Short Introduction to Topological Data Analysis, Persistent Homology and Applications

Stefano De Marchi[1]

Department of Mathematics "Tullio Levi-Civita", University of Padova

Uppsala, August 2025

[1] with Cinzia Bandiziol (UniPD), Federico Lot (Marburg), Francesco Marchetti (UniPD) and Davide Poggiali (ex-UniPD)

Part I

Introduction and basic things on topology/homology

Co-authors

Figure: **Left to right:** Cinzia Bandiziol, Federico Lot, Francesco Marchetti and Davide Poggiali

Reference papers

F. Marchetti, F. De Martino, M. Shamseddin, S. De Marchi and C. Brisken *Variably Scaled Kernels Improve Classification of Hormonally-Treated Patient-Derived Xenografts*, 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS), Bari, pp. 1-6.

S. De Marchi, F. Lot, F. Marchetti and D. Poggiali: *Variably Scaled Persistence Kernels (VSPKs) for persistent homology applications*, J. Comput. Math. and Data Science 4 (2022), 100050.

C. Bandiziol, S. De Marchi: *Persistence symmetric kernels for classification: A comparative study*, Symmetry 16(9) (2024), 1236 - special issue "Algebraic Systems, Models and Applications".

M. Allegra, C. Bandiziol and S. De Marchi, *On intrinsic dimension of point clouds by a persistent homology approach: computational tips,* In preparation.

Cinzia Bandiziol: Applications of Persistent Homology: Data Classification and Intrinsic Dimension of Manifolds, Ph. D. Dissertation (2025).

Motivation

From the Introduction of the first reference papers above

> [...] we first analyze the structure of our data using a clustering technique from the persistent homology framework [7] [....]

[7] G. Carlsson, "Topology and data", Am. Mat. Soc. 46(2) (2009), pp. 255– 308.

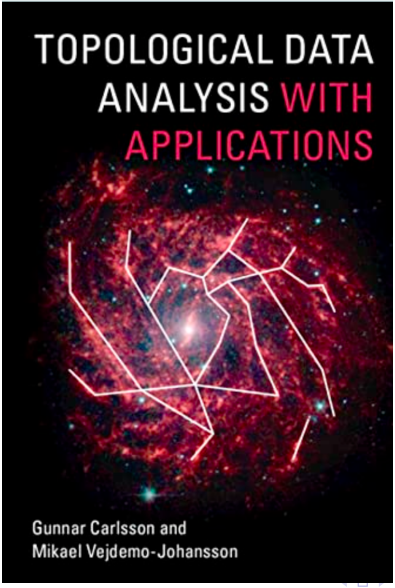
Data analysis is the heart of data science!

Introduction

The birth of Topological Data Analysis (TDA)(*)

- Emerging discipline that examines geometric properties of data using tools from Algebraic Topology
- Big amount of data to analyze, complex and/or of high dimension
- Dispose of tools able to extract new and intrinsic information from data, that is features, related to the "shape of data" or their characteristics;
- The analysis is based on Persistent Holomogy (PH) (studied in algebraic geometry) and in particular by means of *Persistence Diagrams (PD) or Persistent Barcodes (PB)*, connected to some features of the data, for improving the performance of the classification problem by SVM and other information connected to the data.

(*) Gunnar Carlsson, Mikael Vejdemo-Johansson: *Topological Data Analysis with Applications* (2022, Cambridge University Press)


TOPOLOGICAL DATA
ANALYSIS WITH
APPLICATIONS
Gunnar Carlsson and
Mikael Vejdemo-Johansson

Fields of applications of TDA

Here a partial list

- Chemistry
  - Townsend J.; Micucci C.P.; Hymel J. H.; Maroulas V.; Vogiatzis K. D. Representation of molecular structures with persistent homology for machine learning applications in chemistry. Nat. Commun 2020, 11, 3230
- Oncology/Medicine
  - Bukkuri A.; Andor N.; Darcy I. K.: Applications of Topological Data Analysis in Oncology. Front. Artif. Intell. 2021, 4, 659037
  - Moon C.; Li Q.; Xiao G.: Using persistent homology topological features to characterize medical images: Case studies on lung and brain cancers. Ann. Appl. Stat. 2023, 17
- Biomedicine
  - Skaf Y.; Laubenbacher R.: Topological data analysis in biomedicine: A review. Journal of Biomedical Informatics 2022, 130, 104082

'Cont

- Neuroscience
  - Bhattacharya D.; Kaur R.; Aithal N.; Sinha N.; Issac T. G. Persistent homology for MCI classification: a comparative analysis between graph and Vietoris-Rips filtrations. Front. Neurosci. 2025, 19
  - Flammer M.: Persistent Homology-Based Classification of Chaotic Multi-variate Time Series: Application to Electroencephalograms. Sn Computer Science 2024, 5, 107
  - Pachauri D.; Hinrichs C.; Chung M.K.; Johnson S.C.; Singh V.:Topology based Kernels with Application to Inference Problems in Alzheimer's disease. IEEE Transactions on Medical Imaging 2011, 30, 1760–1770
- Computer graphics
  - Bruel-Gabrielsson R.; Ganapathi-Subramanian V.; Skraba P.; Guibas L.J.: Topology-Aware Surface Reconstruction for Point Clouds. Computer Graphics Forum 2020, 39, 197–207
- Physics, Statistics, Agricolture, Engineering applications
- ETC...

Notice: the majority of the references are quite recent.

'Cont

- Neuroscience
  - Bhattacharya D.; Kaur R.; Aithal N.; Sinha N.; Issac T. G. Persistent homology for MCI classification: a comparative analysis between graph and Vietoris-Rips filtrations. Front. Neurosci. 2025, 19
  - Flammer M.: Persistent Homology-Based Classification of Chaotic Multi-variate Time Series: Application to Electroencephalograms. Sn Computer Science 2024, 5, 107
  - Pachauri D.; Hinrichs C.; Chung M.K.; Johnson S.C.; Singh V.:Topology based Kernels with Application to Inference Problems in Alzheimer's disease. IEEE Transactions on Medical Imaging 2011, 30, 1760–1770
- Computer graphics
  - Bruel-Gabrielsson R.; Ganapathi-Subramanian V.; Skraba P.; Guibas L.J.: Topology-Aware Surface Reconstruction for Point Clouds. Computer Graphics Forum 2020, 39, 197–207
- Physics, Statistics, Agricolture, Engineering applications
- ETC...

Notice: the majority of the references are quite recent.

Algebraic topology

Simple definition

Branch of mathematics that uses tools form abstract algebra for studying topological spaces. The main goal of algebraic topology is finding algebraic invariants to classify topological spaces up to homeomorphism (homotopy equivalence).

