

GABRIEL VS. DELAUNAY

Computing the differences between the
persistent homologies

Idoia Gamiz Ugarte

May 2020



Bachelor Project in Mathematics

Contents

1	Introduction	1
2	COMPLEXES	3
2.1	Simplicial complexes	3
2.2	Abstract simplicial complex	4
3	<i>Čech</i> complexes	5
4	Vietoris-rips complexes	5
5	Delaunay complex	6
5.1	Voronoi diagram	6
5.2	Delaunay triangulation	7
6	Homology	7
6.1	Boundary matrices	8
7	Persistent homology	9
7.1	Filtration	10
7.2	Persistence barcode and persistence diagrams	11
8	Subcomplexes of the Delaunay triangulation	12
8.1	Alpha Complexes	13
8.2	Grabriel 1-simplicies in Del , Del_1	13
8.2.1	$Del_1 \subseteq Del$	14
8.2.2	$DR_t = Del \cap VR_t$ and $D_1R_t = Del_1 \cap VR_t$	14
9	\mathbb{R}^2: Are persistent 0-homology of DR_t and D_1R_t isomorphic?	15
9.1	Euclidean Minimum Spanning Tree	15
9.2	Persistent homology of the Delaunay complex	16
9.3	Python experiment and examples	17

10 \mathbb{R}^3: What about persistent 1-homology of $D\check{C}_t$ and $D_2\check{C}_t$?	20
10.1 Python experiments and examples	21
11 Clustering	24
11.1 Persistent Homology and clustering	25
11.2 K -means clustering	28
11.3 Advantages of 0-persistent homology over K -means	28
11.4 Hierarchical clustering dendograms	28
11.5 Hierarchical clustering and persistent homology	29
12 Conclusion	29

1 Introduction

Persistent homology is a very useful tool in *Topological Data Analysis*. In the last years, the Delaunay triangulation has been used to get the persistence diagram, but what if we could find a more efficient manner of getting the same persistent homology?

In this project we will explain the existence of two sub-complexes of the Delaunay triangulation that can help us answering this problem: the alpha complexes and the Gabriel complexes.

The alpha complex is the most widely used method to compute persistent homology for large low-dimensional data sets. It is more common than the Gabriel simplices and it has been studied for more time. In fact, we use it a lot because it works.

Gabriel complex has some problems, but we want to discover in which way can give us information that we can use, even if it does not totally work. That is why we are specially interested in this type of complex, a field to investigate that could give us great benefits. On the one hand, an isomorphism between the 0-persistent homology of both complexes will be proved on the plane. On the other hand, we will see how in three dimensions, despite of not getting the same 1-persistent homology, we get some information for this homologies.

Suppose that S is a set of points in the plane and $Del(S)$ the corresponding Delaunay triangulation. Let K be the simplicial complex consisting on those points and the edges and triangles of $Del(S)$. Let $K1$ be another simplicial complex, which consists just on those simplices with Gabriel edges. By an isomorphism we will see how both of this simplicial complexes end up giving us the same 0-persistent homology.

Furthermore, for the proof of that isomorphism the already known minimum spanning tree will be introduced. This concept works for the 0-persistent homology, but it does not tell anything about the 1-persistent homology. We want to compare the Gabriel simplices with the Delaunay triangulation to find out how much can give us.

The Delaunay triangulation was discovered by the Russian mathematician Borís Deloné in 1934. It is the triangulation that maximizes the minimum interior angle of all the triangles and that is the main reason for its applications in terrain modeling. This technique consists in triangulating points in real three dimensional terrains and then elevating the triangulation. To get an accurate solution the triangles will better approximate the terrain when the length of its edges is shorter and the angles are greater.

Furthermore, the Delaunay triangulation is used to calculate the persistent homology in an efficient way. Persistent homology has two main applications that we are going to explain in this paper: clustering and recognizing noise from data. For example, it was used to identify a subgroup of breast cancer as explained in [1] .

Other concepts that will be explained later in this paper have also important applications. Voronoi diagrams were used by the physician John Snow to determine that a particular pump was responsible for most of the deaths caused by the London cholera of 1854. Alpha shapes were conceived in 1981 as an attempt to define the shape of a finite set of point in the plane. Since

then, it has been used in diverse areas such as pattern recognition, digital shape sampling and processing, and structural molecular biology.

Finally, the Gabriel Graph was used for analysis of geographic variation of data such as the cubic root of the body weight of female red-winged blackbirds in North-America. Even so, it is a field that has not been very investigated and we will study its application in respect of persistent homology.

The remainder of this paper is structured as follows. Section 2 will be just an introduction of some basic concepts so that we can understand definitions and main concepts, directly taken from the book [2]. Section 3, also from [2], will be the explanation of a particular complex, the Čech complex. A similar and more useful complex for our approach denoted by Vietoris-Rips complex will be introduced in Section 4. From that Section, we will start with the real development of the project. There is still information from [2], corroborated and complemented with different sources mentioned along the sections. In Section 5 we introduce the Delaunay complex. Section 6, which contains also definitions taken from [2], is an introduction of concepts for Section 7, which explains the most important notion of this paper, the Persistent Homology. Section 8 introduce the two subcomplexes of the Delaunay complex mentioned before: the Alpha complex and the Gabriel complex. The two experiments will be developed in Sections 9 and 10 and to sum up, we will show the benefits of the persistent homology in terms of clustering in Section 11. Section 12 is just a summary of the conclusions drawn from this paper.

2 COMPLEXES

The concepts from this section are taken from Chapter III.1 in book [2]. A topological space can be represented as a decomposition into simple pieces. If these pieces are topologically simple and their common intersections are lower dimensional pieces of the same kind, then we say that this decomposition is a complex.

2.1 Simplicial complexes

Roughly speaking a simplicial complex is a set of points, edges, triangles, tetrahedrons or n -dimensional counterparts. Each element of the simplicial complex is known as a simplex.

Example 2.1. • A 0-simplex is a point (a vertex).

- A 1-simplex is a line segment (an edge).
- A 2-simplex is a triangle.
- A 3-simplex is a tetrahedron.

For a more mathematical definition, we first need to introduce some concepts. Let u_0, u_1, \dots, u_k be points in \mathbb{R}^d .

A point $x = \sum_{i=0}^k \lambda_i u_i$ is an *affine combination* of the u_i if the λ_i sum to 1. The *affine hull* is the set of affine combinations.

An affine combination is a *convex combination* if all λ_i are non-negative. The set of convex combinations is the *convex hull*.

The $k + 1$ points are *affinely independent* if and only if the k vectors $u_i - u_0$, for $1 \leq i \leq k$, are linearly independent.

Definition 2.2. A k -simplex is the convex hull of $k + 1$ affinely independent points, $\sigma = \text{conv}\{u_0, u_1, \dots, u_k\}$. Those $k + 1$ points are the vertices.

Any subset of affinely independent points is again affinely independent and therefore also defines a simplex.

Definition 2.3. A face of σ , τ , is the convex hull of a non-empty subset of the u_i . We write $\tau \leq \sigma$. If the subset is not the entire set τ is a proper face. $\tau < \sigma$.

A set of size $k + 1$ has 2^{k+1} subsets including the empty set that is not a face, so a simplex σ has exactly $2^{k+1} - 1$ faces, all of which are proper except for σ itself.

Definition 2.4. The boundary of σ , denoted as $\text{bd}\sigma$, is the union of all proper faces, and the interior is everything else, $\text{int}\sigma = \sigma - \text{bd}\sigma$.

Example 2.5. – A 0-simplex has $2 - 1 = 1$ face, the point itself. A point does not have boundary.

- A 1-simplex has $2^{1+1} - 1 = 3$ faces, the edge itself and the vertices that are the proper faces. The boundary of an edge is the union of the vertices.
- A 2-simplex has $2^{2+1} - 1 = 7$ faces, the triangle itself and the three edges and three vertices that are the proper faces. The boundary of a triangle is the union of the edges and the vertices.

Definition 2.6. *A simplicial complex is a finite collection of simplices K such that:*

- *every face from a simplex in K is also in K*
- *the non-empty intersection of any two simplices in K is a face of both*

The dimension of K is the maximum dimension of any of its simplices.

2.2 Abstract simplicial complex

Sometimes we are interested in finding a geometric realization and in constructing a complex in order to put it into the Euclidean space. In that case, it is easier to construct a complex abstractly.

Definition 2.7. *An abstract simplicial complex is a finite collection of sets A such that $\alpha \in A$ and $\beta \subseteq \alpha$ implies $\beta \in A$.*

The sets in A are its simplices and the dimension of A is the maximum dimension of any of its simplices. In this case the dimension of a simplex is $\dim \alpha = \text{card} \alpha - 1$.

Definition 2.8. *A subcomplex is an abstract simplicial complex $B \subseteq A$.*

Theorem 2.9. *(Geometric Realization Theorem) Every abstract simplicial complex of dimension d has a geometric realization in \mathbb{R}^{2d-1} .*

Later in this text when defining some of the complexes we will use the concept of nerves.

