} region. The heuristic works as follow: at first we check the histogram of unpenalized Vol_{norm} on the top few ranked subsets in terms of $\mathfrak{L}$. If spikes in the small Vol_{norm} regions are witnessed in the histogram, taking the union of the subsets and inspecting the histogram of unpenalized Vol_{norm} on the combined set again. If spikes in small Vol_{norm} region diminished,

one can conclude that a larger cardinality size $|S|$ is needed for such manifold.

We illustrate the idea on *swiss roll with hole* dataset in Figure 3.6a. Figure 3.6b is the optimal subset of coordinates $S_* = \{1, 8\}$ selected by the proposed algorithm that best parameterize the underlying manifold. Figure 3.6d suggests one should eliminate $S_0 = \{1, 2\}$ because $\mathfrak{D}(S_0, i) \geq 0$ for all the points by ReguParamSearch. However, as shown in Figure 3.6b, though it has low frequency and having rank 2 for most of the places, set $\{1, 8\}$ might not be suitable for data analysis for the very thin arms in left side of the embedding. Figure 3.6e is the histograms of the point-wise unpenalized Vol_{norm} on different subsets. Purple and yellow curves correspond to the histogram of top two ranked subsets S from IndEigenSearch. Both curves show a concentration of masses in small Vol_{norm} region. The histogram of point-wise unpenalized Vol_{norm} on $\{1, 8, 10\}$ (red curve), which is the union of the aforementioned two subsets, shows less concentration in the small Vol_{norm} region and implies that $|S| = 3$ might be a better choice for data analysis. Figure 3.6f shows the embedding with coordinate $S = \{1, 8, 10\}$, which represents a two dimensional strip embedded in three dimensional space. The thin arc in Figure 3.6b turns out to be a collapsed two dimensional manifold via projection, as shown in the upper right part of Figure 3.6f and left part of Figure 3.6c. Here we have to restate that the embedding in Figure 3.6b, although is a *degenerated* embedding, is still the best set one can choose for $s = 2$ such that the embedding varies slowest and has rank 2. However, choosing $s = 3$ might be better for data analysis.

3.6 Summary

Algorithms that use eigenvectors, such as DM, are among the most promising and well studied in ML. It is known since Goldberg et al. [51] that when the aspect ratio of a low dimensional manifold exceeds a threshold, the choice of eigenvectors becomes non-trivial and that this threshold can be as low as 2. Experimental results in this chapter confirm the need to augment ML algorithms with IES methods to successfully apply ML to real-world problems. Surprisingly, the IES problem has received little attention in the ML literature to

the extent that the difficulty and complexity of the problem have not been recognized. Our method advances state of the art by (i) introducing for the first time a differential geometric definition of the problem; (ii) we highlighting geometric factors such as injectivity radius that, in addition to the aspect ratio, influence the number of eigenfunctions needed for a smooth embedding. Furthermore, we (iii) construccollt selection criteria based on *intrinsic manifold quantities* (iv) that have analyzable asymptotic limits, (v) can be computed efficiently, and (vi) are robust to the noise present in real scientific datasets.

3.7 Appendix–IES problem on all synthetic manifolds

In this chapter, a total of 13 different synthetic manifolds are considered. Table 3.2 summarized the synthetic manifolds constructed and its abbreviations (from $\mathcal{D}_1$ to $\mathcal{D}_{13}$). Embedding results for the synthetic manifolds are in Figures 3.7, 3.8 and 3.10. The ranking of the first few candidate sets S from IndEigenSearch, GreedyIndEigenSearchand LLRCoordSearch can be found in Table 3.3. The table shows the optimal subsets return by three different algorithms are often time the same, with exception for $\mathcal{D}_7$ *high torus* as discussed in Section 6.4.

Table 3.2: Abbreviations for different synthetic manifolds in this chapter. The abbreviation with asterisk represents such dataset is discussed in Figure 3.2.

	Manifold with $s = d$
$\mathcal{D}_1^*$	Two dimensional strip (aspect ratio 2π)
$\mathcal{D}_2$	2D strip with cavity (aspect ratio 2π)
$\mathcal{D}_3$	Swiss roll
$\mathcal{D}_4$	Swiss roll with cavity
$\mathcal{D}_5$	Gaussian manifold
$\mathcal{D}_6$	Three dimensional cube
	Manifold with $s > d$
$\mathcal{D}_7^*$	High torus
$\mathcal{D}_8$	Wide torus
$\mathcal{D}_9$	z-asymmetrized high torus
$\mathcal{D}_{10}$	x-asymmetrized high torus
$\mathcal{D}_{11}$	z-asymmetrized wide torus
$\mathcal{D}_{12}$	x-asymmetrized wide torus
$\mathcal{D}_{13}^*$	Three-torus

Table 3.3: Results returned from different algorithms on different synthetic datasets.

	Exact search					Greedy rank	LLR rank
	1	2	3	4	5		
$\mathcal{D}_1$	[1, 7]	[1, 8]	[1, 9]	[1, 10]	[1, 12]	[1, 7, 6, 4, 3, 2, 5]	[1, 7, 14, 16, 11, 18, 6]
$\mathcal{D}_2$	[1, 4]	[1, 8]	[1, 9]	[1, 10]	[1, 12]	[1, 4, 8, 6, 5, 3, 2]	[1, 4, 8, 5, 17, 11, 14]
$\mathcal{D}_3$	[1, 9]	[1, 10]	[1, 11]	[1, 13]	[1, 18]	[1, 9, 5, 2, 3, 4, 6]	[1, 9, 19, 16, 12, 10, 4]
$\mathcal{D}_4$	[1, 8]	[1, 10]	[1, 11]	[1, 14]	[1, 15]	[1, 8, 3, 2, 4, 10, 5]	[1, 8, 11, 10, 19, 16, 4]
$\mathcal{D}_5$	[1, 6]	[1, 8]	[1, 10]	[1, 11]	[1, 13]	[1, 6, 2, 8, 3, 10, 4]	[1, 6, 19, 8, 18, 14, 12]
$\mathcal{D}_6$	[1, 2, 8]	[1, 2, 11]	[1, 4, 8]	[1, 2, 17]	[1, 2, 13]	[1, 2, 8, 3, 4, 6, 5]	[1, 2, 8, 10, 3, 13, 6]
$\mathcal{D}_7$	[1, 4, 5]	[1, 4, 8]	[1, 5, 7]	[1, 7, 12]	[1, 7, 8]	[1, 5, 4, 3, 6, 2, 8]	[1, 2, 5, 4, 15, 6, 10]
$\mathcal{D}_8$	[1, 2, 7]	[1, 4, 7]	[1, 3, 7]	[1, 2, 9]	[1, 5, 7]	[1, 7, 2, 4, 3, 13, 5]	[1, 2, 7, 13, 12, 15, 14]
$\mathcal{D}_9$	[1, 3, 4]	[1, 3, 7]	[1, 4, 6]	[1, 3, 10]	[1, 7, 9]	[1, 3, 4, 2, 9, 7, 6]	[1, 3, 4, 2, 19, 8, 7]
$\mathcal{D}_{10}$	[1, 2, 4]	[1, 3, 4]	[1, 4, 5]	[1, 6, 9]	[1, 6, 14]	[1, 4, 2, 3, 5, 6, 8]	[1, 4, 2, 3, 8, 5, 6]
$\mathcal{D}_{11}$	[1, 2, 5]	[1, 4, 8]	[1, 4, 5]	[1, 8, 9]	[1, 2, 8]	[1, 5, 2, 4, 8, 3, 9]	[1, 2, 5, 8, 10, 9, 11]
$\mathcal{D}_{12}$	[1, 2, 5]	[1, 4, 5]	[1, 2, 7]	[1, 3, 5]	[1, 2, 8]	[1, 5, 2, 3, 4, 6, 8]	[1, 5, 2, 6, 10, 9, 4]
$\mathcal{D}_{13}$	[1, 2, 5, 10]	[1, 3, 5, 10]	[1, 4, 5, 10]	[1, 5, 6, 10]	[1, 2, 8, 10]	[1, 5, 10, 2, 4, 3, 6]	[1, 2, 10, 5, 14, 15, 16]

Synthetic manifolds with $s = d$. Below summarized the details of generating the datasets.

1. $\mathcal{D}_1^*$: points from this dataset are sampled uniformly from $\mathbf{x}_i \sim \textsc{Unif}([-2,2]\times[-4\pi,4\pi])$.

2. $\mathcal{D}_2$: points are first sampled uniformly from $[-2,2] \times [-4\pi, 4\pi]$. Points i are removed if $|X_{i1}| < 4\pi/3$ and $|X_{i2}| < 2/3$.

3. $\mathcal{D}_3$: first sampling points $\mathbf{X}_{\text{true}} = [\mathbf{x}_0, \mathbf{y}_0]$ uniformly from a two dimensional strip. The data $\mathbf{X}$ can be obtained by the following non-linear transformation.

$$\mathbf{X} = \left[\frac{\mathbf{x}_0 \circ \cos \mathbf{x}_0}{2}, \mathbf{y}_0, \frac{\mathbf{x}_0 \circ \sin \mathbf{x}_0}{2}\right],$$

   with $\circ$ denotes Hadamard (element-wise) product.

4. $\mathcal{D}_4$: sampling points $\mathbf{X}_{\text{true}} = [\mathbf{x}_0, \mathbf{y}_0]$ uniformly from 2D strip with cavity then applying the transformation (3) to get $\mathbf{X}$.

5. $\mathcal{D}_5$: sampling points $\mathbf{X}_{\text{true}}$ uniformly from ellipse $\left\{(x,y) \in \mathbb{R}^2 : \left(\frac{x}{6}\right)^2 + \left(\frac{y}{2}\right)^2 = 1\right\}$. The data is obtained by

$$\mathbf{X} = [\mathbf{X}_{\text{true}}, \mathbf{z}],$$

   with $z_i = \exp\left(-\left(\left(\frac{X_{i1}}{3}\right)^2 + X_{i2}^2\right)/2\right)$

6. $\mathcal{D}_6$: points are sampled uniformly from $[-1,1] \times [-2,2] \times [-4,4]$.

The experimental results are in Figure 3.7 ($\mathcal{D}_4$ in Figure 3.6).

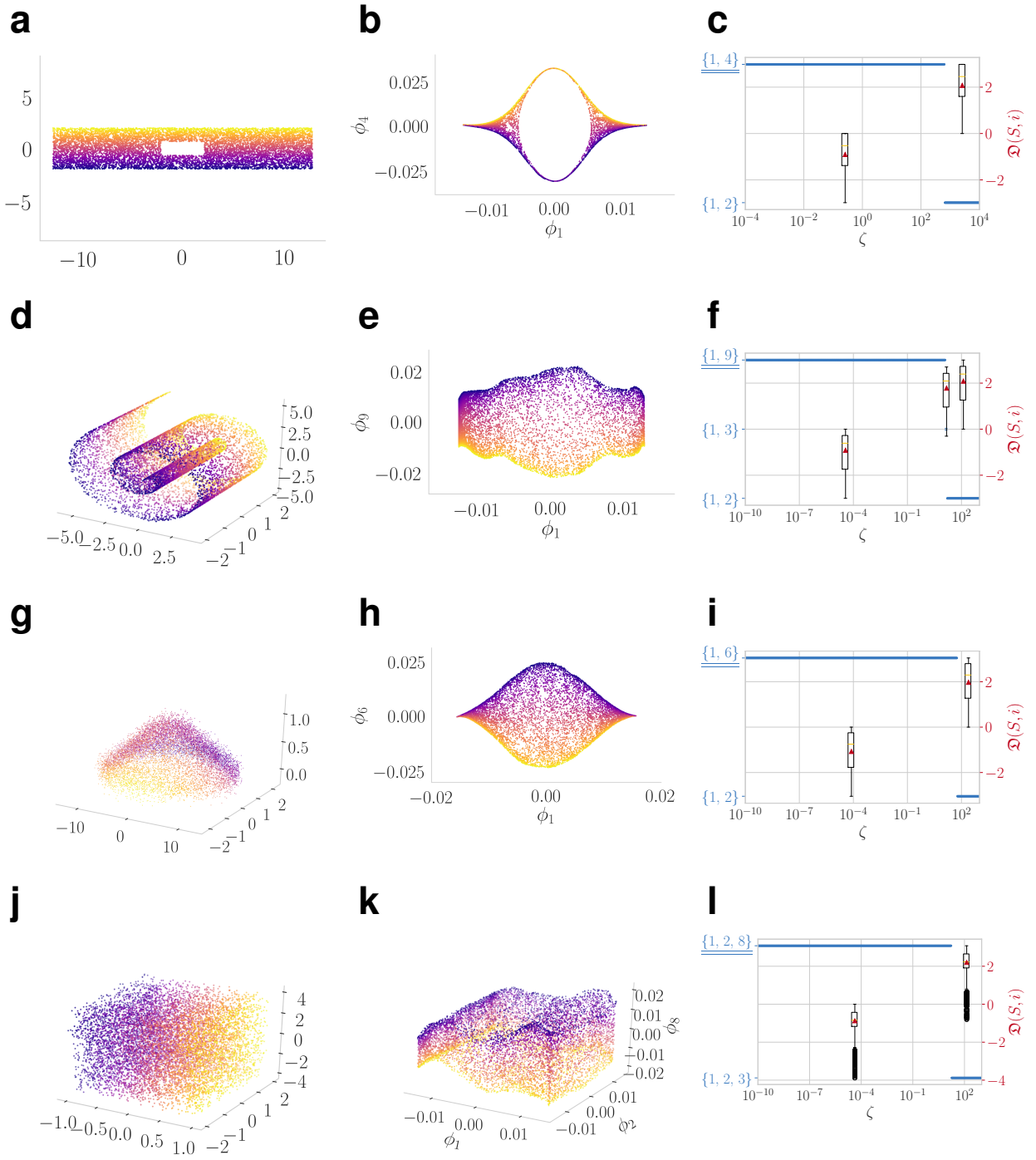

Figure 3.7: Synthetic manifolds with minimum embedding dimension s equals intrinsic dimension d. Rows from top to bottom represent $\mathcal{D}_2$ *two dimensional strip with cavity* whose aspect ratio is $W/H = 2\pi$ (a–c), $\mathcal{D}_3$ *swiss roll* (d–f), $\mathcal{D}_5$ *gaussian manifold* (**g**–**i**), and $\mathcal{D}_6$ *three dimensional cube* dataset (j–l), respectively. Columns from left to right are the original data $\mathbf{X}$, embedding ϕ_{S_*} with optimal coordinate sets S_* chosen by IndEigenSearch and the regularization path, respectively.

Tori and asymmetrized tori. A torus can be parametrized by

$$\begin{aligned} x_1 &= (r_2 + r_1 \cos\theta_1)\cos\theta_2; \\ x_2 &= (r_2 + r_1 \cos\theta_1)\sin\theta_2; \\ x_3 &= h\sin(\theta_2). \end{aligned} \tag{3.8}$$

1. $\mathcal{D}_7^*$: sampling θ_1, θ_2 uniformly from $[0, 2\pi)$ and generating the torus with $(r_1, r_2, h) = (2, 3, 8)$ from (3.8)

2. $\mathcal{D}_8$: generating the torus with $(r_1, r_2, h) = (2, 10, 2)$

3. $\mathcal{D}_9$: generating a high torus with $(r_1, r_2, h) = (2, 3, 8)$ and applying the following transformation

$$z \leftarrow (z - \min(z))^\gamma / \xi, \tag{3.9}$$

   with $(\gamma, \xi) = (3, 1500)$

4. $\mathcal{D}_{10}$: generating a high torus with $(r_1, r_2, h) = (2, 3, 8)$ and applying the following transformation

$$x \leftarrow (x - \min(x))^\kappa / \eta, \tag{3.10}$$

   with $(\kappa, \eta) = (2, 10)$

5. $\mathcal{D}_{11}$: generating a wide torus with $(r_1, r_2, h) = (2, 10, 2)$ and applying transformation (3.9) with $(\gamma, \xi) = (3, 50)$

6. $\mathcal{D}_{12}$: generating a wide torus with $(r_1, r_2, h) = (2, 10, 2)$ and applying transformation (3.10) with$(\kappa, \eta) = (3, 1000)$

The experimental results are in Figures 3.8 and 3.9.

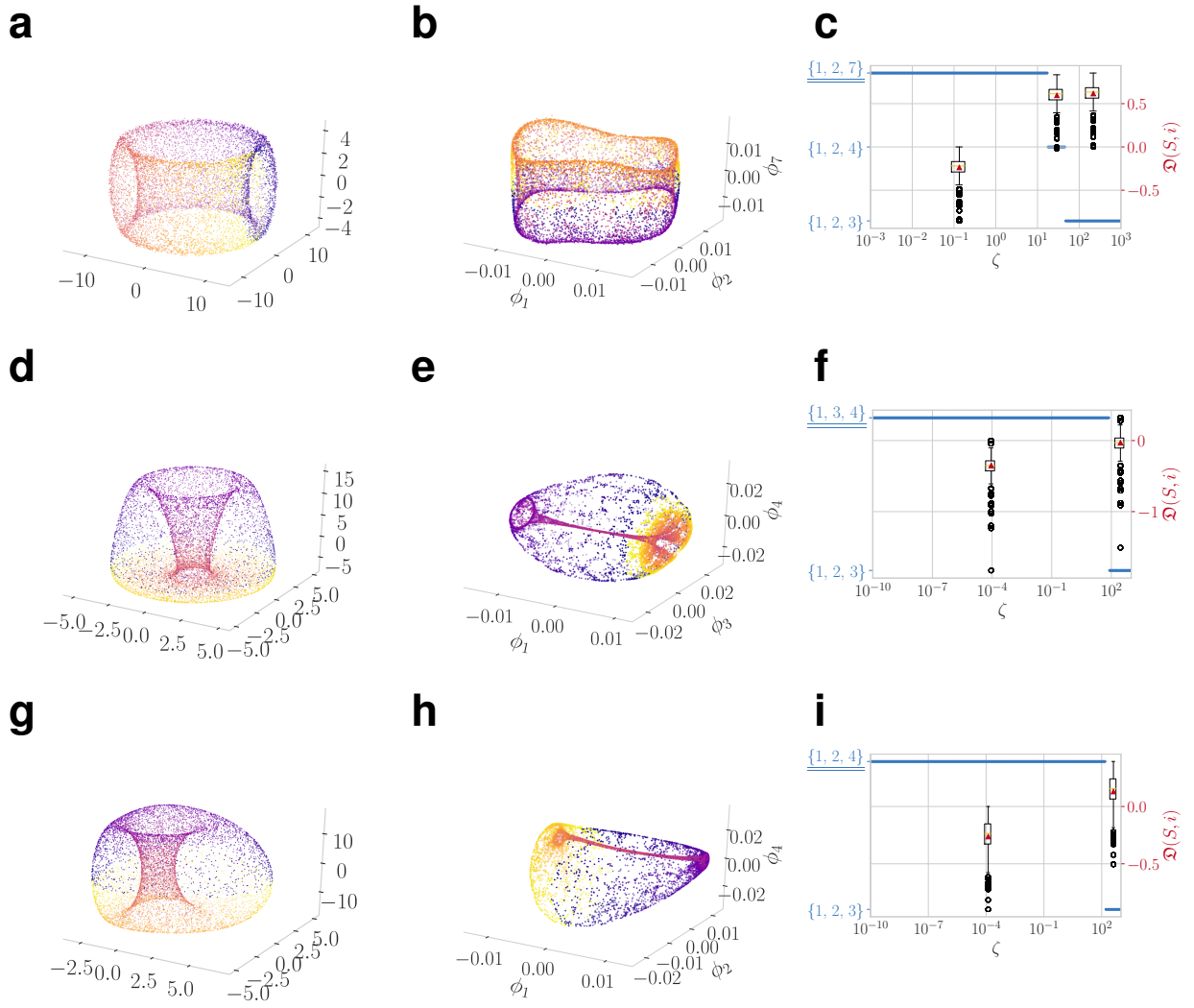

Figure 3.8: Synthetic manifolds with minimum embedding dimension s greater than intrinsic dimension d. Rows from top to bottom represent $\mathcal{D}_8$ *wide torus* (a–c), $\mathcal{D}_9$ *z-asymmetrized high torus* (d–f), and $\mathcal{D}_{10}$ *x-asymmetrized high torus* (g–i), respectively. Columns from left to right are the original data $\mathbf{X}$, embedding ϕ_{S_*} with optimal coordinate sets S_* chosen by IndEigenSearch and the regularization path, respectively.

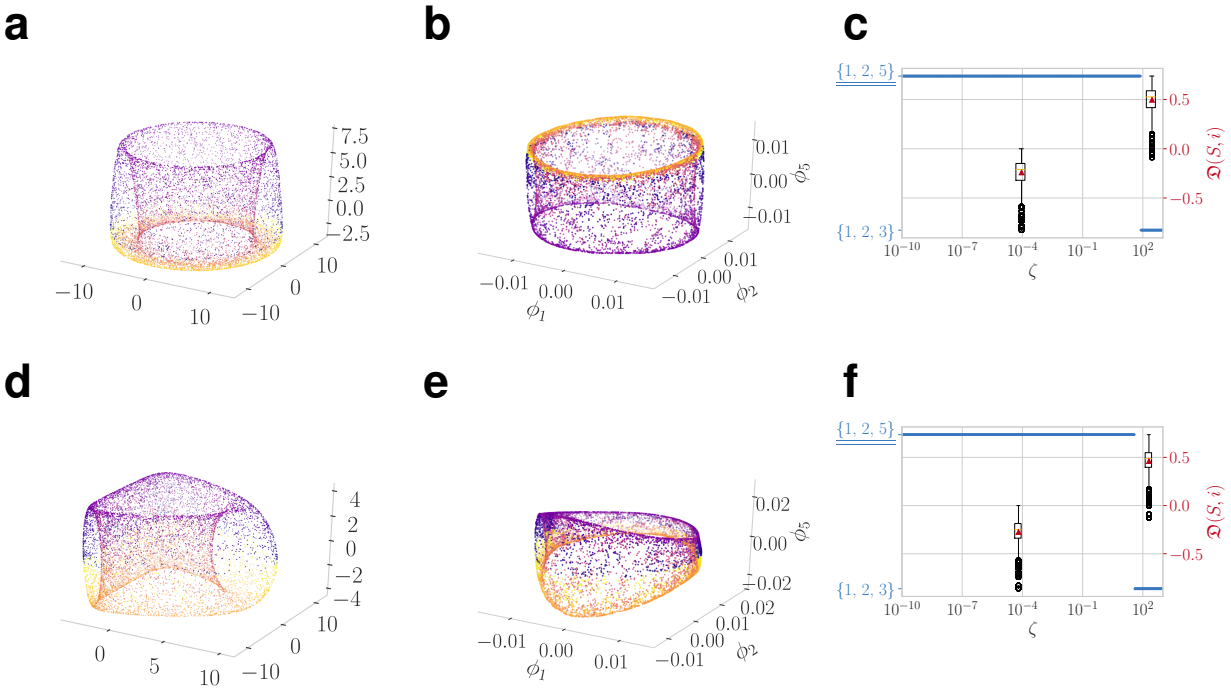

Figure 3.9: Synthetic manifolds with minimum embedding dimension s greater than intrinsic dimension d. Rows from top to bottom represent $\mathcal{D}_{11}$ *z-asymmetrized wide torus* (a–c) and $\mathcal{D}_{12}$ *x-asymmetrized wide torus* (d–f), respectively. Columns from left to right are the original data $\mathbf{X}$, embedding ϕ_{S_*} with optimal coordinate sets S_* chosen by IndEigenSearch and the regularization path, respectively.

Three-torus. The parameterization of the three torus is

$$\begin{aligned} x_1 &= (r_3 + (r_2 + r_1 \cos\theta_1)\cos\theta_2)\cos\theta_3; \\ x_2 &= (r_3 + (r_2 + r_1 \cos\theta_1)\cos\theta_2)\sin\theta_3; \\ x_3 &= (r_2 + r_1 \cos\theta_1)\sin\theta_2; \\ x_4 &= r_1 \sin\theta_1. \end{aligned} \tag{3.11}$$

To generate $\mathcal{D}_{13}$, we sample $\boldsymbol{\alpha}_k$ uniformly from $[0, 2\pi)$ for $k \in [3]$ and apply the transformation (3.11) with $(r_1, r_2, r_3) = (8, 2, 1)$. The sample size for this dataset is $n = 50,000$. The experimental result of three-torus can be found in Figure 3.10.

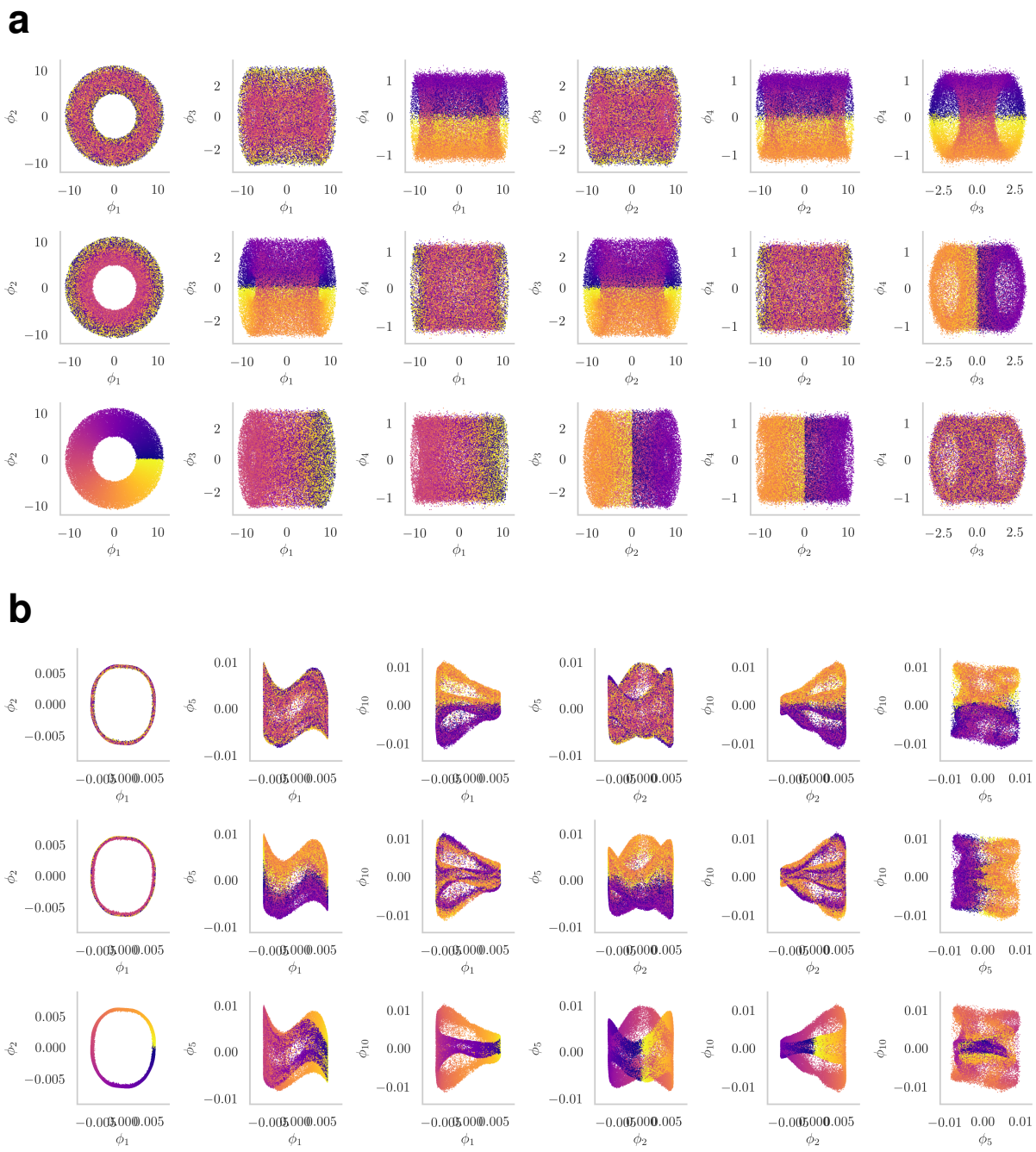

Figure 3.10: IES experiment on $\mathcal{D}_{13}$. (a) Original data $\mathbf{X}$ of three torus. (b) Embedding ϕ_{S_*} with optimal coordinate sets S_* chosen by IndEigenSearch. Rows for both (a) and (b) from top to bottom are embedding colored by the parameterization $(\alpha_1, \alpha_2, \alpha_3)$ in (3.11), respectively.

3.7.1 Verification of the chosen subsets on synthetic manifolds

Unlike 2D strip, the close form solution of the optimal set is oftentimes unknown in general. In this section, we verify the correctness of the chosen subset by reporting the full procrustes distance (disparity score) M^2 [43], which is defined to be the normalized sum of square of the point-wise difference between the procrustes transformed ground truth data $\mathbf{X}_{\text{true}} \in \mathbb{R}^{n\times k}$ and the test data $\mathbf{X}_{\text{test}} \in \mathbb{R}^{n\times k}$. Namely,

$$M^2(\mathbf{X}_{\text{true}}, \mathbf{X}_{\text{test}}) = \min_{\beta,\boldsymbol{\gamma},\boldsymbol{\Gamma}} \|\mathbf{X}_{\text{true}} - \beta \mathbf{X}_{\text{test}}\boldsymbol{\Gamma} - \mathbf{1}_n\boldsymbol{\gamma}^\top\|_F^2,$$
$$\text{s.t. } \beta > 0, \boldsymbol{\gamma} \in \mathbb{R}^k, \Gamma \in SO(k).$$

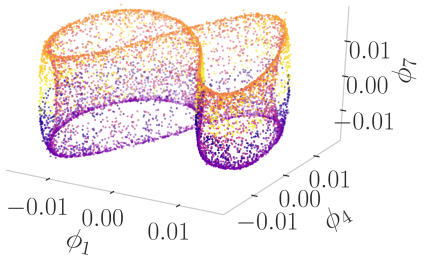

Figure 3.11: An example of an embedding $\phi_{\{1,4,7\}}$ of $\mathcal{D}_8$ that has *crossing*.

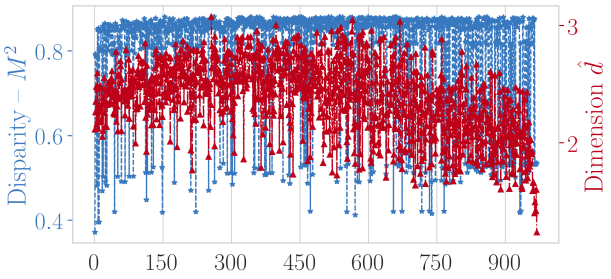

Figure 3.12: Disparity score M^2 and the estimated dimension $\hat{d}$ v.s. ranking of sets based on $\mathfrak{L}$ in (3.1) for $\mathcal{D}_{13}$

Here β is a scale parameter, $\boldsymbol{\gamma}$ is the centering parameter and $\boldsymbol{\Gamma}$ is a $k \times k$ rotation matrix. We further require $\|\mathbf{X}_{\text{true}}\|_F = 1$ so that the disparity score will be between 0 and 1. Intuitively, one can expect the optimal choice of eigencoordinates S_* will yield a small disparity score $M^2(\mathbf{X}_{\text{true}}, \phi_{S_*})$, with score increases as the coordinate set S contains duplicate parameterizations or ϕ_S contains *knots, crossings*, etc. (e.g., Figure 3.11). Note that the score can only be calculated when the ground truth data $\mathbf{X}_{\text{true}}$ is available. For dataset without obtainable ground truth, one cannot proposed to report the disparity score of ϕ_S and the

original data $\mathbf{X}$ as the proxy of $\mathbf{X}_{\text{true}}$, for $\mathbf{X}$ might not be a affine transformation of $\mathbf{X}_{\text{true}}$, e.g., Swiss roll. Besides, small M^2 given ϕ_S does not imply S is optimal, which will be clear in the discussion of Figure 3.6g. Besides disparity scores, we will also report the estimated dimension $\hat{d}$. One can expect the estimated dimension for the optimal set $\dim(\phi_{S_*})$ will be close to the intrinsic dimension d, while the estimated dimension for sets containing duplicate parameterizations will be smaller than the intrinsic dimension. One cannot propose to use it as a criterion to choose the optimal set, for the suboptimal sets can also have estimated dimensions closed to the intrinsic dimension, e.g., Figure 3.3g. Throughout the experiment, the dimension estimation method by Levina and Bickel [72] is used for its ability to estimate dimension among all candidate subsets fairly fast. Blue and red curves in Figures 3.13 and 3.12 show the disparity scores and estimated dimensions versus ranking of coordinate subsets for different synthetic manifolds, respectively. As expected, we have an increasing in M^2 and decreasing in $\hat{d}$ with respect to ranking. We first highlight that the set that produces the lowest disparity score is not necessarily optimal, although S_* does yield a small disparity. This can be shown in the example of $\mathcal{D}_4$ *swiss roll with hole* dataset. Figure 3.6g is the embedding ϕ_{S_3} of $\mathcal{D}_4$, with S_3 is ranked third subset in terms of $\mathfrak{L}(S;\zeta)$, that minimizes the disparity score M^2 in $\mathcal{D}_4$ as shown in Figure 3.13d. This is because the embedding of the subset $S_3 = \{1, 11\}$ has larger area on the left, compared to Figure Figure 3.6b. This balances out the high disparity caused by the *flipped* region between two *knots* in the embedding ϕ_{S_3} when matched with $\mathbf{X}_{\text{true}}$. Since all the ranked first subset has low disparity compared to other subsets, we have higher confidence saying that the ranked 1st subset is indeed the optimal choice for the synthetic manifolds.

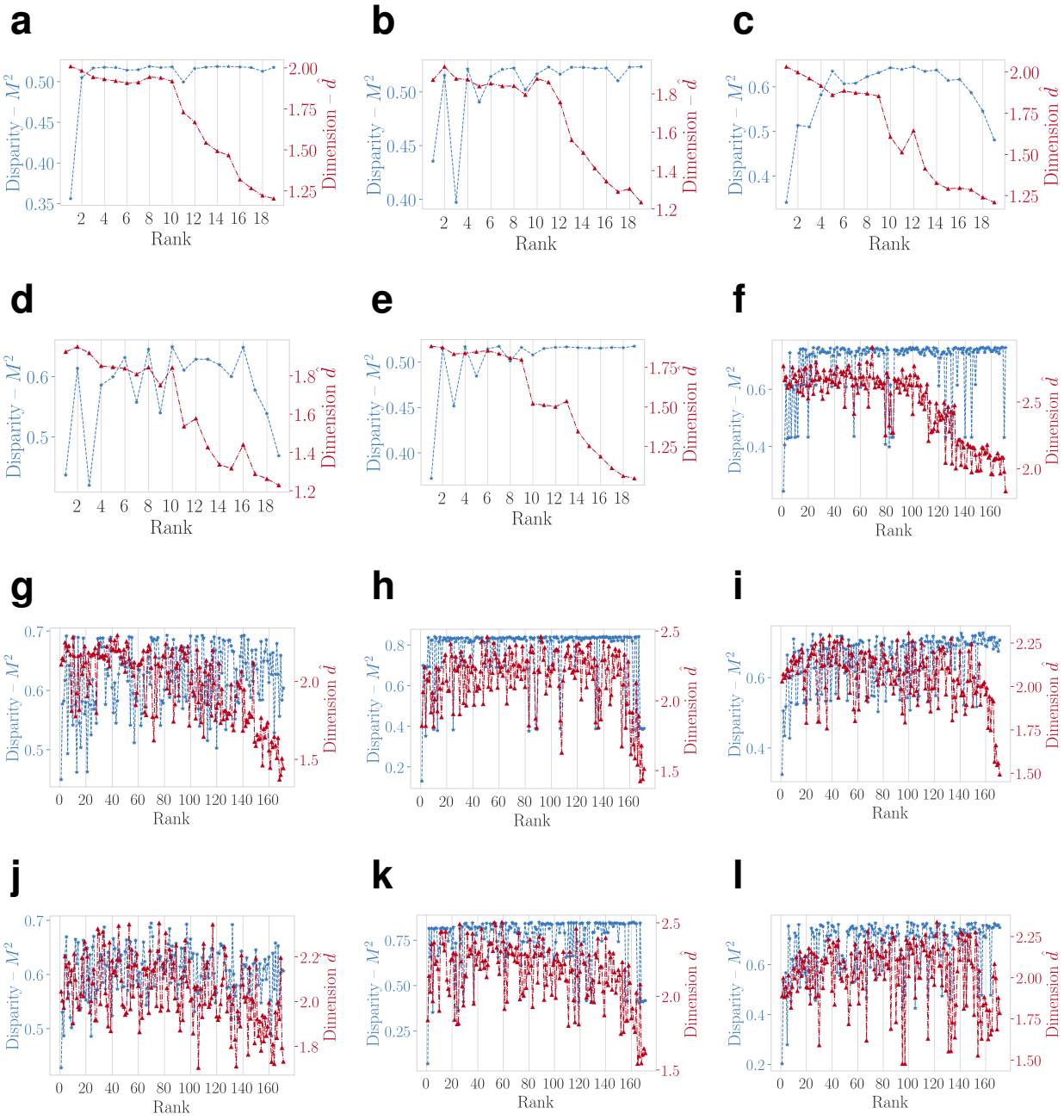

Figure 3.13: (a–l) Verification of the correctness of the chosen sets in synthetic manifolds $\mathcal{D}_1$–$\mathcal{D}_{12}$, respectively.

Chapter 4

THE ESTIMATION OF DISCRETE HELMHOLTZIAN

Laplacians are known to be intimately tied to a manifold's topology and geometry. As discussed in the previous chapter, the *Laplace-Beltrami* operator Δ_0, an operator acting on functions (0-forms), is well studied and pivotal in classical manifold learning as we have discussed in previous chapters; however, estimating the 1-Laplacian Δ_1, an operator acting on vector fields (1-forms), for a manifold has rarely been attempted yet.

The core concept we use in this chapter is the *k-Laplacian matrix* [46] (introduced in Chapter 2), which is a discrete analogue of the continuous k-Hodge Laplacian operator Δ_k. The beauty of the aforementioned framework generated numerous applications in areas such as numerical analysis [5, 40], edge flow learning on graphs [62, 105], pairwise ranking [63], and game theory [20]. Unfortunately, these works mainly focused on abstract graph/simplicial complex, while the theoretical connection between $\mathbf{L}_1$ and Δ_1 in an ML perspective, i.e., with points sampled from a manifold, is currently lacking.

Being able to estimate Δ_1 by a discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$ acting on the edges of a graph can support many applications, just as the Δ_0 estimator by weighted Laplacians successfully did. For instance, (i) topological information, i.e., the first Betti number β_1 [73], can be obtained by the dimension of the null space of $\boldsymbol{\mathcal{L}}_1$; (ii) low dimensional representation of the space of vector fields on a manifold are made possible, similar to the dimensionality reduction algorithms such as Laplacian Eigenmap from the discrete estimates of Δ_0; (iii) the well known *Helmholtz-Hodge decompositon* (HHD) [16, 73] allows us to test, e.g., if a vector field on the manifold $\mathcal{M}$ is approximately a gradient or a rotational field; lastly (iv), edge flow semi-supervised learning (SSL) and unsupervised learning algorithms, i.e., flow prediction and flow smoothing in edge space, can be easily derived from the well-studied node based learning models [10, 90] with the aid of $\boldsymbol{\mathcal{L}}_1$.

In this chapter we initiate the estimation of higher order Laplacian operators from point cloud data, with a focus on the first order Laplacian operator Δ_1 of a manifold. We propose a discrete estimator $\boldsymbol{\mathcal{L}}_1$ of the manifold Helmholtzian Δ_1 with proper triangular weights (Section 4.1) which resembles the well known Vietoris-Rips (VR) complex in the persistent homology theory. We show separately the convergence (Section 4.2) of the down and the up

components of the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$ to the continuous operators Δ_1^{down} (spectrally) and Δ_1^{up} (pointwise; up to a function depending on the edge length). We support our theoretical claims (Section 4.3) by comparing the estimated eigenvalues with the ground truth on simple manifolds; additional experiments on the real small molecules datasets show that the proposed estimator can successfully extract the correct non-trivial topological structure of the manifolds, unlike other existing methods.

4.1 Discrete Helmholtzian estimator

Below we discuss the particular choice of weights ($\mathbf{W}_0$, $\mathbf{W}_1$, and $\mathbf{W}_2$) and a modification for the *random-walk* 1-Laplacian in (2.15); this results in a manifold Helmholtzian Δ_1 estimator called the discrete Helmholtzian shown as follows.

$$\boldsymbol{\mathcal{L}}_1 = a \cdot \mathbf{B}_1^\top \mathbf{W}_0^{-1} \mathbf{B}_1 \mathbf{W}_1 + b \cdot \mathbf{W}_1^{-1} \mathbf{B}_2 \mathbf{W}_2 \mathbf{B}_2^\top \tag{4.1}$$

The core choice is the weight for a triangle $w_2(t)$. Previous choices for w_2 include (i) a constant weight on every t [105] or (ii) weights based on $\mathbf{B}_2^\top$ [52]. The former choice fails to capture the size of a triangle and does not have an analyzable limit, while the latter violates the assumption that we are building SC_2 from a point cloud. Here we introduce weights based on a *kernel* $\kappa(.,.)$; each triangle weight is the product of its edges' kernels.

$$w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \kappa(\mathbf{x}, \mathbf{y}) \cdot \kappa(\mathbf{x}, \mathbf{z}) \cdot \kappa(\mathbf{y}, \mathbf{z}) \text{ for } (x, y, z) \in T. \tag{4.2}$$

We choose $\kappa(\mathbf{x}, \mathbf{y}) = \exp\left(-\|\mathbf{x} - \mathbf{y}\|^2/\varepsilon^2\right)$, the Gaussian kernel with bandwidth parameter ϵ. The exponentially decreasing function κ enables filtering out structures that are topological noise in $\boldsymbol{\mathcal{L}}_1$, which is usually necessary when dealing with real-world data having complicated manifold structures (will be illustrated in Figures 4.3 and 5.3). Note that there is a resemblance between (4.2) and the VR complex; by definition, a triangle $t = (x, y, z)$ in the VR complex is formed if and only if its edge lenghts $\|\mathbf{x}-\mathbf{y}\|$, $\|\mathbf{x}-\mathbf{z}\|$, and $\|\mathbf{y}-\mathbf{z}\|$ are smaller than ε. Hence, the VR complex itself is obtained by using $\kappa(u) = \mathbb{1}(u < 1)$ in (4.2).

Once the weights of the triangles $\mathbf{W}_2$ are given, the weights of the lower dimensional simplices are determined by the consistency conditions required by the boundary operator. Specifically, it follows that the weight of a node $v \in V$ equals its (weighted) degree, i.e. $[\mathbf{W}_0]_{v,v} = \sum_{e \in E} |[\mathbf{B}_1]_{v,e}| [\mathbf{W}_1]_{e,e}$. Similarly for $e \in E$, $[\mathbf{W}_1]_{e,e} = \sum_{t \in T} |[\mathbf{B}_2]_{e,t}| [\mathbf{W}_2]_{t,t}$ (as in the bottom pane of Figure 4.2a).

Asymptotically, as $n \to \infty$, the kernel widths corresponding to $\mathbf{W}_1$ and $\mathbf{W}_2$ must decrease towards 0 at mutually compatible rates; namely, a shrinking δ w.r.t. the increasing n so that there is a sufficient amount of neighbors for each vertex. One can choose δ by Joncas et al. [64] due to the spectral consistency of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ (Section 4.2.1 and Figure 4.1). The analysis of the limit of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ in Theorem 4.3 suggests a choice of $\varepsilon = \mathcal{O}(\delta^{2/3})$; in practice, one has to choose ε in a case-by-case manner.

Since the convergence of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ are up to some factors (details below), we introduce parameters a and b in (4.1) such that $a \cdot \boldsymbol{\mathcal{L}}_1^{\text{down}} + b \cdot \boldsymbol{\mathcal{L}}_1^{\text{up}} \approx \Delta_1$ in large sample limit. In this work, we use a fixed $a = 1/4$ and $b = 1$ based on our empirical observation (Figure 4.2c). Note that since the spaces of *curl* or *gradient* flow are mutually orthogonal (due to HHD), a and b will only scale the eigenvalues but not the eigenspaces; hence, they will only affect the relative ranking of the *curl* and *gradient* components. More discussions on their impact are in Section 4.4.2. We leave the systematic way to choose these parameters (ε, a, and b), either in an analytic form or a data-driven approach, as future work. Finally, with the weights and parameters discussed above, the discrete Helmholtzians $\boldsymbol{\mathcal{L}}_1$ and $\boldsymbol{\mathcal{L}}_1^s$ are obtained using (4.1) and $\boldsymbol{\mathcal{L}}_1^s = a \cdot \mathbf{A}_1^\top \mathbf{A}_1 + b \cdot \mathbf{A}_2 \mathbf{A}_2^\top$ (a modification of (2.16)), respectively.

4.2 Large sample limit of the discrete Helmholtzian

In this section, we study the connection between the large sample limit of $\boldsymbol{\mathcal{L}}_1$ and the manifold Helmholtzian Δ_1. We first show that the *down* Laplacian $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ converges spectrally to Δ_1^{down} (up to a constant). The critical component of the proof is a convenient property that the non-zero spectrum of $\boldsymbol{\mathcal{L}}_1^{\text{down}} = \mathbf{B}_1^\top \mathbf{W}_0^{-1} \mathbf{B}_1 \mathbf{W}_1$ is identical to that of the graph Laplacian $\boldsymbol{\mathcal{L}}_0 = \mathbf{W}_0^{-1} \mathbf{B}_1 \mathbf{W}_1 \mathbf{B}_1^\top$ (a similar relationship holds for Δ_0 and Δ_1^{down}); therefore, one can

complete the proof using the existing spectral convergence of the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ to Δ_0 [15, 32].

The convergence of the *up* Laplacian for every edge (up to a function that depends on the edge length), called *pointwise* convergence, is shown by an asymptotic expansion of the matrix $\mathbf{B}_2\mathbf{W}_2\mathbf{B}_2^\top$ on the edge flow induced by a smooth vector field via the path integral formulation. The functional form of the triangular weights w_T in (4.2) makes it possible to remove the unwanted terms using the odd and even function symmetry.

In this section, we assume the following for our analysis.

Assumption 4.1. *The data* $\mathbf{X}$ *are sampled i.i.d. from a uniform density supported on a* d *dimensional manifold* $\mathcal{M} \subseteq \mathbb{R}^D$ *that is of class* C^3 *and has bounded curvature. W.l.o.g., we assume that the volume of* $\mathcal{M}$ *is 1; and we denote by* μ *the Lebesgue measure on* $\mathcal{M}$*.*

Assumption 4.2. *The kernel* $\kappa(\mathbf{x}, \mathbf{y})$ *of* $w_2(\mathbf{x}, \mathbf{y}, \mathbf{z})$ *in* (4.2) *is of class* C^3 *and has exponential decay.*

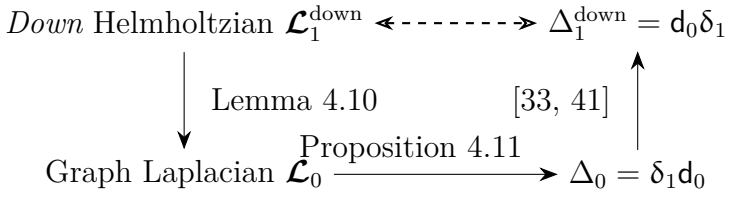

Figure 4.1: Outline for the proof of spectral consistency of the down Helmholtzian. Proposition 4.11 and Lemma 4.10 can be found in Appendix 4.8.

