1 Introduction

Motivation Persistent homology is, as of this time of writing, a well-studied mathematical structure. From its unassuming history starting with postniov towers, the body of research created over the past two decades have established perisstence as not only an intrinsic quantity, but a useful tool. From **, ** , to **, applications abound with persistence; see for an survey. PH is more than just a homology inference tool.

Since its inception, a popular application of persistence in data analysis is its use as a featurization tool. In machine learning, featurization is a means of converting various data representations to a vector format amenable for learning and enhanced training. Classical examples include word2vec in natural language processing, Scale-Invariant Feature Transform (SIFT) in computer vision, extended-connectivity fingerprints (ECFs) in used in chemical informatics and molecular modeling, etc. More recent results include transformers... Through no small feat of engineering, many of these techniques have been incrementally improved and adapted throughout the past decades, and tend to do quite well in terms of their efficiency. As such, they have seen widespread-adoption from more scientific fields trying to harness their power. While certainly useful, one of the pitfalls with such featurizations if the difficulty that comes with interpretation. Difficulty in heavy featurization has lead to qualitative comparisons in scientific fields of the featurization outputs. Many are lead by the same equation: exactly what isa featurization tool capturing that is so useful for training?

Persistent homology is, in some sense, a natural tool for featurization. persistence is a stable invariant that comes equipped mathematical guarantees; thus featurization of diagrams can be interpreted as mapping persistence diagrams to Euclidean space in such a way that maximally preserves the topological information conveyed by the diagram. Moreover, we also know persistence diagrams retain some amount of geometry, such as those of curvature sets and the quasi-isometry theorems in distributed persistence. These results suggest an inverse theory related to persistence. Indeed, a recent injectivity result shows that collections of persistence diagrams are sufficient to uniquely characterize data sets in 2- and 3-dimensions[1], establishes persistence as truly an intrinsic description of shape.

Organization In what follows, we introduce a parameterized expression of the persistent Betti number (PBN) invariant that has certain computational advantages. Namely, we show that a simple augmentation of traditional PBN computation leads to a spectral relaxation that $(1-\epsilon)$-approximates the PBN. Moreover, we show that this relaxation is permutation invariant, obeys certain inclusion-exclusion principles, and admits a notion of stability in a certain sense—all properties useful in parameterized settings.

2 Background & Notation

Diagrams are PBNs are Diagrams

Persistence has been viewed from multiple, equivalent perspectives; we recall a few which are illuminating in this effort. Among the first characterizations of the PBNs were given from an algebraic perspective—Carlsson et al. [9] observed that the persistent homology groups over a filtration may be viewed as the standard homology groups of a particular graded module M over a polynomial ring. In [5], Cohen-Steiner et al. give a more discrete perspective by defining the persistence diagram in terms of a *multiplicities*: given a tame function $f : K \to \mathbb{R}$, its homological critical values $\{a_i\}_{i=1}^n$, and an interleaved sequence $\{b_i\}_{i=0}^n$ satisfying $b_{i-1} < a_i < b_i$ for all i, the p-th persistence diagram $\mathrm{dgm}_p(f) \subset \mathbb{R}^2$ of a filtration induced by f is defined as:

$$\mathrm{dgm}_p(K_\bullet) = \{\, (a_i, a_j) : \mu_p^{i,j} \neq 0 \,\} \;\cup\; \Delta \tag{1}$$

where Δ denotes the diagonal, counted with infinite multiplicity, and $\mu_p^{i,j}$ is the *multiplicity function*, defined as:

$$\mu_p^{i,j} = \left(\beta_p^{i,j\text{-}1} - \beta_p^{i,j}\right) - \left(\beta_p^{i\text{-}1,j\text{-}1} - \beta_p^{i\text{-}1,j}\right) \qquad \text{for } 0 < i < j \leq n+1 \tag{2}$$

For M a decomposable persistence module over $\mathbb{R}$, Chazal [3] redefine the multiplicity function by interpreting $\mu_p^\ast$ as a *persistence measure*:

$$\mu_p(R; M) = \mathrm{card}\left(\mathrm{dgm}_p(M)\big|_R \right) \qquad \text{for all rectangles } R \subset \mathbb{R}^2 \tag{3}$$

The measure-theoretic perspective from (3) views μ_p as a kind of integer-valued measure defined over rectangles in the plane. Since μ_p is also defined in terms of PBNs, this perspective gives an indirect way of interpreting [combinations of] PBNs in measure-theoretic terms. Cerri et al. [2] incorporate this interpretation in their work studying the

---
[1]

