Topological machine learning: descriptors and stability

Mathieu Carrière
INRIA Sophia-Antipolis
mathieu.carriere@inria.fr
What we’ve seen so far
What we’ve seen so far

General topology: topological equivalences + homology.

\[ H_k = \frac{Z_k}{B_k} \]
What we’ve seen so far

General topology: topological equivalences + homology.

\[ H_k = \frac{Z_k}{B_k} \]

Topological visualization tools: Mapper and Reeb spaces.
What we’ve seen so far

General topology: topological equivalences + homology.

$$H_k = \frac{Z_k}{B_k}$$

Topological visualization tools: Mapper and Reeb spaces.

Topological clustering: ToMATo.
Today: topological descriptors built from data

We will see how to build new topological features from data sets...
Today: topological descriptors built from data

We will see how to build new topological features from data sets...

...but why is that interesting?
Today: topological descriptors built from data

Scans

3D shapes

Magnetometer

Galaxies
Today: topological descriptors built from data

Data often come as (sampling of) metric spaces or sets/spaces endowed with a similarity measure with, possibly complex, topological/geometric structure.

Data carrying geometric information is usually high dimensional.
Today: topological descriptors built from data

Features from **Topological Data Analysis** allow to:
- infer relevant topological and geometric features of these spaces.
- take advantage of topol./geom. information for further processing of data (classification, recognition, learning, clustering, parametrization...).
Challenges and advantages

Problem: how to actually compute the homology groups of a data set?
Challenges and advantages

Problem: how to actually compute the homology groups of a data set?

Challenges and goals:
→ no direct access to topological/geometric information: need of intermediate constructions with simplicial complexes;
→ distinguish topological “signal” from noise;
→ topological information may be multiscale;
→ statistical analysis of topological information.
Challenges and advantages

Advantages:
→ **coordinate invariance:** topological features/invariants do not rely on any coordinate system ⇒ no need to have data with coordinates, or to embed data in spaces with coordinates... but the metric (distance/similarity between data points) is important.
→ **deformation invariance:** topological features are invariant under homeomorphism and reparameterization.
→ **compressed representation:** topology offers a set of tools to summarize the data in compact ways while preserving its topological structure.
Persistent homology

What is persistent homology?
What is persistent homology?

→ a mathematical framework for encoding the evolution of the topology (homology) of families of nested spaces (filtered complex, sublevel sets, ...).

→ formalized by H. Edelsbrunner et al. (2002) and G. Carlsson et al. (2005) - wide development during the last decade.
Persistent homology

What is persistent homology?

→ barcodes/persistence diagrams can be efficiently computed.
→ multiscale topological information.
→ stability properties.
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (*filtration*) of sublevel-sets $f^{-1}((-\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (filtration) of sublevel-sets $f^{-1}((-\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family

![Diagram of a function $f$ and real number line $\mathbb{R}$]
Recall: sublevel sets of function

Intuition of persistence:

- Nested family \((\text{filtration})\) of sublevel-sets \(f^{-1}((−\infty, t])\) for \(t\) ranging over \(\mathbb{R}\)
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (*filtration*) of sublevel-sets $f^{-1}((−∞, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (filtration) of sublevel-sets $f^{-1}((-\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (filtration) of sublevel-sets $f^{-1}((\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (*filtration*) of sublevel-sets $f^{-1}((\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (*filtration*) of sublevel-sets $f^{-1}((\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (filtration) of sublevel-sets $f^{-1}((-\infty, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
- Finite set of intervals (barcode) encodes births/deaths of topological features
Recall: sublevel sets of function

Intuition of persistence:

- Nested family (*filtration*) of sublevel-sets $f^{-1}((−∞, t])$ for $t$ ranging over $\mathbb{R}$
- Track the evolution of the topology (homology) throughout the family
- Finite set of intervals (barcode) encodes births/deaths of topological features

```
\begin{itemize}
  \item Alternate representation as a (multi-) set of points in the plane (*persistence diagram*).
\end{itemize}
```
Recall: sublevel sets of distance $= \text{growing balls}$

\[ f_P : \mathbb{R}^2 \rightarrow \mathbb{R} \]
\[ x \mapsto \min_{p \in P} \| x - p \|_2 \]
Recall: sublevel sets of distance $= \text{growing balls}$

$$f_P : \mathbb{R}^2 \to \mathbb{R}$$

$$x \mapsto \min_{p \in P} \| x - p \|_2$$
This example shows one of the connections to topological data analysis. Other connections happen through the study of density estimators (cf. ToMATo).