Definition 2.10. *(From Chapter III.2 of [2]) Let F be a finite collection of sets in \mathbb{R}^d . Without assuming the sets are convex, we define the nerve to consist of all non-empty subcollections whose sets have a non-empty common intersection,*

$$\text{Nrv} F = \{X \subseteq F \mid \bigcap X \neq \emptyset\}.$$

The nerve is an abstract simplicial complex. It is straightforward that if $\bigcap X \neq \emptyset$ and $Y \subseteq X$, then $\bigcap Y \neq \emptyset$.

3 Čech complexes

This section has information from Chapter III.2 of [2] and from [3]. We now consider the special case in which the convex sets are closed geometric balls, all of the same radius, r .

Let S be a finite set of points in \mathbb{R}^d and write $B_x(r) = x + \mathbb{B}^d$ for the closed ball with center x and radius r .

Given $r > 0$, the Čech complex of S and r , $\check{C}_r(S)$, is an abstract simplicial complex that is isomorphic to the nerve of this collection of balls. So by definition of the nerve we have:

$$\check{C}ech = \{\sigma \subseteq S \mid \bigcap_{x \in \sigma} B_x(r) \neq \emptyset\}.$$

That is, for each subset σ of points in S , a closed ball $B_x(r)$ is formed around each point x in σ , and consider σ as a simplex of $\check{C}ech(r)$ if there is a common point contained in the intersection of the balls for all x in σ . It does not necessarily have a geometric realization in \mathbb{R}^d .

The mini ball algorithm, explained in [2], is a fast algorithm to recognize among sets of two points the ones that form simplices in the Čech complex.

In this project instead of in these complexes, we are more interested in an other type of complexes that are easier to compute: The Vietoris-rips complexes. We will use the intersection of this complex with the Delaunay complex along a filtration when doing our experiments.

4 Vietoris-rips complexes

The contents of this section are also taken from Chapter III.2 of [2] and from [3]. For checking if the subset σ is a Čech simplex we verify if there is a common point contained in the intersection of the ball of radius r around x for all x in σ . But if we just check pairs and we see that all balls have pairwise intersection, then σ is a Vietoris-rips simplex.

This can be translate as complex consisting of all subsets of diameter at most $2r$:

$$Vietoris - Rips(r) = \{\sigma \subseteq S \mid diam\sigma \leq 2r\}$$

The edges of the $Vietoris-Rips(r)$ complex are the same as in the Čech complex, but the 2-simplices of higher dimensional simplices are not necessarily the same. Furthermore, $\check{C}ech(r) \subseteq Vietoris-Rips(r)$. In fact, if all the balls have a common point contained in the intersection, they also have pairwise intersection. If we increase the radius by a multiplicative constant we have the opposite implication.

Lemma 4.1. (*Vietoris-rips lemma*). *Let S be a finite set of points in some Euclidean space and $r \geq 0$, then $Vietoris-Rips(r) \subseteq \check{C}ech(\sqrt{2}r)$.*

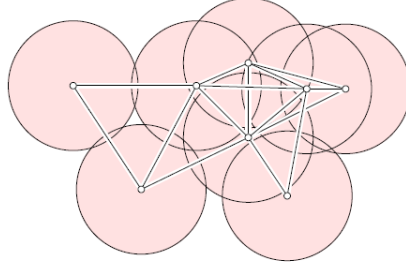


Figure 1: Nine points with pairwise intersections among the disks indicated by straight edges connecting their centers. Nine of the ten possible triangles and the two tetrahedrons belong to the Čech complex. The only difference is the tenth triangle that belongs only to the Vietoris-Rips. From [2].

5 Delaunay complex

The definitions for this Section was taken from Chapter III.3 in the book [2]. In order to construct the notion of a Delaunay complex, we first need to introduce the concept of Voronoi diagram.

5.1 Voronoi diagram

The Voronoi diagram is a particular simplicial complex from a finite set $S \subseteq \mathbb{R}^d$.

Definition 5.1. *Let S be a finite set in \mathbb{R}^d . The Voronoi cell of a point u in S is the set of point for which u is closest,*

$$V_u = \{x \in \mathbb{R}^d \mid \|x - u\| \leq \|x - v\|, v \in S\}.$$

The Voronoi diagram of S is the collection of Voronoi cells of its points.

V_u is a convex polyhedron in \mathbb{R}^d and the union of all this Voronoi cells is the entire space. The intersection of two of this cells is at most a common piece of their boundary and if we join those two vertices we get what we know as the Delaunay triangulation.

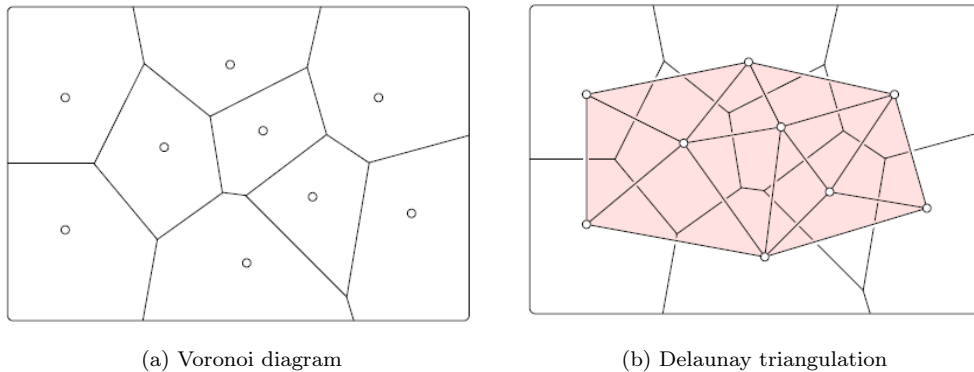


Figure 2: From [2].

5.2 Delaunay triangulation

The Delaunay triangulation partitions the convex hull into d -dimensional simplices.

Let S be a set of points in \mathbb{R}^d , σ be a k -simplex ($0 \leq k \leq d$) whose vertices are in S . A circumsphere of σ is a sphere that passes through all vertices of σ . If $k = d$, σ has a unique circumsphere, otherwise, there are infinitely many circumspheres of σ .

We say that σ is Delaunay if there exists a circumsphere of σ such that no vertex of V lies inside it. A Delaunay triangulation of S is a simplicial complex such that all simplices are Delaunay, and the underlying space of $Del(S)$ is the convex hull of S . The dimension of any simplex in the Delaunay complex is at most d .

Definition 5.2. *The Delaunay complex of a finite set $S \subseteq \mathbb{R}^d$ is isomorphic to the nerve of the voronoi diagram,*

$$Delaunay = \{\sigma \subseteq S \mid \bigcap_{u \in \sigma} V_u \neq \emptyset\}.$$

Definition 5.3. *We say the set S is in general position if no $d+2$ of the points lie on a common $(d-1)$ -sphere. This assumption implies that no $d+2$ Voronoi cells have a non-empty common intersection.*

Otherwise, we say that S contains degeneracies, that is, there are $d+2$ points in S lie on a common sphere. Degeneracies can be removed by applying an arbitrary small perturbation onto the coordinates of points in S . The Delaunay triangulation of S is unique if S is in general position.

In a delaunay triangulation of \mathbb{R}^2 the simplicial complex, K , consists on points, edges and triangles. We partition the convex hull into 2-dimensional simplices. Whereas in \mathbb{R}^3 , since we divide it into 3-dimensional simplices, K consists on points, edges, triangles and tetrahedrons. This is also called Delaunay tetrahedralization.

With all the previous definitions we are ready to introduce the main concept of the paper: the persistent homology. First, let give a general idea of basic concepts for homology.

6 Homology

The material and definitions for this Section are taken from Chapter IV in [2]. When we talk about homology in one way we are talking about how a space is connected.

Let K be a simplicial complex and p a dimension.

Definition 6.1. *A p -chain is a formal sum of p -simplices in K , $c = \sum a_i \sigma_i$, where the σ_i are the p -simplices and the a_i are the coefficients.*

We are interested in *modulo 2 coefficients*, the ones that are either 0 or 1. We denote as $(C_p, +)$ the group of p -chains with the addition operation. To relate this groups, first we need to define the boundary of a p -simplex.

Definition 6.2. The boundary of a p -simplex is the sum of its $(p - 1)$ -dimensional faces.

If we write a simplex as $\sigma = [u_0, u_1, \dots, u_p]$, where u_0, u_1, \dots, u_p are the vertices that form the simplex, then we have that

$$\partial_p \sigma = \sum_{j=0}^p [u_0, u_1, \dots, \hat{u}_j, \dots, u_p],$$

and the hat indicates that u_j is omitted.

There is a similar definition for a p -chain.

