Persistent Homology for Shape Comparison

Ulderico Fugacci

CNR - IMATI

Persistent Homology for Shape Comparison

- *Quick Overview on TDA*
- *Case Study*

Persistent Homology for Shape Comparison

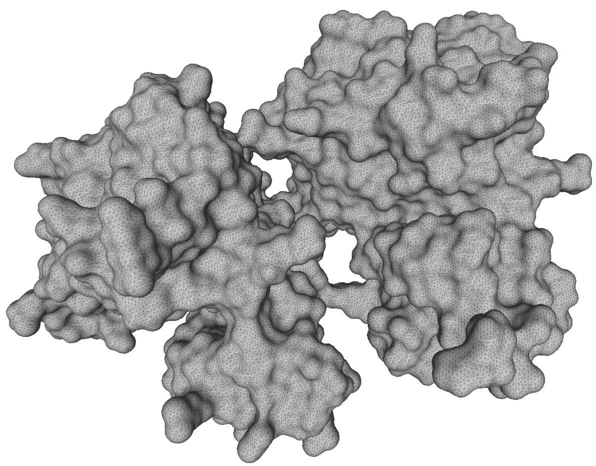
- ***Quick Overview on TDA***
- *Case Study*

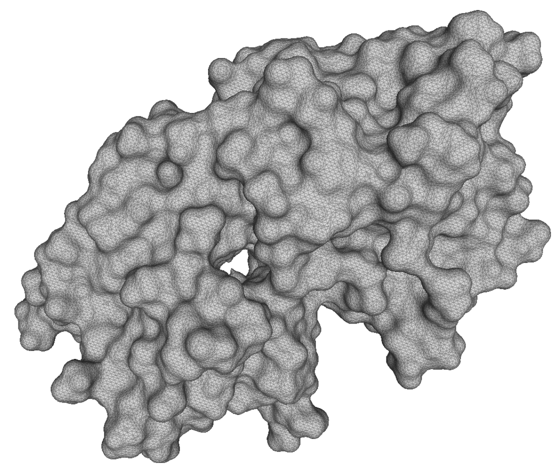
Topological Data Analysis

Topological Data Analysis (TDA)* aims at describing, characterizing, and discriminating data on the basis of their *shape

Example: Consider a dataset consisting of *molecular surfaces* and pick two of them, X and Y

X

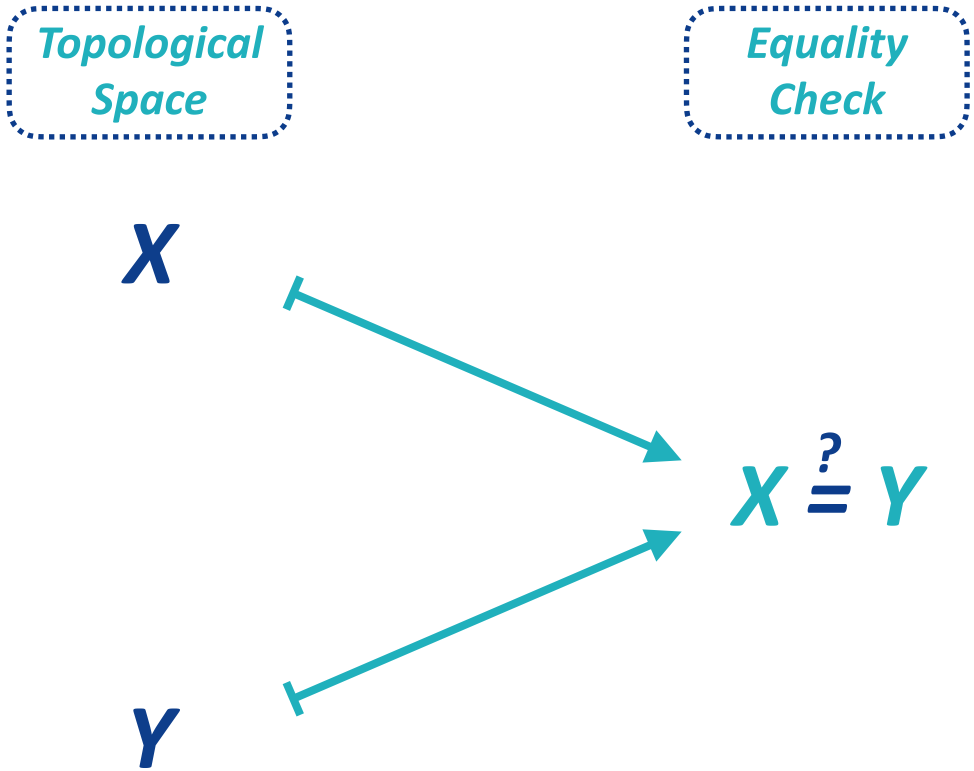




Y

Do X and Y represent the same molecule?

Are X, Y equal?



Do X and Y have the same shape?

Topological Space		Algebraic Structure	Equality Check
X	$\longmapsto$	$\mathit{Shape}(X)$	
			$\mathit{Shape}(X) \stackrel{?}{=} \mathit{Shape}(Y)$
Y	$\longmapsto$	$\mathit{Shape}(Y)$	

Do X and Y have the same shape?

Topological Space	Algebraic Structure	Algebraic Invariant	Equality Check

$$X \longmapsto \mathit{Shape}(X) \longmapsto \mathit{Inv}(\mathit{Shape}(X))$$

$$\mathit{Inv}(\mathit{Shape}(X)) \stackrel{?}{=} \mathit{Inv}(\mathit{Shape}(Y))$$

$$Y \longmapsto \mathit{Shape}(Y) \longmapsto \mathit{Inv}(\mathit{Shape}(Y))$$

Homology

Given a topological space *X* and a field $\mathbb{F}$, the ***homology of X*** is a ***topological invariant***

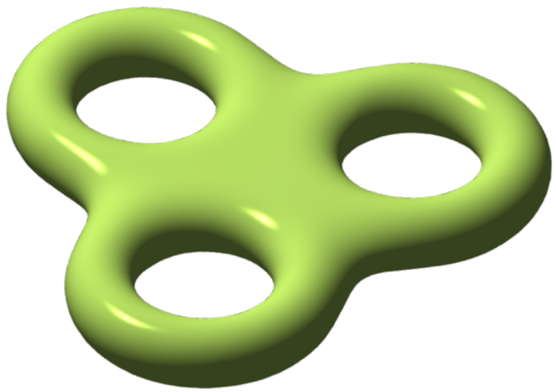
intuition

detecting the "holes" of X

capturing the independent non-bounding cycles of X

measuring how far the chain complex associated with X is from being exact

formalism



$$\longmapsto H_i(X;\mathbb{F}) \cong \begin{cases} \mathbb{F} & \text{for } i=0 \\ \mathbb{F}^6 & \text{for } i=1 \\ \mathbb{F} & \text{for } i=2 \\ 0 & \text{otherwise} \end{cases}$$

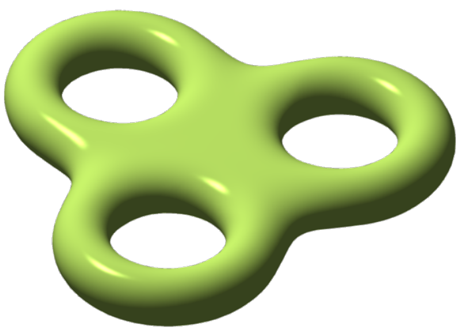
Homology

Theorem:

Any homology group of X with coefficients in $\mathbb{F}$ can be expressed as

$$H_i(X;\mathbb{F}) \cong \mathbb{F}^{\beta_i}$$

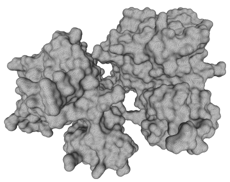
*where β_i is called the **ith Betti number** of X*



$$\longmapsto \quad \beta_i = \begin{cases} 1 & \text{for } i = 0 \\ 6 & \text{for } i = 1 \\ 1 & \text{for } i = 2 \\ 0 & \text{otherwise} \end{cases}$$

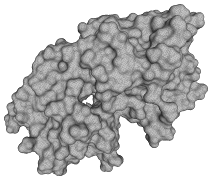
Do X and Y have the same shape?

Topological Space | *Algebraic Structure* | *Algebraic Invariant* | *Equality Check*



$$\longmapsto H_i(X;\mathbb{Z}_2) \cong \begin{cases} \mathbb{Z}_2 & \text{for } i=0 \\ (\mathbb{Z}_2)^4 & \text{for } i=1 \\ \mathbb{Z}_2 & \text{for } i=2 \end{cases} \longmapsto (1,4,1)$$

$$X \neq Y$$



$$\longmapsto H_i(X;\mathbb{Z}_2) \cong \begin{cases} \mathbb{Z}_2 & \text{for } i=0 \\ (\mathbb{Z}_2)^2 & \text{for } i=1 \\ \mathbb{Z}_2 & \text{for } i=2 \end{cases} \longmapsto (1,2,1)$$

Homology Fails

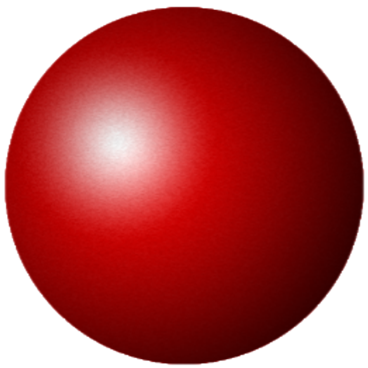
Do they have the same shape?

