Statistics for Persistent Landscapes

Yuxin Ma

December 23, 2023

1 Introduction

This essay is devoted to studying the following problem: How to extract meaningful topological information from 2D/3D image data with an algorithm that is *computationally efficient* and *robust to noise*? The topological information to be extracted includes the number of connected components, holes, voids, and their relative positions. Moreover, the study aims to automatically process large datasets of images and perform statistical analysis and machine learning tasks on topological information. For instance, given the 2D slices of prostate cancer histopathology images depicted in Figure 1, we aim to develop an algorithm that can accurately identify not only the large ring-shaped structures corresponding to glands but also the small dark-coloured components and rings that correspond to the many peaks of nuclei inside the glands. We wish to detect topological patterns in big datasets of such medical images to recognize clinically meaningful features associated with the progression of prostate cancer.

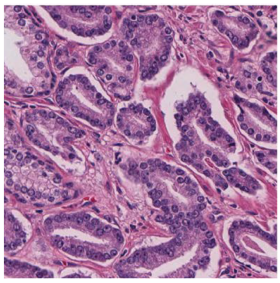


Figure 1: A sample prostate cancer histopathology image

In this essay, we introduce a powerful technique called **persistent homology (PH)** that addresses this challenge. Since its introduction, PH has become the workhorse for the emerging field of **Topological Data Analysis (TDA)**.

The first half of this essay (Sections 2 to 4) focuses on the theoretical foundations of PH. PH is introduced as a generalization of standard homology, and its key properties are derived from homology theory. To this end, we begin by reviewing standard homology theory from a computational perspective in Section

2, emphasizing its informative and computable nature in detecting significant topological patterns in *binary images*. Subsequently, in Section 3, we introduce the key idea of *multi-scale processing* behind the generalisation of homology to PH. To illustrate the idea, we note that in Figure 1, topological information is present at various scales, ranging from miniature cellular structures to large glandular rings. PH aims to simultaneously explore all possible scales, examine how topological features emerge and merge along the scales, and trace their persistence. Finally, in Section 4, we demonstrate that PH can be formulated as a standard homology, enabling us to generalize the decomposability and computability of homology. The decomposition of PH signifies that the abstract concept of PH can be translated into a simple *barcode representation* that encodes pairs of birth and death times across scales. These theoretical properties form the basis for the practical utility of PH in algorithmically extracting a plethora of topological insights presented in a compact, interpretable barcode.

Starting from Section 5, we offer practical demonstrations of the use of PH in image analysis. In Section 5, we detail the standard computational algorithm while also examining efforts aimed at enhancing its computational efficiency, emphasizing the benefits of utilizing the *cubical complex* for computational concerns. Subsequently, in Section 6, we present a practical framework that maps the barcode representation, which is unsuitable for statistical analysis, to a vector space, thereby enabling the utilization of established statistical and machine learning tools. To demonstrate the practical applicability of the techniques outlined in the essay, we present two case studies, one of which centres on the analysis of prostate histology images depicted in Figure 1. We pay particular attention to two popular functional vectorizations, namely the *persistence landscape* and the *persistence intensity*, analyzing their theoretical properties and showcasing their practical usage through the case studies.

Purpose While PH has diverse applications in point clouds, networks, and functions, this essay focuses specifically on its application in image data due to several reasons. Firstly, image data requires unique computational considerations, and statistical analysis strategies may differ. Secondly, existing literature on PH theory usually discusses the theory based on point clouds and simplicial complexes. In contrast, this essay provides a self-contained introduction to PH theory using image data examples, making it accessible to readers interested solely in PH's applications in images.

Furthermore, this essay delves deeper into the theoretical foundations of PH compared to most introductory texts. It places significant emphasis on illuminating the underlying motivations of PH, the deep-rooted link to standard homology theory, and the reasons behind the miracle of PH. By exploring the theoretical foundations, readers will gain a deeper understanding of the theory's capabilities and limitations, as well as its potential for future development.

Prerequisite Knowledge of homology is not required, as all necessary backgrounds are reviewed in Section 2. However, reasonable background in abstract

algebra, topology, statistics and machine learning is assumed. Basic notions of category theory are used to clarify the key ideas and illustrate the relations between concepts in Section 2 to 4, but familiarity with definitions of category and functor would be enough.

Acknowledgement Section 2 is based on the treatment of homology theory in [25], the design of cubical complexes in [29], and the computational perspective in [57]. Section 3 adopts a categorical viewpoint of PH from [9] and [48]. Section 4 mainly draws from [57] and [9], with missing details spelled out in the proof of Theorem 4.1 and the treatment of graded module. The development of the PH theory also consults [20, 47, 46]. The practical parts of the essay (Section 5 and 6) review materials from recent work, which are cited in the text. Persistence computations and diagrams are generated using the Python interface of GUDHI [49].

Notation and convention $\mathbb{N} = \{0, 1, 2, \dots\}$

2 Computational homology

A good starting point for our venture into TDA involves exploring two questions:

1. How to "topologize" the data, i.e. impose a compatible topological structure on the image data? (Section 2.1)
2. Which is a good topological invariant as the focus of our analysis of data topology? (Section 2.2 and 2.3)

2.1 Cubical complex

2D image data consisting of $m \times n$ pixels can be modelled as a scalar function $f : \Omega \to \mathbb{R}$ where domain $\Omega = [0, m] \times [0, n]$ and f is constant on the squares $\{[i, i+1] \times [j, j+1] : i, j \in \mathbb{N}\}$. 3D image data can also be modelled analogously. It is natural to work with the inherent rigid structure of the data, which leads us to define cubical complex as the topological space to work with [29].

Definition 2.1 (Cubical complex)**.** *An* ***elementary interval*** $I \subseteq \mathbb{R}$ *is of the form* $I = [m, m]$ *or* $[m, m+1]$ *where* $m \in \mathbb{Z}$*. If* $I = [m, m]$*, we say it is* ***degenerate****.*

In $\mathbb{R}^n$*, a* d***-cube*** σ *is* $\sigma = \prod_{k=1}^{n} I_k \subseteq \mathbb{R}^n$*, where* d *of the intervals are non-degenerate. We say* σ *has* ***dimension*** d*. In particular, a* 1*-cube is a unit interval, a* 2*-cube is a unit square, and a* 3*-cube is a unit cube. Cube* σ_1 *is a* ***face*** *of* σ_2 *if* $\sigma_1 \subseteq \sigma_2$*.*

A ***cubical complex*** $\mathcal{K}$ *is a finite union of elementary cubes that is closed under taking faces and intersections.*

We can form a cubical complex from a 2D *binary image* $B : [0, m] \times [0, n] \to \{0, 1\}$ in a natural way: take the pixels with value 0 as 2-cubes, and take all edges and vertices of those pixels as 1-cubes and 0-cubes.

Example 2.1. *Figure 2 shows an example of a cubical complex in* $\mathbb{R}^2$ *constructed from a* 2×2 *2D binary image.*

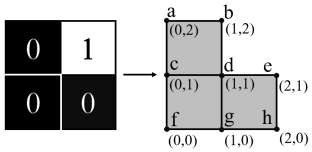


Figure 2: A 2D binary image topologized as a cubical complex consisting of eight 0-cubes, ten 1-cubes and three 2-cubes

Readers who are familiar with the notion of the simplicial complex should find them analogous. The main advantages of using cubical complex over simplicial complex for image data include:

1. Artificial triangulation is necessary to construct simplicial complexes from image data, which can significantly increase the size of the data. Specifically, constructing a triangulation of one n-dimensional cube requires $n!$ simplices [29].

2. Some works extract point clouds from images and build simplicial complexes on them. This approach requires substantial preprocessing tailored to each dataset. In contrast, the cubical complex is immediately usable without additional customization.

3. The rigid structure of cubical complexes allows for a more compact data structure and more efficient query of face relations and orientations. Please refer to Section 5.2 for more details on this.

It turns out that both cubical complexes and simplicial complexes are special cases of a general class of topological spaces that plays a central role in algebraic topology: the CW complex. The CW complex provides a flexible framework for describing spaces constructed inductively with simple building blocks. Throughout the essay, our theory of homology and persistent homology will be defined upon CW complexes. This provides generality that ensures the theory developed is readily applicable to other data types, such as point clouds and networks.

Definition 2.2. *A* N***-dimensional CW complex*** $\mathcal{K}$ *is a topological space constructed inductively as below,*

1. *Start with a discrete set* X_0 *with the discrete topology.*

2. *For each $n = 1, \ldots, N$, the n-**skeleton** X_n is formed by attaching one or more n-**cells** $(\sigma_j^n)_j$, each homeomorphic to an n-disk, to X_{n-1} via the glueing map $\varphi_j^n : \partial\sigma_j^n \to X_{n-1}$. Finally, take $\mathcal{K} = \bigcup_{n=0}^{N} X_n$.*

We shall see in the next section that the purpose of such inductive construction of topological space is to provide a combinatorial structure of cells that enables algebraic computations to be performed.

2.2 Homology

Among many topological invariants, we will introduce homology and illustrate why it is a good focus for our study in TDA. In short, homology is both *informative* and *computable*. Firstly, it provides information on the number of connected components, the number of holes, Euler characteristics, and more, capturing essential features of a space. Secondly, its discrete nature (when considering CW complexes) and algebraic structure make it algorithmically computable with matrix operations. We provide a minimal background on homology for CW complexes in the rest of Section 2.2 and explore its computability in Section 2.3.

We first give an informal definition of "chain complex" and follow it with an intuitive explanation. For a more formal treatment, please read [25].

Definition 2.3 (Cellular chain complex (informal definition))**.** *Given a CW complex $\mathcal{K}$, the associated **cellular chain complex** $(C_\bullet(\mathcal{K}), \partial_\bullet)$ over a ring R is a sequence of R-modules with homomorphisms*

$$\ldots \xrightarrow{\partial_{n+1}} C_n(\mathcal{K}) \xrightarrow{\partial_n} C_{n-1}(\mathcal{K}) \xrightarrow{\partial_{n-1}} \ldots \xrightarrow{\partial_1} C_0(\mathcal{K}).$$

- *The **n-chain** $C_n(\mathcal{K})$ is the free R-module generated by the n-cells $\{\sigma_j^n\}_j$ of $\mathcal{K}$.*
- *The **boundary operator** $\partial_n : C_n(\mathcal{K}) \to C_{n-1}(\mathcal{K})$ is a homomorphism mapping each n-dimensional element of $C_n(\mathcal{K})$ to its $(n-1)$-dimensional boundary.*

The key idea is to treat the topological building blocks, the n-cells, as algebraic objects. This enables us to perform computations on them in a geometrically meaningful way. To achieve this, we define a sequence of R-modules. The boundary operators, which connect the n-chains, also have an algebraic meaning. We summarize the collection of all n-chains and boundary operators as one algebraic object, known as the "chain complex".

We now focus on the concrete structure of the cubical complex. The boundary operator takes the following form [47]:

For a non-degenerate elementary interval $I = [p, p+1]$, we define its **front face** $I^+ = [p+1, p+1]$, and **back face** $I^- = [p, p]$. Consider a cube $\sigma = \prod_{k=1}^{n} I_k$. If I_m is non-degenerate, we define the m**-th front face** σ_m^+ and m**-th back face** σ_m^- of σ by replacing I_m in the direct product with I_m^+ or I_m^- respectively.

Boundary operator ∂_d is generated by its action on d-cube σ given by $\partial_d(\sigma) = \sum_{\sigma' \text{ is a (d-1)-cube}} \kappa(\sigma, \sigma')$, where

$$\kappa(\sigma, \sigma') = \begin{cases} -1 & \text{if } \sigma' = \sigma_m^+ \text{ for even } m \text{ or } \sigma' = \sigma_m^- \text{ for odd } m; \\ 1 & \text{if } \sigma' = \sigma_m^+ \text{ for odd } m \text{ or } \sigma' = \sigma_m^- \text{ for even } m; \\ 0 & \text{otherwise.} \end{cases}$$

Geometrically, the coefficient $\kappa(\sigma, \sigma')$ captures the orientation of σ' in the boundary. We compute the boundary operator on the previous example to illustrate the above ideas.

Example 2.1 (continued)**.** *In the previous example of cubical complex, we have*

$$\begin{aligned} \partial_2(cdgf) &= \partial_2([0,1] \times [0,1]) \\ &= [0,1] \times [0] - [0,1] \times [1] - [0] \times [0,1] + [1] \times [0,1] \\ &= fg + gd - cd - fc \end{aligned}$$

The boundary operator allows us to also define the geometric concept of "boundary" and "cycle" algebraically.

Definition 2.4. *(Boundary, cycle)* $B_n(\mathcal{K}) := \operatorname{img}(\partial_{n+1}), Z_n(\mathcal{K}) := \ker(\partial_n)$ *are* n***-boundaries*** *and* n***-cycles*** *respectively. They are both submodules of* $C_n(\mathcal{K})$*. A well-defined boundary map should guarantee that every boundary is a cycle, i.e.* $B_n \leq Z_n$ *as submodules.*

We finally define homology.

Definition 2.5 (Homology, Betti number)**.** *The* n***-th cellular homology module*** *of CW complex* $\mathcal{K}$ *over ring* R *is the quotient module*

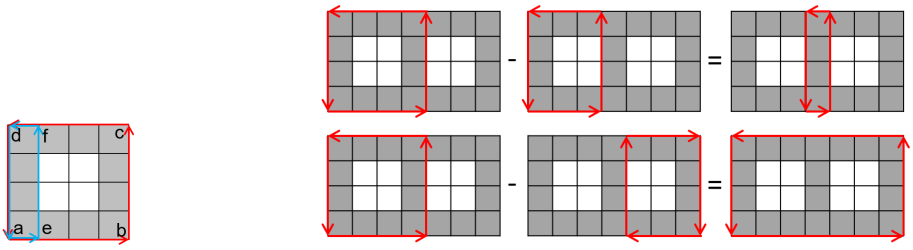
$$H_n(\mathcal{K}) := \ker(\partial_n) / \operatorname{img}(\partial_{n+1}) = Z_n(\mathcal{K}) / B_n(\mathcal{K}).$$

The n***-th Betti number*** $\beta_n(\mathcal{K})$ *is the group rank of* $H_n(\mathcal{K})$ *over integral coefficient* $R = \mathbb{Z}$*. The* n***-th Betti number over field*** $\mathbb{F}$ $\beta_n^{\mathbb{F}}(\mathcal{K})$ *is the vector space dimension of* $H_n(\mathcal{K})$ *over field coefficient* $R = \mathbb{F}$*.*

Intuitively, the n-th homology captures "n-dimensional holes" as "n-cycles quotienting out n-boundaries". As illustrated in Figure 3, cycles that are not boundaries are considered holes, and two cycles are considered equivalent topologically if they differ only by a boundary. The n-th Betti number counts the number of n-dimensional holes in the topological space (β_0 is the number of connected components by convention).

