Thesis

walch

February 2023

CONTENTS

INTRODUCTION

0.1 Structure

This thesis involves theoretical and computative work. Chapter 1 to 3 provide an introduction to the concepts needed for Chapter 4. Chapter 1 starts with an introduction to topological data analysis and explains the complexification of data. Complexification adds the connectivity information needed to extract topological invariants from the data in order to describe their shape. Theoretical justification for these invariants is guaranteed via the Nerve Theorems presented in Section 1.7.

Chapter 2 introduces persistent homology as one approach to analyze the topological shape of data, others include the Mapper algorithm [SMC07] and discrete Morse theory ... which, however, will not be used explicitly within this work.

Chapter 3 introduces neural networks and deep learning. The main focus of Chapter 3 is on self-supervised learning and in particular Autoencoders.

Locality runs like a red thread through this thesis, with its first appearance in Section 2.7, where the Mayer-Vietoris blowup complex is presented.maybe use some more words here include overview figure of chapters and locality This is continued in Chapter 2, where spectral sequences and local persistent homology algorithms are presented. In Chapter 3, it is proven that the Autoencoder architecture (definition in Chapter 3) induces a cover of the data on the input space (definition). In Chapter 4, synergies between persistent homology and self-supervised learning with Autoencoders are investigated. The starting point herein marks the 'topological Autoencoder' [Moo+21]. It is proven that via this choice of architecture, the interleaving distance (definition) between persistence modules in the input, latent and output space (definition in chapter 3) of dimension higher than 1 cannot be controlled. Starting from [TP21] we propose to not regulate the global topology of the input, latent and output space via a loss term added to the training loss (definition Chapter 3), but instead to bound the interleaving between the locally representative subsets of the respective covers

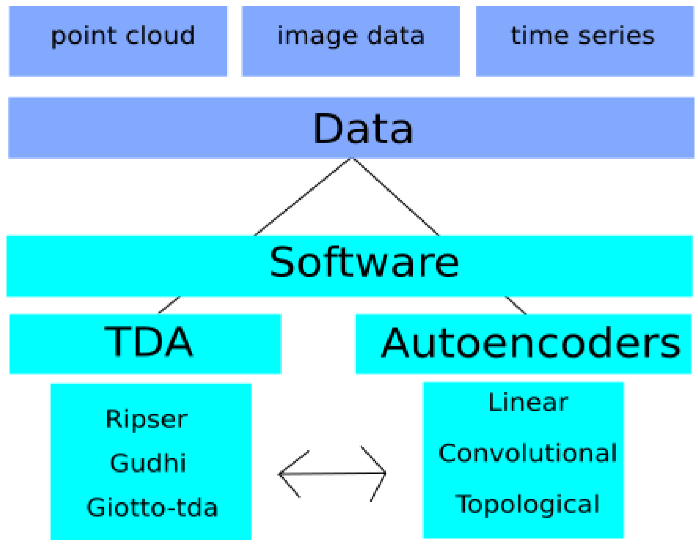


Figure 1: Overview of the data and software used for computations within this work.

induced by the network architecture. Finally, we also propose an algorithm for the implementation of this idea.

0.2 Computations

From the beginning of Chapter 1, the theoretical concepts discussed are illustrated with computational examples. In the following, three data sets and the software used to calculate the examples are presented. The data sets may seem complex for the basic concepts from Chapters 1 and 2, but they were also chosen with Chapter 3 in mind. Neural networks require a certain data size on the one hand and a certain complexity of the data on the other hand to be reasonably applied. In contrast to that, the dimension of the data and its complexity are the two most difficult factors in topological data analysis. The aim of this work is to contribute to resolving this friction. Where necessary, simpler toy datasets are used to give an idea of the concepts presented. However, on the basis of these three datasets, which can be considered emblematic for the use cases of self-supervised learning, the difficulties within a synergy between persistent homology and deep learning shall also be illustrated as shown schematically within Figure 1. This is also done to motivate the local approach taken within Chapter 4 and its main results theorem algorithm.

0.2.1 Data

In the following, we will consider point cloud data, image data and time series data.

Point Cloud Data

As an example for point cloud data, we take the 4-, 8- and 16-dimensional random point cloud data from [Ott+17]. This data has no particular physical meaning, but was used as a benchmark data set for the comparative study on TDA methods presented in [Ott+17]. Figure 2 and Figure 3 show the first three dimensions of the 16- and the 8-dimensional random point cloud. The first one consisting of 50 data points, whereas the latter is much denser with 1000 data points in total.

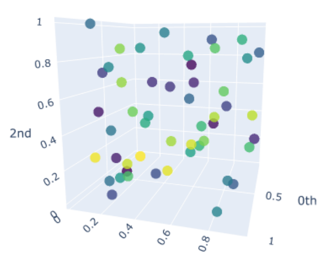


Figure 2: $16D$ point cloud

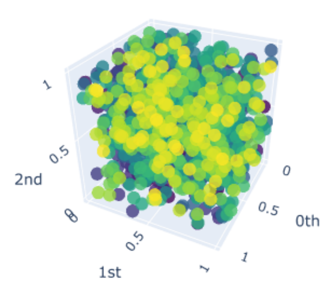


Figure 3: $8D$ point cloud

Image Data

For image data input (and cubical persistence) we use MNIST ([LCB]), a database of handwritten digits commonly used to benchmark image processing systems and algorithms. In addition to that, we use Fashion-MNIST ([XRV17]), a dataset of Zalando's article images. Both datasets are integrated by default in many machine learning libraries, e.g. pytorch, keras or tensorflow. Figure 4 and Figure 5 show two examples thereof.

Time Series Data

As an example for time series data, we take the DeepVALVE data set. It is an industrial data set containing a sequence of random closing and opening events of an industrial valve. In total, there are five different labels: 'start', 'lose', 'linear', 'stuck' and 'end'. These labels were set manually via markers on the current-time measurement of the engine driving the valve. A part of the DeepVALVE data set is shown in Figure 6

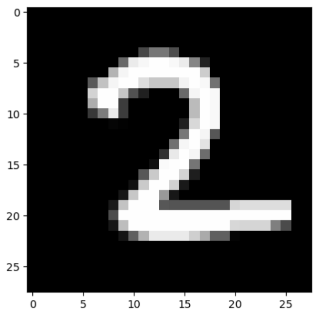


Figure 4: Example of MNIST

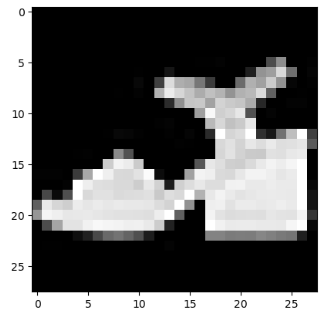


Figure 5: Example of FashionMNIST

0.2.2 TDA Software

Extensive lists (with discussions) of open source software available for topological data analysis and (persistent) (co)homology related calculations can be found in [Ott+17], [Ada23] and [Bub23]. In the following, we will describe **ripser**, **gudhi** and **giotto-tda** more thoroughly, as these will be the implementations used for the illustrative examples in the following. Furthermore, with regard to the machine and deep learning methods presented in Chapter 3, we will focus on the python implementations.

RIPSER

Ripser [Bau21] was developed by Ulrich Bauer and started as a C++ code for the computation of Vietoris-Rips barcodes. With regard to efficiency, ripser has been state of the art until adaptations of the algorithm for parallel computing evolved with ripser++, [ZXW20]. With ripser.py [TSB18] there exists also a python binding and with ripserer.jl [Čuf20] a julia binding for ripser. These also include some plotting routines. Ripser does not offer the wealth of additional topological features that the other libraries do. To cite Ulrich Bauer: 'It can do just this one thing, but does it extremely well.'

GUDHI

Gudhi uses the algorithms of [DFW12] and [SV09] to compute persistent cohomology, in combination with the compressed annotation matrix implementation of [BDM13]. The latter guarantees improved time and memory performance via separation of the representation of the simplicial complex and the representation of the cohomology groups. Gudhi can calculate simplicial and cubical cohomology and thus can also possibly be used for greyscale image analysis. It offers the caclulation of large variety of complexes from either point clouds, distance matrices or correlation matrices. Unfortunately, not all of these functionalities are available in python (for example the Čech complex computation). Persistence intervals can be visualized via persistence diagrams or barcodes. More

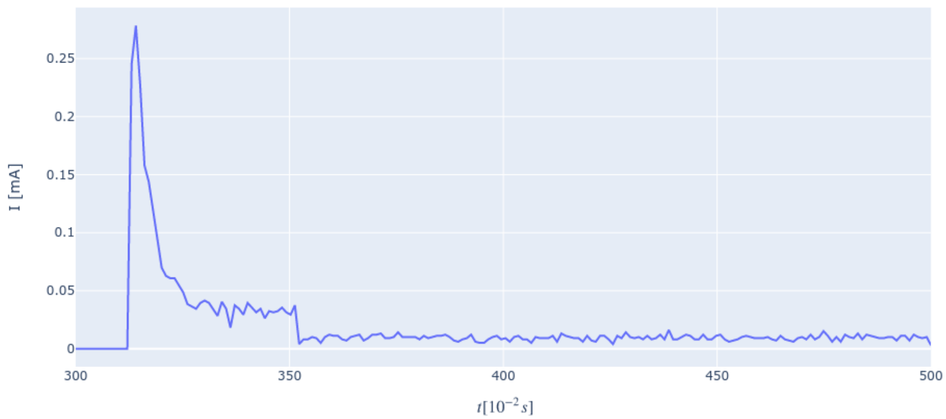


Figure 6: Part of the DeepValve time series data set. The blue line represents the electrical current per time.

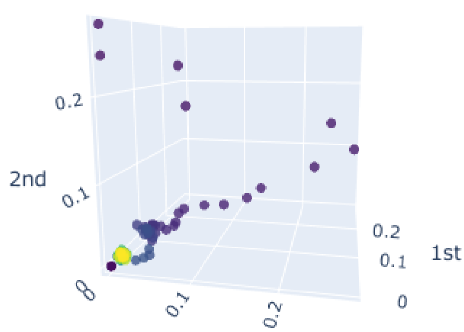


Figure 7: First 3D of Takens embedding of the DeepVALVE data segment shown above.

elaborate visualizations such as e.g. the entropy summary function [AGS20] or persistence landscapes [Bub20] are listed in Table 2.
Gudhi also provides the calculation of metrics (bottleneck and Wasserstein distance) between persistence diagrams, diagram kernels (persistence Fisher kernel [LY18], heat kernel [Rei+14], persistent weighted Gaussian kernel [KFH16] and sliced Wasserstein kernel [CCO17]). Additional diagram features and vectorizations include persistence entropy [AGR17], topological vector [COO15] and complex polynomial [DF15]. For machine learning and deep learning, two libraries are related to gudhi: ATOL (Automatic Topologically-Oriented Learn-

Language	**PH Algorithm**
C++/Python	dual [SMV11]
	twist [CK11]
Coefficient Field	**Homology**
$\mathbb{Z}/p\mathbb{Z}$	simplicial
Input	**Complexes**
point cloud	Vietoris-Rips
distance matrix	
Visualizations	**Additional Features**
persistence diagrams	-
lifetime of generators	
representative (co)cycles	

Table 1: Overview of RIPSER.

ing) [Roy+21] and PersLay [Car+19] (a simple and versatile neural network layer for persistence diagrams).

GIOTTO-TDA

Giotto-tda [Tau+20] is based on a python backend of ripser and gudhi. It also incorporates directed simplicial cohomology via a python binding for flagser [Lue+19], developed as a sibling project. Giotto-tda combines gudhi's edge collapse algorithm [Pri] with ripser and thus improved on the state-of-the-art ([BP20]) runtimes for the computation of Vietoris-Rips barcodes. Giotto-tda was developed to integrate computational topological data analysis into machine learning and deep learning and as such includes parallelizable computations across batches of data. One of the main advantages is the inbuilt preprocessing, which enables the direct integration not only of point cloud data or distance matrices, but also of graphs, image data and time series as input data. Another benefit is the possibility to visualize data, persistence diagrams and persistence features such as Betti curves, persistence landscapes and persistence images. Compared to gudhi, giotto-tda offers less persistence kernels, but as an additional feature, the L^p distance between representations. Also, complexes can be calculated only implicitly on the data as a step involved in calculating persistence (co)homology.

0.2.3 Machine Learning Software

For the implementation of the neural networks, PyTorch [Pas+19] was used. The specific Autoencoder architectures will be described in Chapter 3.

Language	**PH Algorithm**
C++/Python	dual via simplicial maps [DFW12] multifield [BM19]
Coefficient Field	**Homology**
$\mathbb{Z}/p\mathbb{Z}$	simplicial/cubical
Input	**Complexes** (C++)
point cloud distance matrix correlation matrix	Vietoris-Rips Čech Witness α Cover Cubical Tangential Delaunay
Visualizations	**Additional Features**
nerves (e.g. via KeplerMapper) persistence diagram persistence density [CD19] barcode entropy summary function Betti curve persistence landscapes silhouette [Cha+13] persistence image [Ada+15]	topological vector persistence kernels complex polynomial metrics persistence entropy ATOL PersLay

Table 2: Overview of GUDHI.

0.3 Outline of the Main Results

theoretical theorems etc

Language	**PH Algorithm**
Python	Ripser and GUDHI backend
	Flagser backend via pyflagser
Coefficient Field	**Homology**
$\mathbb{Z}/p\mathbb{Z}$	undirected simplicial
	directed simplicial [Lue+19]
	cubical
Input	**Complexes**
point cloud	Vietoris-Rips
distance matrix	Čech
image data	α (weak) [Gio]
time series	cubical
Visualizations	**Additional Features**
persistence diagrams	metrics
Betti curves	persistence kernels
silhouettes	vector representations
heat representation [Rei+14]	curve features
persistence image	complex polynomial
persistence landscapes	amplitude [Tau+20]
	persistence entropy

Table 3: Overview of GIOTTO-TDA.

ABBREVIATIONS

General

	Abbreviation
Field	$\mathbb{F}$
Manifold	$\mathcal{M}$
Metric Space	(M, d_M)
$\mathbb{R} \cup \{-\infty, \infty\}$	$\overline{\mathbb{R}}$

Categories

	Abbreviation
Category of chain complexes of Abelian groups	**Ch**
Category of simplicial complexes and simplicial maps	**Simp**
Category of topological spaces	**Top**
Category of vector spaces and linear maps over a given field $\mathbb{F}$	**Vect**
Category of persistence modules	**Pers**
Category of observable persistence modules	**Obs**

BASIC DEFINITIONS

Term	Definition
Point Cloud // Data Set $\mathcal{S}$	finite metric space, most commonly $\mathcal{S} \subset \mathbb{R}^d$
unit ball E^d in $\mathbb{R}^d$	$E^d = \{x \in \mathbb{R}^d \| \ \|x\| \leq 1\}$
unit sphere S^{d-1} in $\mathbb{R}^d$	$S^{d-1} = \{x \in \mathbb{R}^d \| \ \|x\| = 1\}$, $d \geq 1$
closed ball $B[x, r]$	for a metric space (M, d_M), centered at $x \in M$ with radius r: $\{x' \in X \mid d_M(x, x') \leq r\}$
open ball $B(x, r)$	for a metric space (M, d_M), centered at $x \in M$ with radius r: $\{x' \in X \mid d_M(x, x') < r\}$
d-ball	d-dimensional ball

CHAPTER 1

TOPOLOGY FROM DATA

On a dit, écrivais je (ou à peu près) dans une préface, que la géometrie est l'art de bien raisonner sur des figures mal faites. Oui, sans doute, mais à une condition. Les proportions de ces fiugres peuvent être grossièrement altérées, mais leurs éléments ne doivent pas être transposées et ils doivent conserver leur situation relative.

Henri Poincaré [Poi21]

Abstract

The basic TDA pipeline is presented. Starting with its input step, it will be discussed how structure can be introduced into point cloud data. Simplicial complexes form the basis for the homological computations on the sample data, which are continued in Chapter 2. The Nerve Theorem forms the central theoretical guarantee for the construction of certain simplicial complexes (and approximations thereof) on point clouds.

Contents

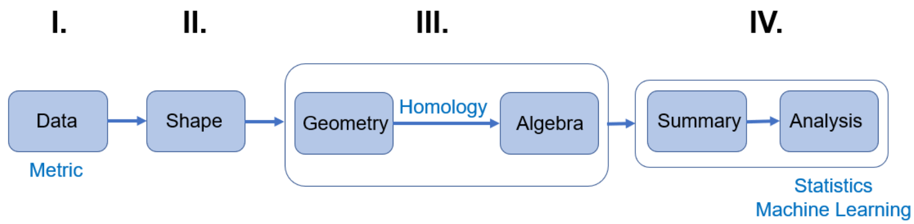


Figure 1.1: TDA Pipeline as described in [CM17].

1.1 The TDA Pipeline

Topological data analysis (TDA) emerged as a synergetic field between applied (algebraic) topology and computational geometry driven by pioneering works of Edelsbrunner [ELZ00], Zomorodian and Carlsson [ZC05; Car09]. The aim of TDA is to provide mathematical, statistical and algorithmic methods in order to qualitatively and quantitatively analyze the structure of data. Following [CM17], the basic TDA pipeline consists of four steps, visualized in Figure 1.1. The TDA pipeline starts with a data input. Data can come from a variety of sources and appear in different forms: For example as images, sound and picture recordings, time series measurements or text modules. In general, data is given as a set of points. Most commonly, these points are elements of the d-dimensional Euclidian space $\mathbb{R}^d$, forming a finite subspace thereof with the inherited metric. Such data is called **point cloud** data. The addition of topological structure introduces connectivity to the data space and is indispensable to study the structure thereof.

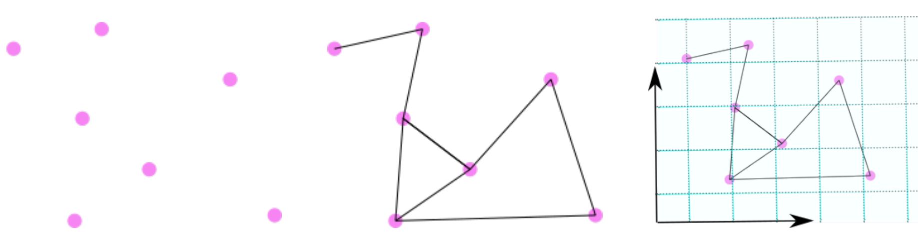


Figure 1.2: From left to right: Space, topological space and metric space.

With this proceeds the next step called 'shape' in Figure 1.1, where the data set $\mathcal{S}$ is partitioned (combinatorically) into subsets and maps between them in order to induce the structure of a non-discrete topological space. Subsequently, the geometric and topological properties of this space are extracted. The methods to do so have to be stable with respect to perturbations and noise in the data. Lastly, the extracted topological and geometric features must be summarized in a way that they can be proceeded with statistical or machine learning methods.

