ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Persistent Homology and the Classification of Liver Lesions

Bachelor's Thesis

Thomas Heinz

Thursday 18th April, 2024

Advisors: Dr. S. Kališnik Hintz
Department of Mathematics, ETH Zürich

Abstract

The goal of this thesis is to define persistent homology and to show how it can be used to classify (hepatic) liver lesions. We introduce the notions of simplicial complexes and homology. After that, we explain what persistent vector spaces are and present the structure theorem for persistent homology. Using this theorem, we are able to represent the persistent vector space we are interested in by a barcode (or a persistent diagram). On the set of barcodes we can even define metrics. All these notions can be used to analyze datasets consisting of point clouds or black-and-white images. Last, but not least, we present a hands-on application where we use persistent homology to classify a dataset of images of liver lesions.

Acknowledgements

I would like to express my deepest appreciation to Dr. Sara Kališnik Hintz for her supervision. She was very committed and patiently helped me whenever I struggled. Her guidance as well as her wish that I benefit from writing this thesis made the whole process a enjoyable experience for me. It helped me grow as a mathematician and I want her to know that I sincerely appreciate that. Moreover, I want to thank my friend Samuel Huber for helping me coming up with a proof idea once when a I was completely stuck, as well as for spell-checking the thesis.

Contents

Chapter 1

Introduction

In the field of *topological data analysis* one analyzes data sets using techniques from (algebraic) topology. Topology is a branch of mathematics that studies the properties of geometric objects that are preserved under continuous deformations. If we think of a geometric object as being made of dough, the object is identified with any object it can be continuously deformed into. Continuous deformations allow for stretching, contracting and twisting. We are not allowed to tear the dough apart or glue one side to another. For example, consider Figure 1.1.

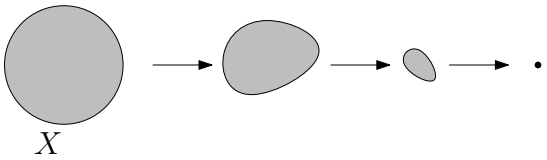

Figure 1.1: Example of a continuous deformation of a space.

The space X consists of a closed disk. As the figure illustrates, we are able to deform it into a small point. Thus, for a topologist a closed disk is considered the same as a point.

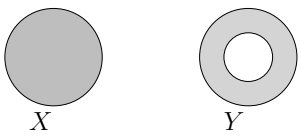

Figure 1.2: Example of spaces with di erent Betti numbers.

Using algebraic tools, one might also find invariants of topological spaces, such as the so-called *Betti numbers* b_i, which count the number of i-dimensional holes a space has. Consider Figure 1.2. The space Y contains one hole, in con-

trast to the space X, which has none. From this we can tell that these spaces are not the same, since their invariants (the Betti numbers) are different.

We wish to use the invariants of a topological space, like the Betti numbers, to capture the shape of a given dataset. Consider the datasets as depicted in Figure 1.3.

(a) A nite subset of a metric space (b) 3 3 black-and-white image

Figure 1.3: Examples of datasets.

The rst dataset, Figure 1.3a, consist of a nite set of points in a metric space. The other one, 1.3b, is a black-and-white image, consisting of 3 3 pixels. In topological data analysis one analyzes these kinds of sets by de ning geometrical objects on them. One way to do this is by using a collection of so-called simplices, where an i-simplexis an object de ned by $(i + 1)$ points that are not contained in an $(i \quad 1)$-dimensional subspace of the metric space. For instance, a 0-simplex is a point, a 1-simplex is line segment, and a 2-simplex is a triangle, and so on. If we glue these simplices along common boundaries, we get a simplicial complex. To give a concrete example, consider Figure 1.4.

Figure 1.4: Construction of geometrical objects on data points.

In the gure we assume a dataset is given as a nite set of points in the plane

$\mathbb{R}^2$. We draw around each data point a closed ball, each with the same radius. We add an i-simplex to the complex if $i+1$ balls intersect. The larger the radius, the more simplices our complex contains. In Figure 1.4 we see how different radii give rise to different complexes. The first complex is just a set of discrete points, the second one forms a square, and the last one forms a tetrahedron.

Using these constructions, we want to find out what the topological features of the dataset are. But as seen in the previous example, on the same dataset we can construct distinct complexes with different topological features. So how can we make any use of them in order to draw conclusion on the "true shape" of the dataset?

Persistent homology tackles that issue by constructing a 1-parameter family of simplicial complexes that depends on a parameter r. In Figure 1.4 the parameter is the radius of the closed balls. For each value of r, we look at what topological features the corresponding complex has, while keeping track of when certain features appear (for example when a hole is formed) and when they disappear (i.e. the hole gets filled in). Features existing for a long time are considered more likely to reflect the properties of the dataset. Those that "die" after a short time are more likely due to errors occurring while sampling the data. One way to represent this is by using barcodes. One example for such a barcode can be seen in Figure 1.5.

Figure 1.5: Barcode of a space for the Betti number b_1.

This barcode tells us that the we can detect three 1-dimensional holes in the space. One appears at T_1 and disappears at T_3, another one "lives" from T_4 until T_5 and a last one appears at T_4 and gets filled in at T_7.

Finally, we apply persistent homology to classify liver lesions. Lesions are abnormal growths in the liver. Most of them are harmless, but some of them might be related to cancer. The method of image analysis, which relies on persistent homology, can help to detect such cancerous lesions. For this, we

construct barcodes from images of different lesion types which then. With the help of machine learning algorithms, we investigate these barcodes and classify the lesions.

Chapter 2

Basic Concepts

The goal of this chapter is to introduce the mathematical background and the tools that we are going to use in this thesis. The chapter is based on Carlsson's Topological pattern recognition for point cloud data ([5]), on Hatcher's Algebraic Topology ([10]), on Computational Topology for Data Analysis by Dey and Wang ([9]), as well as on Topological Data Analysis with Applications by Carlsson and Vejdemo-Johansson ([6]).

2.1 Simplicial Complexes

In this section we will look at topological spaces which can be described in a very combinatorial way. The building blocks consist of points, line segments between points, and more generally, of convex hulls of points. To make the theory work, we must restrict the relative position between the points.

De nition 2.1 Let $S = \{x_0, x_1, \ldots, x_n\} \subset \mathbb{R}^k$ denote a nite subset. We say that S is in general position if it is not contained in any af ne subspace of dimension $n - 1$ in $\mathbb{R}^k$.

Note that in the de nition the enumeration of the points starts at 0. To illustrate the de nition, let us look at an example.

Example 2.2 Consider $S = \{x_0, x_1, x_2\}, P = \{x_3, x_4, x_5\} \subset \mathbb{R}^2$ as in Figure 2.1.

Figure 2.1: Example and counterexample for points in general position.

We note that the points of S are in general position, since no three points of S lie on a line, i.e. in a 1-dimensional subspace of $\mathbb{R}^2$, whereas the points of T are not in general position, since the three points x_3, x_4, x_5 lie on the same line.

Example 2.3 Consider any $S = \{x_0, x_1, \ldots, x_n\} \subset \mathbb{R}^{n-1}$. Then S is not in general position, since its points are all contained in the $(n-1)$-dimensional space $\mathbb{R}^{n-1}$.

The basic building blocks of the objects we will construct are called simplices.

Definition 2.4 Let $S = \{x_0, x_1, \ldots, x_n\} \subset \mathbb{R}^k$ be in general position. We define the simplex spanned by S to be the convex hull $s = s(S)$ of S in $\mathbb{R}^k$, i.e. the set of points which can be expressed as a linear combination $\sum_{i=0}^{n} \lambda_i x_i$ such that

$$\forall i = 0, 1, \ldots, n : \lambda_i \geq 0 \text{ and } \sum_{i=0}^{n} \lambda_i = 1.$$

The points x_i of S are called vertices, and the simplices $s(T)$ spanned by a non-empty subset $T \subseteq S$ are called faces of s.

Remark 2.5 We may write $s = [x_0, x_1, \ldots, x_n]$ or just $x_0 x_1 \ldots x_n$ for the simplex spanned by the vertices $\{x_0, x_1, \ldots, x_n\}$.

Example 2.6 Take $S = \{x_0, x_1, x_2\}$ as seen before on Figure 2.1. The simplex spanned by S, denoted by s, is just the triangle spanned by the three points. Take $T = \{x_1, x_2\} \subset S$. Then $s(T)$, i.e. the edge connecting the vertices x_1 and x_2, is a face of s.

Definition 2.7 If a simplex s is spanned by $k+1$ vertices, we call it a k-simplex.

Example 2.8 Points are 0-simplices, edges are 1-simplices, triangles are 2-simplices, tetrahedrons form 3-simplices. The objects can be seen in Figure 2.2.

(a) 0-simplex (b) 1-simplex

(c) 2-simplex (d) 3-simplex

Figure 2.2: Examples of simplices.

Definition 2.9 By a (finite) simplicial complex, we mean a finite collection of simplices in a Euclidean space $\mathbb{R}^k$, denoted by X, such that the following conditions hold:

(1) For any simplex $s \in X$, all faces of s are also contained in X.

(2) For any two simplices s and t of X, the intersection $s \cap t$ is a simplex, which is a face of both s and t.

Example 2.10 Consider the two lists of simplicies

$$X_1 = \{A, B, C, AB, AC, ABC\},$$
$$X_2 = \{D, E, F, G, DE, FG\}.$$

The objects lie in $\mathbb{R}^2$ as depicted in Figure 2.3. X_1 is not a simplicial complex since the edge connecting B and C is missing, i.e. condition (1) is not satisfied. On the other hand, X_2 is not a simplicial complex since the intersection point of the edges DE and FG is not a simplex in the list, so condition (2) is not satisfied.

(a) X_1 from Example 2.10. (b) X_2 from Example 2.10.

Figure 2.3: Counterexamples for simplicial complexes.

We note that a simplicial complex is a family of subsets of the whole vertex set which determines the relations between the vertices. So, instead of the geometric object described by the simplicial complex, we might only be interested in the combinatorial object it defines. I.e. we only consider the set of vertices and the relations between them. This motivates the next definition.

Definition 2.11 An abstract simplicial complex X is a pair $X = (V(X), S(X))$, where $V(X)$ is a finite set called the vertices of X, and $S(X)$ is a subset of the collection of all non-empty subsets of $V(X)$ called the simplices, satisfying the following condition:

$$\forall\, t \subseteq s \in S(X) \Rightarrow t \in S(X).$$

A simplicial complex X therefore determines an abstract simplicial complex whose vertex set $V(X)$ is given by the set of all vertices of X and where a subset of $V(X)$ is in the collection of simplices $S(X)$ if and only if the elements of that subset form a simplex of X. In other words, considering the abstract simplicial complex X of a simplicial complex X, we do not care about the distances between the vertices of X, but focus on the relation between them, i.e. if they are connected by an edge, for instance.

Example 2.12 Consider the tetrahedron in $\mathbb{R}^3$ given in Figure 2.2d.

This geometric object is described by the vertex set $V(X) = \{A, B, C, D\}$ and the simplices

$$S(X) = \{A, B, C, D, AB, AC, AD, BC, BD, CD, ABC, ABD, ACD, BCD\}.$$

We are also able to construct a simplicial complex from an abstract simplicial complex.

Definition 2.13 Let $X^0 = (V(X^0), S(X^0))$ be an abstract simplicial complex such that $V(X^0) = \{x_0, x_1, \ldots, x_n\}$, and let $S := \{e_0, e_1, \ldots, e_n\}$ be a basis of $\mathbb{R}^{n+1}$. Then a simplicial complex $X = (S, S(X))$ is called a *geometric realisation* of X^0 if the following condition holds:

$$[x_{i_0}, x_{i_1}, \ldots, x_{i_k}] \in S(X^0), \quad [e_{i_0}, e_{i_1}, \ldots, e_{i_k}] \in S(X),$$

where $\{i_0, i_1, \ldots, i_k\} \subseteq \{0, 1, \ldots, n\}$.

Remark 2.14 Note that a geometric realisation of an abstract simplicial complex with vertex set $\{x_0, x_1, \ldots, x_n\}$ does not have to be an $(n+1)$-dimensional object. It can often be embedded in a lower-dimensional subspace of $\mathbb{R}^{n+1}$.

Furthermore, it is possible to define maps between (abstract) simplicial complexes:

Definition 2.15 Let $X = (V(X), S(X))$ and $Y = (V(Y), S(Y))$ be two (abstract) simplicial complexes. We say that f is a *map of (abstract) simplicial complexes* if it is a map in the sets of vertices $f: V(X) \to V(Y)$ such that

$$\forall s \in S(X): f(s) \in S(Y),$$

where $f(s) = f([x_0, x_1, \ldots, x_k]) = [f(x_0), f(x_1), \ldots, f(x_k)]$.

Example 2.16 Consider the two abstract simplicial complexes $X = (V(X), S(X))$ and $Y = (V(Y), S(Y))$, such that

$$V(X) = V(Y) = \{A, B, C, D\},$$
$$S(X) = \{A, B, C, D, AB, AC, AD, BC, ABC\},$$
$$S(Y) = \{A, B, C, D, AB, AD, BC, BD, CD, ABD, BCD\}.$$

The complexes can be seen in Figure 2.4.

Then a map of simplicial complexes $f: V(X) \to V(Y)$ is given via

$$f(A) = B,$$
$$f(B) = C,$$
$$f(C) = D,$$
$$f(D) = A.$$

Figure 2.4: The complexes $X = (V(X), S(X))$ and $Y = (V(Y), S(Y))$ from Example 2.16.

Indeed, one easily verifies that for every simplex s of X we have that $f(s)$ is a simplex of Y.

Moreover, if $f : V(X) \to V(Y)$ is a function on a vertex set, it induces a simplicial complex.

