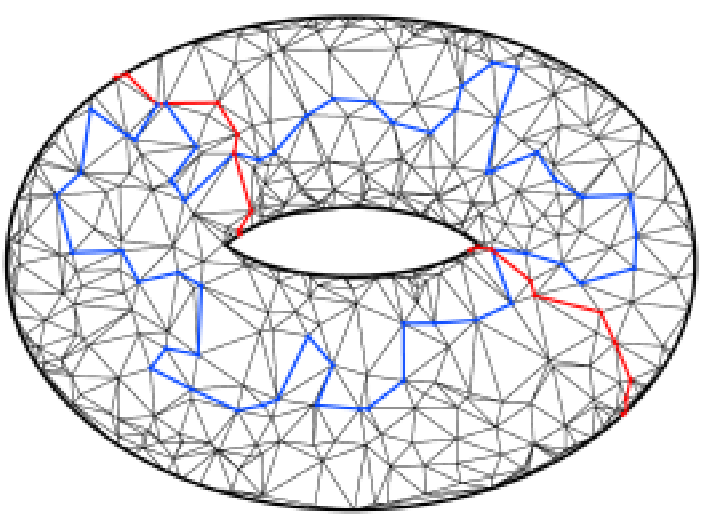
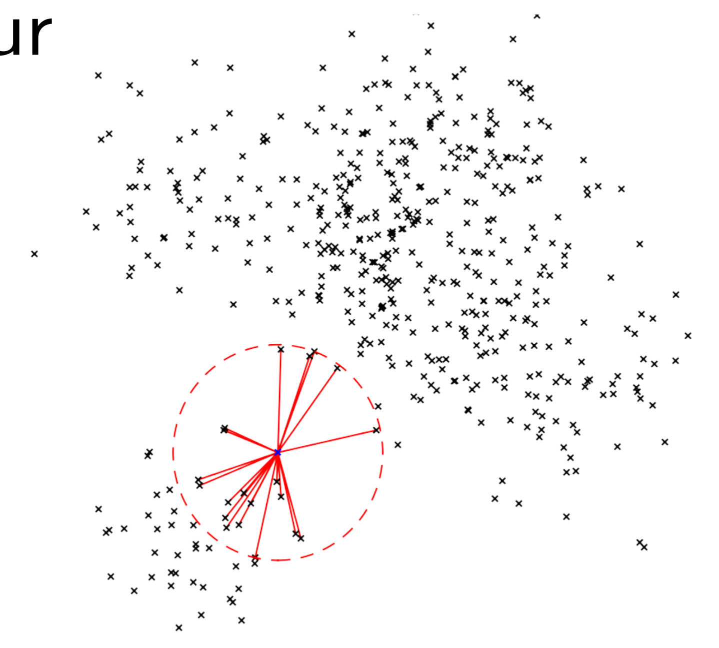
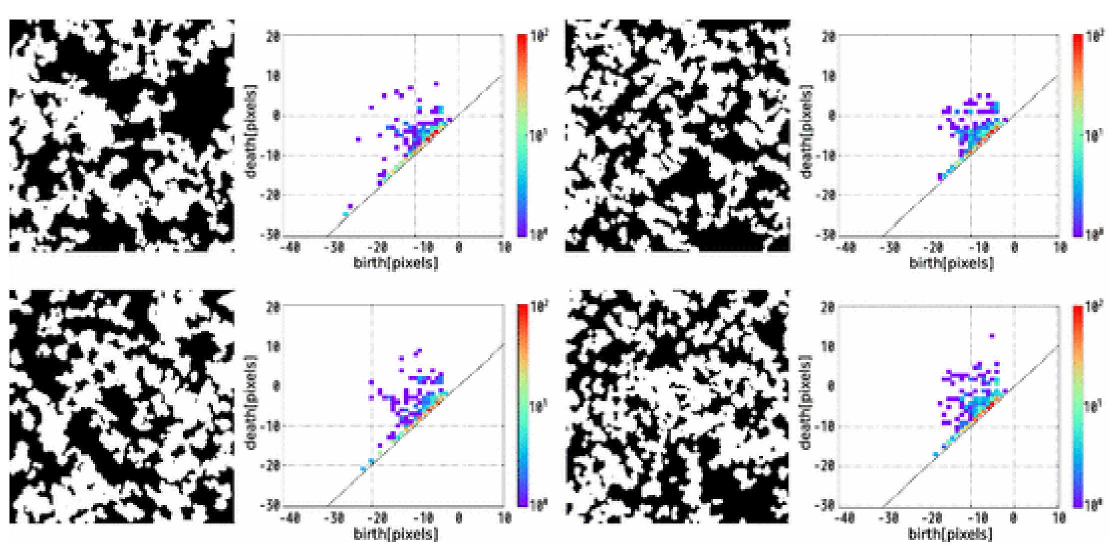
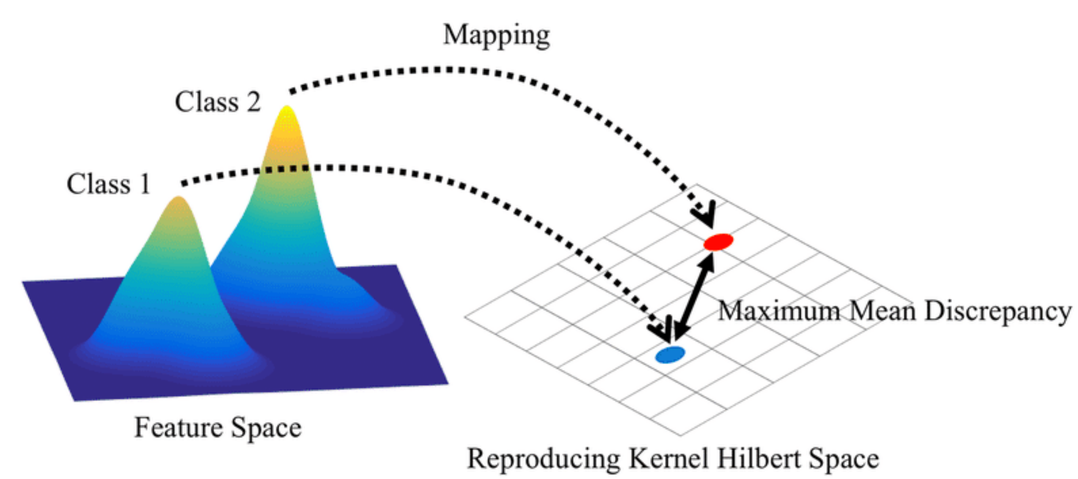
Foundations of Geometric Methods in Data Analysis

Instructors:
Mathieu Carrière & Frédéric Cazals
Centre Inria d'Université Côte d'Azur
`firstname.lastname@inria.fr`









Information about the class

Information about the class

Class outline (each class is 50% lecture 50% practical session)

- 1. Computational Topology (I): Simplicial Complexes
- 2. Nearest Neighbors in Euclidean and metric spaces (I): Data Structures and Algorithms
- 3. Nearest Neighbors in Euclidean and metric spaces (II): Analysis
- 4. Comparing Samplings, Distributions, Clusterings
- 5. Computational Topology (II): Persistence Theory
- 6. Topological Machine Learning (I): An Introduction
- 7. Topological Machine Learning (II): Advanced Topics
- 8. Dimensionality Reduction Algorithms

Information about the class

Class outline (each class is 50% lecture 50% practical session)

- 1. Computational Topology (I): Simplicial Complexes
- 2. Nearest Neighbors in Euclidean and metric spaces (I): Data Structures and Algorithms
- 3. Nearest Neighbors in Euclidean and metric spaces (II): Analysis
- 4. Comparing Samplings, Distributions, Clusterings
- 5. Computational Topology (II): Persistence Theory
- 6. Topological Machine Learning (I): An Introduction
- 7. Topological Machine Learning (II): Advanced Topics
- 8. Dimensionality Reduction Algorithms

Website: `http://www-sop.inria.fr/abs/teaching/centrale-FGMDA/centrale-FGMDA--cazals-carriere.html`

Information about the class

Class outline (each class is 50% lecture 50% practical session)

- 1. Computational Topology (I): Simplicial Complexes
- 2. Nearest Neighbors in Euclidean and metric spaces (I): Data Structures and Algorithms
- 3. Nearest Neighbors in Euclidean and metric spaces (II): Analysis
- 4. Comparing Samplings, Distributions, Clusterings
- 5. Computational Topology (II): Persistence Theory
- 6. Topological Machine Learning (I): An Introduction
- 7. Topological Machine Learning (II): Advanced Topics
- 8. Dimensionality Reduction Algorithms

Website: `http://www-sop.inria.fr/abs/teaching/centrale-FGMDA/centrale-FGMDA--cazals-carriere.html`

Validation is done by group projects, subjects will be available later.

Information about the class

Class outline (each class is 50% lecture 50% practical session)

- 1. Computational Topology (I): Simplicial Complexes
- 2. Nearest Neighbors in Euclidean and metric spaces (I): Data Structures and Algorithms
- 3. Nearest Neighbors in Euclidean and metric spaces (II): Analysis
- 4. Comparing Samplings, Distributions, Clusterings
- 5. Computational Topology (II): Persistence Theory
- 6. Topological Machine Learning (I): An Introduction
- 7. Topological Machine Learning (II): Advanced Topics
- 8. Dimensionality Reduction Algorithms

Website: `http://www-sop.inria.fr/abs/teaching/centrale-FGMDA/centrale-FGMDA--cazals-carriere.html`

Validation is done by group projects, subjects will be available later.

Practical sessions are based on:

गुढी GUDHI

Structural Bioinformatics Library

Information about the class

My classes are about

Topological Data Analysis (TDA)

Information about the class

My classes are about

Topological Data Analysis (TDA)

Goal: Study geometric data sets with techniques coming from *topology*.

Information about the class

My classes are about

Topological Data Analysis (TDA)

Goal: Study geometric data sets with techniques coming from *topology*.

Question: What is topology?

Information about the class

My classes are about

Topological Data Analysis (TDA)

Goal: Study geometric data sets with techniques coming from *topology*.

Question: What is topology?

A: Roughly speaking, the topology of a space X is its number of 'holes'. More formally, it is the class of spaces that can be obtained by continuous deformations of X.

Information about the class

My classes are about

Topological Data Analysis (TDA)

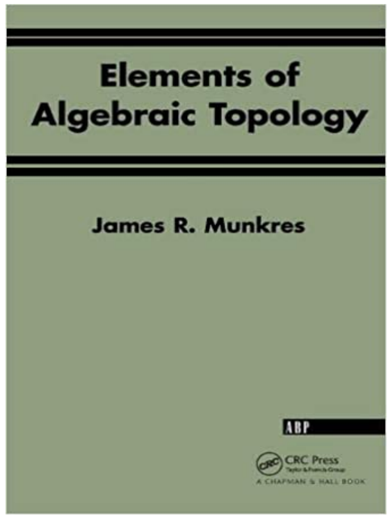
Goal: Study geometric data sets with techniques coming from *topology*.

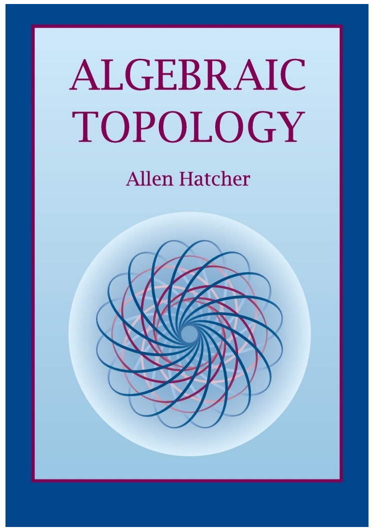
Question: What is topology?

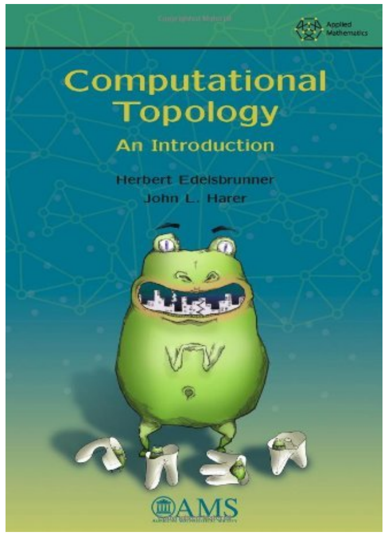
[*Elements of Algebraic Topology*, Munkres, CRC Press, 1984]

[*Algebraic Topology*, Hatcher, Cambridge University Press, 2002]

[*Computational Topology: an introduction*, Edelsbrunner, Harer, AMS, 2010]







Introduction

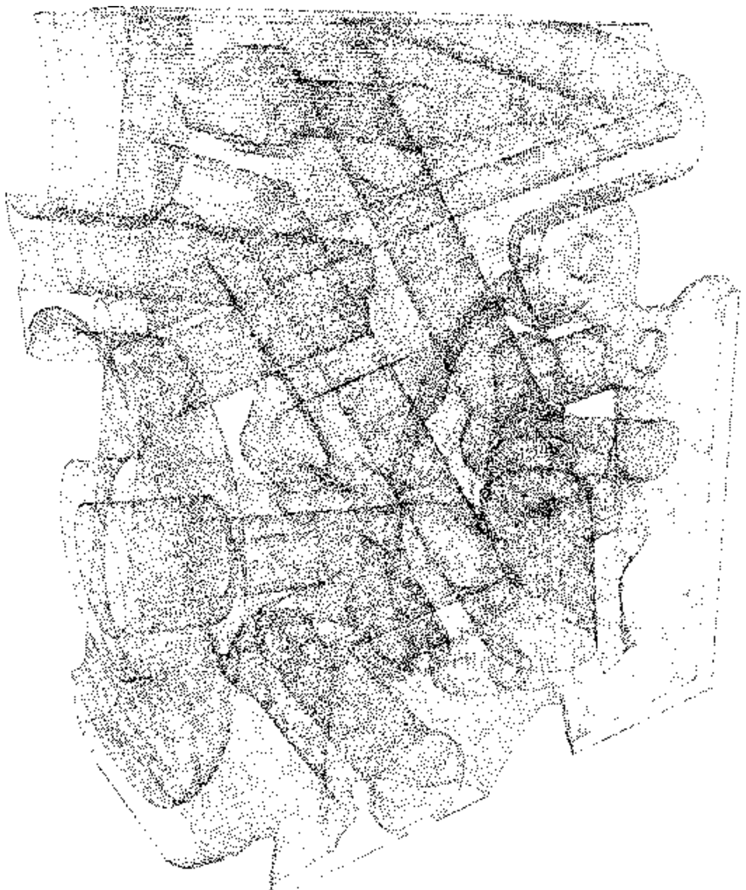
We will see how to build new topological features from data sets...

Introduction

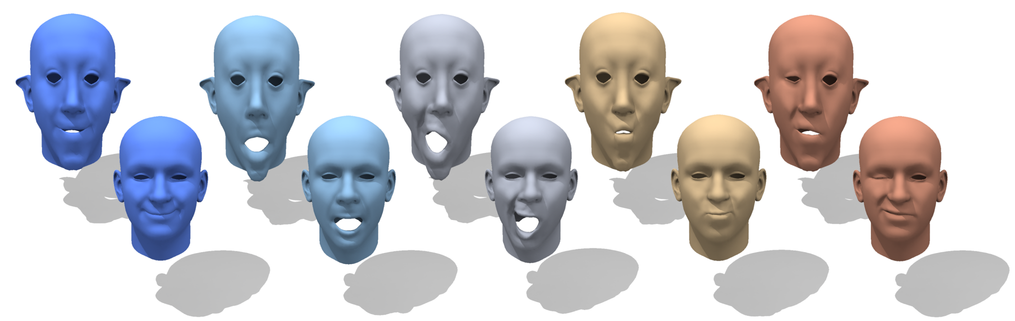
We will see how to build new topological features from data sets...

...but why is that interesting?

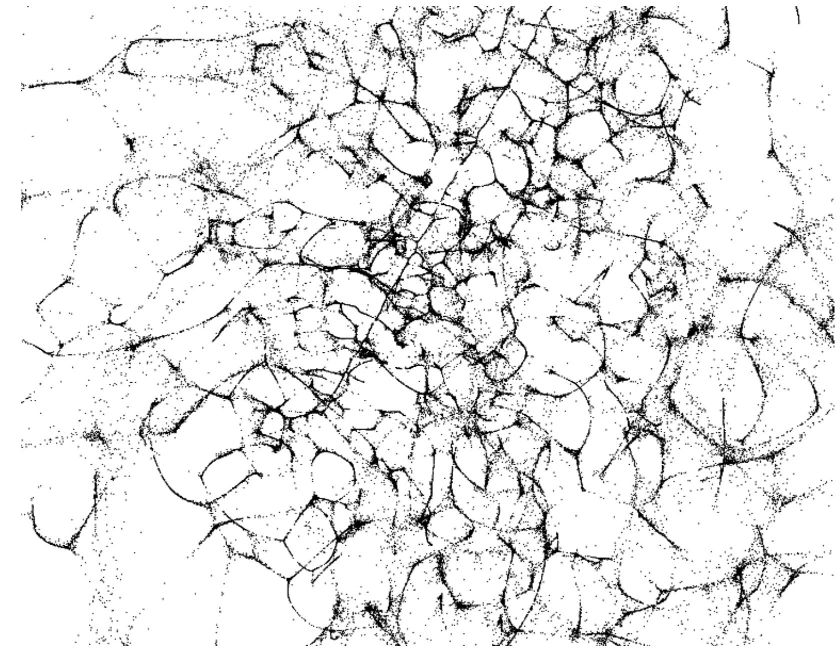
Introduction



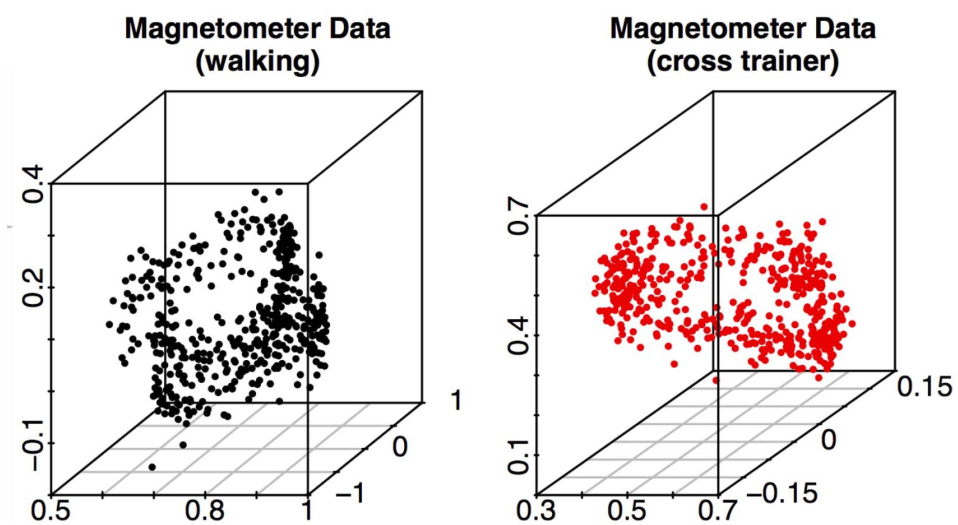
Scans



3D shapes



Galaxies

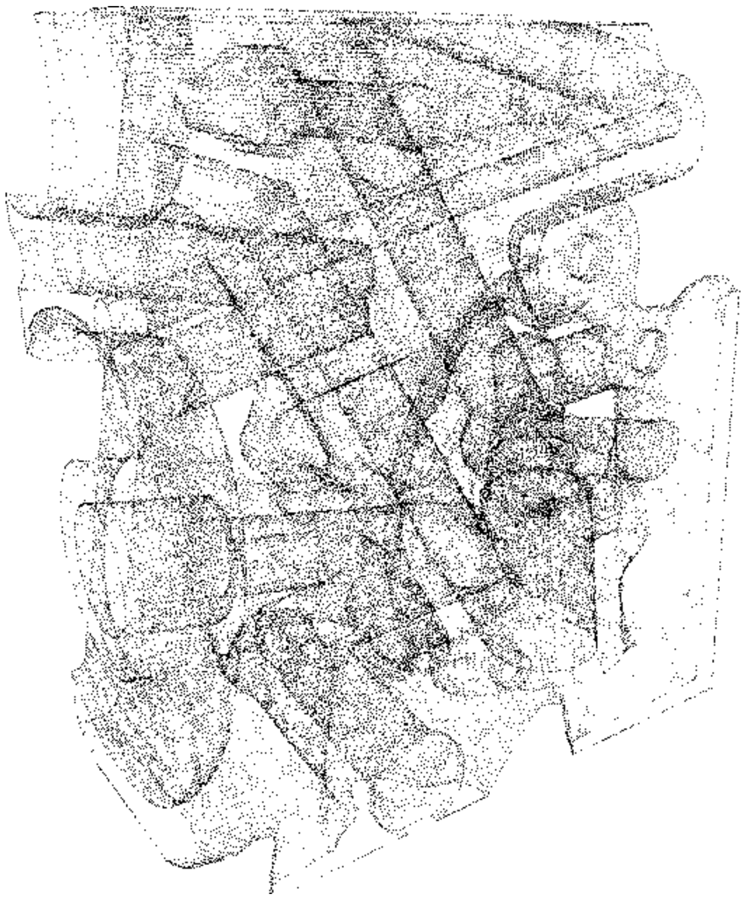


Magnetometer

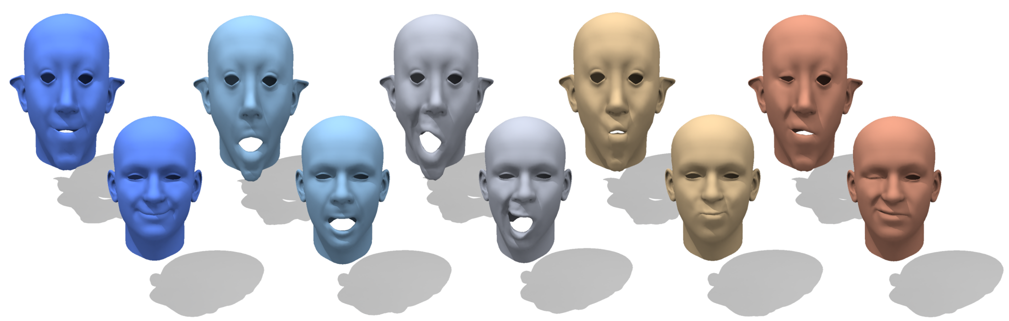
Data often come as (sampling of) metric spaces or sets/spaces endowed with a similarity measure with, possibly complex, topological/geometric structure.

Data carrying geometric information is usually high dimensional.