Recall: sublevel sets of distance = growing balls

\[ f_P : \mathbb{R}^2 \rightarrow \mathbb{R} \]
\[ x \mapsto \min_{p \in P} \|x - p\|_2 \]
Recall: sublevel sets of distance $\equiv$ growing balls

\[ f_P : \mathbb{R}^2 \to \mathbb{R} \]
\[ x \mapsto \min_{p \in P} \| x - p \|_2 \]
Recall: sublevel sets of distance = growing balls

$$f_P : \mathbb{R}^2 \to \mathbb{R}$$
$$x \mapsto \min_{p \in P} \| x - p \|_2$$
Recall: sublevel sets of distance $= \text{growing balls}$

$$f_P : \mathbb{R}^2 \to \mathbb{R}$$

$$x \mapsto \min_{p \in P} \|x - p\|_2$$
Recall: sublevel sets of distance = growing balls

\[ f_P : \mathbb{R}^2 \to \mathbb{R} \]
\[ x \mapsto \min_{p \in P} \| x - p \|_2 \]
Recall: sublevel sets of distance $= \text{growing balls}$

$$f_P : \mathbb{R}^2 \to \mathbb{R}$$
$$x \mapsto \min_{p \in P} \|x - p\|_2$$
Recall: sublevel sets of distance $= \text{growing balls}$

$$f_P : \mathbb{R}^2 \to \mathbb{R}$$

$$x \mapsto \min_{p \in P} \|x - p\|_2$$

3 pillars of persistence theory:

- decomposition theorems (barcode existence)
- persistence algorithm (barcode calculation)
- stability theorem (barcode stability)
Filtered complexes and filtrations

Def: A filtered simplicial complex $S$ is a family $\{S_a\}_{a \in \mathbb{R}}$ of subcomplexes of some fixed simplicial complex $S$ s.t. $S_a \subseteq S_b$ for any $a \leq b$.

Def: A filtration $F$ of a space $X$ is a family $\{F_a\}_{a \in \mathbb{R}}$ of subspaces of $X$ s.t. $F_a \subseteq F_b$ for any $a \leq b$. 
Mathematical foundations

Filtration: $F_1 \subseteq F_2 \subseteq F_3 \subseteq F_4 \subseteq F_5 \cdots$

Example 1: *offsets filtration* (nested family of unions of balls)
Mathematical foundations

Filtration: \( F_1 \subseteq F_2 \subseteq F_3 \subseteq F_4 \subseteq F_5 \cdots \)

Example 1: offsets filtration (nested family of unions of balls)

Example 2: simplicial filtration (nested family of simplicial complexes)

\[
\begin{array}{cccccc}
\text{a} & \text{b} \\
\text{d} & \text{c} \\
F_1 & F_2 & F_3 & F_4 & F_5 & F_6
\end{array}
\]

**Def:** Let \( f \) be a real valued function defined on the vertices of \( K \). For \( \sigma = [v_0, \ldots, v_k] \in K \), let \( f(\sigma) = \max_{i=0,\ldots,k} f(v_i) \), and order the simplices of \( K \) in increasing order w.r.t. the function \( f \) values (and break ties with dimension in case some simplices have the same function value).

**Q:** Show that this is a filtration.
Mathematical foundations

Filtration: \( F_1 \subseteq F_2 \subseteq F_3 \subseteq F_4 \subseteq F_5 \cdots \)

Example 1: *offsets filtration* (nested family of unions of balls)

Example 2: *simplicial filtration* (nested family of simplicial complexes)

Example 3: *sublevel-sets filtration* (family of sublevel sets of function \( f \))

\[ F_\alpha := f^{-1}((-\infty, \alpha]) \]
Mathematical foundations

Filtration: \[ F_1 \subseteq F_2 \subseteq F_3 \subseteq F_4 \subseteq F_5 \cdots \]

\[
H_*(F_1) \to H_*(F_2) \to H_*(F_3) \to H_*(F_4) \to H_*(F_5) \to \cdots
\]

**Def:** A *persistence module* is a sequence of vector spaces connected with linear maps:

\[
H_*(F_1) \to H_*(F_2) \to H_*(F_3) \to H_*(F_4) \to \cdots
\]
Mathematical foundations

