Title

Dimensionality reduction in biology

Author

Xu, Boyan

Publication Date

2024

Peer reviewed|Thesis/dissertation

Dimensionality reduction in biology

by

Boyan Xu

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Mathematics

and the Designated Emphasis

in

Computational and Genomic Biology

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Ksenia V. Krasileva, Co-chair
Professor Steven N. Evans, Co-chair
Professor Noah Whiteman
Professor Javier Arsuaga

Summer 2024

Abstract

Dimensionality reduction in biology

by

Boyan Xu

Doctor of Philosophy in Mathematics

Designated Emphasis in Computational and Genomic Biology

University of California, Berkeley

Professor Ksenia V. Krasileva, Co-chair

Professor Steven N. Evans, Co-chair

Dimensionality reduction techniques play a crucial role in analyzing and interpreting complex biological data. This dissertation explores the application of these techniques in three distinct areas: nonlinear time series analysis, neural data analysis, and protein domain annotation. The work presented here bridges pure mathematics, applied mathematics, and computational biology, showcasing the versatility and power of dimensionality reduction methods in addressing diverse biological questions. In Chapter 1, I introduce the overarching theme of dimensionality reduction in biological data and provide a brief overview of the field's current state. The chapter sets the stage for the detailed explorations in the subsequent chapters. Chapter 2 presents the first paper, "Twisty Takens: a geometric characterization of good observations on dense trajectories." This work focuses on delay embeddings of time series data and the conditions necessary for successful topological reconstructions of trajectories on manifolds. Using persistent cohomology and Eilenberg-MacLane coordinates, we demonstrate methods for reducing the dimensionality of high-dimensional embeddings, allowing for the identification of various topological shapes in naturally occurring phenomena. Chapter 3 offers a discussion linking the first paper to the second, highlighting the common thread of persistent cohomology as a tool for dimensionality reduction and topological analysis. In Chapter 4, the second paper, "Evaluating State Space Discovery by Persistent Cohomology in the Spatial Representation System," is presented. This study evaluates the ability of persistent cohomology to uncover topological structures in high-dimensional neural recordings. By focusing on the firing rates of grid cells in the brain's spatial representation system, we reconstruct the 2D trajectories of an animal, demonstrating the efficacy of circular coordinates for toroidal data. Chapter 5 provides a discussion that transitions from the neural data analysis of the second paper to the structural analysis in the third paper, emphasizing the

application of dimensionality reduction techniques across different biological scales and data types. Chapter 6 contains the third paper, "Structure-Aware Annotation of Leucine-rich Repeat Domains." This work leverages deep learning-based protein structure prediction to improve the annotation of Leucine-rich repeat domains. By employing differential geometry and dimensionality reduction methods, we enhance the accuracy of domain annotation and detect structural features in protein curves, demonstrating the practical application of mathematical techniques in bioinformatics. Finally, Chapter 7 offers a concluding discussion that synthesizes the findings of the three papers, explores their implications for the field of computational biology, and suggests future research directions. The dissertation highlights the interdisciplinary nature of dimensionality reduction techniques and their potential to advance our understanding of complex biological systems.

Contents

Acknowledgments

I thank my advisor, Ksenia Krasileva, for welcoming me into her lab partway through graduate school despite my being a newcomer to biology. Ksenia has inspired and influenced my leadership and mentorship abilities beyond measure. I am extraordinarily lucky to have been mentored by someone as responsive, attentive, relatable, and inviting as she, and it's truly difficult to imagine how I would've made it through my PhD otherwise.

I thank all members of Krasileva Lab for participating in and contributing to such an extraordinary scientific and social community. I swear our lab manager, China Lunde, is the greatest lab manager in the world. She makes all of us feel completely taken care of and never fails to put a smile on my face every time I see her.

I thank my mother and father for raising me and fostering my intellectual development, facilitating my interest in computer programming. I thank Emily Hill for starting a mushroom farm with me, supporting me through all the ups and downs of grad school, and encouraging me to realize my dream of becoming a biologist. I thank Alois Cerbu for being an incredible friend, music mentor, and collaborator whose brilliance and humor never ceases to spark joy in me. I thank Chris Tralie for being such an exceptional mentor and collaborator in scientific computing—I'd be a far less capable data scientist and programmer without the guidance he has provided over the years.

I thank Katherine Baney for motivating me to pursue my interests in entrepreneurship and biosynthetic pathway discovery. I thank my friend and labmate Chandler Sutherland for teaching me plant biology, helping me be the best version of myself, and lifting me up in my moments of self-doubt. I thank Evan Groover for his exuberance and supportiveness, answering my rudimentary questions and fueling me with confidence to pursue research in synthetic biology. I thank Noah Whiteman for helping me enter an exciting new world of organismal biology, genetics, and psychedelic science. Finally, I thank Paul Daley for being such an inspiring role model and friend. He has encouraged me more than anyone else to explore the breadth and depths of my scientific curiosities to the fullest extent.

Chapter 1

Introduction to Dimensionality Reduction in Biology and "Twisty Takens" paper

The complexity and volume of biological data present significant challenges for analysis and interpretation. Understanding the underlying dynamics of biological processes often requires sophisticated methods that can manage high-dimensional data while preserving essential features. Dimensionality reduction techniques are crucial for uncovering patterns, relationships, and structures that might otherwise remain hidden in these vast datasets. This dissertation explores various approaches to dimensionality reduction, focusing on their applications and implications in biological contexts.

The concept of dimensionality reduction is not new to biology; it has been a cornerstone of data analysis for decades. Traditional methods such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) have been widely used to simplify complex datasets. These methods, while powerful, rely on linear transformations and thus may fail to capture the inherent nonlinear structures present in biological data. As biological data have grown in complexity and volume, so too has the need for more advanced techniques. This dissertation delves into some of these advanced methods, emphasizing their geometric foundations and the insights they provide into biological systems.

One of the foundational principles in the study of dynamical systems is the use of delay coordinate embeddings. Originally proposed by Takens in 1981, this method provides a way to reconstruct the state space of a dynamical system from a time series of observations [107]. The delay coordinate mapping, or sliding window embedding, transforms a one-dimensional time series into a geometric object in a higher-dimensional space. This transformation reveals the underlying structure of the system, allowing for the identification of patterns and features that are not apparent in the original time series [78, 62, 25]. However, while the delay embedding increases the dimensionality, it sets the stage for subsequent dimensionality reduction techniques that simplify the resulting high-dimensional representations.

The geometric nature of these embeddings makes them particularly useful for analyzing

biological signals, which often exhibit complex, nonlinear behaviors. For instance, sliding window embeddings have been applied to ECG signals to detect patterns associated with heart conditions [103, 91], and to gene expression data to uncover periodicities in circadian rhythms [89]. The ability to visualize and analyze these geometric structures provides a powerful tool for understanding the dynamics of biological systems.

A significant advancement in managing the high-dimensional outputs of delay embeddings is the use of persistent cohomology and Eilenberg-MacLane coordinates. Persistent cohomology is a method from topological data analysis that captures the shape of data across multiple scales, identifying features that persist over a range of parameters. Eilenberg-MacLane coordinates, also known as "projective PCA," provide a way to project high-dimensional data onto lower-dimensional spaces in a way that preserves the topological and geometric properties of the original data. This method is particularly powerful for identifying underlying structures in data that might otherwise be obscured by noise or complexity.

Eilenberg-MacLane coordinates work by mapping high-dimensional data into lower-dimensional spaces using algebraic topological properties. Essentially, these coordinates help in identifying circular structures within the data, which is particularly useful for systems exhibiting periodic or quasi-periodic behavior. This process significantly reduces the dimensionality of the data while preserving its essential topological features, making it easier to analyze and interpret.

The first paper included in this dissertation, titled "Twisty Takens: A Geometric Characterization of Dynamical Observations and Delay Coordinate Embeddings," extends Takens' embedding theory by providing a geometric framework for understanding when and how delay coordinate embeddings can be effectively used. The paper introduces a set of conditions that characterize the types of observations that yield high-dimensional embeddings, and then demonstrates how persistent cohomology and Eilenberg-MacLane coordinates can be used to reduce these high-dimensional representations into more manageable forms.

This dissertation aims to bridge the gap between theoretical advances in dimensionality reduction and their practical applications in biology. By exploring geometric characterizations of dynamical observations, it provides new tools and methods for analyzing complex biological data. The following chapters will delve deeper into specific applications and case studies, illustrating the power and versatility of these techniques in uncovering the hidden structures within biological systems.

Chapter 2 presents the paper "Twisty Takens," detailing the theoretical foundations and practical implications of the geometric approach to delay coordinate embeddings. This paper serves as a cornerstone for understanding the broader themes of dimensionality reduction in biological systems, setting the stage for subsequent chapters that explore additional methods and applications. Through this exploration, we aim to demonstrate how advanced mathematical techniques can enhance our ability to analyze and interpret complex biological data.

Chapter 2

Twisty Takens: A Geometric Characterization of Good Observations on Dense Trajectories

The contents of this chapter are based on a publication of BX et al. [124].

Abstract

In nonlinear time series analysis and dynamical systems theory, Takens' embedding theorem states that the sliding window embedding of a generic observation along trajectories in a state space, recovers the region traversed by the dynamics. This can be used, for instance, to show that sliding window embeddings of periodic signals recover topological loops, and that sliding window embeddings of quasiperiodic signals recover high-dimensional torii. However, in spite of these motivating examples, Takens' theorem does not in general prescribe how to choose such an observation function given particular dynamics in a state space. In this work, we state conditions on observation functions defined on compact Riemannian manifolds, that lead to successful reconstructions for particular dynamics. We apply our theory and construct families of time series whose sliding window embeddings trace tori, Klein bottles, spheres, and projective planes. This greatly enriches the set of examples of time series known to concentrate on various shapes via sliding window embeddings, and will hopefully help other researchers in identifying them in naturally occurring phenomena. We also present numerical experiments showing how to recover low dimensional representations of the underlying dynamics on state space, by using the persistent cohomology of sliding window embeddings and Eilenberg-MacLane (i.e., circular and real projective) coordinates.

2.1 Introduction

The *delay coordinate mapping*, or *sliding window embedding* [107, 78, 62, 25], posits a time series as a sequence of observations made along trajectories in a hidden state space. Under this scheme, a one dimensional time series, which could otherwise be analyzed with more traditional linear analysis techniques such as ARMA and Fourier/Wavelet analysis, is instead turned into a geometric object via a vector of samples of the time series, which moves along the signal (Equation 6.11). The shape of this geometric object provides information about the system under study. Periodic processes, for example, map to points which concentrate on a topological loop. Sliding window embeddings have been used in this context, for example, to analyze ECG signals of a beating heart [103, 91], to detect chatter in mechanical systems [64], to quantify repetitive motions in human activities [37, 119], to discover periodicity in gene expression during circadian rhythms [89], and to detect wheezing in audio signals [34]. In addition to loops, torus shapes often show up during "quasiperiodicity," which is a state of near-chaos. Sliding window embeddings have witnessed this torus shape in such applications as vocal fold anomalies [50], horse whinnies [13], neural networks [75], and oscillating cylinder flow [44]. Certain time series even concentrate on fractals after a sliding window embedding [107, 100]. Sliding window embeddings have also been used as a tool for shape analysis more generally even when an underlying model for the dynamics is unknown, such as in music structure analysis [9, 98]. We direct the interested reader to [87] for a recent review on how topological data analysis can be used in the analysis of time delay embeddings.

The main theory motivating the use of sliding window embeddings in all of these applications is Takens' delay embedding [107] theorem, which is stated as follows:

Theorem (Takens' embedding theorem [107])**.** *Let M be a compact manifold of dimension m. Suppose X is a smooth vector field with flow $\psi_t : M \to M$ and G is a smooth function on M. For $\tau > 0$, $N \geq 2m$, and pairs (X, G) it is a generic property that $\Psi_\tau^N : M \to \mathbb{R}^{N+1}$ defined by*

$$\Psi_\tau^N(p) = (G(p), G(\psi_\tau(p)), G(\psi_{2\tau}(p)), \ldots, G(\psi_{N\tau}(p)))$$

is an embedding.

A "random" choice of X and G makes the delay coordinate mapping Ψ_τ^N a smooth embedding. Thus, remarkably, the state space M of a dynamical system may in general be reconstructed from a single generic observation function G[1], which gives rise to a 1D time series. However, in practice, Takens' result is ill-suited for computational purposes because it does not provide an explicit characterization of "genericity". In this work, we extend Takens' embedding theory with a geometric characterization of observations which yield high-dimensional delay coordinate embeddings, given a particular flow on a manifold. Our main theoretical result for general compact manifolds is stated in Theorem 2.4.1 in Section 2.4, as follows:

[1]Some texts refer to this as an "observable."

Theorem. *The Takens map Ψ_τ^N is an embedding for some dimension $N > 0$ and flow time $\tau > 0$, if the following conditions hold:*

1. *For any point of $p \in M$ there is an m-tuple $J \in \mathbb{Z}_{\geq 0}^m$ of nonnegative integers such that the m-form*
$$\mathcal{L}_X^{\wedge J} dG := \bigwedge_{j \in J} \mathcal{L}_X^j dG$$
is nonzero at some point on the integral curve $\gamma_p(s)$. Here, $\mathcal{L}_X^j$ denotes the j^{th}-order Lie derivative.

2. *For any pair of distinct points $p, q \in M$ the observation curves $g_p(s)$ and $g_q(s)$ are not identical.*

We first provide several examples in Section 2.3 which satisfy the conditions of our theorem. In the process, we discuss a non-example that violates condition 1 if we're not careful (Example 2.3.3) and show another non-example which violates condition 2 (Example 2.3.2, part 2). We then prove our theorem in Section 2.4, and we explore a special case in Section 2.5 in which Fourier bases can be used to construct observation functions[2].

2.2 Background

In this section, we provide a more detailed overview of several concepts utilized in this work, including sliding window embeddings, persistent (co)homology, and Eilenberg-MacLane coordinates. The latter two tools will be used to empirically validate that our sliding window embeddings recover our chosen state space and the underlying dynamics.

Sliding Window Embeddings

We express a time series $g(t)$ as an observation G along a dense trajectory γ on a manifold M, i.e.
$$g(t) = G(\gamma(t))$$
for $\gamma : \mathbb{R} \longrightarrow M$ and $G : M \longrightarrow \mathbb{R}$. We compute the *sliding window* of g as
$$\text{SW}_\tau^N g(t) := \begin{bmatrix} g(t) \\ g(t+\tau) \\ g(t+2\tau) \\ \vdots \\ g(t+N\tau) \end{bmatrix} \in \mathbb{R}^{N+1} \tag{2.1}$$

[2]The code to generate all figures in this manuscript can be found at http://www.github.com/ctralie/TwistyTakens

where $N \in \mathbb{N}$ is the number of delays, $\tau > 0$ is the delay time, and $N\tau$ is the window length.

We interpret the sliding window $\mathrm{SW}_\tau^N g(t)$ as the evaluation of the Takens map Ψ_τ^N in Theorem 2.1 above on an integral curve $\psi_t(p)$ of a vector field X through a point $p \in M$. For if $N = 2 \cdot \dim M$ and $\gamma(t) = \psi_t(p)$, then

$$\mathrm{SW}_\tau^N g(t) = \Psi_\tau^N(\psi_t(p)).$$

For sufficiently large N and small τ, $\mathrm{SW}_\tau^N g(t)$ densely "traces" the embedding $\Psi_\tau^N(M)$ for appropriate choice of observation G and vector field X.

When g is a periodic function with frequency $\omega \in \mathbb{R}$, it readily follows that the sliding window embedding $\mathrm{SW}_\tau^N g(t)$ traces a closed curve in $\mathbb{R}^{N+1}$. The shape of this curve is closely related to the choice of parameters N and τ, and their relation to ω [88]. In particular, if τ and N are chosen so that N is large enough and $N\tau\omega \approx 1$, then the image of $\mathrm{SW}_\tau^N g$ is in fact a topological circle in $\mathbb{R}^{N+1}$, whose shape is tightly controlled by the Fourier coefficients of g. In other words, the periodic nature of g — a spectral property — is reflected in the circularity of its sliding window, a topological feature. Quasiperiodicity is another spectral notion with a clear geometric/topological counterpart. Indeed, let $1, \omega_1, \ldots, \omega_n \in \mathbb{R}$ be linearly independent over the rational numbers. We say that $f : \mathbb{R} \longrightarrow \mathbb{R}$ is quasiperiodic with frequencies $\omega_1, \ldots, \omega_n$, if it can be written as $f(t) = F(t, \ldots, t)$ for some function $F : \mathbb{R}^n \longrightarrow \mathbb{R}$ whose j-th marginals $f_j(t) = F(t_1, \ldots, t_{j-1}, t, t_{j+1}, \ldots, t_n)$ are periodic with frequency ω_j. In this case, and for appropriate N and τ, the set $\mathrm{SW}_\tau^N f(\mathbb{Z})$ is dense in an n-dimensional torus embedded in $\mathbb{R}^{N+1}$ [84, 38].

Koopman spectra

We now review another relevant tool that goes along with sliding window embeddings. For positive flow time $t > 0$, the flow ψ_t of a vector field X on a compact manifold M defines a diffeomorphism $\psi_t : M \to M$. Then the composition map U^t, or Koopman operator [65, 25, 72] given by

$$U^t G = G \circ \psi_t,$$

is a linear operator on the space of observation functions on M. The coordinates of the delay mapping are thus iterated applications of U^t on an observation G.

For certain classes of dynamical systems, the Koopman operator possesses a discrete spectrum and yields a linear expansion

$$G = \sum_{k=0}^{\infty} G_k \varphi_k$$

where φ_k are eigenfunctions of U^t and G_k are *Koopman modes*. For such systems one "lifts" the dynamics on the state space to an evolution of observables. For a more comprehensive overview of Koopman theory and its applications, please refer to [1].

We will see in Section 2.5 that a high-dimensional delay mapping essentially recovers the Koopman modes of an observation function. We therefore characterize delay embedding

observations in terms of spectral decomposition properties. We examine a special case with a Fourier basis for the Koopman operator on the Torus and Klein bottle, and show via our main Theorem 2.4.1 what is needed of these coefficients.

Persistent Homology

In practice we evaluate the sliding window $\mathrm{SW}_\tau^N g(t)$ at a finite set of evenly sampled time points $t_1 < \cdots < t_J$. This results in a discrete collection of J vectors, referred to as a "sliding window point cloud". The topology of a point cloud with J points is trivial; it consists of J connected components and lacks any other topological features (loops, voids, etc). However, if we use a *simplicial complex* (a discrete object) to approximate the underlying space from which the point cloud is sampled, then we can estimate the underlying topology via combinatorial means. A simplicial complex on a set V of vertices (e.g., a sliding window point cloud) is a collection K of nonempty subsets $\sigma \subset V$, so that if $\emptyset \neq \tau \subset \sigma \in K$, then $\tau \in K$. As an example, suppose we seek a simplicial complex with topology reflecting that of the unit circle S^1. Starting with the set $V = \{a, b, c\}$ of vertices, we let $K = \{a, b, c, \{a, b\}, \{b, c\}, \{a, c\}\}$ be the simplicial complex containing 3 edges between every pair of vertices. Like S^1, K has one connected component, one loop which bounds an empty space, and no higher dimensional features (voids, etc).

So far, our description of simplicial complexes has been purely combinatorial/topological, but one can use geometry to inform their construction. An early scheme in Euclidean space is the alpha complex [33], constructed as a family of subcomplexes of Delaunay triangulations at different scales. An even simpler construction, which works in any metric space, is the so-called "Vietoris-Rips" complex at scale $\alpha \geq 0$, denoted $R_\alpha(V)$. It is comprised of the finite subsets of V which have diameter less than α. Choosing the "appropriate" scale is ill-posed. For instance, Figure 2.1 shows a point cloud in $\mathbb{R}^2$ for which it is impossible to choose an appropriate scale at which the simplicial complex contains the two empty loops that are present in the original shape.

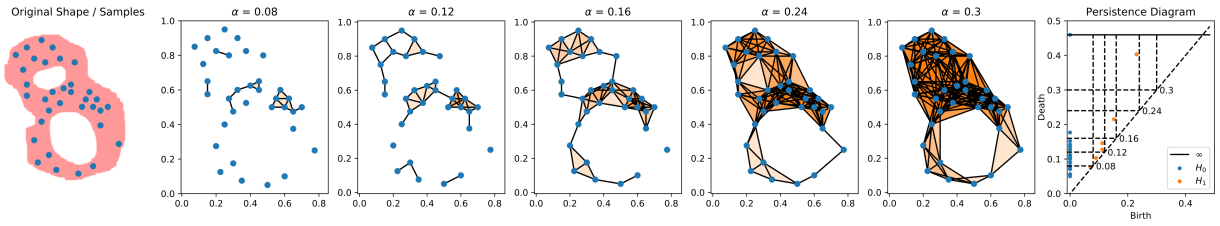

Figure 2.1: An example of the Rips filtration on a point cloud sampled from a thickened figure eight. The Rips complex is shown at different scales, the values of α, producing the persistence diagram on the right.

Specifically, $R_{0.16}(V)$ contains the upper loop, but not the lower loop, and $R_{0.24}(V)$ contains the lower loop, but the upper loop is no longer empty. In fact, it is impossible to

choose an α in which both loops are present and empty in $R_\alpha(V)$ in this example. However, we can still summarize the multiscale topological information of any point cloud by performing a *filtration* of the complex. That is, we evaluate $R_\alpha(V)$ as α varies continuously from 0 to some maximum value, so that $R_{\alpha_1}(V) \subset R_{\alpha_2}(V)$ if $\alpha_1 \leq \alpha_2$. Throughout this process we keep track of topological features as they appear, or are "born," and as they are filled in, or "die". For each such *homology class*, we can produce a point in a scatter plot, known as the *persistence diagram* of the filtration, with birth time on the x-axis and death time on the y-axis. Figure 2.1 shows a persistence diagram associated with our running example[3]. Intuitively, points further from the diagonal correspond to larger topological features which "persist" (stay alive) over longer intervals, and points closer to the diagram correspond to small, "noisy" features which are often artifacts of sampling (e.g. the square and pentagon loop that exist at $\alpha = 0.12$).

For completeness, we extend the above explanation with a brief rigorous presentation. For a more comprehensive treatment, please refer to [31, 30, 16, 42, 82]. Let $(\Gamma, \preceq)$ be a partially ordered set. A Γ-filtered simplicial complex is a collection $\mathcal{K} = \{K_\alpha\}_{\alpha\in\Gamma}$ of simplicial complexes, so that $K_\alpha \subset K_{\alpha'}$ for every $\alpha \preceq \alpha' \in \Gamma$. The typical examples for point cloud data are the Rips filtration, as we mentioned, and the Čech filtration motivated by the nerve lemma [49]. Specifically, let L be a finite subset of a metric space $(\mathbb{M}, \mathbf{d})$. The Rips filtration of L is the $\mathbb{R}$-filtered simplicial complex $\mathcal{R}(L) = \{R_\alpha(L)\}_{\alpha\in\mathbb{R}}$. Similarly, for $\ell \in L$ let

$$B_\alpha(\ell) = \{b \in \mathbb{M} : \mathbf{d}(b, \ell) < \alpha\} \qquad \text{and} \qquad \mathcal{B}_\alpha = \{B_\alpha(\ell) : \ell \in L\}.$$

The Čech complex $\check{C}_\alpha(L)$ is defined as the nerve of $\mathcal{B}_\alpha$; that is $\check{C}_\alpha(L) = \mathcal{N}(\mathcal{B}_\alpha)$ where

$$\sigma \in \mathcal{N}(\mathcal{B}_\alpha) \qquad \text{if and only if} \qquad \bigcap_{\ell\in\sigma} B_\alpha(\ell) \neq \emptyset$$

Hence $\check{C}(L) = \{\check{C}_\alpha\}_{\alpha\in\mathbb{R}}$ is an $\mathbb{R}$-filtered simplicial complex, and $R_\alpha(L) \subset \check{C}_\alpha(L) \subset R_{2\alpha}(L)$ for all $\alpha \in \mathbb{R}$.

The persistent homology (resp. cohomology) of a filtered complex $\mathcal{K} = \{K_\alpha\}_{\alpha\in\Gamma}$, with coefficients in a field $\mathbb{F}$, are defined, respectively, as

$$PH_n(\mathcal{K}; \mathbb{F}) := \bigoplus_{\alpha\in\Gamma} H_n(K_\alpha; \mathbb{F}) \qquad \text{and} \qquad PH^n(\mathcal{K}; \mathbb{F}) := \bigoplus_{\alpha\in\Gamma} H^n(K_\alpha; \mathbb{F})$$

Let $\iota_{\alpha,\alpha'} : H_n(K_\alpha; \mathbb{F}) \longrightarrow H_n(K_{\alpha'}; \mathbb{F})$ and $\jmath^{\alpha',\alpha} : H^n(K_{\alpha'}; \mathbb{F}) \longrightarrow H^n(K_\alpha; \mathbb{F})$ be the $\mathbb{F}$-linear maps induced by the inclusion $K_\alpha \subset K_{\alpha'}$, $\alpha \preceq \alpha'$. A persistent homology (resp. cohomology) class is an element $\bigoplus_{\alpha\in\Gamma} \nu_\alpha \in PH_n(\mathcal{K}; \mathbb{F})$ (resp. $\bigoplus_{\alpha\in\Gamma} \mu^\alpha \in PH^n(\mathcal{K}; \mathbb{F})$) so that $\iota_{\alpha,\alpha'}(\nu_\alpha) = \nu_{\alpha'}$ (resp $\jmath^{\alpha',\alpha}\left(\mu^{\alpha'}\right) = \mu^\alpha$) for every $\alpha \preceq \alpha'$.

When $\Gamma = \mathbb{R}$, a theorem of Crawley-Boevey [22] contends that if each $H_n(K_\alpha; \mathbb{F})$ is finite-dimensional (also known in the literature as being pointwise-finite) then one can choose bases S^α for each $H_n(K_\alpha; \mathbb{F})$, satisfying the following compatibility condition:

[3]We compute persistence diagrams for all examples in this paper using the Python interface to "Ripser" [8, 112]

1. $\iota_{\alpha,\alpha'}(S^\alpha) \subset \left(S^{\alpha'} \cup \{0\}\right)$ for every $\alpha \leq \alpha'$.

2. If $\iota_{\alpha,\alpha'}(\mathbf{v}_j^\alpha) = \iota_{\alpha,\alpha'}(\mathbf{v}_k^\alpha)$ and $j \neq k$, then $\iota_{\alpha,\alpha'}(\mathbf{v}_j^\alpha) = 0$.

The set $S = \bigcup_{\alpha\in\mathbb{R}} S^\alpha$ admits a partial order $\preceq$ given by $S^\alpha \ni \mathbf{v} \preceq \mathbf{v}' \in S^{\alpha'}$ if and only if $\alpha \leq \alpha'$ and $\iota_{\alpha,\alpha'}(\mathbf{v}) = \mathbf{v}'$. The maximal chains in $(S, \preceq)$ are the persistent homology classes. To each maximal chain $\mathscr{C} \subset S$ one can associated the point $(b_\mathscr{C}, d_\mathscr{C}) \in [-\infty, \infty] \times [-\infty, \infty]$ defined by

$$b_\mathscr{C} = \inf\{\alpha \in \mathbb{R} : S^\alpha \cap \mathscr{C} \neq \emptyset\} \quad , \quad d_\mathscr{C} = \sup\{\alpha \in \mathbb{R} : S^\alpha \cap \mathscr{C} \neq \emptyset\}$$

The collection of such pairs, where $\mathscr{C}$ runs over all maximal chains, is the persistence diagram for the persistence homology of the filtered complex $\mathcal{K}$.

Persistent cohomology behaves similarly. Indeed, any basis for $H_n(K_\alpha; \mathbb{F})$ yields a well-defined isomorphism $H_n(K_\alpha; \mathbb{F}) \cong H_n(K_\alpha; \mathbb{F})^*$ with the linear dual space, and the latter is naturally isomorphic to $H^n(K_\alpha; \mathbb{F})$, by the universal coefficient theorem. Hence, these isomorphisms turn the S^α's into a collection of compatible bases for the cohomology groups $H^n(K_\alpha; \mathbb{F})$, showing that persistent homology and cohomology yield the same persistence diagrams.

Persistent Homology of Sliding Window Embeddings

As mentioned in the introduction, there are numerous examples in the literature of persistent homology on sliding window point clouds. For any periodic time series $(x(t) = x(t + kT), k \in \mathbb{Z})$, a sliding window embedding yields a topological loop, and there is a point of high persistence in the persistence diagram for PH_1 [88]. However, the authors of [88] also show, surprisingly, that sliding window embeddings of functions like $x(t) = \cos(t) + a\cos(2t)$, $|a| > 1$, can lie on the boundary of an embedded Möbius strip [88]. We use this to help intuitively explain the time series we obtain for the projective plane (Example 2.3.4) and the Klein Bottle (Section 2.5). Note that this also means that field coefficients other than $\mathbb{Z}_2$ are needed to maximize the maximum persistence in PH_1. In general, for $\cos(t) + a\cos(kt)$, coefficients which are not prime factors of k are needed [88, 115]. Finally, there are works which utilize both PH_1 and PH_2 to quantify the presence of quasiperiodicity in time series data, by estimating the toroidality of a sliding window point cloud [84, 116]. In this work, we extend this suite of examples beyond (possibly twisted) loops and torii to other manifolds.

Eilenberg-MacLane Coordinates

Though persistent homology is informative, one can further utilize it to perform nonlinear dimensionality reduction on sliding window point clouds, for visualization purposes and reconstruction of the underlying dynamics. To this end, we use "Eilenberg-MacLane coordinates", which turn persistent cohomology classes into maps from point clouds to the circle [26, 86], and (real or complex) projective spaces [83]. We present next a more detailed

summary; maps to the projective plane are particularly interesting, as they allow us to "untwist" non-orientable manifolds like the Klein bottle.

More formally, if G is an abelian group [4] and n is a positive integer, then it is possible to construct a connected CW complex $K(G, n)$, called an Eilenberg-MacLane space, whose homotopy type is uniquely determined by two properties:

1. its j-th homotopy group $\pi_j(K(G, n))$ is trivial for all $j \neq n$
2. $\pi_n(K(G, n)) \cong G$

The Brown representability theorem (for CW complexes and singular cohomology) contends that if B is a CW complex, then there is a natural bijection

$$H^n(B; G) \cong [B, K(G, n)] \tag{2.2}$$

between the n-th cohomology of B with coefficients in G, and the set of homotopy classes of maps from B to $K(G, n)$.

