Arnau Mas

Contents

Preface

Lorem ipsum

Chapter 1

The problem

1.1 The problem

The data that has been used for the analysis comes from a study in the field of radiomics. The aim of radiomics is to establish a relationship between a patient's response to treatment and what are known as *radiomic features* of a lesion. These are parameters which give a systematic description of a feature gathered from scans. The hypothesis is that there is information that can be extracted from this more systematic study of medical scans, as opposed to the more traditional qualitative observation by a medical professional.

On a practical level, the radiomic features are extracted from a 3d reconstruction of the lesion, built from a series of 2d sectional scans. It is then the work of a medical professional to manually delimit the contours of the actual lesion, which is known as the *region of interest*. This process is called *segmentation*. Once this is done, the actual work of computing the radiomic features takes place. For the data at hand, these had been extracted with the PyRadiomics Python package [Gri+17]. These features include shape parameters of the triangular mesh determined by the segmented region of interest, as well as statistical descriptors of the distribution of pixel intensity in the region of interest.

Chapter 2

The theory of homology

In this chapter we give a brief explanation of the fundamental concepts involved in the theory of homology, with special emphasis on those wich are relevant to persistent homology. A standard treatment of homology can be found in chapter 2 of [Hat01]. Introductions to persistent homology in particular can be found at [EH08; EM12; EM16].

2.1 Introduction

The standard elevator pitch for algebraic topology is as follows: classifying topological objects is hard, whereas algebraic objects (groups, rings, etc) are better understood, so algebraic topology provides tools to compute algebraic invariants from topological spaces, which aids in their study. The natural setting for these ideas is category theory, so that these "tools" become *functors* from the category of topological spaces and continuous maps, Top, to categories such as Grp, the category of groups and group morphisms, or Ab, the category of abelian groups and their morphisms[1].

One of this tools are homology groups, which are the fundamental object of study in the theory of homology. At the most abstract level, a theory of homology is a family of functors from (some subcategory of) Top to Ab with a series of natural transformations between them, subject to what are known as the *Eilenberg-Steenrod axioms*. For our purposes, however, such a general framework is not necessary, and we will sismply describe one particular theory of homology, which is known as simplicial homology. This theory is built in two steps. The first step has very much to do with the geometry of the space we are considering and can be seen as a functor from the category of *simplicial complexes*, Simp, to the category of *chain complexes of abelian groups*, $\mathsf{Ch}(\mathsf{Ab})$, as explained in section 2.2.

[1] We will make some use of basic concepts from cateogry theory in this chapter, a wonderful book on the topic is [Rie16].

This is described in section 2.2. Then there is a more algebraic step, in essence a functor from Ch(Ab) to Ab, as seen in section 2.3. The hole theory is then the composition of these two steps.

Finally, section 2.5 gives an introduction to the ideas which underpin persistent homology, which are the ones we will apply to the problem outlined in chapter 1.

2.2 The geometric side: simplices and simplicial complexes

As stated before, simplicial homology is restricted to simplicial complexes, which are, roughly speaking, topological spaces assembled out of *simplices*, which are the generalisation of triangles and tetrahedra to higher dimensions. These spaces can be described combinatorially, which makes them very useful for computation with datasets. There are ways to generalise the theory to broader classes of spaces, such as singular homology, but simplicial homology is sufficient for the analysis required.

2.2.1 Simplices

As stated before, the basic building block of the spaces we will be dealing with are simplicies.

Definition 2.1 (Simplex)**.** A simplex of dimension n, or simply an n-simplex, generated by $n+1$ points of $\mathbb{R}^d$ which do not lie in an affine subspace of dimension n[2], is their convex hull, i.e. the smallest convex set which contains them. △

When a set of points generate a simplex they are called *geometrically independent*, see fig. 2.1.

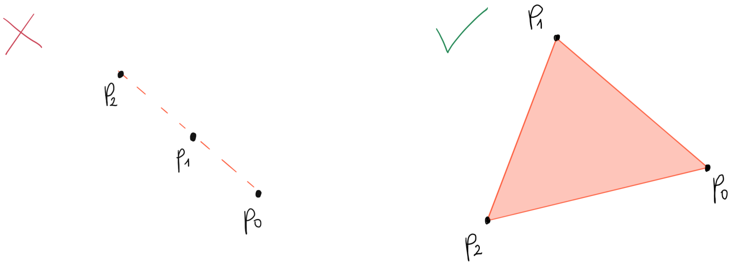

Figure 2.1: The three points on the left are not geometrically independent, whereas the three on the right are and so generate a 2-simplex.

[2] which means the ambient dimension d must be at least n.

