Persistent Homology in Data Science

Stefan Huber
Salzburg University of Applied Sciences
Salzburg, Austria
stefan.huber@fh-salzburg.ac.at

***Abstract*—Topological data analysis (TDA) applies methods of topology in data analysis and found many applications in data science in the recent decade that go well beyond machine learning. TDA builds upon the observation that data often possesses a certain intrinsic shape such as the shape of a point cloud, the shape of a signal or the shape of a geometric object. Persistent homology is probably the most prominent tool in TDA that gives us the means to describe and quantify topological properties of these shapes.**

In this paper, we give an overview of the basic concepts of persistent homology by interweaving intuitive explanations with the formal constructions of persistent homology. In order to illustrate the versatility of TDA and persistent homology we discuss three domains of applications, namely the analysis of signals and images, the analysis of geometric shapes and topological machine learning. With this paper we intend to contribute to the dissemination of TDA and illustrate their application in fields that received little recognition so far, like signal processing or CAD/CAM.

***Index Terms*—topological data analysis, persistent homology, shape, kernel, machine learning, applications**

I. Introduction

The field of topological data analysis leverages mathematical tools from algebraic topology to problems in data science. This is motivated by the observation that data often has intrinsic *shape* that can be captured and quantified by notions from topology, most notably from persistent homology. In recent work a lot of work investigated and demonstrated the application of this topological information for machine learning tasks. Besides machine learning, however, topological methods emerge as a general framework for a broader spectrum of applications.

One illustrative approach to the idea of persistent homology starts with a problem statement of the following type: Assume we are given a continuous function $f\colon [0,1]^2 \to [0,1]$. We could interpret f as a geographic height profile over the unit square or as a grayscale picture, where we assume that the co-domain $[0,1]$ spans from black to white. Now assume we would like to identify mountains and volcanoes from f. Intuitively a volcano displays itself as a "bright ring" the grayscale picture, but it may not be straightforward to come up with an elementary mathematical notion that makes this intuition precise. Mountains, on the other hand, could be defined as local maxima in f. However, in real-world data, which is noisy, we would end up with numerous mountains and it would not be straightforward to make our intuition of a "significant mountain" mathematically precise. Number and significance of these mountains would depend on the statistical properties of the noise. In practice, often heuristics and filters are applied instead, which requires us to choose parameters accordingly and often there is no one-size-fits-all choice.

Persistent homology provides us with precise mathematical notions to (i) capture features like mountains, volcanoes and higher-dimensional counterparts in a natural way in the original, unprocessed data and (ii) quantify their significance.[1]

In all brevity, for the concrete problem of detecting mountains the method of persistent homology is also known as the *watershed transformation* in image processing, which helps to understand the idea of persistent homology in its generality: We consider the super-level sets $U_c = f^{-1}([c,1]) = \{x \in [0,1]^2 \colon f(x) \geq c\}$ of f at levels $c \in \mathbb{R}$, where c starts at ∞ and then declines towards $-\infty$. As c declines, U_c grows and changes its topology, e.g., connected components (islands) are created and re-merge later on, holes emerge and are filled up again. Persistent homology essentially tracks the evolution of these topological features in arbitrary dimensions and tells us for how long they lasted. A mountain or volcano is "more significant" if it lives longer.[2]

Persistent homology provides us with a theoretical framework that cannot only be applied in the identification of peaks and cycles in images, but has found numerous other applications, such as clustering, image analysis, shape recognition, image segmentation, analysis of time series analysis, analysis of biological structures (drug molecules, roots, et cetera), material analysis, and more, see [1]–[3] for an overview. An introductory textbook in the field of computational topology, including persistent homology, has been given by Edelsbrunner and Harrer [1] in 2010. The original paper, in which the concept of persistence was introduced is due to Edelsbrunner et al. [4] in 2002, however, precursors date back to the beginning of the 1990s.

The remainder of this paper is organized as follows: We will give an introduction into persistent homology and the necessary topological requirements. For the sake of simplicity and brevity, however, we will not go into various details, e.g., we skip all kind of generalizations of the concept of persistence. Instead, we will highlight three exemplary domains of applications, where persistent homology can be applied:

Partially supported by Digitales Transferzentrum, Salzburg.

[1] However, to capture certain features, we choose a suitable filtration. In this sense the filtration is a parameter to TDA.

[2] In topography this concept is known as the "prominence" of a mountain.

The analysis of signals and images (e.g., peak detection), the analysis of geometric shapes (e.g., clustering) and its application in machine learning.

II. Persistent homology

The take-away of the watershed model from a topological point of view is the following: Consider a sequence of growing topological spaces – called *filtration* – and track its topological features – namely *homology groups*. Different filtrations lead to different results in the evolution of the homology groups. The way we choose the filtration gives us means to pull in information specific to our application, e.g., geometric information. That is, we choose the filtration according to our application; in case of the watershed model the answer is the so-called super-level set filtration and other applications make use of different filtrations.

A. Simplicial complexes

We first require topological spaces that allow for an computational, algorithmic treatment. Simplicial complexes are prime examples that can be seen as a generalization of graphs.

A geometric n-simplex in $\mathbb{R}^d$ is the convex hull of $n+1$ (affinely independent) points $p_0, \dots, p_n$. That is, a 0-simplex is a point, a 1-simplex is a straight-line segment, a 2-simplex a triangle, a 3-simplex a tetrahedron, and so on. An n-simplex is an n-dimensional convex polytope. Moreover, every face of an n-simplex is a simplex itself, e.g., the vertices, edges and facets of a tetrahedron are simplices themselves.

A simplicial complex $\mathcal{S}$ is a set of simplices with the following two properties: First, for each simplex $\sigma \in \mathcal{S}$ all its faces are elements of $\mathcal{S}$ as well. Secondly, for every non-disjoint pair $\sigma_1, \sigma_2 \in \mathcal{S}$ the intersection $\sigma_1 \cap \sigma_2$ is in $\mathcal{S}$ as well. For short, a simplicial complex is a set of simplices that is closed under (non-empty) intersection and face decomposition, see Fig. 1.

