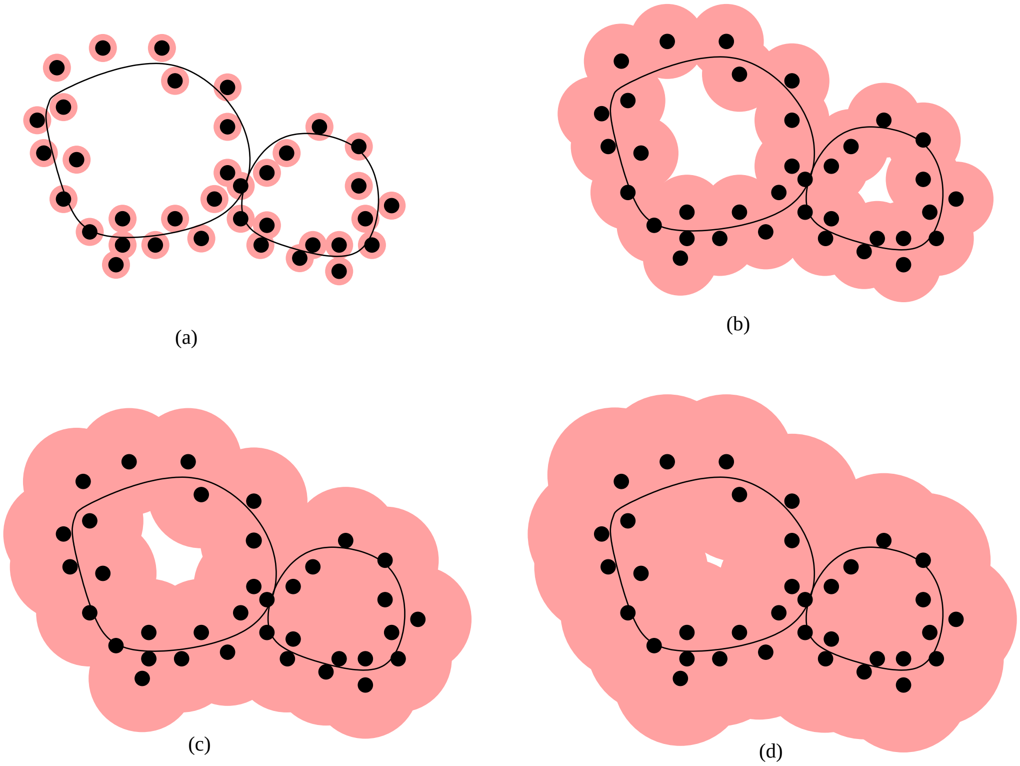
Persistent Homology

Suppose we have a noisy point set (data) sampled from a space, say a curve in $\mathbb{R}^2$ as in Figure 12. Can we get the information that the sampled space had two loops, one bigger and more prominent than the other? Let P denote the data points. Consider the distance function $r : \mathbb{R}^2 \to \mathbb{R}$ defined over $\mathbb{R}^2$ where $r(x)$ equals $d(x, P)$, that is, the minimum distance to the points in P. Now let us look at the sublevel sets of r, that is, $r^{-1}[0, a]$ for some $a \in \mathbb{R}^+ \cup \{0\}$. These sublevel sets are union of closed balls of radius a cenetring the points.


(a)

(b)

(c)

(d)

Figure 12: Noisy sample of a curve with two loops and the growing sublevel sets of the distance function to the sample points: The larger loop appearing as the bigger hole in the complement of the union of balls persists longer than the same for the smaller loop while other spurious holes persist even shorter.

As one can observe, if we increase a starting from zero, we come across different holes surrounded by the union of these balls which ultimately get filled up at different times. However, the two holes corresponding to the original two loops persist longer than the others. So, can we abstract out this observation where we look at the measure of how long a feature (homologies) survives when we scan over the increasing sublevel sets? This seems to weed out the 'false' features (noise) from the true ones. The notion of persistent homology formalizes this idea. *It takes a function defined on a topological space and quantizes the changes in homology as the sublevel sets develop with increasing value of the function.*

14 Definition

First, we consider a general topological space and later we specialize the concept to simplicial complexes.