The two Eilenberg-MacLane spaces we use to generate circular and projective coordinates are: $K(\mathbb{Z}, 1) \simeq S^1$, and $K(\mathbb{Z}/2, 1) \simeq \mathbb{R}\mathbf{P}^\infty = \mathbb{R}^\infty \setminus \{\mathbf{0}\}/\sim$, respectively. Here $\mathbb{R}^\infty$ is the collection of infinite sequences of real numbers $\mathbf{x} = (x_0, x_1, \ldots)$ which are nonzero for all but finitely many x_j's, and $\mathbf{x} \sim \mathbf{y}$ if and only if $\mathbf{x} = r\mathbf{y}$ for some $r \in \mathbb{R} \setminus \{0\}$. One can also regard $\mathbb{R}^\infty$ as the direct limit of the system $\mathbb{R} \subset \mathbb{R}^2 \subset \mathbb{R}^3 \subset \cdots$, where the inclusion $\mathbb{R}^j \hookrightarrow \mathbb{R}^{j+1}$ sends $(x_0, \ldots, x_{j-1})$ to $(x_0, \ldots, x_{j-1}, 0)$. With this interpretation in mind, $\mathbb{R}\mathbf{P}^\infty$ can be regarded as the direct limit of the system $\mathbb{R}\mathbf{P}^0 \subset \mathbb{R}\mathbf{P}^1 \subset \mathbb{R}\mathbf{P}^2 \subset \cdots$, where $\mathbb{R}\mathbf{P}^n = \mathbb{R}^{n+1} \setminus \{\mathbf{0}\}/\sim$. Recently [83], it has been shown that if L is a finite subset of a metric space $(\mathbb{M}, \mathbf{d})$, and for $\ell \in L$ we let $B_\alpha(\ell)$ be the open ball of radius α centered at ℓ, then persistent cohomology classes in $PH^1(\mathcal{R}(L); \mathbb{Z}/2)$ can be used to define projective coordinates

$$f_\mu : \bigcup_{\ell \in L} B_\alpha(\ell) \longrightarrow \mathbb{R}\mathbf{P}^n$$

Similarly, persistent cohomology classes in $PH^1(\mathcal{R}(L); \mathbb{Z}/q)$, for appropriate choices of prime $q > 2$, yield circular coordinates [86]

$$f_{\theta,\tau} : \bigcup_{\ell \in L} B_\alpha(\ell) \longrightarrow S^1$$

In both cases, the resulting coordinates mimic the properties of the bijection (2.2) from Brown's representability.

[4] In this section G will refer to an Abelian group, but it otherwise refers to an observation function.

Projective Coordinates

Here is a sketch of the construction of projective coordinates from persistent cohomology classes. Let $L = \{\ell_0, \dots, \ell_n\} \subset \mathbb{M}$, and fix a cocycle $\mu = \{\mu^\alpha_{jk}\} \in Z^1(R_{2\alpha}(L); \mathbb{Z}/2)$ so that its cohomology class is not in the kernel of the homomorphism

$$\iota^{2\alpha,\alpha} : H^1(R_{2\alpha}(L); \mathbb{Z}/2) \longrightarrow H^1(R_\alpha(L); \mathbb{Z}/2)$$

induced by the inclusion $R_\alpha(L) \subset R_{2\alpha}(L)$. Since $R_\alpha(L) \subset \check{C}_\alpha(L) \subset R_{2\alpha}(L)$, then the rightmost inclusion yields a nonzero class in $H^1(\check{C}_\alpha(L); \mathbb{Z}/2)$. We let

$$\begin{array}{rccl} f_\mu : & \bigcup\limits_{\ell\in L} B_\alpha(\ell) = L^{(\alpha)} & \longrightarrow & \mathbb{R}\mathbf{P}^n \\ & B_\alpha(\ell_j) \ni b & \mapsto & \left[(-1)^{\mu^\alpha_{j0}} |\alpha - \mathbf{d}(b, \ell_0)|_+ : \cdots : (-1)^{\mu^\alpha_{jn}} |\alpha - \mathbf{d}(b, \ell_n)|_+\right] \end{array}$$

where $[x_0 : \cdots : x_n] \in \mathbb{R}\mathbf{P}^n$ denotes the equivalence class of $(x_0, \dots, x_n) \in \mathbb{R}^{n+1} \smallsetminus \{\mathbf{0}\}$, and $|r|_+ := \max\{0, r\}$ for $r \in \mathbb{R}$. Since $\{\mu^\alpha_{jk}\}$ is a cocycle, it readily follows that the point $f_\mu(b) \in \mathbb{R}\mathbf{P}^n$ is independent of the index $j \in \{0, \dots, n\}$ for which $b \in B_\alpha(\ell_j)$. In other words, f_μ is well defined.

If $\{\nu^\alpha_{jk}\} \in Z^1(R_{2\alpha}(L); \mathbb{Z}/2)$ is cohomologous to $\{\mu^\alpha_{jk}\}$, and $f_\nu : L^{(\alpha)} \longrightarrow \mathbb{R}\mathbf{P}^n$ is the associated map, then $f_\mu \simeq f_\nu$ and hence we get a well defined function

$$\begin{array}{ccc} H^1(R_{2\alpha}(L); \mathbb{Z}/2) & \longrightarrow & \left[L^{(\alpha)}, \mathbb{R}\mathbf{P}^n\right] \\ [\mu] & \mapsto & [f_\mu] \end{array}$$

The metric properties of f_μ are also determined by the cohomology class of μ. For if

$$\mathbf{d}_g(\mathbf{x}, \mathbf{y}) := \arccos\left(\frac{|\langle \mathbf{x}, \mathbf{y}\rangle|}{\|\mathbf{x}\| \cdot \|\mathbf{y}\|}\right)$$

denotes the geodesic distance in $\mathbb{R}\mathbf{P}^n$, then it readily follows that

$$\mathbf{d}_g\left(f_\nu(b), f_\nu(b')\right) = \mathbf{d}_g\left(f_\mu(b), f_\mu(b')\right)$$

for all $b, b' \in L^{(\alpha)}$ and μ, ν in the same cohomology class.

Given a finite set $P \subset L^{(\alpha)}$, taking its image through f_μ yields a new point cloud $f_\mu(P) \subset \mathbb{R}\mathbf{P}^n$. A dimensionality-reduction scheme in $\mathbb{R}\mathbf{P}^n$ referred to as principal projective component analysis is also defined in [83]. This procedure yields a sequence of maps

$$P_k : f_\mu(P) \longrightarrow \mathbb{R}\mathbf{P}^k, \qquad k = 0, \dots, n$$

minimizing an appropriate notion of (metric) distortion. In particular, $P_k \circ f_\mu(P)$ and $P_k \circ f_\nu(P)$ are isometric if μ and ν are cohomologous. The point clouds $P_k \circ f_\mu(P) \subset \mathbb{R}\mathbf{P}^k$ are referred to as the projective coordinates of P, induced by the landmarks $L \subset \mathbb{M}$ and the cohomology class $[\mu] \in H^1(R_{2\alpha}(L); \mathbb{Z}/2)$.

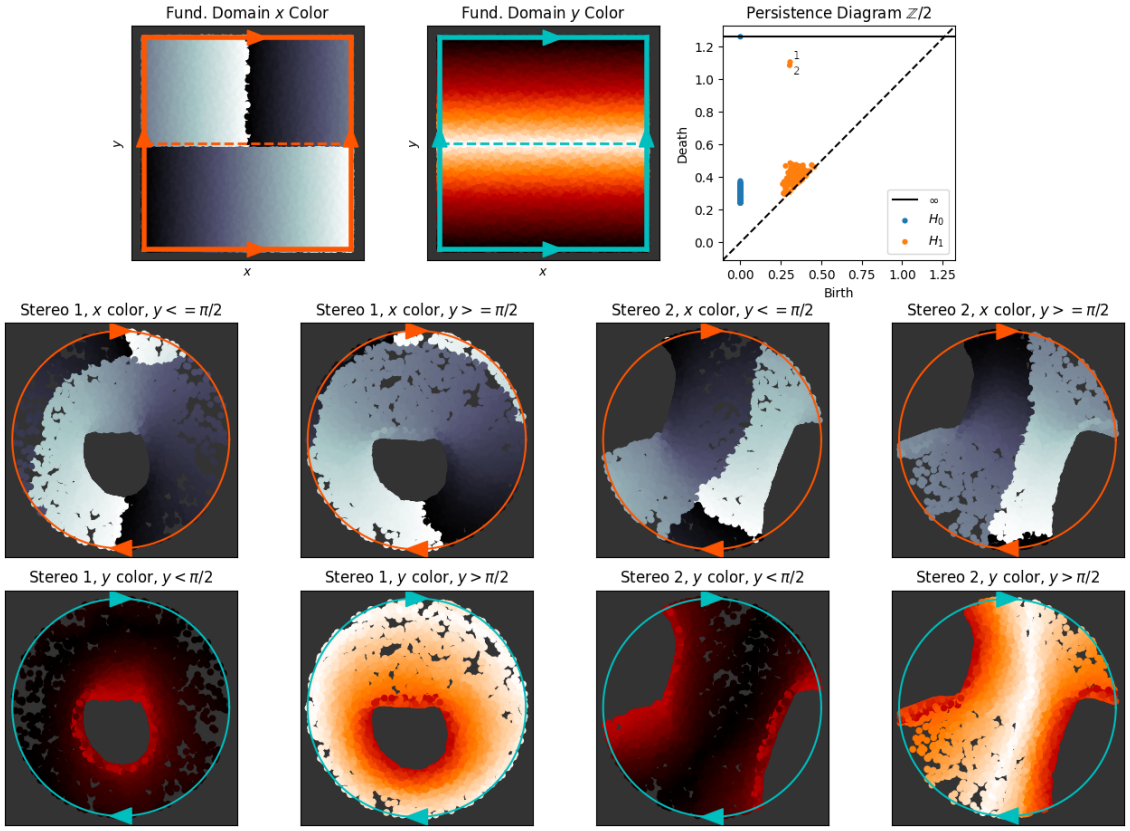

Figure 2.2: An example of projective coordinates for point sampled from a flat Klein bottle obtained as a quotient of the torus via $[x, y] \sim [x + \pi, -y]$. The two coordinates are colored according to their x and y positions on the fundamental domain $[0, 2\pi] \times [0, \pi]$, and we show two different stereographic projections to the plane from $\mathbb{R}\mathbf{P}^2$. If the fundamental domain is split into distinct parts $A = [0, 2\pi] \times [0, \pi/2]$ and $B = [0, 2\pi] \times [\pi/2, \pi]$, then A and B map to two distinct Möbius strips which are attached at their boundaries at $y = \pi/2$ (medium red for the y colors), which is indeed what happens when the Klein bottle is cut down the middle.

As an example, Figure 2.2 shows the projective coordinates onto $\mathbb{R}\mathbf{P}^2$ of points sampled from a Klein bottle $\mathbb{K}$, using the flat metric on the torus $\mathbb{T}$, descended onto the automorphism $\kappa : (x, y) \mapsto (x + \pi, -y)$. We use the cocycle representative which is the sum of the representative cocycles from the two most persistent classes.

In fact, this is a 2 to 1 map, as shown in Figure 2.2. Just as a torus can be obtained from gluing two annuli together at their boundary, the Klein bottle can be obtained by gluing two Möbius strips at their boundary. Each one of these Möbius strips is visible in the odd and even columns of the bottom two rows of Figure 2.2, respectively. In particular, the loops $[0, 2\pi] \times 0$ and $[0, 2\pi] \times \pi$ are at the center of each Möbius strip, and the boundaries of each Möbius strip at $[0, 2\pi] \times \pi/2$ get identified at the center of the projective coordinates plot.

We will observe similar projective coordinates for the sliding window of our Klein bottle time series in Section 2.5.

Circular Coordinates

The idea of using the bijection $H^1(B;\mathbb{Z}) \cong [B, S^1]$ to construct circle-valued functions for data, from persistent cohomology classes, was first introduced by de Silva et. al. [26]. Their construction has shortcomings (not sparse, not transductive) which are addressed in [86]; the latter is the procedure we use in the paper and the one we describe next.

Let $q > 2$ be a prime so that the homomorphism

$$H^1(R_{2\alpha}(L);\mathbb{Z}) \longrightarrow H^1(R_{2\alpha}(L);\mathbb{Z}/q)$$

induced by the projection $\mathbb{Z} \longrightarrow \mathbb{Z}/q$, is surjective. Hence, any $\mu \in Z^1(R_{2\alpha}(L);\mathbb{Z}/q)$ has a lift $\tilde{\mu} \in Z^1(R_{2\alpha}(L);\mathbb{Z})$. Moreover, if $\iota : \mathbb{Z} \hookrightarrow \mathbb{R}$ is the inclusion homomorphism, then there are cochains $\theta \in Z^1(R_{2\alpha}(L);\mathbb{R})$ and $\tau \in C^0(R_{2\alpha}(L);\mathbb{R})$ so that θ is the unique harmonic cocycle representative of $\iota^*([\tilde{\mu}])$ and $\iota^{\#}(\tilde{\mu}) = \theta - \delta^0\tau$. From this data we define

$$\begin{array}{rccl} f_{\theta,\tau}: & \bigcup\limits_{\ell\in L} B_\alpha(\ell) & \longrightarrow & S^1 \subset \mathbb{C} \\ & B_\alpha(\ell_j) \ni b & \mapsto & \exp\left\{2\pi i\left(\tau_j + \sum\limits_{k=0}^{n} \theta_{jk}\varphi_k(b)\right)\right\} \end{array}$$

where

$$\varphi_k(b) = \frac{|\alpha - \mathbf{d}(b,\ell_k)|_+}{\sum\limits_{r=0}^{n} |\alpha - \mathbf{d}(b,\ell_r)|_+}$$

Figure 2.3 shows an example of this algorithm on a point cloud sampled from a torus, using 400 landmarks. In this example, the algorithm is able to find maps from the points to the inner and outer circle of the torus.

2.3 Preliminary Examples: Distance To A Point As Observation Function

To motivate a more general development of good observation functions on manifolds, we first explore a very specific genre of observation functions: those which arise as the distance to a specified point in the manifold. We then verify the geometric integrity of a delay coordinate mapping of the resulting time series using persistent homology and Eilenberg-MacLane coordinates on a few examples. Through these tools and a visual comparison of the time series to known examples, we will already be able to explain quite a lot, including motivating both conditions of Theorem 2.4.1, though a full development of the theory in Section 2.4 is needed to justify these choices of observation functions.

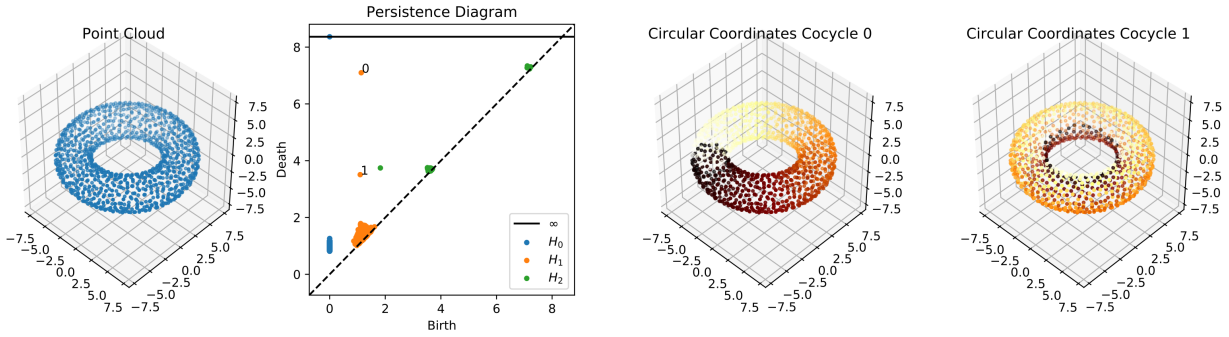

Figure 2.3: An example of the circular coordinates algorithm on a point cloud sampled from a torus in $\mathbb{R}^3$. The third plot shows the coordinates resulting from the representative cocycle of the largest persistence class, which goes around the large circle on the outside, while the fourth plot shows the circular coordinates resulting from the cocycle from the second largest persistence class, which wraps around the inner circle.

In the discussion below, all of our observation functions are of the form $G(x) = d(x, \hat{x})$, where d is some metric chosen on the manifold and $\hat{x}$ is some fixed point on the manifold which is our "reference distance point."

Example 2.3.1. Flat torus $\mathbb{T}$

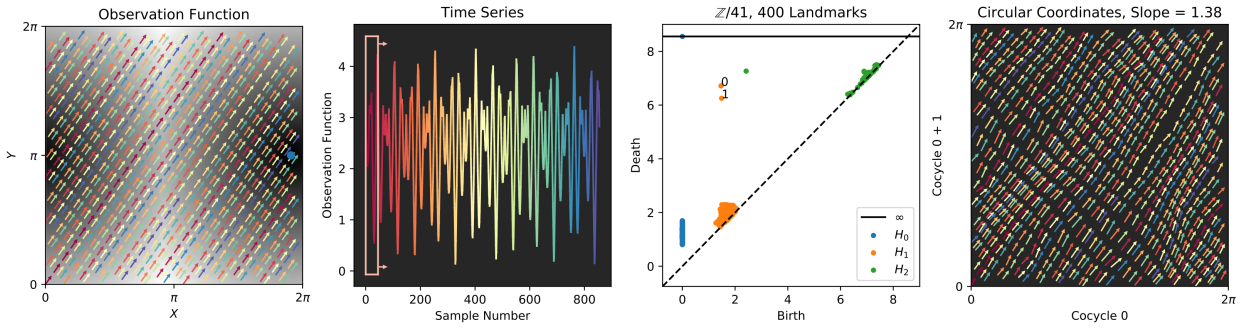

Figure 2.4: An irrational winding on the flat torus, with an observation function as the distance to the point $\hat{x} = (6, \pi)$, which is shown as a blue dot on the left plot. The distance from this point is indicated in gray (dark means close, light means far). The resulting time series is shown in the second plot, with a sliding window indicated with a red box. The third and fourth figures show, respectively, the persistence diagrams of the sliding window point cloud and the resulting circular coordinates. The arrows in the fourth plot are the recovered dynamics; they indicate the order on the sliding windows inherited from the time series. Colors are coordinated between the flows in the first, second, and fourth plots. Similar plotting conventions are present in Figures 2.5, 2.6, 2.7, 2.8, 2.9.

We first examine the planar torus $\mathbb{T} = \mathbb{R}^2/2\pi\mathbb{Z}^2$, parameterized by $(u, v) \in [0, 2\pi] \times [0, 2\pi]$. As our dynamics, we take the irrational winding $\psi_t(u, v) = (u + \sqrt{2}dt, v + dt)$, and the observation $G(u, v)$ is the flat geodesic distance between (u, v) and the point $\hat{x} = (6, \pi)$. This is shown in Figure 2.4. After performing a delay embedding on the resulting time series with window length of 30 samples, we see two persistent H_1 classes and 1 persistent H_2 class, which is the signature of a torus. Furthermore, circular coordinates resulting from the top two persistent classes in H_1 recovered the full original flow specification.

Example 2.3.2. Flat Klein Bottle $\mathbb{K}$

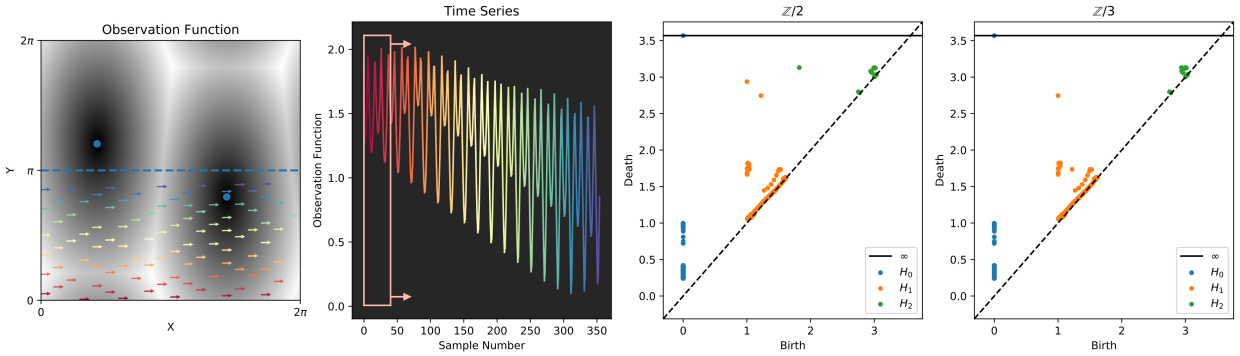

Figure 2.5: A winding with a very shallow slope on the fundamental domain of a flat Klein bottle, which is double covered by the flat torus by the automorphism $(x, y) \sim (x + \pi, -y)$. The observation function is then a scaled L^2 distance from the point $\hat{x} = (4.5, 2.5)$, which descends under the automorphism.

As in our projective coordinates example in Figure 2.2, we now form a quotient on the domain of the flat torus to create a Klein bottle, via the automorphism $\kappa : (x, y) \sim (x+\pi, -y)$. Then, the metric on the torus descends to the Klein bottle via κ. We use a slightly modified weighted L^2 flat metric as our distance measure for the observation function; that is

$$d_{\alpha,\beta}((u_1, u_1), (u_2, u_2)) = \sqrt{\alpha^2(u_1 - u_2)^2 + \beta^2(u_1 - u_2)^2} \tag{2.3}$$

In this particular example, we let $\alpha = 1$ and $\beta = 0.5$, and we take an observation to the point $\hat{x} = (4.5, 2.5)$; that is, $G(u, v) = d_{1,0.5}((u, v), (4.5, 2.5)$. Finally, we use a flow with a very shallow slope, $\psi_t(u, v) = (u + dt, v + 0.05dt)$, in the fundamental domain $y < \pi$. After performing a sliding window embedding with a window length of 30 samples, we see two persistent classes in H_1 and one persistent class in H_2 with $\mathbb{Z}/2$ coefficients, but we only see one class in H_1 and no classes in H_2 with $\mathbb{Z}/3$ coefficients. This is indeed the signature of a Klein bottle. We will show projective coordinates on a similar example with a slightly different observation function in Section 2.5, and we will explain more intuitively visual features of the time series at that point.

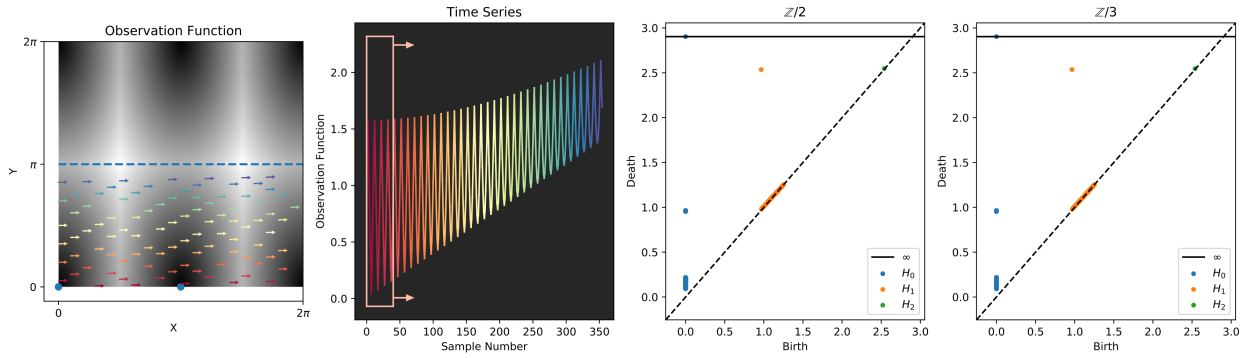

Figure 2.6: Not all distance functions on the Klein bottle work. The conditions here are the same as in Figure 2.5, but the point $\hat{x}$ from which distance is measured has been moved to $(\pi, 0)$. The sliding window embedding degenerates to a cylinder in this example.

Note that not every distance function will lead to a reconstruction of the Klein bottle. For instance, if we use the same flow ψ_t but an observation function $G(u, v) = d((u, v), (\pi, 0))$, as in Figure 2.6, then the sliding window embedding of the resulting time series degenerates to a cylinder, because there exist pairs of points with the same observation curves under the flow. This motivates condition 2 in Theorem 2.4.1.

Example 2.3.3. Sphere $\mathcal{S}^2$

We now reconstruct the sphere from a given trajectory and distance function. Tralie [113] showed empirically that a sliding window embedding of a helical trajectory, under the observation function on the sphere which is the arclength from some point on the sphere, yields an embedding of the sphere. We replicate this here. More specifically, we parameterize the unit sphere in spherical coordinates (φ, θ) (where φ is azimuth and θ is elevation from the north pole), we let $\psi_t(\varphi, \theta)_\alpha = (\varphi + dt, -\pi/2 + \theta dt)$, and the let the observation $G(\varphi, \theta)$ to a point $\hat{x} = (\hat{\varphi}, \hat{\theta})$ be

$$G(\theta, \varphi) = \cos^{-1}\Big(\cos(\varphi)\sin(\theta)\cos(\hat{\varphi})\sin(\hat{\theta}) + \sin(\varphi)\sin(\theta)\sin(\hat{\varphi})\sin(\hat{\theta}) + \cos(\theta)\cos(\hat{\theta})\Big) \tag{2.4}$$

We repeat this here in Figure 2.7. In this example, simple linear dimension reduction via PCA is able to recover the most of the geometry of the sliding window point cloud, though spherical coordinates are also possible in the Eilenberg-MacLane framework [83].

One pitfall in this example is that the observation point $\hat{x}$ cannot lie on the equator or the north or south poles; that is, $\hat{\varphi} \notin \{-\pi/2, 0, \pi/2\}$. In these cases, the helix structure is flattened to a spiral,so the sliding window embedding degenerates to a disc. This motivates the "derivative rank" condition, or condition 1 in Theorem 2.4.1.

Example 2.3.4. Projective plane $\mathbb{R}\mathbf{P}^2$

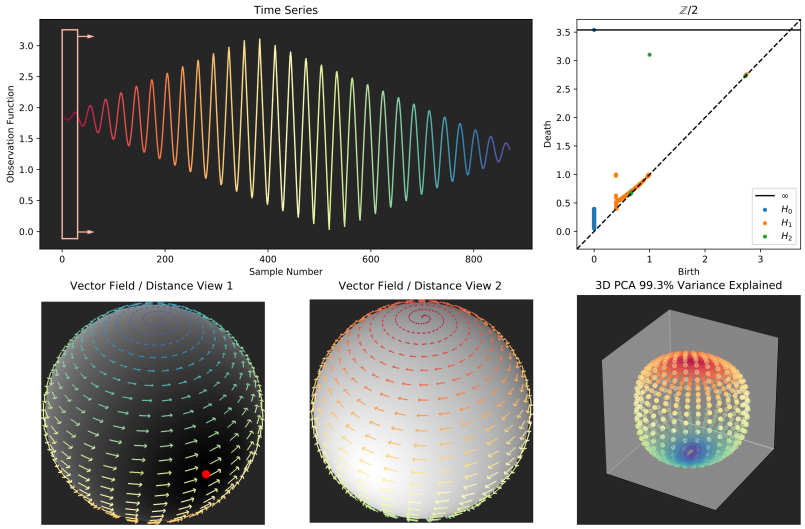

Figure 2.7: An observation function on the sphere which is the geodesic distance from a point $\hat{x}$ drawn in red. The top and bottom views of the vector field are drawn in the left two figures. 3D PCA of the sliding window embedding, which retains nearly all of the variance of the sliding window point cloud, is shown in the bottom right plot.

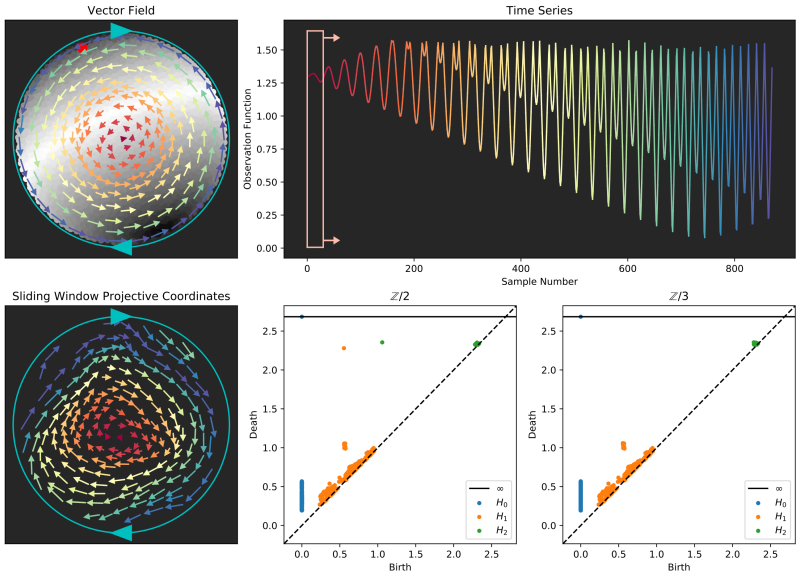

Figure 2.8: An observation function on $\mathbb{R}\mathbf{P}^2$ which is the geodesic distance from a point $\hat{x}$ drawn in red.

We can extend the scheme that we used in Example 2.3.3 to the projective plane $\mathbb{RP}^2$ by taking a flow only on the upper hemisphere and performing the antipodal identification at the equator $x \sim -x$. The flow ψ_t is the same, but the observation function changes to

$$G(\theta, \varphi) = \cos^{-1} \left| \cos(\varphi)\sin(\theta)\cos(\hat{\varphi})\sin(\hat{\theta}) + \sin(\varphi)\sin(\theta)\sin(\hat{\varphi})\sin(\hat{\theta}) + \cos(\theta)\cos(\hat{\theta}) \right| \tag{2.5}$$

Figure 2.8 shows this result, in which a single highly persistent point is present for both H_1 and H_2 using $\mathbb{Z}/2$ coefficients, but in which none are present for $\mathbb{Z}/3$, which is a correct signature of $\mathbb{R}\mathbf{P}^2$. Interestingly, the quotient identification is visible in the time series itself; the time series in Figure 2.8 can be obtained from the time series in Figure 2.7 by reflecting values above the line $y = \pi/2$ across that line. This is because the maximum distance between any two points on $\mathbb{R}\mathbf{P}^2$ is $\pi/2$. Additionally, both the sphere time series and the Möbius loop time series $(\cos(t) + a\cos(2t))$ are visible in Figure 2.8. The time series starts off in a spiral, which fills out a disc, and this disc transitions to a spiraling Möbius loop time series which fills out the strip. This visually reflects the fact that $\mathbb{R}\mathbf{P}^2$ is the connected sum of a disc and the boundary of a cross-cap. We will use a similar intuition to explain the Klein bottle time series in Section 2.5.

Example 2.3.5. Genus 2 surface $\mathbb{T}\#\mathbb{T}$

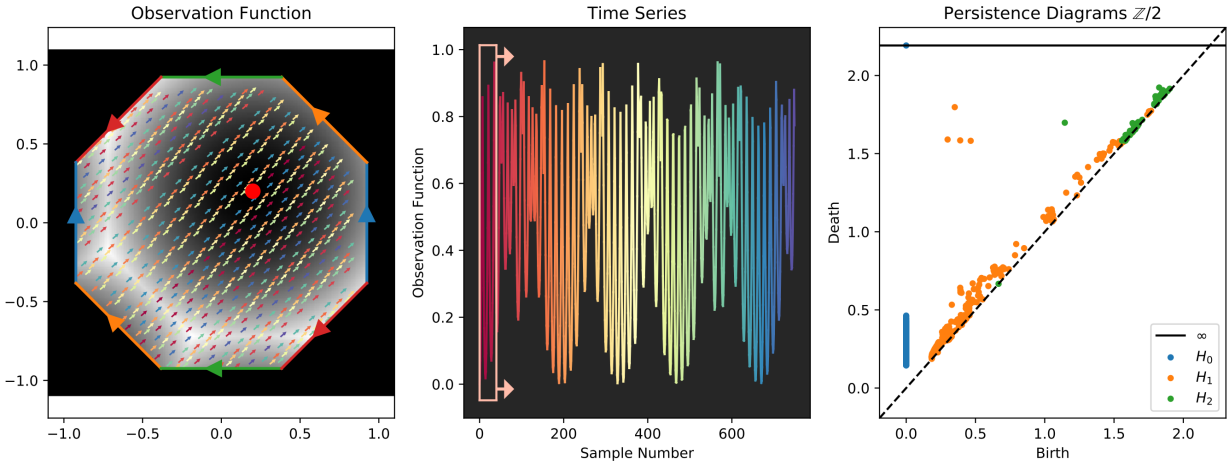

Figure 2.9: An example of a time series resulting from a dense flow on the 2-holed torus, using the flat squared Euclidean distance [126] from an observation point (shown as a red dot).