The n-simplex generated by the standard basis of $\mathbb{R}^{n+1}$, $e_0, \dots, e_n$[3], is called the *standard n-simplex* and written Δ^n. It can be shown that

$$\Delta^n = \left\{ (t_0, \dots, t_n) \in \mathbb{R}^{n+1} \mid \sum_{k=0}^{n} t_k = 1, \forall k \leq n \colon t_k \geq 0 \right\}.$$

The standard simplices provide a model for any other simplex. Indeed, if $\phi\colon \mathbb{R}^{n+1} \to \mathbb{R}^d$ is the linear map defined by $\phi(e_k) = p_k$ for $0 \leq k \leq n$ then $\phi(\Delta^n)$ is precisely the simplex generated by $p_0, \dots, p_n$. $(t_0, \dots, t_n)$ are called the *barycentric coordinates* of the point $\phi(t_0, \dots, t_n)$.

Orientation

For the purposes of homology, it is also important to keep track of the *orientation* of a simplex. We will use the idea of general simplices being the image of standard simplices to model this situation

Definition 2.2 (Ordered simplex)**.** An *ordered n-simplex* generated by $p_0, \dots, p_n \in \mathbb{R}^d$ is a map $\sigma\colon \Delta^n \to \mathbb{R}^d$ such that σ is the restriction to Δ^n of the linear map $\phi\colon \mathbb{R}^{n+1} \to \mathbb{R}^d$ given by $\phi(e_k) = p_k$, provided $p_0, \dots, p_k$ do indeed generate a simplex. △

Observe that giving an ordering to the vertices of a simplex completely determines an ordered simplex, thus we introduce the notation $[p_0, \dots, p_n]$ for an ordered simplex as defined above.

Consider now the natural action of the symmetric group, $\mathfrak{S}_n$, on $\mathbb{R}^n$ by permuting the elements of the standard basis. Then, a reordering of an ordered n-simplex σ is of the form $\tau \circ \sigma$ for some $\tau \in \mathfrak{S}_{n+1}$. Conversely, since the action of $\mathfrak{S}_n$ is free and transitive, for any two ordered simplices with the same vertices, σ_1 and σ_2 there will always exist a unique permutation $\tau \in \mathfrak{S}_n$ such that

$$\sigma_1 = \tau \circ \sigma_2.$$

If τ is even then σ_1 and σ_2 are said to have the same orientation, whereas if τ is odd then they are said to have opposite orientations. This means in particular that there are only two possible orientations for any simplex, which is consistent with the usual intuition for orientation. See fig. 2.2 for an example.

Given a simplex $\sigma = [p_0, \dots, p_n]$, the $n+1$ possible simplices we get by removing one of the generators —i.e. $[p_0, \dots, \hat{p}_k, \dots, p_n]$— are called the *faces* of σ. Note that they are

[3] Because an n-simplex is generated by $n+1$-points, it will be convenient to number things starting at 0, as opposed to 1 as is more common in mathematics.

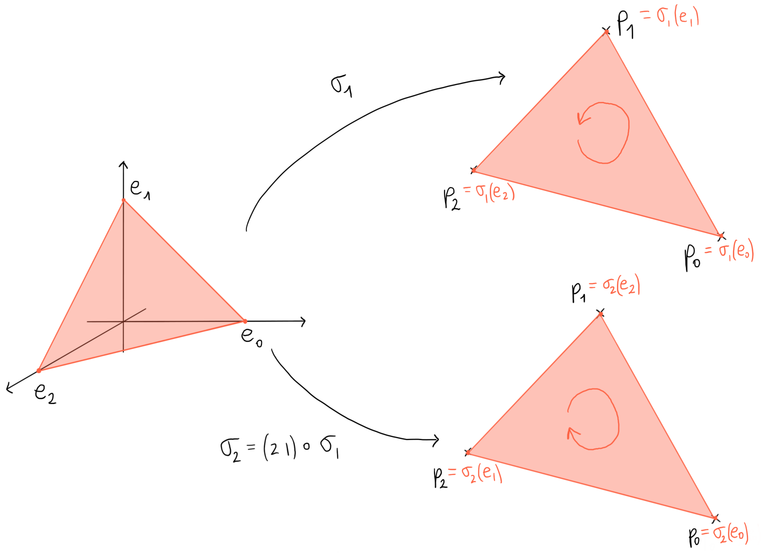

Figure 2.2: Two different orderings of a 2-simplex with the same vertices that determine different orientations.

$(n-1)$-simplices in their own right and they inherit an orientation from the orientation of σ.

2.2.2 Simplicial complexes

Simplicial complexes are spaces assembled out of properly glued together simplices, as the following definition makes precise.

Definition 2.3 (Simplicial complex)**.** A *simplicial complex* K is a collection of simplices of $\mathbb{R}^d$ such that

(i) if $\sigma \in K$ and $\tau \in K$ is a face of σ then $\tau \in K$,

(ii) for any $\sigma, \tau \in K$ then $\sigma \cap \tau \in K$[4].