Remark. *By definition, 2D image has trivial homology* H_n *for* $n \geq 2$*, and 3D image has trivial homology* H_n *for* $n \geq 3$*.*

Last but not least, we state an important property of homology that is frequently used in this essay: homology is not only a topological invariant but



(a) (b)

Figure 3: In (a), cycle $ab + bc - dc - ad$ is a 2-dimensional hole, but cycle $ae + ef - df - ad$ is not because it is a boundary. In (b), the two cycles in the first row are considered equivalent because they differ by a boundary, whereas the two cycles in the second row are considered distinct.

also has a nice functoriality property. Interested readers should refer to [25] for proofs.

We use **CWCplx** to denote the category (CW complexes, cellular maps), use $\mathbf{Ch}_R$ to denote the category of (chain complexes over R, chain maps), and use $\mathbf{Mod}_R$ to denote the category (R-modules, homomorphisms). [1]

Theorem 2.1 (Functorality)**.** *(1) The assignment of n-th cellular chain over ring R to a CW complex is a functor $C_n : \mathbf{CWCplx} \to \mathbf{Mod}_R$.*

(2) The assignment of a cellular chain complex to a CW complex is a functor $C_\bullet : \mathbf{CWCplx} \to \mathbf{Ch}_R$.

(3) The assignment of n-th homology to a chain complex is a functor $H_n : \mathbf{Ch}_R \to \mathbf{Mod}_R$.

2.3 Computation of homology

We shall now demonstrate that if $\mathcal{K}$ is a finite-dimensional CW complex and the coefficient ring R is a principle ideal domain (PID), then the homology module $H_n(\mathcal{K})$ can be presented in a recognizable form which is algorithmically computable.

Taking the canonical basis for $C_n(\mathcal{K})$ (formed by the n-cells) for all n, we can represent the boundary operators ∂_n as matrices over R. We call them **boundary matrices** and denote them as M_n. Since R is PID, each matrix M_n can be transformed by elementary row and column operations into the Smith Normal Form (SNF) $\mathrm{diag}(d_1^n, \ldots, d_{l_n}^n, 0, \ldots, 0)$, where each $d_i^n \neq 0$ and $d_1^n | \ldots | d_{l_n}^n$.

Denote the new basis of $C_n(\mathcal{K})$ and $C_{n-1}(\mathcal{K})$ wrt to the SNF of M_n as $\{\sigma_1^n, \ldots, \sigma_{N_n}^n\}$ and $\{\sigma_1^{n-1}, \ldots, \sigma_{N_{n-1}}^{n-1}\}$ respectively. Then $\{d_1^n \sigma_1^{n-1}, \ldots, d_{l_n}^n \sigma_{l_n}^{n-1}\}$

[1] Cellular maps between two CW complexes are continuous maps taking n-skeletons to n-skeletons; Chain maps between two chain complexes $(A_\bullet, \partial_\bullet^A)$ and $(B_\bullet, \partial_\bullet^B)$ are a sequence of homomorphism $f_n : A_n \to B_n$ satisfying $\partial_n^B \circ f_n = f_{n-1} \circ \partial_n^A$.

is a basis for B_{n-1} and $\text{rank}(B_{n-1}) = l_n$. Also, $\{\sigma^n_{l_n+1}, \ldots, \sigma^n_{N_n}\}$ is a basis for Z_n and $\text{rank}(Z_n) = N_n - l_n$. As $B_{n-1} \leq Z_{n-1} \leq C_{n-1}$ as submodules, we can choose σ^{n-1}_i such that $\{\sigma^{n-1}_1, \ldots, \sigma^{n-1}_{N_{n-1}-l_{n-1}}\}$ form a basis of Z_{n-1}. Therefore we can identify

$$\begin{aligned} H_{n-1}(\mathcal{K}) &= \frac{Z_{n-1}(\mathcal{K})}{B_{n-1}(\mathcal{K})} \\ &\cong \frac{R^{N_{n-1}-l_{n-1}}}{d^n_1 R \oplus \cdots \oplus d^n_{l_n} R} \\ &\cong R^{N_{n-1}-l_{n-1}-l_n} \oplus \left(\bigoplus_{i=1}^{l_n} (R/d^n_i R) \right) \end{aligned} \tag{1}$$

where the first term is the **free part**, and the second term is the **torsional part**.

Recall that the Structure Theorem states that for any PID R, every finitely generated R-module M is uniquely decomposable as the form (1).

On the one hand, the procedure above serves as constructive proof for the existence part of the Structure Theorem applied to the homology module. $H_n(\mathcal{K})$ is guaranteed to admit a compact representation given by (1). In particular, $H_n(\mathcal{K})$ over a field coefficient only has the free part and is fully characterised by its vector space dimension.

On the other hand, the procedure also shows that computing the homology module $H_n(\mathcal{K})$ simply boils down to computing the SNF of the boundary matrices M_n, M_{n+1} over ring R. When R is a field, the standard algorithm based on Gaussian elimination is of time complexity $O(n^3)$ where n is the total number of cells.

Computing homology of images Before the invention of persistent homology, computational homology algorithms have already been widely used for the topological analysis of images, with applications in medical imaging [42], physics [37] and material science [52].

Arguably the biggest obstacle that hinders its applicability is the computational burden. For example, a cubical complex constructed from a 3d $k \times k \times k$ image may contain $O(k^3)$ elementary cubes if a large part of the image has value 1. Hence the standard algorithm has a time complexity of $O(k^9)$ even for the field coefficient, which is prohibitive for even a medium-sized image. There have been multiple efforts to improve computational efficiency, and we will see in Section 5.2.

3 From homology to persistence homology

Figure 4 shows a simple example of a grayscale image that we constructed for illustrative purposes. To apply the computational homology algorithms developed in the previous section, we need to choose a binarization threshold as a

first step. However, no threshold captures all the important pieces (two holes and one piece inside a hole) that are visually obvious from the image.

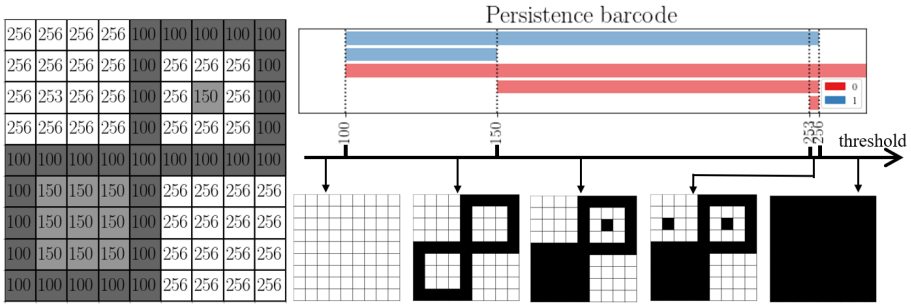


Figure 4: No threshold correctly captures the topology

From this example, we can draw the following insights:

- There is no correct or optimal threshold for binarization, and important information exists across the entire spectrum.
- As the threshold increases, the shape grows and homology classes (connected components, holes) are born and die merging into one another. Tracking their lifespan provides much more useful topological information than simple binarization. In this example, let's consider the 0-th homology first: ignoring the small component that appears at threshold 253 and disappears at 256, one component appears at threshold 100, grows, and persists throughout; another is born at 150 and dies merging with the original one at 256. For the 1-st homology, two 1D holes are born simultaneously at 100; one is filled at 150, and another is filled at 256. This information can be summarized as a *persistence barcode* in Figure 4.
- Such an analysis also serves as a denoising technique. In this example, we intentionally included a noisy pixel of value 253, leading to a transient component from 253 to 256. It is reasonable to assume that short persistence (barcode of short lengths) corresponds to noise.

The main idea is to consider a *multi-scale* version of homology, which evaluates all possible thresholds simultaneously. Since multi-scale patterns are abundantly found in nature, multi-scale approaches are ubiquitous in image analysis. The discrete wavelet transform can split signals into components at different scales and positions, which is useful for image denoising and compression [38]. Another example is the Convolutional Neural Network (CNN), which underlies many state-of-the-art models in image processing. It is designed to detect patterns at a local scale in the first few layers and track features of a larger scale at later layers, while imposing an assumption of *scale separation* [6].

In this section, we incorporate the intuitive idea above to formalise a theory of *persistence homology* upon standard homology, generalizing "CW complex"

and "homology" to the multi-scale concepts of "filtration" and "persistent module of homology" respectively, leading to an algebraic formalization of "birth", "death" and "persistence". We adopt the categorical viewpoint of [9] for its brevity and clarity.

3.1 Filtration and persistence module

Inspired by Figure 4, we now want to study a spectrum of growing CW complexes as a single topological object and assign an algebraic invariant that extends homology to a multi-scale setting.

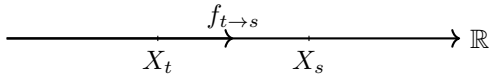
To adopt the multi-scale approach, we first define a "diagram" as a formalised notion of continuous and discrete multi-scale objects.

Definition 3.1 (Diagram)**.** *We use* $(\mathbb{R}, \leq)$ *to denote the category of real numbers with morphisms* $s \to t$ *whenever* $s \leq t$*.*

An $\mathbb{R}$***-indexed diagram*** *of category* $\mathcal{C}$ *is a functor* $F : (\mathbb{R}, \leq) \to \mathcal{C}$*. All such functors, with morphisms given by the natural transformations, form a category, denoted by* $\mathcal{C}^{(\mathbb{R},\leq)}$*.*

The $\mathbb{N}$***-indexed diagram*** *is defined similarly.*

Intuitively, an $\mathbb{R}$-indexed diagram $F \in \mathcal{C}^{(\mathbb{R},\leq)}$ can be seen as a collection $(X_\bullet, f_\bullet)$ with objects $(X_t)_{t\in\mathbb{R}}$ in $\mathcal{C}$, lying on the real axis, and morphisms $f_{t\to s} : X_t \to X_s$ for each $t \leq s$.



A $\mathbb{N}$-indexed diagram $F \in \mathcal{C}^{(\mathbb{N},\leq)}$ can be represented as a sequence

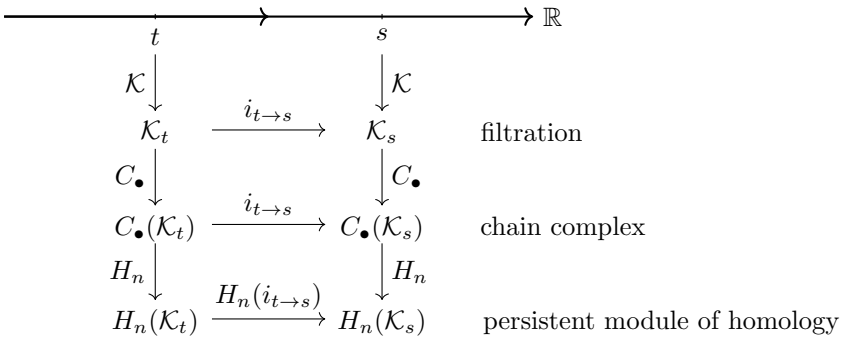
$$X_0 \xrightarrow{f_0} \cdots \xrightarrow{f_{k-1}} X_k \xrightarrow{f_k} X_{k+1} \xrightarrow{f_{k+1}} \cdots .$$

We begin by utilizing $\mathbb{R}$-indexed diagrams to allow for greater generality, although we will show in the next section that in practice, it often reduces to $\mathbb{N}$-indexed.

Using this framework, we can represent our multi-scale topological object of interest as $\mathcal{K} \in \mathbf{CWCplx}^{(\mathbb{R},\leq)}$ with $\mathcal{K}(t \to s)$ given by inclusions. This is referred to as an $\mathbb{R}$**-indexed filtration**. According to Theorem 2.1, by composing the chain complex functor and the n-th homology functor H_n over a field coefficient $\mathbb{F}$[2], we obtain a diagram of vector spaces of homology $H_n \circ C_\bullet \circ \mathcal{K} \in \mathbf{Vec}_\mathbb{F}{}^{(\mathbb{R},\leq)}$. We refer to diagrams in $\mathbf{Vec}_\mathbb{F}{}^{(\mathbb{R},\leq)}$ as $\mathbb{R}$**-indexed persistence modules over** $\mathbb{F}$.

It is helpful to visualize this process using the diagram below. We see that the functorality of homology assigns an algebraic invariant to a filtration, resulting in a persistent module formed by homology vector spaces. This once again connects topology and algebra in the multi-scale setting.

[2]In Section 4, we will explain why we limit ourselves to a field coefficient.



We can now formally define the notion of "birth", "death" and "persistence" with respect to the diagram.

Definition 3.2 (Birth, death, persistence)**.** *Given a persistence module of homology $(H_n(\mathcal{K}_\bullet), H_n(i_\bullet))$, we say that a homology class $[h] \in H_n(\mathcal{K}_j)$ is **born** at time j if $[h] \notin H_n(\mathcal{K}_t)$ for any $t \le j$. $[h]$ **dies** at time $k \ge j$ if k is the smallest number with $H_n(i_{j\to k}))([h]) = 0$ (allowing $k = \infty$ by convention).*

*The **persistence** of $[h]$ is death time − birth time.*

We can also restrict our attention to two timestamps $s \le t$ along the diagram, and consider the homology class at s that persists from s to t, still generating non-trivial homology at t.

Definition 3.3 (Persistent homology, persistent Betti number)**.** *Given a filtration $(\mathcal{K}_\bullet, i_\bullet)$ and $s \le t$, the $(t-s)$-**persistent** n-**th homology** of $\mathcal{K}_s$ is $PH_n^{s\to t}(\mathcal{K}) := \mathrm{img}(H_n(i_{s\to t}))$. The corresponding **persistent Betti number** $\beta^{s\to t}$ is then the group rank of $PH_n^{s\to t}$ over integral coefficient, and $\beta_{\mathbb{F}}^{s\to t}$ is the vector space dimension of $PH_n^{s\to t}$ over field coefficient $\mathbb{F}$.*

Equivalently, $PH_n^{s\to t}(\mathcal{K}) = \frac{i_{s\to t}(Z_n(\mathcal{K}_s))}{i_{s\to t}(Z_n(\mathcal{K}_s)) \cap B_n(\mathcal{K}_t)}$, which has a nice geometric interpretation: the equivalence classes of n-cycles in $\mathcal{K}_s$ that have not been filled and become n-boundaries in K_t.