The dimension of $\mathcal{S}$ is the maximum of the dimensions of its simplices $\sigma \in \mathcal{S}$. For instance, a planar straight-line embedding of a graph G yields an example for a 1-dimensional simplicial complex. The 0-simplices are the vertices and the 1-simplices are the edges. We define the underlying space $|\mathcal{S}|$ of $\mathcal{S}$ as the union of all its simplices together with the topology inherited from the ambient space where $\mathcal{S}$ lives in. We can think of $\mathcal{S}$ being a (generalized) triangulation of $|\mathcal{S}|$.

For topological considerations of simplicial complexes (or graphs), we are often not so much interested in a particular geometric realization but rather in the abstract structure. Hence, in analogy to (abstract) graphs, we define an abstract n-simplex σ as a set of n elements, and all its non-empty subsets $\sigma' \subset \sigma$ are its faces.[3] An abstract simplicial complex is then a system $\mathcal{S}$ of sets that is closed under (non-empty) intersection and building subsets (faces). The vertex set $V(\mathcal{S})$ of $\mathcal{S}$ is the union of all its elements.[4]

A simplicial complex $\mathcal{S}' \subset \mathcal{S}$ is called a simplicial subcomplex of $\mathcal{S}$. In Fig. 3, a simplicial complex and a subcomplex is shown. We define the k-skeleton of $\mathcal{S}$ as the subset of $\mathcal{S}$ that contains only simplices of dimension at most k, which is again a subcomplex of $\mathcal{S}$, see Fig. 1.

B. Homology

In the watershed example in Section I we referred to topological features like connected components and holes. Homology is mathematical tool that allows for a computational treatment of these entities: We add algebraic structures that allow us to *compute* boundaries and holes.

Intuitively, let c be a closed path on the 1-skeleton of a simplicial complex $\mathcal{S}$. If we cannot "continuously contract" c to a point within $|\mathcal{S}|$ then $\mathcal{S}$ contains a hole and c forms a loop around it. In Fig. 1, the loop visiting e_4, e_6, e_7, e_8 gives an example. Similarly, if c is a closed surface on the 2-skeleton of $\mathcal{S}$ that cannot be contracted to a point then $|\mathcal{S}|$ contains a cavity.

We algebraically define a p-chain of $\mathcal{S}$ is a formal sum $\sum \lambda_i \sigma_i$ of p-simplices σ_i of $\mathcal{S}$. For instance, $e_4+e_6+e_7+e_8$ is a 1-chain, but also e_1+e_6. The coefficients λ_i are in $\mathbb{Z}_2$, which means that λ_i counts modulo 2 how often a simplex σ_i is present in the chain.[5] For instance, consider a triangulated quadrilateral with two triangles s_1, s_2 as in Fig. 1. Let $c_1 = e_1 + e_2 + e_3$ denote the 1-chain formed by the sum of the three boundary edges of s_1 and likewise for c_2 and s_2. Then the sum $c_1 + c_2$ can be computed by adding up coefficients and results in $e_1 + e_2 + e_4 + e_5$, which is 1-chain around the quadrilateral. Note that the edge e_3, which is shared by s_1 and s_2, is present $1+1=0$ times due to modulo-2 arithmetic. We denote by C_p the set of all p-chains on $\mathcal{S}$, which forms an algebraic group with the operator $+$. We interpret C_p as the set of "paths" made up by p-simplices and the operator $+$ as an xor-summation of the path's simplices.

We identify holes in $|S|$ by finding "closed" chains around them, such as $e_4+e_6+e_7+e_8$ in Fig. 1. A chain is "closed" – we call it a cycle – if it has no boundary: The boundary

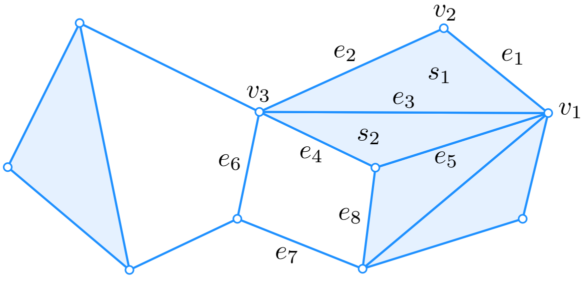


Fig. 1: A simplicial complex $\mathcal{S}$ of dimension 2 consisting of vertices, edges and triangles. The intersection of any two simplices is either empty or a member of $\mathcal{S}$. If we remove the shaded triangles we obtain the 1-skeleton. The 1-chain $c = e_4 + e_6 + e_7 + e_8$ is a 1-cycle as $\partial c = 0$. On the other hand, $\partial(e_1 + e_2) = v_1 + v_3$.

[3] In topology $\subset$ typically includes non-proper subsets and hence $\sigma \subset \sigma$.

[4] We further assume that $V(\mathcal{S}) \cap \mathcal{S} = \emptyset$.

[5] In computational topology most applications work with $\mathbb{Z}_2$, but we could generalize to an arbitrary ring.

$\partial\sigma$ of a p-simplex σ is the $(p-1)$-chain formed by all $(p-1)$-dimensional faces of σ. The boundary of a p-chain $c = \sum_i \lambda_i \sigma_i$ is defined by $\partial c = \sum_i \lambda_i \partial\sigma_i$. In Fig. 1, we can compute $\partial s_1 = e_1+e_2+e_3$ and $\partial(s_1+s_2) = e_1+e_2+e_4+e_5$. We say that the p-chain c is a p-cycle if it has no boundary, i.e., $\partial c = 0$. For instance ∂s_1 in Fig. 1 is a cycle as $\partial\partial s_1 = 0$.

Note that ∂c of a p-chain c forms a $(p-1)$-chain. The map $\partial_p \colon C_p \to C_{p-1} \colon c \mapsto \partial c$ is called the p-th boundary homomorphism. Hence, the set Z_p of p-cycles in C_p can be written as $Z_p = \ker\partial_p$. The set B_p of p-boundaries is the set of p-cycles that form the boundary of some $(p+1)$-cycle and therefore $B_p = \operatorname{im}\partial_{p+1}$. Note that a p-boundary is itself a p-cycle and therefore $B_p \subset Z_p$. In other words, $\partial_p\partial_{p+1} = 0$.

