Computation of Persistent homology

2018

Outline

Set up

The usual pipeline of TDA

Review of Persistence complex and Persistence module

Main Problem

Basic idea and a review of computation of homology

Basic idea

Review of Computation of homology over $\mathbb{Z}$

Computation of persistent homology

What's more?

Outline

Pipeline

1. Build the filtered simplicial complex from a point cloud.
2. Compute the persistent homology from the filtered simplicial complex.
3. Statistics and interpretation.

Outline

Persistence complex

Definition (Zomorodian and Carlsson 2004)

A **persistence complex** C is a family of chain complexes $\{C_*^i\}_{i\geq 0}$ over a field $\mathbb{F}$, together with chain maps $f^i : C_*^i \to C_*^{i+1}$.

$$C_*^0 \xrightarrow{f^0} C_*^1 \xrightarrow{f^1} C_*^2 \xrightarrow{f^2} \cdots . \tag{1}$$

Example

Various filtered simplicial complices.
For example: Vietoris-Rips complex $VR_\epsilon(S)$ of a point cloud S with chosen parameters $\epsilon_0 < \epsilon_1 < \cdots$.

$$VR_{\epsilon_0}(S) \hookrightarrow VR_{\epsilon_1}(S) \hookrightarrow VR_{\epsilon_2}(S) \hookrightarrow \cdots \tag{2}$$

Persistence module

Definition (Zomorodian and Carlsson 2004)

A **persistence module** M is a family of $\mathbb{F}$-vector spaces $\{M^i\}_{i\geq 0}$ together with linear maps $\phi^i : M^i \to M^{i+1}$.

$$M^0 \xrightarrow{\phi^0} M^1 \xrightarrow{\phi^1} M^2 \to \cdots \tag{3}$$

Example

For any n, nth homology of a persistence complex

$$H_n(C^0) \to H_n(C^1) \to H_n(C^2) \to \cdots . \tag{4}$$

Finite type

Definition

A persistence complex or a persistence module is of **finite type** if each component is a finite dimensional $\mathbb{F}$-vector space and the maps stabilizes, that is, the maps become isomorphisms after finite steps.

$$M^0 \to M^1 \to \cdots \to M^N = M^{N+1} = M^{N+2} = \cdots \tag{5}$$

Example

A filtered simplicial complex associated to a point cloud with finitely many stages is a persistence complex of finite type. Taking homology we get persistence module of finite type.

Correspondence

Given a persistence module $M = \{M^i, \phi^i\}_{i \geq 0}$ over field $\mathbb{F}$, we associate a graded $\mathbb{F}[t]$-module

$$M^0 \xrightarrow{\phi^0} M^1 \xrightarrow{\phi^1} M^2 \xrightarrow{\phi^2} \cdots \quad (6)$$

$$\alpha(M) = \bigoplus_{i=0}^{\infty} M^i \quad (7)$$

The variable t is a shifting operator

$$t \cdot (m^0, m^1, m^2, \cdots) = (0, \phi^0(m^0), \phi^1(m^1), \phi^2(m^2), \cdots). \quad (8)$$

Theorem (Zomorodian and Carlsson 2004)

The correspondence α defines an equivalence of categories between the category of persistence module of finite type over $\mathbb{F}$ and the category of finite generated graded(graded by $\mathbb{N}$) modules over $\mathbb{F}[t]$.

Standard structure theorem

Every finitely generated graded module M over $\mathbb{F}[t]$ (in general, graded PID) decomposes uniquely into the form

$$\left(\bigoplus_{i=1}^{n} \Sigma^{\alpha_i}\mathbb{F}[t]\right) \oplus \left(\bigoplus_{j=1}^{m} \Sigma^{\gamma_j}\mathbb{F}[t]/t^{d_j}\mathbb{F}[t]\right) \qquad (9)$$

for some $n, m, \alpha_i, \gamma_j, d_j$ where $d_j \in \mathbb{N}$(t^{d_j} homogeneous) and $d_j \leq d_{j+1}$ ($t^{d_j} | t^{d_{j+1}}$). Σ^d is to shift upward by a degree d.