Example:

\[ \begin{array}{c}
\triangle \subseteq \quad \triangle \subseteq \quad \triangle \subseteq \quad \triangle \subseteq \\
(1, 0) \rightarrow \quad (0, 1) \rightarrow \\
\end{array} \]

(\text{degree-1 homology})
Mathematical foundations

Example:

\[
\begin{array}{cccc}
\triangledown & \subseteq & \triangledown & \subseteq \\
\triangledown & \subseteq & \triangledown & \subseteq \\
\end{array}
\]

\[
\mathbb{Z}_2 \to \mathbb{Z}_2^2 \to \mathbb{Z}_2 \to \mathbb{Z}_2^2
\]

(degree-1 homology)
Example:

\[
\begin{align*}
\mathbb{Z}_2 & \subseteq \mathbb{Z}_2 \times \mathbb{Z}_2 \subseteq \mathbb{Z}_2 \times \mathbb{Z}_2 \subseteq \mathbb{Z}_2 \times \mathbb{Z}_2 \subseteq \mathbb{Z}_2 \times \mathbb{Z}_2 \times \mathbb{Z}_2 \\
\end{align*}
\]

(Mathematical foundations)

\[
\begin{align*}
\mathbb{Z}_2 & \longmapsto \mathbb{Z}_2^2 \longmapsto \mathbb{Z}_2 \longmapsto \mathbb{Z}_2^2 \longmapsto \mathbb{Z}_2^2 \longmapsto \cdots
\end{align*}
\]

(degree-1 homology)
Mathematical foundations

**Thm:** Let $M$ be a persistence module over an index set $T \subseteq \mathbb{R}$. Then, $M$ decomposes as a direct sum of *interval modules* $\mathbb{Z}_2[b,d]$:

$$
0 \to \cdots \to 0 \to \mathbb{Z}_2 \xrightarrow{id} \cdots \xrightarrow{id} \mathbb{Z}_2 \to 0 \to 0 \to \cdots \to 0
$$

$\forall t < [b,d]$

$\forall [b, d]$

$\forall t > [b,d]$

$M \simeq \bigoplus_{j \in J} \mathbb{Z}_2[b_j,d_j]\,$

(The barcode is a complete descriptor of the algebraic structure of $M$)

[The structure and stability of persistence modules, Chazal, de Silva, Glisse, Oudot, Springer, 2016].
**Thm:** Let $M$ be a persistence module over an index set $T \subseteq \mathbb{R}$. Then, $M$ decomposes as a direct sum of *interval modules* $\mathbb{Z}_2[b,d]$: 

\[
\begin{array}{cccccccccc}
0 & \to & \cdots & \to & 0 & \to & \mathbb{Z}_2 & \to & \cdots & \to & \mathbb{Z}_2 & \to & 0 \\
\hspace{1cm} t<[b,d] & & & & & \hspace{1cm} [b,d] & & & & & \hspace{1cm} t>[b,d]
\end{array}
\]

in the following cases:

- $T$ is finite,
- $M$ is *pointwise finite-dimensional* (pfd), i.e., every space $M_t$ has finite dimension.

Moreover, when it exists, the decomposition is unique up to isomorphism and permutation of the terms [Azumaya 1950].
Mathematical foundations

Example:

$$\begin{array}{cccc}
\triangle \subseteq \triangle \subseteq \triangle \subseteq \triangle \subseteq \triangle \subseteq \text{hexagon}
\end{array}$$

$$(\begin{pmatrix} 1 \\ 0 \end{pmatrix}) \rightarrow \mathbb{Z}_2 \rightarrow \mathbb{Z}_2^2 \rightarrow \mathbb{Z}_2 \rightarrow \mathbb{Z}_2^2 \rightarrow \mathbb{Z}_2^2 \rightarrow \cdots$$

(degree-1 homology)
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

positive, i.e., it creates a new homology class

negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

- positive, i.e., it creates a new homology class
- negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:
positive, i.e., it creates a new homology class
negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

- positive, i.e., it creates a new homology class
- negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

- positive, i.e., it creates a new homology class
- negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

positive, i.e., it creates a new homology class

negative, i.e., it destroys an homology class

\begin{align*}
\text{1} & \quad \text{3} \\
\text{4} & \quad \text{2} \\
\text{5} & \quad \text{6} \\
\text{7} & \quad \text{3} \\
\text{1} & \quad \text{4} \\
\text{1} & \quad \text{3} \\
\text{1} & \quad \text{2} \\
\end{align*}
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