Among the ways to classify a topological space we recall: homotopy groups (see $H_n(X)$ below), homology, co-homology (that are sequences of invariant groups), manifolds (each point resembles a Euclidean space).

An example of invariant: $H_n(X)$

Given a topological space (X, τ) (τ is the topology on it), the n-th homology group $H_n(X)$, consists of the n-dimensional holes that characterize the space itself. In applications we usually considers the groups with $n = 0, 1, 2$ (as we see more in details in the sequel).

Example: $X = \mathbb{S}^2$, the 2 sphere

- It's a two-dimensional manifold, meaning that at any point on the sphere, you can find a small region that looks like a piece of a two-dimensional plane
- Counting the number of connected components (0-dimensional holes), loops/tunnels (1-dimensional holes) and cavities/voids (2-dimensional holes) allow to characterize the space X from a qualitative and intrinsic point of view. These are the Betti numbers

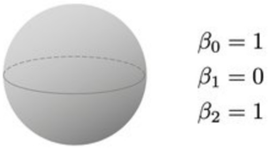
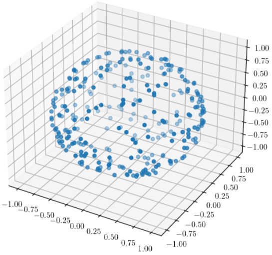
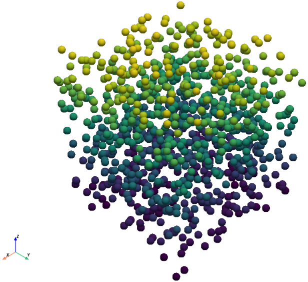


Figure: Sphere with its Betti Numbers

Notice: In general, for $\mathbb{S}^n$, Betti numbers are
$\beta_0 = \beta_n = 1, \beta_k = 0, 1 \leq k \leq n-1, k > n.$

Extension to discrete data sets or point clouds





The ingredient is now the Simplicial Homology

Simplicial Homology

In algebraic topology, simplicial homology is the sequence of homology groups of a simplicial complex (generalization of triangulations of a topological space).

- It formalizes the idea of the number of holes of a given dimension in simplicial complexes.
- It generalizes the number of connected components (the case of dimension 0).
- $\hookrightarrow$ It is the basis of the Persistent Homology

Persistent Homology

- TDA has had a fast development thanks to its strong basis on algebraic geometry with its main tool Persistent Homology (cf. e.g. [Edelsbrunner, Letscher and Zomorodian IEEE Symp. 2000], [Carlsson, Bull. AMS 2009])
- Persistent Homology (PH) is a method that allows the computation of persistent topological features from several objects and is able to extract information about the "shape of data" (a nicer survey on interaction between kernels, frames and PH is by Guillemard, Iske ANHA 2017)

How to compute persistent features?

Persistent Homology

- TDA has had a fast development thanks to its strong basis on algebraic geometry with its main tool Persistent Homology (cf. e.g. [Edelsbrunner, Letscher and Zomorodian IEEE Symp. 2000], [Carlsson, Bull. AMS 2009])
- Persistent Homology (PH) is a method that allows the computation of persistent topological features from several objects and is able to extract information about the "shape of data" (a nicer survey on interaction between kernels, frames and PH is by Guillemard, Iske ANHA 2017)

How to compute persistent features?

Simplicial complexes

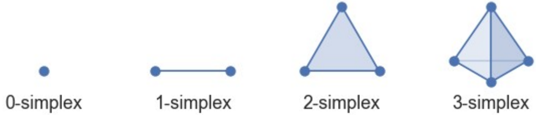
Simplicial complex

A simplicial complex K consists of a set of simplices of certain dimensions that has to meet the following conditions:

- Every face of a simplex in K is also in K
- The non-empty intersection of any two simplices $\sigma_1, \sigma_2 \in K$ is a face of both σ_1 and σ_2

$\hookrightarrow$ The dimension of the complex K is the maximum dimension of simplices that belong to K. $\hookleftarrow$

Simplices and simplicial complex of low dimension



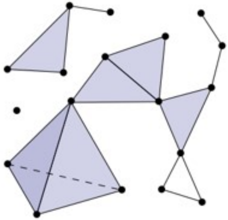


Figure: A simplicial complex of dimension 3, composed of simplices of dimensions 0, 1, 2, 3, respectively (in the first row)

Algebra of simplices

Given a simplex σ of certain dimension n, it is completely defined by its set of vertices denoted by $\{v_1, \dots, v_{n+1}\}$.

- Every subset ρ of $\sigma = \{v_1, \dots, v_{n+1}\}$ represents another simplex, a "face of" σ, briefly denoted by $\rho \leq \sigma$.
- Simplices in K can grouped (depending on their dimension k) and can enumerate them using σ_i^k, which is the i-th simplex of dimension k.
- If $\mathbb{G} = (\mathbb{Z}, +)$ is the well-known Abelian group, we may build linear combinations of simplices with coefficients in $\mathbb{G}$ getting chains of simplices.

k-chain

A integer valued k-dimensional chain is an object of the form

$$c = \sum_i a_i \sigma_i^k, \quad \text{with} \quad a_i \in \mathbb{Z} \, .$$

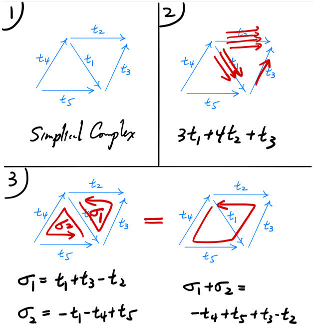
Examples:

- σ_i^k and $-\sigma_i^k$ are the simplest chains.
- A 1-chain like $2\sigma_1 - \sigma_2 + 3\sigma_3$, where σ_i are 1-simplices (line segments). The coefficients indicate how many times each simplex is included and its orientation (positive or negative)

Cont

A chain of simplexes is a sequence where adjacent simplexes share a common facet (a lower-dimensional face).

Example. A chain of 1-simplices (line segments) would be a sequence of line segments where each segment starts where the previous one ends.



Homework: compute and represent $\sigma_2 - \sigma_1$

Summarizing (chain of simplices)

1. Importance in Topology
   - Chains of simplexes are used to define simplicial homology, which is a way to assign algebraic invariants (homology groups) to topological spaces.
   - These homology groups capture information about the "holes" or "connectedness" of the space. As we already saw, the 1-dimensional homology group (also called the fundamental group) captures information about loops in the space. The 2-dimensional homology group captures information about "voids" or "cavities" in the space.
2. Geometric Interpretation
   - A chain of simplexes can be thought of as a "piecewise-linear" approximation of a curve or surface in a topological space. Hence, by taking finer and finer approximations using smaller and smaller simplexes, one can study the topological properties of the space in more detail.

In essense

Chains of simplexes are fundamental building blocks for understanding the topological structure of spaces using simplicial complexes and homology theory.