1.2 Outline

This Chapter 1 starts with a basic introduction to homotopy and homology. It then focuses on the transition of **step I. to II.** within the pipeline in Figure 1.1. Section 1.4 to Section 1.7 explain how topological spaces can be constructed on a given point cloud $\mathcal{S}$. Section 1.8 gives examples for (geometric) simplicial complexes, most of which are also implemented within the tda packages presented in the Introduction. In Section 1.9.1, simplicial homology is introduced, which forms the basis for the concepts illustrated in the subsequent Chapter 2.

1.3 Homotopy and Homology

Intuitively, algebraic topology distinguishes spaces by counting the occurrences of geometric patterns. Doing this counting directly is not feasible, hence *equivalence classes* of occurrences of patterns under certain equivalence relations are counted.

1.3.1 Homotopy

Homotopy is a way to determine whether two given spaces have 'the same shape' in a much coarser sense than homeomorphism [Hat02]. In the following, let X and Y be two topological spaces.

Definition 1.3.1. *If these topological spaces X and Y have distinguished base points x_0 and y_0, we call them **based**. Based (or base-preserving) maps $f : X \to Y$ are maps for which $f(x_0) = y_0$.*

Definition 1.3.2 (Homotopic Maps)**.** *Given two maps $f, g : X \to Y$, f and g are called homotopic $f \simeq g$, if there is a continuous map $H : X \times [0, 1] \to Y$ so that $H(x, 0) = f(x)$ and $H(x, 1) = g(x)$ for all $x \in X$.*

Homotopy is an equivalence relation on the set of continuous functions between X and Y. The set of equivalence classes (**homotopy classes**) with regard to this relation is a discrete invariant giving a high level description of the set of maps from X to Y. For based spaces, homotopies fullfill $H(x_0, t) = y_0$ for all t. Based homotopy is an equivalence relation on the set of based maps from X to Y.

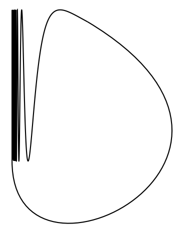


Figure 1.3: Warsaw circle.

Homotopy Group

Let X be the based n-sphere S^n. Then the homotopy classes of base-preserving maps of S^n to Y exhibit a group structure with the group operation given by gluing of two spheres at their basepoint. The group $\pi_n(Y, y_0)$ is called the n-th **homotopy group** of Y. For $n \geq 2$ this group is abelian. Intuitively, the homotopy groups count n-dimensional holes in Y. A space Y is said to have an n-dimensional hole if and only if it contains an n-dimensional sphere that cannot be shrunk to a single point. This holds if and only if the map $S^n \to Y$ is not homotopic to the constant function and this is the case if and only if $\pi_n(Y, y_0) \ncong 0$.

Homotopy Equivalence

Definition 1.3.3 (Homotopy Equivalence)**.** *Two topological spaces X and Y are called homotopy equivalent (or spaces of the same* ***homotopy type****) if there exist continuous maps $f : X \to Y$ and $g : Y \to X$ such that gf is homotopic to the identity map in Y and fg is homotopic to the identity map in X.*

If in the above definition, the base points of x_0 and y_0 of X and Y are preserved respectivley, X and Y are called **based homotopy equivalent**. If X and Y are based homotopy equivalent then all their homotopy groups are isomorphic (Proposition 2.4 in [Car14]). However, converseley, a map inducing isomorphisms on all homotopy groups does not induce a homotopy equivalence but a weaker connection called **weak homotopy equivalence**.

Definition 1.3.4 (Weak Homotopy Equivalence)**.** *A weak homotopy equivalence is a map $f : X \to Y$ that induces isomorphisms $\pi_n(X, x_0) \to \pi_n(Y, f(x_0))$ for all $n \geq 0$ and choices of basepoint x_0.*

The *Warsaw circle* and the point constitute a pair of spaces that are weakly homotopy equivalent but not homotopy equivalent.

Example 1.3.1 (Warsaw Circle)**.** *The Warsaw circle, as depicted in Figure 1.3, is a subset of $\mathbb{R}^2$ defined as:*

$$S_W = \{(x, sin(\pi/x)) \mid 0 < x \leq 1\} \cup \{(0, y) \mid -1 \leq y \leq 1\} \cup C. \qquad (1.1)$$

Herein C is a continuous curve in the plane connecting $(1,0)$ to $(0,0)$ without further intersection of the Warsaw circle. As the Warsaw circle is path connected, it has trivial homotopy groups such as the point. However, these are not homotopy equivalent.

Proof. To prove this, one must show that the Warsaw circle is not contractible. This is done by proving that a continuous map $f : S_W \to S_W/\sim\ \cong S^1$ is not null homotopic, where $S_W/\sim$ is the quotient space obtained after collapsing the interval down to a point. E contrario, assuming f is null homotopic, we can lift it to a map $\hat{f} : S_W \to \mathbb{R}$.

$$\begin{tikzcd} & \mathbb{R} \arrow[d, "g"] \\ S_W \arrow[r, "f"] \arrow[ur, dashed, "\hat{f}"] & S^1 \end{tikzcd}$$

Except for the interval piece, f is injective, which thus also applies to the lift. Thus restricting the domain of $\hat{f}$ to $S_W/\sim\ \cong S^1$ implies an injective map $S_W/\sim\ \simeq S^1 \to \mathbb{R}$, which does not exist by the Borsuk-Ulam theorem. □

Although the definition of homotopy groups appears quite easy, homotopy groups are very difficult to compute. Thus they are not used in practice as an invariant for algebraic topology. Instead, one uses **homology groups**.

1.3.2 Homology

Most generally speaking, homology defines the association of a sequence of algebraic objects such as Abelian groups or modules with other mathematical objects such as topological spaces. Homology was initially defined for combinatorial structures called **simplicial complexes** [Section 1.4.3], the related theory is now refered to as **simplicial homology** [Section 1.9.1]. Samuel Eilenberg proved that the definition of homology groups can be extended to all spaces [ES45]. The five **Eilenberg-Steenrod axioms** [ES45] guarantee properties that all homology theories of topological spaces have in common. Thence, homology can be defined abstractly as a sequence of functors satisfying the Eilenberg-Steenrod axioms. By the first Eilenberg-Steenrod axiom, homology is coarser than homotopy. Bringing this down to the level of homology groups $H_p(X)$, this means that if two topological spaces X and Y are homotopy equivalent, then the homology groups are isomorphic, but the converse is not true.

Example 1.3.2. *The torus $T^2 = S^1 \times S^1$ and the wedge sum $S^1 \vee S^1 \vee S^2$ have isomorphic homology groups but differing homotopy groups, already for π_1 and π_2.*

Referring to the TDA pipeline in Figure 1.1 and the purpose of this work as outlined in the introduction, we will naturally have to deal with simplicial complexes built on finite metric spaces and thus simplicial homology. Simplicial complexes are practicable because homology is calculable on them: It boils down to linear algebra as elaborated in Section 1.9.1. This view yet is quite general: Many topological spaces of interest are either explicitly simplicial complexes or at least homotopy equivalent to such [Car14].

1.4 Constructing Complexes on Data

A complex structure on a data set $\mathcal{S}$ is the power set of certain subsets of $\mathcal{D}$ chosen in a way that allows defining 'gluing' maps between them. This defines a non-trivial topological structure on $\mathcal{S}$ in a combinatorial way. This structure should approximate the structure of the underlying geometric space that the point cloud data is supposed to be drawn from. In the following, CW complexes, Δ-complexes and simplicial complexes will be presented. CW complexes herein denote the most general structure, where the 'gluing' between the subsets called *cells* is defined inductively. Some special CW complexes are Δ-complexes, where the relevant subsets are called *simplices* and the maps between them are restricted. Simplicial complexes are Δ-complexes where simplices must intersect in a specific way.

1.4.1 CW Complexes

A nonempty CW complex X is constructed as an unbased Hausdorff space via a sequence of unbased subspaces called *skeleta*

$$X^0 \subseteq X^1 \subseteq ... \subseteq X^n \subseteq X^{n+1}..., \tag{1.2}$$

whose union is X. For this construction, we need the following definitions of *free maps* and *adjunction spaces*.

Definition 1.4.1 (Free Map)**.** *A free map is a continuous function of unbased spaces or of based spaces without the requirement that the base point is preserved.*

Definition 1.4.2 (Adjunction Space)**.** *Let X and Y be unbased topological spaces, with $W \subseteq X$ and a free map $\phi : W \to Y$. In the disjoint union $Y \amalg X$ introduce the equivalence relation $w \sim \phi(w)$ for all $w \in W$. The* ***adjunction space*** *$Y \cup_\phi X$ is given by the resulting identification space.*

The inductive construction of CW complexes involves the following steps [Hat02; Ark10]:

- Start with $\mathcal{S}$ whose elements are called 0-cells or *vertices*.
- Assume that X^{n-1} is defined for $n \geq 1$ and obtain X^n from X^{n-1} by attaching n-cells via *attaching maps* $\phi_\beta : S^{n-1}_\beta \to X^{n-1}$, where for some index set B and each $\beta \in B$, S^{n-1}_β denotes the $(n-1)$-sphere S^{n-1} centered at β. The ϕ_β determine a free (i.e. non-basepoint preserving) map $\phi : \amalg_{\beta \in B} S^{n-1}_\beta \to X^{n-1}$. Then the n-skeleton X^n is defined as

$$X^n = X^{n-1} \cup_\phi \amalg_{\beta \in B} E^n_\beta, \tag{1.3}$$

where $E^n_\beta = E^n$ is the unit ball in Euclidian n-space $\mathbb{R}^n$ centered around β, with

$$E^n = \{x \in \mathbb{R}^n | \, |x| \leq 1\}. \tag{1.4}$$

- The composition of the inclusion $E^n_\beta \to X^{n-1} \sqcup \amalg_{\beta \in B} E^n_\beta$ and the quotient map $X^{n-1} \sqcup \amalg_{\beta \in B} E^n_\beta \to X^n$ gives a continuous function called the *characteristic function* of pairs $\Phi_\beta = \Phi^n_\beta : (E^n_\beta, S^{n-1}_\beta) \to (X^n, X^{n-1})$. Then the subset $\Phi_\beta(E^n_\beta) \subseteq X^n \subseteq X$ is called an *n-cell* and denoted as $\bar{e}^n_\beta$.
- This inductive process can be either stopped at finite stage, setting $X = X^n$ for some $n < \infty$, or can be continued indefinitly, setting $X = \cup_n X^n$. In the latter case X is given the weak topology meaning that a set $A \subset X$ is open (or closed) iff $A \cap X^n$ is open (or closed) in X^n for each n.

A CW complex is called finite if it has finitely many cells, and finite-dimensional, if, for some N, the N-skeleton $X^N = X$. The smallest N for which this is the case is called the dimension of X, i.e. the maximum dimension of cells of X. Furthermore, a CW complex is said to be **regular** if every attaching map can be chosen to be a homeomorphism on the entire cell being attached.

Example 1.4.1.

- *A 1-dimensional cell complex $X = X^1$ is called a* ***graph*** *and consists of vertices (0-cells) and edges (1-cells) attached to the vertices.*
- *The sphere S^n has the structure of a two-cell complex $\bar{e}^0, \bar{e}^n$. The n-cell is attached by the constant map $S^{n-1} \to \bar{e}^0$.*
- *The n-dimensional real projective space admits a CW structure with one cell in each dimension.*
- *Polyhedra are naturally CW complexes.*

A subcomplex of a cell complex X is defined as a closed subspace $A \subset X$ that is a union of cells of X. A subcomplex of a cell complex is a cell complex in its own right: Since A is closed, the characteristic map and in particular the attaching map of each cell in A has an image contained in A. Up to weak equivalence, every topological space is a CW complex by the following theorem:

Theorem 1.4.1 (CW Approximation Theorem)**.** *If X is any space, then there is a CW complex Y and a map $f : X \to Y$ inducing isomorphisms on all homotopy, homology and cohomology groups.*

Proof. One should consult [[Hat02], Section 4.1, p. 357ff.] for a complete proof. Key to the proof is an inductive construction of a CW complex and a weak homotopy equivalence of this CW complex to X. The inductive step goes as follows [Hat02]:
Let A be a CW complex with a map $f : A \to X$ and let a_γ be a chosen basepoint 0-cell for each component of A. Then for an integer $k \geq 0$ one attaches k-cells to A to form a CW complex B with a map $f : B \to X$ extending the given f such that:

∗ The induced map $f_* : \pi_i(B, a_\gamma) \to \pi_i(X, f(a_\gamma))$ fullfills injectivity for $i = k-1$ and surjectivity for $i = k$, for all a_γ.

□

To construct a CW approximation for a topological space X, one starts with a CW complex A consisting of a point for each path-component of X together with a map $f : A \to X$ mapping each of these points to a path-component of X. According to the induction step described above, this bijection for π_0 is extended by attaching 1-cells to A to create a surjection on π_1 for each path-component, then 2-cells are added to improve this to an isomorphism on π_1 and a surjection on π_2 and so forth for higher π_i. Hence one obtains indeed a weak homotopy equivalence between a so-constructed CW complex A and a topological space X.

Example 1.4.2. *An example for this being only a weak homotopy equivalence is again given by the Warsaw circle, which is not homotopy equivalent to a CW complex.*

Proof. As the Warsaw circle is not contractible, this follows directly from the Whitehead theorem. □

1.4.2 Δ-Complexes

Definition 1.4.3 (Standard n-Simplex)**.** *The standard n-simplex is the convex hull of the unit vectors in the $d+1$-dimensional Euclidian space along the coordinate axes:*

$$\Delta^n = \{(t_0, ..., t_n) \in \mathbb{R}^{n+1} \mid \sum_i t_i = 1 \text{ and } t_i \geq 0 \ \forall i\}, \tag{1.5}$$

where the vertices $(t_0, ..., t_n)$ are ordered.

The ordering of the vertices is crucial for the definition of an arbitrary n-simplex.

Definition 1.4.4 (n-Simplex)**.** *An arbitrary n-simplex on some ordered points $[v_0, ..., v_n]$ in the $n+1$ dimensional Euclidian space is defined as a map from the standard n-simplex to $[v_0, ..., v_n]$ preserving the order of the vertices: $(t_0, ..., t_n) \to \sum_i t_i v_i$. The coefficients t_i are called the* ***barycentric coordinates*** *of the point $\sum_i t_i v_i$ in $[v_0, ..., v_n]$.*

Deleting one of the $n+1$ vertices of an n-simplex $[v_0, ..., v_n]$ results in an $n-1$-simplex, called the **face** of $[v_0, ..., v_n]$. This is a subsimplex of $[v_0, ..., v_n]$, with the remaining vertices being ordered accordingly. The union of all the faces of Δ^n is called the **boundary** $\partial\Delta^n$ of Δ^n, and $\Delta^n - \partial\Delta^n$ is called the interior $\mathring{\Delta}^n$ of Δ^n.

Definition 1.4.5 (Δ-complex)**.** *A Δ-complex is another way to define structure on the data $\mathcal{S}$ via a collection of maps $\sigma_\alpha : \Delta^n \to \mathcal{S}$, with n depending on the index α, such that [Hat02]:*

- *The restriction of σ_α to the interior of Δ^n is injective and each point in $\mathcal{S}$ is the image of exactly one such restriction map.*
- *Each restriction of σ_α to a face of Δ^n consists one of the maps $\sigma_\beta : \Delta^{n-1} \to \mathcal{S}$ for another index β.*
- *A set $A \subset \mathcal{S}$ is open if and only if σ_α^{-1} is open in Δ^n for each σ_α.*

Using the third property, a Δ-complex on $\mathcal{S}$ can be built as a quotient space of a collection of disjoint simplices and thus must be a Hausdorff space.

1.4.3 Simplicial Complexes

Simplicial complexes are examples of regular CW complexes.

Definition 1.4.6 (Abstract Simplicial Complex)**.** *Given a set V, an **abstract simplicial complex** with vertex set V is a set K of finite subsets of V such that the elements of V belong to K and for any $\sigma \in K$ and $\tau \subset \sigma$, also $\tau \in K$.*

The dimension of an abstract simplex σ is given by $\mathrm{card}(\sigma) - 1$ and the dimension of K is given by the largest dimension of its simplices.

Definition 1.4.7 (Simplicial Map)**.** *A simplicial map is a function f between two simplicial complexes K and L with vertex sets V_K and V_L, $f : V(K) \to V(L)$ such that every simplex in K is mapped to a simplex in L.*

Abstract simplicial complexes together with simplicial maps form a category **Simp**. With any abstract simplicial complex, one can associate a **geometric simplicial complex** via the **geometric realization functor**

$$|\cdot| : \quad \mathbf{Simp} \to \mathbf{Top}. \tag{1.6}$$

Definition 1.4.8 (Geometric Simplicial Complex)**.** *A simplicial complex $|K|$ in $\mathbb{R}^d$ (geometric simplicial complex) is a collection of simplices in $\mathbb{R}^d$ such that:*

1. *Every face of a simplex of $|K|$ is in $|K|$.*
2. *The intersection of any two simplices of $|K|$ is a face of each of them.*

The functor from Eq. (1.6) takes abstract simplicial complexes to geometric simplicial complexes and simplicial maps $f : K \to L$ to the induced mpas

$$|f| : |V| \to |L|, \quad \sum_i t_i \cdot v_i \to \sum_i a_i \cdot f(v_i). \tag{1.7}$$

The combinatorial description of any geometric simplicial complex $|K|$ gives rise to an abstract simplicial complex K. Conversely, for a given abstract simplicial complex K, if $|K|$ is a geometric simplicial complex whose combinatorial structure is the same as K, $|K|$ is called the **geometric realization** of $\tilde{K}$.

The Topology of Simplicial Complexes

Geometric simplicial complexes are topological spaces obtained by gluing together dimensional simplices.

Definition 1.4.9 (Subspace Topology)**.** *Let $|K|$ be the subset of $\mathbb{R}^n$ that is the union of the simplices of K. One way to topologize $|K|$ is by giving each simplex its natural topology as a subspace of $\mathbb{R}^n$.*

For **infinite** K one needs a finer topology than the subspace topology $|K| \subset \mathbb{R}^n$.

Definition 1.4.10 (Final Topology)**.** *Let $f_i : F_i \to K$ be a family of maps whose domains carry a topology. The final topology on $|K|$ is the topology for which a subset $U \subset K$ is open whenever $f_i^{-1}(U) \subset F_i$ is open for all i.*

For infinite K, we give $|K|$ the final topology with respect to all maps $|\sigma| \xhookrightarrow{|j|} |K|$, where $\sigma \xhookrightarrow{j} K$ is the inclusion of a finite subcomplex. Thus $U \subset |K|$ is open if and only if $U \cap |\sigma|$ is open for all finite $\sigma \subset K$. The space $|K|$ is called the **underlying space** of K. The realization $|K|$ is Hausdorff and if K is finite, $|K|$ is compact.

Subcomplexes and Stars

A subcollection L of K that contains all faces of its elements is a simplicial complex in its own right, called a subcomplex.
A subset of a simplicial complex useful for talking about local neighborhoods is given by the *star*, which however, not necessarily is a subcomplex.