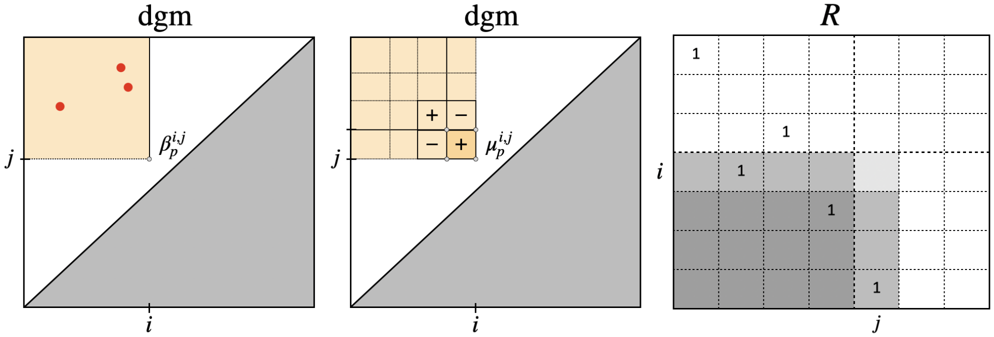

Figure 1: (Left) the persistent Betti number $\beta_p^{i,j}$ counts the number of points (3) in upper left-corner of $\mathrm{dgm}_p(K_\bullet)$. (Middle) The additivity of PBNs can be used to express multiplicity $\mu_p^{i,j}$ of any given box. (Right) The computational interpretation of the Pairing Uniqueness Lemma; in this case $r_R(i,j) = 3 - 2 + 1 - 2 = 0$ yields whether the entry $R[i,j]$ is non-zero.

stability of PBNs in multidimensional persistence. Namely, they show that *proper cornerpoints* in the persistence diagram are points $x = (i,j) \in \Delta_+$ satisfying:

$$x = (i,j) \in \mathrm{dgm}_p(K_\bullet) \iff \mu_p(x) > 0 \iff \min_{\epsilon > 0} \left(\beta_p^{i+\epsilon, j-\epsilon} - \beta_p^{i+\epsilon, j+\epsilon}\right) - \left(\beta_p^{i-\epsilon, j-\epsilon} - \beta_p^{i-\epsilon, j+\epsilon}\right) > 0 \tag{4}$$

One may compare (2) with (4). One of the primary contributions from [2] is a representation theorem expressing the persistent Betti number function as a sum of multiplicity functions. A consequence of this theorem is that any distance between persistence diagrams induces a distance between PBNs, a result which may be used to show that PBNs are stable functions on continuous filtering functions. That is, small changes in continuous scalar-valued filtering functions imply small changes in the corresponding persistent Betti numbers functions. This stability justifies studying PBN functions in continuous, parameterized settings.

3 Main Result

Let $B_p(K_\bullet) \subseteq Z_p(K_\bullet) \subseteq C_p(K_\bullet)$ denote the p-th boundary, cycle, and chain groups of a given filtration $K_\bullet$, respectively. Additionally, let $\partial_p : C_p(K_\bullet) \to C_p(K_\bullet)$ denote the boundary operator sending p-chains to their respective boundaries. With a slight abuse of notation, we also use ∂_p to also denote the filtration boundary matrix with respect to an ordered basis $(\sigma_i)_{1 \leq i \leq m_p}$. The p-th persistent Betti number $\beta_p^{i,j}$ at some index $(i,j) \in \Delta_+^m$ is defined as:

$$\begin{aligned} \beta_p^{i,j} &= \dim(H_p^{i,j}) \\ &= \dim\left(Z_p(K_i)/(Z_p(K_i) \cap B_p(K_j)\right) \\ &= \dim\left(Z_p(K_i)\right) - \dim\left(Z_p(K_i) \cap B_p(K_j)\right) \end{aligned} \tag{5}$$

While $Z_p(K_i) = \mathrm{nullity}(\partial_p(K_i))$ and is thus easily obtained, efficient computation of the intersection term (the persistence part) is a bit more subtle. Zomorodian et al [9] give a procedure to compute a basis for $Z_p(K_i) \cap B_p(K_j)$ via a sequence of boundary matrix reductions; the subsequent Theorem (5.1) reduces the complexity of computing PH groups with coefficients in any PID to that of computing homology groups. However, the standard homology computations require $O(m^2)$ space and $O(m^3)$ time to compute, making the PBN computation no more efficient than the full persistence computation.

Alternatively, both iterative and explicit projector-based methods [1] may also for the intersection term in (5), though these projectors may still be expensive to compute.

In what follows, we outline a different approach to computing (5) that is both simpler and computationally more attractive. To illustrate our approach, we require more notation. If A is a $m \times n$ matrix, let $A^{i,j}$ denote the lower-left submatrix defined by last $m-i+1$ rows (rows i through m, inclusive) and the first j columns. For any $1 \leq i < j \leq m$, define the quantity $r_A(i,j)$ as follows:

$$r_A(i,j) = \mathrm{rank}(A^{i,j}) - \mathrm{rank}(A^{i+1,j}) + \mathrm{rank}(A^{i+1,j-1}) - \mathrm{rank}(A^{i,j-1}) \tag{6}$$

