PROJECTED BARCODES FOR MULTI-PERSISTENT HOMOLOGY

A master thesis written by Luca Nyckees, and
supervised by Nicolas Berkouk, Prof. Kathryn Hess

Expert : Mathieu Carrière

Department of Mathematics
Spring semester of 2022

Abstract

Finding stable and computable invariants in multi-parameter persistent homology is an ongoing project for the topological data analysis (TDA) community, as multi-parameter persistence is, from an algebraic viewpoint, more challenging than single-parameter persistence. In this work, we present the class of projected barcodes and integral sheaf metrics, recently introduced by N. Berkouk and F. Petit through the language of sheaf theory. Projected barcodes have been shown to present nice theoretical properties. In particular, they are stable, in some sense, with respect to the interleaving distance and they successfully generalize the fibered barcode construction. We study a specific case and use the recent advances in topological optimization machine learning to propose a way of computing *linear* projected barcodes and *linear* integral sheaf metrics. We show how the projected barcode construction can be integrated in current TDA pipelines and present an application to image classification. The implementation is based on a regularized automatic-differentiation method and, together with experimental notebooks, will be made publicly available at this GitHub account in a new repository called `sheaf_metric`.

Acknowledgments

For his guidance and his advice, for supporting me through this whole learning experience and for simply making topology even more colorful than it already is, my sincerest thanks go to Nicolas Berkouk. I would like to thank Mathieu Carrière for accepting to correct this work, and for having laid down a publicly available `TensorFlow` topological optimization framework. I would also like to thank Léo Lagast for the precious feedback on my report, and Silvio, Léa and Maria for listening to me mumbling and rumbling about this mysterious «magic metric» for the whole semester. Finally, I would like to thank Prof. Kathryn Hess Bellwald for supervising this project and making all of this possible.

Contents

1 Introduction

Topological data analysis emerged, as a field of mathematics, as a powerful tool to study the shape of data. It has found many applications in domains like computational neuroscience, machine learning and statistical data analysis ([5]). One of the most important concepts in this field is *persistent homology*, which allows to compute stable topological invariants for point clouds data and other spaces. The mathematical setting of persistent homology is very well understood. There is a natural generalization to higher dimensions, called *multi-parameter persistent homology* ([6]), that finds various applications in topological machine learning (see for example [7] by M. Carrière and A. Blumberg). It provides a more complex challenge than standard persistence, in the sense that invariants with nice properties are harder to find and to compute.

A central task in multi-parameter persistent homology is to establish meaningful metrics to distinguish between and classify topological spaces. As opposed to single-parameter persistent homology, there is no *barcode* or *persistence diagram* describing a function $f : X \to \mathbb{R}^d$ with $d \leq 2$. Just as in the one-dimensional case, one can look at the *interleaving distance*. However, it higher dimension, this metric becomes computationally too expensive. Indeed, it has been shown to be **NP**-hard to compute [2]. The quest for other invariants and metrics has led to defining the notion of *fibered barcode* (which has an equivalent invariant called *rank invariant*) for multi-parameter persistence modules. The fibered barcode naturally induces a metric for multi-parameter persistence modules called the *matching distance*. The fibered barcode has been used in various applications in machine learning, and its algorithmic aspects have been widely studied [17]. However, there are topological features that it still fails to capture. In fact, one can find multi-parameter persistence modules that have an arbitrarily large interleaving distance, yet a zero matching distance. Moreover, M. Lesnick and M. Wright show, in [20], that one can compute a matching distance for sublevel set homology of functions $f : K_1 \to \mathbb{R}^2$, $g : K_2 \to \mathbb{R}^2$ in complexity $\mathcal{O}(|K|^{11})$.

Aiming at providing a new invariant that tackles the issues encountered in the theory of multi-parameter persistence, Nicolas Berkouk and François Petit use the language of sheaf theory to introduce a new class of invariants called *projected barcodes*, together with a class of distances called *integral sheaf metrics* [1]. The authors show that projected barcodes enjoy nice theoretical properties. In particular, they are, in some sense, stable with respect to the interleaving distance. Moreover, they successfully generalize the fibered barcode construction. Roughly speaking, the idea behind projected barcodes is the following. Both the fibered barcode and a projected barcode consist of a collection of barcodes of single-parameter persistence modules. Instead of looking at inverse images of embeddings $i_{\mathcal{L}} : \mathbb{R} \hookrightarrow \mathbb{R}^d$, as in the case of the

fibered barcode, one can look at direct images of projections $p : \mathbb{R}^d \to \mathbb{R}$ and obtain a broader family of barcodes. In this work, we aim at introducing the projected barcodes for multi-parameter persistence modules and investigate the challenges and algorithmic aspects of an implementation.

This work is structured as follows. Section 2 lays out the aspects of single parameter persistence whose generalization to multi-parameter persistence is part of this work. More precisely, subsections 2.1 and 2.2 recall the fundamentals of persistent homology, including main results of decompositions and metric stability. Subsection 2.3 focuses on topological optimization, which is the framework necessary the compute a specific type of integral sheaf metrics called the *linear integral sheaf metric*. In section 3.1, we recall the main concepts of multi-parameter persistent homology and highlight core computational invariants, namely the matching distance and the rank invariant. Section 4 introduces the class of projected barcodes and integral sheaf metrics in the language of multi-parameter persistence. In subsection 4.3, we present the computational framework used for the implementation. In section 5, we present applications to image analysis. More precisely, we integrate the LISM signature a standard TDA pipeline and study its behavior for MNIST classification.

Notation. When unspecified, X and Y shall denote topological spaces and K, K_1 and K_2 simplicial complexes. Moreover, we write $H_j(\cdot)$ for the singular homology functor over a fixed field $\mathbf{k}$, *i.e.* $H_j(\cdot) = H_j^{\text{sing}}(\cdot, \mathbf{k})$. Any inclusion of spaces $X \hookrightarrow Y$ induces a linear map of vector spaces $H_j(X) \to H_j(Y)$. We endow $\mathbb{R}^d$ with the standard euclidean topology and the poset structure given by $a \preceq b$ if and only if $a_i \leq b_i$ for all $i \in \{1, ..., d\}$, where $a = (a_1, ..., a_d)$ and $b = (b_1, ..., b_d)$ are two points in $\mathbb{R}^d$. Moreover, we will use the following conventions and notation.

1. $\mathbb{R}^d$ is the category induced by the poset $(\mathbb{R}^d, \preceq)$,
2. $\mathbf{Vect}_{\mathbf{k}}$ is the category of $\mathbf{k}$-vector spaces with linear maps.
3. Let $p \in \mathbb{N}_{>0}$. The Euclidean space $\mathbb{R}^d$ can be endowed with the p-norm defined with $\|x\|_p = (\sum_{i=1}^d x_i)^{\frac{1}{p}}$ for any vector $x = (x_1, ..., x_d) \in \mathbb{R}^d$.
4. Moreover, the Euclidean space $\mathbb{R}^d$ can also be endowed with the ∞-norm defined with $\|x\|_\infty = \max_{i=1,...,d} |x_i|$ for any vector $x = (x_1, ..., x_d) \in \mathbb{R}^d$.
5. We write $\mathbb{R}_+ = \{x \in \mathbb{R} \mid x > 0\}$.

2 Persistent Homology

We recall the fundamentals of 1-parameter persistent homology. One may read [13] by H. Edelsbrunner and J. Harer for a very nice survey about persistent homology.

2.1 Persistence modules

Definition 2.1. *A* persistence module *over* $\mathbb{R}$ *is a functor* $M : \mathbb{R} \to \mathbf{Vect_k}$.

One often writes $M_a = M(a)$ for $a \in \mathbb{R}$ and a persistence module M.

Definition 2.2. *A* filtration *over* $\mathbb{R}$ *is a functor* $F : (\mathbb{R}, \leq) \to (\mathbf{Top}, \subseteq)$.

The *sublevel set filtration* of a function $f : X \to \mathbb{R}$ is defined as the functor

$$\mathbb{R} \to \mathbf{Vect_k} : a \to \{x \in X \mid f(x) \leq a\}.$$

It gives rise to the class of *sublevel set persistence modules*.

Definition 2.3. *Let* $f : X \to \mathbb{R}^d$ *be a continuous function. The persistence module* $S_j(f) : \mathbb{R} \to \mathbf{Vect_k}$ *with*

$$S_j(f)(a) = H_j(\{x \in X \mid f(x) \leq a\})$$

is the j-th sublevel set persistence module of f.

Definition 2.4. *An* interval module *over* $\mathbb{R}$ *is a persistence module of the form*

$$I_{(b,d)}(t) = \begin{cases} \mathbf{k} \text{ if } b \leq t \leq d \\ 0 \text{ otherwise} \end{cases} \quad \text{with } I_{(b,d)}(s \leq t) = \begin{cases} \mathrm{id}_{\mathbf{k}} \text{ if } b \leq s, t \leq d \\ 0 \text{ otherwise} \end{cases}.$$

Consider a persistence module $M : \mathbb{R} \to \mathbf{Vect_k}$.

Theorem 2.5 (Existence of decomposition)**.** *Provided that either all vector spaces* $M_a \in \mathbf{Vect_k}$ *are finite-dimensional or the support* $\{a \in \mathbb{R} \mid M_a \neq 0\} \subseteq \mathbb{R}$ *is finite, the persistence module* M *decomposes as a direct sum of interval modules.*

Theorem 2.6 (Unicity of decomposition)**.** *A direct sum decomposition of a persistence module into interval modules is unique, up to permutation of the summands.*

Persistence diagram and barcode. Consider an interval module decomposition $M \cong \bigoplus_{i=1}^{n} I_{(b_i,d_i)}$. Without loss of generality, one can shift all coordinates so that $(b_i, d_i) \in \mathbb{R}^2_+$ for all $i = 1, ..., n$. The *persistence diagram* of M is defined as the subset of $\mathbb{R}^2_+$ consisting of the half-diagonal $\Lambda = \{(x, y) \in \mathbb{R}^2_+ \mid x = y\}$ together with points (b_i, d_i). A topological descriptor equivalent to the persistence diagram is the so-called *barcode*. The barcode of a persistence module M, denoted $\mathcal{B}(M)$, is given by the collection of intervals $\{[b_i, d_i]\}_{i=1}^{n}$. The j-th persistence diagram (resp. j-th barcode) associated to a function $f : X \to \mathbb{R}$, denoted by $\mathrm{Pers}_j(f)$ (resp. $\mathcal{B}_j(f)$) is simply defined as the persistence diagram (resp. barcode) of the persistence module $S_j(f)$. An example is given in Figure 1, From now on, persistence diagrams and barcodes will both be denoted by $\mathcal{B}_j(\cdot)$ and we will consider them as the same object.

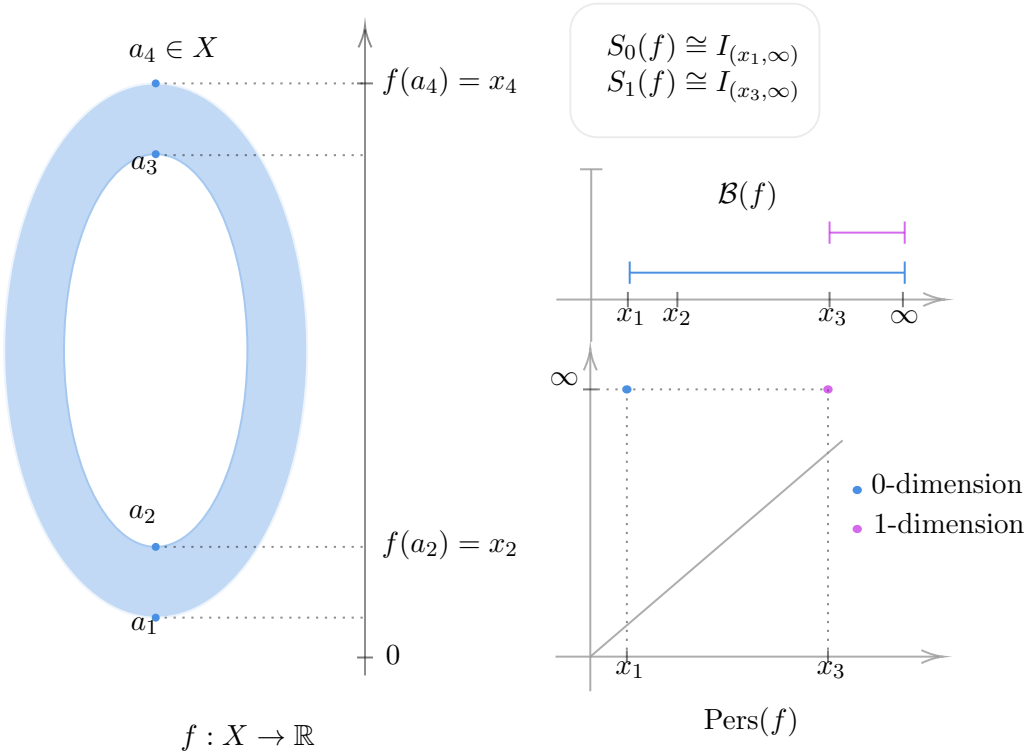


Figure 1: Persistence diagram and barcode of a sublevel set homology of a filtration induced by a function $f : X \to \mathbb{R}$ where $X \subset \mathbb{R}^2$ is an annulus.

Remark 2.1. Combining the two results above, one gets a well-defined map assigning to a function on a finite simplicial complex its persistence diagram.

There exists a more general condition on a persistence module M (or a function $f : X \to \mathbb{R}$) to ensure the existence of its persistence diagram, called *q-tameness*. In particular, this condition does not assume that M is interval-decomposable.

Definition 2.7. *A persistence module M (resp. function $f : X \to \mathbb{R}$) is said to be* q-tame *if* $\operatorname{rank}(M(s \leq t)) < \infty$ *(resp.* $\operatorname{rank}(S_j(f)(s \leq t)) < \infty$*) for all* $s < t \in \mathbb{R}$.

Theorem 2.8. *A q-tame persistence module admits a persistence diagram.*

2.2 Metrics for persistence diagrams

Definition 2.9. *A matching between two persistence diagrams D_1 and D_2 is a subset $m \in D_1 \times D_2$ such that any point in $(D_1 \setminus \Lambda) \cup (D_2 \setminus \Lambda)$ appears exactly once in m.*

We denote by $\mathcal{M}_{D_1}^{D_2}$ the set of matchings between persistence diagrams D_1 and D_2.

Definition 2.10 (2-Wasserstein distance)**.** *The* p-Wasserstein distance *between two persistence diagrams D_1 and D_2 is given by*

$$d_{W_p}(D_1, D_2)^p = \inf_{m \in \mathcal{M}_{D_1}^{D_2}} \sum_{(p,q) \in m} \|p - q\|_\infty^p.$$

When $p = 2$, one omits the index and writes d_W.

Definition 2.11 (Bottleneck distance)**.** *The* Bottleneck distance *between two persistence diagrams D_1 and D_2 is given by*

$$d_B(D_1, D_2) = \inf_{m \in \mathcal{M}_{D_1}^{D_2}} \max_{(p,q) \in m} \|p - q\|_\infty.$$

One can see a persistence module $M : \mathbb{R} \to \mathbf{Vect_k}$ as a collection $\{M_a\}_{a \in \mathbb{R}}$.

Definition 2.12. *An ε-homomorphism $\Phi : M \to N$ between two persistence modules M and N is given by a collection of linear maps $\{\varphi_a : M_a \to N_{a+\varepsilon}\}_{a \in \mathbb{R}}$ that satisfies the commuting condition $\varphi_b \circ M(a \leq b) = N(a + \varepsilon \leq b + \varepsilon) \circ \varphi_a$ for all $a \leq b \in \mathbb{R}$.*

Any persistence module M has an ε-homomorphism called *translation endofunctor*, given by $\varphi_\varepsilon^M : M \to M$ given by the collection of linear maps $\{M(a \leq a + \varepsilon)\}_{a \in \mathbb{R}}$.

Definition 2.13 (ε-interleaving)**.** *Two persistence modules M, N are* ε-interleaved *if there are ε-homomorphisms $f : M \to N$ and $g : N \to M$ satisfying*

$$\begin{cases} g \circ f = \varphi_{2\varepsilon}^M \\ f \circ g = \varphi_{2\varepsilon}^N \end{cases}.$$

Theorem 2.14 (Isometry). *For two q-tame persistence modules M and N, one has*

$$d_B(\mathcal{B}(M), \mathcal{B}(N))) = d_I(M, N).$$

Theorem 2.15 (Stability). *For two q-tame functions $f, g : X \to \mathbb{R}$, one has*

$$d_B(\mathcal{B}_j(f), \mathcal{B}_j(g)) \leq \|f - g\|_\infty$$

for all $j \in \mathbb{N}$.

2.3 Topological optimization

A new approach in topological machine learning is to optimize functions based on persistent homology. However, those can present a highly non-convex behavior. In [8], the authors introduce a general setting, through o-minimal geometry, in which one can successfully compute gradients of such functions and obtain convergence results for standard stochastic gradient algorithms.

2.3.1 Optimizing topological functions

Notation. Let $\mathfrak{S}_n$ denote the set of permutations on n elements.

Definition 2.16 (Parametrized family of filtrations). *Let A be a set. An A-parametrized family of filtrations on K is a map $\Phi : A \to \mathbb{R}^{|K|}$ such that for any two simplices $\sigma_1, \sigma_2 \in K$, and any element $p \in A$, one has $\Phi_{\sigma_1}(p) \leq \Phi_{\sigma_2}(p)$.*

Sublevel sets. Consider a family of functions $\{f_p : K^0 \to \mathbb{R}\}_{p \in A}$ defined on the 0-skeleton (vertices) of K. The associated *sublevel sets filtrations* are the filtrations

$$\Phi(p) : K \to \mathbb{R} : \sigma \mapsto \max_{v \in \sigma} f_p(v)$$

for $p \in A$. This defines an A-parametrized family of filtrations on K.

Definition 2.17 (Function of persistence). *A map $E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R}$ is said to be a function of persistence if, for any permutations $\tau_x \in \mathfrak{S}_{2p}$ and $\tau_y \in \mathfrak{S}_q$, one has*

$$E(x_1, ..., x_{2p}, y_1, ..., y_q) = E(\tau_x(X), ..., \tau_x(x_{2p}), \tau_y(y_1), ..., \tau_y(y_q)).$$

For instance, the following functions are functions of persistence.

1. $E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R} : D \mapsto d_B(D, D')$ (with D' fixed),
2. $E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R} : D \mapsto d_W(D, D')$ (with D' fixed),
3. $E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R} : D \mapsto \sum_{i=1}^p |b_i - d_i|$, where $D = (b_i, d_i)_{i=1}^p \in \mathbb{R}^{2p}$.