Definition 1.4.11 (Star of a Simplex)**.** *The star of a simplex σ consists of all simplices that have σ as their face,*

$$St(\sigma) = \{\tau \in K \,|\tau \leq \sigma\}. \tag{1.8}$$

The star of a simplex can be made into a complex by adding all missing faces. The resulting complex $\bar{St}(\sigma)$ is called the *closed star* of a simplex.

Filtrations

Definition 1.4.12 (Filtration)**.** *A **filtration** of K is defined as a partial ordering on the simplices of K such that every prefix of the ordering is a subcomplex. It is denoted as $\leq_K$.*

A simplicial complex equipped with a filtration is called a **filtered simplicial complex**. This also defines a functor.

Remark 1.4.1. *Let $\mathcal{I}$ be an indexing category. A filtered simplicial complex is a functor $\mathcal{K} : \mathcal{I} \to$ **Simp**, such that $K^s \subseteq K^t$ for any pair $s \leq t$ in $\mathcal{I}$.*

Let $\mathbf{r}$ be an indexing set, most commonly chosen as $\mathbb{R}$. By construction, there exists some smallest filtration parameter a at which no connectivity information is added to the data set $\mathcal{S}$ and the subcomplex K^a hence corresponds to the set of data points $\{\mathcal{S}\}$. Thus, for $a, b, g, ..., t$ some filtration parameters in $\mathbf{r}$, one obtains a sequence of subcomplexes connected via indexed inclusion maps ι_r between them:

$$K^a \xhookrightarrow{\iota_a} K^b \xhookrightarrow{\iota_b} ... \xhookrightarrow{\iota_f} K^g \xhookrightarrow{\iota_g} ... \tag{1.9}$$

If the number of points in $\mathcal{S}$ is finite, a maximum filtration parameter $\bar{\rho}$ exists for which the connectivity stabilizes. Hence, if we choose the respective filtration parameter r large enough, i.e. $r > \bar{\rho}$, the sequence in Eq. (1.9) will attain its final term $K^{\bar{\rho}} = K$, for which all data points are inter-connected.

Each simplex $\sigma \in K$ can be assigned to the minimum scale at which it appears in the filtration. Hence one can define a monotonic function $f : K \to \mathbf{r}$ with $f(\sigma) = t$ if $\sigma \in K^t$ and $\sigma \notin K^s$ for $s < t$. In other words, a filtration induces a monotonic function on the simplices of K, where $f(\sigma) \leq f(\tau)$ if σ is a face of τ.
More abstractly, via association of geometric complexes to the subcomplexes in a filtration and f_e being the inclusion of one subcomplex of the filtration into its subsequent subcomplex, a filtered simplicial complex also defines a **diagram of spaces**.

Definition 1.4.13 (Diagram of Spaces [Hat02])**.** *A **diagram of spaces** consists of an oriented graph Γ together with a space X^μ for each vertex $\mu \in \Gamma$ and a map $f_e : X^\mu \to X^w$ for each edge e of Γ from a vertex v to a vertex w, where edges e are oriented but no commutativity of Γ is assumed. In other words, a diagram of spaces is a functor $\mathcal{D}$ from a given indexing category $\mathcal{I}$ to the category of topological spaces* **Top***:*

$$\mathcal{D} : \mathcal{I} \to \mathbf{Top}. \tag{1.10}$$

For the standard 1−paramter case, the indexing category $\mathcal{I}$ is $\mathbb{R}$ or $\mathbb{N}$. In the following, a **filtered space** describes a topological space obtained form a point cloud data set via simplicial complexification. Other examples for diagrams of spaces are given in Example 1.4.3 below, [Hat02].

Example 1.4.3.

- *A sequence of inclusions of topological spaces X_i*

$$X^1 \hookrightarrow X^2 \hookrightarrow ... \tag{1.11}$$

  determines a diagram of spaces.

- *From a cover (see Definition 1.6.1) $\mathcal{U} = \{U_\alpha\}$ of a space X by subspaces U_α one can form a diagram of spaces $X_\mathcal{U}$ whose vertices are the nonempty finite intersections of $U_{\alpha_1} \cap ... U_{\alpha_n}$ with distinct indices α_j and whose edges are the various inclusions obrtained by omitting some of the subscpaes in such an intersection such as $U_{\alpha_i} \cap U_{\alpha_j} \hookrightarrow U_{\alpha_i}$.*

1.5 Cubical Complexes

Cubical complexes also constitute abstract simplicial complexes according to Definition 1.4.6. However, in contrast to simplicial complexes, geometric cubical complexes are not constructed via standard n-simplices, but instead via *elementary cubes*.

Definition 1.5.1 (Elementary Cube)**.** *An elementary cube is a finite product of elementary intervals I, with $I = [l, l+1]$ or $I = [l, l]$, $l \in \mathbb{Z}$:*

$$Q = I_1 \times I_2 \times ... \times I_d \subset \mathbb{R}^d. \tag{1.12}$$

Definition 1.5.2 (Cubical Set)**.** *A set $X \subset \mathbb{R}^d$ is cubial, if it can be written as a finite union of elementary cubes.*

Examples for the usage of cubical complexes can be found in digital image processing and analysis, the study of dynamical systems or structural medicine. Cubical homology is particularly useful for image data, since the finite unions of cubes can be thought of as representing the pixels in the image. An implementation can be found in gudhi and giotto-tda.

1.6 Reconstruction

Investigating the structure of a given point cloud $\mathcal{S}$ intrinsically includes the assumption that the points are sampled from a shape $\mathbf{S}$. The aim of topological data analysis is to reconstruct the topology of this shape. By definition, a topological space on $\mathcal{S}$ can be constructed via a *cover* of $\mathcal{S}$.

Definition 1.6.1 (Cover)**.** *Let R be set and $\mathcal{U} = \{U_\alpha \mid \alpha \in A\}$ an indexed family of subsets $U_\alpha \subset R$. Then $\mathcal{U}$ is a cover of R if*

$$R = \bigcup_{\alpha \in A} U_\alpha. \tag{1.13}$$

*If the cover consists of open sets, it is called an **open cover**.*

Most intuitively, to study a non-trivial topology of $\mathbf{S}$, one would construct balls of radius α around each point p in $\mathcal{S}$ and investigate their intersections. This construction is called an α-offset of the sample points $p \in \mathcal{S}$.

Definition 1.6.2 (α-Offset)**.** *An α-offset of a set of n points p in $\mathcal{S} \subseteq \mathbb{R}^d$ is a union of n d-balls centered at the sampled points p, with radius α.*

The α-offset of a point cloud $\mathcal{S}$ will be denoted as $\mathcal{S}^\alpha$ in the following.
Let the underlying shape $\mathbf{S}$ of $\mathcal{S}$ be a smooth manifold $\mathcal{M}$, from which the points $\mathcal{S}$ were sampled in a sufficiently dense way. Then [NSW08] proved that the α-offset is homotopic to $\mathcal{M}$ for a suitable α. For this case, the parameter α is bounded from above by another parameter τ dependent on the local and

global curvature properties of the manifold $\mathcal{M}$.
It is hypothesized that the data sets appearing in **step IV.** of the analysis pipeline in Figure 1.1 are manifolds [FMN13]. However, in general, the requirement for the transition of **step I.** to **step II.** in Figure 1.1 is, that the complexification of data at scale α has the homotopy type of the corresponding α-offset $\mathcal{S}^\alpha$. For some geometric simplicial complexes, this property is guaranteed by a **Nerve Theorem**.

1.7 Nerve Theorem

In general, **Nerve Theorems** state that the intersection pattern of a cover of a topological space X fully determines X with respect to its algebraic topological invariants under suitable hypotheses. Among those hypotheses are, for example, contractibility or acyclicity of all non-empty and finite intersections of cover elements. Different versions of Nerve Theorems were given by Borsuk [Bor48], Leray [Ler45], Weil [Wei52], McCord [McC67], Björner [Bjö03] and Barmak [BM06].
Nerves as initially defined by [Ale28], provide a way to construct a computable topological space from a point cloud via a suitable cover. Each set of such a cover is given by a local cluster or a grouping of data points sharing some common properties. The nerve is a way to compactly and globally describe the relationship between these sets through their intersection patterns.

Definition 1.7.1 (Nerve)**.** *Given a simplicial complex K and a cover $\mathcal{U}$, the nerve $\mathcal{N}(\mathcal{U})$ is defined as the simplicial complex whose k-simplices represent the non-trivial intersection of subsets of $\mathcal{U}$, of size $k+1$.*

For any simplicial complex K on a data set $\mathcal{S}$ one can thus define a nerve via a finite collection of subcomplexes L. The nerve $\mathcal{N}(L)$ constitutes the abstract simplicial complex whose vertices are the subcomplexes in L and whose simplices are the subcollection of L with a nonempty intersection.

Theorem 1.7.1 (Nerve Theorem ([Hat02])**.** *Let $\mathcal{U}$ be an open cover of a paracompact space X such that every nonempty intersection of finitely many sets in $\mathcal{U}$ is contractible. Then X is homotopy equivalent to the nerve $N(\mathcal{U})$.*

For a simplicial complex K constructed on a data set $\mathcal{S}$ and a cover of subcomplexes $\mathcal{L} = \{L_i\}_{i\in I}$ it has been proven by [Bjö03] that K and the nerve of the covering $\mathcal{N}(\mathcal{L})$ are homotopy equivalent under the condition of $(k-t+1)$-connectedness for their k-wise intersections.

Definition 1.7.2 (k-Connectedness, [Bjö03])**.** *A space X is called k-connected ($k \geq 0$) if for every $0 \leq r \leq k$, every map of the r-sphere into X is homotopic to a constant map.*

However, this does not imply that every simplicial complex on $\mathcal{S}$ induces a nerve of the same homotopy type. For arbitrary complexes, the intersections of the subcomplex cover elements are not necessarily connected, although the connected components of the non-empty intersections are contractible [Car09]. This case does not fall under the Nerve Theorem.

1.8 Examples of Simplicial Complexes

In the following, we will present and discuss the most common simplicial complexes. These comprise (Vietoris-)Rips, Čech, Delaunay, Witness and Graph-Induced complexes [Ott+17]. An overview is shown in Figure 1.4. Vietoris-Rips and Čech complexes are closely related, but Čech complexes have better theoretical guarantees via the Nerve Theorem 1.7.1. Both are growing exponentially of order $2^{\mathcal{O}(N)}$ in size with respect to their number N of vertices [Ott+17]. The other complexes mentioned represent different approaches to control this size, as for any reasonable point cloud, it becomes uncalculable.

Example 1.8.1. *For the 8-dimensional random point cloud (1000 vertices), see Figure 3, one obtains the following Vietoris-Rips complexes with gudhi (for $\epsilon \leq 12$):*

- ***dimension 1****: 500500 simplices.*
- ***dimension 2****: 166667500 simplices.*

For the deepVALVE data segment shown in figure 6 (192 vertices for 5 dimensions), one obtains:

- ***dimension 1****: 18528 simplices.*
- ***dimension 2****: 1179808 simplices.*
- ***dimension 3****: 56050288 simplices.*

The Delaunay and the alpha complex constitute Voronoi approximations to the Vietoris-Rips and Čech complexes, where the alpha complex is a subcomplex of the Delauany complex and also a subcomplex of the Čech complex. The witness complex is a way to sparsify the vertex set by constructing simplices around a defined subset - the *witnesses*. Lastly, Graph-Induced (GIC) complexes are an intermediate between witness and Rips-complexes. They were designed in order to make the idea of witness complexes applicable to higher dimensions than $(2D, 3D)$.

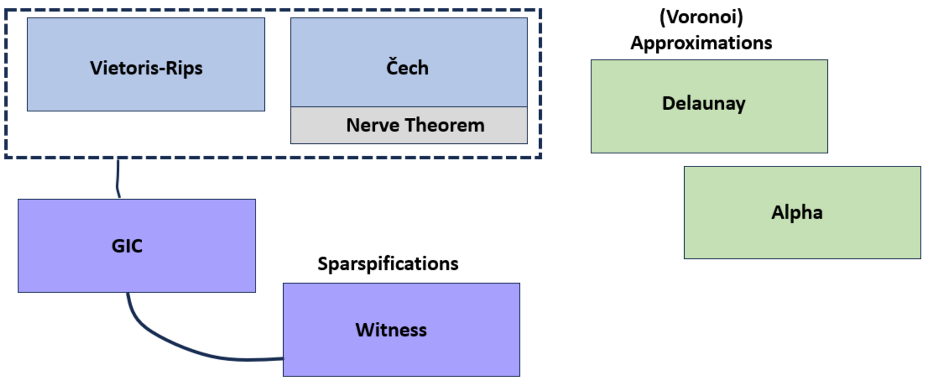


Figure 1.4: Overview of the simplicial complexes presented in the subsequent sections.

1.8.1 Vietoris-Rips and Čech Complex

The Vietoris-Rips complex generalizes proximity graphs to higher dimensions.

Definition 1.8.1 (Vietoris-Rips Complex)**.** *Let* (M, d_M) *be a metric space and let* $\mathcal{S} \subseteq M$*. For* $\epsilon \in \mathbb{R}$*, a simplicial complex* $\mathrm{Rips}(\mathcal{S}, \epsilon)$ *is defined on the vertex set of* $\mathcal{S}$ *as follows:*

$$[p_0, p_1, ..., p_k] \in \mathrm{Rips}(\mathcal{S}, \epsilon) \Leftrightarrow d_M(p_i, p_j) \leq \epsilon \tag{1.14}$$

for all $i, j \in \{0, k\}$*. For* $\epsilon \leq 0$*,* $\mathrm{Rips}(\mathcal{S}, \epsilon)$ *consists of the vertex set alone.*

On the contrary, the Čech Complex for a set of points $\mathcal{S} \subseteq M$ is an example for a *cover complex.* A cover complex is a complex whose simplices are computed by means of a cover of the point cloud. Often, this cover comes from the preimages of intervals covering the image of a function defined on the point cloud. In particular, the Čech complex is defined via a *good cover* of the set.

Definition 1.8.2 (Good Cover)**.** *A good cover is an open cover in which all sets and all nonempty intersections of finitely many sets are contractible.*

Definition 1.8.3 (Čech Complex)**.** *Let* (M, d_M) *be a metric space,* $\mathcal{S} \subseteq M$ *and* $\epsilon \in \mathbb{R}$*. The Čech complex* $\check{C}ech(\mathcal{S}, \epsilon)$ *is defined as follows:*

$$[p_0, p_1, ..., p_k] \in \check{C}ech(\mathcal{S}, \epsilon) \Leftrightarrow \bigcap_{i=0}^{k} B[p_i, \epsilon] \neq \emptyset. \tag{1.15}$$

The Čech complex thus constitutes, by definition, the nerve of a good cover.

Theorem 1.8.1 (Čech Theorem [GM05])**.** *The nerve of a collection of convex sets has the same homotopy type as the union of this set.*

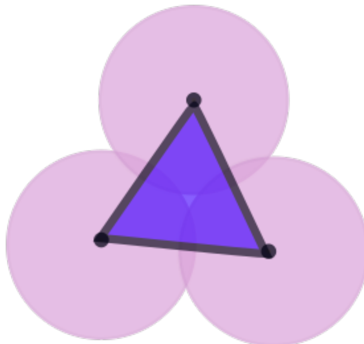


Figure 1.5: Rips-Complex: Graph with a 2-Simplex.

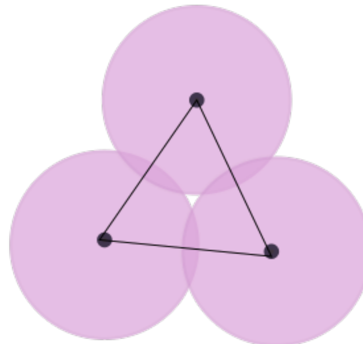


Figure 1.6: Čech Complex: Graph.

By construction, a good cover is built from convex sets and thus Čech complexes are homotopy equivalent to the corresponding α-offset. In contrast to this, the Vietoris-Rips complex is usually not defined as a nerve. For an example using the l_∞-norm and a cover by cubes, see [GM05], Lemma VII.
Therefore, Vietoris-Rips complexes do not fall within the scope of the Nerve Theorem. However, they approximate Čech complexes, as illustrated in the following lemma.

Lemma 1.8.1 (Vietoris-Rips Lemma, p.62 in [EH10])**.** *For all $\epsilon > 0$, the following inclusions hold:*

$$\check{C}ech(\mathcal{S}, \epsilon) \subset \mathrm{Rips}(\mathcal{S}, \epsilon) \subset \check{C}ech(\mathcal{S}, 2\epsilon). \tag{1.16}$$

In [SG07] it has been proven that the above inclusion can be tightened at most to

$$\mathrm{Rips}(\mathcal{S}, \epsilon') \subset \check{\mathrm{C}}\mathrm{ech}(\mathcal{S}, \epsilon) \subset \mathrm{Rips}(\mathcal{S}, \epsilon) \quad \text{for} \quad \frac{\epsilon}{\epsilon'} \geq \sqrt{\frac{2d}{d+1}} \tag{1.17}$$

for a set of points $\mathcal{S} \in \mathbb{R}^d$ and $\check{\mathrm{C}}\mathrm{ech}(\mathcal{S}, \epsilon)$ the Čech complex of the cover of $\mathcal{S}$ by balls of radius $\epsilon/2$.

Figure 1.5 and Figure 1.6 illustrate the difference between the Rips and the Čech complex: For the Rips complex the pairwise intersections between the three disks lead to a two-simplex, whereas for the Čech construction we obtain only a one-simplex as the balls do not have a threefold intersection.

From the Nerve Theorem follows that the Čech complex of a point cloud $\mathcal{S}$ is always homotopic to the union of diameter-ϵ balls about the vertex set of $\mathcal{S}$. Hence the Čech complex can be seen as an appropriate simplicial model for the topology of the point cloud.
This is not the case for the Vietoris-Rips complex. Unlike the Čech complex, even when $\mathcal{S}$ is a finite subset of $\mathbb{R}^d$, the Vietoris-Rips complex $\mathrm{Rips}(\mathcal{S}, \epsilon)$ does

not admit a geometric realization in $\mathbb{R}^d$. It can have nontrivial topological invariants (homotopy or homology groups) of dimension d and above. To illustrate this, consider a point cloud $\mathcal{S} \in \mathbb{R}^d$ and the projection map $p : \mathrm{Rips}(\mathcal{S}, \epsilon) \to \mathbb{R}^d$ taking the vertices of the Rips-complex to $\mathcal{S}$ and the k-simplices of $\mathrm{Rips}(\mathcal{S}, \epsilon)$ to the convex hull of the associated points in $\mathcal{S}$. The d-dimensional image of p in $\mathbb{R}^d$ is called the **shadow** of the Vietoris-Rips complex. In general one wants the images of complexes in $\mathbb{R}^d$ to lie close to the data. However, there is no chance of $\mathrm{Rips}(\mathcal{S}, \epsilon)$ being homeomorphic to its shadow, and also homotopy equivalence is too much to ask, [Cha+10]. In fact, the domain of the projection map p is likely to have higher dimension than the codomain. Herein, the choice of the parameter ϵ is crucial as demonstrated in the following example adapted from [Cha+10], Proposition 5.3.