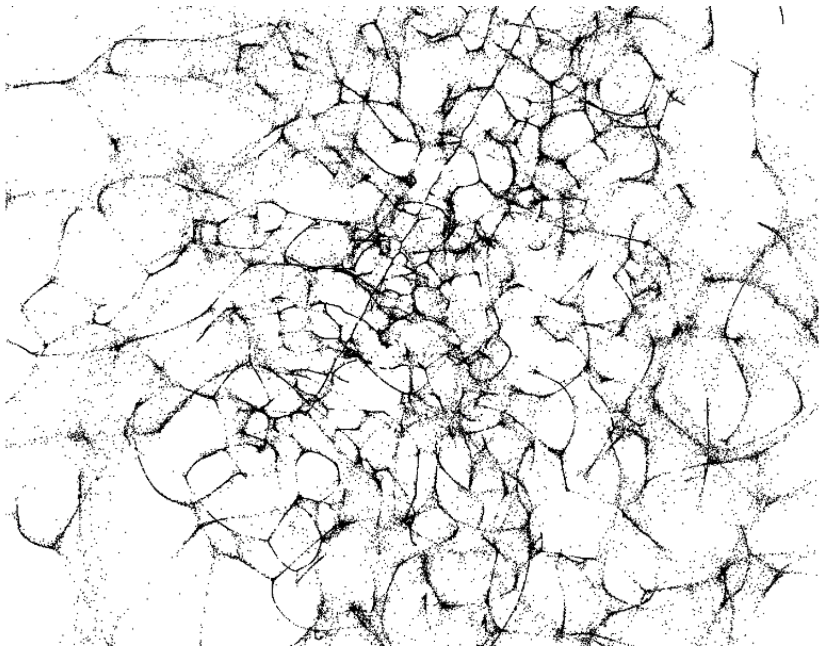
Introduction



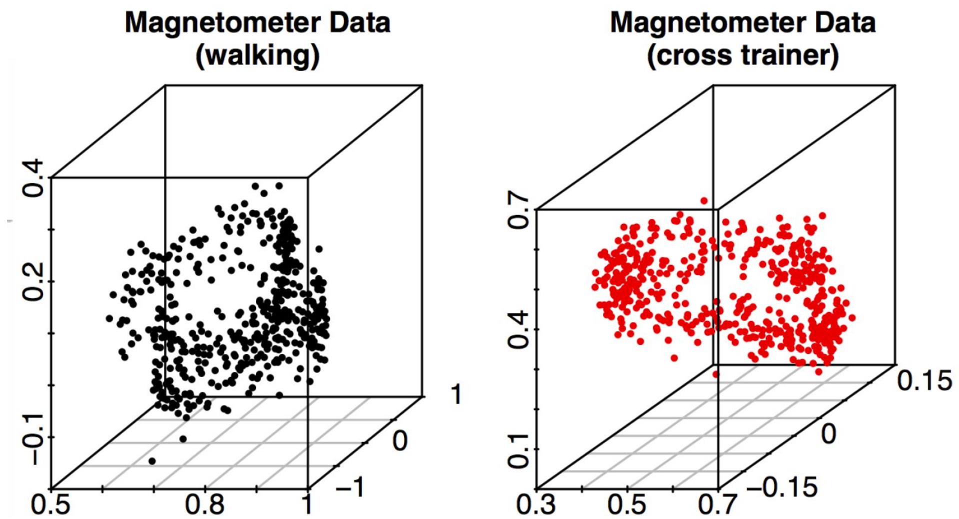
Scans



3D shapes



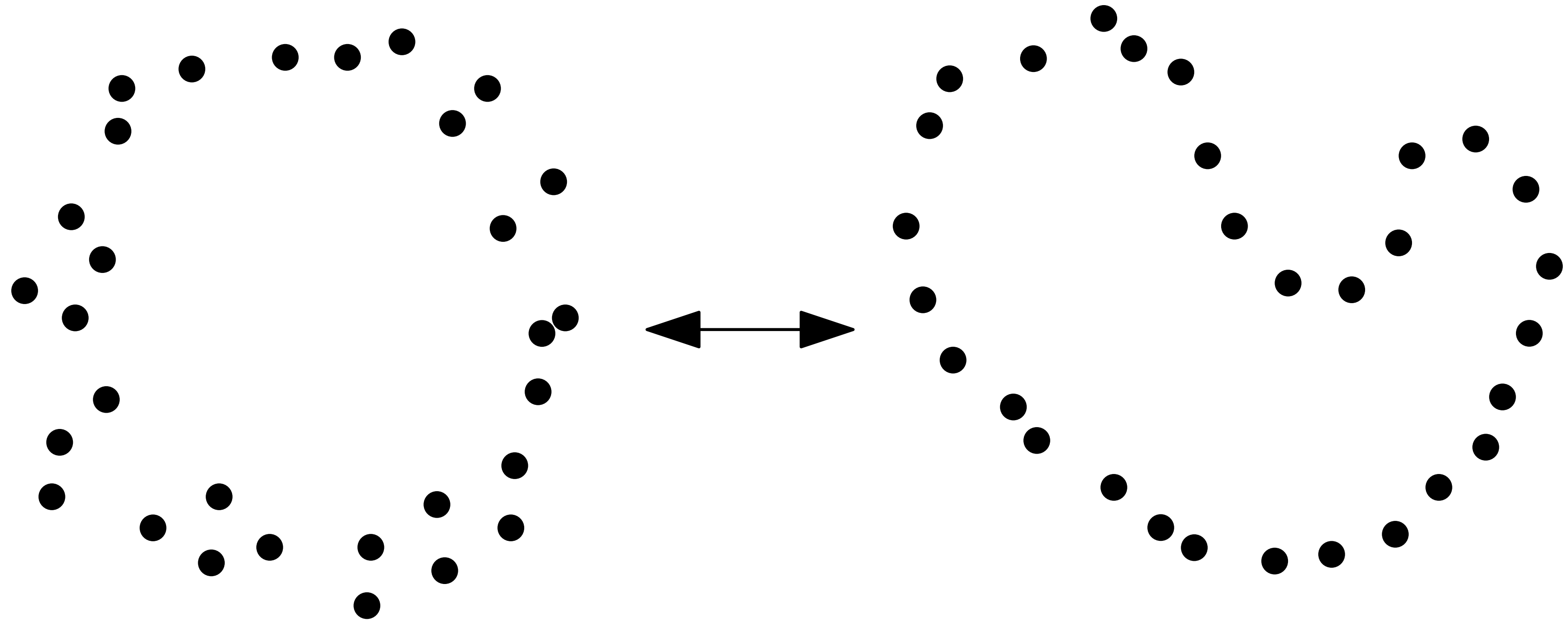
Galaxies



Magnetometer

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...).

Introduction



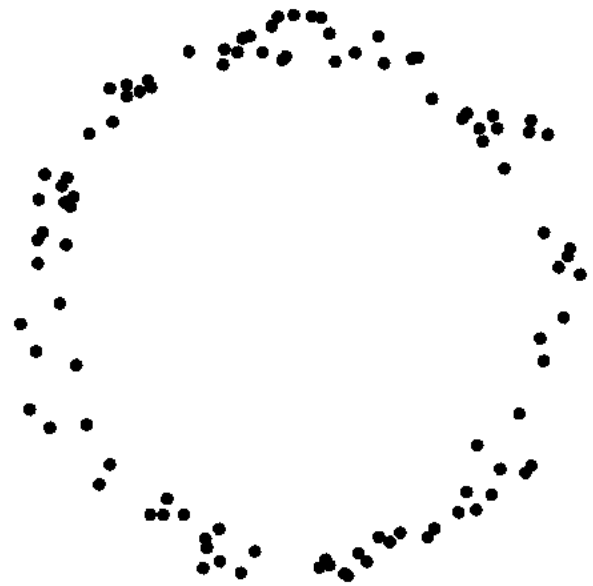
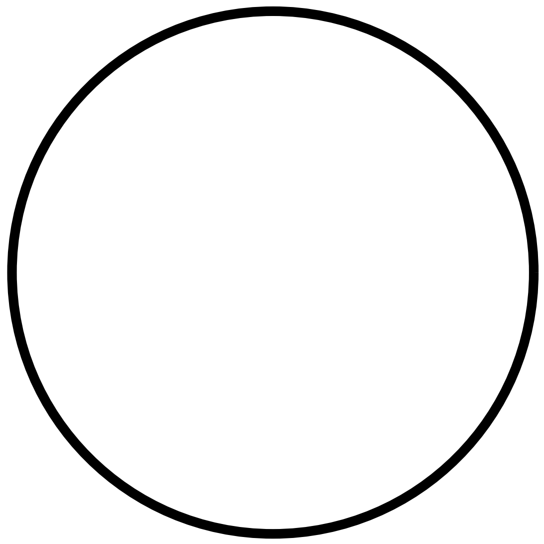
Pros of topology:

• **Coordinate invariance:** topological features/invariants do not rely on any coordinate system so 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.

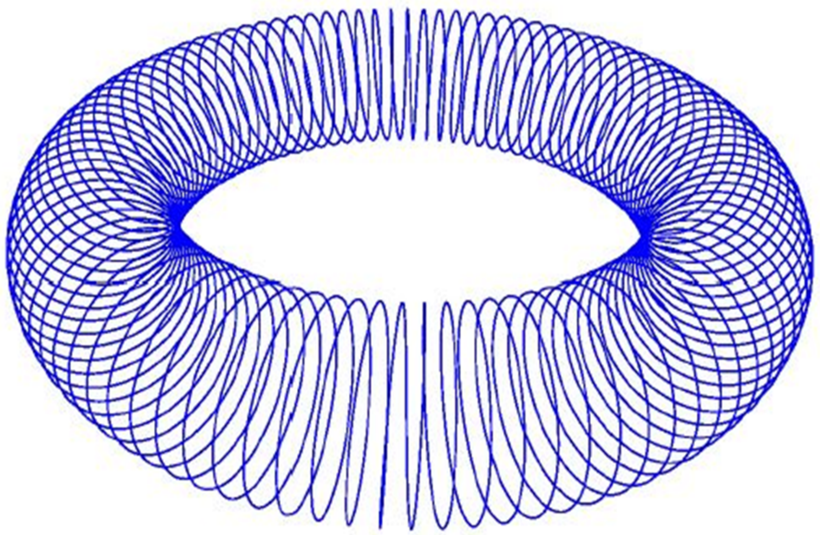
Introduction

Problem: how to define the *topology* of a data set?

Cons of topology:

• 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.

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Pbm: How to encode topological spaces for computational purposes?

Computational Topology (I): Simplicial Complexes and Homology

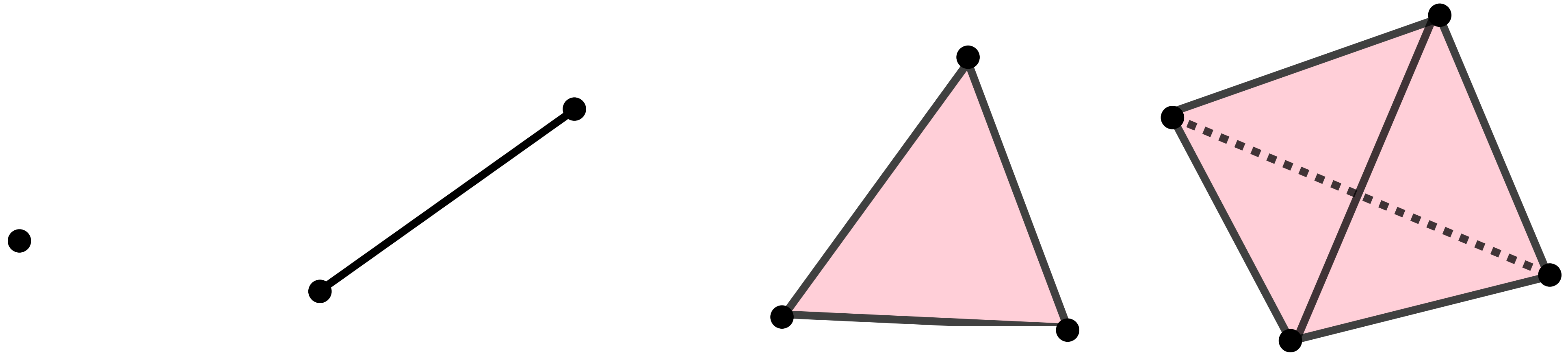
1. **Simplicial Complexes**
2. **Nerve Theorem**
3. **Homology Groups**

Pbm: How to encode topological spaces for computational purposes?

A: Using spaces made of small convex bricks, namely the *simplicial complexes* made of *simplices*.

Simplex and simplicial complex

Simplex and simplicial complex



0-simplex:
vertex

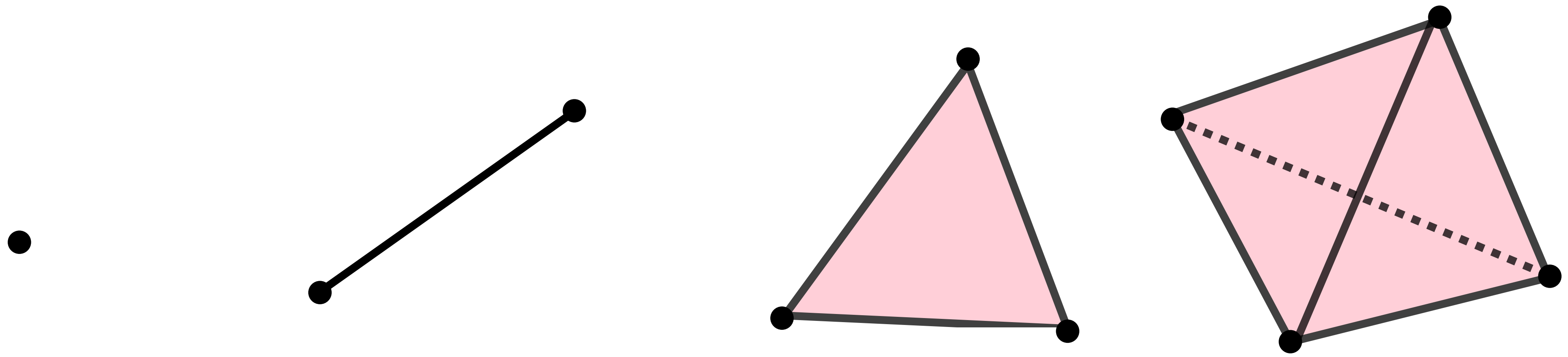
1-simplex:
edge

2-simplex:
triangle

3-simplex:
tetrahedron

etc...

Simplex and simplicial complex



0-simplex: vertex | 1-simplex: edge | 2-simplex: triangle | 3-simplex: tetrahedron | etc...

Def: Given a set $P = \{p_0, \ldots, p_k\} \subset \mathbb{R}^d$ of $k+1$ affinely independent points, the k-dimensional simplex σ (or k-simplex for short) spanned by P is the set of convex combinations

$$\sum_{i=0}^{k} \lambda_i \, p_i, \quad \text{with} \quad \sum_{i=0}^{k} \lambda_i = 1 \quad \text{and} \quad \lambda_i \geq 0.$$

The points $p_0, \ldots, p_k$ are called the vertices of σ.

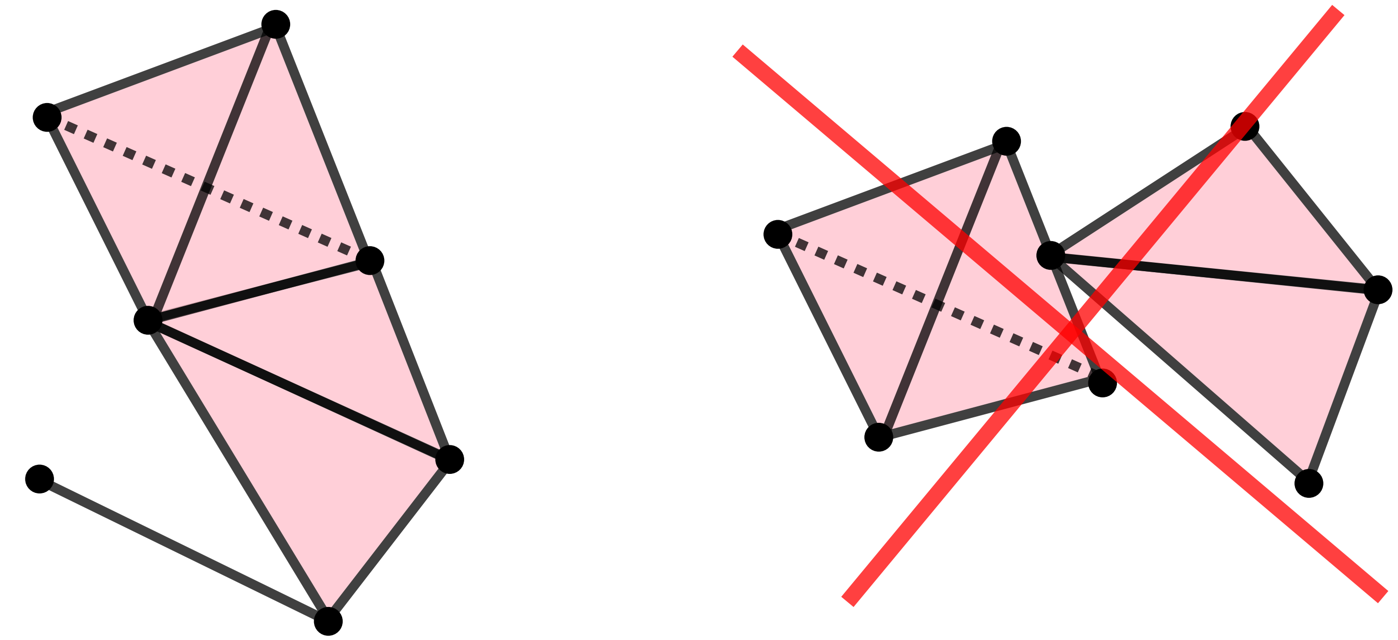
Simplex and simplicial complex

Def: A simplicial complex K in $\mathbb{R}^d$ is a collection of simplices s.t.:

- *(i)* any face of a simplex of K is a simplex of K,
- *(ii)* the intersection of any two simplices of K is either empty or a common face of both.

The underlying space of K, written $|K| \subseteq \mathbb{R}^d$, is the union of its simplices. The k-skeleton of K, written $\mathrm{Skel}_k(K)$, is the smaller complex made of the simplices of K of dimension up to k: $\mathrm{Skel}_k(K) = \{\sigma \in K : \dim(\sigma) \leq k\}$.

Simplex and simplicial complex



Def: A simplicial complex K in $\mathbb{R}^d$ is a collection of simplices s.t.:

- *(i)* any face of a simplex of K is a simplex of K,
- *(ii)* the intersection of any two simplices of K is either empty or a common face of both.

The underlying space of K, written $|K| \subseteq \mathbb{R}^d$, is the union of its simplices. The k-skeleton of K, written $\mathrm{Skel}_k(K)$, is the smaller complex made of the simplices of K of dimension up to k: $\mathrm{Skel}_k(K) = \{\sigma \in K : \dim(\sigma) \leq k\}$.

Simplex and simplicial complex

Remark: Simplicial complexes can be seen at the same time as geometric/topological spaces (good for geometrical/topological inference) and as combinatorial objects (good for computations).

Def: A simplicial complex K in $\mathbb{R}^d$ is a collection of simplices s.t.:

- *(i)* any face of a simplex of K is a simplex of K,
- *(ii)* the intersection of any two simplices of K is either empty or a common face of both.

The underlying space of K, written $|K| \subseteq \mathbb{R}^d$, is the union of its simplices.
The k-skeleton of K, written $\mathrm{Skel}_k(K)$, is the smaller complex made of the simplices of K of dimension up to k: $\mathrm{Skel}_k(K) = \{\sigma \in K : \dim(\sigma) \leq k\}$.

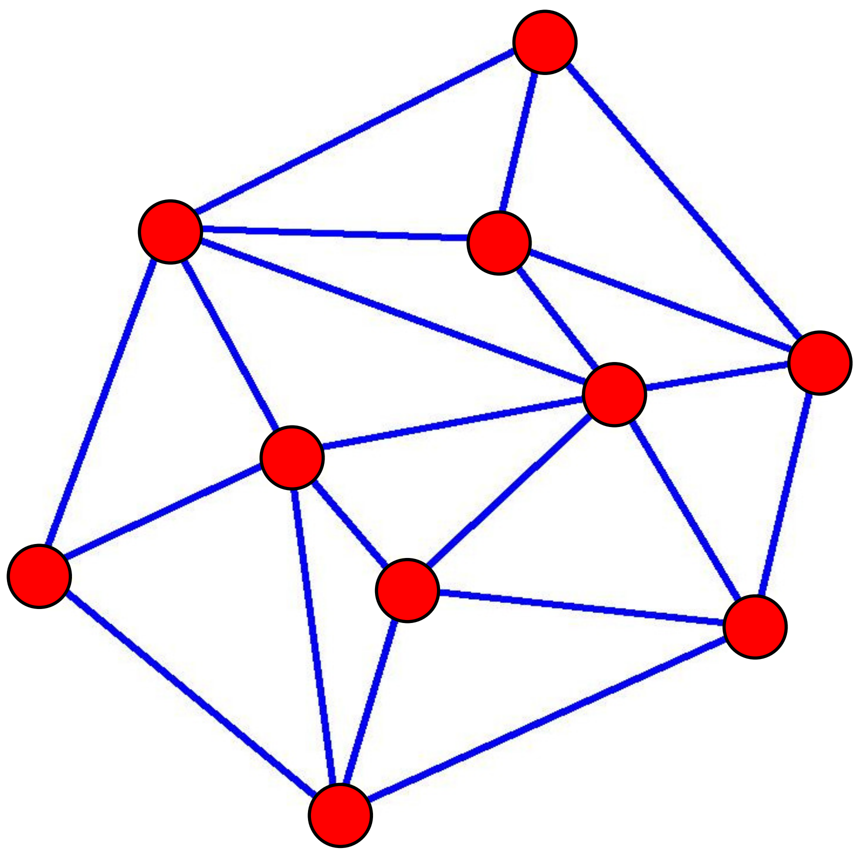
Triangulations

Def: A simplicial complex of dimension d is pure if every simplex is the face of some d-dimensional simplex.

Triangulations

Def: A simplicial complex of dimension d is pure if every simplex is the face of some d-dimensional simplex.

Def: A triangulation of a point cloud $P \subseteq \mathbb{R}^d$ is a pure simplicial complex K s.t. $\mathrm{vert}(K) = P$ and $|K| = \mathrm{conv}(P)$.

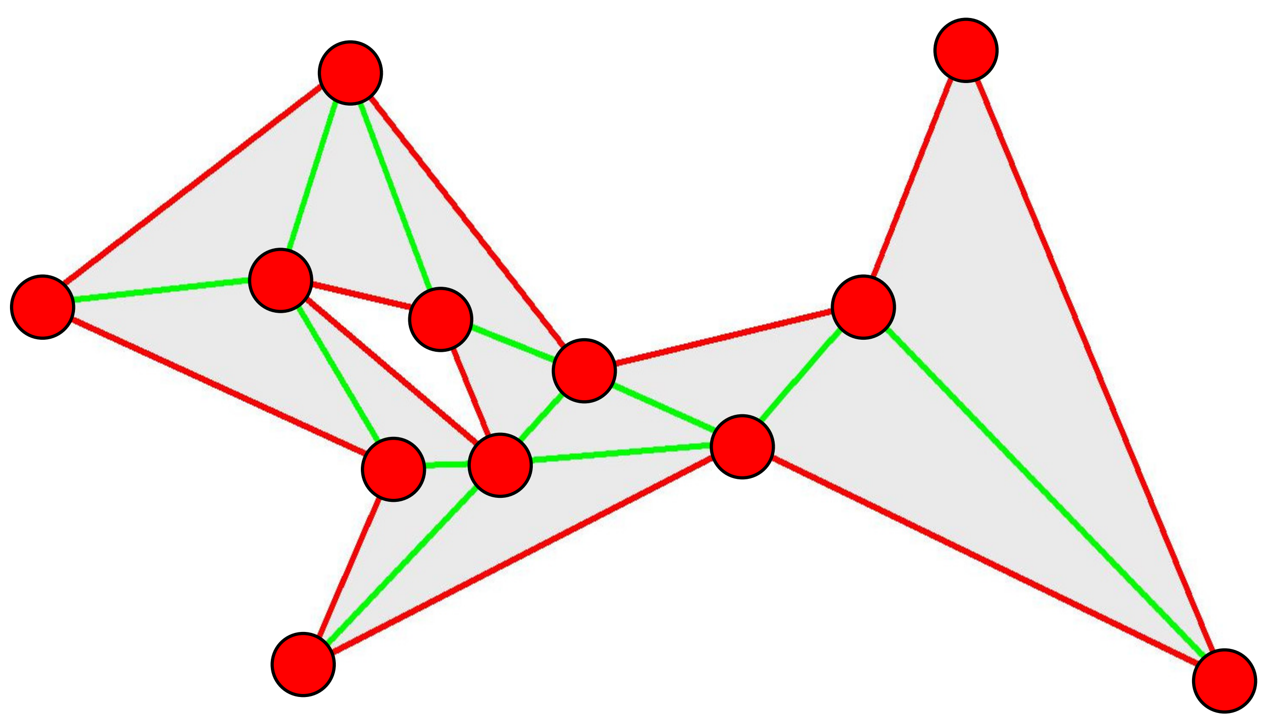


Triangulations

Def: A simplicial complex of dimension d is pure if every simplex is the face of some d-dimensional simplex.

Def: A triangulation of a point cloud $P \subseteq \mathbb{R}^d$ is a pure simplicial complex K s.t. $\mathrm{vert}(K) = P$ and $|K| = \mathrm{conv}(P)$.

Def: A triangulation of a polygonal domain $\Omega \subseteq \mathbb{R}^d$ with vertex set P is a pure simplicial complex K s.t. $\mathrm{vert}(K) = P$ and $|K| = \Omega$.



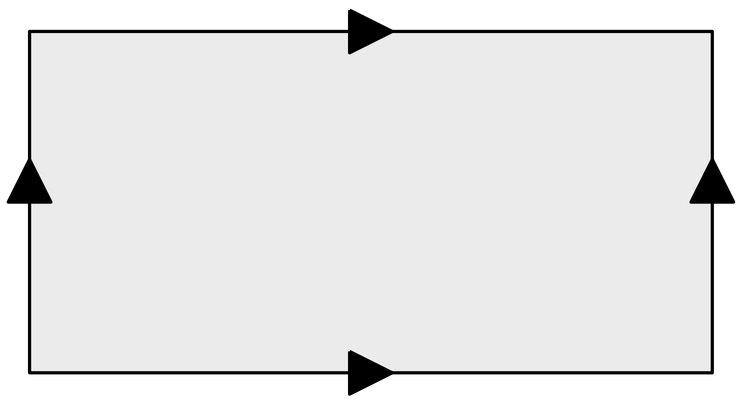
Triangulations

Def: A simplicial complex of dimension d is pure if every simplex is the face of some d-dimensional simplex.

Def: A triangulation of a point cloud $P \subseteq \mathbb{R}^d$ is a pure simplicial complex K s.t. $\mathrm{vert}(K) = P$ and $|K| = \mathrm{conv}(P)$.

Def: A triangulation of a polygonal domain $\Omega \subseteq \mathbb{R}^d$ with vertex set P is a pure simplicial complex K s.t. $\mathrm{vert}(K) = P$ and $|K| = \Omega$.