Note that, in Definition 2.16, a filtration $K \to \mathbb{R}$ is considered as an element of $\mathbb{R}^{|K|}$. Indeed, an element $p \in A$ gives rise to a well-defined filtration $\Phi(p)$ defined as

$$\Phi(p) : K \to \mathbb{R} : \sigma \mapsto \Phi_\sigma(p).$$

Now, consider a parametrized family of filtrations $\Phi : A \to \mathbb{R}^{|K|}$. Oftentimes, one is interested in maximizing or minimizing the functional

$$\mathcal{L} : A \to \mathbb{R} : p \mapsto E \circ \mathrm{Pers} \circ \Phi(p),$$

where $E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R}$ a function of persistence. We follow the ideas of [8] to formulate a convergence result for standard subgradient descent algorithms, but without the setting of o-minimal geometry.

Definition 2.18 (Lipschitz function)**.** *Consider two metric spaces (X, d_X) and (Y, d_y). A function $f : X \to Y$ is* Lipschitz *if there exists a constant $K > 0$ such that, for all $x_1, x_2 \in X$, one has*

$$d_Y(f(x_1), f(x_2)) \leq K \cdot d_X(x_1, x_2).$$

Definition 2.19 (Locally Lipschitz function)**.** *Consider two metric spaces (X, d_X) and (Y, d_y). A function $f : X \to Y$ is* locally Lipschitz *if for every point $x \in X$, there exists an open set $U_x \subseteq X$ such that $f|_{U_x} : U_x \to Y$ is Lipschitz.*

The following is an important result in differential geometry about the differentiability of locally Lipschitz maps. We refer to Section 3.1.2 of [14] for a proof.

Theorem 2.20 (Rademacher)**.** *Let $U \subseteq \mathbb{R}^n$ be an open set, and consider a Lipschitz function $f : U \to \mathbb{R}$. Then the function f is almost everywhere differentiable on U. More precisely, the subset of points in U where f is not differentiable form a set of Lebesgue measure zero.*

In particular, a locally Lipschitz map $f : \mathbb{R}^n \to \mathbb{R}$ is almost everywhere differentiable.

Notation. Consider a family of vectors $\{x_i\}_i \subset \mathbb{R}^n$. One denotes by $\mathrm{Conv}(\{x_i\}_i)$ the convex hull formed by the vectors x_i. More precisely, one defines

$$\mathrm{Conv}(\{x_i\}_i) = \left\{ x \in \mathbb{R}^n \mid \sum_i \lambda_i x_i = x,\ 0 \leq \lambda_i \leq 1,\ \text{and}\ \sum_i \lambda_i = 1 \right\}.$$

Definition 2.21 (Clarke subdifferential)**.** *The* Clarke subdifferential *∂f of an almost everywhere differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is defined by setting, for any $x \in \mathbb{R}^n$,*

$$\partial f(z) = \mathrm{Conv}(\{ \lim_{z_i \to z} \nabla f(z_i) \mid f \text{ is differentiable at } z_i \}).$$

Now, provided that $\mathcal{L}$ is locally Lipschitz, the results above assure that $\mathcal{L}$ has a well-defined Clarke subdifferential $\partial\mathcal{L}$. Hence, one can use a stochastic gradient subgradient algorithm based on iterations

$$x_{k+1} = x_k - \alpha_k(y_k + \zeta_k), \tag{1}$$

where $(\alpha_k)_k$ is a changing learning rate, $(\zeta_k)_k$ are random variables, and $y_k \in \partial\mathcal{L}(x_k)$.

Assumption A. (Assumptions for the stochastic gradient descent)

1. $\alpha_k > 0$ for all $k \in \mathbb{N}$, $\sum_{k\in\mathbb{N}} \alpha_k = \infty$, and $\sum_{k\in\mathbb{N}} \alpha_k^2 < \infty$.
2. Almost surely, one has $\sup_{k\geq 1} \|x_k\| < \infty$.
3. Let $\mathcal{F}_k$ denote the increasing sequence of σ-algebras $\sigma(x_j, y_j, \zeta_j \mid j \leq k)$. Then there exists a function $p : \mathbb{R}^d \to \mathbb{R}_+ \cup \{0\}$, bounded on bounded sets, such that almost surely, for all $k \in \mathbb{N}$, one has $\mathbb{E}[\zeta_k \mid \mathcal{F}_k] = 0$ and $\mathbb{E}[\|\zeta_k\|^2 \mid \mathcal{F}_k] < p(x_k)$.

Definition 2.22 (Locally finite partition)**.** *A partition $X = \bigcup_{i\in I} X_i$ of a non-empty set $X \subseteq \mathbb{R}^m$ endowed with the subspace topology is* locally finite *if, for any $x \in X$, there exists a open neighborhood $U \subseteq X$ with*

$$|\{j \in I \mid X_j \cap U \neq \emptyset\}| < \infty.$$

Definition 2.23 (C^p-stratification)**.** *A* C^p-stratification *$\{X_i\}_{i\in I}$ of a non-empty set $X \subseteq \mathbb{R}^m$ is a locally finite a partition $X = \bigcup_{i\in I} X_i$ where the* strata *X_i are connected submanifolds of $\mathbb{R}^m$ of class C^p and such that, for any indices $i \neq j \in I$,*

$$\bar{X}_i \cap X_j \neq \emptyset \Rightarrow X_j \subseteq \bar{X}_i \setminus X_i.$$

Definition 2.24 (Whitney stratifiable set)**.** *Let $\mathcal{X} = \{X_i\}_{i\in I}$ be a C^p-stratification of a non-empty set $X \subseteq \mathbb{R}^m$. One says $\mathcal{X}$ is* Whitney*, if, for each $x \in \bar{X}_i \cap X_j$ with $i \neq j$, and for each sequence $(x_k)_k \subset X_i$ converging to x such that the sequence of tangent spaces $(T_{x_k}X_i)_k$ converges to a subspace $V \subset T_x\mathbb{R}^m$, one has $T_xX_j \subset V$.*

If X admits a Whitney C^p-stratification, it is called *Whitney C^p-stratifiable.*

Notation. Recall that the *graph* of a function $f : \mathbb{R}^n \to \mathbb{R}^k$ is defined as

$$\Gamma(f) = \{(x, f(x)) \mid x \in \mathbb{R}^n\} \subset \mathbb{R}^{n+k}.$$

Definition 2.25 (Whitney stratifiable function)**.** *A function $f : \mathbb{R}^n \to \mathbb{R}^k$ is* Whitney C^p-stratifiable *if its graph $\Gamma(f) \subset \mathbb{R}^{n+k}$ is Whitney C^p-stratifiable.*

A *critical point* of an almost everywhere differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is a local minima or maxima of f. One denotes the set of critical points of f as $\mathrm{Crit}(f)$.

Theorem 2.26. *([11, Corollary 5.9]) Let $\mathcal{L} : \mathbb{R}^n \to \mathbb{R}$ be a locally Lipschitz function that is Whitney C^d-stratifiable, and consider the subgradient algorithm with iterations 1. Provided that Assumption A is satisfied, almost surely, one has*

$$(x_k)_k \to \lim_{k\to\infty} x_k \in \mathrm{Crit}(\mathcal{L}),$$

and $(\mathcal{L}(x_k))_k$ converges.

In [8], the authors use this strategy, together with the setting of o-minimal geometry, to ensure the convergence (Theorem 4.2) of the subgradient algorithm for the case of functions based on persistent homology, such as the map $\mathcal{L} = E \circ \mathrm{Pers} \circ \Phi(\cdot)$, where

$$E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R} : D \mapsto d_B(D, D'),$$

and Φ is an A-parametrized family of sublevel sets filtrations.

2.3.2 Gradients of topological functions

In this subsection, we recall how one can compute gradients of persistence-based topological functions, following the approach of M. Carrière, F. Chazal, M. Glisse, Y. Ike in [8] and T. Dey and Y. Wang in [12]. Computing gradients of such functions has opened new perspectives for topological machine learning, and plays a fundamental role in computing linear integral sheaf metrics (see Section 4).

Definition 2.27 (Abstract simplicial complex)**.** *Let V be a set. An abstract simplicial complex over V is a set K of subsets of V such that*

- *$\{v\} \in K$ for every $v \in V$,*
- *if $\sigma \in K$ and $\sigma' \subseteq \sigma$, then $\sigma' \in K$.*

Definition 2.28 (k-simplex)**.** *A k-simplex σ is defined as the convex hull of a set $\{x_1, ..., x_{k+1}\} \subset \mathbb{R}^n$, where the points $x_1, ..., x_{k+1}$ are affinely independent. Symbolically, one has*

$$\sigma = \mathrm{Conv}(\{x_1, ..., x_{k+1}\}) = \left\{ x \in \mathbb{R}^d \mid x = \sum_{i=1}^{k+1} \alpha_i x_i,\ 0 \leq \alpha_i \leq 1,\ \sum_{i=1}^{k+1} \alpha_i = 1 \right\}.$$

For a k-simplex σ, one can write $V(\sigma)$ for the set of vertices of σ. Now, for a vertex subset $V' \subseteq V(\sigma)$, one says $\mathrm{Conv}(V')$ is a *face of* σ. A k-simplex σ is a *simplex* of *dimension* $\dim(\sigma) = k$.

Definition 2.29 (Geometric simplicial complex)**.** *A geometric simplicial complex is a finite collection of simplices K such that*

- *if $\sigma \in K$ and $\sigma' \subseteq \sigma$, then $\sigma' \in K$,*
- *for any two $\sigma, \sigma' \in K$, either $\sigma \cap \sigma' = \emptyset$ or $\sigma \cap \sigma'$ is a face of both σ and σ'.*

The *dimension* of a geometric complex K is $\dim(K) = \max_{\sigma \in K} \dim(\sigma)$.

Definition 2.30 (Geometric realization)**.** *Let K be an abstract simplicial complex. A geometric realization of K is a geometric simplicial complex K' if there is exists a bijection $\varphi : K \to K'$ such that $\sigma \in K$ if and only if $\varphi(\sigma) \in K'$.*

One often denotes by $|K|$ the geometric realization of a finite abstract simplicial complex K built as follows. Suppose K has m vertices. Then, for each vertex $v_i \in K$, define $\varphi(v_i) = x_i = (0, ..., 0, 1, 0, ...0) = e_i \in \mathbb{R}^m$. Then, for any simplex $\sigma \in K$, define $\varphi(\sigma) = \text{Conv}(\{\varphi(v) \in \mathbb{R}^n \mid v \in \sigma\})$. This defines a mapping $\varphi : K \to \text{Im}(\varphi) =: |K|$.

Notation. One denotes by K^i the set of i-simplices of a geometric simplicial complex K. It is called the *i-skeleton of K* (K has vertex set K^0). Moreover,

- For K a geometric simplicial complex, one denotes by $|K|$ the underlying space consisting of the union of all the simplices in K. More precisely, if K has dimension n, then $|K|$ is the subspace of $\mathbb{R}^n$ consisting of the union of all simplices $\sigma \in K$ embedded in $\mathbb{R}^n$, and endowed with the subspace topology.
- For K a finite abstract simplicial complex, one denotes by $|K|$ its geometric realization. From now on, we will abuse notation by writing $|K|$ instead of $||K||$, which is the underlying space of the geometric realization of K.

Figure 2 shows the geometric realization of an abstract simplicial complex.

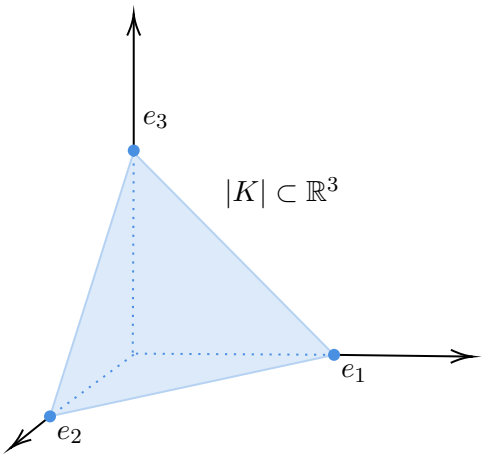
Definition 2.31 (Barycentric coordinates)**.** *Consider a geometric simplicial complex K, and a function $f : K^0 \to \mathbb{R}^n$. Any $x \in |K|$ belongs to the interior of exactly one simplex $\sigma_x = \text{Conv}(\{v_i \mid i \in I_{\sigma_x}\}) \in K$. The barycentric coordinates of x are the unique points $b_i(x)$ with*

$$x = \sum_{i \in I_{\sigma_x}} b_i(x) v_i.$$

Given a finite abstract simplicial complex K and a function $f : K^0 \to \mathbb{R}^n$, one extend f continuously to a function $f_{\text{ext}} : |K| \to \mathbb{R}^n$ with

$$f_{\text{ext}}(x) = \sum_{i \in I_{\sigma_x}}^{n} b_i(x) f(v_i),$$

where $x \in \text{int}(\text{Conv}(\{v_i \mid i \in I_{\sigma_x}\}))$ and $b_i(x)$ are the barycentric coordinates of x.



$$K = \{\{1\},\{2\},\{3\},\{1,2\},\{2,3\},\{1,3\},\{1,2,3\}\}$$

Figure 2: Geometric realization (an equilateral triangle embedded in $\mathbb{R}^3$) of the abstract simplicial complex $K = \{\{1\},\{2\},\{3\},\{1,2\},\{2,3\},\{1,3\},\{1,2,3\}\}$.

Definition 2.32 (Piecewise linear function)**.** *Let K be a geometric simplicial complex with embedded in $\mathbb{R}^n$. A function $f : |K| \to \mathbb{R}^m$ is* piecewise linear (PL) *if, for every simplex $\sigma \in K$, the restriction $f|_\sigma$ is the restriction of an affine map $\mathbb{R}^n \to \mathbb{R}^m$.*

Proposition 2.33. *Let K be a finite abstract simplicial complex, and consider a function $f : K^0 \to \mathbb{R}^m$. The function $f_{\text{ext}} : |K| \to \mathbb{R}^m$ is a piecewise linear.*

Proof. Let $\sigma = \{v_1, ..., v_k\} \in |K|$. We have to show that the restriction of f_{ext} to σ is the restriction of a linear map $\mathbb{R}^n \to \mathbb{R}^m$. Any point $x \in \sigma$ can be written as $x = \sum_{i=1}^k b_i v_i$ and thus, in the basis of barycentric coordinates defined on σ, one has $x = (b_1, ..., b_k)$. By definition, $f_{\text{ext}}|_\sigma$ is a linear map. ■

Notation. Let $\mathfrak{F} \subset \mathcal{C}^0(\mathbb{R}^d)$, and consider piecewise linear functions $f : |X| \to \mathbb{R}^d$ and $f : |Y| \to \mathbb{R}^d$, where X and Y are abstract simplicial complexes. One obtains two parametrized family of continuous functions $\{f_p\}_{p\in\mathfrak{F}}$ and $\{g_p\}_{p\in\mathfrak{F}}$ defined by taking $f_p := p \circ f$ and $g_p := p \circ g$, for each element $p \in \mathfrak{F}$.

For $p \in \mathfrak{F}$, define $\mathcal{B}_j^1(p) := \mathcal{B}_j(f_p)$ and $\mathcal{B}_j^2(p) := \mathcal{B}_j(g_p)$. Let $\mathcal{D}$ denote the set of barcodes $\mu : \mathbb{Z} \times \mathbb{R}^d \to \mathbb{N}$ (or persistence diagrams $D \subset \mathbb{R}^{2d}$). Let $E_1^j \subset \mathbb{R}$ and $E_2^j \subset \mathbb{R}$ be the multisets of coordinates (endpoints) of intervals appearing in the barcodes $\mathcal{B}_j^1(p)$ and $\mathcal{B}_j^2(p)$ respectively.

The main trick. ([21]) Suppose $f_p : |X| \to \mathbb{R}$ and $g_p : |Y| \to \mathbb{R}$ are piecewise-linear and injective. Then there is a map $\rho_1^p : E_1^j \to X$ (resp. $\rho_2^p : E_2^j \to Y$) mapping each

birth and death points in E_1^j (resp. E_2^j) to the simplex of X (resp. Y) generating or annihilating the corresponding interval. Moreover, the inverse functions ρ_1^p and ρ_2^p are, in some sense, locally constant with respect to $p \in \mathfrak{F}$.

Definition 2.34. *Consider a PL function $f : |X| \to \mathbb{R}$. For $\varepsilon > 0$, define the ball*

$$B(f, \varepsilon) = \{g : |X| \to \mathbb{R} \mid \|f - g\|_\infty < \varepsilon\}.$$

Let $\varepsilon > 0$ be the minimum difference between two values of f of simplices of X. In other words, $\varepsilon = \min\{|f(\sigma_1) - f(\sigma_2)| \mid \sigma_1 \neq \sigma_2 \in X\}$. For the next definition, recall that a persistence diagram can be seen as a map $(X, f) \mapsto \{(b_i, d_i)\}_{i \in I}$.

Definition 2.35. *Consider a PL function $f : |X| \to \mathbb{R}$. For $\delta > 0$, define the ball*

$$B(\mathcal{B}(f), \delta) = \{\mathcal{B}(g) \mid g \in B(f, \delta)\}.$$

Consider the map $\eta_f : X \to X^0$ defined by $\eta_f(\sigma) = \operatorname{argmin}_{\nu \in \sigma} f(\nu)$ for $\sigma \in X$, where X^0 is the set of vertices of X. Naturally, for any vertex $\sigma \in X$, one has $\eta_f(\sigma) = f(\sigma)$. Now, note that for any filtration function on a finite simplicial complex, one can obtain a total ordering on the simplices (one can consider using the lexicographic order). The total ordering provides a mapping

$$\zeta_f = (\zeta_f^1, \zeta_f^2) : \mathcal{B}(f) \to X \times X$$

assigning a pair of simplices to each pair of birth-death coordinates (b_i, d_i).

Definition 2.36. *For a PL-function $f : |X| \to \mathbb{R}$, one defines the* inverse mapping

$$\pi_f = (\pi_f^1, \pi_f^2) = (\eta_f \circ \zeta_f^1, \eta_f \circ \zeta_f^2).$$

We will now show that $\pi : f \mapsto \pi_f$ is locally constant.

Lemma 2.37. *There is an $r > 0$ such that $\eta_f = \eta_g$ for any $g \in B(f, r)$.*

Proof. Let $\varepsilon = \min\{|f(\sigma_1) - f(\sigma_2)| \mid \sigma_1 \neq \sigma_2 \in X\}$ defined as above. Clearly, one has $\eta_f = \eta_g$ for any $g \in B(f, \frac{\varepsilon}{2})$ and thus taking $r = \varepsilon/2$ yields the claim. ∎

Lemma 2.38. *There exists an $r > 0$ such that $\zeta_f = \zeta_g$ for any $g \in B(f, r)$.*

Proof. This follows from the stability inequality $d_B(\mathcal{B}(f), \mathcal{B}(g)) \leq \|f - g\|_\infty$ and the fact that ζ_f varies continuously with $\mathcal{B}(f)$ (with the Bottleneck distance). ∎

Proposition 2.39. *There exists an $r > 0$ such that $\pi_f = \pi_g$ for any $g \in B(f, r)$.*

Proof. This immediately follows from Lemma 2.37 and Lemma 2.38. ∎

Notation. To go back to the notations of the main trick above, one sets

$$\begin{aligned}\rho_1^p(b_i) &:= \pi^1_{f_p}(b_i, d_i),\\ \rho_1^p(d_i) &:= \pi^2_{f_p}(b_i, d_i),\\ \rho_2^p(b_i) &:= \pi^1_{g_p}(b_i, d_i),\\ \rho_2^p(d_i) &:= \pi^2_{g_p}(b_i, d_i).\end{aligned}$$

The result below plays a central role in computing gradients of functionals of persistence diagrams (or barcodes). It studies how a persistence diagram $\mathcal{B}^j(h)$ changes with respect to the filtration function h it is based on, in the case where h depends on a set of parameters $p \in \mathfrak{F}$. The fact that ρ_1^p and ρ_2^p are locally constant maps (with respect to u) provides well-defined derivatives.