Group structure

Definition

The set S of integer-valued k-dimensional chains endowed with the binary operation

$$+ : S \times S \to S$$

defined for all $c_1, c_2 \in S$ as

$$c_1 + c_2 = \sum_i a_i \sigma_i^k + \sum_j b_j \sigma_j^k = \sum_l (a_l + b_l) \sigma_l^k \tag{1}$$

is the abelian group of the k-dimensional simplicial integer-valued chains of the simplicial complex K, denoted with $C_k(K)$.

Remarks

- k chains are combinations of k-simplices not necessarily connected
- if in (1) two simplices are different, their coefficients are added separately
- To simplicial complexes we associate the abelian groups $C_0(K), \ldots, C_n(K)$: the generators (...we have a finite number of points).

The boundary operator

Definition

The boundary of a chain is the linear combination of boundaries of the simplices in the chain. The boundary of a k-chain is a $(k-1)$-chain. We denote it with $\partial_k c$.

Note: the boundary of a simplex is not a simplex, but a chain with coefficients 1 or -1 (see below). Thus chains are the closure of simplices under the boundary operator.

Properties

1. ∂ is a linear operator
2. The square of ∂, i.e. ∂^2 is identically 0 (that is, the boundary of a simplex has no boundary)
3. If $\partial(ac) = 0$ with $a \neq 0$ then $\partial c = 0$

In practise, the boundary of $c \in C_k(K)$ is an element in $C_{k-1}(K)$ that we denote as $\partial_k c$. If $c = \sum_{i=1}^{r} a_i \sigma_i$ then $\partial_k c = \sum_{i=1}^{r} a_i \partial_k \sigma_i$

An example

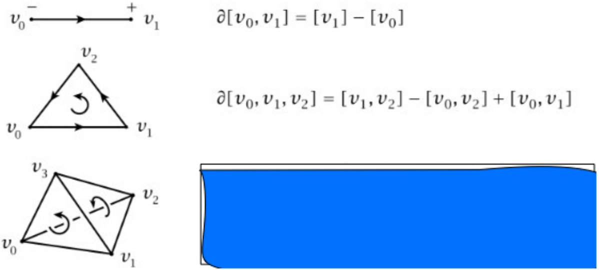
Example

Consider the path from points v_1 to v_4. Letting $s_1 = [v_1, v_2]$, $s_2 = [v_2, v_3]$, $s_3 = [v_3, v_4]$ three 1-simplices and consider the chain $c = s_1 + s_2 + s_3$.

$$\begin{aligned}
\partial_1 c &= \partial_1(s_1 + s_2 + s_3) = \partial_1(s_1) + \partial_1(s_2) + \partial_1(s_3) \\
&= \partial_1([v_1, v_2]) + \partial_1([v_2, v_3]) + \partial_1([v_3, v_4]) \\
&= (v_2 - v_1) + (v_3 - v_2) + (v_4 - v_3) = v_4 - v_1
\end{aligned}$$

Homeworks

Homeworks: • What is the boundary operator of a polygonal open curve $A_1, A_2, ..., A_6$? And if the same curve is closed?



• Given the tetrahedra $T = [v_0, v_1, v_2, v_3]$ in the above figure, with basis $[v_0, v_1, v_2]$ and faces $[v_1, v_2, v_3]$ and $[v_0, v_1, v_3]$, what is ∂T?

Homology group

Some definitions

- A chain c is called a cycle when its boundary is zero, i.e. $\partial c = 0$ (Example is the closed polygonal curve)
- A boundary is a cycle that can be filled in or formed by the boundary of a higher-dimensional object. The fact that every boundary is a cycle is a fundamental property: the boundary of a boundary is always zero, i.e. $\partial^2 = 0$.
- A chain that is the boundary of another chain is called a (chain) boundary.
- Boundaries are cycles (not the opposite!), so chains form a chain complex, whose homology groups (cycles modulo boundaries) are called simplicial homology groups.

Example

The plane punctured at the origin (i.e. the origin is removed!) has nontrivial 1-homology group (it can be shrinked to the unit circle!) i.e. the unit circle which is a cycle, but not a boundary.

Cont

Definition

1. The set of all k-cycles is an abelian group, denoted by $Z_k(K)$ (subgroup of $C_k(K)$).
2. The set of all k-boundary is an abelian group, denoted by $B_k(K)$ (subgroup of $Z_k(K)$).

Definition of k simplicial homological group

Given the simplicial complex K the k-dimensional integer-valued simplicial homological group is the quotient

$$\boxed{H_k(K) := Z_k(K)/B_k(K) = ker(\partial_k)/Im(\partial_{k+1})\ .} \tag{2}$$

Interpretation

The homology groups of K measure "how far" the chain complex associated to K is from being exact.

Cont

Examples: $H_0(K)$ collected the connected components (0-dimensional holes); $H_1(K)$ collects the cycles (1-dimensional holes) and $H_2(K)$ collects the cavities/voids 2-dimensional holes, and so on.

Corollary (see Rotman J. J.: An introduction to Algebraic Topology; Springer 1988)

If K is a simplicial complex of dimension n then

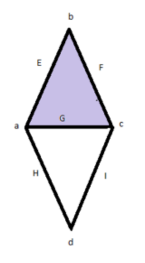
- $H_k(K)$ is finitely generated for every $k \geq 0$
- $H_k(K) = 0$ for $k > n$
- $H_n(K)$ is free abelian group (that is, it has a basis).

Betti numbers

Since $H_k(K)$ has finite independent generators: the number of these generators (the rank of $H_k(K)$), are the Betti numbers

Example 1: two triangles

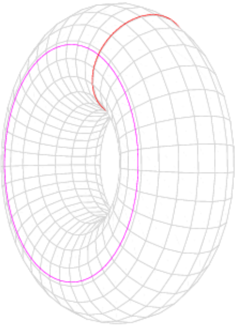
Consider a simplicial complex with 0-simplices: a, b, c, and d, 1-simplices: E, F, G, H and I, and the only 2-simplex is J, which is the shaded region in the figure.



There is one connected component in this figure $b_0 = 1$; one hole, which is the unshaded region $b_1 = 1$ and no "voids" or "cavities" $b_2 = 0$ (trinagles are in the plane).

This means that the rank of H_0 is 1, the rank of H_1 is 1 and the rank of H_2 is 0. The Betti numbers sequence for this figure is 1, 1, 0, 0, ..

Example 2: torus



A torus has

- one connected surface component so $b_0 = 1$,
- two circular holes (one equatorial (red curve) and one meridional (magenta curve)) so $b_1 = 2$,
- one single cavity enclosed within the surface so $b_2 = 1$.

The Poincaré polynomial

Poincaré polynomial

The Poincaré polynomial of a surface is a polynomial whose coefficients are its Betti numbers.