Finally, we show a time series whose sliding window embedding lies on the two holed torus. We use an irrational flow with slope $(dt, dt\sqrt{3}/2)$, with an observation function as the squared flat metric on the fundamental domain [126] represented by an octagon with opposite sides identified. Figure 2.9 shows the result, in which four highly persistent dots are

visible in H_1 and a single persistent dot is visible in H_2, matching what is expected of the homology of a genus 2 surface.

2.4 Main Theorem: Characterizing Good Observation Functions

As our main theoretical contribution, we now state more general conditions for good observation functions. Let M be a compact manifold of dimension m, $G : M \to \mathbb{R}$ a smooth function, and X a vector field with flow ψ_t. Applying G to an integral curve $\gamma_p(t) = \psi_t(p)$ through a point p yields a real-valued function

$$g_p := G \circ \gamma_p$$

in t, the *observation curve* of p. For sufficiently nice G and X, one can recover the point p from a finite uniform sampling of g_p. More precisely, the *Takens map* $\Psi_\tau^N : M \to \mathbb{R}^{N+1}$ defined by

$$\Psi_\tau^N(p) = \big(g_p(0), g_p(\tau), g_p(2\tau), \ldots, g_p(N\tau)\big)$$

is an embedding for some dimension $N > 0$ and flow time $\tau > 0$. For such G and X we say G is a *good observation* for X.

Motivation for the approach

As a simple example, take $M = S^1 = \mathbb{R}/2\pi\mathbb{Z}$, $\psi_t(x) = x + t$, and $G(x) = \cos(x)$. The point x is uniquely determined by sampling the two values $g_x(0) = \cos(x)$ and $g_x(\pi/2) = -\sin(x)$ and the Takens map

$$\Psi_{\pi/2}^1(x) = (\cos(x), -\sin(x))$$

is an embedding, so G is a good observation.

On the other hand, the doubly periodic function $G(x) = \cos(2x)$ is not a good observation function. Indeed, any integral curve $g_x(t)$ is invariant under a π-shift of x, as G cannot distinguish between any flow of x and $x + \pi$. In fact, the good observation functions on S^1 for the rotational dynamic are precisely ones with minimum period 2π

In higher dimensions the task of recovering p from g_p becomes less clear. Consider the torus $\mathbb{T} = S^1 \times S^1$ and $G : \mathbb{T} \to \mathbb{R}$ given by

$$G(x, y) = \cos(x) + \cos(y)$$

and ψ_t an irrational flow

$$\psi_t(x, y) = (x + \alpha t, y + \beta t)$$

and thus for $p = (x, y) \in \mathbb{T}$ we have the observation curve

$$g_p(t) = \cos(x + \alpha t) + \cos(y + \beta t)$$

For G to be good, there must be a τ such that each p is uniquely determined by sampling G along the integral curve γ_p at finitely many τ-steps. Since we are free to shrink τ and increase N, it is natural to examine infinitesimal changes of G along the flow ψ_t. The derivatives

$$\begin{aligned} g_p(0) &= \cos(x) + \cos(y) \\ g_p'(0) &= -\alpha \sin(x) - \beta \sin(y) \\ g_p^{(2)}(0) &= -\alpha^2 \cos(x) - \beta^2 \cos(y) \\ g_p^{(3)}(0) &= \alpha^3 \sin(x) + \beta^3 \sin(y) \end{aligned}$$

up to 3rd order yield the linear equation

$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ -\alpha^2 & -\beta^2 & 0 & 0 \\ 0 & 0 & -\alpha & -\beta \\ 0 & 0 & \alpha^3 & \beta^3 \end{pmatrix} \begin{pmatrix} \cos(x) \\ \cos(y) \\ \sin(x) \\ \sin(y) \end{pmatrix} = \begin{pmatrix} g_p(0) \\ g_p'(0) \\ g_p^{(2)}(0) \\ g_p^{(3)}(0) \end{pmatrix}$$

Equivalently, over $\mathbb{C}$, the linear system

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ i\alpha & -i\alpha & i\beta & -i\beta \\ -\alpha^2 & -\alpha^2 & -\beta^2 & -\beta^2 \\ -i\alpha^3 & i\alpha^3 & -i\beta^3 & i\beta^3 \end{pmatrix} \begin{pmatrix} e^{ix} \\ e^{-ix} \\ e^{iy} \\ e^{-iy} \end{pmatrix} = \begin{pmatrix} g_p(0) \\ g_p'(0) \\ g_p^{(2)}(0) \\ g_p^{(3)}(0) \end{pmatrix}$$

has invertible Vandermonde matrix and one can solve for e^{ix} and e^{iy}. Therefore (x, y) is uniquely determined by $g_p^{(k)}(0)$'s. Choosing τ small enough so that $g_p(\tau)$ is close to the 3rd order Taylor polynomial of g_p about 0, we see that p is uniquely determined (modulo 2π) by a τ-uniform finite sampling of g_p.

Main theorem and proof

The above calculation illustrates our approach to determining whether G is good: by studying the Taylor coefficients $g_p^{(k)}$. Note that $g_p^{(k)}(t)$ is the k-fold derivation of X applied to G at $\psi_t(p)$, i.e., in Lie derivative notation,

$$g_p^{(k)}(t) = \mathcal{L}_X^k G(\psi_t(p)).$$

$\mathcal{L}_X$ is the linear operator on tensor fields which measures infinitesimal change along X, i.e. if T is a tensor then

$$\mathcal{L}_X T(p) = \left.\frac{d}{dt}\right|_{t=0} ((\psi_{-t})_* T_{\psi_t(p)})$$

Writing dG for the differential of G, and $\wedge$ for exterior product, we now state the main result:

Theorem 2.4.1. *The Takens map Ψ_τ^N is an embedding for some $N > 0$ and flow time $\tau > 0$, if the following conditions hold:*

1. *For any point of $p \in M$ there is an m-tuple $J \in \mathbb{Z}_{\geq 0}^m$ of nonnegative integers such that the m-form*
$$\mathcal{L}_X^{\wedge J} dG := \bigwedge_{j \in J} \mathcal{L}_X^j dG$$
is nonzero at some point on the integral curve $\gamma_p(s)$.

2. *For any pair of distinct points $p, q \in M$ the observation curves $g_p(s)$ and $g_q(s)$ are not identical.*

Proof. For Ψ_τ^N to be an immersion, the cotangent vectors

$$dG|_p,\ d(G \circ \psi_\tau)|_p,\ d(G \circ \psi_{2\tau})|_p,\ \ldots,\ d(G \circ \psi_{N\tau})|_p \in T_p^* M$$

must span an m-dimensional space for all $p \in M$. Equivalently, for any point $p \in M$ there must be a strictly increasing m-tuple $I = (i_1, i_2, \ldots, i_m) \in \mathbb{Z}_{\geq 0}^m$ of indices such that the determinant m-form $\bigwedge d(G \circ \psi_{i_k \tau})$ does not vanish at p, i.e.

$$\omega_p^I(\tau) := \bigwedge_{i_k \in I} d(G \circ \psi_{i_k \tau})|_p \neq 0.$$

We require that I be strictly increasing because a wedge product containing identical factors is zero.

The idea is to perform a convolution of the Taylor series of the cotangent curves $d(G \circ \psi_{i_k} t)$ and to use condition 1 above to choose sufficiently small τ so that $\omega_p^I(\tau) \neq 0$ for some I. By compactness of M, one makes a uniform choice of small τ so that Ψ_τ is immersive and each observation curve is distinguished on some integer multiple of τ, thereby making Ψ_τ^N injective.

Let $s \geq 0$ be a time parameter for p such that

$$\mathcal{L}_X^{\wedge J} dG$$

is nonzero at $\gamma_p(s)$. Write $\tilde{p} = \gamma_p(s)$ and $\mathcal{J}_n$ for the set of all strictly increasing m-tuples $J = (j_1, j_2, \ldots, j_m)$ with degree

$$j_1 + j_2 + \ldots + j_m = n$$

satisfying

$$\mathcal{L}_X^{\wedge J} dG|_{\tilde{p}} \neq 0$$

Fix $n > 0$ to be the minimal integer for which $\mathcal{J}_n$ is nonempty (possible by condition 1 above).

Let $A(t)$ be the m by $(n+1)$ matrix with $(k,j)^{\text{th}}$ entry

$$A_{k,j}(t) = \frac{i_k^{j-1}(t-s)^{j-1}}{(j-1)!}$$

and $L : T_{\tilde{p}}M \to \mathbb{R}^{n+1}$ the linear map given by $\mathcal{L}_X^{j-1} dG|_{\tilde{p}}$ in the j^{th} coordinate,

$$L = \left(dG|_{\tilde{p}}, \mathcal{L}_X^1 dG|_{\tilde{p}}, ..., \mathcal{L}_X^n dG|_{\tilde{p}}\right).$$

So the k^{th} component of the composition $A(t) \circ L$, viewed as an m-tuple of t-dependent cotangent vectors, yields the n^{th} order Taylor polynomial about $t = s$ of the cotangent curve $d(G \circ \psi_{i_k t})|_p$:

$$A(t) \circ L = \sum_{j=0}^{n} \frac{i_k^j (t-s)^j}{j!} \mathcal{L}_X^j dG|_{\tilde{p}}.$$

By Cauchy-Binet formula applied to $A(t)$ and L, the top exterior product

$$\omega_p^I(t) = \bigwedge_{i_k \in I} d(G \circ \psi_{i_k t})|_p$$

has n^{th} order Taylor series expansion about $t = s$ with n^{th} coefficient

$$C_n = \frac{\det(V)}{a_n} \sum_{J \in \mathcal{J}_n} |I^J| \cdot \mathcal{L}_X^{\wedge J} dG|_{\tilde{p}}$$

where

- a_n is a nonzero constant depending only on n
- $|I^J| = \prod i_k^{j_k - k + 1}$

and

$$\det(V) = \prod_{k<k'} (i_{k'} - i_k) \neq 0$$

is the nonzero determinant of the $m \times m$ Vandermonde matrix V with $(k,j)^{\text{th}}$ entry

$$V_{k,j} = {i_k}^{j-1}$$

where we take $0^0 = 1$.

By the minimality assumption on $\mathcal{J}_n$, all the lower degree Taylor coefficients, which contain $\mathcal{L}_X^K dG|_{\tilde{p}} = 0$ for m-tuples K with degree strictly less than n,

$$C_j = 0 \text{ for } j < n$$

are zero. So the Taylor expansion of $\omega_p^I(t)$ has the form

$$\omega_p^I(t) = (t-s)^n C_n + R_p^n(t)$$

where $R_p^n(t)$ is the n^{th} order Taylor error term with vanishing limit

$$\lim_{t\to s} \frac{R_p^n(t)}{(t-s)^n} = 0$$

For for suitable choice of I, there will be a dominating term in the sum over $\mathcal{J}_n$ such that C_n is nonzero. For $\tilde{J} \in \mathcal{J}_n$ the colexigraphically maximal element of $\mathcal{J}_n$, let a be the maximal index such that $j_a < \tilde{j}_a$, for all $J < \tilde{J}$. Choose I by making all terms right of $a-1$ large, so that

$$|I^J| << |I^{\tilde{J}}|$$

for all $J < \tilde{J}$

$$\sum_{J\in\mathcal{J}_n} |I^J| \cdot \mathcal{L}_X^{\wedge J} dG|_{\tilde{p}} \neq 0.$$

So the n^{th} Taylor coefficient

$$C_n \neq 0$$

is nonzero. Hence we may choose a time $\eta > s$ sufficiently close to s so that the Taylor error $R_p^n(\eta)$ is small and the inequality

$$\omega_q^I(\eta) \neq 0$$

holds for all q in a neighborhood of p, and this property remains invariant under shrinking η closer to s. By compactness of M there is a finite collection of triples (I_r, η_r, s_r) such that the collection of m-forms

$$\{\omega^{I_r}(\eta_r)\}$$

do not all vanish at any given point of M and the cotangent vectors

$$\{d(G \circ \psi_{i_k\eta_r})|_q\}_{i_k\in I_r}$$

specified by I_r are linearly independent. Choose $\tau > 0$ small enough so that there is an integer multiple of τ lying in the interval (s_r, η_r) for each r. Then the Takens map Ψ_τ^N is an immersion for all $N > 0$ bounding I_r and η_r/τ.

So Ψ_τ^N is locally injective and the difference map

$$\Psi_\tau^N(p) - \Psi_\tau^N(q)$$

does not vanish for all $p \neq q$ in an open neighborhood U of the diagonal in $M \times M$, and this property is invariant under scaling $N \mapsto Nd$ and $\tau \mapsto \tau/d$ for an integer $d > 0$ (with U fixed).

For distinct $(p, q) \in M \times M \setminus U$, we may shrink τ so that g_p and g_q are distinguished on some integer multiple of τ and $\Psi_\tau(p) \neq \Psi_\tau(q)$. By compactness of $M \times M \setminus U$, there is a uniform choice of τ and N making Ψ_τ^N injective, hence an embedding.

□

Remark 2.4.2. While one can provide a lower bound for the dimension N needed to yield a Takens embedding, the formula depends in a complicated way on G and X. In practice, choosing sufficiently large N and small τ amounts to a dense sampling of a discrete time series.

2.5 An Application to Surfaces via Fourier theory

Now that we have our theory in hand, we can examine another class of observation functions which are constructed from Fourier modes, in addition to our distance-based observation functions in Section 2.3.

The Torus

We start by characterizing all smooth observations $G : \mathbb{T} \to \mathbb{R}$ for a vector field X of irrational flow

$$\psi_t(x, y) = (x + \alpha t, y + \beta t)$$

yielding toroidal delay embedding. For G write the Fourier expansion

$$G(x, y) = \sum_{(n,m)\in\mathbb{Z}^2} \hat{G}(n, m) \cdot \exp(i(nx + my))$$

where $\hat{G}(n, m) \in \mathbb{C}$ is the $(n, m)^{\text{th}}$ Fourier coefficient of G. Set

$$\operatorname{Supp} \hat{G} = \{(n, m) \in \mathbb{Z}^2 \mid \hat{G}(n, m) \neq 0\}$$

the *support* of $\hat{G}$.

Theorem 2.5.1. *A smooth function $G : \mathbb{T} \to \mathbb{R}$ is a good observation for an irrational winding if and only if the support $\operatorname{Supp} \hat{G}$ of the Fourier coefficients generates $\mathbb{Z}^2$ as an abelian group.*

Proof. Write $e_{n,m} = \exp(i(nx + my))$ for the $(n, m)^{\text{th}}$ Fourier basis element. The k-fold Lie derivative $\mathcal{L}_X^k G$ has Fourier coefficient

$$\widehat{\mathcal{L}_X^k G}(n, m) = i^k(n\alpha + m\beta)^k \cdot \hat{G}(n, m)$$

and thus Fourier expansion

$$\mathcal{L}_X^k G = \sum_{(n,m)\in\mathbb{Z}^2} i^k(n\alpha + m\beta)^k \hat{G}(n, m) \cdot e_{n,m}$$

Since α/β is irrational, the coefficients

$$c_{n,m} = i \cdot (n\alpha + m\beta)$$

are nonvanishing and pairwise distinct. Therefore the Vandermonde matrix with $(n,m) \times j^{\text{th}}$ entry

$$(c_{n,m}^k)$$

is nonsingular and the projection

$$G * e_{n,m} = \hat{G}(n,m) \cdot \exp(i(nx+my))$$

can be written as an infinite sum

$$\hat{G}(n,m) \cdot \exp(i(nx+my) = \sum_{j=0}^{\infty} b_j \mathcal{L}_X^j G \tag{2.6}$$

Hence the values of $\mathcal{L}_X^k G$ on a point $(u,v) \in \mathbb{T}$ uniquely determine

$$\hat{G}(n,m) \cdot e^{i(nu+mv)}$$

If $\operatorname{Supp} \hat{G}$ generates $\mathbb{Z}^2$, then there is some finite product

$$\prod_{(n_j,m_j) \in \operatorname{Supp} \hat{G}} e^{i(n_j u + m_j v)} = e^{iu}$$

and thus u, and similarly v, are uniquely determined modulo 2π by the observation curve $G \circ \gamma_{u,v}$ and condition 2 of Theorem 2.4.1 above is satisfied.

If $d\mathcal{L}_X^j G \wedge d\mathcal{L}_X^k G$ vanishes at p for all $j,k \geq 0$, then by equation 2.6 above, the 2-form

$$d(G * e_{n,m}) \wedge d(G * e_{n',m'}) = \det \begin{pmatrix} n & m \\ n' & m' \end{pmatrix} \hat{G}(n,m)\hat{G}(n',m') \cdot e_{n,m} e_{n',m'}$$

also vanishes at p for all pairs $(n,m),(n',m') \in \mathbb{Z}^2$. Thus

$$\det \begin{pmatrix} n & m \\ n' & m' \end{pmatrix} = 0 \text{ for all } (n,m),(n',m') \in \operatorname{Supp} \hat{G}$$

and $\operatorname{Supp} \hat{G}$ cannot generate $\mathbb{Z}^2$. So condition 1 of Theorem 2.4.1 is satisfied if $\operatorname{Supp} \hat{G}$ generates $\mathbb{Z}^2$.

Conversely, suppose $\operatorname{Supp} \hat{G}$ does not generate $\mathbb{Z}^2$. By the classification of finitely generated abelian groups, there is a $\mathbb{Z}$-basis

$$(n_1,m_1),(n_2,m_2)$$

for $\mathbb{Z}^2$ such that $\operatorname{Supp} \hat{G}$ is generated by

$$a \cdot (n_1,m_1), b \cdot (n_2,m_2)$$

where a and b are integers not both ± 1. Then there is some $(u, v) \notin 2\pi\mathbb{Z}^2$ such that

$$\begin{pmatrix} an_1 & am_1 \\ bn_2 & bm_2 \end{pmatrix} \cdot \begin{pmatrix} u \\ v \end{pmatrix}$$

takes values in $2\pi\mathbb{Z}$, so that $\exp(i(nu + mv)) = 1$ for all $(n, m) \in \operatorname{Supp} \hat{G}$. So for any point $(x, y) \in \mathbb{T}$, $(x + u, y + v) \in \mathbb{T}$ is a distinct point with the same observation curve, and no Takens map can distinguish between (x, y) and $(x + u, y + v)$. □

Remark 2.5.2. Theorem 2.5.1 can be strengthened to include rational windings. In this case one cannot expect the delay mapping to recover all Fourier modes of an observation function, but only those which are coprime to the slope of the winding.

Remark 2.5.3. For the irrational winding on the torus, the Koopman eigenfunctions are given by the Fourier basis. The Vandermonde inversion in equation (2.6) above shows that the Fourier modes of an observation are determined by its delay mapping. We are not aware of such a connection between Takens and Koopman, though it seems natural in this context.

By Theorem 2.5.1, whether or not G is good for an irrational flow depends only on the support $\operatorname{Supp} \hat{G}$. The quasiperiodic function

$$g(t) = \cos\sqrt{2}t + \cos t \tag{2.7}$$

is the observation of $G(x, y) = \cos(x) + \cos(y)$ along the irrational flow $(\sqrt{2}t, t)$ on the planar torus $\mathbb{T} = \mathbb{R}^2/2\pi\mathbb{Z}^2$. A point cloud densely sampled from the sliding window $\mathrm{SW}_1^{10} g(t)$ coordinates given by 10 uniform shifts of $g(t)$ yields a curve in $\mathbb{R}^{10}$ with toroidal persistence.

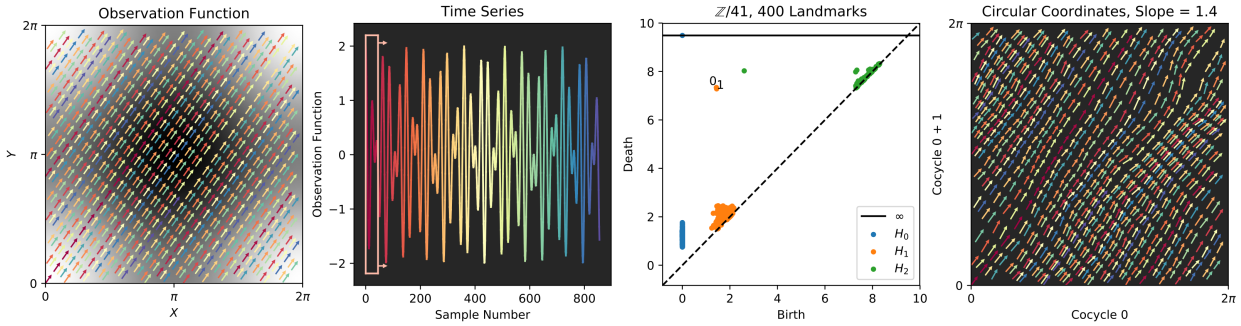

Figure 2.10: The observation function $\cos(x) + \cos(y)$ for the same flow as Figure 2.4

The Klein bottle

As in our example in Section 2.2, we write the Klein bottle $\mathbb{K}$ as the quotient of the torus $\mathbb{T}$ by the automorphism $\kappa : (x, y) \mapsto (x + \pi, -y)$. The irrational flow on the $\mathbb{T}$ is not κ-invariant

since κ is orientation reversing in the y coordinate. To approximate the shallow flow in Figures 2.5 and 2.6 above, we construct a vector field which flows cyclically along a repellor $y = 0$ and an attractor $y = \pi$ by restricting a linear flow to the fundamental domain $[0, 2\pi] \times [0, \pi]$ and flatten it out on the boundary circles $y = 0, \pi$. For $\alpha, \beta \in \mathbb{R}$ with $0 < \alpha/\beta << 1$ irrational, let X_ϵ be a vector field on the rectangle given by

$$X_\epsilon(x, y) = \begin{cases} (\alpha, \rho(y)) & 0 \leq y \leq \epsilon \\ (\alpha, \beta) & \epsilon < y \leq \pi - \epsilon \\ (\alpha, \rho(\pi + \epsilon - y)) & \pi - \epsilon < y \leq \pi \end{cases}$$

where ρ is a smooth function on a neighborhood of $[0, \epsilon]$ with $\rho(0) = 0$, $\rho(\epsilon) = \beta$ making X_ϵ smooth. For example, $\rho = \beta \exp(1/(y/\epsilon - 1)^2 - 1)$. Then X_ϵ extends uniquely to a κ-invariant vector field on on $\mathbb{T}$, and therefore induces a vector field on $\mathbb{K}$.

Theorem 2.5.4. *Let $G : \mathbb{T} \to \mathbb{R}$ be a κ-invariant function on $\mathbb{T}$. For fixed $N\tau$, the Takens map*

$$\Psi_\tau^N : \mathbb{K} \to \mathbb{R}$$

induced by G and X_ϵ for arbitrarily small ϵ and slope $\alpha/\beta << 1$ is an embedding if and only if the following conditions hold:

1. *$G(x, \pi)$ and $G(x, 0)$ have period π in x and do not differ by a shift*
2. *$\operatorname{Supp} \hat{G}$ generates $\mathbb{Z}^2$*

Proof. Suppose G is good for X_ϵ. Since X_ϵ flows horizontally at $y = 0, \pi$, condition 1) must hold so that each point is uniquely determined by its observation curve. Condition 2) must hold as well, since X_ϵ is given by an irrational winding away from the ϵ-neighborhood of $y = 0, \pi$ and the same argument as in Theorem 2.5.1 above applies for sufficiently shallow slope α/β because $N\tau$ is fixed.

Conversely, suppose conditions 1) and 2) hold. X_ϵ is given by an irrational flow away from the ϵ-neighborhood of $y = 0, \pi$. Furthermore, any point in the ϵ-strip with $y \neq 0, \pi$ may be flowed to a point where X_ϵ has irrational slope. The same argument as in Theorem 2.5.1 shows that the Takens map restricts to an embedding on $y \neq 0, \pi$.

By condition 1, the observation curve of a point (x, y) where $y = 0, \pi$ uniquely determines x modulo π, and is periodic and therefore distinct from any observation curve for $y \neq 0, \pi$. So each point is uniquely determined by its observation curve as per condition 2) of Theorem 2.4.1.

It remains to show that the Takens map is immersive at $y = 0, \pi$. If not, then $\frac{\partial G}{\partial y}$ vanishes on the circles $y = 0, \pi$, a neighorhood about which Ψ_τ^N would fail to immerse, a contradiction. □

According to Theorem 2.5.4, the "simplest" κ-symmetric good observation is

$$G(x, y) = \cos 2x + \cos x \sin y + \cos y. \tag{2.8}$$

Indeed, the Fourier coefficients of G are supported at $(\pm 2, 0), (\pm 1, \pm 1), (0, \pm 1)$, which generates $\mathbb{Z}^2$. Along the limit cycles we have $G(x, 0) = \cos 2x + 1$ and $G(x, 0) = \cos 2x - 1$, which are distinct and doubly periodic.

Intuitively, the $\cos 2x$ term is responsible for delay-mapping the limit cycles $y = 0, \pi$ via a double covering. Without this term, the boundary $G(x, 0) = 1$, $G(x, \pi) = -1$ along the bottom and top boundaries, respectively. Not only are these boundaries no longer identified, but they also each map to a single point, turning the Klein bottle into a sphere. The delay mapping of $\cos x \sin y$ fills two Möbius strips in conjuction with $\cos(2x)$, while the $\cos(y)$ term serves to "separate" the Möbius strips, as shown in the right hand side of Figure ??.

We can also see this by parameterizing the flow by a single variable $t = x$ and examining the time series directly. In this case, the time series is

$$g(t) = \cos(2t) + \cos(t) \sin\left(\frac{\alpha}{\beta} t\right) + \sin\left(\frac{\alpha}{\beta} t\right) \tag{2.9}$$

for $\epsilon < \frac{\alpha}{\beta} t < \pi - \epsilon$. Over small ranges of t, the sine terms are approximately constant. The time series is then of the form $\cos(2t) + a \cos(t), |a| < 1$; that is, its sliding window embedding locally parameterizes the boundary of a Möbius strip [88]. As it moves further along, a changes, and so it fills out the strip.

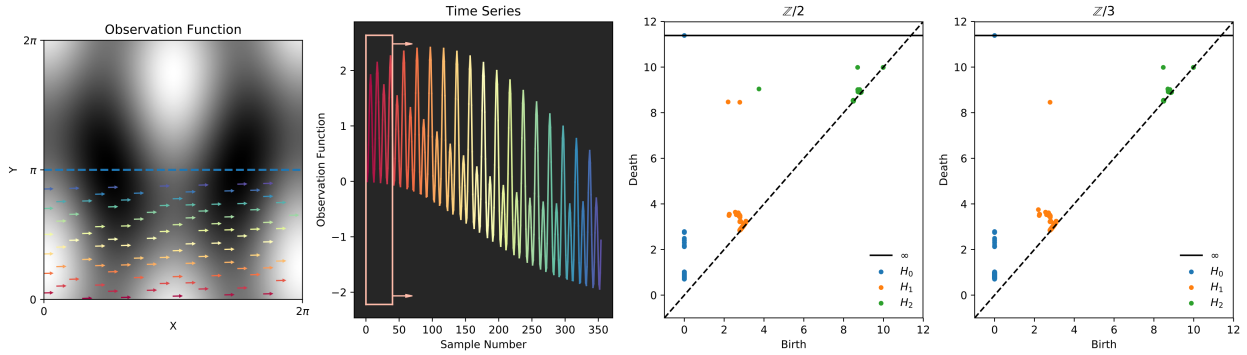

Figure 2.11: The observation function $G(x, y)$ from Equation 2.8 for the same flow as Figure 2.5

. Indeed the good observation function reproduces $\mathbb{K}$, as evidenced by the persistence diagram.

2.6 Discussion

It is clear what circular and toroidal observations look like in the time domain, and as we have mentioned, there are many applications that take advantage of this knowledge. The theory developed in this paper has enabled us to move beyond this and to develop examples of signals recovering other manifolds.

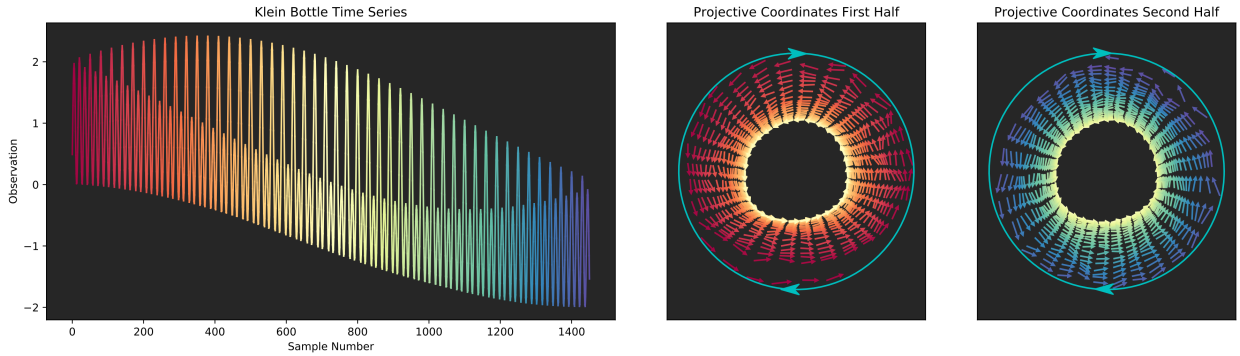

Figure 2.12: Projective coordinates for a Klein bottle with observation function specified in Equation 2.8. We plot the first half in the second subplot, which traces a Möbius strip from its core to its boundary, shown in yellow. Then, that boundary is glued to a second Möbius strip, which corresponds to the second half of the time series, as shown on the right.

Also, by showing the existence of time series whose "attractors" are on twisted spaces, we also provide further motivation for TDA time series users to move beyond exclusively using $\mathbb{Z}/2\mathbb{Z}$ in TDA. The latter is the default option across most applications of TDA in time series analysis, but it is possible that these pipelines are blind to important features, as some of our examples show.

Moreover, just as circular and toroidal sliding window embeddings have interpretations in terms of physical phenomena, the presence of Klein bottles, Moebius strips, spheres, projective planes, etc, should also have practical meaning. It is unlikely that one could recognize the significance of these time series in the wild without such examples in hand, and being primed as such makes it more likely that we will be able to discover physical examples where non-orientable state spaces are natural.

Finally, we note that not only do we have a method for producing time series recovering other manifolds, which we have validated empirically using persistent homology and Eilenberg MacClane coordinates, but the method is backed by a theorem that indicates exactly when it will succeed/fail.

Chapter 3

Concluding remarks to Chapter 2 and Transition to Chapter 4

In Chapter 2, we delved into the geometric framework of delay coordinate embeddings and the conditions under which they can be used to reconstruct the state space of dynamical systems. This exploration, detailed in the paper "Twisty Takens: A Geometric Characterization of Dynamical Observations and Delay Coordinate Embeddings," provided significant insights into how high-dimensional embeddings can be effectively reduced using techniques such as persistent cohomology and Eilenberg-MacLane coordinates. These methods are crucial for simplifying the analysis of complex biological data, enabling us to uncover underlying structures and dynamics that are otherwise obscured by the high dimensionality of the data.

The insights gained from Chapter 2 set the stage for our next exploration into the application of persistent cohomology, this time within the context of neural data. The brain's spatial representation system offers a fascinating arena where low-dimensional manifolds emerge from high-dimensional neural activity. Understanding these manifolds is key to decoding how the brain encodes information and performs computations.

In Chapter 4, we will present the paper "Evaluating State Space Discovery by Persistent Cohomology in the Spatial Representation System," which shifts our focus from the theoretical foundations of delay embeddings to practical applications in neuroscience. This chapter will explore how persistent cohomology can be used to uncover the topological structures in neural recordings, particularly within the brain's spatial representation system.