Q: Triangulate

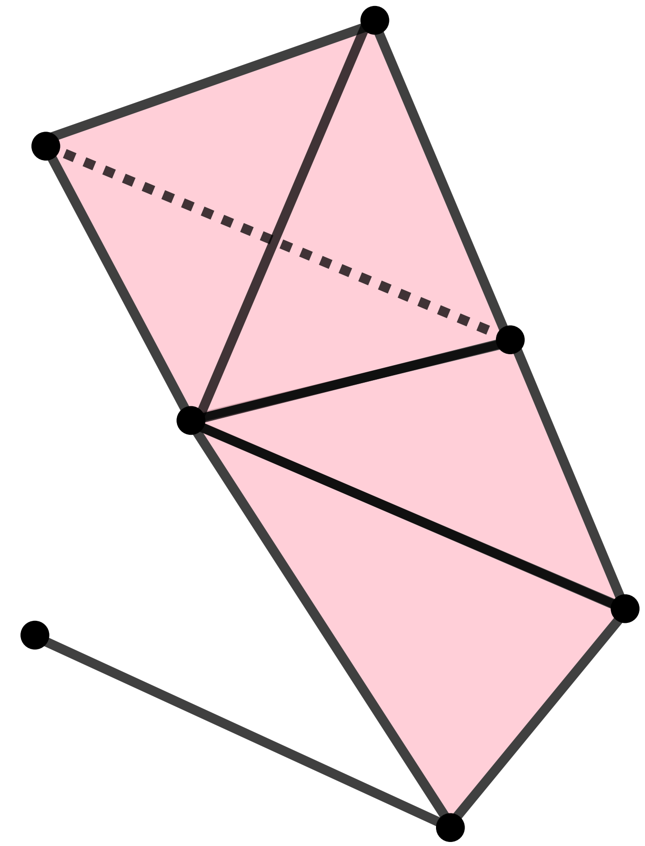


Abstract simplex and simplicial complex

Def: Let $P = \{p_1, \cdots, p_n\}$ be a (finite) set of vertices (*not necessarily embedded in* $\mathbb{R}^d$). An abstract simplicial complex K with vertex set P is a set of subsets of P satisfying the two conditions:

- *(i)* the elements of P belong to K,
- *(ii)* if $\tau \in K$ and $\sigma \subseteq \tau$, then $\sigma \in K$.

The elements of K are the simplices.

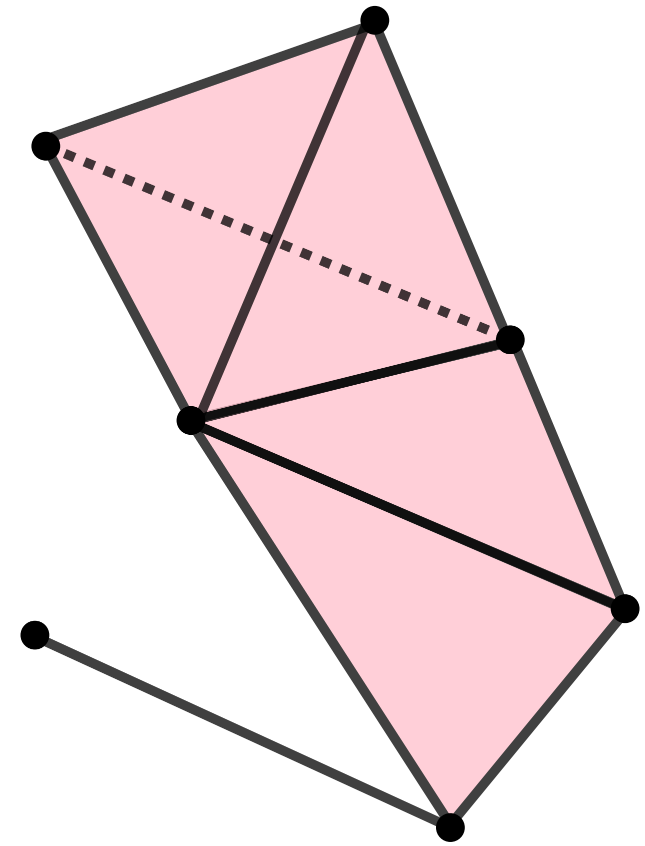


Abstract simplex and simplicial complex

Def: Let $P = \{p_1, \cdots, p_n\}$ be a (finite) set of vertices (*not necessarily embedded in* $\mathbb{R}^d$). An abstract simplicial complex K with vertex set P is a set of subsets of P satisfying the two conditions:

- *(i)* the elements of P belong to K,
- *(ii)* if $\tau \in K$ and $\sigma \subseteq \tau$, then $\sigma \in K$.

The elements of K are the simplices.



Remark: It is possible to define abstract simplicial complexes out of point clouds embedded in $\mathbb{R}^d$—in this case, the dimension of the complex is not necessarily d, see for instance Rips complexes later.

Abstract simplex and simplicial complex

Def: A realization of an abstract simplicial complex K is a geometric simplicial complex K' who is isomorphic to K, i.e., there exists a bijection

$$f : \mathrm{vert}(K) \to \mathrm{vert}(K'),$$

such that $\sigma \in K \iff f(\sigma) \in K'$.

Abstract simplex and simplicial complex

Def: A realization of an abstract simplicial complex K is a geometric simplicial complex K' who is isomorphic to K, i.e., there exists a bijection

$$f : \mathrm{vert}(K) \to \mathrm{vert}(K'),$$

such that $\sigma \in K \iff f(\sigma) \in K'$.

Q: Prove that any simplicial complex with n vertices can be realized in $\mathbb{R}^n$.

Čech and (Vietoris)-Rips complexes

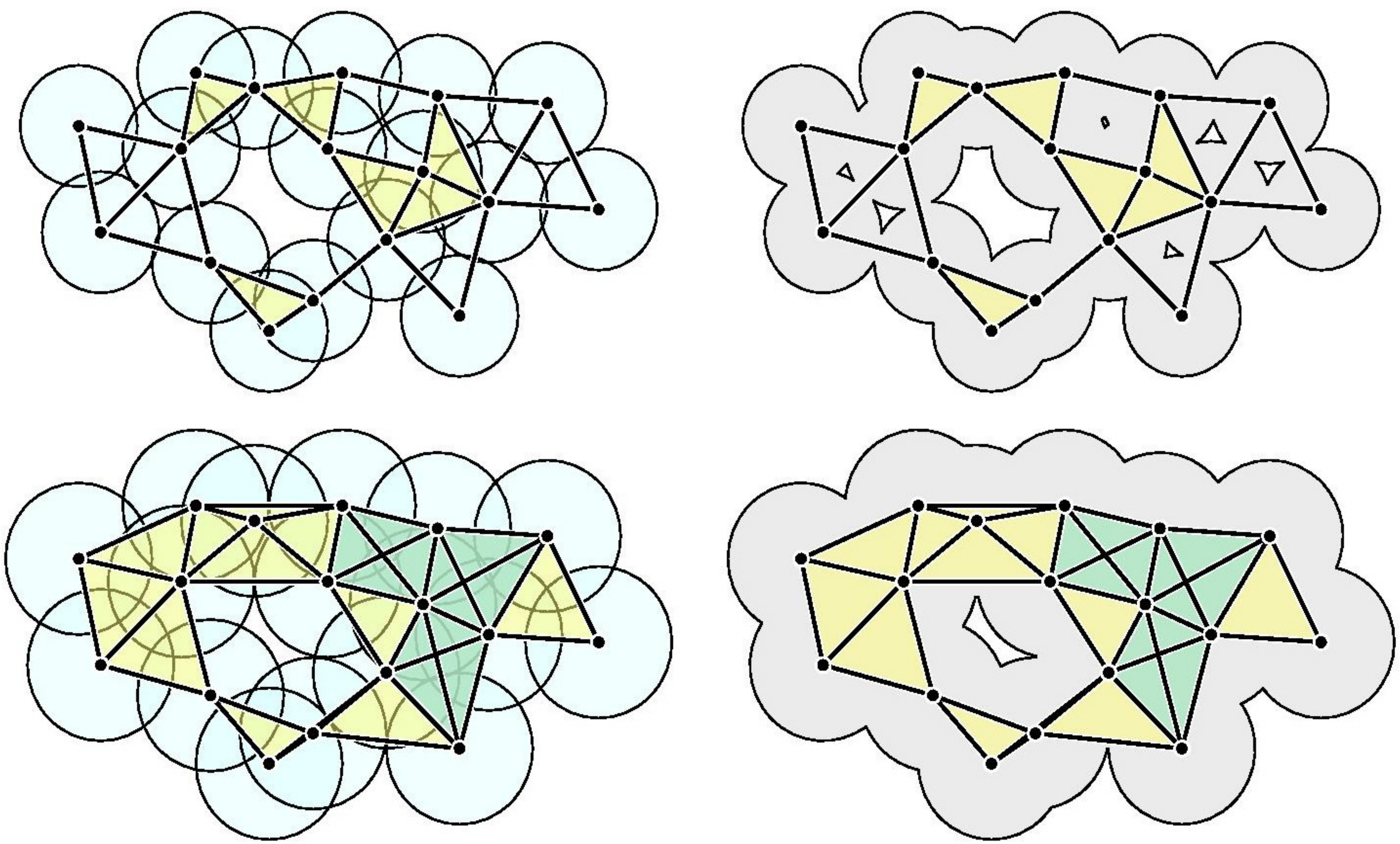
Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P,r)$ s.t. $\mathrm{vert}(C(P,r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P,r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$



Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

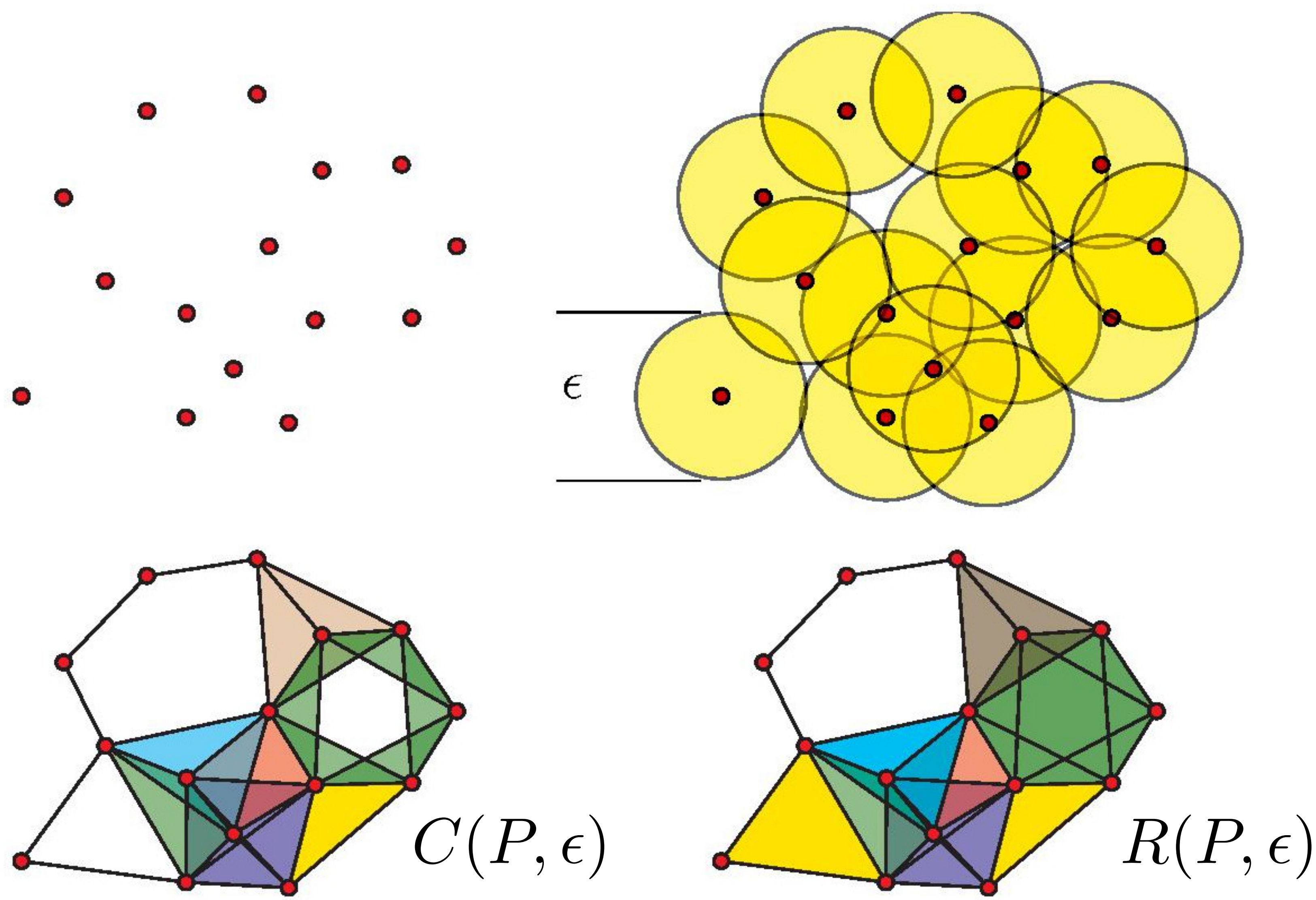
$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Remark: The 1-skeleton $\mathrm{Skel}_1(R(P, r))$ of a Rips complex of radius r is also called the r-*neighborhood graph* of P.

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$



Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Good news is that Rips and Čech complexes are related:

Čech and (Vietoris)-Rips complexes

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Čech complex of radius $r > 0$ is the abstract simplicial complex $C(P, r)$ s.t. $\mathrm{vert}(C(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in C(P, r) \quad \text{iif} \quad \cap_{j=0}^{k} B(P_{i_j}, r) \neq \emptyset.$$

Pbm: Čech complexes can be quite hard to compute.

Def: Given a point cloud $P = \{P_1, \ldots, P_n\} \subset \mathbb{R}^d$, its Rips complex of radius $r > 0$ is the abstract simplicial complex $R(P, r)$ s.t. $\mathrm{vert}(R(P, r)) = P$ and

$$\sigma = [P_{i_0}, P_{i_1}, \ldots, P_{i_k}] \in R(P, r) \quad \text{iif} \quad \|P_{i_j} - P_{i_{j'}}\| \leq 2r, \forall 1 \leq j, j' \leq k.$$

Good news is that Rips and Čech complexes are related:

Prop: $R(P, r/2) \subseteq C(P, r) \subseteq R(P, r)$.

Q: Prove it.

Storing simplicial complexes

[*The Simplex Tree: An Efficient Data Structure for General Simplicial Complexes*, Boissonnat, Maria, Algorithmica, 2014]

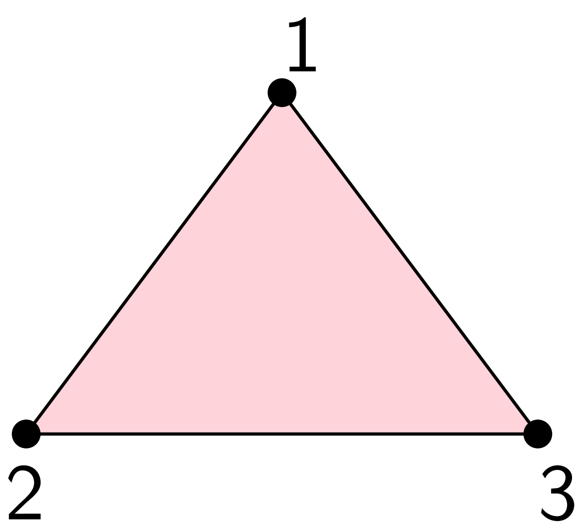
We want to store simplicial complexes with a data structure that allows to perform standard operations (insertion of a simplex, checking if a simplex is present, etc) in a fast and easy way.

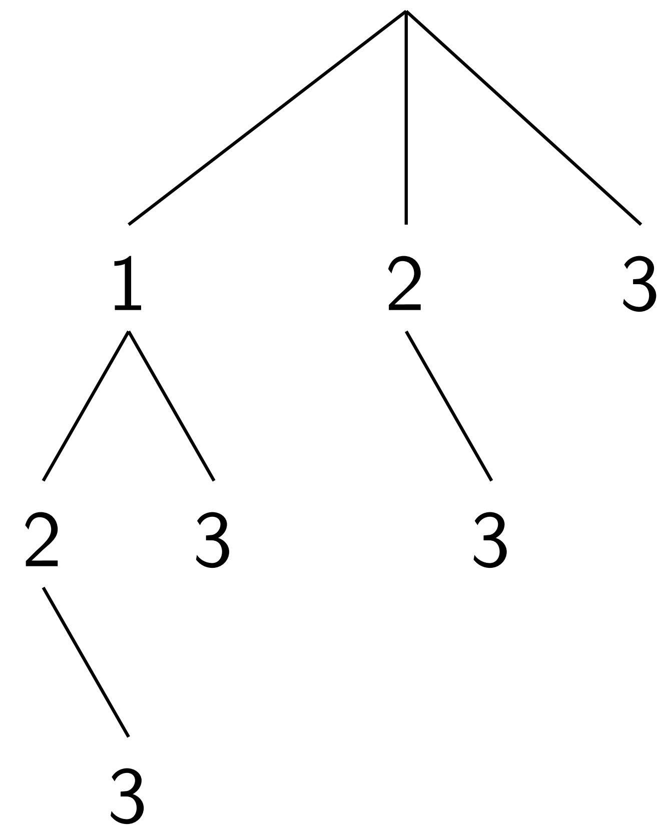
Storing simplicial complexes

[*The Simplex Tree: An Efficient Data Structure for General Simplicial Complexes*, Boissonnat, Maria, Algorithmica, 2014]

We want to store simplicial complexes with a data structure that allows to perform standard operations (insertion of a simplex, checking if a simplex is present, etc) in a fast and easy way.

Idea: store sorted simplices in a prefix tree (also called *trie)*.



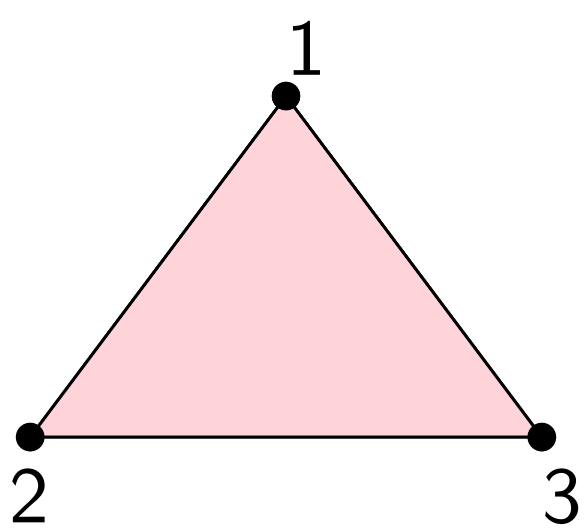


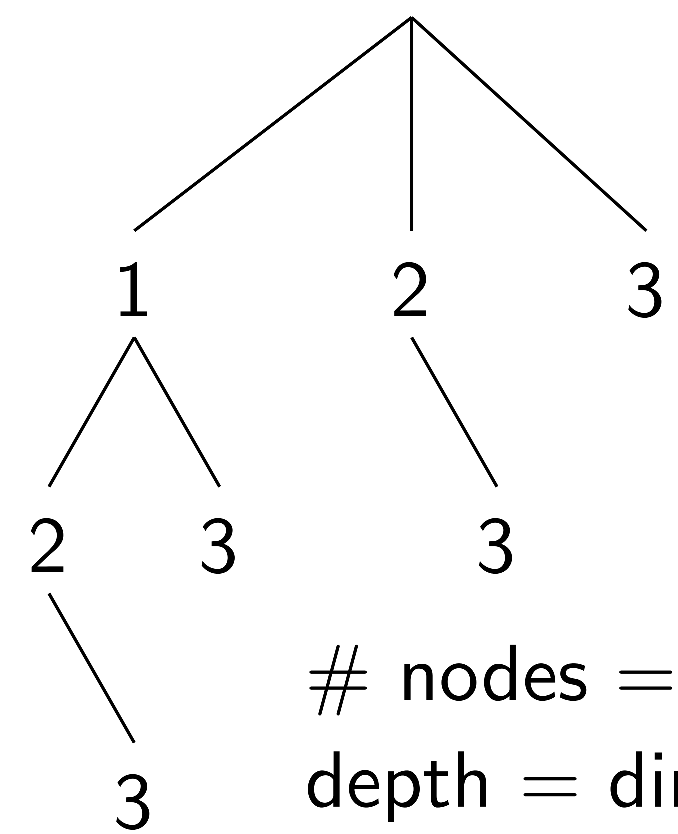
Storing simplicial complexes

[*The Simplex Tree: An Efficient Data Structure for General Simplicial Complexes*, Boissonnat, Maria, Algorithmica, 2014]

We want to store simplicial complexes with a data structure that allows to perform standard operations (insertion of a simplex, checking if a simplex is present, etc) in a fast and easy way.

Idea: store sorted simplices in a prefix tree (also called *trie)*.





$\#$ nodes $= \#$ simplices

depth $=$ dimension $+ 1$

This is called the *simplex tree*.

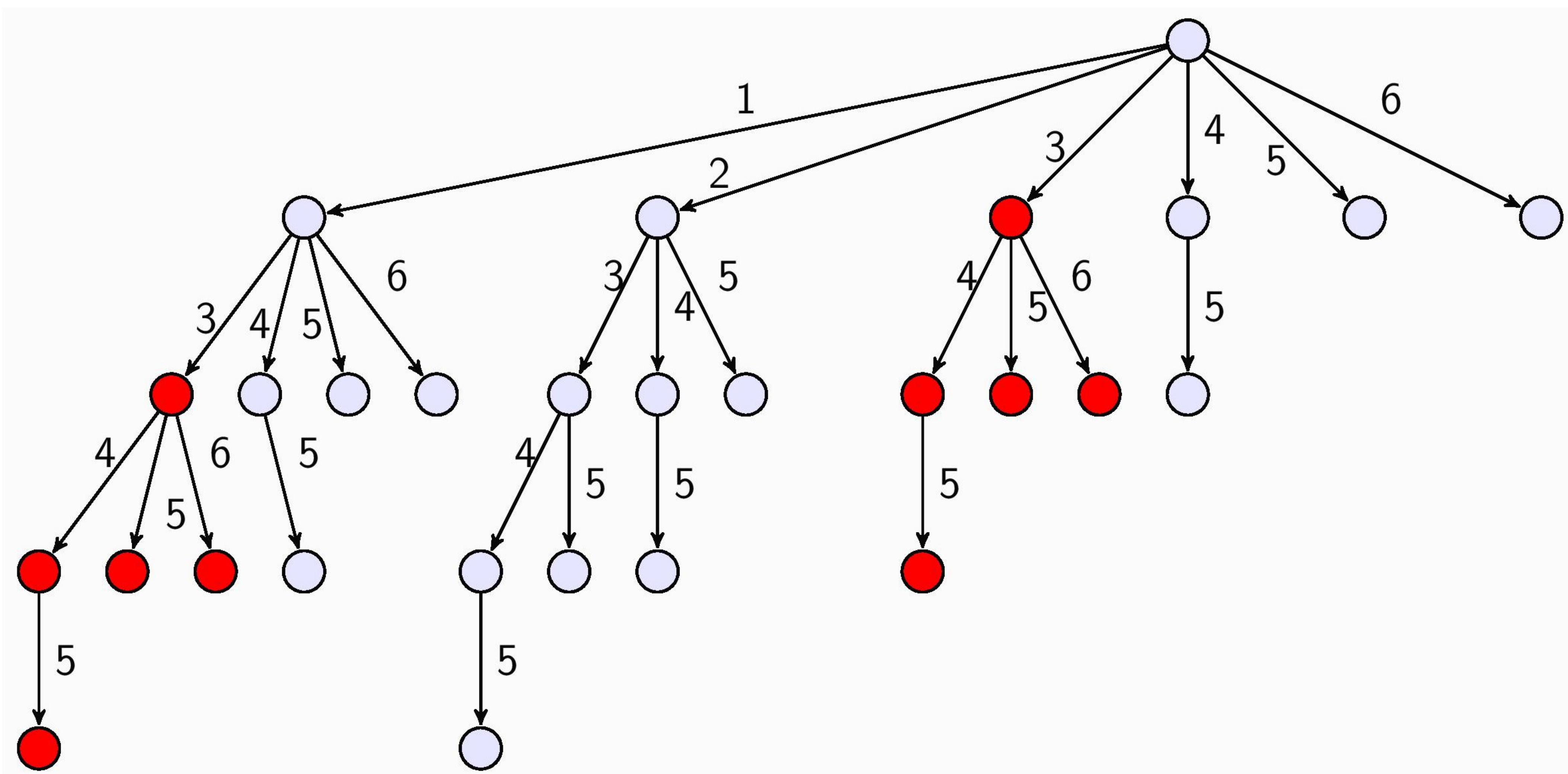
It allows to store all simplices explicitly without storing all adjacency relations, while maintaining low complexity for basic operations.

Storing simplicial complexes

[*The Simplex Tree: An Efficient Data Structure for General Simplicial Complexes*, Boissonnat, Maria, Algorithmica, 2014]

We want to store simplicial complexes with a data structure that allows to perform standard operations (insertion of a simplex, checking if a simplex is present, etc) in a fast and easy way.

Unfortunately, the simplex tree also has redundancies.