The structure theorem from [9] shows that 1-parameter persistence modules can be decomposed in an *essentially unique* way into indecomposables. Computationally, a consequence of this phenomenon is the Pairing Uniqueness Lemma [6], which asserts that if $R = \partial V$ is the decomposition of the boundary matrix, then:

$$r_R(i,j) \neq 0 \Leftrightarrow R[i,j] \neq 0$$

Since the persistence diagram is derived completely from R, this result suggests that information about a diagram can be obtained through rank computations alone. For a more geometric description of this idea, see the third picture in Figure 1. We record a non-trivial fact that follows from this observation:

Lemma 1 (Dey & Wang [7])**.** *Let $R = \partial V$ denote the matrix decomposition of a given filtered boundary matrix ∂ derived from the associated filtration $K_\bullet$. For any pair (i,j) satisfying $1 \leq i < j \leq m$, we have:*

$$\mathrm{rank}(R^{i,j}) = \mathrm{rank}(\partial^{i,j}) \tag{7}$$

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

Lemma 1 was the essential motivating step used by Chen et al [4] in their rank-based persistence algorithm—the first output-sensitive algorithm given for computing persistent homology of a filtered complex. In fact, Lemma 1 may be further generalized to arbitrary rectangles in Δ_+ via *μ-queries* [4]: box-parameterized rank-based queries that count the number of persistence pairs that intersect a fixed "box" placed in the upper half-plane. Our first result of this effort we show is that this Lemma allows us to write the persistent Betti number as a sum of rank functions.

Proposition 1. *For any fixed $p \geq 0$, let ∂_p denote the p-dimensional boundary matrices of filtration $K_\bullet$ of size $m_p = |K_{(p)}|$. For any pair $(i,j) \in ([m_p],[m_{p+1}])$, the persistent Betti number $\beta_p^{i,j}$ at (i,j) is given by:*

$$\beta_p^{i,j} = \mathrm{rank}(I_p^{1,i}) - \mathrm{rank}(\partial_p^{1,i}) - \mathrm{rank}(\partial_{p+1}^{1,j}) + \mathrm{rank}(\partial_{p+1}^{i+1,j}) \tag{8}$$

where $I_p^{1,i}$ denotes the first i columns of the $m_p \times m_p$ identity matrix.

A detailed proof using Lemma 1 is given in the appendix. The main utility this proposition provides is that it enables the persistent Betti number as a combination of rank computations performed directly on the *unfactored* dimension p and $(p+1)$ boundary matrices. We dedicate the rest of the paper to exploring the consequences of this fact.

To get some intuition on what the structure and size of these matrices, we include a picture of each of the terms in Equation (8). Consider a filtration $K_\bullet$ with m simplices $\sigma_1, \sigma_2, \ldots, \sigma_m$, constructed from a p-dimensional complex

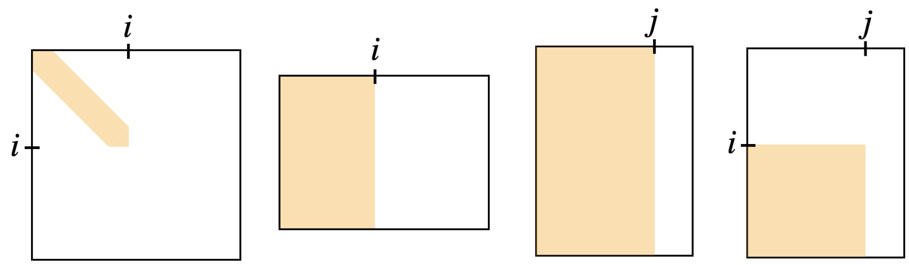

Figure 2: The four matrices whose ranks yield $\beta_p^{i,j}$ in the same order as given in (8). Each solid portion represents (sparse) blocks of non-zero entries, while each white portion is zero. Observe $\partial_{p+1}^{i+1,j}$ can be obtained by intersecting the non-zero entries of $\partial_{p+1}^{1,j}$ with the non-zero entries in the complement of $\partial_p^{1,i}$.

K. As a simplex σ enters its filtration after it's boundary $\partial_p(\sigma)$, observe ∂ is upper-triangular. Moreover, the sparsity of ∂ (and all of its principle submatrices) has storage complexity:

$$\mathrm{nnz}(\partial) \sim O(m \log m) \tag{9}$$

This follows from the simple observation that since each column of ∂_p^* contains exactly $p+1$ non-zero entries, $\mathrm{nnz}(\partial) \sim O((p+1)m)$, and since a p-simplex has $2^{p+1}-1$ faces, $p \leq \log(m+1)-1$. Indeed, the relation $\partial_{p+1}^{i+1,j} \subseteq \partial_{p+1}^{1,j}$ implies the largest number of non-zeros in any matrix in (8) is $O(\max\{p\, i, (p+1)\, j\})$. As a result, we have the following corollary:

Corollary 1. *Given two indices* $(i, j) \in \Delta_+^m$ *and a filtration* $K_\bullet$ *with* $n = |K_i^{(p)}|$ *and* $m = |K_j^{(p+1)}|$*, computing* $\beta_p^{i,j}$ *can be done in time and storage complexity* $O(\max\{R_\partial(n, p), R_\partial(m, p+1)\})$ *where* $R_\partial(a, b)$ *is the complexity of computing the rank of a b-dimensional boundary matrix with* $n\times$

In the next section, we extend these complexity statements to parameterized settings in a natural way. We address the storage and time complexity of $R_\partial(\cdot, \cdot)$ in section 2.