Example

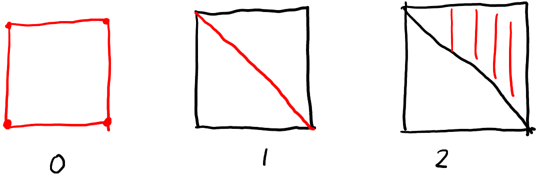
The homology of a persistence complex has such a structure under the correspondence. How to interpret it?

Standard Structure of persistent homology

1. Free part corresponding to $\Sigma^{\alpha_i}\mathbb{F}[t]$ are homology classes that are born at stage α_i and persist till the end.
2. Torsion part corresponding to $\Sigma^{\gamma_j}\mathbb{F}[t]/t^{d_j}\mathbb{F}[t]$ are homology classes that are born at stage γ_j and killed at stage $\gamma_j + d_j$.

Example

simple example and barcode.



Outline

Problem and thoughts

Given a persistence complex and taking homology, how do we compute $m, n, \alpha_i, \gamma_j, d_j$?

$$\left(\bigoplus_{i=1}^{n} \Sigma^{\alpha_i}\mathbb{F}[t]\right) \oplus \left(\bigoplus_{j=1}^{m} \Sigma^{\gamma_j}\mathbb{F}[t]/t^{d_j}\mathbb{F}[t]\right) \tag{10}$$

In other words, how can we find basis that corresponds to basis of the free and torsion parts?

1. Some naive thoughts: find homology basis at each stage and try to connect them up. Why this is not working?
2. Some good idea: find a "consistent" homology basis that can work at every stage.

Example

simple example.

Outline

Basic idea

Want to find a "consistent" homology basis?

Consider the whole persistence complex!

Given a persistence complex $\{C^i, f^i\}_{i\geq 0}$ of finite type, consider the *total persistence complex* as a graded chain complex over $\mathbb{F}[t]$. Boundary maps are termwise boundary maps.

$$C_n := \bigoplus_{i=0}^{\infty} C_n^i \tag{11}$$

$$\cdots \to C_n \xrightarrow{\partial_n} C_{n-1} \xrightarrow{\partial_{n-1}} C_{n-2} \to \cdots \tag{12}$$

graded $\mathbb{F}[t]$-module

Good news

1. $\mathbb{F}[t]$ is a PID, even Euclidean domain.
2. We have algorithm to compute ordinary homology over PID.

Outline

Chain, cycle and boundary

Given a finite simplcial complex, we can construct a chain complex over $\mathbb{Z}$. Every component is a free abelian group.

$$\cdots \to C_n \xrightarrow{\partial_n} C_{n-1} \xrightarrow{\partial_{n-1}} C_{n-2} \to \cdots \tag{13}$$

Cycle group $Z_k := \ker \partial_k \subseteq C_k$
Boundary group $B_k := \operatorname{im} \partial_{k+1} \subseteq C_k$

Figure: chain,circle and boundary

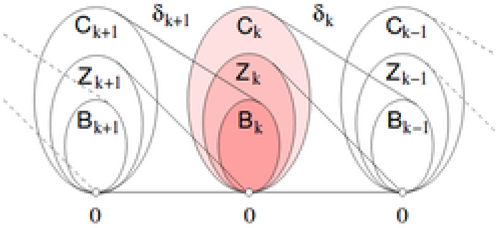


image source: [Zomorodian and Carlsson 2004]

Smith normal form

Start with a matrix representation M_k of $\partial_k : C_k \to Z_{k-1}$ under some free basis of C_k and Z_{k-1}.
Perform elementary row operations and column operations (change of basis in Z_{k-1} and C_k) on M_k to get the Smith normal form with $b_i > 0$ and $b_i | b_{i+1}$

$$\tilde{M}_k = \begin{array}{c} \\ \hat{e}_1 \\ \vdots \\ \hat{e}_{l_k} \\ \hat{e}_{l_{k+1}} \\ \vdots \\ \hat{e}_{l_{n_k}} \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & \cdots & e_{l_k} & e_{l_k+1} & \cdots & e_{m_k} \end{array} \\ \left[\begin{array}{ccc|c} b_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & b_{l_k} & \\ \hline & 0 & & 0 \end{array}\right] \end{array}$$

$\{e_i\}$ is a new basis of C_k. $\{\hat{e}_j\}$ is a new basis of Z_{k-1}.