The sequence $B_p \subset Z_p \subset C_p$ constitutes a sequence of subgroup relationships and we can consider the quotient group of Z_p modulo B_p. In fact, this is the definition of the p-th homology group H_p, namely Z_p/B_p. In other words, H_p is the group of equivalence classes (called homology classes) that are formed by p-cycles that are equivalent modulo p-boundaries. In Fig. 1, the two 1-cycles $c_1 = e_6+e_7+e_8+e_4$ and $c_2 = e_6+e_7+e_8+e_5+e_3$ belong to the same homology class as $c_1 = c_2 + \partial s_2$. Let $c_1, c_2 \in Z_p$ belong to the same homology class, i.e., $c_2 = c_1 + b$ with $b \in B_p$ and therefore $b = c_1 + c_2$ due to $\mathbb{Z}_2$ coefficients. There is a $d \in Z_{p+1}$ with $\partial_{p+1} d = b$. A sketchy but possibly still helpful analogy might be to interpret d as a "cylinder": Its boundary is $\partial d = c_1 + c_2$ and we think of deforming c_1 into c_2 along d.

In the previous example, we had two 1-cycles around the same hole that belong to the same homology class. In some sense the homology group is actually *generated* by the set of holes: If we have a (triangulated) simple polygon P then H_1 contains only the trivial class B_1, as every 1-cycle can be contracted to a point. If we punch a hole into P then we have two classes, B_1 and the set of cycles around the hole. If we punch a second hole then we have four classes: B_1, cycles c_1 that go around one hole, cycles c_2 that go around the other hole and cycles $c_1 + c_2$ that go around both. In this sense, H_1 is a group generated by two holes and the cosets are $B_1, c_1{+}B_1, c_2{+}B_2, (c_1{+}c_2){+}B_1$. An analogous example for H_2 could involve a tetrahedrized sphere where we punch out cavities.

One way to summarize the above said in a more algebraic way is by means of the notion of a so-called chain complex, i.e., a sequence

$$\cdots \xrightarrow{\partial_3} C_2 \xrightarrow{\partial_2} C_1 \xrightarrow{\partial_1} C_0 \xrightarrow{\partial_0} 0$$

of Abelian groups C_p and homomorphisms ∂_p with the property $\partial_p\partial_{p+1} = 0$. Then

$$H_p = \ker\partial_p / \operatorname{im}\partial_{p+1}. \tag{1}$$

The rank of H_p is called the p-th Betti number β_p. In other words, β_2 counts cavities, β_1 counts holes and β_0 counts connected components.[6] A torus has $\beta_0 = 1, \beta_1 = 2$ and $\beta_p = 0$ for $p \geq 2$, just like a polygon with two holes or the simplicial complex in Fig. 1.

C. Persistent homology

The Betti numbers we gain as information from homology are too coarse for most applications in data science. For instance, if we consider a point cloud of n points in $\mathbb{R}^d$ then $\beta_0 = n$ and $\beta_p = 0$ for $p \geq 1$, no matter what "shape" the point cloud resembles. However, if we grow a ball around each point at unit speed and consider the evolution of their union and its Betti numbers, then we may identify clusters: The balls within a cluster quickly merge together (and β_0 drops), but the inter-cluster merges happen later, which enables us to distinguish them, see Fig. 6. Persistent homology is the tool to track the evolution of the β_p and clustering is just one application.

Let $\mathcal{S}$ denote a simplicial complex then a filtration of $\mathcal{S}$ is a sequence $\emptyset = \mathcal{S}_0 \subset \cdots \subset \mathcal{S}_m = \mathcal{S}$ of nested simplicial subcomplexes. We interpret the sequence $(\mathcal{S}_k)_{k=0}^m$ as an evolution of a simplicial complex in which we track the birth and death of homology classes.

Assume that $\mathcal{S}_{i+1} \setminus \mathcal{S}_i$ consists of a single p-simplex σ. The insertion of σ into $\mathcal{S}_i$ may have exactly one of two effects on the homology: (i) A p-cycle is born as this σ closed the cycle or (ii) a $(p-1)$-cycle has died as σ finally filled the cycle up. For instance, adding an 1-simplex (edge) into $\mathcal{S}_i$ either closes a cycle or merges two connected components.[7] Adding a 2-simplex either closes a cavity and β_2 increases or fills up a 1-cycle and β_1 decreases.

Roughly speaking, the persistent p-th homology group $H_p^{i,j}$ contains those homology classes that have been born not later than in $\mathcal{S}_i$ and die no earlier than $\mathcal{S}_j$. In other words, $H_p^{i,j}$ contains those homology classes that were alive throughout the time span $[i,j]$ in the filtration $(\mathcal{S}_k)_{k=0}^m$. The p-th persistent Betti number $\beta_p^{i,j}$ is the rank of $H_p^{i,j}$.

For a more precise definition we recall the definition of H_p based on a chain complex. We stack the chain complexes of the filtration elements $\mathcal{S}_i$ onto each other and obtain the commutative diagram

$$\begin{tikzcd}
\cdots \arrow[r, "\partial_3^0"] & C_2^0 \arrow[r, "\partial_2^0"] \arrow[d, hook] & C_1^0 \arrow[r, "\partial_1^0"] \arrow[d, hook] & C_0^0 \arrow[r, "\partial_0^0"] \arrow[d, hook] & 0 \\
\cdots \arrow[r, "\partial_3^1"] & C_2^1 \arrow[r, "\partial_2^1"] \arrow[d, hook, dotted] & C_1^1 \arrow[r, "\partial_1^1"] \arrow[d, hook, dotted] & C_0^1 \arrow[r, "\partial_0^1"] \arrow[d, hook, dotted] & 0 \\
\cdots \arrow[r, "\partial_3^m"] & C_2^m \arrow[r, "\partial_2^m"] & C_1^m \arrow[r, "\partial_1^m"] & C_0^m \arrow[r, "\partial_0^m"] & 0
\end{tikzcd}$$

where C_p^i denotes $C_p(\mathcal{S}_i)$, i.e., the set of p-chains in $\mathcal{S}_i$, and the vertical maps denote inclusion maps. Then for $0 \leq i \leq j \leq m$ the persistent p-th homology group $H_p^{i,j}$ of the filtration $(\mathcal{S}_k)_{k=0}^m$ is defined as

$$H_p^{i,j} = \ker\partial_p^i / (\operatorname{im}\partial_{p+1}^j \cap \ker\partial_p^i). \tag{2}$$

[6] The boundary of a 0-chain is zero, i.e., $\partial_0 = 0$ and $C_{-1} = 0$.