Examples. The Betti numbers of the torus are 1, 2, and 1; thus its Poincaré polynomial is $1 + 2x + x^2$. The Poincaré polynomial of the two triangles is $1 + x$

The same definition applies to any topological space which has a finitely generated homology

General rule

Given a topological space which has a finitely generated homology, the Poincaré polynomial is defined as the generating function of its Betti numbers, via the polynomial where the coefficient of x^n is b_n, that is

$$p_n(x) = b_n x^n + \cdots + b_0$$

Part II

Persistent Homology

Motivation

In the context of Data Analysis, user usually has only a dataset $\mathcal{X}_m = \{\mathbf{x}_k\}_{k=1,\ldots,m}$ that comes/represent from/a manifold $\mathcal{M}$ or a topological space (X, τ), or simply X, and no simplicial complex structure at hand. It is indeed in this case that: Persistent Homology helps to compute topological invariants of finite structures.

The main objective is to compute homological information of the topological space X using only available data $\mathcal{X}_m$.

Cont

Consider the spaces

$$\mathbb{X}_\epsilon = \bigcup_{i=1}^{m} B(x_i, \epsilon) \quad (3)$$

where $B(x_i, \epsilon)$ denotes the ball centered at x_i with radius $\epsilon > 0$.
If ϵ is big enough, $\mathbb{X}_\epsilon$ cover completely the space X and it could suggest that $\mathbb{X}_\epsilon$ could inherit also topological properties of X (and also the geometric ones).

Unfortunately, this kind of approach has revealed some drawbacks and ends up being unstable. But we have simplicial complexes.

Another "difficult" problem

The simplicial complex recognition problem is: given a finite simplicial complex, decide whether it is homeomorphic to a given geometric object. This problem is undecidable for any d-dimensional manifolds for $d \geq 5$ (but we don't talk!)

Čech Complex

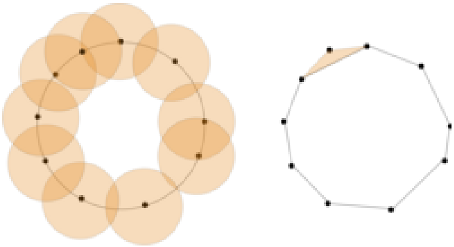
A first example of the simplicial complex which can be constructed from X is the Čech Complex.

The Čech complex is an abstract simplicial complex constructed from a point cloud in any metric space which is meant to capture topological information about the point cloud or the distribution it is drawn from.

Construction

Given a finite point cloud X and $\epsilon > 0$, the Čech complex $\check{C}_\varepsilon(X)$ is constructued as follows

- consider the elements of X as the vertex set of $\check{C}_\varepsilon(X)$
- a simplex σ (an edge, a triangle,...) is added to the complex, i.e. $\sigma \in \check{C}_\epsilon(X)$, if the ϵ-balls centered at points in σ have common intersection



In other words, the Čech complex is the nerve of the set of ε-balls centered at points of X.

Remark. By the nerve lemma (see J. Leray 1945), the Čech complex is homotopy equivalent (by means of some homotopy) to the union of the balls, known as offset filtration

Example

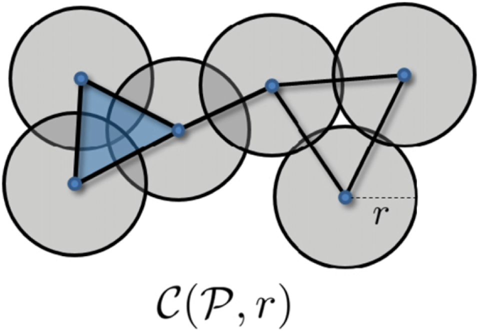
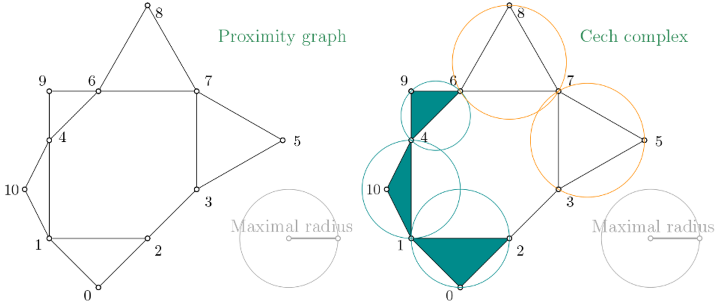


Figure: Čech complex generated on a set $\mathcal{P}$ of 6 points in the plane

Another example

The Čech complex is a simplicial complex constructed from a proximity graph. The set of all simplices is filtered by the radius of their minimal enclosing ball.



On this example, as edges $(x, y), (y, z)$ and (z, x) are in the complex, the minimal ball radius containing the simplex (x, y, z) is computed. Hence (x, y, z) is inserted in the simplicial complex if `min_ball_radius(x,y,z)` $\leq$ `max_radius`

So on, in higher dimensions

Homework

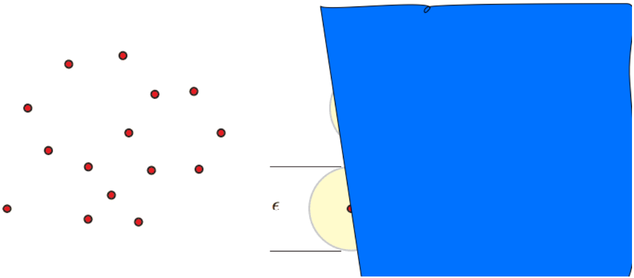


Figure: Build the Čech complex of the set of point on the left using the ϵ-balls as indicated

Vietoris-Rips complexes

Computational problem

The construction of the Čech complex for some $\epsilon > 0$ is costly. In fact, for any subset of vertices we must solve a system of inequalities to find out if the intersection of the ϵ-balls is empty or not.

For this reason data analysts use Vietoris-Rips complexes.

Vietoris-Rips complexes

Data analysts consider Vietoris-Rips complexes associated to a parameter ϵ and to the set $\mathcal{X} = \{x_0, \ldots, x_k\}$, $K = VR(\mathcal{X}, \epsilon)$:
"two vertices are connected by an edge iff $\|x_i - x_j\|_2 \leqslant \epsilon$ and r-dimensional elements are determined by $r+1$ connected $(r-1)$ dimensional faces, $r \leq d$" (d being the space dimension)

In practise: $VK(\mathcal{X}, \epsilon)$ is a simplicial complex that generalizes proximity (ϵ-ball) graphs to higher dimensions.