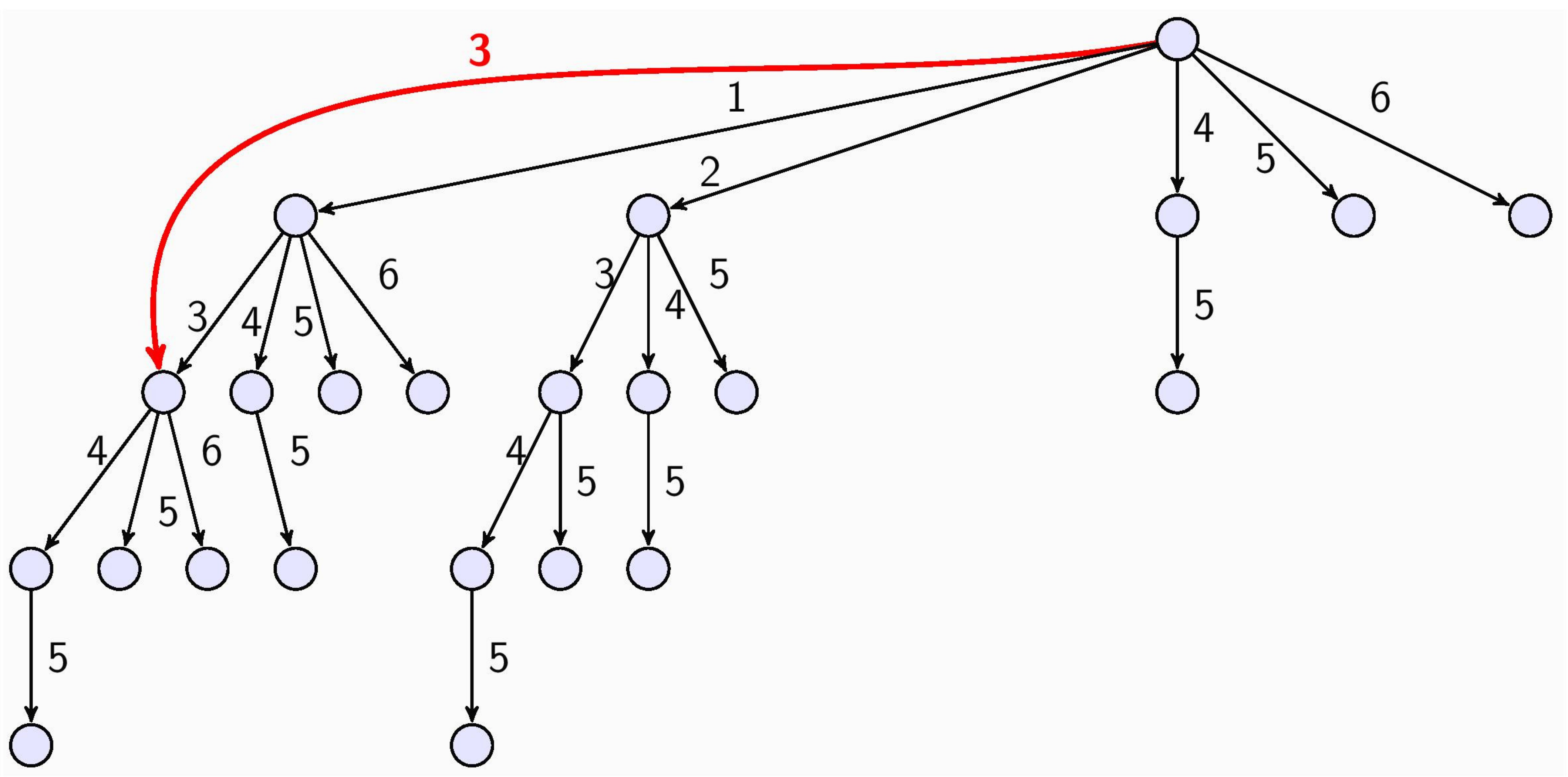


Storing simplicial complexes

[*The Simplex Tree: An Efficient Data Structure for General Simplicial Complexes*, Boissonnat, Maria, Algorithmica, 2014]

We want to store simplicial complexes with a data structure that allows to perform standard operations (insertion of a simplex, checking if a simplex is present, etc) in a fast and easy way.

Unfortunately, the simplex tree also has redundancies.



Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Pbm: How to ensure simplicial complexes are "good" models of topological spaces?

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. **Nerve Theorem**
3. Homology Groups

Pbm: How to ensure simplicial complexes are "good" models of topological spaces?

A: The *Nerve Theorem* ensures that appropriate complexes have the right topology.

Introduction

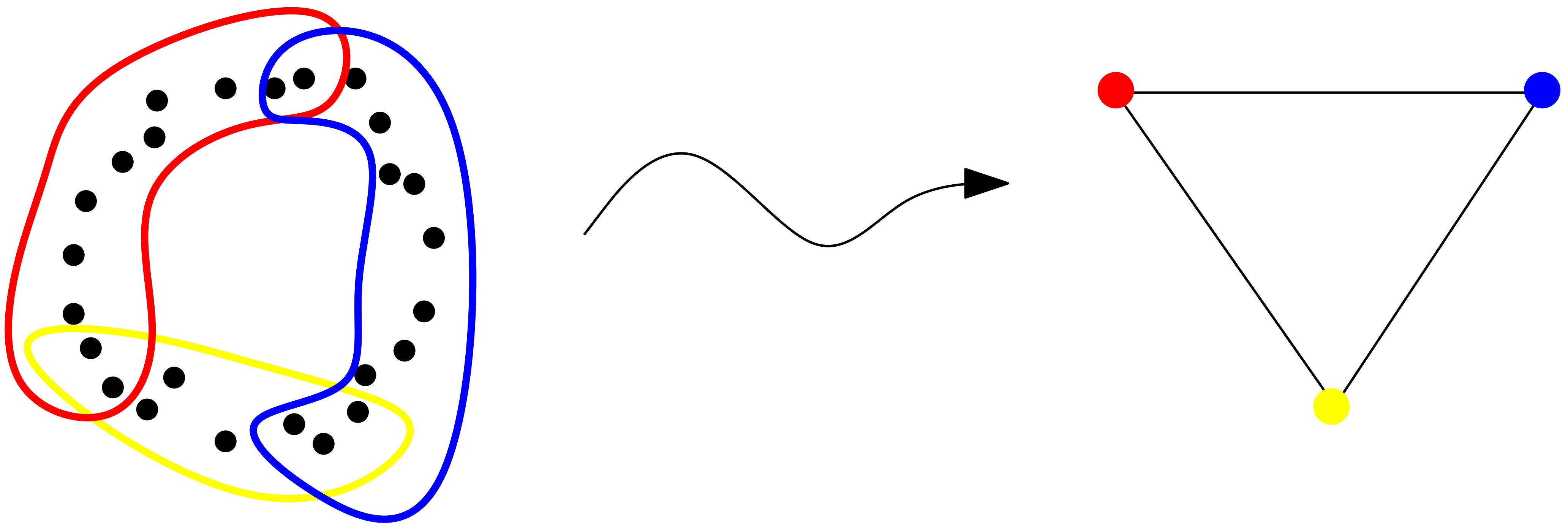
Topology is the art of deformation. It was introduced by Poincaré as a way to classify topological spaces: 'two topological spaces are in the same class if one can deform it into the other'.

The Nerve Theorem provides conditions under which a simplicial complex can be deformed into the topological space it was computed from.

Introduction

Idea: work with cover complexes.

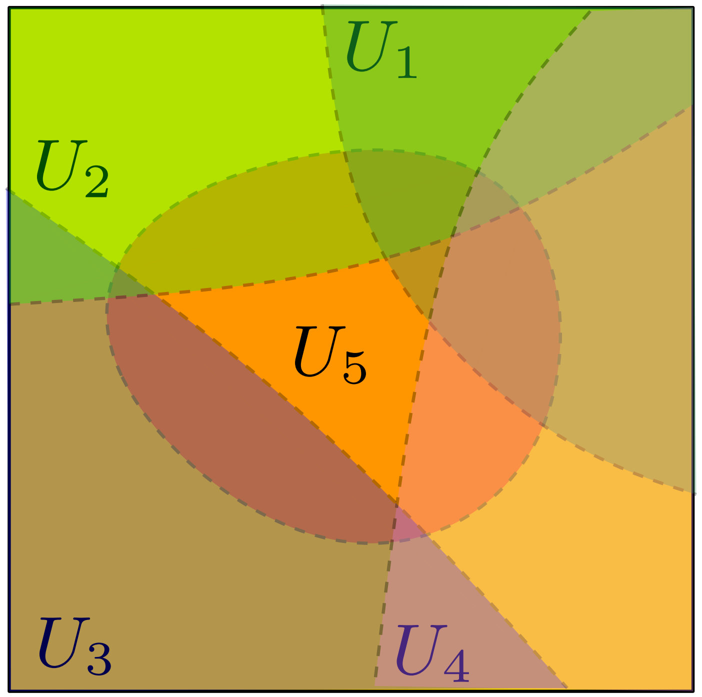
- Group data points in local clusters.
- Summarize the data through the combinatorial/topological structure of the intersection patterns of these local clusters.



Nerve complex

Def: An open cover of a topological space X is a collection $\mathcal{U} = (U_i)_{i \in I}$ of open subsets $U_i \subseteq X$, $i \in I$ where I is a set, such that $X \subseteq \cup_{i \in I} U_i$.

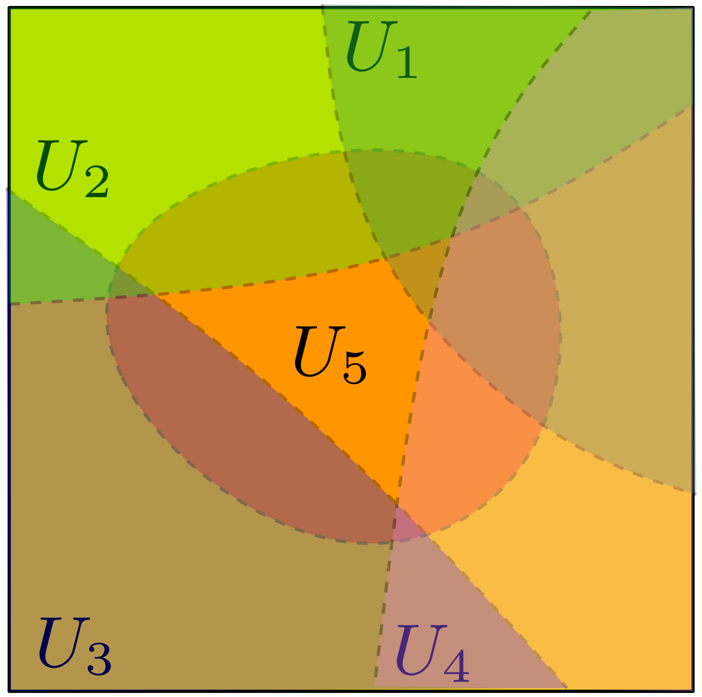
Nerve complex



X

Def: An open cover of a topological space X is a collection $\mathcal{U} = (U_i)_{i \in I}$ of open subsets $U_i \subseteq X$, $i \in I$ where I is a set, such that $X \subseteq \cup_{i \in I} U_i$.

Nerve complex



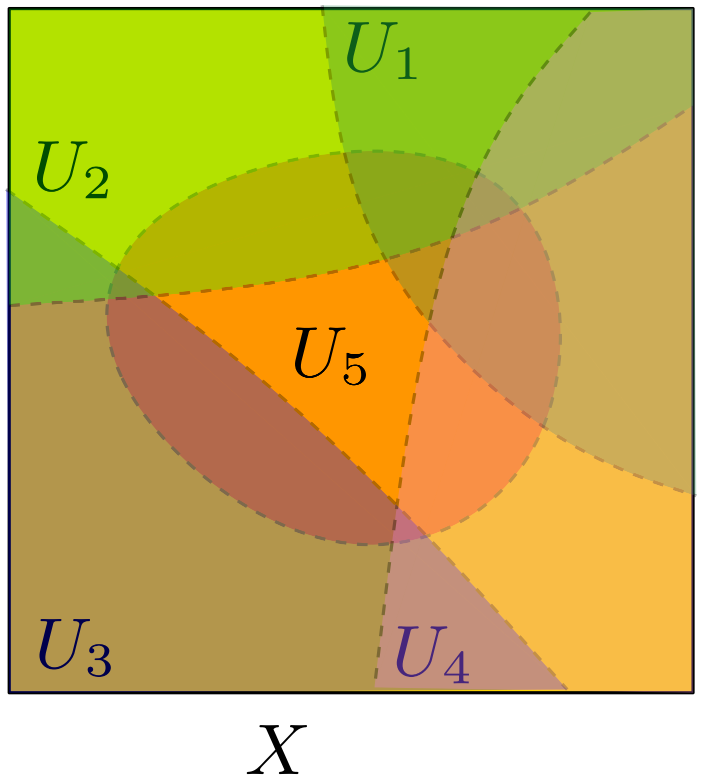
X

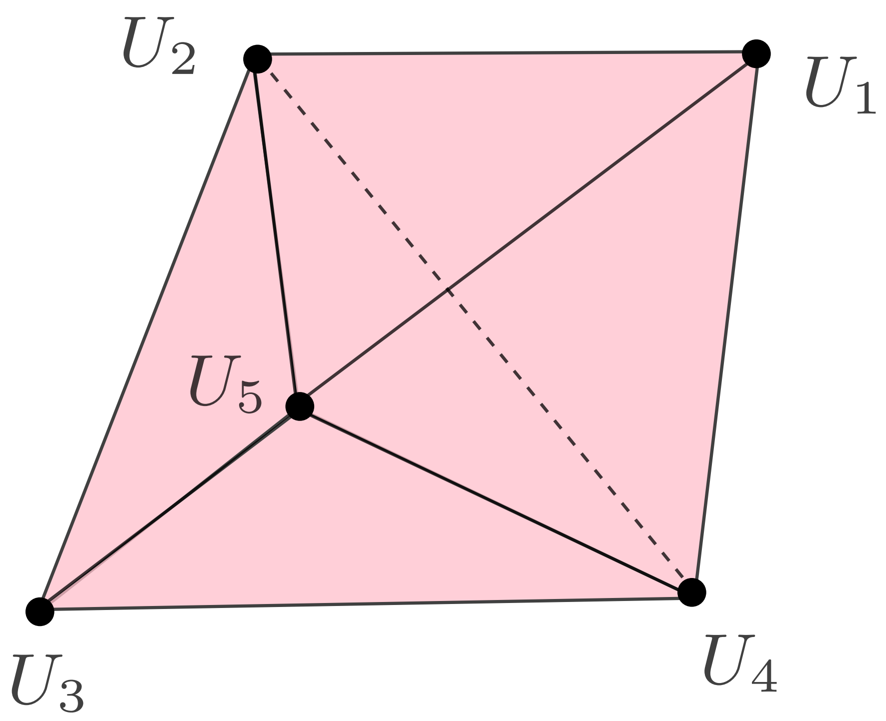
Def: An open cover of a topological space X is a collection $\mathcal{U} = (U_i)_{i \in I}$ of open subsets $U_i \subseteq X$, $i \in I$ where I is a set, such that $X \subseteq \cup_{i \in I} U_i$.

Def: Given a cover of a topological space X, $\mathcal{U} = (U_i)_{i \in I}$, its nerve is the abstract simplicial complex $C(\mathcal{U})$ whose vertex set is $\mathcal{U}$ and s.t.

$$\sigma = [U_{i_0}, U_{i_1}, \ldots, U_{i_k}] \in C(\mathcal{U}) \quad \text{if and only if} \quad \cap_{j=0}^{k} U_{i_j} \neq \emptyset.$$

Nerve complex





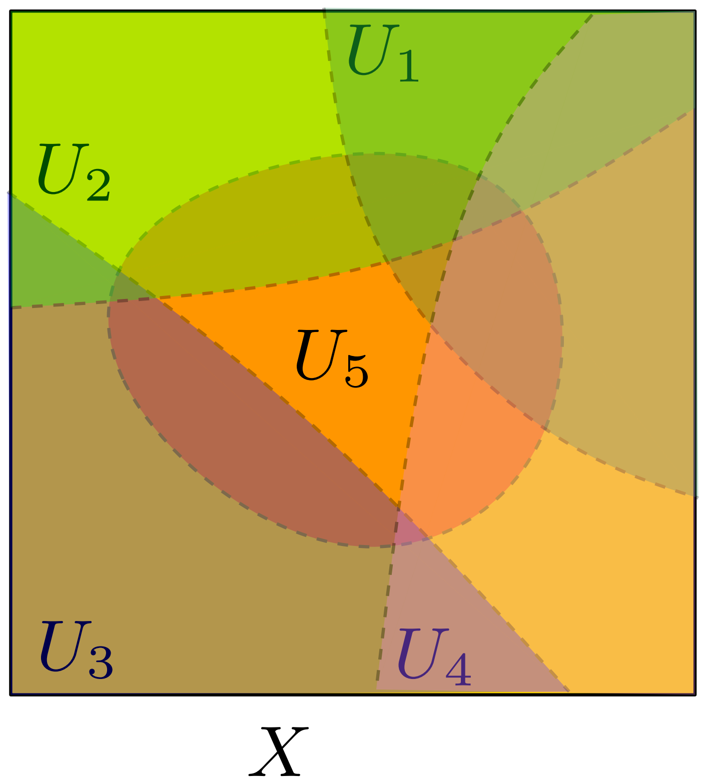
Def: An open cover of a topological space X is a collection $\mathcal{U} = (U_i)_{i\in I}$ of open subsets $U_i \subseteq X$, $i \in I$ where I is a set, such that $X \subseteq \cup_{i\in I} U_i$.

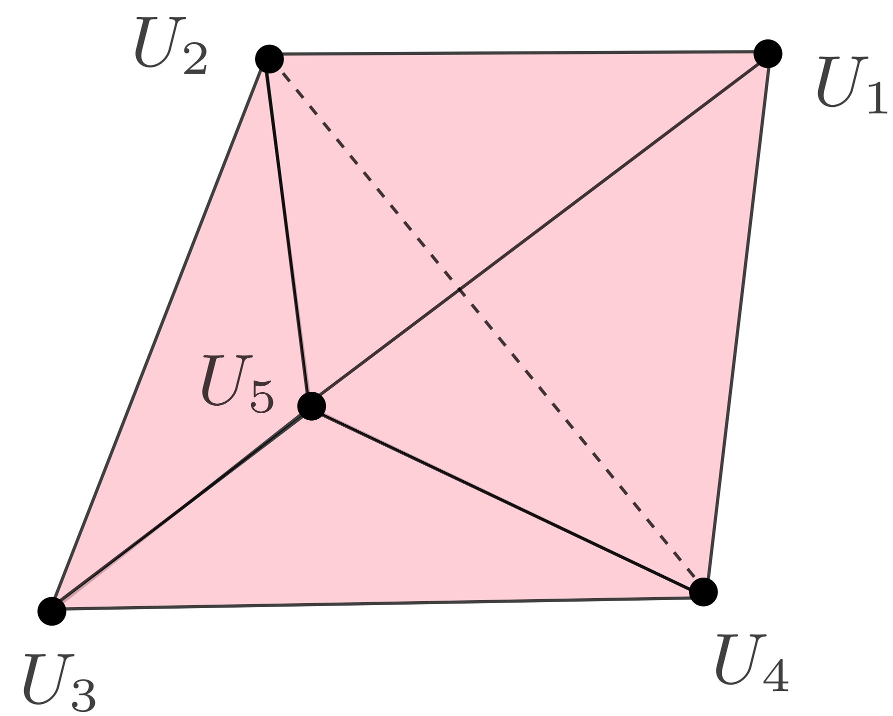
Def: Given a cover of a topological space X, $\mathcal{U} = (U_i)_{i\in I}$, its nerve is the abstract simplicial complex $C(\mathcal{U})$ whose vertex set is $\mathcal{U}$ and s.t.

$$\sigma = [U_{i_0}, U_{i_1}, \ldots, U_{i_k}] \in C(\mathcal{U}) \text{ if and only if } \cap_{j=0}^{k} U_{i_j} \neq \emptyset.$$

Nerve complex

[*On the imbedding of systems of compacta in simplicial complexes*, Borsuk, Fund. Math., 1948]





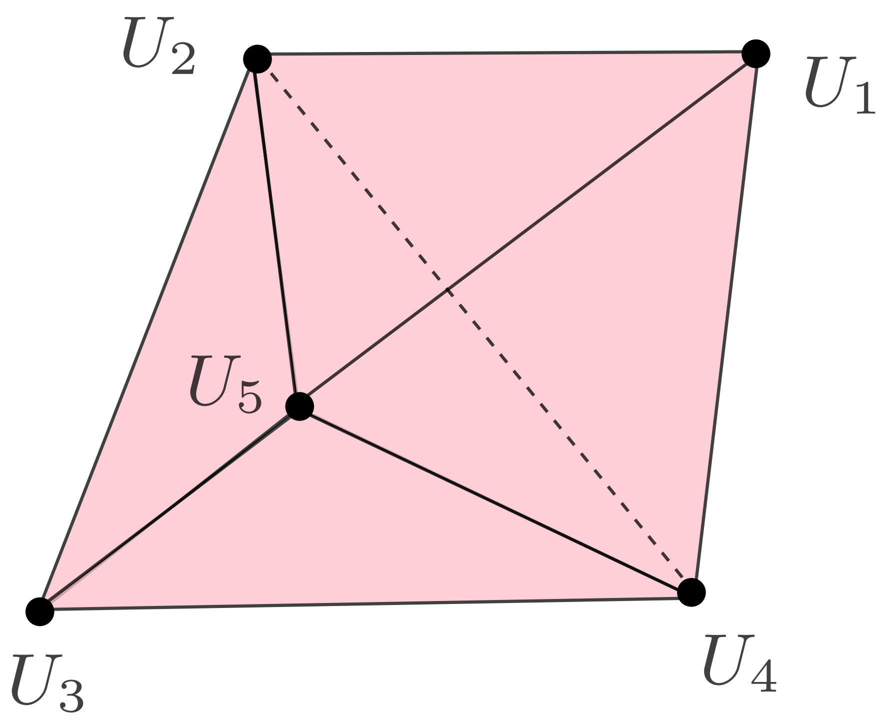
The Nerve Theorem: Let $\mathcal{U} = (U_i)_{i\in I}$ be a finite open cover of a subset X of $\mathbb{R}^d$ such that any intersection of the U_i's is either empty or convex. Then there are continuous deformations $X \to C(\mathcal{U})$ and $C(\mathcal{U}) \to X$.

Remark: More formally, one say they are homotopy equivalent.

Nerve complex

[*On the imbedding of systems of compacta in simplicial complexes*, Borsuk, Fund. Math., 1948]

Two maps $f_0 : X \to Y$ and $f_1 : X \to Y$ are homotopic if $\exists$ a continuous map $F : [0,1] \times X \to Y$ s.t. $\forall x \in X$, $F(0,x) = f_0(x)$ and $F_1(1,x) = f_1(x)$.
The spaces X and Y are homotopy equivalent if $\exists$ continuous maps $f : X \to Y$ and $g : Y \to X$ s.t. $g \circ f$ is homotopic to id_X and $f \circ g$ is homotopic to id_Y.



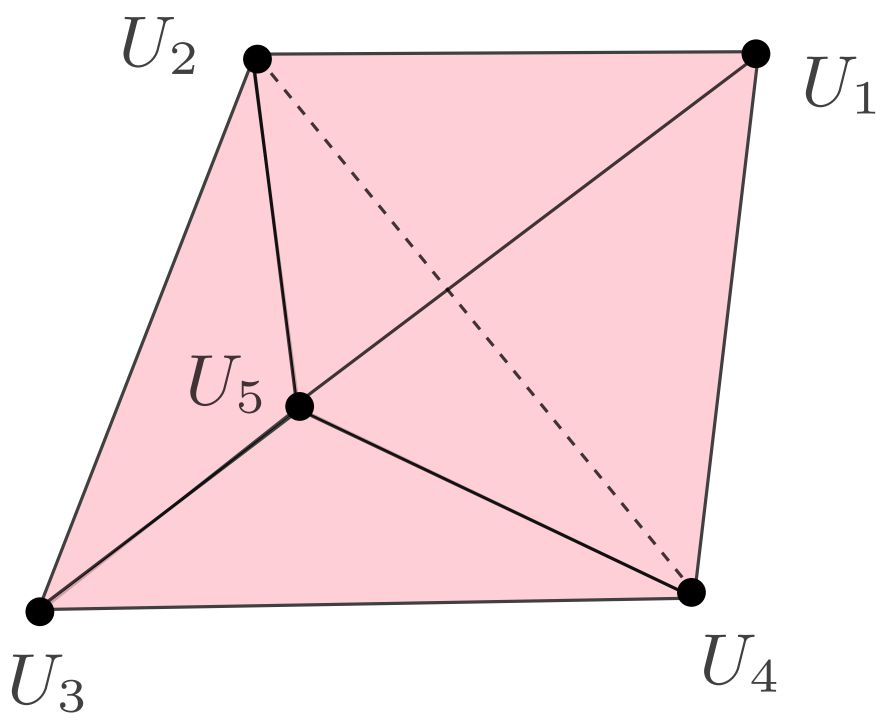
The Nerve Theorem: Let $\mathcal{U} = (U_i)_{i \in I}$ be a finite open cover of a subset X of $\mathbb{R}^d$ such that any intersection of the U_i's is either empty or convex. Then there are continuous deformations $X \to C(\mathcal{U})$ and $C(\mathcal{U}) \to X$.

Remark: More formally, one say they are homotopy equivalent.