[7] In some sense, the space between two connected components got filled up, so 0-cycles can also be seen as inter-component gaps that die when two component get merged.

Note that $\ker \partial_p^i$ is the set of p-cycles in $\mathcal{S}_i$ and $\operatorname{im} \partial_{p+1}^j$ are the p-boundaries in $\mathcal{S}_j$. In essence, $H_p^{i,j}$ consists of the p-cycles in $\mathcal{S}_i$ modulo the p-boundaries in $\mathcal{S}_j$. So for a non-trivial p-cycle[8] to be in $H_p^{i,j}$ it must be born until (exist in) $\mathcal{S}_i$ and it must not die until (be filled up in) $\mathcal{S}_j$, otherwise it would be zero modulo a p-boundary in $\mathcal{S}_j$. Also note that $H_p^{i,i}$ is simply the p-th homology group $H_p(\mathcal{S}_i)$ of $\mathcal{S}_i$ by (1).

The persistent Betti number $\beta_p^{i,j}$ is the number of *independent* p-dimensional homology classes that generate $H_p^{i,j}$, i.e., that are born no later than in $\mathcal{S}_i$ and live until $\mathcal{S}_j$. The number of independent classes that are born until $\mathcal{S}_i$ but die exactly with $\mathcal{S}_j$ is therefore given by $\beta_p^{i,j-1} - \beta_p^{i,j}$. Hence, the number

$$\mu_p^{i,j} = (\beta_p^{i,j-1} - \beta_p^{i,j}) - (\beta_p^{i-1,j-1} - \beta_p^{i-1,j}) \qquad (3)$$

counts the number of independent p-dimensional homology classes that are born exactly at $\mathcal{S}_i$ and die exactly with $\mathcal{S}_j$.

Vice versa, the $\beta_p^{i,j}$ can be reconstructed by summing up $\mu_p^{i,j}$, which gives the fundamental lemma of persistent homology. The so-called p-th persistence diagram is a way to encode all $\mu_p^{i,j}$, as we will see in the next section. So the fundamental lemma says that all the information of persistent homology groups is encoded by persistence diagrams. [1]

D. Monotonic functions and persistence diagrams

So far birth and death of a homology class refers to the index i of the simplicial complex $\mathcal{S}_i$ in a filtration $(\mathcal{S}_k)_{k=0}^m$. But in our initial example of finding mountains and volcanoes – homology classes in dimension 0 and 1 – in a height map we would rather talk about points in time as real numbers.

That is, in a simplicial complex $\mathcal{S}$ we assign each simplex $\sigma \in \mathcal{S}$ a time $\varphi(\sigma)$, with $\varphi\colon \mathcal{S} \to \mathbb{R}$, that tells when σ appears in the filtration. In order for $\mathcal{S}_t = \{\sigma \in \mathcal{S}\colon \varphi(\sigma) \leq t\} = \varphi^{-1}((-\infty, t])$ to form a simplicial complex we require that $\varphi(\sigma') \leq \varphi(\sigma)$ when σ' is a face of σ. We call such a $\varphi\colon \mathcal{S} \to \mathbb{R}$ monotonic. We can interpret $(\mathcal{S}_t)$ as a filtration with a continuous index set $\mathbb{R}$. However, for a finite simplicial complex $\mathcal{S}$ changes only occur at finitely many points in time $t_0 < \dots < t_m$. Hence, we can define $\mathcal{S}_i = \mathcal{S}_{t_i}$ for $0 \leq i \leq m$ and obtain a filtration $(\mathcal{S}_k)_{k=0}^m$ as before.

There are $\mu_p^{i,j}$ independent homology classes that are born at t_i and die at t_j. Their persistence is defined as $t_j - t_i$. For a class that never dies we set the persistence to infinity.

The p-th persistence diagram D_p is a multiset of points (t_i, t_j) with multiplicity $\mu_p^{i,j}$ on the extended plane $\overline{\mathbb{R}}^2 = (\mathbb{R} \cup \{\pm\infty\})^2$, see Fig. 2. Each point in D_p therefore encodes a p-dimensional persistent homology class and its persistence is the vertical distance to the diagonal. Note that no points can be below the diagonal as $t_j \geq t_i$. However, it will turn out convenient to add all diagonal points of infinite multiplicity to D_p. This will become clear when we talk about bottleneck and Wasserstein distance between diagrams. Note that points on the diagonal mean zero persistence and in this sense "do not matter".

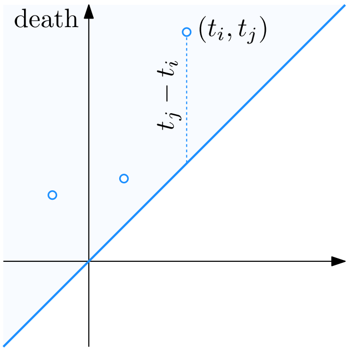


Fig. 2: A persistence diagram D_p is a multiset of points. Each point (t_i, j_j) encodes a persistent homology class that is born at time t_i and dies at time t_j. All points on the diagonal (blue) are contained in D_p with infinite multiplicity.

E. Algorithms

The concept of persistent homology along with an efficient algorithm has been introduced in [4]. The rough idea behind this algorithm is as follows: Let $\sigma_1, \dots, \sigma_n$ be the insertion sequence of the simplices of $\mathcal{S}$ according to the filtration. The so-called boundary matrix B is the $n \times n$ matrix over $\mathbb{Z}_2$ that encodes for a simplex σ_i its boundary simplices $\partial\sigma_i$, i.e., the i-th row contains a 1 in the j-th column if σ_j belongs to $\partial\sigma_i$ and 0 otherwise. So if $v \in \mathbb{Z}_2^n$ encodes a p-chain c then $B \cdot v$ encodes ∂c. Note that B contains ones only above the diagonal.

The algorithm now reduces B similar to Gaussian elimination using column reductions from left to right. For the j-th column C we have a look at the top-most 1, if there is any. If there is a column to the left that has a 1 in the same row then we add it up to C and effectively remove the 1 in C and repeat with the reduction. This procedure results in a reduced boundary matrix R.