3.2 Construction of filtration for image data

Suppose we have greyscale/binary k-d image data. Let $\Omega = [0, n_1] \times \cdots \times [0, n_k]$ be the domain, and $\mathcal{K}$ be the cubical complex taking the entire domain. We now outline how to construct an $\mathbb{R}$-indexed filtration of cubical complexes $(\mathcal{K}_\bullet, i_\bullet)$. The common procedure is:

Step 1 Construct a real-valued image data $f : \Omega \to \mathbb{R}$.

> We can directly use the grayscale image data like in Figure 4; otherwise, if the image data is binary or if we do not expect the grayscale value to contain much information and binarization does not cause significant information loss, we can do a binarization first and construct a task-specific f that is up to the design choice.

Step 2 Extend it to a function on the cubical complex $f : \mathcal{K} \to \mathbb{R}$. As f is already defined on the top-dimensional cells, we define $f(\sigma) = \min\{f(x) : \sigma \subseteq x, x \text{ is a top-dimensional cell}\}$. The sublevel set $(f^{-1}((-\infty, t]))_{t\in\mathbb{R}}$ then defines a filtration, referred to as a **low-star filtration**.

We give some common examples of task-specific choices of f in Step 1. We refer interested readers to [22] for more interesting constructions.

Example 3.1 (Signed distance filtration)**.** *Given a binary image, we define f to be the positive distance to the boundary on the 1-valued pixels/voxels and the negative distance to the boundary on the 0-valued pixels/voxels. This can also be applied to three-class images, where we assign f to ∞ on the parts taking the third value.*

This is useful for e.g. segmented medical images [43]. Geometric information such as size, shape, and relative locations are encoded in the distance function and hence captured by PH through this filtration.

Example 3.2 (Height filtration)**.** *Starting from a k-d binary image $B : \Omega \to \{0, 1\}$, if we are interested in the shapes, we can choose a unit direction vector $v \in \mathbb{R}^k$ and use*

$$f(x) = \begin{cases} \langle x, v\rangle & \text{if } B(x) = 1 \\ \max\limits_{y\in\Omega}\langle y, v\rangle & \text{if } B(x) = 0 \end{cases}$$

We refer readers to [54] to see why the corresponding filtration is useful in extracting shape information.

Example 3.3 (Filtration based on the prediction map)**.** *Deep learning models for image segmentation or reconstruction typically generate a prediction value in the range of [0,1] for each pixel/voxel. It is useful to generate a suplevel set filtration using the prediction map to analyze prediction results with different binarizations. Comparing the PH generated from filtration based on the prediction maps and the ground truth helps identify topological differences so that we can drive the predictions towards the ground truth. We refer readers to [27, 17] for more details.*

By employing a filtration that has a clear geometric interpretation, such as the ones presented in Examples 3.2 and 3.1, PH transcends its original purely topological foundation and incorporates significant geometric insights. PH can then be seen as a geometric analysis of topological features, as it captures the structural characteristics across different geometric scales.

Lastly, we remark that a general $\mathbb{R}$-indexed persistence module would be an infinitely-dimensional object that is computationally intractable. However, as there are only finitely many values in the image data, the corresponding persistent module of homology is usually "tame".

Definition 3.4 (Tame persistent module)**.** *For a persistence module $F \in \mathbf{Vec}^{(\mathbb{R},\leq)}$, we say $t \in \mathbb{R}$ is a **regular value** if there exists an open interval $I \ni t$, such*

that for all $a < b \in I$, $F(a \to b)$ is an isomorphism. Otherwise, t is a ***critical value****.*

If there are only finitely many critical values, and vector spaces $F(t)$ are finite-dimensional for all t, we say F is ***tame****.*

By re-indexing the finite set of critical values with $\mathbb{N}$, the tame persistence module is equivalent to $\mathbb{N}$-indexed persistence module of finite type.

Definition 3.5. *(Finite type) For a $\mathbb{N}$-indexed persistence module $(V_\bullet, f_\bullet)$, we say it is of* ***finite type*** *if V_i is finite dimensional for all i, and there exists $N \in \mathbb{N}$ such that f_n is isomorphism for all $n \geq N$. We denote the category of $\mathbb{N}$-indexed persistent module of finite type as $\mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$.*

We can similarly define a $\mathbb{N}$-indexed filtration $\mathcal{K} \in \mathbf{CWCplx}^{(\mathbb{N},\leq)}$ of ***finite type****.*

In other words, by suitable reindexing, image data leads to a filtration of finite type $\mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \cdots \subseteq \mathcal{K}_k = \mathcal{K}$, with persistent module of homology of finite type $H_n(\mathcal{K}_0) \xrightarrow{f_0} \cdots \xrightarrow{f_{k-1}} H_n(\mathcal{K}_k)$.

4 Barcode representation

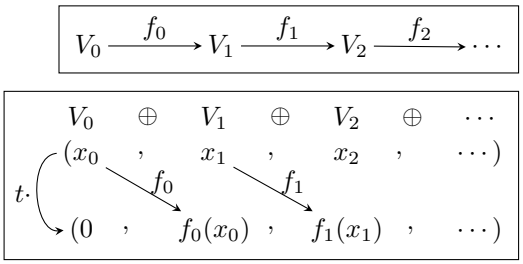
In the previous section, we introduced a new algebraic tool, the persistent module, for studying multi-scale topology. Our formulation of the persistent module as a diagram is intuitive, informative, and allows for natural generalization through functorality. However, it is unclear whether the persistent homology still exhibits the decomposability and computability described in Section 2.3, which are crucial for practical use. Moreover, while Figure 4 illustrates a simple example where multi-scale topological information can be represented by a simple barcode, it is unclear whether such a pairing of birth and death necessarily exists or is unique.

Fortunately, [57] demonstrate that the persistent module of homology can be effectively incorporated into the standard homology framework using a different formulation. This not only elucidates the algebraic structure of the persistent module, but also naturally leads to decomposability and computability. Specifically, this approach yields a unique, computable, and interpretable barcode representation of the persistent module of homology, extending the theoretical idea to a useful practical tool for data analysis.

4.1 Structure of the persistent module

At the end of Section 3.2, we argue that it is sufficient to study the $\mathbb{N}$-indexed persistence module of finite type $(V_k, f_k)_{k\in\mathbb{N}}$. We begin by naturally combining the vector spaces as a direct sum $V = \bigoplus_{i\in\mathbb{N}} V_i$. The morphisms $f_\bullet$ connecting the vector spaces can then be encoded by expanding the field of coefficients $\mathbb{F}$ to be the polynomial ring $\mathbb{F}[t]$. Specifically, we define $t \cdot (x_0, x_1 \dots) = (0, f_0(x_0), f_1(x_1) \dots)$, i.e., shifting along the sequence, for any $(x_0, x_1 \dots) \in V$.

This reformulates a persistence module as a $\mathbb{F}[t]$-module and is illustrated in the diagram below.



To formally establish this construction, we can show an isomorphism in the categorical sense.

Definition 4.1. *A **graded** $\mathbb{F}[t]$**-module** is a $\mathbb{F}[t]$-module M s.t. $M = \bigoplus_{k=0}^{\infty} M_k$, and $t^m M_n \subseteq M_{m+n}$ for any $m, n \in \mathbb{N}$. We say M is of **finite type** if each M_n is finite-dimensional.*

*Any non-zero elements $at^k \in \mathbb{F}[t]$, or $e \in M_k$ are said to be **homogneous of degree** k in $\mathbb{F}[t]$ and M respectively. A **homogeneous basis** of M is a basis consisting of homogeneous elements in $\mathbb{F}[t]$.*

Theorem 4.1. *[[9]] The category of $\mathbb{N}$-indexed persistence module of finite type, $\mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$ is isomorphic to the category of finite-type graded $\mathbb{F}[t]$-module.*

Proof. Use $\mathcal{C}$ to denote the category of finite-type graded $\mathbb{F}[t]$-module.

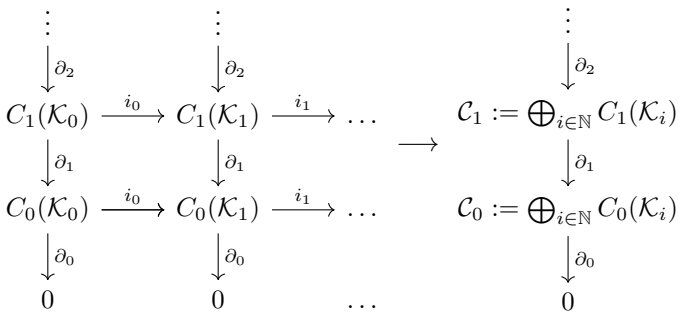
First, we define a functor $F : \mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)} \to \mathcal{C}$. We have already described how F maps the objects of the category. A morphism η of $\mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$ consists of linear maps $\eta_i : V_i \to V_i'$ for $i \in \mathbb{N}$ such that the diagram (left) commutes. $F(\eta) : \bigoplus_{i\in\mathbb{N}} V_i \to \bigoplus_{i\in\mathbb{N}} V_i'$ is defined canonically as $(x_1, x_2, \dots) \mapsto (\eta_1(x_1), \eta_2(x_2), \dots)$. The commutativity of the diagram (left) ensures that $F(\eta)$ respects the action of t; meanwhile by definition, $F(\eta)$ respects the grading of $\mathcal{C}$. Therefore, $F(\eta)$ is a graded morphism of $\mathcal{C}$, and F is indeed a functor.

$$\begin{array}{ccccc} V_1 & \xrightarrow{f_1} & V_2 & \xrightarrow{f_2} & \dots \\ {\scriptstyle\eta_1}\downarrow & & \downarrow{\scriptstyle\eta_2} & & \\ V_1' & \xrightarrow{f_1'} & V_2' & \xrightarrow{f_2'} & \dots \end{array} \longrightarrow \bigoplus_{i\in\mathbb{N}} V_i \xrightarrow{F(\eta)} \bigoplus_{i\in\mathbb{N}} V_i'$$

We can similarly define a functor $G : \mathcal{C} \to \mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$, and check F, G are mutually inverse to each other. □

Finally, we apply the theorem to the context of persistent homology. Starting from a $\mathbb{N}$-indexed filtration of finite type $\mathcal{K} \in \mathbf{CWCplx}^{(\mathbb{N},\leq)}$ mentioned at the end of Section 3.2. By Theorem 2.1, composing with the assignment of n-th cellular chain over field $\mathbb{F}$, we get $C_n \circ \mathcal{K} \in \mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$. Theorem 4.1 then suggests that it can be thought of as a graded $\mathbb{F}[t]$-module of finite type $\mathcal{C}_n := \bigoplus_{i\in\mathbb{N}} C_n(\mathcal{K}_i)$. As shown in the diagram below, this collapses a

structure of persistence complex (left) to a standard chain complex (right). Now the boundary operators $\partial_\bullet$ (right) are graded morphisms, i.e. $\mathbb{F}[t]$-module homomorphisms that respect grading. The persistence module of homology $H_n \circ C_\bullet \circ \mathcal{K} \in \mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$ over field $\mathbb{F}$ can then be thought of as the standard homology over the chain complex formed by $\mathcal{C}_n$ (right). Meanwhile, itself can also be thought of as a graded $\mathbb{F}[t]$-module of finite type $\mathcal{H}_n = \bigoplus_{i\in\mathbb{N}} H_n(\mathcal{K}_i)$.



4.2 Decomposition and computability

With the formulation that puts the theory of persistent homology under the umbrella of standard homology theory, we can now study it in the same vein as in Section 2.3, but extra care is needed for the additional structure of grading.

We will use a running example to illustrate the key ideas.

Example 4.1. *Figure 5 shows a* 1×3 *greyscale image, its corresponding cubical complex and low-star filtration based on Section 3.2. Intuitively, it admits a barcode shown in the Figure. We aim to compute* $\mathcal{H}_1$ *and* $\mathcal{H}_0$ *over field* $\mathbb{Z}_2$*.*

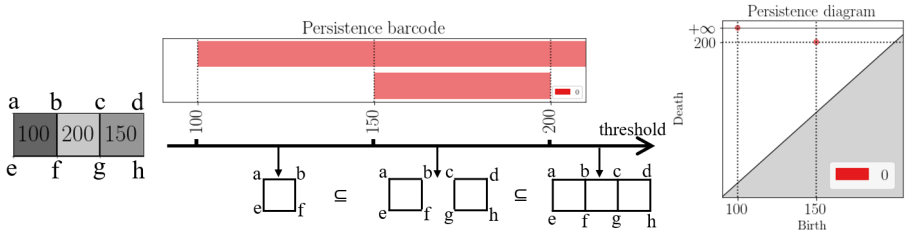


Figure 5: A 2×3 greyscale image, its cubical complex and filtration

Recall that the n-chain $C_n(\mathcal{K}_i)$ over field $\mathbb{F}$ is a vector space generated by n-cells in $\mathcal{K}_i$. Now let $\{\sigma_1^n, \ldots, \sigma_{N_n}^n\}$ be the set of n-cells in $\mathcal{K} := \bigcup \mathcal{K}_i$. A homogeneous basis of the graded $\mathbb{F}[t]$-module $\mathcal{C}_n = \bigoplus_{i=1}^{\infty} C_n(\mathcal{K}_i)$ is then given by $\{t^{\text{birth}(\sigma_1^n)}(\sigma_1^n, 0, \ldots), \ldots, t^{\text{birth}(\sigma_{N_n}^n)}(\sigma_{N_n}^n, 0, \ldots)\}$. Note that the i-th element in the basis is homogeneous with degree $= \text{birth}(\sigma_i^n)$.

It is convenient to overload notation σ_i^n to also represent its corresponding element $t^{\text{birth}(\sigma_i^n)}(\sigma_i^n, 0, \dots)$ in $\mathcal{C}_n$. We say $\mathcal{C}_n$ is freely generated by $\{\sigma_1^n, \dots, \sigma_{N_n}^n\}$ where $\deg(\sigma_i^n) = \text{birth}(\sigma_i^n)$. We can write $\mathcal{C}_n$ as a direct sum $\mathcal{C}_n = \bigoplus_{i=1}^{N_n} t^{\deg(\sigma_i^n)}\mathbb{F}[t]$.