Nerve complex

[*On the imbedding of systems of compacta in simplicial complexes*, Borsuk, Fund. Math., 1948]

Two maps $f_0 : X \to Y$ and $f_1 : X \to Y$ are homotopic if $\exists$ a continuous map $F : [0,1] \times X \to Y$ s.t. $\forall x \in X$, $F(0,x) = f_0(x)$ and $F_1(1,x) = f_1(x)$.
The spaces X and Y are homotopy equivalent if $\exists$ continuous maps $f : X \to Y$ and $g : Y \to X$ s.t. $g \circ f$ is homotopic to id_X and $f \circ g$ is homotopic to id_Y.



The Nerve Theorem: Let $\mathcal{U} = (U_i)_{i \in I}$ be a finite open cover of a subset X of $\mathbb{R}^d$ such that any intersection of the U_i's is either empty or convex. Then there are continuous deformations $X \to C(\mathcal{U})$ and $C(\mathcal{U}) \to X$.

Remark: More formally, one say they are homotopy equivalent.

Ex: There are continuous deformations between the Čech complex $C(P,r)$ and the union of balls $\cup_{p \in P} B(p,r)$.

Cover complexes

Q: How to build meaningful covers?

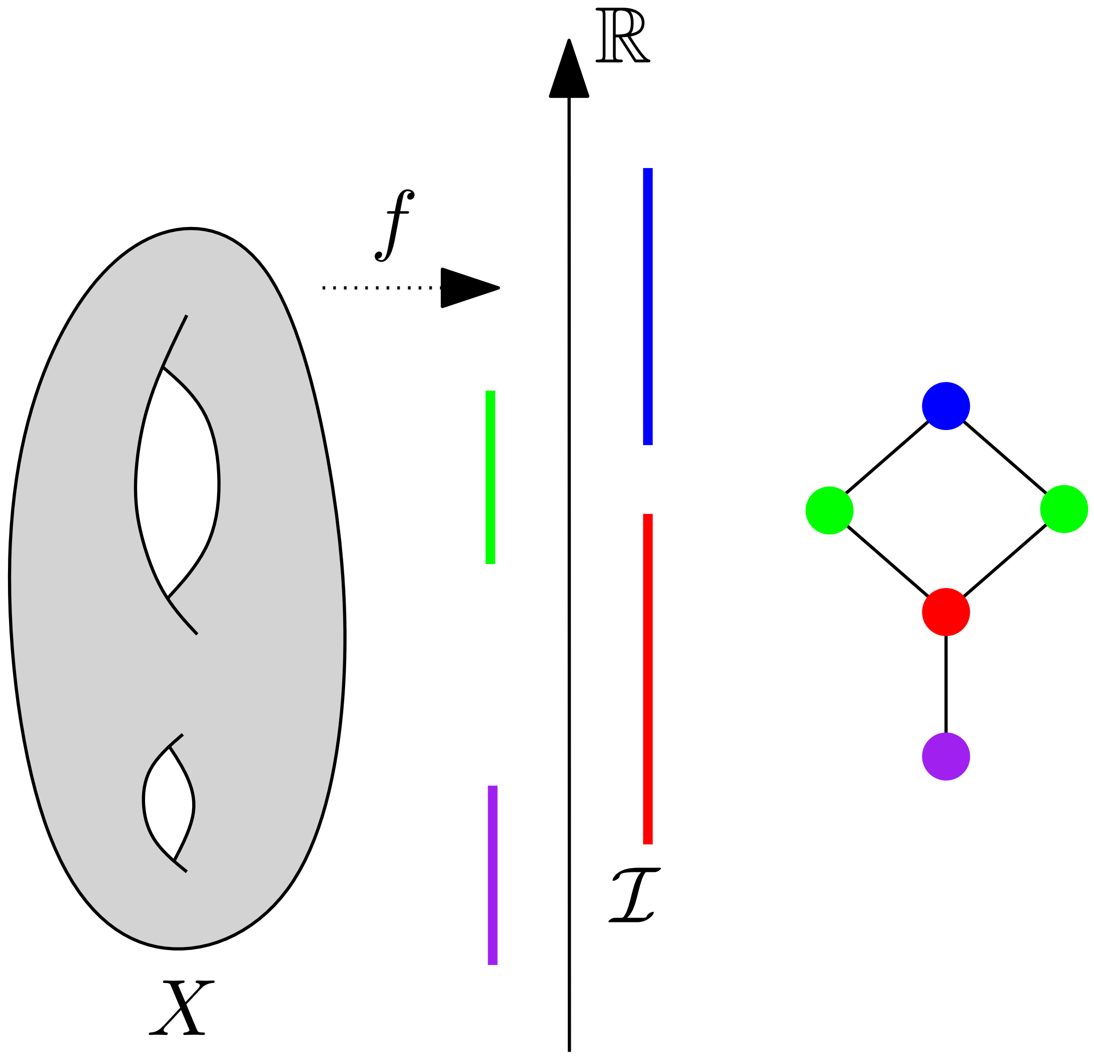
Two directions:

Cover complexes

Q: How to build meaningful covers?

Two directions:

1. Using a function (lens) defined on the data:
$\rightarrow$ the Mapper algorithm
$\rightarrow$ exploratory data analysis

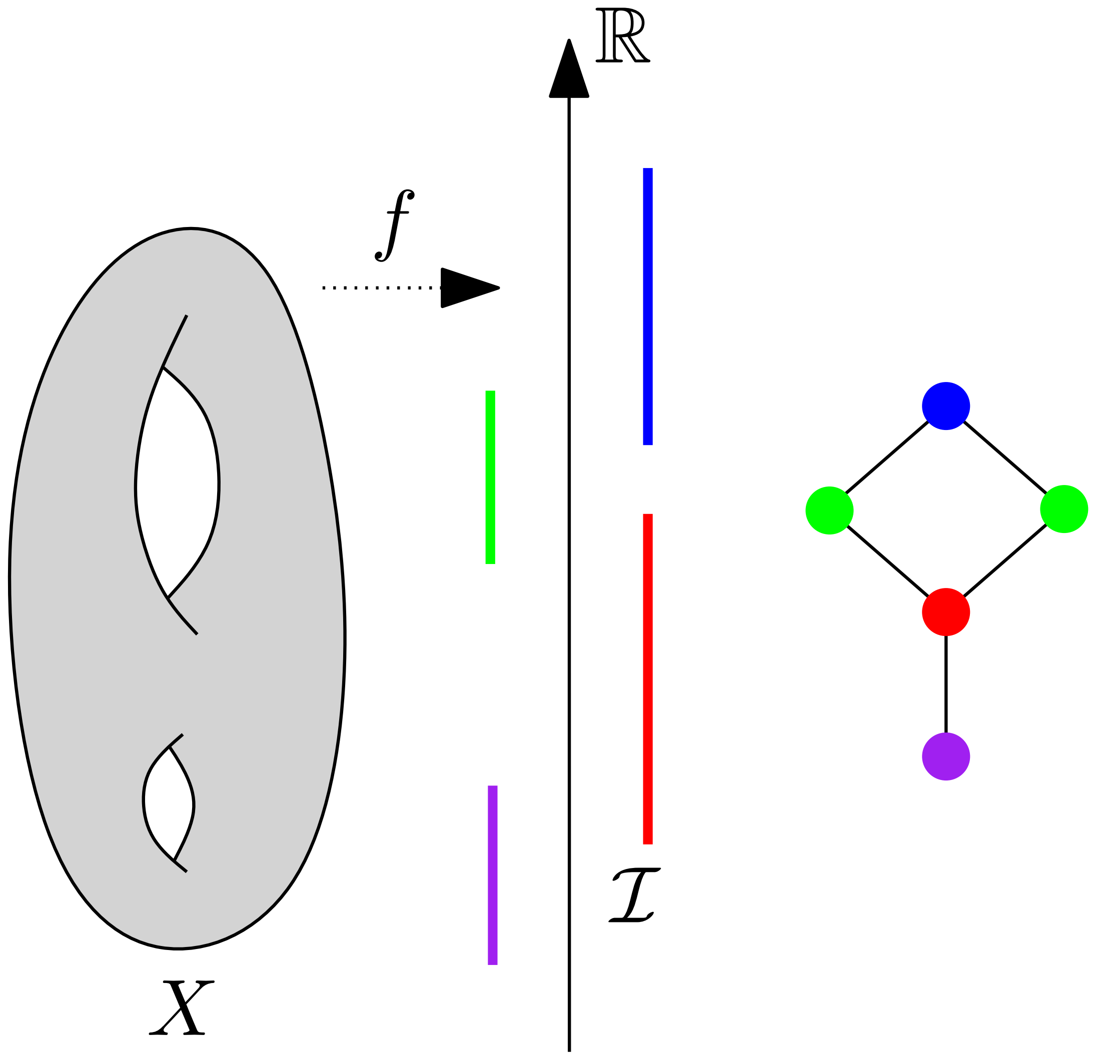


Cover complexes

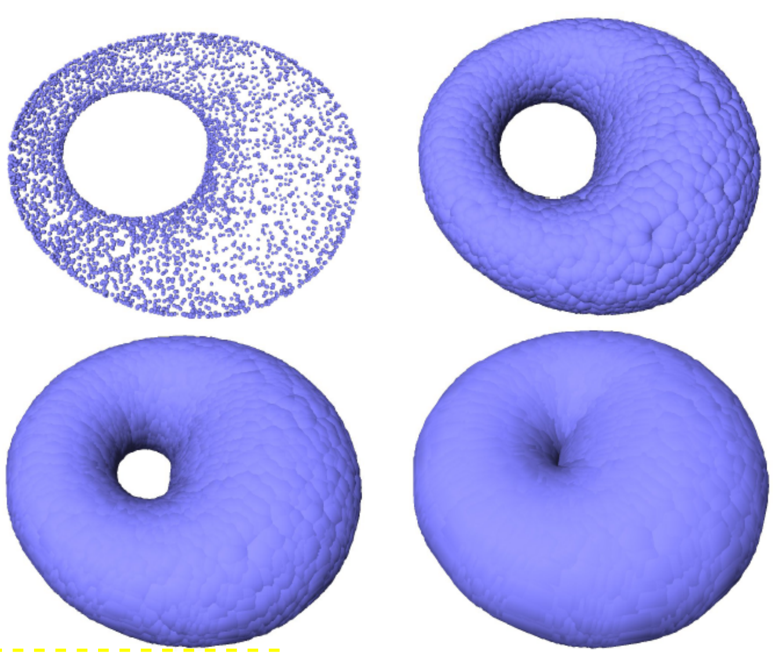
Q: How to build meaningful covers?

Two directions:

1. Using a function (lens) defined on the data:
$\rightarrow$ the Mapper algorithm
$\rightarrow$ exploratory data analysis



2. Covering data by balls:
$\rightarrow$ distance functions frameworks, persistence-based signatures,...
$\rightarrow$ geometric inference, provide a framework to establish various theoretical results in TDA.

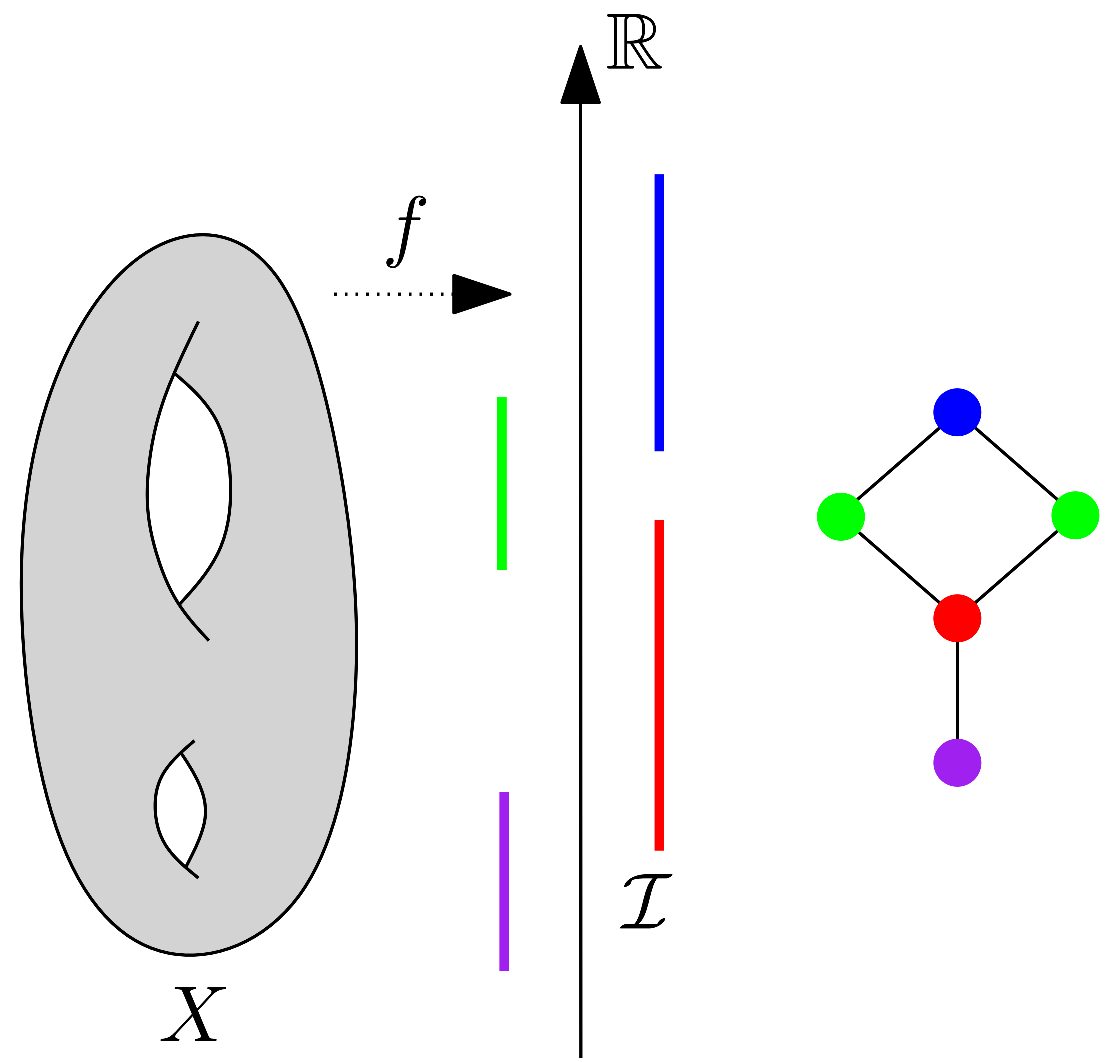


Cover complexes

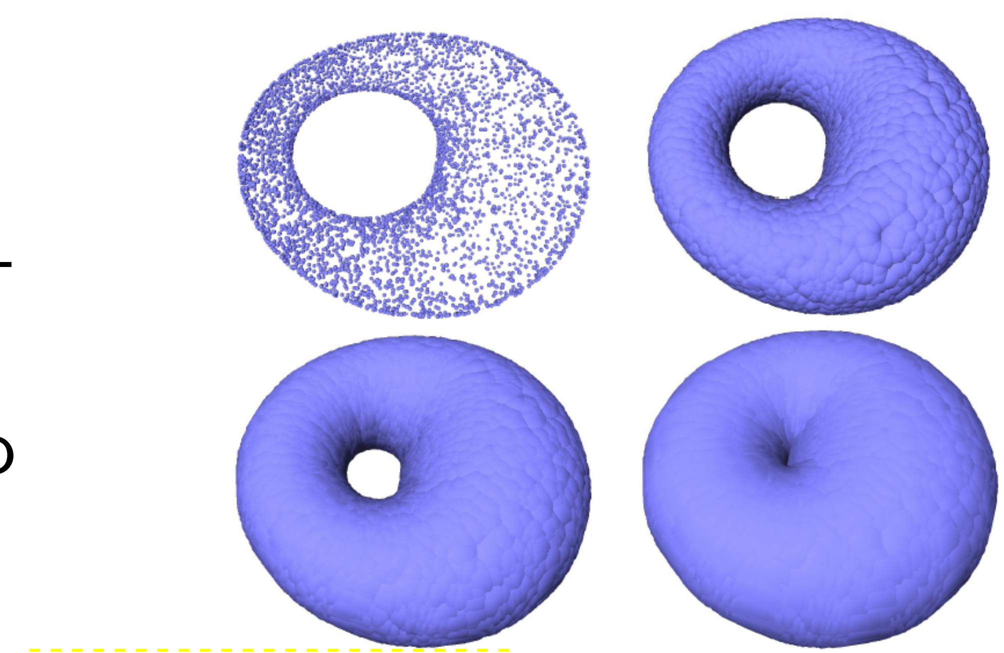
Q: How to build meaningful covers?

Two directions:

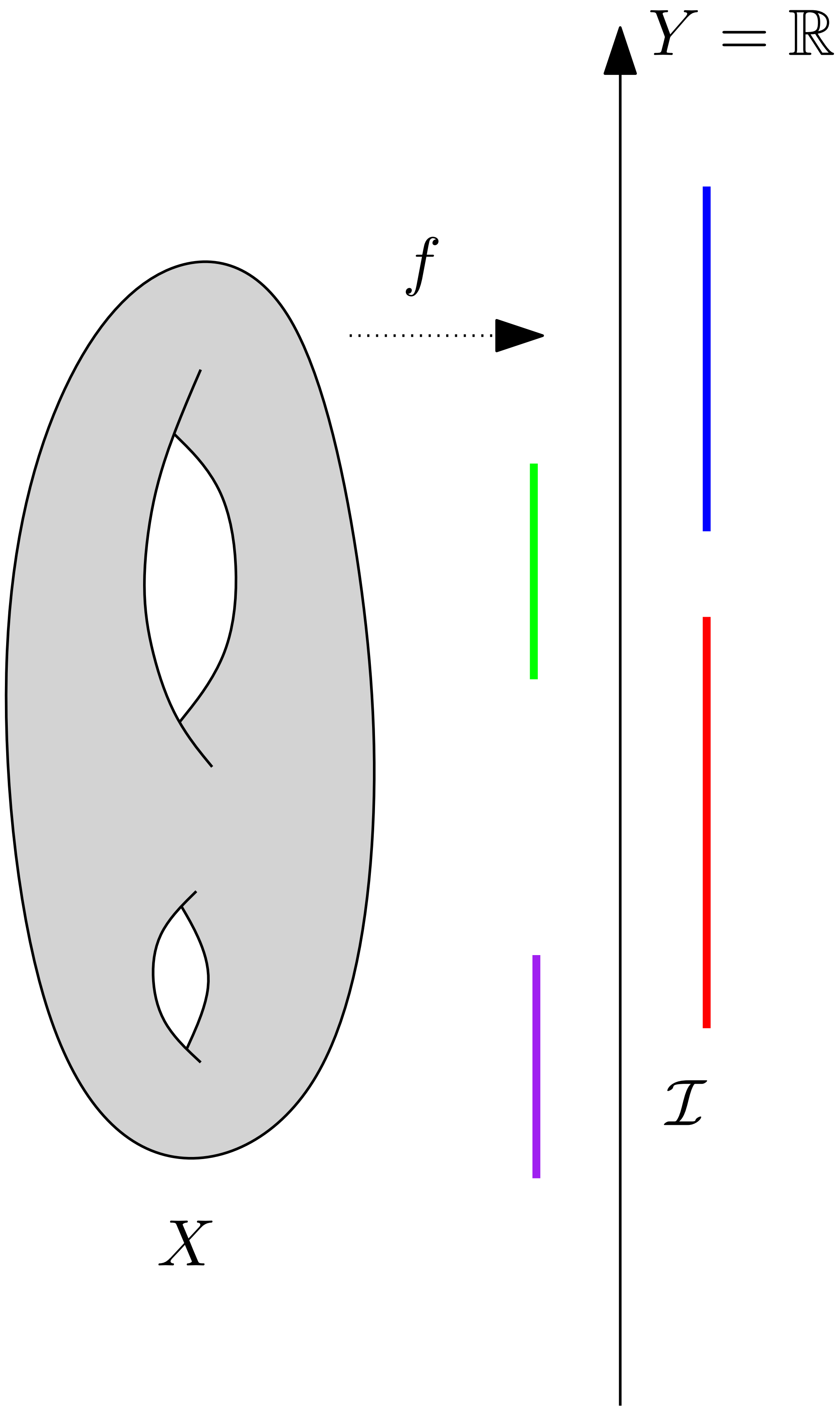
1. Using a function (lens) defined on the data:
$\rightarrow$ the Mapper algorithm
$\rightarrow$ exploratory data analysis



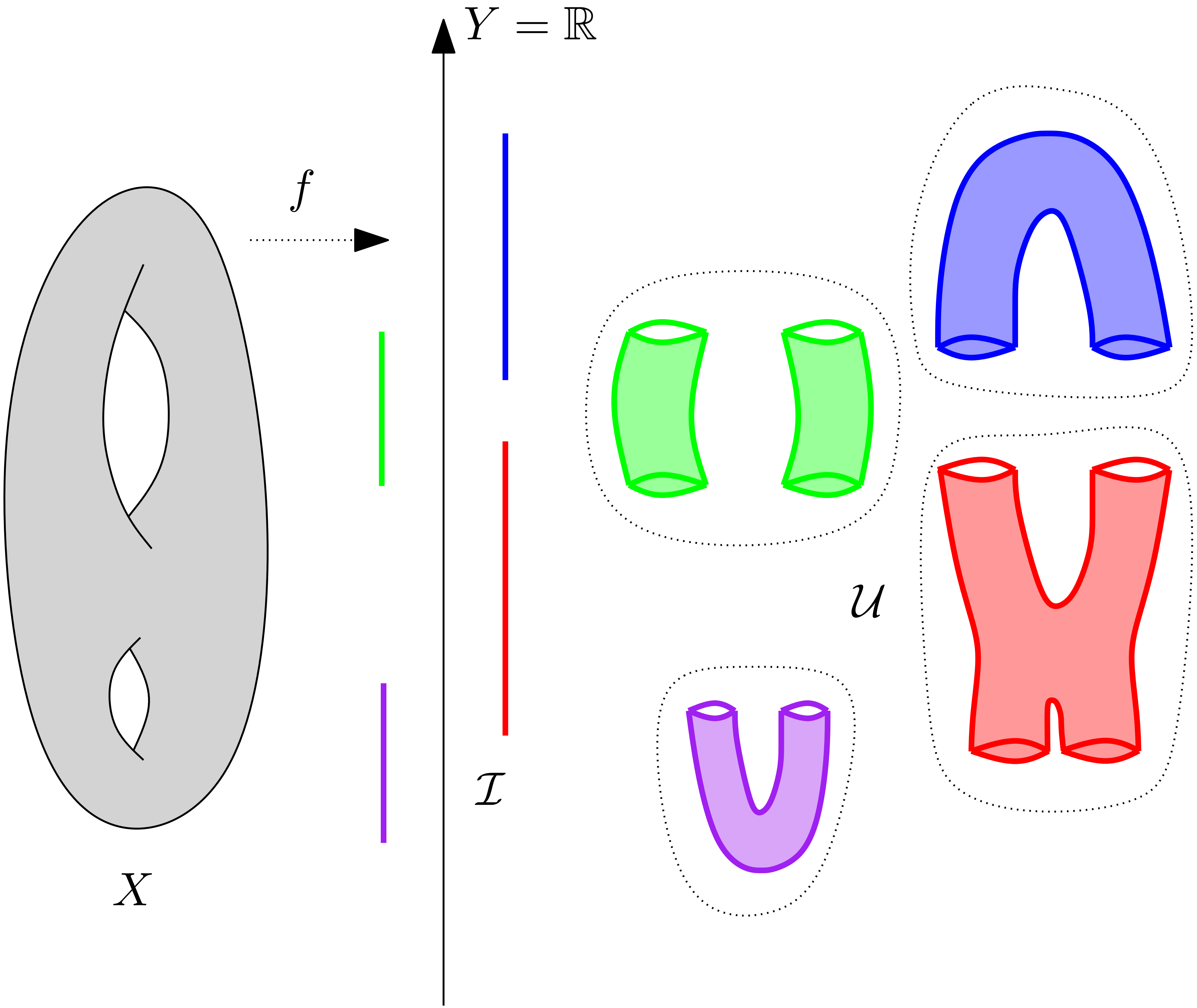
2. Covering data by balls:
$\rightarrow$ distance functions frameworks, persistence-based signatures,...
$\rightarrow$ geometric inference, provide a framework to establish various theoretical results in TDA.