Proposition 2.40. *For $b_i, d_i \in E_1^j$ and $b'_i, d'_i \in E_2^j$, one has*

$$\nabla b_i(p) = \nabla f_p(\rho_1^p(b_i)) \text{ and } \nabla d_i(p) = \nabla f_p(\rho_1^p(d_i)),$$

$$\nabla b'_i(p) = \nabla g_p(\rho_2^p(b'_i)) \text{ and } \nabla d'_i(p) = \nabla g_p(\rho_2^p(b'_i)).$$

Proof. This is the proof of the first equality only, as the three others are completely analogous. Now, $f_p(\rho_1^p(b_i)) = b_i$ for all i. Moreover, ρ_1^p is locally constant and thus

$$\nabla b_i(p) = \nabla_p f_p(\rho_1^p(b_i)).$$

■

Consider a map $E : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ and define, for $j \in \mathbb{Z}$, the map $\mathcal{T}^j : \mathbb{R}^d \to \mathbb{R}$ with

$$\mathcal{T}^j(p) = E(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p)).$$

Applying the chain rule, one gets

$$\begin{aligned}\nabla\mathcal{T}^j(p) &= \sum_{e \in E_1^j \sqcup E_2^j} \frac{\partial E}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p))\nabla e(p)\\ &= \sum_{e \in E_1^j} \frac{\partial E}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p))\nabla e(p) + \sum_{e \in E_2^j} \frac{\partial E}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p))\nabla e(p)\\ &= \sum_{e \in E_1^j} \frac{\partial E}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p))\nabla f_p(\rho_1^p(b)) + \sum_{e \in E_2^j} \frac{\partial E}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p))\nabla g_p(\rho_2^p(b)),\end{aligned}$$

where the last equality holds by Proposition 2.40.

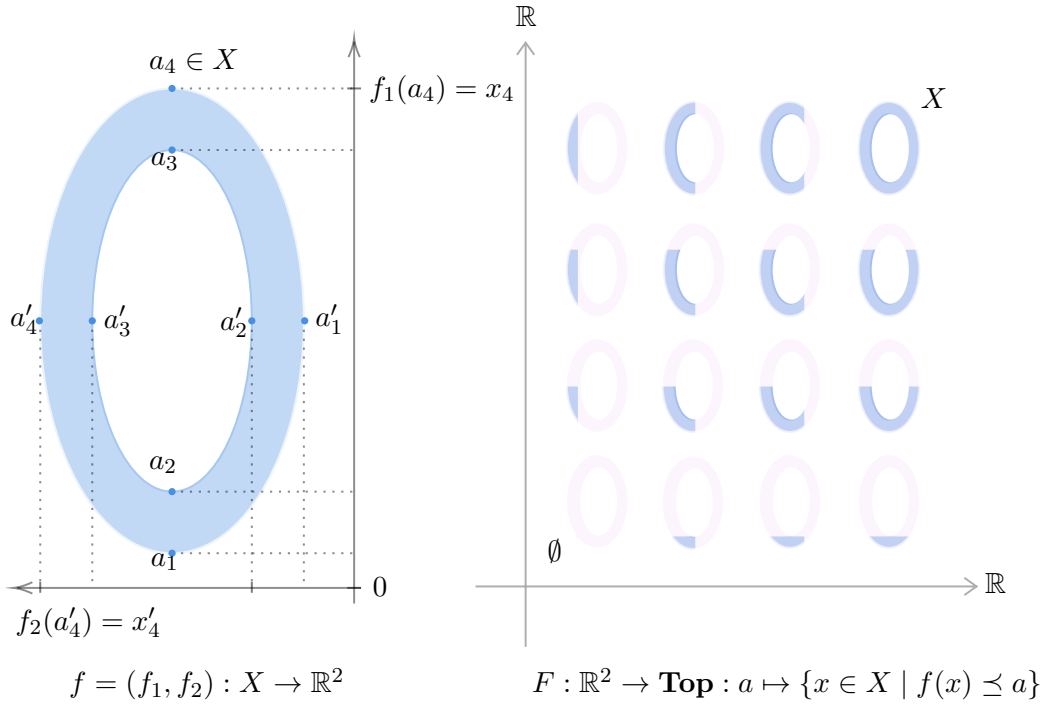


Figure 3: A 2-parameter filtration on an annulus $X \subset \mathbb{R}^2$.

3 Multi-parameter Persistent Homology

In this section, we recall the fundamentals of multi-parameter persistence. We refer to [4] by M. Botnan and M. Lesnick or [6], by G. Carlsson and A. Zomorodian, for a complete introduction to the subject.

3.1 Multi-parameter persistence modules

Definition 3.1 (Multi-parameter filtration)**.** *Let $d \geq 1$. A d-parameter filtration is an order preserving functor $F : (\mathbb{R}^d, \preceq) \to (\mathbf{Top}, \subseteq)$.*

Example 3.2 (d-parameter filtration)**.** *A continuous function $f : X \to \mathbb{R}^d$ induces a d-parameter filtration $F : (\mathbb{R}^d, \preceq) \to (\mathbf{Top}, \subseteq)$ defined as $F(a) = \{x \in X \mid f(x) \preceq a\}$. Figure 3 shows an example of 2-parameter filtration induced by a function $f : X \to \mathbb{R}^2$.*

Definition 3.3 (Multi-parameter persistence module)**.** *Let $d \geq 1$. A d-parameter persistence module is a functor $M : \mathbb{R}^d \to \mathbf{Vect}_\mathbf{k}$.*

Applying a homology functor H_j to a d-parameter filtration F yields a d-parameter persistence module $M : \mathbb{R}^d \to \mathbf{Vect}_\mathbf{k}$ defined by $M(x) = H_j(F(x))$ for any $x \in \mathbb{R}^d$.

Here, morphisms $M(x) \to M(y)$ are induced by the inclusions $F(x) \subseteq F(y)$ for any $x \preceq y \in \mathbb{R}^d$. Combining this observation with Example 3.2, one gets the following.

Definition 3.4. *Let $f : X \to \mathbb{R}^d$ be a continuous function. One defines $S_j(f) : \mathbb{R}^d \to \mathbf{Vect_k}$ to be the* sublevel set persistence module of f*, where*

$$S_j(f)(a) = H_j(x \mid f(x) \preceq a).$$

As seen in Section 2, q-tame 1-parameter persistence modules admit a decomposition into interval modules that is unique up to isomorphism giving place to a persistence diagram and a barcode. Now, in the 1-parameter persistence case, the barcode is a *complete* invariant, in the sense that it completely determines the persistence module it comes from. In dimension $d \geq 2$, there is no topological summary equivalent to the barcode, in the sense that there is no complete invariant.

As in the 1-dimensional case, there is a finiteness condition under which a (multi-parameter) persistence module admits a unique decomposition. A d-parameter persistence module $M : \mathbb{R}^d \to \mathbf{Vect_k}$ is said to be *pointwise finite dimensional (p.f.d)* if, for any $a \in \mathbb{R}^d$, one has $\dim(M_a) < \infty$.

Definition 3.5. *A d-parameter persistence module $B : \mathbb{R}^d \to \mathbf{Vect_k}$ is an* interval module *if it is of the form $I_1 \times ... \times I_d$, where each I_i is a 1-parameter interval module.*

Definition 3.6. *A d-parameter persistence module is* interval-decomposable *if it admits a decomposition $M \cong \bigoplus_{i \in I} M_i$, where each M_i is an interval module.*

Theorem 3.7 (Decomposition)**.** *Any pointwise finite dimensional d-parameter persistence module $M : \mathbb{R}^d \to \mathbf{Vect_k}$ admits an interval decomposition $M \cong \bigoplus_{i \in I} M_i$ that is unique, up to permutation of the summands M_i.*

Hence, barcodes are well-defined for pointwise finite dimensional persistence modules. Figure 4 shows the interval decomposition of a 2-parameter persistence module induced from a function $f : X \to \mathbb{R}^2$, where $X \subset \mathbb{R}^2$ is an annulus.

One can further investigate the algebraic nature of multi-parameter persistence modules through *graded modules*.

Let

$$P_d = \left\{ \sum_i \alpha_i x_1^{j_1^i} \cdots x_d^{j_d^i} \mid j_k^i \in \mathbb{R}_+, k = 1, ..., d \right\}$$

be the polynomial ring with variables $x_1, ..., x_d$, coefficients in K and real positive exponents. For a commutative ring R, we say an R-module M is *d-graded* if there is a K-vector space decomposition $M = \bigoplus_{i \in \mathbb{R}^d} M_i$ such that $\mathbf{x}^j \cdot m \in M_{i+j}$ for any $i \in \mathbb{R}^d$, any monomial $\mathbf{x}^j = \prod_{k=1}^d x_k^{j_k} \in P_d$ and $m \in M_i$. In other words, the

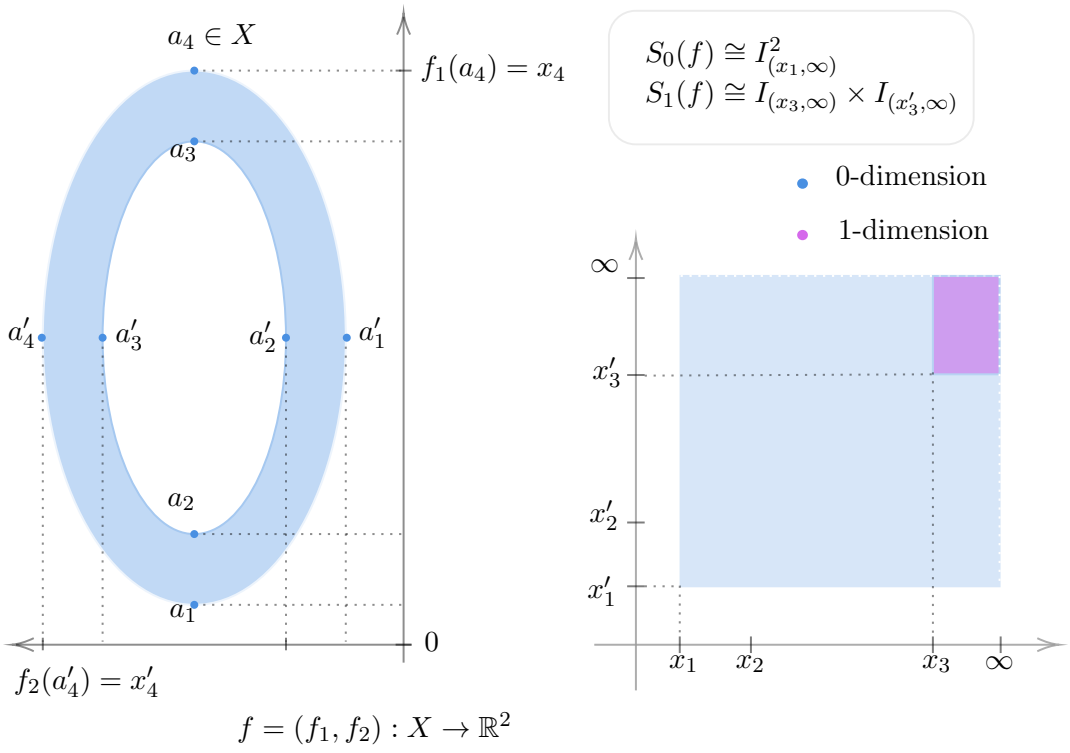


Figure 4: Interval decomposition of the sublevel set homology of a function $f : X \to \mathbb{R}^2$, where $X \subset \mathbb{R}^2$ is an annulus.

K-vector space decomposition is stable under the action of the ring P_d. An element $m \in M$ is said to be *homogeneous* if $m \in M_i$ for some $i \in \mathbb{R}^d$. Clearly, any element $m \in M$ can be written as a sum of homogeneous elements $m = \sum_{r=1}^{l} m_r$. Now, let $\mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k})$ denote the category of d-parameter persistence module, and let **d-mod** be the category of d-graded modules.

Notation. For a functor $M : \mathbb{R}^d \to \mathbf{Vect_k}$ and $i \preceq j \in \mathbb{R}^d$, we denote by $M_{i,j}$ the morphism $M(i \preceq j) : M_i \to M_j$.

Proposition 3.8. *There is an equivalence of categories*

$$\mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k}) \cong \mathbf{d\text{-}mod}.$$

Proof. Define a functor $\phi : \mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k}) \to \mathbf{d\text{-}mod}$ by setting

$$\phi(M) = \bigoplus_{i \in \mathbb{R}^d} M_i$$

for any object M of $\mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k})$. For a homogeneous element $m \in M$, define $x_k(m) := M_{i,i+\mathbf{e}_k}(m)$ for all $i \in \mathbb{R}^d$ and $k \in \{1, ..., d\}$. Then, extend this action on all of $\phi(M)$ linearly, in the following way. For any $m = \sum_{r=1}^{l} m_r$ with m_i homogeneous, set $x_k(m) := \sum_{r=1}^{l} x_k(m_r)$. This naturally yields an action of P_d on $\phi(M)$, giving a well-defined d-graded module $\phi(M)$. It remains to show ϕ is an functor that is also an isomorphism. ∎

When unspecified, M shall denote a d-parameter persistence module, that can be seen either as a functor $\mathbb{R}^d \to \mathbf{Vect_k}$ or as a d-graded P_d-module.

The *internal translation* of M is the map $\varphi_\varepsilon^M : M \to M$ with $\varphi_\varepsilon^M(m) = \mathbf{x}^{\varepsilon\mathbf{1}}(m)$, where $\varepsilon\mathbf{1} = (\varepsilon, ..., \varepsilon)$ and $\mathbf{x}^{\varepsilon\mathbf{1}} = \prod_{k=1}^{d} x_k^\varepsilon$. The *translation endofunctor* $T_\varepsilon : \mathbb{R}^d \to \mathbb{R}^d$ is defined via $T_\varepsilon(i) = i + \varepsilon\mathbf{1}$. This yields an endofunctor

$$T_\varepsilon^* : \mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k}) \to \mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k})$$

with $T_\varepsilon^*(M) = M \circ T_\varepsilon$.

Definition 3.9 (ε-interleaving)**.** *Two d-parameter persistence modules M, N are ε-interleaved if there are natural transformations $f : M \to T_\varepsilon^*(N)$ and $g : N \to T_\varepsilon^*(M)$ satisfying*

$$T_\varepsilon^*(g)f = \varphi_{2\varepsilon}^M,$$
$$T_\varepsilon^*(f)g = \varphi_{2\varepsilon}^N.$$

In other words, two d-parameter persistence modules M, N are ε-interleaved if there are two families of linear maps $\{f_i : M_i \to N_{i+\varepsilon\mathbf{1}}\}_{i \in \mathbb{R}^d}$ and $\{g_i : N_i \to M_{i+\varepsilon\mathbf{1}}\}_{i \in \mathbb{R}^d}$ such that the two conditions below are satisfied.

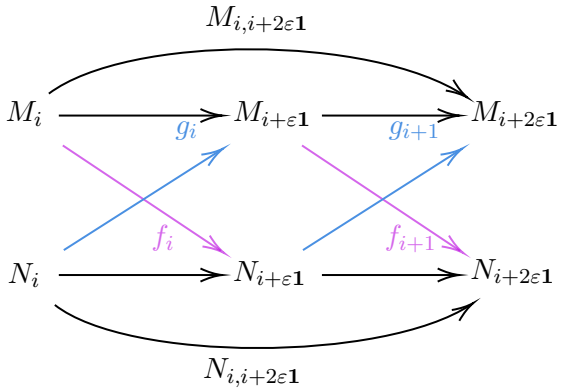


Figure 5: An ε-interleaving between multi-parameter persistence modules M and N.

1. for $i \in \mathbb{R}^d$, $M_{i,i+2\varepsilon\mathbf{1}} = g_{i+\varepsilon\mathbf{1}} \circ f_i$ and $N_{i,i+2\varepsilon\mathbf{1}} = f_{i+\varepsilon\mathbf{1}} \circ g_i$
2. for $i \preceq j \in \mathbb{R}^d$, $f_j \circ M_{i,j} = N_{i+\varepsilon\mathbf{1},j+\varepsilon\mathbf{1}} \circ f_i$ and $g_j \circ N_{i,j} = M_{i+\varepsilon\mathbf{1},j+\varepsilon\mathbf{1}} \circ g_i$

Multi-paramater persistence modules are 0-interleaved if and only if they are isomorphic. Indeed, a 0-interleaving between the modules M and N corresponds to morphisms $f : M \to T_0^*(N) = N$ and $g : N \to T_0^*(M) = M$ such that $f \circ g = \varphi_0^N = \mathrm{id}_N$ and $g \circ f = \varphi_0^M = \mathrm{id}_M$, *i.e.* it corresponds to an isomorphism $M \cong N$.

Definition 3.10 (Interleaving distance)**.** *The* interleaving distance *is defined as*

$$d_I : \begin{array}{l} \mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k}) \times \mathrm{Fun}(\mathbb{R}^d, \mathbf{Vect_k}) \longrightarrow \mathbb{R}_+ \\ (M, N) \longmapsto \inf\{\varepsilon \geq 0 \mid \ M \text{ and } N \text{ are } \varepsilon\text{-interleaved}\} \end{array} .$$

If M and N are not ε-interleaved for any $\varepsilon \in \mathbb{R}_+$, then we set $d_I(M, N) = \infty$.

Definition 3.11 (Matching)**.** *Let S and T be multisets, i.e. sets whose elements are attributed an integer multiplicity. A* matching *$\phi : S \nrightarrow T$ is given by a bijection $S' \leftrightarrow T'$, where $S' \subseteq S$ and $T' \subseteq T$.*

One often says a matching it is a partial bijection.

One says a d-parameter persistence module M is *ε-trivial* if, for any $i \in \mathbb{R}^d$, the linear map $M_i \to M_{i+\varepsilon\mathbf{1}}$ is zero. For a small ε, a ε-trivial d-parameter persistence module is considered as insignificant. This lays down the intuition for the next definition.