Definition 6.3. The boundary of a p -chain is the sum of the boundaries of its simplices,

$$\partial_p c = \sum a_i \partial_p \sigma_i,$$

From this definition we have that a boundary maps a p -chain to a $p - 1$ chain, so we write $\partial_p : C_p \rightarrow C_{p-1}$. Since is a map between groups that commutes with the group operation this is known as the *boundary homomorphism*. If we apply this map recursively with different dimensions we get a *chain complex*, that is, a sequence of chain groups connected by boundary homomorphisms:

$$\dots \xrightarrow{\partial_{p+2}} C_{p+1} \xrightarrow{\partial_{p+1}} C_p \xrightarrow{\partial_p} C_{p-1} \xrightarrow{\partial_{p-1}} \dots$$

We distinguish between two different types of p -chains, the p -cycles and the p -boundaries.

Definition 6.4. A p -cycle is a p -chain with empty boundary $\partial c = 0$.

$Z_p = Z_p(K)$ is the group of p -cycles. This group is the kernel of the p -th boundary homomorphism, $Z_p = \ker \partial_p$.

Definition 6.5. A p -boundary is a p -chain that is the boundary of a $(p + 1)$ -chain, $c = \partial d$ where $d \in C_{p+1}$.

$B_p = B_p(K)$ is the group of p -boundaries. This group is the image of the $(p + 1)$ -th boundary homomorphism, $B_p = \text{im} \partial_{p+1}$.

Every p -boundary is also a p -cycle, so B_p is a subgroup of Z_p and both of them are subgroups of C_p . Then, $B_p \subseteq Z_p \subseteq C_p$. Since $B_p \subseteq Z_p$ we can partition each cycle group into classes of cycles that have different boundaries.

Definition 6.6. The p -th homology group is the p -th cycle group modulo the p -th boundary group, $H_p = Z_p / B_p$. The p -th Betti number is the rank of this group, $\beta_p = \text{rank} H_p$

6.1 Boundary matrices

Our aim is to calculate the persistent homology of the the Delaunay triangulation and the one of its subcomplexes. To do that, we need to explain the concept of boundary matrix.

Let K be a simplicial complex. We represent the $(p - 1)$ -simplices as rows and the p -simplices as columns of the p -th boundary matrix.

In our experiment, we are going to use another version of this, but in general when we talk about the p -th boundary matrix of a simplicial complex we refer to this. In fact, when working in two dimensions with the Delaunay triangulation we are interested both in the boundary of the triangles and the edges. Using the notion of p -th boundary matrix we should need two matrix: a 1-boundary matrix to get the boundary of the edges and a 2-boundary matrix to get the boundary of the triangles. We will do this in a unique matrix, see Section 9.3.

Let n_{p-1} and n_p be the number of $p-1$ - and p -simplices respectively. Assuming an arbitrary but fixed ordering of the simplices, for each dimension, this matrix is $\partial_p = [a_i^j]$, where $i=1, \dots, n_{p-1}$ and $j=1, \dots, n_p$.

$$a_i^j = \begin{cases} 1 & , \text{ if the } i\text{-th } (p-1)\text{-simplex is a face of the } j\text{-th } p\text{-simplex} \\ 0 & , \text{ otherwise} \end{cases} \quad (1)$$

Let our problem be calculating the boundary of our simplicial complex K and suppose it consists on $0, 1, \dots, n$ -simplices. Then, we can use matrix (1) n times to get the p -th boundary of each simplex. In this matrices column j represent the p -th boundary of the simplex j .

Moreover, our goal is to get a unique matrix from where we can extract the already mentioned and so important concept that we are about to introduce: The Persistent Homology, which is an adaptation of homology to point cloud data.

7 Persistent homology

This Section has a mix of contents from [2], [3], [4] and [5].

Nowadays the concepts of *Big Data* and *Data Analysis* are very involved in our society. In fact, due to all the information that is constantly generated, our system is full of data. Most of it is called noisy data, because it is made up of the true signal and the noise, which is a large amount of additional meaningless information.

We are interested in overcoming that noise, but determine if something is noise or not is subjective. Anyway, it will be always delimited by a range of scales so that everything that is smaller or larger is ignored.

In this paper we are focused in the *topological data analysis*, a strategy of data analysis that uses techniques from topology. This allows us to analyze the data in such a way that is insensitive to the particular metric chosen. Furthermore, it provides dimensionality reduction and robustness to noise. It reveals the global structure of a data-set and deals with high-dimensional data-set without using dimensionality reduction.

We need to study the "shape" of the data. There are two ingredients, one geometric, defining a function on a topological space, and the other algebraic, turning the function into measurements. In this way it combines the differentiating of geometry with the classification power of topology.

Persistent homology studies the evolution of k -dimensional holes along a filtration of simplicial complexes. In search of the true signal it plays a big role. In fact, if the same k -dimensional hole

is detected along a large number of subsets of the filtration, then it probably represents a true feature rather than noise.

It detects true signal by varying a parameter instead of finding an optimal parameter value that yields the most accurate representation of the signal. This technique has shown to be effective in various datasets.

7.1 Filtration

This section is taken from Chapter VII.1 in book [2]. Filtrations are generally based on distance between points. We are free to determine what we take as distance and filtration values. In our experiment we will define a Vietoris-Rips filtration function.

Consider a simplicial complex K and let the function $f : K \rightarrow \mathbb{R}$ be monotonic. This means that it is non-decreasing along increasing chains of faces, that is, $f(\sigma) \leq f(\tau)$ whenever σ is a face of τ .

Monotonicity implies that the sublevel set, $K(a) = f^{-1}(-\infty, a]$, is a subcomplex of K for every $a \in \mathbb{R}$. Letting m be the number of simplices in K , we get $n + 1 \leq m + 1$ different subcomplexes, that are organized as an increasing sequence,

$$\emptyset = K_0 \subseteq K_1 \subseteq \dots \subseteq K_n = K.$$

In other words, if $a_1 < a_2 < \dots < a_n$ are the function values of the simplices in K and if $a_0 = -\infty$ then $K_i = K(a_i)$ for each i .

We denote this sequence of complexes a filtration of f and it consists on a construction by adding pieces of simplices at a time. In persistence homology, more than in the sequence of complexes, we are interested in the topological evolution and this is expressed by the corresponding sequence of homology groups.

For every $i \leq j$ we have an inclusion map from the underlying space of K_i to that of K_j and an induced homomorphism for each dimension p , $f_p^{i,j} : H_p(K_i) \rightarrow H_p(K_j)$. The filtration is a sequence of homology groups connected by homomorphisms,

$$0 = H_p(K_0) \rightarrow H_p(K_1) \rightarrow \dots \rightarrow H_p(K_n) = H_p(K)$$

. When we go from K_{i-1} to K_i , we gain new homology classes and we lose some because they become trivial or merge with each other.

The p -th homology group represents which p -th ordered topological features continue alive. We keep the classes that are born at or before a given threshold and die after another threshold in those groups.

Definition 7.1. *The p -th persistent homology groups are the images of the homomorphisms induced by inclusion, $H_p^{i,j} = \text{im} f_p^{i,j}$, for $0 \leq i \leq j \leq n$. The corresponding p -th persistent Betti numbers are the ranks of these groups, $\beta_p^{i,j} = \text{rank} H_p^{i,j}$.*

The persistent homology groups are the homology classes of K_i that continue alive at K_j , $H_p^{i,j} = Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$.

Theorem 7.2. (*First Group Isomorphism Theorem*). If $\varphi : G \rightarrow H$ is a group homomorphism, then $\ker(\varphi)$ is a normal subgroup of G and $G/\ker(\varphi)$ is isomorphic to the image of φ .

In fact, $H_p^{i,j} = \text{im } f_p^{i,j} = \text{im}(Z_p(K_i) \xrightarrow{g} Z_p(K_i)/B_p(K_j))$ and g is an homomorphism, so by Theorem 7.2 $Z_p(K_i)/\ker g$ is isomorphic to $H_p^{i,j}$. Since $\ker g = Z_p(K_i) \cap B_p(K_j)$, then $H_p^{i,j}$ is isomorphic to $Z_p(K_i)/(Z_p(K_i) \cap B_p(K_j))$.

$$\begin{array}{ccc} Z_p(K_i)/B_p(K_i) = H_p(K_i) & \xrightarrow{f_p^{i,j}} & H_p(K_j) = Z_p(K_j)/B_p(K_j) \\ & \nearrow g & \\ & Z_p(K_i) & \end{array}$$

Definition 7.3. Let γ be a class in $H_p(K_i)$.

1. γ is born at K_i if it is not in $H_p^{i-1,i}$.
2. if γ is born at K_i , then it dies entering K_j if $f_p^{i,j-1}(\gamma)$ it is not in $H_p^{i-1,j-1}$, but $f_p^{i,j}(\gamma)$ it is in $H_p^{i-1,j}$.

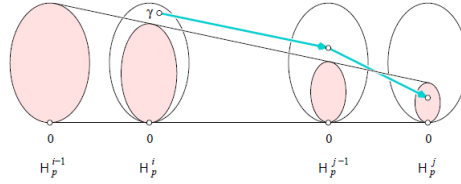


Figure 3: The class γ is born at K_i because it is not in the image of H_p^{i-1} . It dies entering K_j since this is the first time its image merges into the image of H_p^{i-1} . From [2].