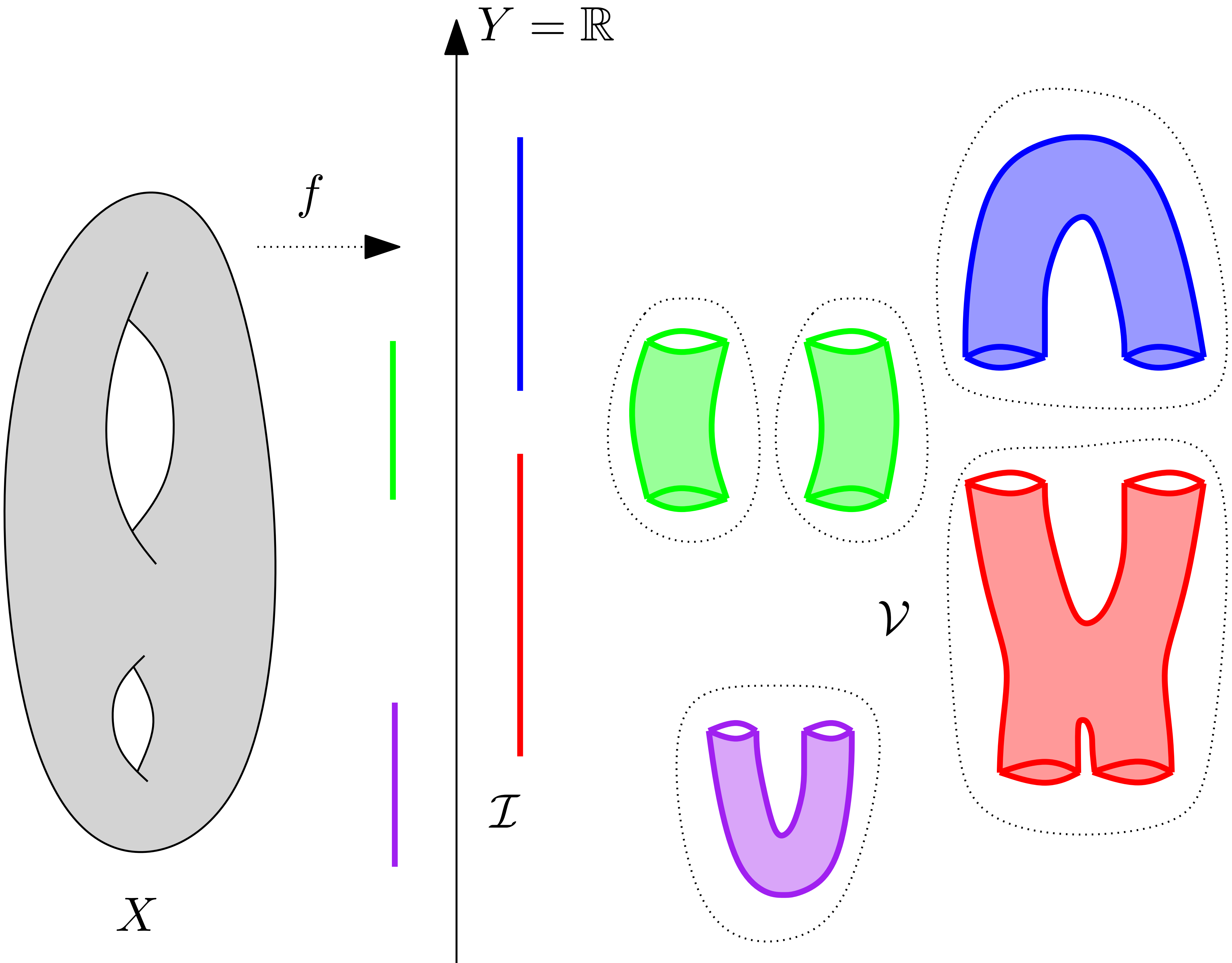
Mapper in the continuous setting



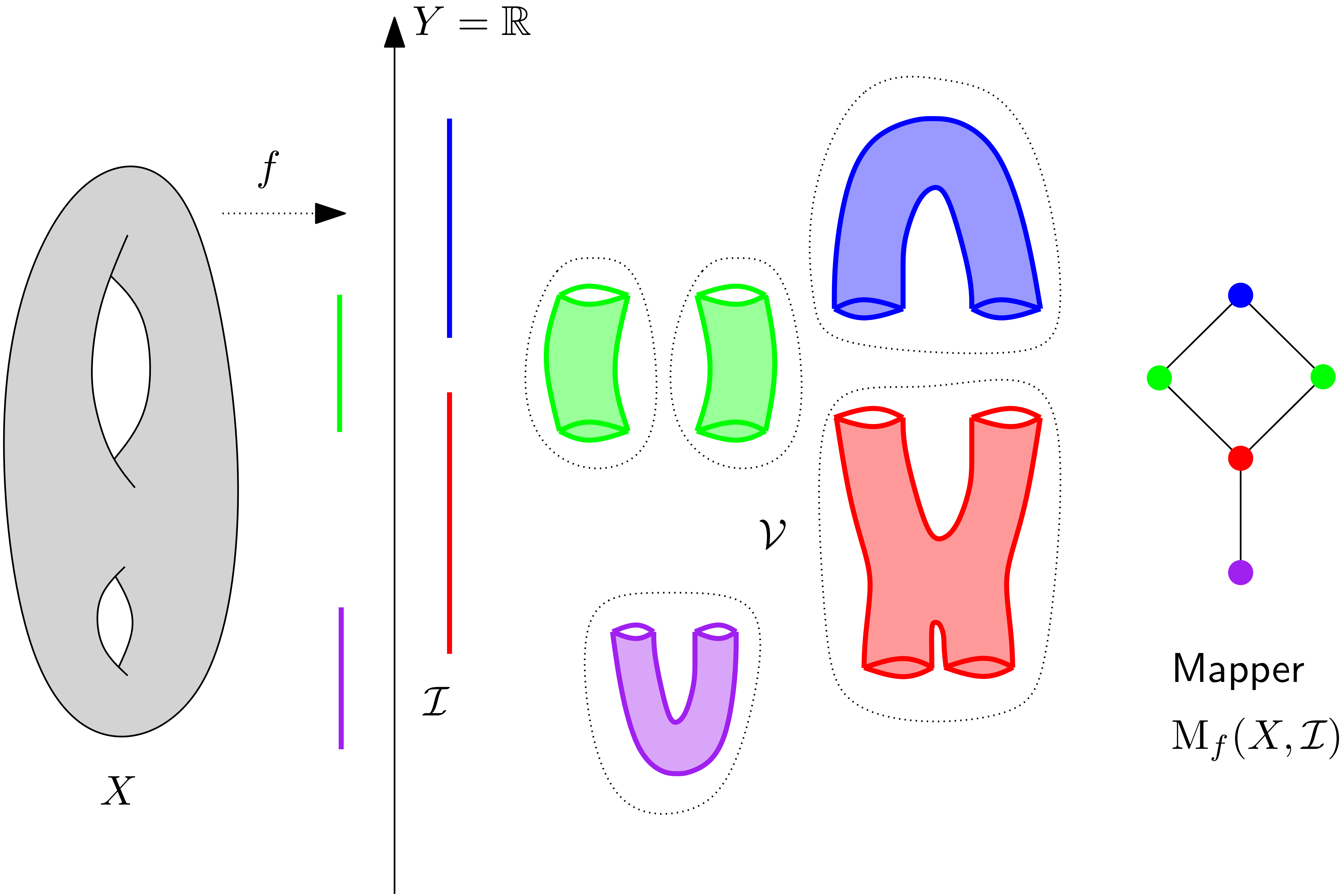
Mapper in the continuous setting



Mapper in the continuous setting



Mapper in the continuous setting



Mapper in the continuous setting

Input:

- topological space X
- continuous function $f : X \to Y$ (99% of the time $Y = \mathbb{R}^D$)
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals: $\mathrm{im}(f) \subseteq \bigcup_{I \in \mathcal{I}} I$

Method:

- Compute *pullback cover* $\mathcal{U}$ of X: $\mathcal{U} = \{f^{-1}(I)\}_{I \in \mathcal{I}}$
- Refine $\mathcal{U}$ by separating each of its elements into its various connected components in $X \to$ connected cover $\mathcal{V}$
- The Mapper is the *nerve* of $\mathcal{V}$:
  - 1 vertex per element $V \in \mathcal{V}$
  - 1 edge per intersection $V \cap V' \neq \emptyset$, $V, V' \in \mathcal{V}$
  - 1 k-simplex per $(k+1)$-fold intersection $\bigcap_{i=0}^{k} V_i \neq \emptyset$, $V_0, \cdots, V_k \in \mathcal{V}$

Mapper in practice

Input:

- point cloud $P \subseteq X$ with metric d_P
- continuous function $f : P \to \mathbb{R}$
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals: $\mathrm{im} f \subseteq \bigcup_{I \in \mathcal{I}} I$

Method:

- Compute *pullback cover* $\mathcal{U}$ of P: $\mathcal{U} = \{f^{-1}(I)\}_{I \in \mathcal{I}}$
- Refine $\mathcal{U}$ by separating each of its elements into its various clusters, as identified by a clustering algorithm $\to$ connected cover $\mathcal{V}$
- The Mapper is the *nerve* of $\mathcal{V}$:
    - 1 vertex per element $V \in \mathcal{V}$
    - 1 edge per intersection $V \cap V' \neq \emptyset$, $V, V' \in \mathcal{V}$
    - 1 k-simplex per $(k+1)$-fold intersection $\bigcap_{i=0}^{k} V_i \neq \emptyset$, $V_0, \cdots, V_k \in \mathcal{V}$

intersections are assessed by the presence of common data points

Mapper in practice

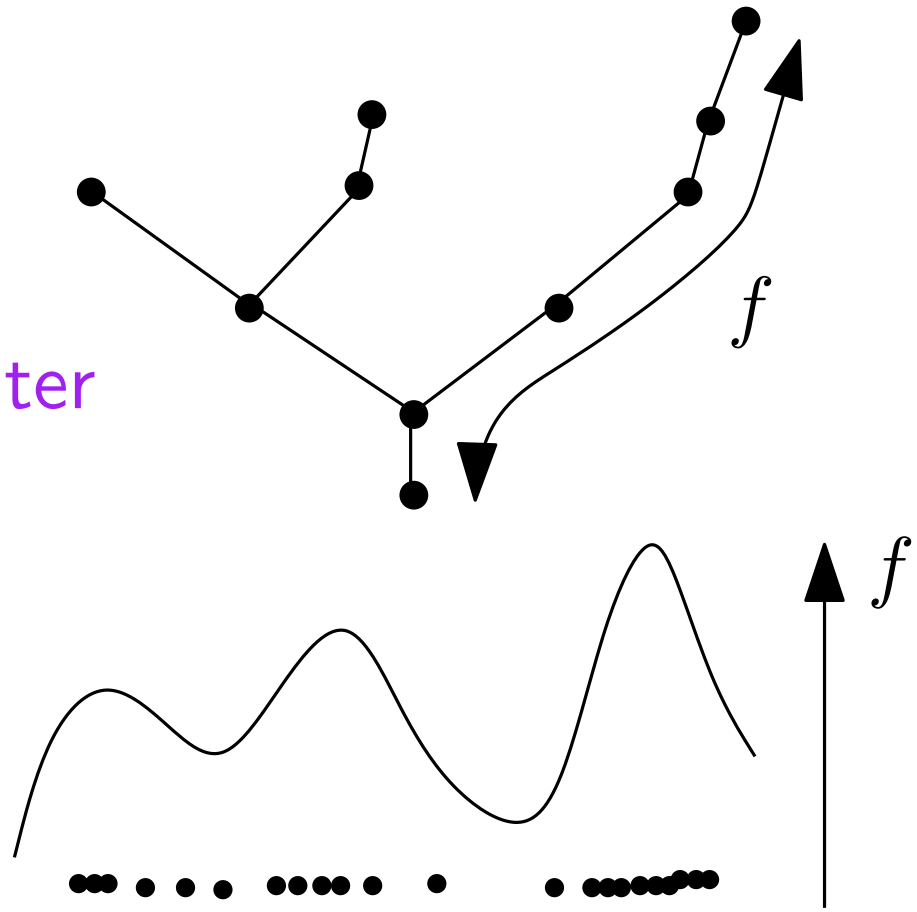
Parameters:

- function $f : P \to \mathbb{R}$
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals
- clustering algorithm $\mathcal{C}$

Mapper in practice

Parameters:

- function $f : P \to \mathbb{R}$ lens or filter
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals
- clustering algorithm $\mathcal{C}$



Classical choices:

- density estimates
- centrality $f(x) = \sum_{y \in X} d(x, y)$
- eccentricity $f(x) = \max_{y \in X} d(x, y)$
- PCA coordinates

- Eigenfunctions of graph laplacians.
- Functions detecting outliers.
- Distance to a root point.
- Prior knowledge

Mapper in practice

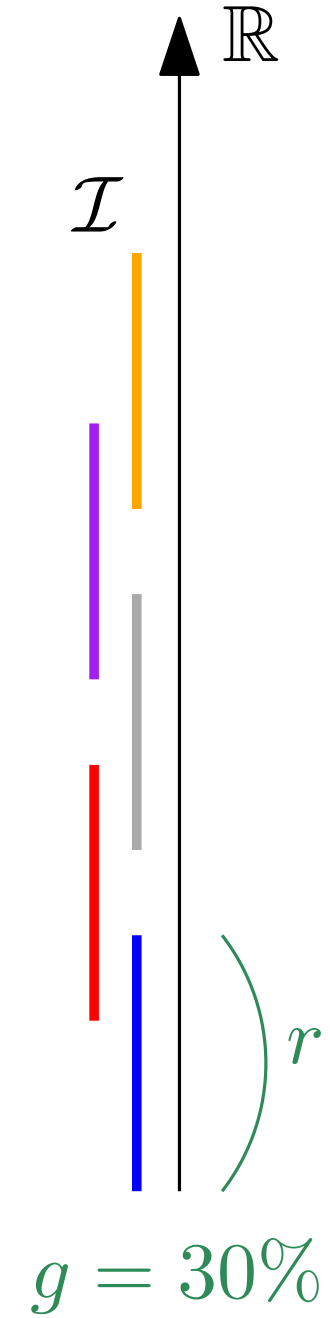
Parameters:

- function $f : P \to \mathbb{R}$

- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals

- clustering algorithm $\mathcal{C}$

range scale



Uniform cover:

- resolution / granularity: r (diameter of intervals)
- gain: g (percentage of overlap)

Mapper in practice

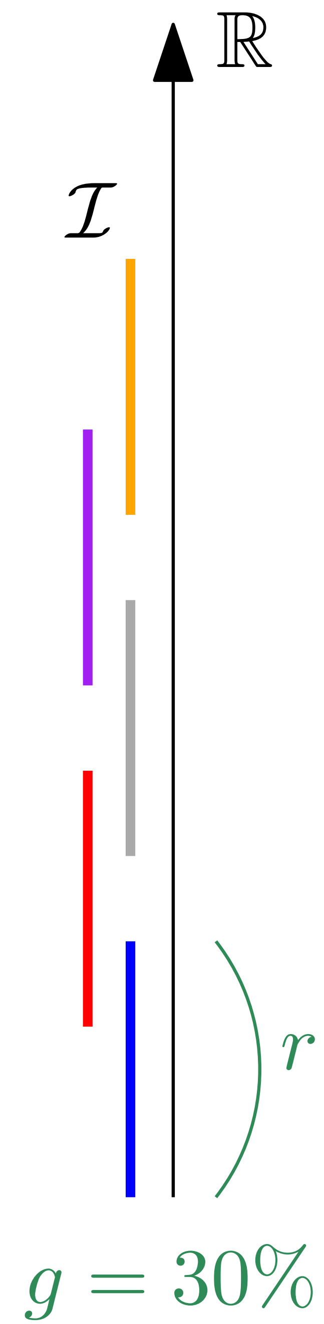
Parameters:

- function $f : P \to \mathbb{R}$
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals
- clustering algorithm $\mathcal{C}$

range scale

Uniform cover:

- resolution / granularity: r (diameter of intervals)
- gain: g (percentage of overlap)



Intuition:

- small $r \to$ finer resolution, more nodes.
- large $r \to$ rougher resolution, less nodes.

- small $g \to$ less connectivity, nerve dimension small.
- large $g \to$ more connectivity, nerve dimension large.

Mapper in practice

Parameters:

- function $f : P \to \mathbb{R}$
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals
- clustering algorithm $\mathcal{C}$

Classical choices:

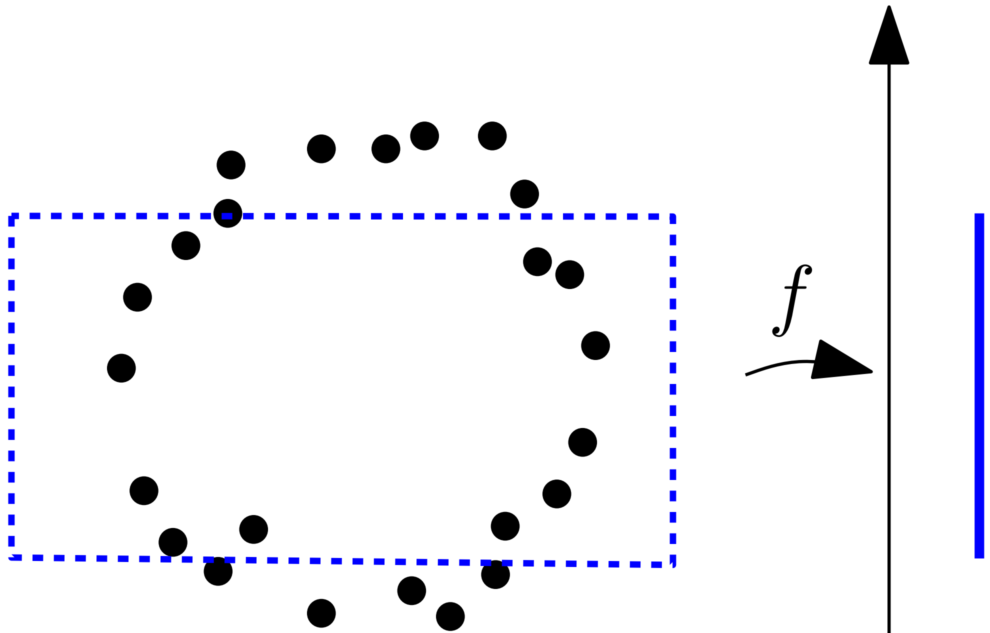
- any clustering algorithm works
- different clustering algorithms/parameters for each preimage
- for theoretical reasons, we prefer to work with hierarchical clustering with (predefined) neighborhood size δ

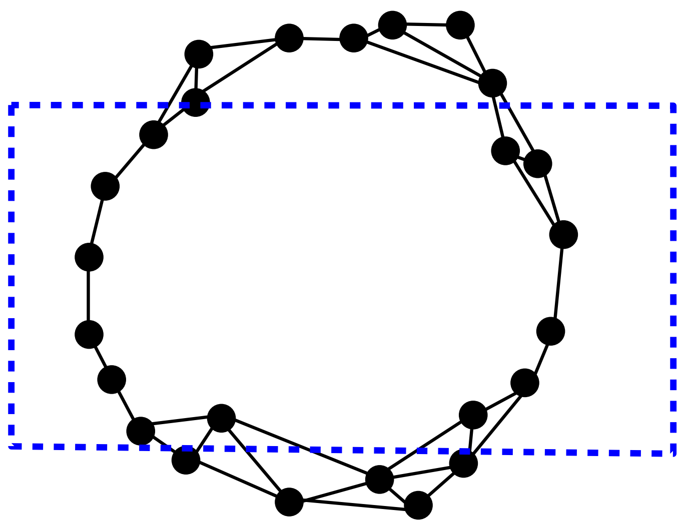
geometric scale

Mapper in practice

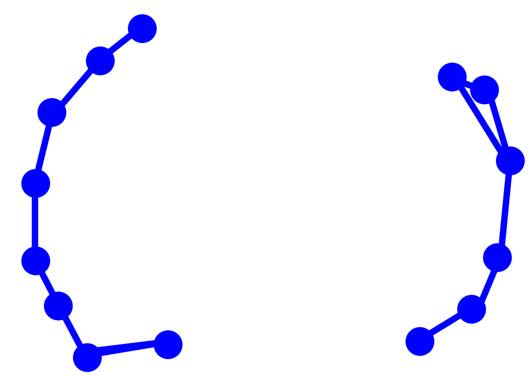
Parameters:

- function $f : P \to \mathbb{R}$
- cover $\mathcal{I}$ of $\mathrm{im}(f)$ by open intervals
- clustering algorithm $\mathcal{C}$



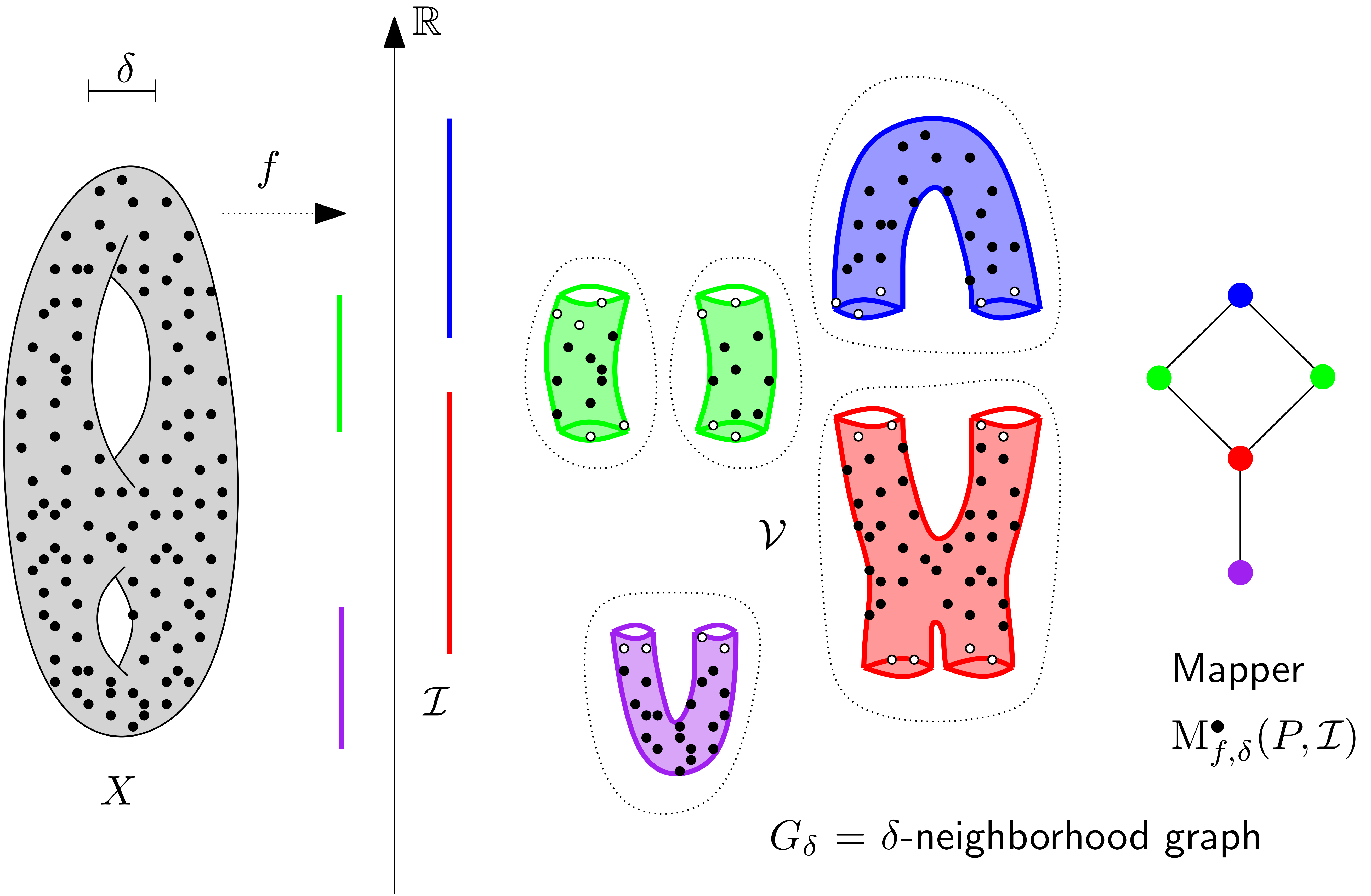


Build a neighboring graph (kNN,...)

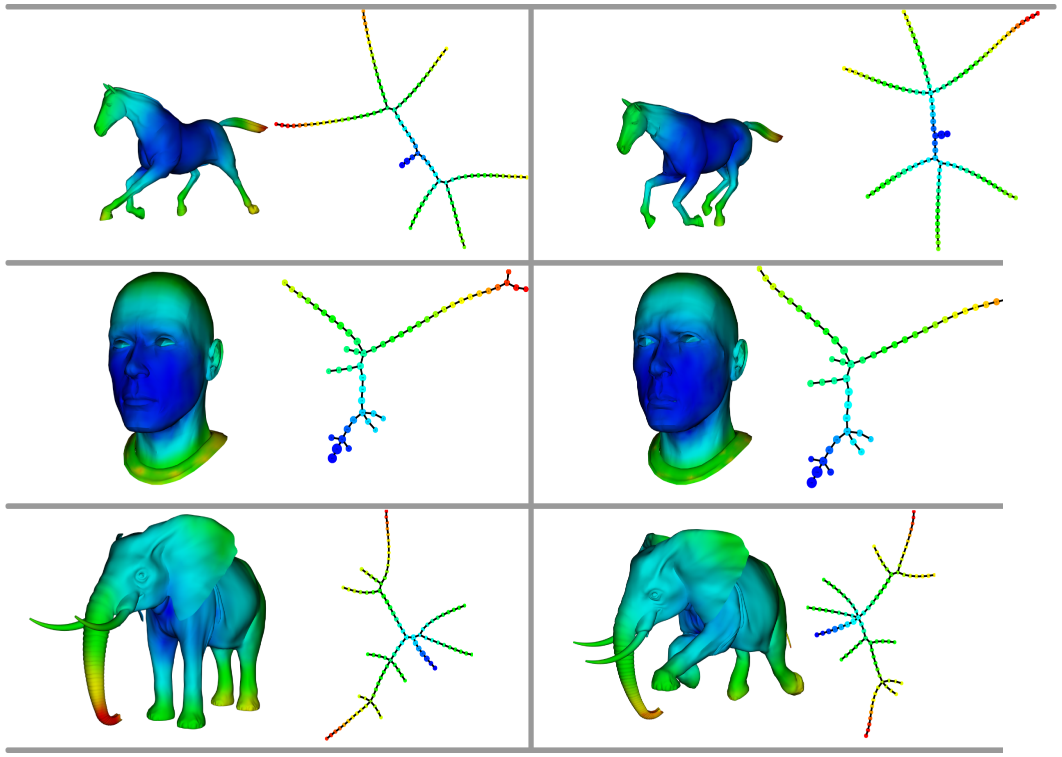


Take the connected components of the subgraph spanned by the vertices in the preimage $f^{-1}(U)$.