A Parameterized Boundary Matrix Relaxation

One advantage of expressing the PBN via equation (8) is that certain properties of rank function may be exploited in *parameterized* settings, i.e. settings where the scalar-valued filter function and corresponding filtration belong to a parametric family. One simple such property is permutation invariance: given any square matrix $A \in \mathbb{R}^{n\times n}$, $\mathrm{rank}(A) = \mathrm{rank}(P^T AP)$ for any permutation matrix P. Observe the boundary matrices ∂_p in (8) need not be in filtration order induced by $(K_\bullet, f)$ to be evaluated—the rank function is permutation invariant so long as the involved matrices have the same essential non-zero pattern. Thus, unlike the vineyards algorithm [6]—which requires a $\approx O(m^2)$ maintenance procedure to simulate persistence across a homotopy[2]—PBNs need no such procedure even in parameterized settings. We elaborate in the following section.

Recall that the boundary operator ∂_p for a finite simplicial filtration $K_\bullet$ with $m = |C_p(K_\bullet)|$ and $n = |C_{p-1}(K_\bullet)|$ can be represented by an $(n\times m)$ boundary matrix ∂_p whose columns and rows correspond to p-simplices and $(p-1)$-simplices, respectively. The entries of ∂_p depend on the choice of $\mathbb{F}$; in general, after orientating the simplices of K arbitrarily, they have the form:

$$\partial_p[k, l] = \begin{cases} c(\sigma_j) & \text{if } \sigma_l \in \partial_p(\sigma_k) \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $c(\sigma_*) \in \mathbb{F}$ is an arbitrary constant satisfying $c(\sigma) = -c(\sigma')$ if σ and σ' are opposite orientations of the same simplex, typically set to ± 1. In what follows, we assume a fixed orientation is given on K, and write $\pm c(\sigma)$ to indicate the sign of $c(\sigma)$ depends on the orientation of σ.

The typical input to the persistence computation is a fixed complex/filter pair $(K_\bullet, f)$, from which the corresponding boundary matrices are constructed. These filter functions typically have a particular geometrical interpretation. For example, a common scenario in practice is let $K_\bullet$ via a Rips filtration a metric space (X, d_X); in this case $f : \mathcal{P}(X) \to \mathbb{R}_+$ is given as the diameter $f(\sigma) = \max_{x,x'\in\sigma} d_X(x, x')$. In many applications it is of interest to study such filter function in parameterized settings, e.g. given some set of parameters $\mathcal{H}$, the goal is to understand a given topological invariant at many parameters $h \in \mathcal{H}$, i.e. treat $f : \mathcal{P}(X) \times \mathcal{H} \to \mathbb{R}_+$. We include several example in section **??**.

Suppose that instead of being given a fixed filtration $(K_\bullet, f)$, the filter was parameterized $f : \mathcal{H} \times K \to \mathbb{R}$ and you wanted to compute $\beta_p^{i,j}$ over $\mathcal{H}$. We give several instantiations of this in section **??**.

Definition 1 (Parameterized boundary matrix)**.** *Let* X *denote a data set of interest of size* $|X| = n$*, equipped with parameterized filtering function* $f : \mathcal{P}(X) \times \mathcal{H} \to \mathbb{R}$*. Define* $(\mathcal{P}(X), \preceq^*)$ *be a fixed linear extension of the face poset of the standard* $(n$*-*$1)$*-simplex. For fixed* $i, j \in \mathbb{F}$*, define the* $\mathcal{H}$-parameterized p-th boundary matrix $\hat{\partial}_p^{i,j}(t)$ at scale (i, j) *to be the* $\binom{n}{p} \times \binom{n}{p+1}$ *matrix ordered by* $\preceq^*$ *for all* $h \in \mathcal{H}$*, and whose entries* (k, l) *satisfy:*

$$\hat{\partial}_p^{i,j}(h)[k, l] = \begin{cases} \pm S_{i,j}(\sigma_k, \sigma_l) & \textit{if } \sigma_k \in \partial_p(\sigma_l) \\ 0 & \textit{otherwise} \end{cases} \tag{11}$$

where $S_{i,j} : \mathcal{P}(X) \times \mathcal{P}(X) \to \{0, 1\}$ *is a* step *function that accepts a face/coface pair* (σ_k, σ_l) *and returns a* 1 *if* $f(\sigma_k) \geq i$ *and* $f(\sigma_l) \leq j$*, and* 0 *otherwise.*