Row and column operations

Starting with a matrix M under basis $\{e_i\}$ of the domain and $\{\hat{e}_j\}$ of the codomain.

1. Three column operations
   a. exchange column i and j $\leftrightarrow$ exchange e_i and e_j
   b. multiply column i by -1 $\leftrightarrow$ $e_i \to -e_i$
   c. add a multiple q of column j onto column i $\leftrightarrow$ add $q \cdot e_j$ onto e_i
2. Three row operations
   a. exchange row i and j $\leftrightarrow$ exchange $\hat{e}_i$ and $\hat{e}_j$
   b. multiply row i by -1 $\leftrightarrow$ $\hat{e}_i \to -\hat{e}_i$
   c. add a multiple q of row j onto row i $\leftrightarrow$ add $-q \cdot \hat{e}_i$ onto $\hat{e}_j$

Read H_{k-1} from Smith normal form

Now the boundary map $\partial_k : C_k \to Z_{k-1}$ under new basis of C_k and Z_{k-1} has the following form.

$$\tilde{M}_k = \begin{array}{c} \\ \hat{e}_1 \\ \vdots \\ \hat{e}_{l_k} \\ \hat{e}_{l_{k+1}} \\ \vdots \\ \hat{e}_{l_{n_k}} \end{array} \begin{array}{c} \begin{array}{cccccc} e_1 & \cdots & e_{l_k} & e_{l_k+1} & \cdots & e_{m_k} \end{array} \\ \left[\begin{array}{ccc|c} b_1 & & 0 & \\ & \ddots & & 0 \\ 0 & & b_{l_k} & \\ \hline & 0 & & 0 \end{array}\right] \end{array}$$

1. $H_{k-1}(C_*) \simeq \left(\bigoplus_{i=l_{k+1}}^{m_k} \mathbb{Z} \right) \oplus \left(\bigoplus_{1 \le i \le l_k, b_i > 1} \mathbb{Z}/(b_i) \right)$
2. $\{e_i | l_k + 1 \le i \le m_k\}$ is a basis for Z_k.
3. represent ∂_{k+1} in terms of basis of C_{k+1} and Z_k and do the next step.

Generalization

1. For general PID, we can get the Smith form mod an algorithm to find gcd.
2. In graded case, we should start with homogeneous basis of C_k and Z_{k-1}.
3. For Euclidean domain, we have Euclidean algorithm to find gcd. So we have complete algorithm to find Smith normal form.
4. If we work over field, we are happier.

Computation complexity

1. Arithmetic operation: $O(m^3)$ in the worst case, m is the number of simplices in the simplicial complex. But...
2. Naive implementation takes exponential bit operations in the worst case. Entries in the intermediate matrix can explode.

Example

$$\begin{bmatrix} 2 & 0 & 0 & \cdots & 0 \\ 1 & 2 & 0 & \cdots & 0 \\ 1 & 1 & 2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 2 \end{bmatrix} \to \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & \cdots & \\ & & & & 2^n \end{bmatrix}$$

Outline

Example

Let's go through the following example and see the key trick used in the computation.

Figure: filtered simplicial complex

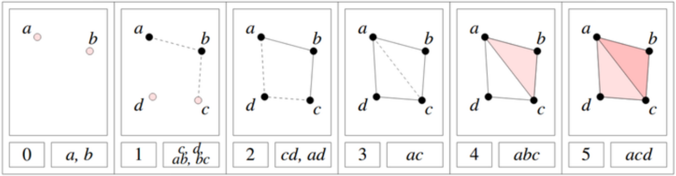


image source: [Zomorodian and Carlsson 2004]

Total persistence complex

a	b	c	d	ab	bc	cd	ad	ac	abc	acd
0	0	1	1	1	1	2	2	3	4	5

image source: [Zomorodian and Carlsson 2004]

Recall that we consider the total persistence complex over $\mathbb{F}[t]$.