4.2.1 Spectral consistency of the down *Laplacian* $\boldsymbol{\mathcal{L}}_1^{\text{down}}$

The proof for spectral consistency of the 1-*down* Laplacian is outlined in the flow chart above. In short, by linking the spectra/eigenfields of Δ_1^{down} to Δ_0 as well as their discrete counterparts (two vertical arrows), one can show the consistency of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ (horizontal dashed line) using the known spectral convergence of the discrete graph Laplacian $\boldsymbol{\mathcal{L}}_0$ to the the Laplace-Beltrami operator Δ_0 [15, 32] (horizontal solid arrow). The details and proofs are in Appendix 4.8.

4.2.2 *Pointwise convergence of the* up *Laplacian* $\boldsymbol{\mathcal{L}}_1^{\text{up}}$

Let $\boldsymbol{\gamma}(t)$ for $t \in [0,1]$ be the geodesic curve connecting x, y with $\boldsymbol{\gamma}(0) = \mathbf{x}, \boldsymbol{\gamma}(1) = \mathbf{y}$, and $\boldsymbol{\gamma}'(t) = d\boldsymbol{\gamma}(t)/dt$. A 1-form (vector field) $\mathbf{v}$ on $\mathcal{M}$ induces the 1-cochain $\boldsymbol{\omega}$ on E by $\boldsymbol{\omega}([x,y]) = \omega_{xy} = \int \mathbf{v}(\boldsymbol{\gamma}(t))^\top \boldsymbol{\gamma}'(t) \mathsf{d}t$ for any edge $[x,y] \in E$. For notational simplicity, let $f_{xyz} = \omega_{xy} + \omega_{yz} + \omega_{zx}$. The goal is to show the consistency of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ for a fixed edge $[x,y]$, i.e., to show that $\boldsymbol{\mathcal{L}}_1^{\text{up}} \boldsymbol{\omega}_{xy} \to c \int_0^1 (\Delta_1 \mathbf{v})(\boldsymbol{\gamma}(t))^\top \boldsymbol{\gamma}'(t) \mathsf{d}t$. First we obtain the discrete form of the unnormalized (weighted) *up* Laplacian operating on a 1-cochain $\boldsymbol{\omega}$.

Lemma 4.1. *Let $\boldsymbol{\omega} \in \mathbb{R}^{n_1}$ be a 1-cochain induced on SC_2 by vector field $\mathbf{v}$. For any $x, y, z \in V$, we denote by $[x', y', z']$ the canonical ordering of the triangle $t \in T$ with vertex set $\{x, y, z\}$ (if one exists). Then, $[\mathbf{B}_2 \mathbf{W}_2 \mathbf{B}_2^\top \boldsymbol{\omega}]_{[x,y]} = \sum_{z \in \{v \in V : [x', y', v'] \in T\}} w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) f_{xyz}$.*

Lemma 4.1 is proved in Appendix 4.7.1. From Lemma 4.1, it is enough to consider *divergence-free* 1-forms for the limit of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$, because $f_{xyz} = 0$ if $\mathbf{v}$ is *curl-free*. The following proposition shows the asymptotic expansion for the integral form of $\sum_z w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) f_{xyz}$ (Lemma 4.1) when n is large.

Proposition 4.2. *If Assumptions 4.1–4.2 hold, $\mathbf{v}$ is divergence-free and of class $C^4(\mathcal{M})$, then*

$$\varepsilon^{-d} \int_{\mathcal{M}} w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) f_{xyz} \mathsf{d}\mu(\mathbf{z}) = \varepsilon^2 c(\|\mathbf{x} - \mathbf{y}\|) \int_0^1 \Delta_1 \mathbf{v}(\boldsymbol{\gamma}(t))^\top \boldsymbol{\gamma}'(t) \mathsf{d}t + \mathcal{O}(\varepsilon^3, \delta^2), \tag{4.3}$$

with $c(\|\mathbf{x} - \mathbf{y}\|) = c_2 - c_1(\|\mathbf{x} - \mathbf{y}\|)$, $c_1(\|\mathbf{x} - \mathbf{y}\|) = \frac{2}{3} \int_{\mathbf{z} \in \mathbb{R}^d} \kappa(\|\mathbf{z} - \mathbf{x}\|^2) \kappa(\|\mathbf{z} - \mathbf{y}\|^2) ((\mathbf{z}_1 - \mathbf{x}_1)^2 + (\mathbf{z}_1 - \mathbf{y}_1)^2 \mathsf{d}\mathbf{z}$ and $c_2 = \frac{4}{3} \int_{\mathbf{z} \in \mathbb{R}^d} \kappa(\|\mathbf{z}\|^2) \kappa'(\|\mathbf{z}\|^2) \mathbf{z}_1^2 \mathbf{z}_2^2 \mathsf{d}\mathbf{z}$. Here $\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1$ represent the first coordinate of $\mathbf{x}, \mathbf{y}, \mathbf{z}$ in the local tangent coordinate system.

Note that in the above, c_1 is indeed a function of $\|\mathbf{x} - \mathbf{y}\|$ for any $x, y \in \mathcal{M}$.

Sketch of proof. We first prove the case when $\mathcal{M} = \mathbb{R}^d$ (Lemma 4.8). Consider a triangle $[x, y, z] \in T$, where z will be integrated over. We parametrize the path segments $x \to y$, $y \to z$, and $z \to x$ by $\mathbf{u}(t)$, $\mathbf{v}(t)$, and $\mathbf{w}(t)$, respectively. By changing the variables $\mathbf{w}(t)$ and

$\mathbf{v}(t)$ to $\mathbf{u}(t)$, we express all three line integrals as integrals along the segment $[x, y]$. Unwanted terms can be removed by odd function symmetry and the structure of triangular weights in (4.2) (Lemma 4.6). The remaining second order term is the 1-cochain $\Delta_1\mathbf{v}$ by Corollary 2.24. Next, when $\mathcal{M} \subseteq \mathbb{R}^D$, we bound the error terms of approximating the integration from $\mathcal{M}$ to the tangent plane $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ at $\mathbf{x}$ with $\mathcal{O}(\varepsilon^3)$ in Lemma 4.9. Combining this Lemma with Lemma 4.8 concludes the proof. ■

Proposition 4.2 implies that one should choose $\varepsilon = \mathcal{O}(\delta^{2/3})$ so that the $\mathcal{O}(\delta^2)$ term has same asymptotic rate as the higher order $\mathcal{O}(\varepsilon^3)$ term. Now we can analyze the pointwise bias of the estimator,

Theorem 4.3. *Under the assumptions of Proposition 4.2, let* $\varepsilon = \mathcal{O}(\delta^{2/3})$ *and* $q(\|\mathbf{x}-\mathbf{y}\|) = \frac{w_1(\|\mathbf{x}-\mathbf{y}\|)}{\varepsilon^2 \cdot c(\|\mathbf{x}-\mathbf{y}\|)}$ *with* $w_1(\|\mathbf{x}-\mathbf{y}\|) = \int_{\mathbb{R}^d} \kappa(\|\mathbf{z}-\mathbf{x}\|)\kappa(\|\mathbf{z}-\mathbf{y}\|)\mathsf{d}\mathbf{z}$. *Then, for any fixed* $x, y \in \mathcal{M}$,

$$\mathbb{E}\left[q(\|\mathbf{x}-\mathbf{y}\|)\left(\boldsymbol{\mathcal{L}}_1^{\mathrm{up}}\boldsymbol{\omega}\right)_{[x,y]}\right] = \int_0^1 \Delta_1\mathbf{v}(\boldsymbol{\gamma}(t))^\top \boldsymbol{\gamma}'(t)\mathsf{d}t + \mathcal{O}(\varepsilon, n^{-1}). \tag{4.4}$$

In the above, the expectation is taken over samples $\mathbf{X}$ *of size* n*, to which the points* $\mathbf{x}, \mathbf{y}$ *are added.*

Sketch of proof. The proof follows from the *Monte Carlo* approximation [15, 32] of the RHS of Lemma 4.1, i.e., $\mathbb{E}\left[\frac{1}{n}\sum_z w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) f_{xyz}\right] = \int_{\mathcal{M}} w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) f_{xyz}\mathsf{d}\mu(\mathbf{z})$. Combining the result of Proposition 4.2 (the ε bias) and the standard ratio estimator (the n^{-1} bias) completes the proof. The proof details are in Appendix 4.7.3. ■

Note that the rate for ε for the Δ_0 estimator ($\boldsymbol{\mathcal{L}}_0$) is slower than n^{-1}, see e.g., Berry and Sauer [15], Singer [111]. One can thus drop the n^{-1} term in (4.4) using the similar bandwidth parameter as the Δ_0 estimator. We note also that the $\boldsymbol{\mathcal{L}}_1^{\mathrm{up}}$ estimator may be biased, due to the dependence of q on $\|\mathbf{x}-\mathbf{y}\|$.

In sum, we have derived the continuous operator limits of the *up* and *down* Laplacian terms. We have shown that $\boldsymbol{\mathcal{L}}_1^{\mathrm{down}}$ converges to (a constant times) Δ_1^{down} spectrally. For $\boldsymbol{\mathcal{L}}_1^{\mathrm{up}}$, we have shown that the pointwise limit exists, and that it equals Δ_1^{up} multiplied with

a function depending only on the edge length. The simulation in Section 4.3 suggests that the bias is not too large.

Since the limits of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ have different scalings, the estimator $\boldsymbol{\mathcal{L}}_1$ of Δ_1 is a weighted sum of the two terms with coefficients a and b as in (4.1). We use $a = 1/4; b = 1$ based on our empirical observations on the synthetic datasets. Please refer to Section 4.3 and Section 4.4.2 for more details.

4.3 Experiments

4.3.1 Spectrum of the Helmholtzian estimator

We illustrate our theoretical results on two synthetic manifolds by comparing the spectrum of $\boldsymbol{\mathcal{L}}_1$ with the ground truth. The first manifold of interest is `CIRCLE` (a one-dimensional circle in $\mathbb{R}^2$) for which the Δ_1 eigenvalues are $\lambda_k = \left(\lceil k/2 \rceil\right)^2$ for $k = 0, 1, \cdots$. Figure 4.2b, shows the estimated (red) eigenvalues for `CIRCLE` overlaid on the ground truth (blue). We rescale the estimated eigenvalues by the minimum non-zero eigenvalue of $\boldsymbol{\mathcal{L}}_1$ (i.e., $\lambda_1(\boldsymbol{\mathcal{L}}_1)$) so that they are comparable to those of Δ_1 (blue). We see that up to $k \approx 30$ the Δ_1 spectrum is accurately estimated. Since the $\boldsymbol{\mathcal{L}}_1$ spectrum is finite, the number of Δ_1 eigenvalue accurately estimated can be increased by increasing the sample size n. Now we turn to a slightly more complicated manicenifold, `FLAT-TORUS` (a two-dimensional *flat-torus* in $\mathbb{R}^4$). Figure 4.2c shows the first fifty estimated eigenvalues (red) of `FLAT-TORUS` overlaid with the ground truth (blue). Since there are two loops in the flat torus, the first two eigenflows correspond to *harmonic* flows ($\beta_1 = 2$). The spectrum of `FLAT-TORUS` has multiplicity, with half of the eigenflows for each distinct eigenvalue being *gradient* and the other half being *curl* flows. With $a = 1/4$ and $b = 1$, one can match the estimated spectrum with the ground truth. The eigenvalues for the *gradient* flows do not precisely match those of the *curl* flows (see, e.g., the small bumps in the fifth plateau of Figure 4.2c). It might be due to a a weaker convergence (pointwise, not spectrally) of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$, and the fact that its limit to Δ_1^{up} is up to a function depending on edge length. Despite having weaker theoretical results, the numerical spectrum does not differ

too much from the ground truth.

The *harmonic* flow space $\mathcal{H}_1$, which parametrizes the non-trivial *holes* in $\mathcal{M}$, is of special significance by being directly related to the loop structure of the given manifold $\mathcal{M}$ (see Chapter 3.7 for more details). Applications related to identifying or analyzing this subspace are called *topological feature discovery*; in next Section, we will illustrate some of these applications.

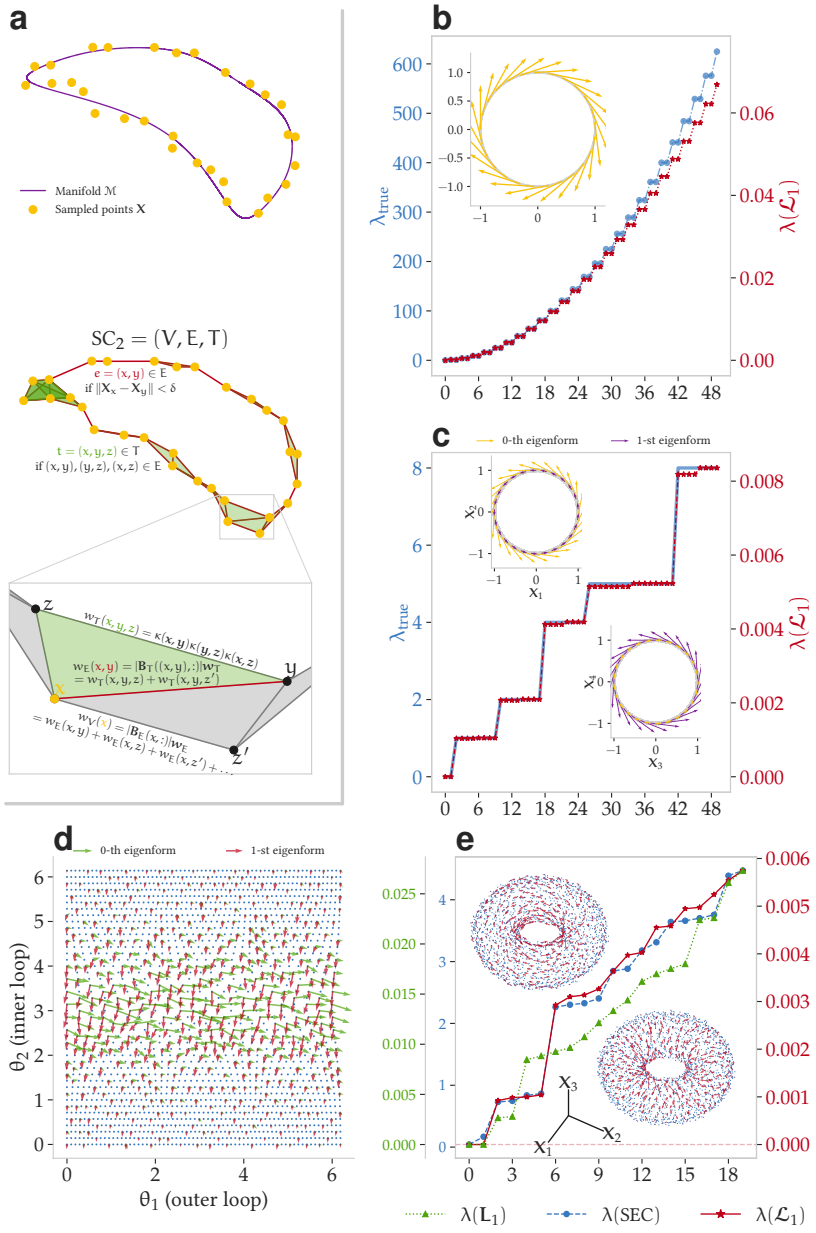

Figure 4.2: Estimations of the first Betti number β_1 on the synthetic manifolds. (a) An illustration for constructing an SC_2 from a point cloud and the choice of weights $\mathbf{W}_2$, $\mathbf{W}_1$, and $\mathbf{W}_0$. Here $\mathbf{W}_2 \in \mathbb{R}^{n_2}$ is an n_2-dimensional vector, with $\mathbf{W}_2 = \text{diag}(\mathbf{W}_2)$ (same for $\mathbf{w}_2$ and $\mathbf{w}_0$). (b and c) Estimated eigenvalues λ's of the graph Helmholtzian $\boldsymbol{\mathcal{L}}_1^s$ (red) overlaid with the ground truth spectrum (blue). The inset plots are the estimated *harmonic* eigneforms plotted on the original space. (d) The first two (*harmonic*) eigenfields in the intrinsic coordinate space of the torus. (e) The estimated spectra of the graph Helmholtzian $\boldsymbol{\mathcal{L}}_1$ (red), the unweighted 1-Laplacian $\mathbf{L}_1$ (green), and SEC (blue). The left and right inset plots are the zeroth and the first eigenfields in the original Euclidean space, respectively.

4.3.2 Topological feature discovery

The spectrum of Δ_1 contains information about the manifold topology. More specifically, the dimension of the null space of Δ_1, equals the number of the independent loops on the manifold $\mathcal{M}$ [73], represented by the first Betti number β_1. This number can be estimated by the number of zero eigenvalues of the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$. Eigenflows that correspond to these eigenvalues span a special vector field basis called *harmonic* flows. These flows parametrize the loops in the manifold; for instance, the estimated yellow/purple *harmonic* flows in the insets of Figures 4.2b and 4.2c reveal the loops in `CIRCLE` and `FLAT-TORUS`, respectively. Therefore, one can identify the loops by visualizing these flows on point cloud data, something which is not immediate possible with TDA.

As discussed earlier, TDA and SEC [13] are prior works in extracting topological features. The estimation of β_1 with TDA is performed by a multiple spatial resolution approach, called the *Persistence diagram* (PD). Statistically significant structures in a PD can be identified by bootstrapping-based methods [21, 128]. The TDA algorithms are powerful in generalizing to higher dimensional *holes*, e.g., cavities in the 3-dimensional space; in contrast, they are less effective in constructing vector field bases on $\mathcal{M}$ since the orientations are not tracked. SEC directly approximates the spectrum of Δ_1 using the eigenfunctions of the graph Laplacian [32].

To illustrate the effectiveness of the proposed Helmholtzian estimator ($\boldsymbol{\mathcal{L}}_1$), we report the eigenvalues of $\boldsymbol{\mathcal{L}}_1$ (red) along with two other baselines: the SEC [13] (blue) and the unweighted (constant triangular weight) Laplacian $\mathbf{L}_1$ (green). Additionally, we present the PD with a 95% confidence interval estimated from 7,000 bootstrap samples on these real datasets that we do not know the ground truth. Figure 4.2e shows the estimated spectra of different algorithms on `TORUS`, a synthetic two-dimensional torus. A torus has two 1-dimensional loops ($\beta_1 = 2$); the first two eigenvalues of $\mathbf{L}_1$ or $\boldsymbol{\mathcal{L}}_1$ are close to zero, but the null space estimated by SEC has dimension 1 (Figure 4.2e). Inset plots show the first two eigenfields estimated from the *harmonic* eigenvectors of $\boldsymbol{\mathcal{L}}_1^s$. We further plot these two

harmonic eigenflows w.r.t. the angles θ_1 and θ_2 (Figure 4.2d) that parametrize two loops in `TORUS` . Figure 4.2d and the insets of Figure 4.2e indicate that the zeroth eigenflow (upper left) is along the outer (bigger) cycle while the first eigenvector (lower right) belongs to the inner (smaller) loop.

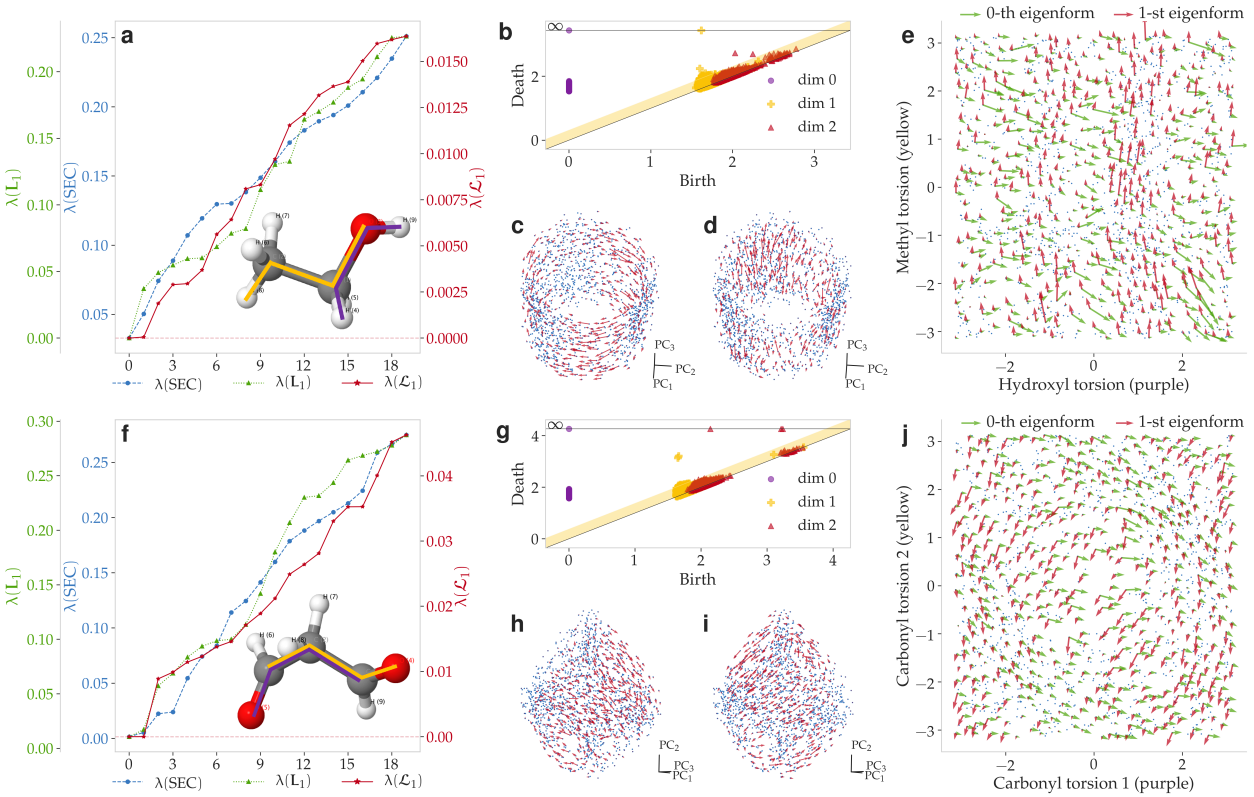

Figure 4.3: Estimations of the first Betti number β_1 on the ethanol (a–e) and malondialdehyde (f–j) datasets. (a and f) The estimated spectra of the $\boldsymbol{\mathcal{L}}_1$ (red) and other baselines. The inset plots are schematics of the molecules with the bond torsions of interest. (b and g) The persistence diagrams of the small molecules data. (c, d, h, and i) The first two *harmonic* flows in the first three PCA spaces of the small molecule datasets. (e and j) The first two *harmonic* eigenflows in the torsion space, whose first two coordinates correspond to the purple and yellow bond torsions in the inset plots of (a and f), respectively.

We apply similar analyses to `ETH` and `MDA`[1] (Figure 4.3), which correspond to another real molecular dynamics simulation trajectories of the ethanol and malondialdehyde molecule [29],

[1]These two datasets are available in `http://quantum-machine.org`.

respectively. We preprocess these two datasets in ways similar to that for `SN2` in Section 3.4.2. In addition to the point cloud $\mathbf{X}$, we also calculate the *bond torsions* for these two small molecule dataset. Each bond torsion is calculated by the dihedral angle of the corresponding chemical bond for every molecular configuration; they are treated as the "dictionaries" for our unsupervised learning task.

The ambient and the (estimated) intrinsic dimensions of `ETH` are $D = 102$ and $d = 2$, respectively. The dataset is known to be a noisy non-uniformly sampled torus parametrized by the Hydroxyl (purple) and Methyl (yellow) functional groups (the inset of Figure 4.3a). The inner loop corresponding to the Methyl group is difficult to detect due to an asymmetric topological structure. We subsample $n = 1,500$ points farthest from each other. By using the proposed triangle weights (4.2), one can successfully remove the topological noise while preserving the smaller inner loop (thus resulting in $\beta_1 = 2$) in the weighted Helmholtzian $\boldsymbol{\mathcal{L}}_1$ (red curve in Figure 4.3a). The unweighted Laplacian $\mathbf{L}_1$ (green) and the SEC (blue) in Figure 4.3a fail to properly encode the geometric information and consequently do not detect the second loop, reporting $\beta_1 = 1$. Contrarily, the PD in Figure 4.3b is prone to topological noise; it detects $\beta_1 > 2$ statistically significant loops.

Apart from β_1, one can also obtain estimates of the two *harmonic* eigenflows from the first two eigenvectors of $\boldsymbol{\mathcal{L}}_1^s$ (see also Section 4.3.3 for the discussion on the first ten eigenflows). These two *harmonic* flows (in Figures 4.3c and 4.3d) correspond to the two independent loops of this manifold. With our prior knowledge that the purple and yellow rotors parametrize the loops in `ETH`, we map the two eigenfields that reside in the PCA space $\mathbf{X}$ to the torsion space by estimating the Jacobian matrix of the torsions w.r.t. $\mathbf{X}$ (in Section 2.5), as in Figure 4.3e. As clearly shown in the figure, the zeroth eigenfield (green) aligns with the direction of increase of the Hydroxyl rotor, while the first eigenfield (red) matches with the derivative of the Methyl torsion.

The estimation of β_1 on `MDA` is shown in the second row of Figure 4.3. This dataset has a similar topological structure to `ETH`, i.e., they are both non-uniformly sampled tori. The two loops are parametrized by the two Carbonyl bond torsions (inset plot of Figure

4.3f). Compared to ETH, MDA is easier in the sense that the two loops have comparable radii. However, this dataset is harder to visualize since the torus is embedded in a four-dimensional space. A clear separation between the zeroth and the first eigenvalues of $\boldsymbol{\mathcal{L}}_1$ can be seen in the estimated spectrum (Figure 4.3a) with the help of triangular weights in (4.2). Even though the estimated dimensions of the null space of SEC and $\mathbf{L}_1$ are both two, we do not observe such well-separated gaps between the first two estimated eigenvalues compared to that of $\boldsymbol{\mathcal{L}}_1$. The bootstrapped PD with a 95% confidence interval (Figure 4.3g) shows that β_1 is at least two. However, some (statistically significant) loops generated from the topological noise are still visible in the vicinity of the diagonal line of the PD. Similar to ETH, we map the first estimated two eigenfields in Figures 4.3h–4.3i to the Carbonyl torsion space, as in Figure 4.3j. It is consistent with our prior knowledge that these two eigenfields parametrize the yellow and the purple torsions.

4.3.3 Low-frequency eigenflows of ETH and MDA

In Figure 4.4, we estimate the first 10 eigenfields by solving the linear system (2.21). The 0-th eigenfield clearly represents the bigger loop parameterized by Hydroxyl rotor as shown in the inset scatter plot of Figure 4.3a. In Figure 4.4c, it is difficult to tell whether the first eigenflow corresponds to Methyl rotor or not. One can overcome this issue by mapping the first eigenfield to the torsion space with prior knowledge as illustrated in Figure 4.3e. *Harmonic* flows often represent a global structure; by contrast, flows in Figure 4.4d–4.4f are more localized, implying that the eigenflows are not *harmonic*. The Helmholtz-Hodge decomposition of the eigenvectors of $\boldsymbol{\mathcal{L}}_1$ in Figure 4.4a confirms this.

Figure 4.5 shows the scatter plot of the first three principal components of the MDA dataset. As clearly shown in the figures, it is difficult to make sense of the topological structure for the manifold of such dataset is a torus embedded in a four-dimensional space. With the aid of the first two *harmonic* eigenfields, one can infer that the first loop travels in the direction of northwest to southeast, while the second loop goes diagonally from northeast to southwest. With proper prior knowledge, one can map the *harmonic* eigenfields to the

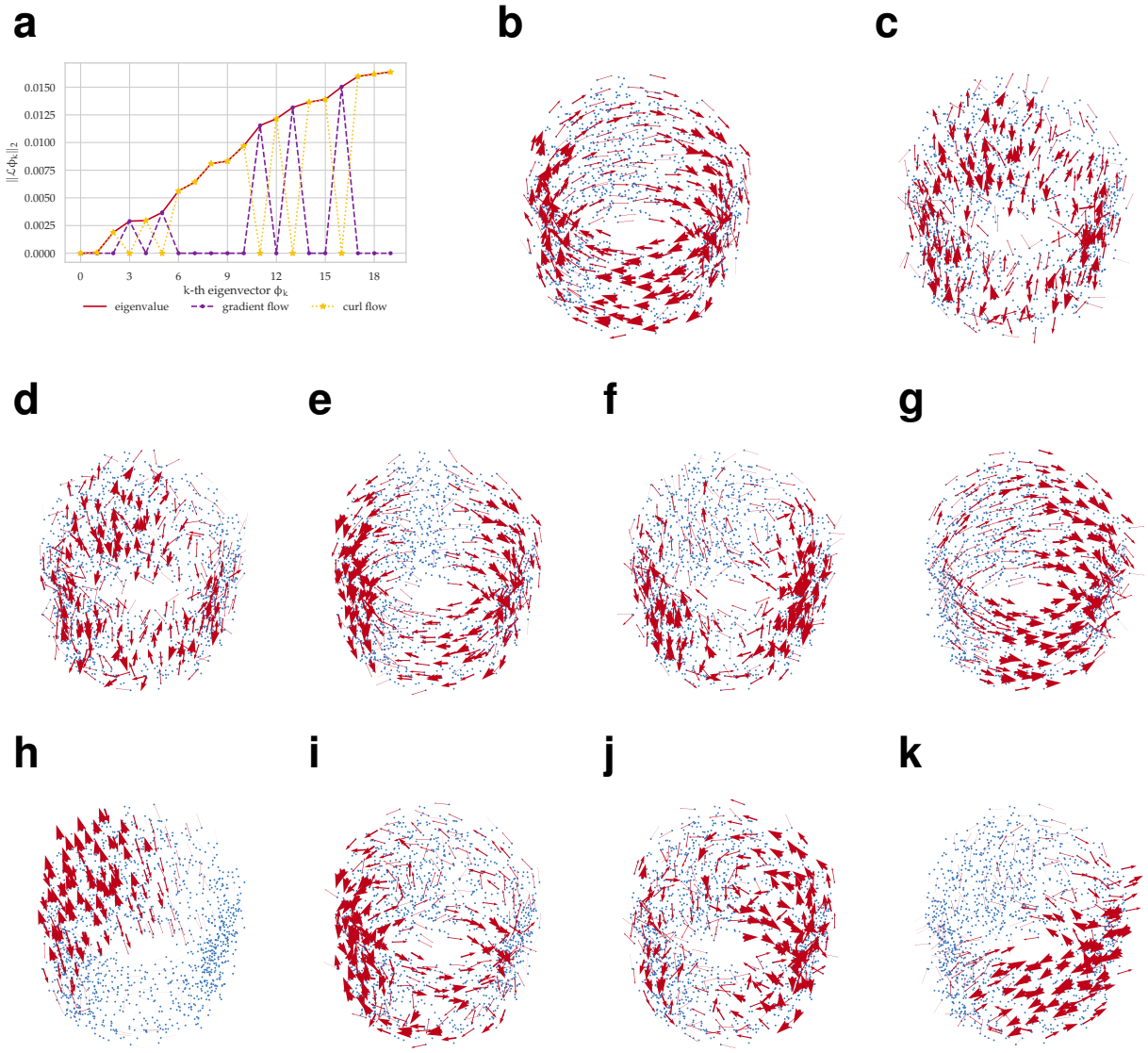

Figure 4.4: (a) HHD on the first 10 eigenfields, showing that the first two eigenflows are *harmonic*; the fourth, sixth, twelfth, fourtheenth, and the seventeenth eigenflows are *gradient* flow, while the rest are *curl* flows. (b–k) The first 10 estimated vertex-wise eigenfields on the original dataset $\mathbf{X}$ by solving the linear system (2.21).

torsion space to get a better visualization, as shown in Figure 4.3j.

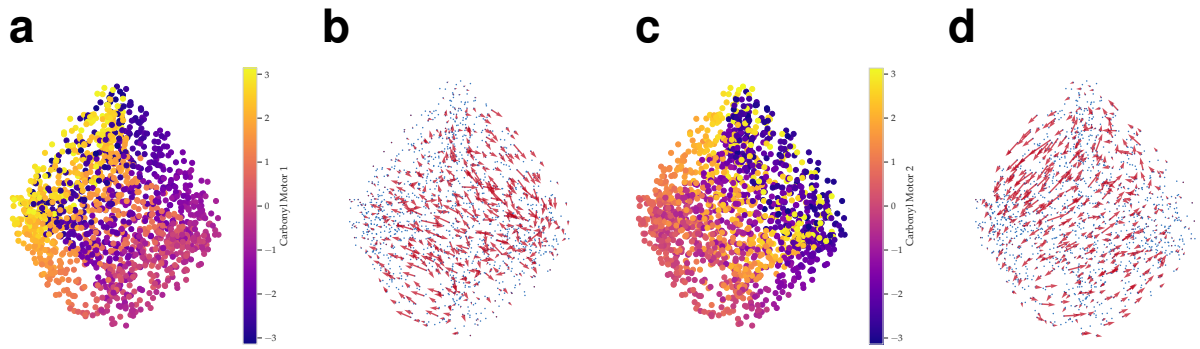

Figure 4.5: The *harmonic* eigenflows of the MDA dataset. (a and c) are the scatter plot of the first three PCs colored by the first and second Carbonyl rotors (purple and yellow in the inset of Figure 4.3f and Figure 4.3j). (b and d) represent the first two *harmonic* eigenfields estimated from the eigenvectors of $\boldsymbol{\mathcal{L}}_1^s$. The zeroth eigenfield in (b) parameterizes the first carbonyl rotor in (a), while the first eigenfield in (d) represents the second carbonyl rotor as in (c). See Figure 4.3j for a better visualization.

4.4 Discussions

4.4.1 Related works

Consistency results of Laplace type operators on a manifold. Numerous *non-linear dimensionality reduction* algorithms from point cloud data (e.g., Belkin and Niyogi [8], Coifman and Lafon [32], Hein et al. [55, 56], Nadler et al. [85], Ting et al. [120]) investigated the consistency of functions (0-forms) on $\mathcal{M}$. Spectral exterior calculus (SEC) [13] extended the existing consistency results of 0-forms to 1-forms by building a frame (overcomplete set) to approximate the spectrum of Δ_1 from $\boldsymbol{\mathcal{L}}_0$. The SEC only has $\mathcal{O}(n^3)$ dependency in computing the eigenvectors of $\boldsymbol{\mathcal{L}}_0$. Therefore, it is well-suited for topological feature discovery when large number of points are sampled from $\mathcal{M}$. Nevertheless, the algorithm involves several fourth order *dense* tensor computations with size m, the number of the eigenvectors of 1-Laplacian to estimate, and results in a $\mathcal{O}(m^4)$ dependency in memory and $\mathcal{O}(m^6)$ in runtime. These dependencies may cause difficulties in applying SEC to the edge flow learning (in Chapter 5) scenarios in real datasets, since higher frequency terms of the 1-Laplacian are

oftentimes needed ($m \geq 100$). On the other end, Singer and Wu [112] studied the discrete approximation of the *Connection Laplacian*, a differential operator acting on tensor bundles of a manifold. This is intrinsically different from the 1 Hodge Laplacian we discussed.

Random walks on discrete k-Laplacian operator. Schaub et al. [105] studied random walks on the edges of normalized 1-Hodge Laplacian of the pre-determined graph. For points sampled from $\mathcal{M}$, they proposed an *ad hoc* hexagonal binning method to construct the SC_2 from the trajectories; theoretical aspects of the binning method were not discussed. On the theoretical front, frameworks of random walks on simplices of down [83] and up [93] k-Laplacian have also been visited. These works focused on the connection between random walks on a simplicial complex and spectral graph theory. Our graph Helmholtzian $\boldsymbol{\mathcal{L}}_1^s$ based on pairwise triangular weights makes it possible to extend their frameworks to the point cloud datasets.

Topological data analysis and Persistent homology. Persistent Homology (PH) theory enables us to study the topological features of a point cloud in multiple spatial resolutions. The direct application of PH is the estimation of the k-th Betti number, i.e., the number of k dimensional *holes*, from $\mathbf{X} \subseteq \mathcal{M}$. PH algorithms applied to real data typically output large numbers of k-holes with low persistences. Therefore, one selects the statistically significant topological features by some bootstrapping-based methods, e.g., the quantile of the *bottleneck distances* between estimated *persistent diagram* (PD) and the bootstrapped PDs. Readers are encouraged to refer to Chazal and Michel [21], Wasserman [128] for more details. The PH theories are powerful in finding β_k when $k \geq 2$; in contrast, the Laplacian based methods are found effective in edge flow learning and smoothing for their abilities to keep track of the orientations; these application will be introduced in Chapter 5.

4.4.2 The choice of the weights between up/down *Laplacians*

From HHD, the space of cochain $\mathbb{R}^{n_1}$ can be decomposed into three different orthogonal subspaces: the image of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ (*gradient*), the image of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ (*curl*), and the kernel of both $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ (*harmonic*). Since these subspaces are orthogonal to each other, rescaling $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ with some constants a and b will only scale the spectra accordingly without altering the eigenvectors. Here, we investigate the spectrum of the rescaled $\boldsymbol{\mathcal{L}}_1$ w.r.t. a, b with the following Corollary.

Corollary 4.4 (Spectrum of the $\boldsymbol{\mathcal{L}}_1$ rescaled by a and b)**.** *The range of the spectra of* $\boldsymbol{\mathcal{L}}_1$ *is* $\lambda(\boldsymbol{\mathcal{L}}_1) \in [0, \max(2a, 3b)]$.

Proof. From Horak and Jost [58], $\lambda(\mathbf{W}_k^{-1}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top) \in [0, k+2]$. From Lemma 4.10, one has $\mathcal{S}(\boldsymbol{\mathcal{L}}_k^{\text{down}}) = \mathcal{S}(\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}})$. Therefore, $\lambda(\mathbf{B}_1^\top\mathbf{W}_0^{-1}\mathbf{B}_1\mathbf{W}_1) \in [0, 2]$ and $\lambda(\mathbf{W}_1^{-1}\mathbf{B}_2\mathbf{W}_2\mathbf{B}_2^\top) \in [0, 3]$. From HHD, an eigenvector can only be either *curl*, *gradient*, or *harmonic* flow. Thus the non-zero spectrum of $\boldsymbol{\mathcal{L}}_1$ will simply be the union of two disjoint eigenvalue set. Since rescaling rescaling the matrix by a constant will only change the scales of the eigenvalues, the union of the (rescaled) *down* and *up* Laplacian will therefore be in the range of $[0, \max(2a, 3b)]$. This completes the proof. ■

Note that by choosing $a = 1/2$ and $b = 1/3$, the spectra of $\boldsymbol{\mathcal{L}}_1$, $\boldsymbol{\mathcal{L}}_1^{\text{down}}$, and $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ are all upper bounded by 1.

Based on the discussion above, different choices of a, b constants will shift the rankings between the *curl* flows ($\boldsymbol{\mathcal{L}}_1^{\text{up}}$) and *gradient* flows ($\boldsymbol{\mathcal{L}}_1^{\text{down}}$). This effect can be seen in Figure 4.6, with the first two *gradient* flows (in purple) in Figure 4.6a corresponds to the ninth and the thirteenth eigenvalues in Figure 4.6c. Considering the case $a = \frac{1}{2}; b = \frac{1}{3}$ when the spectra of $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ are both upper bounded by 1. Since there are only $n_0 - \beta_0$ non-zero eigenvalues in $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ (# of edges needed to form a spanning tree) compared to $n_1 - (n_0 - \beta_0) - \beta_1$ non-zero eigenvalues in $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ (# of independent triangles), the density of the *gradient* flows will be $\mathcal{O}(n_1/n_0)$ less than those of *curl* flow. That is to say, we will

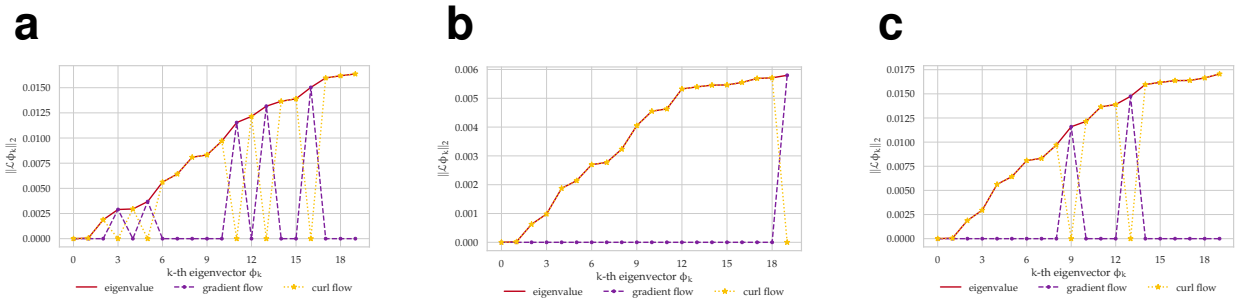

Figure 4.6: Shift in the rankings of the $\boldsymbol{\mathcal{L}}_1^{\text{up}}$, $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ spectrum with different choices of a, b values for ethanol dataset. The a, b values of (**a**–**c**) are $a = 1/4, b = 1$, $a = 1/2, b = 1/3$, $a = b = 1$, respectively. The eigenvector corresponds to the third eigenvalue in (a) is identical to that corresponds to the 18th in (b) and the ninth in (c). Note that the rankings within *gradient* or *curl* flows will not change by different choices of a, b, which can be shown by comparing the *curl* flows (yellow) between (a–c).

observe more *curl* flows than *gradient* flow for a fixed number of eigenvalues as shown in Figure 4.6b. Choosing a smaller a value increases the density of the *gradient* flow in the low frequency region (see a smaller choice of a in 4.6c and an even smaller a in 4.6a) It creates a more balanced distribution of flows in the low frequency regime. Empirical results in Section 4.3 suggest the choice of $a = 1/4; b = 1$ matches perfectly in the manifolds with known ground truth spectrum. This choice also creates the most balanced spectrum within the first 20 eigenvalues, as shown in Figure 4.6.

One can also analyze the random walk in the finite simplicial complex as in Schaub et al. [105]. By letting $a = 1/2; b = 1/3$ with $\mathbf{W}_2 = \mathbf{I}_{n_2}$, they showed the constructed Helmholtzian corresponds to a finite random walk with equal probability ($p = 1/2$) of performing *up* (diffuse to upper adjacent edges by common triangle) and *down* (diffuse to lower adjacent edges by common nodes) random walk. One can easily extend their analysis to non-constant weights $\mathbf{W}_2$ and different a, b values. This results in a random walk with probability $\frac{2a}{2a+3b}$ in performing lower random walk, while performing upper random walk with probability $\frac{3a}{2a+3b}$. Similarly, $a = 1/2; b = 1/3$ will result in an equal probability of upper/lower random walks, as suggested in Schaub et al. [105]. However, it might not be optimal for the transition

probability when performing lower random walk (depending on w_1), which is much larger than the transition probability of the upper adjacent walk (depending on w_2). Hence, one might need to choose a smaller a value to ensure a more balanced random walk across all neighboring edges.

4.5 Summary

The main contributions in this chapter are to (i) propose an estimator of the Helmholtzian Δ_1 of a manifold in the form of a weighted 1-Laplacian of a two-dimensional simplicial complex SC_2, whose vertex set includes points sampled from a manifold. With the proposed kernel function for triangles, which is a core part of the construction of this estimator, (ii) we further derive (Section 4.2) the infinite sample limit of 1 up-Laplacian $\boldsymbol{\mathcal{L}}_1^{\mathrm{up}}$ under the assumption that the points are sampled from a constant density supported on $\mathcal{M}$. (iii) The spectral consistency of the corresponding $\boldsymbol{\mathcal{L}}_1^{\mathrm{down}}$ is also shown using the spectral dependency to the well-studied graph Laplacian.

On the methodological side, we show (iv) applications to topological feature discovery on the synthetic manifolds along with several molecular dynamics datasets having asymmetric topological features. Due to the geometric information encoded in the triangular weights, the proposed Helmholtzian estimator successfully extracts the correct number of loops in every dataset; by contrast, methods such as 1-Laplacian with constant triangular weights, SEC, and PD are sensitive to topological noise. In addition to the validity in estimating the number of loops, the *harmonic* basis vectors obtained from the eigenflows can be used to identify the "loops" in the point cloud data. We present a possibility of loop identification by visualization; this idea will be investigated and extended to the higher-order k-Laplacian setting in Chapter 6.

4.6 Appendix—furthest points sampling method

Throughout this thesis (and for most of the manifold learning algorithms), the sampling density $\psi(\mathbf{x})$ is assumed to be constant; however, this is not always the case for most of the real datasets. To mitigate the non-uniform sampling effects, we use Algorithm 4.1 to subsample points furthest to each other (in terms of the Euclidean metric in the ambient space $\mathbb{R}^D$).

Algorithm 4.1: FURTHESTPOINT: furthest points sampling

Input : Initial point cloud $\tilde{\mathbf{X}} \in \mathbb{R}^{n' \times D}$, number of furthest points n

1 $\mathbf{X} \leftarrow \emptyset$

2 Pick a point $\hat{\mathbf{x}} \in \mathbb{R}^D$ randomly from $\tilde{\mathbf{X}}$

3 **for** $i = 1, \cdots, n-1$ **do**

4 $\quad \mathbf{X} \leftarrow \mathbf{X} \cup \{\hat{\mathbf{x}}\}$ ▷ `Add` $\hat{\mathbf{x}}$ `to` $\mathbf{X}$

5 $\quad \tilde{\mathbf{X}} \leftarrow \tilde{\mathbf{X}} \backslash \{\hat{\mathbf{x}}\}$ ▷ `Remove` $\hat{\mathbf{x}}$ `from` $\tilde{\mathbf{X}}$

6 $\quad \hat{\mathbf{x}} \leftarrow \operatorname{argmax}_{\mathbf{x} \in \mathbf{X}} \min_{\tilde{\mathbf{x}} \in \tilde{\mathbf{X}}} \|\mathbf{x} - \tilde{\mathbf{x}}\|_2$

$\quad$ ▷ `Find the point` $\hat{\mathbf{x}}$ `in` $\mathbf{X}$ `that is furthest from` $\tilde{\mathbf{X}}$

Return : Point cloud $\mathbf{X} \in \mathbb{R}^{n \times D}$

Note that the non-uniform sampling effect is not guaranteed to be fixed using this method since it also depends on the ratio between n and $|\tilde{\mathbf{X}}|$. For instance, if $n = |\tilde{\mathbf{X}}|$ then the procedure simply output all the point, and the density stays intact after the "sampling" procedure. Note also that by using the geodesic distance (by Dijkstra algorithm) instead of Euclidean distance in Step 6, one can get a more uniform sample (regardless of the curvature of $\mathcal{M}$); however, it would result in high computational overhead and memory consumption. Empirically, we do not observe too much difference between these two distance metrics.