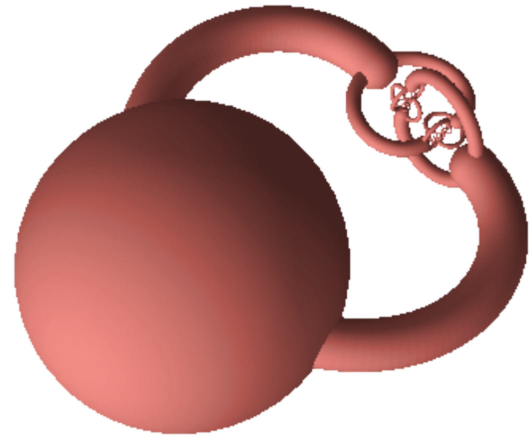
In Practice? ✓

In Theory? ✗

Homology Fails

Do they have the same shape?





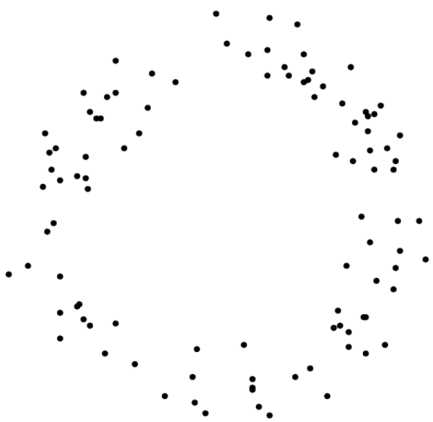
In Practice?

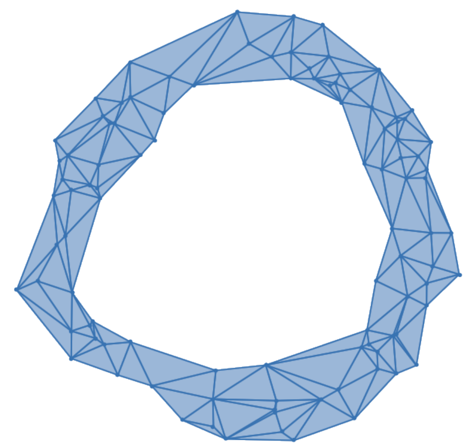
In Theory?

Homology Fails

Which is the shape of a given data?

We would like to retrieve the ***"actual" homological information*** of a data





Point Cloud Dataset

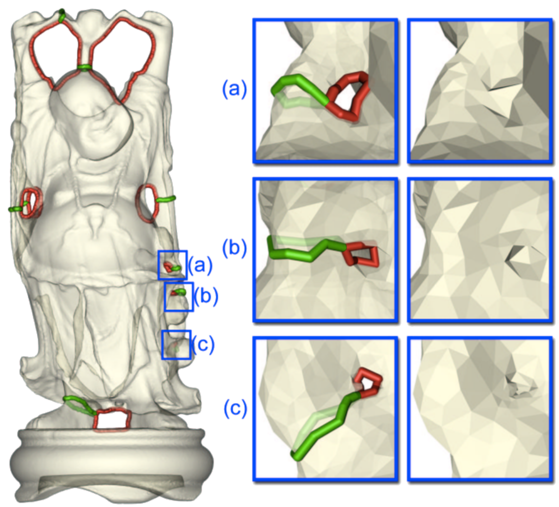
Topological Nature of the "Underlying" Shape

Image from [Bauer 2015]

Homology Fails

Which is the shape of a given data?

We would like to retrieve the ***"actual" homological information*** of a data



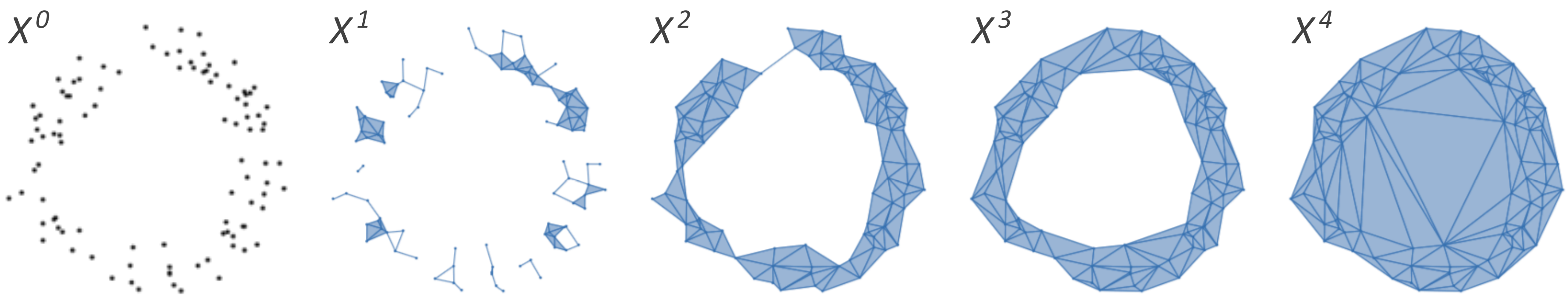
Noisy Dataset → ***Relevant Homological Information***

Image from [Dey et al. 2008]

The Solution? Persistent Homology

In a Nutshell:

Persistent homology allows for describing the homology changes of an evolving object



Do X and Y have the same shape?

Topological Space	Algebraic Structure	Algebraic Invariant	Equality Check

$$\begin{tikzcd}
X \arrow[r, mapsto] & \mathit{Shape}(X) \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(X)) \arrow[dr, mapsto] & \\
 & & & \mathit{Inv}(\mathit{Shape}(X)) \overset{?}{=} \mathit{Inv}(\mathit{Shape}(Y)) \\
Y \arrow[r, mapsto] & \mathit{Shape}(Y) \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(Y)) \arrow[ur, mapsto] &
\end{tikzcd}$$

Do X and Y have the same shape?

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Equality Check

$$X \longmapsto X' \longmapsto Shape(X') \longmapsto Inv(Shape(X'))$$

$$Y \longmapsto Y' \longmapsto Shape(Y') \longmapsto Inv(Shape(Y'))$$

$$Inv(Shape(X')) \stackrel{?}{=} Inv(Shape(Y'))$$

Do X and Y have a *similar* shape?

Input Data | *Topological Representation* | *Algebraic Structure* | *Algebraic Invariant* | *Similarity Check*

$$
\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & Shape(X') \arrow[r, mapsto] & Inv(Shape(X')) \arrow[dr, mapsto] & \\
& & & & d(Inv(Shape(X')),Inv(Shape(Y'))) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & Shape(Y') \arrow[r, mapsto] & Inv(Shape(Y')) \arrow[ur, mapsto] &
\end{tikzcd}
$$

Do X and Y have a similar shape?

Input Data | *Topological Representation* | *Algebraic Structure* | *Algebraic Invariant* | *Similarity Check*

$$\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & \mathit{Shape}(X') \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(X')) \arrow[dr, mapsto] & \\
 & & & & d(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & \mathit{Shape}(Y') \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(Y')) \arrow[ur, mapsto] &
\end{tikzcd}$$

Do X and Y have a similar shape?

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Similarity Check

$$
\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & Shape(X') \arrow[r, mapsto] & Inv(Shape(X')) \arrow[dr, mapsto] & \\
 & & & & d(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & Shape(Y') \arrow[r, mapsto] & Inv(Shape(Y')) \arrow[ur, mapsto] &
\end{tikzcd}
$$

The entire procedure has to be stable

I.e. robust to noise and small perturbations

Do X and Y have a similar shape?