Example 4.1 (continued)**.** *With the notations shown in Figure 5, we can express the canonical homogeneous bases and corresponding decomposition of $\mathcal{C}_n$, as summarised in the Table 1.*

	Homogeneous basis	Decomposition
$\mathcal{C}_0$	$\deg = 0 : a, b, e, f$ $\deg = 1 : c, d, g, h$	$\mathcal{C}_0 \cong \mathbb{F}[t]^4 \oplus (t\mathbb{F}[t])^4$
$\mathcal{C}_1$	$\deg = 0 : ab, ef, ea, fb$ $\deg = 1 : cd, gh, gc, hd$ $\deg = 2 : bc, fg$	$\mathcal{C}_1 \cong \mathbb{F}[t]^4 \oplus (t\mathbb{F}[t])^4 \oplus (t^2\mathbb{F}[t])^2$
$\mathcal{C}_2$	$\deg = 0 : abef$ $\deg = 1 : cdgh$ $\deg = 2 : bcfg$	$\mathcal{C}_2 \cong \mathbb{F}[t] \oplus t\mathbb{F}[t] \oplus t^2\mathbb{F}[t]$

Table 1: Canonical basis and decomposition of $\mathcal{C}_n$

We again want to use the boundary matrices M_n, M_{n-1} to compute $\mathcal{H}_{n-1} = Z_{n-1}/B_{n-1}$ just like in Section 2.3. We highlight the differences arising from the grading:

1. Equation (1) only required knowing $\text{rank}(Z_{n-1})$ from SNF of M_{n-1} and diagonal values in SNF of M_n. However, with the grading structure, we also need to keep track of the grading of basis elements of Z_{n-1}, B_{n-1} after the basis transformation. One consequence is that the initial M_n should be constructed with respect to C_n and Z_{n-1} instead of C_{n-1}.

2. M_n is now a matrix over $\mathbb{F}[t]$. When both row basis $(\sigma_i^{n-1})_i$ and column basis $(\sigma_j^n)_j$ are homogeneous, elements in M_n are all homogeneous too. The constant term is the same as in Section 2.3, and the exponent of t satisfying
$$\deg((M_n)_{ij}) = \deg(\sigma_j^n) - \deg(\sigma_i^{n-1}) \tag{2}$$
for any i, j such that σ_i^{n-1} is a face of σ_j^n (otherwise, $(M_n)_{ij} = 0$).

Example 4.1 (continued)**.** *We first try to compute $\mathcal{H}_0$.*

Note that $Z_0 = C_0$ has a homogeneous basis calculated in Table 1. We can express M_1 using formula (2), and reduce it to the SNF, as shown in the first row of Figure 6. After transformation, we see Z_0 admit a basis $\{c, d + c, g, h + d + c + g, a, b + a, e, f + b + a + e + t(h + d + c + g)\}$, and B_0 has basis $\{c, d + c, g, t(h + d + c + g), a, b + a, e\}$. Hence $\mathcal{H}_0 \cong \mathbb{F}[t]/(t) \oplus \mathbb{F}[t]$.

We now move on to compute $\mathcal{H}_1$. From SNF of M_1, we know Z_1 is freely generated by z_1, z_2, z_3, all of which are homogeneous. It is then easy to write out M_2 with respect to this basis of Z_1. After transforming to SNF, we get $\mathcal{H}_1 = 0$.

You may notice that the row and column bases and matrix elements of M_n remain homogeneous after transforming to SNF. This is not a coincidence.

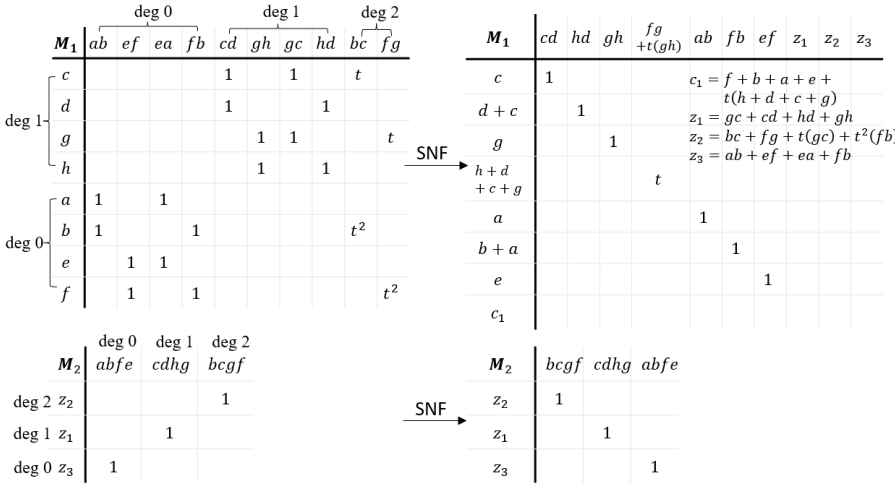


Figure 6: Compute SNF of M_1 and M_2

Proposition 4.1. *In general, the SNF transformation of all boundary matrices can be done in such a way that the row and column bases and all matrix elements remain homogeneous in every step.*

Proof. The proof has two steps:

Step 1 We prove the proposition holds for each M_n if its initialised construction has homogeneous row and column bases.

Step 2 We prove by induction that if we reduce M_n sequentially from $M_0, M_1, \ldots,$ then we can always have homogeneous bases to start with.

We start from Step 1. Fix n, and assume M_n has homogeneous row and column bases. We first order the bases such that the degrees of the basis elements increase from left to right, and bottom to top (just like what is done in the Example). Denote the row basis of M_n as $(\sigma_i^{n-1})_i$ and the column basis as $(\sigma_i^n)_i$. A direct consequence is that Equation (2) ensures that the degrees of elements in M_n are increasing from left to right, and top to bottom.

We consider doing SNF reduction in two steps: (a) Use elementary column operations to reduce it to Column Echelon Form; (b) Use elementary row operations to further reduce it to SNF.

Because of the ordering, Step (a) can be done using only two kinds of column operations, both maintaining the homogeneity of basis and matrix elements.

1. Swapping $\operatorname{col} i, \operatorname{col} j$, corresponding to swapping column basis σ_i^n, σ_j^n.

2. Replacing $\operatorname{col} j$ with $\operatorname{col} j + at^k(\operatorname{col} i)$ where $i < j$, and $k = \deg(\sigma_j^n) - \deg(\sigma_i^n)$. This corresponds to replacing σ_j^n with $\sigma_j^n - at^k(\sigma_i^n)$.

Step (a) results in a Column Echelon Form where the pivots are homogeneous with increasing degrees from top to bottom. As such, Step (b) can be done with only one type of row operation: Replacing $\operatorname{row} j$ with $\operatorname{row} j + at^k(\operatorname{row} i)$ where $i > j$, and $k = \deg(\sigma_j^{n-1}) - \deg(\sigma_i^{n-1})$. This corresponds to replacing σ_i^{n-1} with $\sigma_i^{n-1} - at^k(\sigma_j^{n-1})$. Again, it maintains the homogeneity of basis and matrix elements; meanwhile, all pivot elements and the degree of the row basis elements are unchanged.

We highlight that the proposition holds in both Step (a) and (b).

Now we prove Step 2 by induction. For $n = 0$, $Z_0 = C_0$ has a canonical homogeneous basis. The induction hypothesis on M_n implies that the basis of Z_n obtained from SNF of M_n is homogeneous. Therefore, M_{n+1} is initialised with homogeneous bases. Therefore, the proposition holds. □

Decomposition and barcode representation The first nice consequence of Proposition 4.1 is that it constitutes a constructive proof for the existence of a decomposition that resembles the Structure Theorem (1).

Suppose the SNF of M_n is $\operatorname{diag}(d_1, \ldots, d_{l_n}, 0, \ldots, 0)$ and $b_i := \deg(d_i)$. Suppose the first l_n rows have basis elements of degrees $(\alpha_i)_{i=1}^{l_n}$ and the $N_n - l_n$ zero rows have basis elements of degrees $(\beta_i)_{i=1}^{N_n - l_n}$, then

$$\mathcal{H}_{n-1} \cong \left(\bigoplus_{i=1}^{l_n} t^{\alpha_i} \mathbb{F}[t]/(t^{b_i}) \right) \oplus \left(\bigoplus_{i=1}^{N_n - l_n} t^{\beta_i} \mathbb{F}[t] \right). \tag{3}$$

It is convenient to apply Theorem 4.1 to obtain an isomorphism in the category of $\mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$

$$(H_{n-1}(\mathcal{K}_\bullet), H_{n-1}(i_\bullet)) \cong \left(\bigoplus_{i=1}^{l_n} I(\alpha_i, \alpha_i + b_i) \right) \oplus \left(\bigoplus_{i=1}^{N_n - l_n} I(\beta_i, \infty) \right) \tag{4}$$

where the **interval module** $I(b, d) \in \mathbf{FinVec}_{\mathbb{F}}^{(\mathbb{N},\leq)}$ is defined by the sequence

$$0 \xrightarrow[f_1]{0} \cdots \xrightarrow[f_{b-1}]{0} \mathbb{F} \xrightarrow[f_b]{\mathrm{Id}_{\mathbb{F}}} \cdots \xrightarrow[f_{d-1}]{\mathrm{Id}_{\mathbb{F}}} \mathbb{F} \xrightarrow[f_d]{0} 0 \xrightarrow[f_{d+1}]{0} \cdots$$

The SNF transform of the graded $\mathbb{F}[t]$-module computes a compatible basis for $H_n(\mathcal{K}_\bullet)$ across the persistent module, with respect to which the persistent module of homology is uniquely decomposable as a direct sum of interval modules, and each $I(a, b)$ can be geometrically interpreted as a homology class that is exactly born at a and die at b. It is also easy to prove the decomposition is unique.

Notably, this shows the existence and uniqueness of the compact and interpretable barcode representation that was intuitively introduced at the start of Section 4.

Definition 4.2 (Persistence Barcode/Diagram)**.** *The* ***barcode*** *of a $\mathbb{R}$-indexed persistent module of homology $(H_n(\mathcal{K}_\bullet), H_n(i_\bullet))$ is a multi-set of intervals $\mathcal{B} = [[b_i, d_i] : i \in I, b_i < d_i \in \mathbb{R} \cup \{\infty\}$, multiplicity $m(b_i, d_i) < \infty]$[3], such that $(H_n(\mathcal{K}_\bullet), H_n(i_\bullet)) \cong \bigoplus_{i=1}^{n} I(b_i, d_i)^{m(b_i,d_i)}$.*

An equivalent representation is the ***persistence diagram*** *$\mathcal{D} = [(b_i, d_i) : i \in \mathbb{N}, m(b_i, d_i) < \infty] \cup [(x, x) : x \in \mathbb{R}, m(x, x) = \infty] \subseteq \mathbb{R}^2$, which take the end points of the intervals in the barcodes. We add all diagonal points with infinite multiplicity for convenience in Definition 4.3 in the next section.*

It is easy to see that the persistent Betti number $\beta_{\mathbb{F}}^{s\to t}$ (see Definition 3.3) can be read from the barcode as the number of intervals containing $[s, t]$.

Example 4.1 (continued)**.** *After transforming back to the original filtration index, the previous example yields barcodes for $\mathcal{H}_0$ $[150, 200], [100, \infty]$, and no barcode for $\mathcal{H}_1$. This coincides with the persistence barcode and diagram plotted in Figure 5.*

It is worth noting that the decomposability property we have introduced is not a trivial result. Specifically, similar to the decomposability of standard homology, which necessitates the coefficient ring to be a PID (as discussed in Section 2.3), Proposition 4.1 does not hold for a general polynomial ring $R[t]$, since $\mathbb{F}[t]$ is a PID, but for example, $\mathbb{Z}[t]$ is not. This explains why our focus in persistent homology theory is restricted to field coefficients since, otherwise, the desirable barcode representation may not be achievable.

Computability Furthermore, the approach outlined in Proposition 4.1 is essential to the algorithmic computability of the barcode. Firstly, the homogeneity of the matrix elements satisfying Equation (2) throughout the reduction means that the matrix reduction can be done over $\mathbb{F}$ rather than $\mathbb{F}[t]$, thereby avoiding potentially expanding polynomial elements and achieving computational efficiency. Secondly, the degree of the row basis remains constant, obviating the need to keep track of the basis change.

Finally, it is necessary to automatically express M_{n+1} with respect to the row basis Z_n after obtaining Z_n from SNF of M_n. This can be accomplished by first writing out M_{n+1} with respect to the canonical basis of C_n and C_{n+1} and subsequently removing the rows corresponding to pivot columns in M_n. This follows from the fact that $M_n M_{n+1} = 0$, and the proof is straightforward. Interested readers may refer to Lemma 4.2 in [57] for more details.

4.3 Isometry theorem

We have seen that the persistent module of homology over field coefficient admits a compact, interpretable and computable barcode representation. It turns out that barcode has another very desirable property of being an isometry.

We impose distances on both the space of the persistence module and the space of barcodes, denoted by **Bar**.

[3]We use []to denote a multi-set, where each object has a multiplicity m.

Definition 4.3 (Bottleneck distance, p-Wasserstein distance)**.** *For two persistence diagrams $\mathcal{D}_1, \mathcal{D}_2$, the* ***bottleneck distance*** *between them*

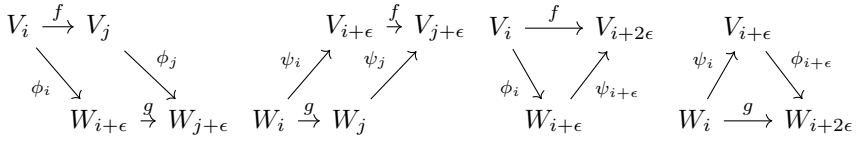
$$W_\infty(\mathcal{D}_1, \mathcal{D}_2) = \inf_{\text{bijections } f:\mathcal{D}_1 \to \mathcal{D}_2} \left(\sup_{(b,d)\in\mathcal{D}_1} \|(b,d) - f(b,d)\|_\infty \right).$$

It can be seen as a special case of the ***p-Wasserstein distance***

$$W_p(\mathcal{D}_1, \mathcal{D}_2) = \inf_{\text{bijections } f:\mathcal{D}_1 \to \mathcal{D}_2} \left(\sum_{(b,d)\in\mathcal{D}_1} \|(b,d) - f(b,d)\|_\infty^p \right)^{1/p}.$$

Note that this distance arises in optimal transport theory, intuitively understood as the optimal transport cost between two "piles". Here, it is a metric on the barcodes of tame persistence modules.