4.7 Appendix—proofs of the pointwise convergence of the up *Helmholtzian*

4.7.1 Proof of Lemma 4.1

Proof. First note that

$$[\mathbf{B}_2\mathbf{W}_2\mathbf{B}^\top\boldsymbol{\omega}]_{[x,y]} = w_1(\mathbf{x},\mathbf{y})\omega_{[x,y]} + (\cdots).$$

Here $w_1(\mathbf{x},\mathbf{y}) = [\mathbf{W}_1]_{xy,xy}$. There are six different cases to consider in the $(\cdots)$ part. Assume $[x',y',z']$ is the canonical permutation/ordering of x,y,z, i.e., $x' < y' < z'$. Note that by the definition of $w_2(\mathbf{x},\mathbf{y},\mathbf{z})$, one has $w_2(\mathbf{x}',\mathbf{y}',\mathbf{z}') = w_2(\mathbf{x},\mathbf{y},\mathbf{z})$,

1. $[x',y'] \times [x',z'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot -1 \cdot \omega_{[x',z']}$
2. $[x',y'] \times [y',z'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot 1 \cdot \omega_{[y',z']}$
3. $[x',z'] \times [x',y'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot -1 \cdot \omega_{[x',y']}$
4. $[x',z'] \times [y',z'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot -1 \cdot \omega_{[y',z']}$
5. $[y',z'] \times [x',y'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot 1 \cdot \omega_{[x',y']}$
6. $[y',z'] \times [x',z'] \to w_2(\mathbf{x},\mathbf{y},\mathbf{z}) \cdot -1 \cdot \omega_{[x',z']}$

Grouping 1 & 2, one obtains

$$w_2(\mathbf{x},\mathbf{y},\mathbf{z})(\omega_{[y',z']} - \omega_{[x',z']}) = w_2(\mathbf{x},\mathbf{y},\mathbf{z})(f_{x'y'z'} - \omega_{[x',y']}).$$

Similarly for 3 & 4; 5 & 6, we have

$$w_2(\mathbf{x},\mathbf{y},\mathbf{z})(-\omega_{[x',y']} - \omega_{[y',z']}) = w_2(\mathbf{x},\mathbf{y},\mathbf{z})(-f_{x'y'z'} - \omega_{[x',z']})$$
$$w_2(\mathbf{x},\mathbf{y},\mathbf{z})(\omega_{[x',y']} - \omega_{[x',z']}) = w_2(\mathbf{x},\mathbf{y},\mathbf{z})(f_{x'y'z'} - \omega_{[y',z']}).$$

Here $f_{x'y'z'} = \omega_{[x',y']} + \omega_{[y',z']} - \omega_{[x',z']}$. To sum up, the $(\cdots)$ part becomes

$$w_2(x,y,z)(\sigma_{xy,xyz} f_{x'y'z'} - f_{[x,y]})$$

Note that $\sigma_{xy,xyz} f_{x'y'z'} = \omega_{xy} + \omega_{yz} + \omega_{zx} = f_{xyz}$, therefore,

$$\begin{aligned}
[\mathbf{B}_2 \mathbf{W}_2 \mathbf{B}_2^\top \boldsymbol{\omega}]_{[x,y]} &= w_1(\mathbf{x},\mathbf{y})\omega_{[x,y]} + \sum_{z \notin \{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})(\sigma_{xy,xyz} f_{x'y'z'} - \omega_{[x,y]}) \\
&= \cancel{w_1(\mathbf{x},\mathbf{y})\omega_{[x,y]}} - \cancel{\sum_{z \notin \{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})\omega_{[x,y]}} + \cdots \\
&= \sum_{z \notin \{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}.
\end{aligned}$$

In the above, we use the fact that $\mathbf{W}_1 = \mathrm{diag}(|\mathbf{B}_2|\mathbf{W}_2 \mathbf{1}_{n_2})$. This completes the proof. ■

4.7.2 Proof of Proposition 4.2

We are interested in showing the asymptotic expansion of of the following form for purely *curl/harmonic* flow for some constant $c = c(\mathbf{x},\mathbf{y})$.

$$\int_{\mathcal{M}} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz} \mathsf{d}\mu(\mathbf{z}) = c \int_{x \to y} (\Delta_1 f(\boldsymbol{\gamma}(t)))\boldsymbol{\gamma}'(t)\mathsf{d}t + \mathcal{O}(\varepsilon^3). \tag{4.5}$$

Here $\boldsymbol{\gamma}(t)$ is the parameterization of geodesic curve on the manifold $\mathcal{M}$. From Corollary 2.24 in Supplement 2.6.2, one has $\Delta_1 \zeta_1 = \sum_i \sum_j \frac{\partial^2 f_i}{\partial s_j^2} \mathsf{d}s_i = \sum_i \Delta_0 f_i \mathsf{d}s_i$ in local coordinate $(s_1, \cdots, s_d)$ with $\zeta_1 = \sum_i f_i \mathsf{d}s_i$. Before we proceed, we first show the upper bound for the error by integrating the integral operator around ε^γ balls around x and y. This lemma is the modification of the similar technique that appeared in Lemma 8 of [32].

Lemma 4.5 (Error bound for localization of exponential decay kernel)**.** *Let $0 < \gamma < 1$, given some bounded function g, the integration of the integral operator $\int_{\mathcal{M}} \kappa(\mathbf{x},\mathbf{z})\kappa(\mathbf{y},\mathbf{z})g(\mathbf{z})\mathsf{d}\mathbf{z}$ that is ε^γ far away from points $x, y \in \mathcal{M}$ can be bounded above by $\mathcal{O}(\varepsilon^3)$.*

Proof. First we focus on the domain of the integral. Points $\mathbf{z} \in \mathcal{M}$ that is ε^γ far away from both $\mathbf{x}$ and $\mathbf{y}$ can be written as $\{\mathbf{z} \in \mathcal{M} : \min(\|\mathbf{z}-\mathbf{x}\|, \|\mathbf{z}-\mathbf{y}\|) > \varepsilon^\gamma\}$. Because of the exponential decay of the kernel, one can follow the same technique of Lemma 8 in [32] to bound the integration by,

$$\begin{aligned}
&\left| \frac{1}{\varepsilon^d} \int\limits_{\substack{\mathbf{z}\in\mathcal{M} \\ \min(\|\mathbf{z}-\mathbf{x}\|,\|\mathbf{z}-\mathbf{y}\|)>\varepsilon^\gamma}} \kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right) g(\mathbf{z})\mathsf{d}\mathbf{z} \right| \\
&\leq \frac{\|g\|_\infty}{\varepsilon^d} \int\limits_{\substack{\mathbf{z}\in\mathcal{M} \\ \min(\|\mathbf{z}-\mathbf{x}\|,\|\mathbf{z}-\mathbf{y}\|)>\varepsilon^\gamma}} \left|\kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\right| \left|\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right)\right| \mathsf{d}\mathbf{z} \\
&\leq \frac{\|g\|_\infty \|\kappa\|_\infty}{\varepsilon^d} \int\limits_{\substack{\mathbf{z}\in\mathcal{M} \\ \|\mathbf{z}-\mathbf{x}\|>\varepsilon^\gamma}} \left|\kappa\left(\frac{\|z-x\|^2}{\varepsilon^2}\right)\right| \mathsf{d}\mathbf{z} \\
&\leq \|g\|_\infty \|\kappa\|_\infty \int\limits_{\substack{\mathbf{z}\in\mathcal{M} \\ \|\mathbf{z}\|>\varepsilon^{\gamma-1}}} \left|\kappa(\|\mathbf{z}\|^2)\right| \mathsf{d}\mathbf{z} \leq C\|g\|_\infty Q\left(\varepsilon^{1-\gamma}\right)\exp\left(-\varepsilon^{\gamma-1}\right).
\end{aligned}$$

Last inequality holds by using the exponential decay of the kernel. Here Q is some polynomial. Since $0 < \gamma < 1$, the term is exponentially small and bounded by $\mathcal{O}(\varepsilon^3)$. ■

Therefore, the original integral operator (LHS of (4.5)) becomes,

$$\varepsilon^{-d}\boldsymbol{\mathcal{L}}_1^{\text{up}} f_{xy} = \varepsilon^{-d} \int\limits_{\substack{\mathbf{z}\in\mathcal{M} \\ \max(\|\mathbf{z}-\mathbf{y}\|,\|\mathbf{z}-\mathbf{x}\|)\leq\varepsilon^\gamma}} \left(\kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right)\oint_{\Gamma(x,y,z)}\zeta\right)\mathsf{d}\mathbf{z} + \mathcal{O}(\varepsilon^3).$$

The following lemma, which is based on odd/even function symmetry, will be useful in terms cancellation when proving Proposition 4.2.

Lemma 4.6 (Odd/Even function preservation)**.** *The following integral is* 0 *if g is an odd function,*

$$\int_{\mathbf{z}\in\mathbb{R}^d} \left(\kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right)(g(\mathbf{z}-\mathbf{x})+g(\mathbf{z}-\mathbf{y}))\right)\mathsf{d}\mathbf{z}. \tag{4.6}$$

Proof. This can be shown by changes of variable,

$$\begin{aligned}
&\int_{\mathbf{z}\in\mathbb{R}^d} \left(\kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right)(g(\mathbf{z}-\mathbf{x})+g(\mathbf{z}-\mathbf{y}))\right)\mathsf{d}\mathbf{z}\\
&= \int_{\mathbb{R}^d} h_1(\|\mathbf{z}-\mathbf{x}\|)h_1(\|\mathbf{z}-\mathbf{y}\|)g(\mathbf{z}-\mathbf{x})\mathsf{d}\mathbf{z} + \int_{\mathbb{R}^d} h_1(\|\mathbf{z}-\mathbf{x}\|)h_1(\|\mathbf{z}-\mathbf{y}\|)g(\mathbf{z}-\mathbf{y})\mathsf{d}\mathbf{z}\\
&= \int_{\mathbb{R}^d} \underbrace{h_1(\|\boldsymbol{\alpha}+\mathbf{x}-\mathbf{y}\|)}_{=h_2(\boldsymbol{\alpha},\mathbf{x},\mathbf{y})} h_1(\|\boldsymbol{\alpha}\|)g(\boldsymbol{\alpha})\mathsf{d}\boldsymbol{\alpha} + \int_{\mathbb{R}^d} \underbrace{h_1(\|\boldsymbol{\alpha}+\mathbf{y}-\mathbf{x}\|)}_{=h_2(\boldsymbol{\alpha},\mathbf{y},\mathbf{x})} h_1(\|\boldsymbol{\alpha}\|)g(\boldsymbol{\alpha})\mathsf{d}\boldsymbol{\alpha}\\
&= \int_{\mathbb{R}^d} h_1(\|\boldsymbol{\alpha}\|)g(\boldsymbol{\alpha})\left(h_2(\boldsymbol{\alpha},\mathbf{x},\mathbf{y})+h_2(\boldsymbol{\alpha},\mathbf{y},\mathbf{x})\right)\mathsf{d}\boldsymbol{\alpha}.
\end{aligned}$$

Since $h_2(-\boldsymbol{\alpha},\mathbf{x},\mathbf{y}) = h_2(\boldsymbol{\alpha},\mathbf{y},\mathbf{x})$, this implies $h_3(\boldsymbol{\alpha}) = h_2(\boldsymbol{\alpha},\mathbf{x},\mathbf{y})+h_2(\boldsymbol{\alpha},\mathbf{y},\mathbf{x}) = h_2(-\boldsymbol{\alpha},\mathbf{y},\mathbf{x})+h_2(-\boldsymbol{\alpha},\mathbf{x},\mathbf{y}) = h_3(-\boldsymbol{\alpha})$ is an even function. Therefore, the integration $\int_{\mathbb{R}^d} h_1(\|\boldsymbol{\alpha}\|)h_3(\boldsymbol{\alpha})g(\boldsymbol{\alpha})\mathsf{d}\boldsymbol{\alpha}$ is zero if g is an odd function. ■

We introduce the last lemma that is useful in removing the bias term in the line integral before proving Proposition 4.2. This lemma will be used again in generating the 1-cochain in Supplement 2.5.

Lemma 4.7 (Linear approximation of line integral)**.** *Assume $f \in \mathcal{C}^2$, let $\mathbf{u}(t) = \mathbf{x}+(\mathbf{y}-\mathbf{x})t$ be a parameterization of straight line between node x, y, one has the following error bound,*

$$f(\mathbf{u}(t)) = f(\mathbf{x}) + ((f(\mathbf{y})-f(\mathbf{x}))t + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2). \tag{4.7}$$

Proof. Since $\mathbf{u}(t) = \mathbf{x}+(\mathbf{y}-\mathbf{x})t$, by Taylor expansion on f, one has

$$f(\mathbf{x}+(\mathbf{y}-\mathbf{x})t) = f(\mathbf{x}) + (\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x})\cdot t + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2).$$

Additionally, $(\mathbf{y}-\mathbf{x})^\top \nabla f(\mathbf{x})$ is the directional derivative, and can be approximate by $f(\mathbf{y}) - f(\mathbf{x}) + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2)$ by Taylor expansion. Therefore

$$f(\mathbf{u}(t)) = f(\mathbf{x} + (\mathbf{y}-\mathbf{x})t) = f(\mathbf{x}) + (f(\mathbf{y}) - f(\mathbf{x}))t + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2).$$

This completes the proof. ■

The outline of the proof is as follow. We first prove the asymptotic expansion of the integral operator $\varepsilon^{-d}\int w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz} \mathsf{d}\mu(\mathbf{z})$. Later on, we bound the error of approximating the manifold by tangent bundles. Lastly, the asymptotic expansion of the integral operator is obtained by incorporating the error terms of tangent plane approximation and the expansion in $\mathbb{R}^d$. The following lemma is the first step, i.e., the asymptotic expansion in $\mathbb{R}^d$.

Lemma 4.8 (Asymptotic expansion of $\boldsymbol{\mathcal{L}}_1^{\mathrm{up}}$ in $\mathbb{R}^d$)**.** *Under Assumption 4.1–4.2, further assume that the corresponding 1-form ζ of 1-cochain $\boldsymbol{\omega}$ is divergence-free and $\zeta = (f_1, \cdots, f_d) \in \mathcal{C}^4(\mathcal{M})$. Let $\mathbf{u}(t) = \mathbf{x} + (\mathbf{y}-\mathbf{x})t$ for t in $[0,1]$ be a parameterization of straight line between nodes x, y, and $\mathbf{u}'(t) = \mathsf{d}\mathbf{u}(t)/\mathsf{d}t$. One has the following asymptotic expansion*

$$\begin{aligned}\varepsilon^{-d}\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}\mathsf{d}\mu(\mathbf{z}) \;=\; & \varepsilon^2(c_2 - c_1(\mathbf{x},\mathbf{y}))\int_0^1 \left[\Delta_1\zeta(\mathbf{u}(t))\right]^\top \mathbf{u}'(t)\mathsf{d}t \\ & + \varepsilon^2 c_3 \int_0^1 \sum_i \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_i^2}(\mathbf{y}-\mathbf{x})_i \mathsf{d}t \\ & + \mathcal{O}(\delta^2) + \mathcal{O}(\varepsilon^3).\end{aligned} \tag{4.8}$$

Here $c_1(\mathbf{x},\mathbf{y}) = \frac{2}{3}\int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}-\mathbf{x}\|^2)\kappa(\|\mathbf{z}-\mathbf{y}\|^2)((z_1-x_1)^2+(z_1-y_1)^2)\mathsf{d}\mathbf{z}$, with z_i be the first coordinate of vector $\mathbf{z}$, $c_2 = \frac{4}{3}\int_{\mathbf{z}\in\mathbb{R}^d}\kappa(\|\mathbf{z}\|^2)\kappa'(\|\mathbf{z}\|^2) z_1^2 z_2^2 \mathsf{d}\mathbf{z}$, and $c_3 = \frac{2}{3}\int_{\mathbf{z}\in\mathbb{R}^d}\kappa(\|\mathbf{z}\|^2)\kappa'(\|\mathbf{z}\|^2)(z_1^4 - 3z_1^2 z_2^2)\mathsf{d}\mathbf{z}$. With the choice of exponential kernel $\kappa(u) = \exp(-u)$, c_3 will be zero thus the bias term can be removed.

Proof. First we define $\mathbf{u}(t), \mathbf{v}(t), \mathbf{w}(t)$ as in Figure 4.7, with $\mathbf{u}(0) = \mathbf{x}, \mathbf{u}(1) = \mathbf{y}$, $\mathbf{v}(0) = \mathbf{y}, \mathbf{v}(1) = \mathbf{z}$, and $\mathbf{w}(0) = \mathbf{z}, \mathbf{w}(1) = \mathbf{x}$. In our analysis, we do not use the conventional

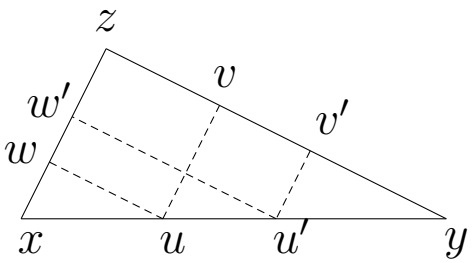

Figure 4.7: Taylor expansion coordinate for simplex x, y, z.

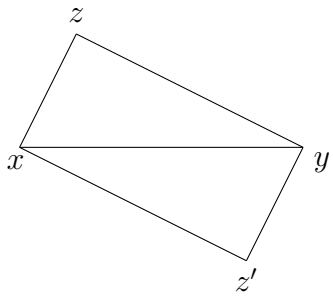

Figure 4.8: Two nodes z and z' that will cancel each other.

parametrization of curve with *unit-speed* since the notation will be more concise this way. One can always use the convention without changing any conclusions. Further define $\Gamma(x, y, z)$ be the loop connecting nodes x, y, z, i.e., $\Gamma(x, y, z) = \{\mathbf{u}(t), \mathbf{v}(t), \mathbf{w}(t)\}$. The loop integral f_{xyz} becomes

$$\oint_{\Gamma(x,y,z)} \zeta_1 = \int_0^1 \mathbf{f}(\mathbf{u}(t))^\top (\mathbf{y} - \mathbf{x}) \mathsf{d}t + \int_0^1 \mathbf{f}(\mathbf{v}(t))^\top (\mathbf{z} - \mathbf{y}) \mathsf{d}t + \int_0^1 \mathbf{f}(\mathbf{w}(t))^\top (\mathbf{x} - \mathbf{z}) \mathsf{d}t.$$

One can do the following coordinate-wise expansion up to 2nd order terms. With slightly abuse of notation, let $\mathbf{v} = \mathbf{v}(t)$ and $\mathbf{u} = \mathbf{u}(1-t)$, and $s_1, \cdots, s_d$ represents the coordinate system.

$$f_i(\mathbf{v}(t)) = f_i(\mathbf{u}(1-t)) + \sum_j \frac{\partial f_i(\mathbf{u})}{\partial s_j} (\mathbf{v} - \mathbf{u})_j + \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u})}{\partial s_j \partial s_k} (\mathbf{v} - \mathbf{u})_j (\mathbf{v} - \mathbf{u})_k + \mathcal{O}(\varepsilon^3).$$

$$f_i(\mathbf{w}(t)) = f_i(\mathbf{u}(1-t)) + \sum_j \frac{\partial f_i(\mathbf{u})}{\partial s_j} (\mathbf{w} - \mathbf{u})_j + \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u})}{\partial s_j \partial s_k} (\mathbf{w} - \mathbf{u})_j (\mathbf{w} - \mathbf{u})_k + \mathcal{O}(\varepsilon^3).$$

With the above expansion, we denote the 0th, 1st, and 2nd order term to be the following

$$f_{xyz} = \int_\Gamma \zeta = f_{xyz}^{(0)} + f_{xyz}^{(1)} + f_{xyz}^{(2)} + \mathcal{O}(\varepsilon^3). \tag{4.9}$$

Claim (Constant term)**.** *The loop integral of constant term $f^{(0)}_{xyz}$ is zero.*

The above claim is true because

$$\begin{aligned} f^{(0)}_{xyz} &= \int_0^1 \sum_i f_i(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i \mathsf{d}t + \int_0^1 \sum_i f_i(\mathbf{u}(1-t))(\mathbf{z}-\mathbf{y})_i \mathsf{d}t + \int_0^1 \sum_i f_i(\mathbf{u}(1-t))(\mathbf{x}-\mathbf{z})_i \mathsf{d}t \\ &= \int_0^1 \sum_i f_i(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i \mathsf{d}t - \int_1^0 \sum_i f_i(\mathbf{u}(t))(\mathbf{z}-\mathbf{y})_i \mathsf{d}t - \int_1^0 \sum_i f_i(\mathbf{u}(t))(\mathbf{x}-\mathbf{z})_i \mathsf{d}t \\ &= \int_0^1 \sum_i f_i(u) \cdot (\mathbf{y}-\mathbf{x}+\mathbf{z}-\mathbf{y}+\mathbf{x}-\mathbf{z})_i \mathsf{d}t = 0. \end{aligned}$$

The second equation holds by change of variable from $1-t \to t$ for the second and third term.

Claim (First order term)**.** *The first order term $f^{(1)}_{xyz}$ can be decomposed into the following four terms,*

$$f^{(1)}_{xyz} = \int_0^1 \left(\ell_1(\mathbf{z}-\mathbf{x})(1-t) - \ell_1(\mathbf{z}-\mathbf{y})t - \ell_2(\mathbf{z}-\mathbf{x})(1-t) - \ell_2(\mathbf{z}-\mathbf{y})t\right) \mathsf{d}t. \tag{4.10}$$

We start with the fact that $(\mathbf{v}(t)-\mathbf{u}(1-t)) = t(\mathbf{z}-\mathbf{x})$, and $(\mathbf{w}(t)-\mathbf{u}(1-t)) = (1-t)(\mathbf{z}-\mathbf{y})$.

$$\int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j}(\mathbf{v}-\mathbf{u})_j(\mathbf{z}-\mathbf{y})_i \mathsf{d}t = \int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j} t(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{y})_i \mathsf{d}t$$

$$= \int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x}+\mathbf{x}-\mathbf{y})_i(1-t)\mathsf{d}t$$

$$= \int_0^1 \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_i(1-t)\mathsf{d}t - \int_0^1 \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{x})_j(\mathbf{y}-\mathbf{x})_i(1-t)\mathsf{d}t$$

$$+ \int_0^1 \sum_i \frac{\partial f_i(\mathbf{u}(t))}{\partial s_i}(\mathbf{z}-\mathbf{x})_i^2(1-t)\mathsf{d}t - \int_0^1 \sum_i \frac{\partial f_i(\mathbf{u}(t))}{\partial s_i}(\mathbf{z}-\mathbf{x})_i(\mathbf{y}-\mathbf{x})_i(1-t)\mathsf{d}t.$$

The last two terms can be cancelled out because of our assumption (ζ is *curl* flow, i.e.,

$\delta\zeta = 0$). Similarly,

$$\int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j}(\mathbf{w}-\mathbf{u})_j(\mathbf{x}-\mathbf{z})_i \mathsf{d}t = \int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j}(\mathbf{z}-\mathbf{y})_j(1-t)(\mathbf{x}-\mathbf{z})_i \mathsf{d}t$$
$$= \int_0^1 \sum_i \sum_j \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{y})_j(\mathbf{x}-\mathbf{y}+\mathbf{y}-\mathbf{z})_i t \mathsf{d}t$$
$$= -\int_0^1 \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{y})_i(\mathbf{z}-\mathbf{y})_j t \mathsf{d}t - \int_0^1 \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{y})_j(\mathbf{y}-\mathbf{x})_i t \mathsf{d}t$$
$$-\int_0^1 \sum_i \frac{\partial f_i(\mathbf{u}(t))}{\partial s_i}(\mathbf{z}-\mathbf{y})_i^2 t \mathsf{d}t - \int_0^1 \sum_i \frac{\partial f_i(\mathbf{u}(t))}{\partial s_i}(\mathbf{z}-\mathbf{y})_i(\mathbf{y}-\mathbf{x})_i t \mathsf{d}t.$$

By defining $\ell_1(\mathbf{v}) = \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j} v_i v_j$ and $\ell_2(\mathbf{v}) = \sum_i \sum_{j\neq i} \frac{\partial f_i(\mathbf{u}(t))}{\partial s_j} v_j(\mathbf{y}-\mathbf{x})_i$, (4.10) is satisfied.

Claim (Second order term)**.** *The second order term $f^{(2)}_{xyz}$ can be decomposed into the following 5 terms,*

$$\begin{aligned} f^{(2)}_{xyz} = \int_0^1 [&- q_1(\mathbf{z}-\mathbf{x})(1-t)^2 - q_1(\mathbf{z}-\mathbf{y})t^2 - q_2(\mathbf{z}-\mathbf{x})(1-t)^2 - q_2(\mathbf{z}-\mathbf{y})t^2 \\ &+ q_3(\mathbf{z}-\mathbf{x})(1-t)^2 - q_3(\mathbf{z}-\mathbf{y})t^2] \mathsf{d}t. \end{aligned} \tag{4.11}$$

Using similar trick as before

$$
\begin{aligned}
&\int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(1-t))}{\partial s_j \partial s_k}(\mathbf{v}-\mathbf{u})_j(\mathbf{v}-\mathbf{u})_k(\mathbf{z}-\mathbf{y})\mathsf{d}t \\
&= \int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(1-t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_k(\mathbf{z}-\mathbf{y})_i t^2 \mathsf{d}t \\
&= \int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_k(\mathbf{z}-\mathbf{x}+\mathbf{x}-\mathbf{y})_i (1-t)^2 \mathsf{d}t \\
&= -\int_0^1 \sum_i \sum_j \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j^2}(\mathbf{z}-\mathbf{x})_j^2(\mathbf{y}-\mathbf{x})_i(1-t)^2\mathsf{d}t \\
&\quad -\int_0^1 \sum_i \sum_j \sum_{k\neq j} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_k(\mathbf{y}-\mathbf{x})_i(1-t)^2\mathsf{d}t \\
&\quad +\int_0^1 \sum_{i,j,k} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_k(\mathbf{z}-\mathbf{x})_i(1-t)^2\mathsf{d}t.
\end{aligned}
$$

And,

$$
\begin{aligned}
&\int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(1-t))}{\partial s_j \partial s_k}(\mathbf{w}-\mathbf{u})_j(\mathbf{w}-\mathbf{u})_k(\mathbf{x}-\mathbf{z})_i\mathsf{d}t \\
&= \int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(1-t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{y})_j(\mathbf{z}-\mathbf{y})_k(\mathbf{x}-\mathbf{z})_i(1-t)^2\mathsf{d}t \\
&= \int_0^1 \sum_i \sum_j \sum_k \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{y})_j(\mathbf{z}-\mathbf{y})_k(\mathbf{x}-\mathbf{y}+\mathbf{y}-\mathbf{z})_i t^2\mathsf{d}t \\
&= -\int_0^1 \sum_i \sum_j \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j^2}(\mathbf{z}-\mathbf{y})_j^2(\mathbf{y}-\mathbf{x})_i t^2\mathsf{d}t \\
&\quad -\int_0^1 \sum_i \sum_j \sum_{k\neq j} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{y})_j(\mathbf{z}-\mathbf{y})_k(\mathbf{y}-\mathbf{x})_i t^2\mathsf{d}t \\
&\quad -\int_0^1 \sum_{i,j,k} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{y})_j(\mathbf{z}-\mathbf{y})_k(\mathbf{z}-\mathbf{y})_i t^2\mathsf{d}t.
\end{aligned}
$$

By letting $q_1(\mathbf{v}) = \sum_{i,j} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j^2} v_j^2(\mathbf{y}-\mathbf{x})_i$, $q_2(\mathbf{v}) = \sum_{i,j\neq k} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k} v_j v_k(\mathbf{y}-\mathbf{x})_i$, and $q_3(\mathbf{v}) = \sum_{i,j,k} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k} v_i v_j v_k$, (4.11) is satisfied.

We next evaluate the integral operator. Note that since there is t dependency in the line integral, one cannot directly use Lemma 4.6 to cancel out the unwanted terms. To solve this issue, we introduce a mirror node z' of z as in Figure 4.8. The node has a property that $\mathbf{z}'-\mathbf{x} = -(\mathbf{z}-\mathbf{y})$ and $\mathbf{z}'-\mathbf{y} = -(\mathbf{z}-\mathbf{x})$, which will be shown to be useful in the cancellation. With the above construction, the integral of $w_2(\mathbf{x},\mathbf{y},\mathbf{z})$ becomes

$$\begin{aligned}\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}\mathsf{d}\mathbf{z} &= \frac{1}{2}\left(\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}\mathsf{d}\mathbf{z} + \int_{\mathbf{z}'\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z}') f_{xyz'}\mathsf{d}\mathbf{z}'\right)\\ &= \frac{1}{2}\left(\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z})(f_{xyz}+f_{xyz'})\mathsf{d}\mathbf{z}\right).\end{aligned}$$

The $1/2$ is dropped for simplicity. We start with the first order term $\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z})(f^{(1)}_{xyz}+ f^{(1)}_{xyz'})\mathsf{d}\mathbf{z}$.

$$\begin{aligned}f^{(1)}_{xyz} + f^{(1)}_{xyz'} &= -\ell_1(\mathbf{z}-\mathbf{y})t - \ell_1(\mathbf{z}'-\mathbf{y})t\\ &\quad -\ell_2(\mathbf{z}-\mathbf{x})(1-t) - \ell_2(\mathbf{z}'-\mathbf{x})(1-t)\\ &\quad -\ell_2(\mathbf{z}-\mathbf{y})t - \ell_2(\mathbf{z}'-\mathbf{y})t.\end{aligned}$$

Since $\ell_1(\mathbf{z}'-\mathbf{x}) = \ell_1(\mathbf{z}-\mathbf{y})$, $\ell_1(\mathbf{z}'-\mathbf{y}) = \ell_1(\mathbf{z}-\mathbf{x})$, $\ell_2(\mathbf{z}'-\mathbf{x}) = -\ell_2(\mathbf{z}-\mathbf{y})$ and $\ell_2(\mathbf{z}'-\mathbf{y}) = -\ell_2(\mathbf{z}-\mathbf{x})$. Additionally, ℓ_1 is an odd function, using Lemma 4.6 one has,

$$\int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}-\mathbf{x}\|^2/\varepsilon^2)\kappa(\|\mathbf{z}-\mathbf{x}\|^2/\varepsilon^2)\int_0^1 \ell_1(\mathbf{z}-\mathbf{x}) + \ell_1(\mathbf{z}-\mathbf{y})(1-t)\mathsf{d}t\mathsf{d}\mathbf{z} = 0.$$

The ℓ_1 terms consisting of t is also zero by the same justification. For the ℓ_2 terms, one

can do the following expansion

$$
\begin{aligned}
&-\int_0^1 \ell_2(\mathbf{z}-\mathbf{x})(1-t)-\ell_2(\mathbf{z}'-\mathbf{x})(1-t)-\ell_2(\mathbf{z}-\mathbf{y})t-\ell_2(\mathbf{z}'-\mathbf{y})t\mathsf{d}t\\
=&-\int_0^1\sum_{i\neq j}\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{x})_j(\mathbf{y}-\mathbf{x})_i(1-t)\mathsf{d}t-\int_0^1\sum_{i\neq j}\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}-\mathbf{y})_j(\mathbf{y}-\mathbf{x})_i t\mathsf{d}t\\
&-\int_0^1\sum_{i\neq j}\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}'-\mathbf{x})_j(\mathbf{y}-\mathbf{x})_i(1-t)\mathsf{d}t-\int_0^1\sum_{i\neq j}\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}(\mathbf{z}'-\mathbf{y})_j(\mathbf{y}-\mathbf{x})_i t\mathsf{d}t\\
=&-\int_0^1\sum_{i\neq j}\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}\left((\mathbf{z}-\mathbf{x})_j(\mathbf{y}-\mathbf{x})_i-(\mathbf{z}-\mathbf{y})_j(\mathbf{y}-\mathbf{x})_i\right)(1-2t)\mathsf{d}t\\
=&-\int_0^1\sum_{i\neq j}\left(\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}-\frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j}\right)(\mathbf{y}-\mathbf{x})_j(\mathbf{y}-\mathbf{x})_i(1-t)\mathsf{d}t=\mathcal{O}(\delta^2).
\end{aligned}
$$

The last equality holds by doing Taylor expansion on $\frac{\partial f_i(\mathbf{u}(t))}{\partial s_j}-\frac{\partial f_i(\mathbf{u}(1-t))}{\partial s_j}=\mathcal{O}(\|y-x\|)$. Since we build a VR complex with maximum distance δ, $\|y-x\|$ will be upper bounded by δ. One $(\mathbf{y}-\mathbf{x})_i$ is left out of the big-O notation to be in the line integral. Integrating this term with $w_2(\mathbf{x},\mathbf{y},\mathbf{z})$ results in a $\mathcal{O}(\delta^2)$ term.

Similar trick applied for the second order term. The q_1 terms can be reduced as follows,

$$
\begin{aligned}
&q_1(\mathbf{z}-\mathbf{x})(1-t)^2+q_1(\mathbf{z}'-\mathbf{x})(1-t)^2+q_1(\mathbf{z}-\mathbf{y})t^2+q_1(\mathbf{z}'-\mathbf{y})t^2\\
=&\,q_1(\mathbf{z}-\mathbf{x})(1-t)^2+q_1(\mathbf{z}-\mathbf{y})(1-t)^2+q_1(\mathbf{z}-\mathbf{y})t^2+q_1(\mathbf{z}-\mathbf{x})t^2\\
=&\,(q_1(\mathbf{z}-\mathbf{x})+q_1(\mathbf{z}-\mathbf{y}))(t^2+(1-t)^2).
\end{aligned}
$$

Therefore, integrating the q_1 terms yields

$$
\begin{aligned}
&-\int_{\mathbf{z}\in\mathbb{R}^d} w_2(\mathbf{x},\mathbf{y},\mathbf{z})\int_0^1(q_1(\mathbf{z}-\mathbf{x})+q_1(\mathbf{z}-\mathbf{y}))(t^2+(1-t)^2)\mathsf{d}t\mathsf{d}\mathbf{z}\\
=&\,\varepsilon^2 c_1(\mathbf{x},\mathbf{y})\int_0^1\sum_i\sum_j\frac{\partial^2 f_i}{\partial s_j^2}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i(t^2+(1-t)^2)\mathsf{d}t.
\end{aligned}
$$

With $c_1(\mathbf{x},\mathbf{y})=\int_{\mathbf{z}\in\mathbb{R}^d}\kappa(\|\mathbf{z}-\mathbf{x}\|^2/\varepsilon^2)\kappa(\|\mathbf{z}-\mathbf{y}\|^2/\varepsilon^2)((\mathbf{z}-\mathbf{x})_1^2+(\mathbf{z}-\mathbf{y})_1^2)^2\mathsf{d}\mathbf{z}$. The $t^2+(1-t)^2=1-2t(1-t)$ term in the line integral can be removed by the following technique.

Let $g_{ij}(\mathbf{u}(t)) = \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j^2}$, and let $\tau = t - 1/2$. Considering the integral term contains $t(1-t)$, we have

$$\int_0^1 g_{ij}(\mathbf{u}(t))t(1-t)\mathsf{d}t = \int_0^1 g_{ij}\left(\frac{\mathbf{x}+\mathbf{y}}{2} + (\mathbf{y}-\mathbf{x})\tau\right)\left(\frac{1}{4} - \tau^2\right)\mathsf{d}\tau.$$

The $1/4$ terms becomes a unbiased line integral, i.e., $\frac{1}{4}\int_0^1 g_{ij}(\mathbf{u}(t))\mathsf{d}t$, therefore we can focus only on the τ^2 part. Taylor expanding the g_{ij}, one has,

$$\int_{-1/2}^{1/2} \tau^2 \left[g_{ij}\left(\frac{\mathbf{x}+\mathbf{y}}{2}\right) + \cancel{\tau(\mathbf{y}-\mathbf{x})^\top \nabla g_{ij}} + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2)\right]\mathsf{d}\tau = \frac{1}{12}\cdot g_{ij}\left(\frac{\mathbf{x}+\mathbf{y}}{2}\right).$$

The first order term becomes zero using the property of odd function, and the $1/12$ comes from $\int_{-1/2}^{1/2}\tau^2\mathsf{d}\tau$. By Jensen's inequality, $g_{ij}((\mathbf{x}+\mathbf{y})/2) = \frac{1}{2}(g_{ij}(\mathbf{x}) + g_{ij}(\mathbf{y})) + \mathcal{O}(\|\mathbf{y}-\mathbf{x}\|^2)$. Therefore,

$$\begin{aligned}
\sum_i \int_{-\frac{1}{2}}^{\frac{1}{2}} g_{ij}(\mathbf{u}(\tau))(\mathbf{y}-\mathbf{x})_i\tau^2\mathsf{d}\tau &= \frac{1}{12}\cdot\frac{1}{2}\left(g_{ij}(\mathbf{x}) + g_{ij}(\mathbf{y})\right)(\mathbf{y}-\mathbf{x})_i + \mathcal{O}(\delta^2) \\
&= \frac{1}{12}\sum_i \int_0^1 [g_{ij}(\mathbf{x}) + (g_{ij}(\mathbf{y}) - g_{ij}(\mathbf{x}))t](\mathbf{y}-\mathbf{x})_i\mathsf{d}t + O(\delta^2) \\
&= \frac{1}{12}\sum_i \int_0^1 g_{ij}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i\mathsf{d}t + \mathcal{O}(\delta^2).
\end{aligned}$$

Last equality holds from Lemma 4.7. Now we have,

$$\int_0^1 \sum_i g_{ij}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i t(1-t)\mathsf{d}t = \left(\frac{1}{4} - \frac{1}{12}\right)\int_0^1 \sum_i g_{ij}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i\mathsf{d}t + \mathcal{O}(\delta^2).$$

Using the fact that $t^2 + (1-t)^2 = 1 - 2t(1-t)$, the q_1 terms become

$$\begin{aligned}
&-\varepsilon^2 c_1(\mathbf{x},\mathbf{y})\int_0^1 \sum_i\sum_j \frac{\partial^2 f_i}{\partial s_j^2}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i(t^2+(1-t)^2)\mathsf{d}t \\
= &-\frac{2}{3}\varepsilon^2 c_1(\mathbf{x},\mathbf{y})\int_0^1 \sum_i\sum_j \frac{\partial^2 f_i}{\partial s_j^2}(\mathbf{u}(t))(\mathbf{y}-\mathbf{x})_i\mathsf{d}t + \mathcal{O}(\varepsilon^2\delta^2).
\end{aligned}$$

By choosing $\delta = c\varepsilon^{3/2}$, the second term is in the order of $\mathcal{O}(\varepsilon^2\delta^2) = \mathcal{O}(\varepsilon^5)$. This corresponds to a higher order term therefore is negligible.

Remark. Note that the 2/3 corresponds to $\int_0^1 t^2 + (1-t)^2 dt$. The result can be interpreted as follows. When $\mathbf{x}$ is sufficiently close to $\mathbf{y}$, the $g_{ij}(\mathbf{u}(t))$ term is roughly a constant therefore the integral can be approximately done separately, i.e., $\int_0^1 g_{ij}(\mathbf{u}(t))(t^2+(1-t)^2)\mathsf{d}t \approx \int_0^1 (t^2 + (1-t)^2)\mathsf{d}t \int_0^1 g_{ij}(\mathbf{u}(t))\mathsf{d}t = \frac{2}{3}\int_0^1 g_{ij}(\mathbf{u}(t))\mathsf{d}t$.

The q_2 term has the same property as q_1, i.e.,

$$\begin{aligned}
& q_2(\mathbf{z}-\mathbf{x})(1-t)^2 + q_2(\mathbf{z}'-\mathbf{x})(1-t)^2 + q_2(\mathbf{z}-\mathbf{y})t^2 + q_2(\mathbf{z}'-\mathbf{y})t^2 \\
=\ & q_2(\mathbf{z}-\mathbf{x})(1-t)^2 + q_2(\mathbf{z}-\mathbf{y})(1-t)^2 + q_2(\mathbf{z}-\mathbf{y})t^2 + q_2(\mathbf{z}-\mathbf{x})t^2 \\
=\ & (q_2(\mathbf{z}-\mathbf{x}) + q_2(\mathbf{z}-\mathbf{y}))(t^2+(1-t)^2).
\end{aligned}$$

However, since q_2 is an odd function therefore this term becomes zero by Lemma 4.6. Now we turn to the q_3 term,

$$\begin{aligned}
& q_3(\mathbf{z}-\mathbf{x})(1-t)^2 + q_3(\mathbf{z}'-\mathbf{x})(1-t)^2 - q_3(\mathbf{z}-\mathbf{y})t^2 - q_3(\mathbf{z}'-\mathbf{y})t^2 \\
=\ & q_3(\mathbf{z}-\mathbf{x})(1-t)^2 - q_3(\mathbf{z}-\mathbf{y})(1-t)^2 - q_3(\mathbf{z}-\mathbf{y})t^2 + q_3(\mathbf{z}-\mathbf{x})t^2 \\
=\ & (q_3(\mathbf{z}-\mathbf{x}) - q_3(\mathbf{z}-\mathbf{y}))(t^2+(1-t)^2).
\end{aligned}$$

Note that since q_3 is an odd function, by Lemma 4.6 the following term will not be zero.

$$\begin{aligned}
& \int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}-\mathbf{x}\|^2/\varepsilon^2)\kappa(\|\mathbf{z}-\mathbf{y}\|^2/\varepsilon^2) \int_0^1 \sum_{i,j,k} (q_3(\mathbf{z}-\mathbf{x}) - q_3(\mathbf{z}-\mathbf{y}))(t^2+(1-t)^2)\mathsf{d}t\mathsf{d}\mathbf{z} \\
=\ & \frac{4}{3}\int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}-\mathbf{x}\|^2/\varepsilon^2)\kappa(\|\mathbf{z}-\mathbf{y}\|^2/\varepsilon^2) \int_0^1 \sum_{i,j,k} q_3(\mathbf{z}-\mathbf{x})\mathsf{d}t\mathsf{d}\mathbf{z} + \mathcal{O}(\varepsilon^3\delta).
\end{aligned}$$

Last equality holds from the fact that $\mathbf{z}-\mathbf{x}$ and $\mathbf{z}-\mathbf{y}$ are symmetric, and the technique of removing $t^2+(1-t)^2$ used in q_1 terms. In the following, we show that by choosing δ decrease slightly faster than ε, one is possible to reduce the term to a Δ_1 term. We started

with Taylor expansion on the kernel function $\kappa(\|\mathbf{z}-\mathbf{y}\|)$,

$$\kappa\left(\frac{\|\mathbf{z}-\mathbf{y}\|^2}{\varepsilon^2}\right) = \kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right) + \varepsilon^{-1}\kappa'\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)\left(\frac{\mathbf{z}-\mathbf{x}}{\varepsilon}\right)^\top(\mathbf{x}-\mathbf{y}) + \varepsilon^{-2}Q_2(\mathbf{z}-\mathbf{x}) + \mathcal{O}(\delta^3/\varepsilon^3).$$

The zeroth and second order terms are zero since q_3 is an odd function. The third order (and higher) terms can be ignored since integrating q_3 gives us a ε^3, resulting in the third (and higher) terms bounded by $\mathcal{O}(\delta^2) = \mathcal{O}(\varepsilon^3)$. We now focus on the first order term.

$$\varepsilon^2 \int_{\mathbf{z}\in\mathbb{R}^d} \kappa\kappa' \int_0^1 \sum_{i,j,k,\ell} \frac{\partial^2 f_i(\mathbf{u}(t))}{\partial s_j \partial s_k}(\mathbf{z}-\mathbf{x})_i(\mathbf{z}-\mathbf{x})_j(\mathbf{z}-\mathbf{x})_k(\mathbf{z}-\mathbf{x})_\ell(\mathbf{x}-\mathbf{y})_\ell \mathsf{d}\mathbf{z}.$$

The following four conditions will make the above integral non-zero: (i) $i=j\neq k=\ell$, (ii) $i=k\neq j=\ell$, (iii) $i=\ell\neq j=k$, and (iv) $i=j=k=\ell$. First we inspect condition (i), the integration becomes

$$\varepsilon^2 c_2 \int_0^1 \sum_{i\neq k} \frac{\partial^2 f_i}{\partial s_i \partial s_k}(\mathbf{x}-\mathbf{y})_k \mathsf{d}t = \varepsilon^2 c_2 \left(\cancel{\int_0^1 \sum_{i,k} \frac{\partial^2 f_i}{\partial s_i \partial s_k}(\mathbf{x}-\mathbf{y})_k \mathsf{d}t} - \int_0^1 \sum_i \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t \right).$$

The first term is zero from the fact that ζ is *divergence-free*, i.e., $\mathsf{d}\delta\zeta = 0$ and $\sum_j \frac{\partial^2 f_j}{\partial s_i \partial s_j} = 0\,\forall\, i \in [d]$. Therefore, the first term can be cancelled, with only $\partial^2 f_i/\partial s_i^2$ term remains. Similarly,

$$\varepsilon^2 c_2 \int_0^1 \sum_{i\neq j} \frac{\partial^2 f_i}{\partial s_j \partial s_i}(\mathbf{x}-\mathbf{y})_j \mathsf{d}t = \varepsilon^2 c_2 \left(\cancel{\int_0^1 \sum_{i,j} \frac{\partial^2 f_i}{\partial s_j \partial s_i}(\mathbf{x}-\mathbf{y})_j \mathsf{d}t} - \int_0^1 \sum_i \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t \right).$$

For condition (iii), $i = \ell \neq j = k$,

$$\begin{aligned}\varepsilon^2 c_2 \int_0^1 \sum_{i\neq j} \frac{\partial^2 f_i}{\partial s_j^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t &= \varepsilon^2 c_2 \int_0^1 \sum_{i,j} \frac{\partial^2 f_i}{\partial s_j^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t - \varepsilon^2 c_2 \int_0^1 \sum_{i} \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t \\ &= \varepsilon^2 c_2 \int_0^1 [\Delta_1 \zeta(\mathbf{u}(t))]^\top \mathbf{u}'(t) \mathsf{d}t - \varepsilon^2 c_2 \int_0^1 \sum_{i} \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t.\end{aligned}$$

Last equality holds by Corollary 2.24 and the assumption that ζ is a *divergence-free* flow. Here $c_2 = \int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}\|^2)\kappa'(\|\mathbf{z}\|^2) z_1^2 z_2^2 \mathsf{d}\mathbf{z}$. The last condition is when $i = j = k = \ell$, that produces a term with $\varepsilon^2 c_3 \int_0^1 \sum_i \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t$ with $c_3 = \int_{\mathbf{z}\in\mathbb{R}^d} \kappa(\|\mathbf{z}\|^2)\kappa'(\|\mathbf{z}\|^2) z_1^4 \mathsf{d}\mathbf{z}$. Putting things together, the q_3 terms become

$$\frac{2}{3}\varepsilon^2 c_2 \int_0^1 [\Delta_1 \zeta(\mathbf{u}(t))]^\top \mathbf{u}'(t) \mathsf{d}t + \frac{2}{3}\varepsilon^2 (c_3 - 3c_2) \int_0^1 \sum_i \frac{\partial^2 f_i}{\partial s_i^2}(\mathbf{x}-\mathbf{y})_i \mathsf{d}t.$$

In general, $c_3 - 3c_2$ will not be zero, however, for exponential kernel, this term disappears because

$$\begin{aligned}\iint \exp(-2x^2 - 2y^2) x^4 \mathsf{d}x\mathsf{d}y &= \frac{3\pi}{128}; \\ \iint \exp(-2x^2 - 2y^2) x^2 y^2 \mathsf{d}x\mathsf{d}y &= \frac{\pi}{128}.\end{aligned}$$

Implying $c_2 - 3c_1$ is zero. For general dimension, one can use the identity

$$\int_{\mathbf{x}\in\mathbb{R}^d} \exp(-2\|\mathbf{x}\|^2) g(x_1, x_2) \mathsf{d}\mathbf{x} = \left(\frac{\sqrt{2\pi}}{4}\right)^{d-2} \iint \exp(-2x_1^2 - 2x_2^2) g(x_1, x_2) \mathsf{d}x_1 \mathsf{d}x_2.$$

Therefore, $c_2 - 3c_1 = \int_{\mathbf{z}\in\mathbb{R}^d} \exp(-2\|\mathbf{z}\|^2)(z_1^4 - 3z_1^2 z_2^2) \mathsf{d}\mathbf{z} = 0$ for exponential kernel. The proof is thus completed by putting things together and renaming the constants $c_i \leftarrow \frac{2}{3} c_i$ for $i = 1, 2, 3$. ■

The second step is to provide the error term induced by change of variable from ambient space $\mathcal{M} \subseteq \mathbb{R}^D$ to local tangent coordinate $\mathcal{T}_{\mathbf{x}}\mathcal{M} \in \mathbb{R}^d$ defined by $\mathbf{x}$. In the following, there are two coordinate systems that we mainly focus on, the *normal coordinate* and *tangent plane coordinate* at $\mathbf{x}$. First, we define $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{M} \subseteq \mathbb{R}^D$ be the points in *ambient space*. We

then let $\mathbf{x}_s, \mathbf{y}_s, \mathbf{z}_s \in \mathbb{R}^d$ be the same set of points in *normal coordinate* in the neighborhood of point $\mathbf{x}$. Same points in *tangent plane coordinate* defined by tangent plane $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ of $\mathbf{x}$ are denoted $\mathbf{x}_p, \mathbf{y}_p, \mathbf{z}_p \in \mathbb{R}^d$. Note that by definition, the origin of these two coordinate systems are $\mathbf{x}$, i.e., $\mathbf{x}_s$, $\mathbf{x}_p$ are zero vectors. The following lemma generalizes the result in [32] for triangular relations $\mathbf{x}, \mathbf{y}, \mathbf{z}$.

