Spectral relaxations of persistent rank invariants

Matt Piekenbrock* and Jose Perea†

Abstract: *Using a duality result between persistence diagrams and persistence measures, we introduce a framework for constructing families of continuous relaxations of the persistent rank invariant for persistence modules indexed over the real line. Like the rank invariant, these families obey inclusion-exclusion, are derived from simplicial boundary operators, and encode all the information needed to construct a persistence diagram. Unlike the rank invariant, these spectrally-derived families enjoy a number of stability and continuity properties typically reserved for persistence diagrams, such as smoothness and differentiability over the positive semi-definite cone. Leveraging a connection to combinatorial Laplacian operators, we find the non-harmonic spectra of our proposed relaxation encode valuable geometric information about the underlying space, prompting several avenues for geometric data analysis. Exemplary applications in topological data analysis and machine learning, such as hyper-parameter optimization and shape classification, are investigated in the full paper.*

Background: Persistent homology related pipelines typically follow a well-established pattern: given input data set X, construct a filtration (K, f) from X such that useful topological or geometric information may be profitably gleaned from its *persistence diagram*—a multiset summary of f constructed by pairing homological critical values $\{a_i\}_{i=1}^n$ with non-zero *multiplicities* $\mu_p^{i,j}$ or sums of Betti numbers $\beta_p^{i,j}$ [5]:

$$\begin{aligned} \mathrm{dgm}_p(f) &\triangleq \{\, (a_i, a_j) : \mu_p^{i,j} \neq 0 \,\} \,\cup\, \Delta \\ \mu_p^{i,j} &\triangleq \left(\beta_p^{i,j\text{-}1} - \beta_p^{i,j}\right) - \left(\beta_p^{i\text{-}1,j\text{-}1} - \beta_p^{i\text{-}1,j}\right) \end{aligned}$$

By pairing simplices using homomorphisms between homology groups, diagrams demarcate homological features succinctly. The essential quality of persistence is that this pairing exists, is unique, and is stable under additive perturbations [5]. Whether for shape recognition [], computer vision, metric learning [10], dimensionality reduction [], or time series analysis [], persistence is the de facto connection between homology and the application frontier.

Though theoretically sound, diagrams suffer from several practical issues: they are sensitive to strong outliers, far from injective, expensive to compute, and expensive to compare. Practitioners have tackled some of these issues by equipping diagrams with additional structure by way of maps to function spaces—examples include persistence images [], persistence landscapes [], and template functions []. These diagram vectorizations have proven useful for learning applications due to their stability and metric configurability []. The scalability issue remains exacerbated though, as these vectorizations require diagrams as part of their input.

Approach: Rather than adding structure to precomputed diagrams, we devise a spectral method that performs both steps, simultaneously and approximately. Our approach constructs a vector-valued mappings over a *parameter space* $\mathcal{A} \subset \mathbb{R}^d$:

$$(X_\alpha, \mathcal{R}, \epsilon, \tau) \mapsto \mathbb{R}^{O(|\mathcal{R}|)}$$

where X_α is an $\mathcal{A}$-parameterized input data set, $\mathcal{R} \subset \Delta_+$ a *sieve* over the upper half-plane Δ_+, and $(\epsilon, \tau) \in \mathbb{R}_+^2$ are approximation/smoothness parameters, respectively. The intuition is that $\mathcal{R}$ is used to filter and summarize the topological and geometric behavior exhibited by X_α for all $\alpha \in \mathcal{A}$, thereby *sifting* the space $\mathcal{A} \times \Delta_+$. Our strategy is motivated both by a technical observation that suggests several advantages exist for the rank invariant computation (see section **??**) and by measure-theoretic perspectives on $\mathbb{R}$-indexed persistence modules [2, 3], which generalize () to arbitrary *corner points* $(\hat{\imath}, \hat{\jmath}) \in \Delta_+$:

$$\begin{aligned} \mu_p(f; R) &\triangleq \mathrm{card}\left(\mathrm{dgm}_p(f)\big|_R \right) \\ &= \min_{\delta > 0} \left(\beta_p^{\hat{\imath}+\delta, \hat{\jmath}\text{-}\delta} - \beta_p^{\hat{\imath}+\delta, \hat{\jmath}+\delta}\right) - \left(\beta_p^{\hat{\imath}\text{-}\delta, \hat{\jmath}\text{-}\delta} - \beta_p^{\hat{\imath}\text{-}\delta, \hat{\jmath}+\delta}\right) \end{aligned}$$