Definition 4.4 (Interleaving distance)**.** *For two $\mathbb{R}$-indexed persistence modules $(V_\bullet, f_\bullet), (W_\bullet, g_\bullet)$, an* ***$\epsilon$-interleaving*** *is two families of linear maps $(\phi_i : V_i \to W_{i+\epsilon})_{i\in\mathbb{R}}$ and $(\psi_i : W_i \to V_{i+\epsilon})_{i\in\mathbb{R}}$ such that the diagrams below commute for all $i \le j$.*



The ***interleaving distance***

$$d_i(V, W) = \inf\{\epsilon > 0 : \textit{there is an } \epsilon \textit{ interleaving between } V, W\}$$

Intuitively, interleaving distance measures how far they are from being isomorphic.

Theorem 4.2 (Isometry theorem [12, 9, 36])**.** *Endowing the space of barcode* **Bar** *with the bottleneck distance W_∞, and the space of persistence modules with the interleaving distance d_i, then the assignment of barcode is an isometry. i.e. for any two persistence modules $(V_\bullet, f_\bullet), (W_\bullet, g_\bullet)$ with barcodes $\mathcal{B}_1, \mathcal{B}_2$, we have $W_\infty(\mathcal{B}_1, \mathcal{B}_2) = d_i(V, W)$.*

The proof of Theorem 4.2 is rather intricate and not closely connected to other parts of this essay. Interested readers may refer to Chapter 3 of [48] for a comprehensive discussion on this matter.

That being said, it is worth highlighting the significance of this result. First, the "$\le$" part ensures the stability of the barcode representation to perturbations of the persistence module. This implies that the barcode is robust to noise and other forms of perturbations. Second, the "$\ge$" part guarantees that the barcode accurately distinguishes differences in persistence modules, making it an informative characterization of topological features of images.

5 Computation of barcode

5.1 Computation algorithm

The computability discussion in Section 4.2 could be streamlined to provide the final algorithm for computing the persistent barcode. Specifically, we present the algorithm for coefficient $\mathbb{Z}_2$ and see that it is sufficient to do matrix reduction on one big boundary matrix. Notably, PH on $\mathbb{Z}_2$ alone is usually sufficiently informative for practical data analysis.

Step 1 Construct the filtration. Order all cells in $\mathcal{K}$ as $\sigma_1, \ldots, \sigma_N$ such that if $\deg(\sigma_i) \leq \deg(\sigma_j)$ or $\deg(\sigma_i) = \deg(\sigma_j), \sigma_i$ is a face of σ_j, then σ_i precedes σ_j. Otherwise, the ordering is arbitrary. Build the $N \times N$ $\mathbb{Z}_2$-valued boundary matrix M with respect to the ordering.

Step 2 Use Algorithm 1 for reduction of the matrix. We define $low(j)$ to be the largest i such that $M_{ij} \neq 0$. When the algorithm terminates, each row of the boundary matrix M can be the $low(j)$ for at most one column j, known as a pivot. This is in essence the same as a reduction to column echelon form with a slight difference in the formulation.

Algorithm 1 Matrix reduction

```
procedure REDUCTION(M)
    for i=1 to N do
        k ← low(i)
        while there exists j < i with low(j) = k ≠ 0 do
            col(i) ← col(i) − col(j) × M_ki/M_kj
        end while
    end for
end procedure
```

Step 3 Read the barcode from the reduced matrix.

For each column j, if $\mathrm{col}(j)$ is zeroed out, and $j = low(i)$ for some i, then it contributes a bar $[\deg(\sigma_i), \infty]$ to PH of dimension $\dim(\sigma_i)$. Intuitively, σ_i persists and does not die at the end of filtration. If $\mathrm{col}(j) \neq 0$, it contributes a bar $[\deg(\sigma_{low(j)}), \deg(\sigma_j)]$ to PH of dimension $\dim(\sigma_{low(j)})$. Intuitively, $\sigma_{low(j)}$ persists until death due to the birth of σ_j.

The time and memory complexity is usually dominated by Step 2, which is $O(n^3)$ where n is the total number of cells in $\mathcal{K}$, the largest complex in the filtration. i.e. computing PH has the same level of complexity as computing the standard homology $\mathcal{K}$. While this is a very desirable result, the applicability of PH in image analysis is still heavily limited by the computational burden as mentioned in 2.3.

Example 4.1 (continued)**.** *We run the algorithm on the previous example. The original and reduced matrix is shown in Figure 7. We left the readers to check that the operations are essentially the same as what was described in Section 4.2.*

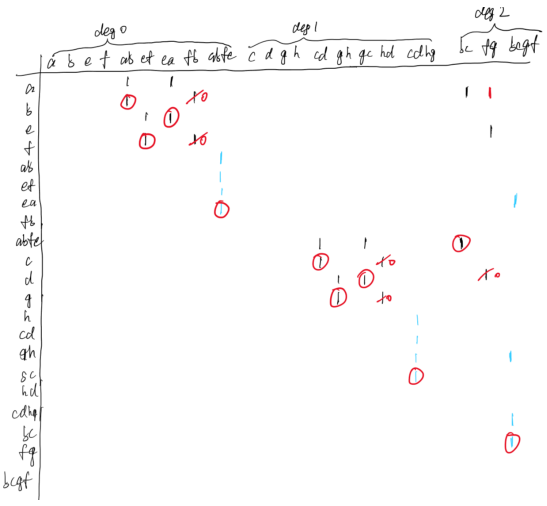


Figure 7: Run algorithm on the example. Red colour indicates the reduced matrix, while the pivots are circled.

5.2 Improving computational efficiency

In this section, we provide a brief overview of various computational efficiency enhancement techniques, with a particular focus on the cubical complex.

Filtration-building and data structure Step 1 of the algorithm on filtration construction is a non-trivial task that necessitates a well-designed data structure for storing and accessing boundary relations in a complex. Although several data structures have been proposed for simplicial complexes, such as the simplex tree [5], the cubical complex offers a more straightforward approach. Due to its rigid structure, a compact data structure called **CubeMap** has been introduced in [56]. For example, as illustrated in Figure 8, a 4×4 2D image can be represented by a 9×9 2D array, where the type of each element (i.e., face, edge, or vertex) can be directly inferred from its coordinates modulo 2. Additionally, boundary relations are also encoded directly in the coordinates.

Sparse matrix representation The matrix M used in the algorithm discussed in the previous section is inherently sparse, and thus, sparse matrix representations, as described in [57] and [19], are typically employed to enhance memory efficiency.

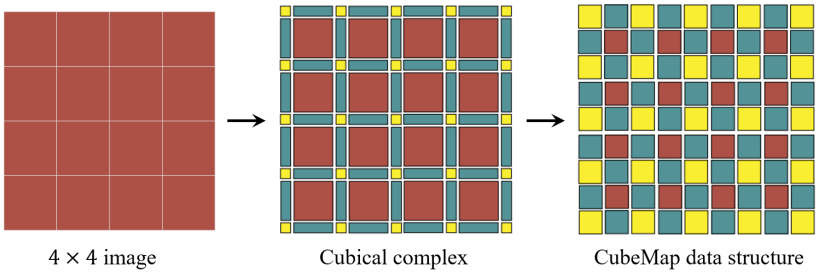


Figure 8: CubeMap data structure (adapted from [56])

Reducing the complex Before the advent of PH, there had been several attempts to improve the computational efficiency of standard homology by pre-processing the data to reduce the size of the complex without changing its homology. Several of these methods have been proven to be particularly useful for cubical complexes, where the rigid use of same-sized cubes often leads to an unnecessarily large number of cubes. Some of the popular methods include:

(1) In [30], an algorithm called *elementary reductions* was introduced, which removes generators from the chain complex iteratively. Topologically, it corresponds to a deformation retract of a free space onto its complex. The effectiveness of this algorithm in the cubical setting was further studied in [31] and [28].

(2) In [44], a coreduction algorithm was introduced, which usually leads to more reduction in the cubical setting compared to (1), as demonstrated empirically.

(3) More recently, [24] suggests using the smaller Morse Complex from the Discrete Morse Theory.

As PH is deeply rooted in the theory of computational homology, these ideas have been successfully generalized to PH. For example, [45] generalized the coreduction algorithm and empirically demonstrated its effectiveness in the cubical complex. Additionally, [23] utilized the rigidity of the cubical complex to present an efficient algorithm for computing the Morse-Smale complex of 3D images. Although later work in [41] presented a more general algorithm for Morse reduction that works for any CW complex, the efficiency gain due to the cubical complex is still appreciated.

PH is also widely employed for point cloud data, i.e., points in a finite-dimensional metric space. In this case, a popular class of methods for reducing the size of the complex is through subsampling. The idea is to select a subset of landmark points to approximate the persistent homology of the entire point cloud. A probabilistic discussion of subsampling is presented in [13]. The idea is further developed in the construction of the witness complex, which chooses

a subset of landmark points but constructs a simplicial complex using information from the remaining points as well [53]. However, such methods are not applicable to the cubical complex, which highlights a significant difference in the computational treatment of the two data types.

6 Statistics and Machine Learning with persistent homology

In the previous sections, we have presented the theoretical framework and computational techniques of PH, which enable the automatic quantification of topological information in *one image*. In the current era of big data, there is a natural inclination to process big datasets of images and carry out statistical analysis and machine learning tasks based on topological features. However, the barcode representation poses several challenges due to the complicated geometry of the space of barcodes equipped with bottleneck distance or p-Wasserstein distance (see Definition 4.3). Although we can naturally use the notion of Fréchet mean[4], the mean is not always unique. Moreover, computing the bottleneck or p-Wasserstein distances is known to be time-consuming, as it requires solving an optimal transport problem.

Despite successful efforts to establish a statistical framework for barcodes [40, 55, 21], a more practically useful alternative is to map the barcode to a vector space, either finite- or infinite-dimensional, to facilitate the application of existing statistical and machine-learning techniques that operate in vector spaces. Various vectorization methods have been proposed in the literature for different purposes. They are discussed in Section 6.1, with a focus on functional summaries. To illustrate how they are employed in practice, we present two case studies for applications in material science and medical imaging respectively, in Sections 6.2 and 6.3. These case studies provide detailed discussions of the two introduced functional summaries, show their pros and cons, and offer heuristics to guide practitioners in selecting suitable vectorizations according to the task.

Finally, it is worth noting that the computational complexity of PH poses a major hurdle to performing statistics and machine learning on big image datasets, especially on 3D or large-sized images. Therefore, improving computational efficiency, as discussed in Section 5.2, has garnered much research attention as an essential step toward improving the practical utility of PH-based methods.

6.1 Vectorization

We now introduce a statistical set-up, treating barcodes as random variables $\mathcal{B} : (\Omega, \mathcal{F}, \mathcal{P}) \to \mathbf{Bar}$ where $(\Omega, \mathcal{F}, \mathcal{P})$ is a probability space, and the space of barcodes **Bar** is endowed with a pushed-forward probability measure. We

[4]For two persistence diagrams $\mathcal{D}, \mathcal{D}'$, their Fréchet mean is defined as the collection of midpoints of the line segments between $(b, d) \in \mathcal{D}$ and $f(b, d) \in \mathcal{D}'$, where f is the bijection of two diagrams that realizes the p-Wasserstein distance

define vectorisation as a measurable function $F : \mathbf{Bar} \to \mathcal{S}$, where the summary space $\mathcal{S}$ is a vector space. We assume that N barcodes, $\mathcal{B}_1, \ldots, \mathcal{B}_N \overset{iid}{\sim} \nu$ and their corresponding feature vectors $F(\mathcal{B}_1), \ldots, F(\mathcal{B}_N)$ are also i.i.d. To ensure tractability in vectorization, we truncate the intervals that persist indefinitely at the maximal filtration value, so that the barcodes are of the form $\mathcal{B} = \{[b_i, d_i] : i \in I, -\infty < b_i < d_i < \infty, \text{multiplicity } m(b_i, d_i) < \infty\}$.

Important design considerations Various proposed vectorizations of barcodes share some important design considerations, which include firstly, the *stability* and *discriminative power*. Recall from Section 4.3 that we discussed the stability and discriminative power of the barcode with respect to the persistence module that underpins its practical usefulness. Thus, it is important for vectorizations of barcodes to retain these properties. Stability is typically expressed in the form $d(F(\mathcal{B}), F(\mathcal{B}')) \leq W_p(\mathcal{B}, \mathcal{B}')$, where d is a metric on the summary space $\mathcal{S}$ and $1 \leq p \leq \infty$. This indicates that similar barcodes produce similar vector representations. On the other hand, while most proposed vectorizations fail to admit a distance that is lower bounded by the corresponding distance on barcodes due to the complicated geometry of the barcode space, it is insightful to consider the notion of being characteristic [15, 34] to indicate the discriminative power of the mean. It is an important property to justify using the mean summary to discriminate between two datasets.

Definition 6.1 (Characteristic)**.** *A vectorization $F : \mathbf{Bar} \to \mathcal{S}$ is **characteristic** if the map $\nu \mapsto \mathbb{E}_{\mathcal{B}\sim\nu}[F(\mathcal{B})]$ sending a probability measure ν on the* **Bar** *to the population mean of summary is injective.*

The second crucial consideration is the *interpretability* of the vectorization in terms of topology. Although statistical analysis and machine learning on vectorizations can be treated as a stand-alone task, it is desirable for the vectorization and the operations on it (such as addition and taking the mean) to have topological meaning and interpretability. This ensures that useful topological insights can be derived from data analysis results.

Thirdly, it is important for vectorization to offer *flexibility* in assigning relative importance to points in the persistence diagrams. This is particularly relevant in practical cases where prior knowledge indicates certain topological features related to points in certain regions of the persistence diagram are more relevant for the specific task at hand.

Functional Summaries We can map barcodes to either a finite- or infinite-dimensional vector space and use established results in multivariate and functional statistics, respectively. However, due to the infinite-dimensional nature and complex geometry of the space of barcodes with bottleneck distance, functional summaries are usually preferred due to their ability to retain more information.

A simple way to summarize the information contained in a barcode is to count the Betti number at each filtration level t. The **Betti curve** of a barcode

$\mathcal{B}$ is defined as $\beta(t) = \sum_{[b,d]\in\mathcal{B}} \mathbf{1}_{[b,d]}(t)$ [51] and is illustrated in Figure 9c, which shows the Betti curve corresponding to the persistence barcodes and diagrams in Figures 9a and 9b. However, the Betti curve is known to be unstable due to the use of indicator functions. A simple remedy is to replace the indicator function with a triangular function, motivating the persistent landscape.