VR complex: example

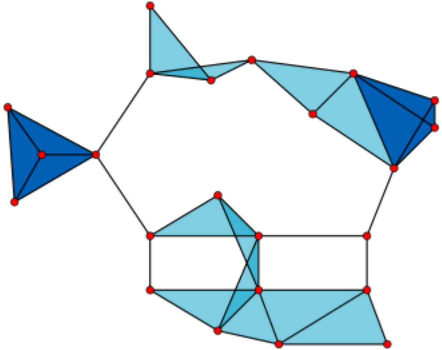


Figure: A Vietoris–Rips complex of a set of 23 points in the Euclidean plane. This complex has sets of up to four points: the points themselves (shown as red circles), pairs of points (black edges), triples of points (pale blue triangles), and quadruples of points (dark blue tetrahedrons)

Relation between Čech complexes and VR complexes

Important

The Vietoris-Rips complex is essentially the same as the Čech complex, except instead of adding a simplex when there is a common point of intersection of all the ϵ-balls, we just do so when all the balls have pairwise intersections.

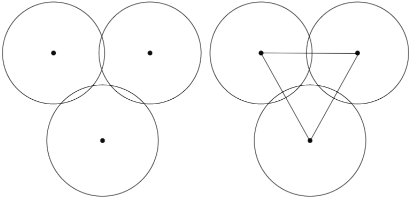


Figure: Given 3 points on an equilateral triangle of unitary sides. Take $\epsilon = 1/2$. On the right $VR_{1/2}$.

What's the Čech complex?

'Cont

Note: Čech complexes are subcomplexes of Vietoris-Rips ones. Moreover,

Theorem

For all $\epsilon > 0$ we have

$$C_\epsilon \subset VR_\epsilon \subset C_{2\epsilon}$$

$\hookrightarrow$ The theorem says that both complexes are homotopy equivalent. So if the Čech complexes for both are good approximations of the underlying data, then so is the Vietoris-Rips complex.

Theorem by de Silva, Ghrist, Alg. Geom. Top. 7(1)(2007)

Let X be a set of point in $\mathbb{R}^d$. Let $\epsilon > 0$, and C_ϵ the Čech complex of X with balls of radius $\epsilon/2$

$$VR_{\epsilon'} \subset C_\epsilon \subset VR_\epsilon, \quad whenever \ \frac{\epsilon}{\epsilon'} \geq \sqrt{\frac{2d}{d+1}}$$

This ratio is the smallest possible for which the inclusion holds.

Filtration and PH group

Persistent Homology analyzes not only simplicial complexes but nested sequences of them and their evolution.

Definition

Given a simplicial complex K, a filtration is a nested family of subcomplexes K_t, $t \in T$ where T is a totally ordered set s.t. for all $t_1, t_2 \in T$, with $t_1 < t_2$, then $K_{t_1} \subset K_{t_2}$ and $K = \cup_{t \in T} K_t$

- In applications $T \subset \mathbb{R}$.
- The previous definition can be extended to a topological space X. If $f : X \to \mathbb{R}$, then the family $(K_t)_{t \in T}$ with $T \subset \mathbb{R}$ defines the so called sublevel set filtration.
- Given a subset $\mathcal{X}$ of a compact metric space, the family of Vietoris-Rips complexes $(VR(\mathcal{X}, \epsilon))_{\epsilon \in \mathbb{R}}$ and the Čech complexes $(\check{C}(\mathcal{X}, \epsilon))_{\epsilon \in \mathbb{R}}$ are filtrations.

Note: the most used are the VR filtrations (computationally less expensive)

Cont

Letting $(0 <)\epsilon_1 < \cdots < \epsilon_l$ be an increasing sequence of real numbers, we obtain the filtration

$$\emptyset = K_0 \subseteq K_1 \subseteq K_2 \subset \cdots \subseteq K_l$$

with $K_i = VR(\mathcal{X}, \epsilon_i)$

s-persistent homologycal group

Given $r \geq 0$ and $i \in \{0, \dots, l\}$. The s-persistent homology group of $\mathcal{X}$ is defined as

$$H_i^{r,s}(\mathcal{X}) = Z_i(K_r)/(Z_i(K_r) \cap B_{i+s}(K_r)).$$

Remarks

- This group contains all homology classes that persist in the interval $[i, i+s]$, i.e they are born before the time/index i and are still alive after s steps.
- The classes that remain alive for large values of s are stable topological features of the set $\mathcal{X}$.

Cont

Remarks continue

- Along the filtration, the topological information appears and disappears, thus it means that they may be represented with a couple of indexes. If p is such a feature, it must be born in some K_i and die in K_j so it can be described as (i, j), $i < j$. We underline here that j can be equal to $+\infty$, since some features can be alive up to the end of the filtration
- Hence, all such topological invariants live in the extended positive plane, that we denote by $\mathbb{R}^2_+ = \mathbb{R}_{\geq 0} \times \{\mathbb{R}_{\geq 0} \cup \{+\infty\}\}$
- Finally, some features can appear more than once: such collection of points are called multisets.

Summarizing

Each element of the persistent homology groups obtained by the whole filtration can be represented by a birth-death pair $(b, d) \in \mathbb{R}^2$, $b = \epsilon_h$, $d = \epsilon_k$ for some $h \in \{0, \ldots, l\}$, $k \in \{0, \ldots, l\} \cup \{\infty\}$, $h < k$

Persistent diagram: definition

Persistent Diagram

A Persistence Diagram (PD), $D_r(\mathcal{X}, \varepsilon)$ related to the filtration $K_1 \subset K_2 \subset \cdots \subset K_l$ with $\boldsymbol{\varepsilon} := (\epsilon_1, \ldots, \epsilon_l)$ is a multiset (due to multiplicities), subset $\mathbb{R}^2$ defined as

$$\boxed{D_r(\mathcal{X}, \varepsilon) := \{(b, d) | (b, d) \in P_r(\mathcal{X}, \varepsilon)\} \cup \Delta}$$

where $P_r(\mathcal{X}, \varepsilon)$ denotes the set of r-dimensional birth-death that came out along the filtration, each (b, d) is considered with its multiplicity, while the points of $\Delta = \{(x, x) \,|\, x \geq 0\}$ have infinite multiplicity.

- Each point $(b, d) \in D_r(\mathcal{X}, \varepsilon)$ is called generator and the difference $d - b$ is called the persistence of the generator, that represents its lifespan and shows the robustness of the topological property.
- We denote by $\mathfrak{D}_r$ all $P_r(\mathcal{X}, \varepsilon)$ for all r

Example: points on $\mathbb{S}^2$

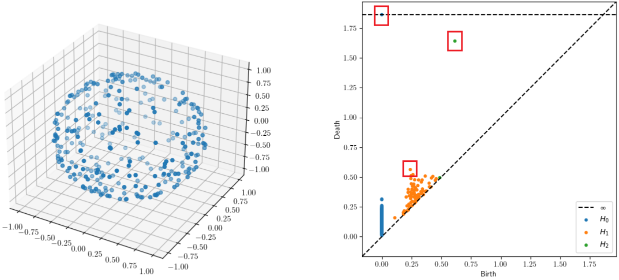


Figure: PD of the collection features of dimension 0 (in blue), of dimension 1 (in orange), and of dimension 2 (in green). Points close to the diagonal represent features with a short lifetime, and so usually they are concerned with noise; instead, features far away are indeed relevant and meaningful, and, based on applications, one can decide to consider both or only the most interesting ones. At the top of the figure, there is a dashed line that indicates infinity and allows us to plot also couples as $(i, +\infty)$. In red, we highlight the most important features: 1 connected component, 1 cycle, and 1 cavity.