Example 2.17 Let $X = (V(X), S(X))$ be a simplicial complex given by the vertex set $V(X) = \{A, B, C, D\}$ and set of simplices

$$S(X) = \{A, B, C, D, AB, AC, AD, BC, CD, ACD\}.$$

Let $f : V(X) \to V(X)$ be a function on $V(X)$ such that

$$f(A) = C,$$
$$f(B) = D,$$
$$f(D) = A,$$
$$f(C) = B.$$

Then f induces the complex $Y = (V(Y), S(Y))$, where $V(Y) = V(X)$ and

$$S(Y) = \{A, B, C, D, CD, CB, CA, DB, BA, CBA\}.$$

The complexes can be seen in Figure 2.5.

Figure 2.5: A map f on the set of vertices induces a simplical complex.

2.2 Homology

In this section, we define the notion of *homology*. As a motivation, consider the abstract simplicial complex X with the list of simplicies

$$S(X) = \{A, B, C, D, E, AB, BC, BE, CD, CE, AD, BCE\}.$$

Figure 2.6: Simplicial complex with two loops, one of which is filled-in.

The geometric realization of this complex, as seen in Figure 2.6, has one loop, the square ABCD, and one loop that is filled in by a simplex, the triangle BCE. The main goal of homology is to investigate how many loops in our abstract simplicial complex are not filled in. Such loops are often called *holes*. To do this, we first introduce the notion of a *chain complex* that enables us to construct vector spaces on simplicial complexes. By using some methods from linear algebra, these vector spaces will allow us to compute the so called Betti numbers, which count the number of i-dimensional holes of the underlying simplicial complex.

2.2.1 Chain Complexes

Definition 2.18 Let k be a field and S a finite set. Then the *free k-vector space* on S, denoted by $V_k(S)$, is the k-span of S.

Definition 2.19 Let k be a field and X be a given (abstract) simplicial complex. Denote by $C_i(X)$ the free k-vector space on the set of i-simplices. We call an element of $C_i(X)$ an *i-chain*.

One important property of $C_i(X)$ is that it forms a vector space over k: If $c = \sum c_j s_j$ and $d = \sum d_j s_j$, we define the sum $c + d$ to be $c + d := \sum (c_j + d_j) s_j$. Also, for $a \in k$ and c as before, we define the scalar multiplication as follows: $ac := \sum a c_j s_j$. Thus, $C_i(X)$ forms a vector space over k, the so called vector space of i chains in X. The neutral element is given by $0 := \sum 0 s_i$.

Remark 2.20 The set of i-simplices forms a basis of $C_i(X)$. Hence, the rank of $C_i(X)$ is given by the number of i-simplices in X. If $i < 0$ or $i > \dim(X)$, we have $C_i(X) = 0$ since there are no simplices of these dimensions.

From now on we will only consider simplicial complexes over the binary field, i.e. $k = Z_2$. That way, the addition of two i-simplices can considered as taking the symmetric difference between them.

Definition 2.21 For an i-simplex s we define its **boundary** $\partial_i s$ to be the sum of its $(i-1)$-dimensional faces, i.e. if $s = [x_0, x_1, \ldots, x_i]$ then

$$\partial_i s = \sum_{j=0}^{i} [x_0, x_1, \ldots, \hat{x}_j, \ldots, x_i]$$

where the notation $\hat{x}_j$ means that x_j is omitted. For an i-chain $c = \sum c_j s_j$, the boundary is the sum of the boundaries of its simplices, $\partial_i c = \sum c_j (\partial_i s_j)$.

Example 2.22 Let us look again at the complex from Figure 2.6. The boundary of the triangle BCE, a 2-simplex, is given by $BC + BE + CE$. The boundary of A, a 0-simplex, is 0.

We observe that "taking boundaries" is a linear map from the set of i-chains into the set of $(i-1)$-chains. This gives rise to the next definition.

Definition 2.23 For a fixed basis of $C_i(X)$ and $C_{i-1}(X)$, we can represent the boundary map $\partial_i : C_i(X) \to C_{i-1}(X)$ by a matrix, called the **boundary matrix**.

We choose the set of i-simplices and as the basis of $C_i(X)$. Similarly, the set of $(i-1)$-simplices forms the basis of $C_{i-1}(X)$. This gives us an $n_{i-1} \times n_i$ matrix, where n_{i-1} and n_i denote the number of $(i-1)$-simplices and i-simplices of X respectively. Each row is indexed by a $(i-1)$-simplex and every column is indexed by an i-simplex. The entries a_{kl} such that $\partial_i = \{a_{kl}\}$ are defined by

$$a_{kl} = \begin{cases} 1, & \text{if the simplex of row } k \text{ is a face of the simplex of column } l \\ 0, & \text{otherwise.} \end{cases}$$

Remark 2.24 We denote both the transformation and the corresponding matrices by ∂_i.

Example 2.25 Let X be the simplicial complex given in Figure 2.6. We want to determine the matrix of ∂_1: The row entries correspond to the 0-simplices of X, i.e. the vertices, whereas the columns correspond to the 1-simplices of X, i.e. the edges. The entry a_{kl} of the matrix is equal to 1 if and only if the vertex of row k is in the boundary of the edge of column l. Thus the boundary matrix is given by

$$\partial_1 = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{cccccc} AB & AD & BC & BE & CD & CE \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}.$$

The next result tells us that that boundary of a boundary needs to be zero.

Lemma 2.26 (Fundamental Lemma of Homology) For a given (abstract) simplicial complex X, it holds that

$$\forall i \geq 1 : \partial_i \circ \partial_{i+1} = 0$$

where $\partial_i : C_i(X) \to C_{i-1}(X)$ and $\partial_{i+1} : C_{i+1}(X) \to C_i(X)$ are the respective boundary maps.

Proof Let $s = [x_0, x_1, \ldots, x_{i+1}]$ be an $(i+1)$-simplex. Then we have:

$$\begin{aligned}
\partial_i \circ \partial_{i+1} s &= \partial_i \sum_{j=0}^{i+1} [x_0, x_1, \ldots, \hat{x_j}, \ldots, x_{i+1}] \\
&= \sum_{k=0}^{i+1} \sum_{\substack{j=0 \\ k \neq j}}^{i+1} [x_0, x_1, \ldots, \hat{x_j}, \ldots, \hat{x_k}, \ldots, x_{i+1}] \\
&= 0
\end{aligned}$$

where the last step of the equation follows from the observation that for all $m, n \in \{0, 1, \ldots, i+1\}$ with $m \neq n$ we can find $j, k \in \{0, 1, \ldots, i+1\}$ such that

$$[x_0, x_1, \ldots, \underbrace{\hat{x_j}}_{m\text{th position}}, \ldots, \underbrace{\hat{x_k}}_{n\text{th position}}, \ldots, x_{i+1}]$$

and

$$[x_0, x_1, \ldots, \underbrace{\hat{x_k}}_{m\text{th position}}, \ldots, \underbrace{\hat{x_j}}_{n\text{th position}}, \ldots, x_{i+1}],$$

both appear in the sum, i.e. we always have a pair of summands where j and k have switched roles. In particular, these two summands are equal, thus by summing up these two expressions, we get

$$2 \cdot [x_0, x_1, \ldots, \hat{x_j}, \ldots, \hat{x_k}, \ldots, x_{i+1}]$$

which is equal to 0, since $C_i(X)$ is a vector space over the field $\mathbb{Z}_2$. Thus, the whole sum sums up to 0 and since the $(i+1)$-simplex s was arbitrary, the statement follows.

Now we are ready to give the main definition of this subsection.

Definition 2.27 A chain complex C over a field k is given by a choice of vector spaces C_i, $\forall i \geq 0$, together with linear transformations $\partial_i : C_i \to C_{i-1}$ such that $\partial_i \circ \partial_{i+1} \equiv 0 \; \forall i \geq 1$.

In particular, the i-chain vector spaces $C_i(X)$ together with the boundary maps ∂_i form a chain complex:

$$\ldots \xrightarrow{\partial_{i+2}} C_{i+1}(X) \xrightarrow{\partial_{i+1}} C_i(X) \xrightarrow{\partial_i} C_{i-1}(X) \xrightarrow{\partial_{i-1}} \ldots \xrightarrow{\partial_1} C_0(X) \xrightarrow{} 0.$$

2.2.2 Cycles, Boundaries, Homology

In this subsection, we want to define two particular types of chains which will help us define homology groups. Throughout the whole section, let X be an simplicial complex over $\mathbb{Z}_2$.

Definition 2.28 Let $c \in C_i(X)$ be an i-chain, where $i \geq 0$. We say that c is an i-cycle if $\partial_i c = 0$, i.e. if c has trivial boundary. The set of all i-cycles is denoted by $Z_i(X)$.

Remark 2.29 $Z_i = \ker(\partial_i)$, so in particular $Z_i(X)$ forms a subspace of $C_i(X)$.

Figure 2.7: A triangulated square.

Example 2.30 Consider the simplicial complex X with list of simplices

$$S(X) = \{A, B, C, D, AB, AC, BC, CD, ABC, ACD\},$$

The complex can be seen in Figure 2.7.

Here an example of a 1-chain is given by $c_1 = AB + BC + AC$. Note that

$$\partial_1 c_1 = A + B + B + C + A + C = 0.$$

Hence, c_1 is a 1-cycle. Consider now the 1-chain given by

$$c_2 = AB + AC + AD + BC + CD.$$

Using the boundary map, we get

$$\begin{aligned} \partial_1 c_2 &= A + B + A + C + A + D + B + C + C + D \\ &= A + C \\ &\neq 0. \end{aligned}$$

Thus, c_2 is not an 1-cycle. Both chains are depicted in Figure 2.8.

Definition 2.31 Let $c \in C_i(X)$ be an i-chain, where $i \geq 0$. We say that c is an i-boundary if there exists a $d \in C_{i+1}(X)$ such that $c = \partial_{i+1} d$. The set of all i-boundaries is denoted by $B_i(X)$.

Remark 2.32 $B_i(X) = \mathrm{im}(\partial_{i+1})$, so in particular $B_i(X)$ forms a subspace of $C_i(X)$.

Figure 2.8: 1-chains from Example 2.30.

Example 2.33 Consider again the simplicial complex from Example 2.30. An example on a 2-chain is given by $c_1 = ABC$. Note that

$$\partial_2 c_1 = AB + BC + AC =: d_1$$

Thus, d_1 defines a 1-boundary. Another example of a 2-chain of X is given by $c_2 = ABC + ACD$. We apply the boundary map ∂_2 and get

$$\begin{aligned}\partial_2 c_2 &= AB + AC + BC + AC + AD + CD \\ &= AB + BC + AD + CD \\ &=: d_2\end{aligned}$$

Hence, d_2 is also a 1-boundary. The chains and the corresponding boundaries are depicted in Figure 2.9.

Figure 2.9: 2-chains and their boundaries from Example 2.33.

Proposition 2.34 An i-chain s is an i-cycle if and only if every simplex in the boundary $\partial_i s$ appears an even number of times.

Proof Since our coefficients live in $\mathbb{Z}_2$, $\partial_i s$ will sum up to zero, if and only if every simplex in the sum appears to an even number of times.

From the fundamental lemma of homology (Lemma 2.26) it follows that $B_i(X) \subseteq Z_i(X)$. From this we can define a quotient space.

Definition 2.35 Let $i \geq 1$. Then we define the i-th homology group of X to be given by $H_i(X) := Z_i(X) / B_i(X)$. The rank $b_i := \operatorname{rank}(H_i(X))$ is called the i-th Betti number of X.

Example 2.36 One way to think of the Betti number is that it counts how many i-dimensional holes the space has: Consider the spaces X, Y, Z as given in Figure 2.10.

Figure 2.10: Spaces X, Y, Z from Example 2.36.