We can record the reduction steps in a matrix V such that

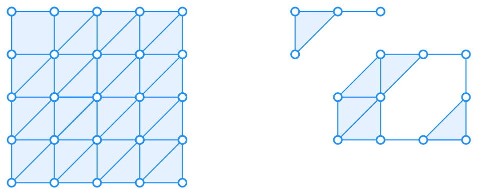
$$R = B \cdot V.$$

That is, we start with $B = B \cdot I$, where I denotes the identity matrix, and apply the reduction steps to the left-hand side B and synchronously to I until we end up with $R = B \cdot V$.

It can be shown that the j-th column of V encodes a p-chain c and $B \cdot c$ is the j-th column in R, which encodes ∂c. When we insert σ_j either a homology class is born or another one has died. It turns out that we can distinguish between these cases by looking at the j-th column of R: If the j-th column of R is a zero column then $\partial c = 0$. In fact, a homology class is born and c is a cycle in it. On the other hand, if the j-th column of R is not zero then a homology class died and the j-th column of R is a cycle of the class.

The reduction algorithm takes $O(n^3)$ time and allows us to construct all persistence diagrams from R in total linear time. In addition, we can read off a particular cycle of the persistent homology classes from R and V. This will be used in the image segmentation task of identifying the boundaries of biological cells or detecting peaks in signals in Section III.

[8] A respresentation of a homology class not equal to the boundary group.



(a) Grid triangulation (b) Subcomplex in the filtration

Fig. 3: Left: A triangulation $\mathcal{S}$ of the pixel grid. Right: Some subcomplex in the super-level set filtration of the grayscale image with one 1-dimensional homology class (volcano).

A couple of improvements on the basic matrix reduction algorithm have been published in recent years, e.g., Chen and Kerber [5] save reduction steps of columns and Bauer et al. [6] gave a distributed algorithm for persistent homology. For some applications cubical complexes rather than simplicial complexes can be used to improve performance [7]. The special case 0-dimensional persistent homology can be computed in $O(n\alpha(n))$ time by tracking the merges of components with a union-find data structure [1]. Software packages, like TDA for R, provide interfaces to efficient implementations, like PHAT [8] for C++.

III. Analysis of Signals and Images

A. Super-level set filtration and the watershed model

Our initial example of detecting mountains and volcanoes in a grayscale image now boils down to computing persistent homology for a specific filtration that relates to the watershed model. We recall that we obtain a growing sequence of spaces as a growing landmass as the water level declines. In a continuous setting, we have a grayscale image $f\colon [0,1]^2 \to [0,1]$ and we consider the super-level sets $U_c = f^{-1}([c,1])$, where c is the water level.

In a discrete setting, we interpret the pixels of the grayscale image as a grid of points, which we triangulate to obtain a simplicial complex $\mathcal{S}$, see Fig. 3a.[9] The watershed model naturally leads to a monotonic function φ that corresponds to $-f$, i.e., the points of higher function values of f appear earlier. More precisely, we define $\varphi(\sigma) = -\min_{p\in\sigma} f(p)$, where we interpret σ as a geometric simplex. The filtration we obtain from φ is called the *super-level set filtration* of f. We can define the sub-level set filtration of a function f analogously.

A mountain in the grayscale image is a point in the 0-th persistence diagram and its significance is its vertical distance to the diagonal. Similarly, a volcano is a point in the 1-th persistence diagram. In Fig. 3b we see a subcomplex of $\mathcal{S}$ in the super-level set filtration where one non-trivial 1-dimensional persistent homology class (volcano) is alive. It will die when all triangles within the cycle have been inserted by the filtration (i.e., reached by the water level).

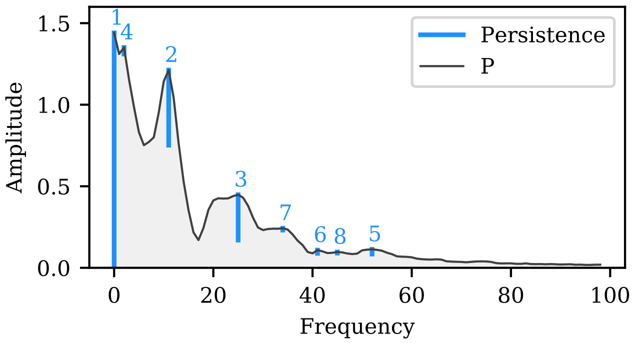


Fig. 4: The function $P\colon |\mathcal{S}| \to [0,\infty)$ and its eight most persistent 0-th dimensional persistent homology classes in the super-level set filtrations. The vertical bars illustrate the persistence. Class 1 has infinite persistence.

The problem of detecting mountains and volcanoes is of a generic nature in image segmentation and comes at all variants, e.g., detecting cell boundaries in microscopy images [1] or detecting traces of animal paws [9].

B. Noisy signals and peaks

Detecting peaks in signals is a common task in all kind of engineering disciplines. For instance, we would like to adjust the feedback controller of a drive in automation industry. To this end, we would like to identify the most significant natural frequency of the mechanics attached to the drive. One way to do so is to impinge an excitation signal and then analyze the Fourier transform $\mathcal{F}(p)$ of the position response signal $p(t)$. Let us denote by $P\colon [0,\infty) \to [0,\infty)$ the absolute value of the Fourier transform $\mathcal{F}(p)$, see Fig. 4. Then we interpret the most dominant peak in P at a non-zero frequency is the most significant natural frequency of the system, which would be peak 2 in Fig. 4.

The global maximum of P is often at $P(0)$, so we look for a local maximum. The second largest local maximum is often due to noise around the global maximum and we would call it an artifact rather than "dominant".

Persistent homology gives us the means to precisely define what we mean by a "dominant peak" on the original data. To this end, assume that the discrete function P is given at finitely many real points $\omega_1 < \cdots < \omega_n$. This leads to a 1-dimensional simplicial complex $\mathcal{S}$ with 0-simplices $\{\omega_i\}$ and 1-simplices $\{\omega_i, \omega_{i+1}\}$. We interpret P is a (piecewise linear) function $|\mathcal{S}| \to [0,\infty)$ and consider its super-level set filtration. Peaks are now points in the 0-th dimensional persistence diagram D_0 and we look for the most persistent point in D_0 that is not due to $P(0)$, which would be class 2 in Fig. 5.[10]

The persistence diagrams are stable with respect to noise. If we perturb all function values of P by at most ϵ then birth and death may be shift by at most ϵ so the persistence may

[9] We interpret f as a piecewise linear function $|\mathcal{S}| \to \mathbb{R}$ that is linear over each simplex $\sigma \in \mathcal{S}$, see [1] for details.