Definition 6.2 (Persistence landscape[7][8])**.** *The* ***persistence landscape*** *of barcode $\mathcal{B}$ is a function λ on $\mathbb{N}\times\mathbb{R}$ with $\lambda(k,t) = \text{kmax}_{[b,d]\in\mathcal{B}}\left(\Lambda_{(b,d)}(t)\right)$, where $\Lambda_{(b,d)}(t) = \min(t-b, d-t)_+$ denotes the right-angle isosceles triangular function based at $[b,d]$.*

It admits a natural mean $\bar{\lambda}(k,t) = \frac{1}{N}\sum_{i=1}^{N}\lambda_i(k,t)$, and L_p norm $\|\lambda\|_p = \sum_{k=1}^{\infty}\left(\int_{-\infty}^{\infty}\lambda_k(t)^p dt\right)^{1/p}$ for $1 \le p < \infty$, and $\|\lambda\|_\infty = \sup_{k,t}\lambda_k(t)$. Figure 9e shows the persistence landscape corresponding to the same example.

Equivalently, $\lambda(k,t) = \sup\{h \ge 0 : \beta_{\mathbb{F}}^{(t-h)\to(t+h)} \ge k\}$, where $\beta_{\mathbb{F}}^{s\to t}$ is the persistence Betti number defined in Definition 3.3. As such, the sample mean $\bar{\lambda}(k,t)$ is topologically meaningful, reflecting the average value of the largest radius interval centred at t that is contained in k intervals in the barcodes $\mathcal{B}_1,\ldots,\mathcal{B}_N$. However, interpreting topological features of the sample mean directly is challenging, as $\bar{\lambda}$ does not have a corresponding barcode.

An important property of the persistence landscape is that the assignment of landscape to a barcode is invertible, ensuring that no information is lost. Furthermore, the method is parameter-free, but the direct usability of the persistence landscape is accompanied by the rigid assumption that more persistent features are more significant. Indeed, $\lambda(k,\cdot)$ intuitively measures the k-th dominant homological feature, as gauged by persistence.

Another functional summary, *persistence intensity* is also a popular choice in various applications.

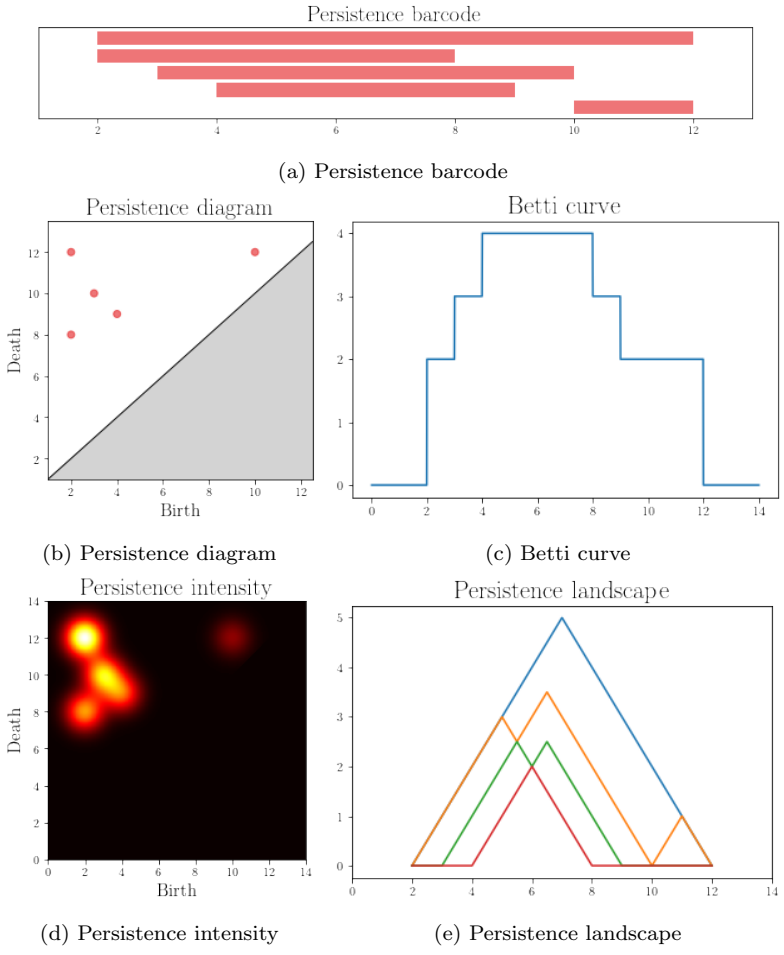
Definition 6.3 (persistence intensity [14][1])**.** *The* ***persistence intensity*** *of $\mathcal{B}$ is a bivariate function $f(x,y) = \sum_{[b,d]\in\mathcal{B}} w(b,d)\cdot K_H(x-b, y-d)$, where w is weighting function set to be 0 when persistence is 0, and K is a 2D-kernel function with bandwidth matrix H. A typical choice is the Gaussian kernel $K_\sigma(x,y) = \frac{1}{2\pi\sigma^2}\exp(-\frac{x^2+y^2}{2\sigma^2})$.*

Figure 9d illustrates the persistence intensity corresponding to the same example, with a Gaussian kernel with $\sigma^2 = 0.5$, and $w(b,d) = \frac{d-b}{\sum_{[b,d]\in\mathcal{B}} d-b}$.

The persistence intensity can be viewed as a kernel smoothing of the point measure $\Phi(x,y) = \sum_{[b,d]\in\mathcal{B}} w(b,d)\delta_{x-b,y-d}$, and hence the sample mean of the summary $\bar{f}$ resembles a kernel density estimator of Φ. $\bar{f}$ is much more interpretable than the mean persistent landscapes, as we shall see in the case studies.

Another advantage of the persistence intensity is its flexibility in choosing weights, kernels, and bandwidth as compared to the parameter-free persistence landscape. In certain applications such as [3], the homology of short/medium persistence is more important, and a customized weight function can reflect this knowledge.

We refer readers to [4] for a detailed review of standard statistical analysis on functional summaries, including the Strong Law of Large Numbers, the Central Limit Theorem, constructing confidence bands via bootstrapping, two-sample tests via permutation tests, etc. Additionally, techniques in Functional Data Analysis, such as Functional PCA and registration techniques to separate amplitude and phase in landscapes, can be readily employed. Examples of such attempts can be found in [43] and [39].



(a) Persistence barcode

(b) Persistence diagram

(c) Betti curve

(d) Persistence intensity

(e) Persistence landscape

Figure 9: Illustration of the proposed functional summaries

Finite-dimensional summaries Most machine learning algorithms require inputs in the form of finite-dimensional vectors. It is straightforward to extract such vectors from the functional summaries discussed above by suitable discretization techniques. One prominent example is the **persistent image** [1],

which is a discretization of the persistence intensity.

There are other simple finite-dimensional summaries that are used in practice. One such example, which will be used in Case Study 2, is the **ranked persistence**. This summary is a concatenation of two vectors, $(\mathbf{H_0}, \mathbf{H_1}) \in \mathbb{R}^{n_0+n_1}$, where $\mathbf{H_i} \in \mathbb{R}^{n_i}$ consists of the persistence in the i-th persistence barcodes, sorted in descending order. As only persistence information is retained, a large amount of information is lost.

Kernel-based vectorizations A rather different approach to vectorization is the kernel-based method. While the metric space of barcodes **Bar** does not possess a Hilbert space structure, a kernel k can be defined on **Bar**. An important property of a kernel is that $k(\mathcal{B}, \mathcal{B}') = \langle F(\mathcal{B}), F(\mathcal{B}') \rangle$ for all $\mathcal{B}, \mathcal{B}' \in \mathbf{Bar}$, for some feature map $F : \mathbf{Bar} \to H$, where H is a real Hilbert space. Therefore, by defining a valid kernel, we can implicitly work in a Hilbert Space. Kernel-based machine learning algorithms, such as Support Vector Machine can then be readily employed in this framework. While we will not go deeper into this topic, we refer readers to [50] and [10] for some prominent examples of kernels on barcodes.

6.2 Case Study 1: Mean persistent landscape characterising microstructure in phase separation dynamics

[18] applied PH to material science, and in particular study the phase separation mechanism of binary alloy in 2D space, as shown in Figure 10b. The study showed that mean persistence landscapes encode rich geometric and topological information about the evolving microstructures. This information is sufficient to solve an inverse problem of recovering the total mass from snapshots of the phase separation process.

6.2.1 Methods and results

Data generation and filtration construction Let $\rho_A(t, \mathbf{x})$ and $\rho_B(t, \mathbf{x})$ be the concentration of the two components of the alloy at time t and position $\mathbf{x} \in \Omega \subseteq \mathbb{R}^2$, such that $\rho_A + \rho_B = 1$. [18] generated numerical simulations for phase separation process using the Cahn-Hilliard-Cook model, a stochastic partial differential equation describing the space-time changes of the relative concentration difference $u = \rho_A - \rho_B$. Each dataset used N different total mass values $\mu_1, \ldots, \mu_N$, and for each μ_i, M solution paths of u were simulated with initial conditions of a random perturbation of a spatially homogeneous state, i.e., $u(0, \mathbf{x}) \approx \mu$ for all $\mathbf{x}$. Solution snapshots were saved at T regular time intervals.

By rescaling the solution range from $[-1, 1]$ to $[0, 256]$, each solution path can be seen as a time-series of real-valued image data, and hence the methods discussed in Section 3.2 could be used to construct filtrations of cubical complexes, where barcodes were then computed. Along the filtration of sublevel

sets, the corresponding cubical complex is initially the part dominated by component A and then grows to include parts dominated by B as the threshold grows beyond 128.

Classification Persistence landscapes $(\lambda^{k,n}_{m,t})_{n\in N,m\in M,t\in T,k\in\{0,1\}}$ were computed from Barcodes $(\mathcal{B}^{k,n}_{m,t})_{n\in N,m\in M,t\in T,k\in\{0,1\}}$ for each k-dimensional PH, n-th mass value, m-th initial condition, at time stamp t. Time-series of landscapes $P := (\lambda_t)_{t\in T}$ were considered, which can be seen as a function on $\mathbb{N}\times\mathbb{N}\times\mathbb{R}$. Therefore, it naturally admits an L_p norm and a well-defined mean. The dataset then contains time-series $P^{k,n}_m = (\lambda^{k,n}_{m,t})_{t\in T}$, and the sample mean $\bar{P}^{k,n}$ of k-dimensional PH and n-th mass value is taken over M different initial conditions. Figure 10a shows some sample outputs of the mean landscapes for 1-dimensional PH.

The inverse problem of recovering the mass value μ can be formulated as a classification problem that assigns a class $n \in N$, indicating mass μ_n, to a new collection of time-series of landscapes $(Q^k)_{k\in\{0,1\}}$, where k is the PH dimension. A simple classification scheme was proposed: For each dimension k, let $S^k_j \subseteq 1,\dots N$ be the first j classes whose sample mean is closest to Q^k in L_p distance. We assign $(Q_k)_k$ to class $n \in N$ if $\bigcap_{k=0}^{\infty} S^k_l = \{n\}$ for some l; otherwise, we say $(Q_k)_k$ fail to be classified.

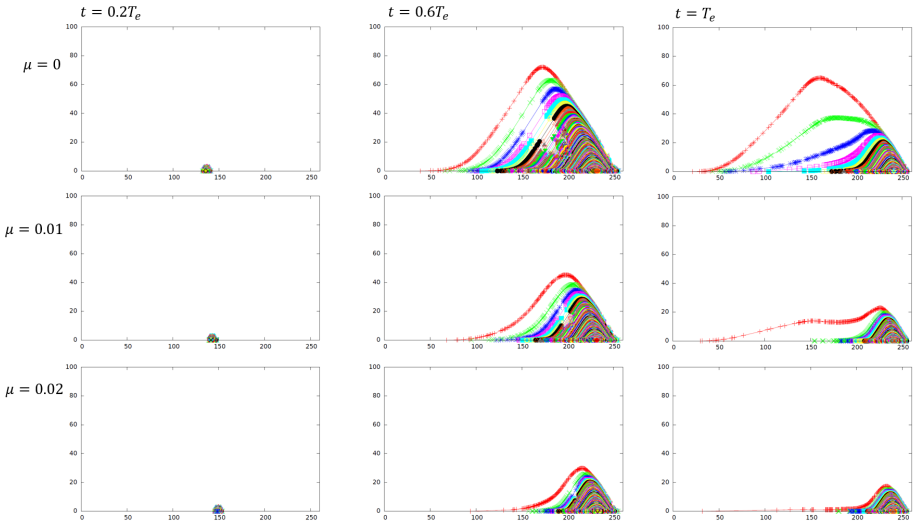
Results Despite being simple, the classification procedure showed high accuracy. With a dataset of $N = 51$ mass values, $T = 25$ time-stamps, and $M = 100$ initial conditions where only $M_T = 10$ out of M are used as the training set, classification using L_1, L_2, and L_∞ norms achieved 99%, 99%, and 94% accuracy on test data up to an error of ± 0.01 (misclassification by one class).

6.2.2 Discussion

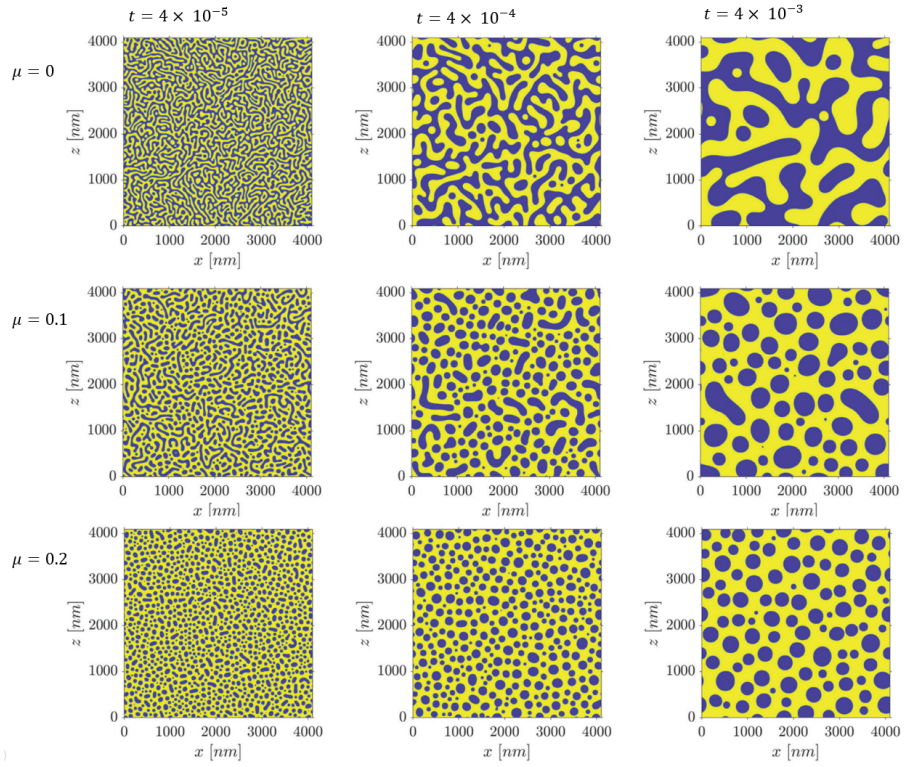
Beyond the work in [18], we aim to investigate deeper into: 1) The theoretical properties of persistence landscapes that underlie the success of the simple mean-based classification scheme; 2) while [18] did not make extensive efforts to derive topological insights from the mean landscapes, we intend to fill this gap by leveraging known qualitative properties of microstructure patterns in phase separation; 3) Finally, we aim to challenge a key conclusion drawn by [18], which we argue may not be fully supported by the theoretical understanding of PH that has been developed in the literature.