The space X has two connected components, i.e. two 0-dimensional holes. The components are given by one filled-in triangle and one filled-in square, in particular they do not contain anymore holes. Hence, the Betti numbers for the space X are

$$b_0(X) = 2, b_i(X) = 0, \forall i > 0.$$

The space Y has only one connected component, consisting of one filled-in square, two filled in triangles and two loops, therefore the Betti numbers for Y are

$$b_0(Y) = 1, b_1(Y) = 2, b_i(Y) = 0, \forall i > 1.$$

Lastly, the space Z has one connected component, consisting of the faces of a cube. In particular, the inside of the cube is not included, i.e. the space has a two dimensional hole. Thus, the Betti numbers for Z are

$$b_0(Z) = 1, b_1(Z) = 0, b_2(Z) = 1, b_i(Z) = 0, \forall i > 2.$$

Example 2.37 Let us come back to the example at the beginning of section 2.2.1, i.e. we are looking at the simplicial complex from Figure 2.6. The vector spaces formed by the i-chains are given as follows:

$$\begin{aligned} C_0(X) &= \langle A, B, C, D, E \rangle, \\ C_1(X) &= \langle AB, AD, BC, BE, CD, CE \rangle, \\ C_2(X) &= \langle BCE \rangle. \end{aligned}$$

Moreover, $\forall i \geq 3 : C_i = 0$. The boundary matrices are

$$\partial_2 = \begin{array}{c} \\ AB \\ AD \\ BC \\ BE \\ CD \\ CE \end{array} \begin{array}{c} BCE \\ \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \end{array}, \quad \partial_1 = \begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{cccccc} AB & AD & BC & BE & CD & CE \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix} \end{array}, \quad \partial_0 = (0).$$

One can easily check that

$$Z_0(X) = \mathrm{Ker}\,\partial_0 = \langle A, B, C, D, E\rangle,$$
$$Z_2(X) = \mathrm{Ker}\,\partial_2 = 0.$$

In order to get $Z_1(X)$, we have to work a little more: Note that our only candidates for cycles that could form a basis of $Z_1(X)$ would be

$$c_1 = AB + BC + CD + AD,$$
$$c_2 = BC + CE + BE,$$
$$c_3 = AB + BE + CE + CD + AD.$$

We notice that $c_1 + c_2 = c_3$. Also, it holds that

$$BE \in \langle AB + BC + CD + AD\rangle,$$
$$AD \in \langle BC + CE + BE\rangle.$$

Thus we conclude that

$$Z_1(X) = \langle AB + BC + CD + AD, BC + CE + BE\rangle.$$

Now we want to determine the boundaries.

$$B_2(X) = \mathrm{Im}\,\partial_3 = 0,$$
$$B_1(X) = \mathrm{Im}\,\partial_2 = \langle BC + CE + BE\rangle,$$
$$B_0(X) = \mathrm{Im}\,\partial_1 = \langle A + B, A + D, B + C, B + E, C + D, C + E\rangle.$$

Notice that

$$(B + C) + (B + E) = C + E.$$

Hence,

$$(C + E) \in \langle A + B, A + D, B + C, B + E, C + D\rangle.$$

Similarly,

$$(A + B) + (B + C) + (C + D) = A + D$$

implies that

$$(A + D) \in \langle A + B, B + C, B + E, C + D\rangle.$$

Thus, $B_0(X)$ simply es to

$$B_0(X) = \langle A + B, B + C, B + E, C + D\rangle.$$

Now we are ready to compute the homology groups.

$$\begin{aligned}
H_2(X) &= Z_2(X)/B_2(X) = 0,\\
H_1(X) &= Z_1(X)/B_1(X)\\
&= \langle AB+BC+CD+AD, BC+CE+BE\rangle / \langle BC+CE+BE\rangle\\
&= \langle AB+BC+CD+AD\rangle\\
&= Z_2,\\
H_0(X) &= Z_0(X)/B_0(X)\\
&= \langle A,B,C,D,E\rangle / \langle A+B, B+C, B+E, C+D\rangle\\
&= Z_2.
\end{aligned}$$

Furthermore, we are able to determine the Betti numbers.

$$\begin{aligned}
b_2 &= \operatorname{rank} H_2 = \operatorname{rank} Z_2 \quad \operatorname{rank} B_2 = 0,\\
b_1 &= \operatorname{rank} H_1 = \operatorname{rank} Z_1 \quad \operatorname{rank} B_1 = 2 \quad 1 = 1,\\
b_0 &= \operatorname{rank} H_0 = \operatorname{rank} Z_0 \quad \operatorname{rank} B_0 = 5 \quad 4 = 1.
\end{aligned}$$

Note that $\forall i \quad 3 : Z_i(X) = 0 = B_i(X)$, i.e. the homology groups are trivial, and $b_i = 0$.

In Section 3.4, we will provide an algorithm that can be used to determine the homology groups.

Another important property we want to mention is the functoriality of the homology groups. The functoriality property says that from every map of abstract simplicial complexes $f : X \to Y$ we can obtain an induced linear transformation $H_n(f) : H_n(X) \to H_n(Y)$ by showing the following:

(1) There are linear transformations $f_i : C_i(X) \to C_i(Y)$ which carry basis elements t of $C_i(X)$ to basis elements $f(t)$ of $C_i(Y)$.

(2) The boundary maps ∂_i respect the maps f_i in the sense that the following diagram commutes:

$$\begin{array}{ccc}
C_i(X) & \xrightarrow{f_i} & C_i(Y)\\
\downarrow{\scriptstyle \partial_i} & & \downarrow{\scriptstyle \partial_i}\\
C_{i-1}(X) & \xrightarrow{f_i} & C_{i-1}(Y)
\end{array}$$

(3) From (2) we conclude that f_i carries $Z_i(X)$ into $Z_i(Y)$ and $B_i(X)$ into $B_i(Y)$.

(4) From (3) it follows that there is an induced homomorphism

$$H_i(X) = Z_i(X)/B_i(X) \xrightarrow{H_i(f)} Z_i(Y)/B_i(Y) = H_i(Y).$$

Chapter 3

Persistent Homology

One goal of topological data analysis is to analyze a given dataset and get information about its topological features. Consider, for instance, the dataset from Figure 3.1. One can see that the set is sampled from a square. However, applying homology to the space of data points is not very insightful. We can only say that it is a space consisting of some points. In order to get more information about the space we have to apply persistent homology

Figure 3.1: Data sample of a square.

This section is based on Topological Data Analysis with Applications by Vejdemo-Johansson and Carlson ([6]), on Classification of hepatic lesions using the matching metric by Adcock, Rubin, Carlson ([1]), on Computational Topology for Data Analysis by Dey and Wang ([9]), as well as on Carlsson's Topological pattern recognition for point cloud data ([5]).

3.1 Filtrations

In this section we define simplicial complexes on our data in order to investigate it. We take a look at some examples that show how one can construct such complexes on a given dataset. In the first example, the data is a finite

subset of a metric space, while in the second one the data consists of black-and-white images. Both examples will serve as a motivation for the terms ltration and sublevel set ltrationof a complex.

3.1.1 Construction of Simplicial Complexes on Point Clouds

De nition 3.1 A point cloud is a nite subset of a metric space.

Example 3.2 The sample from Figure 3.1 forms a point cloud in $\mathbb{R}^2$.

Typically, the metric space in which our data points lie is just the Euclidean space $\mathbb{R}^n$. Our goal is to construct simplicial complexes on this type of dataset. One way to do this is by using nerves of coverings.

De nition 3.3 Let X be a topological space and $\mathcal{U} = \{U_1, \ldots, U_n\}$ be a covering of X, i.e. $X = \bigcup_{i=1}^{n} U_i$. We de ne the nerve of $\mathcal{U}$ to be the simplicial complex $N(\mathcal{U}) := (V_{\mathcal{U}}, S_{\mathcal{U}})$, where the vertex set $V_{\mathcal{U}}$ is given by $V_{\mathcal{U}} = \{x_1, \ldots, x_n\}$, and for $\{i_0, i_1, \ldots, i_k\} \subseteq \{1, \ldots, n\}$ we have that

$$\{x_{i_0}, x_{i_1}, \ldots, x_{i_k}\} \in S_{\mathcal{U}} \iff U_{i_0} \cap U_{i_1} \cap \cdots \cap U_{i_k} \neq \emptyset.$$

This notion is important in topological data analysis because from it one can conclude on important topological features by the so-called Nerve Lemma The lemma roughly states that if $\mathcal{U}$ is a nite cover of the space X and every non-empty intersection of the elements of the cover $\mathcal{U}$ is contractible, i.e. it is homotopy equivalent to a point, then $N(\mathcal{U})$ is homotopy equivalent to X. However, the lemma and its proof would go beyond the scope of this thesis and therefore we refer the interested reader to [3].

Example 3.4 Let $X \subseteq \mathbb{R}^2$ be s space and $\mathcal{U} = \{U_1, U_2, U_3, U_4\}$ a covering of X, as seen in Figure 3.2.

Figure 3.2: (a) A covering $\mathcal{U}$ of a space X and (b) its nerve $N(\mathcal{U})$.

In the nerve $N(U)$ each vertex correspond to one element of the covering. Furthermore, we have the following non empty intersection of the elements $U_1, \dots, U_4$:

$$U_1,\ U_2,\ U_3,\ U_4,$$
$$U_1 \cap U_2,\ U_1 \cap U_3,\ U_2 \cap U_3,\ U_2 \cap U_4,\ U_3 \cap U_4,$$
$$U_1 \cap U_2 \cap U_3.$$

Hence the set of simplices of $N(U)$ is given by

$$S_U = \{x_1, x_2, x_3, x_4, x_1x_2, x_1x_3, x_2x_3, x_2x_4, x_3x_4, x_1x_2x_3\}.$$

This idea of using a covering of a topological space to get a simplicial complex can now be used to get a simplicial complex on a point cloud. In the case of the Cech complex we use closed balls around the data points to get a covering.

Definition 3.5 Let (M, d) be a metric space, where M is a set and d is a metric on M. Let $X \subseteq M$ be a finite subset of M. For a fixed $r \in \mathbb{R}_{>0}$ we define the Cech complex $C(r, X)$ to be the nerve of the covering $U = \{\overline{B}(x, r)\}_{x \in X}$, where

$$\overline{B}(x, r) = \{y \in M \mid d(x, y) \leq r\}$$

is the closed ball of radius r and center $x \in X$.

Hence, for a given point cloud $X \subseteq \mathbb{R}^n$ and given $r > 0$, we get a simplicial complex the by the following procedure:

- Every data point $x \in X$ corresponds to a vertex in $C(r, X)$. By abuse of notation, we will write x for both the data point and the corresponding vertex.
- Around every data point $x \in X$, draw a closed ball of radius r, i.e. $\overline{B}(x, r)$.
- If $\overline{B}(x_0, r) \cap \overline{B}(x_1, r) \cap \dots \cap \overline{B}(x_k, r) \neq \emptyset$, then the k-simplex $x_0x_1 \dots x_k$ is included in the complex $C(r, X)$.

It is important to point out that different values of $r > 0$ give rise to different Cech complexes as the following example illustrates:

Example 3.6 Consider the point cloud $X \subseteq \mathbb{R}^2$ given by

$$X = \{(0, 0), (1, 0), (1, 1)\} =: \{x_1, x_2, x_3\},$$

i.e. the vertices of a rectangular triangle. As we can see in Figure 3.3, for $r < \frac{1}{2}$ none of the balls will intersect, i.e. for $r < \frac{1}{2}$ the Cech complex $C(r, X)$ consists only of the vertices x_1, x_2, x_3 and no other simplices. At $r = \frac{1}{2}$ the balls $\overline{B}(x_1, \frac{1}{2}), \overline{B}(x_2, \frac{1}{2})$ and $\overline{B}(x_2, \frac{1}{2}), \overline{B}(x_3, \frac{1}{2})$ each intersect in one point, thus for all $r < \frac{\sqrt{2}}{2}$ we

Figure 3.3: Different Cech complexes for different radii r.

include the 1-simplices x_1x_2 and x_2x_3 in the complex $C(r, X)$. At $r = \frac{\sqrt{2}}{2}$ all balls will intersect at one point, thus we get additionally the simplex x_1x_3 as well as the 2-simplex $x_1x_2x_3$. Since no other intersection is possible, we conclude that $\forall r \neq \frac{\sqrt{2}}{2} : C(r, X) = C(\frac{\sqrt{2}}{2}, X)$.

Definition 3.7 Let $X = (V(X), S(X))$ be a simplicial complex. Let $\{X_r\}_{r \in R}$ be a sequence of subcomplexes such that

- $|R|$ is finite,
- $\forall r, r' \in R, r < r' : X_r \subseteq X_{r'}$,
- $\exists r \in R : X_r = X$.

Then $\{X_r\}_{r \in R}$ is called a *filtration* of the complex X.

Example 3.8 Since for $0 < r < r'$ and $x, y \in X$ we have that

$$\bar{B}(x, r) \cap \bar{B}(y, r) \neq \emptyset \Rightarrow \bar{B}(x, r') \cap \bar{B}(y, r') \neq \emptyset,$$

we immediately see that every simplex $\sigma \in C(r, X)$ is also included in the complex $C(r', X)$. In particular, we have the inclusion

$$C(r, X) \subseteq C(r', X).$$

Hence, for a finite subset $R \subseteq \mathbb{R}$, we get that $\{C(r, X)\}_{r \in R}$ is a filtration.

3.1.2 Image Filtrations

We want to define simplicial complexes on data sets consisting of images. With the help of the complexes, as well as the notion of $\mathbb{R}$-filtered sets, we are able to look for similarities among images and classify them.

First of all, we restrict ourselves to black-and-white images. Each image I in our dataset can be considered as a two-dimensional collection of pixels, each of which endowed with a value, the so-called grayscale or intensity value of the pixel. We denote $I := (P, g)$, where P is the set of pixels of I and $g: P \to R$ is the function that assigns to each pixel $p \in P$ its grayscale value $g(p) \in R$. We call g the grayscale function of the image I. Next we define a simplicial complex on the set of pixels.

Definition 3.9 We define the pixel complex $K := (V(K), S(K))$ of a given image $I = (P, g)$ to be the complex which is derived as follows:

- Each pixel in $p \in P$ corresponds to a vertex $x_p \in V(K)$.
- If two pixels $p_1, p_2 \in P$ are adjacent in I (where we treat diagonal pixels as adjacent), we add the 1-simplex $x_{p_1} x_{p_2}$ to the complex, i.e. $x_{p_1} x_{p_2} \in S(K)$. Similarly, if three pixels $p_1, p_2, p_3 \in P$ are adjacent in I, then we add the 2-simplex $x_{p_1} x_{p_2} x_{p_3}$ to the list of simplices.

Remark 3.10 We emphasise that we do not include simplices with $r \geq 3$. In particular, if $p_1, p_2, p_3, p_4 \in P$ are adjacent in I, we do not add the simplex $x_{p_1} x_{p_2} x_{p_3} x_{p_4}$ to $S(K)$.

Example 3.11 In Figure 3.4 one can see the image I and next to it the image together with the constructed pixel complex. Note that the pixel complex does not depend on the grayscale values.

Figure 3.4: The image I on the left and its pixel complex on the right.

The pixel complex is a very regular complex, which does not contain much information. Indeed, if P_1, P_2 are two different pixel complexes, the only variation between them is given by the boundary shape of the corresponding image. In order to make any use of this construction we have to consider a filtration of the complex.

Definition 3.12 Let X be any set, equipped with a function $r: X \to R$. Then we call r a filtration function and the pair (X, r) an R-filtered set.