△

The mental picture is that the simplices are only allowed to be glued along their faces.

The simplices of K are often called its *cells*, and we will write K_n for collection of n-cells of K. And the highest dimension of any of the cells of K is the *top dimension* of K.

A simplicial complex K determines a topological space, called its *underlying space*, which is the subset of $\mathbb{R}^d$ determined by the union of (the images of) all of the cells of K equipped with the subspace topology, which we will write as $|K|$. Pursuing the idea of

[4] This is an abuse of notation since what is really meant is $\sigma(\Delta^k) \cap \tau(\Delta^l)$.

realising more general topological spaces as underlying spaces of some simplicial complex is what leads to singular homology.

2.2.3 Abstract simplicial complexes

Notice that a simplex is always determined by its vertices, and vice versa, except for, of course, whenever the vertices do not generate a simplex, in the sense of definition 2.1. This requirement comes from thinking of simplices and simplicial complexes as geometric objects. But if we don't think of them in this way, and simply allow for any $n+1$ vertices to determine an n-simplex they become strictly combinatorial objects. In this case, we also have to lift the second condition in the definition of a simplicial complex, which was related to how the simplices are assembled together as geometrical objects and therefore becomes meaningless in this new context. This more general complexes are called *abstract simplicial complexes* and are purely combinatorial objects. From this viewpoint, an n-simplex is now an (ordered) set of $n+1$ vertices, and its faces are just its subsets of n vertices. Ans so definition 2.3 is replaced by

Definition 2.4 (Abstract simplicial complex)**.** An *abstract simplicial complex K* is any set closed under the relation of inclusion, i.e., if $\sigma \in K$ then if $\tau \subseteq K$, it must be the case that $\tau \in K$. △

For our purposes, we will assume that all of the simplices that make up a complex are finite, that there is a maximum possible dimension and that complexes are at most countably infinite, thus avoiding problems related to size. This is not much of a restriction since the main application are datasets which are finite.

Notice that to specify a complex it suffices to exhibit its *maximal simplices*, i.e. those simplices which are not the face of any other simplex, or equivalently those which are maximal with respect to inclusion[5].

2.2.4 Special complexes

In the context of topological data analysis, one is often handed a bare point cloud and is faced with the task of generating an appropriate simplicial complex. We now describe various ways of doing this. or the remainder of this section, X will denote a finite subset of $\mathbb{R}^d$, which models the raw point cloud we begin with.

[5] They are guaranteed to exist by requiring that the dimensions of all simplices that make up the complex be bounded.

Čech complex

The ϵ-Čech complex of X, $C_\epsilon(X)$, is determined by the following prescription: $\sigma \subseteq X$ is in $C_\epsilon(X)$ if and only if

$$\bigcap_{p\in\sigma} B_\epsilon(p) \neq \emptyset$$

where $B_\epsilon(p)$ is the ball of radius ϵ centered at p.

This does determine a simplicial complex, since if a certain set of vertices is such that the intersection of all the ϵ-balls centered at them is nonempty, then so will the intersection of the corresponding balls for any subset be nonempty.

Vietoris-Rips complex

The prescription for the ϵ-Vietoris-Rips complex, $V_\epsilon(X)$, is as follows: $\sigma \subseteq X$ defines a simplex in $V_\epsilon(X)$ if and only if for every $p, q \in \sigma$

$$B_\epsilon(p) \cap B_\epsilon(p) \neq \emptyset.$$

In other words, a set of points generate a simplex whenever each of them is at distance at most 2ϵ[6] from the rest. This also shows that the Vietoris-Rips complex is indeed a complex.

Notice that the ϵ-Čech complex is always a subcomplex[7] of the ϵ-Vietoris-Rips complex. Indeed, if $[p_0, \dots, p_n]$ is a simplex in $C_\epsilon(X)$, then for any $0 \leq i, j \leq n$

$$\emptyset \neq \bigcap_{k=0}^{n} \subseteq B_\epsilon(p_i) \cap B_\epsilon(p_j)$$

so that $[p_0, \dots, p_n]$ is also a simplex in $V_\epsilon(X)$. The intuition is that since the condition for a set of points to determine a simplex in the Vietoris-Rips complex is weaker than for the Čech complex, the former will contain more simplices than the latter.

Clique complex

If the points of our cloud are the vertices of some graph, G, we can use this information to build the so-called clique complex, $G(X)$, by declaring that $\sigma \subseteq X$ is a simplex of $G(X)$ if and only if it is a clique, i.e. a fully-connected subgraph of G. This is indeed a complex since the graph generated by any subset of vertices of a clique is itself a clique.