ELDER RULE. If γ is born at K_i , a_i , and dies entering K_j , a_j , then the persistence is the difference between this two values: $\text{pers}(\gamma) = a_j - a_i$. If γ is born at K_i but never dies, we take a_j as infinity.

7.2 Persistence barcode and persistence diagrams

This subsection is complemented with some statements from [3]. The persistence diagram is a set of bars (intervals) representing birth and death times of k -dimensional holes along the sequence of simplicial complexes.

Let denote a_i and a_j the birth and death times of a topological feature. Then, the duration of that feature's survival is represented as an interval $[a_i, a_j)$ in the persistence barcode. In order to visualize the persistent Betti numbers, we draw the corresponding points (a_i, a_j) in two dimensions as we will explain below. Some of these points may have infinite coordinates, so we work with the extended real plane, $\overline{\mathbb{R}}^2$ and we denoted by persistence diagram.

These diagrams are very important because they contain all the information about persistent homology groups. Informally, if a k -dimensional hole has a short lifetime it is considered to be topological noise, whereas if it has a long life it is considered to be a topological feature.

The number of classes that are born at or before K_i and die entering K_j is $\beta_p^{i,j-1} - \beta_p^{i,j}$. Whereas the number of classes that are born at or before K_{i-1} and die entering K_j is $\beta_p^{i-1,j-1} - \beta_p^{i-1,j}$.

So we can define $\mu_p^{i,j}$ the number of p -dimensional classes born at K_i and dying entering K_j as

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

for all $i < j$ and all p .

Drawing each point (a_i, a_j) with multiplicity $\mu_p^{i,j}$ we get the p -th persistence diagram of the filtration, denoted as $Dgm_p(f)$. It represents a class by a point whose vertical distance to the diagonal, $a_j - a_i$, is the persistence. Since the multiplicities are defined only for $i < j$, all points lie above the diagonal. We also add the points on the diagonal to the diagram, each with infinite multiplicity.

It is easy to get the persistent Betti numbers from the diagram. In fact, $\beta_p^{k,l}$ is the number of points in the upper, left quadrant with corner point (a_k, a_l) . A class that is born at K_i and dies entering K_j is counted iff $a_i \leq a_k$ and $a_j > a_l$. The quadrant is therefore closed along its vertical right side and open along its horizontal lower side.

Lemma 7.4. (*Fundamental lemma of persistent homology*) *Let*

$$\emptyset = K_0 \subseteq K_1 \subseteq \dots \subseteq K_n = K$$

be a filtration. For every pair of indices $0 \leq k \leq l \leq n$ and every dimension p , the p -th persistent Betti number is

$$\beta_p^{k,l} = \sum_{i \leq k} \sum_{j > l} \mu_p^{i,j}$$

8 Subcomplexes of the Delaunay triangulation

It is clear enough that persistent homology is a very useful tool in *Topological Data Analysis*, not only for clustering, as we will see in Section 11, but also for many applications in problems of great importance.

The Delaunay triangulation has been used to get the persistence diagram and its definition was based on what we call the empty ball property: no point in S is inside the circumsphere of any triangle in $Del(S)$. We will use this to define an alternative for computing the persistent homology.

If there exist a sphere that circumscribes the simplex and does not contain any other point on its bounded side, then the ball bounded by such a sphere is called an empty ball. We say that a simplex is Delaunay if there exist such a circumsphere. When $k = d$, all k -simplices have a unique circumsphere, otherwise, there are infinitely many circumspheres of the simplices.

We can use this property to define sub-complexes of the Delaunay triangulation by imposing more restrictions on the empty balls. Two of those sub-complexes are the Gabriel simplices and the α -shapes.

In this paper we are specially interested on the first ones, a field to investigate that could give us great benefits, but we will first introduce the well known Alpha complexes. The mentioned restrictions are explained in [6].

8.1 Alpha Complexes

The concepts of this subsection are from the survey [7]. The alpha complex is the most widely used method to compute persistent homology for large low-dimensional data sets. It is family of subcomplexes of the Delaunay complex that differ from the *Čech complexes* by having canonical geometric realizations.

Let S be a finite set of points in \mathbb{R}^d and α a non-negative real number. For each $u \in S$, we let $B_u(\alpha) = u + \alpha\mathbb{B}^d$ be the closed ball with center u and radius α . The union of these balls is the set of points at distance at most α from at least one of the points in S . To decompose the union, we intersect each ball with the corresponding Voronoi cell, $R_u(\alpha) = B_u(\alpha) \cap V_u$.

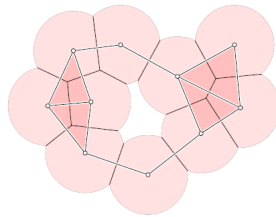


Figure 4: The union of disks decomposed into convex regions and the alpha complex is superimposed. From [2].

Since balls and Voronoi cells are convex, the $R_u(\alpha)$ are also convex. Any two of them are disjoint or overlap along a common piece of their boundaries, and together the $R_u(r)$ cover the entire union. We observe that $R_u(\alpha) \subseteq V_u$, so if a collection of $R_u(\alpha)$ has non-empty common intersection so do the corresponding Voronoi cells. The α -complex is a subcomplex the Delaunay triangulation and isomorphic to the nerve of this cover,

$$Alpha(r) = \{\sigma \subseteq S \mid \bigcap_{u \in \sigma} R_u(r) \neq \emptyset\}.$$

The α -shape is the the union of all simplices in the α -complex.

8.2 Gabriel 1-simplicies in Del , Del_1

The following definitions can be find in book [6]. The Gabriel 1-simplicies are sub-complexes of the Delaunay triangulation as we will proof in the next subsection.

In this case the restriction for the empty balls is explained as follow: a simplex of dimension less than three is called *Gabriel* if its smallest circumsphere is empty. In two dimensions, the restriction is only imposed for the 1-simplices because the circumsphere is not unique, so we need to take the smallest. This means that the triangles of the Delaunay complex are always Gabriel. In three dimensions, the restriction is both for the 1- and 2-simplices. All the tetrahedrons of the Delaunay complex are Gabriel.

The interiors of this smallest circumspheres are also known as Gabriel neighborhoods.

Definition 8.1. (*Gabriel neighborhood*) Let v_1, v_2 be two vertices in K . The Gabriel neighborhood associated with v_1 and v_2 is the interior of the smallest ball touching v_1 and v_2 . The Gabriel neighborhood sphere has radius $d(v_1, v_2)/2$.

The gabriel graph is the geometric graph that contains all Gabriel 1-simplices. Two vertices will be joined if and only if their Gabriel neighborhood is empty. In other words, if we have two vertices v_1 and v_2 , the gabriel graph joins them by an edge if and only if the sphere centered in the middle point of this edge (the Gabriel neighborhood sphere) does not contain any other point of S . That is, for a 1-simplex in \mathbb{R}^2 the middle point is not closer to any other element of S than v_1 and v_2 .

8.2.1 $Del_1 \subseteq Del$

All Gabriel simplices are contained in the Delaunay triangulation.

$$Del_1 \subseteq Del_2 \subseteq \dots \subseteq Del_d = Del$$

Proof. (Similarly explained in Chapter 2 of [8]). Suppose the edge $v_1v_2 \in Del_1$. Then the Gabriel neighborhood of v_1 and v_2 is empty. Since the Gabriel neighborhood associated with v_1 and v_2 is the interior of the smallest ball touching v_1 and v_2 , the sphere with radius $d(v_1, v_2)/2$ is empty. This means that no other point of P is inside this sphere.

If some ball touching two vertices v_1, v_2 is empty, there is also a largest empty ball touching these vertices, which touches one other vertex, so that v_1v_2 is an edge of a 2-simplex that is part of the Delaunay Triangulation. Then v_1v_2 is in the Delaunay Triangulation, and therefore $Del_1 \subseteq Del$. \square

This will be proven in next section when introducing the Euclidean Minimum Spanning Tree by the equivalent contrapositive statement. That concept will be introduced later in this paper.

8.2.2 $DR_t = Del \cap VR_t$ and $D_1R_t = Del_1 \cap VR_t$

In this experiment we define a special type of complex denoted by Delaunay-Rips complex. It consists of the intersection between the Delaunay complex and the Vietoris-Rips complex. Along

the sequence obtained with the filtration we will see how the intersection affects until we finally achieve the whole Delaunay complex.

We are trying to do something new, different from other type of simplicial complex already studied. So we define the simplicial complex K by using the Rips filtration function f as follows:

1. In dimension two, the filtered value of the edge is its length and for the triangle is the largest edge.
2. In dimension three, the filtered value of the edges and the triangles still the same. For the tetrahedrons, it is also its biggest edge.

For D_1R_t , compute exactly the same filtration value and repeat the process getting what we could call Gabriel-Rips complex. In \mathbb{R}^3 , we would also work with an extended version of the Gabriel simplices, but let us leave this explanation for later.

In next section, we will start with the comparison of the persistent-homology obtained from the previous simplicial complexes.

9 \mathbb{R}^2 : Are persistent 0-homology of DR_t and D_1R_t isomorphic?