*positive*, i.e., it *creates a new homology class*

*negative*, i.e., it *destroys an homology class*
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

- positive, i.e., it creates a new homology class
- negative, i.e., it destroys an homology class
Computation with matrix reduction

Input: simplicial filtration

(Persistent) homology can be computed by using the fact that each simplex is either:

- **positive**, i.e., it *creates a new homology class*
- **negative**, i.e., it *destroys an homology class*

Q: Do the same for the homology of the cube.
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Computation with matrix reduction

Input: simplicial filtration
given as *boundary matrix*

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Diagram of a simplicial complex with vertices 1, 2, 3, 4, 5, 6, 7 and edges connecting them.
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & & & \bullet & & & \\
2 & & \bullet & & \bullet & & \\
3 & & & & \bullet & & \\
4 & & & & & & \\
5 & & & & & & \\
6 & & & & & & \\
7 & & & & & & \\
\end{array}
\]
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Diagram of a simplicial complex with vertices labeled 1 to 7 and facets labeled 1 to 7.
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Diagram of simplicial complex:

- Vertices: 1, 2, 3, 4, 5, 6, 7
- Edges: (1, 2), (1, 3), (2, 4), (3, 4), (4, 5), (5, 6), (6, 7)
- Face: 1, 2, 3

[Diagram of simplicial complex showing a triangle formed by vertices 1, 2, and 3 with edges connecting these vertices.]
Computation with matrix reduction

Input: simplicial filtration

given as *boundary matrix*

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>●</td>
<td></td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>●</td>
<td>●</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>●</td>
<td>●</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>●</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>●</td>
<td></td>
</tr>
</tbody>
</table>

for \( j = 1 \) to \( m \) do:

\[
\text{while } \exists k < j \text{ s.t. } \text{low}(k) = \text{low}(j) \text{ do:}
\]

\[
\text{col}(j) = \text{col}(j) + \text{col}(k)
\]
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 & & & & & & \\
2 & & & & & & \\
3 & & & & & & \\
4 & & & & & & \\
5 & & & & & & \\
6 & & & & & & \\
7 & & & & & & \\
\end{array}
\]

for \( j = 1 \) to \( m \) do:

while \( \exists k < j \) s.t. \( \text{low}(k) = \text{low}(j) \) do:

\[
\text{col}(j) = \text{col}(j) + \text{col}(k)
\]

\[
\text{low}(j) = j'
\]
Computation with matrix reduction

Input: simplicial filtration

given as boundary matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

for $j=1$ to $m$ do:

while $\exists k < j$ s.t. $\text{low}(k) = \text{low}(j)$ do:

$\text{col}(j) = \text{col}(j) + \text{col}(k)$

$6 = 6 + 5$

$\text{low}(j) = j'$
Computation with matrix reduction

Input: simplicial filtration

given as boundary matrix

\[ \begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 & \cdot & \cdot & \cdot & & & \\
2 & \cdot & \cdot & \cdot & & & \\
3 & & & \cdot & & & \\
4 & & & & \cdot & & \\
5 & & & & & \cdot & \\
6 & & & & & & \cdot \\
7 & & & & & & & \\
\end{array} \]

for \( j = 1 \) to \( m \) do:

\[ \text{while } \exists k < j \text{ s.t. } \text{low}(k) = \text{low}(j) \text{ do:} \]

\[ \text{col}(j) = \text{col}(j) + \text{col}(k) \]

\[ 6 = 6 + 5 \]

\[ \text{low}(j) = j' \]
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

\[
\begin{array}{ccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 & & & & & & \\
2 & & & & & & \\
3 & & & & & & \\
4 & & & & & & \\
5 & & & & & & \\
6 & & & & & & \\
7 & & & & & & \\
\end{array}
\]

\[
\begin{array}{cccc}
5 & 6 \\
\cdot & \cdot \\
\cdot & \cdot \\
\cdot & \cdot \\
\cdot & \\
\cdot & \\
\cdot & \\
\end{array}
\quad \begin{array}{cccc}
4 & 6 \\
\cdot & \cdot \\
\cdot & \cdot \\
\cdot & \cdot \\
\cdot & \\
\cdot & \\
\cdot & \\
\end{array}
\]