The result of definition 1 is that, in parameterized settings, the PBN can be written in essentially the same form as (8). To simplify the notation, we write $A^x = A^{1,x}$ for the setting where only columns up to x of A are being selected, and let $q = p + 1$. We also write $r(A) = \mathrm{rank}(A)$.

$$\hat{\beta}_p^{i,j} : \mathcal{H} \to \mathbb{N} \tag{12}$$

$$h \mapsto |K_i^{(p)}(h)| - (\mathrm{r} \circ \partial_p^i)(h) - (\mathrm{r} \circ \partial_q^j)(h) + (\mathrm{r} \circ \partial_q^{i+1,j})(h) \tag{13}$$

One remark is in order: note that although definition 1 specifies $\hat{\partial}_p^{i,j}$ as a full $\binom{n}{p} \times \binom{n}{p+1}$ matrix, implying a memory complexity of $O((p+1)n^{p+1})$ for all $h \in \mathcal{H}$, we remark that there is no need to fully allocate this much memory

[2] Strictly speaking, the bound $O(m^2)$ assumes the homotopy changes each filtration value in a monotone way throughout the homotopy.

as the rows/columns corresponding to the set of $p/p+1$ face/coface pairs (σ_k, σ_l) with $f(\sigma_l) < i$ or $f(\sigma_k) > j$ are entirely 0. Indeed, as we will show below, it is enough to have access to the simplices $K_j^{(p+1)}(h)$.

Laplacian Connection We now show a few properties that $\hat{\partial}_p^*(t)$ exhibits which is advantageous for parameterized families. The first such property is a simple parameterized relaxation of PBN:

$$\begin{aligned}
\hat{\beta}_p^{i,j} &= |K_i^{(p)}| - \mathrm{rank}(\partial_p^i) - \mathrm{rank}(\partial_q^j) + \mathrm{rank}(\partial_q^{i+1,j}) && (14)\\
&= |K_i^{(p)}| - \mathrm{rank}((\partial_p^i)(\partial_p^i)^T) - \mathrm{rank}((\partial_q^j)(\partial_q^j)^T) + \mathrm{rank}((\partial_q^{i+1,j})(\partial_q^{i+1,j})^T) && (15)\\
&= |K_i^{(p)}| - \mathrm{rank}(L_p^i) - \mathrm{rank}(L_q^j) + \mathrm{rank}(L_q^{i+1,j}) && (16)
\end{aligned}$$

Computation In this section we discuss a few iterative methods for finding the eigenvalues of a Hermitian matrices. In particular, we focus on the positive semi-definite, diagonally dominant (DD) and strictly diagonally dominant (SDD) cases. We will give analysis of certain aspects of it which we will need.

The success of iterative methods at extracting spectral-information from large, sparse matrices in the past three decades is unprecedented. Core to these methods is the Arnoldi iteration, whose specialization to symmetric matrices yields the Lanczos method []. Although discovered before the widely successful direct methods (Householder transformations, QR factorization, etc.), the Lanczos algorithm became the staple for applications where the underlying full orthogonal decompositon of system was too large to fit into memory.

The means by which the Lanczos method estimates eigenvalues is by projecting onto successive Krylov subspaces. Given a large, sparse, symmetric $n \times n$ matrix A with eigenvalues $\lambda_1 \geq \lambda_2 > \cdots \geq \lambda_r > 0$ and a vector v, the order-k Krylov subspaces are the spaces spanned by:

$$\mathcal{K}_k(A, v) := \mathrm{span}\{v, Av, A^2v, \ldots, A^{k-1}\} = \mathrm{range}(K_k(A, v)) \tag{17}$$

where $K_k(A, v) = [v \mid Av \mid A^2v \mid \cdots \mid A^{k-1}]$ are the corresponding Krylov matrices. To generate an orthonormal basis for $\mathcal{K}_k(A, v)$, the Lanczos method proceeds by constructing a reduced QR factorizations of $K_k(A, v) = Q_k R_k$, for all $k = 1, 2, \ldots, n$. Due orthogonality of Q_k, when A is symmetric and real, we have $q_i^T A q_j = q_j^T A^T q_i = 0$ for $i > j + 1$, implying the corresponding projections $T_k = Q_k^T A Q_k$ have a tridiagonal structure:

$$T_k = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_{k-1} \\ & & & \beta_{k-1} & \alpha_k \end{bmatrix}, \beta_i > 0, i = 2, \ldots, k \tag{18}$$