Example 1.8.2 (Vietoris-Rips Bubbles). *Figure 1.7 shows six hexagonally distributed points $\{x_1, .., x_6\}$ whose shortest inter-point distance is r. The shadow of these points in $\mathbb{R}^2$ is just the hexagon enclosed by these points. Thus it is contractible and has the homotopy type of a point. However, the Vietoris-Rips complex for $r \leq \epsilon \leq \sqrt{3}r$ has the structure of an octahedron and thus the homotopy type of a 2-sphere.*

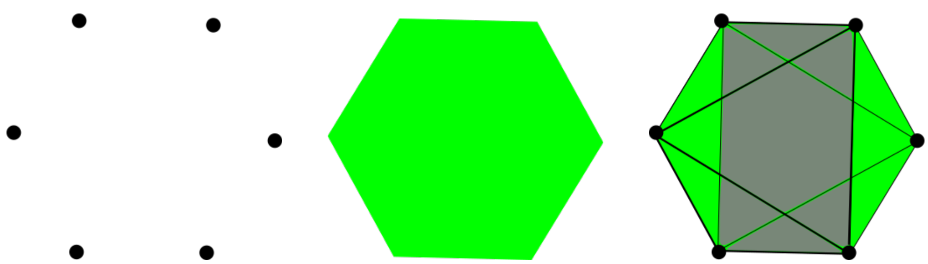


Figure 1.7: From left to right: Hexagonally distributed points in $\mathbb{R}^2$. Shadow of these points in $\mathbb{R}^2$. Vietoris-Rips complex.

The above example demonstrates that already for points in $\mathbb{R}^2$, the Vietoris-Rips complex does not capture the topology of the associated cover by $B[x_i, \epsilon]$. With regard to this, the Vietoris-Rips complex is sensible to the choice of ϵ. Extrapolating Example 1.8.2 to more general situations, this indicates that for inhomogeneously distributed data, where most of the data points have smallest inter-point distances r_M Vietoris-Rips bubbles can appear in denser regions if ϵ is chosen too small, i.e. $\epsilon \ll r_M$. Although these topologically insignificant artefacts might appear, empirically, it is known that Vietoris-Rips constructions capture the topology of large scale structures correctly [GM05].

1.8.2 Delaunay Complex

Delaunay complexes are an approach to limit the dimension of simplices in a complexification to the dimension of the ambient space. This is done via a

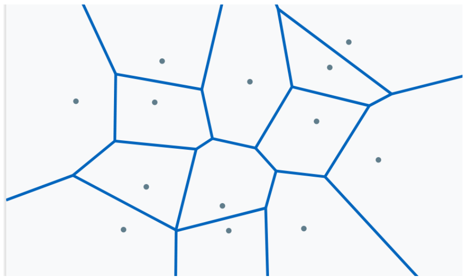


Figure 1.8: Voronoi diagram.

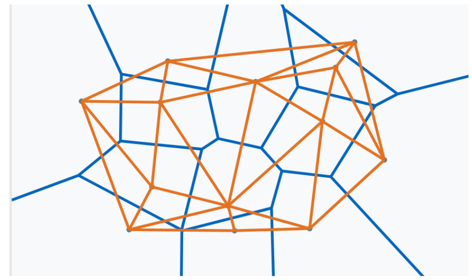


Figure 1.9: Delaunay triangulation.

geometric construction based on Voronoi cells. Given a point set $\mathcal{S} \subseteq \mathbb{R}^d$ of n distinct points, a Voronoi diagram is the partition of $\mathbb{R}^d$ into n Voronoi cells.

Definition 1.8.4 (Voronoi Cell)**.** *The Voronoi cell V_p of a point p in a finite metric space $\mathcal{S} \subseteq \mathbb{R}^d$ is given by the set of points to which p is closest:*

$$V_p = \{x \in \mathbb{R}^d \mid ||x-p|| \leq ||x-q||,\ q \in \mathcal{S}\}. \tag{1.18}$$

The Delaunay complex is constructed as the dual graph of the Voronoi diagram of a given point set. This is illustrated for a two dimensional example in Figures 1.8 and 1.9, which were generated with [Bru23]. For the Delaunay construction, the Voronoi vertices have to be in *general position*, i.e. in $\mathbb{R}^d$ there are no more than $d+1$ neighboring Voronoi cells. Then the Delaunay triangulation is constructed by connection of the Voroni vertices, which results into a partition of the convex hull of $\mathcal{S}$ as follows:

Definition 1.8.5 (Delauany Complex)**.** *The Delaunay complex of a finite metric space $\mathcal{S} \subseteq \mathbb{R}^d$ is isomorphic to the nerve of the collection of Voronoi cells, i.e.*

$$\text{Delaunay} = \{\sigma \subseteq \mathcal{S} | \cap_{p\in\sigma} V_p \neq \emptyset\}. \tag{1.19}$$

In Gudhi, there is a modification of Delaunay complexes implemented called *Tangential* Delaunay commplexes. These are constructed as follows [Jam20]: First, for each point $p \in \mathcal{S}$, the tangent subspcae T_p is estimated. Subsequently, the Voronoi diagram is added. Then for each point, its star in the Delaunay triangulation, restricted to T_p is constructed. The Tangential Delaunay complex is the union of these stars.

1.8.3 Alpha Complex

The alpha complex is a subcomplex of the Delaunay complex, constructed as follows: For each point $p \in \mathcal{S}$, let $B[p,r]$ be the closed ball in $\mathbb{R}^d$ centered around p. The set of points at distance at most r from at least one of the points in $\mathcal{S}$ give the union of these balls:

$$\text{Union}(r) = \{x \in \mathbb{R}^d \mid \exists p \in \mathcal{S} \text{ with } ||x-p|| \leq r\}. \tag{1.20}$$

Intersecting each ball with the corresponding Voronoi cell, $R_p(r) = B[p, r] \cap V_p$ gives a decomposition of this union. Since balls and Voronoi cells are convex, their intersections are also convex. The alpha complex is isomorphic to the nerve of the cover given by $R_p(r)$:

$$\text{Alpha}(r) = \{\sigma \subseteq \mathcal{S} | \bigcap_{p\in\sigma} R_p(r) \neq \emptyset\}. \tag{1.21}$$

Thus like the Čech complex, the alpha complex fullfills the Nerve Theorem and has the same homotopy type as the union of balls, Definition 1.6.2. By construction the alpha complex is a subcomplex of the Delaunay complex. Since $R_p(r) \subseteq B[p, r]$, it is also a subcomplex of the Čech complex $\text{Alpha}(r) \subseteq \check{\text{C}}\text{ech}(r, \epsilon)$. Although the alpha complex is smaller than the Čech complex, it is yet still computationally unfeasible as it relies on a Voronoi construction, Definition 1.8.4.

Example 1.8.3. *For the deepVALVE data segment (Figure 6), the resulting alpha complex computed with gudhi is of dimension 5 with 136731 simplices, whereas, for a (64-Bit GNU/Linux machine with 12 cores 3.60GHz Intel(R) Xeon(R) CPU E5-1650 v4) machine, the Rips complex is only computable up to dimension 3, where it already consists of 56050288 simplices.*

By allowing different radii for the above-mentioned balls $B[p, r]$ around $p \in \mathcal{S}$ one obtains a *weighted* version of alpha complexes, which is useful, for example [Men+19], within the analysis of biomolecules.

1.8.4 Witness Complex

In contrast to Vietoris-Rips complexes, for which a k-simplex is added when $k+1$ vertices lie ϵ-close to each other, witness simplices include k-simplices when $k+1$ vertices lie close to some designated points of $\mathcal{S}$. The latter are called *landmark points*. The key idea is to use the non-landmark points as witnesses to the existence of edges or simplices spanned by combinations of landmark points. This is done in order to remove points from the complexification which are redundant with regard to the reconstruction of the underlying topology. In this sense, witness complexes constitute an approximation to Vietoris-Rips and Čech complexes. The size of witness complexes can be determined via the choice of the number of landmark points. This is illustrated in Figure 1.10, adapted from [GO08].

Remark 1.8.1 (Construction of Witness Complexes)**.** *Witness complexes are constructed as follows: Given the point cloud data $\mathcal{S}$, a subset L of vertices is chosen randomly. Alternatively, the set L can be chosen using a weak optimisation strategy for good distribution. Witness complexes are related to Delaunay complexes as elaborated in [GO08]. When the landmark points lie in general position in a Euclidian space, [Sil03] proved that the witness complex* $\text{witness}(L)$ *is*

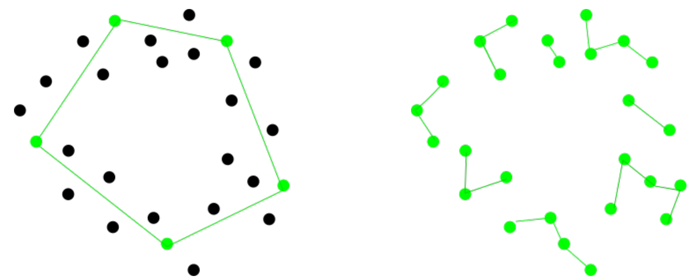


Figure 1.10: A smooth closed curve with noise, sampled by witnesses. Shown are two different sets of landmark points (green) and the corresponding witness complexes.

a subcomplex of the respective Delauany complex Delaunay(L)*. The cardinality of* L *constrains the complexity of the detectable topology. Some* $x_0, x_1, ..., x_k \in L$ *span a* k*-simplex if and only if:*

- *There is a point* $w \in \mathcal{S}$ *whose* $k+1$ *nearest neighbours in* L *are* $x_0, x_1, ..., x_k$ *(weak witness); and*
- w *is equidistant from* $x_0, x_1, ..., x_k$ *(strong witness).*

As the existence of a strong witness point w in a finite point set is almost a probability zero event, one either needs to introduce a tolerance parameter instead of equidistance or restrict the complexification to the use of weak witnesses. Another constraint with the use of witness complexes comes from dimensionality. In [[BGO07], Theorem 2.4] it has been proven that witness complexes are not homotopy equivalent to manifolds already in dimension 4, even for strong sampling conditions. This result contributes to the conclusion that in dimension 3 or higher, witness complexes have too few simplices to capture the topology of the sampled space [BGO07].

1.8.5 Graph-Induced Complex

Graph-induced (GI) complexes rely, like witness complexes, on a sparsification of the vertices used to build the respective simplices. However, in contrast to witness complexes, they are also constructed as a clique complex like the Vietoris-Rips complex.

Remark 1.8.2 (Construction of Graph-induced Complexes [DFW13])**.** *Given a data set* $\mathcal{S} \subseteq \mathbb{R}^d$*, graph-induced complexes are constructed as follows: First of all, a graph is constructed on* $\mathcal{S}$ *equipped with a metric. then a complex induced by this graph is built on a subsample* $Q \subseteq \mathcal{S}$*. Herefore, a simplex with a vertex set* $V \subseteq Q$ *is constructed if a set of points in* $\mathcal{S}$*, each being closest to exactly one vertex in* V*, forms a clique. Input points are grouped according to their proximity to the subsampled vertices.*

Definition 1.8.6 (Graph-induced Complex, Definition 2.3 in [DFW13]). *Let $(\mathcal{S}, d)$ be a finite metric space consisting of a finite point set $\mathcal{S}$ equipped with a metric d. Let $Q \subset \mathcal{S}$ be a subset. Let $\mu_d : \mathcal{S} \to Q$ denote the nearest point map with $\mu_d(p)$ being a point in* $\operatorname{argmin}_{q \in Q} d(p, Q)$*, $p \in \mathcal{S}$. Given a graph $G(\mathcal{S})$ with $\mathcal{S}$ as its vertex set, the graph-induced complex is defined as*
$\mathcal{G}(\mathcal{S}, Q, d) := \mathcal{G}(\mathcal{S}, Q, \mu_d)$.

Hence, the data forming the simplicial complex is sparsified considerably via the clique condition. However, the ability of GI complexes to decipher the topology of the data is dependent on several parameters, among which is the metric chosen. For the Euclidian metric d_E, $G^\alpha(\mathcal{S})$ will be the graph induced complex for $(\mathcal{S}, E^\alpha, d_E)$ with $(p_1, p_2) \in E^\alpha$ if and only if $||p_1 - p_2|| \leq \alpha$. The main guarantees for topological inference by GI complexes stems from a sandwiching relation of $\mathcal{G}^\alpha(\mathcal{S}, Q, d)$ to Rips complexes $\operatorname{Rips}(\mathcal{S}, \alpha)$ and $\operatorname{Rips}(\mathcal{S}, \alpha + 2\delta)$, where Q is a δ-offset of $\mathcal{S}$. More details can be found in [DFW13], Proposition 2.8. The libraries presented in section 0.2.2 do not comprise an implementation of GI complexes, which are not used widely in practice yet.

1.9 Simplicial Homology

Let K a simplicial complex, G an Abelian group and p a dimension for the simplicial complexes to be considered. Homology for simplicial complexes with coefficients in G is defined via formal sums of p-simplices in K, called p-chains. Together with the addition operation the p-chains are defined as free Abelian groups denoted as $C_p = C_p(K)$. For a p-chain $c = \sum a_i \sigma_i$, $a_i \in G$, $\sigma_i \in K$, the boundary is given by the sum of the boundaries of its simplices. Hence, taking the boundary maps a p-chain to a $(p-1)$-chain

$$\partial_p : C_p \to C_{p-1}. \tag{1.22}$$

Simplicial homology is fully defined and calculated via the above boundary map.

Definition 1.9.1 (Simplicial Homology). *For any $p \in \mathbb{N}_0$, the p-th homology of a simplicial complex K is the quotient*

$$H_p(K, G) := \ker(\partial_p)/\operatorname{im}(\partial_{p+1}). \tag{1.23}$$

Simplicial homology groups are defined as quotients of Abelian groups and thus Abelian. For field coefficients, the p-chain groups and hence also the simplicial homology groups become vector spaces.
Elements in the image of ∂_{p+1} are called p-boundaries, and elements in the kernel of ∂_p are called p-cycles. The **Fundamental Lemma of Homology** states that the boundary of a boundary is zero, i.e. $\partial_p \partial_{p+1} c = 0$ for every integer p and every $(p+1)$-chain c, [EH10]. Hence the group of p-boundaries B_p is a subgroup of the cycle group Z_p.

In the following, we will consider simplicial homology with coefficients in a field $\mathbb{F}$ if not otherwise denoted. In this case, the p-chains form vector spaces and the boundary operator ∂_p has a particular matrix representation M_p:

Definition 1.9.2 (Boundary Matrix)**.** *The p-th **boundary matrix** of a simplicial complex K represents the $(p-1)$-simplices as rows and the p-simplices as columns. Fixing an ordering of the simplices, for each dimension, this matrix M_p has entries $M_p = [a_i^j]$, where i ranges from 1 to m_{p-1}, the cardinality of $(p-1)$-simplices, and j ranges form 1 to m_p, the cardinality of p-simplices. The entries a_i^j take values in $\mathbb{F}$ and usually are chosen such that $a_i^j = 1$, if the i-th $(p-1)$-simplex is a face of the j-th p-simplex and $a_i^j = 0$ otherwise.*

A collection of columns of the boundary matrix represents a p-chain and the sum of these columns gives its boundary. Likewise, collections of rows represent $(p-1)$-chains and their sum the coboundaries. An algorithm to extract the p-th homology groups from the boundary matrix M_p will be presented in Section 2.3.2. If K is a simplicial complex of dimension n, then $H_p(K) = 0$ for any $p > n$ and thus one obtains the following exact sequence of vector spaces and linear maps:

$$0 \xrightarrow{\partial_{n+1}} C_n(K) \xrightarrow{\partial_n} ... \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0. \tag{1.24}$$

Simplicial homology counts the cycles that are not boundaries which intuitively can be associated to "holes" in the topological space X. The cardinality of p-dimensional such holes is given by the dimension of $H_p(X, G)$, which is called the p-th **Betti number**. The ranks of the boundary operators appearing in the chain complex associated with a simplicial complex K completely determine its Betti numbers, as illustrated in Example 1.9.1 below, [Nan20].

Example 1.9.1. *The Betti numbers of the **solid** 0**-simplex** over any field $\mathbb{F}$ are*

$$\beta_k(\Delta_0) = \begin{cases} 1 & k = 0 \\ 0 & k > 0. \end{cases} \tag{1.25}$$

The corresponding simplicial chain complex exhibits only one nontrivial chain group C_k for $k = 0$ (as we have only one vertex):

$$... \longrightarrow 0 \longrightarrow 0 \longrightarrow \mathbb{F} \longrightarrow 0;$$

and thus all homology groups except $H_0 = \mathbb{F}$ are trivial.
*On the other hand, for a **hollow** 2**-simplex**, we get Betti numbers as*

$$\beta_k(\partial\Delta_2) = \begin{cases} 1 & k \in \{0, 1\} \\ 0 & k > 1. \end{cases} \tag{1.26}$$

Again, considering the chain complex, which is nontrivial only in dimension 0 and 1 (3 vertices, 3 edges):

$$... \longrightarrow 0 \longrightarrow \mathbb{F}^3 \longrightarrow \mathbb{F}^3 \longrightarrow 0;$$

and the only nontrivial boundary map ∂_1 has rank 2 (as 2 vertices form the boundary of one edge). Thus, its image has dimension 2 and its kernel has dimension 1 and via consideration of the above exact sequence one obtaines $\beta_1 = 3 - 2 = 1$ and $\beta_0 = 1 - 0 = 1$.

Relative Homology

For a given space X and a subspace $A \subset X$, relative homology groups can be defined as follows [Hat02]: Let $C_p(X, A)$ be the quotient group $C_p(X)/C_p(A)$. As a consequence, chains in A are trivial in $C_p(X, A)$. The boundary map for the chains in X induces a boundary map on the quotient

$$\partial : C_p(X, A) \to C_{p-1}(X, A),$$

and thus for variable n, a sequence of boundary maps:

$$... \longrightarrow C_p(X, A) \xrightarrow{\partial} C_{p-1}(X, A) \longrightarrow ...$$

where the relation $\partial^2 = 0$ holds because it also holds before passing to the quotient. The relative homology groups $H_p(X, A)$ are defined analogously to Definition 1.9.1.

Simplicial Homology as a Functor

Definition 1.9.3 (Homology Functor)**.** *A system $H = (H_i)_{i \in \mathbb{Z}}$ of covariant additive functors from an Abelian category $\mathcal{A}$ into an Abelian category $\mathcal{A}_1$ is called a homology functor if the following two axioms are satisfied [Mat22]:*

1. *For each exact sequence in $\mathcal{A}$,*

$$0 \to A' \to A \to A'' \to 0 \tag{1.27}$$

   and each index i, in $\mathcal{A}$ a morphism ∂_i $H_{i+1}(A'') \to H_i(A')$ is given, which is known as the connecting or boundary morphism.