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Similarity Check

$$\begin{array}{ccccccccc}
X & \longmapsto & X' & \longmapsto & \mathit{Shape}(X') & \longmapsto & \mathit{Inv}(\mathit{Shape}(X')) & & \\
 & & & & & & & \searrow & \\
 & & & & & & & & d(X, Y) \\
 & & & & & & & \nearrow & \\
Y & \longmapsto & Y' & \longmapsto & \mathit{Shape}(Y') & \longmapsto & \mathit{Inv}(\mathit{Shape}(Y')) & &
\end{array}$$

In other words,

Similar Data ⇒ Similar Algebraic Invariants

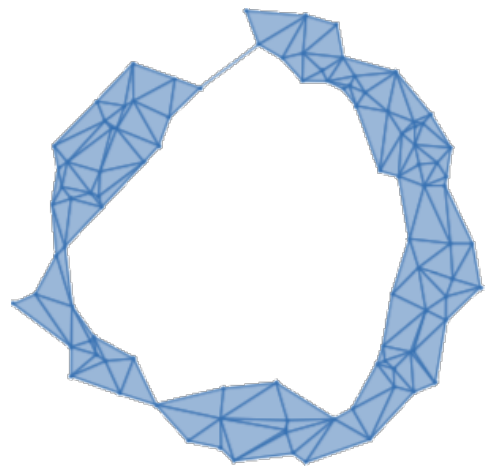
Persistent Homology

From an *Input Data* X to a *Topological Representation* X':

Let X be a ***point cloud*** $V \subseteq \mathbb{R}^d$

We set $X' := \{ X^p \}_{p \in \mathbb{R}}$, *where, chosen a value* $p \in \mathbb{R}$, X^p *is defined as*

$$\{ \sigma \subseteq V \mid d(u, v) \leq p, \ \forall u, v \in \sigma \}$$

X  $\longmapsto$  X^p

Persistent Homology

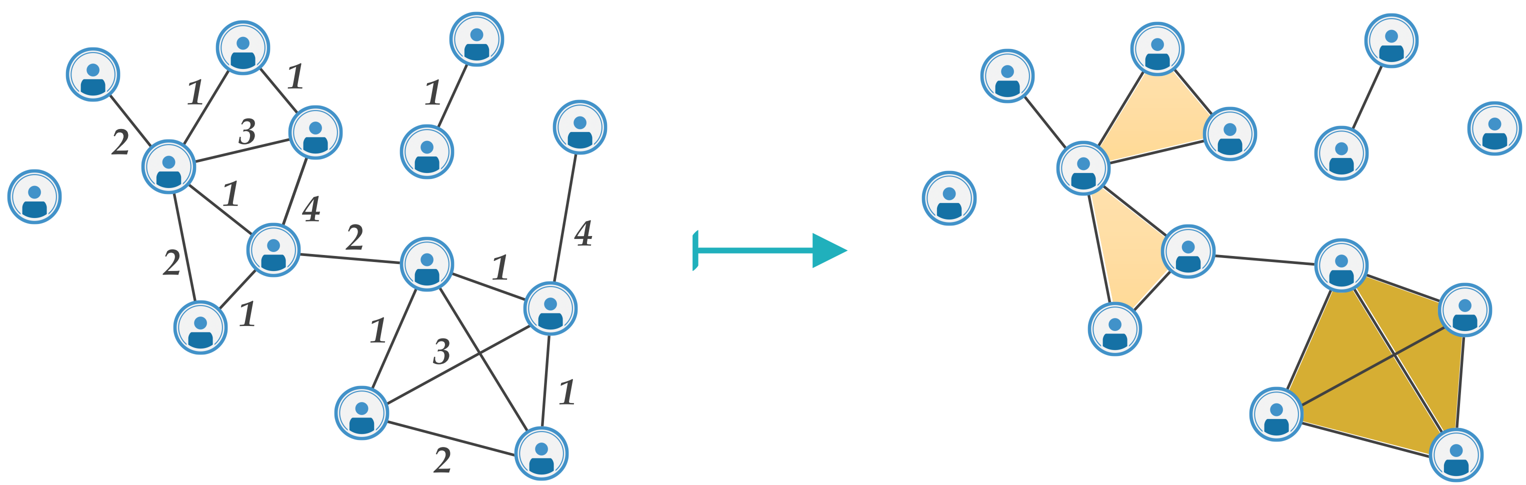
From an Input Data X to a Topological Representation X':

Let X be a ***weighted graph*** $G := (V, E, w: E \longrightarrow \mathbb{R})$

We set $X' := \{ X^p \}_{p \in \mathbb{R}}$*, where, chosen a value* $p \in \mathbb{R}$*,* X^p *is defined as*

$$\{ \sigma \subseteq V \mid w(u, v) \leq p, \ \forall u, v \in \sigma \}$$

X



X^p

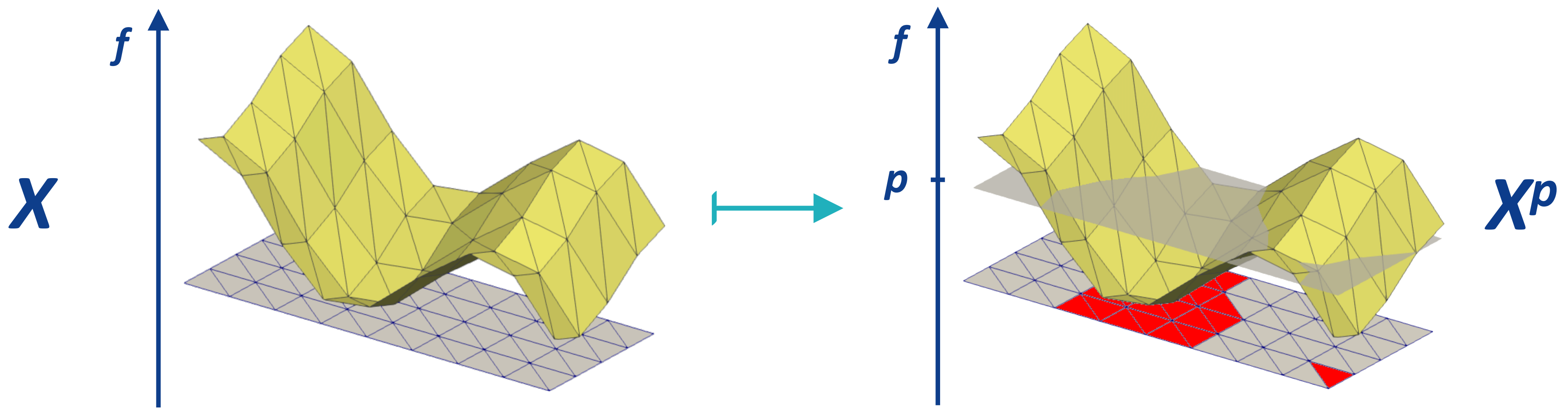
Persistent Homology

From an *Input Data X* to a *Topological Representation X'*:

Let X be a ***function f: D*** $\longrightarrow \mathbb{R}$

We set $X' := \{ X^p \}_{p \in \mathbb{R}}$*, where, chosen a value* $p \in \mathbb{R}$, X^p *is defined as*

$$\{ \sigma \in D \mid f(\sigma) \leq p \}$$



Persistent Homology

From an *Input Data* X to a *Topological Representation* X':

Independently from the construction procedure,

*the collection of topological spaces $\{ X^p \}_{p \in \mathbb{R}}$, called **filtration** of X,*

*satisfies that, **for any p, q** $\in \mathbb{R}$ **such that p ≤ q**,*

$$X^p \subseteq X^q$$

Working Assumption:

***We can always pretend that parameter p varies over** $\mathbb{N}$*

Persistent Homology

From a Topological Representation X' to an Algebraic Structure Shape(X'):

Given a filtration $X' := \{ X^p \}_{p \in \mathbb{N}}$, a value $i \in \mathbb{N}$, and a field $\mathbb{F}$, the ***i^{th} persistence module M of X' over*** *$\mathbb{F}$ is defined as the* ***finitely generated graded $\mathbb{F}[x]$-module***

$$M := \bigoplus_{p \in \mathbb{N}} M_p$$

where:

- ***$M_p := H_i(X^p; \mathbb{F})$*** *, the set of* ***homogeneous elements of grade p***
- *The* ***action $x^{q-p}\,h$ over an element h of grade p*** *is defined as* ***$\mu_i^{\,p,q}(h)$****, where:*
  - ***$\mu_i^{\,p,q}(h)$: $H_i(X^p; \mathbb{F}) \longrightarrow H_i(X^q; \mathbb{F})$*** *is the linear map induced by the inclusion $X^p \subseteq X^q$*

Persistent Homology

From an Algebraic Structure Shape(X') to an Algebraic Invariant Inv(Shape(X')):

Theorem (structure for finitely generated graded modules over a PID):

Any persistence module M can be expressed as

$$M \cong \bigoplus_{k=1}^{n} \mathbb{F}[x](-r_k) \oplus \bigoplus_{j=1}^{m} \left(\mathbb{F}[x]/(x^{q_j - p_j})\right)(-p_j)$$