Unlike the spectral decomposition $A = U\Lambda U^T$, which diagonalizes A into a form such that any other decomposition $A = \tilde{U}\Lambda\tilde{U}^T$ is related by a similarity transform, there is no canonical T_k due to the arbitrary choice of v. Instead, the Lanczos method exploits a particular connection between the iterates $K_k(A, v)$ and the tridiagonalization of A: if $Q^T A Q = T$ is tridiagonal and $Q = [\, q_1 \mid q_2 \mid \cdots \mid q_n \,]$ is an $n \times n$ orthogonal matrix $QQ^T = I_n$, then:

$$K_n(A, q_1) = QQ^T K_n(A, q_1) = Q[\, e_1 \mid Te_1 \mid T^2 e_1 \mid \cdots \mid T^{n-1} e_1 \,] \tag{19}$$

is the QR factorization of $K_n(A, q_1)$—that is, Q may be generated completely by tridiagonalizing A with respect to an orthogonal matrix Q whose first column vector is q_1. More explicitly, the Implicit Q Theorem [8] asserts that if $A, T \in \mathbb{R}^{n \times n}$ where T is upper Hessenburg and has only positive elements on its first subdiagonal and there exists an orthogonal matrix Q such that $Q^T A Q = T$, then Q and T are *uniquely* determined by A and the first column of Q. Thus, at each step $j = 1, 2, \ldots, k$, we may restrict and project A to a its j-th Krylov subspace T_j via:

$$AQ_j = Q_j T_j + \beta_j q_{j+1} e_j^T \tag{20}$$

where $q_{j+1} \in \mathbb{R}^n$ is an orthogonal vector, and e_j is the standard j identity vector, and $\beta_j > 0$. Equating the j-th columns on each side of (20) yields a three-term recurrence:

$$Aq_j = \beta_{j-1} q_{j-1} + \alpha_j q_j + \beta_j q_{j+1} \tag{21}$$

where $\alpha_j = q_j^T A q_j$, $\beta_j = \|r_j\|_2$, $r_j = (A - \alpha_j I)q_j - \beta_{j-1} q_j$, and $q_{j+1} = r_j / \beta_j$. Thus, if q_{k-1}, β_k and q_k are known, then α_k, β_{k+1} and q_{k+1} are completely determined. The sequential process by which (21) is used to construct T_k is called the *Lanczos iteration.* In principle, if the Lanczos iteration doesn't break down due to rounding errors (T_k is full rank), then the characteristic polynomial of T_k is the unique polynomial $p^k \in P^k$ that achieves []:

$$\|p^k(A)b\| = minimum \tag{22}$$

In fact, there is more we can say: if $S = \mathcal{K}_k(A, q_1)$, then applying the Lanczos iteration yield the *Ritz pairs* (θ, y) of A, which are the pairs satisfying $w^T(Ay - \theta y) = 0$ for all $w \in S$. The essential result from [] is that the eigenvalues $\Lambda(T_k) = \{\theta_1, \theta_2, \dots, \theta_k\}$ are 'optimal' in the sense that $T_k = B$ is the matrix that minimizes $\|AQ_k - QkB\|_2$ over the space of all $k \times k$ matrices. Thus, if $\mathrm{rank}(A) = r$, the eigenvalues of T_k for some $k < r$ tend to converge quickly to the largest eigenvalues of A. Moreover, if for some $k < n$ we encounter $r_k = 0$, then $\mathrm{range}(Q_k) = \mathcal{K}_k(A, q_1)$ is invariant for A, i.e. $k = \mathrm{rank}(A)$ and the iteration stops.

The Lanczos iteration is very computationally attractive due to the fact that each step consists of a matrix-vector multiplication, and inner product, and a few vector operations. If one can efficiently compute $v \mapsto Av$ because A is sparse and/or has special structure, then the Lanczos method may require significantly less than the $O(n^3)$ operations needed by the more direct methods. Indeed, the three-term recurrence from (21) implies the Lanczos iteration of A may be carried out with just three $O(n)$-sized vectors. Moreover, if $A \in \mathbb{R}^{n \times n}$ is a symmetric rank-r matrix with an average of ν nonzeros per row, then approximately $(2\nu + 8)n$ flops are needed for a single Lanczos step, implying a $O(n\nu r)$ time complexity for a single iteration [8]. Once T_k has been obtained for sufficient $k > 0$, the corresponding spectrum $\Lambda(T_n) = \{\,\theta_i\,\}$ may be obtained in $O(n \log n)$ time [].

The special structure of ∂_p actually admits a few simplications compared to the traditional sparse matrix. In particular, the operator ∂_p need not be represented explicitly in memory. To see this, observe each column of ∂_p corresponds to a p-chain of the form (10), which is a constant up to difference in sign. Thus, evaluating $y = \partial_p^T v$ reduces to computing the locations $(k_1, k_2, \dots, k_{p+1})$:

$$y_i = y_i \pm S_{i,j}(\sigma_k, \sigma_l) \tag{23}$$

he face relation To make this more precise, we start with a useful lemma:

Lemma 2. *For any fixed $p \geq 0$, given simplicial complex K with m (n, respectively) simplices of dimension p ($p-1$, respectively) and an arbitrary vector v of size $|v| = n$, the operation $v \mapsto \partial_p \partial_p^T v$ can be evaluated $O(\max\{m, n\})$ time and $O(\max\{m, n\})$ memory.*

From this lemma, we have the following result:

Proposition 2 ([]). *Given a simplicial complex K with $m = |K^{(p+1)}|$ and $n = |K^{(p)}|$. Without loss of generality, assume $n < m$, and that $\mathrm{rank}(\partial_p) = r$. Moreover, assume we may carry out the computation $v \mapsto \partial_p \partial_p^T v$ with coefficients in $\mathbb{F} = \mathbb{R}$ in exact arithmetic in $O(m)$ time. Then the Lanczos iteration from [] yields the quantity:*

$$\dim(B_{p-1}(K; \mathbb{R})) = \mathrm{rank}(\partial_p(K)) \tag{24}$$

in $O(mr)$ time and $O(m)$ storage complexity.