for \( j=1 \) to \( m \) do:

while \( \exists k < j \) s.t. \( \text{low}(k) = \text{low}(j) \) do:

\[
\text{col}(j) = \text{col}(j) + \text{col}(k)
\]

\[
\text{low}(j) = j'
\]

\[
\begin{array}{cccc}
& & & \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\end{array}
\]

\[
\begin{array}{cccc}
& & & \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot \\
\end{array}
\]

\[
6 = 6+5
\]

\[
6 = 6+4
\]
Computation with matrix reduction

Input: simplicial filtration
given as boundary matrix

\[
\begin{array}{cccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 &   &   &   &   &   &   \\
2 &   &   &   &   &   &   \\
3 &   &   &   &   &   &   \\
4 &   &   &   &   &   &   \\
5 &   &   &   &   &   &   \\
6 &   &   &   &   &   &   \\
7 &   &   &   &   &   &   \\
\end{array}
\]

for \( j = 1 \) to \( m \) do:

\[
\text{while } \exists k < j \text{ s.t. } \text{low}(k) = \text{low}(j) \text{ do:}
\]

\[
\text{col}(j) = \text{col}(j) + \text{col}(k)
\]

\[
\text{low}(j) = j'
\]

\[
6 = 6 + 5
\]

\[
6 = 6 + 4
\]
Computation with matrix reduction

Input: simplicial filtration
Output: boundary matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Computation with matrix reduction

**Input:** simplicial filtration

**Output:** boundary matrix

reduced to column-echelon form

### Example

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>*</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Computation with matrix reduction

Input: simplicial filtration
Output: boundary matrix

- simplex pairs give finite intervals: 
  \([2, 4), [3, 5), [6, 7)\]
- unpaired simplices give infinite intervals: \([1, +\infty)\)

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 &   &   & * &   &   &   & * \\
2 &   & * &   & * &   &   &   \\
3 &   &   & * &   &   &   & * \\
4 &   &   &   &   & * &   &   \\
5 &   &   &   &   &   & * &   \\
6 &   &   &   &   &   &   & * \\
7 &   &   &   &   &   &   &   \\
\end{array}
\]

\[
\begin{array}{cccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 \\
1 &   & * &   &   &   &   &   \\
2 &   &   & * &   &   &   &   \\
3 &   &   &   & * &   &   &   \\
4 & 1 &   &   &   &   &   &   \\
5 &   &   &   &   & * &   &   \\
6 &   &   &   &   &   & * &   \\
7 &   &   &   &   &   &   &   \\
\end{array}
\]
Computation with matrix reduction

Input: simplicial filtration
Output: boundary matrix
reduced to column-echelon form

Q: Complexity?
Computation with matrix reduction

Input: simplicial filtration
Output: boundary matrix
        reduced to column-echelon form

Q: Complexity?

PLU factorization:
  • Gaussian elimination
  • fast matrix multiplication (divide-and-conquer)
  • random projections?
Computation with matrix reduction

Input: simplicial filtration
Output: boundary matrix
       reduced to column-echelon form

Q: Complexity?

PLU factorization:

• Gaussian elimination
  - PLEX / JavaPLEX (http://appliedtopology.github.io/javaplex/)
  - Dionysus (http://www.mrzv.org/software/dionysus/)
  - Perseus (http://www.sas.upenn.edu/~vnanda/perseus/)
  - Gudhi (http://gudhi.gforge.inria.fr/)
  - PHAT (https://bitbucket.org/phat-code/phat)
  - DIPHA (https://github.com/DIPHA/dipha/)
  - CTL (https://github.com/appliedtopology/ctl)
**Thm:** For any pfd functions $f, g : X \to \mathbb{R}$ and homological dimension $k$,

$$d_B(D_k(f), D_k(g)) \leq \|f - g\|_\infty,$$

where $\|f - g\|_\infty = \sup_x |f(x) - g(x)|$. 

**Stability properties**
Stability properties

Persistence diagram \( \equiv \textbf{finite} \) multiset in the open half-plane \( \Delta \times \mathbb{R}_{>0} \).
Stability properties

Persistence diagram \(\equiv\) finite multiset in the open half-plane \(\Delta \times \mathbb{R}_{>0}\).