[6] Some texts use a slightly different convention such that the vertices of a simplex in $V_\epsilon(X)$ are at distance at most ϵ (rather than 2ϵ) from each other

[7] A subcomplex of an abstract simplicial complex is a subset which is itself an abstract simplicial complex.

The Vietoris-Rips complex is a special case of this, since it is the clique complex of the graph obtained by declaring to points of X adjacent when they are at distance at most 2ϵ. The sort of complex that will be used later is the clique complex of the MkNN graph of the point cloud.

2.3 The algebraic side: chain complexes and homology groups

So far we have dealt with the special kinds of spaces whose homologies we wish to compute. We now turn to the actual task of computing. The main algebraic idea which requires introduction is that of a chain complex, which, as opposed to the more restricted simplex, plays an important role in the more algebraic aspects of any theory of homology, what is aptly named hmological algebra. Out of the chain complexes we get the actual homology groups, by a quotienting operation we will describe.

2.3.1 Chain complexes

Definition 2.5 (Chain complex)**.** A *chain complex*[8] (of abelian groups) is a family of abelian groups $\{C_n\}_{n\in\mathbb{N}}$ together with a family of morphisms $\partial_n\colon C_n \to C_{n-1}$ such that $\partial_n \circ \partial_{n+1}$. We write (C_*, ∂_*) for the whole complex. △

The elements of C_n are called *n-chains.* The n-chains which lie in $\ker \partial_n$ are the *n-cycles.* And the chains in $\operatorname{im} \partial_{n+1}$ are called the *n-boundaries.* It follows from the definition of a chain complex that

$$\operatorname{im} \partial_{n+1} \subseteq \ker \partial_n$$

i.e. that every boundary is a cycle. This means the following definition makes sense.

Definition 2.6 (Homology groups)**.** The *n-th homology group* of the chain complex (C_*, ∂_*) is the quotient

$$H_n(C) := \ker \partial_n / \operatorname{im} \partial_{n+1}.$$

△

2.3.2 Simplicial chain complexes

The next step is building a chain complex out of the simplicial complexes we have built so far and computing its homology groups. These will then be the homology goups of the

[8] The word complex is being used for two different concepts: simplicial complexes and chain complexes. To avoid confusion we will adopt the convention that complex without qualifier refers to a simplicial complex whereas to refer to a chain complex we will always use both words.

space.

this chain complex is called, perhaps confusingly, a simplicial chain complex. Given a simplicial complex K, the corresponding simplicial chain complex, which we now define, is written $(C_*(K), \partial_*)$. We need each $C_n(K)$ to be an abelian group, so an easy way to achieve this is to define $C_n(K)$ as the free abelian group generated by the n-cells of K. This is not quite it since we want the orientation to play nicely with the group structure. We add the additional condition that for any n-simplex σ and permutation $\tau \in \mathfrak{S}_n$,

$$\tau \circ \sigma = (-1)^\tau \sigma$$

where $(-1)^\tau$ is the sign of τ. That is, if we reorder the vertices of σ by an even permutation, so we don't change the orientation of σ, nothing changes. But if we reorder by an odd permutation, so that the orientation of σ changes, a minus sign is picked up.

In fact, once we have fixed an orientation for each of the cells of K we can treat $C_n(K)$ as a free group, so that an n-chain $c \in C_n(K)$ is of the form

$$\sum_{\sigma \in K_n} a_\sigma \sigma$$

for $a_\sigma \in \mathbb{Z}$.

Of course the other half of a chain complex are the boundary morphisms, which are defined as follows

Definition 2.7 (Boundary morphisms)**.** We define, for $n \geq 1$, the morphisms

$$\partial_n \colon C_n(K) \longrightarrow C_{n-1}(K)$$

by giving their action on a generator as

$$\partial_n [p_0, \dots, p_n] := \sum_{k=0}^{n} (-1)^k [p_0, \dots, \hat{p}_k, \dots, p_n]$$

and extend them on any other chain by

$$\partial_n \left(\sum_{\sigma \in K_n} a_\sigma \sigma \right) := \sum_{\sigma \in K_n} a_\sigma \partial_n \sigma.$$

△

We need to show that all of this is indeed a chain complex, which is a consequence of the following result.