Example 3.13 A grayscale function g of an image $I = (P, g)$ is an example for a filtration function, hence an image is a $\mathbb{R}$-filtered set.

Definition 3.14 Let $X = (V(X), S(X))$ be a simplicial complex and $f: V(X) \to \mathbb{R}$ be a filtration function on the vertex set. Let $f_{min} := \min\{f(v) \mid v \in V(X)\}$ and $f_{max} := \max\{f(v) \mid v \in V(X)\}$. Furthermore, for $i \in \mathbb{R}$ let $X_{f,i} \subseteq X$ be a subcomplex, given by $X_{f,i} = (V(X), S_{f,i}(X))$, where

$$S_{f,i}(X) := \{S \in S(X) \mid \forall v \in S : f(v) \leq i\}.$$

Let $i_1 < i_2 < \dots$ be an increasing sequence of positive real values. Then we define the *sublevel set filtration* of the complex X to be the sequence

$$X_{f,f_{min}} \subseteq X_{f,f_{min}+i_1} \subseteq X_{f,f_{min}+i_2} \subseteq \dots \subseteq X_{f,f_{max}} = X.$$

Remark 3.15 It is also possible to reverse the inequality in Definition 3.14, i.e.

$$S_{f,i}(X) := \{S \in S(X) \mid \forall v \in S : f(v) \geq i\}$$

We would end up with an equally valid filtration

$$X_{f,f_{max}} \subseteq X_{f,f_{max}-i_1} \subseteq X_{f,f_{max}-i_2} \subseteq \dots \subseteq X_{f,f_{min}} = X.$$

If we define $S_{f,i}(X)$ as in Definition 3.14 we will also refer to it as an *increasing sublevel set filtration*, in the case where we defined it with the reversed inequality we will call it a *decreasing sublevel set filtration*.

Hence, a given filtration function will decide when a specific simplex of the initial complex will appear (or disappear) in our filtration. Since we want to analyze images one obvious candidate as our filter function of our pixel complex K is the grayscale function g of the corresponding image I.

Definition 3.16 Let $I = (P, g)$ be an image and $K = (V(X), S(K))$ its pixel complex. For a given increasing sequence of real values $i_1 < i_2 < \dots$ we define the *intensity filtration* to be the sequence

$$K_{g,g_{min}} \subseteq K_{g,g_{min}+i_1} \subseteq K_{g,g_{min}+i_2} \subseteq \dots \subseteq K_{g,g_{max}} = K,$$

where $K_{g,i} = (V(K), S_{g,i}(K))$ with $S_{g,i}(K) = \{S \in K \mid \forall v \in V(K) : g(v) \leq i\}$.

Example 3.17 In Figure 3.5, one can see an example of the intensity filtration of an image $I = (P, g)$, where the function $g: P \to \mathbb{R}$ takes only values in $\{1, 2, 3\}$. We wrote inside each pixel the respective grayscale value. We will then get a filtration

$$K_{g,1} \subseteq K_{g,1} \subseteq K_{g,3} = K,$$

where K denotes the pixel complex of I.

Figure 3.5: Intensity filtration of an image $I = (P, g)$.

3.2 Persistent Homology

We have already seen in Section 3.1.1 that it is possible to define a simplicial complex on a given point cloud. Moreover, with the notion of homology, we are able to investigate the features, more precisely the holes, of a given complex. Thus, by computing the homology groups of a simplicial complex of a point cloud, we might be able to draw conclusion on some topological features of our dataset. However, some problems might occur while constructing and analyzing these complexes, as the following two examples will show: In the first one, we will see that it can be hard to construct a complex that properly captures the topological features of the object our data sample is sampled from, while in the second example we will see how errors in a data sample might affect the analysis.

Example 3.18 Assume the space X we want to investigate is the boundary of a square and the sample X_ε is given by the four corners, as seen in Figure 3.6. We want to reconstruct the space X with the help of the Čech complex. If we choose the radii of the balls too small, the Čech complex consists only of four vertices. If we enlarge the radius such that each closed ball intersects with exactly two other balls, the corresponding complex consists of four vertices, each of which is connected to

Figure 3.6: Data sample from a square.

exactly two other vertices by an edge. Hence, the resulting complex corresponds to the original space. But as soon as the radius is too big, we get additional simplices which might even fill the square in. In particular, this space would then be contractible, i.e. homotopy equivalent to a point, but the original space X is not. Examples of different Čech complexes for different parameter values r are depicted in Figure 3.7.

Figure 3.7: Attempts to reconstruct the square out of the data sample.

Example 3.19 Sometimes it can happen that while getting a sample, one includes measuring errors, often called noise. Consider again the space X which is given by the boundary of a square. This time, our sample the of the space X includes a point v_4 that does not lie on the original square, as seen in Figure 3.8. We set r such that each closed ball around the corner vertices intersects exactly two other balls around the corners. Unlike in the previous example, we do not get a space that resembles the square anymore. Instead, we get one that consists of a square and a triangle, caused by the intersection of the ball around the noise point with the balls around the vertices that correspond to the corners of the square. But, if we choose the radius smartly, as done in Figure 3.9 with r_3, we can get a Čech complex such that the

Figure 3.8: Data sampling of a square with noise point.

triangle is filled in by a simplex, but the square is still not. This might be interesting from a topological point of view, since this filled-in triangle would be contractible. I.e. if we contract the triangle, the complex $C(r_3, X)$ could be deformed in one with only one (non-contractible) 2-dimensional hole, like in the original space X.

Figure 3.9: Attempts to reconstruct the square out of the data sampling with noise.

What both examples show is that it is hard to find the "right" value r (if it even exists) for a complex that captures best the "true" shape of the dataset X. If the value is too small, some topological features might not have appeared yet, whereas if we choose it too large, we might have already lost information. Persistent homology tackles these issues by considering a filtration of a given complex and keeping track of when some topological feature (holes) appear and when they disappear, with the interpretation that the longer a certain feature "lives", i.e. the hole is not filled in, the more likely it is that the original space also has this feature.

3.2.1 Persistent Vector Spaces

Definition 3.20 Let k be any field. A *persistent vector space* over k is a family of k-vector spaces $\{V_r\}_{r \in \mathbb{R}}$ together with linear transformations $L_V(r, r') : V_r \to V_{r'}$, where $r \leq r'$, such that

$$L_V(r', r'') \circ L_V(r, r') = L_V(r, r'') \quad \forall r \leq r' \leq r''.$$

A *sub-persistence vector space* of $\{V_r\}_{r \in \mathbb{R}}$ is a family $\{U_r\}_{r \in \mathbb{R}}$ of k-subspaces $U_r \subseteq V_r$ such that

$$L_V(r, r')(U_r) \subseteq U_{r'} \quad \forall r \leq r'.$$

Example 3.21 Consider $\{C(r, X)\}_{r \in \mathbb{R}}$, where we set $C(r, X) = 0, \forall r < 0$. By Example 3.8 we have that

$$C(r, X) \subseteq C(r', X), \text{ whenever } r \leq r'.$$

Let $i \in \mathbb{N}$. By applying the i-th homology group H_i to the family $\{C(r, X)\}_{r \in \mathbb{R}}$, we obtain a family $\{H_i(C(r, X))\}_{r \in \mathbb{R}}$ of vector spaces, which by the functoriality of homology has the structure of a persistent vector space, where the linear transformations $L_V(r, r')$ are induced by the inclusion maps $C(r, X) \hookrightarrow C(r', X)$.

Definition 3.22 A *linear transformation* $f : \{V_r\}_{r \in \mathbb{R}} \to \{W_r\}_{r \in \mathbb{R}}$ of persistent vector spaces $\{V_r\}_{r \in \mathbb{R}}, \{W_r\}_{r \in \mathbb{R}}$ over k is a family of linear transformations $f_r : V_r \to W_r$ with the property that for all pairs (r, r') such that $r \leq r'$, the diagrams

$$\begin{array}{ccc} V_r & \xrightarrow{L_V(r,r')} & V_{r'} \\ {\scriptstyle f_r}\downarrow & & \downarrow{\scriptstyle f_{r'}} \\ W_r & \xrightarrow{L_W(r,r')} & W_{r'} \end{array}$$

commute, meaning

$$f_{r'} \circ L_V(r, r') = L_W(r, r') \circ f_r.$$

Definition 3.23 If $f : \{V_r\}_{r \in \mathbb{R}} \to \{W_r\}_{r \in \mathbb{R}}$ is a linear transformation of persistent vector spaces, the *image* of f, denoted by $\mathrm{im}(f)$, is the sub-persistent vector space $\{\mathrm{im}(f)_r\}_{r \in \mathbb{R}}$. Moreover, we call f an *isomorphism* if it admits a two-sided inverse.

It is possible to extend the notion of quotient spaces to persistent vector spaces:

Definition 3.24 Let $\{U_r\}_{r \in \mathbb{R}}$ be a sub-persistence vector space of $\{V_r\}_{r \in \mathbb{R}}$. Then we can form the *quotient space* of the persistent vector space $\{V_r / U_r\}_{r \in \mathbb{R}}$, where the linear transformations $L_{V/U}(r, r') : V_r / U_r \to V_{r'} / U_{r'}$, for $r \leq r'$, are given by sending $[v] \in V_r / U_r$ to the equivalence class $[L_V(r, r')(v)] \in V_{r'} / U_{r'}, \forall v \in V_r$.

Next we extend the notion of a free vector space to persistent vector spaces:

De nition 3.25 By the free persistent vector space on the pair (X, r), where X is a set and $r : X \to \mathbb{R}$ is a ltration function, we mean the persistent vector space $\{V_k(X, r)_r\}_{r \in \mathbb{R}}$, where $V_k(X, r)_r \quad V_k(X)$ is the k-linear span of the set $\{x \in X \mid r(x) \quad r\}$. We say that a persistent vector space is free if it is isomorphic to one of the form $V_k(X, r)$ for some (X, r). Moreover, if X is nite, we say $V_k(X)$ is nitely generated .

Remark 3.26 If X is nite, we can nd some suf ciently large $r \in \mathbb{R}$ such that $V_k(X, r)_r = V_k(X)$. For instance, take $r = \max_{x \in X} r(x)$.

Remark 3.27 It often happens that one wants to restrict the function r to the non-negative real numbers $\mathbb{R}_{\geq 0} = [0, \infty)$. In this case, we can still work with the de nition above by simply setting $V_k(X, r)_r = 0, \forall r < 0$.

Proposition 3.28 A linear combination $\sum_{x \in X} a_x x \in V_k(X)$ lies in $V_k(X, r)_r$ if and only if $a_x = 0 \ \forall x \in X$ with $r(x) > 0$.

Proof Let $W := \sum_{x \in X} a_x x$ be some linear combination in $V_k(X)$ and $r \in \mathbb{R}$ xed. First assume there exists an $x \in X$ with $r(x) > r$ and $a_x \neq 0$. Then, since x does not lie in the span of the set $\{x \in X \mid r(x) \quad r\}$, the linear combination W cannot lie in $V_k(X, r)_r$. Conversely, if for every $x \in X$ with $r(x) > r$ it holds that $a_x = 0$, we can rewrite W as:

$$\sum_{x \in X} a_x x = \sum_{\substack{x \in X \\ r(x) \quad r}} a_x x.$$

Since in the last sum we are only summing over elements which lie in the span of $\{x \in X \mid r(x) \quad r\}$, we conclude that $W \quad V_k(X, r)_r$.

Example 3.29 We construct a ltration that does arise from a nite metric space. For this, consider the simplical complex $X = (V(X), S(X))$ with the set of vertices $V(X) = \{a, b, c, d\}$ and the set of simplices

$$S(X) = \{a, b, c, d, ab, ac, ad, bc, ad, abc\}.$$

We lter the complex X as given in Figure 3.10. At time $T_0 = 0$, we only have the 0-simplices a and b. At time T_1, we add the 0-simplices c and d, as well as the 1-simplices ab and bc, and so on. We want to compute the persistent vector spaces consisting of families of the i-chains $_i(X)$:

Figure 3.10: Filtration of the complex X from Example 3.29 at times $T_0, T_1, \ldots, T_4$.

$$C_0(X)_r = \begin{cases} \{0\}, & \text{if } r < T_0 \\ \langle a, b \rangle, & \text{if } T_0 \leq r < T_1, \\ \langle a, b, c, d \rangle, & \text{if } T_1 \leq r, \end{cases}$$

$$C_1(X)_r = \begin{cases} \{0\}, & \text{if } r < T_1, \\ \langle ab, bc \rangle, & \text{if } T_1 \leq r < T_2, \\ \langle ab, bc, ad, cd \rangle, & \text{if } T_2 \leq r < T_3, \\ \langle ab, bc, ad, cd, ac \rangle, & \text{if } T_3 \leq r, \end{cases}$$

$$C_2(X)_r = \begin{cases} \{0\}, & \text{if } r < T_4, \\ \langle abc \rangle, & \text{if } T_4 \leq r. \end{cases}$$

Definition 3.30 A persistent vector space is *finitely presented* if it is isomorphic to a persistent vector space of the form $\{W_r\}_{r \in R} / \operatorname{im}(f)$ for some linear transformation $f : \{V_r\}_{r \in R} \to \{W_r\}_{r \in R}$ between finitely generated free persistent vector spaces $\{V_r\}_{r \in R}$ and $\{W_r\}_{r \in R}$.

The choice of a basis of two vector spaces V, W allows us to represent linear transformations $f : V \to W$ by matrices. We wish to have a similar representation of linear transformations between persistent vector spaces:

Definition 3.31 Let (X, Y) be a pair of finite sets and k be a field. A (X, Y)-*matrix* is an array $[a_{xy}]$ of elements $a_{xy} \in k$, indexed by the elements $x \in X, y \in Y$.