Definition 3.12 (ε-matching)**.** *Let M and N be two d-parameter persistence modules, with decompositions into indecomposable summands given by*

$$M \cong \bigoplus_{k=1}^{K} M_k \text{ and } N \cong \bigoplus_{l=1}^{L} N_l$$

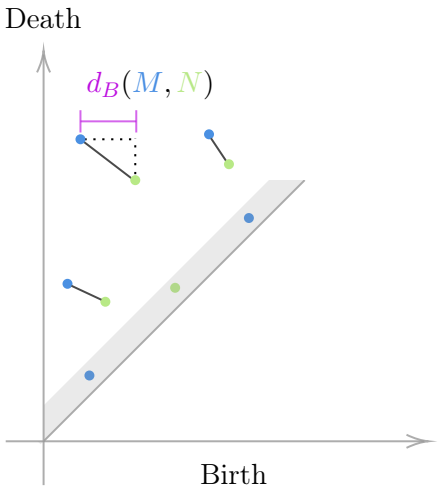


Figure 6: Bottleneck distance between two 1-parameter persistence modules M and N with interval decompositions (barcodes) given by persistence diagrams.

(here M_k and N_l are multi-parameter persistence modules, and not vector spaces). One says M and N are ε-matched, provided that there is a matching $\phi : T = \{1, ..., K\} \twoheadrightarrow \{1, ..., L\} = L$ with underlying bijection $\bar{\phi} : S' \leftrightarrow T'$, satisfying the following.

1. *For any $k \in S \setminus S'$, M_k is 2ε-trivial.*
2. *For any $l \in T \setminus T'$, N_l is 2ε-trivial.*
3. *For any $k \in S'$, M_k and $N_{\phi(k)}$ are ε-interleaved.*

We denote by $\mathcal{M}_d(\varepsilon)$ the set of pairs of ε-matched d-parameter persistence modules. Intuitively, one has $(M, N) \in \mathcal{M}_d(\varepsilon)$ if there is a map that matches summands that are *close*, and leaves unmatched summands that are *persistent-wise insignificant*.

Definition 3.13 (Bottleneck distance d_B)**.** *The Bottleneck distance between two d-parameter persistence modules M, N is defined as (see Figure 6 for an illustration)*

$$d_B(M, N) = \inf\{\varepsilon \mid (M, N) \in \mathcal{M}_d(\varepsilon)\}.$$

By definition, any ε-interleaving induces an ε-matching. Thus, one has the inequality

$$d_I \leq d_B.$$

Notation. For two d-parameter persistence modules M and N, a *Bottleneck matching* between them is a $d_B(M, N)$-matching.

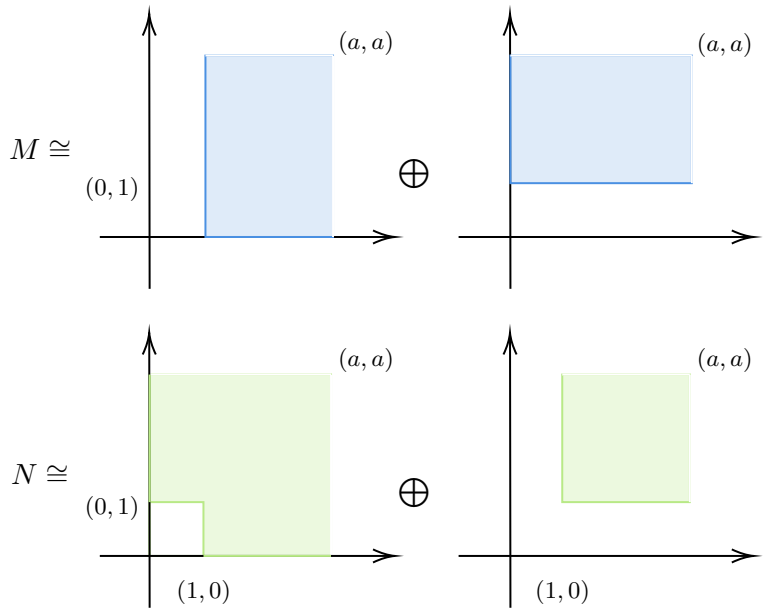


Figure 7: A pair of 2-parameter persistence modules $M, N : \mathbb{R}^2 \to \mathbf{Vect_k}$ with matching distance $d_{\text{match}}(M,N) = 0$ and interleaving distance $d_I(M,N) = 1$.

3.2 The curse of dimensionality

The algebraic objects used in multi-parameter persistent homology are quite complex, in comparison with persistent homology. In [6], the authors show that, in dimension $d \geq 2$, there exists no complete descriptor similar to the barcode. Finding nice computational invariants and distances is an important research direction. The fibered barcode, together with the matching distance, provide a computable and stable invariant for multi-parameter persistence modules. In [20], M. Lesnick and M. Wright introduce the RIVET software, with which one can compute a matching distance for sublevel set homology of functions $f : K_1 \to \mathbb{R}^2$, $g : K_2 \to \mathbb{R}^2$ in complexity $\mathcal{O}(|K|^{11})$. Moreover, in dimension $d \geq 2$, the interleaving distance is **NP**-hard to compute ([2]).

3.3 The fibered barcode

In general, there is no barcode structure existing for multi-parameter persistence modules. Instead, one can obtain a family of barcodes of 1-parameter submodules called the *fibered barcode*, defined below. In the space $\mathbb{R}^d$, the slope $\mathbf{m} = (m_1, ..., m_d)$ of a line $\mathcal{L} = \{t\mathbf{m} + \mathbf{c}\}_{t \in \mathbb{R}}$ is said to be *positive*, if $m_i > 0$ for all $i \in \{1, ..., d\}$. Let Λ be the set of all lines in $\mathbb{R}^d$ with positive slope.

Definition 3.14 (Fibered barcode)**.** *Let $\mathcal{L}$ be a line in $\mathbb{R}^d$ with positive slope. Let $\iota_{\mathcal{L}} : \mathbb{R} \to \mathbb{R}^d$ be the unique order-preserving $\|\cdot\|_\infty$-isometric embedding satisfying*

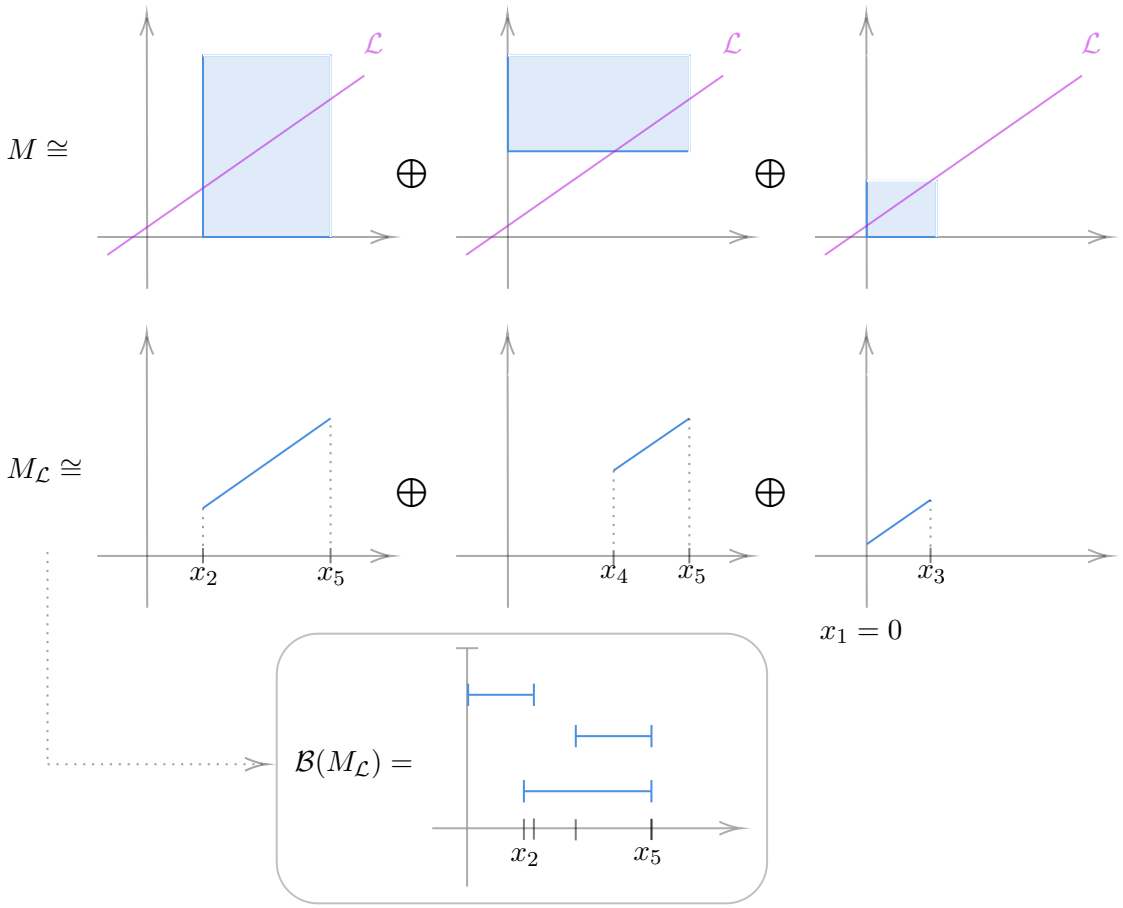


Figure 8: Fibered barcode of a 2-parameter persistence module.

$\mathrm{Im}_{\iota_{\mathcal{L}}} = \mathcal{L}$. *Let* $M_{\mathcal{L}}$ *be the* 1*-parameter persistence module* $M \circ \iota_{\mathcal{L}} : \mathbb{R} \to \mathbf{Vect}_{\mathbf{k}}$. *The* fibered barcode *of* M *is the collection*

$$\mathrm{Fib}(M) = \{\mathcal{B}(M_{\mathcal{L}}))\}_{\mathcal{L} \in \Lambda},$$

where $\mathcal{B}\cdot$*) is the barcode associated to a* 1*-parameter persistence module.*

For example, let $f : X \to \mathbb{R}$ be a continuous function. For any two points $p \preceq q \in \mathbb{R}^d$ on a line $\mathcal{L} \subset \mathbb{R}^d$ with positive slope, one can look at the parametrization $\gamma : \mathbb{R} \to \mathbb{R}^d$ with $\gamma(t) = tp + (1-t)q$ and consider the 1-parameter persistence module

$$M : \mathbb{R} \to \mathbf{Vect}_{\mathbf{k}} : t \mapsto H_j(\{x \in X \mid f(x) \preceq \gamma(t)\}).$$

One may think of $M_{\mathcal{L}}$ as the restriction of M to the line $\mathcal{L}$ (see Figure 8).

Definition 3.15 (Push maps)**.** *Let* $\mathcal{L} \subset \mathbb{R}^d$ *be a line with positive slope. One defines the map* $\mathrm{push}_{\mathcal{L}} : \mathbb{R}^d \to \mathcal{L}$ *with* $\mathrm{push}_{\mathcal{L}}(p) = \min_{a \in \mathcal{L}}\{a \mid a \succeq p\}$.

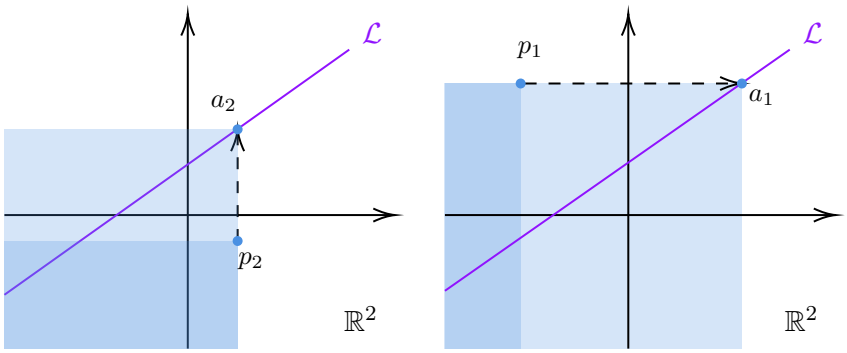


Figure 9: Schematic functioning of the map $\mathrm{push}_{\mathcal{L}}(\cdot)$ associated to a line $\mathcal{L}$. Here, $a_i = \mathrm{push}_{\mathcal{L}}(p_i)$ for $i = 1, 2$. Shaded areas correspond to subposets $\{\mathbf{x} \in \mathbb{R}^2 \mid \mathbf{x} \preceq p_i\}$ (dark shade) and $\{\mathbf{x} \in \mathbb{R}^2 \mid \mathbf{x} \preceq a_i\}$ (light shade), for $i \in \{1, 2\}$.

A map $\mathrm{push}_{\mathcal{L}}(\cdot)$ acts by *pushing* every point of $\mathbb{R}^d$ to the line $\mathcal{L}$ in a way that preserves the partial order $\preceq$, *i.e.* $\mathrm{push}_{\mathcal{L}}(p) \preceq \mathrm{push}_{\mathcal{L}}(q)$ for any two $p \preceq q \in \mathbb{R}^d$.

Proposition 3.16. *Let $\mathcal{L} \in \Lambda$. One has $\mathrm{push}_{\mathcal{L}}(p) \preceq \mathrm{push}_{\mathcal{L}}(q)$ for any $p \preceq q \in \mathbb{R}^d$.*

Proof. One has

$$\begin{aligned} \mathrm{push}_{\mathcal{L}}(p) &= \min_{a \in \mathcal{L}}\{a \mid a \succeq p\} \\ &\leq \min_{a \in \mathcal{L}}\{a \mid a \succeq q\} \\ &= \mathrm{push}_{\mathcal{L}}(q), \end{aligned}$$

because $a \succeq q, a \in \mathcal{L} \Rightarrow a \succeq p, a \in \mathcal{L}$ and thus $\{a \in \mathcal{L} \mid a \succeq q\} \subseteq \{a \in \mathcal{L} \mid a \succeq p\}$. ■

Computing the interleaving distance for multi-parameter modules is an **NP**-hard problem ([2]). To tackle this issue, one considers another metric on the set of d-parameter persistence modules called the *matching distance* (defined in [9]), that is stable with respect to the interleaving distance (Theorem 3.20). The matching distance between two multi-parameter persistence modules M and N is formulated based on taking a weighted supremum of all the Bottleneck distances $d_B(\cdot)$ of 1-parameter persistence submodules obtained by restricting M and N to lines in Λ.

Definition 3.17 (Weight function w)**.** *One defines the* weight function $w : \Lambda \to \mathbb{R}_+$ *in the following way. For a line $\mathcal{L} \in \Lambda$ parametrized by*

$$\mathcal{L} : t \mapsto t \begin{pmatrix} m_1 \\ \vdots \\ m_d \end{pmatrix} + \begin{pmatrix} b_1 \\ \vdots \\ b_d \end{pmatrix},$$

one defines its weight *as*

$$w(\mathcal{L}) = \min_{i=1}^{d} m_i.$$

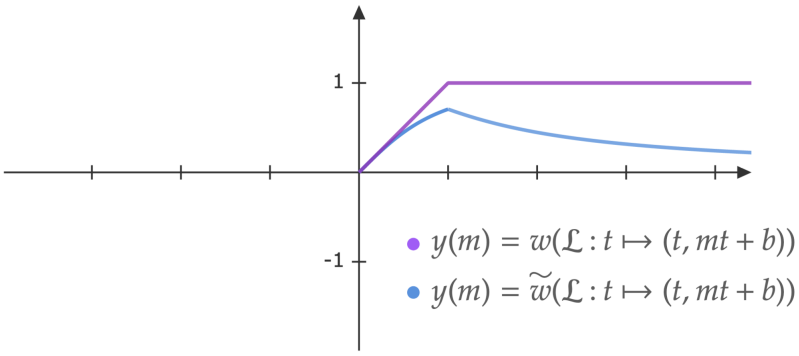


Figure 10: Schematic functioning of the weight functions $w(\cdot)$ and $\tilde{w}(\cdot)$ for $d = 2$.

Definition 3.18 (Matching distance d_{match})**.** *Let $M, N \in \mathbf{Vect}_{\mathbf{k}}$ be pointwise finite dimensional d-parameter persistence modules. The matching distance between M and N is defined as*

$$d_{\text{match}}(M, N) = \sup_{\mathcal{L} \in \Lambda} w(\mathcal{L}) \cdot d_B(M_{\mathcal{L}}, N_{\mathcal{N}}).$$

A note of the weight function w**.** Some authors define the weight of a line $\mathcal{L} \in \Lambda$ parametrized by

$$\mathcal{L} : t \mapsto t \begin{pmatrix} m_1 \\ \vdots \\ m_d \end{pmatrix} + \begin{pmatrix} b_1 \\ \vdots \\ b_d \end{pmatrix}$$

as the quantity $\tilde{w}(\mathcal{L}) = \|\text{push}_{\mathcal{L}}(\mathcal{L}(0) + \mathbf{1}) - \mathcal{L}(0)\|_2$.

Consider the case where $d = 2$. Let $\mathcal{L} \subset \mathbb{R}^2$ be the line given by $t \mapsto (t, tm + b)$ with $m > 0$. The weight of such a line is then given by

$$\tilde{w}(\mathcal{L}) = \begin{cases} \frac{1}{\sqrt{1+m^2}} & \text{if } m \geq 1, \\ \frac{1}{\sqrt{1+\frac{1}{m^2}}} & \text{if } m < 1. \end{cases}$$

The example above shows that the weight function $\tilde{w}(\cdot)$ is defined so as to regularize the distance $d_B(M_{\mathcal{L}}, N_{\mathcal{L}})$ between the 1-parameter persistence submodules taken over a line $\mathcal{L} \subset \mathbb{R}^2$ that is close to one of the axes of $\mathbb{R}^2$. The role of both weight functions $w(\cdot)$ and $\tilde{w}(\cdot)$ is to make the matching distance stable with respect to the interleaving distance. Figure 10 illustrates the case $d = 2$: one has

$$\lim_{m \to 0^+} \tilde{w}(\mathcal{L} : t \mapsto mt + b) = \lim_{m \to \infty} \tilde{w}(\mathcal{L} : t \mapsto mt + b) = 0.$$

Lemma 3.19. *([19, Lemma 1]) An ε-interleaving between d-parameter persistence modules M and N induces an $\frac{\varepsilon}{w(\mathcal{L})}$-interleaving between the submodules $M_{\mathcal{L}}$ and $N_{\mathcal{L}}$.*

Theorem 3.20 (Stability)**.** *For d-parameter persistence modules M and N, one has*

$$d_{\text{match}}(M, N) \leq d_I(M, N).$$

Proof. It follows from Lemma 3.19 that $d_B(M_{\mathcal{L}}, N_{\mathcal{L}}) \leq \frac{1}{w(\mathcal{L})} d_I(M, N)$ for any line $\mathcal{L} \in \Lambda$. Hence, one has $d_{\text{match}}(M, N) = \sup_{\mathcal{L}\in\Lambda} w(\mathcal{L}) \cdot d_B(M_{\mathcal{L}}, N_{\mathcal{N}}) \leq d_I(M, N)$. ■

In [17], M. Kerber, M. Lesnick and S. Oudot describe an exact way to compute the matching distance for the case of 2-parameter persistence modules (called *bimodules*).