MP4 videos

Growing, H_0.

GROWING. We track when the balls intersects. When two balls touch they become one connected component, that is a first death and thus the first point in the PD

Collapsing, H_0.

COLLAPSING. We emphasize that the persistence increases when the noise of each cluster decreases.

Barcode [Barannikov (1994), Carlsson et al. (2004)]

A persistence barcode consists of a multiset of intervals in $\mathbb{R} \cup \{+\infty\}$, where the length of each interval (counterpart of points in the PD) corresponds to the lifetime of a topological feature in a filtration.

Longer intervals in a barcode correspond to more robust features, whereas shorter intervals are more likely to be noise in the data.

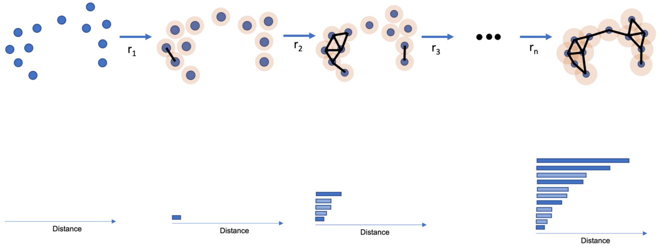
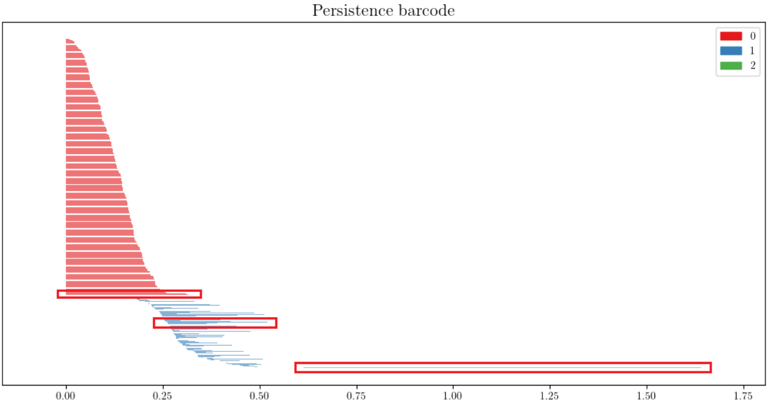


Figure: A series of four nested simplicial complexes and the 0-dimensional (i.e. connected components) persistence barcode of the resulting filtration.

The example of the sphere



Stability

Persistent Diagrams are stable under perturbation of the data. How to measure it?

For two nonempty sets $X, Y \subset \mathbb{R}^2$ with the same cardinality, the Hausdorff distance is

$$d_H(X, Y) := \max\{\sup_{x\in X} \inf_{y\in Y} \|x - y\|_\infty, \sup_{y\in Y} \inf_{x\in X} \|y - x\|_\infty\}.$$

The p-Wasserstein distance, $p > 0$,

$$d_{W,p}(X, Y) = \inf_{\gamma} \sum_{x\in X} \|x - \gamma(x)\|_\infty^p$$

where $\Gamma = \{\gamma : X \to Y | \gamma \text{ bijection}\}$. Taking $p \to +\infty$, we get the bottleneck distance

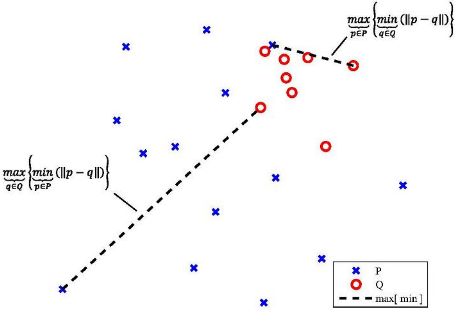
$$d_{W,\infty}(X, Y) = d_B(X, Y) := \inf_{\gamma\in\Gamma} \sup_{x\in X} \|x - \gamma(x)\|_\infty \tag{4}$$

where

$$\|v - w\|_\infty = \max\{|v_1 - w_1|, |v_2 - w_2|\}, \quad \text{for } v = (v_1, v_2), w = (w_1, w_2) \in \mathbb{R}^2$$

Note: Wasserstein distance, also called the Earth mover's distance or the optimal transport distance or Monge problem, is a similarity metric between two probability distributions

Haussdorff distance



MATLAB Central File Exchange
Zachary Danziger (2025). Hausdorff Distance
(https://www.mathworks.com/matlabcentral/fileexchange/26738-hausdorff-distance)
Homework: construct point cloud sets and their H-distances using this Matlab function and comment the results.

Wasserstein distance computation

- Matlab: https://github.com/nklb/wasserstein-distance
- Python:

Compute the Wasserstein distance between two three-dimensional samples, each with two observations.

```
>>> from scipy.stats import wasserstein_distance_nd
>>> wasserstein_distance_nd([[0, 2, 3], [1, 2, 5]], [[3, 2, 3], [4, 2, 5]])
3.0
```

Compute the Wasserstein distance between two two-dimensional distributions with three and two weighted observations, respectively.

```
>>> wasserstein_distance_nd([[0, 2.75], [2, 209.3], [0, 0]],
...                          [[0.2, 0.322], [4.5, 25.1808]],
...                          [0.4, 5.2, 0.114], [0.8, 1.5])
174.15840245217169
```

This is 1-WSD for n-dimensional distributions. The last line represents the weights for each set

Example of bottleneck distance

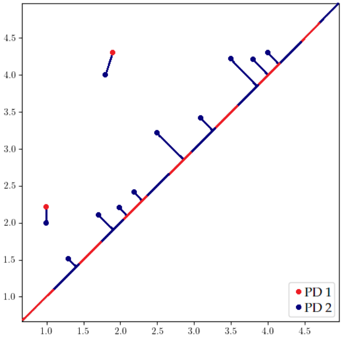


Figure: We show two different PDs overlapped, consisting of Δ plus 2 points in red and 11 points in blue, respectively. First, to apply the definition (4), we need two sets with the same cardinality. For this aim, it is necessary to add points of Δ, more precisely the orthogonal projection onto the diagonal of the 9 blue points closer to it, to reach 11. Lines between points and Δ represent the bijection that realizes the best matching between points in definition (4).