The spatial representation system of the brain includes various neural populations, such as grid cells, head direction cells, and conjunctive cells, which encode spatial information in a periodic manner. These cells operate within a high-dimensional phase space, yet their activity often lies on low-dimensional manifolds with nontrivial topological structures [40, 39, 54, 97]. For example, grid cells fire in a triangular lattice pattern, creating a toroidal structure in the neural activity space [47, 105]. Persistent cohomology allows us to detect and analyze these topological features by examining how the shape of the data changes across different scales [30, 32, 100].

The ability to decode an animal's trajectory from neural activity is a powerful demon-

stration of how topological data analysis can be applied in neuroscience. By using persistent cohomology, we can identify the low-dimensional manifolds sampled by the data and extract meaningful information about the animal's movement and orientation. This technique has shown promise in both simulated and experimental neural recordings, providing insights into the organization and function of spatial representation circuits [93, 17, 41].

The study presented in Chapter 4 systematically evaluates the performance of persistent cohomology in detecting topological structures within neural data. Through comprehensive simulations, we explore how various factors, such as dataset dimensions, spatial tuning variations, and noise, affect the success of topological discovery. We also investigate the conditions under which combinations of neural populations form product topologies, further enhancing our understanding of the brain's encoding mechanisms.

By bridging the theoretical foundations laid out in Chapter 2 with the practical applications discussed in Chapter 4, this dissertation aims to demonstrate the versatility and effectiveness of dimensionality reduction techniques in diverse biological contexts. The methods developed and refined here advance our understanding of dynamical systems and provide valuable tools for analyzing complex neural data.

As we transition into Chapter 4, we will delve deeper into the intricacies of the brain's spatial representation system, exploring how persistent cohomology can reveal the hidden topological structures that underpin neural computations. This investigation will highlight the capabilities of topological data analysis and its potential to enhance our understanding of neural systems and their functions.

Chapter 4

Evaluating state space discovery by persistent cohomology in the spatial representation system

The contents of this chapter are based on a publication of Louis Kang et al [61].

4.1 Abstract

Persistent cohomology is a powerful technique for discovering topological structure in data. Strategies for its use in neuroscience are still undergoing development. We comprehensively and rigorously assess its performance in simulated neural recordings of the brain's spatial representation system. Grid, head direction, and conjunctive cell populations each span low-dimensional topological structures embedded in high-dimensional neural activity space. We evaluate the ability for persistent cohomology to discover these structures for different dataset dimensions, variations in spatial tuning, and forms of noise. We quantify its ability to decode simulated animal trajectories contained within these topological structures. We also identify regimes under which mixtures of populations form product topologies that can be detected. Our results reveal how dataset parameters affect the success of topological discovery and suggest principles for applying persistent cohomology, as well as persistent homology, to experimental neural recordings.

4.2 Introduction

The enormous number of neurons that constitute brain circuits must coordinate their firing to operate effectively. This organization often constrains neural activity to low-dimensional manifolds, which are embedded in the high-dimensional phase space of all possible activity patterns [40, 39, 54, 97]. In certain cases, these low-dimensional manifolds exhibit nontrivial topological structure [23]. This structure may be imposed externally by inputs that are

periodic in nature, such as the orientation of a visual stimulus or the direction of an animal's head. It may also be generated internally by the network itself; for example, the grid cell network constructs periodic representations of physical space which outperform non-periodic representations in several ways [35, 102, 70, 104, 122, 94, 76]. In either case, detecting and interpreting topological structure in neural data would provide insight into how the brain encodes information and performs computations.

One promising method for discovering topological features in data is persistent cohomology [30, 32, 100]. By tracking how the shape of the data changes as we examine it across different scales—thickening data points by growing balls around them—persistent cohomology detects prominent topological features in the data, such as loops and voids. This knowledge helps to identify the low-dimensional manifolds sampled by the data, and in particular to distinguish between tori of different intrinsic dimensions. Furthermore, it enables parametrization of the data and navigation of the underlying manifolds.

We characterize how persistent cohomology can discover topological structure in neural data through simulations of the brain's spatial representation system. This system contains several neural populations whose activity exhibits nontrivial topology, which we term *periodic neural populations* (Fig. 1A). Grid cells fire when an animal reaches certain locations in its environment that form a triangular lattice in space [47]. In each animal, grid cells are partitioned into 4–10 modules [105]. Within each module, grid cells share the same scale and orientation but their lattices have different spatial offsets. Modules appear to increase in scale by a constant ratio and exhibit differences in orientation [105, 67]. Head direction cells fire when an animal's head is oriented in a certain direction relative to its environment [110]. They respond independently of the animal's position. Finally, conjunctive grid $\times$ head direction cells respond when an animal is located at the vertices of a triangular lattice and is oriented in a certain direction [95]. Like grid cells, conjunctive cells are also believed to be partitioned into modules.

We also consider neural populations whose activity exhibits trivial topology, which we will term *non-periodic neural populations* (Fig. 1A). Place and non-grid spatial cells are part of the spatial representation system, and they fire in one or multiple regions of the environment [79, 28, 48]. These two populations are found in different brain regions, and the former tend to have sharper spatial selectivity compared to the latter. Finally, we simulate neurons with irregular activity that exhibits no spatial tuning. We imagine these *random cells* may be responding to non-spatial stimuli or representing internal brain states

Persistent cohomology, as well as the closely related technique persistent homology, has recently been applied to experimental neural recordings within the spatial representation system. It was used to discover topological structure [93, 17] and decode behavioral variables [93] from head direction cells. It was also used to do the same for grid cell recordings [41], and researchers have demonstrated topological discovery in simulated grid cell data [17]. These works have improved our understanding of the large-scale organization of spatial representation circuits through persistent cohomology.

In contrast to the research described above, we aim to comprehensively explore the capabilities of persistent cohomology for simulated datasets. With complete control over the

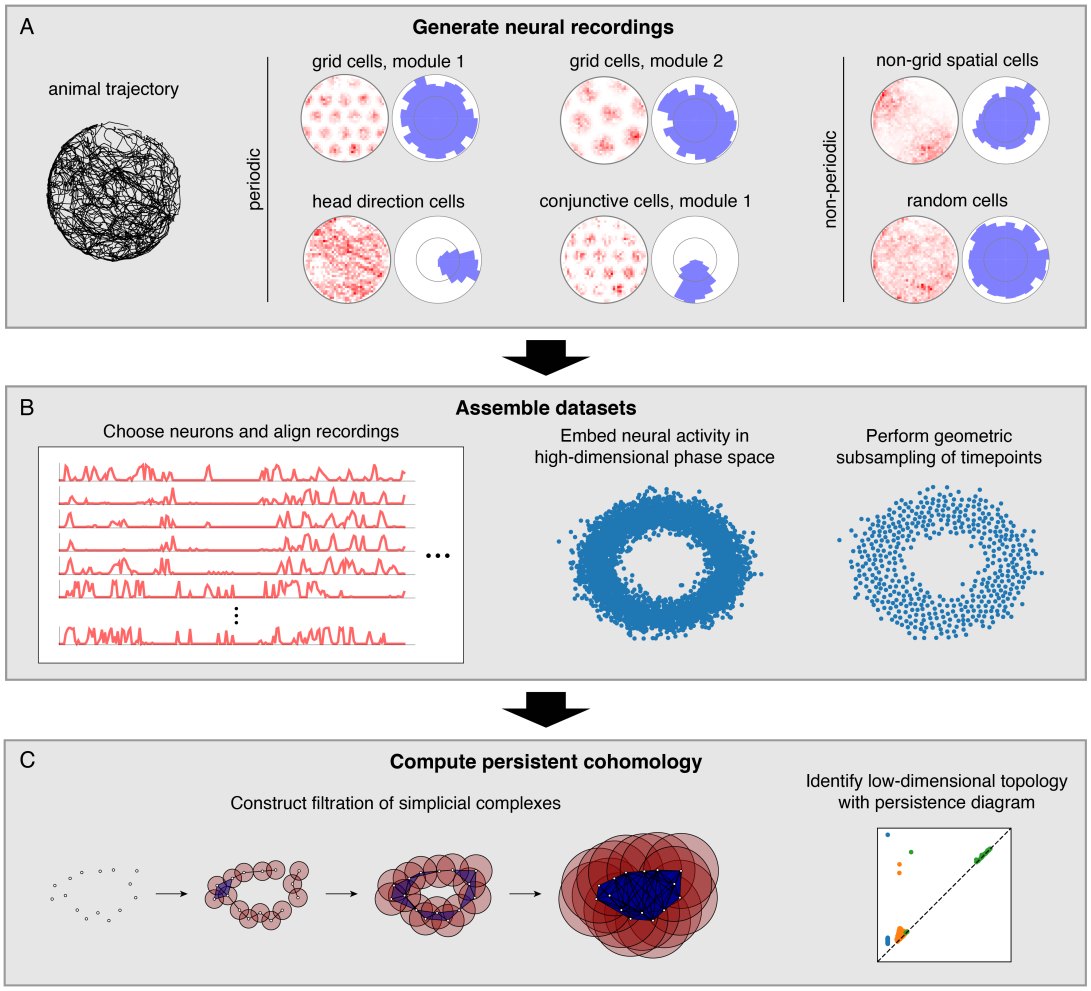

Figure 4.1: Pipeline for simulations and data analysis. (**A**) We generate activities for multiple neural populations along an experimentally recorded rat trajectory. For each population, we plot activity maps as a function of position (left) and direction (right) for one example neuron. (**B**) Then we choose neurons for topological analysis and form a high-dimensional vector of their firing rates at each timepoint along the trajectory. For computational tractability, we eliminate the most redundant points using a geometric subsampling algorithm. (**C**) We compute persistent cohomology on these subsampled timepoints to identify low-dimensional topological structure.

data, we can identify features that improve topological discovery and features that disrupt it. We can also freely generate datasets with varied quantities and proportions of different neural populations. A greater number of neurons embeds underlying activity manifolds in higher dimensions, which can strengthen the signal. However, experimental limitations

impose bounds to this number. Our simulations allow us to evaluate persistent cohomology in regimes currently accessible by experiments, as well as in regimes that may soon become experimentally tractable due to advances in recording technology [58].

4.3 Results

Overview of methods and persistence diagrams

In this work, we simulate neural populations within the spatial representation system, prepare the simulated data for topological analysis, and compute persistent cohomology to discover topological structure within the data (Fig. 1). We will now briefly describe each of these three stages; a complete explanation is provided in the Methods section.

To generate neural recordings, we define tuning curves as a function of position and direction. For each grid module, we first create a triangular lattice in space. Each grid cell has peaks in its positional tuning curves at a randomly chosen offset from each lattice point. Its directional tuning curve is uniform. Head direction cells have peaks in their directional tuning curves at a randomly chosen angle and have uniform positional tuning curves. Conjunctive cells have positional tuning curves like grid cells and directional tuning curves like head direction cells. We describe tuning curves for the non-periodic neural populations in the Methods section.

These tuning curves are applied to an experimentally extracted trajectory of a rat exploring its circular enclosure, producing an activity, or firing rate, for each neuron at 0.2 s intervals. This simulates the simultaneous recording of a large number of neurons from the medial entorhinal cortex and the binning of their spikes into firing rates. The time series spans 1000 s, or 5000 data points. Figure 1A shows examples of these time series data mapped back onto spatial coordinates. Next, we choose a subset of these neurons and pre-process it for topological data analysis (Fig. 1B). We form a vector of neural activities at each timepoint, which produces a point cloud in high-dimensional phase space. We wish to subsample it for computation tractability while maintaining as much evidence of topological structure embedded within it. To do so, we use a geometric subsampling algorithm that roughly eliminates the most redundant points, reducing the 5000 timepoints down to 1000.

Finally, we apply persistent cohomology to this subsampled point cloud (Fig. 1C). We describe this technique colloquially here, in terms of its dual, persistent homology. Both produce the same persistence diagrams, but we use cohomology throughout the paper both because it is faster to compute and because it allows us to parametrize the data. See the Methods section for a precise description. From the point cloud, we form a Vietoris–Rips filtration, which is a nested sequence of simplicial complexes. Each complex consists of all cliques in the near-neighbor graph, which contains all edges between points at distance at most r apart. As the threshold r increases, more edges enter the graph, and more cliques enter the Vietoris–Rips complex. Throughout this process, cycles (e.g., 1-dimensional loops)

appear and get filled in the complex (Fig. 2A). There is a unique way to pair the distance thresholds at which cycles are born and die.

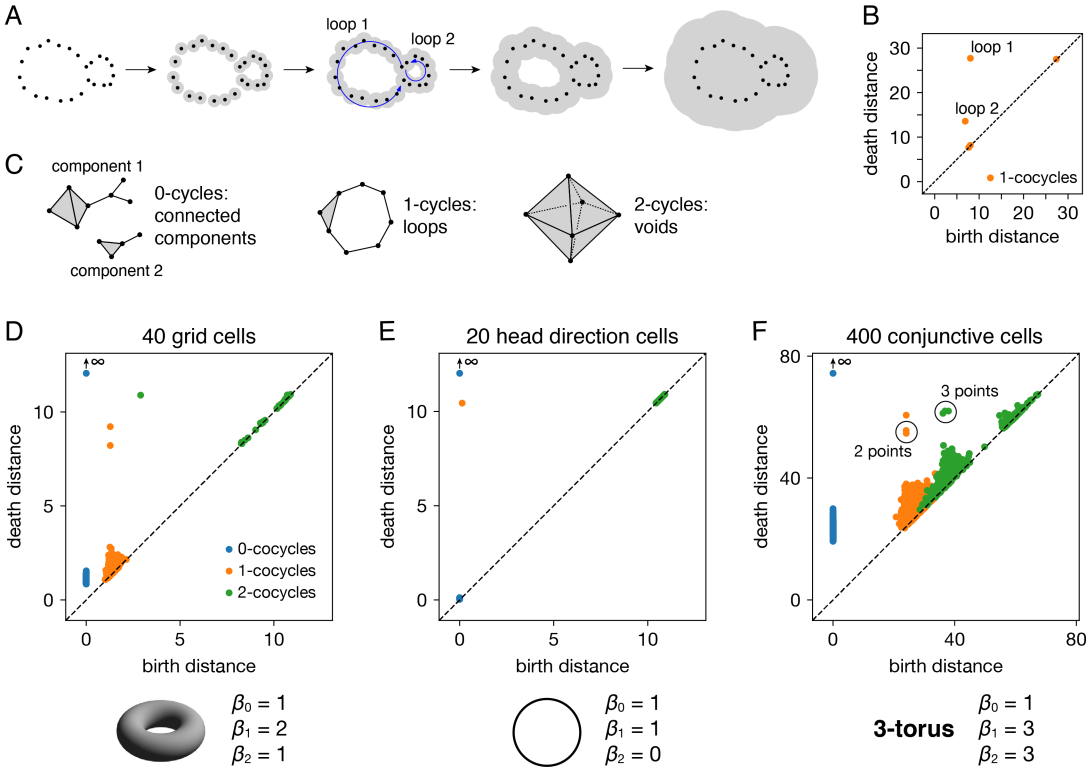

Figure 4.2: Persistence diagrams for periodic neural populations. (**A**–**C**) Interpreting persistence diagrams. (**A**) Persistent (co)homology involves generating a filtration of a dataset in which we connect points within ever-expanding distances. One-dimensional loops will be born and then die at various distances. (**B**) These values are plotted in a persistence diagram where each point corresponds to a single loop. The points located farthest from the diagonal are the most persistent and correspond to significant topological features. (**C**) This process can be performed for features of arbitrary dimension k. (**D**–**F**) Persistence diagrams for periodic neural populations (top) and identified topological spaces (bottom). We compare the number of persistent k-(co)cycles to the k-th Betti numbers β_k of different topological spaces to infer the underlying topological structure of the dataset. (**D**) Grid cells from module 1 exhibit one persistent 0-cocycle, two persistent 1-cocycles, and one persistent 2-cocycle, which corresponds to a torus. (**E**) Head direction cells exhibit one persistent 0-cocycle, one persistent 1-cocycle, and no persistent 2-cocycles, which corresponds to a circle. (**F**) Conjunctive cells exhibit one persistent 0-cocycle, three persistent 1-cocycles, and three persistent 2-cocycles, which corresponds to a 3-torus.

All such birth and death distances are collected into a persistence diagram (Fig. 2B). The points farthest from the diagonal correspond to the most persistent cycles that appear for the

longest range of distance thresholds. They recover topological structure in the space sampled by the point cloud, which corresponds to the processes underlying the data—in our case, the spatial representation networks and external inputs. Persistent (co)homology is stable: the persistent points will remain in the diagram if we make small changes to the data, such as selecting slightly different timepoints or perturbing their values by a small amount of noise. The points closest to the diagonal would appear even if the processes underlying the data lack topological structure, and they are usually interpreted as noise.

The process we described above keeps track of cycles of different dimensions (Fig. 2C). Besides loops (1-dimensional cycles), it tracks connected components (0-dimensional cycles), voids (2-dimensional cycles) and higher-dimensional topological features, which lack a colloquial name. The number of independent k-cycles is called the k-th Betti number and is a topological invariant of a space. We can infer the topology of a dataset by comparing the number of persistent k-(co)cycles to the k-th Betti numbers of conjectured ideal spaces, such a circle or a torus. Note that for every dataset, the 0-(co)cycle corresponding to the entire point cloud will never die, so we consider its death distance to be infinity.

Persistent cohomology for periodic neural populations

Each periodic neural population spans a particular topological space. We recover these relationships when we compute persistent cohomology of our simulated data (Fig. 2D–F). Each grid cell is active at one location in a unit cell that is tiled over 2D space. Grid cells within a single module share the same unit cell but differ in their active location, so each grid module spans a torus, which is periodic in two directions [23]. Similarly, head direction cells span a circle and each conjunctive cell module spans a 3-torus. The correspondence between our results and predicted topological spaces validates the basic capabilities of our methods.

The ability of persistent cohomology to discover topological structure depends on the number of neurons in the dataset, or equivalently, the dimension of the time series embedding. Using the grid cell population as an exemplar, we form multiple datasets with randomly selected neurons to measure the success rate of persistent cohomology as a function of neuron count (Fig. 3A). To measure success, we only use the first cohomology group H^1, which contains 1-cocycles. We define successful discovery of the grid cell torus as a persistence diagram with two persistent 1-cocycles, and we define what it means to be persistent precisely using the commonly used *largest-gap heuristic*. We calculate the lifetime of each cocycle, which is the difference between its death and birth and corresponds to the vertical distance to the diagonal of its point in the persistence diagram (Fig. 3A,B). We find the largest gap in the lifetimes and consider points above this gap to be persistent (Fig. 3B). Figure 3C shows that reliable discovery of the torus using this heuristic can be achieved with ≈20 simulated, idealized grid cells. It also shows that increasing the number of grid cells improves the success rate. This occurs because topological discovery relies on having enough grid cells such that their fields provide a sufficiently uncorrelated coverage of the unit cell. Instead of using data spanning the full 1000 s-long trajectory, which corresponds to 5000 timepoints, we extract portions spanning various lengths. Discovery of the torus requires enough samples of its

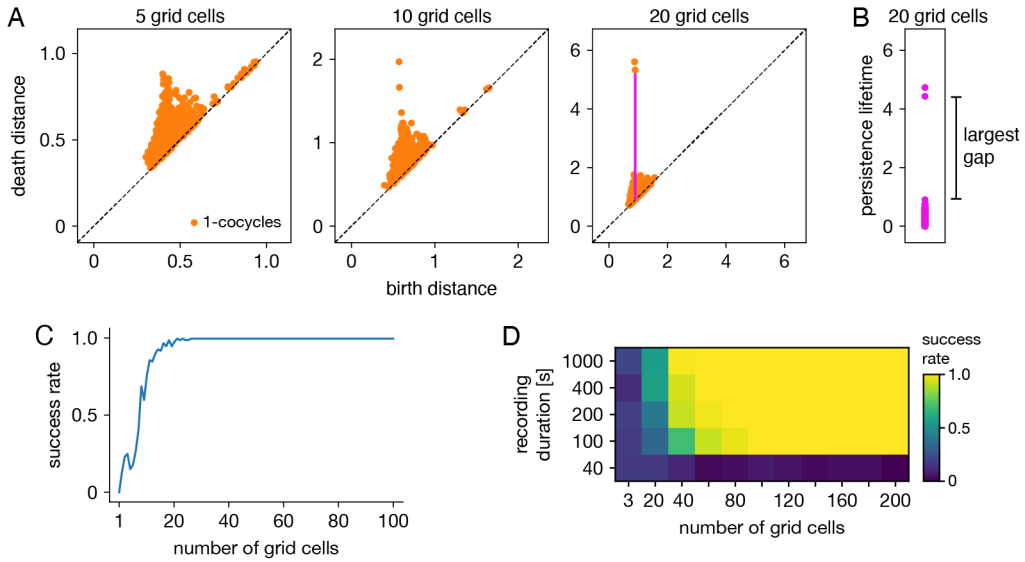

Figure 4.3: Success rates of persistent cohomology for grid cells. (**A**) As the number of grid cells increases, two persistent 1-cocycles emerge. To define persistence precisely, we consider the persistence lifetime of each 1-cocycle, which is the difference between its birth and death distances (length of the magenta line). (**B**) We identify the largest gap in persistence lifetimes and define persistent cocycles as those above this gap. (**C**) Since a grid module has a toroidal topology for which $\beta_1 = 2$, we define success as a persistence diagram with two persistent 1-cocycles. We determine success rates by generating 100 replicate datasets. Success rate increases with the number of grid cells. (**D**) Success rates for different durations extracted from the full simulated recording. Topological discovery benefits from longer recording durations and more neurons.

manifold structure, which is achieved in this case starting at 100 s (Fig. 3D). Thus, persistent cohomology generally thrives in the large-dataset limit with long neural recordings of many neurons.

Persistent cohomology can succeed for mixed signals. Separation of raw electrode recordings into single-neuron spike trains may not always be possible or desired. To address this scenario, we form multi-neuron units by linearly combining time series of neural activity across different grid cells. The mixing coefficients are drawn from a uniform random distribution and then normalized. Example activity maps of these multi-neuron units as a function of position are shown in Fig. 4A. The combination of many neurons destroys the classic responses exhibited by individual grid cells. Yet, multi-neuron units retain topological information associated with the grid module that can be recovered by persistent cohomology (Fig. 4B). The success rate for discovering toroidal topology is remarkably independent of the number of grid cells in each unit. Successful recovery from randomly mixed signals is not entirely unexpected. The preservation of distances under random projections has been

studied extensively in statistics (cf. Johnson–Lindenstrauss lemma [55]).

Persistent cohomology can also succeed in the presence of spiking noise. To simulate such noise, we use our generated activity as a raw firing rate that drives a Poisson-like random process (see Methods). We construct this process to have different Fano factors, which is the variance in the random process for a given firing rate divided by the firing rate. When the Fano factor is 1, the random process is Poisson. Figure 4C shows activity time series for two grid cells that have very similar tuning curves and thus very similar raw firing rates, which can be seen in the noise-free condition (top left). Higher Fano factors lead to more variability both across time for each neuron and across neurons. Persistent cohomology can still recover the toroidal topology of the grid module, though more neurons are required for higher Fano factors (Fig. 4D). In the mammalian cortex, Fano factors lie around $\sim$0.5–1.5 [99, 14]. Applying this regime to our simulations implies that $\approx$80 grid cells are required for reliable topological discovery, but we acknowledge the large differences between simulated and experimental data which may substantially increase this number.

We further test the robustness of persistent cohomology across wide ranges of properties associated with grid cells (Figure 4E–K). Using a trajectory extracted from an animal exploring a square enclosure does not meaningfully change the success rate (Fig. 4E). Grid modules have been shown to favor a lattice orientation of 7.5° relative to square enclosures [106], but this orientation does not affect topological discovery. Similarly, persistent cohomology is not strongly affected by the aspect ratio of the triangular grid (Fig. 4H). Changes in grid dimensions can have a stronger effect. Persistent cohomology fails when the grid scale exceeds the size of the enclosure, which makes sense because the grid module unit cell can no longer be fully sampled (Fig. 4F). When the scale does not change but the size of each firing field is decreased, more grid cells are required to produce toroidal structure in the data since each neuron covers less of the unit cell (Fig. 4G). Under various forms of variability within grid cell tuning curves, the success of persistent cohomology can be maintained if more neurons are recorded, up to a degree (Fig. 4I–K). Beyond critical values, however, variability appears to catastrophically disrupt topological discovery in a way that cannot be overcome with more grid cells. The extra heterogeneity conveyed by an additional neuron overwhelms its contribution to topological structure. Thus, an assessment of tuning curve variability may be a crucial component in the application of persistent cohomology to neural data.

Animal trajectory decoded through topological coordinates

Persistent cohomology can not only discover topological structure in neural data, but it can also decode information embedded within this structure. Recall that a grid module defines a triangular lattice in physical space with fields of each grid cell offset in the rhombic unit cell (Fig. 5A). The periodicities of this unit cell along the two lattice vectors are detected by persistent cohomology as two persistent 1-cocycles belonging to a torus (as seen in Fig. 2D). We can assign circular coordinates [26, 86] for these 1-cocycles (Fig. 5B). These coordinates describe the topological space of the torus and should map back onto the rhombic unit cell that tiles physical space. To explore this relationship, we project the entire time series of

neural activities onto these coordinates. For each neuron, we find the data points for which that neuron is the most active within the population. These points are clustered and define firing fields in topological space (Fig. 5C). The center of masses of these clusters are used to evaluate distances between grid cells; these topological distances are highly correlated with the physical distances between grid offsets within the rhombic unit cell (Fig. 5D). Thus, the topological coordinates defined in neural activity space indeed capture the organization of grid cells in physical space.

Furthermore, persistent cohomology can leverage the mapping between physical and topological spaces to decode trajectories in the former by trajectories in the latter (Fig. 5E). To do so, we trace the circular coordinates depicted in Fig. 5C through time to form a raw trajectory through topological space. We then unshear it by 60° and unfold this trajectory by identifying large jumps with wrapping through the boundary to produce a reconstructed segment. These steps do not require knowledge of the animal's true trajectory (although some general expectations about the animal's motion are required for unshearing; see Methods Sec. 4.5 for details). We find that this reconstruction can be translated, rotated, reflected, and/or uniformly scaled to match the true trajectory very well. Without spiking noise, almost all reconstructions deviate from the true trajectory by less than 4 cm averaged across time, which is much less than the enclosure's diameter of 180 cm (Fig. 5F,G). This error decreases with more neurons or more timepoints in the simulated recording. Similar trends are observed with the introduction of Poisson noise that mimics spiking noise, but more neurons are required, more outliers with poor fit are observed, and geometric subsampling cannot be used to improve computational tractability (Fig. 5H,I).

Persistent cohomology for mixtures of neural populations

Persistent cohomology can discover topological structure in mixtures of neural populations. When neurons are recorded from a periodic neural population and a non-periodic neural population, the latter adds additional dimensions to the point cloud embedding, but the topological structure contained within the former may persist. We test if persistent cohomology can recover this information in mixed datasets with neurons from both a periodic population (either grid or conjunctive) and a non-periodic population (either non-grid spatial or random). Reliable discovery of the torus formed by grid cells is possible when the number of spatial or random cells is less than twice the number of grid cells (Fig. 6A,B). Detection of the 3-torus formed by conjunctive cells requires more neurons, but it can also be reliably achieved in the presence of non-periodic populations (Fig. 6C,D). Thus, persistent cohomology demonstrates robustness to the inclusion of non-periodic populations. The size of the non-periodic population that can be tolerated increases with the size of the periodic population.

When neurons are recorded from multiple periodic neural populations, their structures are preserved within projected subspaces of high-dimensional activity space. We explore persistent cohomology in this scenario by forming mixed datasets with neurons from two periodic populations. When the two populations respond to unrelated signals—such as grid and head direction cells—the combined topological space should be the Cartesian product

of those of the separate populations. Indeed, that persistent cohomology can discover the resultant 3-torus at intermediate mixing ratios (Fig. 7A,B). If one population contributes many more neurons—and thus embedding dimensions—than the other, we instead detect the corresponding single-population structure (Fig. 7B).

When the two populations respond to related signals—such as grid and conjunctive cells—the activity space of one is contained in the activity space of the other. Grid cells and conjunctive cells from the same module encode position with the same toroidal structure; they both tile space with the same rhombic unit cell of neural activity. In addition, the conjunctive population encodes direction with a circular topology. Thus, the mixed dataset should span a 3-torus, which can be detected by persistent cohomology (Fig. 7C). For reliable discovery, at least $\approx$120 conjunctive cells and at least $\approx$240 total neurons are required. However, discovery of the product topology is disrupted if the number of grid cells exceeds the number of conjunctive cells by more than a factor of $\approx$1.5. Thus, persistent cohomology can best detect product topologies when the mixed dataset is not dominated by one population.

Finally, we consider the case of mixing grid cells from multiple modules. Grid modules have different rhombic unit cells with different scales and orientations, so they map the same physical space onto different topological coordinates. Thus, a mixed dataset from two different modules should exhibit the product topology of two 2-tori, which is the 4-torus. However, we are unable to reliably discover this structure using the grid modules illustrated in Fig. 1A; they are too sparse. To produce a point cloud that embeds the toroidal structure for one grid module, the animal trajectory should densely sample its rhombic unit cell. This is achieved since the enclosure contains many unit cells. However, to produce a point cloud that embeds the 4-torus formed by two grid modules, the animal trajectory should densely sample all combinations of unit cells. This is not achieved by the grid modules illustrated in Fig. 1A because the enclosure contains too few rhombic unit cells for them to overlap in many different configurations.

Thus, we generate two grid modules separated by the same scale ratio as in Fig. 1A, but with one-fourth of its scale (Fig. 7D). In addition, we explore different relative orientations between the modules by generating different orientations for module 2. Notably, these scale ratios and orientation differences are not chosen such that the two rhombic unit cells would share a simple geometric relationship with each other [60], which would limit their possible overlap configurations and collapse the expected 4-torus structure to a 2-torus. As we include more neurons from both modules into our dataset, we see that four persistent 1-cocycles eventually emerge from the points close to the diagonal that represent sampling noise (Fig. 7E). The success of persistent cohomology is independent of the orientation difference between the two modules (Fig. 7F). Note that if we obtained independent activity samples from each module, combined datasets formed from the original grid modules with larger scales should exhibit 4-torus structure. However, samples taken from a single animal trajectory are not independent across modules, so smaller grid scales (or equivalently, larger environments) are required to fully sample the 4-torus structure.

4.4 Discussion

We demonstrate that persistent cohomology can discover topological structure in simulated neural recordings with as few as tens of neurons from a periodic neural population (Fig. 3). From this structure, it can decode the trajectory of the animal using only the time series of neural activities (Fig. 5). It can also discover more complex topological structures formed by combinations of periodic neural populations if each population is well-represented within the dataset (Fig. 7).

By comprehensively adjusting a wide range of parameters related to grid cells, we find that persistent cohomology generally behaves in three different ways with respect to parameter variation. First, topological discovery can be unaffected by some manipulations, such as combining grid cells into multi-neuron units and changing global geometric features such as enclosure geometry and lattice aspect ratio (Fig. 4). Second, topological discovery may be impeded in a way that is counteracted by increasing neuron number. Spiking noise, small field sizes, and inclusion of non-periodic populations are examples of parameters that exhibit this behavior (Figs. 4 and 6). Third, topological discovery can fail catastrophically in certain parameter regimes without the possibility of recovery by including more neurons. This happens if tuning curves are inherently too variable or if discovery of product topologies are desired when one neural population vastly outnumbers the other (Figs. 4 and 7). These conclusions can help researchers understand and overcome obstacles to topological discovery with persistent (co)homology and may guide its use across a variety of neural systems.