Example 3.21. *[23, Example 2.1] Consider the 2-parameter persistence modules M and N given by the block-decompositions shown in Figure 7. For any line $\mathcal{L} \in \lambda$, the barcodes $\mathcal{B}(M_{\mathcal{L}})$ and $\mathcal{B}(N_{\mathcal{L}})$ coincide. It follows that $\text{Fib}(M) = \text{Fib}(N)$ and thus $d_{\text{match}}(M, N) = 0$. However, one has $d_I(M, N) = 1$ (details for this computation can be found in [23]).*

Another invariant used to analyse multi-parameter persistence modules is the so-called *rank invariant* ρ_M. Let p_d denote the subset of $\mathbb{R}^d \times \mathbb{R}^d$ consisting of pairs $(p, q) \in \mathbb{R}^d \times \mathbb{R}^d$ with $p \preceq q$.

Definition 3.22 (Rank invariant ρ_M)**.** *The rank invariant of a d-parameter persistence module M is the map $\rho_M : p_d \to \mathbb{N}$ defined with*

$$\rho_M(p, q) = \text{rank}(M_{p,q}).$$

Proposition 3.23. *([6, Theorem 12]) In dimension $d = 1$, the barcode of the (1-parameter) persistence module M is equivalent to its rank invariant*

Proof. The proof of this equivalence relies on a correspondence between the intervals (b, d) of the barcode $\mathcal{B}(M)$ and the non-zero evaluations $\rho_M(b, d)$. This is illustrated in Figure 11. Each point or interval $(b, d) \in \{(b_i, d_i)\}_{i\in I} = \mathcal{B}(M)$ corresponds to a triangle T_i defined as the convex hull of $\{(b, d), (b, b), (d, d)\}$. For a pair $p \leq q$, the value $\rho_M(p, q)$ is the number of such triangles containing (p, q). In other words, one has the formula

$$\rho_M(p, q) = |\{i \in I \mid (p, q) \in T_i\}|.$$

From there, one can define the correspondence. In fact, for any $p \leq q$, one has

$$\rho_M(p, q) = |\{i \in I \mid b_i \leq p, q \leq d_i\}|.$$

■

This equivalence makes the rank invariant *complete* for 1-parameter persistence modules, in the sense that, up to isomorphism, such a module is entirely determined by its rank invariant, as the barcode is a complete invariant for $d = 1$.

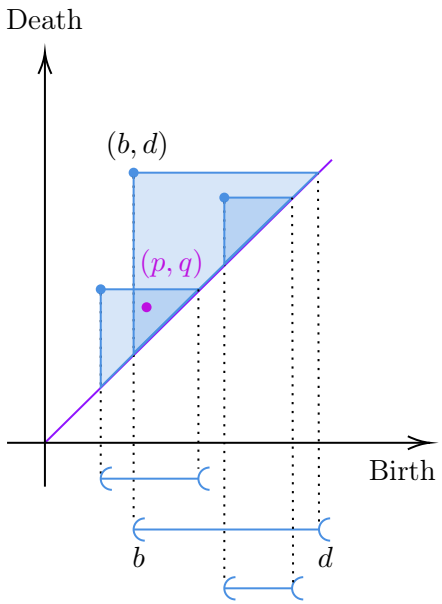


Figure 11: The correspondence giving the bijection $\rho_M \leftrightarrow \mathcal{B}(M)$ for a 1-parameter persistence module M. Each point in the persistence diagram corresponds to an interval in the barcode $\mathcal{B}(M)$. The value $\rho_M(p, q)$ of a pair $p \leq q$ is equal to the number of triangles containing (p, q). In this example, one has $\rho_M(p, q) = 2$.

4 The Projected Barcode

We present the class of *projected barcodes* and *integral sheaf metrics*, originally introduced in the language of sheaf theory by N. Berkouk and F. Petit in [1].

4.1 Motivation

Figure 7 shows a well-known example of 2-parameter persistence modules M and N with $d_{\text{match}}(M,N) = 0$ and $d_I(M,N) = 1$. Now, for any element $r \in \mathbb{R}_+$, one can scale this example to obtain persistence modules with $d_{\text{match}}(M,N) = 0$ and $d_I(M,N) = r$. In other words, one can find persistence modules that are similar to the eye of the matching distance, yet arbitrarily far away to the eye of the interleaving distance. The class of projected barcodes generalizes the fibered barcode, and successfully tackles the problem of the example mentioned above. Just as one defines the matching distance, based on the fibered barcode, one can define the class of integral sheaf metrics, based on the class of projected barcodes.

4.2 Theoretical framework

Consider a continuous map $f : X \to \mathbb{R}^d$.

Black box. Now, d-parameter persistence modules can be seen as derived sheaves on $\mathbb{R}^d$ for the Euclidean topology. Let $p : \mathbb{R}^d \to \mathbb{R}$ be a 1-Lipschitz map. Through one of the six Grothendieck operations, called the direct image, one can assign to a d-parameter persistence module M a single-parameter persistence module Rp_*M.

Notation. Let $\text{Lip}_C(\mathbb{R}^n) = \{p : \mathbb{R}^n \to \mathbb{R} \mid p \ C\text{-Lipschitz}\}$.

Definition 4.1 ($\mathfrak{F}$-projected barcode)**.** *Let $\mathfrak{F} \subset \text{Lip}_1(\mathbb{R}^d)$. The $\mathfrak{F}$-projected barcode associated to f is the collection*

$$\text{Proj}_{\mathfrak{F}}(f) = \{\mathcal{B}(Rp_*S(f))\}_{p\in\mathfrak{F}}.$$

Definition 4.2 ($\mathfrak{F}$-integral sheaf metric)**.** *Let $\mathfrak{F} \subset \text{Lip}_1(\mathbb{R}^d)$. The $\mathfrak{F}$-integral sheaf metric between continuous maps $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$ is given by*

$$d_{\mathfrak{F}}(f,g) = \sup_{p\in\mathfrak{F}}\sup_{j\in\mathbb{N}} d_B\Big(\mathcal{B}(R^jp_*S(f)), \mathcal{B}(R^jp_*S(g))\Big).$$

The class of integral sheaf metrics admits the following stability result.

Theorem 4.3 (Stability)**.** *Let $\mathfrak{F} \subseteq \text{Lip}_C(\mathbb{R}^n)$ with $C \leq 1$. One has*

$$d_{\mathfrak{F}}(f,g) \leq d_I(S(f), S(g)),$$

for any continuous maps $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$.

The class of integral sheaf metrics generalizes the matching distance.

Theorem 4.4 (Lower bound d_{match})**.** *Let $\mathfrak{F} \subseteq \text{Lip}_1(\mathbb{R}^n)$, with $\{\text{push}_{\mathcal{L}}/w(\mathcal{L})\} \subseteq \mathfrak{F}$. One has*

$$d_{\text{match}}(S_j(f), S_j(g)) \leq d_{\mathfrak{F}}(f, g),$$

for any continuous maps $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$.

Theorem 4.5. *([1, Lemma 4.19]) Let X be a compact topological space, and let $f : X \to \mathbb{R}^d$ be continuous. Then*

$$Rp_* S_\bullet(f) = S_\bullet(p \circ f).$$

Another metric is the so-called *k-sliced convolution distance.*

4.2.1 The linear projected barcode

Definition 4.6 (Dual space)**.** *Let $(E, \|\cdot\|)$ be a normed space. The dual space $(E, \|\cdot\|)^*$ is the vector space of linear maps $f : E \to \mathbb{R}$ endowed with the dual norm*

$$\|f\|_* = \sup_{e \in E} \{|f(e)| \mid \|e\| \leq 1\}.$$

Definition 4.7 (Isomorphism of normed vector spaces)**.** *An isomorphism between normed vector spaces $(E, \|\cdot\|_E)$ and $(F, \|\cdot\|_F)$ is a map $\phi : E \to F$ that is an isomorphism of vector spaces and an isometry, meaning that, for any $e \in E$,*

$$\|\phi(e)\|_F = \|e\|_E.$$

Proposition 4.8. *There is an isomorphism of normed vector spaces*

$$(\mathbb{R}^d, \|\cdot\|_\infty)^* \cong (\mathbb{R}^d, \|\cdot\|_1).$$

Proof. Consider the map $\phi : \mathbb{R}^d \to (\mathbb{R}^d)^* : a \mapsto f_a$, where $f_a : \mathbb{R}^d \to \mathbb{R}$ is a linear form defined with

$$f_a(x_1, ..., x_d) = \sum_{i=1}^{d} a_i x_i.$$

The map ϕ is an isomorphism of vector spaces. Indeed, one has $\phi(0) = 0_{(\mathbb{R}^d)^*}$ and

$$\begin{aligned}
\phi(a) + \phi(a')(x_1, ..., x_d) &= f_a(x_1, ..., x_d) + f_{a'}(x_1, ..., x_d) \\
&= \sum_{i=1}^{d} a_i x_i + \sum_{i=1}^{d} a'_i x_i \\
&= \sum_{i=1}^{d} (a_i + a'_i) x_i \\
&= f_{a+a'}(x_1, ..., x_d) \\
&= \phi(a + a')(x_1, ..., x_d),
\end{aligned}$$

for any $(x_1, ..., x_d) \in \mathbb{R}^d$ and $a, a' \in \mathbb{R}^d$, proving linearity. Moreover, ϕ is bijective. Now, it remains to show that ϕ preserves the norms. Let $\|\cdot\|_*$ denoted the dual norm of $\|\cdot\|_\infty$. Let $a \in \mathbb{R}^d$. One has

$$\begin{aligned}
\|f_a\|_* &= \sup_{x\in\mathbb{R}^d} \Big\{ |f_a(x)| \mid \|x\|_\infty \leq 1 \Big\} \\
&= \sup_{x\in\mathbb{R}^d} \Big\{ |\sum_{i=1}^{d} a_i x_i| \mid \|x\|_\infty \leq 1 \Big\} \\
&\leq \sup_{x\in\mathbb{R}^d} \Big\{ \sum_{i=1}^{d} |a_i||x_i| \mid \|x\|_\infty \leq 1 \Big\} \\
&\leq \sup_{x\in\mathbb{R}^d} \Big\{ \|x\|_\infty \sum_{i=1}^{d} |a_i| \mid \|x\|_\infty \leq 1 \Big\} \\
&\leq \sum_{i=1}^{d} |a_i| = \|a\|_1,
\end{aligned}$$

which concludes the proof. ■

Notation. We set

$$\mathfrak{L} := \{p \in (\mathbb{R}^d_+, \|\cdot\|_\infty)^* \mid \|p\|_* = 1\},$$

the set of linear forms on $\mathbb{R}^d$ with positive coefficients and dual norm 1. The isomorphism in Proposition 4.8 restricts to $\mathfrak{L}$ and we have an isomorphism of normed vector subspaces $\mathfrak{L} \cong \{p \in \mathbb{R}^d_+ \mid \|p\|_1 = 1\}$. We will abuse notation by writing $p = (p_1, ..., p_d) \in \mathfrak{L}$, where $p \in \mathbb{R}^d_+$ has 1-norm $\|p\|_1 = 1$.

For the case $d = 2$, one can consider the parametrization defined as (segment)

$$\mathfrak{L} = \{(t, 1-t)) \mid t \in]0, 1[\} \subset \mathbb{R}^2.$$

The case $d = 3$ is illustrated in Figure 12 (filled triangle).

Definition 4.9 (Linear projected barcode)**.** *The linear projected barcode of f is*

$$\mathrm{Proj}_{\mathfrak{L}}(f) = \{\mathcal{B}(Rp_*S(f))\}_{p\in\mathfrak{L}} = \{\mathcal{B}(p \circ f))\}_{p\in\mathfrak{L}}.$$

Definition 4.10 (Linear integral sheaf metric)**.** *The linear integral sheaf metric between continuous maps $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$ is given by*

$$\begin{aligned}
d_{\mathfrak{L}}(f, g) &= \sup_{p\in\mathfrak{F}} \sup_{j\in\mathbb{N}} d_B\Big(\mathcal{B}(Rp_*S_j(f)), \mathcal{B}(Rp_*S_j(g))\Big) \\
&= \sup_{p\in\mathfrak{L}} \sup_{j\in\mathbb{N}} d_B\Big(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)\Big).
\end{aligned}$$

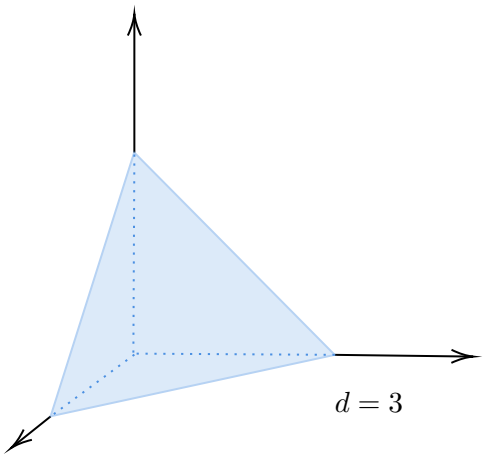


Figure 12: The class $\mathfrak{L} = \{x \in \mathbb{R}^3_+ \mid \|x\|_1 = 1\}$.

One of the underlying properties of the LISM between two multi-filtrations $f = (f_1, ..., f_d)$ and $g = (g_1, ..., g_d)$ is that it captures more information than the Bottleneck distance does. More precisely, by considering the linear forms given by the canonical basis vectors of $\mathbb{R}^d$, one can imagine that, for any $i \in \{1, ..., d\}$,

$$d_B(\mathcal{B}(f_i), \mathcal{B}(g_i)) \leq d_{\mathfrak{L}}(f, g).$$

However, the basis vectors are not contained in $\mathfrak{L}$ and Theorem 4.5 does not hold for those vectors. Although the result is true, it is actually highly non-trivial.

Proposition 4.11. *([1, Corollary 4.27]) Let X and Y be compact topological spaces. Consider continuous maps $f = (f_1, ..., f_d) : X \to \mathbb{R}^d$, $g = (g_1, ..., g_d) : X \to \mathbb{R}^d$. Then*

$$d_B(\mathcal{B}(f_i), \mathcal{B}(g_i)) \leq d_{\mathfrak{L}}(f, g),$$

for any $i \in \{1, ..., d\}$.

In [1], the authors introduce another distance called the *k-sliced convolution distance*.

Definition 4.12 (linear k-sliced convolution distance)**.** *The linear k-sliced convolution distance between continuous maps $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$ is*

$$\mathcal{S}_k(f, g) = \frac{1}{\text{vol}(\mathfrak{L})} \left(\int_{\mathfrak{L}} \sup^k_{j \in \mathbb{N}} d_B(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)) dp \right)^{\frac{1}{k}}.$$

4.3 Implementation

We provide the details to the implementation. We consider a restricted framework given by 2-*Wasserstein linear integral sheaf metrics* and use a `Tensorflow` automatic-differentiation, built on top of M. Carrière's repository.

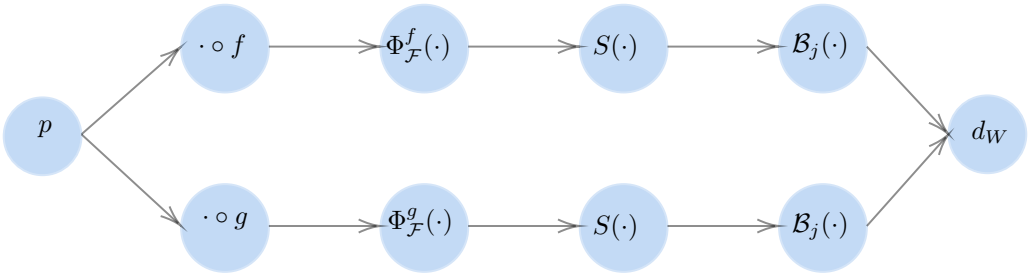


Figure 13: Auto-differentiation tree for LISM computation.

In the context of computing an integral sheaf metric, one is given two functions $f : K_1^0 \to \mathbb{R}^d$ and $g : K_2^0 \to \mathbb{R}^d$ defined on the vertices of finite (geometric) simplicial complexes K_1 and K_2. One can extend those functions two piecewise-linear maps (similarly denoted) $f : |K_1| \to \mathbb{R}^d$ and $g : |K_2| \to \mathbb{R}^d$. Note that $|K_1|$ and $|K_2|$ compact topological spaces.

Two families $\{f_p\}_{p\in\mathfrak{L}}$ and $\{g_p\}_{p\in\mathfrak{L}}$ induce parametrized families of filtrations

$$\Phi^f_{\mathfrak{L}} : \mathfrak{L} \to \mathbb{R}^{|K|} : p \mapsto (f_p(\sigma_1), \cdots, f_p(\sigma_{|K|}),$$
$$\Phi^g_{\mathfrak{L}} : \mathfrak{L} \to \mathbb{R}^{|K|} : p \mapsto (g_p(\sigma_1), \cdots, g_p(\sigma_{|K|}).$$

We are interested in maximizing the functional

$$\mathfrak{L}_j \to \mathbb{R} : p \mapsto d_B(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)).$$

The gradient computation can be implemented with a subgradient algorithm with iterations of the form 1, since an explicit gradient formula was provided. However, we choose to rely on the topological automatic differentiation algorithm whose tree is shown in Figure 13, and implemented with `TensorFlow`.

4.3.1 Wasserstein integral sheaf metrics

One may consider the setting of integral sheaf metrics by replacing the Bottleneck distance d_B with the so-called (see Definition 2.10) 2-Wasserstein distance. All mentioned results and reasoning hold. For example, the map

$$E : \mathbb{R}^{2p} \times \mathbb{R}^q \to \mathbb{R} : D \mapsto d_W(D, D')$$

is also a function of persistence making the functional

$$\mathcal{L}_j : \mathfrak{L} \to \mathbb{R} : p \mapsto d_W(\mathcal{B}_j(p \circ f), D')$$

locally Lipschitz, and thus the topological optimization convergence results hold for

$$\mathcal{T}^j_{f,g} : \mathfrak{L} \to \mathbb{R} : p \mapsto d_W(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g))$$

too (see Section 4.3.4). There are two main advantages to do so. First, the Wasserstein integral sheaf metric admits a more compact gradient derivation and is thus more fit to optimization machine learning tasks. Second, by definition, the 2-Wasserstein distance is a more sensitive measure than the Bottleneck distance as it takes into account all differences accounted for by an optimal matching, as opposed to the Bottleneck distance, which only considers one difference.

Definition 4.13 (Wasserstein $\mathfrak{F}$-integral sheaf metric)**.** *Let $\mathfrak{F} \subset \mathcal{C}^0(\mathbb{R}^d)$. For two continuous maps $f, g : X \to \mathbb{R}$, one defines the* Wasserstein $\mathfrak{F}$-integral sheaf metric *as*

$$d_{\mathfrak{F}}(f, g) = \sup_{p \in \mathfrak{F}} \sup_{j \in \mathbb{N}} d_W(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)).$$

In this section, we shall consider only Wasserstein integral sheaf metrics. For simplicity, we will only speak of *integral sheaf metrics*.