In the full paper, we show that in $\approx O(m)$ memory and $\approx O(mn)$ time, where m, n are the number of $p+1, p$ simplices in the complex, respectively (section **??**). The approximation is spectral-based and is particularly efficient when executed on parameterized families of inputs. When the parameters ϵ and τ made small enough, both invariants are recovered exactly. In deriving the approximation, we obtain families of continuous rank invariants which are Lipshitz continuous, stable under perturbations, and differentiable on the positive semi-definite cone. Unlike existing dynamic persistence algorithms, our approach is simple in that it requires no complicated data structures or maintenance procedures to implement. The proposed relaxation is also *matrix-free*, requiring only

*Northeastern University

†Testing

as much memory as is needed to enumerate simplices in the underlying complex K. Interestingly, our results also imply the existence of an efficient output-sensitive algorithm for computing Γ-persistence pairs with at least $(\Gamma > 0)$-persistence (via [4]) that requires the operator $x \mapsto \partial x$ as its only input, which we consider to be of independent interest.

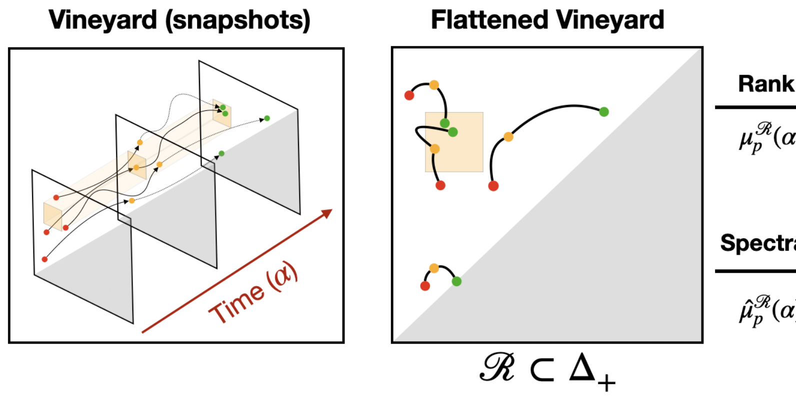

Figure 1: From left to right: Vineyards analogy of diagrams at 'snapshots' over time; vineyard curves flattened with a sieve $\mathcal{R} \subset \Delta_+$; (right) the integer-valued multiplicity function $\mu_p^{\mathcal{R}}(\alpha)$ as a function of time $\alpha \in \mathbb{R}$ (top) and a real-valued spectral relaxation (bottom)

Using the duality between rank functions and diagrams, we not only avoid explicitly constructing diagrams, but we in fact avoid using the reduction algorithm ([7]) entirely. As the vectorization we propose continuously interpolates between the rank invariant and a spectral operator, we elucidate a connection between persistence and other areas of applied mathematics, such as Tikhonov regularization, compressive sensing, and iterative subspace methods. Moreover, inspired by a relationship established between the persistent Betti numbers and combinatorial Laplacian operators [], we show our vectorization able to harvest the rich geometric information such operators encode for tasks like shape classification and filtration optimization.

Relaxing rank invariant

Let K denote a fixed simplicial complex constructed from the data set X and $\mathcal{A} \subset \mathbb{R}^d$ a *parameter space* which indexes a continuously-varying filter function f_α of K:

$$(K, f_\alpha) \triangleq \{ f_\alpha : K \to \mathbb{R} \mid f_\alpha(\tau) \leq f_\alpha(\sigma) \} \qquad (0.1)$$

for all $\tau \subseteq \sigma \in K, \alpha \in \mathcal{A}$. Exemplary choices of f_α include filtrations geometrically realized from methods that themselves are have parameters, such as density filtrations or time-varying filtrations over dynamic metric spaces [10].