We have characterized the capabilities of persistent cohomology using simple simulated data, but our results may generalize to real neural data. A key requirement for generalization is the separation of two timescales. The macroscopic timescale at which topological structures are explored—here, the time required to traverse a rhombic unit cell of a grid module or 360° of head direction—must be much longer than the microscopic timescale at which neuronal activity is generated. This enables us to coarse-grain over spikes and describe the activity by a firing rate. Indeed, the inputs to our analysis pipeline are firing rates over 0.2 s time bins, which averages over many neurophysiological processes, including major neural oscillations found in the hippocampal region [68]. Similar forms of coarse-graining were used by [93], [17], and [41] to successfully apply persistent (co)homology to experimental recordings in the spatial representation system.

For comparison, we attempt to modify manifold learning algorithms to enable discovery and interpretation of topological structure (Supplementary Information and Supp. Fig. 1). We find that Isomap [111] followed by Independent Components Analysis (ICA) [52] can successfully identify and decode from toroidal structure with modified grid cells whose tuning curves exhibit a square lattice. Unlike persistent cohomology, it does not work for triangular lattices and does not consider topological features with dimensionality greater than 1. UMAP [71] can be used to embed grid cell data directly into a 2-dimensional space with a toroidal metric, but we find that its coordinates do not correspond well to the physical grid periodicity. In short, persistent cohomology performs better and requires adjusting fewer parameters for topological data analysis compared to these alternative methods, which

were not designed for such analysis. Note that [17] formulate an alternative method for decoding topologically encoded information; they also use persistent homology for discovery of topological structure.

The application of persistent (co)homology to neuroscience data is still in its developing stages. In addition to the research on spatial representation circuits described above [93, 17, 41], notable lines of work include: simulations of hippocampal place cells in spatial environments with nontrivial topology [24, 101, 19, 4, 3]; analysis of EEG signals, for classification and detection of epileptic seizures [90, 121] and for construction of functional networks in a mouse model of depression [63]; inferring intrinsic geometric structure in neural activity [43]; and detection of coordinated behavior between human agents [125]. There is potential for persistent (co)homology to provide insight to a wide range of neural systems. Topological structures generally can be found wherever periodicities exist. These periodicities can take many forms, such as the spatial periodicities in our work, temporal regularities in neural oscillations, motor patterns, and neural responses to periodic stimuli.

The toolbox of topological data analysis has more methods beneficial to the analysis of neural data. The methods described in this paper, including geometric subsampling, are sensitive to outliers. This problem can be addressed within the same framework of persistent cohomology by using the distance-to-a-measure function [18]. In practice, this would translate into a slightly more elaborate construction [46] of the Vietoris–Rips complex. Furthermore, our analysis pipeline benefits from having neural activity embedded in a high dimensional space, i.e., from having many more neurons than the intrinsic dimensions of the recovered tori. It is possible to adapt this technique to the regime of limited neural recordings (even to a single neuron) by using time-delay embeddings [107]. However, for spatial populations, such a technique would require control over the smoothness of the animal's trajectory, which may not be feasible in practice. Also, the method we present cannot make inferences on network topology. If connectivity information were present in neural activities, they should appear on fast timescales related to synaptic integration, action potential propagation, and synaptic delay. By averaging neural activity into 0.2 s time bins, we destroy this information, but it is possible that a modified method may access it.

Our results also suggest research directions in topological data analysis. Throughout the paper, we relied on 1-dimensional persistent cohomology to infer whether we recovered a particular torus. But that is a relatively weak method: many topological spaces have cohomology groups of the same dimension. Although the trajectories that we recover via circular coordinates serve as a convincing evidence that we are indeed recovering the tori, it is possible to confirm this further by exploiting cup product structure in cohomology, which is a particular kind of a topological operation that turns cohomology into a ring. Computing a "persistent cup product" would provide additional evidence about the structure of the recovered spaces.

4.5 Methods

Generating neural recordings

Animal trajectory

We simulate the simultaneous recording of neurons from a rat as it explores a circular enclosure of diameter 1.8 m. We use 1000 s from a trajectory extracted from an experimental animal [47, 14]. This trajectory is provided as velocities sampled at 0.5 ms intervals along with the initial position. We average these to positions and directions at 0.2 s intervals as follows: the position of the animal is simply the average position within each 0.2 s time bin, and the direction of the animal is the circular mean of the velocity vector angle within each 0.2 s time bin. We ignore the distinction between body direction and head direction.

Periodic neural populations

We generate tuning curves as a function of position and/or direction for each neuron. These localized tuning curves are based on a shifted and truncated cosine function:

$$f(z) = \begin{cases} \dfrac{1+\cos \pi z}{2} & |z| < 1 \\ 0 & |z| \geq 1. \end{cases} \tag{4.1}$$

For each grid module, we set a scale l and an orientation φ. This defines a transformation matrix from the space of phases $\left[-\frac{1}{2}, \frac{1}{2}\right) \times \left[-\frac{1}{2}, \frac{1}{2}\right)$ to the rhombic unit cell of the grid module in physical space:

$$\mathsf{A} = l \begin{pmatrix} \cos\varphi & \cos\left(\varphi + \frac{\pi}{3}\right) \\ \sin\varphi & \sin\left(\varphi + \frac{\pi}{3}\right) \end{pmatrix}. \tag{4.2}$$

Unless otherwise specified, we use $l = 40$ cm and $\varphi = 0$. The inverse of this matrix A^{-1} maps the rhombic unit cell onto the space of phases. We also define $\|\cdot\|$ as the vector norm and $\langle a \rangle_m \equiv (a + m \bmod 2m) - m$ as the shifted modulo operation. The tuning curve of a grid cell as a function of position $\mathbf{x}$ is then

$$s_{\text{grid}}(\mathbf{x}; \mathbf{b}) = f\left(\tfrac{1}{0.45l} \left\| \mathsf{A} \langle \mathsf{A}^{-1}\mathbf{x} - \mathbf{b} \rangle_{1/2} \right\| \right) \tag{4.3}$$

Each grid cell is shifted by a uniformly random phase offset $\mathbf{b}$. The full width at half maximum of each grid field is $0.45l$. In physical space, the offset of a grid cell is $\mathsf{A}\mathbf{b}$, where integers can be added to either component of $\mathbf{b}$; the shortest distance between these offsets for two grid cells is the physical distance shown in Fig. 5D.

The tuning curve of a head direction cell as a function of direction θ is

$$s_{\text{dir}}(\theta; c) = f\left(4\langle \theta - c \rangle_\pi\right). \tag{4.4}$$

Each head direction cell is shifted by a uniformly random direction offset c. The full width at half maximum of the head direction field is $\pi/2$.

The tuning curve of a conjunctive cell is simply the product

$$s_{\text{conj}}(\mathbf{x}, \theta; \mathbf{b}, c) = s_{\text{grid}}(\mathbf{x}; \mathbf{b}) s_{\text{dir}}(\theta; c). \tag{4.5}$$

Each conjunctive cell has uniformly random offsets $\mathbf{b}$ and c.

Non-periodic neural populations

For non-grid spatial cells, we generate tuning curves as a function of position for each neuron of the form

$$s_{\text{spatial}}(\mathbf{x}) = \frac{1}{2}\left(\mathrm{e}^{-\|\mathbf{x}-\mathbf{d}_1\|^2/2\sigma^2} + \mathrm{e}^{-\|\mathbf{x}-\mathbf{d}_2\|^2/2\sigma^2}\right), \tag{4.6}$$

where $\sigma = 40\,\text{cm}$ and the $\mathbf{d}_i$'s are chosen uniformly randomly between $(0\,\text{cm}, 0\,\text{cm})$ and $(180\,\text{cm}, 180\,\text{cm})$.

For random neurons, we obtain activity time series by sampling from a distribution every 2 s, or 10 timepoints, and interpolating in between using cubic polynomials. The distribution is Gaussian with mean 0 and width 0.5, truncated between 0 and 1.

From tuning curves to time series

To obtain activity time series for all populations except for random neurons, we apply the tuning curves to each trajectory timepoint. Whenever the velocity decreases below 5 cm/s, we set the activity to be 0. This threshold simulates the behavior of neurons in the hippocampal region that exhibit high activity during locomotion and low activity during idle periods [95, 66, 51].

Multi-neuron units for grid cells (Fig. 4A,B)

We generate multi-neuron units (Fig. 4) by linearly combining activity time series from multiple grid cells. Each mixing coefficient is chosen from a uniform random distribution between 0 and 1. The activity is then normalized by the sum of squares of the mixing coefficients.

Spiking noise for grid cells (Fig. 4C,D)

The activities described above are dimensionless, and we typically do not need to assign a scale because we divide each time series by its mean before applying persistent cohomology. To create spiking noise, however, we must set the firing rate. We linearly rescale the rate given by Eq. 4.3:

$$\lambda = 0.4 + 7.6 s_{\text{grid}}. \tag{4.7}$$

This sets the maximum firing rate to be 8 and creates a baseline rate of 0.4; with 0.2 s time bins, these values correspond to 40 Hz and 2 Hz, respectively. However, we still set the firing rate to 0 Hz when the animal's velocity decreases below 5 cm/s.

Using λ, we generate Poisson-like spiking noise with different levels of variability (Fig. 4). At each timestep, the noisy activity is given by

$$s_{\text{noisy}} = F \cdot X, \qquad X \sim \text{Pois}(\lambda/F). \tag{4.8}$$

F sets the Fano factor of the random process, which is its variance divided by its mean (for any given λ). The $F = 1$ case corresponds to a Poisson random process; $F < 1$ implies sub-Poissonian noise and $F > 1$ implies super-Poissonian noise.

Generating grid cells with various properties (Fig. 4E–K)

- Square enclosure (Fig. 4E): We use 1000 s from a trajectory extracted from an experimental animal in a square enclosure of width 1.5 m [105], binned in the same way as for the circular enclosure (Sec. 4.1.1). To change the lattice orientation, we use a nonzero value for φ in Eq. 4.2.

- Scale factor (Fig. 4F): Grid scale is modified by changing l in Eq. 4.2. The scale factor is l divided by the diameter of the enclosure, 1.8 m.

- Field size factor (Fig. 4G): Field size is modified by replacing 0.45 in Eq. 4.3 by the field size factor. In other words, the field size factor multiplied by the grid scale is the full width at half maximum of each grid field.

- Aspect ratio (Fig. 4H): The grid lattice is elongated to an aspect ratio $\varepsilon \geq 1$, which is the ratio between the major and minor axes of the ellipse that circumscribes each hexagonal lattice domain. This is accomplished by replacing the transformation matrix in Eq. 4.2 with

$$\mathsf{A} = l \cdot \frac{\sqrt{3}}{2} \sec 2\psi \begin{pmatrix} 2\cos\psi\cos\varphi & \cos\big(\varphi+\psi\big) \\ 2\cos\psi\sin\varphi & \sin\big(\varphi+\psi\big) \end{pmatrix}, \tag{4.9}$$

  where $\psi \leq 60°$ is the angle between the two lattice vectors. ψ is related to ε by

$$\psi = \text{arcsec}\left(\frac{3}{\varepsilon} - 1\right). \tag{4.10}$$

  When $\varepsilon = 3/(1+\sqrt{2}) \approx 1.24$, $\psi = 45°$ and the lattice becomes square.

- Field jitter (Fig. 4I): Jitter in the position of grid fields is introduced by shifting each field in physical space by

$$U_1 l \begin{pmatrix} \cos\varphi \\ \sin\varphi \end{pmatrix} + U_2 l \begin{pmatrix} \cos\big(\varphi+\frac{\pi}{3}\big) \\ \sin\big(\varphi+\frac{\pi}{3}\big) \end{pmatrix}, \tag{4.11}$$

  where l is the grid scale, φ is the grid orientation, and $U_i \sim \text{Unif}(-\text{field jitter}, \text{field jitter})$.

- Field variability (Fig. 4J): Variability across grid fields is introduced by choosing random numbers $U_{ij} \sim \text{Unif}(-\text{field var.}, \text{field var.})$ at 20 cm-intervals throughout the enclosure for each grid cell. A linear interpolation is constructed for these points, which is smoothed by a Gaussian filter of width 20 cm. This resulting random function is added to the standard tuning curve in Eq. 4.3, and all values under 0 are clipped to 0.

- Lattice variability (Fig. 4K): Variability in lattices across grid cells is introduced by randomly perturbing the transformation matrix for each grid cell (Eq. 4.2) according to

$$\mathsf{A} = \begin{pmatrix} 1+W_{11} & W_{12} \\ W_{21} & 1+W_{22} \end{pmatrix} l \begin{pmatrix} \cos\varphi & \cos\left(\varphi+\frac{\pi}{3}\right) \\ \sin\varphi & \sin\left(\varphi+\frac{\pi}{3}\right) \end{pmatrix}, \tag{4.12}$$

  where

$$W_{ij} \sim \text{Unif}(-\text{lattice var.}, \text{lattice var.}). \tag{4.13}$$

Neural activity maps

We construct activity maps for each neuron as a function of position or direction. To do so, we simply tally the total amount of activity in each positional or directional bin. Note that these maps do not depict firing rate because we do not divide by the occupancy of each bin; we decided against this in order to show the activity experienced through the animal trajectory.

Processing neural recordings

For each neuron, we first divide its activity at every timepoint by its mean activity. We then delete all timepoints whose neural activities are all smaller than a small limit 1×10^{-4}. This simple procedure removes points at the origin that we do not expect to participate in topological structures.

To improve computational efficiency, we reduce the number of input points while preserving their topological structure by applying a geometric subsampling strategy. We pick the first point at random, and then iteratively add a point to our subsample that is the furthest away from the already chosen points. Specifically, if P is the input point set and Q_i is the subsample after i iterations, we form Q_{i+1} by adding the point q_{i+1} chosen as

$$\arg\max_{p\in P} \min_{q\in Q_i} \|p-q\|. \tag{4.14}$$

Fig. 1B illustrates a result of this strategy. By construction, the subsample Q_i forms an ε_{i+1}-net of the input point sample, which means the largest distance from any input point to the nearest point of the subsample does not exceed $\varepsilon_{i+1} = \min_{q\in Q_i} \|q_{i+1}-q\|$. Because persistent cohomology is stable, this guarantees that the persistence diagram we compute for the subsample Q_i is at most ε_{i+1} away, in bottleneck distance [21], from the persistence diagram of the full point set P. We generally select 1000 points through this process. The

results in Fig. 5H,I were obtained by applying persistent cohomology on the full dataset without subsampling.

To measure success rate, we apply persistent cohomology on 100 replicate datasets and measure the proportion of successes as determined by the largest-gap heuristic.

Applying persistent cohomology

We refer the reader to extensive literature on persistent (co)homology [30, 32] for the full details. More details on the involved constructions are presented in the Supplementary Information. Here, we only briefly mention some of them. For technical reasons—both to recover the circular coordinates and for computational speed—we work with persistent cohomology, which is dual to persistent homology, which a reader might be more familiar with.

To recover the topology of the space sampled by a point set P, we construct a Vietoris–Rips simplicial complex. Given a parameter r, Vietoris–Rips complex consists of all subsets of the point set P, in which every pair of points is at most r away from each other,

$$\mathrm{VR}(P, r) = \{\sigma \subseteq P \mid \|p - q\| \leq r \ \forall p, q \in \sigma\}. \tag{4.15}$$

The cohomology group, $H^k(\mathrm{VR}(P, r))$, defined formally in the Supplementary Information, is an algebraic invariant that describes the topology of the Vietoris–Rips complex. Its rank, called the k-th Betti number, counts the number of "holes" in the complex.

As we vary the radius r in the definition of the Vietoris–Rips complex, the simplicial complexes nest: $\mathrm{VR}(P, r_1) \subseteq \mathrm{VR}(P, r_2)$, for $r_1 \leq r_2$. The restriction of the larger complex to the smaller induces a linear map on cohomology groups, and all such maps form a sequence:

$$H^k(\mathrm{VR}(P, r_1)) \leftarrow H^k(\mathrm{VR}(P, r_2) \leftarrow H^k(\mathrm{VR}(P, r_3)) \leftarrow \dots \tag{4.16}$$

It is possible to track when cohomology classes (i.e., "holes") appear and disappear in this sequence. Recording all such birth–death pairs (r_i, r_j), we get a persistence diagram, which completely describes the changes in the sequence of cohomology groups.

For a persistent class, i.e., one with a large difference between birth and death, [100] describe a procedure for turning it into a map from the input data points into a circle, which assigns a circular coordinate to each data point. [86] extends that procedure to allow computation of persistent cohomology on a subsample of the data, e.g., the geometric subsample mentioned in the previous subsection.

Reconstructing animal trajectory from circular coordinates

The process for obtaining circular coordinates outlined in the previous subsection (and presented in greater detail in the Supplementary Information) outputs one value between 0 and 2π for each persistent 1-cocycle at each timepoint. A grid module yields two persistent 1-cocycles, so our circular coordinates form a vector $\mathbf{u}_t = (u_{t1}, u_{t2})$ at each timepoint

$t = 1, \ldots, T$. We divide each coordinate value by 2π so that each $u_{ti} \in [0, 1)$. No matter the recording duration used to obtain circular coordinates, we only reconstruct the first 100 s of the animal trajectory.

We first perform a preliminary unfolding of the circular coordinates. We calculate all the difference vectors between adjacent timepoints and cancel out changes by more than $\pm 1/2$:

$$\Delta u_{ti}^{\text{unfolded}} = \langle u_{ti} - u_{t-1,i} \rangle_{1/2}, \quad \text{where} \quad \langle a \rangle_{1/2} \equiv (a + 1/2 \bmod 1) - 1/2. \tag{4.17}$$

Next, we seek to unshear the coordinates. The rhombic unit cell in physical space is sheared by 30° relative to the orthogonal axes of topological space. We wish to apply this transformation to the difference vectors to restore the unsheared trajectory. There are two possible unshearing matrices

$$\mathsf{S}^{\pm} = \begin{pmatrix} \cos 0 & \cos\left(\frac{\pi}{2} \pm \frac{\pi}{6}\right) \\ \sin 0 & \sin\left(\frac{\pi}{2} \pm \frac{\pi}{6}\right), \end{pmatrix} \tag{4.18}$$

and we could perform the rest of the analysis for both transformations separately, knowing that one trajectory is unsheared and the other is doubly shared. Instead, we assume knowledge that the animal is exploring an open field environment in which all directions of motion should generally be sampled uniformly. We calculate the covariance matrix for both sets of transformed difference vectors:

$$\Sigma_{ij}^{\pm} = \frac{1}{T} \sum_t \Delta u_{ti}^{\pm} \Delta u_{tj}^{\pm}, \quad \text{where} \quad \Delta \mathbf{u}_t^{\pm} = \mathsf{S}^{\pm} \Delta \mathbf{u}_t^{\text{unfolded}}. \tag{4.19}$$

The proper unshearing S produces the covariance matrix whose ratio of eigenvalues is closest to 1. This heuristic could be changed assuming different statistics of animal motion, for example those corresponding to a linear track.

After identifying the unshearing matrix S, we return to the raw coordinates and apply this transformation first, before unfolding:

$$\Delta \mathbf{u}_t^{\text{unsheared}} = \mathsf{S}(\mathbf{u}_t - \mathbf{u}_{t-1}). \tag{4.20}$$

Since shear transformations change distances, we reevaluate our unfolding. We compare the norm of every difference vector $\Delta \mathbf{u}_t^{\text{unsheared}}$ to its norm after possible unfoldings along the unsheared lattice vectors given by the columns of S. The shortest vector at each timepoint is the reconstructed difference $\Delta \mathbf{u}_t^{\text{recon}}$, and the reconstructed trajectory segment shown in Fig. 5E is their accumulation $\mathbf{u}_t^{\text{recon}} = \sum_t \Delta \mathbf{u}_t^{\text{recon}}$.

So far, this reconstruction has not incorporated detailed information about the animal trajectory. To judge its quality, we now fit the reconstruction to the true segment of animal trajectory $\mathbf{x}_t$ through rigid transformation and uniform scaling. We first determine whether or not the reconstruction needs to be unreflected. To judge this, we calculate the signed vector angles between adjacent steps for both the reconstruction and the true trajectory

$$\begin{aligned} \Delta\theta_t^{\text{recon}} &= \left\langle \arctan(\Delta u_{t2}^{\text{recon}} / \Delta u_{t1}^{\text{recon}}) - \arctan(\Delta u_{t-1,2}^{\text{recon}} / \Delta u_{t-1,1}^{\text{recon}}) \right\rangle_{\pi} \\ \Delta\theta_t &= \langle \arctan(\Delta x_{t2} / \Delta x_{t1}) - \arctan(\Delta x_{t-1,2} / \Delta x_{t-1,1}) \rangle_{\pi}, \end{aligned} \tag{4.21}$$

where $\langle a \rangle_\pi \equiv (a+\pi \bmod 2\pi)-\pi$. If the mean square difference between $-\Delta\theta_t^{\text{recon}}$ and $\Delta\theta_t$ is less than that between $\Delta\theta_t^{\text{recon}}$ and $\Delta\theta_t$, then we reflect our reconstruction $\mathbf{u}_t^{\text{oriented}} = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right) \mathbf{u}_t^{\text{recon}}$. Otherwise, $\mathbf{u}_t^{\text{oriented}} = \mathbf{u}_t^{\text{recon}}$.

Finally, we fit the reconstruction to the true trajectory segment by minimizing the mean squared error after uniform scaling a, translation $\mathbf{b}$ and rotation θ:

$$\min_{a,\mathbf{b},\theta} \sum_t \left\| \mathbf{x}_t - a \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \mathbf{u}_t^{\text{oriented}} + \mathbf{b} \right\|^2. \tag{4.22}$$

Note that we chose to try particular unshearing matrices in Eq. 4.18 and use the general assumption of isotropic animal motion to select between them, all before fitting the reconstruction to the actual animal trajectory in Eq. 4.22. These choices were designed to obtain the most faithful reconstruction without using the animal trajectory, which is the information that we would like to infer. This process assumes knowledge of the angle between directions of periodicity, which is approximately 60° for grid modules. Moreover, it also relies on the circular coordinates implementation (explained in greater detail in Supplementary Information) selecting two of the shortest nonparallel vectors between lattice vertices as its basis—that selection is not guaranteed, but empirically, it frequently occurs in our case, as indicated by low reconstruction errors in Fig. 5. An alternative method for unshearing before comparison with the true trajectory is to find a transformation matrix that yields a covariance matrix for the transformed difference vectors (analogously to Eq. 4.19) with a desired ratio of eigenvalues. Otherwise, if a sheared reconstruction is sufficient, unshearing can be skipped.

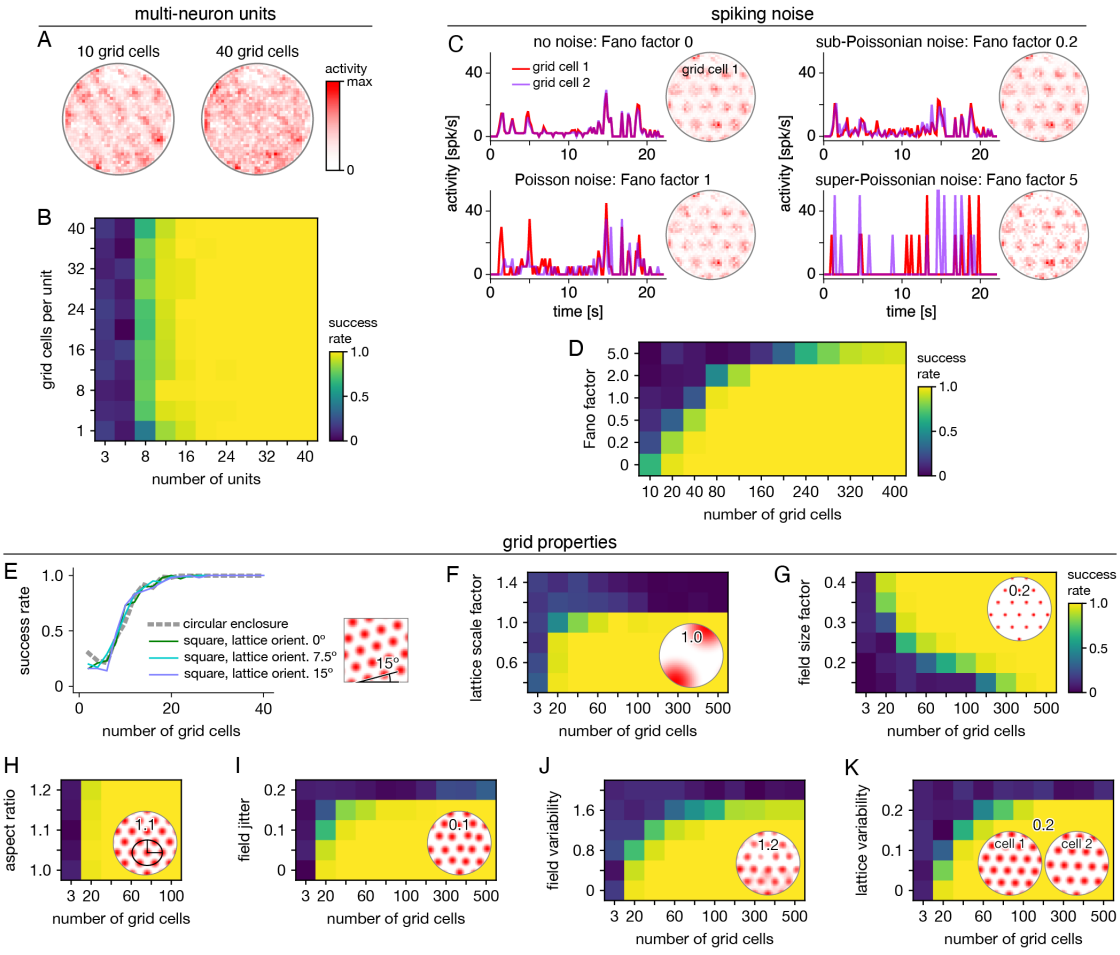

Figure 4.4: Robustness of persistent cohomology for grid cells. (**A,B**) Linearly combining activities from multiple neurons before applying persistent cohomology. (**A**) Example positional activity maps for multi-neuron units. (**B**) The success rate is largely independent of the number of neurons per multi-neuron unit. (**C,D**) Introducing Poisson-like spiking noise with different amounts of variability, as measured by the Fano factor. (**c**) For each Fano factor, time series for two grid cells with very similar offsets (left) and positional activity map of the first (right). (**D**) Persistent cohomology requires more neurons to achieve success with higher Fano factors. (**E**–**K**) Success rates across many different grid cell variables with insets depicting example values. See Methods Sec. 4.5 for a complete description of each variable. (**E**) Introducing a square enclosure at various lattice orientations. (**F**) Changing the lattice scale factor, which is grid scale divided by enclosure diameter. (**G**) Changing the field size factor, which is grid field diameter divided by grid scale. (**H**) Stretching triangular grids to various aspect ratios. (**I**) Jittering the position of each field by random vectors multiplied by grid scale. (**J**) Introducing variability in shape and overall activity across grid fields. (**K**) Introducing variability in lattices across grid cells.

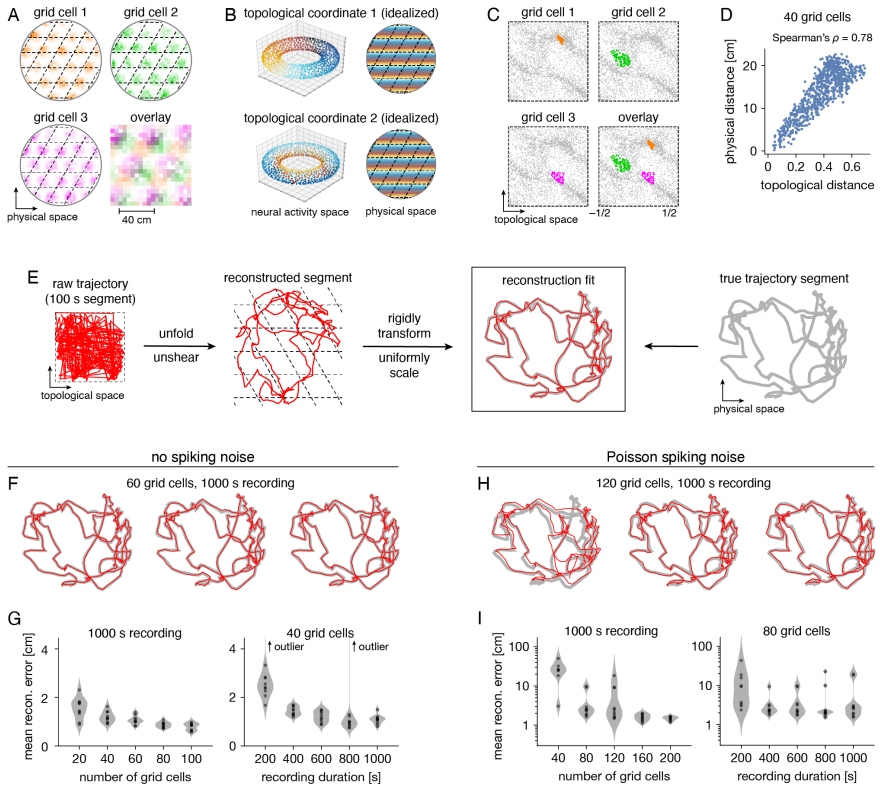

Figure 4.5: Correspondence between spatial and topological coordinates for grid cells. (**A**) Positional activity maps. A grid module tiles physical space with a rhombic unit cell (black dashed lines) with a different offset for each grid cell. (**B**) Schematic of circular coordinates (color) for the two persistent 1-cocycles of a grid module that define a topological space (left) and their mapping onto physical space (right). (**C**) Firing fields in the topological space of neural activities. All timepoints (gray) and those for which a grid cell is the most active in the population (color). (**D**) Topological distances between firing fields depicted in **C** are correlated with physical distances between grid offsets depicted in **A**. Each point represents distances between two grid cells. (**E**) Reconstructing a segment of animal trajectory in the topological space of neural activities and fitting it to the true trajectory segment. (**F,G**) Reconstructions for grid cells without spiking noise. (**F**) Sample reconstruction fits. (**G**) Mean reconstruction error is the distance between reconstructed and actual positions averaged over timepoints. For each condition, we generate 10 replicate datasets, shown as violin plots with points representing individual values. Error generally decreases with increasing grid cell number and trajectory duration. (**H,I**) Same as **F,G** but for grid cells generated with Poisson spiking noise and error plotted on a logarithmic scale. Geometric subsampling was not performed. Persistent cohomology can achieve low reconstruction error with large enough grid cell number and trajectory duration.