Consider a real-valued function $f : \mathbb{T} \to \mathbb{R}$ defined on a topological space $\mathbb{T}$. Let $\mathbb{T}_a = f^{-1}(-\infty, a]$ denote the sublevel set for the function value a. Certianly, we have inclusions:

$$\mathbb{T}_a \subseteq \mathbb{T}_b \text{ for } a \leq b.$$

This inclusion induces a map in the homology groups. So, if $\iota : \mathbb{T}_a \to \mathbb{T}_b$ denotes the inclusion map $x \mapsto x$, we have an induced map

$$f = \iota_* : \mathsf{H}_p(\mathbb{T}_a) \to \mathsf{H}_p(\mathbb{T}_b).$$

Now consider a sequence of distinct values $a_1 < a_2 < \ldots, < a_n$ corresponding to which we have the sequence of homomorphisms induced by inclusions

$$0 \to \mathsf{H}_p(\mathbb{T}_{a_1}) \to \mathsf{H}_p(\mathbb{T}_{a_2}) \to \cdots \to \mathsf{H}_p(\mathbb{T}_{a_n}) \to \mathsf{H}_p(\mathbb{T})$$

This sequence of maps commutes because inclusion satisfies the transitive relation, $\mathbb{T}_{a_1} \subseteq \mathbb{T}_{a_2} \subseteq \mathbb{T}_{a_3}$ implies $\mathbb{T}_{a_1} \subseteq \mathbb{T}_{a_3}$. So, we have a homomorphism

$$f_p^{ij} : \mathsf{H}_p(\mathbb{T}_{a_i}) \to \mathsf{H}_p(\mathbb{T}_{a_j})$$

for all p and $1 \leq i \leq j \leq n$. The homomorphism f_p^{ij} takes the homology classes of the sublevel set $\mathbb{T}_{a_i}$ to those of the sublevel sets of $\mathbb{T}_{a_j}$. Some of these classes may die or get merged with other classes while the others survive. The image $\mathrm{Im}\, f_p^{ij}$ contains this information.

Definition 38 (Persistence.)**.** The p-th persistent homology groups are the images of the homomorphisms; $\mathsf{H}_p^{ij} = \mathrm{im}\, f_p^{ij}$, for $1 \leq i \leq j \leq n$. The p-th persistent betti numbers are the ranks $\beta_p^{ij} = \text{rank } \mathsf{H}_p^{ij}$.

The p-th persistent homology groups contain an important information, namely when a class is born and when it dies. The issue of birth and death of a class becomes more subtle because when a new class is born, many other classes that are sum of this new class and any other existing class also are born. Similarly, when a class ceases to exist, many other classes also do so along with it. Therefore, we need a mechanism to pair births and deaths canonically. We will do it below in the discrete case.

Filtrations. Consider a simplicial complex $\mathcal{K}$ and a function $f : \mathcal{K} \to \mathbb{R}$ on it. We require that the function f is *monotonic* which means it satisfies the property: for every $\sigma' \subseteq \sigma$, we have $f(\sigma') \leq f(\sigma)$. This property ensures that the sublevel sets $f^{-1}(-\infty, a]$ are subcomplexes of $\mathcal{K}$ for every $a \in \mathbb{R}$. Denoting $\mathcal{K}_i = f^{-1}(-\infty, a_i]$, we get a nested sequence of subcomplexes of $\mathcal{K}$ which is called a *filtration*:

$$\emptyset = \mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \cdots \subseteq \mathcal{K}_n = \mathcal{K}.$$

Naturally, this filtraton gives rise to a sequence of homomorphisms induced by inclusions

$$0 \to \mathsf{H}_p(\mathcal{K}_1) \to \cdots \to \mathsf{H}_p(\mathcal{K}_n) = \mathsf{H}_p(\mathcal{K}).$$

By definition, the p-th persistent homology groups consist of classes that survive from $\mathcal{K}_i$ to $\mathcal{K}_j$, that is, the classes which do not get 'quotient out' by the boundaries in $\mathcal{K}_j$. We can write this with notation, $\mathsf{H}_p^{ij} = Z_p(\mathcal{K}_i)/(B_p(K_j) \cap Z_p(K_i))$. We now formally state when a class is born or dies.