Given a **partial matching** \(M : A \leftrightarrow B\):

- cost of a matched pair \((a, b) \in M\): \(c_p(a, b) := \|a - b\|_\infty^p\),
- cost of an unmatched point \(c \in A \sqcup B\): \(c_p(c) := \|c - \overline{c}\|_\infty^p\),
- **cost of** \(M\):

\[
c_p(M) := \left( \sum_{(a, b) \text{ matched}} c_p(a, b) + \sum_{c \text{ unmatched}} c_p(c) \right)^{1/p}
\]
Stability properties

Persistence diagram \(\equiv\) finite multiset in the open half-plane \(\Delta \times \mathbb{R}_{>0}\).

Given a partial matching \(M : A \leftrightarrow B\):

- cost of a matched pair \((a, b) \in M\): \(c_p(a, b) := \|a - b\|_p^n\),
- cost of an unmatched point \(c \in A \sqcup B\): \(c_p(c) := \|c - \bar{c}\|_p^n\),
- cost of \(M\):

\[
c_p(M) := \left( \sum_{(a, b) \text{ matched}} c_p(a, b) + \sum_{c \text{ unmatched}} c_p(c) \right)^{1/p}
\]

Def: \(p\)-th diagram distance (extended metric):

\[
d_p(A, B) := \inf_{M : A \leftrightarrow B} c_p(M)
\]

Def: bottleneck distance:

\[
d_B(A, B) = d_\infty(A, B) := \lim_{p \to \infty} d_p(A, B)
\]
Def: Let $V$ be a point cloud (in a metric space). The Čech complex $\check{C}ech(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

$$\sigma = [p_0, p_1 \ldots, p_k] \in \check{C}ech(V, \alpha) \iff \cap_{i=0}^{k} B(p_i, \alpha) \neq \emptyset$$
**Def:** Let $V$ be a point cloud (in a metric space $(X,d)$). The **Vietoris-Rips complex** $\text{Rips}(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

\[
\sigma = [p_0, p_1, \ldots, p_k] \in \text{Rips}(V, \alpha) \iff \forall i, j \in \{0, \ldots, k\}, \ d(p_i, p_j) \leq \alpha
\]
Def: Let $V$ be a point cloud (in a metric space $(X, d)$). The Vietoris-Rips complex $\text{Rips}(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

$$\sigma = [p_0, p_1, \ldots, p_k] \in \text{Rips}(V, \alpha) \iff \forall i, j \in \{0, \ldots, k\}, \ d(p_i, p_j) \leq \alpha$$

Easy to compute and fully characterized by its 1-skeleton.
**Def:** Let $V$ be a point cloud (in a metric space $(X,d)$). The Vietoris-Rips complex $\text{Rips}(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

$$\sigma = [p_0, p_1, \ldots, p_k] \in \text{Rips}(V, \alpha) \iff \forall i, j \in \{0, \ldots, k\}, \ d(p_i, p_j) \leq \alpha$$

Easy to compute and fully characterized by its 1-skeleton.

**Rips-Čech interleaving:** for any $\alpha > 0$,

$$\text{Čech}(V, \alpha/2) \subseteq \text{Rips}(V, \alpha) \subseteq \text{Čech}(V, \alpha)$$
PH for point cloud: summary

- build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
- compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
- compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
- statistical properties of signatures.

Advantages of Čech/Rips filtrations:

- build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
- compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
- compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
- statistical properties of signatures.
Advantages of Čech/Rips filtrations:

- build a geometric filtered simplicial complex on top of \( \hat{X}_n \) \( \rightarrow \) multiscale topol. structure.
- compute the persistent homology of the complex \( \rightarrow \) multiscale topol. signature.
- compare the signatures of “close” data sets \( \rightarrow \) robustness and stability results.
- statistical properties of signatures.
PH for point cloud: summary

- build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
- compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
- compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
- statistical properties of signatures.

Advantages of Čech/Rips filtrations:

- build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
- compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
- compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
- statistical properties of signatures.
PH for point cloud: summary

Advantages of Čech/Rips filtrations:

• build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
• compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
• compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
• statistical properties of signatures.
PH for point cloud: summary

Advantages of Čech/Rips filtrations:

- build a geometric filtered simplicial complex on top of $\hat{X}_n \rightarrow$ multiscale topol. structure.
- compute the persistent homology of the complex $\rightarrow$ multiscale topol. signature.
- compare the signatures of “close” data sets $\rightarrow$ robustness and stability results.
- statistical properties of signatures.
**Thm:** If $X$ and $Y$ are pre-compact metric spaces, then

$$d_B(D_k(\text{Rips}(X)), D_k(\text{Rips}(Y))) \leq d_{GH}(X, Y).$$

**Rem:** This result also holds for other families of filtrations (particular case of a more general theorem).
Application: non rigid shape classification

- Non rigid shapes in a same class are almost isometric, but computing Gromov-Hausdorff distance between shapes is extremely expensive.
- Compare diagrams of sampled shapes instead of shapes themselves.