Definition 4.14 (Wasserstein matching)**.** *A* Wasserstein matching *between two persistence diagrams D_1 and D_2 is a matching $\mu : D_1 \to D_2$ such that*

$$d_W(D_1, D_2)^2 = \sum_{p \in D_1} \|p - \mu(p)\|_2^2.$$

Proposition 4.15. *Let $k \in \{1, 2\}$. Consider the functionals given by*

$$d^1_W : D_1 \mapsto d_W(D_1, D_2) \text{ and } d^2_W : D_2 \mapsto d_W(D_1, D_2).$$

Then, for any $e \in D_k$, one has $\frac{\partial d^k_W}{\partial e}(D_1, D_2) = 2(e - \mu(e))$, where $\mu : D_1 \to D_2$ is a Wasserstein matching.

It follows that

$$\begin{aligned}\nabla \mathcal{T}^j_{f,g}(p) = &\sum_{\substack{b \in E^j_1, \\ b \text{ birth}}} \frac{\partial d^1_W}{\partial b}\Big(\mathcal{B}_j(f_p), \mathcal{B}_j(g_p)\Big)\nabla b(p) + \sum_{\substack{d \in E^j_1, \\ d \text{ death}}} \frac{\partial d^1_W}{\partial d}\Big(\mathcal{B}_j(f_p), \mathcal{B}_j(g_p)\Big)\nabla d(p) \\ &+ \sum_{\substack{b \in E^j_2, \\ b \text{ birth}}} \frac{\partial d^2_W}{\partial b}\Big(\mathcal{B}_j(f_p), \mathcal{B}_j(g_p)\Big)\nabla b(p) + \sum_{\substack{d \in E^j_2, \\ d \text{ death}}} \frac{\partial d^2_W}{\partial d}\Big(\mathcal{B}_j(f_p), \mathcal{B}_j(g_p)\Big)\nabla d(p)\end{aligned}$$

and thus, using Proposition 2.40 and Proposition 4.15, one has

$$\nabla \mathcal{T}^j_{f,g}(p) = \sum_{\substack{b\in E^j_1,\\ b \text{ birth}}} 2(b-\mu(b))\nabla f_p(\rho_1^p(b)) + \sum_{\substack{d\in E^j_1,\\ d \text{ death}}} 2(d-\mu(d))\nabla f_p(\rho_1^p(d)) + \sum_{\substack{b\in E^j_2,\\ b \text{ birth}}} 2(b-\mu(b))\nabla f_p(\rho_2^p(b)) + \sum_{\substack{d\in E^j_2,\\ d \text{ death}}} 2(d-\mu(d))\nabla f_p(\rho_2^p(d)),$$

where $\mu : E_1 \to E_2$ is a Wasserstein matching. Now, with the scalar product action $h_u = \sum_i u_i h_i$ for any continuous function $h : K \to \mathbb{R}^d$, one has that

$$\nabla f_p(\rho_1^p(b)) = \left(\frac{\partial f_p}{\partial p_1}(\rho_1^p(b)), ..., \frac{\partial f_p}{\partial p_d}(\rho_1^p(b))\right)^{\mathrm{T}} = \begin{pmatrix} f_1(\rho_1^p(b)) \\ \vdots \\ f_d(\rho_1^p(b)) \end{pmatrix} = f(\rho_1^p(b))$$

and similarly for the terms $\nabla f_p(\rho_1^p(d))$, $\nabla g_p(\rho_2^p(b))$ and $\nabla g_p(\rho_2^p(d))$.

Finally, one obtains

Gradient formula for 2-Wasserstein LISM computation

$$\nabla \mathcal{T}^j_{f,g}(p) = \sum_{\substack{b\in E^j_1,\\ b \text{ birth}}} 2(b-\mu(b))f(\rho_1^p(b)) + \sum_{\substack{d\in E^j_1,\\ d \text{ death}}} 2(d-\mu(d))f(\rho_1^p(d)) + \sum_{\substack{b\in E^j_2,\\ b \text{ birth}}} 2(b-\mu(b))g(\rho_2^p(b)) + \sum_{\substack{d\in E^j_2,\\ d \text{ death}}} 2(d-\mu(d))g(\rho_2^p(d)),$$

where $\mu : E_1 \to E_2$ is a Wasserstein matching.

4.3.2 Further computations for the (Bottleneck) LISM

For completeness, we also compute the gradient of the functional $\mathcal{T}^j_{f,g}$ for the case of the Bottleneck distance, *i.e.* of the map

$$\mathcal{T}^j_{f,g} : \mathfrak{L} \to \mathbb{R} : p \mapsto d_B(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)).$$

Upon setting $E = d_B$, the results of Section 2.3.2 yield the following.

$$\nabla \mathcal{T}^j_{f,g}(p) = \sum_{e\in E^j_1} \frac{\partial d_B}{\partial e}(\mathcal{B}^j_1(p), \mathcal{B}^j_2(p))\nabla f_p(\rho_1^p(b)) + \sum_{e\in E^j_2} \frac{\partial d_B}{\partial e}(\mathcal{B}^j_1(p), \mathcal{B}^j_2(p))\nabla g_p(\rho_2^p(b)).$$

Note. We assume that in a persistence diagram D, all endpoints are unique. Hence, to a birth $b \in D$ (resp. a death $d \in D$), one can associate a unique point $(b, d) \in D$.

Proposition 4.16. *([21]) Consider a birth $b_i \in E_1^j$, and let $\mu : E_2^j \to E_1^j$ be a Bottleneck matching. Let $p_{\max} = (b_{\max}, d_{\max}) = \operatorname{argmax}\{d(p, \mu(p)) \mid p \in E_2^j\}$ and $p'_{\max} = (b'_{\max}, d'_{\max}) = \mu(p_{\max})$.*

If $p_i = (b_i, d_i) = p_{\max}$ and $|b_{\max} - b'_{\max}| > |d_{\max} - d'_{\max}|$, then

$$\frac{\partial d_B}{\partial b_i}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p)) = \begin{cases} 1 \text{ if } b_i < \mu(b_i), \\ -1 \text{ otherwise.} \end{cases}$$

If $p_i = (b_i, d_i) \neq p_{\max}$ or $|b_{\max} - b'_{\max}| \leq |d_{\max} - d'_{\max}|$, then

$$\frac{\partial d_B}{\partial b_i}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p)) = 0.$$

One can derive similar results for $d_i \in E_1^j$, $b_i \in E_2^j$ or $d_i \in E_2^j$. For any $e \in E_1^j \cup E_2^j$, we simplify the notation by writing

$$A(e) := \frac{\partial d_B}{\partial e}(\mathcal{B}_1^j(p), \mathcal{B}_2^j(p)).$$

Consider a Bottleneck matching $\mu : E_2^j \to E_1^j$. Define the point

$$p_{\max} = (b_{\max}, d_{\max}) = \operatorname{argmax}_{p \in E_2^j} \|p - \mu(p)\|_2 \in E_2^j$$

and set $p'_{\max} = (b'_{\max}, d'_{\max}) = \mu(p_{\max}) \in E_1^j$. We use the following notation.

- Let A denote the event that $|b_{\max} - b'_{\max}| > |d_{\max} - d'_{\max}|$.
- Let B denote the event that $|b_{\max} - b'_{\max}| < |d_{\max} - d'_{\max}|$.
- For an event C, one writes

$$\mathbf{1}_C = \begin{cases} 1 \text{ if } C \text{ holds,} \\ 0 \text{ otherwise.} \end{cases}$$

All in all, by distinguishing when $e \in E_1^j \cup E_2^j$ is a birth or a death point, Proposition 4.16 and its analog cases yield

$$\begin{aligned} \nabla \mathcal{T}_{f,g}^j(p) = &\sum_{\substack{b_i \in E_1^j, \\ b_i \text{ birth}}} A(b_i) \nabla f_p(\rho_1^p(b)) + \sum_{\substack{d_i \in E_1^j, \\ d_i \text{ death}}} A(d_i) \nabla f_p(\rho_1^p(b)) \\ &+ \sum_{\substack{b_i \in E_2^j, \\ b_i \text{ birth}}} A(b_i) \nabla g_p(\rho_2^p(b)) + \sum_{\substack{d_i \in E_2^j, \\ d_i \text{ death}}} A(d_i) \nabla g_p(\rho_2^p(b)). \end{aligned}$$

It follows that

$$\nabla \mathcal{T}^j_{f,g}(p) = \mathbf{1}_A \cdot \Big(\operatorname{sgn}(b'_{\max} - b_{\max}) \cdot \Big(\nabla f_p(\rho_1^p(b_{\max})) - \nabla g_p(\rho_2^p(b'_{\max})) \Big) \Big) + \mathbf{1}_B \cdot \Big(\operatorname{sgn}(d'_{\max} - d_{\max}) \cdot \Big(\nabla f_p(\rho_1^p(d_{\max})) - \nabla g_p(\rho_2^p(d'_{\max})) \Big) \Big).$$

Here, sgn denotes the *sign function* defined with $\operatorname{sgn}(x) = 1$ if $x > 0$, $\operatorname{sgn}(x) = -1$ if $x < 0$ and $\operatorname{sgn}(0) = 0$ for any $x \in \mathbb{R}$. Note that if neither condition A nor condition B holds, then one has $\frac{\partial \mathcal{T}^j}{\partial p}(p) = 0$. Finally, one obtains the formula

Gradient formula for (Bottleneck) LISM computation

$$\nabla \mathcal{T}^j_{f,g}(p) = \mathbf{1}_A \cdot \Big(\operatorname{sgn}(b'_{\max} - b_{\max}) \cdot \Big(f(\rho_1^p(b_{\max})) - g(\rho_2^p(b'_{\max})) \Big) \Big) + \mathbf{1}_B \cdot \Big(\operatorname{sgn}(d'_{\max} - d_{\max}) \cdot \Big(f(\rho_1^p(d_{\max})) - g(\rho_2^p(d'_{\max})) \Big) \Big).$$

4.3.3 Estimating the sliced convolution distance

Linear 1-sliced convolution distance $\mathcal{S}_1$. Concretely, we are interested in the 2-Wasserstein version of $\mathcal{S}_1$, which is approximated with a Monte Carlo integral. We present the fundamental reasoning and result of the theory of Monte Carlo methods, and refer to [22] for a rigorous introduction to the subject.

Convention. For simplicity, the normalization factor $\frac{1}{\operatorname{vol}(\mathfrak{L})}$ in the definition of the k-sliced convolution distance $\mathcal{S}_k$ is omitted for the rest of this section.

Consider two continuous functions $f : X \to \mathbb{R}^d$ and $g : Y \to \mathbb{R}^d$. Denoting by $\mathcal{T}^j_{f,g}$ the functional $\mathfrak{L} \to \mathbb{R} : p \mapsto d_W(\mathcal{B}_j(f_p), \mathcal{B}_j(g_p))$, one can write

$$\mathcal{S}_1(f,g) = \int_{\mathfrak{L}} \sup_{j \in \mathbb{N}} \mathcal{T}^j_{f,g}(p) dp.$$

Now, consider an integral

$$I(\varphi) = \int_\Omega \varphi(x) dx,$$

where $\varphi : \mathbb{R}^d \to \mathbb{R}$ is a continuous function and $\Omega \subseteq \mathbb{R}^d$ is a domain. The idea behind Monte Carlo integration is based on the concept of *expectation* of a random variable of the form $\psi(X)$. More precisely, let X be a random variable taking values

in a domain $\Omega \subseteq \mathbb{R}^d$, with density function φ_X, and let $\psi : \mathbb{R}^d \to \mathbb{R}$ be a continuous function. Then the expectation of the random variable $\psi(X)$ is given by

$$\mathbb{E}[\psi(X)] = \int_\Omega \psi(x)\varphi_X(x)dx.$$

Theorem 4.17 (Central limit)**.** *Let $\{X_i\}_{i\in\mathbb{N}}$ be independent and identically distributed random variables with $\mathbb{E}[X_i] = \mu$ and $\mathrm{Var}(X_i) = \sigma^2$. Then, in the L^2 norm,*

$$\frac{1}{n}\sum_{i=1}^{n} X_i \to \mu.$$

Given a function $\psi : \mathbb{R}^n \to \mathbb{R}$ such that $\psi\varphi_X = \varphi$, one has

$$\mathbb{E}[\psi(X)] = \int_\Omega \psi(x)\varphi_X(x)dx = \int_\Omega \varphi(x)dx = I(\varphi).$$

As a consequence, the arithmetic mean of the collection $\{\psi(X_i)\}_{i=1}^n$ converges to the integral $I(\varphi)$. In our case, we consider

- $\varphi : \mathfrak{L} \to \mathbb{R}$ with $\varphi(p) = \sup_{j\in\mathbb{N}} \mathcal{T}_{f,g}^j(p)$
- $\psi : \mathfrak{L} \to \mathbb{R}$ with $\psi(p) = \mathrm{Area}(\mathfrak{L}) \sup_{j\in\mathbb{N}} \mathcal{T}_{f,g}^j(p)$
- $\varphi_P : \mathfrak{L} \to \mathbb{R}$ with $\varphi_P(p) = \frac{1}{\mathrm{Area}(\mathfrak{L})}$ ($P \sim U(\mathfrak{L})$, the uniform distribution on $\mathfrak{L}$)

Here, $\mathrm{Area}(\mathfrak{L})$ denotes the area of $\mathfrak{L}$, and $n \in \mathbb{N}$ is a large number.

To summarize, we estimate the quantity $\mathcal{S}_1(f, g)$ with the empirical mean

$$\hat{\mathcal{S}}_1(f, g) = \frac{1}{n}\sum_{i=1}^{n} \psi(P_i),$$

where $\{P_i\}_{i\in\mathbb{N}}$ is a collection of independent and identically distributed random variables following a uniform law on $\mathfrak{L}$. Note that $\mathfrak{L} \subset \mathbb{R}^d$ can be written as the restriction of the hypersurface

$$\{(x_1, ..., x_d) \in \mathbb{R}^d \mid \sum_{i=1}^{d} x_i = 1\} \subset \mathbb{R}^d$$

to $\mathbb{R}^d_+$. Generally speaking, one has the following result.

Theorem 4.18 (Area of hypersurface)**.** *Consider a differentiable function $f : \mathbb{R}^{d-1} \to \mathbb{R}$. The area of the hypersurface $S \subset \mathbb{R}^d$ given by the equation $f(x_1, ..., x_{d-1}) = x_d$ is*

$$\mathrm{Area}(S) = \int \sqrt{1 + \|\nabla f\|_2^2}.$$

In our case, we want to parametrize the hypersurface $S = \{(x_1, ..., x_d) \in \mathbb{R}^d \mid \sum_{i=1}^d x_i = 1\} \subset \mathbb{R}^d$. To this end, we consider the function $f(x_1, ..., x_{d-1}) = -\sum_{i=1}^{d-1} x_i + 1)$. Applying Theorem 4.18, one finds that

$$\begin{aligned}\text{Area}(S) &= \int_{-1}^{1} \sqrt{1 + \|\nabla f\|_2^2} \\ &= \int_{-1}^{1} \sqrt{1 + \sum_{i=1}^{d-1} (-1)^2} \\ &= \int_{-1}^{1} \sqrt{d} \\ &= 2\sqrt{d}.\end{aligned}$$

In the case $d = 2$, for example, one has

$$\text{Area}(\mathfrak{L}) = \frac{\text{Area}(S)}{2} = \sqrt{2}.$$

In the case $d = 3$, one has

$$\text{Area}(\mathfrak{L}) = \frac{\text{Area}(S)}{4} = \frac{\sqrt{3}}{2}.$$

4.3.4 Convergence of subgradient algorithm

Proposition 4.19. *([1, Corollary 6.14]) The functional*

$$\mathcal{T}_{f,g}^j : \mathfrak{L} \to \mathbb{R} : p \mapsto d_W(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)).$$

is Lipschitz.

Hence, the Rademacher theorem (Theorem 2.20), together with Proposition 4.19, ensure that $\mathcal{T}_{f,g}^j$ is differentiable almost everywhere and one can use the subgradient algorithm with iterations of the form 1, that respect Assumption A.

Black box. The map $\mathcal{T}_{f,g}^j$ is C^d-stratifiable.

As a consequence of Theorem 2.26, one has the following.

Theorem 4.20. *Consider two PL maps $f : K_1 \to \mathbb{R}^d$ and $g : K_2 \to \mathbb{R}^d$, and*

$$\mathcal{T}_{f,g}^j : \mathfrak{L} \to \mathbb{R} : p \mapsto d_W(\mathcal{B}_j(p \circ f), \mathcal{B}_j(p \circ g)).$$

Consider the subgradient algorithm with iterations of the form 1. Then, provided that Assumption A is satisfied, almost surely, one has

$$(p_k)_k \to \lim_{k\to\infty} p_k \in \text{Crit}(\mathcal{T}_{f,g}^j),$$

and $(\mathcal{T}_{f,g}^j(p_k))_k$ converges.

Remark 4.1. One can replace d_W with d_B in the theorem above.

5 Applications

There are several ways to use projected barcodes in real world data analysis problems. For instance, one can perform unsupervised learning to classify graphs (simplicial complexes) or images (cubical complexes) that come together with a multi-filtration.

5.1 Image Analysis

We study the classification of the MNIST dataset. More precisely, we perform unsupervised learning on images of hand-written digits, represented as arrays of size 28×28. We present two main pipelines, namely the *standard TDA pipeline*, in which one assigns a topological signature to an image ([15]), and the *LISM pipeline*, in which one compute pairwise distance between images, based on their projected barcode. We also implement a method based on the sliced convolution distance.

Note. An image $I \in \mathbb{R}^{n \times n}$ can also be seen as a map $\phi_I : [n] \times [n] \to \mathbb{R}$ with $\phi_I(i,j) = I_{ij}$. Here, one writes $[n] = \{1, ..., n\}$ for any integer $n \in \mathbb{N}$. We will often write $I(ij)$ to denote the value $I_{ij} = \phi_I(i,j)$, for any two indices $i, j \in [n]$.

5.1.1 Image filtrations

Given an image $I \in \mathbb{R}^{n \times n}$, examples of relevant filtrations include the family of *height* and *radial filtrations* introduced below. Another choice of filtration is the *gray-scale filtration*, denoted I_{gray}, that simply returns the original image, *i.e.* $I_{\text{gray}} = I$. Both families of height and radial filtrations are based on the so-called *binary filtration*.

Definition 5.1 (Binary filtration)**.** *Consider an image $I \in \mathbb{R}^{n \times n}$ and a threshold $t \in]0, 1[$. The t-binary filtration of I is the image $B_t \in \mathbb{R}^{n \times n}$ given by*

$$B_t(i,j) = \begin{cases} 0 & \text{if } I(ij) < t \\ 1 & \text{if } I(ij) \geq t. \end{cases}$$

When unspecified, the threshold is always set to $t = 0.4$ and write $B = B_t$.

Definition 5.2 (Radial filtration)**.** *Consider an image $I \in \mathbb{R}^{n \times n}$. Given a center $c \in [n] \times [n]$, the c-radial filtration of I is the image $R_c \in \mathbb{R}^{n \times n}$ given by*

$$R_c(i,j) = \begin{cases} \| c - \begin{pmatrix} i \\ j \end{pmatrix} \|_2 & \text{if } B(i,j) = 1 \\ R_\infty & \text{if } B(i,j) = 0, \end{cases}$$

where R_∞ is the maximum of the distances of pixels to the center.