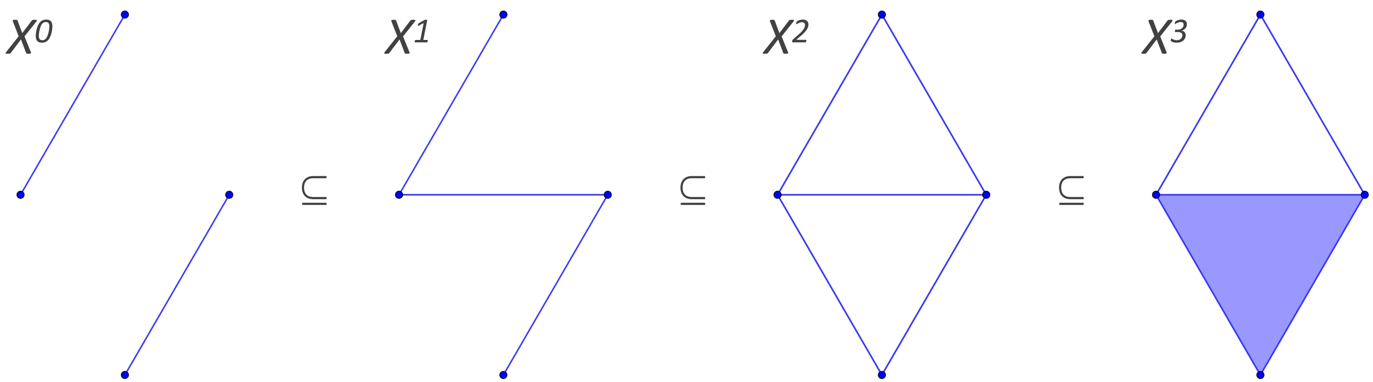
So, ***M is completely determined by the collection of values r_k and of pairs (p_j, q_j)***

Such descriptors are typically expressed as pairs, called ***persistence pairs*** of M, of the kind ***(r_k, ∞)*** and ***(p_j, q_j)***

Persistent Homology

Intuitively:

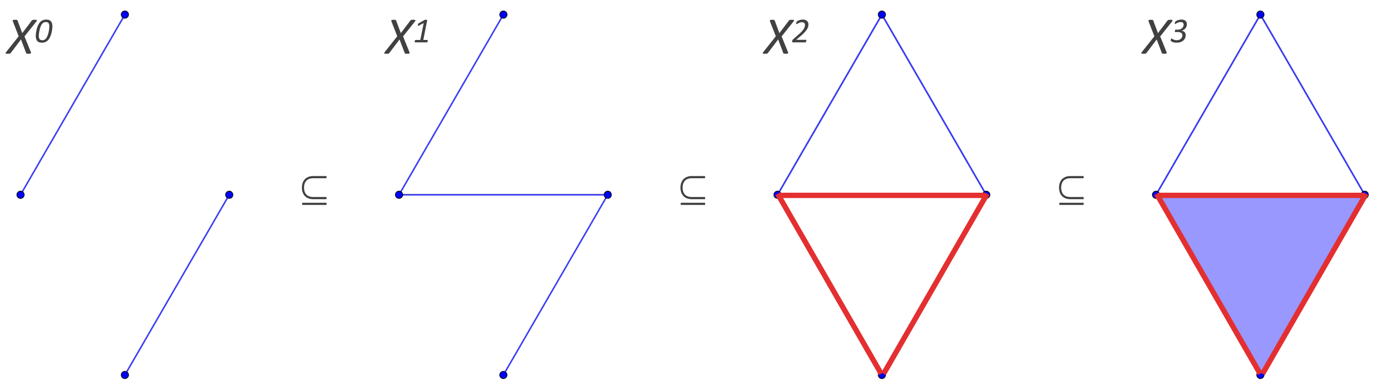
Given a filtration $X' := \{ X^p \}_{p \in \mathbb{N}}$, a ***persistence pair (p,q)*** $\in \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ *with $p < q$ represents a* ***homological class*** *that is* ***born at step p*** *and* ***dies at step q***



Persistent Homology

Intuitively:

Given a filtration $X' := \{ X^p \}_{p \in \mathbb{N}}$, a ***persistence pair (p,q)*** $\in \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ *with $p < q$ represents a* ***homological class*** *that is* ***born at step p*** *and* ***dies at step q***

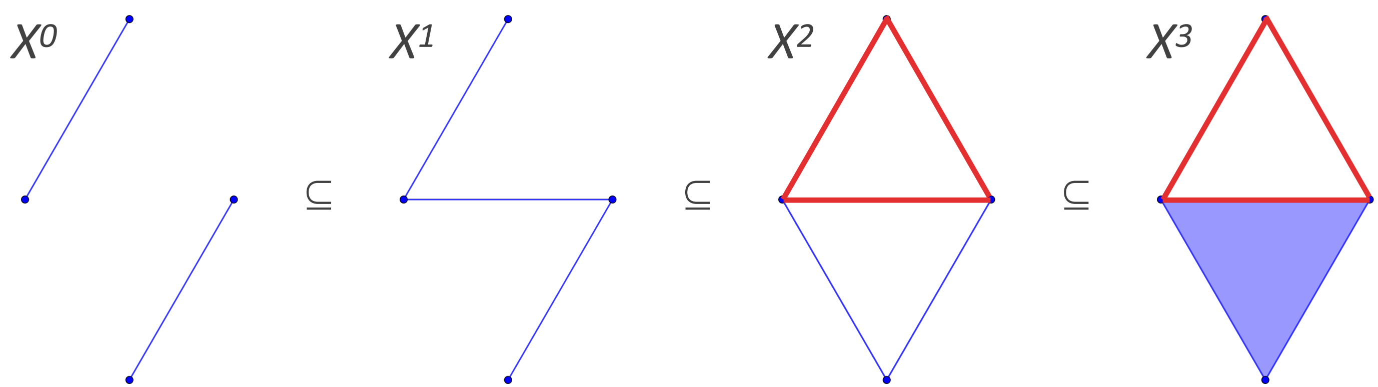


(2, 3)

Persistent Homology

Intuitively:

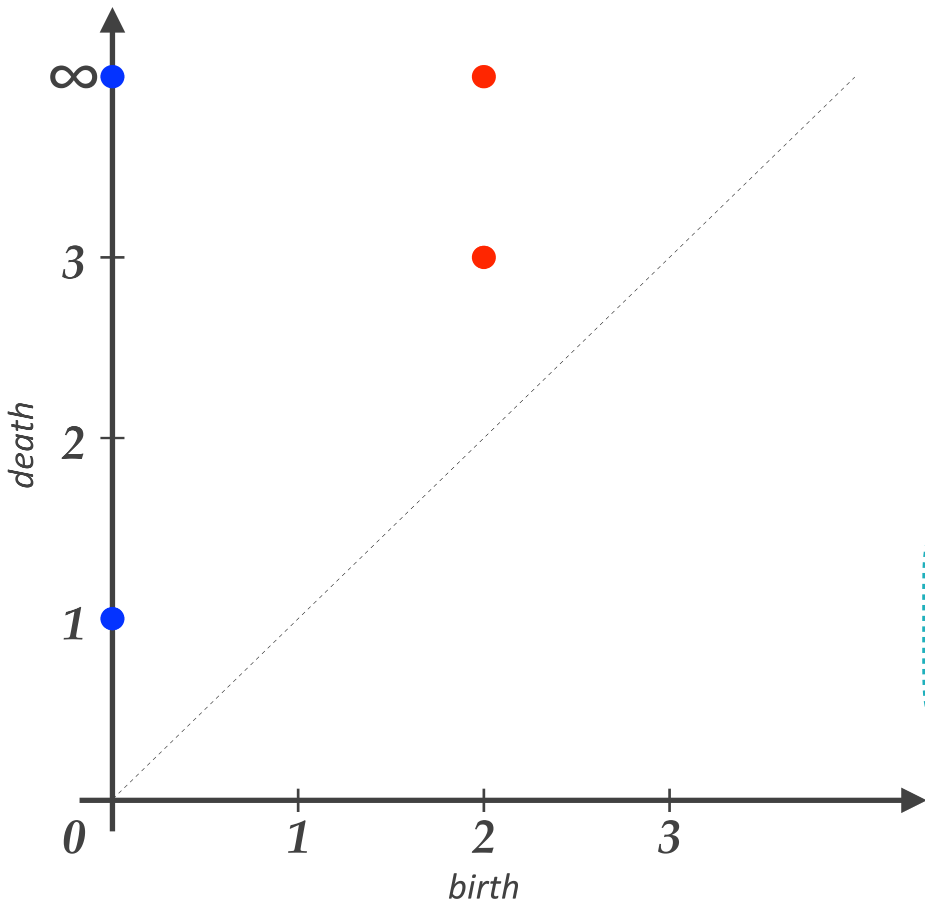
Given a filtration $X' := \{ X^p \}_{p \in \mathbb{N}}$, a ***persistence pair (p,q)*** $\in \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ *with $p < q$ represents a* ***homological class*** *that is* ***born at step p*** *and* ***dies at step q***



(2, ∞) essential pair

Persistent Homology

Persistence Diagrams: *Persistence pairs are represented as* ***points in*** $\mathbb{R} \times (\mathbb{R} \cup \{\infty\})$



H_0	$(0, 1)$ $(0, \infty)$	H_1	$(2, 3)$ $(2, \infty)$

Formally, a persistence diagram is a ***multi-set***

I.e. points are endowed with **multiplicity**

Persistent Homology

From an Algebraic Invariant Inv(Shape(X')) to a Similarity Check d(X, Y):

Given two persistence diagrams D_X and D_Y, their ***bottleneck distance*** $\boldsymbol{d_B}$ *is defined as*

$$d_B(D_X, D_Y) := \inf_{\gamma} \left\{ \sup_{x \in D_X} \left\{ \|x - \gamma(x)\|_\infty \right\} \right\}$$

where γ ranges over all bijections from D_X to D_Y

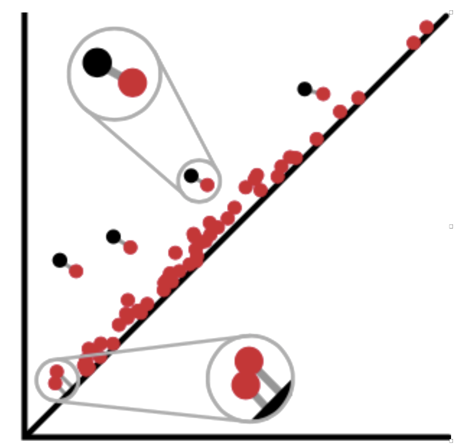


Image from [Rieck 2016]

Do X and Y have a similar shape?