Definition 3.32 Consider two R-filtered finite sets $(X, r), (Y, s)$ and the field k. We say that a (X, Y)-matrix $A = [a_{xy}]$ with entries $a_{xy} \in k$ is (r, s)-*adapted* if $a_{xy} = 0$ whenever $r(x) > s(y)$.

Example 3.33 Consider the filtered simplicial complex from Example 3.29. We set $r : S(X) \to R$ such that $r(s) = i$ if s appears in the filtration at time T_i. We

compute the boundary matrices of X:

$$\partial_1 = \begin{array}{c} \\ C \\ D \\ B \\ A \end{array} \begin{array}{c} \begin{array}{ccccc} BC & AB & AD & CD & AC \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array}, \partial_2 = \begin{array}{c} \\ BC \\ AB \\ AD \\ CD \\ AC \end{array} \begin{array}{c} \begin{array}{c} ABC \end{array} \\ \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{array}.$$

Note that if the simplex t is in the boundary simplex s, t must appear no later than s in the filtration, i.e. $r(t) \leq r(s)$. Hence, ∂_1 is (r_0, r_1)-adapted and ∂_2 is (r_1, r_2)-adapted, where r_i denotes the restriction of r to the set of i-simplices of X.

We consider an (X,Y)-matrix for X, Y finite and denote the row corresponding to $x \in X$ by $r(x)$ and the column corresponding to $y \in Y$ by $c(y)$. For the finitely generated free persistent vector spaces $\{V_k(X,r)_r\}_{r \in R}$ and $\{V_k(Y,s)_r\}_{r \in R}$, where $r : X \to R$ and $s : Y \to R$, we know by Remark 3.26 that we can find an $r \in R$ sufficiently large such that $V_k(X,r)_r = V_k(X)$ and $V_k(Y,s)_r = V_k(Y)$. Thus, for any linear transformation f from $\{V_k(Y,s)_r\}_{r \in R}$ to $\{V_k(X,r)_r\}_{r \in R}$, we can obtain a linear transformation $f_\infty : V_k(Y) \to V_k(X)$ between finite-dimensional vector spaces over k. By fixing the bases $\{x\}_{x \in X}$ of $V_k(X)$ and $\{y\}_{y \in Y}$ of $V_k(Y)$, we can represent f as an (X,Y)-matrix $A(f) = [a_{xy}]$ with entries in k.

Proposition 3.34 *The (X,Y)-matrix $A(f)$ is (r,s)-adapted and any (X,Y)-matrix A which is (r,s)-adapted uniquely determines a linear transformation of persistent vector spaces*

$$f_A : \{V_k(Y,s)_r\}_{r \in R} \to \{V_k(X,r)_r\}_{r \in R}.$$

In particular the correspondences $f \mapsto A(f)$ and $A \mapsto f_A$ are inverse to each other.

Proof Note that basis vector $y \in Y$ lies in $V_k(Y,s)_{s(y)}$. On the other hand we have that

$$f(y) = \sum_{x \in X} a_{xy} x.$$

Using proposition 3.28 we can see that $\sum_{x \in X} a_{xy} x$ lies in $V_k(X,r)_{s(y)}$ if and only if all coefficients $a_{xy} = 0$ if $r(x) > s(y)$, i.e. if the matrix $A = [a_{xy}]$ is (r,s)-adapted.

Proposition 3.35 *Let X, Y be finite and the pairs $(X,r), (Y,s)$ be R-filtered sets. Let $A = [a_{xy}]$ be a (r,s)-adapted (X,Y)-matrix. Then A determines a persistent vector space via the correspondence*

$$A \mapsto \{(V_k(X,r) / \mathrm{im}(f_A))_r\}_{r \in R},$$

where f_A is the uniquely determined transformation of vector spaces from Proposition 3.34. Moreover, this space is finitely presented.

Proof This follows immediately from the correspondence between linear transformations and matrices as seen in Proposition 3.34.

Remark 3.36 We often just write $q(A) = \{q(A)_r\}_{r \in R}$ for the quotient space $\{(V_k(X,r)/\mathrm{im}(f_A))_r\}_{r \in R}$.

The next statement will give us a criterion for when $q(A)$ will be equal to $q(A')$, where A, A' are two matrices. This turns out to be very useful, especially when proving the main statement in the next section.

Proposition 3.37 Let (X,r) and (Y,s) be R-filtered sets, and A be a (r,s)-adapted (X,Y)-matrix, with entries in a field k. Let B be a (r,r)-adapted (X,X)-matrix, and C be a (s,s)-adapted (Y,Y)-matrix, both matrices with entries in k. Then the matrix BAC is (r,s)-adapted, and the persistent vector space $q(A)$ is isomorphic to $q(BAC)$.

Proof First we want to show that the matrix AC is (r,s)-adapted. Write $A = [a_{xy}]$, where $x \in X, y \in Y$, and $C = [c_{y\tilde{y}}]$, where $y, \tilde{y} \in Y$. Note that the rows of $AC = [d_{xy}]$ correspond to the elements of X, and the columns to the elements of Y. Furthermore, we have

$$d_{xy} = \sum_{\tilde{y} \in Y} a_{x\tilde{y}} c_{\tilde{y}y}.$$

Since A is (r,s)-adapted, we have that $a_{x\tilde{y}} = 0$, whenever $r(x) > s(\tilde{y})$. Similarly, $c_{\tilde{y}y} = 0$ whenever $s(\tilde{y}) > s(y)$. In particular, we can write

$$d_{xy} = \sum_{\substack{\tilde{y} \in Y \\ r(x)\ s(\tilde{y})\ s(y)}} a_{x\tilde{y}} c_{\tilde{y}y}.$$

From this, one can immediately see that $d_{xy} = 0$ whenever $r(x) > s(y)$, thus the matrix AC is (r,s)-adapted. Analogously one can show that the matrix BAC is (r,s)-adapted.

We are left to show that $q(A) = q(BAC)$. First note, since $(f_{BAC})_r$ is induced by matrices, we have that $(f_{BAC})_r = (f_B)_r \circ (f_A)_r \circ (f_C)_r$, for all $r \in R$, denoted as $f_{BAC} = f_B \circ f_A \circ f_C$. Furthermore, the maps $(f_B)_r$ and $(f_C)_r$ define isomorphisms on $V_k(X,r)_r$ and $V_k(Y,s)_r$ respectively. Now we fix some $r \in R$ and consider $q(A)_r = (V_k(X,s)/\mathrm{im}(f_A))_r$ and $q(BAC)_r = (V_k(X,s)/\mathrm{im}(f_{BAC}))_r$. Consider the map

$$\tilde{q}_r : (V_k(X,r)/\mathrm{im}(f_A))_r \to (V_k(X,r)/\mathrm{im}(f_{BAC}))_r,$$

which maps the equivalence class $[x] \in (V_k(X,r)/\mathrm{im}(f_A))_r$ to the equivalence class $[(f_B)_r(x)] \in (V_k(X,r)/\mathrm{im}(f_{BAC}))_r$. Consider now some $x \in V_k(X,r)_r$ and some $y \in \mathrm{im}((f_A)_r)$, i.e. $\exists z \in V_k(Y,s)_r$ such that $(f_A)_r(z) = y$. Using the

linearity of $(f_B)_r$ we get

$$\begin{aligned}(f_B)_r(x+y) &= (f_B)_r(x) + (f_B)_r(y)\\ &= (f_B)_r(x) + (f_B)_r \circ (f_A)_r(z)\\ &= (f_B)_r(x) + (f_B)_r \circ (f_A)_r \circ (f_C)_r \circ (f_C)_r^{-1}(z)\\ &= (f_B)_r(x) + (f_B)_r \circ (f_A)_r \circ (f_C)_r(w),\end{aligned}$$

where we used in the last step that $(f_C)_r$ is an isomorphism, therefore we can find some $w \in V_k(Y,s)_r$ such that $(f_C)_r(w) = z$. In particular, we can see from this equation that the equivalence classes will be preserved under $\tilde{q}$, i.e.

$$\tilde{q}_r([x]) = [x].$$

Thus, since the $(f_B)_r$ are isomorphisms, we have that $\tilde{q} = \{\tilde{q}_r\}_{r\in R}$ is a family of well defined isomorphisms. In particular, we have that

$$q(A) = q(BAC).$$

In order to perform matrix operations on adapted matrices, we need to clarify which operations are allowed, i.e. if a matrix is (r,s)-adapted we only want to allow operations that conserve the (r,s)-adapted property.

Definition 3.38 Let $(X,r),(Y,s)$ be two R-filtered sets and A be a (r,s)-adapted matrix. We define an *adapted row operation* to be an operation that adds multiples of $r(x)$ to $r(x')$ whenever $r(x) \leq r(x')$, where $x, x' \in X$. Similarly, we define an adapted column operation to be an operation which adds multiples $c(y)$ to $c(y')$, whenever $s(y) \leq s(y')$ where $y, y' \in Y$.

Remark 3.39 The matrix B from Proposition 3.37 corresponds to adapted row operations applied to the matrix A. Similarly, the matrix C corresponds to adapted column operations.

3.3 Structure Theorem for Persistent Vector Spaces

In this section, we will present a result that enables us to classify all finitely presented persistent vector spaces up to isomorphism. We will later make use of it while introducing the notion of *barcodes* and persistent diagrams.

Definition 3.40 Let k be a field. Choose $a \in R$ and $b \in R \cup \{+\infty\}$ such that $a < b$. Let $P(a,b) = \{P(a,b)_r\}_{r\in R}$ be the persistent vector space defined by

$$P(a,b)_r = \begin{cases} k, & \text{if } r \in [a,b),\\ \{0\}, & \text{if } r \notin [a,b),\end{cases}$$

where the linear transformations $L_{P(a,b)}(r,r')$, with $r < r'$, are given by

$$L_{P(a,b)}(r,r') = \begin{cases} id_k, & \text{if } r,r' \in [a,b), \\ 0, & \text{otherwise.} \end{cases}$$

We call $P(a,b)$ the *interval persistence vector space* of the pair (a,b).

Note that $P(a,b)$ is finitely presented. The next example underlines the basic concept of the upcoming proof.

Example 3.41 Let a,b be as in Definition 3.40. Assume b is finite, i.e. $b \neq \infty$. Let (X,r) and (Y,s) be $\mathbb{R}$-filtered sets, where the sets X and Y consist only of one element, i.e. $X = \{x\}, Y = \{y\}$, with $r(x) = a, s(y) = b$. Consider the (1×1)-matrix

$$A = \begin{matrix} \\ (x,a) \end{matrix} \overset{(y,b)}{\begin{pmatrix} 1 \end{pmatrix}}$$

One can think of A as the matrix, that maps the element $y \in Y$, which appears at $s(y) = b$, to the value $x \in X$, which appears at $r(x) = a$. Note that this matrix is a (r,s)-adapted (X,Y)-matrix, since $a \leq b$. Next we want to determine the persistent vector spaces $\{V_k(X)_r\}_{r \in \mathbb{R}}$.

$$V_k(X)_r = \begin{cases} \{0\} & \text{if } r < a, \\ k & \text{if } r \geq a. \end{cases}$$

Moreover, we have for the image of the corresponding linear map f_A that

$$\operatorname{im}(f_A)_r = \begin{cases} 0, & \text{if } r < b, \\ k, & \text{if } r \geq b. \end{cases}$$

In conclusion, we get that

$$q((1))_r = q(A)_r = (V_k(X)/\operatorname{im}(f_a))_r = \begin{cases} k, & \text{if } r \in [a,b), \\ \{0\}, & \text{if } r \notin [a,b), \end{cases}$$

Hence, $q((1))$ is isomorphic to $P(a,b)$. Now assume that $b = \infty$. Then we have that the image of the map f_A is given by

$$\operatorname{im}(f_A)_r = 0, \forall r \in \mathbb{R}.$$

Hence we have that $q(A)$ is isomorphic to $V(X,r)$. But also $P(a,b)$ is in that case isomorphic to $V(X,r)$. In total we get that

$$\forall a \in \mathbb{R}, b \in \mathbb{R} \cup \{+\infty\} : q(A) = P(a,b).$$

We are now ready to state the main theorem of this section.

Lemma 3.42 (Structure Theorem) Every finitely presented vector space $\{V_r\}_{r\in R}$ over a field k is isomorphic to a finite sum of the form

$$\bigoplus_{i=1}^{n} P(a_i, b_i) = P(a_1, b_1) \oplus P(a_2, b_2) \oplus \cdots \oplus P(a_n, b_n), \tag{3.1}$$

where $a_i \in R$, $b_i \in R \cup \{+\infty\}$ and $a_i < b_i$ for all $i = 1, 2, \ldots, n$.