For a unitary vector $v \in \mathbb{R}^2$, we denote by $P_v \subset \mathbb{R}^2$ the hyperplane defined by v. In other words, we set $P_v = \{x \in \mathbb{R}^2 \mid \langle x, v \rangle = 0\}$.

Definition 5.3 (Height filtration)**.** *Consider an image $I \in \mathbb{R}^{n\times n}$. Given a norm-one direction $v \in \mathbb{R}^d$, the v-height filtration of I is the image $H_v \in \mathbb{R}^{n\times n}$ given by*

$$H_v(i,j) = \begin{cases} \langle \begin{pmatrix} i \\ j \end{pmatrix}, v \rangle & \text{if } B(i,j) = 1 \\ H_\infty & \text{if } B(i,j) = 0, \end{cases}$$

where H_∞ is the value of the pixel that is the farthest from the hyperplane P_v. Here, we write $\langle \begin{pmatrix} i \\ j \end{pmatrix}, v \rangle = iv_1 + jv_2$, with $v = (v_1, v_2)$.

Filtrations. Note that, in this section, a *filtration* is a transformed image, and thus a functor $(\mathbb{R}, \preceq) \to (\mathbf{Top}, \subseteq)$. Indeed, consider an image I viewed as a function $\phi_I : [n] \times [n] \to \mathbb{R}$. Denoting by K a cubical complex built on the grid $[n] \times [n]$, one can obtain a linear extension $f_I : |K| \to \mathbb{R}$. This implicitly gives rise to the functor

$$F_I : \begin{array}{c} (\mathbb{R}, \preceq) \to (\mathbf{Top}, \subseteq) \\ a \mapsto \{x \in |K| \mid f(x) \leq a\} \end{array}.$$

In this section, one looks at functions defined on cubical complexes, as those are the topological objects that are most fit to image analysis in machine learning. Cubical complexes are analogous to simplicial complexes, but built with *cubes* instead of simplices. We refer to [18] for a rigorous introduction to cubical homology, and to [10] for a presentation of cubical homology-based machine learning.

5.1.2 Standard TDA pipeline

In [15], the authors propose a general TDA pipeline for the classification of the MNIST digits. It is based on the following principle. To an image $I \in \mathbb{R}^{n\times n}$, one can associate a filtration $I_{\text{filt}} \in \mathbb{R}^{n\times n}$ capturing some specific feature, and look at the so-called 2-*Wasserstein amplitude*, denoted $A_W(\cdot)$, of the resulting cubical persistence diagram $\mathcal{B}_{\text{Cub}}(I_{\text{filt}})$.

The p-Wasserstein distance of a persistence diagram is defined as its p-Wasserstein distance to the empty diagram (with only the diagonal), multiplied by a factor $\frac{\sqrt{2}}{2}$.

Definition 5.4 (p-Wasserstein amplitude)**.** *Consider a persistence diagram $D = \{(b_i, d_i)\}_{i=1}^N$. The p-Wasserstein amplitude of D is defined as*

$$A_{W_p}(D) = \frac{\sqrt{2}}{2} \|\bar{d} - \bar{b}\|_p,$$

where $\bar{b} = (b_1, ..., b_N)$ and $\bar{d} = (d_1, ..., d_N)$.

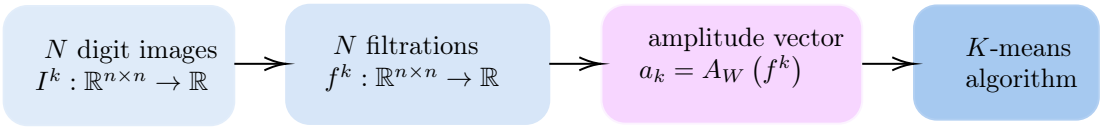


Figure 14: Image-processing TDA pipeline for MNIST classification ([15]).

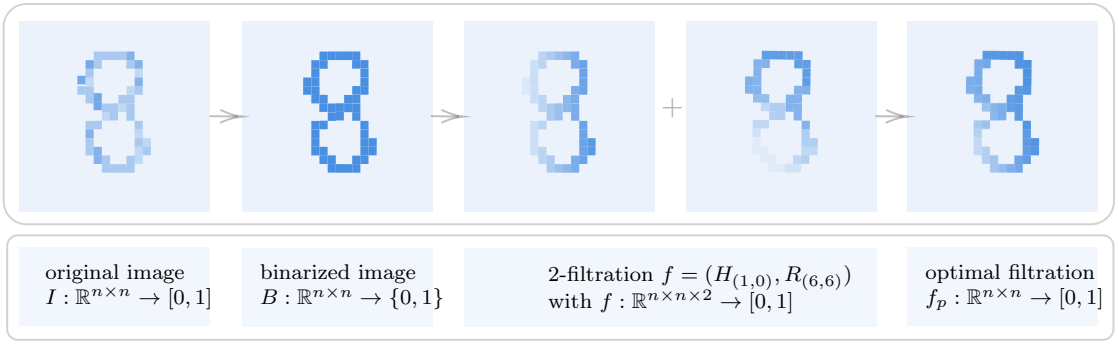


Figure 15: Image processing of a digit during LISM computation between images of a zero and an eight. The experiment is done with a bi-filtration given by $(H_{(1,0)}, R_{(6,6)})$.

Notation. When $p = 2$, one omits the index and writes $A_W(\cdot) = A_{W_2}(\cdot)$.

The standard TDA image processing pipeline is illustrated in Figure 14.

General idea. The idea is to assign a topological signature to an image. Concretely, given an image $I \in \mathbb{R}^{n\times n}$, one can choose a to apply a radial filtration or a height filtration to I, and obtain a filtration $f : \mathbb{R}^{n\times n} \to \mathbb{R}$. Then, one computes the 2-Wasserstein amplitude of the cubical persistence diagram of f : $A_W(\mathcal{B}(f))$. Now, given N images, one obtains a vector of N amplitudes. A clustering algorithm can then perform unsupervised classification on this vector.

In [15], the authors make a list of the eight most important uncorrelated features (filtrations). The idea is to study how some of those features can by combined for classification through the LISM pipeline. The motivation behind this is the fact that some features perform well when comparing two different classes of labels, but fail to classify images when confronted with more classes. The LISM pipeline takes advantage of several features by combining their information.

5.1.3 LISM pipeline

In this subsection, we present the main benchmarking application of this work.

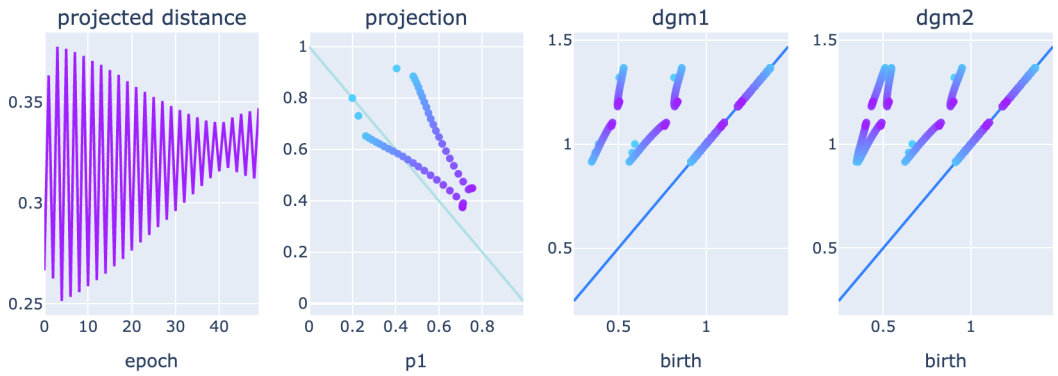


Figure 16: Optimization summary for a LISM computation with $d = 2$ in 1-homology. From left to right : (1) d_W distance evolution (2) projection evolution (3) 1^{rst} persistence diagram evolution (4) 2^{nd} persistence diagram evolution.

General idea. Concretely, given a pair of images $I, J \in \mathbb{R}^{n\times n}$, one can obtain a multi-filtration $f = (f_1, ..., f_d) : \mathbb{R}^{n\times n} \to \mathbb{R}^d$. For $i \in \{1, ..., d\}$, set $I_k = f_k(I)$ and $J_k = f_k(J)$. By optimizing over linear forms $p \in \mathfrak{L}$, one can compute a linear combinations that maximizes the 2-Wasserstein distance between the cubical persistence diagrams associated to $\sum_{i=1}^d p_i I_i$ and $\sum_{i=1}^d p_i J_i$. In other words, we search for the optimal linear form

$$\text{argmax}_{p\in\mathfrak{L}} d_W\left(\sum_{i=1}^d p_i I_i, \sum_{i=1}^d p_i J_i\right).$$

The LISM pipeline is represented in Figure 17. In the implementation, we control the norm of the parameter p by considering the regularized optimization problem

$$\text{argmax}_{p\in\mathbb{R}^2_+} d_W\left(\sum_{i=1}^d p_i I_i, \sum_{i=1}^d p_i J_i\right) - \lambda(\|p\|_1 - 1)^2,$$

where $\lambda > 0$ is a heavily-weighted hyper-parameter. There are other ways of ensuring p has norm $\|p\|_1 = 1$ with `TensorFlow`, like using the `tf.keras.constraints` module or by implementing a *projected* gradient descent. It turns out that, for our experiments, using the regularizer hyper-parameter $\lambda = 1$ and projecting onto $\mathfrak{L}$ at the last step T of the descent (*i.e.* setting $p_{\text{opt}} = \frac{p_T}{\|p_T\|_1}$) approximates the optimal linear form very well. To observe this, we ran proof-check simulations with both the optimization process and a grid search method.

Now, to classify a collection of N images $\{I^k\}_{k=1}^N \subset \mathbb{R}^{n\times n}$, one computes a family of image-induced multi-filtrations $\{f^k : \mathbb{R}^{n\times n} \to \mathbb{R}^d\}_{k=1}^N$, and computes the distance matrix $D = (d_{kl})_{k,l=1}^N$ with

$$d_{kl} = d_{\mathfrak{L}}(f^k, f^l).$$

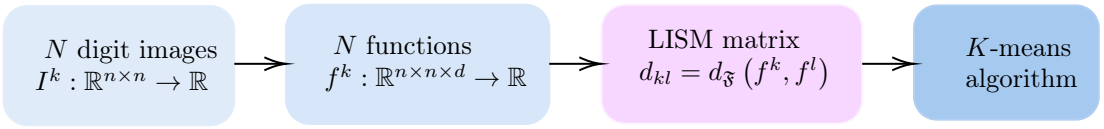


Figure 17: LISM pipeline

One can feed this distance matrix as input to a clustering algorithm.

Example. Let $n = 28$ and $d = 2$. Consider the size-$n \times n$ image representations of the hand-written zero and the hand-written eight shown in Figure 15 (left-most step), denoted I and J respectively. Let

$$f = (f_1, f_2) = (R_{(13,13)}(I), H_{(0,1)}(I)) : \mathbb{R}^{n\times n} \to \mathbb{R}^2,$$
$$g = (g_1, g_2) = (R_{(13,13)}(J), H_{(0,1)}(J)) : \mathbb{R}^{n\times n} \to \mathbb{R}^2.$$

Figure 15 shows the filtrating process of the two digits, as well as the optimized images (right-most step). More precisely, the right-most zero and eight respectively represent the projected bi-filtrations

$$f_{\hat{p}} = \hat{p}_1 f_1 + \hat{p}_2 f_2 : \mathbb{R}^{n\times n} \to \mathbb{R},$$
$$g_{\hat{p}} = \hat{p}_1 g_1 + \hat{p}_2 g_2 : \mathbb{R}^{n\times n} \to \mathbb{R},$$

where $\hat{p} = \text{argmax}_{p\in\mathfrak{L}} d_W(f_p, g_p)$.

Visualization feature for single LISM computation. In Figure 16, we show the optimization process for the case of a 2-filtration given by $R_{(13,13)}$ and $H_{(0,1)}$, in 1-homology. From left to right, the first plot represents the curve $y(t) = d_W(\mathcal{B}_1(f_{p_t}), \mathcal{B}_1(g_{p_t}))$ for $t = 1, ..., N_{\text{steps}}$. The second plot shows the evolution of the projection $p_t \in \mathbb{R}^2$, with a color scale varying along the gradient descent steps. Now, each projection is normalized before being plotted, but in practice, we only have $\|p_t\| \simeq 1$: the norm is controlled with a regularizer $\lambda(\|p\| - 1)_2^2$ (with $\lambda = 1$). The third and fourth plots are scatter plots showing the evolution of the persistence diagrams associated to the $f_p : \mathbb{R}^{n\times n} \to \mathbb{R}$ and $g_p : \mathbb{R}^{n\times n} \to \mathbb{R}$, respectively. To summarize, we plot the following, where N is the number of steps in the optimization :

1. First plot : curve $y(t) = d_W(\mathcal{B}_1(f_{p_t}), \mathcal{B}_1(g_{p_t}))$ for $t = 1, ..., N$
2. Second plot : scatter plot $(p_t)_{t=1}^N \subset \mathbb{R}^2$
3. Third plot : scatter plot $(\mathcal{B}_1(f_{p_t}))_{t=1}^N$
4. Fourth plot : scatter plot $(\mathcal{B}_1(g_{p_t}))_{t=1}^N$

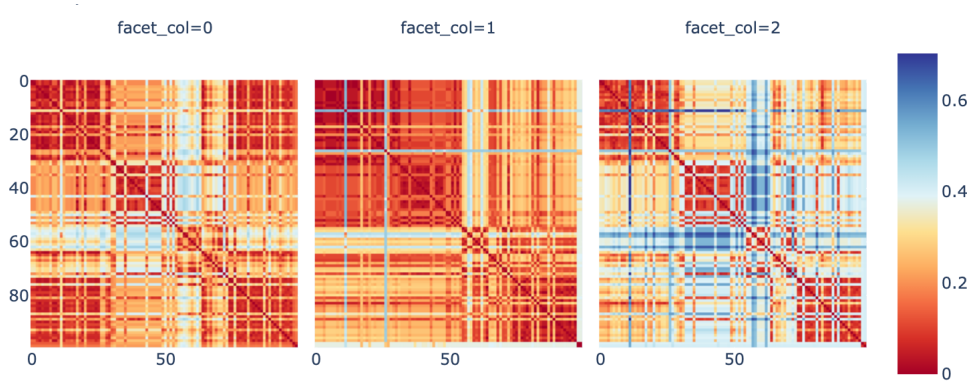


Figure 18: Heat maps of the Wasserstein and LISM distance matrices. The right matrix is $D = (d_{ij})_{i,j=1}^{N}$, where $d_{ij} = d_{\mathfrak{L}}(I^i, I^j)$. For this example, we took $N = 50$ images $(I^k)_{k=1}^{N}$ and consider filtrations $H_{(0,1)}$ and $R_{(13,0)}$. The matrices are block-organized, in the sense that the images are grouped according to their true label.

Visualization feature of distance matrices. Consider the binary classification of digits of 6s and 9s, with filtrations $R_{(13,0)}$ and $H_{(0,1)}$. Figure 18 reports heat maps of three distance matrices

$$D_R = (d_W(f_1^k, f_1^l)_{k,l=1}^{N},$$
$$D_H = (d_W(f_2^k, f_2^l)_{k,l=1}^{N},$$
$$D = (d_{\mathfrak{L}}(f^k, f^l))_{k,l=1}^{N},$$

where $f_1^m = R_{(13,0)}(I_m)$, $f_2^m = H_{(0,1)}(I_m)$ and $f^m = (f_1^m, f_2^m)$.

Clustering. For N images, one obtains a distance matrix of size $N \times N$, that can be fed as input to a clustering algorithm. We choose to use the K-means clustering algorithm, if the N images are distributed among K classes.

Results. Table 1 reports the results of LISM tasks for MNIST classification and compares them with corresponding 1-dimensional subproblems. Given any multi-filtration $(f_1, ..., f_d)$, the LISM performs better than the filtrations f_i individually. In cases where the single-filtrations complement each other and are not highly correlated, the LISM effectively boosts the classification accuracy.

5.1.4 Interpretation and multi-dimensional scaling

For MNIST classification, the standard TDA pipeline with a single well-chosen 1-filtration can sucessfully tackle the problem if there are only two or three digit classes. However, one can increase the complexity of the task by adding more classes. For

metric	multi-filtration	N	digits	dimension	accuracy
d_W	$H_{(0,1)}$	50	$(6,9)$	1	90%
d_W	$R_{(13,0)}$	50	$(6,9)$	1	92%
$d_{\mathfrak{L}}$	$(H_{(0,1)}, R_{(13,0)})$	50	$(6,9)$	1	94%
d_W	$R_{(6,20)}$	50	$(0,1,8)$	1	94%
d_W	$R_{(20,13)}$	50	$(0,1,8)$	1	94%
$d_{\mathfrak{L}}$	$(R_{(6,20)}, R_{(20,13)})$	50	$(0,1,8)$	1	94%
d_W	$R_{(13,13)}$	100	$(1,3,8)$	0	57%
d_W	$H_{(1,0)}$	100	$(1,3,8)$	0	68%
$d_{\mathfrak{L}}$	$(R_{(13,13)}, H_{(1,0)})$	100	$(1,3,8)$	0	71%
d_W	$R_{(13,13)}$	100	$(1,3,8)$	$0,1$	57%
d_W	$H_{(1,0)}$	100	$(1,3,8)$	$0,1$	90%
$d_{\mathfrak{L}}$	$(R_{(13,13)}, H_{(1,0)})$	100	$(1,3,8)$	$0,1$	92%
d_W	$R_{(6,20)}$	100	$(0,2,8,9)$	$0,1$	56%
d_W	$H_{(0,1)}$	100	$(0,2,8,9)$	$0,1$	65%
$d_{\mathfrak{L}}$	$(R_{(6,20)}, H_{(0,1)})$	100	$(0,2,8,9)$	$0,1$	74%
d_W	$R_{(13,13)}$	100	$(3,6,8,9)$	$0,1$	60%
d_W	$H_{(0,1)}$	100	$(3,6,8,9)$	$0,1$	73%
$d_{\mathfrak{L}}$	$(R_{(13,13)}, H_{(0,1)})$	100	$(3,6,8,9)$	$0,1$	81%
d_W	$R_{(6,20)}$	100	$(0,2,8,9)$	1	56%
d_W	$H_{(0,1)}$	100	$(0,2,8,9)$	1	67%
d_W	$R_{(6,6)}$	100	$(0,2,8,9)$	1	74%
$d_{\mathfrak{L}}$	$(R_{(6,20)}, H_{(0,1)}, R_{(6,6)})$	100	$(0,2,8,9)$	1	76%
d_W	$R_{(6,20)}$	100	$(0,2,8,9)$	1	56%
d_W	$H_{(0,1)}$	100	$(0,2,8,9)$	1	67%
d_W	$R_{(6,6)}$	100	$(0,2,8,9)$	1	74%
$\mathcal{S}_1$	$(R_{(6,20)}, H_{(0,1)}, R_{(6,6)})$	100	$(0,2,8,9)$	1	76%

Table 1: Results for MNIST classification

example, classifying digits in $\{3, 6, 8, 9\}$ requires more features (filtrations). The advantage of the projected barcode is that it combines information from several filtrations, making it both a dimensional-reduction tool and a way to extract interesting filtrations from a set of pre-existing features. To illustrate this phenomenon, we use multi-dimensional scaling (MDS) embeddings of distance matrices.