Let S be a set of points in the plane and $Del(S)$ the corresponding Delaunay triangulation. Let K be the simplicial complex consisting of those points and the edges and triangles of $Del(S)$.

In this section, we will discuss a way of obtaining a different simplicial complex K_1 , which consists on the same points but just some of those edges.. We will see how both of this simplicial complexes end up giving us the same 0-persistent homology.

The answer for this question is that they are isomorphic. In the following subsection the concept of Euclidean minimum Spanning Tree will be introduced as well as its application for proving this fact. All this information was found in [9].

9.1 Euclidean Minimum Spanning Tree

Let S be a set of n points in \mathbb{R}^d . The Euclidean minimum spanning tree or EMST is a minimum spanning tree of S .

The weight of an edge is the Euclidean distance between its vertices. A EMST connect a set of dots using lines in such a way that the total length of all the lines is minimized. So each vertex must be joined to its nearest vertex. Furthermore, any dot can be reached from any other by following the lines. The Euclidean Minimum Spanning Tree need not be unique.

Let M be the euclidean minimum Spanning Tree, then :

$$M \subseteq Del_1 \subseteq Del$$

Proof. (Can be found in [9]). All edges of an EMST are edges of a relative neighborhood graph, which in turn are edges of a Gabriel graph, which are edges in a Delaunay triangulation of the points.

This can be proven by the equivalent contrapositive statement: every edge not in a Delaunay triangulation is also not in any EMST.

The proof is based on two properties of minimum spanning trees and Delaunay triangulations:

1. The cycle property of minimum spanning trees: For any cycle C in the graph, if the weight of an edge e of C is larger than the weights of other edges of C , then this edge cannot belong to a MST .
2. A property of Delaunay triangulations: If there is a circle with two of the input points on its boundary which contains no other input points, the line between those two points is an edge of every Delaunay triangulation.

Consider an edge e between two input points p and q . Let suppose that it is not an edge of a Delaunay triangulation. Then, by Property 2 the circle with e as its diameter must contain some other point r inside. But then r is closer to both p and q than they are to each other. By definition of a Gabriel simplex, e is not in Del_1 . Furthermore, the edge from p to q is the longest edge in the cycle of points $p \rightarrow q \rightarrow r \rightarrow p$, and by property 1 e is not in any $EMST$. \square

It is known that the persistent 0-homology of the EMST is isomorphic to the one of Delaunay, with the previous proof is enough to state that the Delaunay 0-homology will be also isomorphic.

9.2 Persistent homology of the Delaunay complex

Persistence can be computed efficiently by an algorithm based on matrix reduction. This reduction will be applied to a special version of the boundary matrix explained in Section 6. The following procedure is explained in Chapter VII of the book [2].

A single reduction can give all the information. We use our filtration function to get a compatible ordering of the simplices: a sequence $\sigma_1, \sigma_2, \dots, \sigma_m$ such that $f(\sigma_i) < f(\sigma_j)$ implies $i < j$ and so does σ_i being a face of σ_j .

Every subsequence of simplices forms a subcomplex of K and we use this sequence to create the boundary matrix, ∂ . The rows and columns are ordered like the simplices in the total ordering and the boundary of a simplex is kept in its column. The simplices of all dimension are stored in a single matrix, that is,

$$\partial[i, j] = \begin{cases} 1 & , \text{ if the } i\text{-th } (p-1)\text{-simplex is a face of the } j\text{-th } p\text{-simplex} \\ 0 & , \text{ otherwise} \end{cases} \quad (2)$$

The algorithm uses column operations to reduce ∂ to another 0-1 matrix R . This reduction is made by adding columns from left to right.

Let $\text{low}(j)$ be the row index of the lowest one in column j . If the entire column is zero then $\text{low}(j)$ is undefined. These lowest ones are directly linked with the persistent homology groups and they are unique.

We call R *reduced* if $\text{low}(j) \neq \text{low}(j_0)$ whenever $j \neq j_0$ represent two non-zero columns. The reduced matrix R is not unique, but we can stop the operations once we get a reduced matrix.

In [2] there is a characterization of the lowest ones that does not depend on the reduction process.

Column j reaches its final form at the end of the j -th iteration of the outer loop. So by that moment we already have the reduced matrix for the complex consisting of the first j simplices in the total ordering.

There are two cases, the one in which column j is zero and the other in which it has a lowest one.

1. Column j of R is zero. We call σ_j *positive* because it gives a birth to a new homology class. Its addition creates a new cycle.
2. Column j of R is nonzero. We call σ_j *negative* because it gives a death to a homology class. These columns give a death for the column corresponding to the low.

If $i = \text{low}(j)$ column j it is born at the time the simplex of its lowest one, σ_i .

Proof. (From [2]) The cycle in column j of R just died and all other cycles that die with it have ones below row i , else we could further reduce the matrix and obtain $\text{low}(j) < i$, which contradicts the algorithm. \square

The lowest ones correspond to the points in the persistence diagrams. More precisely, (a_i, a_j) is a finite point in $Dgm_p(f)$ if and only if $i = \text{low}(j)$, being σ_i a simplex of dimension p and σ_j a simplex of dimension $p + 1$.

Furthermore, (a_i, ∞) in $Dgm_p(f)$ if and only if column i is zero but row i does not contain a lowest one. In other words, σ_i is positive but it does not get paired with a negative simplex. There is not a death for that birth.

9.3 Python experiment and examples

This Section explains an experiment conducted through python in dimension two. The entire code is available as appendixA at <https://github.com/idoiagamiz/GABRIEL-VS.-DELAUNAY-Computing-the-differences-between-the-persistent-homologies.git>

The number of points, k , of the data set is chosen. In this paper, two different sizes of data will be explained simultaneously.

On the one hand, let $k = 4$ be a simple example to illustrate more clearly the results.

On the other hand, since when working with real problems the size of the data is usually considered to be very big, let k be a higher number that exemplify better this situation. Take $k = 100$.

Let S be a randomly created array which keep the coordinates of the k points. Each point will be represented by an index from 0 to $k - 1$. For $k = 4$:

0:(0.03251924,0.26965579), 1:(0.52825821,0.99552744),
2:(0.83619923,0.41257578) and 3:(0.94820483,0.38014814)

We apply to S a Python function that gives the triangles(the 2-simplices) of the Delaunay triangulation. For $k = 4$: $[3 \ 2 \ 0]$, $[2 \ 1 \ 0]$, $[1 \ 2 \ 3]$ From this triangles it is easy to obtain the 1-simplices by doing combinations.

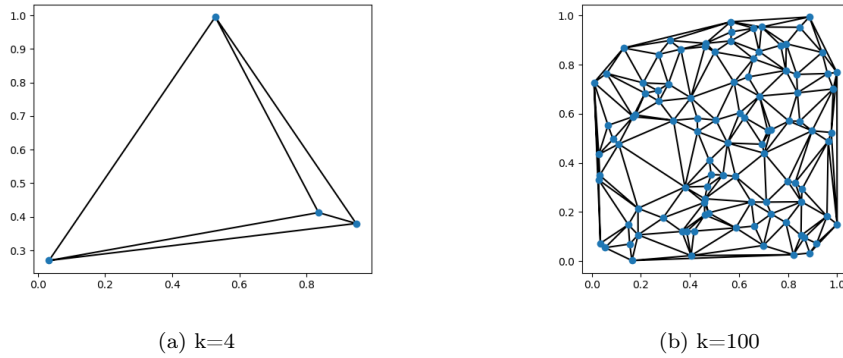


Figure 5: The Delaunay triangulation for the two values of k .

In the next step, we choose the edges that are Gabriel between the edges of the Delaunay complex. Recall that an edge is Gabriel if and only if the midpoint of the vertices p and q of the edge is not closer to any other element of S than p and q . That is, the distance can not be smaller or equal than the radius of the ball centered at the midpoint.

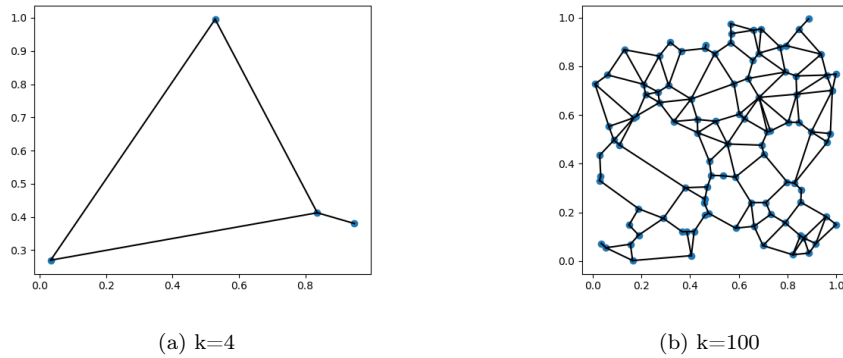


Figure 6: The Gabriel 1-simplices for the two values of k

Once we have the Delaunay triangulation, we define the simplicial complex K consisting on the

vertices, the edges and the triangles. Let $|K|=n$.