Lemma 4.9 (Error term induced from change of coordinates)**.** *Define $Q_{\mathbf{x},m}(\cdot)$ be a homogenous polynomial of order m with coefficient defined by $\mathbf{x}$. Further let $\boldsymbol{\gamma}$ be geodesic curve in $\mathcal{M}$ connecting two points $\mathbf{a} \in \{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$ and $\mathbf{b} \in \{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$. If $\mathbf{y}, \mathbf{z} \in \mathcal{M}$ are in a Euclidean ball of radius ε around $\mathbf{x}$, then for ε sufficiently small, we have the following four approximations.*

$$[\mathbf{z}_s]_i = [\mathbf{z}_p]_i + \mathcal{O}(\varepsilon^3); \tag{4.12a}$$

$$\det\left(\frac{\mathsf{d}\mathbf{z}}{\mathsf{d}\mathbf{z}_p}\right) = 1 + Q_{\mathbf{x},2}(\mathbf{z}_p) + \mathcal{O}(\varepsilon^3); \tag{4.12b}$$

$$\|\mathbf{a} - \mathbf{b}\|^2 = \|\mathbf{a}_p - \mathbf{b}_p\| + Q_{\mathbf{x},4}(\mathbf{a}_p, \mathbf{b}_p) + \mathcal{O}(\varepsilon^5); \tag{4.12c}$$

$$\boldsymbol{\gamma}(t) = \mathbf{a}_p + (\mathbf{b}_p - \mathbf{a}_p)t + \mathcal{O}(\varepsilon^3). \tag{4.12d}$$

Proof. (4.12a) and (4.12b) follows naturally from Lemma 6 & 7 of [32] since there is no triplet-wise $(\mathbf{x}, \mathbf{y}, \mathbf{z})$ relationship. For (4.12c), if either $\mathbf{a} = \mathbf{x}$ or $\mathbf{b} = \mathbf{x}$ the result follows from Lemma 7 of [32]. Now we consider the case that neither $\mathbf{a}$ nor $\mathbf{b}$ are equal to $\mathbf{x}$. Without loss of generality let $\mathbf{a} = \mathbf{z}$ and $\mathbf{b} = \mathbf{y}$. Note that the submanifold in ambient space is locally parameterized by $(\mathbf{v}_p, g(\mathbf{v}_p)) \in \mathbb{R}^D$ for $g : \mathbb{R}^d \to \mathbb{R}^{D-d}$. Since $\mathbf{z}$ is not the original of the normal coordinate system, one can do a Taylor expansion of g from $\mathbf{x}_p$. Denote $\mathbf{s} = (s_1, \cdots, s_d)$ the local coordinate of $\mathcal{T}_{\mathbf{x}}\mathcal{M}$, because $g_i(\mathbf{x}_p) = 0$ and $\frac{\partial g_i(\mathbf{x}_p)}{\partial s_j} = 0$ for $i, j \in [d]$ by definition. We write $g_i(\mathbf{z}_p) = H_{i,\mathbf{x}}(\mathbf{z}_p) + \mathcal{O}(\varepsilon^3)$, and $g_i(\mathbf{y}) = H_{i,\mathbf{x}}(\mathbf{y}_p) + \mathcal{O}(\varepsilon^3)$ by Taylor expansion. Here H is the Hessian of g_i at origin. Therefore,

$\|\mathbf{z}-\mathbf{y}\|^2 = \|\mathbf{z}_p-\mathbf{y}_p\|^2 + \sum_{i=d+1}^{D}(g_i(\mathbf{z}_p)-g_i(\mathbf{y}_p))^2 = \|\mathbf{z}-\mathbf{y}\|^2 + Q_{\mathbf{x},4}(\mathbf{y}_p,\mathbf{z}_p) + \mathcal{O}(\varepsilon^5)$. To prove (4.12d), first note that one can project the geodesic onto $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ with $\mathcal{O}(\varepsilon^3)$ by (4.12a), i.e., $\boldsymbol{\gamma} = \boldsymbol{\gamma}_{\mathcal{T}_{\mathbf{x}}\mathcal{M}}(t) + \mathcal{O}(\varepsilon^3)$. Therefore, we only need to consider the error term caused by approximating the projected geodesic $\boldsymbol{\gamma}_{\mathcal{T}_{\mathbf{x}}\mathcal{M}}(t)$ by a straight line $\mathbf{a}_p + (\mathbf{b}_p - \mathbf{a}_p)t$. Denote $\boldsymbol{\gamma}_{\mathcal{T}_{\mathbf{x}}\mathcal{M}} = \boldsymbol{\gamma}_{\mathcal{T}}$ for simplicity, further let $\mathrm{dist}_{\mathcal{T}}(\mathbf{y},\mathbf{z})$ be the arc length of the projected geodesic $\boldsymbol{\gamma}_{\mathcal{T}}$ connecting $\mathbf{a}_p, \mathbf{b}_p$, from Taylor expansion, one has

$$\boldsymbol{\gamma}_{\mathcal{T}}(t) = \boldsymbol{\gamma}_{\mathcal{T}}(0) + t\boldsymbol{\gamma}'_{\mathcal{T}}(0)\mathrm{dist}_{\mathcal{T}}(\mathbf{a}_p,\mathbf{b}_p) + \frac{1}{2}t^2\mathrm{dist}^2_{\mathcal{T}}(\mathbf{a}_p,\mathbf{b}_p)\boldsymbol{\gamma}''_{\mathcal{T}}(0) + \mathcal{O}(\varepsilon^3). \tag{4.13}$$

The first step is to show that $\gamma''_{\mathcal{T}}(0) = \mathcal{O}(\varepsilon)$. Since $\mathrm{dist}_{\mathcal{T}}(\mathbf{a}_p,\mathbf{b}_p) = \mathcal{O}(\varepsilon)$, if $\gamma''_{\mathcal{T}}(0) = \mathcal{O}(\varepsilon)$, one can bound the second order term by $\mathcal{O}(\varepsilon^3)$. Note that if $\mathbf{a} = \mathbf{x}$, by the definition of geodesic (covariant derivative is normal to $\mathcal{M}$), we have $\gamma''_{\mathcal{T}}(0) = 0$ and the error term is bounded by $\mathcal{O}(\varepsilon^3)$. If $\mathbf{b} = \mathbf{x}$, one can switch $\mathbf{b}$ with $\mathbf{a}$ and same proof can go through. We now deal with the situation when $\mathbf{a}, \mathbf{b} \neq \mathbf{x}$. One can show that the *Levi-Civita connection* of the local (orthonormal) basis vector can be written as a linear combination of the basis vectors by *method of moving frame* [31], with coefficients determined by the local curvature/torsion at that point. The first order approximation of the local basis vector $[\mathbf{e}_{\mathbf{a}}]_i \in \mathbb{R}^D$ for $i \in [d]$ at $\mathbf{a}$ can be approximated by $[\mathbf{e}_{\mathbf{a}}]_i = [\mathbf{e}_{\mathbf{x}}]_i + \mathcal{O}(\varepsilon)$ by *method of moving frame* along the geodesic connecting $\mathbf{x} \to \mathbf{a}$. By Gram-Schmidt process, the basis of the normal space $[\mathbf{n}_{\mathbf{a}}]_j \in \mathbb{R}^D$ at $\mathbf{a}$ for $j = d+1, \cdots, D$, can also be written as a first order approximation of the normal basis at $\mathbf{x}$, i.e., $[\mathbf{n}_{\mathbf{a}}]_j = [\mathbf{n}_{\mathbf{x}}]_j + \mathcal{O}(\varepsilon)$. Since the second derivate of geodesic $\boldsymbol{\gamma}''_{\mathcal{M}}(0)$ at $\mathbf{a}$ is in the normal space $\mathcal{T}^{\perp}_{\mathbf{a}}\mathcal{M}$, the covariant derivative of the projected geodesic $\boldsymbol{\gamma}''(0)$ at $\mathbf{a}$ onto $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ can be bounded by $\mathcal{O}(\varepsilon)$ if the manifold $\mathcal{M}$ has bounded curvature. The term consisting $\boldsymbol{\gamma}''(0)$ can thus be bounded by $\mathcal{O}(\varepsilon^3)$ if $\mathbf{a}, \mathbf{b}$ are sufficiently close to $\mathbf{x}$. Hence, (4.13) becomes

$$\boldsymbol{\gamma}_{\mathcal{T}}(t) = \boldsymbol{\gamma}_{\mathcal{T}}(0) + t\boldsymbol{\gamma}'_{\mathcal{T}}(0)\mathrm{dist}_{\mathcal{T}}(\mathbf{a}_p,\mathbf{b}_p) + \mathcal{O}(\varepsilon^3).$$

Plugging in $t = 1$ gives us $\boldsymbol{\gamma}'(0)\text{dist}_{\mathcal{T}}(\mathbf{a}_p, \mathbf{b}_p) = \mathbf{b}_p - \mathbf{a}_p + \mathcal{O}(\varepsilon^3)$. The following approximation of $\boldsymbol{\gamma}(t)$ thus holds.

$$\boldsymbol{\gamma}(t) = \mathbf{a}_p + (\mathbf{b}_p - \mathbf{a}_p)t + \mathcal{O}(\varepsilon^3).$$

This completes the proof. ■

With the above two lemmas in hand, we can finally start the proof of Proposition 4.2.

Proof of Proposition 4.2. First note that from (4.12d), the geodesic distance can be changed into straight line in local tangent plane $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ with error $\mathcal{O}(\varepsilon^3)$. Therefore, $f_{xyz} = \oint_{\mathcal{M}} \zeta = f_{x_p y_p z_p} + \mathcal{O}(\varepsilon^3)$. Similar expansion as in Lemma 4.8 therefore holds. Let $\kappa_\varepsilon(\mathbf{z}, \mathbf{x}) = \kappa\left(\frac{\|\mathbf{z}-\mathbf{x}\|^2}{\varepsilon^2}\right)$, from the similar construction as Lemma 4.8,

$$\begin{aligned}
&\varepsilon^{-d}\int_{\mathbf{z}\in\mathcal{M}} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}\mathsf{d}\mathbf{z} \overset{(i)}{=} \varepsilon^{-d}\int_{\substack{\mathbf{z}\in\mathcal{M}\\ \max(\|\mathbf{z}-\mathbf{y}\|,\|\mathbf{z}-\mathbf{x}\|)<\varepsilon^\gamma}} w_2(\mathbf{x},\mathbf{y},\mathbf{z}) f_{xyz}\mathsf{d}\mathbf{z} + \mathcal{O}(\varepsilon^3)\\
&\overset{(ii)}{=} \varepsilon^{-d}\int_{\mathbf{z}\in\mathbb{R}^d}\Bigg[\left(\kappa_\varepsilon(\mathbf{z},\mathbf{x})\kappa_\varepsilon(\mathbf{z},\mathbf{y}) + \left(\frac{Q_{\mathbf{x},4}(\mathbf{z},\mathbf{y})}{\varepsilon^2}\right)\left(\kappa'_\varepsilon(\mathbf{z},\mathbf{x})\kappa_\varepsilon(\mathbf{z},\mathbf{y}) + \kappa_\varepsilon(\mathbf{z},\mathbf{x})\kappa'_\varepsilon(\mathbf{z},\mathbf{y})\right)\right)\\
&\qquad\qquad \cdot\left(f^{(0)}_{xyz} + \varepsilon f^{(1)}_{xyz} + \varepsilon^2 f^{(2)}_{xyz}\right)\cdot\left((1 + Q_{\mathbf{x},2}(\mathbf{z})\right)\Bigg]\mathsf{d}\mathbf{z} + \mathcal{O}(\varepsilon^3)\\
&= \varepsilon^2(c_2 - c_1(\mathbf{x},\mathbf{y}))\int_0^1 (\Delta_1\zeta)(\mathbf{u}(t))^\top\mathbf{u}'(t)dt + \mathcal{O}(\delta^2) + \mathcal{O}(\varepsilon^3).
\end{aligned}$$

Equality (i) holds by Lemma (4.5), while equality (ii) is true by projecting $\mathbf{x}, \mathbf{y}, \mathbf{z}$ to $\mathcal{T}_{\mathbf{x}}\mathcal{M}$, changing the variable of $\mathbf{x}, \mathbf{y}, \mathbf{z} \leftarrow \mathbf{x}_p, \mathbf{y}_p, \mathbf{z}_p$, and using Lemma (4.5) again. Here $f^{(0)}_{xyz}, f^{(1)}_{xyz}, f^{(2)}_{xyz}$ represent the constant, first, and second order term from the Taylor expansion in local coordinate system at $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ as in (4.9). Note that the bias term consisting of $\frac{\partial^2 f_i}{\partial s_i^2}$ can be removed if $\kappa(\cdot)$ is exponential kernel. From Lemma 4.8, $f^{(0)}_{xyz} = 0$; therefore the constant term consists of $Q_{\mathbf{x},4}(\mathbf{z},\mathbf{y})$ or $Q_{\mathbf{x},2}(\mathbf{z})$ disappeared. The first and second order terms consist of $Q_{p,4}(\mathbf{x},\mathbf{y})$ or $Q_{p,2}(\mathbf{x},\mathbf{y})$ are in the order of $\mathcal{O}(\varepsilon^3)$ therefore can be merged into the last error term. In Lemma 4.8, The differentiation is in the coordinate on local tangent plane $\mathcal{T}_{\mathbf{x}}\mathcal{M}$. One can change the differentiation to partial derivative on normal

coordinate system by (4.12b). Additionally, one can again approximate the line integral $\int_0^1(\Delta_1\zeta)(\mathbf{u}(t))^\top\mathbf{u}'(t)\mathsf{d}t = \int_0^1(\Delta_1\zeta)(\boldsymbol{\gamma}(t))^\top\boldsymbol{\gamma}'(t)\mathsf{d}t + \mathcal{O}(\varepsilon^3)$ by (4.12d) with $\boldsymbol{\gamma}$ be the geodesic connecting $\mathbf{x},\mathbf{y}$, implying

$$\varepsilon^{-d}\int_{\mathcal{M}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})f_{xyz}\mathsf{d}\mu(\mathbf{z}) = \varepsilon^2 c(\mathbf{x},\mathbf{y})\int_0^1[\Delta_1\zeta(\boldsymbol{\gamma}_{\mathcal{M}}(t))]^\top\boldsymbol{\gamma}'_{\mathcal{M}}(t)\mathsf{d}t + \mathcal{O}(\varepsilon^3) + \mathcal{O}(\delta^2).$$

With $c(\mathbf{x},\mathbf{y}) = c_2 - c_1(\mathbf{x},\mathbf{y})$. This completes the proof. ■

4.7.3 Proof of Theorem 4.3

Proof of Theorem 4.3. Using $\boldsymbol{\mathcal{L}}_1^{\mathrm{up}} = \mathbf{W}_1^{-1}\cdot\mathbf{B}_2\mathbf{W}_2\mathbf{B}_2^\top$ and Lemma 4.1, the expected value beomces,

$$\begin{aligned} q(\|\mathbf{x}-\mathbf{y}\|)\mathbb{E}\left[(\boldsymbol{\mathcal{L}}_1^{\mathrm{up}})_{[x,y]}\right] &= \frac{w_1(\|\mathbf{x}-\mathbf{y}\|)}{\varepsilon^2 c(\|\mathbf{x}-\mathbf{y}\|)}\mathbb{E}_{z,v}\left[\frac{\frac{\varepsilon^{-d}}{n}\sum_{z\notin\{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})f_{xyz}}{\frac{\varepsilon^{-d}}{n}\sum_{v\notin\{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{v})}\right] \\ = \mathbb{E}_z\left[\frac{\frac{\varepsilon^{-d}}{n}\sum_{z\notin\{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})f_{xyz}}{\varepsilon^2 c(\|\mathbf{x}-\mathbf{y}\|)}\right]&\cdot\mathbb{E}_v\left[\frac{w_1(\mathbf{x},\mathbf{y})}{\frac{\varepsilon^{-d}}{n}\sum_{v\notin\{x,y\}} w_2(\mathbf{x},\mathbf{y},\mathbf{v})}\right]. \end{aligned} \tag{4.14}$$

Last equality holds by independence of random variables z, v. By *Monte-Carlo* approximation,

$$\begin{aligned} \mathbb{E}\left[\frac{1}{n}\sum_z w_2(\mathbf{x},\mathbf{y},\mathbf{z})f_{xyz}\right] &= \int_{\mathcal{M}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})f_{xyz}\mathsf{d}\mu(\mathbf{z}) \\ &= \varepsilon^2 c(\|\mathbf{x}-\mathbf{y}\|)\int_0^1(\Delta_1\zeta)(\boldsymbol{\gamma}(t))\boldsymbol{\gamma}'(t)\mathsf{d}t + \mathcal{O}(\varepsilon^3). \end{aligned}$$

Last equality holds by Proposition 4.2. It is not difficult to show that $\mathbb{E}\left[\frac{1}{n}\sum_z w_2(\mathbf{x},\mathbf{y},\mathbf{z})\right] = w_1(\mathbf{x},\mathbf{y})$ following the proof of Proposition 4.2, implying that the $\mathbb{E}_v$ term in (4.14) is $1 + \mathcal{O}(n^{-1})$ by standard ratio estimates. Therefore, one has

$$\mathbb{E}\left[q(\|\mathbf{x}-\mathbf{y}\|)(\boldsymbol{\mathcal{L}}_1^{\mathrm{up}})_{[x,y]}\right] = \int_0^1(\Delta_1\zeta)(\boldsymbol{\gamma}(t))\boldsymbol{\gamma}'(t)\mathsf{d}t + \mathcal{O}(\varepsilon) + \mathcal{O}(n^{-1}).$$

This completes the proof. ■

4.8 Appendix—proofs of the spectral consistency of the down *Helmholtzian*

4.8.1 Outline of the proof

The proof for spectral consistency of 1 *down* Laplacian is outlined in Figure **??**. Instead of directly showing the consistency of 1 *down* Laplacian (dashed line), one can use the existing spectral consistency of the Laplace-Beltrami operator Δ_0 to show the spectral consistency of Δ_1. The first step of the proof is to show the *spectral dependency* of *down* Helmholtzian $\boldsymbol{\mathcal{L}}_1^{\text{down}} = \mathbf{B}_1^\top \mathbf{W}_0^{-1} \mathbf{B}_1 \mathbf{W}_1$ to the corresponding graph Laplacian, i.e., $\boldsymbol{\mathcal{L}} = \mathbf{W}_0^{-1} \mathbf{B}_1 \mathbf{W}_1 \mathbf{B}_1^\top$; or more generally, the spectral dependency between $k-1$ *up* and k *down*Laplacians. Proposition 1.2 of [98] first showed the spectra of $\mathbf{L}_1^{\text{down}}$ and $\mathbf{L}_0$ away from 0 agree including multiplicity. [58] later pointed out the aforementioned agreement between non-zero spectra could be extended to k unweighted Laplacian ($\mathbf{L}_k^{\text{down}}$ and $\mathbf{L}_{k-1}^{\text{up}}$). Here we provide an extension of the results to the weighted k Laplacians.

Lemma 4.10 (Spectral dependency of $\boldsymbol{\mathcal{L}}_k$)**.** *Let $\mathcal{S}(\mathbf{A})$ be the non-zero spectrum of matrix $\mathbf{A}$. The non-zero spectra of $\boldsymbol{\mathcal{L}}_k^{\text{down}}$ and $\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}}$ agree including multiplicity, i.e., $\mathcal{S}(\boldsymbol{\mathcal{L}}_k^{\text{down}}) = \mathcal{S}(\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}})$.*

Proof of the lemma can be found in Supplement 4.8.2 as well as corollaries on finding the eigenvector of $\mathbf{L}_{k-1}^{\text{up}}$ from $\mathbf{L}_k^{\text{down}}$ (or vice-versa). This lemma points out that the non-zero spectrum of $\boldsymbol{\mathcal{L}}_k^{\text{down}}$ is identical to the non-zero spectrum of $\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}}$. It indicates that the *down* discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ is consistent if the corresponding *random walk* Laplacian $\boldsymbol{\mathcal{L}} = \mathbf{I}_n - \mathbf{D}^{-1}\mathbf{K} = \mathbf{W}_0^{-1}\mathbf{B}_1\mathbf{W}_1\mathbf{B}_1^\top$ is consistent, with $\mathbf{W}_0$, $\mathbf{W}_1$ constructed from $\mathbf{W}_2$ as discussed in Section 4.1. The next Proposition investigates the consistency of $\boldsymbol{\mathcal{L}}$.

Proposition 4.11 (Consistency of the random walk Laplacian $\boldsymbol{\mathcal{L}}$ with kernel (4.2))**.** *Under Assumption 4.1–4.2, same spectral consistency result as in Theorem 5 of [15] can be obtained for the corresponding $\boldsymbol{\mathcal{L}}_0 = \mathbf{W}_0^{-1}\mathbf{B}_1\mathbf{W}_1\mathbf{B}_1^\top$, with weights calculated by $\mathbf{W}_k = \text{diag}(|\mathbf{B}_{k+1}|\mathbf{W}_{k+1}\mathbf{1}_{n_{k+1}})$ for $k = 0, 1$.*

Sketch of proof. We first investigate the scenario when $w_2(\mathbf{x}, \mathbf{y}, \mathbf{z})$ is constant, i.e., $w_2(\mathbf{x}, \mathbf{y}, \mathbf{z}) = \mathbb{1}(\|\mathbf{x} - \mathbf{y}\| < \varepsilon)\,\mathbb{1}(\|\mathbf{x} - \mathbf{y}\| < \varepsilon)\,\mathbb{1}(\|\mathbf{x} - \mathbf{y}\| < \varepsilon)$. If $\kappa(\cdot)$ has exponential decay, one can show that the corresponding $w_1(\mathbf{x}, \mathbf{y})$ has exponential decay, implying the consistency of [15]. The above analysis can be naturally extended to general kernel $\kappa(\cdot)$ for $\kappa(\cdot)$ is upper bounded by the indicator kernel. ■

The last part of the proof is to show the agreement in the spectra of the continuous operators Δ_0 and Δ_1. It was shown by using *Courant-Fischer-Weyl min-max principle* on Δ_1 that the non-zero spectrum of Δ_1^{down} (also known as the spectrum of the *co-exact/curl* 1-form) is a copy of non-zero eigenvalues of Δ_0 [33, 41]. With the right arrow completed in Figure 4.1, the *down* Helmholtzian $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ is hence shown to converge *spectrally* to the spectrum of $\Delta_1^{\text{down}} = \mathsf{d}_0\delta_1$.

4.8.2 Spectral dependency and related corollaries

Proof of Lemma 4.10. From [105], the spectra of the *random walk* k-Laplacian and of the symmetrized k-Laplacian are identical, implying that one can study the spectrum of $\boldsymbol{\mathcal{L}}_k^s$ (symmetric) instead of $\boldsymbol{\mathcal{L}}_k$. Following the proof of [98], one has $\mathbf{W}_{k-1}^{-1/2}\mathbf{B}_k\mathbf{W}_k^{1/2}\boldsymbol{\mathcal{L}}_k^{s,\text{down}} = \boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}}\mathbf{W}_{k-1}^{-1/2}\mathbf{B}_k\mathbf{W}_k^{1/2}$ and $\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1/2}\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}} = \boldsymbol{\mathcal{L}}_k^{s,\text{down}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1/2}$. This implies that the mapping between the images of $\boldsymbol{\mathcal{L}}_k^{s,\text{down}}$ and $\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}}$ are isomorphisms. In mathematical terms, if $\tilde{\mathbf{B}}_k = \mathbf{W}_{k-1}^{-1/2}\mathbf{B}_k\mathbf{W}_k^{1/2} : \text{im}(\boldsymbol{\mathcal{L}}_k^{s,\text{down}}) \to \text{im}(\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}})$ and $\tilde{\mathbf{B}}_k^* = \mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1/2} : \text{im}(\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}}) \to \text{im}(\boldsymbol{\mathcal{L}}_k^{s,\text{down}})$, we have $\tilde{\mathbf{B}}_k$ and $\tilde{\mathbf{B}}_k^*$ are isomorphisms. Since $\boldsymbol{\mathcal{L}}_k^{s,\text{down}} = \tilde{\mathbf{B}}_k^*\tilde{\mathbf{B}}_k$ and $\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}} = \tilde{\mathbf{B}}_k\tilde{\mathbf{B}}_k^*$, the isomorphisms of two operator implies $\dim(\text{im}(\boldsymbol{\mathcal{L}}_k^{s,\text{down}})) = \dim(\text{im}(\boldsymbol{\mathcal{L}}_{k-1}^{s,\text{up}}))$ and $\mathcal{S}(\boldsymbol{\mathcal{L}}_k^{\text{down}}) = \mathcal{S}(\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}})$. This completes the proof. ■

Below are some corollaries of Lemma 4.10 which are not used in our analysis but are useful in practice. These lemmas connect the eigenvectors of $\mathbf{L}_k$ (or $\boldsymbol{\mathcal{L}}_k$) with the eigenvectors of $\mathbf{L}_{k-1}$ (or $\boldsymbol{\mathcal{L}}_{k-1}$).

Corollary 4.12 (Eigenvectors of $\mathbf{L}_k$ and $\mathbf{L}_{k-1}$)**.**

C.I *Let $\phi_{k-1} \in \mathbb{R}^{n_{k-1}}$ be the eigenvector of $\mathbf{L}_{k-1}$ with eigenvalues λ, then $\mathbf{B}_k^\top \phi_{k-1}$ is the eigenvector of $\mathbf{L}_k^{down}$ with same eigenvalue.*

C.II *Let $\phi_k \in \mathbb{R}^{n_k}$ be the eigenvector of $\mathbf{L}_k$ with eigenvalues λ, then $\mathbf{B}_k \phi_k$ is the eigenvector of $\mathbf{L}_{k-1}^{up}$ with same eigenvalue.*

Proof. For C.I, Let ϕ_{k-1} be the non-trivial eigenfunction of $\mathbf{L}_{k-1}$ with eigenvalue λ, this implies $\mathbf{L}_{k-1}\phi_{k-1} = \lambda\phi_{k-1}$, therefore

$$\lambda \mathbf{B}_k^\top \phi_{k-1} = \mathbf{B}_k^\top \mathbf{L}_{k-1}\phi_{k-1} = \left(\cancel{\mathbf{B}_k^\top \mathbf{B}_{k-1}^\top} \mathbf{B}_{k-1} + \mathbf{B}_k^\top \mathbf{B}_k \mathbf{B}_k^\top\right) \phi_{k-1} = \mathbf{L}_k^{\text{down}} \mathbf{B}_k^\top \phi_{k-1}.$$

Therefore, $\mathbf{B}_k^\top \phi_{k-1}$ will be the eigenfunction of $\mathbf{L}_k^{\text{down}}$. Similarly, for C.II,

$$\lambda \mathbf{B}_k \phi_k = \mathbf{B}_k \mathbf{L}_k \phi_k = \left(\mathbf{B}_k \mathbf{B}_k^\top \mathbf{B}_k + \cancel{\mathbf{B}_k \mathbf{B}_{k+1}^\top} \mathbf{B}_{k+1}^\top\right) \phi_k = \mathbf{L}_{k-1}^{\text{up}} \mathbf{B}_k \phi_k.$$

This completes the proof. ■

Corollary 4.13 (Eigenvectors of $\boldsymbol{\mathcal{L}}_k$ and $\boldsymbol{\mathcal{L}}_{k-1}$)**.**

C.I *Let $\phi_{k-1} \in \mathbb{R}^{n_{k-1}}$ be the eigenvector of $\boldsymbol{\mathcal{L}}_{k-1}$ with eigenvalues λ, then $\mathbf{B}_k^\top \phi_{k-1}$ is the eigenvector of $\boldsymbol{\mathcal{L}}_k^{down}$ with same eigenvalue.*

C.II *Let $\phi_k \in \mathbb{R}^{n_k}$ be the eigenvector of $\boldsymbol{\mathcal{L}}_k$ with eigenvalues λ, then $\mathbf{W}_{k-1}^{-1} \mathbf{B}_k \mathbf{W}_k \phi_k$ is the eigenvector of $\boldsymbol{\mathcal{L}}_{k-1}^{up}$ with same eigenvalue.*

Proof. For C.I, since ϕ_{k-1} is the non-trivial eigenfunction of $\boldsymbol{\mathcal{L}}_{k-1}$ with eigenvalue λ, we have

$$\lambda \mathbf{B}_k^\top \phi_{k-1} = \mathbf{B}_k^\top \boldsymbol{\mathcal{L}}_{k-1}\phi = \left(\cancel{\mathbf{B}_k^\top \mathbf{B}_{k-1}^\top} \mathbf{W}_{k-2}^{-1} \mathbf{B}_{k-1} \mathbf{W}_{k-1} + \mathbf{B}_k^\top \mathbf{W}_{k-1}^{-1} \mathbf{B}_k \mathbf{W}_k \mathbf{B}_k^\top\right) \phi = \boldsymbol{\mathcal{L}}_k^{\text{down}} \mathbf{B}_k^\top \phi_k.$$

For the case in C.II, we need some little algebraic tricks to get rid of the weights $\mathbf{W}$ before boundary operator.

$$\begin{aligned}
\lambda \mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k\phi_k &= \mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k\boldsymbol{\mathcal{L}}_k\phi_k \\
&= \Big(\underbrace{\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k\mathbf{B}_k^\top}_{\boldsymbol{\mathcal{L}}_{k-1}^{\mathrm{up}}}\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k + \mathbf{W}_{k-1}^{-1}\underbrace{\mathbf{B}_k\cancel{\mathbf{W}_k}\cancel{\mathbf{W}_k^{-1}}\mathbf{B}_{k+1}}_{=0}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\Big)\phi_k \\
&= \boldsymbol{\mathcal{L}}_{k-1}^{\mathrm{up}}\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k\phi_k.
\end{aligned}$$

This completes the proof. ■

4.8.3 Proof of Proposition 4.11

We first start with the following lemma that derives the closed form of $w_1(\mathbf{x},\mathbf{y})$ when the weight on the triangles is an indicator function. Note that in the construction below, we ignore the $\kappa(\mathbf{x},\mathbf{y})$ factor, i.e., we assume that $w_2(\mathbf{x},\mathbf{y},\mathbf{z}) = \kappa(\mathbf{x},\mathbf{z})\kappa(\mathbf{y},\mathbf{z})$, for a more concise notation; the $\kappa(\mathbf{x},\mathbf{y})$ factor will be added back later.

Lemma 4.14 (The integral form of constant triangular weight)**.** *Let ε be a bandwidth parameter. Further assume a constant triangular weight, i.e., $w_2(\mathbf{x},\mathbf{y},\mathbf{z}) = \mathbb{1}(\|\mathbf{z}-\mathbf{x}\| < \varepsilon)\mathbb{1}(\|\mathbf{z}-\mathbf{y}\| < \varepsilon)$, then*

$$\begin{aligned}
w_1(\mathbf{x},\mathbf{y}) &= \mathbb{1}(\|\mathbf{x}-\mathbf{y}\| < \delta)\int_{\mathbf{z}\in\mathcal{M}} w_2(\mathbf{x},\mathbf{y},\mathbf{z})\mathsf{d}\mathbf{z} \\
&= \mathbb{1}(\|\mathbf{x}-\mathbf{y}\| < \delta)C\cdot I_{1-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{4(\gamma\varepsilon)^2}}\left(\frac{d+1}{2},\frac{1}{2}\right) + \mathcal{O}(\varepsilon^2).
\end{aligned} \tag{4.15}$$

Where $C = \varepsilon^d\cdot p\cdot C_d$, C_d is the volume of unit d-ball, i.e., $C_d = \pi^{d/2}/\Gamma(d/2+1)$, and $I_{\mathbf{x}}(a,b)$ is the regularized incomplete beta function.

Proof. In the continuous limit, with constant sampling density p, we have

$$\begin{aligned}
&\varepsilon^{-d}\int_{\mathbf{z}\in\mathcal{M}} w_2 \mathsf{d}\mathbf{z} = \varepsilon^{-d}\int_{\mathbf{z}\in\mathbb{R}^d} w_2 \mathsf{d}\mathbf{z} + \mathcal{O}(\varepsilon^2) \\
=\ & p\cdot 2\cdot \mathrm{Vol}_{\mathrm{cap}}\left(\varepsilon - \frac{\|\mathbf{x}-\mathbf{y}\|_2}{2}; \varepsilon, d\right) = p\cdot C_d I_{1-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{4\varepsilon^2}}\left(\frac{d+1}{2}, \frac{1}{2}\right).
\end{aligned}$$

The first equality holds from projecting $\mathbf{x}, \mathbf{y}, \mathbf{z}$ onto $\mathcal{T}_{\mathbf{x}}\mathcal{M}$ and using Lemma 4.9, $\mathcal{O}(\varepsilon^2)$ is from $Q_{\mathbf{x},4}(\mathbf{z},\mathbf{y})$ and $Q_{\mathbf{x},2}(\mathbf{z})$ of (4.12c) and (4.12b), respectively. Last equality holds because

$$\begin{aligned}
\mathrm{Vol}_{\mathrm{cap}}(h; r, d) \ &= \int_0^{\phi} C_{d-1} r^{d-1}\sin^{d-1}\theta r\sin\theta d\theta = C_{d-1} r^d \int_0^t \nu^{\frac{d-1}{2}}(1-\nu)^{-\frac{1}{2}} d\nu \\
&= C_{d-1} r^d B\left(\frac{d+1}{2}, \frac{1}{2}\right) I_{(2rh-h^2)/r^2}\left(\frac{d+1}{2}, \frac{1}{2}\right) \\
&= C_d r^d I_{(2rh-h^2)/r^2}\left(\frac{d+1}{2}, \frac{1}{2}\right).
\end{aligned}$$

And,

$$C_{d-1}\cdot B\left(\frac{d+1}{2}, \frac{1}{2}\right) = \frac{\pi^{(d-1)/2}}{\Gamma\left(\frac{d-1}{2}+1\right)}\frac{\Gamma\left(\frac{d+1}{2}\right)\Gamma(1/2)}{\Gamma\left(\frac{d}{2}+1\right)} = C_d.$$

This completes the proof. ■

Proof of Proposition 4.11. It suffices to prove that the corresponding $w_1(\mathbf{x},\mathbf{y})$ has exponential decay, and the $\mathcal{O}(\varepsilon^2)$ error term can be ignored in the asymptotic expansion of graph Laplacian operator. Let $w_1^{\mathbb{1}}(\|x-y\|; r)$ be (4.15). The integral operator of general kernel κ can be decomposed into two parts, i.e.,

$$\begin{aligned}
w_1(\mathbf{x},\mathbf{y}) = \int_{\mathcal{M}} \kappa(\mathbf{x},\mathbf{z})\kappa(\mathbf{y},\mathbf{z})\mathsf{d}\mathbf{z} \ =& \int\limits_{\substack{z\in\mathcal{M} \\ \max(\|\mathbf{z}-\mathbf{y}\|, \|\mathbf{z}-\mathbf{x}\|)\le\varepsilon^{\gamma}}} \kappa(\mathbf{x},\mathbf{z})\kappa(\mathbf{y},\mathbf{z})\mathsf{d}\mathbf{z} \\
&+ \int\limits_{\substack{z\in\mathcal{M} \\ \min(\|\mathbf{z}-\mathbf{y}\|, \|\mathbf{z}-\mathbf{x}\|)>\varepsilon^{\gamma}}} \kappa(\mathbf{x},\mathbf{z})\kappa(\mathbf{y},\mathbf{z})\mathsf{d}\mathbf{z}.
\end{aligned}$$

With $\min(\|\mathbf{z}-\mathbf{x}\|, \|\mathbf{z}-\mathbf{y}\|) < \varepsilon^{\gamma}$, one has $\kappa(\mathbf{x},\mathbf{y}) \le \mathbb{1}(\|\mathbf{x}-\mathbf{y}\| < \varepsilon^{\gamma})$. Therefore,

the first term can be bounded by $w_1^{\mathbb{1}}(\mathbf{x}, \mathbf{y}; \varepsilon^\gamma)$. The second term can be bounded by $\mathcal{O}(\varepsilon^3)$ using Lemma 4.5 with $g(\cdot) = 1$. Note that the result can be generalized to manifold with boundaries by the following. For the points that is ε^γ within the boundary $\partial\mathcal{M}$, the above inequality is still valid, for one can use a modified kernel $\kappa'(\mathbf{x}, \mathbf{y})$ with $\kappa'(\mathbf{x}, \mathbf{y}) = \kappa(\mathbf{x}, \mathbf{y})$ if $\mathbf{y} \in \mathcal{M}$ and 0 otherwise. Putting in $\kappa(\mathbf{x}, \mathbf{y})$, one has,

$$w_1(\mathbf{x}, \mathbf{y}) \leq \kappa(\mathbf{x}, \mathbf{y}) w_1^{\mathbb{1}}(\mathbf{x}, \mathbf{y}; \varepsilon^\gamma) + \kappa(\mathbf{x}, \mathbf{y}) \cdot \mathcal{O}(\varepsilon^2) \leq C\kappa(\mathbf{x}, \mathbf{y}) + \kappa(\mathbf{x}, \mathbf{y}) \cdot \mathcal{O}(\varepsilon^2).$$

Last inequality holds since $w_1^{\mathbb{1}}(\mathbf{x}, \mathbf{y}; \varepsilon^\gamma) \leq C$ from Lemma 4.14. The above inequality shows that $w_1(\mathbf{x}, \mathbf{y})$ can be decomposed into a term that has fast enough decay and another term which is bounded by $\mathcal{O}(\varepsilon^2)$. Note that the graph is built with radius δ, the second order expansion of the graph Laplacian integral operator [15] has a δ^2 term, implying that $\kappa(\mathbf{x}, \mathbf{y}) \cdot \mathcal{O}(\varepsilon^2)$ term can be bounded by $\mathcal{O}(\varepsilon^2\delta^2) = \mathcal{O}(\delta^{10/3})$. Hence, spectral consistency as in Theorem 5 of [15] with bias & variance determined by $\mathcal{O}(\delta^{4/3}) = \mathcal{O}(\varepsilon^2)$ can be achieved. More specifically, since points are sampled with constant density from the manifold $\mathcal{M}$, the spectrum of $\boldsymbol{\mathcal{L}} = \mathbf{W}_0^{-1}\mathbf{B}_1\mathbf{W}_1\mathbf{B}_1^\top$ with weight $\mathbf{w}_1 = |\mathbf{B}_2|\mathbf{w}_2$ and $\mathbf{w}_0 = |\mathbf{B}_1|\mathbf{w}_1$ converges to the spectrum of Laplace-Beltrami operator Δ_0 with bias & variance in the order of $\mathcal{O}(\varepsilon^2)$.

■

Chapter 5

VECTOR FIELD LEARNING USING THE HELMHOLTZIAN ESTIMATOR

Many natural phenomena are described by vector fields such as velocity, force, or displacement fields. They capture physical systems across many scales: from the microscopic scale, such as the properties of atoms or molecules [29] and cell development [69], to global scales including the atmosphere [23] and the oceans [49] flow, and eventually to astronomical scales [60]. Furthermore, social scientists and economists also find vector fields beneficial in describing population dynamics in physical space [62, 99] or within the job market [96]. The increased resolution of instruments produces ever more detailed measurements from the aforementioned complex systems, which can be abstractly thought of as high dimensional point clouds, equipped with (sparsely sampled) vector fields. Fortunately, these phenomena usually can be well described by a small number of variables that represent the underlying laws of the system; hence, being able to discover and analyze these variables from the observed vector fields is of crucial importance.

A vector field is formally defined on the Euclidean space $\mathbb{R}^D$ or on non-linear subspaces (e.g., spheres or tori) of $\mathbb{R}^D$ called *manifolds*. Hodge theory and the time-honored *Helmholtz-Hodge decomposition* (HHD) [16, 73], given in (2.4), establish a mathematical framework to analyze vector fields; they define concepts such as sources and sinks (where fluxes are generated or disappear), cyclicity (*harmonics*), and turbulence (categorized by *curl*) in the systems.

Under the framework of the Hodge theory and HHD, if the laws of nature are known, i.e., if one can write out the mathematical descriptor Δ_1 explicitly, then the HHD can reveal these high-level features in an analytic (or numeric) form. More generally, Δ_1 will provide a basis to decompose any vector field for the system of interest. However, for many natural or social phenomena/systems, the mathematical laws are unknown; or it is challenging to write them out explicitly in closed-form solutions. Even if the aforementioned tasks are possible, the modeling task will require specific domain knowledge, making it difficult to generalize to other realms of study. Having the capability to estimate the underlying structure purely from data without explicit functional forms, called *non-parametric* (machine learning) modeling, is crucial to understand these large complex systems. In the past two decades, the success

of Isomap [118] in estimating intrinsic coordinates (functions) motivated numerous similar algorithms (DM [32], LTSA [131], tSNE [124], etc.) known collectively as *manifold learning* (ML) algorithms. ML algorithms achieved remarkable successes in numerous domains, including RNA single-cell sequencing, astronomy, and molecular dynamic [26].

In this chapter, we propose *Helmholtzian Eigenmap* (HE), which expands the paradigm inaugurated by Isomap to the non-parametric learning of vector fields. Specifically, we will obtain a basis of vector fields from the observations, which will allow us to learn from the (partially) observed vector fields. Our work builds upon the discrete 1-Laplacian matrix $\boldsymbol{\mathcal{L}}_1$ discussed in Chapter 4; the versatility of the HE framework is supported by various applications in vector field smoothing, edge flow SSL, and estimating underlying velocity field from the partially observed trajectories on datasets with non-trivial manifold structure from chemistry, oceanography, and cell biology.

5.1 Eigenflows of the discrete Helmholtzian

The eigenflows of the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$ provide an orthogonal basis for the vector fields on $\mathcal{M}$. While the *harmonic* flows characterize the non-trivial topology of the manifold, the *gradient* and *curl* flows describe the sources, the sinks, and the (local) turbulences in the system; these two flow spaces will have essential roles in the edge flow learning we will discuss below.

We exemplify the estimated eigenflows on the *Global Lagrangian Drifter Data*[1], which were collected by NOAA's Atlantic Oceanographic and Meteorological Laboratory. The dataset was used in [49] to analyze the Lagrangian coherent structures in the ocean current, showing that certain flow structures stay coherent over time. Each point in the dataset is a buoy at a specific time, with buoy ID, location (in latitude and longitude), date/time, velocity, and water temperature available to the practitioner. We extract the buoys that were in the North Pacific ocean dated between 2010–2019. The original sample size is around 3 million.

[1]This dataset can be downloaded from `http://www.aoml.noaa.gov/envids/gld/`

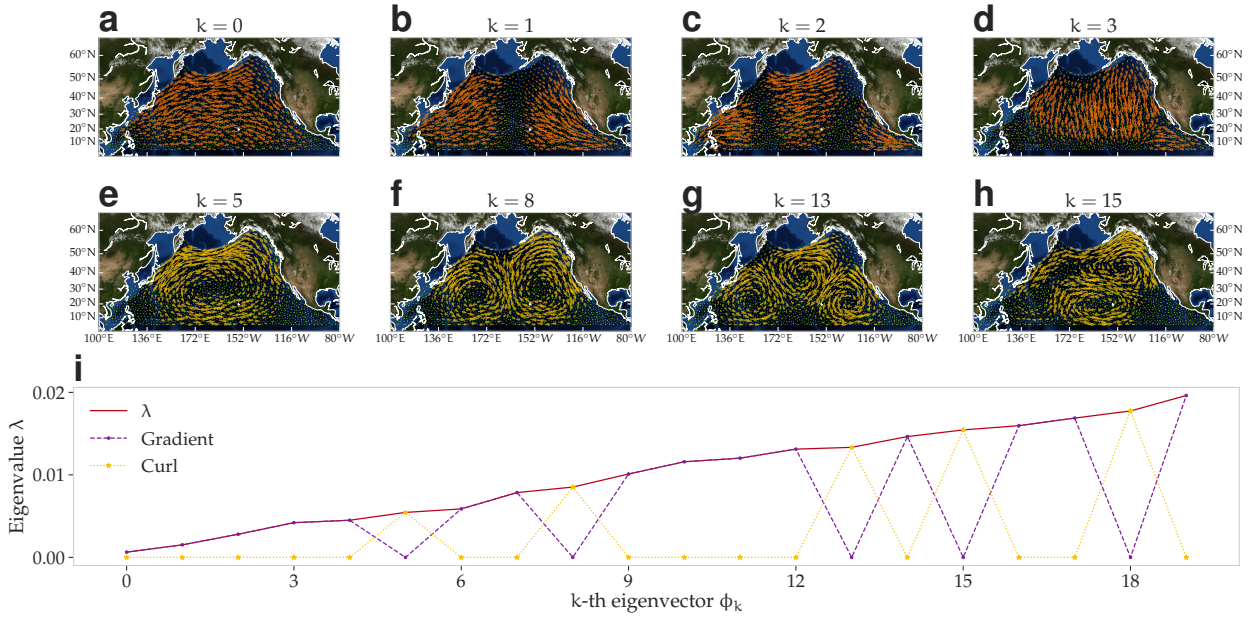

Figure 5.1: Vector field basis estimated from the eigenflows of $\boldsymbol{\mathcal{L}}_1^s$ on the North Pacific Ocean buoys dataset. (a–d) The first five *gradient* eigenfields. (e–h) The first five *curl* eigenfields. (i) Categorizing the first twenty eigenflows of $\boldsymbol{\mathcal{L}}_1^s$ by HHD. Any eigenvalue λ (red) corresponding to an eigenflow $\boldsymbol{\phi}$ can either be equal to $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{down}} \boldsymbol{\phi}$ or $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{up}} \boldsymbol{\phi}$. Eigenflows with non-zeros $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{down}} \boldsymbol{\phi}$ are *gradient* (purple); while $\boldsymbol{\phi}$ with non-zero $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{up}} \boldsymbol{\phi}$ represent *curl* flows (yellow).

We sample $1,500$ furthest buoys that meet the above criteria and convert the coordinate from latitude/longitude to the *earth-centered, earth fixed* (ECEF) coordinate system; we call this dataset `PACIFIC`.