Proof By the correspondence between finitely presented persistent vector spaces and adapted matrices (Proposition 3.34) we will use the latter in order to prove our statement. For this, first consider a (r, s)-adapted (X, Y)-matrix A such that every row and column has at most one non-zero entry, which is equal to 1. Hence, w.l.o.g we can assume A is of the form

$$A = \begin{matrix} I_n & 0 \\ 0 & 0 \end{matrix},$$

where I_n denotes the identity matrix. Otherwise, we could switch rows and columns of A, until it is in the desired form. Let $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$ be the pairs (x_i, y_i) such that $a_{x_i y_i} = 1$. We then obtain the following decomposition

$$\begin{aligned} V_k(X,r)/\mathrm{im}(f_A) &= \bigoplus_{x\in X} V_k(x,r)/\mathrm{im}(f_A) \\ &= V_k(x_1,r)/\mathrm{im}(f_A) \oplus \cdots \oplus V_k(x_n,r)/\mathrm{im}(f_A) \oplus \bigoplus_{x\in X\setminus\{x_1,\ldots,x_n\}} V_k(x,r)/\mathrm{im}(f_A) \\ &= P(r(x_1), s(y_1)) \oplus \cdots \oplus P(r(x_n), s(y_n)) \oplus \bigoplus_{x\in X\setminus\{x_1,\ldots,x_n\}} P(r(x), +\infty). \end{aligned}$$

where the last equality follows from the fact that $V_k(x_i,r)/\mathrm{im}(f_A)$ only depends on the image of the values y_i and from the computation from Example 3.41. Now we consider some general finitely presented vector space $\{V_r\}_{r\in R} = V_k(X,r)/\mathrm{im}(f_A)$, for some linear transformation f and A the corresponding matrix from Proposition 3.34. If we are able to find a (r, r)-adapted (X, X)-matrix B and a (s, s)-adapted (Y, Y)-matrix C such that BAC has the property that each column and row have at most one non-zero entry, which is one, we are done, since by Proposition 3.37, we have that $q(A) = q(BAC)$, i.e. $q(A)$ would be isomorphic to the form described in the statement. To find such matrices B and C, we will use adapted row and column operations. Note that the (r, s)-adapted operations consist of

- all possible multiplications of a row or a column by a non-zero element of k,

- all possible additions of a multiple of $r(x)$ to $r(x')$ if $r(x) \leq r(x')$,
- all possible additions of a multiple of $c(y)$ to $c(y')$ if $s(y) \leq s(y')$.

Now we look for a $y \in Y$ such that $c(y)$ has at least one non-zero entry and the value $s(y)$ is minimized. Such a y can be found since Y is finite. Next we find some $x \in X$ such that $a_{xy} \neq 0$ and $r(x)$ is maximized. By the way we have chosen our x we are able to add multiples of the row $r(x)$ to the other rows such that the entries of the column $c(y)$ will be equal to zero, except for the entry a_{xy}. Similarly, the way we have chosen y enables us to add multiples of the column $c(y)$ to the other columns until the row $c(x)$ has only zero-entries, except the entry a_{xy}. Last but not least, we multiply the row $r(x)$ by the value $\frac{1}{a_{xy}} \in k$. The result is a matrix in which both $r(x)$ and $r(y)$ have exactly one non-zero entry $a_{xy} = 1$. By deleting $r(x)$ and $c(y)$ we end up with a $(X \setminus \{x\}, Y \setminus \{y\})$-matrix which is (r', s')-adapted, where r' and s' are the restrictions of r and s to the sets $X \setminus \{x\}$ and $Y \setminus \{y\}$ respectively. Now, we repeat the whole process for the new $(X \setminus \{x\}, Y \setminus \{y\})$-matrix until there is no non-zero entry left. Each of the required row and column operations can been interpreted as a row or column operations on the original matrix A, since the $r(x)$ and $c(y)$ will remain unaffected. The composition of all used row operations will be represented by the (r, r)-adapted (X, X)-matrix B, and all used column operations by the (s, s)-adapted (Y, Y)-matrix C. We will end up with a matrix BAC, such that each row and column have exactly one non-zero entry equal to 1, and thus the statement follows.

In summary, we are able to identify finitely presented vector spaces as sums of interval persistence vector spaces. The following statement shows, that the persistence intervals in the sum are even unique to some extent.

Proposition 3.43 Let $\{V_r\}_{r \in R}$ be a finitely presented persistence vector space over a field k. Suppose we have the two compositions

$$\{V_r\}_{r \in R} = \bigoplus_{i \in I} P(a_i, b_i) \text{ and } \{V_r\}_{r \in R} = \bigoplus_{j \in J} P(c_j, d_j),$$

where I and J are finite sets and $a_i, c_j \in R, b_i, d_j \in R \cup \{+\infty\}$ with $a_i < b_i$ and $c_j < d_j$ for all $i \in I, j \in J$. Then $|I| = |J|$, i.e. I and J have the same cardinality, and for all $i \in I$ there exists an unique $j \in J$ such that $(a_i, b_i) = (c_j, d_j)$.

Proof First let a_{min} and c_{min} be the smallest value of a_i and c_j respectively. Note that we can express those values as $a_{min} = \min\{r \in R \mid V_r \neq 0\}$. In particular $a_{min} = c_{min}$. Next we define $b_{min} = \min\{b_i \mid i \in I \text{ and } a_i = a_{min}\}$, as well as $d_{min} = \min\{d_j \mid j \in J \text{ and } c_j = c_{min}\}$. Note that we characterize them as $b_{min} = \min\{r' \in R \mid \ker(L(r, r')) \neq \{0\}\} = d_{min}$, where $L(r, r')$ is the linear transformation as given in Definition 3.20. This means that $P(a_{min}, b_{min}) = P(c_{min}, d_{min})$ appears in both decompositions. Moreover,

$P(a_{min}, b_{min})$ is isomorphic to some sub-persistent vector space $\{W_r\}_{r \in R}$ of $\{V_r\}_{r \in R}$, which can be characterized by the kernel of the linear transformation

$$L(r, b_{min}) \; j_{im(L(a_{min}, r))} : im(L(a_{min}, r) \; ! \quad V_{b_{min}}.$$

This implies that the number of summands of the form $P(a_{min}, b_{min})$ in both compositions is the same, i.e. for $I^0 = \{i \in I \mid a_i = a_{min} \text{ and } b_i = b_{min}\}$ and $J^0 = \{j \in J \mid c_j = c_{min} \text{ and } d_j = d_{min}\}$ we have that $|I^0| = |J^0|$. By forming the quotients, we get the decompositions

$$\{V_r\}_{r \in R} / \{W_r\}_{r \in R} = \bigoplus_{i \in I \setminus I^0} P(a_i, b_i) \text{ and } \{V_r\}_{r \in R} / \{W_r\}_{r \in R} = \bigoplus_{j \in J \setminus J^0} P(c_j, d_j).$$

Repeating the whole procedure on the newly obtained quotient space, since we have finitely many summands, will give us the desired result.

Let us come back to the original problem: Let $\{X_r, r\}_{r \in R}$ be a filtered simplicial complex and $r : X \; ! \; R$ its filtration function. Recall that the i-th homology group of X is defined as $Z_i(X) / B_i(X)$, where $Z_i = \ker(\partial_i)$ and $B_i = im(\partial_{i+1})$, in particular $H_i(X)$ is finitely presented.

Definition 3.44 The persistent homology group $\{H_i(X_r, r)\}_{r \in R}$ of a filtered simplical complex $\{X_r, r\}_{r \in R}$ is given by the finitely presented persistent vector space

$$\{H_i(X_r, r)\}_{r \in R} = \{Z_i(X_r, r)\} / \{B_i(X_r, r)\}_{r \in R}.$$

Thanks to all our preparatory work, we are able to apply all our results for persistent vector spaces to the special case of persistent homology. Assume $\{H_i(X_r, r)\}_{r \in R}$ is the persistent homology group of a filtered complex $\{X_r, r\}_{r \in R}$ which is defined on a dataset. Applying the structure theorem, we are able to identify it with a sum of interval persistent vector spaces

$$\{H_i(X_r, r)\}_{r \in R} = \bigoplus_{i=1}^{n} P(a_i, b_i).$$

The interpretation of that is that each summand $P(a_i, b_i)$ corresponds to an i-dimensional hole in the filtered complex $\{X_r, r\}_{r \in R}$, that appears at the time a_i and disappears at b_i. The larger the difference between b_i and a_i, the longer the corresponding hole exists and the more likely it is that it represents a true feature of underlying topological space, i.e. the more relevant it is considered to be.

3.4 Algorithm for Computing Persistent Homology

In this section, we will look at an algorithm that allows us to compute the homology groups of persistent vector spaces where the vector spaces are given as i-chains of a given simplicial complex.

Recall that by the fundamental property of homology, the boundary matrices satisfy $\partial_i \cdot \partial_{i+1} = 0$. In order to establish our algorithm, we first want to study general pairs of matrices satisfying this property.

Definition 3.45 Let $(X, r), (Y, s), (Z, t)$ be $\mathbb{R}$-filtered sets. Furthermore, let A be a (r, s)-adapted (X, Y)-matrix and B be a (s, t)-adapted (Y, Z) matrix, both with entries in the field k, such that

$$A \cdot B = 0.$$

Then we define the *admissible pair operations* on the pair (A, B) to be the following set of operations:

- Arbitrary adapted row operations on A,
- Arbitrary adapted column operations on B,
- Perform an adapted column operation and an adapted row operation on B simultaneously as follows:
  - If the adapted column operation on A is a multiplication of the i-th column by non-zero constant $a \in k$, then the corresponding adapted row operation on B is the multiplication of the i-th row by the constant a^{-1}.
  - If the adapted column operation on A is the transposition of the i-th and the j-th column, the the corresponding adapted row operation on B is the transposition of the i-th and the j-th row.
  - Let $a \in k$. If the adapted column operation on A is the addition of a times the i-th column to the j-th column, the corresponding adapted row operation on B is the subtraction of a times the j-th row from the i-th row.

We observe the following useful property:

Proposition 3.46 Let $(X, r), (Y, s), (Z, t)$ be $\mathbb{R}$-filtered sets and let A be a (r, s)-adapted (X, Y)-matrix and B be a (s, t)-adapted (Y, Z) matrix, both with entries in the field k, such that $A \cdot B = 0$. Then we can perform admissible pair operations on the pair (A, B) to obtain a pair (A', B') such that

$$(A', B') = \left(\begin{bmatrix} I_n & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot A, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I_m \end{bmatrix} \right) \tag{3.2}$$

where I_n and I_m denote identity matrices. Moreover, the pair (A', B') is uniquely determined by the pair (A, B).

Proof As in the proof of the structure theorem for persistent vector spaces (Lemma 3.42), we use adapted row and column operations to obtain

$$(A, B) \quad \begin{bmatrix} I_n & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot A, \begin{bmatrix} B'_{11} & B'_{12} & B'_{13} \\ B'_{21} & B'_{22} & B'_{23} \\ B'_{31} & B'_{32} & B'_{33} \end{bmatrix} =: (A', \tilde{B})$$

Where the matrix $\tilde{B}$ is the result of performing the corresponding operations on the matrix B. By the condition $A \cdot B = 0$, the same must hold for the pair we get by performing admissible pair operations. In particular, this forces $B^0_{11} = B^0_{12} = B^0_{13} = 0$. If B^0_{21} has s rows and B^0_{31} has r rows, then we perform only adapted row operations involving the last $s + r$ rows, since the upper rows are already equal to 0. Each of these operations will have no effect on the matrix A^0, i.e. the corresponding operations will only affect the $r + s$ rightmost columns, which all have only 0 entries. Thus, we can use adapted column and row operations on the matrix $\tilde{B}$, without affecting A^0, such that we achieve

$$(A^0, \tilde{B}) \quad \begin{pmatrix} I_n & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & I_m \end{pmatrix} =: (A^0, B^0)$$

The uniqueness of this representation follows from the fact that $n = \operatorname{rank}(A)$ and $m = \operatorname{rank}(B)$.

Remark 3.47 Since the row and column operations on a matrix correspond to isomorphisms, we have that $\ker(A) = \ker(A^0)$ and $\operatorname{im}(A) = \operatorname{im}(A^0)$.

Remark 3.48 To make the computation shorter and clearer, we order the rows of the matrix A in the pair (A, B) by decreasing value of the function and its columns by increasing value of f.

We will now look at an example to see how we can make use of this algorithm in order to get the homology groups.

Example 3.49 (Persistent Homology Algorithm) We want to determine boundary matrices of a filtered simplicial complex, which will be denoted $(\partial_\#)_\nu$. In order to not lose any information, we have to attach an extra label to each simplex, such that each label tells us when the respective simplex appears in the filtration: Consider the filtered complex from Example 3.29, as seen in Figure 3.10. Each simplex is labeled with i, where T_i marks the point where the simplex appears in the filtration. The boundary matrices representing ∂_1 and ∂_2, as computed in Example 3.33, are

$$(\partial_1)_\nu = \begin{array}{c} \\ (C,1) \\ (D,1) \\ (B,0) \\ (A,0) \end{array} \begin{array}{c} \begin{array}{cccccc} (BC,1) & (AB,1) & (AD,2) & (CD,2) & (AC,3) \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix} \end{array}, (\partial_2)_\nu = \begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD,2) \\ (AC,3) \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{array}.$$

Our goal is to bring the pair of matrices $((\partial_1)_\nu, (\partial_2)_\nu)$ into the form in Equation (3.2), using admissible pair operations. Note that in the following we will only

keep the labels of the rows (i.e. we only track the values of the filtration function) of the right matrix, since the corresponding elements will not be relevant for our computation, and we will not keep track of the elements corresponding to the rows of the right matrix, since they will be the same as the elements corresponding columns of the left matrix, also with the same label. We consider

$$((\pi_1)_*, (\pi_2)_*) = \begin{pmatrix} & \text{(BC,1)} & \text{(AB, 1)} & \text{(AD,2)} & \text{(CD,2)} & \text{(AC,3)} \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}, \begin{matrix} \text{(ABC, 4)} \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{matrix},$$

We denote by $r_{\pi_1}(i)$ and $c_{\pi_1}(j)$ the current i-th row and j-th column of the left matrix. Similarly, $r_{\pi_2}(j)$ and $c_{\pi_2}(k)$ denote the current j-th row and k-th column of the right matrix. We proceed as follows: First we have to make sure to have the 1 at the position $(1,1)$ of the left matrix. If not, we perform row and column switching operations to get the desired entry there. Since we have already the 1 at the desired position, we do not have to switch any columns or rows. We proceed by eliminating all non-zero entries in the column $c_{\pi_1}(1)$ by performing adapted row operations (note that here the matrix on the right remains unchanged while performing adapted row operations).