$$C_0 = \overbrace{\mathbb{F}[t]a \oplus \mathbb{F}[t]b}^{\text{degree } 0} \oplus \overbrace{\mathbb{F}[t]c \oplus \mathbb{F}[t]d}^{\text{degree } 1}$$

$$C_1 = \overbrace{\mathbb{F}[t]ab \oplus \mathbb{F}[t]bc}^{\text{degree } 1} \oplus \overbrace{\mathbb{F}[t]cd \oplus \mathbb{F}[t]ad}^{\text{degree } 2} \oplus \overbrace{\mathbb{F}[t]ac}^{\text{degree } 3}$$

$$C_2 = \overbrace{\mathbb{F}[t]abc}^{\text{degree } 4} \oplus \overbrace{\mathbb{F}[t]acd}^{\text{degree } 5}$$

Boundary maps

$\partial_1 : C_1 \to C_0 = Z_0$

$$M_1 = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{ccccc} ab & bc & cd & ad & ac \end{array} \\ \begin{bmatrix} -t & 0 & 0 & -t^2 & -t^3 \\ t & -t & 0 & 0 & 0 \\ 0 & 1 & -t & 0 & t^2 \\ 0 & 0 & t & t & 0 \end{bmatrix} \end{array}$$

$\partial_2 : C_2 \to C_1$

$$M_2 = \begin{array}{c} \\ ab \\ bc \\ cd \\ ad \\ ac \end{array} \begin{array}{c} \begin{array}{cc} abc & acd \end{array} \\ \begin{bmatrix} t^3 & 0 \\ t^3 & 0 \\ 0 & t^3 \\ 0 & -t^3 \\ -t & t^2 \end{bmatrix} \end{array}$$

The most important trick

Order homogeneous basis of $Z_0 = C_0$ in reverse degrees.
Order homogeneous basis of C_1 in increasing degrees.

$$M_1 = \begin{array}{c} d \\ c \\ b \\ a \end{array} \overset{\begin{array}{ccccc} ab & bc & cd & ad & ac \end{array}}{\begin{bmatrix} 0 & 0 & t & t & 0 \\ 0 & 1 & -t & 0 & t^2 \\ t & -t & 0 & 0 & 0 \\ -t & 0 & 0 & -t^2 & -t^3 \end{bmatrix}}$$

Only do column reduction to get column echelon form, one column after another

$$\tilde{M}_1 = \begin{array}{c} d \\ c \\ b \\ a \end{array} \overset{\begin{array}{ccccc} cd+t(bc+ab) & bc+ab & ab & z_1 & z_2 \end{array}}{\begin{bmatrix} t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & t & 0 & 0 \\ -t^2 & -t & -t & 0 & 0 \end{bmatrix}}$$

$z_1 = ad - cd - t(bc + ab)$, $z_2 = ac - t^2(bc + ab)$

Black magic of TDA

1. There is no need to do row reduction. Just read torsion coefficients from the diagonal. Free part are those non-pivotal rows. Shift of degrees corresponds to degree of the corresponding row

$$\tilde{M}_1 = \begin{array}{c} \\ d \\ c \\ b \\ a \end{array} \begin{array}{c} \begin{array}{ccccc} cd + t(bc+ab) & bc+ab & ab & z_1 & z_2 \end{array} \\ \begin{bmatrix} t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & t & 0 & 0 \\ -t^2 & -t & -t & 0 & 0 \end{bmatrix} \end{array}$$

$$H_0 \simeq \mathbb{F}[t]/(t) \oplus \Sigma^1 \mathbb{F}[t]/(t) \oplus \mathbb{F}[t]$$

Black magic of TDA

2. Next step: represent ∂_2 in terms of homogeneous basis of C_2 and Z_1.
   Homogeneous basis for Z_1:

$$z_1 = ad - cd - t(cd + ab)$$
$$z_2 = ac - t^2(bc + ab)$$

   To find matrix representation of the next boundary map $\partial_2 : C_2 \to Z_1$, we only have to delete the corresponding rows. Why is that?

$$M_2 = \begin{array}{c} \\ ab \\ bc \\ cd \\ ad \\ ac \end{array} \begin{array}{c} \begin{array}{cc} abc & acd \end{array} \\ \begin{bmatrix} t^3 & 0 \\ t^3 & 0 \\ 0 & t^3 \\ 0 & -t^3 \\ -t & t^2 \end{bmatrix} \end{array} \to \begin{array}{c} \\ z_1 \\ z_2 \end{array} \begin{array}{c} \begin{array}{cc} abc & acd \end{array} \\ \begin{bmatrix} 0 & -t^3 \\ -t & t^2 \end{bmatrix} \end{array}$$

Black magic of TDA

Change of basis in domain and codomain have different effects on matrix.