What we need to do next is to define our filtration value function f based on distance between points. Let $w(\sigma)$ be the weight of the edge. Then the function f is defined as follows;

$$f(\sigma) = \begin{cases} 0 & , \text{ if } \sigma \text{ is a 0-simplex} \\ w(\sigma) & , \text{ if } \sigma \text{ is a 1-simplex} \\ \max w(\sigma_i) & , \text{ if } \sigma \text{ is a 2-simplex} \end{cases} \quad (3)$$

where σ_i represent each of the edges of the triangle σ .

Applying 3 to the simplicial complex K we get a compatible ordering of the simplices. If we have the same filtered value for an edge and a triangle, then the edge always goes first in the total ordering. In fact, that equality implies that the line is an edge of the triangle.

We sort the filtration values and we reorder K following the same rule. The next step is to create the unique boundary matrix of size $n \times n$ and to apply the algorithm as explained in Section 9.2. Both the rows and the columns of the matrix represent the elements of K in order.

From this moment, we will follow with the example for $k = 4$ because it is not worth continuing with the other example due to the lack of space.

[0. 0. 0. 0. 0. 0. 0. 0. 1. 1. 0. 1. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0. 0.]	[0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0. 0. 0.]	[0. 0. 0. 0. 1. 1. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0.]	[0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]

(a) ∂
(b) R

Figure 7: The boundary matrix ∂ of K and the corresponding reduced matrix R .

$K : \{0\}, \{1\}, \{2\}, \{3\}, \{2, 3\}, \{1, 2\}, \{1, 3\}, \{1, 2, 3\}, \{0, 2\}, \{0, 1\}, \{0, 1, 2\}, \{0, 3\}, \{0, 2, 3\}$

To sum up, we need to repeat the same procedure but for the Gabriel simplices. We define $K1$ to be the simplicial complex that contains the 0-,1- simplices that are Gabriel and the 2-simplices that has all it edges Gabriel. Recall that in \mathbb{R}^2 all the 2-simplices are Gabriel, but we are not going to use all of them to construct our simplicial complex. Let $|K1| = m$, where $m < n$. The matrices that we get are of smaller dimension, $m \times m$.

[0. 0. 0. 0. 0. 0. 1. 1. 0.]	[0. 0. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 1. 0. 1. 0.]	[0. 0. 0. 0. 0. 1. 1. 0. 0.]
[0. 0. 0. 0. 1. 1. 1. 0. 0.]	[0. 0. 0. 0. 1. 1. 0. 0. 0.]
[0. 0. 0. 0. 1. 0. 0. 0. 0.]	[0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0.]

(a) $\partial 1$
(b) $R1$

Figure 8: The boundary matrix $\partial 1$ of $K1$ and the corresponding reduced matrix $R1$.

$K1 : \{0\}, \{1\}, \{2\}, \{3\}, \{2, 3\}, \{1, 2\}, \{0, 2\}, \{0, 1\}, \{0, 1, 2\}$

In this matrices R and $R1$ the columns that are zero give a birth, whereas the columns that are nonzero give a death for the column corresponding to the low.

In this matrices we observe two type of cycles. We keep the information in two dictionaries, one for each simplicial complex, where the keys are the births and the values are the corresponding deaths.

1. 0-persistent homology: A vertex gives birth to a 0-cycle that an edge kills. Since the two 0-persistent homologies are isomorphic, we get the same dictionary.

$$K : \{0 : \text{None}, 1 : \{0, 2\}, 2 : \{1, 2\}, 3 : \{2, 3\}\}$$

$$K1 : \{0 : \text{None}, 1 : \{0, 2\}, 2 : \{1, 2\}, 3 : \{2, 3\}\}$$

We can see that the vertex "0" is the one that has been chosen as the representative of the 0-persistence homology. This cycle never dies.

2. 1-persistent homology: An edge gives birth to a 1-cycle that a triangle kills. We do not get the same dictionary, but the one of $K1$ is inside the one of K .

$$K : \{\{2, 3\} : \text{None}, \{1, 2\} : \text{None}, \{1, 3\} : \{1, 2, 3\}, \{0, 2\} : \text{None}, \{0, 1\} : \{0, 1, 2\}, \{0, 3\} : \{0, 2, 3\}\}.$$

$$K1 : \{\{2, 3\} : \text{None}, \{1, 2\} : \text{None}, \{0, 2\} : \text{None}, \{0, 1\} : \{0, 1, 2\}\}.$$

10 \mathbb{R}^3 : What about persistent 1-homology of $D\check{C}_t$ and $D_2\check{C}_t$?

There is not an isomorphism between this two 1-homologies, so first we will compute an experiment that is going to fail. After that, we will try to define an extended version of the Gabriel 1- and 2-simplices such that the isomorphism is fulfilled. This process is based on try and error until we reach the desired isomorphism.

When deciding which of the Delaunay 1-simplices are also Gabriel we usually check that the number of points of S that are closer to the midpoint of the edge pq than p and q is zero. Now

we will allow this number of points to be less or equal to 1,2,3... until we get what we want. It takes sense because in each step we are getting more and more closer to the real Delaunay simplices.

10.1 Python experiments and examples

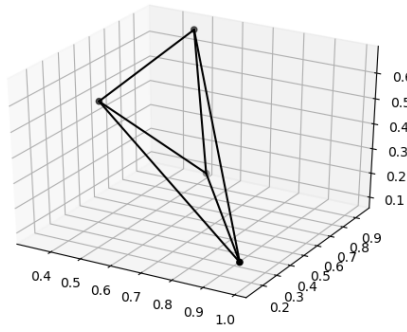
The entire code is available as appendixB2 at <https://github.com/idoiagamiz/GABRIEL-VS-DELAUNAY-Computing-the-differences-between-the-persistent-homologies.git>

Let $k = 4$ for the first example and $k = 10$ for the second one. For $k = 4$:

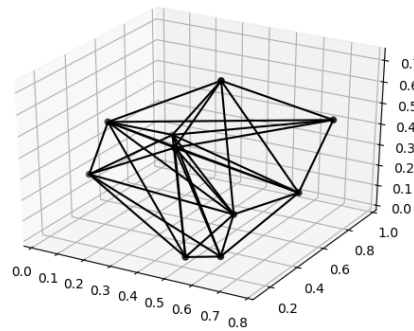
0: (0.33291177, 0.55892937, 0.47344252), 1: (0.46523839, 0.949858, 0.66214016)

2: (0.51876311, 0.92521649, 0.09629331) and 3: (0.9883815, 0.14287922, 0.15805861)

Since we are working with $d=3$, when we apply the Delunay Python function to S , instead of getting triangles we get the tetrahedrons(3-simplices) of the tetrahedralization. For $k = 4$: [1 2 3 0]. From this tetrahedrons it is easy to obtain the 1- and 2-simplices by doing combinations.



(a) $k = 4$

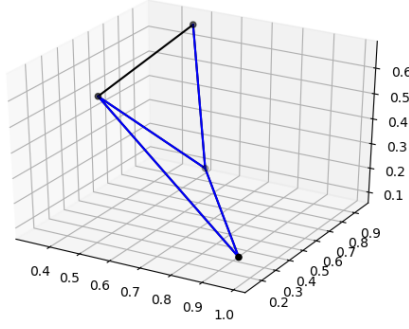


(b) $k = 10$

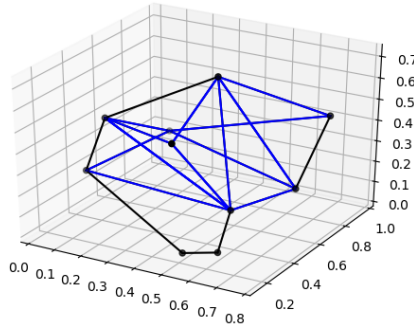
Figure 9: The Delaunay tetrahedralization for the two values of k

In the next step, we choose the edges that are Gabriel exactly in the same manner as in dimension

two. Nevertheless, when we talk about $Del_2(S)$ we usually refer to the Gabriel 2-simplices, that is, to the triangles that are Gabriel. In this experiment we are not interested in calculating those simplices because we want to create something that only depends of the Gabriel 1-simplices. Between all the Delaunay triangles we take just the ones that have all its edges Gabriel.



(a) $k = 4$



(b) $k = 10$

Figure 10: The Gabriel 1-simplices(black) and the Delaunay 2-simplices with Gabriel edges(blue) for the two values of k . For $k = 4$, $(1, 3)$ is the only edge that is not Gabriel, and therefore the triangles $(1, 3, 0)$ and $(1, 2, 3)$ does not appear in blue.

Once we have the Delaunay tetrahedralization, we define the simplicial complex K consisting on the vertices, the edges, the triangles and the tetrahedrons. Let $|K| = n$.