Persistent Homology Pipeline:

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Similarity Check

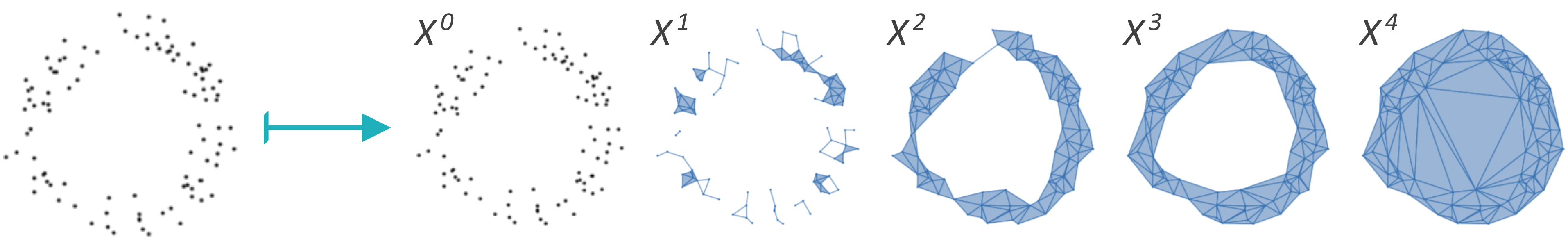
$$\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & Shape(X') \arrow[r, mapsto] & Inv(Shape(X')) \arrow[dr, mapsto] & \\
 & & & & d(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & Shape(Y') \arrow[r, mapsto] & Inv(Shape(Y')) \arrow[ur, mapsto] &
\end{tikzcd}$$

Do X and Y have a similar shape?

Persistent Homology Pipeline:

Input Data

Topological Representation



X

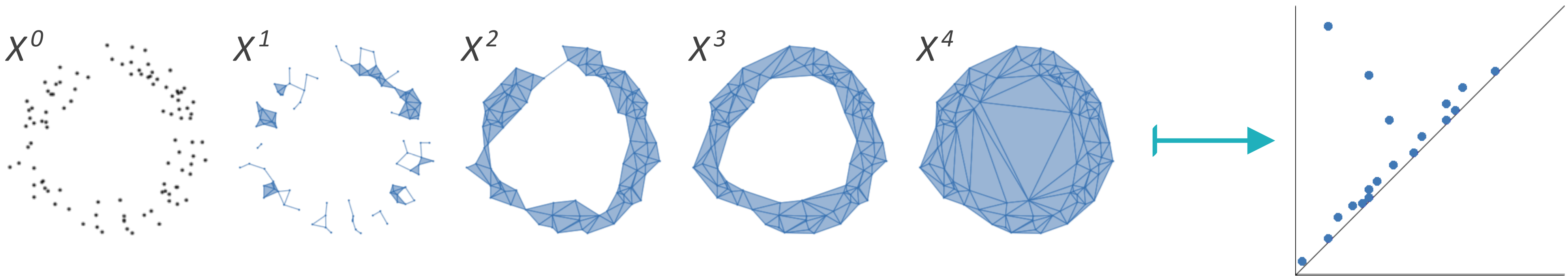
X'

Do X and Y have a similar shape?

Persistent Homology Pipeline:

Topological Representation

Algebraic Invariant



X'

$Inv(Shape(X'))$

Do X and Y have a similar shape?

Persistent Homology Pipeline:

Input Data | *Topological Representation* | *Algebraic Structure* | *Algebraic Invariant* | *Similarity Check*

$$
\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & \mathit{Shape}(X') \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(X')) \arrow[dr, mapsto] & \\
 & & & & d(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & \mathit{Shape}(Y') \arrow[r, mapsto] & \mathit{Inv}(\mathit{Shape}(Y')) \arrow[ur, mapsto] &
\end{tikzcd}
$$

Do X and Y have a similar shape?

AI-Oriented Alternative:

Input Data	*Topological Representation*	*Algebraic Structure*	*Feature Vector*	*Learning Process*

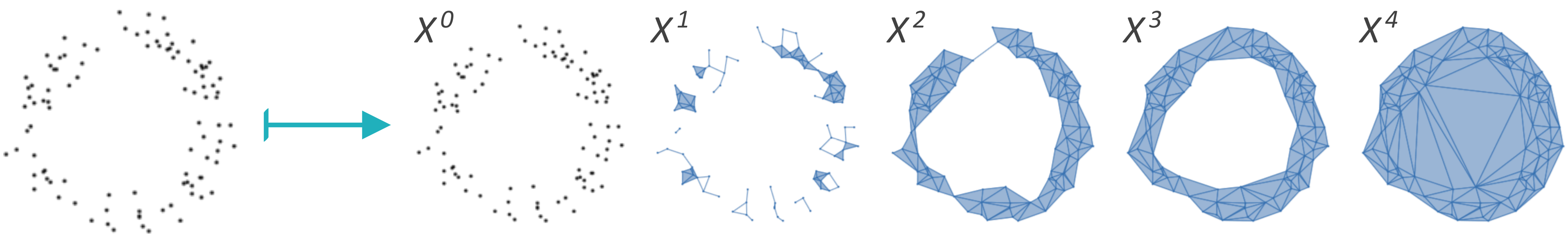
$$
\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & \mathit{Shape}(X') \arrow[r, mapsto] & \phi(\mathit{Shape}(X')) \arrow[dr, mapsto] & \\
 & & & & k(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & \mathit{Shape}(Y') \arrow[r, mapsto] & \phi(\mathit{Shape}(Y')) \arrow[ur, mapsto] &
\end{tikzcd}
$$

Do X and Y have a similar shape?

AI-Oriented Alternative:

Input Data

Topological Representation



X

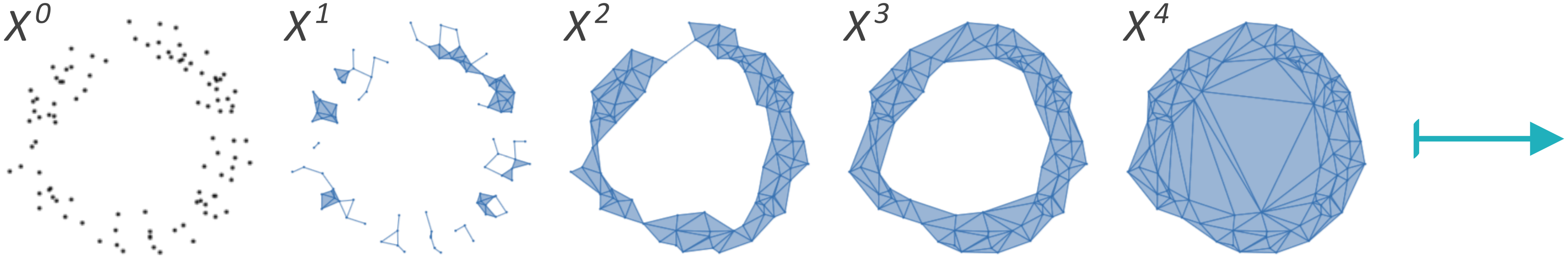
X'

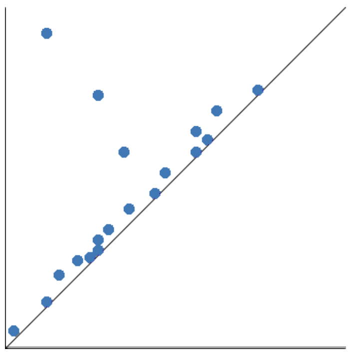
Do X and Y have a similar shape?

AI-Oriented Alternative:

Topological Representation

Algebraic Invariant





X'

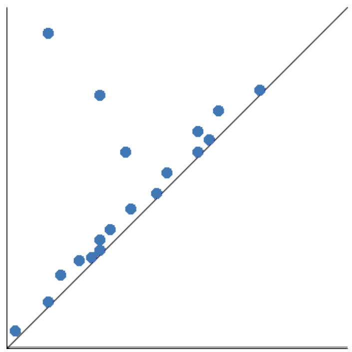
Inv(Shape(X'))

Do X and Y have a similar shape?