[10] The matrix R in Section II-E tells which simplex gave birth to this class.

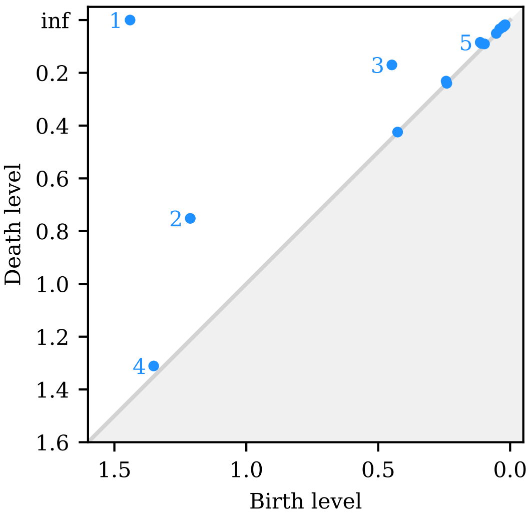


Fig. 5: The 0-th persistence diagram D_0 of the function P in Fig. 4. The axes labels correspond to the function levels of P, so higher values appear earlier in the super-level set filtration.

change by at most 2ϵ and the loci of points in the persistence diagram may change by ϵ in the ∞-norm. Adding noise will create many low-persistence points, i.e., many points will be pulled out from the diagonal[11] and previously low-persistence points may be absorbed by the diagonal, but all that happens only within a vertical ϵ-neighborhood to the diagonal.

Put in different words, persistence diagrams are stable with respect to the bottleneck distance. That is, let F and G denote two persistence diagrams. In order to measure their similarity, we measure how close we can match the individual diagram points, i.e., we consider bijections $\mu\colon F \to G$. Based on this intuition, the bottleneck distance d_B is defined as

$$d_B(F,G) = \inf_{\mu} \sup_{x \in F} \|x - \mu(x)\|_\infty .$$

Let now F and G denote the p-th persistence diagrams we obtain from the super-level set filtrations of two functions f, g. Then bottleneck stability of persistence diagrams holds in the following sense [10]:

$$d_B(F,G) \le \|f - g\|_\infty$$

This crucial property makes persistent homology an attractive tool if we have to deal with noisy data.

IV. Shape Analysis

As we initially mentioned in Section II, persistent homology analyses the evolution of a filtration, but how we obtain a filtration is a different story. The examples in Section III employed the super-level set filtration of functions. In this section we present filtrations that stem from growing geometric sets, e.g., according to the Minkowski sum of sets. The Minkowski sum of two sets A and B in $\mathbb{R}^d$ is defined as $A \oplus B = \{x + y\colon x \in A, y \in B\}$. In the following B_t denotes the ball of radius t centered at the origin.

A. Point sets and alpha complexes

Often in data analysis we have to deal with finite point sets P in $\mathbb{R}^d$ and often this point sets display some sort of shape. For instance, clustering algorithms tessellate the set P into clusters of points that are close to each other.

Persistent homology gives us the means to investigate the shape of P. The Minkowski sum $P_t = P \oplus B_t$ is the union of balls of radius t placed at each point of P, see Fig. 6. Let us consider the sequence of sets P_t as we increase t. We start at $P_0 = P$ and as t grows, balls merge together. Points that form a cluster will be merged sooner while the inter-cluster merges will happen later. The sequence (P_t) can be interpreted as a filtration and the 0-th persistence diagram D_0 visualizes the previous observation in the sense that points in D_0 of higher persistence encode more dominant clusters in P.

The alpha complex is a simplicial complex that capture the topology of P_t in the sense that they are homotopy equivalent [1]. It is a subcomplex of the Delaunay triangulation of P containing those edges where the respective balls intersect, see Fig. 6. Instead of investigating the sets P_t, which are not simplicial complexes, we can instead investigate filtrations that stem from the alpha complex. In fact, as mentioned in [1], alpha complexes were the starting point for the work on persistent homology. The alpha complex is closely related to the Čech complex [11], which together with the Vietoris-Rips complex present the more prominent complexes over point sets in computational topology.

B. Polygonal shapes and offset filtrations

Often data is not modeled as a point cloud but rather as a geometric shape or as an image that resembles a geometric shape, e.g., in medical imaging, geographical information systems or CAD/CAM. Such a shape is often modeled by (a set of) polygons, possibly with holes. By a polygon P with holes we mean a polygon from which we remove polygons in the interior, as illustrated in Fig. 7. In image processing and computational geometry the concept of a topological skeleton of P – such as the medial axis, the Voronoi diagram and the

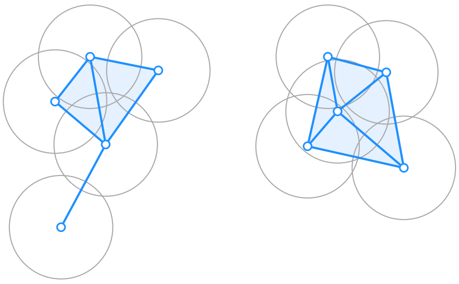


Fig. 6: A point set P with balls B_t at each point. The alpha complex (blue) is a subcomplex of the Delaunay triangulation of P and has the same homotopy type as $P \oplus B_t$.

[11] Recall that each point of the diagonal has infinite multiplicity.

straight skeletons – has been developed as versatile tool in the past five decades. Persistent homology is in close relationship to these [12].

Let us consider the following shrinking process of P: We move all edges of P inwards in parallel at unit speed, see Fig. 7. This moving wavefront changes its shape at certain points in time: Edges collapse to zero length and the topology may change when edges are split into parts. At any point in time, this wavefront forms a so-called *mitered offset curve* of P. In CAD/CAM, for instance, this offset curves are used for tool path generation or tool radius correction. Let us denote by P_t the inset version of P at time t so that its boundary ∂P_t is the corresponding mitered offset curve at distance t to ∂P. So P_0 corresponds to the original P and at some large enough time t^* the inset P_{t^*} becomes the empty set.