Limitations

**Thm:** If $X$ and $Y$ are pre-compact metric spaces, then

$$d_B(D_k(\text{Rips}(X)), D_k(\text{Rips}(Y))) \leq d_{GH}(X, Y).$$

→ Vietoris-Rips (or Čech, witness) filtrations become quickly prohibitively large as the size of the data increases: $O(|X|^d)$, making the practical computation of persistence almost impossible.
Limitations

**Thm:** If \( X \) and \( Y \) are pre-compact metric spaces, then

\[
\text{d}_B(D_k(\text{Rips}(X)), D_k(\text{Rips}(Y))) \leq d_{GH}(X, Y).
\]

→ Vietoris-Rips (or Čech, witness) filtrations become quickly prohibitively large as the size of the data increases: \( O(|X|^d) \), making the practical computation of persistence almost impossible.

→ Persistence diagrams of Vietoris-Rips (as well as Čech, witness,..) filtrations and Gromov-Hausdorff distance are very sensitive to noise and outliers.
Limitations

**Thm:** If $X$ and $Y$ are pre-compact metric spaces, then

$$d_B(D_k(\text{Rips}(X)), D_k(\text{Rips}(Y))) \leq d_{GH}(X, Y).$$

→ Vietoris-Rips (or Čech, witness) filtrations become quickly prohibitively large as the size of the data increases: $O(|X|^d)$, making the practical computation of persistence almost impossible.

→ Persistence diagrams of Vietoris-Rips (as well as Čech, witness,..) filtrations and Gromov-Hausdorff distance are very sensitive to noise and outliers.
The Distance To Measure (DTM)

Preliminary distance function to a measure $P$: let $u \in ]0,1[$ be a positive mass, and $P$ a probability measure on $\mathbb{R}^d$:

$$\delta_{P,u}(x) = \inf\{ r > 0 : P(B(x,r)) \geq u \}$$
The Distance To Measure (DTM)

Preliminary distance function to a measure $P$: let $u \in ]0, 1[$ be a positive mass, and $P$ a probability measure on $\mathbb{R}^d$:

$$\delta_{P,u}(x) = \inf\{r > 0 : P(B(x,r)) \geq u\}$$

$\delta_{P,u}$ is the smallest distance needed to capture a mass of at least $u$. 

The Distance To Measure (DTM)

Preliminary distance function to a measure $P$: let $u \in ]0, 1[$ be a positive mass, and $P$ a probability measure on $\mathbb{R}^d$:

$$\delta_{P,u}(x) = \inf\{r > 0 : P(B(x, r)) \geq u\}$$

$\delta_{P,u}$ is the smallest distance needed to capture a mass of at least $u$.

$\delta_{P,u}$ is the quantile function at $u$ of the r.v. $\|x - X\|$ where $X \sim P$. 

**The Distance To Measure (DTM)**

**Preliminary distance function to a measure** $P$: let $u \in ]0, 1]$ be a positive mass, and $P$ a probability measure on $\mathbb{R}^d$:

$$\delta_{P,u}(x) = \inf\{r > 0 : P(B(x,r)) \geq u\}$$

**Def:** Given a probability measure $P$ on $\mathbb{R}^d$ and $m > 0$, the distance function to the measure $P$ (DTM) is defined by

$$d_{P,m} : x \in \mathbb{R} \mapsto \left(\frac{1}{m} \int_0^m \delta_{P,u}^2(x) \, du\right)^{1/2}$$
The Distance To Measure (DTM)

Preliminary distance function to a measure $P$: let $u \in ]0, 1[$ be a positive mass, and $P$ a probability measure on $\mathbb{R}^d$:

$$\delta_{P,u}(x) = \inf\{r > 0 : P(B(x, r)) \geq u\}$$

**Def:** Given a probability measure $P$ on $\mathbb{R}^d$ and $m > 0$, the distance function to the measure $P$ (DTM) is defined by

$$d_{P,m} : x \in \mathbb{R} \mapsto \left( \frac{1}{m} \int_0^m \delta_{P,u}^2(x) du \right)^{1/2}$$