Definition 39 (Birth and death.)**.** A p-th homology class $[c]$ is born at $\mathcal{K}_i$ if $[c] \in \mathsf{H}_p(\mathcal{K}_i)$, but $[c] \notin \mathsf{H}_p(\mathcal{K}_{i-1})$. It dies entering $\mathcal{K}_j$ if it merges with a class that is born earlier. Formally stated, $f^{i,j-1}([c]) \notin \mathsf{H}_p^{i-1,j-1}$, but $f^{i,j}([c]) \in \mathsf{H}_p^{i-1,j}$.

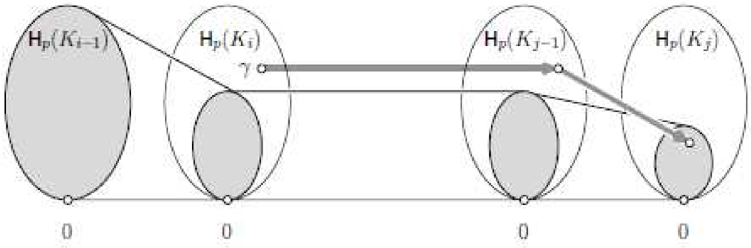


Figure 13: Birth and death of classes, taken from [1]: The class γ is born at $\mathcal{K}_i$ since it is not in the image of $\mathsf{H}_p(\mathcal{K}_{i-1})$. It dies entering $\mathcal{K}_j$ since this is the first time its image merges with the image of $\mathsf{H}_p(\mathcal{K}_{i-1})$.

The definition of birth is almost straightforward. But, the definition of death is a little subtle. When a class is merged with another class, we *choose* to kill the class that is born the *latest*. If we made a different choice, the lifetime of a class would be different. But, we make this canonical choice of killing the *yougest* class when merging happens. We define the persistence of a class, $\text{Pers}([c]) = a_j - a_i$ where the class $[c]$ is born at the function value a_i and dies at the function value a_j. Sometimes, emphasizing on the index we take its *index persistence* as $j - i$.

15 Algorithm

We describe the persistence algorithm originally proposed in [2]. For simplicity we assume that we add one simplex at a time into a filtartion. This means, given a filtration

$$\mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \mathcal{K}_2 \subseteq \cdots \subseteq \mathcal{K}_n = \mathcal{K}$$

where $\mathcal{K}_{i+1} \setminus \mathcal{K}_i = \sigma$, a single simplex.

Fact 7. *When a p-simplex $\sigma = \mathcal{K}_{i+1} \setminus \mathcal{K}_i$ is added, exactly one of the following two possibilities occurs:*

1. *A non-boundary p-cycle c along with its classes $[c]+h$ for any class $h \in \mathsf{H}_p(\mathcal{K}_i)$ are created. In this case we call σ a* positive *simplex.*
2. *An existing $(p-1)$-cycle c along with its classes $[c] + h$ for any class $h \in \mathsf{H}_p(\mathcal{K}_i)$ are destroyed (killed). In this case we call σ a* negative *simplex.*