Theoretical guarantees of persistence landscapes In Section 6.1, we briefly touched upon the desirable properties of vectorizations, namely stability and discriminative power. In this section, we shall delve into specific details for persistence landscapes, which explain the high accuracy of the simple classification scheme proposed in [18].[18].

Recent reviews have demonstrated that persistence landscapes exhibit arguably the strongest proven stability, as evidenced by Table 1 of [16]. While



(a) Sample persistence landscapes for 1-dimensional PH for the microstructures evolving with time.[18]



(b) Sample snapshots for microstructures pattern produced by a different work [33].

Figure 10: Phase separation mechanisms simulated by the Cahn-Hillard-Cook model

most proposed vectorizations have only been shown to be stable with respect to a specific choice of metric, it is worth noting that persistence landscapes have multiple stability results.

Theorem 6.1 (Stability of persistence landscape [7]). *Let λ, λ' be the persistence landscapes of diagrams $\mathcal{D}, \mathcal{D}'$. We have*

(1) $\|\lambda - \lambda'\|_\infty \leq W_\infty(\mathcal{D}, \mathcal{D}')$;

(2) For $p \geq 1$, $\min(1, C\|\lambda - \lambda'\|_p^p) \leq W_p(D, D')^p$ *for some constant* $C > 0$.

Here $\|\cdot\|_p, 1 \leq p \leq \infty$ is the L_p distance on persistence landscapes (see Definition 6.2), and W_p, W_∞ are the p-Wasserstein distance and bottleneck distance on persistence diagrams (see Definition 4.3).

On the other hand, the persistence landscape is one of the few vectorization techniques that are provably discriminative. In particular, [7] establishes that for certain generic sets of persistence diagrams, it is possible to reconstruct the set of persistence diagrams from their mean persistence landscapes. This suggests that the persistence landscape is *characteristic* for empirical measures on such generic sets.

In the context of phase separation, and with the notation introduced in Section 6.2.1, we denote the time-series of barcodes as $\mathcal{B}_m^{k,n} = (\mathcal{B}_{m,t}^{k,n})_{t \in T}$. We assume that different initial conditions generate i.i.d. time-series of barcodes, $(\mathcal{B}_m^{k,n})_{m \in M} \overset{iid}{\sim} \nu^{k,n}$, where $\nu^{k,n}$ is a density that is distinct for each k and n indicating the dimension of PH and the mass value, respectively. The characteristic property of the persistence landscape implies that $\nu^{k,n}$ must induce distinct sample means $\bar{P}^{k,n}$, thereby providing theoretical guarantees for using sample means as classifiers.

Moreover, the stability property of the persistence landscape suggests that for any time-series of barcodes $\mathcal{B} \sim \nu^{k,n}$, its corresponding time-series of persistence landscapes are likely to be close to the sample mean $\bar{P}_m^{k,n}$ in L_p distance for any p. This justifies why classification using any norm yields reasonable outcomes.

Topological interpretations of the mean landscapes In Section 6.1, we acknowledged that the mean persistence landscape is topologically meaningful, while also highlighting its interpretive challenges. This perhaps explains why [18] presented their results in Figure 10a without providing any explicit interpretation. To address this gap, we endeavour to derive some insights in the present study. To this end, we supplement our analysis with reference to the work of [33] for the simulation snapshots depicted in Figure 10b, even though they were generated under slightly different conditions. These snapshots enable us to grasp the pattern of how the microstructure topology evolves, and we establish that the persistence landscapes are consistent with the qualitative understanding of the phase separation mechanism.

As per [33] and illustrated in Figure 10b, the process comprises two stages. Initially, the phases separate, after which the system undergoes further evolution by coarsening the separated phase, wherein domains of different phases grow and merge. Specifically, for $\mu \approx 0$ (i.e., when the two components are approximately equal in proportion), the phase separation primarily follows an intertwined snake-like pattern. In contrast, as μ increases, the morphology increasingly features ellipsoidal and spherical droplets.

This is exactly reflected in the persistence landscapes presented in Figure 10a. Initially, the curves are densely concentrated within a narrow range of filtration levels around the point corresponding to μ after rescaling. This is because the initial conditions are nearly homogeneous at $\mu = 0$. However, as the process evolves, the horizontal range of the landscape generally widens over time, which corresponds to phase separation as the relative concentration difference u moves towards the two extremals -1 and 1. During the coarsening phase (from $0.6T_e$ to T_e), the curve amplitudes decrease, and the majority of them shift towards the right. This phenomenon can be attributed to the fact that there are fewer 1-dimensional holes, and the existing ones are more sparse and born later in filtration.

Furthermore, by comparing the evolution of landscapes for different values of mass, we observe that mass values closer to 0 correspond to curves with a wider horizontal range. This is because the symmetrical, highly intertwined, bicontinuous structures create holes very early on in the filtration. Conversely, for larger mass values, thresholding at the first half of the filtration is dominated by droplet shapes, which only get connected and create loops very late in the filtration.

We can infer that the topological information that we can directly read from the mean landscape is limited, and much less than what is encoded. This is due to two reasons: firstly, a large number of overlapping curves, and secondly, the challenge in interpreting the slope of the curve after averaging. This contrasts with the high interpretability of the persistence intensity, which we will explore in the next Case Study.

Does PH capture geometric information? Based on the experimental results, [18] drew the conclusion that the mass value has a greater impact on topology than previously thought, to the extent that the persistence landscape alone can be used to recover the mass. This assertion is based on the author's belief that PH is a purely topological method and does not capture any size and shape information.

However, this assertion is dubious. While persistent homology is primarily a topological method, it typically captures geometric information through the filtration process. Figure 11 provides a detailed snapshot example that demonstrates a high correlation between the relative concentration difference at each point and its distance to the interfacial boundary formed in the decomposition. Thus, the distance information is implicitly encoded in the filtration. Small and densely populated holes tend to be less persistent than larger and sparser

ones because they are quickly filled during filtration. Moreover, as we have discussed in Section 3.2, geometric information may be incorporated into PH in a more explicit way through specified filtration constructions, such as the signed distance filtration and height filtration.

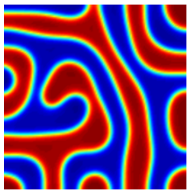


Figure 11: An example snapshot in phase separation.

6.3 Case Study 2: Clustering of histopathology images

The Gleason grading system is the standard approach for evaluating the morphology of prostate cancer. This system assigns cancerous tissue architecture to one of five grades, with increasing severity (refer to Figure 12a). Despite its widespread use, this system is prone to inconsistency due to the subjective nature of pathologists' assessments and neglects to identify significant subpatterns within each grade. Recently, in [35], researchers employed PH to analyze histopathology images, demonstrating its potential as an objective, quantitative, and reliable tool that is consistent with the Gleason grading system while offering more granular analysis.

6.3.1 Methods and results

Image preprocessing and construction of filtration The real-valued image data used in this study was obtained from the intensity of the hematoxylin channel, which is a nuclear-specific colour channel extracted computationally through colour deconvolution from the coloured 2D histological slice (see Figure 12b). To construct the low-star filtration, the process outlined in Section 3.2 was followed. Two sample slices from the filtration are presented in Figure 12c, demonstrating that the filtration initially captures small-scale structures represented by a series of small cycles corresponding to the peaks of the nuclei. As these small components connect and grow across the filtration, larger cycles emerge, which correspond to the two glands. The multi-scale nature of histological architecture makes PH a suitable tool for this type of analysis.

The study presented in [35] focused on histological images associated with Gleason grades 3, 4, and 5. To ensure a balanced dataset, the bootstrapping technique was used to subsample an equal number of images from each class. One hundred bootstrap datasets were generated and used to assess the stability of clustering results across bootstraps.

Vectorization, dimension reduction and clustering To perform clustering, [35] employed a finite-dimensional vectorization known as ranked persistence, as outlined in Section 6.1. This approach involves creating a numerical feature vector comprising the "longest 0D persistence," "second-longest 0D persistence," and so on, up to the "longest 1D persistence," "second-longest 1D persistence," and so forth.

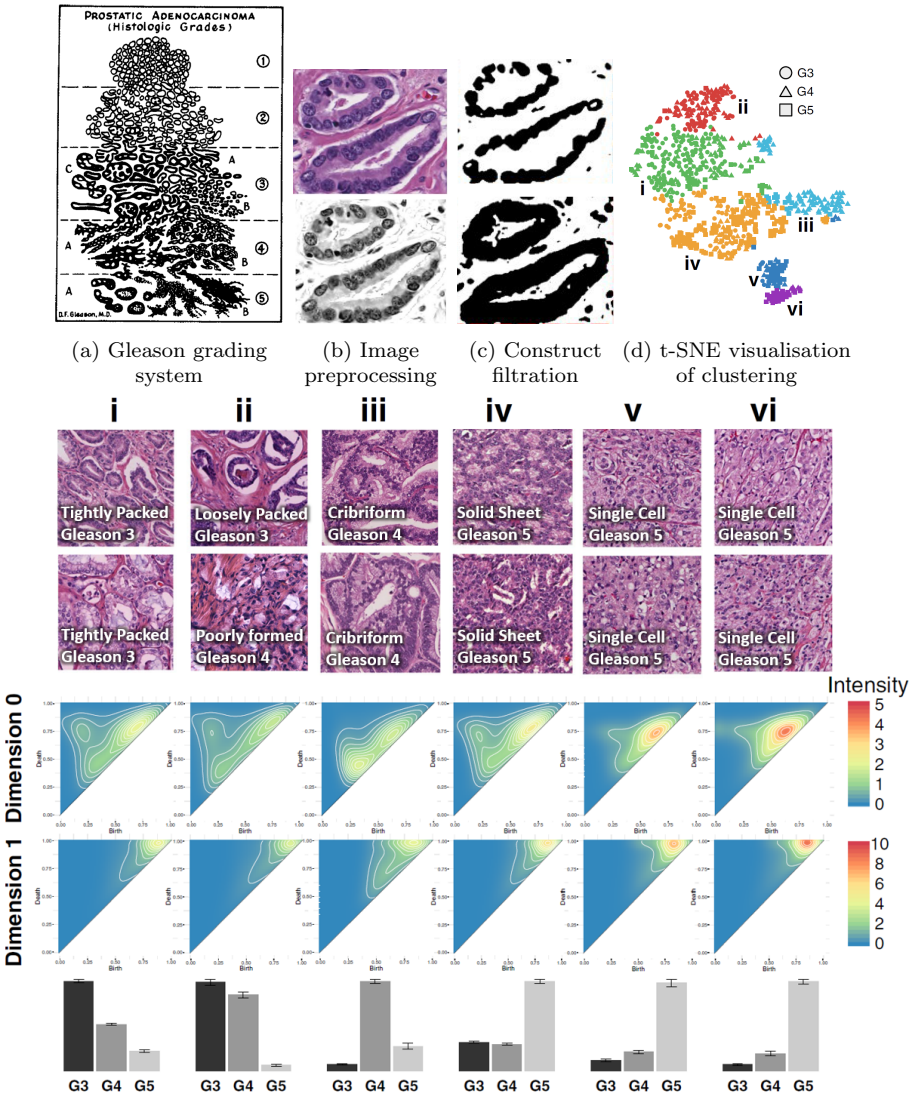
Principal Component Analysis (PCA) was performed on the ranked persistence vectors for dimension reduction. The first six principal components were selected, as they account for at least 99% of the explained variance. Importantly, PCA was conducted without standardization, resulting in a preference for longer persistence automatically. However, unlike rigidly selecting the six longest persistence values, PCA selects linear combinations of the persistence features based on the variabilities present in the dataset, allowing for a data-specific feature selection.

The 6 principal components were then utilized for Hierarchical Ward Clustering. At each step, this method merges a pair of clusters that lead to a minimum increase in total within-cluster variance. The optimal number of clusters $k = 6$ was determined by selecting the value that maximizes the bootstrap mean Gap statistics within the range of $k = 2$ to 10. Finally, t-SNE[26] was used to visualize the clustering result of the 6D data in a 2D space. This technique was designed to preserve the local structure of the data in the lower-dimensional projection. The results are displayed in Figure 12d. Following clustering, another vectorization method, persistence intensity, is utilized. For each cluster, the mean persistence intensity for H_0 and H_1 are computed to provide topological interpretations.

Results The six clusters obtained were arranged in order of increasing aggressiveness. Figure 12e displays two representative slices, the mean persistence intensity, and the composition of Gleason gradings for each cluster. The figure illustrates a continuum of architectural patterns that align with the Gleason grading system. Moreover, PH is capable of capturing subpatterns within and bridging Gleason patterns, thereby providing additional insights beyond the Gleason grading system.

6.3.2 Discussion

[35] provides a detailed discussion of the mean persistence intensity shown in Figure 12e to elucidate the topological and geometric distinctions between clusters and to establish their clinical relevance. In this regard, we highlight some of the principal findings to underscore the remarkable interpretability of the mean persistence intensity. Additionally, as [35] employed vectorizations of ranked persistence and persistence intensity without providing an extensive rationale for their choice, we examine the advantages and disadvantages of this approach relative to alternative methods.