Figure 5.1 illustrates the first five *eigenflows* of the *gradient* (Figures 5.1a–5.1d) and *curl* (Figures 5.1e–5.1h) vector fields on `PACIFIC`. Being eigenfunctions of $\boldsymbol{\mathcal{L}}_1$, eigenflows are uniquely constrained by the shape of the manifold, given by the finite point cloud $\mathbf{X}$. Hence, they do o not depend on any pre-existing vector fields. With `PACIFIC`, the point cloud $\mathbf{X}$ corresponds to buoy locations at time. Both gradient and curl eigenflows become less smmoth become less smooth when the corresponding eigenvalue increases (from left to right in Figure 5.1); we call the flows with lower eigenvalues the *low frequency* flows. In Figure fig:ocean-vf-basis, it is apparent that the first three *gradient* and first three *curl* flows

parametrize the longitude while the fourth ones parameterize the latitude. This is an effect of the manifold *aspect ratio* – larger deployment in longitude than in latitude which has been studied in the case of the Diffusion Maps Laplacian $\boldsymbol{\mathcal{L}}_0$ [32] by [26]. To determine whether an eigenflow $\boldsymbol{\phi}$ is *gradient* or *curl*, one simply calculates the magnitude of $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{up}} \boldsymbol{\phi}$ (*curl*) or $\boldsymbol{\phi}^\top \boldsymbol{\mathcal{L}}_1^{\text{down}} \boldsymbol{\phi}$ (*gradient*). From HHD, the images of $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ are mutually orthogonal; therefore, an eigenvalue will only be linked to either a *curl* or a *gradient* eigenflow (Figure 5.1i).

5.2 Smoothing vector field measurements

Measurements of vector fields are usually noisy. Therefore, having theoretically justified smoothing algorithms is crucial for scientists to interpret the noisy measured vector fields correctly. As discussed previously, lower frequency eigenflows aggregate information over larger scales; hence, they are less sensitive to measurement noise. By projecting a noisy flow onto the low-frequency eigenflows, one can obtain a *smoothed* version of any partially observed vector field. This allows us to remove the high-frequency measurement noise and to visualize the large-scale structure in the flow.

Here, we present a method that balances smoothness and representation accuracy; this method is inspired by the long line of prior work on node-based signal smoothing with the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ [90]. Previous work [104] proposed an edge flow smoothing algorithm using the *unnormalized down Laplacian* $\mathbf{L}_1^{\text{down}} = \mathbf{B}_1^\top \mathbf{B}_1$; this has no smoothing effect on the curl component of a flow since any curl flow lies in the null space of $\mathbf{L}_1^{\text{down}}$. We propose to smooth a given flow **omega** by $\hat{\boldsymbol{\omega}}$, the solution of the regularized least-squares problem

$$\hat{\boldsymbol{\omega}} = \underset{\mathbf{v}}{\operatorname{argmin}} \|\mathbf{v} - \boldsymbol{\omega}\|^2 + \alpha \mathbf{v}^\top \boldsymbol{\mathcal{L}}_1^s \mathbf{v}. \tag{5.1}$$

The optimal solution to the above minimization problem is $\hat{\boldsymbol{\omega}} = (\mathbf{I}_{n_1} + \alpha \boldsymbol{\mathcal{L}}_1^s)^{-1} \boldsymbol{\omega}$. In (5.1), $\boldsymbol{\mathcal{L}}_1^s$ is the symmetrized weighted Laplacian that encodes the information from both the *up* and *down* Laplacians and $\alpha > 0$ represents a *regularization parameter*. This formulation can

successfully smooth out the high-frequency noise in both the curl and the gradient spaces. A larger α puts more weight on the $\boldsymbol{\mathcal{L}}_1^s$ term than the mean-squared error (MSE) in (5.1); this essentially penalizes the high frequency components of $\boldsymbol{\omega}$ by projecting the flow into the subspaces spanned by low-frequency eigenflows.

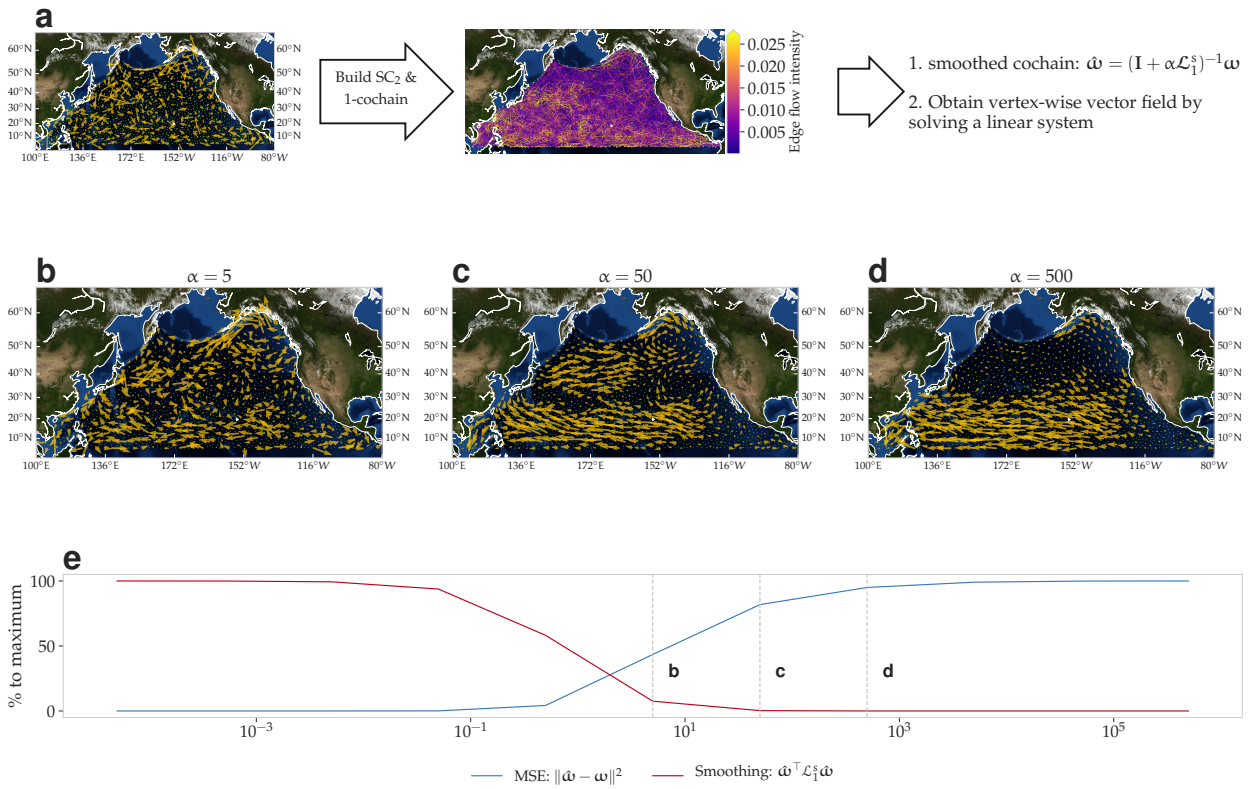

Figure 5.2: Vector field smoothing on the North Pacific Ocean dataset. (a) An illustration of the vector field smoothing algorithm; specifically, the (noisy) vertex-wise vector field is converted to an edge flow on SC_2 with a linear map approximation (middle pane). We remove the noisy high frequency terms of the original edge flow by $\boldsymbol{\mathcal{L}}_1^s$ and an appropriate α; the smoothed vector field on each point is estimated from the smoothed flow with a regularized least squares (details in Section 2.5). (b–d) The smoothed vector fields for different values of regularization parameter α. (e) The tradeoff between mean-squared error $\|\hat{\boldsymbol{\omega}} - \boldsymbol{\omega}\|^2$ and smoothing term $\hat{\boldsymbol{\omega}}^\top \boldsymbol{\mathcal{L}}_1^s \hat{\boldsymbol{\omega}}$ vs. α.

We exemplify this method with `PACIFIC`. The subsampled $n = 1,500$ buoys overlaid with the (finite difference) velocity field are in the left pane of Figure 5.2a. The observed edge flow

(the middle pane of Figure 5.2a) is constructed using the linear integration approximation (in Appendix) of the original vertex-wise velocity field with each edge e colored by the intensity of the flow $|\boldsymbol{\omega}_e|$. The smoothed signal $\hat{\boldsymbol{\omega}}$ represents edge flow in the edge space $\mathbb{R}^{n_1}$; we convert the smoothed edge flow back to the vertex-wise vector field by solving a linear system involving $\mathbf{B}_1$ (in Appendix), in the same way as in Figures 4.2 and 4.3. Figures 5.2b–5.2d show the smoothed vector fields with different smoothing parameters α. Several patterns, e.g., North Equatorial and Kuroshio currents, are visible in the original velocity field. However, a well-known North Pacific current at 40°N is not as apparent as the above two currents. By contrast, the smoothed flow with $\alpha = 50$ makes the North Pacific Gyre visible. The model with $\alpha = 5$ in Figure 5.2b corresponds to "under-smoothing", and that with $\alpha = 500$ represents the case of "over-smoothing".

5.3 Completing sparsely sampled flows by a discrete Helmholtzian regularizer

As discussed earlier, the large-scale structure of a vector field is constrained by the shape of the underlying manifold. Hence, flow values at a relatively small number of points on the manifold can inform on the flow everywhere else on $\mathcal{M}$ given that the flow is smooth. The influence is in proportion to the components of the vector field contained in the low-frequency spectrum of $\boldsymbol{\mathcal{L}}_1$, in the same way that a few Fourier coefficients can successfully reconstruct a smooth function. Following machine learning terminology in Semi-Supervised Learning (SSL), we call the completion of a vector field at the desired set of points from observations *semi-supervised edge flow completion*. Specifically, we propose a method to estimate the flow on all edges of a graph, whose nodes are points on a manifold, from observations of the flow along with a subset of the graph edges. The proposed SSL framework is inspired by *Laplacian Regularized Least Squares* (LaplacianRLS) [10] on scalar functions on nodes. Define the kernel between two edges to be $\mathcal{K}(e_i, e_j) = 1$ if e_i, e_j share the same triangle or node and 0 otherwise. Let $\mathfrak{H}$ be the corresponding reproducing kernel Hilbert space (RKHS). Here we denote the observed set $S \subseteq [n_1]$ as the training set. The relative size of the training set w.r.t. n_1 is the *training ratio*. Given the training set S, we optimize over edge flows

$g \in \mathfrak{H}$ the loss function

$$\frac{1}{|S|}\sum_{e\in S}(g(e)-\omega_e)^2+\lambda_1\|g\|_{\mathfrak{H}}^2+\frac{\lambda_2}{n_1^2}g^\top\boldsymbol{\mathcal{L}}_1^s g. \tag{5.2}$$

From the *representer theorem*, the optimal solution is $g^*(e)=\sum_{e'\in E}\alpha_{e'}^*\mathcal{K}(e',e)$, with $\boldsymbol{\alpha}^*=\left(\operatorname{diag}(\mathbf{1}_S)\boldsymbol{\mathcal{K}}+\lambda_1|S|\mathbf{I}_{n_1}+\frac{\lambda_2|S|}{n_1^2}\boldsymbol{\mathcal{L}}_k^s\boldsymbol{\mathcal{K}}\right)^{-1}\boldsymbol{\omega}$, where $\boldsymbol{\mathcal{K}}$ is a $n_1\times n_1$ kernel matrix with $[\boldsymbol{\mathcal{K}}]_{e_i,e_j}=\mathcal{K}(e_i,e_j)$. One can further extend the proposed $\boldsymbol{\mathcal{L}}_1$-RLS by weighting the importance between $\boldsymbol{\mathcal{L}}_1^{\text{up}}$ and $\boldsymbol{\mathcal{L}}_1^{\text{down}}$ differently, i.e., changing the second regularization term to $\frac{\lambda_2^{\text{up}}}{n_1^2}g^\top\boldsymbol{\mathcal{L}}_1^{s,\text{up}}g+\frac{\lambda_2^{\text{down}}}{n_1^2}g^\top\boldsymbol{\mathcal{L}}_1^{s,\text{down}}g$. We call the proposed variant UpDownLaplacianRLS; it will become LaplacianRLS if $\lambda_2^{\text{down}}=\lambda_2^{\text{up}}$. Note that it is possible to extend other variants of node-based SSL algorithms (e.g., the label propagation algorithm [133] for classification) to edge-based SSL; or introduce more powerful kernels $\mathcal{K}$ that capture the similarities of edges. Here we simply use manifold regularization regression with a simple binary $\mathcal{K}$ to illustrate the effectiveness of the proposed Helmholtzian.

5.3.1 SSL experiment on edge flows

In the following, we study the performance of SSL completion algorithm on one synthetic and three real vector fields. An illustration of the procedure can be found in Figure 5.3 We include the results of the following five models: LaplacianRLS (in purple), UpDownLaplacianRLS (in yellow), ridge regression on the first $m=100$ eigenvectors estimated by $\boldsymbol{\mathcal{L}}_1^s$ (in red), ridge regression by SEC (in green), and lastly the LaplacianRLS using $\mathbf{L}_1$ (with constant triangle weights, in blue). The training set S is sampled uniformly from all n_1 edges, and the training ratio ranges from 0.05 to 0.95. For each training ratio, we repeat the experiments fifty times and report the coefficient of determination (R^2, Figures 5.3f–5.3i) as well as the accuracy of sign prediction (Figures 5.3j–5.3m). The shaded areas represent the 5th to 95th inter-percentile range of the performance metrics for different train/test splits; the solid lines are the median of the corresponding metrics. The hyperparameters (all in the range $[10^{-5},10^5]$) of all models on `PLANE` (a 2D plane) and `PACIFIC` are selected by a 5-fold cross-validation

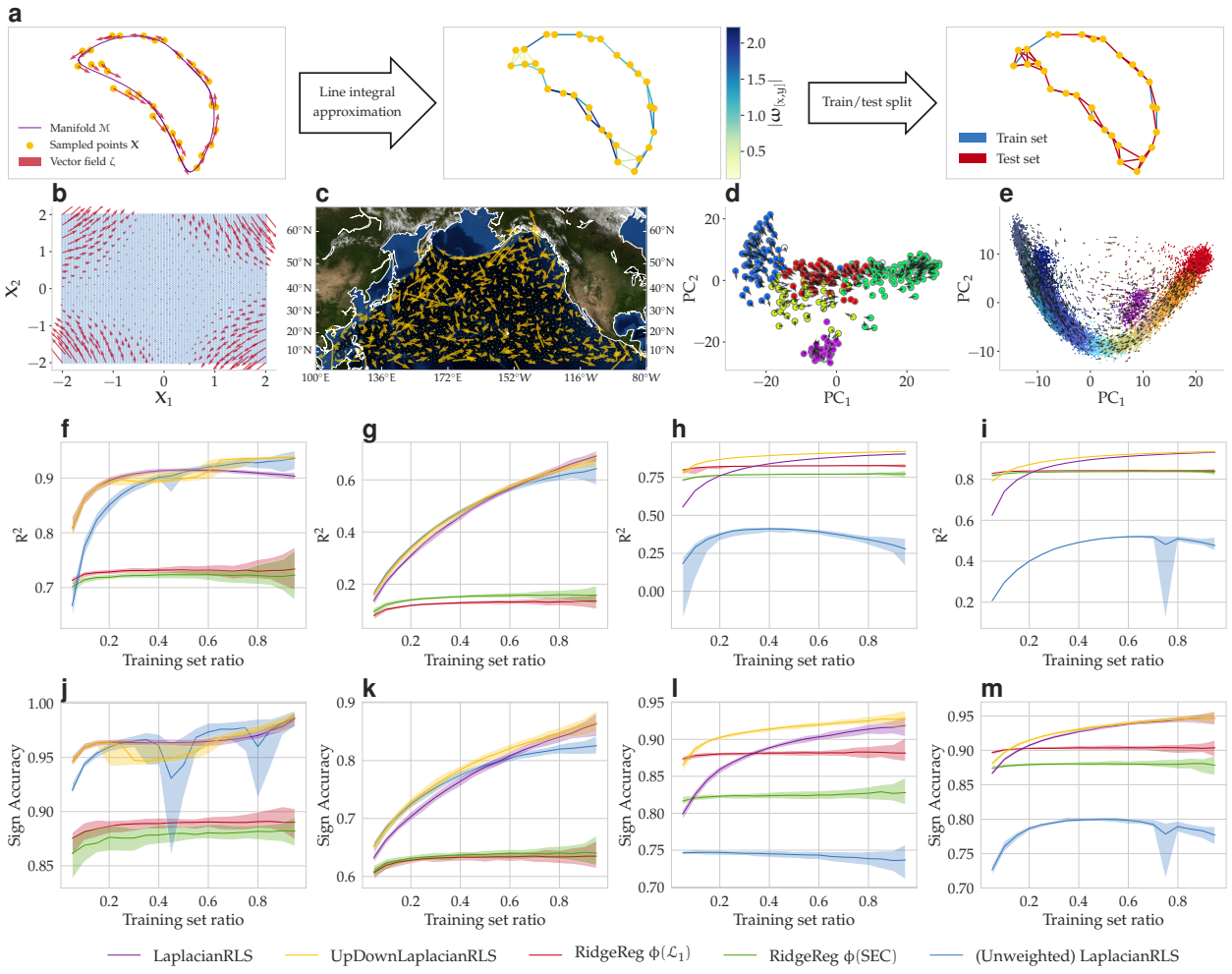

Figure 5.3: Edge flow SSL results for synthetic and real vector fields. (a) An illustration of the SSL cochain construction from the vertex-wise velocity field and train/test split. (b–e) Vector fields of the 2D synthetic strip, the North Pacific Ocean, the chromaffin cell differentiation, and mouse hippocampus cell differentiation datasets, respectively. (f–g) The R^2 scores of different edge flow SSL algorithms applied on the datasets in (B–E), respectively. The bands represent the 5-th to 95-th inter-percentile range of the 50 repeated runs. (j–m) The predicted sign accuracy of different edge flow SSL algorithms.

(CV) for each training ratio split. To reduce the computation time, the hyperparameters of the three Laplacian-based models (purple, yellow, and blue) on the larger datasets (RNA velocity datasets) are chosen by a 5-fold CV when the training set ratio is 0.2 and are used

for all the other train sample sizes.

We illustrate the vector field completion algorithms on `PLANE`, a simple synthetic field, composed of 70% *curl* flow and 30% *gradient* flow (Figure 5.3b). The accuracies of different algorithms are in Figure 5.3f. The gaps between yellow (or purple) and red (or green) curves indicate that the lowest hundred eigenflows are not enough to even predict the simple field in Figure 5.3b. It is reasonable to assume that these gaps will be smaller if we increase m, the number of eigenflows input to the ridge regression model; however, larger m might be challenging for SEC due to its computational overhead. Specifically, SEC involves several fourth-order *dense* tensor computations with size m, indicating that the dependencies in memory and runtime are $\mathcal{O}(m^4)$ and $\mathcal{O}(m^6)$, respectively. In addition to the R^2 score, we provide the sign accuracy, i.e. the accuracy of predicting the correct flow direction along each edge, (Figure 5.3j) as another performance metric. Note that R^2 considers both the direction and the intensity, but the small intensity flows have more influence on the sign accuracy than on the R^2 score. This exemplified by `PLANE` vector field. The vector field in the middle part of the 2D strip manifold is close to zero; a non-robust method like the unweighted LaplacianRLS (blue) results in unstable performances in predicting the orientation of the flows (see the large uncertainty band of the blue curve in Figure 5.3j).

We then apply the method to several real datasets. Figure 5.3g and 5.3k report the R^2 scores and sign accuracy of the edge flow prediction on `PACIFIC`, respectively. Since the velocity field is non-vanishing in the North Pacific Ocean, these two plots show qualitatively similar results. As clearly shown in the plot (red/green curves v.s. the others), higher frequency eigenvectors are needed to complete the edge flow accurately. However, a high precision flow interpolation might be challenging with SEC because large m is usually needed.

Next, we investigate the edge flow SSL on the RNA single-cell datasets equipped with estimated RNA velocities [69] in the third and the fourth column of Figure 5.3. These data have non-trivial manifold structures, making the SSL task more challenging. We apply our framework to two datasets: `CHROMAFFIN` (the Chromaffin cell differentiation dataset with $n = 384$ and $D = 5$) and `HIPPOCAMPUS` (the mouse hippocampus dataset having $n = 18,140$

cells and $D = 10$). We subsample the farthest $n = 800$ cells in `HIPPOCAMPUS` while using all the cells in `CHROMAFFIN`. The RNA velocity fields of `CHROMAFFIN` and `HIPPOCAMPUS` in the first two principal components are in Figures 5.3d and 5.3e, respectively. As expected, the LaplacianRLS (purple) and UpDownLaplacianRLS (yellow) algorithms outperform the SSL algorithms that rely only on low-frequency eigenflows of the estimated Δ_1 (red and green). Note also that compared with the SSL in the simple manifolds in the first two columns, the UpDownLaplacianRLS for the RNA velocity data has better gains compared to the purple curve when the training set sizes are small. The performances of the unweighted LaplacianRLS (blue) for simple manifolds (`PLANE` and `PACIFIC`) seem to align well with the weighted versions (purple and yellow). However, for RNA velocity data, the performances of $\boldsymbol{\mathcal{L}}_1^s$ using the weights by (4.2) (purple) is categorically superior to the naive but intuitive choice of constant triangle weights (blue). As we pointed out earlier, the geometric information is encoded in the triangle weights w_T; discarding this information will not jeopardize the model performance occasionally, in simple cases. Nonetheless, it not the case for the complicated manifolds, as shown in Figures 5.3h, 5.3l, 5.3i, and 5.3m.

The superior performace of the proposed SSL algorithms compared to previous work that only uses the information of unnormalized *down* Laplacian $\mathbf{B}_1^\top \mathbf{B}_1$ [62], shows that the former are adaptable to real flow data beyond those that arise only in *divergence-free* flows (i.e., *curl* and *harmonic* flows). For completeness, in Appendixwe also provide experimental comparisons *divergence-free* flows, created by removing *gradient* components of the real flow.

5.3.2 SSL experiments on the divergence-free flows

The $\mathbf{B}_1$-SSL algorithm proposed by Jia et al. [62] works on (approximately) divergence-free flow. However, the assumption is not always valid for the flows observed in the many real datasets are often times a mixture of *gradient*, *curl*, and *harmonic* flows. Applying the $\mathbf{B}_1$-SSL algorithm by Jia et al. [62] do not results in a good result, as shown in Figures 5.4a–5.4d. Note that these figures are identical to Figures 5.3f–5.3i, but with the SSL results from Jia et al. [62] (blue curves) added. Except for the synthetic flow, the performances of $\mathbf{B}_1$-SSL

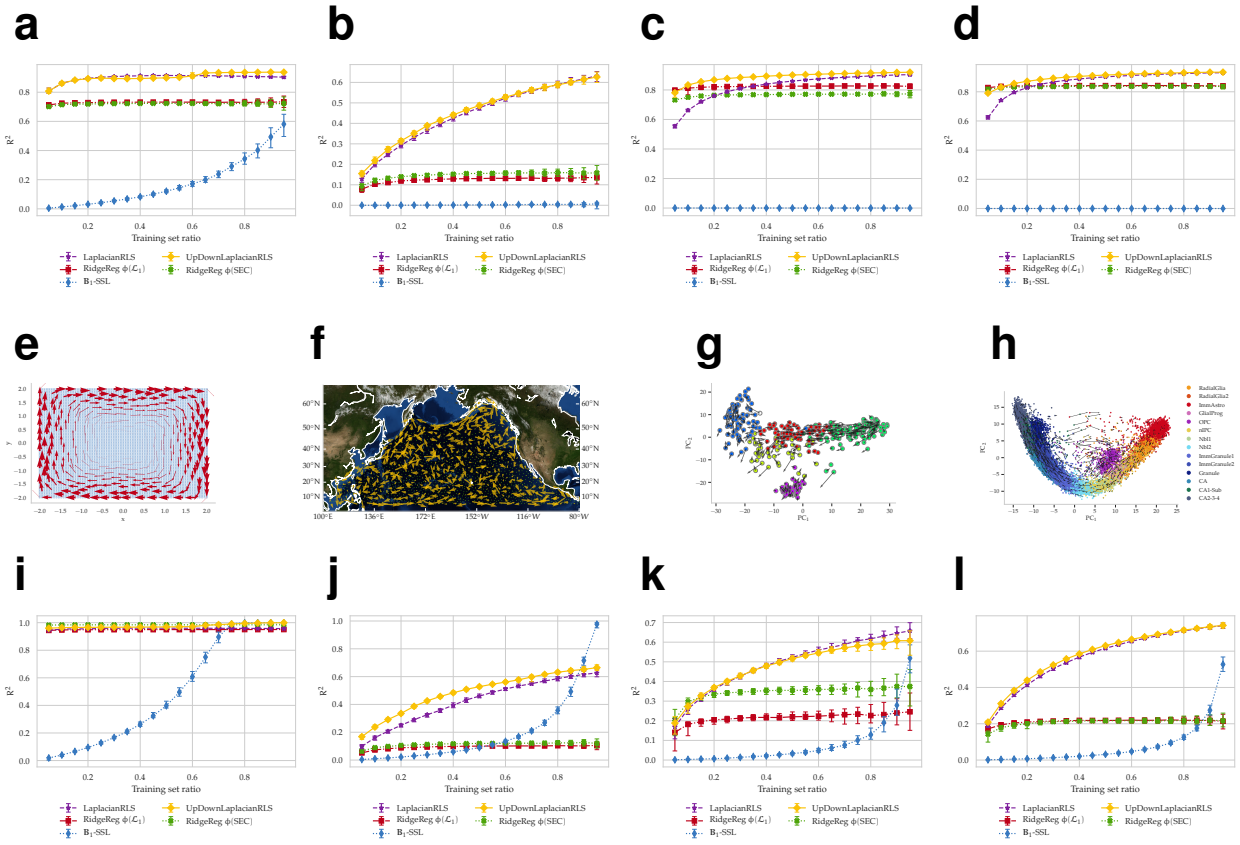

Figure 5.4: SSL results on various datasets with the $\mathbf{B}_1$-SSL proposed by Jia et al. [62]. Columns from left to right correspond to the results of 2D strip, ocean buoy, Chromaffin cell differentiation, and mouse hippocampus cell differentiation dataset. The top row shows the SSL results on the original velocity field with the result of $\mathbf{B}_1$-SSL (blue curve) added. The second row represents the *curl* component of the flows in Figure 5.3 using HHD. The third row are the SSL results of the data with the *curl* flow shown in the second row.

are as bad as random guess.

To further evaluate $\mathbf{B}_1$-SSL, in comparison with our $\boldsymbol{\mathcal{L}}_1$ based SSL algorithms, we artificially create data that satisfies the $\mathbf{B}_1$-SSL assumptions. Namely, we extract the *curl* component from the computed 1-cochain using HHD. In mathematical terms, we first solve the following linear system to get the vector potential $\hat{\mathbf{v}} = \operatorname{argmin}_{\mathbf{v}\in\mathbb{R}^{n_2}} \|\mathbf{B}_2\mathbf{v} - \boldsymbol{\omega}\|_2$. The *curl* component is obtained by projecting 1-cochain $\boldsymbol{\omega}$ onto the image of $\mathbf{B}_2$, i.e., $\boldsymbol{\omega}_{\text{curl}} = \mathbf{B}_2\hat{\mathbf{v}}$.

The estimated velocity field from the *curl* cochains for each datasets can be found in Figures 5.4e–5.4h. As shown in Figure 5.4i–5.4l, the proposed algorithms based on both the *up* and *down* Laplacian out-perform [62] for small train/test ratio. In fact, the performance of $\mathbf{B}_1$-SSL is always weak until the proportion of labeled examples exceeds about 0.7. For manifolds with simple structure, i.e., 2D plane and ocean dataset, [62] can achieve almost perfect predictions ($R^2 \approx 1$) when train-test ratio ≥ 0.9. However, this is not the case for manifolds with complex structures, e.g., RNA velocity datasets.

5.4 Estimating the underlying velocity field from the trajectories using SSL

One application of the Edge flow SSL is to estimate the underlying velocity field given observed "trajectories" that visit a subset of the points. A trajectory can be thought of as a partially observed edge flow with the value of $e = (i, j)$ constructed by counting the number of times the trajectory goes from node i to node j (counted as negative if passing from j to i). In comparison with the previous application, where the inputs were observations of vector field values and the outputs were edge flows on the SC_2 constructed from the samples $\mathbf{X}$, in this application the inputs are (noisy) observed edge flows, i.e. trajectories, and the output is the vector field at all nodes $\mathbf{X}$. We use the UpDownLaplacianRLS algorithm and show the estimated fields from the interpolated edge flow in Figure 5.5. The parameters of the SSL algorithm are chosen using a 5-fold CV when the training ratio is 0.4. For comparison, we also present the velocity fields estimated by the zero-padded edge flow, i.e., the edge flow with all unobserved entries set to zero. For `PACIFIC`, where the original data are trajectories, we sampled 20 trajectories as shown in Figure 5.5a. We obtain a highly interpretable velocity field corresponding to the North Pacific Gyre using SSL (Figure 5.5c), compared with the estimated velocity field from the zero-padded edge flow (Figure 5.5b). The algorithm is surprisingly powerful in the sense that the sampled trajectories do not even cover the west-traveling buoys at 40°N nor the south-bounding drifters near the west coast of the U.S.

We apply a similar analysis to `GLUTAMATERGIC`, the human glutamatergic neuron cell

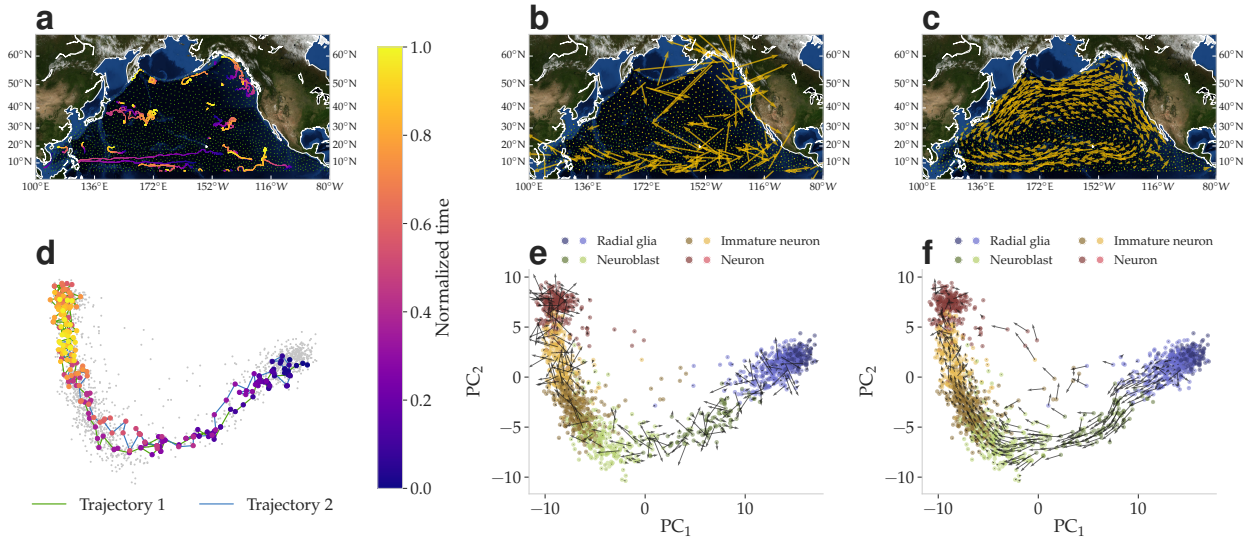

Figure 5.5: Estimating underlying velocity field from partially observed trajectories. The first (a–c) and second (d–f) rows are the North Pacific Ocean and the human glutamatergic neuron cell differentiation datasets, respectively. Columns from left to right present the observed trajectories, the estimated velocity field from a zero-padded edge flow, and the estimated field from the SSL interpolated edge flow.

differentiation data (also from [69]). These data do not contain temporal information; to illustrate our method, we estimate the transition probability matrix of a Markov chain from the RNA velocity field computed from a 550-nearest neighbor (NN) graph. We then sampled two random walk trajectories (Figure 5.5d) from the constructed Markov chain on the 550-NN graph. Note that the k-NN graph above, which is used to estimate the RNA velocity field, can be different from the 1-skeleton of the simplicial complex $\mathrm{SC}_2 = (V, E, T)$ used to construct $\boldsymbol{\mathcal{L}}_1$. The estimated velocity field using the UpDownLaplacianRLS in Figure 5.5c shows a smooth vector field on the manifold compared with that estimated from the zero-padded edge flow (Figure 5.5b).

5.5 Summary

In this chapter, we propose the Helmholtzian Eigenmap (HE) to replicate the functionality of Hodge theory for 1-Laplacian Δ_1 on manifolds. HE is a versatile framework for analyzing vector fields on a high-dimensional point cloud and can support various new realms of research. For example, numerous popular functional learning algorithms on manifolds using the graph Laplacian [32] can be easily extended to vector field learning with the topological and geometric information encoded in the discrete Helmholtzian $\boldsymbol{\mathcal{L}}_1$.

In this chapter, some of these possibilities are explored under the HE framework. Unlike the PD-based algorithms that do not keep track of the orientations, HE can be (i) adapted to vector field learning using the eigenflows of $\boldsymbol{\mathcal{L}}_1$ in a non-parametric manner. Furthermore, the high-frequency information can be easily obtained, compared with SEC having relatively higher computational and memory dependencies. Using the proposed Helmholtzian estimator $\boldsymbol{\mathcal{L}}_1$, (ii) we extend the well-studied vertex-based function smoothing and SSL on manifolds to the domain of vector field learning. Specifically, we present applications in vector field smoothing, (iii) in completing partially observed edge flows, and in (iv) estimating underlying velocity field from trajectories on the synthetic, oceanography, and RNA single-cell sequencing dataset; some of these datasets have complicated manifold structures. These experiments have shown that the proposed HE framework is more effective than SEC, for the higher-frequency information is utilized more efficiently, and more robust than $\mathbf{L}_1$, which uses constant weights on triangles, in learning vector fields on a manifold.

Chapter 6

THE DECOMPOSITION OF THE HOMOLOGY EMBEDDING OF THE k-LAPLACIAN

The k-th *homology vector space* $\mathcal{H}_k$ provides rich geometric information on manifolds or networks. For instance, the zeroth, the first, and the second homology vector spaces identify the connected components, the loops, and the cavities in the manifold, respectively. Topological Data Analysis (TDA) [128], as well as other early works in this field, aims to extract the dimension of $\mathcal{H}_k$ and has found wide use in analyzing biological [67, 102], human behavior [2, 132], or other complex systems [128]. Even though they easily generalize to $k \geq 1$, additional efforts are needed to extract topological features (e.g., instances of loops) besides ranks due to the combinatorial complexity of the structures that support them.

Spectral methods based on k-Laplacians ($\boldsymbol{\mathcal{L}}_k$), by contrast, investigate $\mathcal{H}_k$ in a linear algebraic manner; abundant geometric information can be extracted from the *homology embedding* $\mathbf{Y}$ (the null space eigenvectors of $\boldsymbol{\mathcal{L}}_k$) of $\mathcal{H}_k$. Analysis of the eigenfunctions (of $\mathcal{H}_0$) [32, 79, 87, 106] of the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ is pivotal in providing guarantees for spectral clustering and community detection algorithms. Recent advances in this field [6, 28, 105] extend the existing spectral algorithms based on $\boldsymbol{\mathcal{L}}_0$ to $k \geq 1$; however, theoretical analysis in $\mathbf{Y}$ of $\mathcal{H}_k$, unlike spectral clustering, is less developed, in spite of intriguing empirical results by Ebli and Spreemann [45].

In this chapter, we put these observations on a formal footing based on the concepts of *connected sum* and *prime decomposition* of manifolds (Section 6.1). We examine these operations through the lens of the (subspace) perturbation to the homology embedding $\mathbf{Y}$ of the discrete k-Laplacian $\boldsymbol{\mathcal{L}}_k$ on finite samples (Section 6.2). This framework finds applications in, i.e., identifying the *shortest homologous loops* (Section 6.3). Lastly, we support our theoretical claims with numerous empirical results from point clouds and images.

6.1 Definitions, theoretical/algorithmic aims, and prior works

6.1.1 Connected sum and manifold (prime) decomposition

First, we introduce the concept of *connected sum*, which is the core tool we will use in this chapter. The connected sum [71] of two d dimensional manifolds $\mathcal{M} = \mathcal{M}_1 \sharp \mathcal{M}_2$ is built from

removing two d dimensional "disks" from each manifold $\mathcal{M}_1$, $\mathcal{M}_2$ and gluing together two manifolds at the boundaries (technical details in [71]). The analog of the connected sum for the abstract complexes will be defined in Section 6.1. The connected sum is a core operation in topology and is related to the concept of manifold (prime) decomposition. Informally speaking, the prime decomposition aims to factorize a manifold $\mathcal{M}$ into κ smaller building blocks ($\mathcal{M} = \mathcal{M}_1 \sharp \cdots \sharp \mathcal{M}_\kappa$) so that each $\mathcal{M}_i$ cannot be further expressed as a connected sum of other manifolds. The well-known *classification theorem of surfaces* [4] states that any oriented and compact surface is the finite connected sum of manifolds homeomorphic to either a circle $\mathbb{S}^1$, a sphere $\mathbb{S}^2$, or a torus $\mathbb{T}^2$. Classification theorems for $d > 2$ are currently unknown; fortunately, the uniqueness of the prime decomposition for $d = 3$ was shown (Kneser-Milnor theorem [82]). Recently, Bokor et al. [18] (Corollary 2.5) showed the existence of factorizations of manifolds with $d \geq 5$, even though they might not be unique.

6.1.2 Definitions

The data $\mathbf{X}$ is sampled from a d-dimensional *oriented* manifold $\mathcal{M}$ that can be decomposed into κ prime manifolds ($\mathcal{M} = \mathcal{M}_1 \sharp \cdots \sharp \mathcal{M}_\kappa$). Let $\mathcal{I}_i$ be an index set of the data points in $\mathbf{X}$ sampled from $\mathcal{M}_i$, for $i = 1, \ldots \kappa$. Denote by SC_k, $\boldsymbol{\mathcal{L}}_k$, $\mathcal{H}_k(\mathcal{M})$, and β_k the simplicial complex, the symmetrized k-Laplacian[1], the k-homology space, and the k-th Betti number of $\mathcal{M}$. Furthermore, let $\hat{\mathrm{SC}}_k^{(i)} = (\hat{\Sigma}_0^{(i)}, \cdots, \hat{\Sigma}_k^{(i)})$, $\hat{\boldsymbol{\mathcal{L}}}_k^{(ii)}$, $\mathcal{H}_k(\mathcal{M}_i)$, $\beta_k(\mathcal{M}_i)$ be the same quantities for manifold $\mathcal{M}_i$ (supported on $\mathcal{I}_i$ for $i \leq \kappa$). $\hat{\mathrm{SC}}_k$ and $\hat{\boldsymbol{\mathcal{L}}}_k$ (without superscript i) are the comparable notations for the disjoint manifolds $\mathcal{M}_i$'s, i.e. $\hat{\mathrm{SC}} = \cup_{i=1}^{\kappa} \hat{\mathrm{SC}}^{(i)} = (\hat{\Sigma}_0, \cdots, \hat{\Sigma}_k)$ with $\hat{\Sigma}_\ell = \cup_{i=1}^{\kappa} \hat{\Sigma}_\ell^{(i)}$ for $\ell \leq k$, and $\hat{\boldsymbol{\mathcal{L}}}_k$ is a block diagonal matrix with the i-th block being $\hat{\boldsymbol{\mathcal{L}}}_k^{(ii)}$. Additionally, let $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ (both in $\mathbb{R}^{n_k \times \beta_k}$) be the homology basis of $\boldsymbol{\mathcal{L}}_k$ and $\hat{\boldsymbol{\mathcal{L}}}_k$, respectively. Let $\mathcal{S}_i$ be the index set of columns of $\hat{\mathbf{Y}}$ corresponding to homology subspace $\mathcal{H}_k(\mathcal{M}_i)$, with $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ for $i \neq j$, $|\mathcal{S}_i| = \beta_k(\mathcal{M}_i)$, and $\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_\kappa = \{1, \cdots, \beta_k\}$. Since $\hat{\mathbf{Y}}$ is the homology embedding of a block diagonal matrix $\hat{\boldsymbol{\mathcal{L}}}_k$, it follows that $[\hat{\mathbf{Y}}]_{\sigma,m}$ equals the

[1]With slight abuse of notations, we will denote the *symmetrized* k-Laplacian as in (2.16) by $\boldsymbol{\mathcal{L}}_k$ in this chapter.

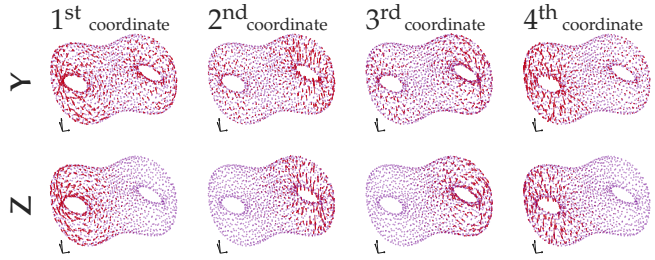

Figure 6.1: Harmonic vector fields obtained by solving a least-squares [28] with $\mathbf{Y}$ (top) and $\mathbf{Z}$ (bottom).

homology embedding of $\hat{\boldsymbol{\mathcal{L}}}_k^{(ii)}$ if $\sigma \in \hat{\Sigma}_k^{(i)}$ with column $m \in \mathcal{S}_i$ and is zero otherwise. Namely, $\hat{\mathbf{Y}}$ lies in the direct sum of subspaces $\mathcal{H}_k(\mathcal{M}_i)$ for $i \leq \kappa$.

Based on the construction above, we are interested in the following: given finite samples from $\mathcal{M}$, which is a κ-fold connected sum of $\mathcal{M}_i$, can this decomposition be recovered from the discrete homology embedding $\mathbf{Y}$ of $\mathcal{M}$? If so, how can we recover this information?

6.1.3 Theoretical aim

We are interested in the geometric properties of the null space eigenvectors $\mathbf{Y}$, and specifically in recovering the homology basis $\hat{\mathbf{Y}}$ of the prime manifolds. Hence, we aim to bound the *distance* between the spaces spanned by $\mathbf{Y}$ and $\hat{\mathbf{Y}}$. Under a small perturbation, one can provide an analogous argument to the *orthogonal cone* structure [87, 106] in spectral clustering (the zeroth homology embedding). The main technical challenge is that the connected sum of manifolds is a highly localized perturbation; namely, most cells are not affected at all, while those involved in the gluing process gain or lose $\mathcal{O}(1)$ (co)faces. Without properly designing $\boldsymbol{\mathcal{L}}_k$ and $\hat{\boldsymbol{\mathcal{L}}}_k$, one might get a trivial bound.

6.1.4 Algorithmic aim

We exemplify the algorithmic aim using $k = 1$, $d = 2$, and $\kappa = 2$, particularly the genus-2 surface shown in Figure 6.1. The null space basis $\mathbf{Y}$ of $\boldsymbol{\mathcal{L}}_k$ is only identifiable up to a unitary

matrix due to the multiplicity of the zero eigenvalues. For instance, the top and bottom rows of Figure 6.1 are both valid bases for the edge flow in $\mathcal{H}_1$. However, the basis vector fields in the second row of Figure 6.1 are more interpretable than those in the top row because $\mathbf{Y}$ (the first row) is a linear combination of $\mathbf{Z}$ (the second row), with each basis (column in the figure) corresponding to a single homology class (loop). Therefore, here we propose a *data-driven* approach to obtain the optimal basis $\mathbf{Z}$ such that the coupling from other manifolds/subspaces is as weak as possible. Being able to obtain $\mathbf{Z}$ from an arbitrary $\mathbf{Y}$ can support numerous applications (more in Section 6.3); however, it is difficult to design a criterion for finding the optimal $\mathbf{Z}$ without knowing the geometric structure of $\mathcal{H}_k$.

6.1.5 Prior works

The shape of the embedding of the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ is pivotal for showing the guarantees of spectral clustering algorithms for point cloud data or the inference algorithms for the stochastic block model. The analyses used either the matrix perturbation theory [87, 125, 126] or assume a mixture model [106]. For the higher-order k-Laplacian, it is reported empirically that the homology embedding is approximately distributed on the union (directed sum) of subspaces [45]; subspace clustering algorithms [65] were applied to partition edges/triangles under their framework.

6.2 Main result: connected sum as a matrix perturbation

6.2.1 Embedding perturbation

In this section, we analyze the geometric structure of $\mathbf{Y}$ by viewing the operation of *connected sum* through the lens of matrix perturbation theory [115]. We show that, under certain conditions, the homology embedding $\mathbf{Y}$ of the joint Laplacian $\boldsymbol{\mathcal{L}}_k$ is approximated by $\hat{\mathbf{Y}}$ for the simplexes that are not created/destroyed during connected sum. In matrix terms, we show that $\mathbf{Y} \approx \hat{\mathbf{Y}}\mathbf{O}$ (Theorem 6.1) with $\mathbf{O}$ a unitary transformation.

We first prepare our assumptions suited for SC built from point clouds. Most of the

assumptions (except Assumption 6.1 for which the connected sum might not be defined) can be extended to the clique complex (for networks) or cubical complex (for images) without too many modifications.

Assumption 6.1. *The point cloud* $\mathbf{X} \in \mathbb{R}^{n\times D}$ *is sampled from a* d*-dimensional oriented and compact manifold* $\mathcal{M} \subseteq \mathbb{R}^{n\times D}$*; the homology vector spaces* $\mathcal{H}_k(\mathrm{SC})$ *formed by the simplicial complex constructed from* $\mathbf{X}$ *are isomorphic to the homology group* $\mathcal{H}_k(\mathcal{M})$ *of* $\mathcal{M}$*, i.e.,* $\mathcal{H}_k(\mathrm{SC}) \cong \mathcal{H}_k(\mathcal{M})$*. Furthermore, assume that* $\mathcal{M} = \mathcal{M}_1 \sharp \cdots \sharp \mathcal{M}_\kappa$*, and that* $\mathcal{H}_k(\hat{\mathrm{SC}}^{(i)}) \cong \mathcal{H}_k(\mathcal{M}_i)$ *for* $i = 1, \cdots, \kappa$*.*

This assumption is the minimal assumption needed for the analysis of the embedding of the $\boldsymbol{\mathcal{L}}_k$; it states that any procedure to construct the simplicial complex or weight function for $\boldsymbol{\mathcal{L}}_k$ is accepted as long as the isomorphic condition holds. The construction of the SC from the point cloud is out of the scope of this chapter (see, e.g., Chapter 4 or Chen et al. [28] for building $\boldsymbol{\mathcal{L}}_1$ from $\mathbf{X}$ with an analyzable limit). The last condition requires that the manifold $\mathcal{M}$ can be decomposed; this is most likely true, except for the known hard case of $\mathcal{M}$ with $d = 4$ discussed in Section 6.1.1. To make this assumption hold for networks or images, one can require that $\boldsymbol{\mathcal{L}}_k$ constructed from these two datasets can be roughly factorized into block-diagonal entries. Below we provide two other assumptions that are valid for both SC and CB (with some modifications): the first one controls the eigengap and the second one ensures a small perturbation in the spectral norm of $\boldsymbol{\mathcal{L}}_k - \hat{\boldsymbol{\mathcal{L}}}_k$. By construction, $\boldsymbol{\mathcal{L}}_k$ is positive semi-definite; since we are interested in the stability of its null space, we define, for any matrix $\mathbf{L} \succeq 0$, the *eigengap* as the the smallest *non-zero* eigenvalue of $\mathbf{L}$ and denote it $\lambda_{\min}(\mathbf{L})$.