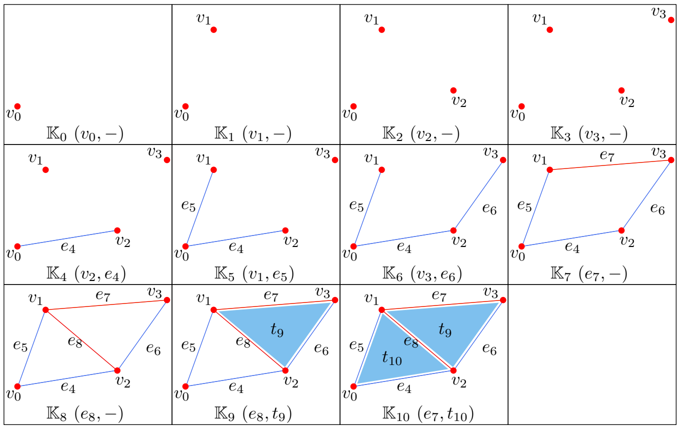

v_0
$\mathbb{K}_0$ $(v_0, -)$
v_1
$\mathbb{K}_1$ $(v_1, -)$
v_2
$\mathbb{K}_2$ $(v_2, -)$
v_3
$\mathbb{K}_3$ $(v_3, -)$
e_4
$\mathbb{K}_4$ (v_2, e_4)
e_5
$\mathbb{K}_5$ (v_1, e_5)
e_6
$\mathbb{K}_6$ (v_3, e_6)
e_7
$\mathbb{K}_7$ $(e_7, -)$
e_8
$\mathbb{K}_8$ $(e_8, -)$
t_9
$\mathbb{K}_9$ (e_8, t_9)
t_{10}
$\mathbb{K}_{10}$ (e_7, t_{10})

Figure 14: Red simplices are positive and blue ones are negative. The simplices are indexed to coincide with their order in the filtration. $(\cdot, \cdot)$ in each subcomplex shows the pairing between the positive and the negative. The second component missing in the parenthesis shows the introducing of a positive simplex.

We elaborate the above two changes through an example depicted in Figure 14. When one moves from $\mathcal{K}_6$ to $\mathcal{K}_7$, a non-boundary loop which is a 1-cycle $(e_4 + e_5 + e_6 + e_7)$ is created after adding edge e_7. Strictly speaking, a positive p-simplex may create more than one p-cycle. Only one of them is independent and the others are its linear combinations with the existing ones in $\mathcal{K}_{i-1}$. From $\mathcal{K}_7$ to $\mathcal{K}_8$, the introduction of edge e_8 creates two non-boundary loops $(e_4 + e_5 + e_8)$ and $(e_6 + e_7 + e_8)$. But any one of them is the linear combination of the other one with the existing loop $(e_4 + e_5 + e_6 + e_7)$. Notice that there is no canonical way to choose an independent one. However, the creation of a loop is reflected in the increase of the rank of H_1. In other words, in general, the Betti number β_p increases by 1 for a positive simplex. For a negative simplex, we get the opposite effect. In this case β_{p-1} decreases by 1 signifying a death of a cycle. However, unlike positive simplices, the killed cycle is determined uniquely up to homology, which is the equivalent class carried by the boundary of σ_i. For example, in Figure 14, the loop $(e_6 + e_7 + e_8)$ gets killed by triangle t_9 when we go from $\mathcal{K}_8$ to $\mathcal{K}_9$.

Pairing. We already saw that killing of a class is uniquely paired with a creation of a class through the 'youngest first' rule. This means that each negative simplex is paired uniquely with a positive simplex. The goal of the persistence algorithm is to find out these pairs.

Consider the birth and death of the classes by addition of simplices into a filtration. When a p-simplex σ is added, we explore if it kills the class $[c]$ of its boundary $c = \partial\sigma$. The cycle c was created when the youngest $(p-1)$-simplex in it, say d, was added. If d, a positive $(p-1)$-simplex, has already been paired with a p-simplex σ', then a class also created by d got merged when σ' appeared. We can get the $(p-1)$-cycle representing this merged class and add it to $\partial\sigma$. The addition provides a cycle that represents a class which was merged to create $\partial\sigma$, but existed before σ'. We update c to be this new cycle and look for the youngest $(p-1)$-simplex d in c and continue the process till we find one that is unpaired, or the cycle c becomes empty. In the latter case, we discover that $c = \partial\sigma$ was a boundary cycle already and thus σ creates a new class in $\mathsf{H}_p(\mathcal{K}_{i+1})$. In the other case, we discover that σ is a negative p-simplex which merges classes among which the youngest one created by d is chosen to be killed by σ. We pair σ with d.

