Persistent Homology Computation

Padma Ragaleena Tanikella

GTDAML 2023

Introduction and initial set up

- In Topological data analysis, one works with **point-cloud data set** (X, d) i.e. the data set X modeled as a finite metric space.
- We assume X to be a discrete sample from a larger topological space S whose shape we are interested in.

Aim: Utilize X in-order to obtain "shape" information about S.

- $|X| < \infty \implies X$ is equipped with the discrete topology which is neither topologically useful nor interesting.
- To overcome this problem, we store shape information in X using an algebraic object called simplicial complex which enables us to use tools from Algebraic Topology to extract topological information.

Why use a simplicial complex?

- We assume the underlying topological space S is triangulable , hence it makes sense to associate it with a simplicial complex that allows us to use the area of simplicial homology within Algebraic topology to extract topological information.
- Although the fundamental group of a space contains all the topological information, computing the fundamental group of spaces is often very difficult.
- Therefore at the cost of losing topological information, one prefers simplicial homology groups, as they are easier to compute under triangulization assumption.

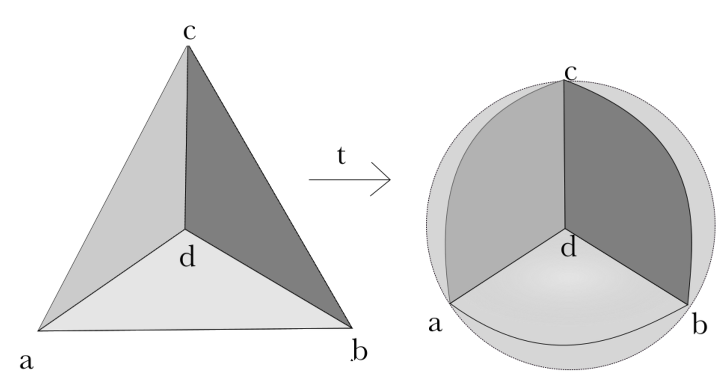


Figure 1. 2-D sphere is triangulable.

Associating data with a Simplicial Complex

- Our goal is to build a simplicial complex on data set X such that the homology of the complex approximates the homology of the original space S.

Complex	Size	Guarantee
Čech	$\mathcal{O}(2^n)$	Nerve Theorem
Vietoris-Rips	$\mathcal{O}(2^n)$	Approximates Čech
Alpha		Nerve Theorem
Witness		In Euclidean space

Table 1. Some ways of associating a simplicial complex to data

- The worst case size of Čech and Rips complex is exponential. However, the process of constructing the complex takes exponential for Čech and polynomial time, $\mathcal{O}(n^3)$, for Rips.

Persistent Homology Basics

X = Simplicial complex.

- The k-th chain group $C_k(X; R)$ is a free abelian group if the set of coefficients R come from a $\mathbb{Z}$. This in turn induces a series of group homomorphisms $\partial_k : C_k(X; R) \to C_{k-1}(X; R)$.
- On the other hand, if the coefficients R come from a field then $C_k(X; R)$ is a vector space over R which induce a series of linear transformations $\partial_k : C_k(X; R) \to C_{k-1}(X; R)$.
- Define $Z_k(X; R) = \ker(\partial_k)$ and $B_k(X; R) = \operatorname{Im}(\partial_{k+1})$.
- k-th homology group = $H_k(X; R) = \dfrac{Z_k(X; R)}{C_k(X; R)}$