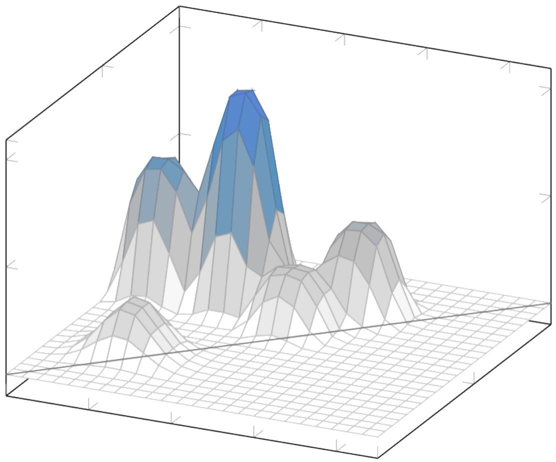
AI-Oriented Alternative:

Algebraic Invariant

Feature Vector



$\longmapsto$



Inv(Shape(X'))

ϕ(Shape(X'))

Do X and Y have a similar shape?

Input Data | Topological Representation | Algebraic Structure | Algebraic Invariant | Similarity Check

$$\begin{tikzcd} X \arrow[r, mapsto] & X' \arrow[r, mapsto] & Shape(X') \arrow[r, mapsto] & Inv(Shape(X')) \arrow[dr, mapsto] & \\ & & & & d(X, Y) \\ Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & Shape(Y') \arrow[r, mapsto] & Inv(Shape(Y')) \arrow[ur, mapsto] & \end{tikzcd}$$

Do X and Y have a similar shape?

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Similarity Check

$$X \longmapsto X' \longmapsto Shape(X') \longmapsto Inv(Shape(X'))$$

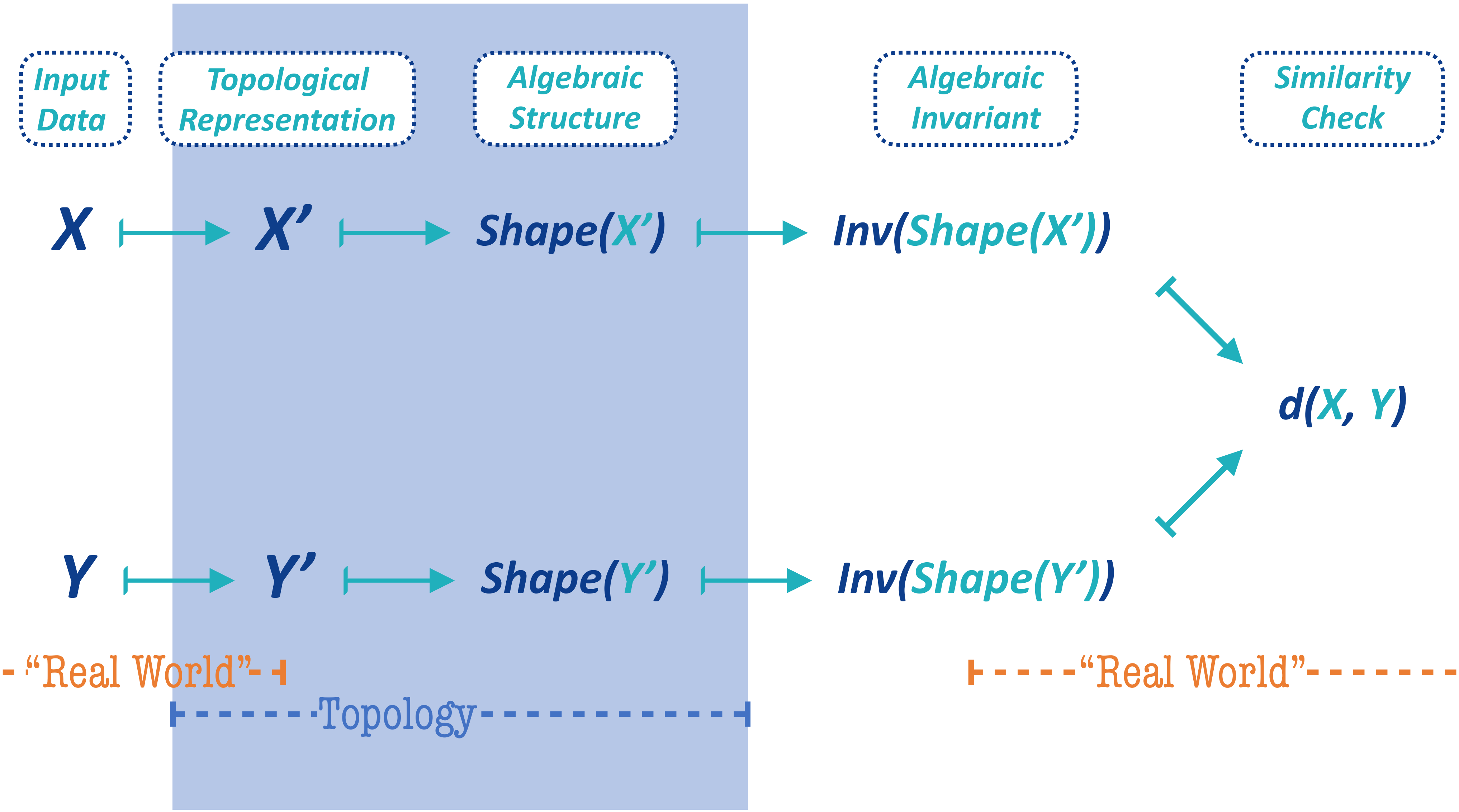
$$d(X, Y)$$

$$Y \longmapsto Y' \longmapsto Shape(Y') \longmapsto Inv(Shape(Y'))$$

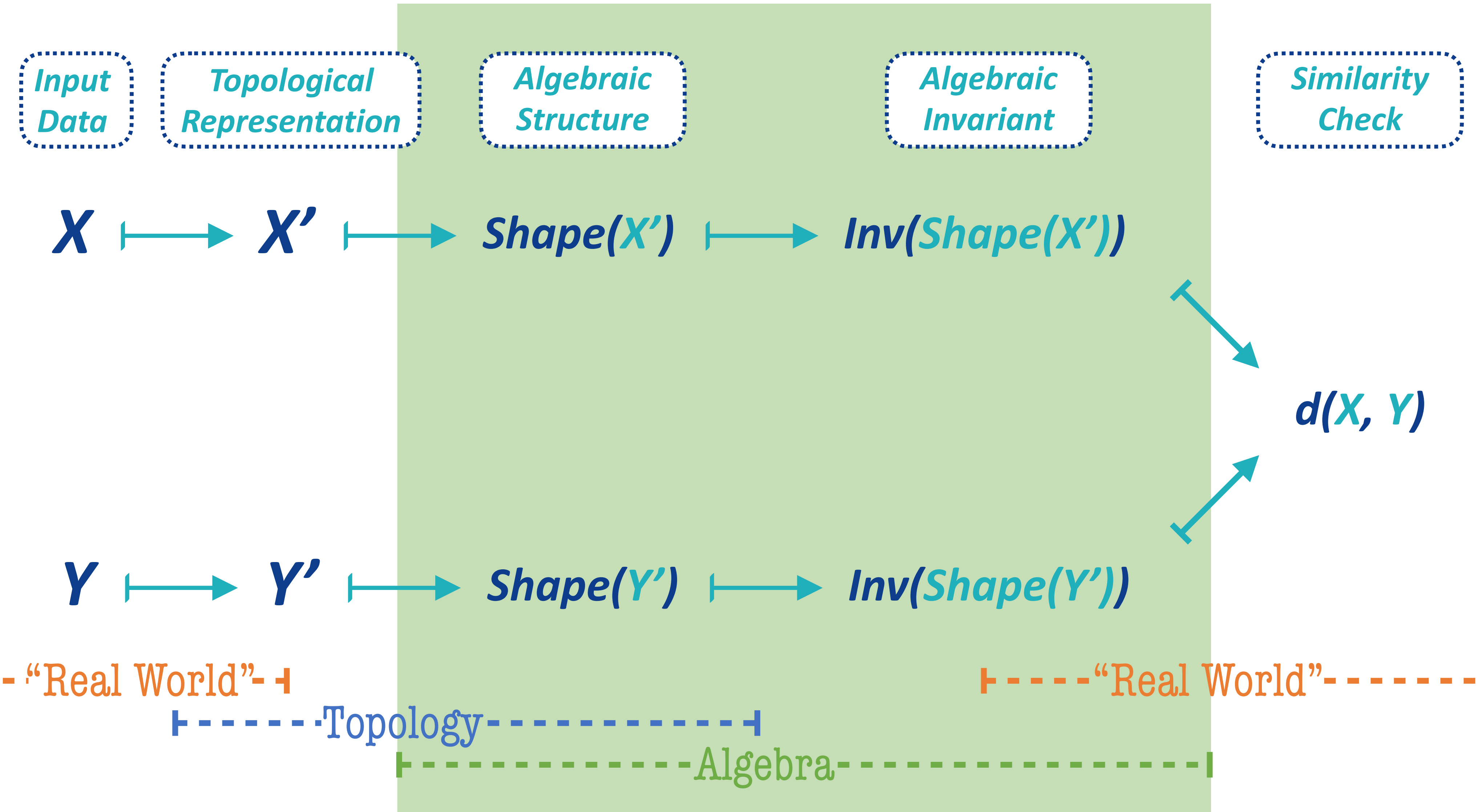
"Real World"

"Real World"

Do X and Y have a similar shape?