In dimension three our filtration value function f is defined as follows:

$$f(\sigma) = \begin{cases} 0 & , \text{ if } \sigma \text{ is a 0-simplex} \\ w(\sigma) & , \text{ if } \sigma \text{ is a 1-simplex} \\ \max w(\sigma_i) & , \text{ if } \sigma \text{ is a 2-simplex} \\ \max w(\sigma_i) & , \text{ if } \sigma \text{ is a 3-simplex} \end{cases} \quad (4)$$

where σ_i represent each of the edges of the triangle or the tetrahedron σ . We sort the filtration values and we reorder K . As before, we will continue with the smallest sample with $k=4$.

[0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 1. 0. 1. 0. 0. 0. 0. 1. 0. 0. 0.]	[0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 1. 0. 1. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 1. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 1. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 1. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]

(a) ∂
(b) R

Figure 11: The boundary matrix ∂ of K and the corresponding reduced matrix R .

$K = \{0\}, \{1\}, \{2\}, \{3\}, \{0, 1\}, \{0, 2\}, \{1, 2\}, \{0, 1, 2\}, \{0, 3\}, \{2, 3\}, \{0, 2, 3\}, \{1, 3\}, \{0, 1, 3\}, \{1, 2, 3\}, \{0, 1, 2, 3\}$

We will repeat the same procedure for the Gabriel simplices. Define $K2$ to be the simplicial complex that contains the 0-,1- simplices that are Gabriel and 2-,3-simplices with all its edges Gabriel. Recall that in \mathbb{R}^3 all the 3-simplices are Gabriel, but we are not going to use all of them to construct our simplicial complex. Let $|K2| = m$, where $m < n$.

[0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0.]	[0. 0. 0. 0. 1. 1. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 1. 0. 1. 0. 0. 0. 0.]	[0. 0. 0. 0. 1. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 1. 1. 0. 0. 1. 0.]	[0. 0. 0. 0. 0. 1. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 1. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 1. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 1.]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]	[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]

(a) $\partial 2$
(b) $R2$

Figure 12: The boundary matrix $\partial 2$ of $K2$ and the corresponding reduced matrix $R2$

$K2 = \{0\}, \{1\}, \{2\}, \{3\}, \{0, 1\}, \{0, 2\}, \{1, 2\}, \{0, 1, 2\}, \{0, 3\}, \{2, 3\}, \{0, 2, 3\}$

In this case we can see an additional type of cycle:

1. 0-persistent homology: A vertex gives birth to a 0-cycle that an edge kills. We get the same dictionary.

$$K : \{0 : \text{None}, 1 : \{0, 1\}, 2 : \{0, 2\}, 3 : \{0, 3\}\}$$

$$K2 : \{0 : \text{None}, 1 : \{0, 1\}, 2 : \{0, 2\}, 3 : \{0, 3\}\}$$

We can see that the vertex "0" is the one that has been chosen as the representative of the 0-persistence homology. This cycle never dies.

2. 1-persistent homology: An edge gives birth to a 1-cycle that a triangle kills. We do not get the same dictionary.

$$K: \{\{0, 1\}: \text{None}, \{0, 2\}: \text{None}, \{1, 2\}: \{0, 1, 2\}, \{0, 3\}: \text{None}, \{2, 3\}: \{0, 2, 3\}, \{1, 3\}: \{0, 1, 3\}\}$$

$$K2: \{\{0, 1\}: \text{None}, \{0, 2\}: \text{None}, \{1, 2\}: \{0, 1, 2\}, \{0, 3\}: \text{None}, \{2, 3\}: \{0, 2, 3\}\}$$

3. 2-persistent homology: A triangle gives birth to a 2-cycle that a tetrahedron kills. In this particular example $K2$ does not have any tetrahedron.

As we can see we do not obtain the same 1-persistent homology. But, let us try with the extended version of Gabriel simplices. By trying an error, for this example, we have to fix the number of points to less or equal to 2. In this case, we get $K2 = K$ and therefore the two persistent homologies are isomorphic. This result is quite obvious because we end up having the same simplicial complex, but for a bigger number of points it could be possible to get the same persistent 1-homology without getting the same simplicial complex.

11 Clustering

In this Section we will introduce clustering, which is an important application of the persistent homology.

It consists on a partition of data into groups of similar objects. Each of these groups are called clusters and the objects that compose it are similar to each other and dissimilar to objects from other groups. It is not just a way of creating different groups, but also a tool for detecting hidden patterns in complex or high-dimensional data.

For example, in statistics it is used to classify people by different factors such as age, income, quality of life, type of family or educational level. So sorting data into clusters leads to more investigation into the data.

Clustering is quite subjective in many cases because it depends on what you perceive as common threads in the data. We need to define clearly which are the limits. There are many types of clustering: variational, spectral, k-means, hierarchical, ... In this section we are particularly interested in the relation between the persistent homology and the clustering, specially the k -means and the hierarchical.

11.1 Persistent Homology and clustering

Persistent homology studies the evolution of topological features by filtration. At a fixed filtration, we can use it to find different topological features in a data set. One of the advantages is that it is able to identify certain patterns in survival data sets.

When working with 0-persistent homology the features are connected components, in 1-persistent homology are the loops and in k -persistent homology are the k -dimensional holes.

We will explain in detail some examples for the first two dimensions. This examples have been taken and summarized from the article [10]. We can also find other examples there. .

0-persistent homology

Suppose that we start with a set of points and that each point is in its own cluster. We will use the 0-persistent homology to separate it into two clusters that represent two connected components.

The 0-persistent homology can be explained as growing balls simultaneously around each point. The radius of the balls is increased until we have the number of clusters that we want, in this case two.

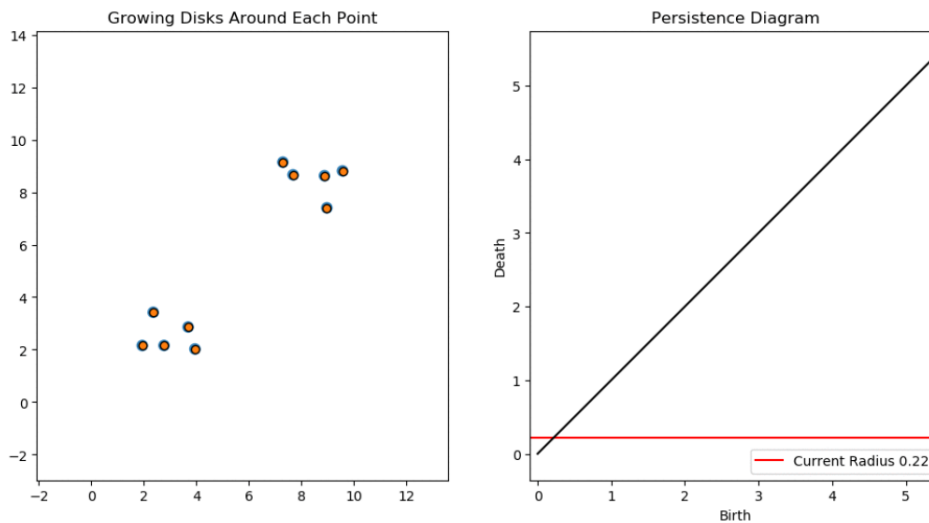


Figure 13: Data set at the beginning. From [10].

It is useful in the search of connected components. In fact, it records when a ball in one connected component first intersects a ball of different connected component. When they touch, they will become part of the same connected component. In that moment, we have our first death of a connected component that is represented with a point on the persistence diagram.

At the end, we have:

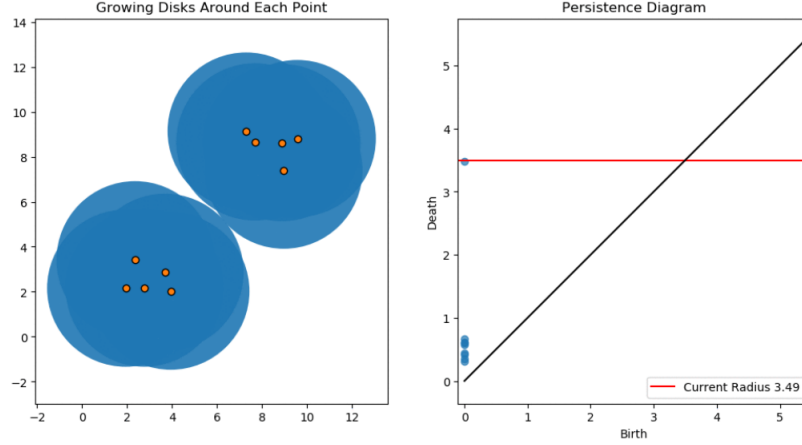


Figure 14: Two noisy clusters of data (left), and the corresponding 0d persistence diagram (right). From [10], to see a virtual graphic of the process go there.

If we do not stop in two clusters and we continue, when we have one connected component, there cannot be any additional death. In this situation, we can add a pair (birth, ∞) to represent it.

The interpretation of the persistence diagram can be explained as follows:

1. The points of the persistence diagram that are closer to 0 represent the noisiness of the clusters.
2. Any point above this, specially the separate and higher persistence value, represent the separation of the two clusters.

This means that the gaps between the points in the diagram explain the manner in which the data is clustered depending on its noisiness.