$$r_{\pi_1}(3) \mapsto r_{\pi_1}(3) + r_{\pi_1}(1)$$

$$\left.\begin{pmatrix} & \text{(BC,1)} & \text{(AB, 1)} & \text{(AD,2)} & \text{(CD,2)} & \text{(AC,3)} \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}, \begin{matrix} \text{(ABC, 4)} \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{matrix}\right),$$

We continue like this until the left matrix is a upper triangular matrix with 1s in the upper left part of the diagonal entries

$$r_{\pi_1}(2) \leftrightarrow r_{\pi_1}(3)$$

$$\left.\begin{pmatrix} & \text{(BC,1)} & \text{(AB, 1)} & \text{(AD,2)} & \text{(CD,2)} & \text{(AC,3)} \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}, \begin{matrix} \text{(ABC, 4)} \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{matrix}\right),$$

$$r_{\P_1}(4) \mapsto r_{\P_1}(4) + r_{\P_1}(2)$$

$$\begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD,2) \\ (AC,3) \end{array} \begin{array}{c} \begin{array}{ccccc} (BC,1) & (AB,1) & (AD,2) & (CD,2) & (AC,3) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \end{array} , \begin{array}{c} \\ (ABC,4) \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

$$r_{\P_1}(4) \mapsto r_{\P_1}(4) + r_{\P_1}(3)$$

$$\begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD,2) \\ (AC,3) \end{array} \begin{array}{c} \begin{array}{ccccc} (BC,1) & (AB,1) & (AD,2) & (CD,2) & (AC,3) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} , \begin{array}{c} \\ (ABC,4) \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

Now, we want to get rid of all the 1's which do not lie on the diagonal of the matrix. For this, we need to perform column operations on the right matrix, and therefore the corresponding row operations on the right matrix.

$$c_{\P_1}(4) \mapsto c_{\P_1}(4) + c_{\P_1}(1)$$
$$c_{\P_1}(5) \mapsto c_{\P_1}(5) + c_{\P_1}(1)$$
$$r_{\P_2}(1) \mapsto r_{\P_2}(1) \quad r_{\P_2}(4) \quad r_{\P_2}(5)$$

$$\begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD+BC,2) \\ (AC+BC,3) \end{array} \begin{array}{c} \begin{array}{ccccc} (BC,1) & (AB,1) & (AD,2) & (CD+BC,2) & (AC+BC,3) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} , \begin{array}{c} \\ (ABC,4) \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

$$c_{\P_1}(4) \mapsto c_{\P_1}(4) + c_{\P_1}(2)$$
$$c_{\P_1}(5) \mapsto c_{\P_1}(5) + c_{\P_1}(2)$$
$$r_{\P_2}(2) \mapsto r_{\P_2}(2) \quad r_{\P_2}(4) \quad r_{\P_2}(5)$$

$$\begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD+BC+AB,2) \\ (AC+BC+AB,3) \end{array} \begin{array}{c} \begin{array}{ccccc} (BC,1) & (AB,1) & (AD,2) & (CD+BC+AB,2) & (AC+BC+AB,3) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} , \begin{array}{c} \\ (ABC,4) \\ \\ \\ \\ \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

$$c_{\P_1}(4) \mapsto c_{\P_1}(4) + c_{\P_1}(3)$$
$$r_{\P_2}(3) \mapsto r_{\P_2}(3) \quad r_{\P_2}(4)$$

$$\begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD+BC+AB+AD,2) \\ (AC+BC+AB,3) \end{array} \begin{array}{c} \begin{array}{cccccc} (BC,1) & (AB,1) & (AD,2) & (CD+BC+AB+AD,2) & (AC+BC+AB,3) & (ABC,4) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array} \quad , \quad \begin{array}{c} \\ (BC,1) \\ (AB,1) \\ (AD,2) \\ (CD+BC+AB+AD,2) \\ (AC+BC+AB,3) \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \end{array}$$

So, we have arrived at the desired form. Now recall that $\ker(\P_1)/\operatorname{im}(\P_2)$ determines the 1st homology group. Similarly, we are able to read off the decomposition of the persistent homology group by looking at the restriction of $\P_2$ to $\ker((\P_1)_{¥})$:

$$\begin{array}{c} \\ (CD+BC+AB+AD,2) \\ (AC+BC+AB,3) \end{array} \begin{array}{c} \begin{array}{c} (ABC,4) \end{array} \\ \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{array}$$

We can read from this that the 1-cycle $CD + BC + AB + AD$ appears at time 2 and never becomes a boundary of a 2-simplex, i.e. it never disappears. The corresponding interval in the decomposition is therefore given by $[2,¥)$. On the other hand, the 1-cycle $AC + BC + AB$ appears at time 3 and then, at time 4, it becomes a boundary of the 2-simplex ABC, thus the corresponding interval is given by $[3,4)$. In total, the persistent homology groups $\{H_i(X_{r,r})\}_{r2R}$ can be represented by

$$\{H_i(X_{r,r})\}_{r2R} = [2,¥) \quad [3,4).$$

The cycles corresponding to the interval can be seen in Figure 3.11. The red cycle, which appears at time T_2, gets a bit smaller at T_3 and never disappears, corresponds to the interval $[2,¥)$, whereas the blue cycle that appears at T_3 and gets filled in at T_4 corresponds to the interval $[3,4)$.

Figure 3.11: Filtered complex studied in Example 3.49.

3.5 Barcodes and Persistent Diagrams

We have seen in the previous section that there is a one-to-one correspondence between the isomorphism classes of finitely presented persistent vector spaces and finite subsets of the form (a,b), where $a \in \mathbb{R}, b \in \mathbb{R} \cup \{+\infty\}$ and $a < b$. This allows us to give some visual representations for a finitely presented persistent vector space. The first one does this via families of intervals in $\mathbb{R}$:

Definition 3.50 Let $\{V_r\}_{r\in\mathbb{R}} = \bigoplus_{i=1}^{n} P(a_i, b_i)$ be a finitely presented persistent vector space. Then the barcode of $\{V_r\}_{r\in\mathbb{R}}$ is given as the (disjoint) family of intervals $[a_i, b_i) \subset \mathbb{R}$, $i = 1, \ldots, n$.

Example 3.51 Assume we have a persistent vector space $\{V_r\}_{r\in\mathbb{R}}$ which can be identified with the sum

$$P(0,1) \oplus P(0,2) \oplus P(1,4) \oplus P(3,6) \oplus P(5,7). \tag{3.3}$$

Then the barcode of $\{V_r\}_{r\in\mathbb{R}}$ is given by the intervals $\{[0,1), [0,2), [1,4), [3,6), [5,7)\}$, as given in Figure 3.12a.

The next representation will represent $\{V_r\}_{r\in\mathbb{R}}$ as points in the plane $\mathbb{R}^2$.

Definition 3.52 Let $\{V_r\}_{r\in\mathbb{R}} = \bigoplus_{i=1}^{n} P(a_i, b_i)$ be a finitely presented persistent vector space. Then the persistent diagram of $\{V_r\}_{r\in\mathbb{R}}$ is given by the collection of points $(a_i, b_i) \in \mathbb{R}^2$, $i = 1, \ldots, n$.

Example 3.53 Let $\{V_r\}_{r\in\mathbb{R}}$ be given as in Example 3.51, i.e. it can be identified with the sum in Equation 3.3. Then the persistent diagram of $\{V_r\}_{r\in\mathbb{R}}$ consists of the points $(0,1), (0,2), (1,4), (3,6), (5,7)$, as seen in Figure 3.12b.

(a) Barcode (b) Persistent Diagram

Figure 3.12: Barcode and persistent diagram of the persistent vector space (3.3).

Remark 3.54 In the context of persistent homology, long intervals in the barcode correspond to longer existing holes. In the persistent diagram, the further a point lies away from the line $\{(x,x) \mid x \in \mathbb{R}\}$, the longer the corresponding hole exists.

3.6 The Bottleneck Distance and the Matching Distance

We are able to associate to a dataset a barcode (or a persistence diagram). One question that naturally comes up is how these barcodes (or persistence diagrams) will change when we have "small" changes in our data. To answer this, we first have to clarify how to "measure" differences between barcodes (or persistence diagrams), by defining a metric on them.

Definition 3.55 Let $I = [a_1, b_1] \subset \mathbb{R}$ and $J = [a_2, b_2] \subset \mathbb{R}$ be two intervals. Then we define the *distance* between the intervals I and J to be

$$D(I, J) := \max\{|a_2 - a_1|, |b_2 - b_1|\}.$$

Furthermore, we define the *lambda-Value* of an interval I to be

$$l(I) := \frac{b_1 - a_1}{2}.$$

Remark 3.56 $D(I, J)$ is the l^∞-distance between the intervals, while $l(I)$ is the l^∞-distance between I and the closest point of the form $(x, x) \in \mathbb{R}^2$ to I.

With these two notions, we are ready to give a notion of distance between barcodes, i.e. families of intervals.

Definition 3.57 Let $\{I_a\}_{a \in A}$ and $\{J_b\}_{b \in B}$ be families of interval in $\mathbb{R}$, where A, B are finite sets. Let $F : A' \to B'$ be a bijection from a subset $A' \subset A$ to a subset $B' \subset B$. We define the *penalty* of the bijection F, denoted by $P(F)$, to be

$$P(F) := \max\{\max_{a \in A'} D(I_a, J_{F(a)}), \max_{a \in A \setminus A'} l(I_a), \max_{b \in B \setminus B'} l(J_b)\},$$

where we set

$$\max_{a \in \emptyset} D(I_a, J_{F(a)}) = \max_{a \in \emptyset} l(I_a) = \max_{b \in \emptyset} l(J_b) = 0.$$

Then we define the *bottleneck distance* between $\{I_a\}_{a \in A}$ and $\{J_b\}_{b \in B}$, denoted by $d_\infty(\{I_a\}_{a \in A}, \{J_b\}_{b \in B})$, to be given by

$$d_\infty(\{I_a\}_{a \in A}, \{J_b\}_{b \in B}) := \min_F P(F),$$

where the minimum runs over all possible bijections F between subsets of A and B.

Example 3.58 Consinder the two families $I = \{[0, 1], [2, 5]\}$ and $J = \{[4, 6]\}$. Our only options for bijections between subsets of them are given by

$$F_1 : \{[0, 1]\} \to \{[4, 6]\}, [0, 1] \mapsto [4, 6],$$
$$F_2 : \{[2, 5]\} \to \{[4, 6]\}, [2, 5] \mapsto [4, 6].$$

For the penalty of F_1 we get

$$\begin{aligned} P(F_1) &= \max\{D([0,1],[4,6]), l([2,5])\} \\ &= \max\{\max\{|4-0|,|6-1|\}, \frac{5-2}{2}\} \\ &= \max\{5, \frac{3}{2}\} \\ &= 5. \end{aligned}$$

Similarly,

$$\begin{aligned} P(F_2) &= \max\{D([2,5],[4,6]), l([0,1])\} \\ &= \max\{\max\{|4-2|,|6-5|\}, \frac{1-0}{2}\} \\ &= \max\{2, \frac{1}{2}\} \\ &= 2. \end{aligned}$$

Hence, the bottleneck distance is

$$d_\infty(I,J) = \min\{P(F_1), P(F_2)\} = 2.$$

One can think of the bottleneck distance as the penalty of "optimal" subsets $A' \subseteq A, B' \subseteq B$ and bijections between these, such that the values $D(a, F(a))$, where $a \in A'$, as well as the Lambda-Values for the elements that not contained in the subsets A', B' are as small as possible. In the context of persistence diagrams, a "good" bijection F would match points such that points are matched together if they are close to each other, and all the unmatched points (i.e. the points in $A \setminus A', B \setminus B'$) are as close as possible to the set $\{(x,x) \mid x \in \mathbb{R}\}$.

Moreover, d_∞ is a special case of the $p = \infty$ version of the family of the so called Wasserstein metric:

De nition 3.59 The Wasserstein metric d_p is de ned by

$$P_p(F) := \sum_{a \in A'} D(I_a, J_{F(a)})^p + \sum_{a \in A \setminus A'} l(I_a)^d + \sum_{b \in B \setminus B'} l(J_b)^p,$$

where we set $d_p(\{I_a\}_{a \in A}, \{J_b\}_{b \in B}) := (\min_F P_p(F))^{\frac{1}{p}}$.

Another metric, of which we will make use of in chapter 4, is the so called matching distance:

De nition 3.60 Let $B_1 = \{I_a\}_{a \in A}$ and $B_2 = \{J_b\}_{b \in B}$, where A, B are nite index sets, be two barcodes. For two intervals $I, J \subseteq \mathbb{R}$ we de ne the symmetric

difference $d(I,J) = m((I \cup J) \setminus (I \cap J))$, where m is the Lebesgue-measure on $\mathbb{R} \cup \{\infty\}$. A matching M in B_1, B_2 is a set of intervals

$$M(B_1,B_2) \quad B_1 \quad B_2 = \{(I_a, J_b) \mid a \in A, b \in B\}$$

such that if $(I_a, J_b) \in M(B_1,B_2)$ then $(I_a, J_{b'}) \notin M(B_1,B_2)\ \forall b' \in B \setminus \{b\}$ and $(I_{a'}, J_b) \notin M(B_1,B_2)\ \forall a' \in A \setminus \{a\}$, i.e. each interval occurs in at most one pair (I_a, J_b). Define N_M to be the set of unmatched intervals, i.e.

$$N_M := \{I_a \mid a \in A \text{ and } \nexists b \in B : (I_a, J_b) \in M(B_1,B_2)\} \cup \{J_b \mid b \in B \text{ and } \nexists a \in A : (I_a, J_b) \in M(B_1,B_2)\}.$$

Then the *matching distance* is defined to be

$$D(B_1,B_2) := \min_M \Big(\sum_{(I,J) \in M(B_1,B_2)} d(I,J) + \sum_{L \in N_M} m(L) \Big),$$

where the minimum runs over all possible matchings on B_1, B_2.