Mapper in practice

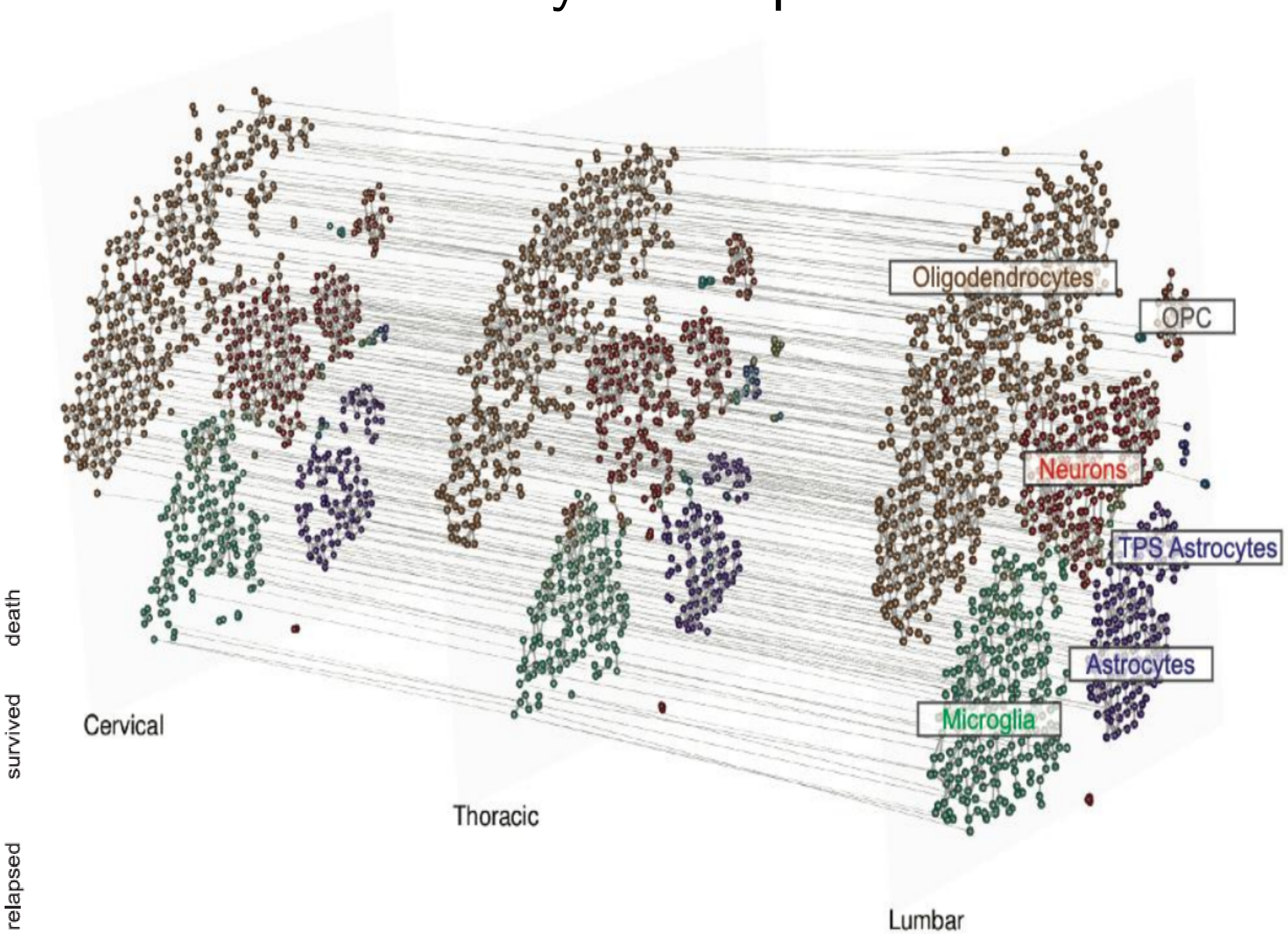


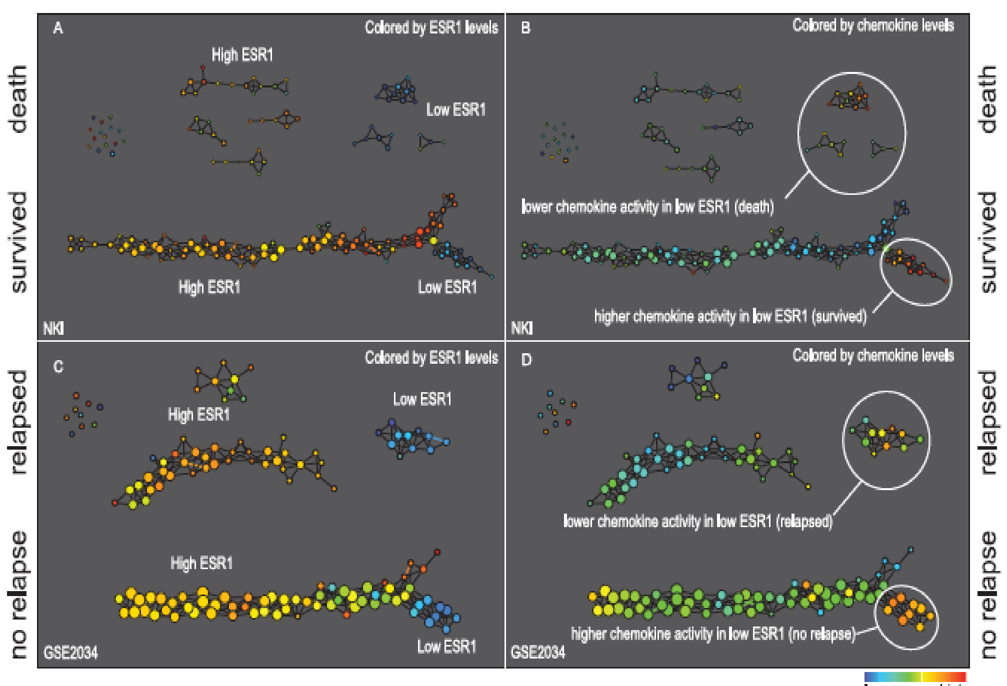
Applications



3D shape classification

Genomic analysis of spinal cord





Breast cancer subtype identification

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Pbm: Looking for homotopy equivalences is extremely difficult. Are there mathematical quantities that are invariant to homotopy equivalences **and** easy to compute?

Computational Topology (I): Simplicial Complexes and Homology

1. Simplicial Complexes
2. Nerve Theorem
3. Homology Groups

Pbm: Looking for homotopy equivalences is extremely difficult. Are there mathematical quantities that are invariant to homotopy equivalences **and** easy to compute?

A: The *holes*, encoded in the *homology groups* H_k, $k \in \mathbb{N}$.

Introduction

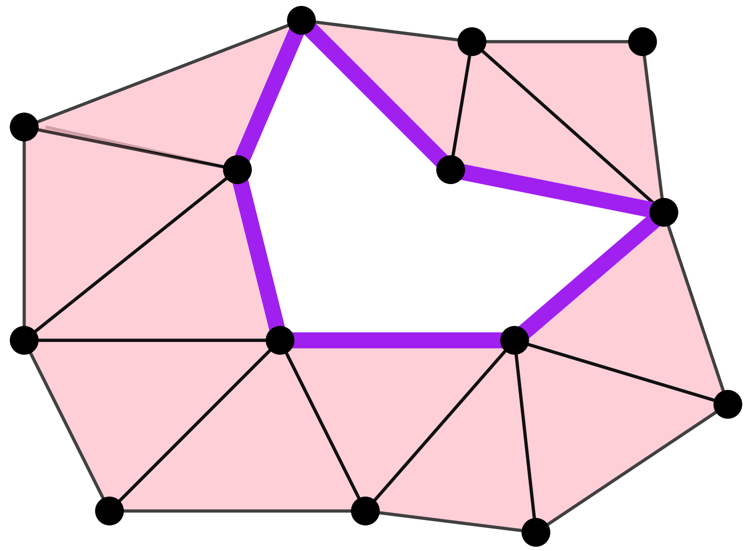
Introduction

Q: How to characterize a hole in a simplicial complex?

Introduction

Q: How to characterize a hole in a simplicial complex?

A: A hole (in 1D) is a path whose first and end points are the same, a loop.

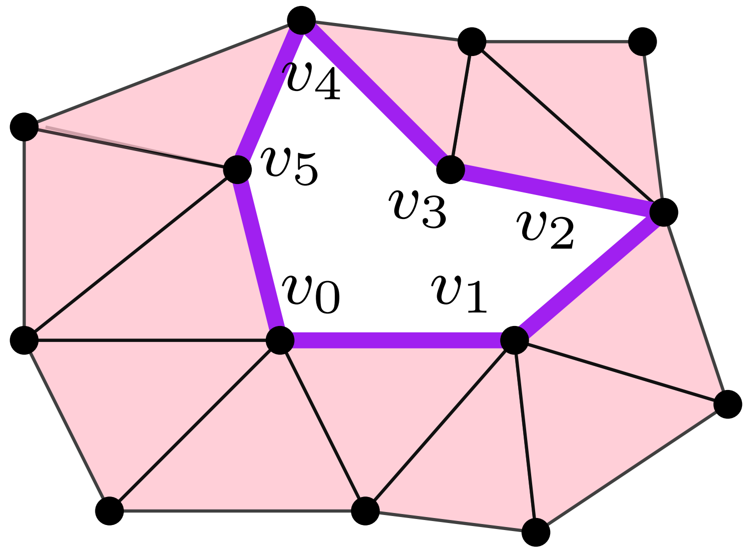


Introduction

Q: How to characterize a hole in a simplicial complex?

A: A hole (in 1D) is a path whose first and end points are the same, a loop.

The sequence of 1-dimensional simplices $[v_0, v_1]$, $[v_1, v_2]$, $[v_2, v_3]$, $[v_3, v_4]$, $[v_4, v_5]$, $[v_5, v_0]$ is a hole.

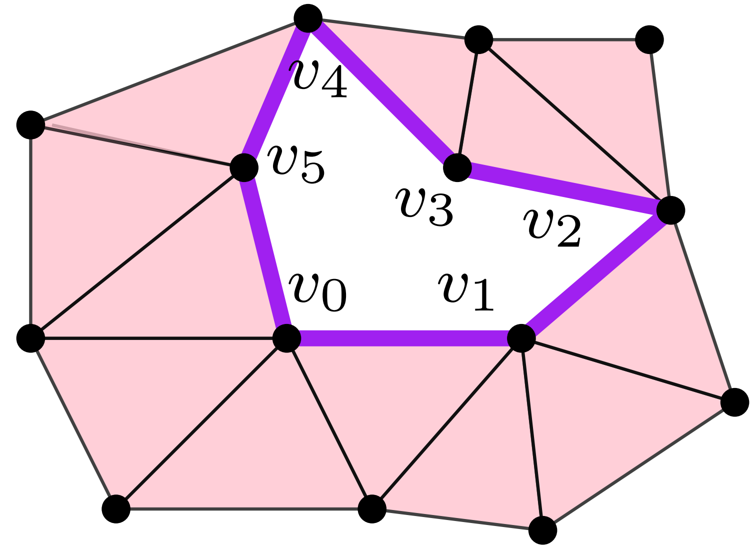


Introduction

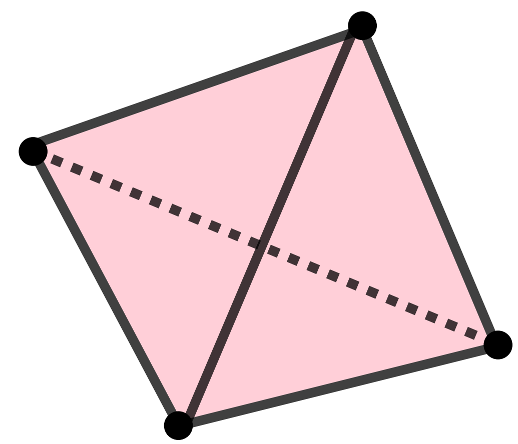
Q: How to characterize a hole in a simplicial complex?

A: A hole (in 1D) is a path whose first and end points are the same, a loop.

The sequence of 1-dimensional simplices $[v_0, v_1]$, $[v_1, v_2]$, $[v_2, v_3]$, $[v_3, v_4]$, $[v_4, v_5]$, $[v_5, v_0]$ is a hole.



But what about higher dimensional holes (like the inside of a tetrahedron)?

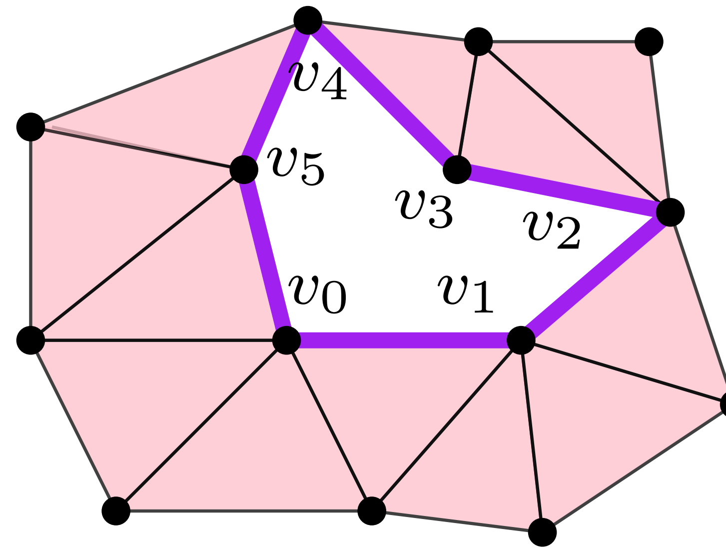


Introduction

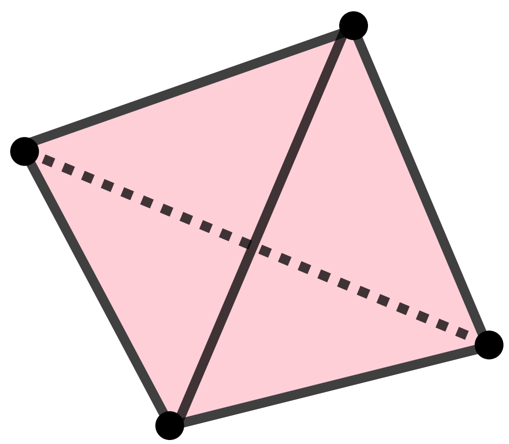
Q: How to characterize a hole in a simplicial complex?

A: A hole (in 1D) is a path whose first and end points are the same, a loop.

The sequence of 1-dimensional simplices $[v_0, v_1]$, $[v_1, v_2]$, $[v_2, v_3]$, $[v_3, v_4]$, $[v_4, v_5]$, $[v_5, v_0]$ is a hole.



But what about higher dimensional holes (like the inside of a tetrahedron)?



A: A hole in dimension d is a simplicial complex in which each $(d-1)$-simplex appears an even number of times.

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

Def: The *boundary* ∂_d of a d-simplex is the $(d-1)$-chain:

$$\partial_d[v_1, \ldots, v_{d+1}] = \sum_{i=1}^{d+1} [v_1, \ldots, v_{i-1}, v_{i+1}, \ldots, v_{d+1}]$$

It extends linearly to d-chains.

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

Ex: Let $C = [v_0, v_1] + [v_1, v_2] + [v_2, v_3] + [v_3, v_4] + [v_4, v_5] + [v_5, v_0]$.

$$\partial_1 C = \partial_1 [v_0, v_1] + \partial_1 [v_1, v_2] + \partial_1 [v_2, v_3] + \partial_1 [v_3, v_4] + \partial_1 [v_4, v_5] + \partial_1 [v_5, v_0]$$

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

Ex: Let $C = [v_0, v_1] + [v_1, v_2] + [v_2, v_3] + [v_3, v_4] + [v_4, v_5] + [v_5, v_0]$.

$$\partial_1 C = \partial_1[v_0, v_1] + \partial_1[v_1, v_2] + \partial_1[v_2, v_3] + \partial_1[v_3, v_4] + \partial_1[v_4, v_5] + \partial_1[v_5, v_0]$$
$$= [v_0] + [v_1] + [v_1] + [v_2] + [v_2] + [v_3] + [v_3] + [v_4] + [v_4] + [v_5] + [v_5] + [v_0]$$

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

Ex: Let $C = [v_0, v_1] + [v_1, v_2] + [v_2, v_3] + [v_3, v_4] + [v_4, v_5] + [v_5, v_0]$.

$$\partial_1 C = \partial_1[v_0, v_1] + \partial_1[v_1, v_2] + \partial_1[v_2, v_3] + \partial_1[v_3, v_4] + \partial_1[v_4, v_5] + \partial_1[v_5, v_0]$$

$$= [v_0] + \cancel{[v_1] + [v_1]} + \cancel{[v_2] + [v_2]} + \cancel{[v_3] + [v_3]} + \cancel{[v_4] + [v_4]} + \cancel{[v_5] + [v_5]} + [v_0]$$

$$= [v_0] + [v_0] = 0.$$

Def: A d-*cycle* is a d-chain C s.t. $\partial_d C = 0$.

The homology groups

Def: A d-*chain* C is a formal sum of d-simplices with coefficients in $\mathbb{Z}/2\mathbb{Z}$:

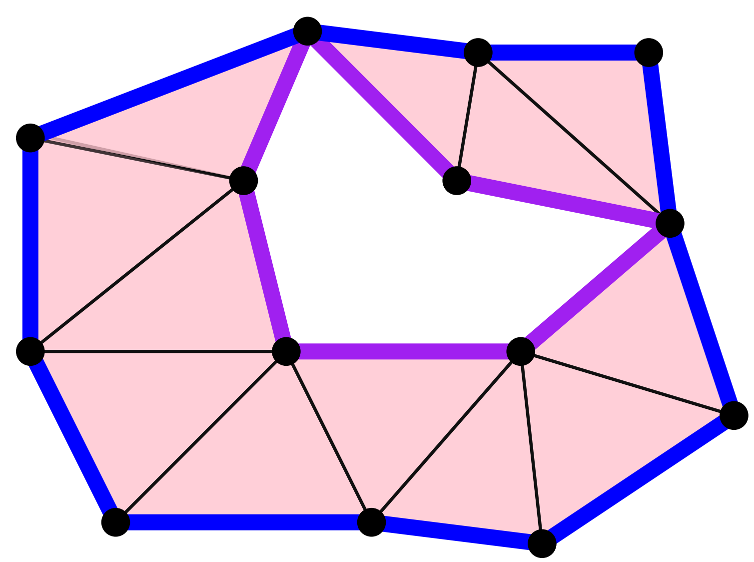
$$C = \sum_{\dim(\sigma)=d} \alpha_\sigma \delta_\sigma, \quad \text{where } \delta_\sigma : \tau \mapsto \begin{cases} 1 & \text{if } \tau = \sigma \\ 0 & \text{otherwise} \end{cases}, \quad \alpha_\sigma \in \mathbb{Z}/2\mathbb{Z}$$

Ex: Let $C = [v_0, v_1] + [v_1, v_2] + [v_2, v_3] + [v_3, v_4] + [v_4, v_5] + [v_5, v_0]$.

$$\partial_1 C = \partial_1[v_0, v_1] + \partial_1[v_1, v_2] + \partial_1[v_2, v_3] + \partial_1[v_3, v_4] + \partial_1[v_4, v_5] + \partial_1[v_5, v_0]$$
$$= [v_0] + \cancel{[v_1] + [v_1]} + \cancel{[v_2] + [v_2]} + \cancel{[v_3] + [v_3]} + \cancel{[v_4] + [v_4]} + \cancel{[v_5] + [v_5]} + [v_0]$$
$$= [v_0] + [v_0] = 0.$$

Def: A d-*cycle* is a d-chain C s.t. $\partial_d C = 0$.

Pb: Different cycles can represent the same hole.



The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

Q: Prove it.

The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

$$C \sim C' \Longleftrightarrow C + C' \in \mathrm{im}(\partial_{d+1})$$

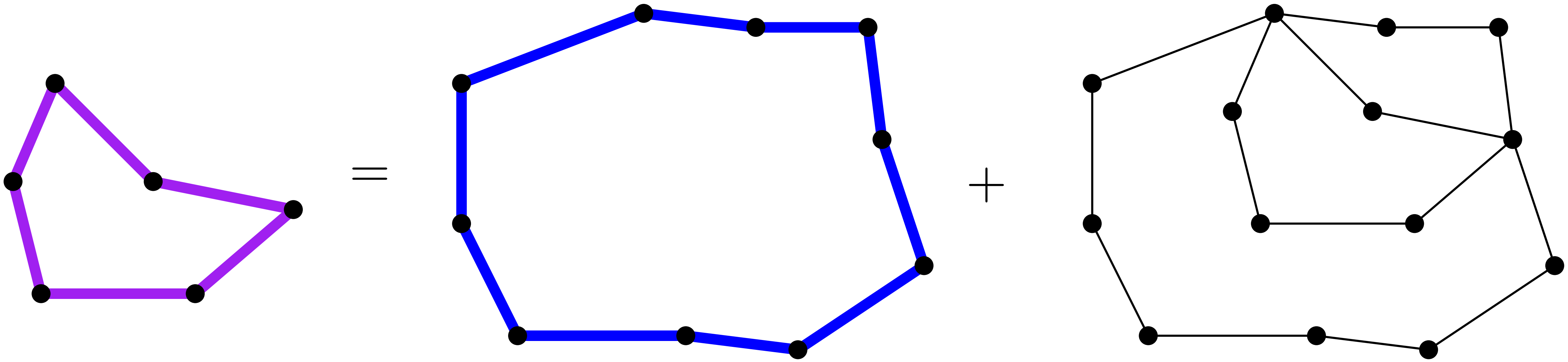
The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

$$C \sim C' \iff C + C' \in \mathrm{im}(\partial_{d+1})$$



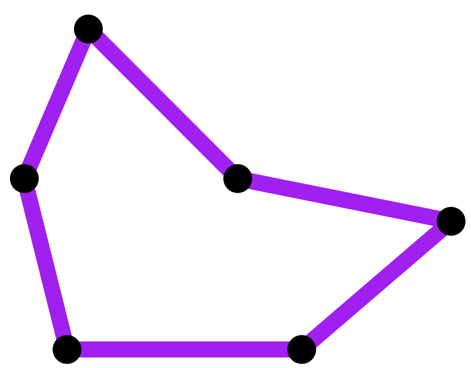
The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

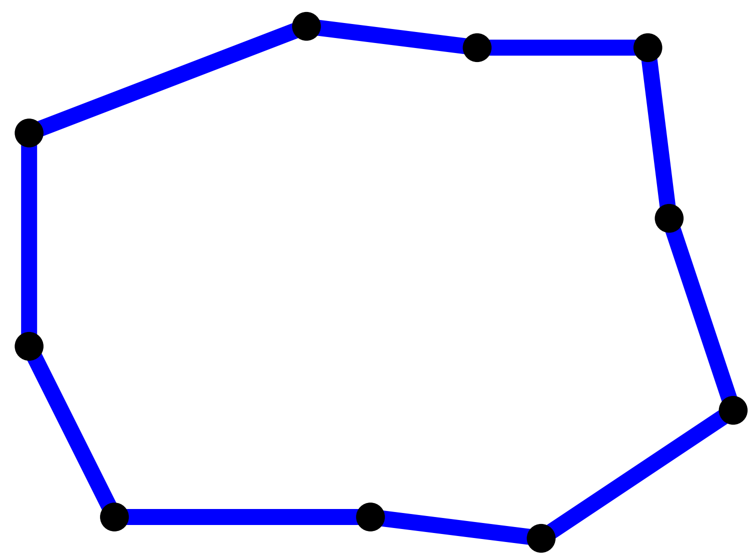
Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

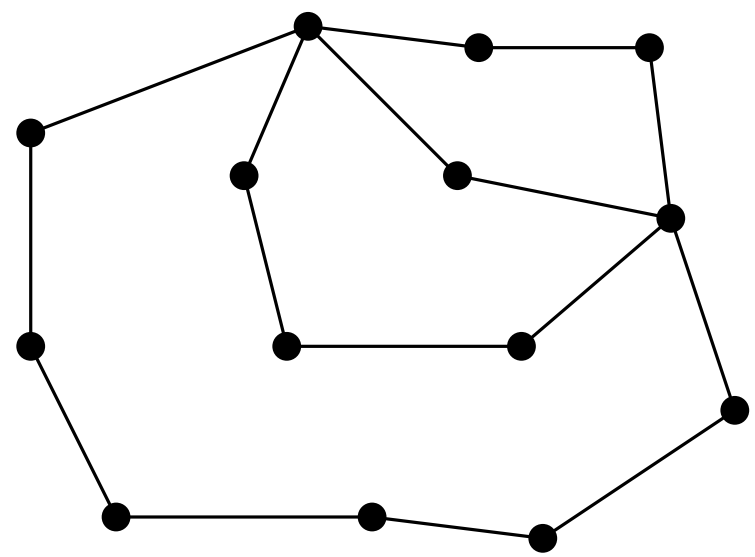
$$C \sim C' \iff C + C' \in \mathrm{im}(\partial_{d+1})$$



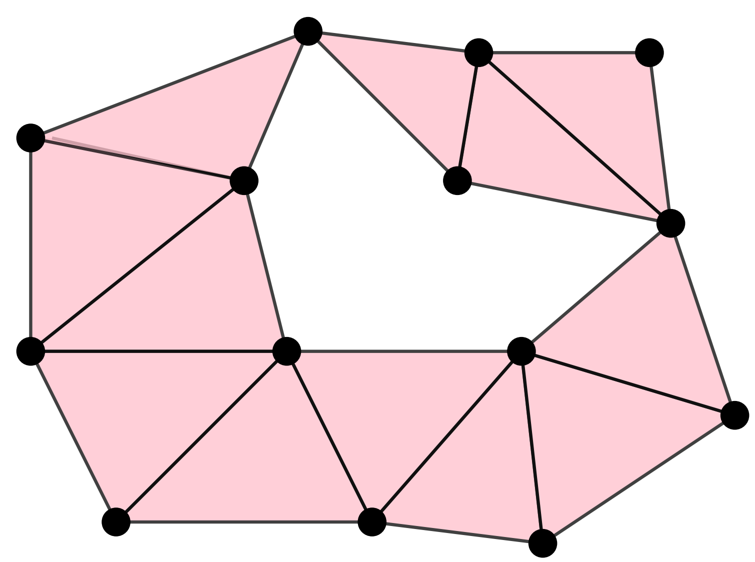
$=$



$+$



$= \partial($



$)$

The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0$.

Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

$$C \sim C' \iff C + C' \in \mathrm{im}(\partial_{d+1})$$

Prop: The space of d-chains $C_d(K)$ is a vector space with basis

$$\{\sigma \in K : \dim(\sigma) = d\}.$$

The space of d-cycles $Z_d(K)$ is a linear subspace of $C_d(K)$.

The boundary operator $\partial_{d+1} : C_{d+1}(K) \to C_d(K)$ is linear and $\mathrm{im}(\partial_{d+1})$ is a linear subspace of $C_d(K)$.

Def: Given a vector space V, and a linear subspace $W \subseteq V$, their quotient is the vector space: $V/W := \{[v] = \{v + w : w \in W\} : v \in V\}$.

Remark: v_1 and v_2 are mapped to the same element of V/W iif $v_1 + v_2 \in W$.

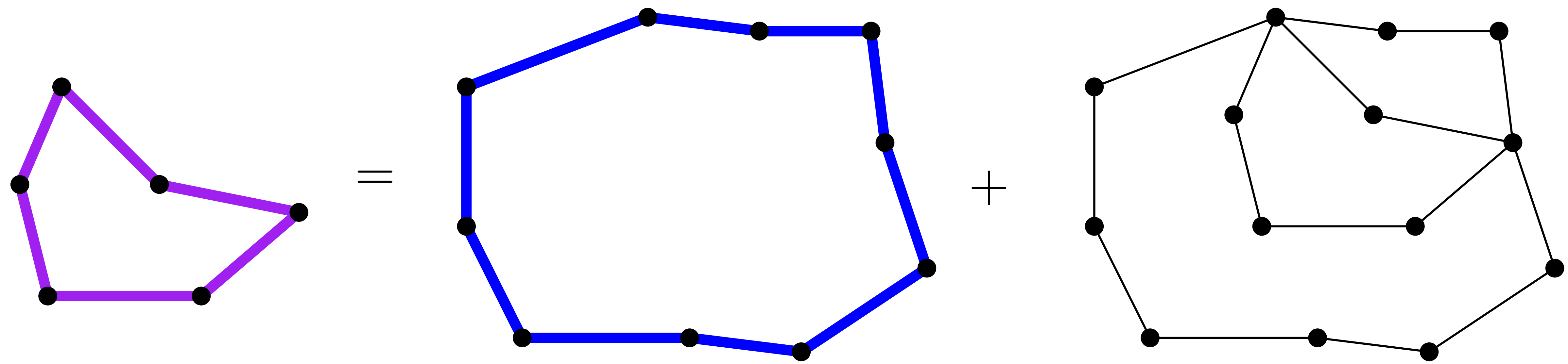
The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

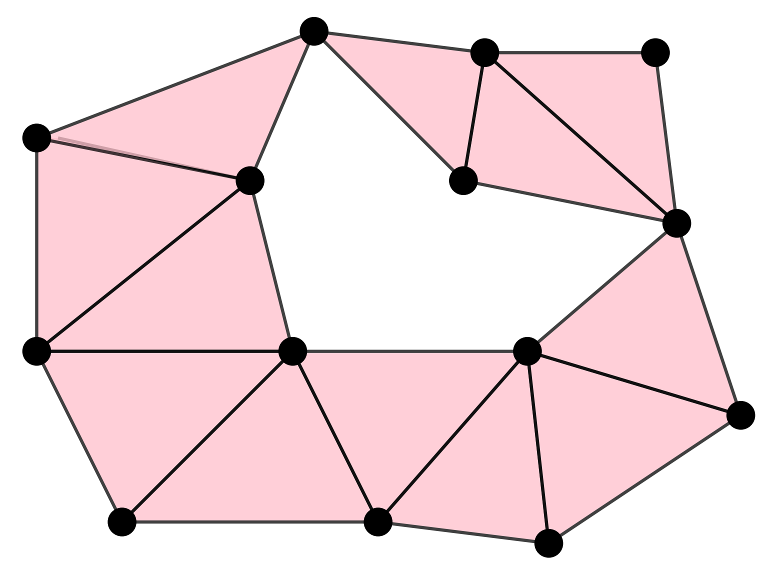
Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\operatorname{im}(\partial_{d+1})$':

$$C \sim C' \iff C + C' \in \operatorname{im}(\partial_{d+1})$$



$$H_d = Z_d / \operatorname{im}(\partial_{d+1})$$

$= \partial($  $)$

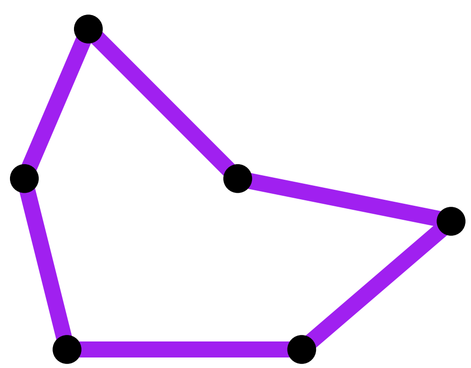
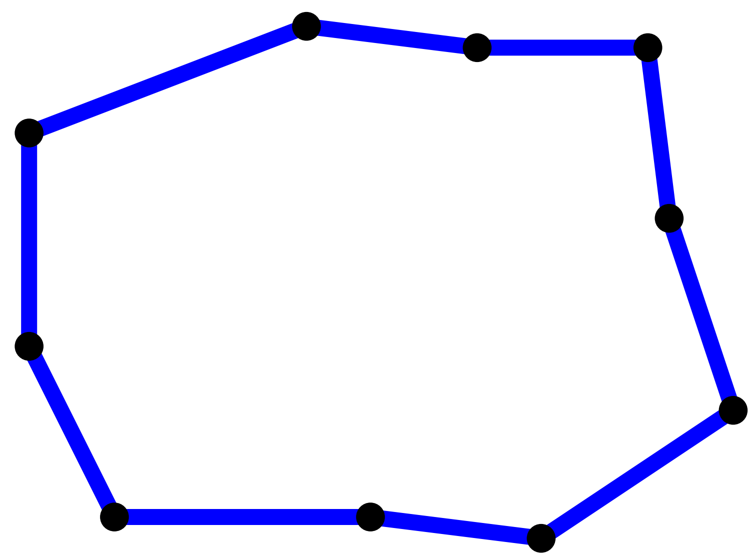
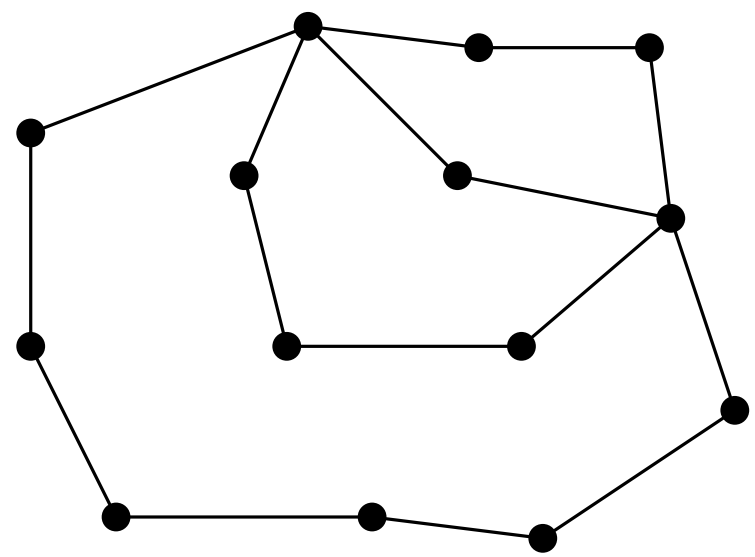
The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0$.

Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

$$C \sim C' \iff C + C' \in \mathrm{im}(\partial_{d+1})$$

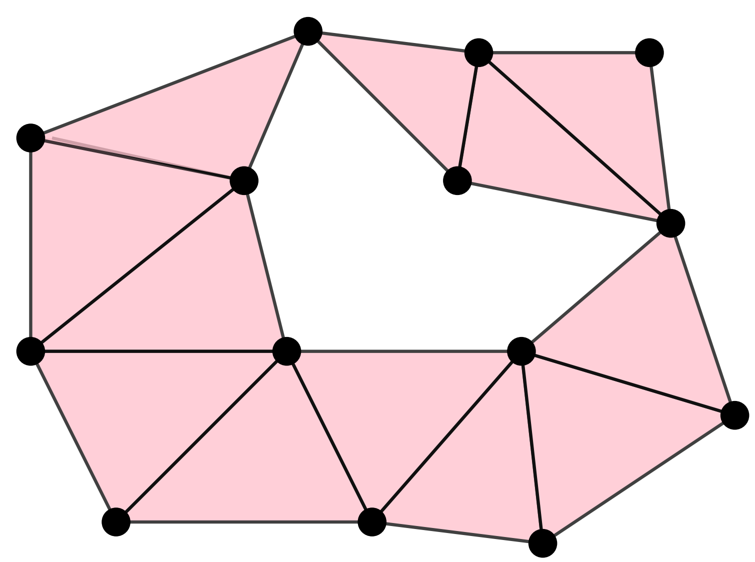
 $=$  $+$ 

$$H_d = Z_d / \mathrm{im}(\partial_{d+1})$$

group of d-cycles

group of 'cycles minus boundaries'

group of d-boundaries

$= \partial($  $)$

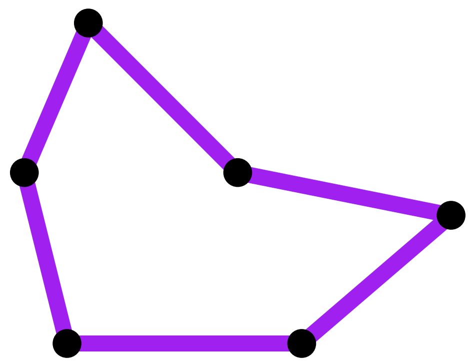
The homology groups

Lemma: $\partial_{n-1} \circ \partial_n = 0.$

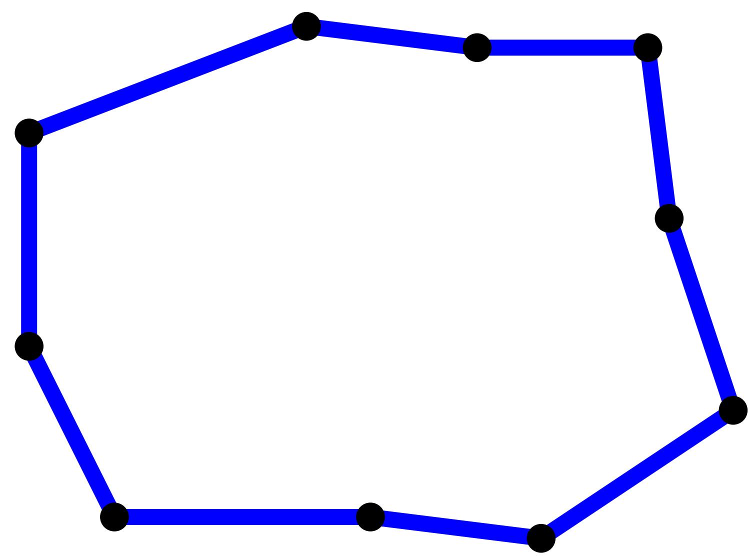
Q: Prove it.

Def: Two d-cycles are homologous if 'their combination is in $\mathrm{im}(\partial_{d+1})$':

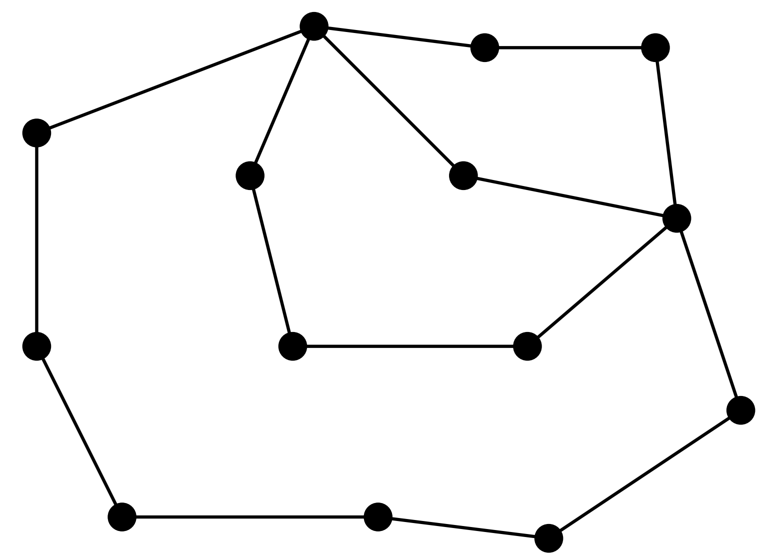
$$C \sim C' \iff C + C' \in \mathrm{im}(\partial_{d+1})$$



$=$



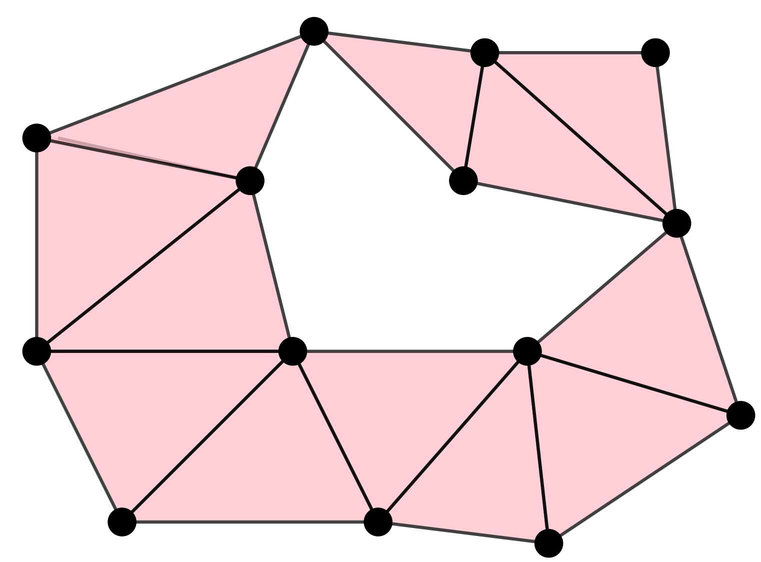
$+$



$$H_d = \{[C] \,:\, C \in Z_d(K)\}$$

where

$$[C] = \{C' \,:\, C \sim C'\}$$

$= \partial($  $)$

The homology groups

H_d is a vector space in which each element is an equivalence class of cycles associated to the same hole.

Def: The dimension of H_d is called the *Betti number* β_d.

Minimum number of (classes of) cyles needed to create a basis, i.e., to be able to write *any* cycle as a linear combination of cycles in the basis.

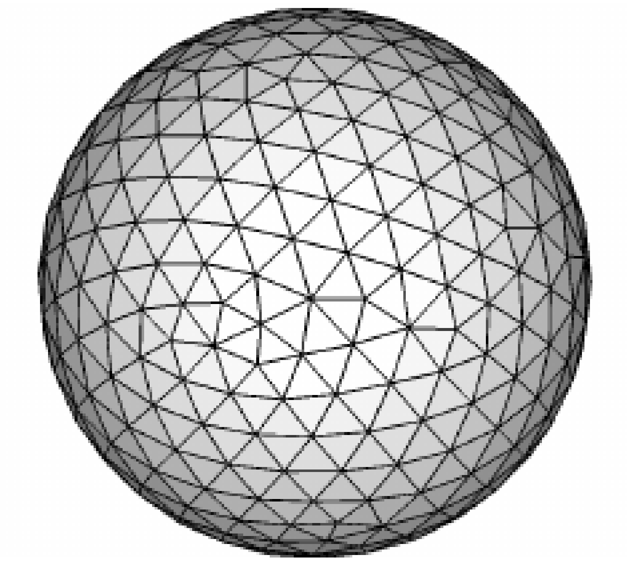
β_0 counts the connected components, β_1 counts the loops, β_2 counts the cavities, and so on...

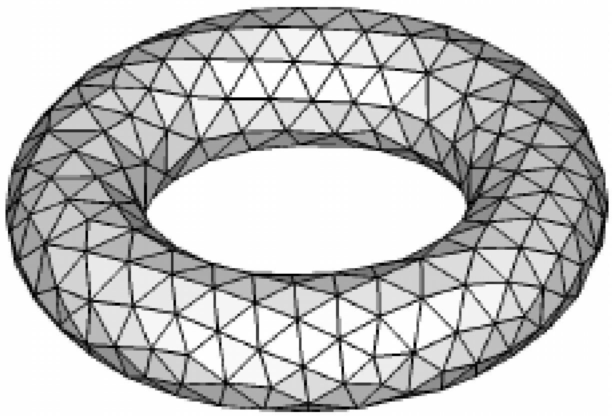
The homology groups

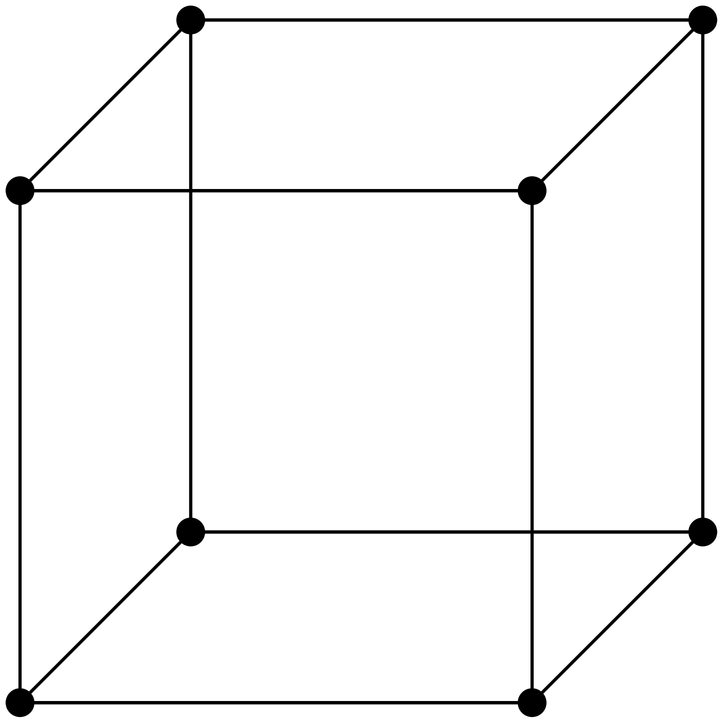
H_d is a vector space in which each element is an equivalence class of cycles associated to the same hole.

Def: The dimension of H_d is called the *Betti number* β_d.

Q: What are the Betti numbers of:





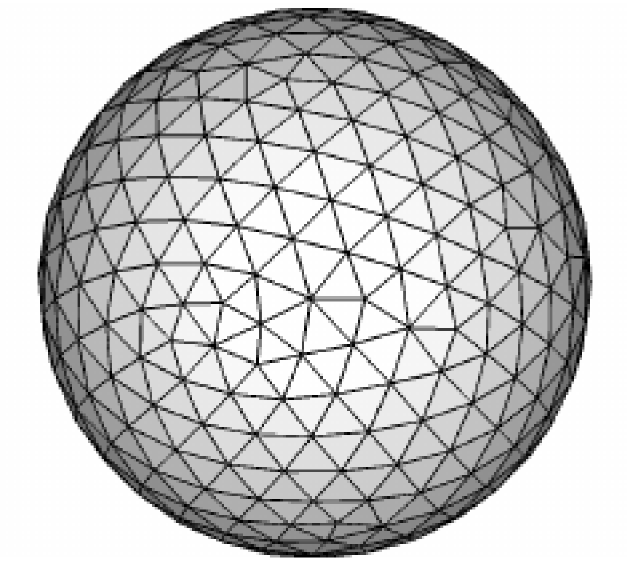


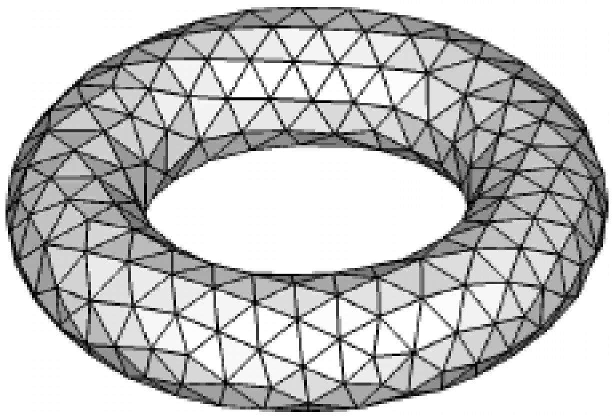
The homology groups

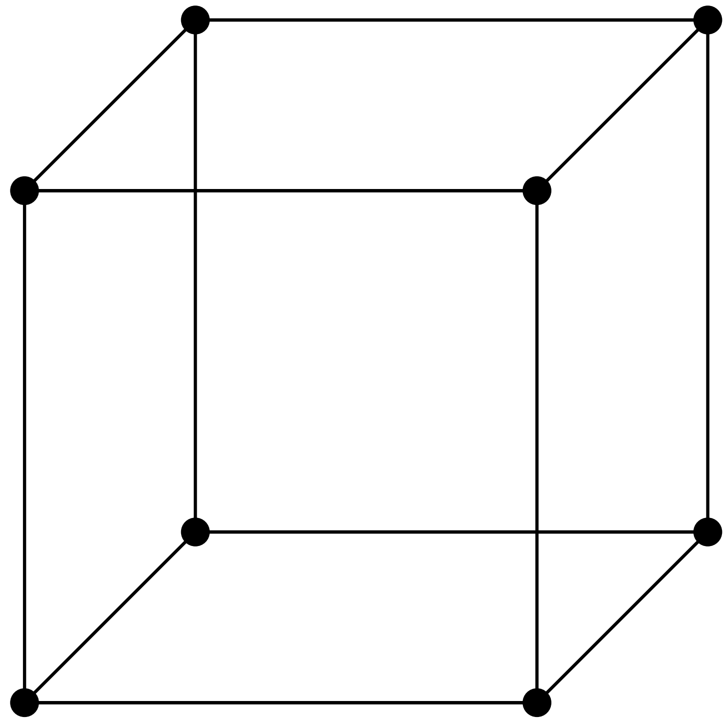
H_d is a vector space in which each element is an equivalence class of cycles associated to the same hole.

Def: The dimension of H_d is called the *Betti number* β_d.

Q: What are the Betti numbers of:







The whole point of homology groups and Betti numbers is that they satisfy:

$$H_d(X) \neq H_d(Y) \Longrightarrow X, Y \text{ are not homotopy equivalent.}$$

Summary

In this class, I introduced the basic bricks of Topological Data Analysis.

We have seen how to encode data sets as topological spaces using combinatorial models called simplicial complexes.

We have seen simplicial complex constructions, e.g., Mapper, that are based on the Nerve Theorem which guarantees that the topology is correct.

We have seen how to quantify topology in simplicial complexes with homology groups and Betti numbers.

Next week, we will see an extension of homology groups, called persistent homology, that allows to create richer descriptors for data science, called persistence diagrams, out of simplicial complexes.