Lemma 2.8. *For any $n > 1$, $\partial_{n-1} \circ \partial_n = 0$.*

Proof. It suffices to show this on any generator, which is a simple calculation:

$$\begin{aligned}
(\partial_{n-1} \circ \partial_n)[p_0, \dots, p_n] &= \partial_{n-1} \left(\sum_{k=0}^{n} (-1)^k [p_0, \dots, \hat{p}_k, \dots, p_n] \right) \\
&= \sum_{k=0}^{n} (-1)^k \partial_{n-1} [p_0, \dots, \hat{p}_k, \dots, p_n] \\
&= \sum_{k=0}^{n} (-1)^k \left(\sum_{l=0}^{k-1} (-1)^l [p_0, \dots, \hat{p}_l, \dots, \hat{p}_k, \dots, p_n] \right. \\
&\quad \left. + \sum_{l=k+1}^{n} (-1)^{l-1} [p_0, \dots, \hat{p}_k, \dots, \hat{p}_l \dots, p_n] \right) \\
&= \sum_{\substack{k=0 \\ l<k}}^{n} (-1)^k (-1)^l [p_0, \dots, \hat{p}_l, \dots, \hat{p}_k, \dots, p_n] \\
&\quad - \sum_{\substack{k=0 \\ l>k}}^{n} (-1)^k (-1)^l [p_0, \dots, \hat{p}_k, \dots, \hat{p}_l, \dots, p_n] \\
&= 0
\end{aligned}$$

because both summands in the last line are the same. □

Thus, if d is the top dimension of K, we have the simplicial chain complex

$$\begin{tikzcd}
C_d(K) \arrow[r, "\partial_d"] & C_{d-1}(K) \arrow[r] & \cdots \arrow[r] & C_1(K) \arrow[r, "\partial_1"] & C_0(K) \arrow[r] & 0
\end{tikzcd}$$

We will drop the subscripts from the boundary morphisms whenever they can be deduced from the context.

Interpretation

The chain groups are related to the concatenation of loops in a general topological space[9] Indeed, the sum of two simplices can be understood as their concatenation. Then, the boundary of a simplex is the sum of its faces. Except not quite, because if we simply used the induced orientation for the faces we would find that $\partial_n \circ \partial_{n+1} \neq 0$. Geometrically, what is happening is that with the induced orientation, the faces don't "go around nicely", so to say, and about half of them need their orientation flipped, which is what the alternating sign accounts for. fig. 2.3 shows this for the boundary of a 2-simplex. The more general interpretation of homology is discussed in section 2.4, however, notice that the cycles and boundaries in a simplicial chain complex are cycles and boundaries in the geometric sense and in fact this is the origin of the terminology.

[9] As explained in [Hat01], the homology groups are in some sense the abelianisation of the fundamental groups.

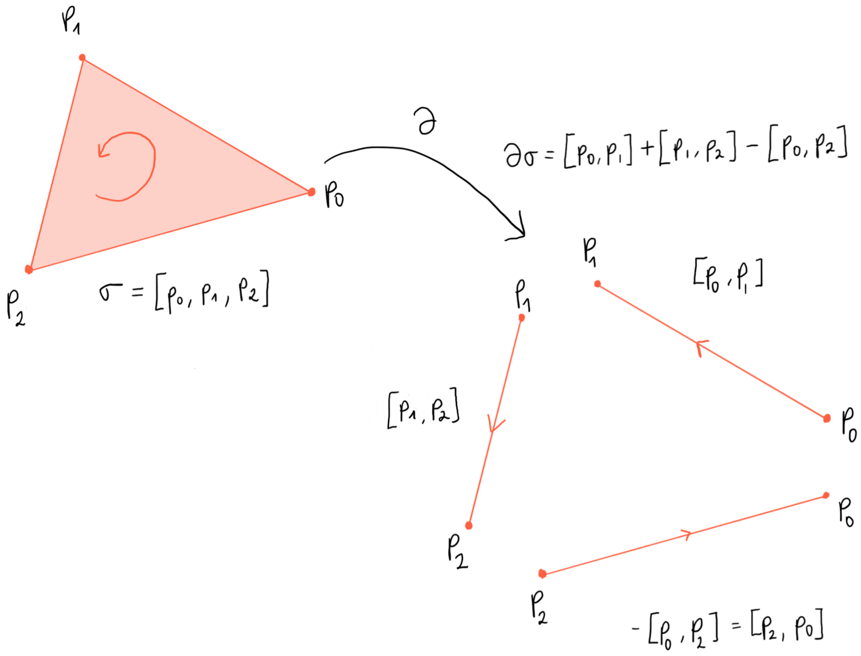
p_1
p_0
p_2
$\sigma = [p_0, p_1, p_2]$
∂
$\partial\sigma = [p_0, p_1] + [p_1, p_2] - [p_0, p_2]$
p_1
$[p_0, p_1]$
p_1
p_0
$[p_1, p_2]$
p_2
p_0
p_2
$-[p_0, p_2] = [p_2, p_0]$

Figure 2.3: This is a representation of the boundary of a 2-simplex. Notice that the face $[p_0, p_2]$ is flipped, which results in the orientations of the boundary going around counterclockwise.