Basic python code for the BND of 2 simple diagrams

```
import matplotlib.pyplot as plt
import numpy as np

diag1 = [[2.7, 3.7], [9.6, 14.0], [14.2, 14.974], [3.0, float("Inf")]]
diag2 = [[2.8, 4.45], [9.5, 14.1], [15.2, 10.1], [3.2, float("Inf")]]
da1=np.array(diag1); da2=np.array(diag2)

message = "diag1=" + repr(diag1); print(message)
message = "diag2=" + repr(diag2); print(message)
message = "Bottleneck distance approximation=" + repr(
    gudhi.bottleneck_distance(diag1, diag2, 0.1)); print(message)
message = "Bottleneck distance exact value=" + repr(
    gudhi.bottleneck_distance(diag1, diag2)); print(message)
```

Bottleneck distance approximation=0.722013466408238
Bottleneck distance exact value=0.75

Characterization

Proposition

Let X and Y be finite subset in a metric space (M, d_M). Then, the the Hausdorff and the bottleneck distances of the persistence diagrams $D(X, \varepsilon)$, $D(Y, \varepsilon)$ satisfy

$$d_B(D(X, \varepsilon), D(Y, \varepsilon)) \leqslant d_H(X, Y).$$

For any further details see, for example

Rotman J. J. An introduction to Algebraic Topology, Springer, **1988**.

Python libraries

- Gudhi: is a generic open source C++ library, with a Python interface, for Topological Data Analysis (TDA) and Higher Dimensional Geometry Understanding. The library offers state-of-the-art data structures and algorithms to construct simplicial complexes, compute persistent homology, show persistence diagrams and persistent barcodes, prune a filtration.

  `https://gudhi.inria.fr/`

- Ripser: it is a lean PH package for Python. Building on the blazing fast C++ Ripser package as the core computational engine, mainly it can visualize persistence diagrams and compute lower star filtrations on images,

  `https://ripser.scikit-tda.org/en/latest/`

Cont

- Giotto-tda: it is a high-performance topological machine learning toolbox in Python built on top of scikit-learn and is distributed under the GNU AGPLv3 license. It allows us to apply the theory of PH to a lot of different kind of data, such as points cloud data, images, graphs, and series as well as persistence Images, Betti curves and Persistence Landscapes.

  `https://github.com/giotto-ai/giotto-tda`

- Dionysus: it is a computational topology package focused on persistent homology. It is written in C++, with Python bindings. It may compute filtration, PH, and distances among PD and plot the results into PDs.

  `https://github.com/nonabelian/tda_dionysus`

- DIPHA: It stands for (a Distributed Persistent Homology Algorithm). This C++ software package computes persistent homology. Besides supporting parallel execution on a single machine, DIPHA may also be run on a cluster of several machines using MPI.

  `https://github.com/DIPHA/dipha`

Code example

```
from gudhi.datasets.generators import points
import ripser
import matplotlib.pyplot as plt
import numpy as np
from mpl_toolkits import mplot3d

# Create 300 random points of a sphere with radius 1
sphere_points = points.sphere(n_samples = 300, ambient_dim = 3,
radius = 1, sample = "random")
# Compute persistent features using ripser library
ripsobj = ripser.Rips(maxdim=2)
dgms = ripsobj.fit_transform(sphere_points)

# plot the points on the sphere
a=np.array(sphere_points)
x = a[0:299,0]; y = a[0:299,1]; z = a[0:299,2]
fig = plt.figure(figsize = (10,10))
ax = plt.axes(projection='3d'); ax.grid()
ax.scatter(x, y, z, color = 'blue', marker='o')
ax.set_title('3D Scattered points on the sphere')

# Set axes label
ax.set_xlabel('x', labelpad=20); ax.set_ylabel('y', labelpad=20); ax.set_zlabel('z', labelpad=20)
plt.show()

# Plot the corresponding PD
ripsobj.plot(dgms)
plt.show()
```

Figures

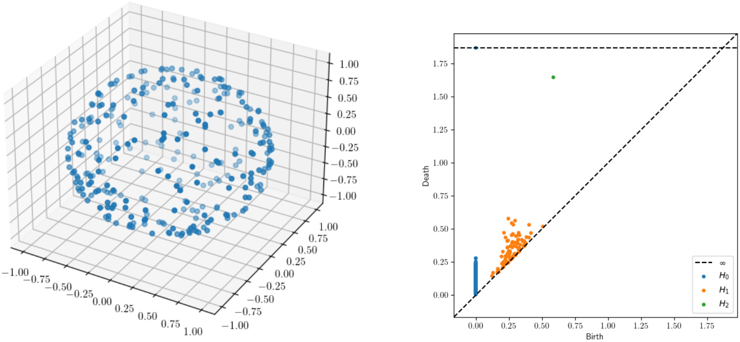


Figure: Points sampled from the sphere (left) and an example of PD (right)

Part III

Applications to Classification and Intrinsic Dimension Estimation

Preamble

Classification is a relevant task (big data to store, make accessible, analyzed) in fields like medicine, economics, psychology, image analysis/processing, etc...

Example: e-mail SPAM

For instance, one wants to provide an algorithm able to filter out if an incoming e-mail is SPAM or not. During the so called Training Phase, the algorithm analyzes a group of e-mails labeled as SPAMs and a group of regular ones in order to find out patterns and features that can make it able to distinguish them. This set of examples is known as Training Set. After that, the algorithm can predict, hopefully in a satisfactory manner, if a new incoming e-mail is SPAM or not. This is the case of the supervised classification problem.

Classification: history and (most) used tools

- This task takes its origin some time ago with the K-Nearest Neighbors (KNN) algorithm, developed in 1951 by Fix and Hodges.

Fix, Evelyn; Hodges, Joseph L. (1951). **Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties** (PDF) (Report). USAF School of Aviation Medicine, Randolph Field, Texas. Archived (PDF) from the original on September 26, 2020.

- During the years, a lot of different methods and variants have been developed: the most famous are: K-th Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree (DT), and Random Forest, only to name a few.
- We focus mainly on KNN and SVM.

Useful notation

- Let $\Omega \subset \mathbb{R}^d$ and consider two subsets, $\mathcal{X}_l = \{\mathbf{x}_1, ..., \mathbf{x}_m\}$, of labeled points and $\mathcal{X}_u \subset \Omega$ be set of unlabelled ones.
- Let $Y_l = \{y_1, \ldots, y_m\}$ be the set of corresponding labels of $\mathcal{X}_l$ where $y_i \in L = \{l_1, \ldots, l_s\}$, the set of labels or classes.
- The set of couples $\{(\mathbf{x}_i, y_i)\}_{1 \le i \le m}$ is the training set while $\mathcal{X}_u$ is called the test set.
- We denote their union as $\mathcal{X}_{lu} = \mathcal{X}_l \cup \mathcal{X}_u$.
- If $L = \{-1, 1\}$, it is called binary problem.