Algorithm 1 Pair(σ)

```
c = ∂_p σ
d is the youngest positive (p − 1)-simplex in c.
while d is paired and c is not empty do
    Let c′ be the cycle killed by the simplex paired with d
    c = c′ + c  \*this addition may cancel simplices*\
    Update d to be the youngest positive (p − 1)-simplex in c
end while
if c is not empty then
    σ is a negative p-simplex and paired with d
else
    σ is a positive p-simplex
end if
```

Let us again consider the example in Figure 14 and see how the algorithm Pair works. From $\mathcal{K}_6$ to $\mathcal{K}_7$, e_7 is added. Its boundary is $c = (v_1 + v_3)$. The vertex v_3 is the youngest positive vertex in c but it is paired with e_6 in $\mathcal{K}_6$. Thus, c is updated to $(v_2+v_3+v_3+v_1) = (v_2+v_1)$. The vertex v_2 becomes the youngest positive one but it is paired with e_4. So, c is updated to (v_0+v_1). The vertex v_1 becomes the youngest positive one but it is paired with e_5. So, c is updated to empty. Hence e_7 is a positive edge. Now we examine the addition of the triangle t_{10} from $\mathcal{K}_9$ to $\mathcal{K}_{10}$. The boundary of t_{10} is $c = (e_4 + e_5 + e_8)$. The youngest positive edge e_8 is paired with t_9. Thus, c is updated by adding the cycle killed by t_9 to $(e_4 + e_5 + e_6 + e_7)$. Since e_7 is the youngest positive edge that is not yet paired, t_{10} finds e_7 as its paired positive edge. Observe that, we finally obtain a loop that is killed by adding the negative triangle. For example, we obtain the loop $(e_4 + e_5 + e_6 + e_7)$ by adding t_{10}.

16 Persistence diagram

A visual representation of the the persistent homology can be created by drawing a collection of points in the plane. Consider the extended plane $(\mathbb{R} \cup \{\pm\infty\})^2$ on which we represent a birth paired with the death as a point with two coordinates. Some of the classes may never die and thus represented as points at infinity. Some others may have same coordinates because they may be born and die at the same time. This happens only when we allow mutiple homology classes being created or destroyed at the same function value or filtration point.

Let μ_p^{ij} be the number of independent p-dimensional classes that are born at $\mathcal{K}_i$ and die entering $\mathcal{K}_j$.

Fact 8. *$\mu_p^{ij} = (\beta^{i,j-1} - \beta_p^{i,j}) - (\beta_p^{i-1,j-1} - \beta_p^{i-1,j})$ for all $i < j$ and all p.*

The first difference on the RHS counts the number of independent classes that are born at or before $\mathcal{K}_i$ and die entering $\mathcal{K}_j$. The second diffrence counts the number of independent classes that are born at or before $\mathcal{K}_{i-1}$ and die entering $\mathcal{K}_j$. The difference between the two differences thus counts the number of independent classes that are born at $\mathcal{K}_i$ and die entering $\mathcal{K}_j$.

Definition 40 (Persistence diagram.)**.** The persistence diagram $\mathrm{Dgm}_p(f)$ of a filtration induced by a function f is obtained by drawing a point (i, j) with multiplicity μ_p^{ij} on the extended plane where the diagonal D is added with infinite multiplicity.

The addition of the diagonal is a technical necessity for results that we will see afterward.

Fact 9.