A proof of this proposition is given in the appendix. Extending from this, corollary **??** implies p-th PBN at index $(i, j) \in \Delta_+^m$ can be computed in... note that for any $p \geq 1$, this is significant reduction in complexity compared to e.g. the $O(m^3)$ reduction algorithm. Indeed, if K has n vertices, the $\Theta(m^3)$ complexity of the reduction algorithm implies computing the p-th persistence diagram requires reducing ∂_{p+1}; one deduces loose though asymptotically accurate bounds as $O(n^3(p + 2))$ operations, i.e. $O(n^9)$ for $p = 1$, $O(n^{12})$ for $p = 2$, etc. In contrast, combining proposition 2 with corollary **??**, we have that the PBN at index i, j may be compute in... a substantial difference indeed.

Obviously, the assumption of exact arithmetic is too strong to be of any practical use. Round-off errors plague the simple Lanczos iteration.

The iteration typically converges geometrically and can be stopped as soon as the desired accuracy is reached.

4 Applications

A Appendix

Proofs

Proof of Lemma 1

Proof. The Pairing Uniqueness Lemma [7] asserts that if $R = \partial V$ is a decomposition of the total $m \times m$ boundary matrix ∂, then for any $1 \leq i < j \leq m$ we have $\mathrm{low}_R[j] = i$ if and only if $r_\partial(i,j) = 1$. As a result, for $1 \leq i < j \leq m$, we have:

$$\mathrm{low}_R[j] = i \iff r_R(i,j) \neq 0 \iff r_\partial(i,j) \neq 0 \tag{25}$$

Extending this result to equation (7) can be seen by observing that in the decomposition, $R = \partial V$, the matrix V is full-rank and obtained from the identity matrix I via a sequence of rank-preserving (elementary) left-to-right column additions. $\square$

Proof of Proposition 1

Proof. We first need to show that $\beta_p^{i,j}$ can be expressed as a sum of rank functions. Note that by the rank-nullity theorem, so we may rewrite (5) as:

$$\beta_p^{i,j} = \dim\left(C_p(K_i)\right) - \dim\left(B_{p-1}(K_i)\right) - \dim\left(Z_p(K_i) \cap B_p(K_j)\right)$$

The dimensions of groups $C_p(K_i)$ and $B_p(K_i)$ are given directly by the ranks of diagonal and boundary matrices, yielding:

$$\beta_p^{i,j} = \mathrm{rank}(I_p^{1,i}) - \mathrm{rank}(\partial_p^{1,i}) - \dim\left(Z_p(K_i) \cap B_p(K_j)\right)$$

To express the intersection term, note that we need to find a way to express the number of p-cycles born at or before index i that became boundaries before index j. Observe that the non-zero columns of R_{p+1} with index at most j span $B_p(K_j)$, i.e $\{\, \mathrm{col}_{R_{p+1}[k]} \neq 0 \mid k \in [j] \,\} \in \mathrm{Im}(\partial_{p+1}^{1,j})$. Now, since the low entries of the non-zero columns of R_{p+1} are unique, we have:

$$\dim(Z_p(K_i) \cap B_p(K_i)) = |\Gamma_p^{i,j}| \tag{26}$$

where $\Gamma_p^{i,j} = \{\, \mathrm{col}_{R_{p+1}[k]} \neq 0 \mid k \in [j], 1 \leq \mathrm{low}_{R_{p+1}}[k] \leq i \,\}$. Consider the complementary matrix $\bar{\Gamma}_p^{i,j}$, given by the non-zero columns of R_{p+1} with index at most j that are not in $\Gamma_p^{i,j}$, i.e. the columns satisfying $\mathrm{low}_{R_{p+1}}[k] > i$. Combining rank-nullity with the observation above, we have:

$$|\bar{\Gamma}_p^{i,j}| = \dim(B_p(K_j)) - |\Gamma_p^{i,j}| = \mathrm{rank}(R_{p+1}^{i+1,j}) \tag{27}$$

Combining equations (26) and (27) yields:

$$\dim(Z_p(K_i) \cap B_p(K_j)) = |\Gamma_p^{i,j}| = \dim(B_p(K_j)) - |\bar{\Gamma}_p^{i,j}| = \mathrm{rank}(R_{p+1}^{1,j}) - \mathrm{rank}(R_{p+1}^{i+1,j}) \tag{28}$$