Example 3.61 Let us look again at the families $I = \{[0,1],[2,5]\}$ and $J = \{[4,6]\}$. The only possible matchings between I and J are

$$M_1(I,J) = \{([0,1],[4,6])\}, M_2(I,J) = \{([2,5],[4,6])\}.$$

The corresponding sets of unmatched intervals are

$$N_1 = \{[2,5]\}, N_2 = \{[0,1]\}.$$

We compute

$$\begin{aligned}
D_1(I,J) &= d([0,1],[4,6]) + m([2,5]) \\
&= m(([0,1] \cup [4,6]) \setminus ([0,1] \cap [4,6])) + m([2,5]) \\
&= m([0,1] \cup [4,6]) + m([2,5]) \\
&= (1+2) + 3 \\
&= 6, \\
D_2(I,J) &= d([2,5],[4,6]) + m([0,1]) \\
&= m(([2,5] \cup [4,6]) \setminus ([2,5] \cap [4,6])) + m([0,1]) \\
&= m([2,6] \setminus [4,5]) + m([0,1]) \\
&= (4-1) + 1 \\
&= 4.
\end{aligned}$$

Hence the matching distance is given by

$$D(I,J) = \min\{D_1(I,J), D_2(I,J)\} = D_2(I,J) = 4.$$

Remark 3.62 A matching can be viewed as a bijection between subsets $B_1' \subseteq B_1$ and $B_2' \subseteq B_2$, where each $b \in B_1'$ is assigned to a $a \in B_2'$.

Chapter 4

Application: Classification of Liver Lesions

After all the prior work, we now look at a concrete application of persistent homology. In this chapter, we show how persistent homology can be used in order to classify images of liver lesions and possibly detect cancerous lesions. The approach and results appear in Classification of hepatic lesions using matching metrics by Adcock, Rubin and Carlsson ([1]).

First, we briefly introduce liver lesions and present types of lesions our dataset contains. For this introduction we refer the reader to [4],[2] [8], [7], [13] and [14]. After that, we describe the methods we use to analyze and classify the dataset with the help of persistent homology. For the computation, we need a machine learning tool called support vector machine ([12] and [11]).

4.1 Liver Lesions

Hepatic lesions, or liver lesions, are abnormal growths of liver cells. Most of them can be categorized into benign lesions, which typically are no reason for concern, or into liver cancer, which are less common but more serious. Hence, classifying these liver lesions is of great interest.

The dataset we use consists of computed tomography (CT) scans of 132 hepatic (liver) lesions, together with diagnosis and semantic descriptors of each lesion. The lesions in the set can be categorized into the following types:

- Cysts (45 lesions) are fluid-filled sacs, which might be already present at birth, but can also develop later in life. Cysts are examples for benign lesions, i.e. they are noncancerous.
- Metastases (45 lesions) are cancerous liver lesions which occur when tumors from other parts of the body spread to the liver.

- Hemangiomas (18 lesions) consist of abnormal blood vessels. They are the most common type of benign lesions and can be found in up to 5% of adults.
- Hepatocellular carcinomas (HCC, 11 lesions) are the most common cancerous liver lesions. They develop in people with liver damage caused by viral hepatitis or alcoholism.
- Focal codules (5 lesions) often occur in women and have a “scar-like appearance”.
- Liver abscesses (3 lesions) are pus- lled pockets of uid within the liver. There are many causes of abscesses, such as infections or other damages to the liver. Although they are noncancerous, they can be life-threatening, where the risk for death is higher the more liver abscesses a person has.
- Neuroendocrine neoplasms (NeN, 3 lesions) are rare types of cancerous liver lesions.
- A single liver laceration, i.e. a liver injury caused by some trauma to the liver.
- One single fat deposit

An example of each can be seen in Figure 4.1.

Figure 4.1: Examples for lesion types in the dataset (Source [1]).

In Ct image-based decision support system for categorization of liver metastases into primary cancer sites: Initial results by Ben-Cohen et al. ([4]) it has been proven useful to classify hepatic lesions by looking for visually identi able structures within the lesions, though it turned out to be quite challenging to nd quantitative measures of the structure. To illustrate this dif culty, let us consider Figure 4.2. It shows three hemangiomas of the given dataset. The structure of hemangiomas is typically given by a large dark center together with dense white regions on the outer edge. But as we can see, for the there hemangiomas in Figure 4.2 there is no rational orientation for the hemangiomas. They have different numbers of the two regions and the formations also differ in size and shape.

Figure 4.2: Images of hemangiomas in the dataset (Source [1]).

4.2 Classi cation of the Dataset via Persistent Homology

Our goal is to classify the given lesion images with the help of persistent homology. First, we need a ltered complex de ned on the dataset. We want to make use of the intensity ltration we introduced in Section 3.1.2. In order to get better classi cation results than by using the intensity ltration alone we combined it with the so-called border ltration : For a given image I one de nes a ltration function $b: P \to R_0$, that associates to each pixel its distance to the lesion border, such that increasing the border will produce an "annulus" which eventually will ll out the lesion, or if one uses a decreasing ltration, it delivers a "misshapen disc" which expands from the center of the lesion.

Let $K_{g,i}$ represent complexes from the intensity ltration, and $K_{b,j}$ complexes from the border ltration. At each ltration slice j of the border ltration, we use the intensity ltration to determine the persistent homology of the subcomplex $K_{b,j}$, so that we end up with the ltered complex

$$K_{(b,g),(j,g_{min})} = K_{b,j} \setminus K_{g,g_{min}} \qquad K_{b,j} \setminus K_{g,g_{max}} K_{(b,g),(j,g_{max})}.$$

Thus, for each j we get a barcode, each of which can be considered as a different measurement of the lesion. We proceed as follows: In order to take into account that different image formatting or different CT scanners lead to differences into the pixel scaling, we normalize the pixel range of each image to the range from 0 to 1. We divide the range of the border ltration into 20 equally spaced slices. Together with the option of considering increasing and decreasing ltrations for both the intensity and the border ltration, which results in the barcodes b_0 and b_1 producing eight barcodes at each slice, we ended up with 160 barcodes computed on each of the lesions. We stop in nite barcodes at the value 1.1 so that two lesions were not immediately separated by a different number of in nite bars.

In order to use machine learning tools to classify the images, we need to create a vector of measurement for each lesion, also known as feature vectorUsing the matching distance from Section 3.6 we create a vector of measurement by computing the matching distance between each lesion and all other lesions (including itself). In other words, we use all 132 images in the dataset as a comparison set to generate the feature vectors. We emphasize that we do this even if only a smaller subset of lesions might be of interest, because that way we can obtain information from lesion types which might be too small for classi cation. Since the combination of intensity and border ltration yields 160 barcodes for each lesion. We then sum up the 160 distances to create a vector of size 132. This feature vector can then by used in machine learning algorithms. As in [1], we use an implementation of the support vector machine called LibSVM in order to test the classi cation results.

Figure 4.3: Example of SVM algorithm (Source [12]).

Support vector machines, or SVM for short, are machine learning algorithms which are primarily used in classi cation problems. These algorithms seek for the best line or decision boundary that separates the underlying space into classes, so that each new data point can easily be assigned to one of

the classes in the future. The best decision boundary is called hyperplane. For this SVM chooses “extreme vectors” that help create these hyperplanes. These “extreme vectors” are called support vectors. In Figure 4.3 one can see an example where support vectors are used to de ne a hyperplane that classi es two different categories. For further information on support vector machines, we refer to [12].

4.3 Classi cation Results

For intuition, by using the feature vectors for classical multidimensional scaling (CMDS) on the distance matrix, we produce 2D and 3D visualizations of the lesions, which can be seen in Figure 4.4.

Figure 4.4: Visualization of the topological features of the lesions (Source [1]).

Since the number of representatives of a lesion class is quite unbalanced in the dataset, for example there are more than 4 times as many images of metastases than of hepatocellular carcinomas, we present the results for four different subsets of the data. The rst one is the full set. The second subset is the set of HCCs, hemangiomas, cysts and metastases. This subset is denoted by HcHeCM. The third subset, labeled by HeCM, consists of hemangiomas,

cysts an metastases. Lastly, in the fourth subset, denoted by CM, only cysts and metastases are considered. The results of the calculations are given in Table 1

Table 1 SVM Classi cation accuracies of 1D and 2D ltrations				
Filtration	Full (%)	HcHeCM (%)	HeCM (%)	CM (%)
1D (intensity)	55.30	59.66	63.89	80.00
2D	66.67	72.27	80.56	85.56

In a next step, we make use of the Gaussian kernel $e^{\frac{sjju\ vjj^2}{2}}$, where u and v are feature vectors in combination with the SVM. Support Vector Machines use kernel methods to transform the input data into a higher-dimensional features space. In such a space, it is simpler to distinguish between the different classes. For more details on this we refer to [11]. The classi cation rates of each lesion type this method achieved are given in Table 2

Table 2 HeCM % classi cation accuracy by lesion type				
Filtration	% of HeCM	% of Heman.	% of Cysts	% of Metas.
1D	63.89	27.78	77.78	64.44
2D	80.56	72.22	88.89	75.56

Adcock et al. noticed while examining the misclassi ed lesions in [1] that many of those lesions are signi cantly larger than the median lesion of the dataset. Hence, we perform once again the analysis from before on the HeCM subset, but this time we remove lesions with various pixel areas. The results can been seen inTable 3

Table 3 Classi cation by lesion size of HeCM				
Lesion size by area	% Accu.	# of Heman.	# of Cysts	# of Metas.
All	80.56	18	45	45
< 10,000 px	83.50	18	42	43
< 5000 px	86.96	16	39	37
< 2500 px	86.25	14	32	34
< 1250 px	91.53	8	28	23

The interpretation of this is that larger lesions might have more potential topological generators than smaller ones. Thus, because of unmatched bars, the matching metric will tell us that a larger lesion is a great distance away from a smaller one.

Nevertheless, the results demonstrate how powerful persistent homology becomes if combined with geometry (i.e. the border ltration). It signi cantly improves the accuracy of classi cation via barcodes. According to Adcock et al. [1], this technique could be a powerful alternative to the classic machine learning approaches.

Bibliography

[1] Aaron Adcock, Daniel Rubin, and Gunnar Carlsson. Classification of hepatic lesions using the matching metric. *Computer Vision and Image Understanding*, 121:36–42, 2014.

[2] Tolu Ajiboye. Liver lesions causes and treatment, 2023. https://www.verywellhealth.com/liver-lesions-5077003 [Accessed: (06.01.2024)].

[3] Ulrich Bauer, Michael Kerber, Fabian Roll, and Alexander Rolle. A unified view on the functorial nerve theorem and its variations. *Expositiones Mathematicae*, 41(4):125503, 2023.

[4] Avi Ben-Cohen, Eyal Klang, Idit Diamant, Noa Rozendorn, Stephen P. Raskin, Eli Konen, Michal Marianne Amitai, and Hayit Greenspan. Ct image-based decision support system for categorization of liver metastases into primary cancer sites: Initial results. *Academic Radiology*, 24(12):1501–1509, 2017.

[5] Gunnar Carlsson. Topological pattern recognition for point cloud data. *Acta Numerica*, 23:289–368, 2014.

[6] Gunnar Carlsson and Mikael Vejdemo-Johansson. *Topological Data Analysis with Applications*. Cambridge University Press, 2021.

[7] Cleveland Clinic. Liver cancer. https://my.clevelandclinic.org/health/diseases/9418-liver-cancer [Accessed: (07.02.2024)].

[8] Cleveland Clinic. Liver lesion. https://my.clevelandclinic.org/health/diseases/14628-liver-lesions [Accessed: (07.02.2024)].

[9] Tamal Krishna Dey and Yusu Wang. *Computational Topology for Data Analysis*. Cambridge University Press, 2022.

[10] Allen Hatcher. *Algebraic Topology*. Cambridge University Press, 2001.

[11] Javatpoint. Major kernel functions in support vector machine. https://www.javatpoint.com/major-kernel-functions-in-support-vector-machine [Accessed: (07.01.2024)].

[12] Javatpoint. Support vector machine algorithm. https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm [Accessed: (07.01.2024)].

[13] Medline Plus. Pyogenic liver abscesses. https://medlineplus.gov/ency/article/000261.htm [Accessed: (07.02.2024)].

[14] J.A. Mannick R.G. Holzheimer. *Surgical Treatment: Evidence-Based and Problem-Oriented.* Zuckschwerdt, 2001.

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every written paper or thesis authored during the course of studies. In consultation with the supervisor, one of the following three options must be selected:

(●) I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used no generative artificial intelligence technologies[1].

( ) I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used and cited generative artificial intelligence technologies[2].

( ) I confirm that I authored the work in question independently and in my own words, i.e. that no one helped me to author it. Suggestions from the supervisor regarding language and content are excepted. I used generative artificial intelligence technologies[3]. In consultation with the supervisor, I did not cite them.

Title of paper or thesis:

Authored by:
If the work was compiled in a group, the names of all authors are required.

Last name(s):	**First name(s):**

With my signature I confirm the following:
- I have adhered to the rules set out in the Citation Guide.
- I have documented all methods, data and processes truthfully and fully.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for originality.

Place, date	**Signature(s)**
	Th. Heinz

If the work was compiled in a group, the names of all authors are required. Through their signatures they vouch jointly for the entire content of the written work.

[1] E.g. ChatGPT, DALL E 2, Google Bard
[2] E.g. ChatGPT, DALL E 2, Google Bard
[3] E.g. ChatGPT, DALL E 2, Google Bard