If we reverse the inset process of P_t in time, we start with the $P_{t^*} = \emptyset$ and end up with $P_0 = P$. This gives a growing sequence of sets and therefore a filtration of P and we can again apply persistent homology, cf. [12]. This filtration has the interesting property that 1-dimensional homology classes never dies. Points in the 0-th persistence diagra D_0 correspond to the "main hunks" of the shape P, which could be used for polygon decomposition. One particular application could be the computation of high-speed NC machining tool paths [13].

We motivated this filtration by means of offset curves, however, we can put this filtration into the setting of super-level set filtrations as well: Consider the offset wavefront ∂P_t and project their evolution into three-space by interpreting the third dimension as time. That is, we consider the set $\bigcup_{t \geq 0} \partial P_t \times \{t\} \subset \mathbb{R}^3$, which we interpret as the function graph of a function $f\colon P \to [0, \infty)$. Note that for a point $p \in P$ the function value $f(p)$ tells when the wavefront hit the point p. Then the isolevel $f^{-1}(t)$ is exactly ∂P_t again. Hence, the super-level set filtration of f is the offset filtration of of P.

In [12], an algorithm is given that computes persistent homology of the filtration of mitered offsets of P by means of the so-called straight skeletons, see Fig. 7. The straight skeleton of P results from the traces of the vertices of the wavefront vertices and can be used to solve a multitude of geometric problems in CAD/CAM, GIS, terrain and roof generation, mathematical origami and so on, cf. [14]. In particular, it can be used to efficiently compute mitered offset curves in linear time. Moreover, in [12] it has been shown that the straight skeleton does not only encode the topology (homotopy type) of P, but in some sense also the topology (persistent homology) of the mitered offset filtration.

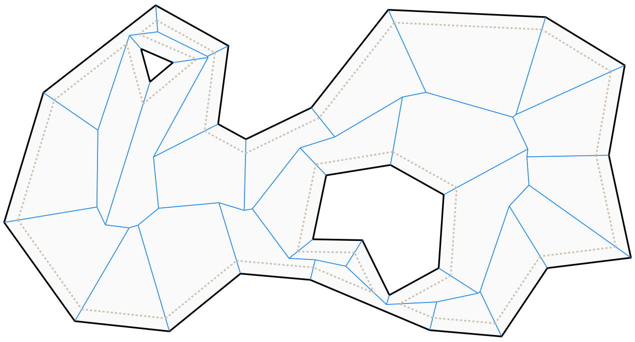


Fig. 7: A polygon P with two holes and a mitered offset ∂P_t in gray. The straight skeleton (blue) results from the traces of the vertices of the offset wavefront.

Instead of so-called mitered offset curves we could have also considered offset curves that are related to Minkowski sums: We define P_t as the largest subset of P with $P_t \oplus B_t \subset P$. These Minkowski-based offset curves ∂P_t can be efficiently computed by Voronoi diagrams. Although the underlying shape P is the same, we obtain different filtrations for the two kind of offsets. In some sense, the different filtrations pull in different geometric information of P.

V. Topological Machine Learning

Persistent homology gives us a summary description in form of persistence diagrams[12]. In order to leverage topological data analysis for machine learning, we need to apply machine learning methods on persistent diagrams, i.e., on multi-sets of points in the plane.

A. A scale-space kernel on persistence diagrams

Reininghaus et al. [15] presented a kernel on the set of persistence diagrams, which enables us to use all kernel-based machine learning techniques, like kernel-based SVM, k-means or PCA. A kernel k on a set X is a bivariate, symmetric, positive-definite function $k\colon X^2 \to \mathbb{R}$. Alternatively, a function $k\colon X \times X \to \mathbb{R}$ is a kernel if there is a Hilbert space H, called the feature space, and a map $\Phi\colon X \to H$, called the feature map, such that $k(x, y) = \langle \Phi(x), \Phi(y) \rangle$ for all $x, y \in H$. In this sense, a kernel plays the role of an inner product. In [15], this feature map has been explicitly constructed in order to obtain a kernel.

Let $\Omega = \{(x, y) \in \mathbb{R}^2 \colon y \geq x\}$ denote the half-plane above the diagonal of persistence diagrams. Then function space $L_2(\Omega)$ is chosen as Hilbert space and a suitable feature map $\Phi\colon \mathcal{D} \to L_2(\Omega)$ from the set $\mathcal{D}$ of persistence diagrams is constructed. Note that diagram points at the diagonal $\partial\Omega$ are of persistence zero and shall therefore not contribute to a meaningful kernel.

In order to achieve this in a natural way, Φ is constructed as a solution to a heat-diffusion partial differential equation: The boundary condition forces solutions to be zero at $\partial\Omega$. Furthermore, the initial condition says that the heat diffusion starts with a Dirac delta distribution δ_p at each point p of the diagram D, i.e., with $\sum_{p \in D} \delta_p$. For each time $t \in (0, \infty)$, we consider the solution $u_t\colon \Omega \to [0, \infty)$ of the partial differential equation as a target for Φ. These solutions u_t turn out to be a sum of Gaussians that cancel out on the diagonal $\partial\Omega$. That is, Reininghaus et al. [15] actually constructed a scale-space kernel k_t with a feature map $\Phi_t(D) = u_t$ that maps into $L_2(\Omega)$.

[12] There are also similar descriptions, for instance, persistent barcodes.

B. Stability

In order to learn topological features of data through a kernel k_t in a meaningful way, we require that small changes in the data lead to small changes in the kernel evaluation. Since we use persistence diagrams as a summary description of the data's topology, we also require stability for the persistence diagrams.

We already mentioned bottleneck stability in Section III. To prove stability of the kernel, we require a generalization of the bottleneck distance: The Wasserstein distance $d_{W,q}$ between diagrams F and G is defined as

$$d_{W,q}(F,G) = \left(\inf_{\mu} \sum_{x \in F} \|x - \mu(x)\|_{\infty}^{q} \right)^{\frac{1}{q}}.$$

Note that the bottleneck distance is a special case of the Wasserstein distance in the sense that $d_B = d_{W,\infty}$.