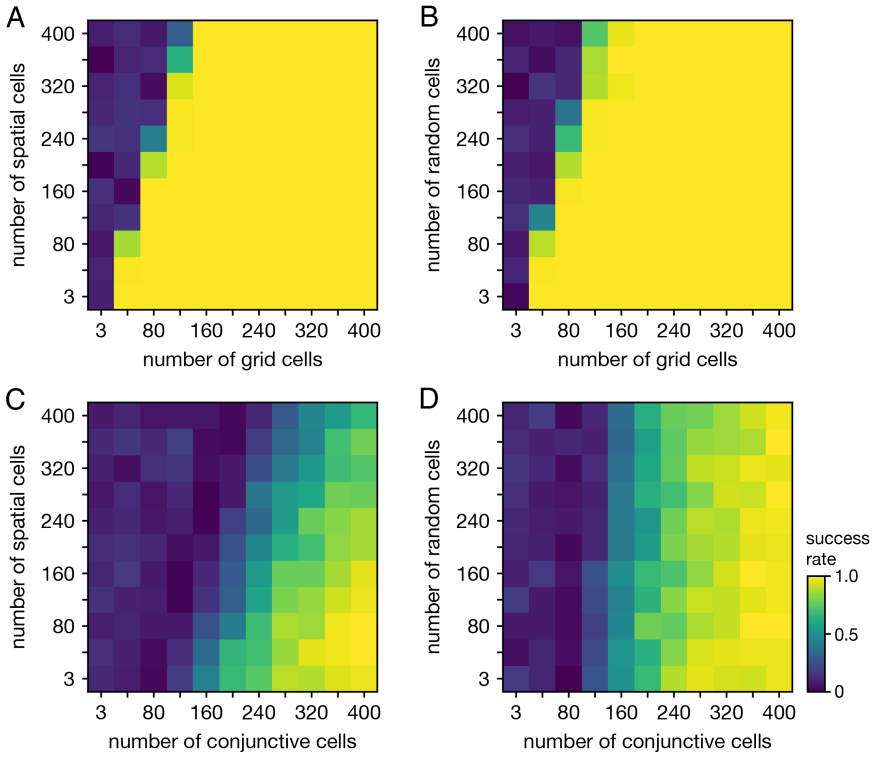

Figure 4.6: Persistent cohomology in combinations of periodic and non-periodic neural populations. Success is defined by observing the number of persistent 1-cocycles expected from the periodic population, which is two for grid cells and three for conjunctive cells. (**A**) Grid cells from module 1 and non-grid spatial cells. (**B**) Grid cells from module 1 and random cells. (**C**) Conjunctive cells from module 1 and non-grid spatial cells. (**D**) Conjunctive cells from module 1 and random cells.

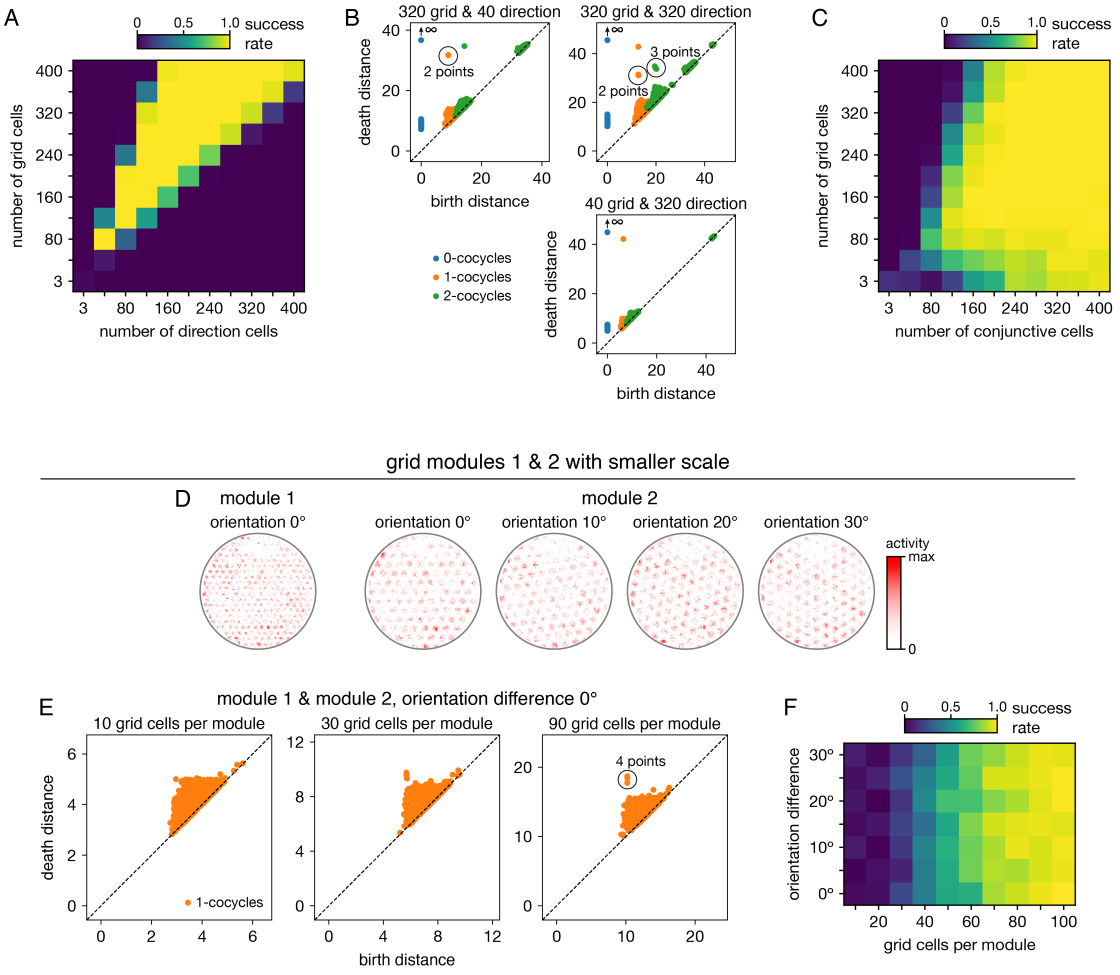

Figure 4.7: Persistent cohomology in combinations of periodic neural populations. Success is defined by observing the number of persistent 1-cocycles expected from the product topology. (**A,B**) Discovering the 3-torus for grid cells from module 1 and head direction cells. (**A**) Success rates. (**B**) Example persistence diagrams. If one population contributes many more neurons than the other, persistent cohomology detects the topology of the predominant population. (**C**) Success rates for discovering the 3-torus for grid cells and conjunctive cells, both from the same module 1. (**D**–**F**) Combining grid cells from two modules with smaller scales. (**D**) Positional activity maps for example neurons from module 1 and four different orientations of module 2. (**E**) Example persistence diagrams. (**F**) Success rates for discovering the 4-torus.

Chapter 5

Transition to Chapter 6

In the previous chapter, we explored the application of persistent cohomology to neural data, specifically focusing on the spatial representation system in the brain. This work highlighted the power of topological methods to uncover low-dimensional structures within high-dimensional data, providing insights into the organization and function of neural circuits. As we transition to the next topic, we shift our focus from neural data to protein structure, while continuing to leverage advanced mathematical techniques for biological discovery.

The subsequent chapter delves into the structural annotation of leucine-rich repeat (LRR) domains within proteins. Unlike the neural data explored previously, protein structures present a unique set of challenges and opportunities for topological and geometric analysis. With the advent of deep learning models such as AlphaFold 2, we now have unprecedented access to high-quality protein structure predictions. This advancement opens new avenues for improving domain annotation methods that have traditionally relied on sequence-based approaches [57].

While the domains of neural data and protein structures may seem disparate, both fields benefit from the application of dimensionality reduction and topological data analysis (TDA). In neural data, persistent cohomology allowed us to detect and quantify topological features within high-dimensional activity spaces [32, 26]. Similarly, in the realm of protein structure, we aim to identify and annotate repeating structural motifs, such as the LRR domains, by leveraging geometric information. Persistent cohomology's ability to reveal underlying structures in complex datasets aligns with our goal of enhancing protein domain annotation. Traditional sequence-based methods, such as Hidden Markov Models (HMMs), often fall short in accurately delineating domain boundaries and identifying repeat units due to the lack of structural context [36, 7]. By incorporating geometric data from predicted protein structures, we can improve the precision of these annotations, providing a more comprehensive understanding of protein functionality.

Leucine-rich repeat domains are characterized by their repetitive nature and curved solenoid structure. These domains play crucial roles in various biological processes, including immune response, where they are involved in recognizing pathogen molecules. Existing annotation tools, such as LRRPredictor, utilize sequence motifs to predict the presence

and boundaries of LRR domains. However, these tools often struggle with highly divergent sequences and irregular motifs [69].

The upcoming chapter presents a novel method that integrates structural data from AlphaFold 2 predictions with geometric and topological analysis to enhance the annotation of LRR domains. This method involves flattening the three-dimensional structure of the protein into a two-dimensional plane, enabling the identification of coiling patterns and the precise delineation of repeat units. By computing the winding number of the protein's curve in this flattened representation, we can accurately determine the boundaries of the LRR domain and detect structural anomalies [15, 109].

The research presented in Chapter 6 aims to achieve several key objectives: (a) enhance the precision of LRR domain annotations by incorporating structural information, overcoming the limitations of sequence-based methods; (b) identify structural irregularities within the LRR domain that may have functional implications, providing new insights into protein behavior; and (c) demonstrate the potential of this method for annotating other types of solenoid domains, paving the way for future research in protein structure analysis.

The transition from persistent cohomology in neural data to structure-aware annotation of LRR domains exemplifies the versatility and power of mathematical and computational methods in advancing our understanding of complex biological systems. As we explore this new application, we build on the foundational principles of topological data analysis and geometric modeling to address challenges in protein annotation and contribute to the broader field of structural biology. In the next chapter, we will delve into the details of this innovative approach, presenting the methodology, results, and implications of our findings in the context of LRR domains and their role in immune response. This work not only enhances our understanding of protein structures but also exemplifies the synergy between advanced computational tools and biological research.

The transition from neural data analysis to protein structure annotation underscores the interconnectedness of different biological domains and the unifying power of mathematical methods. By leveraging the strengths of persistent cohomology and geometric analysis, we aim to push the boundaries of what is possible in protein domain annotation, providing more accurate and insightful tools for researchers. As we move forward, the principles and techniques developed in the context of neural data will continue to inform and enhance our approach to protein structure analysis, highlighting the enduring impact of topological and geometric methods in advancing biological science.

In Chapter 6, we will explore the specifics of our structure-aware annotation method for LRR domains, showcasing its application to a set of proteins from *Arabidopsis thaliana* and validating our approach against a benchmark dataset. This work represents a significant step forward in the integration of structural data into protein annotation, offering new perspectives and tools for the scientific community.

Chapter 6

Structure-Aware Annotation of Leucine-rich Repeat Domains

The contents of this chapter are based on a preprint of BX et al [123].

Abstract

Protein domain annotation is typically done by predictive models such as HMMs trained on sequence motifs. However, sequence-based annotation methods are prone to error, particularly in calling domain boundaries and motifs within them. These methods are limited by a lack of structural information accessible to the model. With the advent of deep learning-based protein structure prediction, existing sequenced-based domain annotation methods can be improved by taking into account the geometry of protein structures. We develop dimensionality reduction methods to annotate repeat units of the Leucine Rich Repeat solenoid domain. The methods are able to correct mistakes made by existing machine learning-based annotation tools and enable the automated detection of hairpin loops and structural anomalies in the solenoid. The methods are applied to 127 predicted structures of LRR-containing intracellular innate immune proteins in the model plant *Arabidopsis thaliana* and validated against a benchmark dataset of 172 manually-annotated LRR domains.

Author summary

In immune receptors across various organisms, repeating protein structures play a crucial role in recognizing and responding to pathogen threats. These structures resemble the coils of a slinky toy, allowing these receptors to adapt and change over time. One particularly vital but challenging structure to study is the Leucine Rich Repeat (LRR). Traditional methods that rely just on analyzing the sequence of these proteins can miss subtle changes due to rapid evolution. With the introduction of protein structure prediction tools like AlphaFold 2, annotation methods can study the coarser geometric properties of the structure. In this study,

we visualize LRR proteins in three dimensions and use a mathematical approach to 'flatten' them into two dimensions, so that the coils form circles. We then used a mathematical concept called winding number to determine the number of repeats and where they are in a protein sequence. This process helps reveal their repeating patterns with enhanced clarity. When we applied this method to immune receptors from a model plant organism, we found that our approach could accurately identify coiling patterns. Furthermore, we detected errors made by previous methods and highlighted unique structural variations. Our research offers a fresh perspective on understanding immune receptors, potentially influencing studies on their evolution and function.

Introduction

Solenoid domains are a class of protein structures defined by a repeating helical arrangement of their backbone chain. These domains are found in a diverse range of proteins and play important roles in a variety of biological processes, including protein-protein interactions, molecular recognition, and scaffolding [53]. The coil shape of solenoid domains arises from a repeating motif of amino acid residues, known as *tandem repeat units*. The specific amino acid sequence and length of the repeating unit can vary between solenoid domains, resulting in differences in the overall structure and function of the domain. The modular nature of solenoid domains allows for the construction of complex structures by combining different domains in a predictable and controlled manner [81].

Leucine-rich repeat (LRR) domains are a type of curved solenoid domain with repeated units of about 20 - 30 residues long which contain leucine residues in a beta-strand conformation. These domains are found in a wide range of proteins, including cell surface receptors, enzymes, and structural proteins, and are known to play important roles in protein-protein interactions, signal transduction, and immune recognition [77].

Leucine-rich repeats play a critical role in the function of the NOD-like receptor (NLR) family of proteins in the innate immune system of plants and animals [56]. NLRs are intracellular immune receptors that recognize pathogen-derived molecules and activate downstream signaling pathways to initiate an immune response. NLRs are involved in the recognition of a wide range of pathogens, including bacteria, fungi, and viruses. NLRs typically consist of three domains: an N-terminal domain, a central nucleotide-binding domain, and a C-terminal LRR domain. The LRR domain is responsible for recognizing and binding to pathogen-derived molecules, such as effector proteins or pathogen-associated molecular patterns (PAMPs) [108]. In particular, the LRR domains of plant NLRs are highly diverse and can recognize a wide range of pathogen-derived molecules, allowing plants to mount a robust and specific immune response to a broad range of pathogens. Understanding LRR domains in plant NLRs is important for developing strategies to enhance plant immunity and improve crop resistance to pathogens.

The concave surface of the leucine-rich repeat domain is generally responsible for binding to ligands [80]. The amino acid residues on the concave surface of the LRR domain form a

specific pattern of hydrophobic, polar, and charged residues that can interact with specific ligands, such as proteins, peptides, carbohydrates, or nucleic acids. The specificity of ligand binding by LRR domains is determined by the overall shape and chemical properties of the concave surface, which can be highly variable between different LRR-containing proteins [92][6]. Additionally, LRR domains can contain variable regions and insertions that can modify the binding specificity and affinity of the domain. More recently, studies such as [96] have shown that "post-LRR" domains which lie at the C-terminal end of the LRR are required for successful plant immune response. Accurate annotation of these domains and their constituent repeat units is thus essential to understanding the components which govern protein shape and binding specificity.

Existing methods for annotating LRR domains give unreliable and inconsistent results due to irregularities in sequence motifs. Profile hidden Markov models (HMMs) are widely used, e.g. by HMMER [36], to annotate protein domains in genomic sequences, but they are sensitive to the size and diversity of the protein family being analyzed and do not perform accurately for rapidly-evolving, highly-divergent families such as LRR [7]. Profile HMMs are also unable to delineate tandem repeat units.

An existing tool, LRRPredictor [69], uses an ensemble of 8 machine learning classifiers to determine the residues which comprise the basic LRR motif of the form "LxxLxL" (where "L" refers to Leucine or other hydrophobic amino acid, and "x" can be any amino acid). We found that LRRPredictor often makes mistakes, particularly in identifying divergent motifs near the C- and N-terminal boundaries of the LRR. Because LRRPredictor, like an HMM, is trained on a specific set of LRR sequences taken from Protein Data Bank [11] (PDB), it incorrectly annotates LRR sequences which diverge from its training set.

With AlphaFold 2 [57], a deep-learning-based model, reliable protein structure prediction has become readily available, enabling domain annotation methods with direct access to geometric data from the protein. We leverage this geometric information to annotate essential features of the LRR domain: start/end position, post-LRR detection, repeat unit delineation, and structural irregularities.

From the perspective of differential geometry, a coiling curve in 3D space is characterized by a linearly increasing winding number around a core curve. We therefore detect the coiling LRR region, as the loci where the winding number is sufficiently close to a line of a fixed slope; the post-LRR domain is then decided as C-terminal sequence downstream from the point at which steady winding terminates. The methods section below describes our procedure for computing the winding number across the length of the protein. In contrast to HMM-based or other data driven techniques, our method is completely unsupervised and driven by simple mathematical methods.

Methods

Datasets used in this study

161 NLR protein sequences, i.e. *NLRome*, were obtained from the reference proteome of *A. thaliana* Col-0 TAIR10 as described previously using hmmsearch [73] and the extended NB-ARC Hidden Markov Model [5]. Of these 161 NLRs, 127 had AlphaFold-predicted structures available on AlphaFoldDB [57][117]. The training dataset used for LRRpredictor, which contained manual annotations of LRR motif positions, was downloaded from supplemental data of [69]. We ran AlphaFold 2 prediction on a supercomputer cluster with default parameters and selected the best-scoring model for further analysis. We have included the protein amino acid sequences and corresponding `pdb` files in the GitHub repository where we host all the code used in this study.

Outline of methods

Our treatment of protein structures follows the outline below. Figure 6.1 shows the results of steps $1-4$, while Figure 6.5 shows the results of steps $5-6$.

1. ***Obtaining the backbone***. Given the space curve $\gamma(t)$ representing the positions of the α-carbons, obtain a smoothed backbone curve $\gamma_\sigma(t)$ by convolving γ with a Gaussian.

2. ***Parallel transport & framing***. Parallel-transport a frame along the backbone to produce, at each position t, an orthonormal basis for the plane normal to $\gamma'_\sigma(t)$. This yields a two-dimensional coordinate system $A(t)$ for each t.

3. ***The flattened representation***. For each t, compute the coordinates of $\gamma(t) - \gamma_\sigma(t)$ according to $A(t)$. This produces a two-dimensional "flattened" curve $\varphi(t)$ representing the position of γ relative to its backbone.

4. ***Cumulative winding number***. Compute the cumulative winding number $W_\varphi(t)$ of φ about the origin.

5. ***Secant line statistics; median slope***. Compute the median slope of secant lines to W_φ to infer the number m of residue positions per helical repeat unit in the LRR domain.

6. ***Piecewise-linear regression & gradient descent***. By gradient descent on an appropriate loss function, find a piecewise-linear regression of W_φ with slopes alternating

between zero and m. Regions of the regression with slope m correspond to solenoidal regions of the protein structure.

Obtaining the backbone

Let $\gamma(t)$, $t \in \{0, \ldots, n\}$ be a discrete space curve representing the positions of the α-carbons in a protein structure. This curve can be represented as three scalar functions of t: $\gamma(t) = (\gamma_x(t), \gamma_y(t), \gamma_z(t))$. Let g_σ be the mean-zero Gaussian distribution with standard deviation σ:

$$g_\sigma(t) = \frac{1}{\sigma\sqrt{2\pi}} e^{-t^2/(2\sigma^2)}. \tag{6.1}$$

We define the "backbone" to the structure

$$\gamma_\sigma(t) := \Big((g_\sigma \star \gamma_x)(t), (g_\sigma \star \gamma_y)(t), (g_\sigma \star \gamma_z)(t) \Big), \tag{6.2}$$

where $\star$ is the convolution, defined $(p \star q)(t) := \sum_s p(t)q(t-s)$, where the sum is over all sensible indices s. Throughout in our computations, we set $\sigma = 20$.

Parallel transport & framing

First, we compute the tangent vector $\gamma'_\sigma(t)$ to the backbone by convolving γ with the derivative of a Gaussian, i.e. with

$$g'_\sigma(t) = -\frac{t}{\sigma^3\sqrt{2\pi}} e^{-t^2/(2\sigma^2)}. \tag{6.3}$$

This is a standard technique [74] for defining derivatives of discrete data, since convolution associates with differentiation as $(d/dt)\big(p \star q\big) = \big((dp/dt) \star q\big) = \big(p \star (dq/dt)\big)$. In order to measure the winding of γ around its backbone γ_σ, we need a consistent representation of the position of γ relative to γ_σ; in effect, we need to "straighten" the backbone and carry γ along for the ride.

Now that we have $\gamma'_\sigma(t) = (g'_\sigma \star \gamma)(t)$, we will produce a sequence of orthonormal bases for the planes orthogonal to $\gamma'_\sigma(t)$ at each residue t. Our method starts with a frame at $t = 0$ and parallel-transports it along the backbone as follows:

1.25Given $\gamma'_\sigma(0)$, the initial tangent to the backbone, let $A(0)$ be any 3×2 real matrix with orthonormal columns such that $A(0)^T \gamma'_\sigma(0) = \mathbf{0}$ (i.e., the columns of $A(0)$ complete $\gamma'_\sigma(0)$ to an orthonormal basis for $\mathbb{R}^3$). Given $A(t-1)$, let $B(t)$ be the matrix whose columns are orthogonal projections of the columns of $A(t-1)$ onto the complement of $\gamma'_\sigma(t)$. Symbolically,

$$B(t) = A(t-1) - \frac{1}{\|\gamma'_\sigma(t)\|^2} \gamma'_\sigma(t)^T A(t) \gamma'_\sigma(t). \tag{6.4}$$

The columns of $B(t)$ are likely not orthonormal. Let $A(t)$ be the 3×2 matrix with orthonormal columns that is closest (in the Frobenius norm) to $B(t)$. Numerically,

$A(t)$ is found by computing the SVD of $B(t)$ and replacing its singular values with 1's (the standard solution to the "Orthogonal Procrustes Problem" [120, 59]). Note that the columns of $A(t)$ span the same subspace as those of $B(t)$, so $A(t)$ has columns guaranteed orthogonal to $\gamma'_\sigma(t)$. Repeat steps 2 and 3 for $t = 1, \ldots, n$.

The flattened representation

The flattened representation is now a plane curve $\varphi(t) = A(t)^T\Big(\gamma(t) - \gamma_\sigma(t)\Big)$. It can be thought of as γ from the perspective of an observer traveling along the backbone γ_σ and oriented according to the frames $A(t)$.

Cumulative winding number

For a continuous-time plane curve $z(t) = \big(x(t), y(t)\big)$ with polar representation $\big(r(t)\cos(\theta(t)), r(t)\sin(\theta(t))\big)$, the winding number is defined

$$W_z(t) = \frac{1}{2\pi}\int_0^t \theta'(s)ds = \frac{1}{2\pi}\int_0^t \frac{x(s)y'(s) - y(s)x'(s)}{x(s)^2 + y(s)^2}ds. \tag{6.5}$$

This quantity tracks the total number of rotations accumulated by a ray pointing at $z(s)$, as s moves in the interval $[0, t]$.

In our case, given the discrete plane curve $\varphi(t) = \big(x(t), y(t)\big)$, we define a discrete version of the cumulative winding number by

$$W_\varphi(t) = \frac{1}{2\pi}\sum_{s=1}^{t} \arctan\left(\frac{y(s)x(s-1) - x(s)y(s-1)}{x(s)x(s-1) + y(s)y(s-1)}\right). \tag{6.6}$$

The summand accumulates the angle between rays to consecutive points $\varphi(s-1)$ and $\varphi(s)$ along the discrete curve. Figure 6.1 provides a graphical example of the backbone, parallel-transported normal bundle, flattened representation, and cumulative winding number plot.

Secant line statistics; median slope

To make piecewise-linear regression tractable, we remove slope as an optimization parameter, and instead infer it from the statistics of secant lines to W_φ. First we choose parameters $0 < d < D$ (in the `median_slope` method, these are `small = 100` and `big = 250`, respectively). We will consider only secant lines with endpoints a, b where $d \le b - a \le D$. Associated to such a secant line is a slope $m_{a,b} = \big(W_\varphi(b) - W_\varphi(a)\big)/\big(b - a\big)$, and a score $S_{a,b}$ computed as follows. First define

$$R_{a,b} = \sum_{t=a}^{b}\left[\Big(W_\varphi(t) - m_{a,b}t\Big) - \frac{1}{b-a+1}\sum_{s=a}^{b}\Big(W_\varphi(s) - m_{a,b}s\Big)\right]^2. \tag{6.7}$$

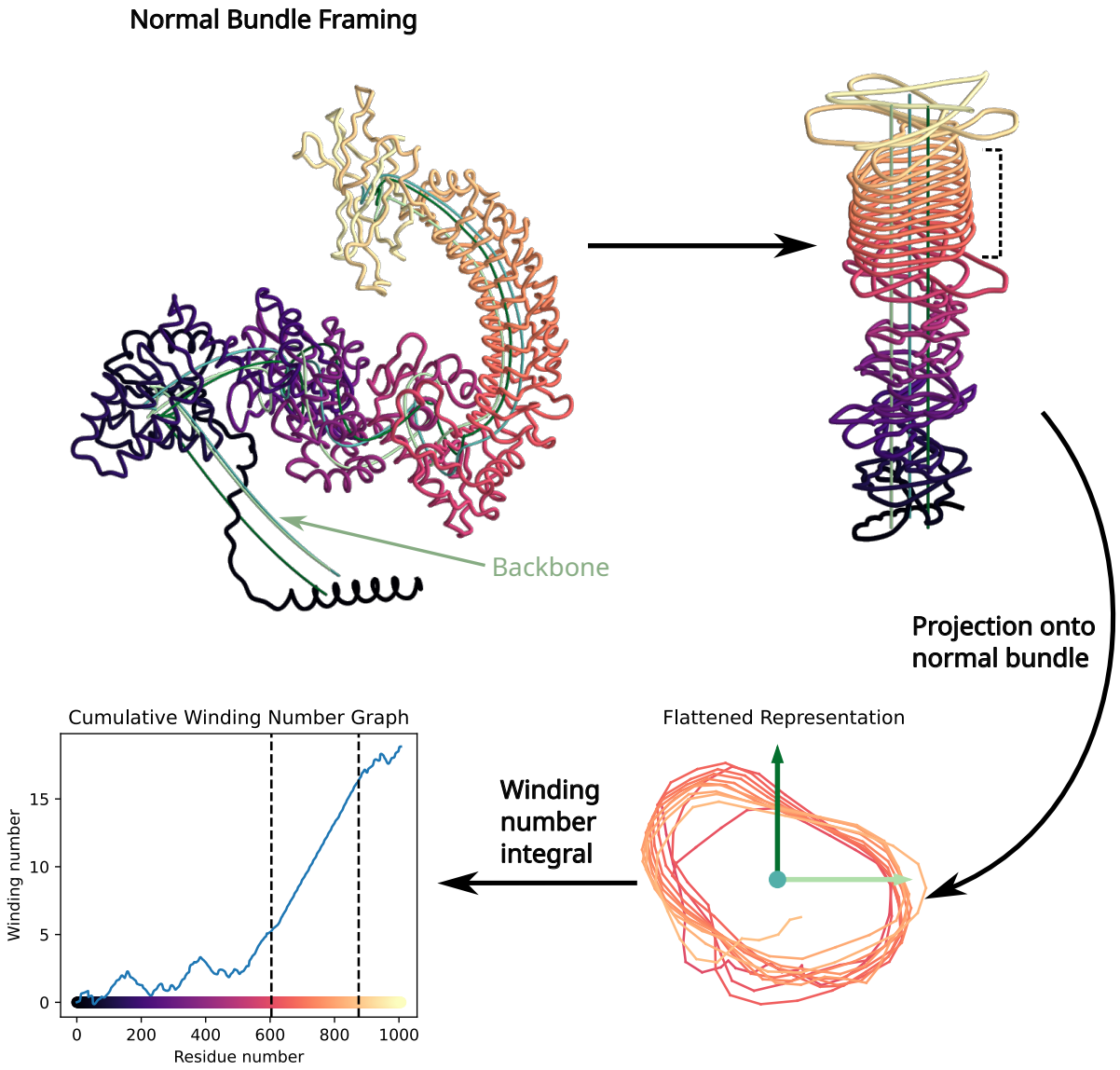

Figure 6.1: Embedding of protein backbone curve into normal bundle followed by projection onto an orthonormal frame yields a 2D curve containing a flattened slinky shown in lower right. The cumulative winding number, computed using the classic formula from calculus, is computed from the projection. Sloped linear segments of the winding number curve indicate coiling. Protein shown is *A. thaliana* NLR with TAIR [10] ID AT3G44400.2.

In other words, $R_{a,b}$ measures the total squared deviation of $\Big(W_\varphi(t) - m_{a,b}t\Big)$ away from its mean; we have $R_{a,b} = 0$ if and only if W_φ coincides with its secant line on $t \in [a, b]$. Now let the score $S_{a,b} = (b-a)/(1+R_{a,b})$, rewarding long secant lines and penalizing deviations from linear behavior.

The "median slope" is chosen by a voting process. First we determine the minimum and maximum slopes, call them m and M. We create $\sqrt{N}$ score bins, where N is the number of secant lines, i.e. the number of pairs (a, b) with $0 \le a, b \le n$ and $d \le b - a \le D$. For each secant line with endpoints a, b, its score $S_{a,b}$ accumulates in the bin with index $\Big\lfloor \big(m_{a,b} - m\big)/\big(M - m\big) \Big\rfloor$. After this procedure, the slope returned is $m + \big(i/\sqrt{N}\big)(M - m)$, where i is the index of the bin with largest score. We use this slope in subsequent regression tasks.

Our "median slope" computation is conceptually similar to the Hough transform [29], a computer vision method for detecting segments in images via a voting process across a parametrized space of lines in the plane.

Piecewise-linear regression & gradient descent

The "median slope" m associated to the winding W_φ approximates the reciprocal of residues per repeat unit in the LRR domain – as residue position t changes by m, winding number increases by 1, i.e. φ completes one revolution around the origin. To annotate the domain in which W_φ exhibits this linear, slope-m behavior, we fit a piecewise-linear, discontinuous function which is constant in the pre-LRR region, slope-m in the LRR domain, and constant in the post-LRR region. More precisely, associated to a choice of breakpoints $(0 = a_0 < a_1 < \cdots < a_k = n)$ is a regression function that is constant on $[a_0, a_1)$, slope-m on $[a_1, a_2)$, constant on $[a_2, a_3)$, and so on. Most of the cumulative winding number plots were well-approximated with $k = 2$ (two breakpoints); we discuss larger k below.

We define a loss function, similar in spirit to 6.7, as follows. First, for a function $f(t)$ and endpoints $a < b$, define $V_{a,b}(f)$ to be the total squared deviation of f from its mean on $[a, b)$:

$$V_{a,b}(f(t)) = \sum_{t=a}^{b-1} \left(f(t) - \frac{1}{b-a} \sum_{s=a}^{b-1} f(s) \right)^2. \tag{6.8}$$

Choose constants C, D, and define the loss associated to the partition $(a_0, \ldots, a_k)$:

$$L(a_0, \ldots, a_k) = \sum_{j=0}^{k/2} \left(C\, V_{a_{2j}, a_{2j+1}}\Big(W_\varphi(t)\Big) + D\, V_{a_{2j+1}, a_{2j+2}}\Big(W_\varphi(t) - mt\Big) \right). \tag{6.9}$$

The loss is a weighted measurement of the total squared deviation between W_φ and the regression function we are fitting, with the weights C and D determining how harshly we penalize deviations from linearity (slope-m behavior) in the LRR region. In our code, we

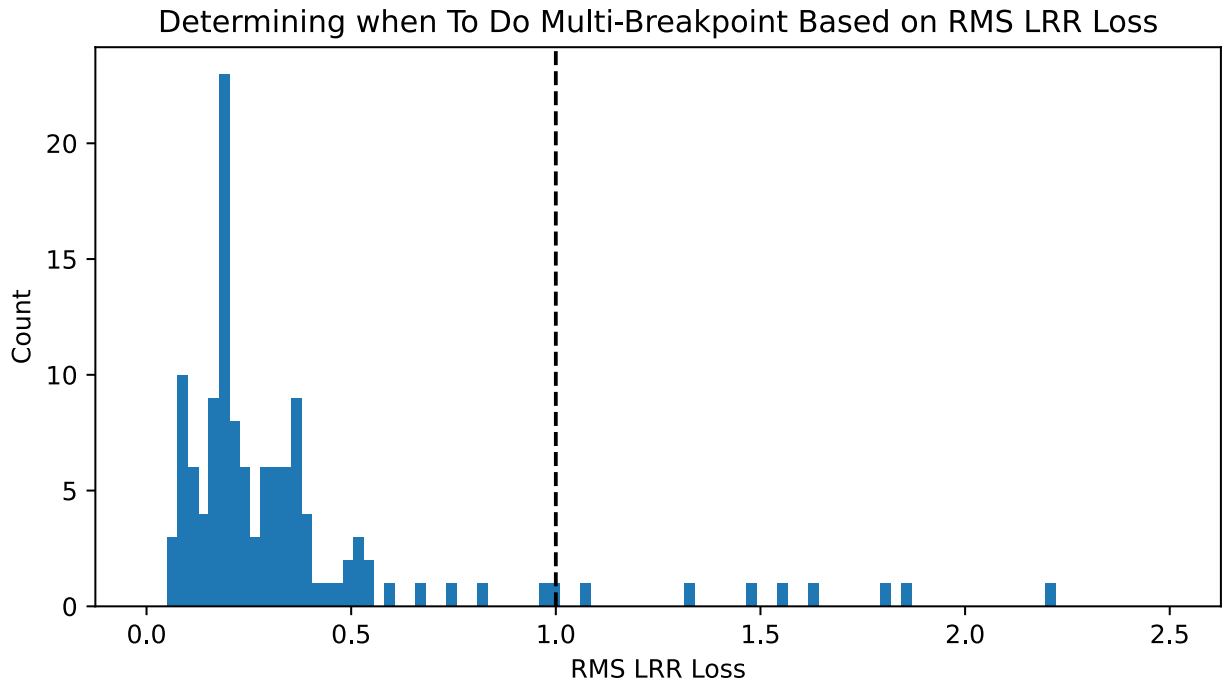

Figure 6.2: We determine when to redo the regression using 4 breakpoints by examining the RMS of the LRR component of the loss. This term is above our threshold of 1 for 9/127 of the proteins in *A. thaliana*.

found that $C = 1$ and $D = 1.5$ worked well. Our optimization problem now becomes: find $(a_0, \ldots, a_k)$ minimizing $L(a_0, \ldots, a_k)$.