Observing the final matrices in (28) are *lower-left* submatrices of R_{p+1}, the final expression (8) follows by applying Lemma 1 repeatedly. $\square$

Proof of boundary matrix properties

Proof. First, consider property (1). For any $t \in T$, applying the boundary operator ∂_p to $K_t = \mathrm{Rips}_\epsilon(\delta_\mathcal{X}(t))$ with non-zero entries satisfying (10) by definition yields a matrix ∂_p satisfying $\mathrm{rank}(\partial_p) = \dim(\mathrm{B}_{p-1}(K_t))$. In contrast, definition (1) always produces p-boundary matrices of Δ_n; however, notice that the only entries which are non-zero are precisely those whose simplices σ that satisfy $\mathrm{diam}(\sigma) < \epsilon$. Thus, $\mathrm{rank}(\partial_p^t) = \dim(\mathrm{B}_{p-1}(K_t))$ for all $t \in T$. < (show proof of (2))> Property (3) follows from the construction of ∂_p and from the inequality $\|A\|_2 \leq \sqrt{m}\|A\|_1$ for an $n \times m$ matrix A, as $\|\partial_p^t\|_1 \leq (p+1)\,\epsilon$ for all $t \in T$.

$\square$

Dynamic Metric Spaces Consider an $\mathbb{R}$-parameterized metric space $\delta_X = (X, d_X(\cdot))$ where X is a finite set and $d_X(\cdot) : \mathbb{R} \times X \times X \to \mathbb{R}_+$, satisfying:

1. For every $t \in \mathbb{R}, \delta_X(t) = (X, d_X(t))$ is a pseudo-metric space[3]

[3]This is required so that if one can distinguish the two distinct points $x, x' \in X$ incase $d_X(t)(x,x') = 0$ at some $t \in \mathbb{R}$.

2. For fixed $x, x' \in X$, $d_X(\cdot)(x, x') : \mathbb{R} \to \mathbb{R}_+$ is continuous.

When the parameter $t \in \mathbb{R}$ is interpreted as *time*, the above yields a natural characterization of a "time-varying" metric space. More generally, we refer to an $\mathbb{R}^h$-parameterized metric space as *dynamic metric space*(DMS). Such space have been studied more in-depth [] and have been shown...

Application: Time-varying Let $\delta_{\mathcal{X}}$ denote an T-parameterized metric space $\delta_{\mathcal{X}}(\cdot) = (X, d_X(\cdot))$, where $d_X : \mathrm{T} \times X \times X \to \mathbb{R}_+$ is called a *time-varying metric* and X is a finite set with fixed cardinality $|X| = n$. δ_X as called a *dynamic metric space* (DMS) iff $d_X(\cdot)(x, x')$ is continuous for every pair $x, x' \in X$ and $\delta_{\mathcal{X}}(t) = (X, d_X(t))$ is a pseudo-metric space for every $t \in \mathrm{T}$. For a fixed $t \in \mathrm{T}$, the Rips complex at scale $\epsilon \in \mathbb{R}$ is the abstract simplicial complex given by

$$\mathrm{Rips}_\epsilon(\delta_{\mathcal{X}}(t)) := \{\sigma \subset X : d_X(t)(x, x') \leq \epsilon \text{ for all } x, x' \in \sigma\} \tag{29}$$

As before, the family of Rips complexes for varying $\epsilon > 0$ yields a filtration whose inclusion maps induce linear maps at the level of homology. The time-varying counterpart is analogous. In this context, we write the p-th persistent Betti number with respect to fixed values $i, j \in I$ as a function of $t \in \mathrm{T}$:

$$\beta_p^{i,j}(t) = \left(\dim \circ \mathrm{H}_p^{i,j} \circ \mathrm{Rips} \circ \delta_{\mathcal{X}}\right)(t) \tag{30}$$

References

[1] Adi Ben-Israel. Projectors on intersection of subspaces. *Contemporary Mathematics*, 636:41–50, 2015.

[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] David Cohen-Steiner, Herbert Edelsbrunner, and Dmitriy Morozov. Vines and vineyards by updating persistence in linear time. In *Proceedings of the twenty-second annual symposium on Computational geometry*, pages 119–126, 2006.

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

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

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

A Boundary matrix factorization

Definition 2 (Boundary matrix decomposition)**.** *Given a filtration $K_\bullet$ with m simplices, let ∂ denote its $m \times m$ filtered boundary matrix. We call the factorization $R = \partial V$ the* boundary matrix decomposition *of ∂ if:*

I1. V is full-rank upper-triangular

I2. R satisfies $\mathrm{low}_R[i] \neq \mathrm{low}_R[j]$ iff its i-th and j-th columns are nonzero

where $\mathrm{low}_R(i)$ denotes the row index of lowest non-zero entry of column i in R or null *if it doesn't exist. Any matrix R satisfying property (I2) is said to be* reduced*; that is, no two columns share the same low-row indices.*