Assumption 6.2. *We denote the set of destroyed and created* k*-simplexes during connected sum by* $\mathfrak{D}_k$ *and* $\mathfrak{C}_k$*, respectively; let the set of non-intersecting simplexes be* $\mathfrak{N}_k = \Sigma_k \backslash \mathfrak{C}_k = \hat{\Sigma}_k \backslash \mathfrak{D}_k$*. We have: (1) no* k*-homology class is created during the connected sum process, i.e.,* $\beta_k(\mathrm{SC}) = \sum_{i=1}^{\kappa} \beta_k(\hat{\mathrm{SC}}^{(i)})$*. (2) The eigengaps of* $\boldsymbol{\mathcal{L}}_k^{\mathfrak{C},\mathfrak{C}}$ *and* $\hat{\boldsymbol{\mathcal{L}}}_k^{\mathfrak{D},\mathfrak{D}}$ *are bounded away from*

the eigengaps of $\boldsymbol{\mathcal{L}}_k^{(ii)}$*, i.e.,* $\min\{\lambda_{\min}(\boldsymbol{\mathcal{L}}_k^{\mathfrak{C},\mathfrak{C}}), \lambda_{\min}(\hat{\boldsymbol{\mathcal{L}}}_k^{\mathfrak{D},\mathfrak{D}})\} \gg \min\{\delta_1, \cdots, \delta_\kappa\}$*, where* δ_i *is the* eigengap *of* $\boldsymbol{\mathcal{L}}_k^{(ii)}$*.*

The first condition requires that the intersecting simplexes $\mathfrak{D}_k \cup \mathfrak{C}_k$ do not create or destroy any k-th homology class; this holds, for instance, when the manifold $\mathcal{M}$ has dimension $d > k$. Under this condition, we have $\mathcal{H}_k(\mathcal{M}_1 \sharp \mathcal{M}_2) \cong \mathcal{H}_k(\mathcal{M}_1) \oplus \mathcal{H}_k(\mathcal{M}_2)$ [71]. A counterexample for this condition is, e.g., inspecting the cavity space ($k = 2$) of a genus-2 surface built from gluing two tori together. That is, β_2 of a genus-2 surface is 1, while the sum of β_2 of two tori is 2. The second condition requires that the principal submatrix of $\boldsymbol{\mathcal{L}}_k$ described by the block of $\mathfrak{C}_k \cup \mathfrak{D}_k$ has large eigengap. This happens, e.g., when $\mathfrak{C}_k$ and $\mathfrak{D}_k$ are cliques and are contained in small balls.

Assumption 6.3 (Informal)**.** *Let* $\tilde{\mathbf{w}}_k = |\mathbf{B}_{k+1}[\mathfrak{N}_k, \mathfrak{N}_{k+1}]|\mathbf{w}_{k+1}$*,* $\tilde{\mathbf{w}}_{k-1} = |\mathbf{B}_k[:, \mathfrak{N}_k]|\tilde{\mathbf{w}}_k$*. For* $\ell = k$ *or* $k-1$*, we have* $\max_{\sigma \in \mathfrak{N}_\ell} \{w_\ell(\sigma)/\tilde{w}_\ell(\sigma) - 1\} \leq \epsilon_\ell$*,* $\max_{\sigma \in \mathfrak{N}_\ell} \{\hat{w}_\ell(\sigma)/\tilde{w}_\ell(\sigma) - 1\} \leq \epsilon_\ell$*, and* $\max_{\sigma \in \mathfrak{N}_\ell} \{|w_\ell(\sigma)/\hat{w}_\ell(\sigma) - 1|\} \leq \epsilon'_\ell$*. Assumption 6.4 is the formal version of this assumption.*

For $k = 1$, it states that not too many triangles are being created or destroyed during connected sum. For this assumption to hold, the density in the connected sum region should be smaller than in other regions, i.e., the manifold $\mathcal{M}$ should be sparsely connected (e.g., Figure 6.2a). Empirically, we observed that the perturbation is small even when $\mathcal{M}$ is not sparsely connected (more discussions in Section 6.4). Note also that $\epsilon'_\ell \ll \epsilon_\ell$, for ϵ'_ℓ represents the *net* change in the degree after connected sum. It might be possible to obtain a tighter bound fully by ϵ'_ℓ's, which do not depend on the relative density between the connected sum region and the remaining manifolds; we leave it as future work.

Theorem 6.1. *Let* $\mathsf{DiffL}_k^{\text{down}}$ *be the modified difference (defined in Appendix 6.6) of* $\boldsymbol{\mathcal{L}}_k^{\text{down}}$ *and* $\hat{\boldsymbol{\mathcal{L}}}_k^{\text{down}}$*, same for that (*$\mathsf{DiffL}_k^{\text{up}}$*) of* up *Laplacians. Under Assumptions 6.1–6.3 with notations defined as before and* $\lambda_k = k + 2$*, if*

$$\left\|\mathsf{DiffL}_k^{\text{down}}\right\|^2 \leq \left[2\sqrt{\epsilon'_k} + \epsilon'_k + \left(1 + \sqrt{\epsilon'_k}\right)^2 \sqrt{\epsilon'_{k-1}} + 4\sqrt{\epsilon_{k-1}}\right]^2 \lambda_{k-1}^2,$$

and

$$\|\mathsf{DiffL}_k^{\text{up}}\|^2 \leq \left[2\sqrt{\epsilon'_k} + \epsilon'_k + 2\epsilon_k + 4\sqrt{\epsilon_k}\right]^2 \lambda_k^2,$$

then there exists a unitary matrix $\mathbf{O} \in \mathbb{R}^{\beta_k \times \beta_k}$ *such that*

$$\left\|\mathbf{Y}_{\mathfrak{N}_k,:} - \hat{\mathbf{Y}}_{\mathfrak{N}_k,:}\mathbf{O}\right\|_F^2 \leq \frac{8\beta_k\left[\left\|\mathsf{DiffL}_k^{\text{down}}\right\|^2 + \|\mathsf{DiffL}_k^{\text{up}}\|^2\right]}{\min\{\delta_1, \cdots, \delta_\kappa\}}. \tag{6.1}$$

Sketch of proof. The proof (in Section 6.6) is based on bounding the error between $\boldsymbol{\mathcal{L}}_k$ and $\hat{\boldsymbol{\mathcal{L}}}_k$ with $\tilde{\boldsymbol{\mathcal{L}}}_k$ (the Laplacian after removal of k-simplices during connected sum), the use of a variant of the Davis-Kahan theorem [130], and the bound of the spectral norm of $\boldsymbol{\mathcal{L}}_k$ for a simplicial complex, i.e., $\|\boldsymbol{\mathcal{L}}_k\|_2 \leq \lambda_k = k + 2$ [58]. ■

What is unusual for the bound is that the LHS of (6.1) contains only the simplices in $\mathfrak{N}_k$. It is unlikely that one can get a small bound for the simplices in $\mathfrak{C}_k \cup \mathfrak{D}_k$ since they do not exist before or after gluing manifolds together. Nonetheless, (6.1) makes sure that the (unbounded) perturbations in the embedding of $\mathfrak{C}_k \cup \mathfrak{D}_k$ do not propagate to the rest of the simplices. The bound in (6.1) can be extended to CB (Corollary 6.2) by changing the λ_k value from $(k + 2)$ to $2k + 2$. The $2k + 2$ term here is the maximum eigenvalue of the $\boldsymbol{\mathcal{L}}_k$ built from any cubical complex (Proposition 6.7).

Corollary 6.2 (For $\boldsymbol{\mathcal{L}}_k$ built from a CB)**.** *Under Assumptions 6.2–6.3 with* $\mathsf{DiffL}_k^{\text{up}}$ *as well as* $\mathsf{DiffL}_k^{\text{down}}$ *defined in Theorem 6.1 and* $\lambda_k = 2k + 2$*, there exists a unitary matrix* $\mathbf{O}$ *such that* (6.1) *holds.*

6.2.2 Subspace identification

We propose to (approximately) separate the columns of the coupled basis $\mathbf{Y}$ to an independent basis $\mathbf{Z}$ (as an approximation to $\hat{\mathbf{Y}}$), with columns being a permutation of $\{1, \cdots, \beta_k\}$, by *blind source separation*, as described by Algorithm 6.1. Specifically, $\mathbf{Z}$ is obtained by

Algorithm 6.1: SUBSPACEIDENTIFY: identify the k-homology space $\mathcal{H}_k$ of a simplicial/cubical complex using ICA

Input : SC, k, weights $\mathbf{W}_{k+1}$

1 $\mathbf{B}_k, \mathbf{B}_{k+1} = \text{BOUNDARYMAPS}(\text{SC}, k)$ ▷ `refer to Sections 2.4.1 and 2.6.1`

2 **for** $\ell = k, k-1$ **do**

3 $\quad \mathbf{W}_\ell \leftarrow \text{diag}\{|\mathbf{B}_{\ell+1}|\mathbf{W}_{\ell+1}\mathbf{1}_{n_{\ell+1}}\}$

4 $\quad \mathbf{A}_{\ell+1} \leftarrow \mathbf{W}_\ell^{-1/2}\mathbf{B}_{\ell+1}\mathbf{W}_{\ell+1}^{1/2}$

5 $\boldsymbol{\mathcal{L}}_k = \mathbf{A}_k^\top \mathbf{A}_k + \mathbf{A}_{k+1}\mathbf{A}_{k+1}^\top$

6 $\mathbf{Y} \in \mathbb{R}^{n_k \times \beta_k} \leftarrow \text{NULLSPACE}(\boldsymbol{\mathcal{L}}_k)$

7 $\mathbf{Z} \leftarrow \text{ICANOPREWHITE}(\mathbf{Y})$

Return : Independent basis $\mathbf{Z}$

Infomax ICA [11] on $\mathbf{Y}$ of $\boldsymbol{\mathcal{L}}_k$, with a modification (Line 7) that preserves the necessary properties of harmonic cochains (i.e., they are *divergence-free* and *curl-free*, see also Proposition 6.3). Algorithm 6.1 works for CB as well by using the appropriate $\mathbf{B}_k$, $\mathbf{B}_{k+1}$ construction method (Line 1).

6.3 Applications: homologous loops detection, clustering, and visualization

6.3.1 Homologous loop detection

In addition to the *rank* information available from classical TDA methods, one might find it beneficial to extract the shortest cycle of the corresponding $\mathcal{H}_k$ generator. This application is found useful in domains including finding minimum energy trajectories in molecular dynamics datasets, trajectory inference in RNA single-cell sequencing [101], and segmenting circular structures in medical images [113]. We propose a *spectral* shortest homologous loop detection algorithm (Algorithm 6.2) based on the shortest path algorithm (Dijkstra) as follows: for each dimension $i = 1, \cdots, \beta_1$, the algorithm reverses every edge e having negative $[\mathbf{z}_i]_e$ to generate a weighted digraph $G_i = (V, E_i)$ (Lines 2–3), with the weight of edge $e \in E_i$ equal to the Euclidean distance $[\mathbf{d}]_{(i,j)} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$. The algorithm finds a shortest (in terms of $\mathbf{d}$) loop on this weighted digraph for each i and outputs it as the homologous loop representing

the i-th class. We present the following proposition to support Algorithm 6.2; it implies that if each coordinate of $\mathbf{Z}$ extracted from Algorithm 6.1 corresponds to a homology class, then the detected homologous loop for each homology class is the shortest.

Proposition 6.3. *Let $\mathbf{z}_i$ for $i = 1, \cdots, \beta_1$ be the i-th homology basis that corresponds to the i-th homology class. For every $i = 1, \cdots, \beta_1$, (1) there exist at least one* cycle *in the digraph G_i such that every vertex $v \in V$ can traverse back to itself (*reachable*); (2) the corresponding* cycle *will enclose at least one homology class (*no short-circuiting*).*

Proof. Reachable: the harmonic flow is divergence free, indicating that the incoming flow must be equal to the outgoing flow. If there exists a vertex that is not *reachable* to itself, then this vertex will either be a *source* or *sink* in the digraph. It violates the assumption that the flow is divergence free. Therefore such vertex will not exist.

No short-circuiting: the harmonic flow is curl free; from Stoke's theorem (or Poincaré Lemma [71]), we have that any path-integral travel along any homology class will be a constant. If there exists a loop such that it does not traverse along with any homology class, the loop integral along this cycle will be zero (by Stoke's theorem). By assumption, the path-integral will always be positive. To generate a loop whose integral is zero, one has to travel "upward" in the digraph; this violates the assumption that we are finding a cycle in the digraph, implying that every loop will traverse along at least one homology class. ■

Since every vertex is *reachable* from itself, we are guaranteed to find a loop for any starting/ending pair (Lines 9–12). Additionally, there will be no short-circuiting for any loop; each loop we found from Dijkstra is guaranteed to be non-trivial. However, there is one caveat from the second property: even though the i-th loop is non-trivial, it might not always be corresponding to the i-th homology class due to the noise in small $[\mathbf{z}_i]_e$. Namely, loops that do not represent i-th homology class can be formed with edges e having small $[\mathbf{z}_i]_e$, resulting in the instability and the (possible) duplication of the identified loops. To address the issue, we propose a heuristic thresholding, by which we keep the n_1/β_1 edges with the largest absolute value in $|[\mathbf{z}_i]_e|$ (Lines 4–5). We chose to keep n_1/β_1 by treating

Algorithm 6.2: LoopFind: spectral homologous loop detection from the factorized subspace $\mathbf{Z}$

Input : $\mathbf{Z} = [\mathbf{z}_1, \cdots, \mathbf{z}_{\beta_1}]$, V, E, edge distance $\mathbf{d}$

for $i = 1, \cdots, \beta_1$ **do**
- $E_i^+ \leftarrow \{(s,t) : (s,t) \in E \text{ and } [\mathbf{z}_i]_{(s,t)} > 0\}$
- $E_i^- \leftarrow \{(t,s) : (s,t) \in E \text{ and } [\mathbf{z}_i]_{(s,t)} < 0\}$
- $\tau \leftarrow \textsc{Percentile}(|\mathbf{z}_i|, 1 - 1/\beta_1)$
- $E_i^\times \leftarrow \{e \in E_i^+ \cup E_i^- : |[\mathbf{z}_i]_e| < \tau\}$
- $E_i \leftarrow E_i^+ \cup E_i^- \backslash E_i^\times$
- $G_i \leftarrow (V, E_i)$, with weight of $e \in E_i$ being $[\mathbf{d}]_e$
- $d_{\min} =$ `inf`
- **for** $e = (t, s_0) \in E_i$ **do**
  - $\mathcal{P}^*, d^* \leftarrow \textsc{Dijkstra}(G_i, \texttt{from=}s_0, \texttt{to=}t)$ ▷ `Note that` $\mathcal{P}^* = [s_0, s_1, \cdots, t]$
  - **if** $d^* < d_{\min}$ **then**
    - $\mathcal{C}_i \leftarrow [t, s_0, s_1, \cdots, t]$

Return: $\mathcal{C}_1, \cdots, \mathcal{C}_{\beta_1}$

each homology class equally, i.e., each class has roughly n_1/β_1 edges.

Compared with previous approaches that find the shortest loops [36] combinatorially, our approach has better time complexity; specifically, the algorithm of Dey et al. [36] has time complexity $\mathcal{O}(nn_1^3 + nn_1^2 n_2)$, whereas Algorithm 6.2 runs in time $\mathcal{O}(n_1^{2.37\cdots} + \beta_1^2 n_1 + \beta_1 n_1 n \log n)$. The first, second, and third terms correspond to the time complexity of eigendecomposition of $\boldsymbol{\mathcal{L}}_1$, the Infomax ICA, and the Dijkstra algorithm on every digraph G_i, respectively. Note that if the simplicial complex is built from point clouds, the number of triangles n_2 may be large; this dependency on n_2 makes the algorithm [36] hard to scale. On the other hand, our framework requires that $\mathbf{z}_i$ are each supported on one homology class; therefore, loops can only be correctly identified using Algorithm 6.2 if the manifold is sparsely connected (Assumptions 6.1–6.3).

6.3.2 Classifying any 2-dimensional manifold

The Betti number β_1 of a torus is 2, which is equal to that of two disjoint circles; hence one cannot distinguish these two manifolds *only* by rank information. Fortunately, they can be categorized using the homology embedding $\mathbf{Z}$. By the classification theorem [4], any 2D surface is the connected sum of circles $\mathbb{S}^1$ and tori $\mathbb{T}^1$; therefore, Theorem 6.1 indicates that embedding lies approximately in the directed sum of homology subspace of $\mathbb{S}^1$ and/or $\mathbb{T}^2$. The homology embedding of $\mathbb{S}^1$ is a line since it is in $\mathbb{R}^1$. On the other hand, any loop in a torus can be a convex combination of the two homology classes, implying that the intrinsic dimension of the homology embedding is 2. It is hard to obtain $\mathbf{Z}$ of any arbitrary torus; we present the homology embedding of the flat m-torus below by expressing the null space basis (1-cochains) as the path integrals of the corresponding harmonic 1-forms [28, 129].

Proposition 6.4. *The envelope of the first homology embedding (1-cochain) induced by the harmonic 1-form on the flat m-torus $\mathbb{T}^m$ is an m-dimensional ellipsoid.*

Proof. The harmonic vector field in an m-flat torus $\mathbb{T}^m$ is a constant in each coordinate, i.e., $\mathbf{v} = [v_1, \cdots, v_m] \in \mathbb{R}^m$. The manifold $\mathbb{T}^m$ is an m-dimensional cube with the periodic boundary condition, i.e., $0 = 2\pi$. From [28, 129], the edge flow $\boldsymbol{\omega}_e$ for an edge $e = (i, j) \in E$ can be written exactly as a linear map, i.e.,

$$\begin{aligned}\omega_e &= \int_0^1 \mathbf{v}^\top(\boldsymbol{\gamma}(t))\gamma'(t)\mathsf{d}t = \int_0^1 [\mathbf{v}(\mathbf{x}_i) + (\mathbf{v}(\mathbf{x}_j) - \mathbf{v}(\mathbf{x}_i))t]^\top(\mathbf{x}_j - \mathbf{x}_i)\mathsf{d}t \\ &= \frac{1}{2}(\mathbf{v}(\mathbf{x}_i) + \mathbf{v}(\mathbf{x}_j))^\top(\mathbf{x}_j - \mathbf{x}_i)\end{aligned}$$

Where $\boldsymbol{\gamma}(t)$ is the geodesic on $\mathcal{M}$ connecting $\mathbf{x}_i$ and $\mathbf{x}_j$ with $\boldsymbol{\gamma}(0) = \mathbf{x}_i$ and $\boldsymbol{\gamma}(1) = \mathbf{x}_j$. Any point $\mathbf{x} \in \mathbb{R}^m$, with $r = \|\mathbf{x}\|$, can be written as $\mathbf{x} = [rf_1(\boldsymbol{\Phi}), rf_2(\boldsymbol{\Phi}), \cdots, rf_m(\boldsymbol{\Phi})]$, where $\boldsymbol{\Phi} \in \mathbb{R}^{m-1}$ is the high-dimensional polar coordinate; for instance, a point in 2D is $[r\cos(\theta), r\sin(\theta)]$ with $\boldsymbol{\Phi} = [\theta]$, while a point in 3D having $\boldsymbol{\Phi} = [\theta, \varphi]$ is $[r\cos\varphi\sin\theta, r\sin\varphi\sin\theta, r\cos\theta]$. The conditional distribution given a fixed $\boldsymbol{\Phi}$ is simply the distribution of edge lengths, i.e., $p(rv_1f_1, \cdots, rv_mf_m|\boldsymbol{\Phi}) = p(r)$. Since $p(r)$ is bounded by some constant δ representing the

maximum edge length, the envelope of the distribution is bounded by $[\delta v_1 f_1(\mathbf{\Phi}), \cdots, \delta v_m f_m(\mathbf{\Phi})]$, indicating that it is an m-ellipsoid with the length of the i-th semi-axes being δv_i. ■

Proposition 6.4 and the classification theorem suggest that the first homology embedding is either a line, a disk, or a combination of the two (with replacement). See an example for the genus-2 surface in Figures 6.2j and 6.4.

6.3.3 Other applications

As pointed out earlier, one can visualize the basis of the harmonic vector fields (of $\mathcal{H}_k$) by overlaying the columns of $\mathbf{Y}$ onto the original dataset (Figure 6.1). Being able to successfully extract a decoupled basis $\mathbf{Z}$ increases the interpretability of $\mathcal{H}_k$, as shown in the second row of Figure 6.1. Theorem 6.1 also supports the use of subspace clustering algorithm in the higher-order simplex clustering framework [45].

6.4 Experiments

We demonstrate our approach by computing $\mathbf{Y}, \mathbf{Z}$ and the shortest loops for five synthetic manifolds: two of them are prime manifolds (`TORUS` *torus*, `3-TORUS` *three-torus*) and three (`PUNCTPLANE` *punctured plane with two holes*, `GENUS-2` *genus-2*, and `TORI-CONCAT` *concatenation of 4 tori*) are factorizable manifolds. Furthermore, five additional real point clouds (`ETH` and `MDA` from chemistry, `PANCREAS` from biology, `3D-GRAPH` from 3D modeling, and `SOUTH-ISLANDS` from oceanography) are analyzed under this framework. For all the point clouds, we build the VR complex SC from the CkNN kernel [15] so that the resulting $\boldsymbol{\mathcal{L}}_1$ is sparse and the topological information is preserved. Note that other methods for building an SC from $\mathbf{X}$ can also be used as long as $\mathcal{H}_k$ is successfully identified (Assumption 6.1). Lastly, we illustrate the efficacy of our framework to a non-manifold data: `RETINA` from medical imaging.

6.4.1 Synthetic manifolds

The results for the synthetic manifolds are in Figure 6.2. Figure 6.2a (the harmonic embedding of `PUNCTPLANE`) confirms Theorem 6.1 that $\mathbf{Y}$ is approximately distributed on two subspaces (yellow and red), with each loop parametrizing a single hole (inset of Figure 6.2a). As discussed previously in Figure 6.1, the harmonic vector bases (green and blue) are mixtures of the separate subspaces; therefore, these bases have poor interpretability compared with the independent subspace $\mathbf{Z}$ identified by Algorithm 6.1. The shortest loops (Figure 6.2b) corresponding to $\mathbf{z}_1$ (yellow), $\mathbf{z}_2$ (red) are obtained by running Dijkstra on the digraphs induced by $\mathbf{z}_1$ and $\mathbf{z}_2$ separately (Algorithm 6.2). Figures 6.2c–6.2f show the results of the two simple *prime manifolds*: `TORUS` and `3-TORUS`. The harmonic embeddings of `TORUS` (Figure 6.2d) and `3-TORUS` (Figure 6.2f) are a two-dimensional disk and a three-dimensional ellipsoid, respectively; this confirms the conclusion from Proposition 6.4. The shortest loops obtained from Algorithm 6.2 for these two datasets are in Figures 6.2c and 6.2e, showing that these loops travel around the holes in `TORUS` (or `3-TORUS`). Note that we plot `3-TORUS` in the intrinsic coordinate because a three torus can not be embedded in 3D without breaking neighborhood relationships. Three lines in Figure 6.2e are indeed loops due to the periodic boundary condition, i.e., $0 = 2\pi$, in the intrinsic coordinate. Figures 6.2h and 6.2j show the embedding of the coupled harmonic basis ($\mathbf{Y}$) and that corresponding to the independent subspace ($\mathbf{Z}$) obtained by Algorithm 6.1. Compared with $\mathbf{Y}$, each coordinate of $\mathbf{Z}$ corresponds to a subspace, i.e., the left or right handle of `GENUS-2`, and does not couple with other homology generators. $\mathbf{Z}$ is thus a union of two 2D disks, with each disk approximating the harmonic embedding of a torus (see Figure 6.4 for more detail). Compared with the loops obtained by running Algorithm 6.2 on $\mathbf{Y}$ (Figure 6.2g), each loop in Figure 6.2i identified from $\mathbf{Z}$ parameterizes the corresponding homology generator without being homologous to other loops. Similar results on `TORI-CONCAT` are in Figures 6.2k and 6.2l, which correspond to the loops obtained from $\mathbf{Y}$ and $\mathbf{Z}$, respectively. The pairwise scatter plots of the eight-dimensional $\mathbf{Z}$ (or $\mathbf{Y}$) are in Figure 6.5. Note that `PUNCTPLANE` is an example of a

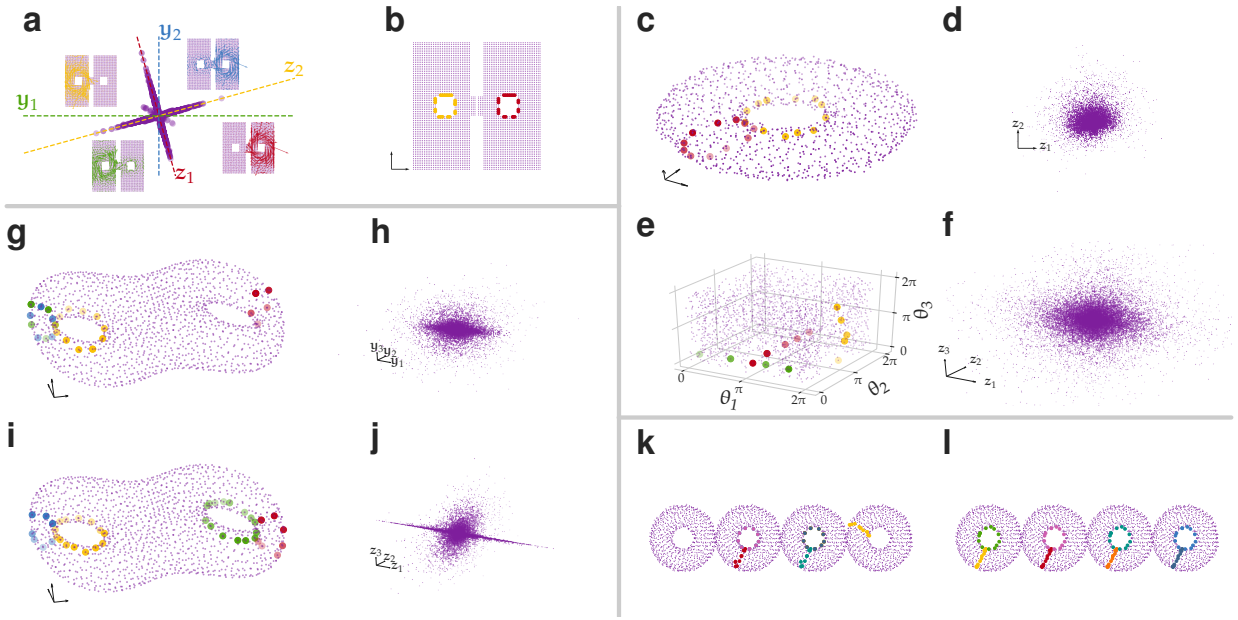

Figure 6.2: (a) The first homology embedding of PUNCTPLANE. The harmonic vector fields are overlaid on the data in the inset plots; green, blue, red, and yellow arrows correspond to $\mathbf{y}_1$, $\mathbf{y}_2$, $\mathbf{z}_1$, and $\mathbf{z}_2$, respectively. (b), (c), (e), (i), and (l) are the detected loops using Dijkstra on $\mathbf{Z}$ for PUNCTPLANE (colors are in (a)), TORUS, 3-TORUS, GENUS-2, and TORI-CONCAT, respectively. (g) and (k) represent the identified loops on the coupled embedding $\mathbf{Y}$ for GENUS-2 and TORI-CONCAT, respectively. (d), (f), (h), and (j) present the embeddings used to detect loops in (c), (e), (g), and (i), respectively.

sparsely connected manifold (see the low-density area in the middle), with $\epsilon_1 \approx 0.035$ and $\epsilon_0 \approx 0.038$. Manifolds of other synthetic/real datasets might not be sparsely connected due to the (approximately) constant sampling densities; nevertheless, the perturbations to the subspaces remain small for these datasets.

6.4.2 Small molecule data [29]

Figures 6.3a–6.3c and 6.3d–6.3f show our analysis on ETH and MDA, respectively. These two small molecule datasets, whose ambient dimensions are $D = 102$ and $D = 98$, are suggested to be noisy non-uniformly sampled tori [117]; the harmonic embeddings of these two datasets (Figures 6.3c and 6.3f) confirm this idea. Finding the minimum trajectories corresponding to

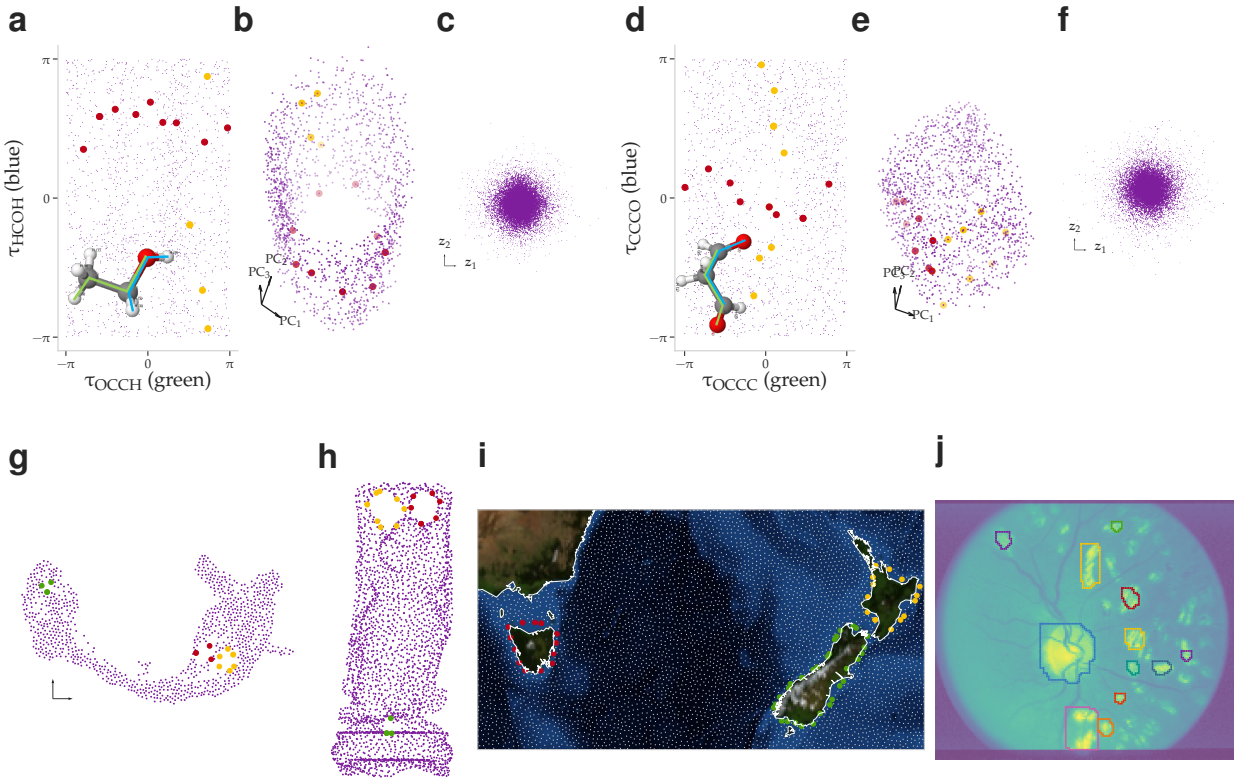

Figure 6.3: (a) and (b) are the detected loops of `ETH` using Dijkstra on $\mathbf{Z}$ (in (c)) in the torsion space (inset of (a)) and in the PCA space, respectively. (d)–(f) are the results for `MDA` that are similar to those for `ETH` in (a)–(c). (g)–(j) show the identified loops using $\mathbf{Z}$ for `PANCREAS`, `3D-GRAPH`, `SOUTH-ISLANDS`, and `RETINA`, respectively.

a specific bond torsion is of interest in chemistry; in these two molecular dynamics systems, this problem can be translated into finding the homologous loops in the point cloud. The homologous loops found by Algorithm 6.2 overlaid on the first three *principal components* (PCs) for these two datasets can be found in Figures 6.3b (for `ETH`) and 6.3e (for `MDA`). The identical homologous loops plot in the bond torsion space (with definition in the insets) based on our prior knowledge are in Figures 6.3a and 6.3d. Similar to the discussion for `3-TORUS` (Figure 6.2e), the yellow/red trajectories form loops due to the periodic boundary condition of the bond torsions.

6.4.3 RNA single-cell sequencing data [12]

The *trajectory inference* methods [101] for analyzing the RNA single-cell sequencing datasets aim to order the cells (points in high-dimensional expression space) along developmental trajectories, which are inferred from the structure of the point clouds. Identifying loops in the dataset can serve as a building block for delineating a correct trajectory, especially for determining cell cycle and cell differentiation.

We illustrate the idea by analyzing `PANCREAS`[12], another single-cell RNA sequencing data of the pancreas cells that exhibits cell cycles in the developmental stage. The 1-Laplacian is computed on the CkNN kernel [15] constructed on the UMAP [76] embedding (Algorithm 6.1). Figure 6.3g shows the identified loops from Algorithm 6.2, with the green loop being the cycle of ductal cells and yellow/red loops representing a trifurcation (endocrine cell differentiation).

6.4.4 Additional point cloud datasets.

`3D-GRAPH` [34] is a 3D model of a Buddha statue[2] with a precomputed triangulation by [34]. In other words, the simplicial complex $\mathrm{SC}_2' = (V', E', T')$ is available beforehand, with $n' \approx 500\text{k}$ and $n_1' \approx 2\text{M}$. To illustrate the efficacy of our framework (and Theorem 6.1), we treat `3D-GRAPH` as a point cloud and build SC from the subsampled $n = 3,000$ furthest points by Algorithm 4.1. $\boldsymbol{\mathcal{L}}_1$ is obtained from the VR complex of the CkNN kernel. Note that with this small sample size, two smaller loops near the waist of the statue are not detectable. Hence, the number of zero eigenvalues of $\boldsymbol{\mathcal{L}}_1$ is 3, with the corresponding homology generators shown in Figure 6.3h.

`SOUTH-ISLANDS` [49], which contains ocean buoys around the Tasman sea, is the other point cloud in our analysis. To generate this data, we subsample $n = 5,000$ furthest points/buoys (Algorithm 4.1) from the ocean drifter data (discussed in Chapter 5) that were dated between 2010–2019 with longitudes within 142°E–179°E and latitudes between

[2]The dataset is available in `https://www.cc.gatech.edu/projects/large_models/`.

48°S–33°S. The estimated β_1 is 3, with the detected loops being the North Island of New Zealand, the South Island of New Zealand, and the main island of Tasmania (Figure 6.3i).

6.4.5 Non-manifold dataset.

Our framework for identifying subspaces is still valid for cubical complexes built from images (by Corollary 6.2). We demonstrate the idea on `RETINA`, a medical retinal image [57]. This dataset is one of the medical images of the STARE project[3], a retinal imaging data collection consisting of around 400 raw images of human retinas. We use the retinal image with ID being 179, which has numerous bright (circular) spots visible. We construct the cubical complex by intensity thresholding (also called the sub-level set method in TDA [128]) and morphological closing on the binary image to remove small cavities. The weight for every rectangle $\mathbf{w}_2(\sigma)$ is set to 1; the estimated null space dimension of the $\boldsymbol{\mathcal{L}}_1$ built from CB is $\beta_1 = 12$, with the identified homologous loops in Figure 6.3j. The result shows the robustness of the proposed framework even for large β_1.

6.4.6 Pairwise scatter plots

In this section, we show the pairwise scatter plots for $\mathbf{Z}$ (blue) and $\mathbf{Y}$ (red); specifically, we would like to show that the independent homology embedding $\mathbf{Z}$ obtained by Algorithm 6.1 is (approximately) factorizable. The blue embeddings (lower diagonal) in Figures 6.4–6.9 confirm this. By contrast, most coordinate of the red embeddings $\mathbf{Y}$ do not correspond to a subspace, except for `PANCREAS` and `3D-GRAPH` in Figures 6.8 and 6.6, respectively.

[3]The diagnosis codes, the segmented blood vessel, and the detected optic nerve are available in `http://cecas.clemson.edu/~ahoover/stare/`.

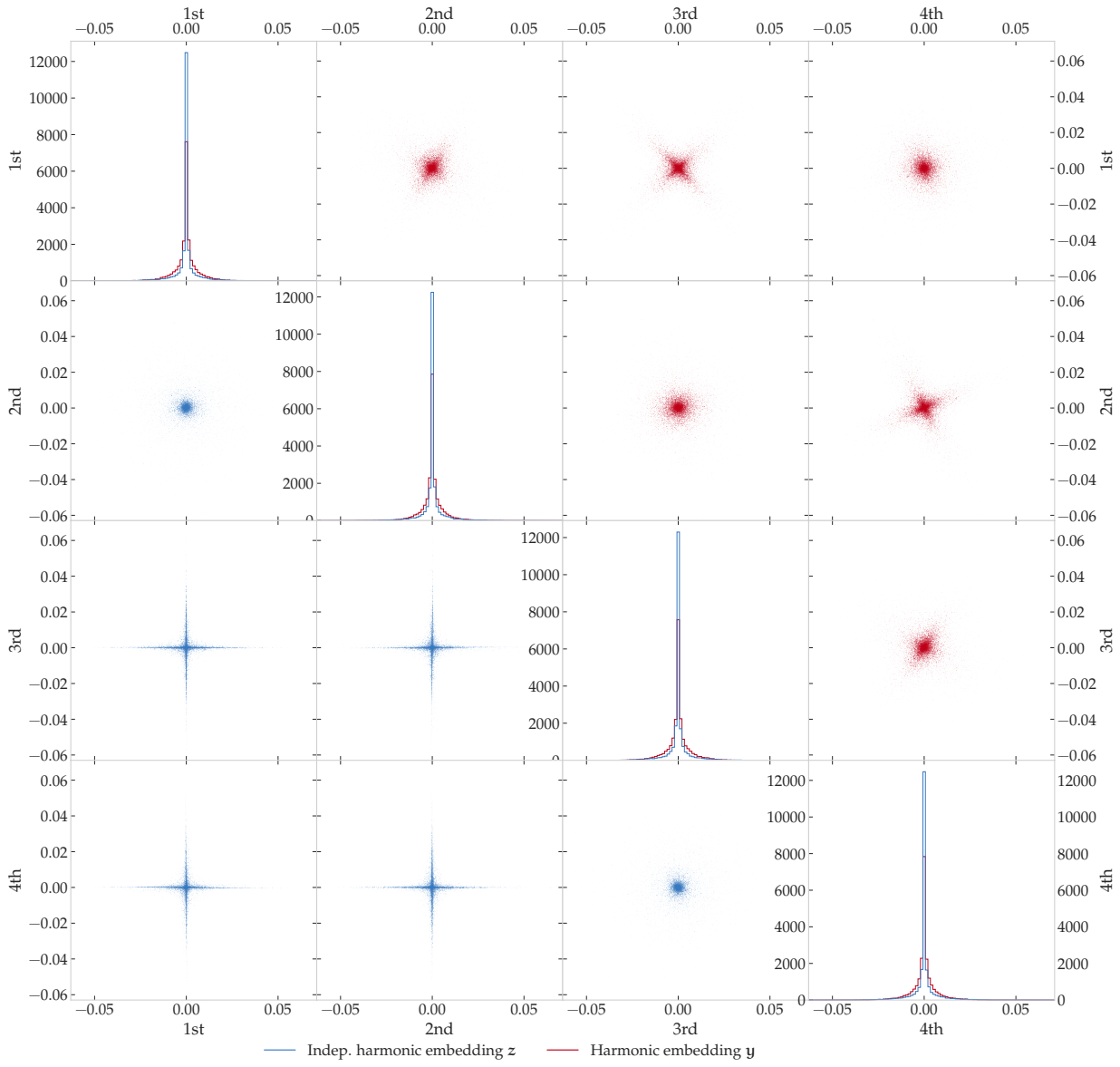

Figure 6.4: The independent ($\mathbf{z}$, in blue) and the coupled ($\mathbf{y}$, in red) homology embeddings of `GENUS-2`. The (i, j)-th (off-diagonal) subplot represents the two-dimensional scatter plot with the i-th and j-th coordinates of the embedding; the i-th diagonal term is the histograms of the i-th coordinate of the corresponding embedding.

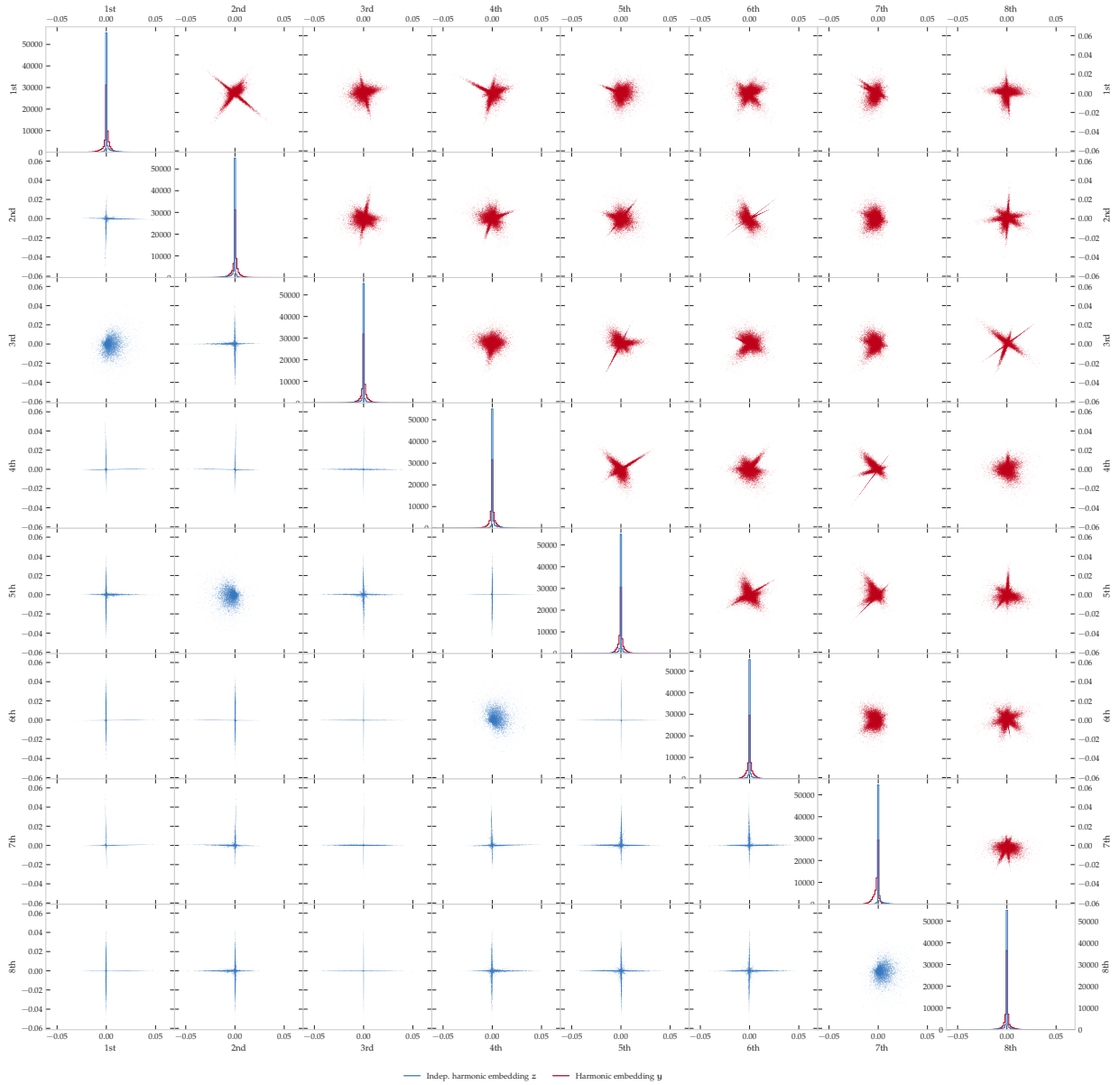

Figure 6.5: The independent ($\mathbf{z}$, in blue) and the coupled ($\mathbf{y}$, in red) homology embeddings of `TORI-CONCAT`.

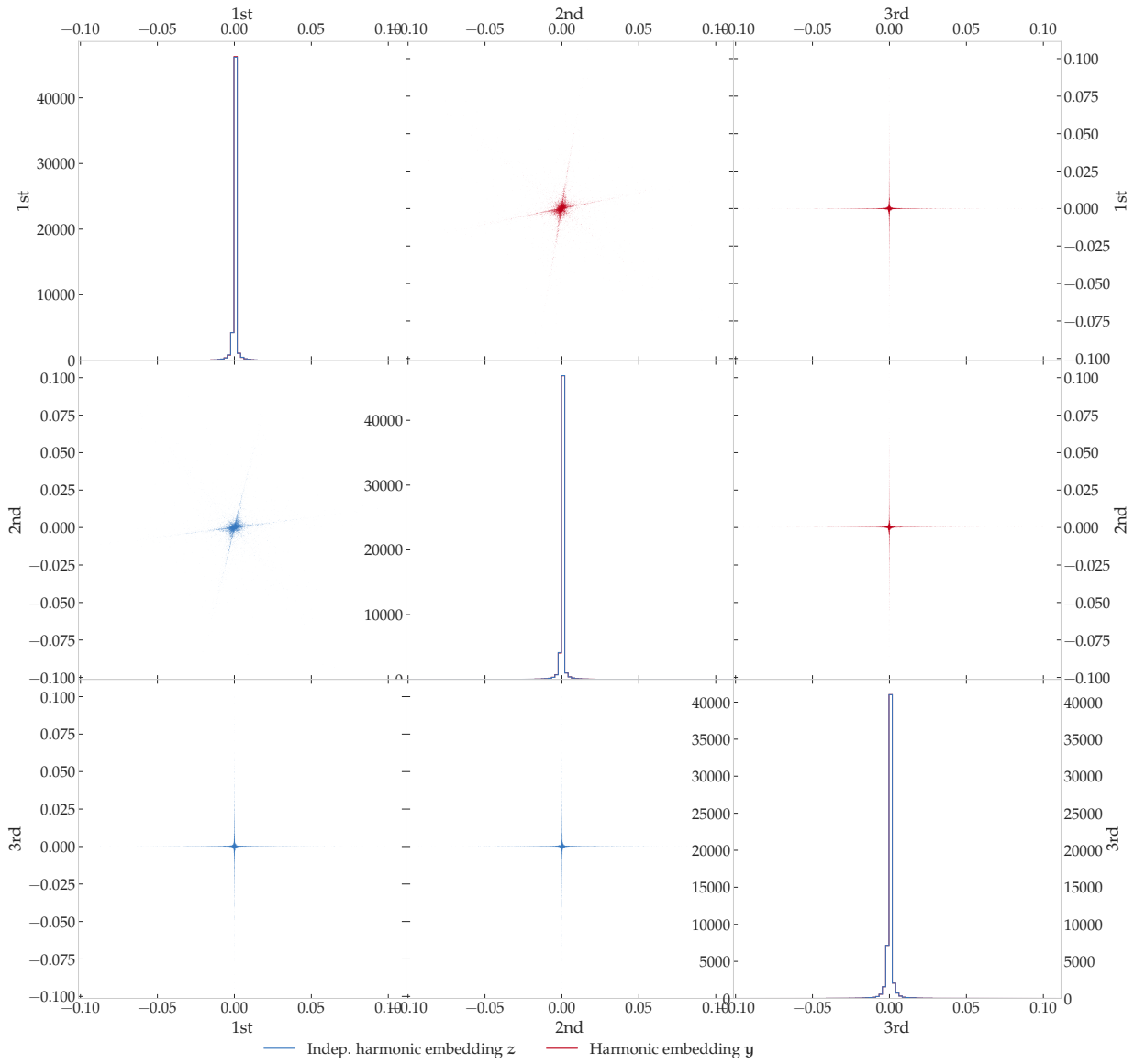

Figure 6.6: The independent (**z**, in blue) and the coupled (**y**, in red) homology embeddings of `3D-GRAPH`.