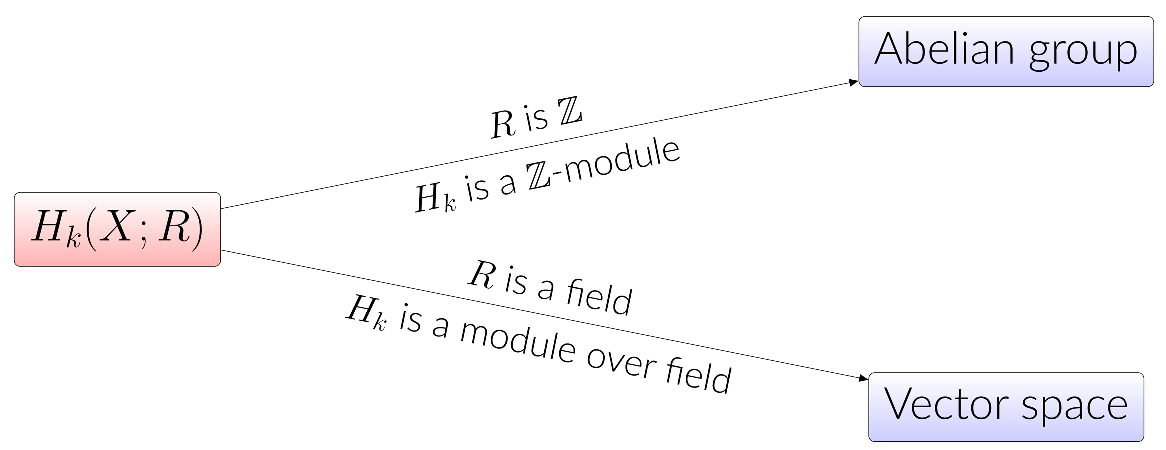


Figure 2. The choice of coefficient PID (eg. $\mathbb{Z}$, field) influences the structure of $H_k(X; R)$. Note that free abelian groups have properties similar to vector spaces.

Reduction algorithm to compute $H_k(X; R)$

- The algorithm is originally due to Poincaré. Although Poincaré was unaware, the algorithm is equivalent to Smith's algorithm published earlier.
- Let $R = \mathbb{Z}$ for simplicity.
- The Smith normal form of a $r \times c$ matrix M over $\mathbb{Z}$ is a matrix product

$$\mathbf{S_{r\times r}}\,\tilde{\mathbf{M}}\,\mathbf{T_{c\times c}} = \mathbf{M_{r\times c}} \text{ where } \tilde{\mathbf{M}} = \begin{bmatrix} \operatorname{Diag}(d_1, \cdots, d_m) & 0_{m\times(c-m)} \\ 0_{(r-m)\times m} & 0_{(r-m)\times(c-m)} \end{bmatrix}$$

such that matrices $\mathbf{S}, \mathbf{T}$ are invertible and $d_i | d_{i+1}$ for all i.

- Reduction algorithm computes $H_k(X; R), B_k(X; R)$ and , $Z_k(X; R)$ by reducing boundary matrix to Smith Normal form. Let

$$\partial_k = S_k \tilde{\partial}_k T_k \ \forall k \text{ such that } \operatorname{rank}(\partial_k) = m_k, \text{ then,}$$

$$H_k(X; \mathbb{Z}) \cong \mathbb{Z}^{n_k - m_k - m_{k+1}} \oplus \bigoplus_{i=1}^{m_{k+1}} (\mathbb{Z}/d_{ki}\mathbb{Z})$$

Structure theorem for modules over PID

Theorem: Let $M \neq \emptyset$ be a finitely generated module over a PID R.

1. Then $\dfrac{M}{\operatorname{Tor}(M)}$ is free and there exists a free submodule F in M such that
$$M \cong \operatorname{Tor}(M) \oplus F \text{ where } F \cong \frac{M}{\operatorname{Tor}(M)}$$
2. For $r \in \mathbb{N} \cup \{0\}$ and $a_1, \ldots, a_m \in R$ (which are not units in R) satisfying $a_1 | a_2 | \cdots | a_m$, we have
$$M \cong R^r \oplus \frac{R}{\langle a_1 \rangle} \oplus \cdots \frac{R}{\langle a_m \rangle}$$

The structure theorem completely characterizes the structure of finitely generated modules over PID.