Multi-dimensional scaling. Multi-dimensional scaling (MDS) is an instance of non-linear dimensionality reduction methods. MDS plays an important role in data visualization and is a key concept in the domain of manifold learning [16]. It provides a way of visualizing a notion of distance or dissimilarity within a given data set. More precisely, given a distance matrix of size $N \times N$ containing pairwise distance between points of a data sample of N points, the MDS method computes an embedding of these N points into a low-dimension Euclidean space. In this work, we use the method `sklearn.manifold.MDS` to visualize how far away two images are, according to different metrics, namely d_W, $d_{\mathfrak{L}}$ and $\mathcal{S}_1$.

A specific example. In this example, we consider N images $\{I_k\}_{k=1}^N$ distributed among four classes, namely digits in $\{3, 6, 8, 9\}$. We consider both 0-dimensional homology and 1-dimensional homology. We study the behavior of the LISM pipeline for the two filtrations given by $R_{(13,13)}$ and $H_{(0,1)}$, in comparison with the standard Wasserstein distance matrices for each of these filtrations. Figure 20 shows, from top to bottom, the MDS embedding of the distance matrices

$$D_R = (d_W(f_1^k, f_1^l)_{k,l=1}^N,$$
$$D_H = (d_W(f_2^k, f_2^l)_{k,l=1}^N,$$
$$D = (d_{\mathfrak{L}}(f^k, f^l))_{k,l=1}^N,$$

where $f_1^m = R_{(13,13)}(I_m)$, $f_2^m = H_{(0,1)}(I_m)$ and $f^m = (f_1^m, f_2^m)$. Results are shown in Table 2 and the heat maps of the matrices are displayed in Figure 19. The projected barcode construction successfully combines the information of both the height and radial filtration and effectively boosts the classification accuracy. Both the heat maps and the MDS embeddings allow to understand why. The heat maps are block-organized, in the sense that they store distances d_ij, where the data sample is ordered by label. More precisely, for this example, if Lab is the vector of labels corresponding to the images $\{I_k\}_{k=1}^N$, then one has

$$\text{Lab} = \{3, ..., 3, 6, ..., 6, 8, ..., 8, 9, ..., 9\}$$

. This means that, the more the the classification is accurate, the more the corresponding matrix will present distinguished blocks. For example, the three matrices in Figure 19 have a red block for the first 30 entries. This is because all the images with label 3 look alike, and thus their pairwise differences are very low (the red color denotes close-to-zero values). Now, in the first (left) distance matrix, one can see two

metric	multi-filtration	N	digits	dimension	accuracy
d_W	$R_{(13,13)}$	100	$(3,6,8,9)$	$0,1$	60%
d_W	$H_{(0,1)}$	100	$(3,6,8,9)$	$0,1$	73%
$d_{\mathfrak{L}}$	$(R_{(13,13)}, H_{(0,1)})$	100	$(3,6,8,9)$	$0,1$	81%

Table 2: Results for the example in Section 5.1.4.

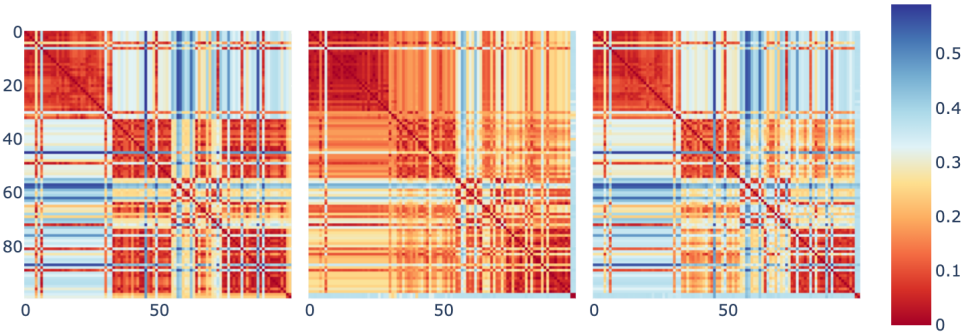


Figure 19: Heat maps of distance matrices for example in Section 5.1.4. From left to right : (1) Wasserstein distance matrix for the $R_{(13,13)}$ filtration, (2) Wasserstein distance matrix for the $H_{(0,1)}$ filtration, and (3) LISM distance matrix for the corresponding 2-filtration. The setting involves $N = 100$ images with labels in $\{3,6,8,9\}$ and both 0-homology and 1-homology.

red blocks that should not appear in an ideal classification. Those two red blocks are the pairwise entries between images with labels 6 and 9. This is coherent with the first MDS plot : labels of 6 and 9 are not distinguished at all. The main trick lies in the fact that, *together*, the height and radial filtrations understand more differences than individually, and complement each other. For instance, although the radial filtration fails to establish a difference between images with labels 6 and 9, the height filtration does, and thus the linear integral sheaf metric does too. Intuitively, given a multi-filtration $(f_1, ..., f_d)$, if each pair of labels is effectively distinguished by at least one of the single-filtrations, then the linear integral sheaf metric performs successfully at distinguishing all labels.

5.1.5 Further considerations

Sliced convolution distance. We also reproduce the experiments by replacing $d_{\mathfrak{L}}$ with the 1-sliced convolution distance $\mathcal{S}_1$. Results are reported in Table 1. Roughly speaking, the sliced convolution distance takes into account all «intermediary» filtrations, as opposed to the LISM, which only considers an optimal filtration. This means, intuitively, that the sliced convolution distance takes account of the differences between two images, from a variety of features, whereas the LISM considers

Figure 20: MDS embeddings visualization for example in Section 5.1.4. From top to bottom : (1) embedding of the Wasserstein distance matrix for the $R_{(13,13)}$ filtration, (2) embedding of the Wasserstein distance matrix for the $H_{(0,1)}$ filtration, and (3) embedding of the LISM distance matrix for the corresponding 2-filtration. The setting involves $N = 100$ images with labels in $\{3, 6, 8, 9\}$ and both 0 and 1-homology.

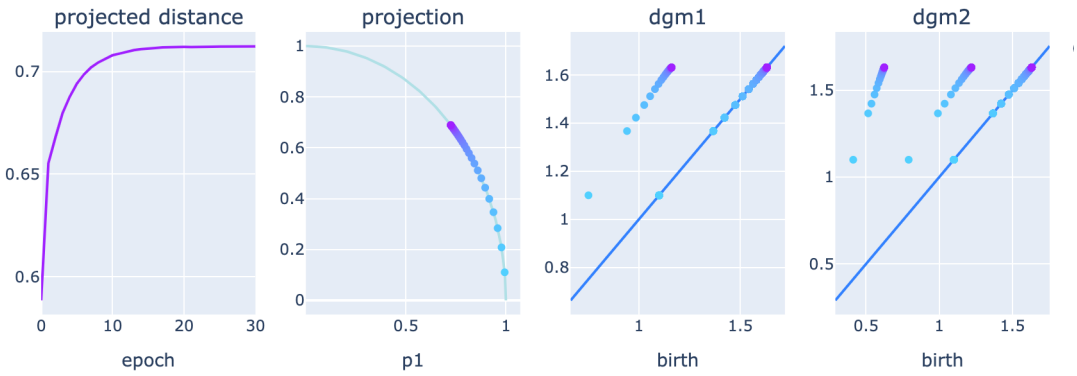


Figure 21: Optimization summary for a LISM computation with $d = 2$ in 1-homology. From left to right : (1) d_W distance evolution (2) projection evolution (3) 1^{rst} persistence diagram evolution (4) 2^{nd} persistence diagram evolution. Here, the optimization ranges over projections in the set $\{x \in \mathbb{R}^d_+ \mid \|x\|_2 = 1\}$ (*i.e.* the $\|\cdot\|_2$-version of the set $\{x \in \mathbb{R}^d_+ \mid \|x\|_1 = 1\}$).

only the feature that makes two images the furthest apart from one another.

A word on optimization. Computing a LISM involves an optimization objective restricted to the set $\mathfrak{L} \cong \{x \in \mathbb{R}^d_+ \mid \|x\|_1 = 1\}$. As mentioned above, in the implementation, this constraint is respected through regularization of the projection norm. Experimentally, one can observe that the $\|\cdot\|_2$-variant of the problem, that is with constraint set equal to $\{x \in \mathbb{R}^d_+ \mid \|x\|_2 = 1\}$, presents an optimization landscape that is both smoother and presents nicer convexity properties (see Figure 21). Convergence to critical points of the loss function is achieved in both cases, but in the first one, the evolution of the projection coordinates during optimization is somewhat erratic and irregular. This behavior is, however, well-understood in the theory of the dynamics of subgradient method for Whitney stratifiable and locally Lipschitz functions [3].

6 Conclusion

The quest for stable and computable invariants in multi-parameter persistent homology is an ongoing project for the computational topology community. The class of projected barcodes, together with integral sheaf metrics, introduced by Nicolas Berkouk and François Petit, aims at solving problems encountered with the current invariants for multi-parameter persistence. Through the language of sheaf theory, they show that the projected barcodes not only present theoretical properties making them stable and computable, but also generalize the fibered barcode construction. In this work, we start by recalling the fundamental results of multi-persistent ho-

mology, present the current state-of-the-art invariants, namely the fibered barcode and its equivalent, the rank invariant. We motivate the quest for new invariants and introduce projected barcodes and integral sheaf metrics for multi-parameter persistence modules. The main aspect of this project is to implement the so-called linear projected barcode and linear integral sheaf metric. Details to both theoretical and practical frameworks are provided. More precisely, we used the recent advances in topological optimization machine learning to formulate a convergence result for the dynamic of the subgradient algorithm designed to compute linear integral sheaf distances. Implementing this algorithm is done via regularized automatic-differentiation. The final part of this work involves benchmarking the linear projected barcode construction by studying applications to image classification. In this last part, we create an image-processing pipeline and interpret the behavior of linear integral sheaf metrics in the context of image analysis. We show how one can use the linear projected barcode to extract new features for MNIST classification and train a machine learning model. The implementation, together with illustrative notebooks and reproducible experiments will soon be made publicly available at this GitHub account (note : a part of the implementation is built on top of Mathieu Carrière's `difftda` Github Repository, that contains a nice topological optimization framework).

6.1 Future work

The work of N. Berkouk and F. Petit opens a new gate in multi-persistent homology. This project can be continued by investigating the following aspects and directions.

1. *Extend the optimization framework to Vietoris-Rips filtrations.* The Vietoris-Rips filtrations fall within the considerations of the optimization framework in [8], and thus theoretical convergence of subgradient methods is guaranteed.

2. *Consider extended persistence instead of post-processing diagrams.* A lot of information is lost when processing diagrams by discarding their essential parts. A remedy to this is to consider extended persistence.

3. *Extend the optimization framework to include push maps* $\mathrm{push}_{\mathcal{L}}$. Note that we are not far away from this. Indeed, consider a line $\mathcal{L} \in \Lambda$ for $d = 2$, passing through the origin and parametrized as $\gamma(t) = mt$. Then one has

$$\mathrm{push}_{\mathcal{L}}(x_1, x_2) = \begin{cases} \frac{x_2}{m} & \text{if } (x_1, x_2) \in \mathcal{L}^+ \\ mx_1 & \text{if } (x_1, x_2) \in \mathcal{L}^-, \end{cases}$$

   where $\mathcal{L}^+$ and $\mathcal{L}^-$ denote the parts of $\mathbb{R}^2$ that are above and below $\mathcal{L}$, respectively. In other words, $\mathrm{push}_{\mathcal{L}}|_{\mathcal{L}^+} = (0, \frac{1}{m})$ and $\mathrm{push}_{\mathcal{L}}|_{\mathcal{L}^-} = (m, 0)$ in terms of projections (p_1, p_2). The similar reasoning works in higher dimensions, and thus one just needs to optimize over a class $\mathfrak{F}$ made a piecewise projections.

However, note that the theoretical behaviour of push maps remains not understood (this goes behind the scope of this work, but the idea is that the key formula $Rp_*(f) = S(p \circ f)$ in [1], that is valid for $p \in \mathfrak{L}$, may not hold in general for push maps).

4. *Make a rigorous study of the algorithmic complexity of LISM computation.* One can study the complexity of computing one single gradient of the form $\frac{\partial d_W}{\partial p}$ through the forward and backward passes that occur in the automatic-differentiation tree shown in Figure 13.

5. *Consider further applications to graph analysis.* In applications, there is a variety of data sets containing graphs with multi-filtrations defined on their 0-skeleton. One can perform various graph classification tasks. Note that this setting is already implemented.

6. *Integrate the LISM concept in more complex machine learning pipelines.* One could think of training a model. Suppose we are given N images $(I^k)_{k=1}^N \subset \mathbb{R}^{n\times n}$, each having a label $L_k \in \{1, ..., m\}$. Consider a choice of d filtration functions, so that one obtains N^2 pairs of d-filtrations. Now, for each pair $(k, l) \in [N] \times [N]$, the projected barcode method returns an optimal projection $p_{k,l} \in \mathbb{R}^d$. For each pair of classes $(a, b) \in [m] \times [m]$, define

$$p^{a,b} = \frac{1}{C_{a,b}} \sum_{\substack{k,l=1 \\ L_k=a, L_l=b}}^{N} p_{k,l},$$

where $C_{a,b} = \{(k, l) \in [N] \times [N] \mid L_k = a, L_l = b\}$. This way, for each pair of digit labels, we store a new filtration (or feature) that is designed to distinguish between those two labels specifically. This provides us with a new class of features of images, made of linear combinations of radial and height filtrations.

References

[1] Nicolas Berkouk and Francois Petit. *Projected distances for multi-parameter persistence modules*. 2022. DOI: 10.48550/ARXIV.2206.08818. URL: https://arxiv.org/abs/2206.08818.

[2] HÃ¥vard Bjerkevik, Magnus Botnan, and Michael Kerber. "Computing the Interleaving Distance is NP-Hard". In: *Foundations of Computational Mathematics* 20 (Nov. 2019). DOI: 10.1007/s10208-019-09442-y.

[3] Jerome Bolte, Edouard Pauwels, and Rodolfo Rios-Zertuche. *Long term dynamics of the subgradient method for Lipschitz path differentiable functions*. 2020. DOI: 10.48550/ARXIV.2006.00098. URL: https://arxiv.org/abs/2006.00098.

[4] Magnus Bakke Botnan and Michael Lesnick. *An Introduction to Multiparameter Persistence*. 2022. DOI: 10.48550/ARXIV.2203.14289. URL: https://arxiv.org/abs/2203.14289.

[5] Gunnar Carlsson and Mikael Vejdemo-Johansson. *Topological Data Analysis with Applications*. Cambridge University Press, 2021. DOI: 10.1017/9781108975704.

[6] Gunnar Carlsson and Afra Zomorodian. "The Theory of Multidimensional Persistence". In: *Discrete and Computational Geometry* 42 (June 2007), pp. 71–93. DOI: 10.1007/s00454-009-9176-0.

[7] Mathieu Carriere and Andrew Blumberg. "Multiparameter Persistence Image for Topological Machine Learning". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 22432–22444. URL: https://proceedings.neurips.cc/paper/2020/file/fdff71fcab656abfbefaabecab1a7f6d-Paper.pdf.

[8] Mathieu Carrière et al. *Optimizing persistent homology based functions*. 2021. arXiv: 2010.08356 [cs.CG].

[9] Andrea Cerri et al. "Betti numbers in multidimensional persistent homology are stable functions". In: *Mathematical Methods in The Applied Sciences* 36 (2013), pp. 1543–1557.

[10] Seungho Choe and Sheela Ramanna. "Cubical Homology-Based Machine Learning: An Application in Image Classification". In: *Axioms* 11.3 (2022). ISSN: 2075-1680. DOI: 10.3390/axioms11030112. URL: https://www.mdpi.com/2075-1680/11/3/112.

[11] Damek Davis et al. *Stochastic subgradient method converges on tame functions*. 2018. DOI: 10.48550/ARXIV.1804.07795. URL: https://arxiv.org/abs/1804.07795.

[12] Tamal Krishna Dey and Yusu Wang. *Computational Topology for Data Analysis*. Cambridge University Press, 2022. DOI: 10.1017/9781009099950.

[13] Herbert Edelsbrunner and John Harer. “Persistent homology survey”. In: *Discrete Computational Geometry - DCG* 453 (Jan. 2008). DOI: 10.1090/conm/453/08802.

[14] Lawrence C. Evans and Ronald F. Gariepy. *Measure Theory and Fine Properties of Functions*. Chapman and Hall/CRC, 2015. DOI: https://doi.org/10.1201/b18333.

[15] Adélie Garin and Guillaume Tauzin. “A Topological "Reading" Lesson: Classification of MNIST using TDA”. In: *CoRR* abs/1910.08345 (2019). arXiv: 1910.08345. URL: http://arxiv.org/abs/1910.08345.

[16] Mark Hansel, Joseph Kruskal, and Myron Wish. “Multidimensional Scaling”. In: *Contemporary Sociology* 9 (Jan. 1980), p. 107. DOI: 10.2307/2065614.

[17] Michael Kerber, Michael Lesnick, and Steve Oudot. *Exact computation of the matching distance on 2-parameter persistence modules*. 2019. arXiv: 1812.09085 [math.AT].

[18] Piotr Kot. “Homology Calculation of Cubical Complexes in Rn”. In: *computational methods in science and technology* 12 (2006), pp. 115–121.

[19] Claudia Landi. “The rank invariant stability via interleavings”. In: *CoRR* abs/1412.3374 (2014). arXiv: 1412.3374. URL: http://arxiv.org/abs/1412.3374.

[20] Michael Lesnick and Matthew Wright. *Computing Minimal Presentations and Bigraded Betti Numbers of 2-Parameter Persistent Homology*. 2019. DOI: 10.48550/ARXIV.1902.05708. URL: https://arxiv.org/abs/1902.05708.

[21] Adrien Poulenard, Primoz Skraba, and Maks Ovsjanikov. “Topological Function Optimization for Continuous Shape Matching”. In: *Computer Graphics Forum* 37.5 (2018), pp. 13–25. DOI: 10.1111/cgf.13487. URL: https://hal.archives-ouvertes.fr/hal-02953321.

[22] Christian P. Robert and George Casella. “Monte Carlo Integration”. In: *Introducing Monte Carlo Methods with R*. New York, NY: Springer New York, 2010, pp. 61–88. ISBN: 978-1-4419-1576-4. DOI: 10.1007/978-1-4419-1576-4_3. URL: https://doi.org/10.1007/978-1-4419-1576-4_3.

[23] Oliver Vipond. *Local Equivalence of Metrics for Multiparameter Persistence Modules*. 2020. arXiv: 2004.11926 [math.AT].