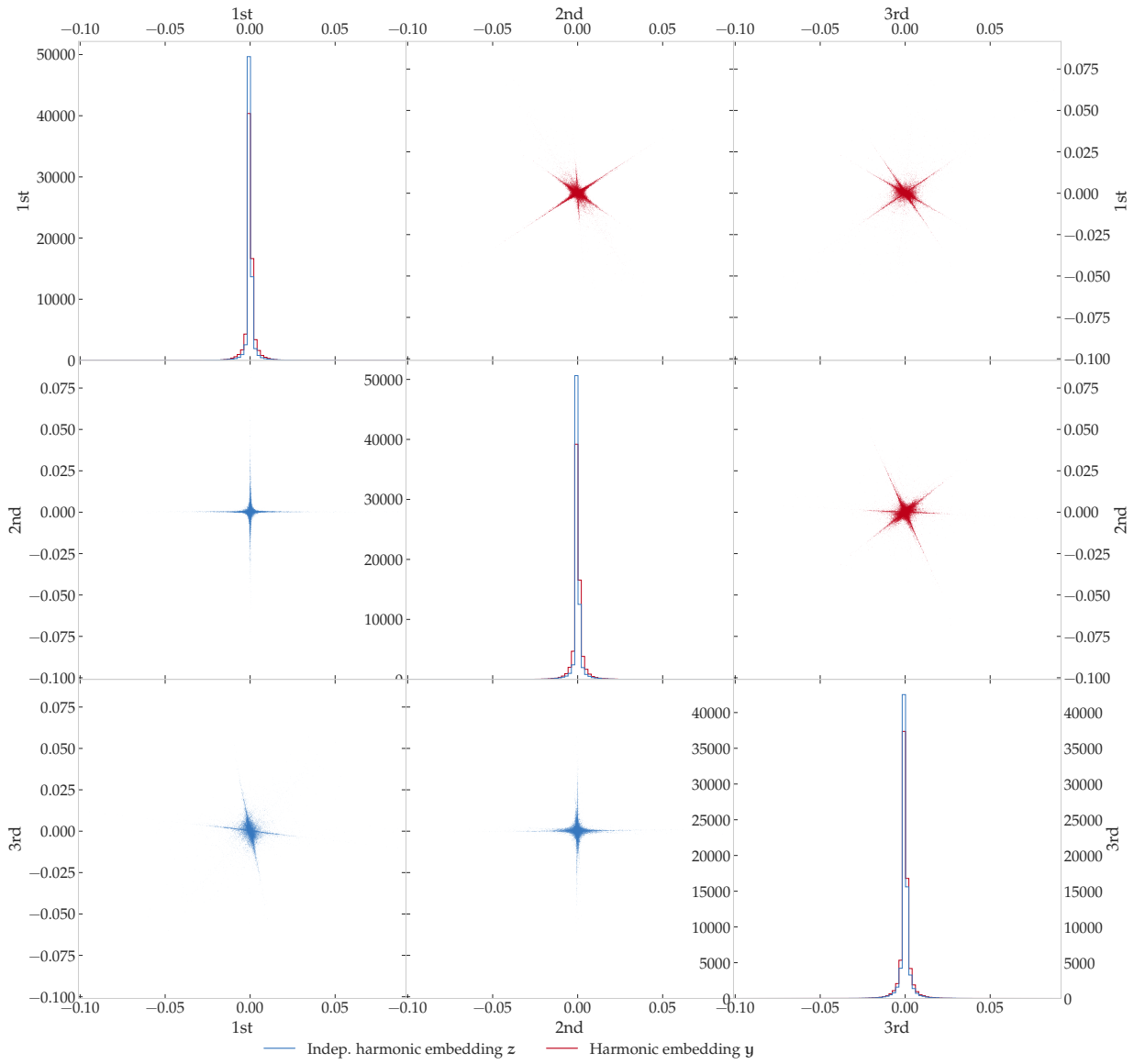

Figure 6.7: The independent ($\mathbf{z}$, in blue) and the coupled ($\mathbf{y}$, in red) homology embeddings of `SOUTH-ISLANDS`.

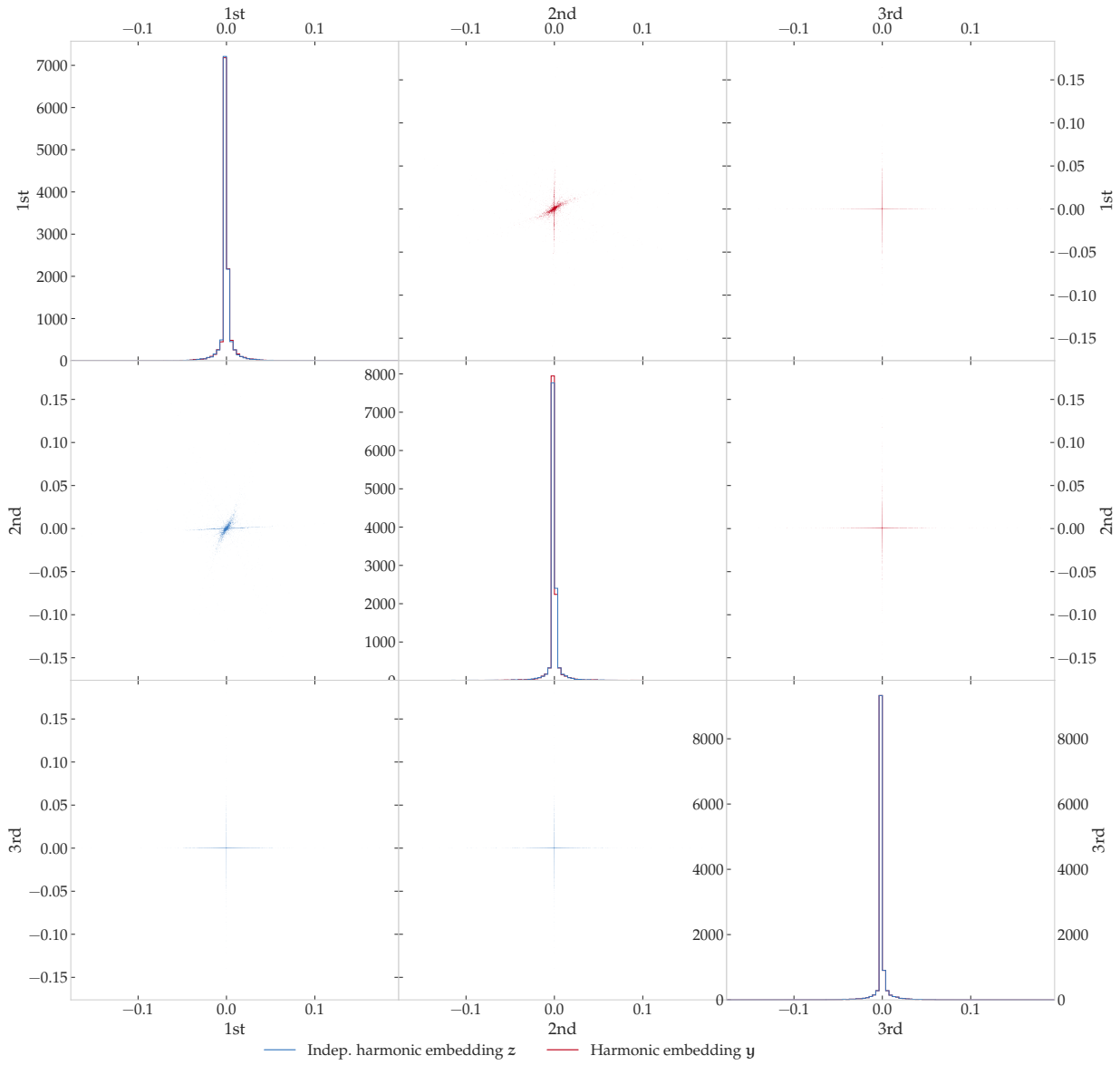

Figure 6.8: The independent ($\mathbf{z}$, in blue) and the coupled ($\mathbf{y}$, in red) homology embeddings of PANCREAS.

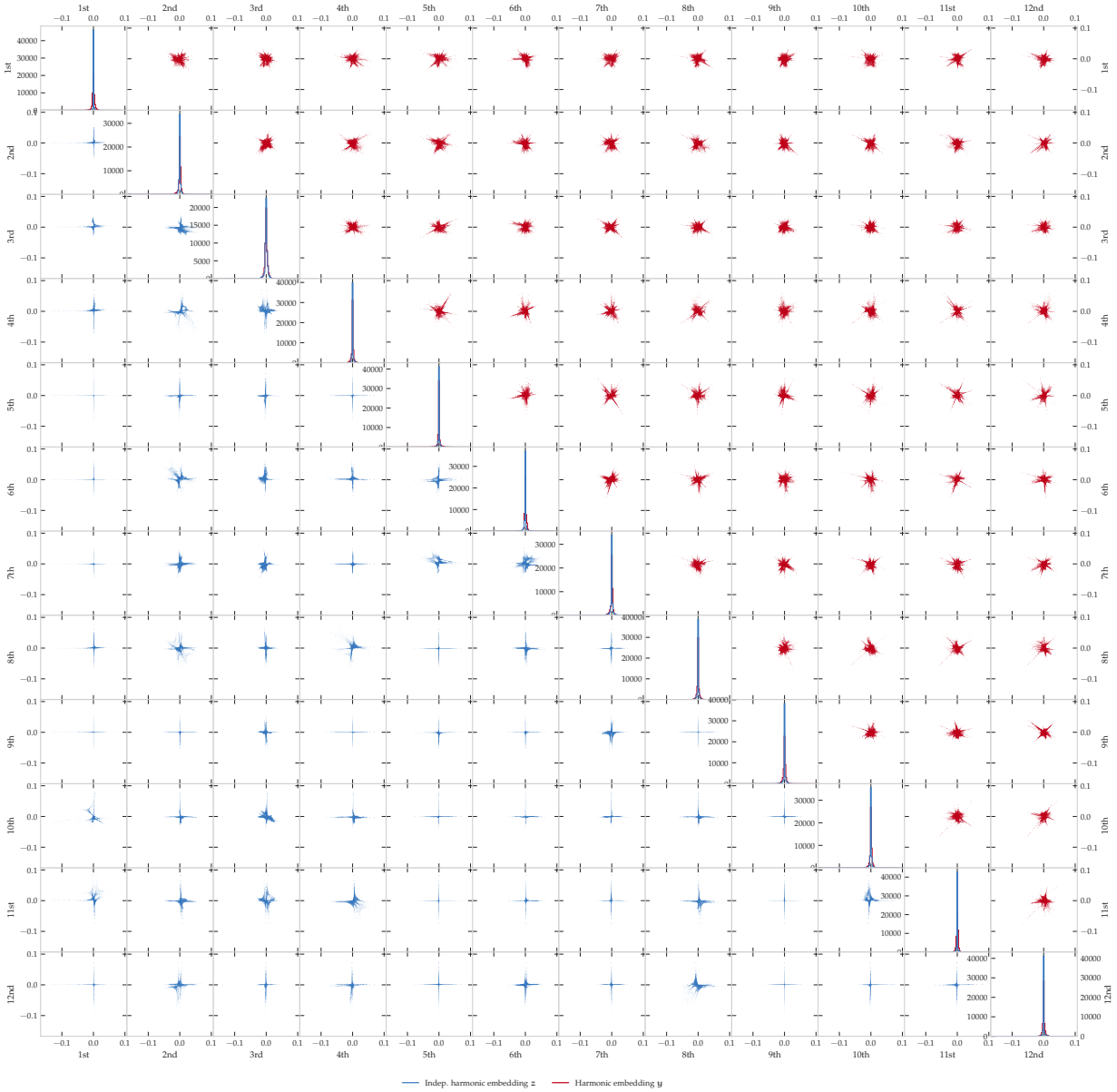

Figure 6.9: The independent ($\mathbf{z}$, in blue) and the coupled ($\mathbf{y}$, in red) homology embeddings of `RETINA`.

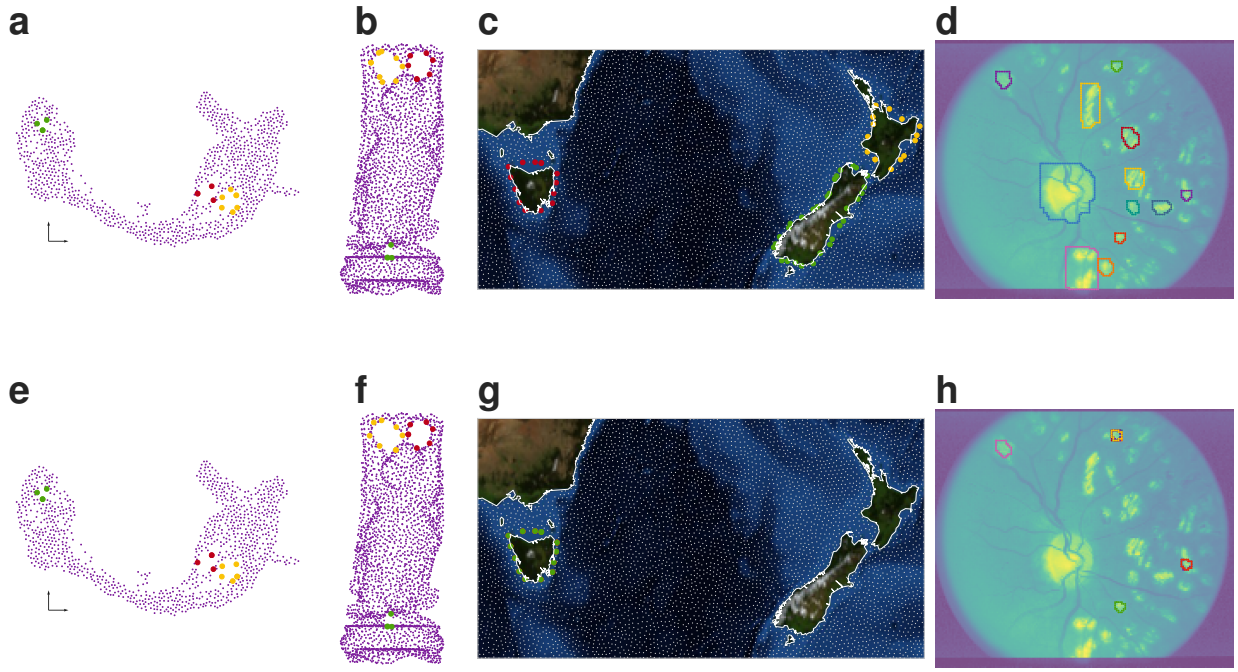

Figure 6.10: Comparison of the homologous loop detections on $\mathbf{Z}$ (the first row) and $\mathbf{Y}$ (the second row). The first, the second, the third, and the fourth columns present the results on `PANCREAS`, `3D-GRAPH`, `SOUTH-ISLANDS`, and `RETINA`, respectively. Note that (a)–(d) are identical to Figures 6.3g–6.3j.

6.4.7 Shortest homologous loops obtained from the coupled embedding

Figure 6.10 shows the results of the shortest homologous loop detection algorithm applied on the coupled homology embeddings $\mathbf{Y}$ on the real datasets. Note that Figures 6.10a–6.10d are identical to Figures 6.3g–6.3j; they are presented here as comparisons to the loops detected from $\mathbf{Y}$ (the second row). As shown in Figures 6.10g and 6.10h, duplicated loops might be extracted if using the coupled embedding $\mathbf{Y}$; these loops are clearly sub-optimal.

6.5 Summary

Our contributions in the emerging field of spectral algorithms for k-Laplacians $\boldsymbol{\mathcal{L}}_k$ [28, 45, 73, 105] are summarized as follows. (i) We extend the study of the homology embedding of vertices by the graph Laplacian $\boldsymbol{\mathcal{L}}_0$ (spectral clustering) to those of higher-order simplices by $\boldsymbol{\mathcal{L}}_k$. Specifically, the k-th homology embedding can be approximately factorized into parts, with each corresponding to a prime manifold given a small perturbation (small ϵ_ℓ and ϵ'_ℓ for $\ell = k, k-1$). (ii) The analysis is made possible by expressing the κ-fold *connected sum* as a matrix perturbation. This convenient property of the homology embedding supports (iii) the use of ICA to identify each decoupled subspace and motivates (iv) the application to the shortest homologous loop detection problem.

Our analysis provides insight into the structure of the k-th harmonic embedding. This framework can inspire researchers in developing *spectral* topological data analysis algorithms (e.g., visualization, clustering, tightest higher-order *cycles* for $k \geq 2$ [39, 89]) similar to those that were inaugurated by spectral clustering two decades ago. These applications are especially beneficial to scientists (chemists, biologists, oceanographers, etc.) who use high-dimensional data analysis techniques for studying complex systems. Similar to the limitation of other unsupervised learning algorithms, practitioners without solid understandings of *both* the analyzed datasets and the used algorithm might draw controversial conclusions (see, e.g., discussions in [3, 88]). Possible approaches to mitigate the negative consequences are to design proper validation and causal inference algorithms for this framework; we leave them as potential directions we will explore.

6.6 Appendix—proofs of Theorem 6.1 and Corollary 6.2

6.6.1 A formal version of Assumption 6.3

Assumption 6.4. *Let* $\tilde{\mathbf{w}}_k = |\mathbf{B}_{k+1}[\mathfrak{N}_k, \mathfrak{N}_{k+1}]|\mathbf{w}_{k+1}$, $\tilde{\mathbf{w}}_{k-1} = |\mathbf{B}_k[:, \mathfrak{N}_k]|\tilde{\mathbf{w}}_k$, *with* $\mathbf{w}_k$ *and* $\hat{\mathbf{w}}_k$ *defined in Section 6.1. Additionally, write*

$$\begin{aligned}
\mathbf{W}_{k+1} &= \tilde{\mathbf{W}}_{k+1} + \boldsymbol{\mathcal{E}}_{k+1,+},\\
\hat{\mathbf{W}}_{k+1} &= \tilde{\mathbf{W}}_{k+1} + \boldsymbol{\mathcal{E}}_{k+1,-},\\
\mathbf{W}_k^{1/2} &= \tilde{\mathbf{W}}_k^{1/2}(\mathbf{I} + \mathbf{E}_{k,+}^{+}) + \boldsymbol{\mathcal{E}}_{k,+}^{1/2},\\
\mathbf{W}_k^{-1/2} &= \tilde{\mathbf{W}}_k^{-1/2}(\mathbf{I} - \mathbf{E}_{k,+}^{-}) + \boldsymbol{\mathcal{E}}_{k,+}^{-1/2},\\
\hat{\mathbf{W}}_k^{1/2} &= \tilde{\mathbf{W}}_k^{1/2}(\mathbf{I} + \mathbf{E}_{k,-}^{+}) + \boldsymbol{\mathcal{E}}_{k,-}^{1/2},\\
\hat{\mathbf{W}}_k^{-1/2} &= \tilde{\mathbf{W}}_k^{-1/2}(\mathbf{I} - \mathbf{E}_{k,-}^{-}) + \boldsymbol{\mathcal{E}}_{k,-}^{-1/2},\\
\mathbf{W}_{k-1}^{-1/2} &= \tilde{\mathbf{W}}_{k-1}^{-1/2}(\mathbf{I} - \mathbf{E}_{k-1,+}),\\
\hat{\mathbf{W}}_{k-1}^{-1/2} &= \tilde{\mathbf{W}}_{k-1}^{-1/2}(\mathbf{I} - \mathbf{E}_{k-1,-}).
\end{aligned}$$

There exists $\epsilon_\ell > 0$ *and* $\epsilon'_\ell > 0$ *for* $\ell = k, k-1$ *such that the following conditions hold*

1. *Not too many* $(k+1)$*-simplices are created (small* $|\mathfrak{C}_{k+1}|$*)*

$$\|\mathbf{E}_{k,+}^{+}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left[\mathbf{E}_{k,+}^{+}\right]_{\sigma,\sigma}\right\} = \max_{\sigma\in\mathfrak{N}_k}\left\{\frac{w_k^{1/2}(\sigma)}{\tilde{w}_k^{1/2}(\sigma)} - 1\right\} \le \sqrt{\epsilon_k}; \tag{6.2a}$$

$$\|\mathbf{E}_{k,+}^{-}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left[\mathbf{E}_{k,+}^{-}\right]_{\sigma,\sigma}\right\} = \max_{\sigma\in\mathfrak{N}_k}\left\{\frac{\tilde{w}_k^{-1/2}(\sigma)}{w_k^{-1/2}(\sigma)} - 1\right\} \le \sqrt{\epsilon_k}; \tag{6.2b}$$

$$\max_{\sigma\in\mathfrak{N}_k}\left\{\frac{w_k(\sigma)}{\tilde{w}_k(\sigma)} - 1\right\} \le \epsilon_k; \tag{6.2c}$$

$$\|\mathbf{E}_{k-1,+}\| = \max_{\nu\in\Sigma_{k-1}}\left\{\left[\mathbf{E}_{k-1,+}\right]_{\nu,\nu}\right\} = \max_{\nu\in\Sigma_{k-1}}\left\{\frac{\tilde{w}_{k-1}^{-1}(\nu)}{w_{k-1}^{-1}(\nu)} - 1\right\} \le \sqrt{\epsilon_{k-1}}; \tag{6.2d}$$

$$\max_{\nu\in\Sigma_{k-1}}\left\{\frac{w_{k-1}(\nu)}{\tilde{w}_{k-1}(\nu)} - 1\right\} \le \epsilon_{k-1}. \tag{6.2e}$$

2. Not too many $(k+1)$-simplices are destroyed (small $|\mathfrak{D}_{k+1}|$)

$$\|\mathbf{E}^{+}_{k,-}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left[\mathbf{E}^{+}_{k,-}\right]_{\sigma,\sigma}\right\} = \max_{\sigma\in\mathfrak{N}_k}\left\{\frac{\hat{w}_k^{1/2}(\sigma)}{\tilde{w}_k^{1/2}(\sigma)}-1\right\}\leq\sqrt{\epsilon_k}; \tag{6.3a}$$

$$\|\mathbf{E}^{-}_{k,-}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left[\mathbf{E}^{-}_{k,-}\right]_{\sigma,\sigma}\right\} = \max_{\sigma\in\mathfrak{N}_k}\left\{\frac{\tilde{w}_k^{-1/2}(\sigma)}{\hat{w}_k^{-1/2}(\sigma)}-1\right\}\leq\sqrt{\epsilon_k}; \tag{6.3b}$$

$$\max_{\sigma\in\mathfrak{N}_k}\left\{\frac{\hat{w}_k(\sigma)}{\tilde{w}_k(\sigma)}-1\right\}\leq\epsilon_k; \tag{6.3c}$$

$$\|\mathbf{E}_{k-1,-}\| = \max_{\nu\in\Sigma_{k-1}}\left\{\left[\mathbf{E}_{k-1,-}\right]_{\nu,\nu}\right\} = \max_{\nu\in\Sigma_{k-1}}\left\{\frac{\tilde{w}_{k-1}^{-1}(\nu)}{\hat{w}_{k-1}^{-1}(\nu)}-1\right\}\leq\sqrt{\epsilon_{k-1}}; \tag{6.3d}$$

$$\max_{\nu\in\Sigma_{k-1}}\left\{\frac{\hat{w}_{k-1}(\nu)}{\tilde{w}_{k-1}(\nu)}-1\right\}\leq\epsilon_{k-1}. \tag{6.3e}$$

3. The net *changes on $\mathbf{w}_k$ and $\mathbf{w}_{k-1}$ are small*

$$\|\mathbf{E}^{+}_{k,+}-\mathbf{E}^{+}_{k,-}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left|\frac{\hat{w}_k^{1/2}(\sigma)}{w_k^{1/2}(\sigma)}-1\right|\right\}\leq\sqrt{\epsilon'_k}; \tag{6.4a}$$

$$\|\mathbf{E}^{-}_{k,+}-\mathbf{E}^{-}_{k,-}\| = \max_{\sigma\in\mathfrak{N}_k}\left\{\left|\frac{w_k^{1/2}(\sigma)}{\hat{w}_k^{1/2}(\sigma)}-1\right|\right\}\leq\sqrt{\epsilon'_k}; \tag{6.4b}$$

$$\max_{\sigma\in\mathfrak{N}_k}\left\{\left|\frac{\hat{w}_k(\sigma)}{w_k(\sigma)}-1\right|\right\}\leq\epsilon'_k; \tag{6.4c}$$

$$\|\mathbf{E}_{k-1,+}-\mathbf{E}_{k-1,-}\| = \max_{\nu\in\Sigma_{k-1}}\left\{\left|\frac{w_{k-1}^{-1}(\nu)}{\hat{w}_{k-1}^{-1}(\nu)}-1\right|\right\}\leq\sqrt{\epsilon'_{k-1}}; \tag{6.4d}$$

$$\max_{\nu\in\Sigma_{k-1}}\left\{\left|\frac{\hat{w}_{k-1}(\nu)}{w_{k-1}(\nu)}-1\right|\right\}\leq\epsilon'_{k-1}. \tag{6.4e}$$

6.6.2 Definitions of $\mathcal{L}_k$ and $\hat{\mathcal{L}}_k$

Given a manifold $\mathcal{M}$ which is constructed by a series of connected sum, i.e., $\mathcal{M}=\mathcal{M}_1\sharp\cdots\sharp\mathcal{M}_\kappa$. Let the Simplicial complex corresponding to $\mathcal{M}$ be $\mathrm{SC}_\ell=(\Sigma_0,\cdots,\Sigma_\ell)$, with the disjoint simplicial complex (of $\cup_{i=1}^{\kappa}\mathcal{M}_i$) being $\hat{\mathrm{SC}}_\ell=(\hat{\Sigma}_0,\cdots,\hat{\Sigma}_\ell)$. For each k, the simplex sets can be decomposed into the following

$$\Sigma_k = \underbrace{\bigcup_{i=1}^{\kappa} \Sigma_k^{(i)}}_{\text{non-intersecting set:}\mathfrak{N}_k} \cup \underbrace{\bigcup_{j>i}^{\kappa} \Sigma_k^{(ij)+}}_{\text{created set:}\mathfrak{C}_k} .$$

Similarly,

$$\hat{\Sigma}_k = \underbrace{\bigcup_{i=1}^{\kappa} \Sigma_k^{(i)}}_{\text{non-intersecting set:}\mathfrak{N}_k} \cup \underbrace{\bigcup_{j>i}^{\kappa} \Sigma_k^{(ij)-}}_{\text{destroyed set:}\mathfrak{D}_k} .$$

W.l.o.g., one can assume that the $(k-1)$-simplices set can be perfectly separated, i.e., $\mathfrak{C}_{k-1} = \mathfrak{D}_{k-1} = \emptyset$ (when analyzing the k-Laplacian). The above construction matches our intuition; by definition, a connected sum is a process of carving out a d-disk ($\mathfrak{D}_k$) and gluing two manifolds together ($\mathfrak{C}_k$).

We are interested in the perturbation of the k-Laplacian $\boldsymbol{\mathcal{L}}_k$ w.r.t. the ideal (disjoint) Laplacian $\hat{\boldsymbol{\mathcal{L}}}_k$. Without carefully define both $\boldsymbol{\mathcal{L}}_k$ and $\hat{\boldsymbol{\mathcal{L}}}_k$, the perturbation on the subspaces might be unbounded. With slight abuse of notation, we let $\mathbf{L} \leftarrow \boldsymbol{\mathcal{L}}_k$, $\mathbf{L}_d \leftarrow \boldsymbol{\mathcal{L}}_k^{\text{down}}$, and $\mathbf{L}_u \leftarrow \boldsymbol{\mathcal{L}}_k^{\text{up}}$ (similar definitions for $\hat{\boldsymbol{\mathcal{L}}}$'s). The k is omitted and can be inferred from the context. The $\hat{\mathbf{L}}$ and $\mathbf{L}$ are defined as follows. $\hat{\mathbf{L}}$ is a block diagonal matrix, with the i-th (diagonal) block $\mathbf{L}^{(i)}$ described by $\mathcal{M}_i$ constructed from the sub-complex $\hat{\text{SC}}^{(i)}$ ($\hat{\Sigma}_{k-1}^{(i)}$, $\hat{\Sigma}_k^{(i)}$, and $\hat{\Sigma}_{k+1}^{(i)}$). Due to manifolds being disjoint (i.e., $\cup_{i=1}^{\kappa} \mathcal{M}_i$), the Laplacian corresponding to such block, denoted $\hat{\mathbf{L}}^{(i,i),(i,i)}$, will be a valid Laplacian. As for the intersecting k-simplices $\mathfrak{C}_k \cup \mathfrak{D}_k$, we let $\hat{\mathbf{L}}^{(i,j),(k,l)} = \mathbf{L}^{(i,j),(k,l)}$ for all $ij, kl \in \binom{[k]}{2}$ so that the corresponding blocks of $\hat{\mathbf{L}} - \mathbf{L}$ will be zero. Under this scenario, the unbounded increase of $(k+1)$-simplices caused by the intersecting k-simplices can be removed. Lastly, the off-diagonal blocks of $\hat{\mathbf{L}}$ are set to zero. Specifically, $\hat{\mathbf{L}}$ is,

$$\hat{\mathbf{L}} = \left[\begin{array}{ccc|ccc|ccc} \hat{\mathbf{L}}^{(1,1),(1,1)} & & & & & & \hat{\mathbf{L}}^{(1,1),(1,2)-} & \cdots & \hat{\mathbf{L}}^{(1,1),(k-1,k)-} \\ & \ddots & & & \mathbf{0} & & \vdots & \ddots & \vdots \\ & & \hat{\mathbf{L}}^{(k,k),(k,k)} & & & & \hat{\mathbf{L}}^{(k,k),(1,2)-} & \cdots & \hat{\mathbf{L}}^{(k,k),(k-1,k)-} \\ \hline & & & \mathbf{L}^{(1,2)+,(1,2)+} & \cdots & \mathbf{L}^{(1,2)+,(k-1,k)+} & & & \\ & \mathbf{0} & & \vdots & \ddots & \vdots & & \mathbf{0} & \\ & & & \mathbf{L}^{(k-1,k)+,(1,2)+} & \cdots & \mathbf{L}^{(k-1,k)+,(k-1,k)+} & & & \\ \hline \hat{\mathbf{L}}^{(1,2)-,(1,1)} & \cdots & \hat{\mathbf{L}}^{(1,2)-,(k,k)} & & & & \hat{\mathbf{L}}^{(1,2)-,(1,2)-} & \cdots & \hat{\mathbf{L}}^{(1,2)-,(k-1,k)-} \\ \vdots & \ddots & \vdots & & \mathbf{0} & & \vdots & \ddots & \vdots \\ \hat{\mathbf{L}}^{(k-1,k)-,(1,1)} & \cdots & \hat{\mathbf{L}}^{(k,k)-,(k,k)} & & & & \hat{\mathbf{L}}^{(k-1,k)-,(1,2)-} & \cdots & \hat{\mathbf{L}}^{(k-1,k)-,(k-1,k)-} \end{array}\right].$$

Similarly, one can define $\mathbf{L}$ to be

$$\mathbf{L} = \left[\begin{array}{ccc|ccc|ccc} \mathbf{L}^{(1,1),(1,1)} & & & \mathbf{L}^{(1,1),(1,2)+} & \cdots & \mathbf{L}^{(1,1),(k-1,k)+} & & & \\ & \ddots & & \vdots & \ddots & \vdots & & \mathbf{0} & \\ & & \mathbf{L}^{(k,k),(k,k)} & \mathbf{L}^{(k,k),(1,2)+} & \cdots & \mathbf{L}^{(k,k),(k-1,k)+} & & & \\ \hline \mathbf{L}^{(1,2)+,(1,1)} & \cdots & \mathbf{L}^{(1,2)+,(k,k)} & \mathbf{L}^{(1,2)+,(1,2)+} & \cdots & \mathbf{L}^{(1,2)+,(k-1,k)+} & & & \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & & \mathbf{0} & \\ \mathbf{L}^{(k-1,k)+,(1,1)} & \cdots & \mathbf{L}^{(k,k)+,(k,k)} & \mathbf{L}^{(k-1,k)+,(1,2)+} & \cdots & \mathbf{L}^{(k-1,k)+,(k-1,k)+} & & & \\ \hline & & & & & & \hat{\mathbf{L}}^{(1,2)-,(1,2)-} & \cdots & \hat{\mathbf{L}}^{(1,2)-,(k-1,k)-} \\ & \mathbf{0} & & & \mathbf{0} & & \vdots & \ddots & \vdots \\ & & & & & & \hat{\mathbf{L}}^{(k-1,k)-,(1,2)-} & \cdots & \hat{\mathbf{L}}^{(k-1,k)-,(k-1,k)-} \end{array}\right].$$

Under this construction, the four lower right blocks, which correspond to the k-simplices in $\mathfrak{C}_k \cup \mathfrak{D}_k$, will be zero. If no new homology class is created/destroyed (Assumption 6.1) and the minimum eigenvalues of the last two diagonal blocks are bounded away from zero (Assumption 6.2), then the eigengap of $\mathbf{L}$ will simply be the minimum eigengap of each $\hat{\mathbf{L}}^{(i)}$, i.e., $\text{eigengap}(\mathbf{L}) = \min\{\delta_1, \cdots, \delta_\kappa\}$.

Now we formally define our formulation. Following the notations introduced in Section 6.1, and let $\mathcal{I}_\sigma$ be the index set of the k-simplex $\sigma \in \mathfrak{N}_k$ sampled from $\mathcal{M}_i$. Note that $\mathcal{I}_\sigma$ is

defined only for $\sigma \in \mathfrak{N}_k$, which can be extended from the index set $\mathcal{I}_v$ for $v \in V$ introduced in Section 6.1 by $\mathcal{I}_\sigma = \{\sigma \in \mathfrak{N}_k : v \in \mathcal{I}_v \text{ for } v \in \sigma\}$. Note also that similar to $\mathcal{I}_v$ for V, $\mathcal{S}_\sigma$ can be larger than 1. For instance, if the manifold is constructed by gluing a torus (indexed by 1) and a circle (indexed by 2), then $\mathcal{S}_1 = \{1, 2\}$ and $\mathcal{S}_2 = \{3\}$; for an edge e belongs to the torus, we have $\mathcal{S}_{\mathcal{I}_e} = \{1, 2\}$. For every $\sigma \in \mathfrak{N}_k$, we write,

$$\sum_{\sigma\in\mathfrak{N}_k}\sum_{i\notin S_{\mathcal{I}_\sigma}} \mathbf{Y}_{\sigma,i}^2 \le \sum_{\sigma\in\mathfrak{N}_k}\sum_{i=1}^{\beta_1}(\mathbf{Y}_{\sigma,i} - \hat{\mathbf{Y}}_{\sigma,i})^2 \le \sum_{\sigma\in\Sigma_k\cup\hat{\Sigma}_k}\sum_{i=1}^{\beta_1}(\mathbf{Y}_{\sigma,i} - \hat{\mathbf{Y}}_{\sigma,i})^2 = \|\mathbf{YO} - \hat{\mathbf{Y}}\|_F^2.$$

Let $\mathsf{DiffL}_k^{\text{down}} = \mathbf{L}_d - \hat{\mathbf{L}}_d$ and $\mathsf{DiffL}_k^{\text{up}} = \mathbf{L}_u - \hat{\mathbf{L}}_u$, from [130] and the triangular inequality,

$$\begin{aligned}
\left\|\mathbf{Y}_{\mathfrak{N}_k,:} - \hat{\mathbf{Y}}_{\mathfrak{N}_k,:}\right\|_F^2 &= \sum_{\sigma\in\mathfrak{N}_k}\sum_{i\notin\mathcal{S}_{\mathcal{I}_\sigma}} \mathbf{Y}_{\sigma,i}^2 \le \|\mathbf{Y} - \hat{\mathbf{Y}}\mathbf{O}\|_F^2 \\
&\le \frac{8\cdot\min\left\{\beta_k\left\|\mathbf{L} - \hat{\mathbf{L}}\right\|^2, \|\mathbf{L} - \hat{\mathbf{L}}\|_F^2\right\}}{\min\{\delta_1, \cdots, \delta_\kappa\}} \\
&\le \frac{8\cdot\min\left\{\beta_k\left\|\mathsf{DiffL}_k^{\text{down}}\right\|^2 + \beta_k\left\|\mathsf{DiffL}_k^{\text{up}}\right\|^2, \|\mathsf{DiffL}_k^{\text{down}}\|_F^2 + \|\mathsf{DiffL}_k^{\text{up}}\|_F^2\right\}}{\min\{\delta_1, \cdots, \delta_\kappa\}} \\
&\overset{\dagger}{\le} \frac{8\beta_k\left(\left\|\mathsf{DiffL}_k^{\text{down}}\right\|^2 + \left\|\mathsf{DiffL}_k^{\text{up}}\right\|^2\right)}{\min\{\delta_1, \cdots, \delta_\kappa\}}.
\end{aligned}$$

Remark. The bound w.r.t. the Frobenius norm is omitted (the last inequality †) based on two reasons: (i) $\boldsymbol{\mathcal{L}}_k$ has complicated forms for large k, therefore, it is hard to derive a concise expression; and (ii) $\|\cdot\|_F$ is usually larger than $\beta_k\|\cdot\|$.

6.6.3 Useful lemmas

Here we omit the k for $\mathfrak{N}$, $\mathfrak{C}$, and $\mathfrak{D}$ for simplicity. Let $\lambda_k = \|\boldsymbol{\mathcal{L}}_k\|$ be the bound on the spectral norm of k-Laplacian. Here, $\lambda_k = k + 2$ for $\boldsymbol{\mathcal{L}}$'s built from simplicial complexes; $\lambda_k = 2k + 2$ for those built from cubical complexes (see also Proposition 6.7). The following two lemmas bound the effects of $\boldsymbol{\mathcal{E}}_{k,+}$, $\boldsymbol{\mathcal{E}}_{k,-}$, $\boldsymbol{\mathcal{E}}_{k+1,+}$, and $\boldsymbol{\mathcal{E}}_{k+1,-}$ in their changes to the

weights ($\mathbf{W}_k$ and $\mathbf{W}_{k-1}$) of the k and $(k-1)$-simplices; we will find them useful in proving Theorem 6.1.

Lemma 6.5. *Let $\mathbf{W}_k$, $\hat{\mathbf{W}}_k$, $\boldsymbol{\mathcal{E}}_{k,+}$, and $\boldsymbol{\mathcal{E}}_{k,-}$ defined in Assumption 6.4, we have*

$$\left\|\boldsymbol{\mathcal{E}}_{k,+}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}\right\| \leq \lambda_{k-1}\epsilon_{k-1},$$
$$\left\|\boldsymbol{\mathcal{E}}_{k,-}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,-}\right\| \leq \lambda_{k-1}\epsilon_{k-1}.$$

Proof. We first inspect the case of $\mathfrak{C}$, i.e., the first equation involving $\boldsymbol{\mathcal{E}}_{k,+}$,

$$[\boldsymbol{\mathcal{E}}_{k,+}]_{\sigma,\sigma} = \begin{cases} w_k^{1/2}(\sigma) & \text{if } \sigma \in \mathfrak{C}; \\ 0 & \text{otherwise.} \end{cases}$$

for any $\nu \in \Sigma_{k-1}$, we have,

$$\begin{aligned} w_{k-1}(\nu) &= |\mathbf{B}_k(\nu)|\mathbf{w}_k; \\ \tilde{w}_{k-1}(\nu) &= |\mathbf{B}_k(\nu)|\tilde{\mathbf{w}}_k. \end{aligned}$$

Therefore,

$$\begin{aligned} \epsilon_{k-1}w_{k-1}(\nu) &\geq \epsilon_{k-1}\tilde{w}_{k-1}(\nu) \geq w_{k-1}(\nu) - \tilde{w}_{k-1}(\nu) = |\mathbf{B}_k(\nu)|(\mathbf{w}_k - \tilde{\mathbf{w}}_k) \\ &= |\mathbf{B}_k(\nu)|\left[\tilde{\mathbf{w}}_k\mathbf{E}_k + \boldsymbol{\mathcal{E}}_{k,+}\right] \geq |\mathbf{B}_k(\nu)|\boldsymbol{\mathcal{E}}_{k,+} = \deg(\nu). \end{aligned}$$

Let f_m be the k-eigencochain corresponding to the largest eigenvalue of $\boldsymbol{\mathcal{E}}_{k,+}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}$. From Eq. (3.6) of [58], we have,

$$\begin{aligned}\|\boldsymbol{\mathcal{E}}_{k,+}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}\|_2 &\le \|\mathbf{L}_d\|\cdot\frac{\sum_{\nu\in\Sigma_{k-1}}f_m^2(\nu)\deg(\nu)}{\sum_{\nu\in\Sigma_{k-1}}f_m^2(\nu)w_{k-1}(\nu)}\\ &\le \lambda_{k-1}\epsilon_{k-1}\cdot\frac{\sum_{\nu\in\Sigma_{k-1}}f_m^2(\nu)w_{k-1}(\nu)}{\sum_{\nu\in\Sigma_{k-1}}f_m^2(\nu)w_{k-1}(\nu)}=\lambda_{k-1}\epsilon_{k-1}.\end{aligned}$$

The case of $\mathfrak{D}$ follows similarly. ■

The following lemma bounds the changes in $(k+1)$-simplices with ϵ_k.

Lemma 6.6. *Let* $\mathbf{W}$ *be either* $\mathbf{W}_k$ *or* $\hat{\mathbf{W}}_k$*, and* $\boldsymbol{\mathcal{E}}$ *be either* $\boldsymbol{\mathcal{E}}_{k+1,+}$ *or* $\boldsymbol{\mathcal{E}}_{k+1,-}$ *defined in Assumption 6.4, we have*

$$\left\|\mathbf{W}\mathbf{B}_{k+1}\boldsymbol{\mathcal{E}}\mathbf{B}_{k+1}^\top\mathbf{W}\right\|\le\lambda_k\epsilon_k.$$

Proof. Consider the case of $\mathbf{W}_k$ and $\boldsymbol{\mathcal{E}}_{k+1,+}$. For any $\sigma\in\Sigma_k$,

$$\begin{aligned}w_k(\sigma) &= |\mathbf{B}_{k+1}(\sigma)|\mathbf{w}_{k+1};\\ \tilde{w}_k(\sigma) &= |\mathbf{B}_{k+1}(\sigma)|\tilde{\mathbf{w}}_{k+1}.\end{aligned}$$

Therefore, for any $\sigma\in\mathfrak{N}$ (do not count the one in $\boldsymbol{\mathcal{E}}_{k,\pm}$) we have,

$$\epsilon_k w_k(\sigma)\ge\epsilon_k\tilde{w}_k(\sigma)\ge w_k(\sigma)-\tilde{w}(\sigma)=|\mathbf{B}_{k+1}(\sigma)|(\mathbf{w}_{k+1}-\tilde{\mathbf{w}}_{k+1})=|\mathbf{B}_{k+1}(\sigma)|\boldsymbol{\mathcal{E}}_{k+1,+}.$$

Let f_m be the k-eigencochain corresponding to the largest eigenvalue of the matrix $\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\boldsymbol{\mathcal{E}}_{k+1,+}\mathbf{B}_{k+1}^\top\mathbf{W}_k^{-1/2}$. From Eq. (3.6) of Horak and Jost [58],

$$\begin{aligned}\left\|\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\boldsymbol{\mathcal{E}}_{k+1,+}\mathbf{B}_{k+1}^\top\mathbf{W}_k^{-1/2}\right\| &\le (k+2)\cdot\frac{\sum_{\sigma\in\mathfrak{N}}f_m^2(\sigma)\deg(\sigma)}{\sum_{\sigma\in\mathfrak{N}}f_m^2(\sigma)w_k(\sigma)}\\ &\le \lambda_k\epsilon_k\frac{\sum_{\sigma\in\mathfrak{N}}f_m^2(e)w_k(e)}{\sum_{\sigma\in\mathfrak{N}}f_m^2(e)w_k(e)}=\lambda_k\epsilon_k\end{aligned}$$

Here $\deg(\sigma) = |\mathbf{B}_{k+1}(\sigma)\,\mathrm{diag}(\boldsymbol{\mathcal{E}}_{k+1,+})|$. Consider the case when $\mathbf{W} \leftarrow \hat{\mathbf{W}}_k$ and $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{E}}_{k+1,+}$, we have,

$$\epsilon_k \hat{w}_k(\sigma) \geq \epsilon_k \tilde{w}_k(\sigma) \geq w_k(\sigma) - \tilde{w}(\sigma) = |\mathbf{B}_{k+1}(\sigma)|(\mathbf{w}_{k+1} - \tilde{\mathbf{w}}_{k+1}) = |\mathbf{B}_{k+1}(\sigma)|\boldsymbol{\mathcal{E}}_{k+1,+}.$$

The result follows similarly for $\boldsymbol{\mathcal{E}} \leftarrow \boldsymbol{\mathcal{E}}_{k+1,-}$; this completes the proof. ■

6.6.4 Proof of Theorem 6.1

Now we start the formal proof of Theorem 6.1. We will break the proof into two parts, i.e., the *down* and *up* parts involving $\mathsf{DiffL}_k^{\text{down}}$ and $\mathsf{DiffL}_k^{\text{up}}$, respectively.

Proof of the $\mathsf{DiffL}_k^{\text{down}}$ term in Theorem 6.1. The explicit form of the *down* Laplacian can be written as

$$\hat{\mathbf{L}}_d = \left[\begin{array}{c|c|c} \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{0} & \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}} \\ \hline \mathbf{0} & \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{0} & \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}} \end{array}\right].$$

And,

$$\mathbf{L}_d = \left[\begin{array}{c|c|c} \mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k^{1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k^{1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}} \end{array}\right].$$

Here, $\mathbf{M}_{\mathfrak{N}}$, $\mathbf{M}_{\mathfrak{C}}$, and $\mathbf{M}_{\mathfrak{D}}$ are diagonal masks for k-simplex sets $\mathfrak{N}$, $\mathfrak{C}$, and $\mathfrak{D}$, respectively.