We solve the optimization problem by gradient descent on L: we form a finite-difference gradient ∇L whose j^{th} entry is

$$(\nabla L)_j = \Big(L(a_0, \ldots, a_j + 1, \ldots, a_k) - L(a_0, \ldots, a_j, \ldots, a_k)\Big), \tag{6.10}$$

choose a learning rate $\epsilon > 0$, increment the vector of breakpoints by $-\epsilon \nabla L$, and iterate.

Refinements And Alternatives

Loss Histograms & Four-Breakpoint Regressions

A small number (ten out of 127) of proteins in our dataset contained hairpin loops or other localized deviations from solenoidal geometry in the LRR region, and regressions with $k = 2$ breakpoints were not satisfactory. We found the standard deviation of the difference between W_φ and the regressing function inside the LRR region, i.e. $V_{a_1,a_2}/(a_2 - a_1)$, is high in such cases. Figure 6.2 shows the distribution of these values. We repeat the regression with four breakpoints, instead of two, to deal with these edge cases.

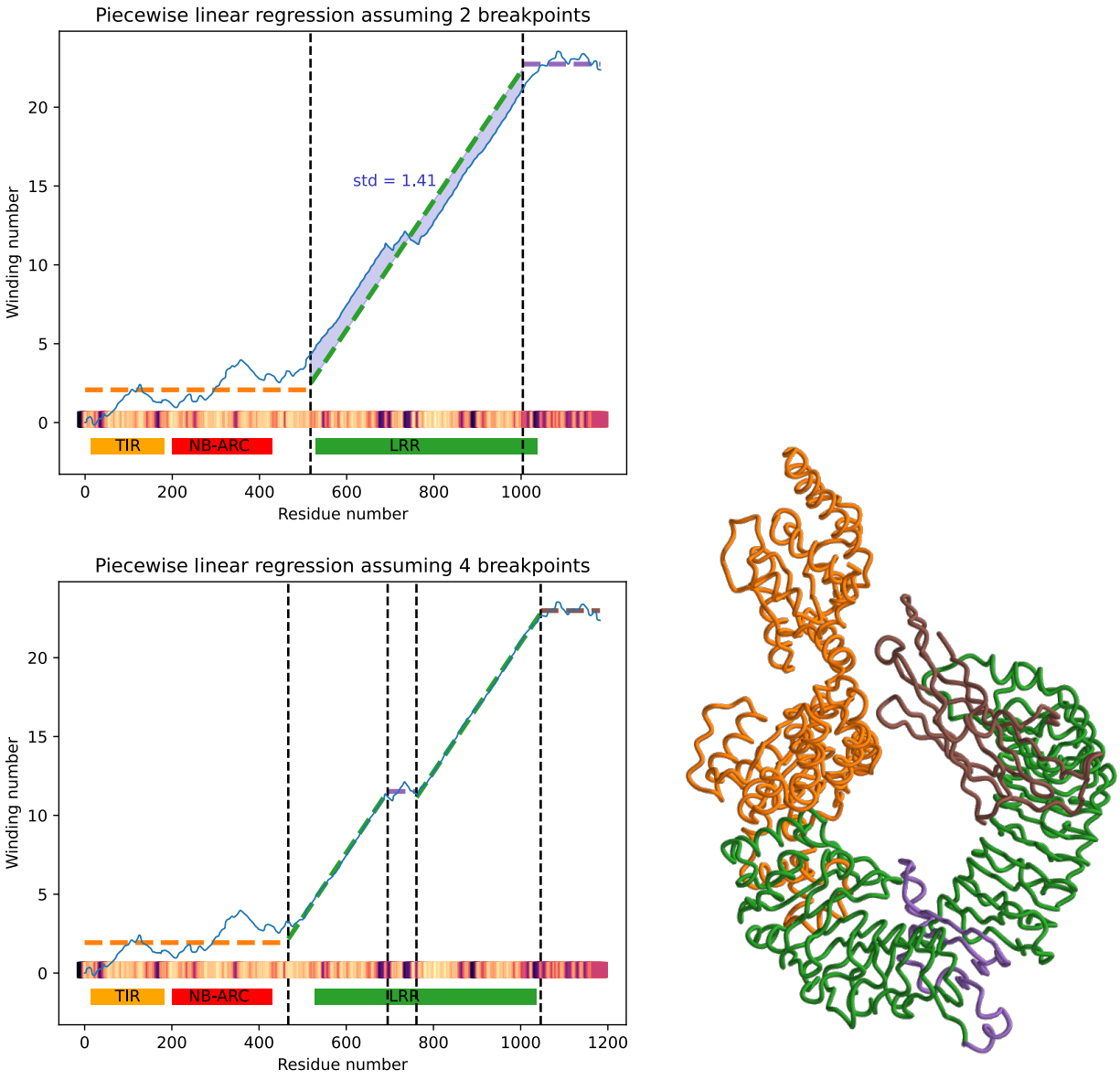

Figure 6.3: Four breakpoint piecewise linear regression enables detection of a non-coiling structure (highlighted in purple at right) which deviates from the usual coiling in the LRR domain. Below regression plot, a heat map shows the pLDDT (predicted local distance difference test), a per-residue confidence measure given by AlphaFold 2 which is elevated in the non-coiling region. Bottom of plot shows HMM-based InterPro domain annotations which fail to detect non-coiling region within LRR domain. TAIR ID is AT1G72840.2.

Figure 6.3 shows the result of fitting a regressing function with four, instead of two breakpoints.

Laplacian Circular Coordinates

In the previous sections, we used piecewise linear regression on the cumulative winding number to isolate the LRR domain. In the process, we estimated the winding number, which can also give us *instantaneous phase*, or the angle along each loop, on the LRR domain sequence. In this section, we briefly describe another technique based on graph theory for estimating instantaneous phase of LRR regions, which we evaluate alongside the parallel transport method in Section 6.

Before setting up the graph, we perform some preprocessing to make LRR solenoid region as circular as possible. First, we nullify some of the torsion by once again computing the tangent vectors on the LRR solenoid. This time, however, we set $\sigma = 1$, and convolve $\gamma(t)$ with $g_1'(t)$ instead of $g_{20}'(t)$ to preserve the loop structures. To further accentuate periodic features, we perform a multivariate sliding window embedding [114] of window size 24 (roughly the length of the LRR period) with delay time 1 on each component of the tangent vector field. The formula for such a sliding window embedding of some sequence $f[t]$ is

$$\mathrm{SW}_1^{24} f[t] := \begin{bmatrix} \gamma[t] \\ \gamma[t+1] \\ \gamma[t+2] \\ \vdots \\ \gamma[t+24] \end{bmatrix} \in \mathbb{R}^{24+1} \tag{6.11}$$

We concatenate together $\mathrm{SW}_1^{24}\, \gamma_{i\sigma}'$ for each of the three components of the tangent vector $\gamma_{i\sigma}'$, resulting in a sequence in 75-dimensional Euclidean space. We then construct a 50-mutual-nearest-neighbors graph on the sliding window embedding.

From the mutual-NN graph we compute leading eigenvectors of the unweighted graph Laplacian [20]. An example is shown in Figure 6.4. Intuitively, the graph Laplacian is a generalization of a discrete second derivative operator to graphs. For the same reason that sines and cosines are eigenfunctions of the second derivative operator with associated eigenvalues proportional to the frequency, eigenvectors of the graph Laplacian on a graph of a circle are sine-cosine pairs, up to a phase, that go through an integer number of cycles over one revolution of the circle, and lower frequency pairs have smaller eigenvalues [45]. We expect a near circular graph in the mutual-NN graph in the periodic LRR region, and the Laplacian eigenvectors are known to degrade gracefully in the presence of imperfections. Therefore, we expect the two eigenvectors with the smallest eigenvalue to be approximately periodic and $\pi/4$-phase shifted. If we use the two entries of these eigenvectors as x- and y-coordinates, respectively, we obtain a projection of the LRR coil onto a circle winding in the plane. Our phase estimation θ along the LRR coil is simply obtained as $\theta = \tan^{-1}(\frac{y}{x})$, as shown in Figure 6.4 below.

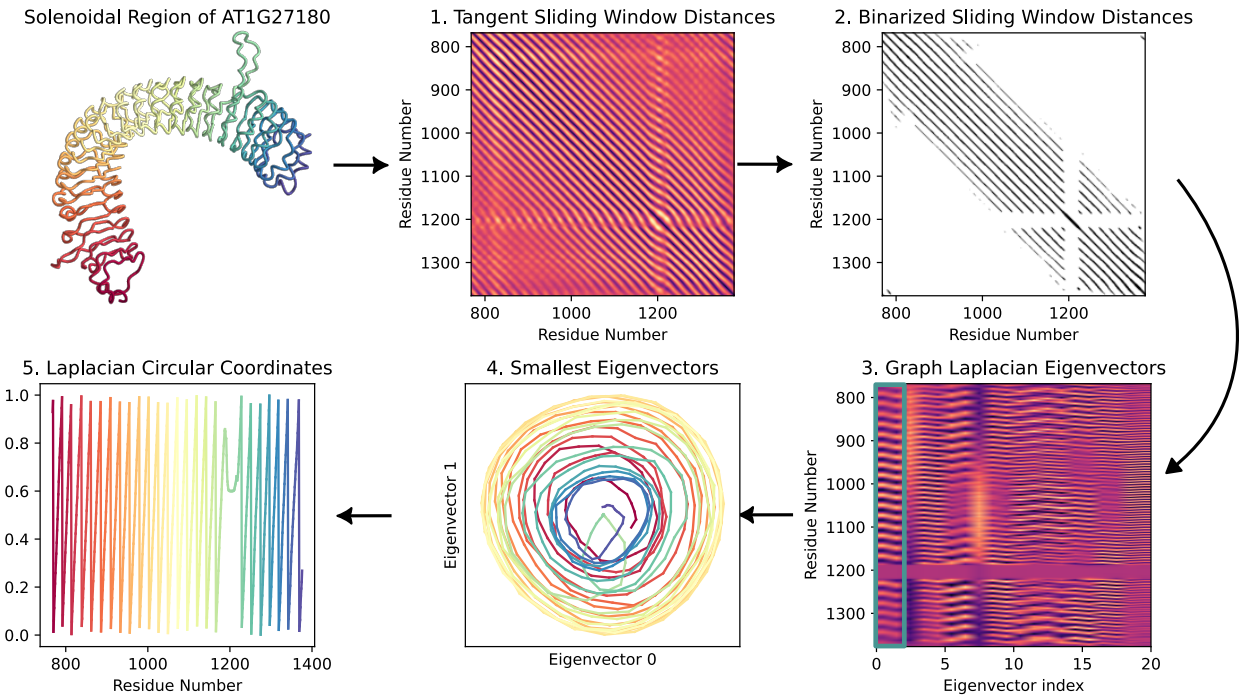

Figure 6.4: Graph Laplacian eigenvectors of mutual nearest neighbor graph on LRR solenoid curve tangent vectors. LRRPredictor residues are shown as blue horizontal lines on eigenmatrix plot. The 0th and 1st eigenvectors have period matching the expected period of the solenoid as determined by LRRPredictor. Leading eigenvectors of graph Laplacian are periodic and are $\pi/4$-phase shifted, thereby yielding projections of LRR coil onto a winding around a circle in a 2D-plane. Phase estimation using the formula $\theta = \tan^{-1}(\frac{y}{x})$ of LRR coil at bottom taking values between $-\pi$ and π.

We note that a similar phase-estimation scheme with the graph Laplacian of mutual nearest neighbors has been used to order photographs along a loop [2] and to parameterize periodic videos [114]. Furthermore, a spiritually similar but more computationally intensive topological phase estimation based on cohomology [27, 85] has been used to recognize patterns in motion capture data [118] and to detect head orientation from neural data [**rybakken2019decoding**].

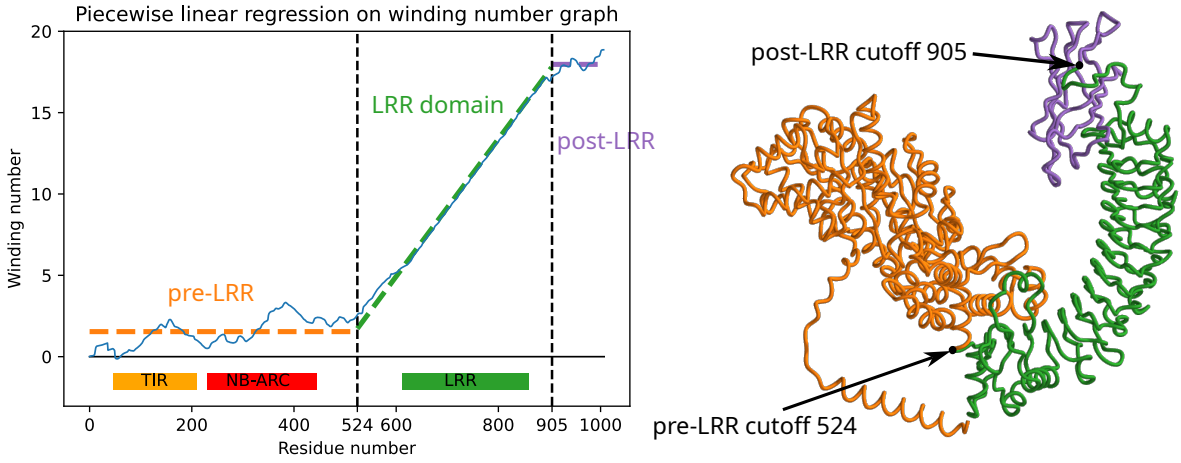

Figure 6.5: A discontinuous clipped ReLU function is regressed on the graph of the winding number function for *A. thaliana* NLR with TAIR [10] ID AT3G44400.2. The breakpoints of the regression yields the start and end positions of the LRR domain, highlighted in green. InterPro [12] domain annotations are shown below regression plot.

Results

Cumulative winding number reveals errors made by ML-based LRR repeat unit delineator

We ran the LRR annotation tool LRRPredictor [69] on the 127 NLRs from *A. thaliana* to obtain predicted locations of the LRR motif "LxxLxL." Let $R_1, \ldots, R_\ell$ denote the starting residues for the LRR motifs predicted by LRRPredictor. The analogous measurement in our model is to record the residues at which our cumulative winding number W_φ crosses integers.

To compare the two prediction schemes, we evaluate our cumulative winding number at the residues returned by LRRPredictor. That is, we form the list of numbers $(W_\varphi(R_1), \ldots, W_\varphi(R_\ell))$. If the models are in agreement, the running difference $(W_\varphi(R_2) - W_\varphi(R_1), \ldots, W_\varphi(R_\ell) - W_\varphi(R_{\ell-1})$ should equal the all-ones vector $(1, \ldots, 1)$ (that is, the structure should wind exactly once around the core between residues R_{j-1} and R_j). The "discrepancy"

$$D(R_1, \ldots, R_\ell) := \sqrt{\sum_{j=2}^{\ell} \Big(W_\varphi(R_j) - W_\varphi(R_{j-1}) - 1 \Big)^2} \tag{6.12}$$

quantifies the extent to which this is not the case. A number of LRRPredictor outputs contained false predictions in which consecutive motif start sites R_j and R_{j-1} appear close together – often only a couple residues apart. Such duplicate predictions result in a high discrepancy $D(R_1, ..., R_\ell)$ because the difference $W_\varphi(R_j) - W_\varphi(R_{j-1})$ as computed in formula

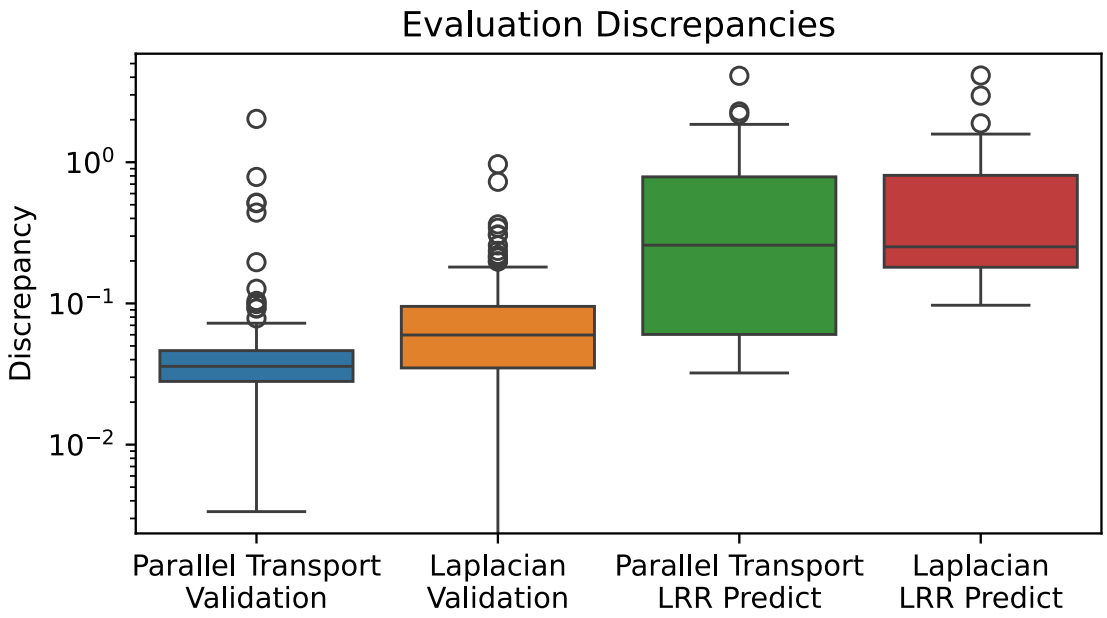

Figure 6.6: Discrepancies for LRRpredictor outputs on 127 *A. thaliana* (green and red) NLRs are higher than those for manually-annotated LRR repeat units used as the training set for the LRRpredictor model (blue, orange). Thus, the cumulative winding number computation faithfully recapitulates the periodicity of the LRR coil.

(6.12) above is close to 0.

To test the validity of our winding number computation, we ran the discrepancy computation on the LRRPredictor outputs on the 127 *A. thaliana* reference proteome NLRs as well as AlphaFold 2 structures for the training dataset for LRRpredictor, a manually-annotated "ground truth" dataset of LRR motifs on 172 experimentally-derived LRR structures taken from Protein Data Bank. These PDB protein structures were derived from a diverse set of organisms comprising bacteria, fungi, plants, and animals.

We found consistently low discrepancy values for the ground truth set with mean 0.127. By comparison, *A. thaliana* NLRome discrepancy values were generally low with mean 0.373, but exhibited higher values in cases where LRRpredictor made mistakes. Figure 6.6 below shows a pair of overlaid histograms comparing discrepancy values for both the validation dataset and NLRome dataset (S1 and S2 Table). The discrepancy values are much lower on the LRRPredictor ground truth dataset compared to the NLRome dataset, implying that our technique makes fewer mistakes than LRRPredictor does on new data. Figure 6.7 demonstrates how the discrepancy is able to catch duplicate motif predictions made by LRRPredictor. These results demonstrate not only the winding number's ability to accurately model the LRR coil, but also its generalizability to non-NLR LRR's derived from species other than *A. thaliana*.

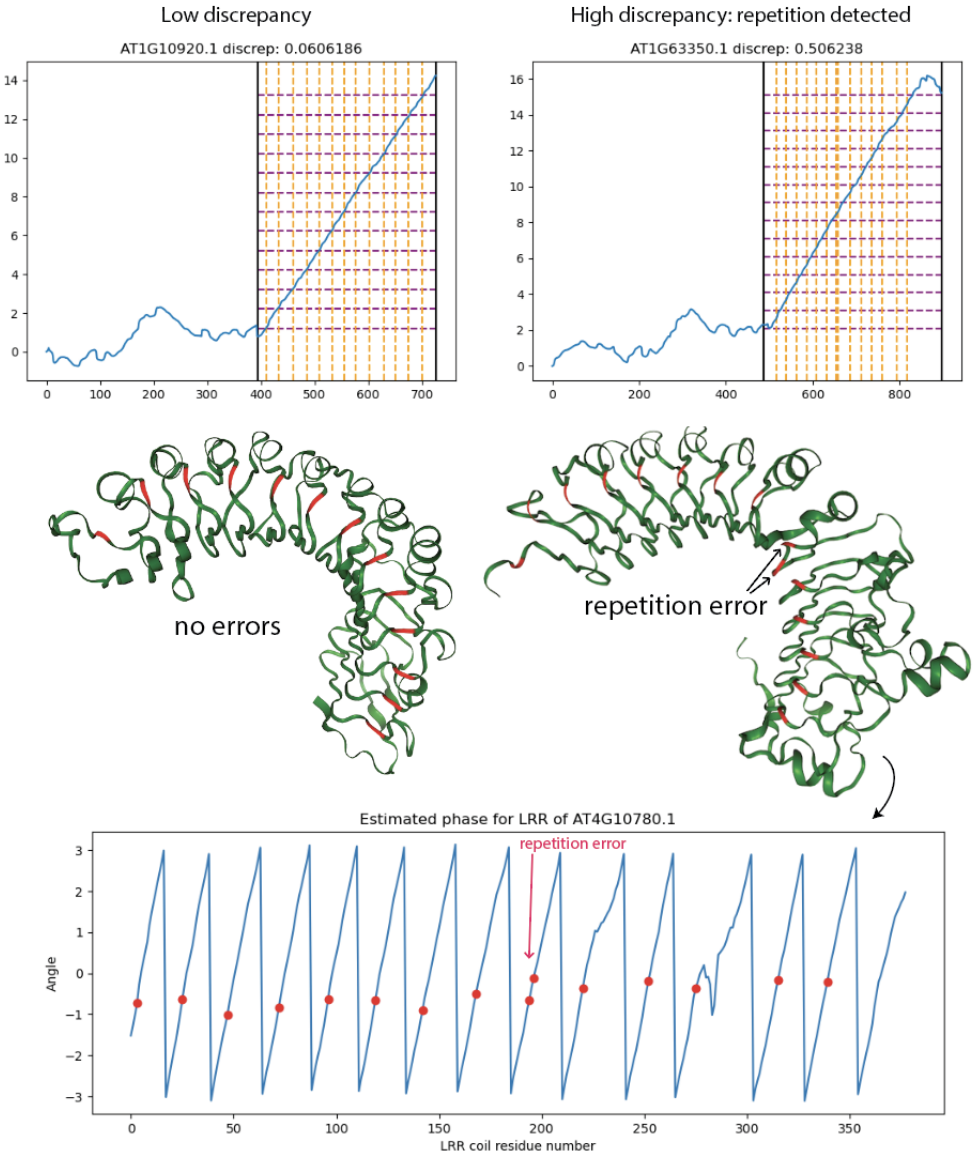

Figure 6.7: **LRRPredictor discrepancy computation reveals proteins with erroneously repeated predictions**. NLRs with high-discrepancy LRRPredictor outputs tend to carry repetition errors or missing motif annotations. Orange vertical lines overlaid on winding number plot depict LRRPredictor residues, while purple horizontal lines depict the integer-spaced grid which best approximates the winding number graph evaluated at LRRPredictor residues. A repetition error can be seen in the grid representation as a doubled orange line around residue 685. At bottom, LRRPredictor residues are mapped onto graph Laplacian eigenvector phase estimation, revealing an pair of duplicates with adjacent phase.

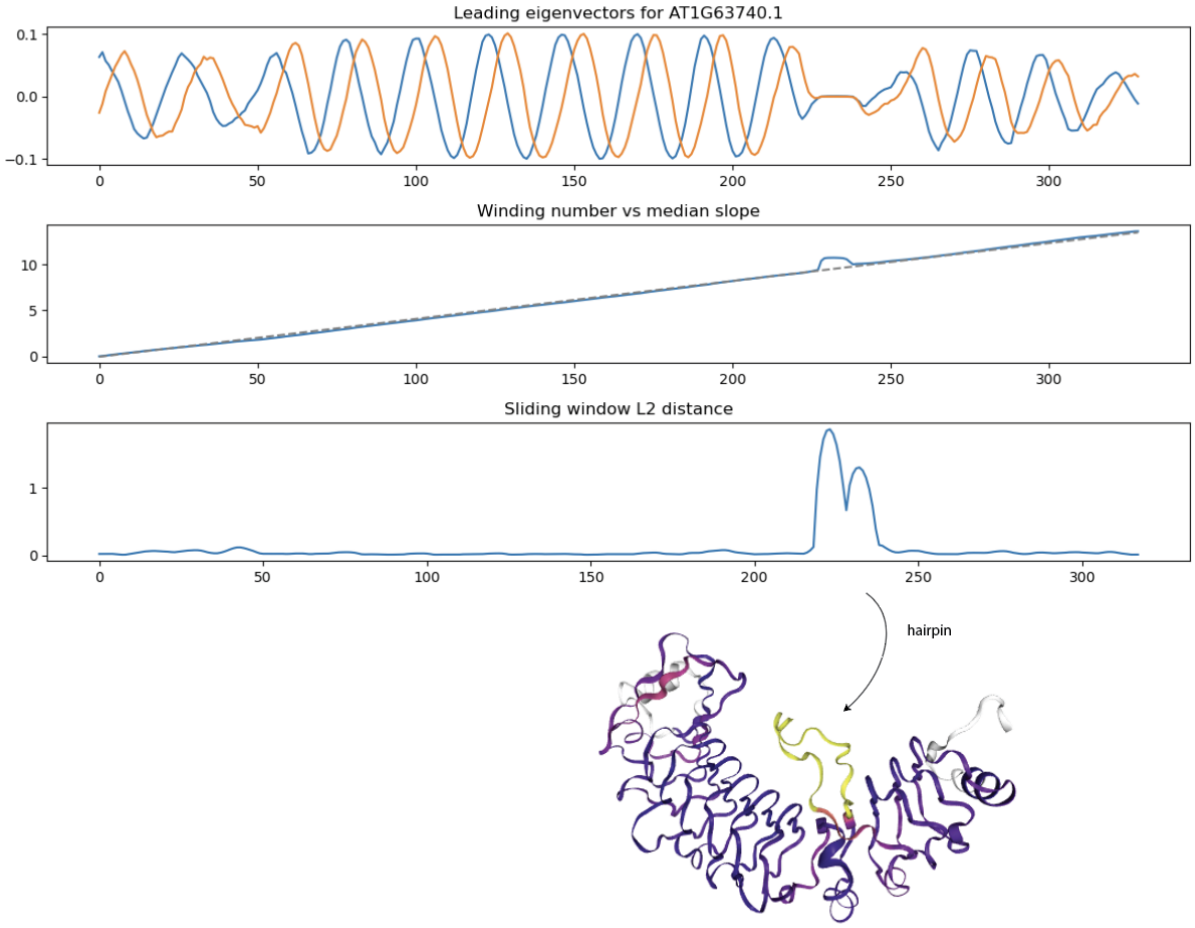

Figure 6.8: Sliding window L^2 distance (SWL2D) from winding number to median secant line detects small hairpins/insertions in LRR coil domain. Structure at bottom is colored according to SWL2D where yellow values are higher.

Structural anomaly detection by sliding window L^2 distance from Laplacian eigenvector winding number to line

Many LRR coils have hairpin loops and other structural anomalies which deviate from coiling. In these anomalous regions, the leading eigenvectors deviate from their usual periodic behavior. Applying the winding number formula (Equation 6.6) above to the pair of leading graph Laplacian eigenvectors leads to a cumulative winding number within the LRR domain which is better able to discern small hairpins compared to the previous winding number computation based on normal bundle projection. As shown in Figure 6.8 below, we detect a small hairpin as a spike in L^2 distance between the winding number and its median slope.

Discussion

The emergence of AlphaFold 2 has catalyzed a paradigm shift in protein structure prediction, facilitating access to genome-wide high-quality structural models. Traditional sequence homology-based domain annotation techniques, like LRRPredictor, often face challenges with LRRs, especially in proteins with high sequence divergence. While evolutionary divergence might veil the sequence homology of LRR units, their core structural topology, characterized by 20-30 amino acid stretches typically involved in protein-protein interactions, often remains conserved, acting as a distinct structural signature.

This study uses AlphaFold 2 to generate a 3D space curve from a protein sequence, which subsequently is projected into the 2D plane by identifying a series of "slinky" cross-sections. Through computing the cumulative winding number on the resultant 2D curve and employing piecewise linear regression, the linearly sloped region, identified as the LRR domain, is discerned. Our method pivots on the application of geometric data analysis to illuminate structural motifs that remain elusive to sequence analysis alone.

The use of geometric and topological concepts in our method aligns with previous studies that have explored Topological Data Analysis (TDA) in protein structure and dynamics [109, 15]. For instance, SINATRA Pro has been used to identify biophysical signatures in protein dynamics by detecting topological differences between protein structures [109]. Similarly, TopologyNet integrates TDA with deep learning for biomolecular property predictions [15]. Our approach builds on these foundational ideas by leveraging large-scale AI/ML-derived databases like AlphaFoldDB, showcasing the potential of combining AI-based structural predictions with geometric and topological analyses for advanced domain annotation. The amalgamation of advanced protein structure prediction technologies and mathematical models, as demonstrated in our approach, underscores the potential for widening our understanding of protein function across varied biological systems.

Our method yields several kinds of precise results: (a) it identifies the start and end sites of the LRR domain with greater accuracy than HMM-based methods, (b) it annotates repeat units more reliably than the existing LRRPredictor, (c) it identifies misannotations by other annotation/prediction tools, and (d) it reveals structural anomalies within the LRR domain that deviate from conventional coiling behaviors. These findings not only underscore the utility of our approach but also present a robust framework for delving into the intricate structural patterns intrinsic to LRR domains.