Select a *sieve* $\mathcal{R} \subset \Delta_+ = \{ (i, j) \in \bar{\mathbb{R}}^2 \mid i < j \}$

that is decomposable to a disjoint set of rectangles:

$$\mathcal{R} = R_1 \cup R_2 \cup \cdots \cup R_h \tag{0.2}$$

This choice typically requires a priori knowledge and is application-dependent. In section **??** we give evidence random sampling may be sufficient for vectorization or exploratory purposes, when $\mathcal{R}$ is unknown.

Fix a homology dimension $p \geq 0$ and parameters $(\epsilon, \tau) \in \mathbb{R}^2_+$ representing how *closely* and *smoothly* the relaxation should model the quantity:

$$\mu_p(\mathcal{R} \times \mathcal{A}) \triangleq \{\, \mathrm{card}(\mathcal{R} \cap \mathrm{dgm}(K, f_\alpha)) \mid \alpha \in \mathcal{A} \,\} \tag{0.3}$$

We will show in section **??**, letting both $\tau \to 0$ and and $\epsilon \to 0$ yields the multiplicity function μ_p exactly. Choose a combinatorial Laplacian operator $\mathcal{L}$ to associate to $\mathcal{R}$:

$$\mathcal{L} : C^p(K, \mathbb{R}) \times \mathcal{A} \to C^p(K, \mathbb{R}) \tag{0.4}$$

The choice of $\mathcal{L}$ determines how the geometric and topological information about (K, f_α) is encoded. For each corner point (i, j) in the boundary of $\mathcal{R}$, restrict and project $\mathcal{L}$ onto a Krylov subspace:

$$\mathcal{K}_n(\mathcal{L}, v) \triangleq \mathrm{span}\{v, \mathcal{L}v, \mathcal{L}^2 v, \ldots, \mathcal{L}^{n-1} v\}, \qquad v \in \mathrm{span}(\mathbf{1})^\perp$$

The eigenvalues of $T = \mathrm{proj}_{\mathcal{K}} \mathcal{L}\big|_{\mathcal{K}}$ form the basis of the (ϵ, τ)-approximation of (0.3) (see section **??**). The remaining steps of the relaxation depend on the application in mind. The duality between diagrams and rank functions suggests any application exploiting vectorized persistence information may benefit from our relaxation; examples include characterizing swarm and flocking behavior with Betti curves [], topology-guided image denoising [], detecting bifurcations in dynamical systems [], generating metric invariants for shape classification and metric learning [], and so on. Moreover, the differentiability of our relaxation enables learning applications seeking to optimize persistence information, such as filtration optimization, incorporating topological priors into loss functions, and....

The following results will be used in proofs and serve as the primary technical motivations for this effort, beginning with a derivation of a lesser known expression of the persistent Betti number. Given a filtration $K_\bullet$ of size $N = |K_\bullet|$, its p-th persistent Betti number $\beta_p^{i,j}$ at index $(i, j) \in \Delta_+^N$ is defined as follows:

$$\begin{aligned} \beta_p^{i,j} &= \dim\left(Z_p(K_i)/B_p(K_j)\right) \\ &= \dim\left(Z_p(K_i)/(Z_p(K_i) \cap B_p(K_j))\right) \\ &= \dim\left(Z_p(K_i)\right) - \dim\left(Z_p(K_i) \cap B_p(K_j)\right) \end{aligned} \tag{0.5}$$

By definition, the boundary and cycle groups $B_p(K_j)$ and $Z_p(K_i)$ are subspaces of the operators $\partial_p(K_i)$ and $\partial_{p+1}(K_j)$, yielding:

$$\beta_p^{i,j} = \dim\big(\,\mathrm{Ker}(\partial_p(K_i))\,\big) - \dim\big(\,\mathrm{Ker}(\partial_p(K_i)) \cap \mathrm{Im}(\partial_{p+1}(K_j))\,\big) \tag{0.6}$$