Homology with different coefficients

Instead of considering the free abelian groups generated by the cells, we could think about the F-vector space they generate, for any field F, subject to the compatibility with orientation. Then we would get a chain complex of vector spaces. The homology groups of this complex would be F-vector spaces. In this case, one defines the *Betti numbers* as

$$\beta_n(X, F) = \dim H_n(X, F)$$

where $H_n(X, F)$ are the homology groups of the F-vector space chain complex.

The case of $F = \mathbb{F}_2$ will be of particular interest to us since the algorithm implemented later computes homology with coefficients in $\mathbb{F}_2$. In this case we have

$$\tau \circ \sigma = (-1)^{\tau}\sigma = \sigma$$

so that this homology is orientation blind in some sense.

2.4 The meaning of homology

So we have described how to calculate, at least in principal, these homology groups, and the claim at the beginning of this chapter was that they give us useful information about the space at hand. the obvious question is then: what information exactly?

The first important point is to understand what homology *cannot* tell us. It can be shown that (singular) homology is invariant under homeomorphisms. But in fact it is also invariant under homotopy equivalence, which is a much weaker relationship than homeomorphism: a point is homotopically equivalent to $\mathbb{R}^n$. This means it cannot be used as a tool to classify spaces, at least not in the most general setting.

The easiest homology group to interpret is H_0, since it can be shown that β_0 is the number of connected components of the space. Higher homology groups have to do with n-dimensional holes in the space. Indeed, a cycle which is not the boundary of anything indicates the presence of some sort of "obstacle" in the space. For instance, β_1 is 1 for S^1, which makes sense because S^1 is in some sense the prototypical 1-dimensional hole. For S^2, $\beta_1 = 0$ but $\beta_2 = 1$, reflecting the fact that S^2 is "hollow".

2.5 The idea of persistence

So far we have seen various ways of extracting information related to the shape of our data using homology. All of this ways, however, carry with them an ammount of arbitrary choice. For example, both the Vietoris-Rips and Čech complexes depend on a scale parameter ϵ, and the appropriate choice of ϵ is generally not clear from the data. The idea of persistent homology is to sidestep this problem altogether by considering the homology of the data at every scale to determine which are the features really reflect geometric aspects and which are byproducts of background noise. The first will be, roughly speaking, those which are present at a large range of scales, i.e. those which are *persistent*. We now formalise these ideas.

Definition 2.9 (Filtration)**.** A *filtration* is a family of simplicial complexes $\{K_i\}_{i=0}^n$ such that K_i is a subcomplex of K_{i+1},

$$K_0 \subseteq K_1 \subseteq \cdots \subseteq K_n$$

where we will mostly assume $K_0 = \emptyset$. △

The inclusions $\iota_i^j \colon K_i \hookrightarrow K_j$, because of the functoriality[10] of homology, give rise to maps between the homology groups at different steps of the filtration,

$$H_p(\iota_i^j) \colon H_p(K_i) \longrightarrow H_p(K_j)$$

which we will write as f_i^j for short. We say a certain class is *born at i* if it is not in the image of any f_k^i for any $k \leq i$. And we say it *dies at j* if it is in the kernel of f_i^j but not of f_i^k for $k < j$. The difference $j - i$ is called its *persistence* or *lifetime*.

[10] Again, see [Rie16] for a detailed introduction to category theory.

This is a reflection at the level of homology of how the shape of the complex changes at every step of the filtration. See fig. 2.4 for a detailed example.