2. *The sequence*

$$... \to H_{i+1}(A') \to H_{i+1}(A) \to H_{i+1}(A'') \to H_i(A') \to ..., \tag{1.28}$$

   called the homology sequence, is exact.

Field coefficient simplicial homology H in a given dimension p is a functor from **Simp** to the category **Vect**. This functor H can be used to convert any diagram in **Simp** to a diagram in **Vect**.
Likewise, **simplicial cohomology groups** are defined as a contravariant version of this functor.

1.10 Cubical Homology

Homology can also be defined via elementary cubes. Analogeously to simplicial homology, elementary cubes define the basis elements of elementary chains, for which a cubical boundary operator can be defined. This is used to define a cubical chain complex for a cubical set $X \subset \mathbb{R}^d$. Just as in Definition 1.9.1, the p-th cubical homology group is defined as the quotient group of cycles modulo boundaries. Cubical homology is used when the construction of cubical simplices appears more natural, such as for the use cases mentioned in Section 1.5.

1.11 Outlook to Persistent Homology

Persistent homology was invented by [ELZ00] in particular to extract robust information from topological patterns with noise. In general, noise is likely to appear on small scales with respect to the relevant topological features. One goal of persistent homology is to make evident these relevant scales. Also the resulting topological invariants should be stable with respect to the noise model. The first step in calculating stable invariants is to construct simplicial complexes for a range of scales instead of one single parameter. This naturally induces a filtered simplicial complex. By drawing an analogy from the indexing set to a time interval, this filtered complex can also be metaphorically viewed as a space changing over time. In this sense, the *persistence modules* (Definition 2.2.4) are the time-dependent analogs of the homology groups and their *barcodes* (Definition **??**) the time dependent analogues of the Betti numbers.

From an algorithmic perspective, there are three main tasks connected to persistence [AJ05]:

- **Persistence**: Find efficient algorithms for computing persistence over arbitrary coefficients. In Chapter 2 we will see two algorithmic approaches to computing persistent homology. Firstly, the *matrix reduction algorithm* in Section **??**, which can be seen as a generalization of the standard algorithm for calculating homology. Secondly, the spectral sequence algorithm in Section **??**, which outputs the same barcode albeit in different order.

- **Topological Simplification**: Find algorithms for simplifying topology based on persistence. This means attributes should be sorted into noise and features within the filtration via increasing persistence. add some text

- **Cycles and Manifolds**: This problem statement includes finding algorithms for computing representations. The persistence algorithm is designed to track the subspcaes that express nontrivial topological attributes and can be modified to identify these subspaces (cycles) as well as subspaces that eliminate them (manifolds). quelle

The theory and the algorithms for persistent homology will be presented in the subsequent Chapter 2.

CHAPTER 2

PERSISTENT HOMOLOGY

La Topologie algébrique vise à étudier les espaces topologiques en leur associant foncotriellement diverses structures algébriques (modules, groupoïdes, etc.) dont les propriétés reflètent celles des espaces considérés.

[Bou16]

Abstract

Persistent (co)homology is presented from an algebraic and algorithmic perspective with a view towards localized homology. Concepts are illustrated with the implementations presented in Section 0.2.2 on the samples of the data from Section 0.2.1.

Contents

2.1 Introduction

There are many excellent resources for an introduction to persistent homology: A first short but comprehensive one can be found in [EH08]. Persistent homology from a (quiver) representation-theoretic point of view is presented in [Ghr14] and [Oud15]. A theoretical and algorithm-heavy introduction can be found in [AJ05]. A reference focusing on shape recognition and feature detection in high-dimensional point cloud data sets is given by [Ghr08]. Also [Car09] is a good starting point for a broader introduction from the data analysis perspective. In [BS14], the authors redevelop persistent homology from a categorical point of view. A brief historical overview is presented in [Per18].

Our current understanding of persistent homology goes back to the work of [Fro92; Rob99]. In the following, we will mention some of the milestones of the last 25 years, which at the same time also correspond to those concepts that we will discuss in more detail within this Chapter 2. This presentation should enable the reader to get a quick broad overview of the points of the field relevant for the present work. The subsequent sections, in which the terms and concepts used are explained more thoroughly, are linked respectively.

2.1.1 Milestones

In general, two approaches to persistent homology can be distinguished: The **geometric-computative** and the **algebraic** one.
The following milestones should be noted in the **geometric-computative track**:

- **1993-2000:** In 1993, [DE95] developed an algorithm for the incremental computation of Betti numbers of time $\mathcal{O}(N\alpha(N))$, where N is the number of simplices and α denotes the inverse Ackermann function ([Cor+09]). Based on this algorithm, [ELZ00] released an algorihm for the efficient computation of Betti numbers for filtered simplicial complexes in the restricted setting of subcomplexes of S^3, and $\mathbb{Z}_2$ coefficients. This can be seen as the first efficient algorithm for the computation of persistent homology, which was extended to generality by [ZC05]. A short presentation of persistent homology algorithms will be given in Section 2.3.

- **2000:** In [ELZ00] introduced the **persistence diagram** (Section 2.2.2) as a visualization tool for persistence.

- **2004:** In [ZC05], **barcodes** (Section 2.2.5) came up as an equivalent way of visualizing persistent homology.

- **2007:** In [CEH05], Cohen-Steiner, Harer and Edelsbrunner introduced a metric between barcodes, the **bottleneck distance** d_B (Section XY) and proved its stability (Theorem) with respect to changes in the input filtration.

On the other hand, the **algebraic track** started, among others, with the work of [ZC05] in 2005, and has recently become a very active research field. The subsequent listing of milestones highlights some of the most important work with a view towards **distributed persistent homology** [YG20; Cas20; TP21], a new approach to persistent homology partly based on old ideas [Ser55] , which brings the local-to-global principle to the algebraic calculation of persistent homology as well as its practical implementation.

- **2005:** In [ZC05], the authors define and study the **persistence module** (Definition 2.2.4) as an algebraic structure that represents the persistent homology of a filtered simplicial complex. As a consequence, they could use the standard Structure Theorem 2.3.1 for finitely generated modules over a principal ideal domain to describe homology groups as a set of intervals. Their work was also a follow up to [ELZ00] and allowed the algorithmic computation of persistent homology for arbitrary dimensional spaces and coefficients in an arbitrary principal ideal domain.

- **2008:** In [Cha+09] the **interleaving distance** d_I (Definition) was introduced as an extended pseudometric between persistence modules. This allowed to reformulate stability in an algebraic manner. Namely, [Cha+09] showed that for a q**-tame** (Table 2.1) persistence module V, the barcode $\mathrm{bcd}(V)$ is well defined. Even more, [Cha+09] related algebraic stability in terms of the interleaving distance between persistence modules to geometric stability in terms of the bottleneck distance between barcodes:

  $$d_B(\mathrm{bcd}(W), \mathrm{bcd}(V)) \leq d_I(W, V), \tag{2.1}$$

  for two q-tame persistence modules W and V.

- **2013:** In [Les15], Lesnick proved that the relation in Eq. (2.1) is in fact an equivalence:

  $$d_B(\mathrm{bcd}(W), \mathrm{bcd}(V)) = d_I(W, V). \tag{2.2}$$

  This result is now formally known as the **Isometry Theorem** (reference) and it states that the algebraic and geometric view on persistent homology are exchangeably equivalent. The interested reader may consult Section XY for more details

- **2008:** In [ZC08] Zomorodian and Carlsson introduced **localized homology**, a tool to (also algorithmically) localize the topological attributes of a topological space X relative to a given cover of X.

- **2013:** In [Cur14], Curry developed cellular (co)sheaves as a tool for topological data analysis. definition

- **2019-2021:** In [YG20], the authors construct a persistence module in a distributed manner, that is isomorphic to the *global* persistence module of interest. Unlike the geometric construction of the Mayer-Vietoris blowup complex presented in Section 2.7, they use cosheaf homology (definition) to combine local information. Their work has partial overlap with [Cas20] who solved the *extension problem* (see Section ref) for distribution covers with nerve dimension higher than 1. Finally, in [TP21] the authors found conditions under which ϵ-interleavings (see Definition ref) exist between the spectral sequences associated to two different covers. Bezug zu Ch 4 herstellen - we will elaborate why this is important with relation to data processed in Autoencoder architectures in Section bla

2.1.2 Overview and Outline

Flow chart graphik für das chapter, vllt mit überschriften und darunter den ausgerechneten Beispielen, Lesefluss beschreiben This Chapter 2 starts with an introduction to persistent homology. With respect to the algorithmic part, a summary of [ZC05], in which the persistent homology algorithm currently used as a standard was developed. Based on this, the algorithms implemented within the computative persistent homology libraries presented in Section 0.2.2, and listed herein in Tables 1, 2 and 3 are also explained. vllt kurzer runtime test für die verschiedenen Daten Examples, implementations locality multiparamter peristence concepts to be explained

2.2 Persistent Homology

Persistent homology aims to calculate the homology groups for a given filtered complex on a data set $\mathcal{S}$. This filtration might arise from a pre-ordered set of scales at which complexes, such as those described in Section 1.8.1 or Section 1.8.4, have been computed. Persistent homology is defined via application of the (simplicial) homology functor (Definition 1.9.3) to the resulting finite filtration (Eq. 1.9). In the following, we will take $\mathbb{R}$ as indexing category, if not otherwise denoted.

Definition 2.2.1 (p-th persistent Homology; Definition 2.6 in [Cas20])**.** *Let K be a simplicial complex filtered over $\mathbb{R}$ and $p \geq 0$. The p-th persistent homology of K is defined as the functor $PH_p(K) : \mathbb{R} \to Vect$.*

This functor is a composition of the filtration functor (Remark 1.4.1) and the simplicial homology functor (Definition 1.9.3), i.e. $PH_n(K) = H_n \circ \mathcal{F}$.

2.2.1 Persistent Homology Groups

In the following, let R be a ring and i be a filtration index, denoted by superscription. The k-persistent p-th homology group of a complex K^i in a filtered complex K is defined as:

$$H_p^{i,k} = Z_p^i / (B_p^{i+k} \cap Z_p^i), \tag{2.3}$$

for $i, k \geq 0$. The k-persistent p-th Betti number of K^i is defined as

$$\beta_p^{i,k} = \text{rank} H_p^{i,k}. \tag{2.4}$$

Persistent homology means to calculate the cycles that are not boundaries for a given dimension and filtration index i and to track them over the subsequential filtration parameters $i+k$. To do so, one would need compatible bases for H_p^i and H_p^{i+k}. To avoid this requirement, [ZC05] introduced persistence modules as algebraic structures that combine the homology of *all* complexes in the filtration into *one* algebraic object.

2.2.2 Persistence Diagrams

The collection of persistent Betti numbers can be visualized in two dimensions via *persistence diagrams*.

Definition 2.2.2 (Persistence Diagram)**.** *Let K be a filtered simplicial complex and $PH_p(K)$ its p-th persistent homology. Then the persistence diagram of $PH_p(K)$, denoted $\mathrm{dgm}_p(K)$ is the collection of pairs (i,j) in $\bar{\mathbb{R}}^2$ with nonzero multiplicity*

$$\mu_p^{i,j} = (\beta_p^{i,j-1} - \beta_p^{i,j}) - (\beta_p^{i-1,j-1} - \beta_p^{i-1,j}) \tag{2.5}$$

for all $i < j$ and all p. Points on the diagonal are added to the diagram with infinite multiplicity.

The number of homology classes that are born at K^i and die entering K^j for a certain homology dimension p is given by $\mu_p^{i,j}$. In this sense, the persistence diagram $\mathrm{dgm}_p(K)$ of a filtered simplicial complex represents a homology class by a point whose vertical distance to the diagonal is its persistence. The persistence diagram encodes all information about persistent homology gropus by the following lemma.

Lemma 2.2.1 (Fundamental Lemma of Persistent Homology, [EH10])**.** *Let $\emptyset = K^0 \subseteq K^1 \subseteq ... \subseteq K^n = K$ be a filtered simplicial complex. For every pair of indices $0 \le i \le j \le n$ and every dimension p, the p-th persistent Betti number is*

$$\beta_p^{i,j} = \sum_{k \le i} \sum_{l \ge j} \mu_p^{k,l}. \tag{2.6}$$

2.2.3 Persistence Modules

In short, persistence modules arise when applying the simplicial homology functor, Definition 1.9, to an object that stores all the chains within the filtration of the complex - the *persistence complex.*

Persistence Complex

Definition 2.2.3 (Persistence Complex)**.** *A persistence complex $\mathcal{C}$ is a family of chain complexes $\{C_*^i\}_{i \ge 0}$ over a ring R, together with chain maps $f^i : C_*^i \to C_*^{i+1}$, such that*

$$C_*^0 \xrightarrow{f^0} C_*^1 \xrightarrow{f^1} C_*^2 \xrightarrow{f^2} \cdots . \tag{2.7}$$

A filtered simplicial complex together with the simplicial inclusion maps constitute an example of a persistence complex. The Diagram below shows a portion of the expansion of Eq. 2.7 for this example.

$$\begin{tikzcd}
{} \arrow[d, "\partial_3"'] & {} \arrow[d, "\partial_3"'] & {} \arrow[d, "\partial_3"'] & \\
C_2^0 \arrow[r, "f^0"] \arrow[d, "\partial_2"'] & C_2^1 \arrow[r, "f^1"] \arrow[d, "\partial_2"'] & C_2^2 \arrow[r, "f^2"] \arrow[d, "\partial_2"'] & \cdots \\
C_1^0 \arrow[r, "f^0"] \arrow[d, "\partial_1"'] & C_1^1 \arrow[r, "f^1"] \arrow[d, "\partial_1"'] & C_1^2 \arrow[r, "f^2"] \arrow[d, "\partial_1"'] & \cdots \\
C_0^0 \arrow[r, "f^0"] & C_0^1 \arrow[r, "f^1"] & C_0^2 \arrow[r, "f^2"] & \cdots
\end{tikzcd}$$

The filtration index increases horizontally to the right, whereas the dimension decreases vertically to the bottom under the boundary operators ∂_p (Equation (1.22)) for a dimension p.

The Category of Persistence Modules

Generalized persistence modules as introduced in [BSS14] are defined as functors $\mathbf{P} \to \mathbf{D}$ from any preordered set (category) $\mathbf{P}$ to any target category $\mathbf{D}$. For the example above, i.e. the homology of a persistence complex as introduced in [ZC05], the indexing category is the category of natural numbers and the target category is **Vect**.

Definition 2.2.4 (Persistence Module, [CCS14]). *Let $(P, \leq)$ be a totally ordered set. The category* **Pers** *of persistence modules over P has the following* ***objects****:*

- *A persistence module V is a functor from P, considered as a category, to* **Vect**. *It consists of indexed vector spaces V_t and linear maps $\rho_{ts} : V_s \to V_t$ for $s \leq t$. The latter are called structure maps and satisfy $\rho_{ts} = \rho_{tu}\rho_{us}$ for all $s \leq u \leq t$ and $\rho_{tt} = 1_{V_t}$ $\forall t$.*

For the morphisms, [CCS14] give the following two alternative definitions:

- *A morphism $\phi : V \to W$ is a natural transformation between functors. Thus, for σ_{ts} being the structure maps for W, it is a collection of linear maps $\phi_t : V_t \to W_t$ such that $\phi_t \rho_{ts} = \sigma_{ts}\phi_s$ for $s \leq t$.*

$$\begin{tikzcd}
V_s \arrow[r, "\rho_{ts}"] \arrow[d, "\phi_s"'] & V_t \arrow[d, "\phi_t"] \\
W_s \arrow[r, "\sigma_{ts}"] & W_t
\end{tikzcd}$$

- *Alternatively, a morphism $\phi : V \to W$ is a collection of linear maps $\phi_{ts} : V_s \to V_t$ for $s \leq t$ such that $\phi_{ts} = \sigma_{tv}\phi_{vu}\rho_{us}$ whenever $s \leq u \leq v \leq t$.*

A persistence module is called *ephemeral* if $\rho_{ts} = 0$ whenever $s < t$. Ephemeral modules form a subcategory **Eph** of **Pers**.

2.2.4 Tameness and Interval Decomposition

Interval Modules

An **interval** in a totally ordered set $\mathbf{T}$ is a subset $J \subseteq \mathbf{T}$ with the following property: If $r \in J$ and $t \in J$ and $r < s < t$, then $s \in J$.

Definition 2.2.5 (Interval Module, [Cha+16])**.** *For any nonempty interval $J \subseteq \mathbf{T}$, the interval module $\mathbb{I} = \mathbb{F}^J$ is defined to be the $\mathbf{T}$-persistence module with vector spaces*

$$I_t = \begin{cases} \mathbb{F} \text{ if } t \in J, \\ 0 \text{ otherwise.} \end{cases} \tag{2.8}$$

and linear maps

$$i_{ts} = \begin{cases} 1 \text{ if } s, t \in J, \\ 0 \text{ otherwise} \end{cases} \tag{2.9}$$

Interval Decomposition

The following Theorem 2.2.1 gives an answer to the question whether a decomposition into interval modules exists for a given persistence module.

Theorem 2.2.1 (Gabriel, Auslander, Ringel-Tachikawa, Webb, Crawley-Boevey, [Cha+16])**.** *Let V be a persistence module over $\mathbf{T} \subseteq \mathbb{R}$. Then V can be decomposed as a direct sum of interval modules under the following conditions:*

1. *Each V_t is finite dimensional; or*
2. *$\mathbf{T}$ is a finite set.*
3. *In contrast to that, there exists a persistence module over $\mathbb{Z}$ which cannot be decomposed into interval modules.*

Proof. 1. The first statement follows from Gabriel's theorem [ZC05; CS08].

2. The second statement was proven in Theorem 3 of [Web85].
3. Three examples can be found in [[Cha+16], Theorem 2.8], one of which is the following: Let $W = \prod_{n \geq 0} \mathbb{F}[-n, 0]$ be a persistence module with

$$\begin{aligned} W_0 &= \{\text{sequences} \quad (x_1, x_2, ...) \quad \text{of scalars}\} \\ W_{-n} &= \{\text{such sequences with} \quad x_1 = ... = x_n = 0\} \quad (n \geq 1). \end{aligned} \tag{2.10}$$

The structure maps $\sigma_{-m,-n}$ are taken to be the canonical inclusion maps. Since the structure maps $\sigma_{-n,-n-1}$ are injective, all of the intervals must be of the form $[-n, 0]$ or $(-\infty, 0]$. The multiplicity of each $[-n, 0]$ is 1, wheareas the multiplicity of $(-\infty, 0]$ is zero. If the latter would not hold, there would be a nonzero element of W_0 in the image of $\sigma_{-n,0}$ for all $n \geq 0$. However, this is impossible as $\bigcap_{n \geq 0} W_{-n} = \{0\}$. This implies that $W \cong \bigoplus_{n \geq 0} \mathbb{F}[-n, 0]$. However, the dimension of W_0 is uncountable and thus the module W does not exhibit an interval decomposition.