While we benchmarked our work on LRR domains in NLR proteins, the intrinsic methodology has the capacity for broader applications, likely extending to other linear solenoid protein domains like armadillo (ARM), tetratricopeptide (TPR), and ankyrin (ANK) repeats, all of which feature distinctive repeat sequences and structural configurations. However, the method is unlikely to work well on circular solenoid domains such as beta propellers (e.g. WD40) because, unlike linear solenoids, those structures do not consistently wind around around a core curve.

Our method does come with limitations. For instance, while it can detect non-coiling structural anomalies within the LRR domain, the origin, authenticity, and potential function-

ality of these regions remain ambiguous. Moreover, our structure-based annotation method, albeit effective for domains with a straightforward geometric description like LRRs, might not be universally applicable to other protein domains without developing a new geometric model tailored to them. This underscores a potential limitation when juxtaposing sequence-based versus structure-based domain annotation, highlighting a future avenue warranting exploration: developing geometric models for other protein domains.

Chapter 7

Conclusion

This dissertation has traversed a diverse landscape of biological research, utilizing mathematical and computational techniques to uncover intricate details in neural data and protein structures. Each chapter has contributed uniquely to our understanding of complex biological systems, demonstrating the power of interdisciplinary approaches in addressing fundamental scientific questions.

In Chapter 4, we focused on the application of persistent cohomology to neural data, specifically within the spatial representation system of the brain. By employing topological methods, we were able to uncover low-dimensional structures embedded in high-dimensional neural activity spaces. This approach not only provided insights into the organization and function of neural circuits but also demonstrated the robustness of topological data analysis in revealing underlying patterns in complex datasets. The findings highlighted the potential of persistent cohomology to decode animal trajectories from neural activity, revealing the intricate relationship between neural representations and physical space.

Transitioning to Chapter 6, we shifted our focus from neural data to the structural annotation of leucine-rich repeat (LRR) domains in proteins. Leveraging the high-quality structural predictions made possible by AlphaFold 2, we integrated geometric and topological analysis to enhance the precision of LRR domain annotations [57]. This novel method involved flattening the three-dimensional structure of proteins into a two-dimensional plane and calculating the winding number to accurately delineate repeat units. Our approach addressed the limitations of traditional sequence-based annotation methods, which often struggle with highly divergent sequences and irregular motifs [69]. By incorporating structural information, we were able to detect structural anomalies and improve the accuracy of domain boundary predictions.

Our method yields several precise results: it identifies the start and end sites of the LRR domain with greater accuracy than HMM-based methods, annotates repeat units more reliably than existing machine learning-based tools, identifies misannotations by other tools, and reveals structural anomalies within the LRR domain that deviate from conventional coiling behaviors. This comprehensive approach allowed us to correct mistakes made by existing annotation tools and enabled the automated detection of hairpin loops and structural

irregularities in the solenoid. We applied our methods to 127 predicted structures of LRR-containing intracellular innate immune proteins in the model plant *Arabidopsis thaliana* and validated our results against a benchmark dataset of 172 manually-annotated LRR domains.

The broader implications of our work extend to other linear solenoid protein domains such as armadillo (ARM), tetratricopeptide (TPR), and ankyrin (ANK) repeats, which feature distinctive repeat sequences and structural configurations. However, our method is unlikely to work well on circular solenoid domains such as beta propellers (e.g., WD40) because, unlike linear solenoids, those structures do not consistently wind around a core curve. Nonetheless, our structure-based annotation method, driven by geometric and topological insights, provides a robust framework for understanding protein function and structural motifs across varied biological systems.

Throughout this dissertation, we have seen the unifying power of mathematical and computational techniques in advancing biological research. The application of persistent cohomology to neural data provided a new lens through which to view the spatial representation system, uncovering topological features that were previously hidden. Similarly, the integration of structural data with geometric analysis in protein annotation opened new avenues for understanding protein functionality and the evolution of immune receptors.

Our work on LRR domains in Chapter 6 exemplifies the potential of combining AI-based structural predictions with geometric and topological analyses. This approach not only enhanced the precision of domain annotations but also revealed structural patterns that are critical for understanding protein-protein interactions in immune responses. The successful application of these methods to proteins from *Arabidopsis thaliana* demonstrates their broader applicability to other solenoid domains, paving the way for future research in protein structure analysis [15, 109].

The findings and methods presented in this dissertation have the potential to inform future research, offering new tools and perspectives for exploring complex biological systems. By leveraging the strengths of persistent cohomology and geometric analysis, we have pushed the boundaries of what is possible in both fields, providing more accurate and insightful tools for researchers.

In conclusion, the work presented in this dissertation exemplifies the power of interdisciplinary research in addressing complex scientific challenges. By bridging the gap between mathematics, computer science, and biology, we have developed novel methods that enhance our understanding of neural and protein structures. This journey has significantly advanced our knowledge in these fields, and the principles and techniques developed here will continue to inform and enhance future research. The integration of computational and biological research will remain a key driver of innovation, contributing to a deeper understanding of complex biological systems.

Bibliography

[1] Hassan Arbabi. "Introduction to Koopman Operatory Theory of Dynamical Systems". In: (2018).

[2] Hadar Averbuch-Elor and Daniel Cohen-Or. "Ringit: Ring-ordering casual photos of a temporal event". In: *ACM Transactions on Graphics (TOG)* 34.3 (2015), pp. 1–11.

[3] Andrey Babichev, Dmitriy Morozov, and Yuri Dabaghian. "Replays of spatial memories suppress topological fluctuations in cognitive map." In: *Network neuroscience (Cambridge, Mass.)* 3.3 (2019), pp. 707–724. DOI: 10.1162/netn_a_00076.

[4] Andrey Babichev, Dmitriy Morozov, and Yuri Dabaghian. "Robust spatial memory maps encoded by networks with transient connections". In: *PLOS Computational Biology* 14.9 (2018). Ed. by Peter Hellyer, e1006433. DOI: 10.1371/journal.pcbi.1006433. URL: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006433.

[5] Paul C Bailey et al. "Dominant integration locus drives continuous diversification of plant immune receptors with exogenous domain fusions". In: *Genome biology* 19.1 (2018), pp. 1–18.

[6] A Cristina Barragan and Detlef Weigel. "Plant NLR diversity: the known unknowns of pan-NLRomes". In: *The Plant Cell* 33.4 (2021), pp. 814–831.

[7] Alex Bateman, Penny Coggill, and Robert D Finn. "DUFs: families in search of function". In: *Acta Crystallographica Section F: Structural Biology and Crystallization Communications* 66.10 (2010), pp. 1148–1152.

[8] Ulrich Bauer. "Ripser: a lean C++ code for the computation of Vietoris-Rips persistence barcodes". In: *Software available at https://github. com/Ripser/ripser* (2017).

[9] Juan P Bello. "Measuring structural similarity in music". In: *IEEE Transactions on Audio, Speech, and Language Processing* 19.7 (2011), pp. 2013–2025.

[10] Tanya Z Berardini et al. "The Arabidopsis information resource: making and mining the "gold standard" annotated reference plant genome". In: *genesis* 53.8 (2015), pp. 474–485.

[11] Helen M Berman et al. "The protein data bank". In: *Nucleic acids research* 28.1 (2000), pp. 235–242.

[12] Matthias Blum et al. "The InterPro protein families and domains database: 20 years on". In: *Nucleic acids research* 49.D1 (2021), pp. D344–D354.

[13] Elodie F Briefer et al. "Segregation of information about emotional arousal and valence in horse whinnies". In: *Scientific reports* 4 (2015), p. 9989.

[14] Yoram Burak and Ila R Fiete. "Accurate Path Integration in Continuous Attractor Network Models of Grid Cells". In: *PLOS Computational Biology* 5.2 (2009), e1000291. DOI: 10.1371/journal.pcbi.1000291.

[15] Zixuan Cang and Guo-Wei Wei. "TopologyNet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions". In: *PLoS computational biology* 13.7 (2017), e1005690.

[16] Gunnar Carlsson. "Topology and data". In: *Bulletin of the American Mathematical Society* 46.2 (2009), pp. 255–308.

[17] Rishidev Chaudhuri et al. "The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep". In: *Nature Neuroscience* 22.9 (2019), pp. 1512–1520. DOI: 10.1038/s41593-019-0460-x. URL: https://www.nature.com/articles/s41593-019-0460-x.

[18] Frédéric Chazal, David Cohen-Steiner, and Quentin Mérigot. "Geometric inference for measures based on distance functions". In: *Foundations of computational mathematics* 11.6 (2011), pp. 733–751. ISSN: 1615-3375. URL: https://hal.archives-ouvertes.fr/inria-00383685/.

[19] Samir Chowdhury, Bowen Dai, and Facundo Mémoli. "The importance of forgetting: Limiting memory improves recovery of topological characteristics from neural data". In: *PLOS ONE* 13.9 (Sept. 2018), pp. 1–20. DOI: 10.1371/journal.pone.0202561. URL: https://doi.org/10.1371/journal.pone.0202561.

[20] Fan RK Chung. *Spectral graph theory*. Vol. 92. American Mathematical Soc., 1997.

[21] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. "Stability of persistence diagrams". In: *Discrete & computational geometry* 37 (2007), pp. 103–120. ISSN: 0179-5376.

[22] William Crawley-Boevey. "Decomposition of pointwise finite-dimensional persistence modules". In: *Journal of Algebra and its Applications* 14.05 (2015), p. 1550066.

[23] Carina Curto. "What can topology tell us about the neural code?" In: *Bulletin of the American Mathematical Society* 54.1 (2016), pp. 63–78. ISSN: 0273-0979. DOI: 10.1090/bull/1554.

[24] Y Dabaghian et al. "A Topological Paradigm for Hippocampal Spatial Map Formation Using Persistent Homology". In: *PLOS Computational Biology* 8.8 (2012). Ed. by Ila Fiete, e1002581. DOI: 10.1371/journal.pcbi.1002581. URL: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1002581.

[25] Suddhasattwa Das and Dimitrios Giannakis. "Delay-coordinate maps and the spectra of Koopman operators". In: *arXiv preprint arXiv:1706.08544* (2017).

[26] Vin De Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. "Persistent cohomology and circular coordinates". In: *Discrete & Computational Geometry* 45.4 (2011), pp. 737–759.

[27] Vin De Silva and Mikael Vejdemo-Johansson. "Persistent cohomology and circular coordinates". In: *Proceedings of the twenty-fifth annual symposium on Computational geometry*. 2009, pp. 227–236.

[28] Geoffrey W Diehl et al. "Grid and Nongrid Cells in Medial Entorhinal Cortex Represent Spatial Location and Environmental Features with Complementary Coding Schemes." In: *Neuron* 94.1 (2017), 83–92.e6. DOI: 10.1016/j.neuron.2017.03.004.

[29] Richard O. Duda and Peter E. Hart. "Use of the Hough transformation to detect lines and curves in pictures". In: *Commun. ACM* 15.1 (Jan. 1972), pp. 11–15. ISSN: 0001-0782. DOI: 10.1145/361237.361242. URL: https://doi.org/10.1145/361237.361242.

[30] Herbert Edelsbrunner and John Harer. *Computational topology: an introduction*. American Mathematical Soc., 2010.

[31] Herbert Edelsbrunner and John Harer. "Persistent homology-a survey". In: *Contemporary mathematics* 453 (2008), pp. 257–282.

[32] Herbert Edelsbrunner and Dmitriy Morozov. "Persistent Homology". In: *Handbook of Discrete and Computational Geometry*. Ed. by Jacob E Goodman, Joseph O'Rourke, and Csaba D Tóth. CRC Press, 2017. ISBN: 9781498711395. URL: https://www.csun.edu/~ctoth/Handbook/chap24.pdf.

[33] Herbert Edelsbrunner and Ernst P Mücke. "Three-dimensional alpha shapes". In: *ACM Transactions on Graphics (TOG)* 13.1 (1994), pp. 43–72.

[34] Saba Emrani, Harish Chintakunta, and Hamid Krim. "Real time detection of harmonic structure: A case for topological signal analysis". In: *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE. 2014, pp. 3445–3449.

[35] I R Fiete, Y Burak, and T Brookings. "What Grid Cells Convey about Rat Location". In: *Journal of Neuroscience* 28.27 (2008), pp. 6858–6871. DOI: 10.1523/jneurosci.5684-07.2008. URL: http://www.jneurosci.org/content/28/27/6858.short.

[36] Robert D Finn, Jody Clements, and Sean R Eddy. "HMMER web server: interactive sequence similarity searching". In: *Nucleic acids research* 39.suppl_2 (2011), W29–W37.

[37] Jordan Frank, Shie Mannor, and Doina Precup. "Activity and Gait Recognition with Time-Delay Embeddings." In: *AAAI*. Citeseer. 2010.

[38] Hitesh Gakhar and Jose A Perea. "Sliding Window Persistence of Quasiperiodic Functions". 2018.

[39] Juan A. Gallego et al. "Neural Manifolds for the Control of Movement". In: *Neuron* 94.5 (2017), pp. 978–984. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2017.05.025.

[40] Peiran Gao and Surya Ganguli. "On simplicity and complexity in the brave new world of large-scale neuroscience". In: *Current Opinion in Neurobiology* 32 (2015), pp. 148–155. ISSN: 0959-4388. DOI: 10.1016/j.conb.2015.04.003.

[41] Richard J. Gardner et al. "Toroidal topology of population activity in grid cells". In: *bioRxiv* (2021). DOI: 10.1101/2021.02.25.432776. URL: https://www.biorxiv.org/content/early/2021/02/25/2021.02.25.432776.

[42] Robert W Ghrist. *Elementary applied topology*. Createspace, 2014.

[43] Chad Giusti et al. "Clique topology reveals intrinsic geometric structure in neural correlations". In: *Proceedings of the National Academy of Sciences of the United States of America* 112.44 (Nov. 2015), pp. 13455–13460. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1506407112. URL: http://www.pnas.org/content/112/44/13455.

[44] Bryan Glaz et al. "Quasi-periodic intermittency in oscillating cylinder flow". In: *Journal of Fluid Mechanics* 828 (2017), pp. 680–707.

[45] Chris Godsil and Gordon F Royle. *Algebraic graph theory*. Vol. 207. Springer Science & Business Media, 2001.

[46] Leonidas Guibas, Dmitriy Morozov, and Quentin Mérigot. "Witnessed k-Distance". In: *Discrete & computational geometry* 49.1 (Jan. 2013), pp. 22–45. ISSN: 0179-5376, 1432-0444. DOI: 10.1007/s00454-012-9465-x. URL: https://doi.org/10.1007/s00454-012-9465-x.

[47] Torkel Hafting et al. "Microstructure of a spatial map in the entorhinal cortex". In: *Nature* 436.7052 (2005), pp. 801–806. DOI: 10.1038/nature03721. URL: http://www.nature.com/nature/journal/v436/n7052/abs/nature03721.html.

[48] Kiah Hardcastle et al. "A Multiplexed, Heterogeneous, and Adaptive Code for Navigation in Medial Entorhinal Cortex." In: *Neuron* 94.2 (2017), 375–387.e7. DOI: 10.1016/j.neuron.2017.03.025.

[49] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[50] Hanspeter Herzel et al. "Analysis of vocal disorders with methods from nonlinear dynamics". In: *Journal of Speech, Language, and Hearing Research* 37.5 (1994), pp. 1008–1019.

[51] James R Hinman et al. "Multiple Running Speed Signals in Medial Entorhinal Cortex". In: *Neuron* 91.3 (2016), pp. 666–679. DOI: 10.1016/j.neuron.2016.06.027. URL: http://www.sciencedirect.com/science/article/pii/S0896627316303051.

[52] Aapo Hyvärinen. "Fast and robust fixed-point algorithms for independent component analysis". In: *IEEE Transactions on Neural Networks* 10.3 (1999), pp. 626–634. ISSN: 1045-9227. DOI: 10.1109/72.761722.

[53] HyeIn Jang et al. "The leucine-rich repeat signaling scaffolds Shoc2 and Erbin: cellular mechanism and role in disease". In: *The FEBS journal* 288.3 (2021), pp. 721–739.

[54] Mehrdad Jazayeri and Arash Afraz. "Navigating the Neural Space in Search of the Neural Code". In: *Neuron* 93.5 (2017), pp. 1003–1014. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2017.02.019.

[55] William B Johnson and Joram Lindenstrauss. "Extensions of Lipschitz mappings into a Hilbert space". In: *Contemporary Mathematics* 26.189-206 (1984), p. 1. ISSN: 0271-4132. URL: https://www.researchgate.net/profile/William_Johnson16/publication/235008656_Extensions_of_Lipschitz_maps_into_a_Hilbert_space/links/55e9abf908aeb65162649527.pdf.

[56] Jonathan DG Jones, Russell E Vance, and Jeffery L Dangl. "Intracellular innate immune surveillance devices in plants and animals". In: *Science* 354.6316 (2016), aaf6395.

[57] John Jumper et al. "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596.7873 (2021), pp. 583–589.

[58] James J Jun et al. "Fully integrated silicon probes for high-density recording of neural activity". In: *Nature* 551.7679 (2017), pp. 232–236. DOI: 10.1038/nature24636. URL: https://www.nature.com/articles/nature24636.

[59] Wolfgang Kabsch. "A solution for the best rotation to relate two sets of vectors". In: *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography* 32.5 (1976), pp. 922–923.

[60] Louis Kang and Vijay Balasubramanian. "A geometric attractor mechanism for self-organization of entorhinal grid modules". In: *eLife* 8 (2019), e46687. DOI: 10.7554/elife.46687. URL: https://elifesciences.org/articles/46687.

[61] Louis Kang, Boyan Xu, and Dmitriy Morozov. "Evaluating state space discovery by persistent cohomology in the spatial representation system". In: *Frontiers in computational neuroscience* 15 (2021), p. 616748.

[62] Holger Kantz and Thomas Schreiber. *Nonlinear time series analysis*. Vol. 7. Cambridge university press, 2004.

[63] Arshi Khalid et al. "Tracing the evolution of multi-scale functional networks in a mouse model of depression using persistent brain network homology". In: *NeuroImage* 101 (2014), pp. 351–363. DOI: 10.1016/j.neuroimage.2014.07.040. URL: https://www.sciencedirect.com/science/article/pii/S1053811914006235.

[64] Firas A Khasawneh and Elizabeth Munch. "Chatter detection in turning using persistent homology". In: *Mechanical Systems and Signal Processing* 70 (2016), pp. 527–541.

[65] B. O. Koopman. "Hamiltonian Systems and Transformation in Hilbert Space". In: *Proceedings of the National Academy of Sciences* 17.5 (1931), pp. 315–318. ISSN: 0027-8424. DOI: 10.1073/pnas.17.5.315. eprint: http://www.pnas.org/content/17/5/315.full.pdf. URL: http://www.pnas.org/content/17/5/315.

[66] Emilio Kropff et al. "Speed cells in the medial entorhinal cortex". In: *Nature* 523.7561 (2015), pp. 419–424. DOI: 10.1038/nature14622.

[67] Julija Krupic et al. "Grid cell symmetry is shaped by environmental geometry". In: *Nature* 518.7538 (2015), pp. 232–235. DOI: 10.1038/nature14153. URL: http://www.nature.com/nature/journal/v518/n7538/abs/nature14153.html.

[68] John E Lisman and Ole Jensen. "The Theta-Gamma Neural Code". In: *Neuron* 77.6 (2013), pp. 1002–1016. DOI: 10.1016/j.neuron.2013.03.007. URL: http://www.sciencedirect.com/science/article/pii/S0896627313002316%5C#bib117.

[69] Eliza C Martin et al. "LRRpredictor—a new LRR motif detection method for irregular motifs of plant NLR proteins using an ensemble of classifiers". In: *Genes* 11.3 (2020), p. 286.

[70] Alexander Mathis, Andreas V M Herz, and Martin Stemmler. "Optimal Population Codes for Space: Grid Cells Outperform Place Cells". In: *Neural Computation* 24.9 (2012), pp. 2280–2317. ISSN: 0899-7667. DOI: 10.1162/neco_a_00319.

[71] Leland McInnes et al. "UMAP: Uniform Manifold Approximation and Projection". In: *Journal of Open Source Software* 3.29 (2018), p. 861. DOI: 10.21105/joss.00861.

[72] Igor Mezić. "Analysis of Fluid Flows via Spectral Properties of the Koopman Operator". In: ().

[73] Jaina Mistry et al. "Challenges in homology search: HMMER3 and convergent evolution of coiled-coil regions". In: *Nucleic acids research* 41.12 (2013), e121–e121.

[74] Farzin Mokhtarian and Alan K Mackworth. "A theory of multiscale, curvature-based shape representation for planar curves". In: *IEEE transactions on pattern analysis and machine intelligence* 14.8 (1992), pp. 789–805.

[75] Katherine Morrison et al. "Diversity of emergent dynamics in competitive threshold-linear networks: a preliminary report". In: *arXiv preprint arXiv:1605.04463* (2016).

[76] Noga Mosheiff et al. "An efficient coding theory for a dynamic trajectory predicts non-uniform allocation of entorhinal grid cells to modules". In: *PLOS Computational Biology* 13.6 (2017). Ed. by Gaute T Einevoll, e1005597. DOI: 10.1371/journal.pcbi.1005597. URL: http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005597.

[77] Aylwin Ng and Ramnik J Xavier. "Leucine-rich repeat (LRR) proteins: integrators of pattern recognition and signaling in immunity". In: *Autophagy* 7.9 (2011), pp. 1082–1084.

[78] David D Nolte. "The tangled tale of phase space". In: *Physics today* 63.4 (2010), pp. 33–38.

[79] J O'Keefe and J Dostrovsky. "The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat". In: *Brain Research* 34.1 (1971), pp. 171–175. DOI: 10.1016/0006-8993(71)90358-1. URL: http://www.sciencedirect.com/science/article/pii/0006899371903581.

[80] Meenu Padmanabhan, Patrick Cournoyer, and SP Dinesh-Kumar. "The leucine-rich repeat domain in plant innate immunity: a wealth of possibilities". In: *Cellular microbiology* 11.2 (2009), pp. 191–198.

[81] Keunwan Park et al. "Control of repeat-protein curvature by computational protein design". In: *Nature structural & molecular biology* 22.2 (2015), pp. 167–174.

[82] Jose A Perea. "A Brief History of Persistence". In: *arXiv preprint arXiv:1809.03624* (2018).

[83] Jose A Perea. "Multiscale Projective Coordinates via Persistent Cohomology of Sparse Filtrations". In: *Discrete & Computational Geometry* 59.1 (2018), pp. 175–225.

[84] Jose A Perea. "Persistent homology of toroidal sliding window embeddings". In: *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE. 2016, pp. 6435–6439.

[85] Jose A Perea. "Sparse circular coordinates via principal $\mathbb{Z}$-bundles". In: *Topological Data Analysis: The Abel Symposium 2018*. Springer. 2020, pp. 435–458.

[86] Jose A Perea. "Sparse Circular Coordinates via Principal $\mathbb{Z}$-Bundles". In: *arXiv preprint arXiv:1809.09269* (2018).

[87] Jose A Perea. "Topological Time Series Analysis". In: *Notices of the American Mathematical Society* 66.5 (2019).

[88] Jose A Perea and John Harer. "Sliding windows and persistence: An application of topological methods to signal analysis". In: *Foundations of Computational Mathematics* 15.3 (2015), pp. 799–838.

[89] Jose A Perea et al. "Sw1pers: Sliding windows and 1-persistence scoring; discovering periodicity in gene expression time series data". In: *BMC bioinformatics* 16.1 (2015), p. 257.

[90] Marco Piangerelli et al. "Topological classifier for detecting the emergence of epileptic seizures". In: *BMC research notes* 11.1 (June 2018), p. 392. ISSN: 1756-0500. DOI: 10.1186/s13104-018-3482-7. URL: http://dx.doi.org/10.1186/s13104-018-3482-7.

[91] Emil Plesnik et al. "Detection and Delineation of the Electrocardiogram Qrs-complexes from Phase Portraits". In: (2014).

[92] Daniil M Prigozhin and Ksenia V Krasileva. "Analysis of intraspecies diversity reveals a subset of highly variable plant immune receptors and predicts their binding sites". In: *The Plant Cell* 33.4 (2021), pp. 998–1015.

[93] Erik Rybakken, Nils Baas, and Benjamin Dunn. "Decoding of Neural Data Using Cohomological Feature Extraction". In: *Neural Computation* 31.1 (2019), pp. 68–93. ISSN: 0899-7667. DOI: 10.1162/neco_a_01150.

[94] A. Sanzeni et al. "Complete coverage of space favors modularity of the grid system in the brain". In: *Physical Review E* 94.6 (2016), p. 062409. ISSN: 2470-0045. DOI: 10.1103/physreve.94.062409. eprint: 1610.04844. URL: https://journals.aps.org/pre/abstract/10.1103/PhysRevE.94.062409.

[95] Francesca Sargolini et al. "Conjunctive representation of position, direction, and velocity in entorhinal cortex." In: *Science* 312.5774 (2006), pp. 758–762. DOI: 10.1126/science.1125572.

[96] Simon B Saucet, Daniel Esmenjaud, and Cyril Van Ghelder. "Integrity of the post-LRR domain is required for TIR-NB-LRR function". In: *Molecular Plant-Microbe Interactions* 34.3 (2021), pp. 286–296.

[97] Shreya Saxena and John P Cunningham. "Towards the neural population doctrine". In: *Current Opinion in Neurobiology* 55 (2019), pp. 103–111. ISSN: 0959-4388. DOI: 10.1016/j.conb.2019.02.002.

[98] Joan Serra, Xavier Serra, and Ralph G Andrzejak. "Cross recurrence quantification for cover song identification". In: *New Journal of Physics* 11.9 (2009), p. 093017.

[99] Michael N. Shadlen and William T. Newsome. "The Variable Discharge of Cortical Neurons: Implications for Connectivity, Computation, and Information Coding". In: *Journal of Neuroscience* 18.10 (1998), pp. 3870–3896. ISSN: 0270-6474. DOI: 10.1523/jneurosci.18-10-03870.1998.

[100] Vin de Silva, Primoz Skraba, and Mikael Vejdemo-Johansson. "Topological analysis of recurrent systems". In: *Workshop on Algebraic Topology and Machine Learning, NIPS*. 2012.

[101] Gard Spreemann et al. "Using persistent homology to reveal hidden information in neural data". In: *arXiv* (2015). eprint: 1510.06629. URL: https://arxiv.org/abs/1510.06629.

[102] Sameet Sreenivasan and Ila Fiete. "Grid cells generate an analog error-correcting code for singularly precise neural computation". In: *Nature Neuroscience* 14.10 (2011), pp. 1330–1337. DOI: 10.1038/nn.2901.

[103] Cornelis J Stam. "Nonlinear dynamical analysis of EEG and MEG: review of an emerging field". In: *Clinical Neurophysiology* 116.10 (2005), pp. 2266–2301.

[104] Martin Stemmler, Alexander Mathis, and Andreas V M Herz. "Connecting multiple spatial scales to decode the population activity of grid cells". In: *Science Advances* 1.11 (2015), e1500816–e1500816. DOI: 10.1126/science.1500816. URL: http://advances.sciencemag.org/content/1/11/e1500816.

[105] Hanne Stensola et al. "The entorhinal grid map is discretized". In: *Nature* 492.7427 (2012), pp. 72–78. DOI: 10.1038/nature11649. URL: http://www.nature.com/nature/journal/v492/n7427/full/nature11649.html.

[106] Tor Stensola et al. "Shearing-induced asymmetry in entorhinal grid cells". In: *Nature* 518.7538 (Feb. 2015), pp. 207–212. DOI: 10.1038/nature14151. URL: http://www.nature.com/nature/journal/v518/n7538/full/nature14151.html.

[107] Floris Takens et al. "Detecting strange attractors in turbulence". In: *Lecture notes in mathematics* 898.1 (1981), pp. 366–381.

[108] Janina Tamborski and Ksenia V Krasileva. "Evolution of plant NLRs: from natural history to precise modifications". In: *Annual review of plant biology* 71 (2020), pp. 355–378.

[109] Wai Shing Tang et al. "A topological data analytic approach for discovering biophysical signatures in protein dynamics". In: *PLoS computational biology* 18.5 (2022), e1010045.

[110] J S Taube, R U Muller, and J B Ranck. "Head-direction cells recorded from the postsubiculum in freely moving rats. I. Description and quantitative analysis." In: *Journal of Neuroscience* 10.2 (1990), pp. 420–435.

[111] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. "A Global Geometric Framework for Nonlinear Dimensionality Reduction". In: *Science* 290.5500 (2000), pp. 2319–2323. ISSN: 0036-8075. DOI: 10.1126/science.290.5500.2319.

[112] Christopher Tralie, Nathaniel Saul, and Rann Barr-On. "Ripser.py: A Lean Persistent Homology Library for Python". In: *Journal of Open Source Software (JOSS)* (2018).

[113] Christopher J Tralie. "Geometric Multimedia Time Series". Duke Ph.D. Dissertation. Department of Electrical and Computer Engineering, Duke University, 2017.

[114] Christopher J Tralie and Matthew Berger. "Topological eulerian synthesis of slow motion periodic videos". In: *2018 25th IEEE international conference on image processing (ICIP)*. IEEE. 2018, pp. 3573–3577.

[115] Christopher J Tralie and John Harer. "Moebius Beats: The Twisted Spaces of Sliding Window Audio Novelty Functions with Rhythmic Subdivisions". In: *18th International Society for Music Information Retrieval (ISMIR), Late Breaking Session*. 2017.

[116] Christopher J. Tralie and Jose A. Perea. "(Quasi)Periodicity Quantification in Video Data, Using Topology". In: *SIAM Journal on Imaging Sciences* 11.2 (2018), pp. 1049–1077. DOI: 10.1137/17M1150736. eprint: https://doi.org/10.1137/17M1150736. URL: https://doi.org/10.1137/17M1150736.

[117] Mihaly Varadi et al. "AlphaFold Protein Structure Database in 2024: providing structure coverage for over 214 million protein sequences". In: *Nucleic Acids Research* 52.D1 (2024), pp. D368–D375.

[118] Mikael Vejdemo-Johansson et al. "Cohomological learning of periodic motion". In: *Applicable algebra in engineering, communication and computing* 26.1 (2015), pp. 5–26.

[119] Vinay Venkataraman, Karthikeyan Natesan Ramamurthy, and Pavan Turaga. "Persistent homology of attractors for action recognition". In: *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE. 2016, pp. 4150–4154.

[120] Grace Wahba. "A least squares estimate of satellite attitude". In: *SIAM review* 7.3 (1965), pp. 409–409.

[121] Yuan Wang, Hernando Ombao, and Moo K Chung. "Topological Data Analysis of Single-Trial Electroencephalographic Signals". In: *The annals of applied statistics* 12.3 (Sept. 2018), pp. 1506–1534. ISSN: 1932-6157. DOI: 10.1214/17-AOAS1119. URL: http://dx.doi.org/10.1214/17-AOAS1119.

[122] X-X Wei, J Prentice, and V Balasubramanian. "A principle of economy predicts the functional architecture of grid cells". In: *eLife* 4 (2015), e08362. DOI: 10.7554/elife.08362.001.

[123] Boyan Xu et al. "Structure-Aware Annotation of Leucine-rich Repeat Domains". In: *bioRxiv* (2023).

[124] Boyan Xu et al. "Twisty takens: A geometric characterization of good observations on dense trajectories". In: *Journal of Applied and Computational Topology* 3 (2019), pp. 285–313.

[125] Mengsen Zhang et al. "Topological portraits of multiscale coordination dynamics". In: *Journal of Neuroscience Methods* 339 (2020), p. 108672. DOI: 10.1016/j.jneumeth.2020.108672. URL: https://www.sciencedirect.com/science/article/pii/S0165027020300947?via%5C%3Dihub.

[126] Anton Zorich. "Flat surfaces". In: *Frontiers in number theory, physics, and geometry I* (2006), pp. 439–585.