(a) Gleason grading system (b) Image preprocessing (c) Construct filtration (d) t-SNE visualisation of clustering

(e) Interpretation of the clustering results

Figure 12: Clustering of prostate cancer histopathology images (edited from [35])

Topological interpretation of clustering results H_0, which captures the emergence of connected components, provides insights into the density and distribution of nuclei. The first few mean persistence intensities of H_0 exhibit a distinct separation into three regions with high intensity. The lower left region of the persistence intensity corresponds to early-birth, early-death H_0 features, representing densely populated nuclei that soon merge. An example of such features is illustrated in the top diagram of Figure 12c. The upper left region of the persistence intensity corresponds larger components such as glands and clusters of glands with high persistence, as captured by the bottom diagram of Figure 12c. Lastly, the late-birth, late-death H_0 features in the upper right region of the persistence intensity correspond to more sparsely populated regions of stroma, as shown in the light-coloured portions in the image slices. Along the continuum of clusters, we observe the merging of the three dominant components, which corresponds to an increase in the homogeneity of gland distribution. The increasingly higher concentration of late-birth features in clusters v and vi is consistent with the uniformly distributed single cancer cells surrounded by intervening stroma.

H_1, on the other hand, captures the persistence of 1D cycles. Along the continuum of clusters, there is a gradual decrease in the width of the distribution of birth times, which corresponds to a decrease in the variances of the density of one-dimensional cycles. Gleason 3 and 4 patterns consist of well-circumscribed glands, resulting in cycles arising from both densely packed and sparsely distributed nuclei. The narrowing range in Gleason 5 is once again a consequence of the increasing homogeneity as cancer cells invade the surrounding stroma, resulting in single-cell patterns.

The topological interpretation of persistence intensity provides a more detailed insight into the identified subpatterns, offering a higher level of granularity than Gleason grading. For instance, clusters i and ii distinguish tightly packed and loosely packed Gleason 3 patterns. The higher degree of merging of the three dominant H_0 features and the narrower range of H_1 features in the persistence intensity of cluster i, as compared to cluster ii, are consistent with a more tightly packed pattern and an increase in gland distribution homogeneity.

Choices of vectorization The chosen vectorization method for clustering, ranked persistence, only extracts persistence information, resulting in a significant loss of information, including the birth and death times of features. Despite this, the clustering results using ranked persistence are reasonably good. This may be attributed to the fact that persistence barcodes are a rich descriptor of the geometric and topological information in the image, where a small proportion of the information is sufficient for accurate clustering. However, it is possible to use more informative finite-dimensional feature vectors to improve the results. One option is to use persistence images, which are the discretization of persistence intensity. Empirical studies in [1] have shown that persistence images achieve higher accuracy in multiple classification and clustering tasks, indicating their strong discriminative power. Alternatively, one may enrich the

information on ranked persistence by adding other summary statistics. For example, [2] proposed using the mean, standard deviations, and percentiles of birth time, death time, and midpoints, as well as the total number of bars, and an entropy measure to capture the randomness in the barcode. Comparative studies in [2] have shown that using comprehensive summary statistics as features for classifications achieves the highest accuracy among 13 popular vectorizations in all three classification experiments, indicating their superior discriminative power.

Furthermore, in regard to the choice of persistence intensity, we have observed a wealth of topological information that can be directly extracted from the mean persistence intensity. This is because the mean persistence intensity can be viewed as an estimator of the density on the persistence diagram, while also retaining the original readable form of the persistence diagram. In contrast, the persistence landscape, another popular functional summary, was shown to be difficult to interpret in Case Study 1. However, it should be noted that the theoretical guarantees of stability and discriminative power, as well as the statistical theories, of the persistence intensity are not as strong as those of the landscape. Nevertheless, in the context of studying cancer histopathology images where the focus is on results interpretation, the persistence intensity is a justifiably optimal choice.

7 Conclusion

This essay developed the theoretical foundations of PH, exploring its decomposability and computability that are inherited from standard homology theory, and the consequent barcode representation that underpins the success of PH as a practical tool in data analysis. Furthermore, the essay provided a blueprint for applying TDA to image data, consisting of the following steps:

Step 1 Construct a suitable filtration (see Section 3.2 for several choices suited for different purposes).

Step 2 Compute the barcode representation (see Section 5 for computational algorithms and attempts to improve efficiency).

Step 3 Choose one or more suitable vectorization methods and apply statistics and machine learning techniques to the feature vectors (see Section 6 for various choices of vectorizations and their applicability).

To further explore the potential of PH in image analysis, a promising new area of investigation is the incorporation of PH into deep learning-based image processing frameworks. While deep learning models, such as CNNs and transformers, have achieved remarkable success in image classification, segmentation, reconstruction, and generation, they often struggle to capture intricate topological features that require specific long-range interactions between pixels. To overcome this limitation, researchers have investigated integrating PH-based techniques into existing deep-learning models to enhance their topology-based

performance. For instance, PH-based loss functions have been explored to improve image segmentation by making it consistent with ground truth or imposing known topology priors [27, 17]. Additionally, PH has been used to develop layers that can be seamlessly integrated into neural network architectures, allowing the fusion of PH-based methods with deep learning frameworks [32, 11]. However, incorporating PH into neural networks faces significant computational challenges as it requires the repeated computation of PH on a large number of images during the training epochs of the neural network. We hope that progress in the directions outlined in Section 5.2 will alleviate this problem.

References

[1] Henry Adams et al. "Persistence images: A stable vector representation of persistent homology". In: *Journal of Machine Learning Research* 18 (2017).

[2] Dashti Ali et al. "A Survey of Vectorization Methods in Topological Data Analysis". In: 2022.

[3] Paul Bendich et al. "Persistent homology analysis of brain artery trees". In: *The annals of applied statistics* 10.1 (2016), p. 198.

[4] Eric Berry et al. "Functional summaries of persistence diagrams". In: *Journal of Applied and Computational Topology* 4.2 (2020), pp. 211–262.

[5] Jean-Daniel Boissonnat and Clément Maria. "The simplex tree: An efficient data structure for general simplicial complexes". In: *Algorithmica* 70 (2014), pp. 406–427.

[6] Michael M Bronstein et al. "Geometric deep learning: going beyond euclidean data". In: *IEEE Signal Processing Magazine* 34.4 (2017), pp. 18–42.

[7] Peter Bubenik et al. "Statistical topological data analysis using persistence landscapes." In: *J. Mach. Learn. Res.* 16.1 (2015), pp. 77–102.

[8] Peter Bubenik. "The persistence landscape and some of its properties". In: *Topological Data Analysis: The Abel Symposium 2018*. Springer. 2020, pp. 97–117.

[9] Peter Bubenik and Jonathan A Scott. "Categorification of persistent homology". In: *Discrete & Computational Geometry* 51.3 (2014), pp. 600–627.

[10] Mathieu Carrière, Marco Cuturi, and Steve Oudot. "Sliced Wasserstein Kernel for Persistence Diagrams". In: *International Conference on Machine Learning*. 2017.

[11] Mathieu Carrière et al. "Perslay: A neural network layer for persistence diagrams and new graph topological signatures". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 2786–2796.

[12] Frédéric Chazal et al. "Proximity of persistence modules and their diagrams". In: *Proceedings of the twenty-fifth annual symposium on Computational geometry*. 2009, pp. 237–246.

[13] Frédéric Chazal et al. "Subsampling Methods for Persistent Homology". In: *International Conference on Machine Learning*. 2014.

[14] Yen-Chi Chen et al. "Statistical analysis of persistence intensity functions". In: *arXiv preprint arXiv:1510.02502* (2015).

[15] Ilya Chevyrev, Vidit Nanda, and Harald Oberhauser. "Persistence paths and signature features in topological data analysis". In: *IEEE transactions on pattern analysis and machine intelligence* 42.1 (2018), pp. 192–202.

[16] Yu-Min Chung and Austin Lawson. "Persistence curves: A canonical framework for summarizing persistence diagrams". In: *Advances in Computational Mathematics* 48.1 (2022), p. 6.

[17] James R Clough et al. "A topological loss function for deep-learning based image segmentation using persistent homology". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.12 (2020), pp. 8766–8778.

[18] Paweł Dłotko and Thomas Wanner. "Topological microstructure analysis using persistence landscapes". In: *Physica D: Nonlinear Phenomena* 334 (2016), pp. 60–81.

[19] Herbert Edelsbrunner and John Harer. "Computational Topology - an Introduction". In: 2009.

[20] Herbert Edelsbrunner, John Harer, et al. "Persistent homology-a survey". In: *Contemporary mathematics* 453.26 (2008), pp. 257–282.

[21] Brittany Terese Fasy et al. "Confidence sets for persistence diagrams". In: *The Annals of Statistics* (2014), pp. 2301–2339.

[22] Adélie Garin and Guillaume Tauzin. "A topological" reading" lesson: Classification of MNIST using TDA". In: *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*. IEEE. 2019, pp. 1551–1556.

[23] David Günther et al. "Efficient computation of 3D Morse–Smale complexes and persistent homology using discrete Morse theory". In: *The Visual Computer* 28 (2012), pp. 959–969.

[24] Shaun Harker et al. "Discrete Morse Theoretic Algorithms for Computing Homology of Complexes and Maps". In: *Foundations of Computational Mathematics* 14 (2014), pp. 151–184.

[25] A. Hatcher. *Algebraic Topology*. Algebraic Topology. Cambridge University Press, 2002. ISBN: 9780521795401.

[26] Geoffrey E Hinton and Sam Roweis. "Stochastic neighbor embedding". In: *Advances in neural information processing systems* 15 (2002).

[27] Xiaoling Hu et al. "Topology-preserving deep image segmentation". In: *Advances in neural information processing systems* 32 (2019).

[28] Tomasz Kaczynski, Konstantin Mischaikow, and Marian Mrozek. "Computing Homology". In: 2001.

[29] Tomasz Kaczynski, Konstantin Michael Mischaikow, and Marian Mrozek. *Computational homology*. Vol. 3. 7. Springer, 2004.

[30] Tomasz Kaczynski, Marian Mrozek, and Maciej Slusarek. "Homology Computation by Reduction of Chain Complexes". In: *International Journal of Computer Mathematics* (1998).

[31] William D. Kalies, Konstantin Mischaikow, and Greg M. Watson. "Cubical approximation and computation of homology". In: *Banach Center Publications* 47 (1999), pp. 115–131.

[32] Kwangho Kim et al. "Pllay: Efficient topological layer based on persistent landscapes". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 15965–15977.

[33] Björn König, Olivier JJ Ronsin, and Jens Harting. "Two-dimensional Cahn–Hilliard simulations for coarsening kinetics of spinodal decomposition in binary mixtures". In: *Physical Chemistry Chemical Physics* 23.43 (2021), pp. 24823–24833.

[34] Roland Kwitt et al. "Statistical topological data analysis-a kernel perspective". In: *Advances in neural information processing systems* 28 (2015).

[35] Peter Lawson et al. "Persistent homology for the quantitative evaluation of architectural features in prostate cancer histology". In: *Scientific reports* 9.1 (2019), p. 1139.

[36] Michael Lesnick. "The theory of the interleaving distance on multidimensional persistence modules". In: *Foundations of Computational Mathematics* 15.3 (2015), pp. 613–650.

[37] N. G. Makarenkoa, L. M. Karimovab, and M. M. Novakc. "Investigation of global solar magnetic field by computational topology methods". In: 2007.

[38] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[39] James Matuk, Sebastian Kurtek, and Karthik Bharath. "Robust Persistent Homology using Elastic Functional Data Analysis". In: *arXiv preprint arXiv:2106.15436* (2021).

[40] Yuriy Mileyko, Sayan Mukherjee, and John Harer. "Probability measures on the space of persistence diagrams". In: *Inverse Problems* 27.12 (2011), p. 124007.

[41] Konstantin Mischaikow and Vidit Nanda. "Morse theory for filtrations and efficient computation of persistent homology". In: *Discrete & Computational Geometry* 50 (2013), pp. 330–353.

[42] Konstantin Mischaikow et al. "Analysis of blood vessel topology by cubical homology". In: *Proceedings. International Conference on Image Processing* 2 (2002), pp. II–II.

[43] Chul Moon, Qiwei Li, and Guanghua Xiao. "Using Persistent Homology Topological Features to Characterize Medical Images: Case Studies on Lung and Brain Cancers". In: *arXiv preprint arXiv:2012.12102* (2020).

[44] Marian Mrozek and Bogdan Batko. "Coreduction Homology Algorithm". In: *Discrete & Computational Geometry* 41 (2009), pp. 96–118.

[45] Marian Mrozek and Thomas Wanner. "Coreduction homology algorithm for inclusions and persistent homology". In: *Computers & Mathematics with Applications* 60.10 (2010), pp. 2812–2833.

[46] Vidit Nanda. "Computational Algebraic Topology Lecture Notes". 2022-2023.

[47] Vidit Nanda. "Discrete Morse theory for filtrations". PhD thesis. Rutgers University, 2012.

[48] Steve Oudot. "Persistence Theory - From Quiver Representations to Data Analysis". In: *Mathematical Surveys and Monographs*. 2015.

[49] The GUDHI Project. *GUDHI User and Reference Manual*. 3.8.0. GUDHI Editorial Board, 2023. URL: `https://gudhi.inria.fr/doc/3.8.0/`.

[50] Jan Reininghaus et al. "A stable multi-scale kernel for topological machine learning". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2014), pp. 4741–4748.

[51] Bastian Rieck, Filip Sadlo, and Heike Leitte. "Topological machine learning with persistence indicator functions". In: *Topological Methods in Data Analysis and Visualization V: Theory, Algorithms, and Applications 7*. Springer. 2020, pp. 87–101.

[52] Gregory S. Rohrer and Herbert M. Miller. "Topological characteristics of plane sections of polycrystals". In: *Acta Materialia* 58 (2010), pp. 3805–3814.

[53] Vin de Silva and Gunnar E. Carlsson. "Topological estimation using witness complexes". In: *Symposium on Point Based Graphics*. 2004.

[54] Katharine Turner, Sayan Mukherjee, and Doug M Boyer. "Persistent homology transform for modeling shapes and surfaces". In: *Information and Inference: A Journal of the IMA* 3.4 (2014), pp. 310–344.

[55] Katharine Turner et al. "Fréchet means for distributions of persistence diagrams". In: *Discrete & Computational Geometry* 52 (2014), pp. 44–70.

[56] Hubert Wagner, Chao Chen, and Erald Vuçini. "Efficient Computation of Persistent Homology for Cubical Data". In: 2012.

[57] Afra Zomorodian and Gunnar Carlsson. "Computing persistent homology". In: *Proceedings of the twentieth annual symposium on Computational geometry*. 2004, pp. 347–356.