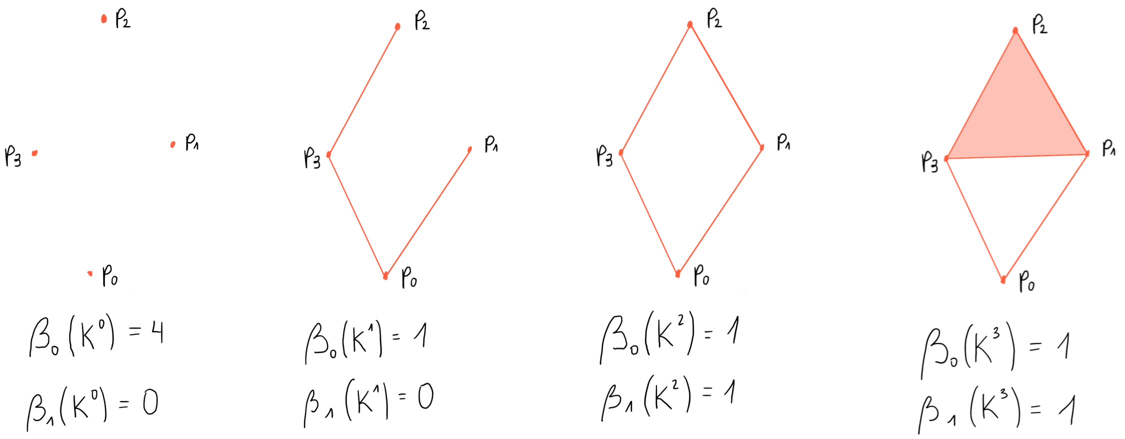

Figure 2.4: Four different steps of a filtration and their first two Betti numbers. At the first step, the complex consist of four isolated points. At the next step, three 1-simplices connect all the points but no combination of them is a cycle, so the dimension of $H_1(K^1)$ is still 0. At the next step, the simplex $[p_1, p_2]$ closes the cycle $c = [p_0, p_1] + [p_1, p_2] + [p_2, p_3] + [p_3, p_0]$ which is not the boundary of anything (there are no 2-simplices), and so the dimension of $H_1(K^2)$ becomes 1. Finally, a 2-cell is born, but this is not sufficient to kill the class of c. Indeed, $f_2^3(c) = c = [p_0, p_1] + [p_1, p_3] + [p_3, p_0] + \partial[p_1, p_2, p_3]$.

Chapter 3

The algorithm and its implementation

3.1 Justification of the algorithm

The main idea of the algorithm used to compute the persistent homology is taken from [Cam18]. Recall that the homology groups are defined as a quotient of the subgroup of cycles by the subgroup of boundaries. But we can just as well consider the quotient of the *whole* chain group. More specifically, consider a filtration $\{K^i\}_{i=0}^N$ and its chain complex, $C_*(K^i)$. We then have the corresponding boundary morphisms,

$$\partial_p^i \colon C_p(K^i) \to C_{p-1}(K^i)$$

where the superscript i refers to the step of the filtration. refers to the step of the filtration. The group of p-cycles is $\ker \partial_p^i$ and the group of p-boundaries is $\operatorname{im} \partial_{p+1}^i$. Since $\partial_p \circ \partial_{p+1} = 0$ the boundaries are a subgroup of the cycles, which means the homology groups are well defined. But the boundaries are a subgroup of $C_p(K^i)$, which means we can define what we will call *prehomology groups* by

$$P_p(K^i) = C_p(K^i) / \operatorname{im} \partial_{p+1}^i$$

which is the result of identifying all homologous chains, not just the cycles.

Now, because $\operatorname{im} \partial_{p+1}^i \subseteq \ker \partial_p^i$, the boundary maps factor through the prehomology groups, which means the following is well-defined

$$\begin{aligned} \bar{\partial}_p^i \colon P_p(K^i) &\longrightarrow P_p(K^j) \\ [c] &\longmapsto [\partial_p^i c]. \end{aligned}$$

In particular,

$$\ker \bar{\partial}_p^i = H_p(K^i).$$

3.2 Mutual k-Nearest Neighbours Graph

As stated before, the original way of determining clusters in the point cloud was by finding the maximal cliques of the Mutual k-Nearest Neighbours graph, MkNN graph for short. We now describe this construction. Given a point cloud, C, its Nearest Neighbour directed graph is constructed according to the following prescription: there is an edge from p to q if and only if $d(p,q) = \min_{r\in C} d(p,r)$, so q is the closest point to p, its *nearest neighbour*. The reason the graph is directed is because the relation of being a nearest neighbour is not symmetric, as evidenced by the figure

This generalises to the k-Nearest Neighbours graph, in which each point is connected to its k-th nearest neighbours, that is, the k-th closest points to it. The resulting graph is still directed. We obtain the undirected *Mutual* k-Nearest Neighbours graph by connecting two points if and only if there are edges between them in both directions, i.e. if they are *mutual* k-nearest neighbours.

The procedure to encode this graph is as follows. First, the distance matrix, D, for the data set is computed. If we index the points of the cloud by $C = \{p_i\}_{i=1}^n$, then this matrix contains the difference between every pair of points:

$$D_{ij} = d(p_i, p_j).$$

This can be done with NumPy and is implemented as a method of the `Cloud` class, which models a point cloud. Then, again using NumPy, the indices of each column of D are sorted into a new matrix such that the i-th column of this new matrix lists the points of C by increasing order of distance to p_i. In particular, the first k are p_i's k-th nearest neighbours. From this we can then construct the adjacency matrix of the kNN graph, A, which is in general not symmetric since the graph is directed. The adjacency matrix of the MkNN graph, M, is given by

$$M = A \odot A^\top$$

where $\odot$ denotes the entrywise prodct. Indeed, we have

$$M_{ij} = A_{ij}A^\top_{ji} = A_{ij}A_{ji}$$

so that M_{ij} is nonzero if and only if both A_{ij} and A_{ji} are nonzero, i.e. if there is an edge going from p_i to p_j and one from p_j to p_i. Furthermore

$$M_{ij} = A_{ij}A_{ji} = A_{ji}A_{ij} = M_{ji}$$

which means that M is symmetric and therefore the adjacency matrix of an undirected graph, as we claimed. This adjacency matrix is used to represent the MkNN graph as a NetworkX graph.