$\square$

A persistence module that is a direct sum of *finitely many* interval modules is most commonly called 'tame'. If a persistence module can be expressed via a direct sum of interval modules, then the Krull-Remak-Schmidt-Azumaya theorem guarantees that the intervals appearing within this decomposition represent invariants.

Theorem 2.2.2 (Krull-Remak-Schmidt-Azumaya, [Cha+16])**.** *Suppose there exist two different ways for expressing a persistence module over* $\mathbf{T} \subseteq \mathbb{R}$ *as a direct sum of interval modules:*

$$V \cong \bigoplus_{l \in L} \mathbb{F}^{J_l} \cong \bigoplus_{m \in M} \mathbb{F}^{K_m}. \tag{2.11}$$

Then there is a bijection $\sigma : L \to M$ *such that* $J_l = K_{\sigma(l)}$ *for all* l*.*

Proof. Two approaches for a proof can be found in [[AF92], Theorem 12.6] and [[Fac98], Theorem 2.12], which the reader may consult for full detail. In the following, let R be a ring, M be an R-module and $\aleph$ a cardinal.

- **Idea of Proof in [AF92]:**
  Let $M = \bigoplus_{i \in I} M_i$ be a direct indecomposable sum decomposition for M. A direct summand M' of M is maximal if M' has an indecomposable complement in M. A direct sum decomposition $M = \bigoplus_{i \in I} M_i$ is said to complement direct summands if for every direct summand M' of M, there exists $J \subseteq I$ with $M = M' \oplus (\bigoplus_{i \in J} M_i)$. Let each endomorphism ring $\mathrm{End}(M_i)$ be local. The locality property of the endomorphism ring is needed in [AF92] to prove that the decomposition $M = \bigoplus_{i \in I} M_i$ complements maximal direct summands (see [AF92], p. 144ff. for more details). From the indecomposability of the direct summands then follows that two direct sum decompositions of such a module M are isomorphic [[AF92], Theorem 12.4].

- **Idea of Proof in [Fac98]:**
  The proof in [[Fac98], Theorem 2.12] is based on the *exchange property* which by Theorem 2.8 in [Fac98] is equivalent to the locality of the endomoprhism ring of M for M being indecomposable. M is said to have the $\aleph$-exchange property if for any R-module G and any two direct sum decompositions
  $$G = M' \oplus N = \bigoplus_{i \in I} A_i, \tag{2.12}$$
  where $M' \cong M$ and $|I| \leq \aleph$, there are R-submodules B_i of A_i, $i \in I$, such that $G = M' \oplus (\oplus_{i \in I} B_i)$. M has the exchange property, if it has the $\aleph$-exchange property for every cardinal $\aleph$. It is said to have the *finite* exchange property if it has the $\aleph$-exchange property for every finite cardinal $\aleph$. Let $M = \bigoplus_{i \in I} M_i$ and $M = \bigoplus_{j \in J} N_j$ be two direct sum decompositions of M, where all M_i and N_j are indecomposable. The second

decomposition is a *refinement* of the first if there is a surjective mapping $\phi : J \to I$ such that $\bigoplus_{j\in\phi(i)} N_j = M_i$ for every $i \in I$. For $I' \subseteq I$ and $J' \subseteq J$ let

$$M(I') = \bigoplus_{i\in I'} M_i \quad \text{and} \quad N(J') = \bigoplus_{j\in J'} N_j. \tag{2.13}$$

Using the exchange property, it is proven in [Fac98] that the two decompositions $M(I') = \bigoplus_{i\in I'} M_i$ and $N(J') = \bigoplus_{j\in J'} N_j$ have isomorphic refinements. From the indecomposability of the M_i and N_j then follows that there exists a one-to-one correspondence $\phi : I' \to J'$ such that $M_i \cong N_{\phi(i)}$ for every $i \in I'$. For every R-module A set

$$I_A = \{i \in I \mid M_i \cong A\} \quad \text{and} \quad J_A = \{j \in J \mid N_j \cong A\}. \tag{2.14}$$

The rest of the proof consists in showing that $|I_A| = |J_A|$ for the finite and infinite case.

- *Application to Tame Persistence Modules*
Tame persistence modules have a direct sum decomposition into interval modules by definition. The endomorphism ring of interval modules over $\mathbf{T} \subseteq \mathbb{R}$, $\mathrm{End}(\mathbb{I}) = \mathbb{F}$ (Proposition 2.5 in [Cha+16]) and thus local. From this follows that interval modules are indecomposable (Proposition 2.6 in [Cha+16]). Thus tame persistence modules fullfill the requirements of Theorem 12.6 in [AF92] and Theorem 2.12. in [Fac98] and hence are decomposed uniquely into interval modules.

□

Q-Tameness

There are different conditions for *tameness* in persistence modules, a comparison of which is given in Section 3.8 of [Cha+16]. A summary is shown in Table 2.1 below. Many of the persistence modules that occur when following the pipeline in Figure 1.1 are not of finite type or 'tame' in the above mentioned sense, but q-tame.

Example 2.2.1. *Examples for q-tame persistence modules appearing in topological data analysis include*

- *the realization of a simplicial complex K as a topological space X, filtered via a continuous function $f : X \to \mathbb{R}$ ([Cha+16], Theorem 3.33).*
- *Also, in [CSO13], Proposition 5.1 it is proven that the Vietoris-Rips and Cěch homology of totally bounded sets (and thus finite subsets of $\mathbb{R}^d$) is q-tame.*

The diagram of implications between the different tameness conditions mentioned in Table 2.1 is as follows [Cha+16]:

$$\text{finite type} \Rightarrow \text{locally finite} \Rightarrow \text{pfd} \Rightarrow q\text{-tame}. \tag{2.15}$$

Proposition 2.2.1. *The inclusions in Eq.* (2.15) *are strict.*

Proof. ausfuehrlicher In [Dro12] an example is given for the Vietoris-Rips homology of a compact metric space that is q-tame but not pointwise finite-dimensional. □

Level of Tameness	Description
finite type	A persistence module is of finite type if it is a direct sum of finitely many interval modules.
locally finite	A persistence module is locally finite if it is a direct sum of interval modules and any bounded subset of **P** meets only finitely many intervals.
pointwise finite-dimensional (pfd)	A persistence module is pointwise finite-dimensional if each vector space V_t is finite-dimensional.
q-tame	Q-tame persistence modules have finite-rank structure maps ρ_{ts} whenever $s < t$.

Table 2.1: Levels of tameness for persistence modules.

Persistence modules of finite type and locally finite persistence modules are interval decomposable by definition. By [Cra14], Theorem 1.1, also any pointwise finite-dimensional persistence module is a direct sum of interval modules. However, this may not be the case for q-tame persistence modules. The presentation of q-tameness in [Cha+16] relies on a measure-theoretic understanding, on which three further levels of tameness can also be based: h-tameness, v-tameness and r-tameness. The interested reader may find details about h-,v-or r-tameness in Section 3.8 et seq. of [Cha+16]. These tameness conditions ascertain the existence of *persistence diagrams* (Definition 2.2.2) over certain subsets of the extended half-plane. The four conditions differ at infinity, while except for the diagonal, the finite part is always included. Q-tameness guarantees the existence of a persistence diagram over the set

$$\{(p^*, q^*) | -\infty \leq p < q \leq +\infty\}. \tag{2.16}$$

The Category Obs

Although a persistence diagram is definable for every q-tame persistence module, this is not a complete invariant. This means that two non-isomorphic q-tame persistence modules can have the same persistence diagram. Furthermore, q-tame persistence modules do not fall under the scope of Theorem 2.2.1. In [CCS14] it is shown that q-tame persistence modules do have a uniquely defined persistence diagram and an interval decomposition, when **Pers** is substituted by a quotient category **Obs**. This category is obtained from **Pers** via Serre localization at the subcategory **Eph** of ephemeral modules.

2.2.5 Barcodes

Definition 2.2.6 (Barcode)**.** *Let $\Omega = \{(b,d) \in \mathbb{R}^2 : d > b\}$ denote the upper-diagonal part of the real plane. A barcode is a multi-set over Ω.*

Barcodes are another way of visualizing persistent homology where instead of points, persistence pairs (i,j) are depicted as intervals.

2.3 Persistent Homology Algorithms

Key idea to the subsequently described algorithm of [ZC05] for the computation of bases for the groups in Eq. (2.3) is the application of the **Structure Theorem** to **persistence modules**. The specialization of this algorithm to field coefficients is denoted nowadays as the **standard algorithm**. pcoh and prow erklraeren weil das nach "dualities in persistent homology" für den dualen algorithmus gebraucht wird

Theorem 2.3.1 (Structure Theorem, 12.1.5 in [DF03])**.** *If D is a PID, then every finitely generated D-module is isomorphic to a direct sum of cyclic D-modules. It decomposes uniquely into the form:*

$$D^\beta \oplus \left(\bigoplus_{i=1}^{m} D/d_i D \right), \tag{2.17}$$

where $d_i \in D$, $\beta \in \mathbb{Z}$, such that $d_i | d_{i+1}$.

Thus finitely generated D-modules are decomposed into two parts: A *free* portion on the left and the *torsional* portion on the right of Eq. (2.17). Before a version of this Theorem can be used for *persistence* homology modules, we will first describe its direct application to homology groups.

In Section 1.9.1, p-dimensional homology groups were defined as a quotient of Abelian groups Z_p/B_p, whose elements are classes of homologous cycles. These also can be interpreted as modules over the integers as a ring. Consequently, this view allows exchanging the ground ring of coefficients. If the ground ring is a PID, the Betti numbers, first introduced in Section 1.3.2 as the dimension of the p-th homology group bei den Homology gruppen n und p fixen, get also a module-theoretic interpretation: The Betti number β_p of the p-th homology group, interpreted as a module, is the rank of its free submodule in the decomposition of Theorem 2.3.1. On the other hand, d_i are its *torsion coefficients*, also called *invariant factors* [DF03]. While Betti numbers have the interpretation outlined already in Section 1.3.2, namely of counting *holes* of a certain dimension, torsion captures features such as non-orientability of a surface. An example including torsion is given by the Klein bottle. vllt bsp auch mit persistence diagram und bildchen

2.3.1 Torsion

symbole fixen, X topologischer raum etc vllt am anfang des kapitels Let X be a topological space and $\mathcal{R}$ be the ring of integers $\mathbb{Z}$. By the *fundamental theorem of finitely generated abelian groups*

When considering field coefficients such as $\mathbb{Z}/p\mathbb{Z}$, the torsion part in Eq. (2.17) vanishes, such that

$$H_p(X, \mathbb{F}) \cong \mathbb{F}^{\beta_p(\mathbb{F})}. \tag{2.18}$$

Given a topological space X, the **universal coefficient theorem for homology** establishes a relationship between integral homology $H_*(X, \mathbb{Z})$ and homology $H_*(X, \mathbb{Z}/p\mathbb{Z})$ with coefficients in the field $\mathbb{Z}/p\mathbb{Z}$.

Theorem 2.3.2 (Universal Coefficient Theorem for Homology, [[Hat02], Theorem 3A.3)**.** *] Let G be an abelian group and C be a chain complex of free abelian groups with homology groups denoted by $H_n(C)$ for a dimension n. Tensoring C with G gives another complex $C \otimes G$ whose homology groups are denoted by $H_n(C; G)$. Then there are natural short exact sequences*

$$0 \longrightarrow H_n(C) \otimes G \longrightarrow H_n(C; G) \longrightarrow \mathrm{Tor}(H_{n-1}(C), G) \longrightarrow 0$$

for all n and all G and these sequences split.

All implementations mentioned in Section 0.2.2 use fields $\mathbb{Z}/p\mathbb{Z}$ as the ground ring and thus do not have to deal with torsion.

2.3.2 Reduction Algorithm

The algorithm developed in [ZC05] is a modification of the previously known **reduction algorithm** ([Mun84]) for the calculation of standard simplicial homology over a PID, which will presented subsequently for clarity. The aim of the homology algorithm is to diagonalize the boundary matrix M_p. In other words, the boundary matrix is brought into a form so that one can read off the basis elements of the chain complex $C_p(K)$ generating the homology group $H_p(K)$.

Definition 2.3.1 (Smith Normal Form)**.** *For a matrix $A : \mathbb{F}^m \to \mathbb{F}^n$ with rank r, there exist invertible matrices $P : \mathbb{F}^n \to \mathbb{F}^n$ and $Q : \mathbb{F}^m \to \mathbb{F}^m$ such that*

$$D = PAQ. \tag{2.19}$$

The matrix D is $n \times m$ diagonal with entries given by matrixeintrag notation fixen

$$D_{ij} = \begin{cases} 1 & \text{if } i = j \leq r, \\ 0 & \text{otherwise.} \end{cases} \tag{2.20}$$

This matrix D is called the Smith normal form of A.

Within the Smith decomposition, the first r columns of the inverse P^{-1} form a basis for $\mathrm{im}A \subset \mathbb{F}^n$ while the last $(m-r)$ columns of Q form a basis of $\ker A \subset \mathbb{F}^m$. Calculating the Smith decompositions of the boundary matrices M_p for each homology dimension p allows to find the basis vectors for the homology groups H_p as explained in the following proposition, [Proposition 3.15 in [Nan20]].

Proposition 2.3.1. *Let $D_p = P_p M_p Q_p$ be the Smith decomposition of M_p for the p-dimensional boundary operator $\partial_p : C_p \to C_{p-1}$. For $n_p = \dim C_p$ and $r_p = \mathrm{rank}\partial_p$, let G_p be the matrix of size $(r_{p+1} + n_p - r_p) \times n_p$ given by the block decomposition*

$$G_p = [B_p|Z_p]. \tag{2.21}$$

The submatrix B_p equals the first r_{k+1} columns of P_{k+1}^{-1} while Z_p equals the last $(n_p - r_p)$ columns of Q_p. The matrix G_p is brought into reduced row echelon form $E_p = [B'_p|Z'_p]$ by performing elementary row operations over $\mathbb{F}$, see Section 2.5 in [ZC05] for a detailed description. Then the columns of Z_p corresponding to the pivot columns of Z'_p form a basis for H_p.

Analogously, one can also manipulate the columns to reduce the boundary matrix. The above proposition only holds for field coefficients. For integer coefficients, one cannot neglect the torsion part which introduces a more interesting structure to the Smith normal form of the boundary matrix. A detailed discussion of this case with an example can be found in [[ZC05], Section 2.5].

2.3.3 Correspondence and Decomposition

To apply the above reduction algorithm to persistence modules, we also need an adapted version of the structure theorem. In Theorem 3.1 of [ZC05] it was proven that the category of persistence modules of finite type over R and the category of finitely generated non-negatively graded modules over $R[t]$ are equivalent. Since persistence modules represent a structure that includes the simplicial complexes over the entire filtration parameters, this means that the time at which a simplex is created in this filtration is associated to this structure by means of a polynomial coefficient. When the ground ring R is a field, the structure of persistence modules can thus be described via the standard structure theorem for graded modules over a graded PID (see Theorem 2.1 in [ZC05]):

$$\left(\bigoplus_{i=1}^{n} \Sigma^{\alpha_i} F[t]\right) \oplus \left(\bigoplus_{j=1}^{m} \Sigma^{\gamma_j} F[t]/(t^{n_j})\right). \tag{2.22}$$

In the above Eq. 2.22, Σ^α denotes an α upward shift in grading. describe The isomorphism classes of $F[t]$-modules now have a special kind of parametrization, that also allows for a descriptive and intuitive interpretation, as will be discussed in Section XY.

Definition 2.3.2 ($\mathcal{P}$-interval, Definition 3.4 in [ZC05])**.** *A $\mathcal{P}$-interval is an ordered pair (i, j) with $0 \leq i < j \in \mathbb{Z}^\infty = \mathbb{Z} \cup +\infty$.*

There exists a bijection Q to associate a graded $F[t]$-module to a set $\mathcal{S}$ of $\mathcal{P}$-intervals. For one $\mathcal{P}$-interval (i, j), this bijection Q is defined as

$$Q(i, j) = \Sigma^i F[t]/(t^{j-i}), \tag{2.23}$$

with $Q(i, +\infty) = \Sigma^i F[t]$. For the whole set $\mathcal{S}$, one obtains

$$Q(\mathcal{S}) = \bigoplus_{l=1}^{n} Q(i_l, j_l). \tag{2.24}$$

Via the correspondence $\mathcal{S} \to Q(\mathcal{S})$ thus results a bijection between the finite sets of $\mathcal{P}$-intervals and the finitely generated graded modules over the graded ring $F[t]$. As a result, the isomorphism classes of persistence modules of finite type over F are in bijective correspondence with the finite sets of $\mathcal{P}$-intervals. This result is now known as the **Fundamental Theorem of Persistent Homology**, [Ott+17]. It follows that for a filtered simplicial complex K with subcomplexes K^i in the filtration, there is a choice of basis vectors of $H_p(K^i)$ such that one can construct a diagram of a well-defined and unique collection of disjoint half-open intervals, known as the **barcode**, which will be described below in Section XY. todo: triangle region verstehen, p8 in zomorodian barcode schon oben eingeführt

2.3.4 Standard Algorithm for Fields

In each homology dimension p, the homology of a complex K_i at filtration parameter i becomes a vector space over a field, which is fully described by its Betti number β_p^i. The persistence module corresponding to K is then a direct sum of all vector spaces in the filtration. The structure theorem guarantees the existence of compatible bases of these vector spaces over the filtration to compute its entire homology. Each $\mathcal{P}$-interval (i, j) describes a basis element for the homology vector spaces, persistent from time i until time $j - 1$. This element corresponds to a p-cycle e that created at time i and joined to the boundary group B_p^j at time j.

Definition 2.3.3 (Boundary Matrix of a Filtered Simplicial Complex)**.** *Let $\{\sigma_1, \sigma_2, ..., \sigma_N\}$ be the ordered simplices of a filtered simplicial complex K. Then the boundary matrix is defined as an $N \times N$ matrix D described as follows. Let (i, j) be an indexing pair in $\{1, ..., N\}^2$, then the entries of D at (i, j) are given as:*

$$D_{ij} = \begin{cases} \pm 1 & \text{if } \sigma_i \leq \sigma_j \quad \text{with } \dim\sigma_j - \dim\sigma_i = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{2.25}$$

The sign $\pm$ depends on the ordering of the vertices of K and can be ignored for $\mathbb{Z}/2\mathbb{Z}$.

To determine the characteristic intervals, one has to transform the boundary matrix to *echelon form* (see Lemma 4.1 in [ZC05]), so that one can read off these intervals directly. There are two ways to do this: either via *row* or *column* operations. These two decompositions give identical outputs, as proven in [[SMV11], **Theorem 3.1**]. Adapting the notation of [SMV11], we denote these two algorithms as **pHcol** and **pHrow**. Probably more standard is the column-form **pHcol**, which was also introduced as such in [ZC05]. Basically, it is just Gaussian elimination on the boundary matrix performed using column operations. It will be described in more detail in the following. For this purpose, one needs a helper function low(j) to be the largest index value i such that $D_{i,j} \neq 0$ [Ott+17]. This is also called a **pivot**.