Assume matrix under basis $\{e_i\}$ of the domain and $\{\hat{e}_i\}$ of the codomain is M.

1. $e_i \to e_i + q \cdot e_j \Leftrightarrow$ replace *col i* by *col i* $+ q \cdot$ *col j*
2. $\hat{e}_i \to \hat{e}_i + q \cdot \hat{e}_j \Leftrightarrow$ replace *row j* by *row j* $- q \cdot$ *row i*. Leaving row i unchanged.

Z_1 Basis: $z_1 = ad - cd - t(cd + ab)$, $z_2 = ac - t^2(bc + ab)$

$$M_2 = \begin{array}{c} \\ ab \\ bc \\ cd \\ ad \\ ac \end{array} \begin{array}{c} \begin{array}{cc} abc & acd \end{array} \\ \begin{bmatrix} t^3 & 0 \\ t^3 & 0 \\ 0 & t^3 \\ 0 & -t^3 \\ -t & t^2 \end{bmatrix} \end{array} \text{ and } M_1 \circ M_2 = 0$$

Algorithm

1. Put a total order on all simplices first by dimension and then by degree.
2. Order homogeneous basis of Z_{k-1} in reverse degree, do column reduction on M_k, read H_{k-1} and get a basis of Z_k
3. Delete corresponding columns in M_{k+1}. Go to step 2.

remark

As long as we keep track of the degree of the basis, there is no need to work over $\mathbb{F}[t]$. We can just work over $\mathbb{F}$.

Computation complexity

1. In the worst case it's bounded by $O(m^3)$ field operations where m is the number of simplices. It's more like computing homology over $\mathbb{F}$ than over $\mathbb{F}[t]$.
2. This bound is sharp by an example in Morozov 2005.
3. In pratice, people see linear behavior because of sparcity of the matrix.

Improve efficiency

1. Reduce size of input complexes while preserving persistent homology: simplicial collapses.
2. Improve the standard algorithm to compute persistent homology. Best known: $O(M(m) + m^2 log^2 m)$ in [Milosavljević, Morozov, and Skraba 2011] where $M(m)$ is the complexity of multiplying two $m \times m$ matrices. The best upper bound of $M(m)$ is $m^{log 7} \approx m^{2.8}$ by Strassen 1969.

What else?

1. Zigzag persistence. [Carlsson and De Silva 2010]
   $C_i \to C_{i+1}$ and $C_i \leftarrow C_{i+1}$.
2. Multidimensional filtration. [Carlsson and Zomorodian 2009]

Carlsson, Gunnar and Vin De Silva (2010). “Zigzag persistence”. In: *Foundations of computational mathematics* 10.4, pp. 367–405.

Carlsson, Gunnar and Afra Zomorodian (2009). “The theory of multidimensional persistence”. In: *Discrete & Computational Geometry* 42.1, pp. 71–93.

Milosavljević, Nikola, Dmitriy Morozov, and Primoz Skraba (2011). “Zigzag persistent homology in matrix multiplication time”. In: *Proceedings of the twenty-seventh annual symposium on Computational geometry*. ACM, pp. 216–225.

Morozov, Dmitriy (2005). “Persistence algorithm takes cubic time in worst case”. In:

Strassen, Volker (1969). “Gaussian elimination is not optimal”. In: *Numerische mathematik* 13.4, pp. 354–356.

Zomorodian, Afra and Gunnar Carlsson (2004). “Computing persistent homology”. In: *Proceedings of the twentieth annual symposium on Computational geometry*. ACM, pp. 347–356.