3.3 Building the filtration

The work of constructing the filtration is packaged up in the `Filtration` class. This class is passed an instance of `Cloud`. The most straightforward way to represent a filtration is to simply store an ordered list of the simplices, in this case cliques, which appear at every step. There are more efficient data structures that can be used, see , but since the data sets analysed are of small size, this more naive approach sufficed.

The NetworkX package has methods which can compute every clique in a graph. In a loop over k from 1 to one minus the number of points in the cloud, the correspoinding MkNN graph is computed and a list of its cliques is extracted. Of these, all those which are new and have size at most one more than the ambient dimension —such that the simplices they determine have dimension at most the ambient dimension— are appended to a list. The `Simplex` wraps a clique as a list of its points as well as keeping track of the step, k, at which the clique is born. In addition it implements methods used in the actual computation of the homology groups.

This list is then sorted by birth, such that cliques born earlier appear firts, and cliques with the same birth are sorted by increasing size. This guarantees that any simplex is always preceded by its faces, which is required for later computations.

3.4 Computing the homology

The classes `Simplex` and `Chain` can be used to calculate with the chain groups. As mentioned before, a `Simplex` simply wraps a clique as well as its birth. It also has the property `faces`, which returns a list of the faces of the simplex.

The elements of the chain group are linear combinations of simplices (of the same dimension), but since we are working over $\mathbb{F}_2$, this amounts to a list of simplices, which is what `Chain` stores. Furthermore, this class implements the addition of chains which, again because the field we ar working over has characteristic 2, reduces to taking the symmetric difference of the lists of simplices of the two chains we are adding. Indeed, if we have two chains of the form $c_1 = \sum_{i=1}^n \epsilon_i \sigma_i$ and $c_2 = \sum_{i=1}^n \delta_i \sigma_i$ with $\epsilon_i, \delta_i \in \mathbb{F}_2$, then

$$c_1 + c_2 = \sum_{i=1}^{n} (\epsilon_i + \delta_i)\sigma_i$$

which means the coefficient of σ_i in $c_1 + c_2$ is $\epsilon_j + \delta_j$. And this will be 1 provided only one of ϵ_j or δ_j is equal to 1, and will be 0 whenever *both* ϵ_j and δ_j are 1 or 0. So σ_j will be present in $c_1 + c_2$ whenever it is present in c_1 or c_2, but not both.

This makes it very easy to implement the boundary morphisms. For a single simplex, wrap the list of faces inside a `Chain` object. And for a larger chain, add the boundaries of each of its constituent simplices. Again, this works because we are taking coefficients from $\mathbb{F}_2$, so that, as explained in the alternating signs that appear in the definition of the boundary all disappear.

With all of this in place we can

Bibliography

[Cam18] Martín Campos. “Filtraciones en homología persistente mediante estimadores kernel de densidad”. Universitat Autònoma de Barcelona, Facultat de Matemàtiques, June 2018.

[EH08] H. Edelsbrunner and J. Harer. “Persistent homology—a survey”. In: *Discrete & Computational Geometry - DCG* 453 (Jan. 2008). DOI: 10.1090/conm/453/08802. URL: https://pub.ist.ac.at/~edels/Papers/2008-B-02-PersistentHomology.pdf.

[EM12] H. Edelsbrunner and D. Morozov. “Persistent homology: theory and practice”. In: *Proceedings of the European Congress of Mathematics.* (July 2–7, 2012). Europ. Soc. of Mathematics. Krakow, Jan. 2012, pp. 31–50. URL: https://pub.ist.ac.at/~edels/Papers/2012-P-11-PHTheoryPractice.pdf.

[EM16] H. Edelsbrunner and D. Morozov. “Persistent Homology”. In: *Handbook of Computational and Discrete Geometry, 3rd ed.* Ed. by J. E. Goodman, J. O’Rourke, and C. D. Tóth. Boca Raton, FL: CRC Press, 2016. Chap. 24. URL: https://pub.ist.ac.at/~edels/Papers/2016-B-01-PersDM.pdf.

[Gri+17] J. J. M. van Griethuysen et al. “Computational radiomics system to decode the radiographic phenotype”. In: *Cancer Research* 77.21 (2017), pp. 104–107. DOI: 10.1158/0008-5472.can-17-0339.

[Hat01] Allen Hatcher. *Algebraic Topology.* Cambridge University Press, 2001. URL: pi.math.cornell.edu/~hatcher/AT/AT.pdf.

[Rie16] Emily Riehl. *Cateogry Theory in Context.* Dover Publications, 2016. URL: http://www.math.jhu.edu/~eriehl/context.pdf.