Do X and Y have a similar shape?



Do X and Y have a similar shape?

Input Data	Topological Representation	Algebraic Structure	Algebraic Invariant	Similarity Check

$$
\begin{tikzcd}
X \arrow[r, mapsto] & X' \arrow[r, mapsto] & Shape(X') \arrow[r, mapsto] & Inv(Shape(X')) \arrow[dr, mapsto] & \\
 & & & & d(X, Y) \\
Y \arrow[r, mapsto] & Y' \arrow[r, mapsto] & Shape(Y') \arrow[r, mapsto] & Inv(Shape(Y')) \arrow[ur, mapsto] &
\end{tikzcd}
$$

"Real World" "Real World"

Topology

Algebra

Persistent Homology for Shape Comparison

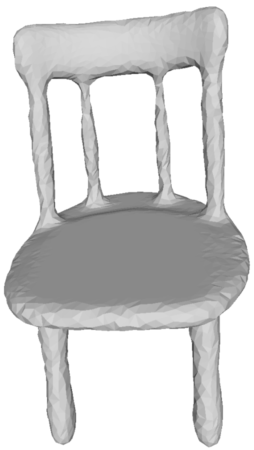
- *Quick Overview on TDA*
- ***Case Study***

Case Study

Goal: *Given a dataset of **80 surfaces**, classify them on the basis of their shape*

Case Study

Homology is not enough

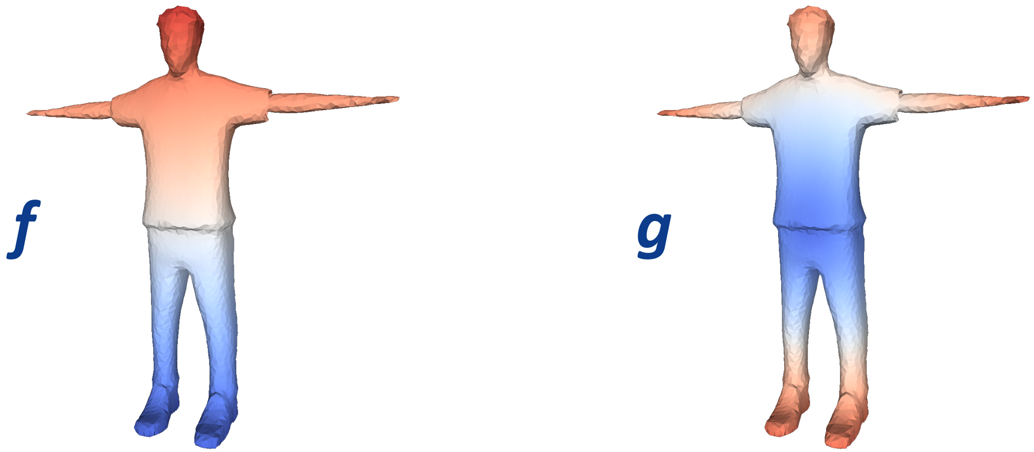





Case Study

In order to adopt ***persistent homology****,*
we need to filter each surface through a function:

- $f(x, y, z) := z$
- $g(x, y, z) :=$ distance of (x, y, z) from the model barycentre



Software Packages

Several software packages for computing persistent homology have been developed:

- ***javaPlex***
- ***jHoles***
- ***Dionysus***
- ***Perseus***
- ***PHAT***
- ***DIPHA***
- ***Gudhi***
- ***SimpPers***
- ***Ripser***
- ***giotto-tda***
- ***TDAstats***
- ***…***

Software Packages

PHAT:

- ***Language:***
  - *C++ (with Python bindings)*
- ***Algorithms:***
  - ***Standard, Dual, Twist, Chunk, Spectral Sequences***
- ***Coefficient Fields:***
  - $\mathbb{Z}_2$
- ***Homology:***
  - ***Simplicial, Cubical***
- ***Accepted Inputs:***
  - ***Boundary Matrices***
- ***Computed Filtrations:***
  - -
- ***Visualizations:***
  - -
- ***Additional Features:***
  - -

Software Packages

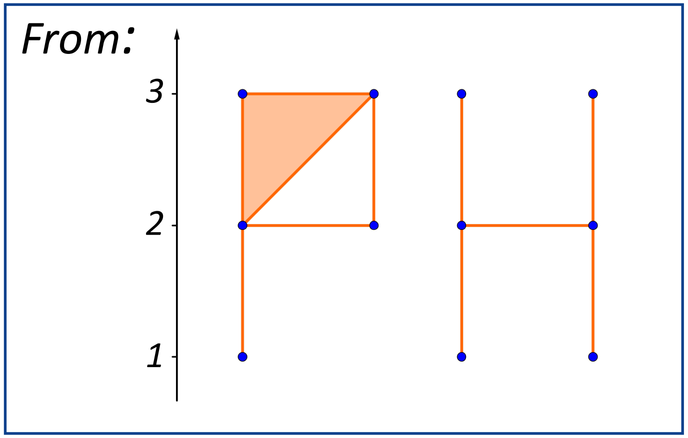
Ripser:

- ***Language***:
  - *C++ (with Python bindings)*
- ***Algorithms***:
  - *Dual, Twist*
- ***Coefficient Fields***:
  - $\mathbb{Z}_p$
- ***Homology***:
  - *Simplicial*
- ***Accepted Inputs***:
  - *Point Clouds, Distance Matrices*
- ***Computed Filtrations***:
  - *Vietoris-Rips and Čech complexes, Alpha-Shapes, Lower Star of Cubical complexes*
- ***Visualizations***:
  - *Persistence Diagrams (through Persim: Persistence Images)*
- ***Additional Features***:
  - *Representative Cocycles (through Persim: Bottleneck distance, modified Gromov–Hausdorff distance, Sliced Wasserstein kernel, Heat kernel)*

Persistent Homology Computation

Standard Algorithm:

[Zomorodian & Carlsson 2005]



To:

H_0 [1, 2] [1, ∞) [1, ∞)

H_1 [3, ∞)

$i\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1								1															
2									1			1											
3										1		1											
4								1			1						1	1					
5											1												
6									1											1			
7										1											1		
8																							
9																							
10																							
11																							
12																							
13																	1						
14																		1					
15																				1			
16																					1		
17																							1
18																							1
19																							
20																							
21																							
22																							1
23																							
low								4	6	7	5	3					13	14		15	16		22

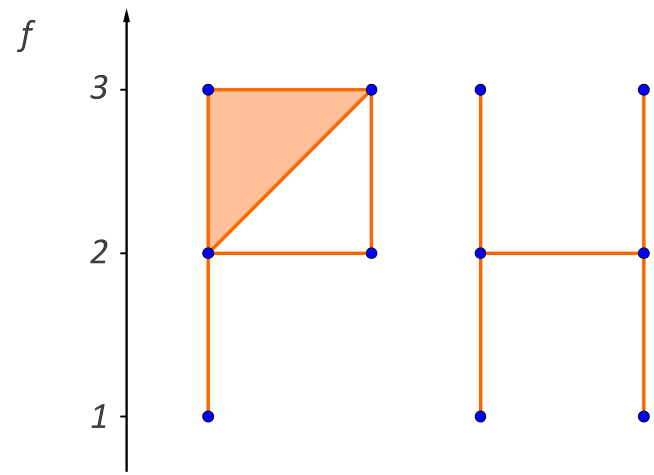
Compute a ***reduced boundary matrix*** for K^f from which easily read the persistence pairs

Persistent Homology Computation

Given a filtered simplicial complex, let us consider its ***filtering function f***:

$$f(\sigma) := \min \{ p \mid \sigma \in K^p \}$$

Conversely, $K^p := \{ \sigma \in K \mid f(\sigma) \leq p \}$



Total Ordering on K^f:

A sequence $\sigma_1, \sigma_2, \ldots, \sigma_n$ of the simplices of K^f such that:

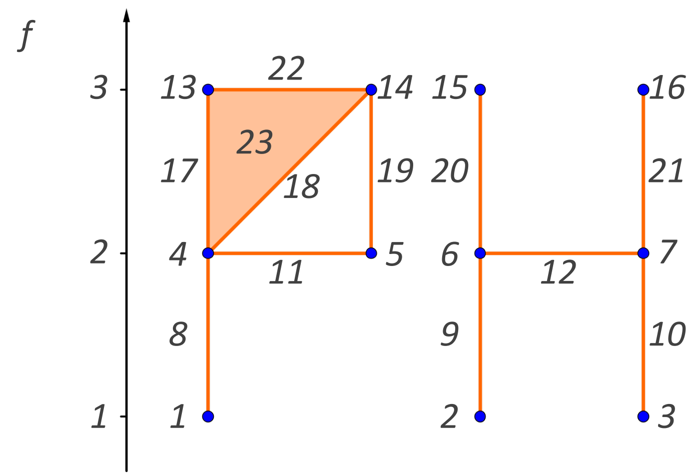
- *if $f(\sigma_i) < f(\sigma_j)$, then $i < j$*
- *if σ_i is a proper face of σ_j, then $i < j$*