Algorithm 1 Standard Algorithm for Boundary Matrix Reduction to Barcodes

```
for j=1 to n do
  while there exists i<j with low(i)=low(j) do
    add column i to column j
  end while
end for
```

Once this algorithm has terminated and the boundary matrix is given in reduced form, one can read off the intervals of the barcode by pairing the simplices in the following way [Ott+17]:

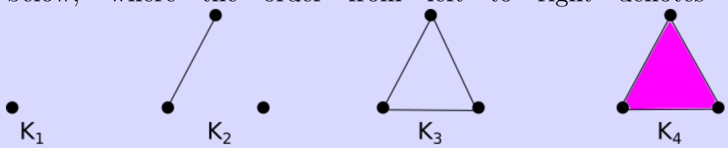
- If low$(j) = i$ the simplex σ_j is paired with σ_i and the entrance of σ_i causes the birth of a feature that dies with the entrance of σ_j. Such a simplex is called a *creator simplex*.

- If low(j) is undefined, then the entrance of the σ_j causes the birth of a feature, that might be destroyed by a simplex σ_k of low$(k) = j$. Such a simplex is called a *destroyer simplex*. If no such σ_k exists, σ_j will remain unpaired. These kind of simplices are called *essential*.

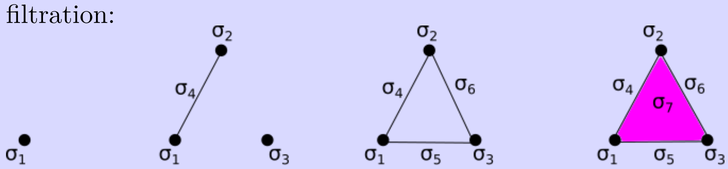
A pair (σ_i, σ_j) corresponds to a half-open interval in the barcode, whereas an unpaired simplex results in an half-open infinite interval. An example for this algorithm for a very simple filtration adapted from [Ott+17] is illustrated in the following.

Example: PH Calculation with the standard algorithm after Section 5.3.1 in [Ott+17]

[ht] (a) We start with the filtered simplicial complex depicted below, where the order from left to right denotes inclusion:



(b) A total order is added on the simplices that is compatible with the filtration:



where σ_i denotes the ith simplex in this order.

(c) On the left side below is the boundary matrix M for the filtered simplicial complex in (a) with respect to the orientation given in (b). On the right side is the reduced column-echelon form $\bar{M}$:

$$M = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \bar{M} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{2.26}$$

where coefficients were taken in $\mathbb{Z}_2$.
(d) The intervals can be read off the matrix $\bar{M}$ as follows:

- σ_1 is positive and unpaired giving the interval $[1, \infty)$ in H_0.
- σ_2 is positive and paired with σ_4 giving no interval, as σ_2 and σ_4 enter the filtration at the same time.
- σ_3 is positive and paired with σ_5 giving the interval $[2, 3)$ in H_0.
- σ_6 is positive and paired with σ_7 giving the interval $[3, 4)$ in H_1.

Algorithmus beschreiben, mit illustrativem Beispiel aus Otter was zur runtime vom Algorithmus sagen

2.3.5 Algorithmic Modifications

die Algorithmen in den Implementation beschreiben =¿ inwiefern weichen sie vom Standard Algorithmus ab, was gibt es noch, vlllt auf die runtime von der barcode berechnung verweisen As the reader may have noticed, none of the implementations presented in Section 0.2.2 uses the standard algorithm directly. Rather they use some modifications to speed up the calculations and make high dimensional data more accessible. In the following, we will briefly present these modifications, namely: The dual algorithm, the dual algorithm via simplicial maps, the multifield dual algorithm and the twist algorithm. flagser?

Dual Algorithm

Analogous to the calculation of relative homology in Section 1.9, relative persistent homology can also be calculated transferring from the Definition in Eq. (2.3). Hence, there arise four natural persistent objects calculable from a filtered space X [SMV11]:

- persistent absolute (co)homology
- persistent relative (co)homology

By functoriality of simplicial homology, there is a standard duality interchanging homology and cohomology. In our case of field coefficient simplicial homology, this is just a vector space duality called **pointwise** duality in the following. However, as proven in [SMV11], there exists another so-called **global** duality:

$$\begin{aligned} \text{absolute homology} &\leftrightarrow \text{relative cohomology} \\ \text{absolute cohomology} &\leftrightarrow \text{relative homology} \end{aligned} \tag{2.27}$$

This global duality holds also algorithmically, meaning that an algorithm producing as output the persistent relative homology also computes the persistent absolute cohomology and vice versa (Corollaries 2.7 and 2.9 in [SMV11]). Let D be the strictly upper-triangular matrix whose entries $D_{i,j}$ are the coefficients of the boundary map acting on the persistence complex associated to X. This matrix represents the chain complex for persistent absolute homology. Flipping the matrix D across its minor diagonal gives the **anti-transpose** $D^{\perp}$ whose entries are defined as $D^{\perp}_{i,j} = D_{n+1-j,n+1-i}$. This represents the cochain complex for persistent cohomology relative to X_{n-i} in the filtration of a complex

$$X: \; X_1 \subset X_2 \subset ... \subset X_n = X_\infty, \tag{2.28}$$

associated to X by simplification. From this duality follows that any algorithm applied to the matrix D that computes the intervals and generators of persistent absolute (relative) homology, when applied to the matrix $D^{\perp}$, gives the intervals and generators of persistent relative (absolute) cohomology. Hence the cohomology algorithm **pCoh** presented in [SMV11] is based on **pHrow** when applied to $D^{\perp}$. Thus, it is based on the calculation of the coboundaries. This results in a speed up of calculations, as pointed out and verified in [SMV11]. The logic of the standard algorithm is based on the fact that a simplex can only enter the filtration at an index i if all of its faces are already present. However, this causes also the necessity that dead cycles have to be stored, because in principle, they could form the faces of a newly born cycle in the future. Contrary to this, **pCoh** goes 'back in time' by associating faces to cycles and thus can drop a column once it has been used in the update. The dual algorithm is implemented in ripser and thus via the ripser backend also in giotto-tda.

fehlt: dual via simplicial maps and multifield dual

Dual Algorithm via Simplicial Maps

In Section Section über rips complexe umschreiben, section 1.4.6 theoretical and computative difficulties it was demonstrated how the size of a Rips filtration becomes a bottleneck due to the inclusive nature of Rips complexes. Subsequently, alternative complexes such as Witness and GIC complexes were presented, which constitute sparsified versions thereof. Via this sparsification, the original sequence of Rips complexes connected by inclusion maps gets approximated by a sequence of sparsified Rips complexes or even sparser GIC complexes, which are connected via simplicial maps $f : K_i \to K_{i+1}$ Symbol fuer inclusion maps fixen. For the latter, standard persistent homology algorithms do not apply directly any more. In other words, the object of Eq. 2.7 gets more complicated and an explicit representation of the induced homomorphisms between the homology groups of two consecutive complexes in a sequence, $f_* : H_*(K_i) \to H_*(K_{i+1})$, is needed. As introduced in [CS08], this can be done by generating an intermediate complex $\hat{K}$ from K_i such that the simplicial map can be represented by inclusions. However, the computation of this intermediate complex has $\mathcal{O}(n^4)$-complexity in contrast to cubical complexity for the standard computation. Thus, in [DFW12] an algorithm is presented that allows to simulate simplicial maps via a sequence of inclusions and vertex collapses (i.e. deletions) in monotone directions without the need for such an intermediate complex. ein wenig mehr details mit formel um uli glücklich zu machen - denn Mathematik ist ja nicht PROSAISCH XD Key idea is...

Multifield Algorithm

The multifield algorithm allows the simultaneous computation of persistent homology over various coefficient fields.

Twist Algorithm

The twist algorithm introduced in [CK11] adds a speedup to the standard algorithm **pHcol** without matrix reduction. It is based on the dimension-wise deletion of creator columns to reduce the boundary matrix without pivot operations. For a column j in the reduction, $\mathrm{low}(j) = i$ means that σ_j kills the homology class created by σ_i. Thus the corresponding column i can be deleted.

For the straight-forward reduction order, the deletion of column i means no speedup, as it is already identified as a creator and thus not needed any more. However, when the reduction order is changed so that one proceeds in decreasing dimensions, starting at $d = \dim K$, the identification of a pivot i corresponds to a simplex σ_i of dimension $\dim\sigma_j - 1$, which by the new order has not been used in the reduction yet. Thus its deletion reduces the number of columns to be processed by the standard algorithm.

2.4 Persistence

Persistence ranks topological attributes by their lifetime in filtration. From an algorithmic perspective, there are three main tasks connected to persistence [AJ05]:

- **Persistence**: Find efficient algorithms for computing persistence over arbitrary coefficients.
- **Topological Simplification**: Find algorithms for simplifying topology based on persistence. This means attributes should be sorted into noise and features within the filtration via increasing persistence.
- **Cycles and Manifolds**: This problem statement includes finding algorithms for computing representations. The persistence algorithm is designed to track the subspcaes that express nontrivial topological attributes and can be modified to identify these subspaces (cycles) as well as subspaces that eliminate them (manifolds).

2.5 Persistent (Co)Homology

2.6 Locality

2.6.1 The Localization Problem

Taking one step back from the local-to-global perspective outlined above, determining the location of topolocial features within a space am Anfang der Arbeit eine Definitionsliste: was ist mit space gemeint etc is important for several reasons:

1. undersampling and noise;
2. a wide scale variance in the dataset;
3. non-uniform distribution of data;
4. need for fast computation;
5. proper design of compression algorithms.

This is called the **localization problem**, [ZC08]. Some of the above mentioned points are addressed by distributed persistence but can also be overcome by **multiparameter persistence** developed in [Fro92; CZ07]. Multiparameter persistence describes the branch of TDA which studies data via **multifiltrations**. The latter comprises diagrams of spaces (see Definition 1.4.13) where the indexing category $\mathcal{I}$ equals a product of n totally ordered sets, e.g. $\mathbb{R}^n$ or $\mathbb{N}^{\mathrm{op}} \times \mathbb{R}$.

Examples [BL23]

Beispiele besser, mit Literaturverweisen

- **Data density** can be used as a second parameter for persistence, as standard 1-parameter persistence is notoriously unstable to outliers.
- The **values of a real-valued function** can also be used as a second parameter. For example, for time varying data one would take time as a second coordinate. Other examples can be found in biology, where the spatial structure of proteins is encoded in a real-valued function.

Hence, multiparameter persistence and the distributed calculation of persistent homology have a concurrent objective in that the global view of the data and the global computation of algebraic invariants via 1-parameter filtrations may result in the geometry of the data space not being properly captured. Multiparameter persistence is an open field in its own right, as it turned out that the natural generalization of a barcode to the multiparameter case is to complicated to work with in practice. This gets more illucidated by taking the quiver representation theoretic view on the problem, which is discussed on an introductory level in Section 8 of [BL23].

2.7 The Mayer-Vietoris Blowup Complex

needed for Chapter 4, [TP21]

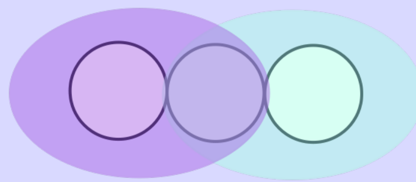
As the worst case time and space complexities for the computation of persistent homology are $\mathcal{O}(n^3)$ and $\mathcal{O}(n^2)$, with n the total number of simplices in the filtration, this shows that sparsified versions of complexes as well as data partition and parallellism are highly needed to remain computable. Especially in view of the fact that the size of the datasets processed in deep learning is much larger than what is usual in TDA application. Unfortunately, the literature and even more the implementation of parallel computing in topology is comparably young. One approach developed by [LZ14] relies on the **Mayer-Vietoris blowup complex**, a spatial version of the **Mayer-Vietoris spectral sequence** [ZC05] (the latter is further explained in Chapter 2). Parallelized homology firstly requires a decomposition of the space into pieces and secondly a recipe to put these parts together to a whole again. When a cover is used to decompose a space into pieces - corresponding to subspaces and their various intersections - this recipe is given by the Mayer-Vietoris spectral sequence which expresses the relationship between the homology of these subspaces to the homology of the space itself. The Mayer-Vietoris blowup complex is a topological space which encodes the data given as input to the spectral sequence and whose homology groups are isomorphic to those of the original space.

Definition 2.7.1 (Blowup Complex [LZ14])**.** *Given a simplicial complex K and a cover $\mathcal{U} = \{U_i\}_{i\in\{1,n-1\}}$ of n subcomplexes, let $K^J = \cap_{k\in J} U_k$ for $J \subseteq \{1, n-1\}$. The Mayer-Vietoris blowup complex $K^{\mathcal{U}} \subseteq K \times \Delta^{n-1}$ of K and $\mathcal{U}$ is defined as:*

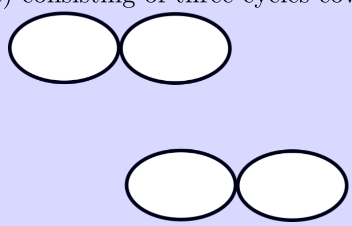
$$K^{\mathcal{U}} = \bigcup_{\emptyset \neq J} K^J \times \Delta^J, \tag{2.29}$$

where $\times$ denotes the Cartesian product and Δ^J is a face of the nerve $\mathcal{N}(\mathcal{U})$.

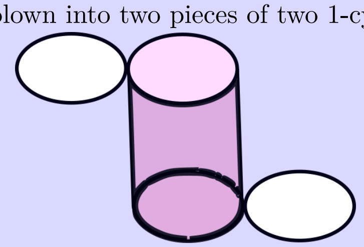
Example taken from [ZC08]



(a) Graph (space) consisting of three cycles covered by two sets.



(b) The graph is blown into two pieces of two 1-cycles each at $t = 0$.



(c) Blowup at $t = 1$.

The middle cycle of the original space in (a) is contained in the intersection of the cover sets, which makes it appear in both local pieces in step (b). To recover the global topology, the two copies of the middle cycle are equated by gluing a cylinder to them which results in construction (c). This is called the *Mayer-Vietoris blowup complex*, which has the same number of cycles as the original space but also incorporates the geometric cover information within its structure.

The Mayer-Vietoris blowup complex is defined as a subspace of a product space, which is triangulated with a simplicial complex as described in [Ito87], page 262. The homology of the product space is obtained from the homology of its factors through the Alexander-Whitney map. As was illustrated in the example above, to construct the blowup complex, one does not tear or glue, but only stretch certain pieces. The blowup complex has the same topology as the original space for open coverings of spaces homeomorphic to finite CW-complexes and also for coverings of finite simplicial complexes by subcomplexes (Lemma 1 in [ZC08]).

CHAPTER 3

MACHINE LEARNING AND AUTOENCODERS

BIBLIOGRAPHY

[Ada+15] Henry Adams et al. "Persistence Images: A Stable Vector Representation of Persistent Homology". In: (2015). DOI: 10.48550/ARXIV.1507.06217. URL: https://arxiv.org/abs/1507.06217.

[Ada23] Adams, Henry. *What applied software options do I have?* https://www.math.colostate.edu/~adams/advising/appliedTopologySoftware/, Last accessed on 2023-03-10. 2023.

[AF92] Frank W. Anderson and Kent R. Fuller. *Rings and Categories of Modules*. 2nd ed. Springer-Verlag, 1992.

[AGR17] Nieves Atienza, Rocio Gonzalez-Diaz, and Matteo Rucco. *Persistent Entropy for Separating Topological Features from Noise in Vietoris-Rips Complexes*. 2017. arXiv: 1701.07857 [cs.OH].

[AGS20] Nieves Atienza, Rocio Gonzalez-Dıéaz, and Manuel Soriano-Trigueros. "On the stability of persistent entropy and new summary functions for topological data analysis". In: *Pattern Recognition* 107 (Nov. 2020), p. 107509. DOI: 10.1016/j.patcog.2020.107509. URL: https://doi.org/10.1016%2Fj.patcog.2020.107509.

[AJ05] Zomorodian A.J. *Topology for computing*. CUP, 2005. ISBN: 0511082207.

[Ale28] P. Alexandroff. "Über den allgemeinen Dimensionsbegriff und seine Beziehungen zur elementaren geometrischen Anschauung". In: *Mathematische Annalen* 98 (1928), pp. 617–635. URL: http://eudml.org/doc/159232.

[Ark10] Martin Arkowitz. *Introduction to Homotopy Theory*. Springer, 2010.

[Bau21] Ulrich Bauer. "Ripser: efficient computation of Vietoris-Rips persistence barcodes". In: *J. Appl. Comput. Topol.* 5.3 (2021), pp. 391–423. ISSN: 2367-1726. DOI: 10.1007/s41468-021-00071-5. URL: https://doi.org/10.1007/s41468-021-00071-5.

[BDM13] Jean-Daniel Boissonnat, Tamal K. Dey, and Clément Maria. "The Compressed Annotation Matrix: an Efficient Data Structure for Computing Persistent Cohomology". In: (2013). DOI: 10.48550/ARXIV.1304.6813. URL: https://arxiv.org/abs/1304.6813.

[BGO07] Jean-Daniel Boissonnat, Leonidas Guibas, and Steve Oudot. "Manifold Reconstruction in Arbitrary Dimensions Using Witness Complexes". In: *Discrete and Computational Geometry* 42 (Jan. 2007). DOI: 10.1145/1247069.1247106.

[Bjö03] Anders Björner. "Nerves, fibers and homotopy groups". In: *Journal of Combinatorial Theory, Series A* 102.1 (2003), pp. 88–93. ISSN: 0097-3165. DOI: https://doi.org/10.1016/S0097-3165(03)00015-3. URL: https://www.sciencedirect.com/science/article/pii/S0097316503000153.

[BL23] Magnus Bakke Botnan and Michael Lesnick. *An Introduction to Multiparameter Persistence*. 2023. arXiv: 2203.14289 [math.AT].

[BM06] Jonathan Ariel Barmak and Elias Gabriel Minian. *Simple Homotopy Types and Finite Spaces*. 2006. DOI: 10.48550/ARXIV.MATH/0611158. URL: https://arxiv.org/abs/math/0611158.

[BM19] Jean-Daniel Boissonnat and Clément Maria. "Computing persistent homology with various coefficient fields in a single pass". In: *Journal of Applied and Computational Topology* 3.1-2 (Apr. 2019), pp. 59–84. DOI: 10.1007/s41468-019-00025-y. URL: https://doi.org/10.1007%2Fs41468-019-00025-y.

[Bor48] Karol Borsuk. "On the imbedding of systems of compacta in simplicial complexes". In: *Fundamenta Mathematicae* 35 (1948), pp. 217–234.

[Bou16] N. Bourbaki. *Éléments de Mathématique. Topologie algébrique. Chapitres 1 à 4*. Springer, 2016.