By triangular inequality,

$$\left\|\mathbf{L}_d - \hat{\mathbf{L}}_d\right\| \leq \underbrace{\left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\mathbf{W}_k^{1/2}\mathbf{M}_{\mathfrak{N}} - \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\mathbf{M}_{\mathfrak{N}}\right\|}_{(*)} + 2\left[\underbrace{\left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}}\right\|}_{(\dagger)} + \left\|\mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}}\right\|\right]. \tag{6.5}$$

Expand the $\mathbf{W}_k$ with $\hat{\mathbf{W}}_k$ and omit $\mathbf{M}_{\mathfrak{N}}$ for simplicity, the first term of (6.5) can be bounded by

$$\begin{aligned}
(*) \leq & \left\|\hat{\mathbf{W}}_k^{1/2}\left(\mathbf{I} + \left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)\right)\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\left(\mathbf{I} - \left(\mathbf{E}_{k-1,+} - \mathbf{E}_{k-1,-}\right)\right)\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\left(\mathbf{I} + \left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)\right)\right. \\
& \left. -\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\right\| \\
\leq & \left[\left(\left\|2\cdot\left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)\right\| + \left\|\left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)^2\right\|\right)\cdot\left\|\hat{\mathbf{L}}_d\right\| + \right. \\
& \left.\left(1+\sqrt{\epsilon_k'}\right)^2\left\|\hat{\mathbf{W}}_k^{1/2}\mathbf{B}_k^\top\hat{\mathbf{W}}_{k-1}^{-1}\left(\mathbf{E}_{k-1,+} - \mathbf{E}_{k-1,-}\right)\mathbf{B}_k\hat{\mathbf{W}}_k^{1/2}\right\|\right] \\
\leq & \left[\left\|2\cdot\left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)\right\| + \left\|\left(\mathbf{E}_{k,+}^+ - \mathbf{E}_{k,-}^+\right)^2\right\| + \left(1+\sqrt{\epsilon_k'}\right)^2\left\|\mathbf{E}_{k-1,+} - \mathbf{E}_{k-1,-}\right\|\right]\cdot\|\hat{\mathbf{L}}_d\| \\
\overset{*}{\leq} & \left[2\sqrt{\epsilon_k'} + \epsilon_k' + \left(1+\sqrt{\epsilon_k'}\right)^2\sqrt{\epsilon_{k-1}'}\right]\cdot\left\|\tilde{\mathbf{L}}_d\right\|.
\end{aligned}$$

The last two terms of (6.5) can be bounded using Lemma 6.5, i.e.,

$$\begin{aligned}
(\dagger) = \left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}}\right\| &\leq \left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{1/2}\mathbf{B}_k^\top\mathbf{W}_{k-1}^{-1/2}\right\|\cdot\left\|\mathbf{W}_{k-1}^{-1/2}\mathbf{B}_k\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathcal{I}}\right\| \\
&\leq \|\mathbf{L}_d\|\sqrt{\epsilon_{k-1}}.
\end{aligned}$$

The last term of (6.5) can also be bounded by $\|\tilde{\mathbf{L}}_d\|\sqrt{\epsilon_{k-1}}$ using Lemma 6.5. Since $\|\mathbf{L}_d\|$, $\left\|\hat{\mathbf{L}}_d\right\|$, and $\left\|\tilde{\mathbf{L}}_d\right\|$ have the same upper bound λ_{k-1}, we have

$$\left\|\mathbf{L}_d - \hat{\mathbf{L}}_d\right\|^2 \leq \left[2\sqrt{\epsilon'_k} + \epsilon'_k + \left(1+\sqrt{\epsilon'_k}\right)^2 \sqrt{\epsilon'_{k-1}} + 4\sqrt{\epsilon_{k-1}}\right]^2 \lambda_{k-1}^2.$$

■

Proof of the $\mathsf{DiffL}_k^{\mathrm{up}}$ term in Theorem 6.1. The explicit form of $\hat{\mathbf{L}}_u$ is,

$$\hat{\mathbf{L}}_u = \left[\begin{array}{c|c|c} \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\hat{\mathbf{W}}_k^{-1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{0} & \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}} \\ \hline \mathbf{0} & \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\hat{\mathbf{W}}_k^{-1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{0} & \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{M}_{\mathfrak{D}} \end{array}\right].$$

And,

$$\mathbf{L}_u = \left[\begin{array}{c|c|c} \mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\mathbf{W}_k^{-1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\mathbf{W}_k^{-1/2}\mathbf{M}_{\mathfrak{N}} & \mathbf{M}_{\mathfrak{C}}\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{M}_{\mathfrak{C}} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{M}_{\mathfrak{D}}\boldsymbol{\mathcal{E}}_{k,-}^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{-1/2}\mathbf{M}_{\mathfrak{D}} \end{array}\right].$$

The perturbation is,

$$\left\|\mathbf{L}_u - \hat{\mathbf{L}}_u\right\| \leq \underbrace{\left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\mathbf{W}_k^{-1/2}\mathbf{M}_{\mathfrak{N}} - \mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\hat{\mathbf{W}}_k^{-1/2}\mathbf{M}_{\mathfrak{N}}\right\|}_{(*)} + 2\left[\underbrace{\left\|\mathbf{M}_{\mathfrak{N}}\mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,+}^{1/2}\mathbf{M}_{\mathfrak{C}}\right\|}_{(\dagger)} + \left\|\mathbf{M}_{\mathfrak{N}}\hat{\mathbf{W}}_k^{-1/2}\mathbf{B}_{k+1}\hat{\mathbf{W}}_{k+1}\mathbf{B}_{k+1}^\top\boldsymbol{\mathcal{E}}_{k,-}^{1/2}\mathbf{M}_{\mathfrak{D}}\right\|\right]. \tag{6.6}$$

The first term of (6.6) can be bounded by expanding $\mathbf{W}_{k+1}$ w.r.t. $\hat{\mathbf{W}}_{k+1}$, i.e., $\mathbf{W}_{k+1} = \hat{\mathbf{W}}_{k+1} + (\boldsymbol{\mathcal{E}}_{k+1,+} - \boldsymbol{\mathcal{E}}_{k+1,-})$. As slight abuse of notation, we let $\mathbf{W}_k \leftarrow \mathbf{W}_k[\mathfrak{N},\mathfrak{N}]$, $\mathbf{B}_{k+1} \leftarrow \mathbf{B}_{k+1}[\mathfrak{N},:]$. The first term $(*)$ of (6.6) becomes

$$
\begin{aligned}
(*) &\leq \left\| \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \hat{\mathbf{W}}_{k+1} \mathbf{B}_{k+1}^\top \mathbf{W}_k^{-1/2} - \hat{\mathbf{W}}_k^{-1/2} \mathbf{B}_{k+1} \hat{\mathbf{W}}_{k+1} \mathbf{B}_{k+1}^\top \hat{\mathbf{W}}_k^{-1/2} \right\| \\
&\leq \left\| \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \hat{\mathbf{W}}_{k+1} \mathbf{B}_{k+1}^\top \mathbf{W}_k^{-1/2} - \hat{\mathbf{W}}_k^{-1/2} \mathbf{B}_{k+1} \hat{\mathbf{W}}_{k+1} \mathbf{B}_{k+1}^\top \hat{\mathbf{W}}_k^{-1/2} \right\| + \\
&\quad \left\| \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,+} \mathbf{B}_{k+1}^\top \mathbf{W}_k^{-1/2} - \hat{\mathbf{W}}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,-} \mathbf{B}_{k+1}^\top \hat{\mathbf{W}}_k^{-1/2} \right\| \\
&\overset{\ddagger}{\leq} \left(2\|\mathbf{E}_{k,+}^- - \mathbf{E}_{k,-}^-\| + \left\| \left(\mathbf{E}_{k,+}^- - \mathbf{E}_{k,-}^- \right)^2 \right\| \right) \cdot \left\| \hat{\mathbf{L}}_u \right\| + \\
&\quad \left\| \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,+} \mathbf{B}_{k+1}^\top \mathbf{W}_k^{-1/2} - \hat{\mathbf{W}}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,-} \mathbf{B}_{k+1}^\top \hat{\mathbf{W}}_k^{-1/2} \right\| \\
&\overset{\S}{\leq} \left[2\sqrt{\epsilon_k'} + \epsilon_k' + 2\epsilon_k \right] \lambda_k
\end{aligned}
$$

The ‡ term holds by expanding $\mathbf{W}_k^{-1/2} = \hat{\mathbf{W}}_1^{-1/2} \left(\mathbf{I} - (\mathbf{E}_{k,+}^- - \mathbf{E}_{k,-}^-) \right)$ and following a similar approach of the *down* Laplacian. The § term holds by bounding $\mathbf{E}_{k,+}^- - \mathbf{E}_{k,-}^-$ with Assumption 6.4 (ϵ_k') and using Lemma 6.6 (ϵ_k).

The (†) term in (6.6) can be bounded by ϵ_k using Lemma 6.6, i.e.,

$$
\begin{aligned}
(\dagger) &\overset{\ddagger}{=} \left\| \mathbf{M}_{\mathfrak{N}} \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,+} \mathbf{B}_{k+1}^\top \boldsymbol{\mathcal{E}}_{k,+}^{-1/2} \mathbf{M}_{\mathfrak{C}} \right\| \\
&\leq \left\| \mathbf{M}_{\mathfrak{N}} \mathbf{W}_k^{-1/2} \mathbf{B}_{k+1} \boldsymbol{\mathcal{E}}_{k+1,+}^{1/2} \right\| \cdot \left\| \boldsymbol{\mathcal{E}}_{k+1,+}^{1/2} \mathbf{B}_{k+1}^\top \boldsymbol{\mathcal{E}}_{k,+}^{-1/2} \right\| \\
&\leq \sqrt{\lambda_k \epsilon_k} \cdot \left\| \boldsymbol{\mathcal{E}}_{k+1,+}^{1/2} \mathbf{B}_{k+1}^\top \boldsymbol{\mathcal{E}}_{k,+}^{-1/2} \right\| \\
&\overset{\S}{\leq} \sqrt{\epsilon_k} \lambda_k.
\end{aligned}
$$

‡ holds because the intersection of triangles of $\boldsymbol{\mathcal{E}}_{k,+}$, and $\mathbf{W}_k$ is the triangles with non-zero entries in $\boldsymbol{\mathcal{E}}_{k+1,+}$. § holds (the $\sqrt{\lambda_k}$ term) because $\boldsymbol{\mathcal{E}}_{k+1,+}^{1/2} \mathbf{B}_{k+1}^\top \boldsymbol{\mathcal{E}}_{k,+}^{-1/2}$ is a submatrix of $\mathbf{W}_{k+1}^{1/2} \mathbf{B}_{k+1}^\top \mathbf{W}_k^{-1/2}$; hence, the spectral norm will be upper bounded by the *up* Laplacian $\|\mathbf{L}_u\| \leq \lambda_k$.

Therefore, we have

$$
\left\| \mathbf{L}_u - \hat{\mathbf{L}}_u \right\|^2 \leq \left[2\sqrt{\epsilon_k'} + \epsilon_k' + 2\epsilon_k + 4\sqrt{\epsilon_k} \right]^2 \lambda_k^2.
$$

Combining the bound involving $\mathsf{DiffL}_k^{\text{down}}$ completes the proof of Theorem 6.1. ■

6.6.5 The maximum eigenvalue of $\mathcal{L}_k$ constructed from a cubical complex (Corollary 6.2)

In this section, we would like to show the bound on the spectral norm of $\boldsymbol{\mathcal{L}}_k$ built from a cubical complex. The property is found useful in extending Theorem 6.1 to Corollary 6.2; namely, the goal is to show that $\|\boldsymbol{\mathcal{L}}_k\|_2 \leq \lambda_k = (2k+2)$. Note that $\|\boldsymbol{\mathcal{L}}_k^{\text{down}}\| = \|\mathbf{A}_k^\top \mathbf{A}_k\| = \|\mathbf{A}_k \mathbf{A}_k^\top\| = \|\boldsymbol{\mathcal{L}}_{k-1}^{\text{up}}\|$. W.l.o.g., one can inspect only the up-Laplacian. We provide the following proposition that is largely based on the similar analysis [58] of $\|\boldsymbol{\mathcal{L}}_k\|$ for SC.

Proposition 6.7. *Given an* up *k-Laplacian $\boldsymbol{\mathcal{L}}_k^{\text{up}} = \mathbf{A}_{k+1}\mathbf{A}_{k+1}^\top$ with $\mathbf{A}_{k+1} = \mathbf{W}_k^{-1/2}\mathbf{B}_{k+1}\mathbf{W}_{k+1}^{1/2}$ built from a cubical complex, we have*

$$\|\boldsymbol{\mathcal{L}}_k^{\text{up}}\|_2 \leq \lambda_k = 2k+2.$$

Proof. From Schaub et al. [105], the eigenvalues of the k-th renormalized *up*-Laplacian $\boldsymbol{\mathcal{L}}_k^{\text{up}}$ are identical to those of the k-th *random-walk up*-Laplacian $\boldsymbol{\mathcal{L}}_k^{\text{rw}} = \mathbf{W}_k^{-1}\mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}$. Further, let $\mathbf{L}_k^{\text{up}} = \mathbf{B}_{k+1}\mathbf{W}_{k+1}\mathbf{B}_{k+1}^\top$, following the analysis of [58], we have

$$\begin{aligned}
\mathbf{f}^\top \mathbf{L}_k^{\text{up}} \mathbf{f} &= \left(\mathbf{W}_{k+1}^{1/2}\mathbf{B}_{k+1}^\top \mathbf{f}\right)^\top \left(\mathbf{W}_{k+1}^{1/2}\mathbf{B}_{k+1}^\top \mathbf{f}\right) \\
&= \sum_{\sigma \in K_k} \sum_{\tau \in \text{coface}(\sigma)} f^2(\sigma) w_{k+1}(\tau) \\
&\overset{\dagger}{\leq} (2k+2) \sum_{\sigma \in K_k} f^2(\sigma) \sum_{\tau \in \text{coface}(\sigma)} w_{k+1}(\tau) \\
&= (2k+2) \sum_{\sigma \in K_k} f^2(\sigma) \deg(\sigma).
\end{aligned}$$

The inequality † holds using the Cauchy-Schwarz inequality; the $2k+2$ term comes from the fact that a $(k+1)$-cube has $(2k+2)$ faces. Following the rest of the proof in [58], we

have

$$\|\boldsymbol{\mathcal{L}}_k^{\mathrm{up}}\| = \|\boldsymbol{\mathcal{L}}_k^{\mathrm{rw,up}}\| = \frac{\|\mathbf{L}_k^{\mathrm{up}}\|}{\mathbf{f}^\top \mathbf{W}_k \mathbf{f}} \leq (2k+2) \frac{\sum_{\sigma \in K_k} f^2(\sigma) \deg(\sigma)}{\sum_{\sigma \in K_k} f^2(\sigma) w_k(\sigma)} = 2k+2.$$

The first equality holds due to the identical eigenvalues of $\boldsymbol{\mathcal{L}}_k$ and $\boldsymbol{\mathcal{L}}_k^{\mathrm{rw}}$; the last inequality holds because we have $\mathbf{w}_k(\sigma) = |\mathbf{B}_{k+1}(\sigma)|\mathbf{w}_{k+1} = \deg(\sigma)$ for all $\sigma \in K_k$. ■

Chapter 7

CONCLUSION

In this thesis, we studied the discovery of topological structures and the vector fields learning on the high-dimensional point cloud dataset. The core mathematical tools used in this thesis are the graph Laplacian, the discrete Helmholtzian, and the k-Laplacians on simplicial/cubical complexes; they serve as the finite-sample estimators of the Laplace-Beltrami operator Δ_0 and its higher-order extensions Δ_k. The geometric information contained in these estimators is obtained by solving an eigenproblem or linear system involving $\boldsymbol{\mathcal{L}}_k$. In each chapter, we propose theoretically-grounded algorithms built upon these estimators for numerous applications in the domains of chemistry, astronomy, oceanography, and biology.

In Chapter 3, we study the well-documented deficiency, called the IES problem, in the DM embeddings when the aspect ratio of the data manifold $\mathcal{M}$ is large. We propose an efficient bicriterial algorithm based on a loss function $\mathfrak{L}$ with its minimizer corresponding to the subset of eigencoordinates forming a smooth embedding. We also analyze its large sample limit and show its resemblance to a Kullback-Leibler divergence. The proposed IES algorithm has lent itself to opportunities for future investigations; for instance, (i) instead of designing a set-based loss function $\mathfrak{L}$, one can extend this notion to find an optimal rotation/projection matrix such that the corresponding embedding is smooth (e.g., Figure 3.6f). Note that the set-based loss function is a special case of such extension, for we enforce a discrete set to be chosen. We conjecture that the loss function of this problem will be a *difference of convex* function; therefore, solvers exist for this functional form. Additionally, (ii) one might be able to extend the proposed algorithm to other popular learning algorithms such as HLLE and LTSA. The challenging part of this extension is in estimating the "smoothness" (the regularization term in (3.1)), since these algorithms usually do not provide an estimate of the smoothness of the embedding just as DM does in (3.6). Finally, we show that (in Section 3.5.2) UMAP, a popular data visualization algorithm used in single-cell RNA datasets, also suffers from the IES problem. This observation opens up new problems for us to answer, i.e., (iii) understanding the limitations and deficiencies of these loss-based embedding algorithms. Moreover, (iv) proposing a validation framework, such as Meila [78], Savvides et al. [103], for testing whether the structures obtained from these unsupervised learning algorithms are

signals or noises.

In Chapter 4, we introduce an estimator for the manifold Helmholtzian Δ_1, whose triangles are weighted by the product of pairwise kernels in (4.2). We analyze separately the large sample limit of the *up*/*down* Laplacians with these special triangle weights and show their pointwise and spectral convergences to Δ_1^{up} and Δ_1^{down}, respectively. We assume that the sampling density $\psi(\mathbf{x})$ is a constant supported on the manifold $\mathcal{M}$ for our theoretical analysis; unfortunately, this is not always the case for real datasets. We partially solve this issue by sampling the furthest points using Algorithm 4.1; however, we conjecture that (i) one might get similar asymptotic results using the δ-CkNN graph [15]; this approach has been explored partially in Chapter 6.

An indirect benefit here is that by using the graph or simplicial complex constructed by the nearest-neighbor-based methods of (2.6) and (2.7), one can usually get a smaller number of edges n_1 compared to that from a δ-radius graph. Dependency on the number of edges n_1 has a direct impact on the computational overhead. Namely, we often need to solve an eigenproblem or linear system with $\boldsymbol{\mathcal{L}}_1$ whose size is $n_1 \times n_1$. (ii) One possible future direction to explore is the *spectral sparsification* of $\boldsymbol{\mathcal{L}}_k$. Principled methods exist for graph Laplacians (see, e.g., Spielman and Srivastava [114]); Osting et al. [91] is the first to extend the framework of spectral sparsification to higher-order Laplacians. However, limitations still apply, e.g., the efficient sparsification is only possible if the *leverage score* can be easily approximated, which in general is not currently possible for $\boldsymbol{\mathcal{L}}_k$ without providing any assumption on the simplicial complexes built from the point cloud. (iii) Sparsifying the simplicial complex or approximating existing complexes built from the point cloud (combinatorially) is another possible approach. Examples include the approximation of the VR complex [38, 109], the approximation of the Čech complex [66], or the development of completely new complexes (e.g., the graph induced complex [37]). The goal of these approaches is to preserve information in the *persistent homology* sense. Ways to connect these algorithms with the spectral sparsification methods might also be another topic to explore.

In Chapter 5, we reanalyze some of the datasets in oceanography and single-cell RNA. We demonstrate the possibility of extracting the geometric and topological information from the eigenforms of the Helmholtzian estimator $\boldsymbol{\mathcal{L}}_1$ as well as the applications in noisy vector field smoothing, semi-supervised edge flow learning, and vector field reconstruction from trajectories. The proposed framework is a significant building block for new applications and algorithms in other domains of study. For instance, (i) a paragraph of text can be viewed as a trajectory of words; therefore, one might be able to analyze the underlying vector fields/systems using the well-studied word embeddings [81, 94]. A prior study with a heuristic for creating the temporal information in the small/simple rhymes datasets [132] has been explored. We expect our proposed framework can serve as a theoretically justified alternative for analyzing these datasets; this will allow us to build better language models using high-order relationships. Similarly, (ii) topological feature discovery on various (scientific) datasets studied priorly using PDs [2, 50, 102, 123] can also be extended to the edge flow learning scenario with their temporal (trajectory) information. (iii) Low dimensional representations of the flows, especially the independent flow subspaces [26], can be applied in visualization and in diagnosing the phase transitions [75] of vector fields. (iv) Moreover, one can extract parametric decompositions (e.g., different terms of the Navier Stokes in fluid flows) from the non-parametric Helmholtz-Hodge decomposition using a similar approach as Meila et al. [80].

Designing a better mapping of the vector fields between different representations can benefit the analyses of vector fields in single-cell RNA sequencing. La Manno et al. [69] have proposed a method based on the mapping between the Markov chains between the graph of the original space and the embedding space; however, the theoretical insights are currently lacking. (v) One possible differential geometry-based approach is to use the mapping method introduced in Section 2.5.3 assisted with diffeomorphism conditions between embeddings. In Section 4.3, the mapping of the eigenforms in PCA space (e.g., Figures 4.3c and 4.3d) to torsion space (e.g., Figure 4.3e) is not a diffeomorphism. To ensure the validity of this assumption, we present a partial solution by introducing the periodic boundary condition.

However, diffeomorphism might not always hold in practice since most of the loss-based embedding algorithms (UMAP/tSNE) break the neighborhood relationships. Additionally, the proper boundary condition to maintain diffeomorphism might be intractable due to the complexity of the used data/algorithm. Since diffeomorphism can be encoded in the neighborhood graph, estimating a mapping between the neighborhood graphs constructed from the two embeddings might be a viable approach. (vi) Another way to tackle this problem is to use the notion of isometry. More precisely, if the mapping is an isometry, then the 1-cochain for every edge is preserved by Definition 2.4. One might be able to achieve this goal using the pushforward metric introduced by Perraul-Joncas and Meila [95].

In Chapter 6, we study the problem of the decomposition of the homology embedding for a manifold [27]. Under the lens of matrix perturbation theory, we show that the homology vector space $\mathcal{H}_k$, where the embedding resides, is approximately factorizable if the targeted manifold is decomposable. We derive a perturbation bound for the homology embedding under the assumption that the manifold is a (sparsely) connected sum of disjoint manifolds. Empirically, we observe that the sparsely connected assumption can be relaxed; therefore, (i) one potential direction for further research is to derive a tighter bound of Theorem 6.1 based on a similar formulation. Additionally, (ii) we expect more spectral algorithms, such as higher-order clustering and shortest homologous loop extraction, to be built upon this framework. The proposed perturbation bound can be beneficial to derive the correctness of these algorithms just as spectral clustering and community detection [126] algorithms do.

The proposed framework is an extension of the spectral clustering (which uses the information of the zeroth homology space) to the higher-order setting; it is particularly motivated by the studies in the geometry of the spectral embedding of null space [87, 106]. A practical use case for the spectral clustering algorithms is when the data have β_0 weakly connected clusters; under this scenario, only the first eigenvalue will be zero, while the rest $(\beta_0 - 1)$ eigenvalues will be small but bounded away from zero. The above analyses for spectral clusterings (e.g., Ng et al. [87], Schiebinger et al. [106]) can be generalized to the weakly connected scenario; however, it is unclear whether it applies to general $\mathcal{H}_k$. Therefore, (iii)

one interesting direction to explore is: under what conditions and assumptions, the proposed decomposition framework can be extended to data that have "weak" loop structures, e.g., when the loop is slightly twisted together at some regions.

Another potential future work is to (iv) extend the Metric learning algorithms [68], which use the clustering information $\mathcal{H}_0$ to the high-order setting $\mathcal{H}_k$. The classical metric learning algorithms aim at finding the appropriate metrics, in terms of the Mahalanobis distance, from supervised labels so that the resulting embedding is consistent with the given labels/clusters. Extending the framework to higher-order scenarios by enforcing topological constraints can increase interpretability. For instance, if we are interested in analyzing `ETH` with a prior knowledge that $\beta_1 = 2$, the goal of the higher-order metric learning embedding will have a more detectable inner loop than that from the original point cloud (PCA) or the DM embedding.

BIBLIOGRAPHY

[1] Kevork N Abazajian, Jennifer K Adelman-McCarthy, Marcel A Agüeros, Sahar S Allam, Carlos Allende Prieto, Deokkeun An, Kurt SJ Anderson, Scott F Anderson, James Annis, Neta A Bahcall, et al. The seventh data release of the sloan digital sky survey. *The Astrophysical Journal Supplement Series*, 182(2):543, 2009.

[2] Saad Ali, Arslan Basharat, and Mubarak Shah. Chaotic Invariants for Human Action Recognition. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, October 2007. doi: 10.1109/ICCV.2007.4409046.

[3] Jose Alquicira-Hernandez, Joseph Powell, and Tri Giang Phan. No evidence that plasmablasts transdifferentiate into developing neutrophils in severe COVID-19 disease. *bioRxiv*, 2020.

[4] Mark Anthony Armstrong. *Basic Topology*. Springer Science & Business Media, 2013.

[5] Douglas Arnold, Richard Falk, and Ragnar Winther. Finite element exterior calculus: From Hodge theory to numerical stability. *Bulletin of the American Mathematical Society*, 47(2):281–354, 2010. ISSN 0273-0979, 1088-9485. doi: 10.1090/S0273-0979-10-01278-4.

[6] Sergio Barbarossa and Stefania Sardellitti. Topological signal processing over simplicial complexes. *IEEE Transactions on Signal Processing*, 68:2992–3007, 2020.

[7] Jonathan Bates. The embedding dimension of Laplacian eigenfunction maps. *Applied and Computational Harmonic Analysis*, 37(3):516–530, November 2014. ISSN 1063-5203. doi: 10.1016/j.acha.2014.03.002.

[8] Mikhail Belkin and Partha Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, June 2003. ISSN 0899-7667. doi: 10.1162/089976603321780317.

[9] Mikhail Belkin and Partha Niyogi. Convergence of laplacian eigenmaps. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 129–136. MIT Press, 2007.

[10] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(Nov):2399–2434, 2006.

[11] Anthony J. Bell and Terrence J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159, 1995.

[12] Volker Bergen, Marius Lange, Stefan Peidli, F. Alexander Wolf, and Fabian J. Theis. Generalizing RNA velocity to transient cell states through dynamical modeling. *Nature biotechnology*, 38(12):1408–1414, 2020.

[13] Tyrus Berry and Dimitrios Giannakis. Spectral exterior calculus. *Communications on Pure and Applied Mathematics*, 73(4):689–770, 2020.

[14] Tyrus Berry and John Harlim. Variable Bandwidth Diffusion Kernels. *arXiv:1406.5064 [math]*, June 2014.

[15] Tyrus Berry and Timothy Sauer. Consistent manifold representation for topological data analysis. *Foundations of Data Science*, 1(1):1, 2019. doi: 10.3934/fods.2019001.

[16] H. Bhatia, G. Norgard, V. Pascucci, and P. Bremer. The Helmholtz-Hodge Decomposition—A Survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(8): 1386–1404, August 2013. doi: 10.1109/TVCG.2012.316.

[17] Yochai Blau and Tomer Michaeli. Non-Redundant Spectral Dimensionality Reduction. *ArXiv*, abs/1612.03412, 2017. doi: 10.1007/978-3-319-71249-9_16.

[18] Imre Bokor, Diarmuid Crowley, Stefan Friedl, Fabian Hebestreit, Daniel Kasprowski, Markus Land, and Johnny Nicholson. Connected sum decompositions of high-dimensional manifolds. *arXiv:1909.02628 [math]*, September 2020.

[19] Ingwer Borg and Patrick JF Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer Science & Business Media, 2005.

[20] Ozan Candogan, Ishai Menache, Asuman Ozdaglar, and Pablo A. Parrilo. Flows and Decompositions of Games: Harmonic and Potential Games. *Mathematics of Operations Research*, 36(3):474–503, August 2011. ISSN 0364-765X, 1526-5471. doi: 10.1287/moor.1110.0500.

[21] Frédéric Chazal and Bertrand Michel. An introduction to Topological Data Analysis: Fundamental and practical aspects for data scientists. *arXiv:1710.04019 [cs, math, stat]*, October 2017.

[22] Frédéric Chazal, David Cohen-Steiner, and Quentin Mérigot. Geometric Inference for Probability Measures. *Foundations of Computational Mathematics*, 11(6):733–751, December 2011. ISSN 1615-3383. doi: 10.1007/s10208-011-9098-0.

[23] Dudley B. Chelton, Michael G. Schlax, Michael H. Freilich, and Ralph F. Milliff. Satellite measurements reveal persistent small-scale features in ocean winds. *science*, 303 (5660):978–983, 2004.

[24] Guangliang Chen, Anna V. Little, and Mauro Maggioni. Multi-resolution geometric analysis for data in high dimensions. In *Excursions in Harmonic Analysis, Volume 1*, pages 259–285. Springer, 2013.

[25] Yen-Chi Chen. A tutorial on kernel density estimation and recent advances. *Biostatistics & Epidemiology*, 1(1):161–187, 2017.

[26] Yu-Chia Chen and Marina Meilă. Selecting the independent coordinates of manifolds with large aspect ratios. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1086–1095. Curran Associates, Inc., 2019.

[27] Yu-Chia Chen and Marina Meilă. The decomposition of the higher-order homology embedding constructed from the k-Laplacian. *arXiv:2107.10970 [stat.ML]*, July 2021.

[28] Yu-Chia Chen, Marina Meilă, and Ioannis G. Kevrekidis. Helmholtzian Eigenmap: Topological feature discovery & edge flow learning from point cloud data. *arXiv:2103.07626 [stat.ML]*, March 2021.

[29] Stefan Chmiela, Alexandre Tkatchenko, Huziel E Sauceda, Igor Poltavsky, Kristof T Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science advances*, 3(5):e1603015, 2017.

[30] Fan Chung. *Spectral Graph Theory*, volume 92 of *CBMS Regional Conference Series in Mathematics*. American Mathematical Society, December 1996. ISBN 978-0-8218-0315-8 978-0-8218-8936-7 978-1-4704-2452-7. doi: http://dx.doi.org/10.1090/cbms/092.

[31] Jeanne N. Clelland. *From Frenet to Cartan: The Method of Moving Frames*, volume 178. American Mathematical Soc., 2017.

[32] Ronald R. Coifman and Stéphane Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, July 2006. ISSN 1063-5203. doi: 10.1016/j.acha.2006.04.006.

[33] Bruno Colbois. Spectral Theory and Geometry. pp.34:cel–00392158, 2006.

[34] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 303–312, 1996.

[35] Sanjoy Dasgupta and Kaushik Sinha. Randomized partition trees for exact nearest neighbor search. In *Conference on Learning Theory*, pages 317–337, 2013.

[36] Tamal K. Dey, Jian Sun, and Yusu Wang. Approximating loops in a shortest homology basis from point data. In *Proceedings of the Twenty-Sixth Annual Symposium on Computational Geometry*, pages 166–175, 2010.

[37] Tamal K. Dey, Fengtao Fan, and Yusu Wang. Graph Induced Complex on Point Data. *arXiv:1304.0662 [cs, math]*, April 2013.

[38] Tamal K. Dey, Dayu Shi, and Yusu Wang. SimBa: An Efficient Tool for Approximating Rips-filtration Persistence via Simplicial Batch-collapse. *arXiv:1609.07517 [cs, math]*, page 16 pages, 2016. doi: 10.4230/LIPIcs.ESA.2016.35.

[39] Tamal K. Dey, Tao Hou, and Sayan Mandal. Computing minimal persistent cycles: Polynomial and hard cases. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2587–2606. SIAM, 2020.

[40] Jozef Dodziuk. Finite-Difference Approach to the Hodge Theory of Harmonic Forms. *American Journal of Mathematics*, 98(1):79–104, 1976. ISSN 0002-9327. doi: 10.2307/2373615.

[41] Józef Dodziuk and Jeffrey McGowan. The Spectrum of the Hodge Laplacian for a Degenerating Family of Hyperbolic Three Manifolds. *Transactions of the American Mathematical Society*, 347(6):1981–1995, 1995. ISSN 0002-9947. doi: 10.2307/2154917.

[42] David L. Donoho and Carrie Grimes. Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences*, 100(10):5591–5596, May 2003. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1031596100.

[43] I. L. (Ian L.) Dryden. *Statistical Shape Analysis : With Applications in R*. Wiley Series in Probability and Statistics. Wiley, Chichester, West Sussex, England, 2nd ed. edition, 2016. ISBN 1-119-07250-6.

[44] Carmeline J Dsilva, Ronen Talmon, Ronald R Coifman, and Ioannis G Kevrekidis. Parsimonious representation of nonlinear dynamical systems through manifold learning: A chemotaxis case study. *Applied and Computational Harmonic Analysis*, 44(3):759–773, 2018.

[45] Stefania Ebli and Gard Spreemann. A Notion of Harmonic Clustering in Simplicial Complexes. *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 1083–1090, December 2019. doi: 10.1109/ICMLA.2019.00182.

[46] Beno Eckmann. Harmonische Funktionen und Randwertaufgaben in einem Komplex. *Commentarii mathematici Helvetici*, 17:240–255, 1944. ISSN 0010-2571; 1420-8946/e.

[47] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[48] Kelly L. Fleming, Pratyush Tiwary, and Jim Pfaendtner. New approach for investigating reaction dynamics and rates with ab initio calculations. *Jornal of Physical Chemistry A*, 120(2):299–305, 2016. doi: DOI:10.1021/acs.jpca.5b10667.

[49] Gary Froyland and Kathrin Padberg-Gehle. A rough-and-ready cluster-based approach for extracting finite-time coherent sets from sparse and incomplete trajectory data. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 25(8):087406, July 2015. ISSN 1054-1500. doi: 10.1063/1.4926372.

[50] Chad Giusti, Robert Ghrist, and Danielle S. Bassett. Two's company, three (or more)

is a simplex. *Journal of Computational Neuroscience*, 41(1):1–14, August 2016. ISSN 1573-6873. doi: 10.1007/s10827-016-0608-6.

[51] Yair Goldberg, Alon Zakai, Dan Kushnir, and Ya'acov Ritov. Manifold learning: The price of normalization. *Journal of Machine Learning Research*, 9(Aug):1909–1939, 2008.

[52] Leo J. Grady and Jonathan Polimeni. *Discrete Calculus: Applied Analysis on Graphs for Computational Science*. Springer-Verlag, London, 2010. ISBN 978-1-84996-289-6.

[53] David A Harville. Matrix algebra from a statistician's perspective. 1998.

[54] Allen Hatcher. *Algebraic Topology*. , 2005.

[55] Matthias Hein, Jean-Yves Audibert, and Ulrike Von Luxburg. From graphs to manifolds–weak and strong pointwise consistency of graph Laplacians. In *International Conference on Computational Learning Theory*, pages 470–485. Springer, 2005.

[56] Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. Graph laplacians and their convergence on random neighborhood graphs. *Journal of Machine Learning Research*, 8(6), 2007.

[57] Adam Hoover and Michael Goldbaum. Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels. *IEEE transactions on medical imaging*, 22 (8):951–958, 2003.

[58] Danijela Horak and Jürgen Jost. Spectra of combinatorial Laplace operators on simplicial complexes. *Advances in Mathematics*, 244:303–336, September 2013. ISSN 0001-8708. doi: 10.1016/j.aim.2013.05.007.

[59] Roger A Horn, Roger A Horn, and Charles R Johnson. *Matrix Analysis*. Cambridge university press, 1990.

[60] Ryohko Ishikawa, Javier Trujillo Bueno, Tanausú del Pino Alemán, Takenori J. Okamoto, David E. McKenzie, Frédéric Auchère, Ryouhei Kano, Donguk Song, Masaki Yoshida, Laurel A. Rachmeler, Ken Kobayashi, Hirohisa Hara, Masahito Kubo, Noriyuki Narukage, Taro Sakao, Toshifumi Shimizu, Yoshinori Suematsu, Christian Bethge, Bart De Pontieu, Alberto Sainz Dalda, Genevieve D. Vigil, Amy Winebarger, Ernest Alsina Ballester, Luca Belluzzi, Jiří Štěpán, Andrés Asensio Ramos, Mats Carlsson, and Jorrit Leenaarts. Mapping solar magnetic fields from the photosphere to the base of the corona. *Science Advances*, 7(8):eabe8406, February 2021. ISSN 2375-2548. doi: 10.1126/sciadv.abe8406.

[61] Rishabh Iyer and Jeff Bilmes. Algorithms for approximate minimization of the difference between submodular functions, with applications. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, UAI'12, pages 407–417, Arlington, Virginia, United States, 2012. AUAI Press. ISBN 978-0-9749039-8-9.

[62] Junteng Jia, Michael T. Schaub, Santiago Segarra, and Austin R. Benson. Graph-based Semi-Supervised & Active Learning for Edge Flows. In *KDD*, 2019. doi: 10.1145/3292500.3330872.

[63] Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. Statistical ranking and combinatorial Hodge theory. *Mathematical Programming*, 127(1):203–244, March 2011. ISSN 1436-4646. doi: 10.1007/s10107-010-0419-x.

[64] Dominique Joncas, Marina Meila, and James McQueen. Improved Graph Laplacian via Geometric Self-Consistency. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4457–4466. Curran Associates, Inc., 2017.

[65] Karin Kailing, Hans-Peter Kriegel, and Peer Kröger. Density-connected subspace clustering for high-dimensional data. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pages 246–256. SIAM, 2004.

[66] Michael Kerber and R. Sharathkumar. Approximate Čech Complex in Low and High Dimensions. In Leizhen Cai, Siu-Wing Cheng, and Tak-Wah Lam, editors, *Algorithms and Computation*, Lecture Notes in Computer Science, pages 666–676, Berlin, Heidelberg, 2013. Springer. ISBN 978-3-642-45030-3. doi: 10.1007/978-3-642-45030-3_62.

[67] Violeta Kovacev-Nikolic, Peter Bubenik, Dragan Nikolić, and Giseon Heo. Using persistent homology and dynamical distances to analyze protein binding. *Statistical applications in genetics and molecular biology*, 15(1):19–38, 2016.

[68] Brian Kulis. Metric Learning: A Survey. *Foundations and Trends® in Machine Learning*, 5(4):287–364, 2013. ISSN 1935-8237, 1935-8245. doi: 10.1561/2200000019.

[69] Gioele La Manno, Ruslan Soldatov, Amit Zeisel, Emelie Braun, Hannah Hochgerner, Viktor Petukhov, Katja Lidschreiber, Maria E. Kastriti, Peter Lönnerberg, and Alessandro Furlan. RNA velocity of single cells. *Nature*, 560(7719):494–498, 2018.

[70] John M. Lee. Introduction to smooth manifolds. 2003.

[71] John M. Lee. Introduction to Smooth Manifolds. In *Introduction to Smooth Manifolds*, pages 1–31. Springer, 2013.

[72] Elizaveta Levina and Peter J Bickel. Maximum likelihood estimation of intrinsic dimension. In *Advances in Neural Information Processing Systems*, pages 777–784, 2005.

[73] Lek-Heng Lim. Hodge laplacians on graphs. *Siam Review*, 62(3):685–715, 2020.

[74] Chuanjiang Luo, Issam Safa, and Yusu Wang. Approximating gradients for meshes and point clouds via diffusion metric. In *Computer Graphics Forum*, volume 28, pages 1497–1508. Wiley Online Library, 2009.

[75] Eran Lustig, Or Yair, Ronen Talmon, and Mordechai Segev. Identifying Topological Phase Transitions in Experiments Using Manifold Learning. *Physical Review Letters*, 125(12):127401, 2020.

[76] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[77] James McQueen, Marina Meilă, Jacob VanderPlas, and Zhongyue Zhang. Megaman: Scalable manifold learning in python. *Journal of Machine Learning Research*, 17(148): 1–5, 2016.

[78] Marina Meila. How to tell when a clustering is (approximately) correct using convex relaxations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 7407–7418. Curran Associates, Inc., 2018.

[79] Marina Meilă and Jianbo Shi. A random walks view of spectral segmentation. In *International Workshop on Artificial Intelligence and Statistics*, pages 203–208. PMLR, 2001.

[80] Marina Meila, Samson Koelle, and Hanyu Zhang. A regression approach for explaining manifold embedding coordinates. *arXiv:1811.11891 [cs, stat]*, November 2018.

[81] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[82] John Milnor. A unique decomposition theorem for 3-manifolds. *American Journal of Mathematics*, 84(1):1–7, 1962.

[83] Sayan Mukherjee and John Steenbergen. Random walks on simplicial complexes and harmonics. *Random structures & algorithms*, 49(2):379–405, 2016.

[84] Sayan Mukherjee and Qiang Wu. Estimation of gradients and coordinate covariation in classification. *Journal of Machine Learning Research*, 7(Nov):2481–2514, 2006.

[85] Boaz Nadler, Stéphane Lafon, Ronald R. Coifman, and Ioannis G. Kevrekidis. Diffusion

maps, spectral clustering and reaction coordinates of dynamical systems. *Applied and Computational Harmonic Analysis*, 21(1):113–127, 2006.

[86] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming*, 14(1):265–294, 1978.

[87] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[88] John Novembre and Matthew Stephens. Interpreting principal component analyses of spatial population genetic variation. *Nature genetics*, 40(5):646–649, May 2008. ISSN 1061-4036. doi: 10.1038/ng.139.

[89] Ippei Obayashi. Volume Optimal Cycle: Tightest representative cycle of a generator on persistent homology. *arXiv:1712.05103 [cs, math]*, December 2017.

[90] Antonio Ortega, Pascal Frossard, Jelena Kovačević, José M. F. Moura, and Pierre Vandergheynst. Graph Signal Processing: Overview, Challenges, and Applications. *Proceedings of the IEEE*, 106(5):808–828, May 2018. ISSN 1558-2256. doi: 10.1109/JPROC.2018.2820126.

[91] Braxton Osting, Sourabh Palande, and Bei Wang. Spectral Sparsification of Simplicial Complexes for Clustering and Label Propagation. *arXiv:1708.08436 [cs]*, February 2019.

[92] Nina Otter, Mason A. Porter, Ulrike Tillmann, Peter Grindrod, and Heather A. Harrington. A roadmap for the computation of persistent homology. *EPJ Data Science*, 6 (1):17, December 2017. ISSN 2193-1127. doi: 10.1140/epjds/s13688-017-0109-5.

[93] Ori Parzanchevski and Ron Rosenthal. Simplicial complexes: Spectrum, homology and random walks. *Random Structures & Algorithms*, 50(2):225–261, March 2017. ISSN 1098-2418. doi: 10.1002/rsa.20657.

[94] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[95] Dominique Perraul-Joncas and Marina Meila. Non-linear dimensionality reduction: Riemannian metric estimation and the problem of geometric discovery. *arXiv:1305.7255 [stat]*, May 2013.

[96] Dominique Perrault-Joncas and Marina Meila. Estimating Vector Fields on Manifolds and the Embedding of Directed Graphs. *arXiv:1406.0013 [cs, stat]*, May 2014.

[97] Jacobus W. Portegies. Embeddings of Riemannian manifolds with heat kernels and eigenfunctions. *Communications on Pure and Applied Mathematics*, 69(3):478–518, 2016.

[98] Olaf Post. First Order Approach and Index Theorems for Discrete and Metric Graphs. *Annales Henri Poincaré*, 10(5):823–866, August 2009. ISSN 1424-0661. doi: 10.1007/s00023-009-0001-3.

[99] David Rey, Hillel Bar-Gera, Vinayak V. Dixit, and S. Travis Waller. A branch-and-price algorithm for the bilevel network maintenance scheduling problem. *Transportation Science*, 53(5):1455–1478, 2019.

[100] Steven Rosenberg and Rosenberg Steven. *The Laplacian on a Riemannian Manifold: An Introduction to Analysis on Manifolds*. Number 31. Cambridge University Press, 1997.

[101] Wouter Saelens, Robrecht Cannoodt, Helena Todorov, and Yvan Saeys. A comparison of single-cell trajectory inference methods. *Nature biotechnology*, 37(5):547–554, 2019.

[102] Manish Saggar, Olaf Sporns, Javier Gonzalez-Castillo, Peter A. Bandettini, Gunnar Carlsson, Gary Glover, and Allan L. Reiss. Towards a new approach to reveal dynam-

ical organization of the brain using topological data analysis. *Nature communications*, 9(1):1–14, 2018.

[103] Rafael Savvides, Andreas Henelius, Emilia Oikarinen, and Kai Puolamäki. Significance of patterns in data visualisations. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1509–1517, 2019.

[104] M. T. Schaub and S. Segarra. Flow Smoothing And Denoising: Graph Signal Processing In The Edge-Space. In *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 735–739, November 2018. doi: 10.1109/GlobalSIP.2018.8646701.

[105] Michael T. Schaub, Austin R. Benson, Paul Horn, Gabor Lippner, and Ali Jadbabaie. Random walks on simplicial complexes and the normalized Hodge 1-Laplacian. *SIAM Review*, 62(2):353–391, 2020.

[106] Geoffrey Schiebinger, Martin J. Wainwright, and Bin Yu. The geometry of kernelized spectral clustering. *Annals of Statistics*, 43(2):819–846, April 2015. ISSN 0090-5364, 2168-8966. doi: 10.1214/14-AOS1283.

[107] Michael Schweinberger. Instability, sensitivity, and degeneracy of discrete exponential families. *Journal of the American Statistical Association*, 106(496):1361–1370, 2011.

[108] Cosma Rohilla Shalizi and Alessandro Rinaldo. Consistency under sampling of exponential random graph models. *Annals of statistics*, 41(2):508, 2013.

[109] Donald R. Sheehy. Linear-size approximations to the Vietoris–Rips filtration. *Discrete & Computational Geometry*, 49(4):778–796, 2013.

[110] Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis*. Routledge, 2018.

[111] A. Singer. From graph to manifold Laplacian: The convergence rate. *Applied and Computational Harmonic Analysis*, 21(1):128–134, July 2006. ISSN 1063-5203. doi: 10.1016/j.acha.2006.03.004.

[112] Amit Singer and Hau-tieng Wu. Vector Diffusion Maps and the Connection Laplacian. *arXiv:1102.0075 [math, stat]*, January 2011.

[113] Nikhil Singh, Heather D. Couture, J. S. Marron, Charles Perou, and Marc Niethammer. Topological descriptors of histology images. In *International Workshop on Machine Learning in Medical Imaging*, pages 231–239. Springer, 2014.

[114] Daniel A. Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. *SIAM Journal on Computing*, 40(6):1913–1926, 2011.

[115] G. W. Stewart, J. W. Stewart, and Ji-guang Sun. *Matrix Perturbation Theory*. Elsevier Science, July 1990. ISBN 978-0-12-670230-9.

[116] Walter A Strauss. *Partial Differential Equations: An Introduction*. Wiley, 2007.

[117] Michael Tabor. *Chaos and Integrability in Nonlinear Dynamics: An Introduction*. Wiley, 1989.

[118] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, December 2000. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.290.5500.2319.

[119] Daniel Ting and Michael I. Jordan. On Nonlinear Dimensionality Reduction, Linear Smoothing and Autoencoding. *arXiv:1803.02432 [stat]*, March 2018.

[120] Daniel Ting, Ling Huang, and Michael Jordan. An Analysis of the Convergence of Graph Laplacians. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, 2010.

[121] Nicolas Garcia Trillos and Dejan Slepčev. A variational approach to the consistency of spectral clustering. *Applied and Computational Harmonic Analysis*, 45(2):239–281, 2018.

[122] Nicolás García Trillos, Moritz Gerlach, Matthias Hein, and Dejan Slepčev. Error estimates for spectral convergence of the graph Laplacian on random geometric graphs toward the Laplace–Beltrami operator. *Foundations of Computational Mathematics*, 20(4):827–887, 2020.

[123] Rien Van De Weygaert, Gert Vegter, Herbert Edelsbrunner, Bernard JT Jones, Pratyush Pranav, Changbom Park, Wojciech A. Hellwing, Bob Eldering, Nico Kruithof, and EGP Patrick Bos. Alpha, betti and the megaparsec universe: On the topology of the cosmic web. In *Transactions on Computational Science XIV*, pages 60–101. Springer, 2011.

[124] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. ISSN ISSN 1533-7928.

[125] Ulrike von Luxburg. A Tutorial on Spectral Clustering. *arXiv:0711.0189 [cs]*, November 2007.

[126] Yali Wan and Marina Meila. A class of network models recoverable by spectral clustering. In *NIPS*, pages 3285–3293, 2015.

[127] Frank W. Warner. *Foundations of Differentiable Manifolds and Lie Groups*, volume 94. Springer Science & Business Media, 2013.

[128] Larry Wasserman. Topological data analysis. *Annual Review of Statistics and Its Application*, 5:501–532, 2018.

[129] Hassler Whitney. *Geometric Integration Theory*. Dover Publications, Mineola, N.Y, December 2005. ISBN 978-0-486-44583-0.

[130] Yi Yu, Tengyao Wang, and Richard J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

[131] Zhenyue Zhang and Hongyuan Zha. Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment. *SIAM Journal of Scientific Computing*, 26:313–338, 2002.

[132] Xiaojin Zhu. Persistent Homology: An Introduction and a New Text Representation for Natural Language Processing. In *Twenty-Third International Joint Conference on Artificial Intelligence*, June 2013.

[133] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. 2002.

[134] Xiaojin Zhu, Zoubin Ghahramani, and John Lafferty. Semi-supervised learning using Gaussian fields and harmonic functions. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, ICML'03, pages 912–919, Washington, DC, USA, August 2003. AAAI Press. ISBN 978-1-57735-189-4.