The DTM is robust, i.e., stable under Wasserstein perturbations:

$$\|d_{P,m} - d_{Q,m}\|_{\infty} \leq \frac{1}{\sqrt{m}} W_2(P, Q)$$
The Distance To Measure (DTM)

Def: Let $X_1, \ldots, X_n$ sampled according to $P$ and let $P_n$ be the empirical measure. Then

$$d_{P_n,k/n}(x) = \frac{1}{k} \sum_{i=1}^{k} \|x - X(i)\|^2,$$

where $\|X(1) - x\| \leq \|X(2) - x\| \leq \cdots \leq \|X(k) - x\| \leq \cdots \leq \|X(n) - x\|$. [Geometric inference for probability measures, Chazal, Cohen-Steiner, Mérigot, Found. Comput. Math., 2011]
The Wasserstein distance

Let \((X, d)\) be a metric space and let \(\mu, \nu\) be probability measures on \(X\) with finite \(p\)-moments \((p \geq 1)\). The Wasserstein distance \(W_p(\mu, \nu)\) quantifies the optimal cost of pushing \(\mu\) onto \(\nu\), the cost of moving a small mass \(dx\) from \(x\) to \(y\) being \(d(x, y)^p dx\).

- Transport plan: \(\Pi\) a probability measure on \(X \times X\) s.t. \(\Pi(A \times \mathbb{R}^d) = \mu(A)\) and \(\Pi(\mathbb{R}^d \times B) = \nu(B)\) for any borelian sets \(A, B \subseteq X\).
- Cost of a transport plan:
  \[
  C(\Pi) = \left( \int_{X \times X} d(x, y)^p d\Pi(x, y) \right)^{\frac{1}{p}}
  \]
- \(W_p(\mu, \nu) = \inf_{\Pi} C(\Pi)\).
The Wasserstein distance

Ex: If $P = \{p_1, \ldots, p_n\}$ is a point cloud, and $P' = \{p_1, \ldots, p_{n-k-1}, o_1, \ldots, o_k\}$ with $d(o_i, P) = R$, then

$$d_H(P, P') \geq R \quad \text{but} \quad W_2(\mu_P, \mu_{P'}) \leq \sqrt{\frac{k}{n}}(R + \text{diam}(P))$$
**DTM-based filtrations**


**Def:** Let $V$ be a point cloud (in a metric space). The **DTM-based complex** $W(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

$$
\sigma = [p_0, p_1 \ldots, p_k] \in W(V, \alpha) \iff \bigcap_{i=0}^{k} B(p_i, r_{p_i}(\alpha)) \neq \emptyset
$$

where $r_{p}(\alpha) = 0$ if $\alpha \leq d_{P_n,k/n}(p)$ and $|\alpha^q - d_{P_n,k/n}(p)^q|^{1/q}$ otherwise.
**Def:** Let $V$ be a point cloud (in a metric space). The **DTM-based complex $W(V)$** is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

$$\sigma = [p_0, p_1 \ldots, p_k] \in W(V, \alpha) \iff \cap_{i=0}^k B(p_i, r_{p_i}(\alpha)) \neq \emptyset$$

where $r_p(\alpha) = 0$ if $\alpha \leq d_{P_n,k/n}(p)$ and $|\alpha^q - d_{P_n,k/n}(p)^q|^{1/q}$ otherwise.
DTM-based filtrations

Def: Let $V$ be a point cloud (in a metric space). The DTM-based complex $W(V)$ is the filtered simplicial complex indexed by $\mathbb{R}$ whose vertex set is $V$ and whose other simplices are defined with

\[ \sigma = [p_0, p_1, \ldots, p_k] \in W(V, \alpha) \iff \bigcap_{i=0}^{k} B(p_i, r_{p_i}(\alpha)) \neq \emptyset \]

where $r_p(\alpha) = 0$ if $\alpha \leq d_{P_n, k/n}(p)$ and $|\alpha^q - d_{P_n, k/n}(p)^q|^{1/q}$ otherwise.

Thm: $d_B(W(X), W(Y)) \leq \sqrt{\frac{n}{k}} W_2(X, Y) + 2^{1/q} d_H(X, Y)$. 