Now, consider computing $\beta_p^{i,j}$ via (0.6) from matrix representatives $\partial_p \in \mathbb{F}^{n \times m}$. Since the nullity of an operator may be reduced to a rank computation, the complexity of first term may be reduced to the complexity of computing the rank of a (sparse) $n \times m$ matrix. In contrast, the second term—the persistence term—typically requires finding a basis in intersection of the two subspaces via either column reductions or projection-based techniques. In general, direct methods that accomplish this require $\Omega(N^3)$ time and $\Omega(N^2)$ memory [9].

To illustrate an alternative approach, we will require a key property of persistence. The structure theorem from [12] shows that 1-parameter persistence modules can be decomposed in an *essentially unique* way into indecomposables. One consequence of this result is the Pairing Uniqueness Lemma [8], which asserts that if $R = \partial V$ decomposes the boundary matrix ∂ to a *reduced* matrix $R \in \mathbb{R}^{m \times n}$ using left-to-right column operations, then:

$$R[i,j] \neq 0 \Leftrightarrow \mathrm{rank}(R^{i,j}) - \mathrm{rank}(R^{i+1,j}) + \mathrm{rank}(R^{i+1,j-1}) - \mathrm{rank}(R^{i,j-1}) \neq \tag{0.7}$$

where $R^{i,j}$ denotes the lower-left submatrix defined by the first j columns and the last $m - i + 1$ rows (rows i through m, inclusive). In other words, the existence of non-zero "pivot" entries in R may be inferred entirely from the ranks of certain submatrices of R. As we will use this fact frequently in this paper, we record it formally with a lemma.

Lemma 1. *Given filtration $K_\bullet$ of size $N = |K|$, let $R = \partial V$ denote the decomposition of the filtered boundary matrix $\partial \in \mathbb{F}^{N \times N}$ given in equation (**??**). Then, for any pair (i, j) satisfying $1 \leq i < j \leq N$, we have:*

$$\mathrm{rank}(R^{i,j}) = \mathrm{rank}(\partial^{i,j}) \tag{0.8}$$

Equivalently, all lower-left submatrices of ∂ have the same rank as their corresponding submatrices in R.

An explicit proof of this can be found in [6], though it was also noted in passing by Edelsbrunner [8]. It can be shown by combining the Pairing Uniqueness Lemma with the fact that R is obtained from ∂ via left-to-right column operations, which preserves the ranks of every such submatrix. Lemma 1 is remarkable in that although R is not unique, its non-zero pivots are, and these pivots *define* the persistence diagram. This seems like a minor observation at first, however it is far more general, as recently noted by [1]:

Proposition 1 (Bauer et al. [1]). *Any persistence algorithm which preserves the ranks of the submatrices $\partial^{i,j}(K_\bullet)$ for all $i, j \in [N]$ is a valid persistence algorithm.*

A lesser-known fact that exploits Lemma 1—also pointed out in [6]—is that (0.8) enables the PBN to be written as a sum of ranks of submatrices of ∂_p and ∂_{p+1}:

Proposition 2 (Dey & Wang [6]). *Given a fixed $p \geq 0$, a filtration $K_\bullet$ of size $N = |K_\bullet|$, and any pair $(i, j) \in \Delta_+^N$, the persistent Betti number $\beta_p^{i,j}(K_\bullet)$ at (i, j) is given by:*

$$\beta_p^{i,j}(K_\bullet) = |K_i^p| - \mathrm{rank}(\partial_p^{1,i}) - \mathrm{rank}(\partial_{p+1}^{1,j}) + \mathrm{rank}(\partial_{p+1}^{i+1,j}) \tag{0.9}$$

For completeness, we give our own detailed proof of Proposition 2 in the appendix. By combining Proposition 2 with (**??**), we recover a submatrix-rank-based p-th multiplicity function $\mu_p^R(\cdot)$, which to the authors knowledge was first pointed out by Chen & Kerber [4]:

Proposition 3 (Chen & Kerber [4]). *Given a fixed $p \geq 0$, a filtration $K_\bullet = \{K_i\}_{i\in[N]}$ of size $N = |K|$, and a $R = [i, j] \times [k, l]$ whose indices (i, j, k, l) satisfy $0 \leq i < j \leq k < l \leq N$, the p-th multiplicity μ_p^R of $K_\bullet$ is given by:*

$$\mu_p^R(K_\bullet) = \mathrm{rank}(\partial_{p+1}^{j+1,k}) - \mathrm{rank}(\partial_{p+1}^{i+1,k}) - \mathrm{rank}(\partial_{p+1}^{j+1,l}) + \mathrm{rank}(\partial_{p+1}^{i+1,l}) \tag{0.10}$$

For more geometric intuition of these propositions, see Figure **??**. Note the differences between these two quantities: whereas $\beta_p^{i,j}$ captures points on the diagram that may have unbounded persistence ("essential" classes [8]), the multiplicity function μ_p^R by definition is restricted to classes with bounded persistence[1].

Compared to the classical reduction methods [7, 12], the primary advantage of the rank-based expressions from (0.9)-(0.10) is that they imply the complexity of obtaining either $\beta_p^{i,j}(K_\bullet)$ or $\mu_p^R(K_\bullet)$ may be reduced to the complexity of computing the rank of a set of submatrices of ∂—a fact that actually motivated the rank-based persistence algorithm from Chen et al [4]. Our contributions in this effort stem from the observation that the constitutive terms in these expressions are *unfactored* boundary (sub)matrices—thus, the operation $x \mapsto \partial x$ can be implemented without actually constructing ∂ in memory, enabling the use of e.g. iterative Krylov or subspace acceleration methods [9, 11] for their computation. Indeed, this line of thought suggests other algebraic properties of the rank function—such as invariance under permutations and adjoint multiplication—may simplify these rank-based expressions even further. The rest of the paper explores these questions and their ramifications in detail.

[1] One may always *cone* the filtration to extend μ_p^R to the unbounded case, see [4]

References

[1] Ulrich Bauer, Talha Bin Masood, Barbara Giunti, Guillaume Houry, Michael Kerber, and Abhishek Rathod. Keeping it sparse: Computing persistent homology revised. *arXiv preprint arXiv:2211.09075*, 2022.

[2] Andrea Cerri, Barbara Di Fabio, Massimo Ferri, Patrizio Frosini, and Claudia Landi. Betti numbers in multidimensional persistent homology are stable functions. *Mathematical Methods in the Applied Sciences*, 36(12):1543–1557, 2013.

[3] Frédéric Chazal, Vin De Silva, Marc Glisse, and Steve Oudot. *The structure and stability of persistence modules*, volume 10. Springer, 2016.

[4] Chao Chen and Michael Kerber. An output-sensitive algorithm for persistent homology. In *Proceedings of the twenty-seventh annual symposium on Computational geometry*, pages 207–216, 2011.

[5] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. Stability of persistence diagrams. In *Proceedings of the twenty-first annual symposium on Computational geometry*, pages 263–271, 2005.

[6] Tamal Krishna Dey and Yusu Wang. *Computational topology for data analysis.* Cambridge University Press, 2022.

[7] Herbert Edelsbrunner and John L Harer. *Computational topology: an introduction.* American Mathematical Society, 2022.

[8] Herbert Edelsbrunner, David Letscher, and Afra Zomorodian. Topological persistence and simplification. In *Proceedings 41st annual symposium on foundations of computer science*, pages 454–463. IEEE, 2000.

[9] Gene H Golub and Charles F Van Loan. *Matrix computations.* JHU press, 2013.

[10] Woojin Kim and Facundo Mémoli. Spatiotemporal persistent homology for dynamic metric spaces. *Discrete & Computational Geometry*, 66:831–875, 2021.

[11] Beresford N Parlett. Do we fully understand the symmetric lanczos algorithm yet. *Brown et al*, 3:93–107, 1994.

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