1-persistent homology

Suppose that we have a noisy circle of data.

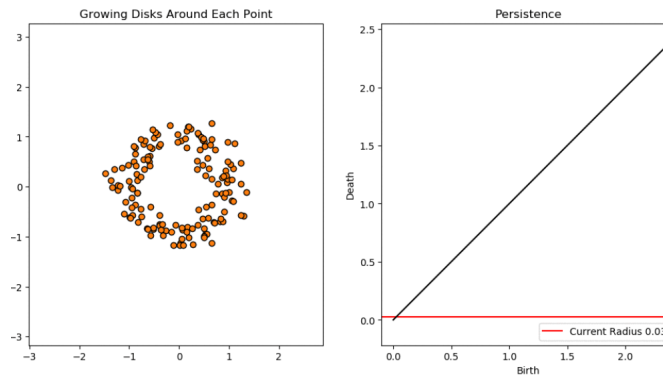


Figure 15: The data set at the beginning. From [10].

The 1-persistent homology also makes balls grow simultaneously around each point, as in the 0-persistent homology. The difference is that in this case it does not just track connected components, but also it realizes when loops form and disappear.

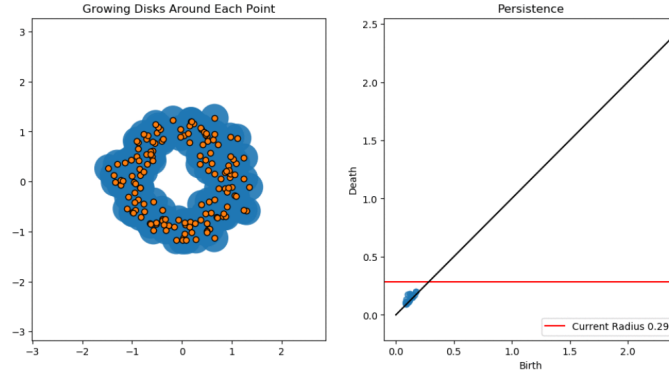


Figure 16: The balls around each point forming the biggest ring. From [10].

In this process rings appeared and a loop is formed when the white space inside the ring is totally enclosed. This loop is unique. As the threshold increases the ring thickens and the white space of the middle of the ring is reduced, until finally the growing disks eventually fill in the interior of the ring around radius b .

Then, we say that the loop is born at a and its death occurs at b .

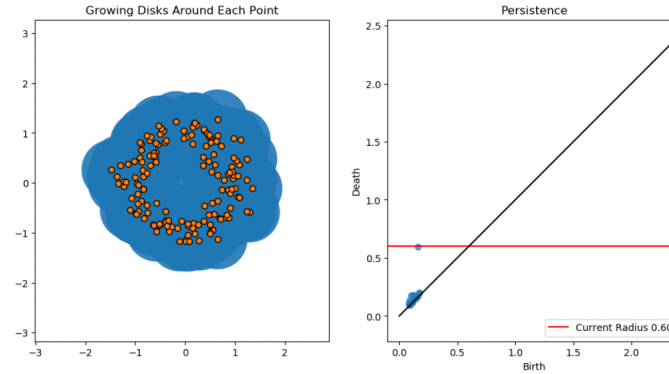


Figure 17: The outside totally enclosed ring represent by the separated point. From [10], see the third graphics.

The noisy persistence values on the diagonal when they take very low persistence values represents instances of loops forming early on. These loops have a small radius and are thus quickly filled in by the expanding disks. This means that death values are close to their birth values and therefore the pairs on the persistence diagram are close to the diagonal. The only loop with big radius is the outside loop, formed with all the points. It is the only point separated from the diagonal.

11.2 K -means clustering

K -means clustering is an iterative data-partitioning algorithm. The k represent the number of clusters that we want to identify in our data and is chosen before the algorithm starts. It starts with initial estimates for the k centroids, which can either be randomly generated or randomly selected from the data set. So this centroids will be the initial k clusters.

The algorithm iterates between two steps until a stopping criteria is met:

1. Data assignment : assign each point to one of the k clusters. Each centroid defines one of the clusters and each data point is assigned to its nearest centroid, based on the squared Euclidean distance.
2. Centroid update: the centroids are recomputed. This is done by taking the mean of all data points assigned to that centroid's cluster.

11.3 Advantages of 0-persistent homology over K -means

As explained in Section 11.1 the 0-persistent homology can be used to make clusters between data points. In fact, it has some important advantages because it give us additional information that the K -means do not. This advaneges are taken from [10].

On the one hand, just one run of k -means give us less information than the 0-persistent homology because it tells you nothing about the stability of k clusters relative to any other number of clusters. However, computing 0-persistent homology gives you a measure by the separation of values on the persistence diagram. To get this information from K -means we would need more than one run.

On the other hand, k -means are not necessarily stable. In fact, it is possible to run it twice with the same parameters and to get different answers. Nevertheless, 0-persistent homology is a stable result.

11.4 Hierarchical clustering dendograms

The concepts for this subsection are explained in [11]. Assume we have n data points and that we need K clusters. Initialize n clusters where each data point is in its own cluster. Then, we merge two nearest clusters into one and we update distances of new cluster to existing ones. A pair of clusters is joined based on similarities, giving one less cluster. We repeat the process of merging similar clusters until K clusters are formed or until reaching another stopping criterion.

The dendrogram is a graph that shows hierarchical clusters. It consist on a cluster tree to represent data, we will explain the agglomerative (bottom-up) technique: we start with single point cluster and merge it to a parent cluster. Is a tree so that:

- each node represent a cluster,
- the root node contains the whole data,

- each internal node has two daughters, corresponding to the clusters that were merged to obtain it.

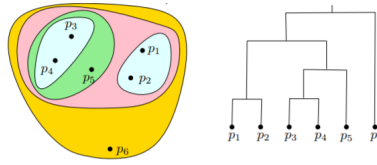


Figure 18: From [11].

11.5 Hierarchical clustering and persistent homology

One of the drawbacks of hierarchical clustering is that dendrograms may be unstable. In fact, small perturbations on the input data or the choice of the neighborhood graph may lead to wide change in the structure of the trees. The problem is that there are more than one ways of merging the clusters.

Nevertheless, persistence diagrams can be efficiently computed and have stability properties.

12 Conclusion

The aim of this project was to create an alternative way of computing the persistent homology and to show some of its applications and advantages. As we can see in the last section, it has notable advantages over other types of clustering and this makes it an efficient way of separating noise from data.

In \mathbb{R}^2 we have achieved our goal of getting the same persistent 0-homology by taking the simplices that has Gabriel 1-simplices as faces. In terms of Data Analsys, this means that having a set of data points we could apply this method in order to get the persistence diagram. Once we have that, if the k -dimensional hole has a short lifetime it is considered to be topological noise.

In \mathbb{R}^3 , by the moment we have not discovered a way of computing the persistent 1-homologies by calculating just the Gabriel 1-simplices, but we have found a clue on how to do it. The next step would be to determine a way of calculating such Gabriel extensions for each particular example in such a way that we get the same persistent homology before obtaining exactly the same simplicial complex.

References

- [1] Nicolau, M., Levine, A. J., & Carlsson, G. (2011). Topology based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival. *Proceedings of the National Academy of Sciences*, 108(17), 7265-7270.
- [2] Edelsbrunner, H., Harer, J., & American Mathematical Society. (2010). *Computational topology : An introduction*. Providence, R.I: American Mathematical Society.
- [3] Atienza, N., Gonzalez-Diaz, R., & Rucco, M. (2019). Persistent entropy for separating topological features from noise in vietoris-rips complexes. *Journal of Intelligent Information Systems*, 52(3), 637-655.
- [4] Noisy data.(2020,January 19). In Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Noisy_data
- [5] Topological data analysis.(2020, April 12). In Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Topological_data_analysis
- [6] Boissonnat, Jean-Daniel, & Teillaud, Monique. (2006). *Effective Computational Geometry for Curves and Surfaces*. Berlin, Heidelberg: Springer Berlin Heidelberg.
- [7] Edelsbrunner, H. (2010). Alpha shapes—a survey. *Tessellations in the Sciences*, 27, 1-25.
- [8] Veltkamp, R., & Springerlink. (1994). *Closed Object Boundaries from Scattered Points* (Vol. 885, *Lecture Notes in Computer Science*). Berlin, Heidelberg: Springer Berlin Heidelberg.
- [9] Euclidean minimum spanning tree.(2019, October 24) In Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Euclidean_minimum_spanning_tree
- [10] Gary Koplik(Oct 29, 2019).Persistent Homology: A Non-Mathy Introduction with Examples. Towards Data Science <https://towardsdatascience.com/persistent-homology-with-examples-1974d4b9c3d0>
- [11] Frédéric Chazal. Clustering algorithms and introduction to persistent homology.Foundations of Geometric Methods in Data Analysis 2017-18. <http://www-sop.inria.fr/teams/abs/classes/centrale-FGMDA/course-material/fchazal-course-ClusteringPersistence.pdf>