Also for the Wasserstein distance a stability result is known by [16]: For our setting, let f denote a 1-Lipschitz function on a simplicial complex $\mathcal{S}$. Then for any $k \geq 1$ the degree k total persistence, $\sum_{(b,d) \in D} |d-b|^k$ of the p-th persistence diagram D of f is bounded by a constant C. Let now f, g be two L-Lipschitz functions on $\mathcal{S}$ and F, G their p-th dimensional persistence diagrams, then for all $q \geq k$ it holds that

$$d_{W,q}(F,G) \leq \left(LC \cdot \|f-g\|_{\infty}^{q-k} \right)^{\frac{1}{q}}$$

Furthermore, it can now be shown that also the kernel k_t on persistence diagrams is in turn stable w.r.t. the 1-Wasserstein distance [15]:

$$\|\Phi_t(F) - \Phi_t(G)\|_{L_2(\Omega)} \leq \frac{1}{2\sqrt{\pi}t} d_{W,1}(F,g),$$

which completes the argument for the stability of the scale-space kernel k_t.

Bubenik [17] introduced the concept of persistence landscapes, which provide a map from persistence diagrams into the Banach space $L_p(\mathbb{R}^2)$, which was originally intended for statistical computations. In case of $L_2(\mathbb{R}^2)$, persistence landscapes also yields a stable kernel on persistence diagrams, see [15] for a comparison. Kwitt et al. [18] later obtained a universal kernel in the sense of Steinwart as modified version the scale-space kernel from [15]. Recently, Hofer et al. [19] presented a deep learning approach on persistence diagrams that is related to [15]. The key idea is that some regions of Ω could be more important to a given learning problem than others and they were able to include a weight over Ω as part of the learning task.

VI. Conclusion

The aim of this paper is to support the dissemination of persistent homology. Topological data analysis (TDA), and persistent homology in particular, gained a lot of momentum in the recent decade. In some sense, persistent homology contributes to data science in two ways:

First, persistence diagrams provide a natural way to make various methods of data science applicable to disciplines, where shape plays a crucial role, like signal processing, image processing, computational geometry, and computational biology, to name a few.

Secondly, it is a tool within data science to understand data by providing a mechanism to describe the intrinsic shape of data. For instance, explainable AI tackles the big challenge of understanding the inner workings of (deep) neural networks. One approach has been presented by Carlsson and Gabrielsson [20], who apply persistent homology to the internal states of a CNN to gain insight in what the CNN has learned.

References

[1] H. Edelsbrunner and J. Harer, *Computational Topology – An Introduction*. American Mathematical Society, 2010, iSBN 978-0-8218-4925-5.

[2] C. S. Pun, K. Xia, and S. X. Lee, "Persistent-homology-based machine learning and its applications – a survey," *SSRN Electronic Journal*, 2018.

[3] L. Wasserman, "Topological data analysis," *Annual Review of Statistics and Its Application*, vol. 5, no. 1, pp. 501–532, 2018.

[4] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological persistence and simplification," *Discrete Comp. Geom.*, vol. 28, no. 4, pp. 511–533, Nov. 2002.

[5] C. Chen and M. Kerber, "Persistent homology computation with a twist," in *Proc. 27th Europ. Workshop on Comp. Geom. (EuroCG '11)*, Mar. 2011.

[6] U. Bauer, M. Kerber, and J. Reininghaus, "Distributed computation of persistent homology," in *Proc. 16thWorkshop Alg. Eng. and Exp. (ALENEX '14)*, Mar. 2014, pp. 31–38.

[7] H. Wagner, C. Chen, and E. Vuçini, "Efficient computation of persistent homology for cubical data," in *Topological Methods in Data Analysis and Visualization II*, ser. Mathematics and Visualization. Springer-Verlag, 11 2012, pp. 91–106.

[8] U. Bauer, M. Kerber, J. Reininghaus, and H. Wagner, "Phat - persistent homology algorithms toolbox," *J. Symb. Comput.*, vol. 78, pp. 76—90, Jan. 2017.

[9] Stackoverflow: Peak detection in a 2d array. [Online]. Available: https://stackoverflow.com/questions/3684484/peak-detection-in-a-2d-array/47190183#47190183

[10] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, "Stability of persistence diagrams," *Discrete Comp. Geom.*, vol. 37, no. 1, pp. 103–120, 2007.

[11] U. Bauer and H. Edelsbrunner, "The morse theory of Čech and delaunay complexes," *Transactions of the American Mathematical Society*, vol. 369, no. 5, pp. 3741–3762, 2017.

[12] S. Huber, "The topology of skeletons and offsets," in *Proc. 34th Europ. Workshop on Comp. Geom. (EuroCG '18)*, Mar. 2018.

[13] M. Held and C. Spielberger, "Improved spiral high-speed machining of multiply-connected pockets," *Comp. Geom. Theory & Appl.*, vol. 11, no. 3, pp. 346 – 357, 2014.

[14] S. Huber, *Computing Straight Skeletons and Motorcycle Graphs: Theory and Practice*. Shaker Verlag, Apr. 2012, iSBN 978-3-8440-0938-5.

[15] J. Reininghaus, S. Huber, U. Bauer, and R. Kwitt, "A stable multi-scale kernel for topological machine learning," in *Proc. 2015 IEEE Conf. Comp. Vision & Pat. Rec. (CVPR '15)*, Boston, MA, USA, Jun. 2015, pp. 4741–4748.

[16] D. Cohen-Steiner, H. Edelsbrunner, J. Harer, and Y. Mileyko, "Lipschitz functions have L_p-stable persistence," *Found. Comput. Math.*, vol. 10, no. 2, pp. 127–139, 2010.

[17] P. Bubenik, "Statistical topological data analysis using persistence landscapes," *Journal of Machine Learning Research*, vol. 16, pp. 77–102, Jan. 2015.

[18] R. Kwitt, S. Huber, M. Niethammer, W. Lin, and U. Bauer, "Statistical topological data analysis - a kernel perspective," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 3070–3078.

[19] C. Hofer, R. Kwitt, M. Niethammer, and A. Uhl, "Deep learning with topological signatures," in *Conf. Neural Inf. Proc. Sys.*, 2017.

[20] G. E. Carlsson and R. B. Gabrielsson, "Topological approaches to deep learning," *CoRR*, vol. abs/1811.01122, 2018. [Online]. Available: http://arxiv.org/abs/1811.01122