[BP20] Jean-Daniel Boissonnat and Siddharth Pritam. "Edge Collapse and Persistence of Flag Complexes". In: *SoCG 2020 - 36th International Symposium on Computational Geometry*. Zurich, Switzerland, June 2020. DOI: 10.4230/LIPIcs.SoCG.2020.19. URL: https://hal.inria.fr/hal-02873740.

[Bru23] Brunnengräber, Moritz. *Delaunay Triangulation and Voronoi Diagram*. https://cartography-playground.gitlab.io/playgrounds/triangulation-delaunay-voronoi-diagram/, Last accessed on 2023-09-01. 2023.

[BS14] Peter Bubenik and Jonathan A. Scott. "Categorification of Persistent Homology". In: *Discrete and Computational Geometry* 51.3 (Jan. 2014), pp. 600–627. DOI: 10.1007/s00454-014-9573-x. URL: https://doi.org/10.1007%2Fs00454-014-9573-x.

[BSS14] Peter Bubenik, Vin de Silva, and Jonathan Scott. "Metrics for Generalized Persistence Modules". In: *Foundations of Computational Mathematics* 15.6 (Oct. 2014), pp. 1501–1531. DOI: 10.1007/s10208-014-9229-5. URL: https://doi.org/10.1007%2Fs10208-014-9229-5.

[Bub20] Peter Bubenik. "The Persistence Landscape and Some of Its Properties". In: *Topological Data Analysis*. Springer International Publishing, 2020, pp. 97–117. DOI: 10.1007/978-3-030-43408-3_4. URL: https://doi.org/10.10072F978-3-030-43408-3_4.

[Bub23] Bubenik, Peter. *Software*. https://people.clas.ufl.edu/peterbubenik/software/, Last accessed on 2023-03-10. 2023.

[Car+19] Mathieu Carrière et al. *PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures*. 2019. DOI: 10.48550/ARXIV.1904.09378. URL: https://arxiv.org/abs/1904.09378.

[Car09] Gunnar Carlsson. "Topology and Data". In: *Bulletin of The American Mathematical Society - BULL AMER MATH SOC* 46 (Apr. 2009), pp. 255–308. DOI: 10.1090/S0273-0979-09-01249-X.

[Car14] Gunnar Carlsson. "Topological pattern recognition for point cloud data". In: *Acta Numerica* 23 (May 2014), pp. 289–368. DOI: 10.1017/S0962492914000051.

[Cas20] Álvaro Torras Casas. *Distributing Persistent Homology via Spectral Sequences*. 2020. arXiv: 1907.05228 [math.AT].

[CCO17] Mathieu Carrière, Marco Cuturi, and Steve Oudot. *Sliced Wasserstein Kernel for Persistence Diagrams*. 2017. arXiv: 1706.03358 [cs.CG].

[CCS14] Frederic Chazal, William Crawley-Boevey, and Vin de Silva. *The observable structure of persistence modules*. 2014. arXiv: 1405.5644 [math.RT].

[CD19] Frédéric Chazal and Vincent Divol. *The density of expected persistence diagrams and its kernel based estimation*. 2019. arXiv: 1802.10457 [cs.CG].

[CEH05] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. "Stability of Persistence Diagrams". In: vol. 37. June 2005, pp. 263–271. DOI: 10.1007/s00454-006-1276-5.

[Cha+09] Frédéric Chazal et al. "Proximity of persistence modules and their diagrams". In: *SoCG 2009 - 25th Annual Symposium on Computational Geometry*. Aarhus, Denmark: ACM Press, June 2009, p. 10. DOI: 10.1145/1542362.1542407. URL: https://inria.hal.science/hal-02292996.

[Cha+10] Erin Chambers et al. "Vietoris–Rips Complexes of Planar Point Sets". In: *Discrete and Computational Geometry* 44 (July 2010), pp. 75–90. DOI: 10.1007/s00454-009-9209-8.

[Cha+13] Frédéric Chazal et al. *Stochastic Convergence of Persistence Landscapes and Silhouettes*. 2013. DOI: `10.48550/ARXIV.1312.0308`. URL: `https://arxiv.org/abs/1312.0308`.

[Cha+16] Frédéric Chazal et al. *The Structure and Stability of Persistence Modules*. SppringerBriefs in Mathematics. Springer, 2016.

[CK11] Chao Chen and Michael Kerber. "Persistent Homology Computation with a Twist". In: *27th European Workshop on Computational Geometry (EuroCG 2011) (pp. 1–4)* (2011). URL: `http://hdl.handle.net/20.500.12708/53902`.

[CM17] Frédéric Chazal and Bertrand Michel. "An Introduction to Topological Data Analysis: Fundamental and Practical Aspects for Data Scientists". In: *Frontiers in Artificial Intelligence* 4 (Oct. 2017). DOI: `10.3389/frai.2021.667963`.

[COO15] Mathieu Carrière, Steve Y. Oudot, and Maks Ovsjanikov. "Stable Topological Signatures for Points on 3D Shapes". In: *Computer Graphics Forum* (2015). DOI: `10.1111/cgf.12692`.

[Cor+09] T.H. Cormen et al. *Introduction to Algorithms, third edition*. Computer science. MIT Press, 2009. ISBN: 9780262033848. URL: `https://books.google.de/books?id=i-bUBQAAQBAJ`.

[Cra14] William Crawley-Boevey. *Decomposition of pointwise finite-dimensional persistence modules*. 2014. arXiv: `1210.0819 [math.RT]`.

[CS08] Gunnar Carlsson and Vin de Silva. *Zigzag Persistence*. 2008. arXiv: `0812.0197 [cs.CG]`.

[CSO13] Frederic Chazal, Vin de Silva, and Steve Oudot. *Persistence stability for geometric complexes*. 2013. arXiv: `1207.3885 [math.AT]`.

[Čuf20] Matija Čufar. "Ripserer.jl: flexible and efficient persistent homology computation in Julia". In: *Journal of Open Source Software* 5 (Oct. 2020), p. 2614. DOI: `10.21105/joss.02614`.

[Cur14] Justin Curry. *Sheaves, Cosheaves and Applications*. 2014. arXiv: `1303.3255 [math.AT]`.

[CZ07] Gunnar Carlsson and Afra Zomorodian. "The Theory of Multidimensional Persistence". In: *Discrete and Computational Geometry* 42 (June 2007), pp. 71–93. DOI: `10.1007/s00454-009-9176-0`.

[DE95] Cecil Jose A. Delfinado and Herbert Edelsbrunner. "An incremental algorithm for Betti numbers of simplicial complexes on the 3-sphere". In: *Comput. Aided Geom. Des.* 12 (1995), pp. 771–784.

[DF03] D.S. Dummit and R.M. Foote. *Abstract Algebra*. Wiley, 2003. ISBN: 9780471433347. URL: `https://books.google.de/books?id=KJDBQgAACAAJ`.

[DF15] Barbara Di Fabio and Massimo Ferri. *Comparing persistence diagrams through complex vectors*. 2015. DOI: `10.48550/ARXIV.1505.01335`. URL: `https://arxiv.org/abs/1505.01335`.

[DFW12] Tamal K. Dey, Fengtao Fan, and Yusu Wang. *Computing Topological Persistence for Simplicial Maps*. 2012. DOI: `10.48550/ARXIV.1208.5018`. URL: `https://arxiv.org/abs/1208.5018`.

[DFW13] Tamal K. Dey, Fengtao Fan, and Yusu Wang. *Graph Induced Complex on Point Data*. 2013. arXiv: `1304.0662 [cs.CG]`.

[Dro12] Jean-Marie Droz. *A subset of Euclidean space with large Vietoris-Rips homology*. 2012. arXiv: `1210.4097 [math.GT]`.

[EH08] Herbert Edelsbrunner and John Harer. "Persistent homology—a survey". In: *Discrete and Computational Geometry - DCG* 453 (Jan. 2008). DOI: `10.1090/conm/453/08802`.

[EH10] Herbert Edelsbrunner and John L. Harer. *Computational Topology. An Introduction*. AMS, 2010.

[ELZ00] H. Edelsbrunner, D. Letscher, and A. Zomorodian. "Topological persistence and simplification". In: *Proceedings 41st Annual Symposium on Foundations of Computer Science*. 2000, pp. 454–463. DOI: `10.1109/SFCS.2000.892133`.

[ES45] Samuel Eilenberg and Norman E. Steenrod. "Axiomatic Approach to Homology Theory". In: *Proceedings of the National Academy of Sciences of the United States of America* 31.4 (1945), pp. 117–120. ISSN: 00278424. URL: `http://www.jstor.org/stable/87896` (visited on 03/06/2023).

[Fac98] Alberto Facchini. *Module Theory. Endomorphism rings and direct sum decompositions in some classes of modules*. 2nd ed. Springer Basel AG, 1998.

[FMN13] Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. *Testing the Manifold Hypothesis*. 2013. arXiv: `1310.0425 [math.ST]`.

[Fro92] Patrizio Frosini. "Measuring shapes by size functions". In: *Other Conferences*. 1992.

[Ghr08] Robert Ghrist. "Barcodes: The persistent topology of data". In: *BULLETIN (New Series) OF THE AMERICAN MATHEMATICAL SOCIETY* 45 (Feb. 2008). DOI: `10.1090/S0273-0979-07-01191-3`.

[Ghr14] Robert Ghrist. *Elementary Applied Topology*. 2014. URL: `https://www2.math.upenn.edu/~ghrist/notes.html` (visited on 03/01/2023).

[Gio] Giotto-tda. *Weak Alpha Persistence*. `https://giotto-ai.github.io/gtda-docs/0.3.0/modules/generated/homology/gtda.homology.WeakAlphaPersistence.html`, Last accessed on 2023-04-03.

[GM05] Robert Ghrist and Abubakr Muhammad. "Coverage and hole-detection in sensor networks via homology". In: May 2005, pp. 254–260. ISBN: 0-7803-9201-9. DOI: `10.1109/IPSN.2005.1440933`.

[GO08] Leonidas Guibas and Steve Oudot. "Reconstruction Using Witness Complexes". In: *Discrete and computational geometry* 40 (Oct. 2008), pp. 325–356. DOI: 10.1007/s00454-008-9094-6.

[Hat02] Allen Hatcher. *Algebraic Topology.* Cambridge University Press, 2002.

[Ito87] Kiyosi Ito. *Encyclopedic Dictionary of Mathematics.* 2nd ed. MIT Press, 1987.

[Jam20] Clément Jamin. *Tangential Complex.* 2020. URL: https://gudhi.inria.fr/doc/3.1.1/group__tangential__complex.html (visited on 09/25/2023).

[KFH16] Genki Kusano, Kenji Fukumizu, and Yasuaki Hiraoka. *Persistence weighted Gaussian kernel for topological data analysis.* 2016. arXiv: 1601.01741 [math.AT].

[LCB] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. *THE MNIST DATABASE of handwritten digits.* http://yann.lecun.com/exdb/mnist/, Last accessed on 2023-03-16.

[Ler45] J. Leray. *Topologie algebrique.sur la forme des espaces topologiques et sur le points fixes des representations etc ...* Journal de mathematiques pures et appliquees. Gauthıéer-Villars, 1945. URL: https://books.google.de/books?id=E3uMGwAACAAJ.

[Les15] Michael Lesnick. "The Theory of the Interleaving Distance on Multidimensional Persistence Modules". In: *Foundations of Computational Mathematics* 15.3 (Mar. 2015), pp. 613–650. DOI: 10.1007/s10208-015-9255-y. URL: https://doi.org/10.1007%2Fs10208-015-9255-y.

[Lue+19] Daniel Luetgehetmann et al. *Computing persistent homology of directed flag complexes.* 2019. DOI: 10.48550/ARXIV.1906.10458. URL: https://arxiv.org/abs/1906.10458.

[LY18] Tam Le and Makoto Yamada. "Persistence Fisher Kernel: A Riemannian Manifold Kernel for Persistence Diagrams". In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems.* NIPS'18. Montréal, Canada: Curran Associates Inc., 2018, pp. 10028–10039.

[LZ14] Ryan H. Lewis and Afra Zomorodian. *Multicore Homology via Mayer Vietoris.* 2014. arXiv: 1407.2275 [cs.CG].

[Mat22] Encyclopedia of Mathematics. *Homology functor.* http://encyclopediaofmath.org/index.php?title=Homology_functor&oldid=52353 [Accessed: 19.07.2023]. 2022.

[McC67] Michael C. McCord. "Homotopy Type Comparison of a Space with Complexes Associated with its Open Covers". In: *Proceedings of the American Mathematical Society* 18.4 (1967), pp. 705–708. ISSN: 00029939, 10886826. URL: http://www.jstor.org/stable/2035443 (visited on 03/03/2023).

[Men+19] Zhenyu Meng et al. *Weighted persistent homology for biomolecular data analysis*. 2019. arXiv: 1903.02890 [q-bio.BM].

[Moo+21] Michael Moor et al. *Topological Autoencoders*. 2021. arXiv: 1906.00722 [cs.LG].

[Mun84] James R. Munkres. "Elements of algebraic topology". In: 1984.

[Nan20] Vidit Nanda. *Computational Algebraic Topology*. 2020. URL: https://people.maths.ox.ac.uk/nanda/cat/.

[NSW08] Partha Niyogi, Stephen Smale, and Shmuel Weinberger. "Finding the Homology of Submanifolds with High Confidence from Random Samples". In: *Discrete and Computational Geometry* 39 (Mar. 2008), pp. 419–441. DOI: 10.1007/s00454-008-9053-2.

[Ott+17] Nina Otter et al. "A roadmap for the computation of persistent homology". In: *EPJ Data Science* 6.1 (Aug. 2017). DOI: 10.1140/epjds/s13688-017-0109-5. URL: https://doi.org/10.1140%2Fepjds%2Fs13688-017-0109-5.

[Oud15] Steve Oudot. *Persistence Theory: From Quiver Representations to Data Analysis*. AMS, 2015.

[Pas+19] Adam Paszke et al. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. 2019. arXiv: 1912.01703 [cs.LG].

[Per18] Jose A. Perea. *A Brief History of Persistence*. 2018. arXiv: 1809.03624 [math.AT].

[Poi21] Henri Poincaré. "Analyse des travaux scientifiques de Henri Poincaré faite par lui-même". In: *Acta mathematica* 38 (1921), pp. 1–135.

[Pri] Siddharth Pritam. *Edge collapse*. https://gudhi.inria.fr/doc/3.4.1/group__edge__collapse.html, Last accessed on 2023-04-03.

[Rei+14] Jan Reininghaus et al. *A Stable Multi-Scale Kernel for Topological Machine Learning*. 2014. DOI: 10.48550/ARXIV.1412.6821. URL: https://arxiv.org/abs/1412.6821.

[Rob99] Vanessa Robins. "Towards computing homology from approximations". In: *Topology Proceedings* 24 (Jan. 1999).

[Roy+21] Martin Royer et al. "ATOL: Measure Vectorization for Automatic Topologically-Oriented Learning". In: *AISTATS 2021 - 24th International Conference on Artificial Intelligence and Statistics*. The 24th International Conference on Artificial Intelligence and Statistics. Virtual conference, France, Apr. 2021. URL: https://hal.science/hal-02296513.

[Ser55] Jean-Pierre Serre. "Faisceaux Algebriques Coherents". In: *Annals of Mathematics* 61.2 (1955), pp. 197–278. ISSN: 0003486X. URL: http://www.jstor.org/stable/1969915 (visited on 05/05/2023).

[SG07] Vin Silva and Robert Ghrist. "Coverage in sensor networks via persistent homology". In: *Algebraic and Geometric Topology* 7 (Apr. 2007). DOI: 10.2140/agt.2007.7.339.

[Sil03] Vin de Silva. *A weak definition of Delaunay triangulation*. 2003. arXiv: cs/0310031 [cs.CG].

[SMC07] Gurjeet Singh, Facundo Memoli, and Gunnar Carlsson. "Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition". In: *Eurographics Symposium on Point-Based Graphics*. Ed. by M. Botsch et al. The Eurographics Association, 2007. ISBN: 978-3-905673-51-7. DOI: 10.2312/SPBG/SPBG07/091-100.

[SMV11] Vin de Silva, Dmitriy Morozov, and Mikael Vejdemo-Johansson. "Dualities in persistent (co)homology". In: *Inverse Problems* 27.12 (Nov. 2011), p. 124003. DOI: 10.1088/0266-5611/27/12/124003. URL: https://doi.org/10.1088%2F0266-5611%2F27%2F12%2F124003.

[SV09] Vin de Silva and Mikael Vejdemo-Johansson. "Persistent cohomology and circular coordinates". In: *Proceedings of the twenty-fifth annual symposium on Computational geometry*. ACM, June 2009. DOI: 10.1145/1542362.1542406. URL: https://doi.org/10.1145%2F1542362.1542406.

[Tau+20] Guillaume Tauzin et al. "giotto-tda: A Topological Data Analysis Toolkit for Machine Learning and Data Exploration". In: (2020). DOI: 10.48550/ARXIV.2004.02551. URL: https://arxiv.org/abs/2004.02551.

[TP21] Álvaro Torras and Ulrich Pennig. *Interleaving Mayer-Vietoris spectral sequences*. 2021. arXiv: 2105.03307 [math.AT].

[TSB18] Christopher Tralie, Nathaniel Saul, and Rann Bar-On. "Ripser.py: A Lean Persistent Homology Library for Python". In: *Journal of Open Source Software* 3.29 (2018), p. 925. DOI: 10.21105/joss.00925. URL: https://doi.org/10.21105/joss.00925.

[Web85] Cary Webb. "Decomposition of graded modules". In: 1985. URL: https://api.semanticscholar.org/CorpusID:115146035.

[Wei52] André Weil. "Sur les théorèmes de de Rham." In: *Commentarii mathematici Helvetici* 26 (1952), pp. 119–145. URL: http://eudml.org/doc/139040.

[XRV17] Han Xiao, Kashif Rasul, and Roland Vollgraf. *Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms*. Aug. 28, 2017. arXiv: cs.LG/1708.07747 [cs.LG].

[YG20] Hee Rhang Yoon and Robert Ghrist. *Persistence by Parts: Multiscale Feature Detection via Distributed Persistent Homology*. 2020. arXiv: 2001.01623 [math.AT].

[ZC05] Afra Zomorodian and Gunnar Carlsson. "Computing Persistent Homology". In: *Discrete and Computational Geometry* 33 (Feb. 2005), pp. 249–274. DOI: 10.1007/s00454-004-1146-y.

[ZC08] Afra Zomorodian and Gunnar Carlsson. "Localized Homology". In: *Comput. Geom.* 41 (Jan. 2008), pp. 126–148. DOI: `10.1109/SMI.2007.25`.

[ZXW20] Simon Zhang, Mengbai Xiao, and Hao Wang. *GPU-Accelerated Computation of Vietoris-Rips Persistence Barcodes*. 2020. DOI: `10.48550/ARXIV.2003.07989`. URL: `https://arxiv.org/abs/2003.07989`.