Persistent Homology Computation

Given a filtered simplicial complex, let us consider its ***filtering function f*:**

$$f(\sigma) := \min \{ p \mid \sigma \in K^p \}$$

Conversely, $K^p := \{ \sigma \in K \mid f(\sigma) \leq p \}$



A Possible Choice:

Set $\sigma < \sigma'$ if:

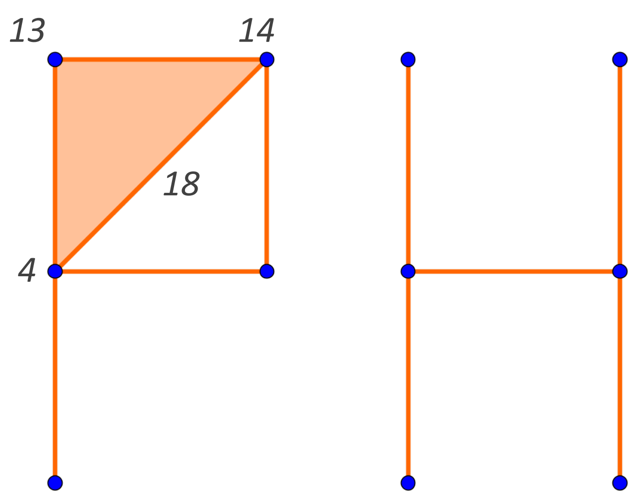
- $f(\sigma) < f(\sigma')$
- $f(\sigma) = f(\sigma')$ and $\dim(\sigma) < \dim(\sigma')$
- $f(\sigma) = f(\sigma')$, $\dim(\sigma) = \dim(\sigma')$, *and σ precedes σ' w.r.t. the* ***lexicographic order*** *of their vertices*

Persistent Homology Computation

Boundary Matrix:

A square matrix ***M*** of size *n* x *n* defined by

$$M_{i,j} := \begin{cases} 1 & if\ \sigma_i\ \ is\ \ a\ \ face\ \ of\ \sigma_j\ \ s.t.\ \ dim(\sigma_i) = dim(\sigma_j) - 1 \\ 0 & otherwise \end{cases}$$



E.g.

- $M_{4,18} = 1$
- $M_{14,18} = 1$
- $M_{13,18} = 0$

Persistent Homology Computation

Reduced Matrix:

Given a non-null column *j* of a boundary matrix *M*,

$$low(j) := \max \{ i \mid M_{i,j} \neq 0 \}$$

A matrix ***R*** *is called* ***reduced*** *if, for each pair of non-null columns* j_1, j_2,

$$low(j_1) \neq low(j_2)$$

Equivalently, if low function is ***injective*** on its domain of definition

Persistent Homology Computation

$i\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1								1															
2									1														
3										1													
4								1			1						1	1					
5											1								1				
6									1			1								1			
7										1		1									1		
8																							
9																							
10																							
11																							
12																							
13																	1					1	
14																		1	1			1	
15																				1			
16																					1		
17																							1
18																							1
19																							
20																							
21																							
22																							1
23																							
low								4	6	7	5	7					13	14	14	15	16	14	22

low(10) = 7 = low(12) ⇒ *M* is **not** reduced

Persistent Homology Computation

Reduction Algorithm:

Matrix $R = M$
`for` $j = 1, \ldots, n$ `do`
  `while` $\exists\ j'' < j$ *with* $low(j') = low(j)$ `do`
    $R.column(j) = R.column(j) + R.column(j')$
  `endwhile`
`endfor`
`return` R

Time Complexity:

At most n^2 column additions ⇒ *$O(n^3)$ in the worst case*

$i \backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1								1															
2									1														
3										1													
4								1			1						1	1					
5											1								1				
6									1			1								1			
7										1		1									1		
8																							
9																							
10																							
11																							
12																							
13																	1					1	
14																		1	1			1	
15																				1			
16																					1		
17																							1
18																							1
19																							
20																							
21																							
22																							1
23																							
low								4	6	7	5	7					13	14	14	15	16	14	22

Initialize ***R*** to ***M*,** where

M is the ***boundary matrix*** of K^f

expressed according with a ***total ordering*** of its simplices

$i\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1								1															
2									1			1											
3										1		1											
4								1			1						1	1					
5											1												
6									1											1			
7										1											1		
8																							
9																							
10																							
11																							
12																							
13																	1						
14																		1					
15																				1			
16																					1		
17																							1
18																							1
19																							
20																							
21																							
22																							1
23																							
low								4	6	7	5	3					13	14		15	16		22

The algorithm returns the above **reduced matrix *R***

Persistent Homology Computation

Retrieving Persistence Pairs:

- For each $i = 1, \ldots, n$,

  if there exists j such that $low(j) = i$ ⇨ **$[i, j]$ is a pair for R**

- Once every i has been parsed,

  if i is an **unpaired** value ⇨ **$[i, \infty)$ is a pair for R**

From pairs of R to the **"actual" persistence pairs of K^f**:

$[i, j]$ corresponds to $[f(\sigma_i), f(\sigma_j)]$

$[i, \infty)$ corresponds to $[f(\sigma_i), \infty)$

(homological degree = $dim(\sigma_i)$)

Persistent Homology Computation

H_0

[1, ∞)

[2, ∞)

[3, 12]

[4, 8]

[5, 11]

[6, 9]

[7, 10]

[13, 17]

[14, 18]

[15, 20]

[16, 21]

H_1

[19, ∞)

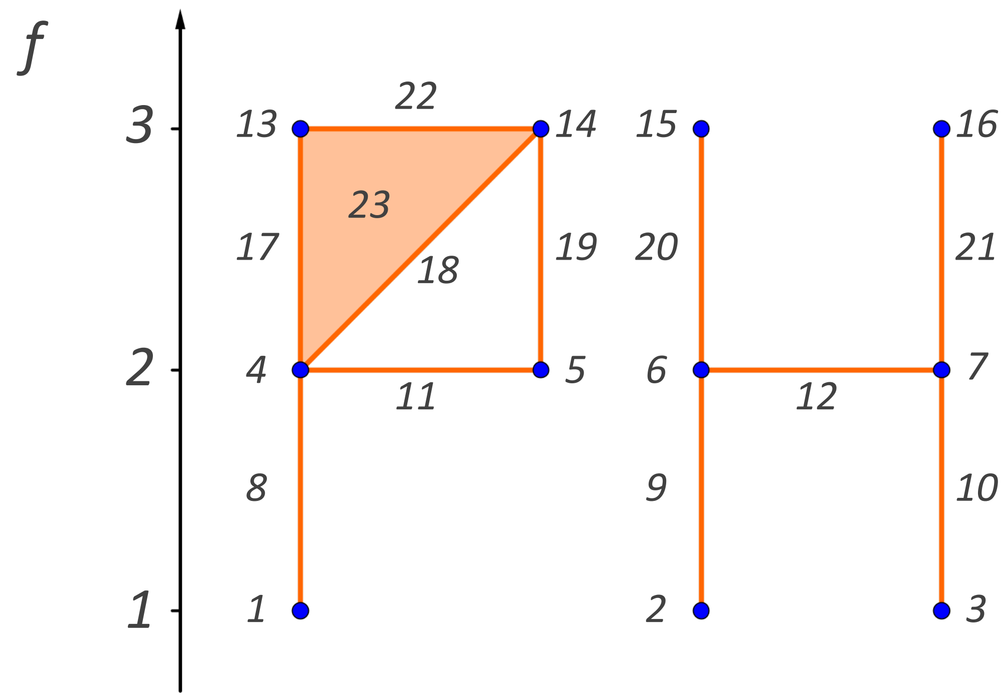
[22, 23]

$i\backslash j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1								1															
2									1			1											
3										1		1											
4								1			1						1	1					
5											1												
6									1											1			
7										1											1		
8																							
9																							
10																							
11																							
12																							
13																	1						
14																		1					
15																				1			
16																					1		
17																							1
18																							1
19																							
20																							
21																							
22																							1
23																							
low								4	6	7	5	3					13	14		15	16		22

Persistent Homology Computation

H_0

[1, ∞)	**[1, ∞)**
[2, ∞)	**[1, ∞)**
[3, 12]	**[1, 2]**
[4, 8]	[2, 2]
[5, 11]	[2, 2]
[6, 9]	[2, 2]
[7, 10]	[2, 2]
[13, 17]	[3, 3]
[14, 18]	[3, 3]
[15, 20]	[3, 3]
[16, 21]	[3, 3]



H_1

[19, ∞)	**[3, ∞)**
[22, 23]	[3, 3]