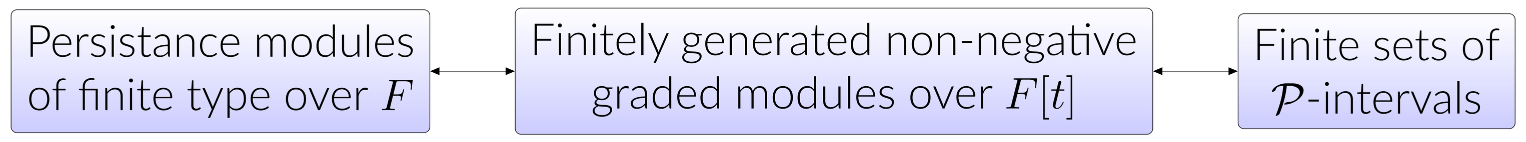
Barcodes: Alternate way of storing homology information

- **Idea:** Combine the homology of all complexes in the filteration and view it as a single algebraic structure called the Persistence Module.
- Let $\{X_i\}_{i=1}^n$ be a filtered simplicial complex. Then $\mathcal{M} = \{(H_k(X_i))_{i=1}^n, (f_{ij})_{\forall i,j}\}$ defines a persistence module.
- Given a persistence module $\mathcal{M}_1 = \{M_i, \phi_i\}_{i \geq 0}$ where each M_i is an R-module, there exists corresponding well-defined $R[t]$-module $\alpha(\mathcal{M}_1) = \bigoplus_{i \geq 0} M_i$

Ring R	$R[t]$	Can we characterize $\alpha(\mathcal{M}_1)$?
PID, E.g. $\mathbb{Z}$	Not a PID	Known to be a difficult classification problem
Field, E.g. $\mathbb{Z}/p\mathbb{Z}$	PID	Use the Structure theorem

Table 2. Significance of using field coefficients

- Define $\mathcal{P}$-interval to be an ordered pair (i, j) such that $0 \leq i < j \in \mathbb{Z} \cup \{\infty\}$.
- Let $R = F$ be a field. Now, associate an $F[t]$-module to a finite set of $\mathcal{P}-$intervals using the map $\mathcal{Q}$:
- $\mathcal{Q}(i, j) = \sum_i \dfrac{F[t]}{(t^{j-i})}$ (Torsion part)
- $\mathcal{Q}(i, \infty) = \sum_i F[t]$ (Free part)
- Let $\mathcal{S}$ be set of $\mathcal{P}$-intervals, then $\mathcal{Q}(\mathcal{S}) = \bigoplus_{(i,j)\in\mathcal{S}} \mathcal{Q}(i, j)$
- **Corollary**: The correspondence $\mathcal{S} \longrightarrow \mathcal{Q}(\mathcal{S})$ defines a bijection between finite sets of $\mathcal{P}$-intervals and finitely generated graded modules over $F[t]$.



The Standard algorithm for $R = \mathbb{Z}/2\mathbb{Z}$

- The algorithm gives a way of obtaining barcodes, i.e. a set of $\mathcal{P}$ intervals for a filtered simplicial complex over field F without having to construct the persistence module.
- Place a total order on $X = \{\sigma_1, \cdots, \sigma_n\}$.
- Define: $\delta(i, j) = \mathbb{I}(\sigma_i \leq \sigma_j$ of codim 1) and $\operatorname{low}(j) = i := \operatorname{argmax}\{x | \delta(x, j) \neq 0\}$.
- We say that δ is reduced if low is injective. Note that $\operatorname{low}(j)$ is undefined when $\delta(x, j) = 0 \ \forall\, x \in \{1, \cdots, n\}$.

Once low is reduced, following is how we get barcodes:

1. If $\operatorname{low}(j) = i \implies$ pair σ_j with σ_i which corresponds to $[\operatorname{dg}(\sigma_i), \operatorname{dg}(\sigma_j))$
2. If $\operatorname{low}(j) =$ undefined, then it corresponds to $[\operatorname{dg}(\sigma_j), \infty)$

References

[1] Jeff Ericson.
Lecture notes in Computational Topology (CS598: UIUC), Fall 2009.

[2] Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington.
A roadmap for the computation of persistent homology.
EPJ Data Science, 6(1), August 2017.

[3] Afra Zomorodian and Gunnar Carlsson.
Computing persistent homology.
Discrete & Computational Geometry, 33(2):249–274, November 2004.

Pennsylvania State University, University Park

pbt5137@psu.edu

Department of Statistics, Eberly College of Science