1. *If a class has persistence s, then the point representing it will be at distance s from the diagonal D*
2. *Since all points (i, j) representing a class have $i < j$, they lie above the diagonal.*
3. *$\beta_p^{k,\ell}$ is the number of points in the upper left quadrant of the corner (k, ℓ). A class that is born at $\mathcal{K}_i$ and dies enetering $\mathcal{K}_j$ is counted for $\beta_p^{k,\ell}$ iff $i \leq k$ and $j > \ell$. The quadrant is therefore closed on the right and open on the bottom.*

Theorem 7. *For every pair of indices $0 \leq k \leq \ell \leq n$ and every p, the p-th persistent Betti number is $\beta_p^{k,\ell} = \sum_{i \leq k} \sum_{j > \ell} \mu_p^{i,j}$.*

Stability of persistence diagrams. A persistence diagram $\mathrm{Dgm}_p(f)$, as a set of points in the extended plane $\overline{\mathbb{R}^2}$, summarizes certain topological information of a space in relation to the function f defined on it. However, this is not useful in practice unless we can be certain that a slight change in f does not change this diagram dramatically. In practice f is seldom measured accurately, and if its persistence diagram can be approximated from a slightly perturbed version, it becomes useful. Fortunately, persistence diagrams are stable. To formulate this stability, we need a notion of distances between persistence diagrams.

Let $\mathrm{Dgm}_p(f)$ and $\mathrm{Dgm}_p(g)$ be two persistence diagrams for two monotonic functions f and g defined on a complex $\mathcal{K}$. We want to consider bijections between points from $\mathrm{Dgm}_p(f)$ and $\mathrm{Dgm}_p(g)$. However, they may have different caridinality of off-diagonal points. Recall that persistence diagrams include the points on the diagonal D each with infnite multiplicity. This addition allows us to borrow points from the diagonal when necessary to define the bijections.

Definition 41 (Bottleneck distance.)**.** Let $B = \{b\}$ denote the set of all bijections $b : \mathrm{Dgm}_p(f) \to \mathrm{Dgm}_p(g)$. Consider the distance between two points $x = (x_1, x_2)$ and $y = (y_1, y_2)$ in L_∞-norm $||x - y||_\infty = max\{|x_1 - y_1|, |x_2 - y_2|\}$. The bottleneck distance between the two diagrams is:

$$W_\infty(\mathrm{Dgm}_p(f), \mathrm{Dgm}_p(g)) = \inf_{b \in B} \sup_{x \in \mathrm{Dgm}_p(f)} ||x - b(x)||_\infty.$$

Fact 10. *W_∞ is a metric on the space of persistence diagrams. Clearly, $W_\infty(X, Y) = 0$ iff $X = Y$. Moreover, $W_\infty(X, Y) = W_\infty(Y, X)$ and $W_\infty(X, Y) \leq W_\infty(X, Z) + W_\infty(Z, Y)$.*

The following theorem originally proved in [4] quantifies the notion of the stabilty of the persistence diagram.

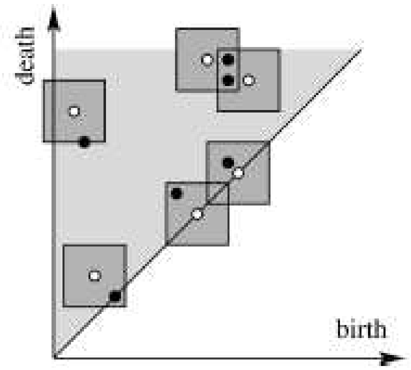


Figure 15: Two persistence diagrams and their bottleneck distance which is half of the side lengths of the squares representing bijections. Figure taken from [1].

Theorem 8 (Stability theorem.)**.** *Let $f, g : \mathcal{K} \to \mathbb{R}$ be two monotonic functions defined on a simplicial complex $\mathcal{K}$. Then, for every $p \geq 0$,*

$$W_\infty(\mathrm{Dgm}_p(f), \mathrm{Dgm}_p(g)) \leq \|f - g\|_\infty.$$

17 Matrix reduction algorithm

There is an algorithm for computing persistent homology that uses only matrix operations. First notice the following:

- The boundary operator $\partial_p : C_p \to C_{p-1}$ can be represented by a matrix D where the columns correspond to the p-simplices and rows correspond to $(p-1)$-simplices.
- It represents the transformation of a basis of C_p given by the set of p-simplices to a basis of C_{p-1} given by the set of $(p-1)$-simplices.
- $$D[i, j] = \begin{cases} 1 & \text{if } \sigma_i \in \partial\sigma_j \\ 0 & \text{otherwise.} \end{cases}$$

Consider a simplicial complex $\mathcal{K}$ and a filtration $\mathcal{K}_0 = \emptyset \subset \mathcal{K}_1 \subset \ldots \subset \mathcal{K}_m = \mathcal{K}$ induced by an ordering of simplices $(\sigma_1, \sigma_2, \ldots, \sigma_m)$ in $\mathcal{K}$. Let D denote the boundary matrix for simplices in $\mathcal{K}$.

Given any matrix M, let $\mathrm{row}_M[i]$ and $\mathrm{col}_M[j]$ denote the ith row and jth column of M, respectively. We abuse the notation slightly to let $\mathrm{col}_M[j]$ denote also the chain $\{\sigma_i \mid M[i, j] = 1\}$, which is the collection of simplices corresponding to 1's in the column $\mathrm{col}_M[j]$.

Definition 42 (Reduced matrix.)**.** Let $\mathrm{low}_M[j]$ denote the row index of the last 1 in the jth column of M, which we call the *low-row index* of the column j. It is undefined for empty columns. The matrix M is *reduced* (or is in *reduced form*) if $\text{low}_M[j] \neq \text{low}_M[j']$ for any $j \neq j'$; that is, no two columns share the same low-row indices.

We define a matrix M to be *upper-triangular* if all of its diagonal elements are 1, and there is no entry $M[i, j] = 1$ with $i > j$.

Proposition 9 ([3])**.** *Let $R = DV$, where R is in reduced form and V is upper triangular. Then, the simplices σ_i and σ_j form a persistent pair if and only if* $\text{low}_R[j] = i$.

Notice that there are possibly many R and V for a fixed D forming the *reduced-form decomposition* as described in the above proposition. The above result implies that the persistent pairing is independent of the particular contents of R and V. Furthermore, consider a column $\text{col}_V[j]$ of V. Let c_j be the p-chain corresponding to this column, that is, $c_j = \text{col}_V[j]$. It follows from the relation $R = DV$ that the jth column of R, $\text{col}_R[j]$, corresponds to the $(p-1)$-chain ∂c_j.

Proposition 10. *Let c_j and c'_j be the p- and $(p-1)$-chains corresponding to the columns* $\text{col}_V[j]$ *and* $\text{col}_R[j]$ *respectively where $R = DV$. Then, $c'_j = \partial_p(c_j)$.*

In light of Proposition 9, we have the following algorithm to compute the persistent pairs of simplices. We process the columns of D from left to right which correspond to the order in which they appear in the filtration. Notice that row indices also follow the same order. Suppose we have processed all columns up to $j-1$ and now are going to process the column j. We check if the row $\text{low}_D[j]$ contains any other lowest 1 for any column j' to the left of j, that is $j' < j$. If so, we add $\text{col}_D[j']$ to $\text{col}_D[j]$. This moves $\text{low}_D[j]$ upward. We continue this process until either we turn all entries in $\text{col}_D[j]$ 0, or settle on $\text{low}_D[j]$ that does not conflict with any other $\text{low}_D[j']$ to its left. In the former case, we declare σ_j a positive simplex. In the latter case, σ_j is a negative p-simplex that pairs with the positive $(p-1)$-simplex $\sigma_{\text{low}_D[j]}$.

References

[1] H. Edelsbrunner and J. Harer. Computational Topology. American Mathematical Society, 2009.

[2] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Discrete Comput. Geom.* **28** (2002), 511–533.

[3] D. Cohen-Steiner, H. Edelsbrunner, and D. Morozov. Vines and vineyards by updating persistence in linear time. *Proc. 22nd Annu. Sympos. Comput. Geom.* (2006), 119–134.

[4] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. *Discrete Comput. Geom.* **37** (2007), 103-120.