KNN idea

"Similar points are closer to each other."

To determine the belonging class of a new point, the only thing to do is to infer such a prediction by analyzing its neighbors.

KNN search

- Fix $k \in \mathbb{N}$.
- Consider $x \in X_u$ and the k points in X_l that are closer to it using a prescribed distance (for ex: Euclidean norm, sup norm, Manhattan distance[a]).
- Once extracted these k points, it assigns to x the most recurrent label among them.

[a] This is also known as taxicab distance, i.e. the distance between to points in a grid-like path

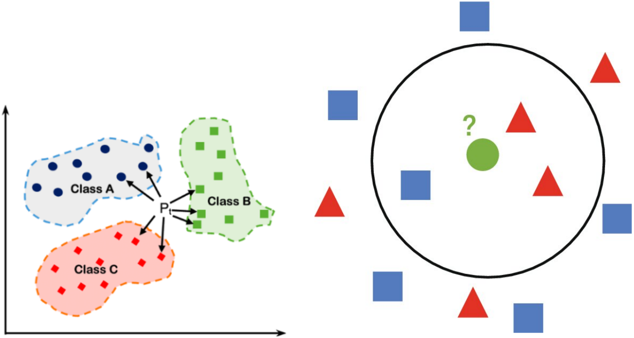


Figure: The idea of similarity of points (left) and how KNN works with $k = 3$ (right)

The test sample (green dot) should be classified either to blue squares or to red triangles. If $k = 3$ (solid line circle) it is assigned to the red triangles because there are 2 triangles and only 1 square inside the inner circle.
Homework. If $k = 5$ is it assigned to the blue squares or to the red triangles?

Variant of KNN: ANN

- Approximate Nearest Neighbors (ANN): can be used to speed up the computation of by reducing the number of pairwise comparisons needed (for instance: fix a query distance, takes the one(s) that is(are) c-times this query)
- ANN is particularly useful in high-dimensional spaces, which are common in modern ML-AI applications. In high dimensions, it needs a dimension reduction pre-prossesing (for instance by PCA).
- The algorithms behind the search are, among them, Hashing-based methods, Tree-based methods, Greedy-search in the proximity graph,...

Notice: K-nearest neighbors (KNN) sits between NN and ANN by giving faster results while maintaining high accuracy.

KNN by Persistent Homology

Kindelan R.; Frías J.; Cerda M.; Hitschfeld N. A topological data analysis based classifier. Advances in Data Analysis and Classification **2024**, Issue 2/2024

developed a new technique that infers labels exploiting the structure of data given by simplicial complexes.

The authors called the method Link-based label propagation function and the goal is to define a proper label function that allows to associate the right label to an unlabeled point.

How to construct the simplicial complexes and then filtrate them?

The authors called selectors the methods for chosing suitable simplicial complexes.

We refer to this approach as Global TDA.

KNN by Persistent Homology

Kindelan R.; Frías J.; Cerda M.; Hitschfeld N. A topological data analysis based classifier. Advances in Data Analysis and Classification **2024**, Issue 2/2024

developed a new technique that infers labels exploiting the structure of data given by simplicial complexes.

The authors called the method Link-based label propagation function and the goal is to define a proper label function that allows to associate the right label to an unlabeled point.

How to construct the simplicial complexes and then filtrate them?

The authors called selectors the methods for chosing suitable simplicial complexes.

We refer to this approach as Global TDA.

Some selectors

Let P_K be the set of all persistent features $p = (b, d)$ and $pers(p) := d - b$ the lifetime of p

- **AVG**:

$$p_{avg} = (\bar{b}, \bar{d}) := \min_{p \in P_K} |\text{pers}(p) - \text{avg}|$$

  where **avg** is the average of all pers(p) within P_K.

- **HAVG**:

$$p_{havg} = (\bar{b}, \bar{d}) := \min_{p \in P_K} |\text{pers}(p) - \text{havg}|$$

  where **havg** is the harmonic mean that, for a set of positive numbers $\{x_1, \ldots, x_n\}$ is $\text{havg}(x_1, \ldots, x_n) = \dfrac{n}{\frac{1}{x_1} + \cdots + \frac{1}{x_n}} = 1/avg(1/x_1 \cdots, 1/x_n)$

- **MAX**:

$$p_{max} = (\bar{b}, \bar{d}) := \max_{p \in P_K} \text{pers}(p)$$

Cont

- **MEDIAN**:

$$p_{med} = (\bar{b}, \bar{d}) := \min_p |\mathrm{pers}(p) - \mathrm{median}|$$

  where **median** is the median of all pers(p) with $p \in P_K$ (**median** requires the ordering of the points and it is assumed at the position $(n+1)/2$)
- **RANDOM**: $p_{random} = (\bar{b}, \bar{d})$ is chosen uniformly at random among all persistent features $p \in P_K$

After choosing one of the previous options, that is $p \in \{p_{max}, p_{random}, p_{med}, p_{avg}, p_{havg}\}$, the selected simplicial complex turns out to be $K_i = f^{-1}((-\infty, \bar{d}))$.

Association function

Let $\mathbf{E} = span\{\mathbf{e}_1, \ldots, \mathbf{e}_s\}$. The association function $\phi : \mathcal{X}_l \to \mathbf{E}$ is defined at a vertex (or 0-dimensional simplex) $v \in \mathcal{X}_l$, as $\phi(v) = \mathbf{e}_s$ for $v \in \mathcal{X}_l$, $\mathbf{0}$ ($\in \mathbb{R}^s$) otherwise. Then, its "extension to any simplex in $\sigma \in K$ is given by

$$\Phi(\sigma) = \sum_{v \in \sigma} \phi(v)$$

Cont

- **MEDIAN**:

$$p_{med} = (\bar{b}, \bar{d}) := \min_p |\mathrm{pers}(p) - \mathrm{median}|$$

where **median** is the median of all pers(p) with $p \in P_K$ (**median** requires the ordering of the points and it is assumed at the position $(n+1)/2$)

- **RANDOM**: $p_{random} = (\bar{b}, \bar{d})$ is chosen uniformly at random among all persistent features $p \in P_K$

After choosing one of the previous options, that is $p \in \{p_{max}, p_{random}, p_{med}, p_{avg}, p_{havg}\}$, the selected simplicial complex turns out to be $K_i = f^{-1}((-\infty, \bar{d}))$.

Association function

Let $\mathbf{E} = span\{\mathbf{e}_1, \ldots, \mathbf{e}_s\}$. The association function $\phi : \mathcal{X}_l \to \mathbf{E}$ is defined at a vertex (or 0-dimensional simplex) $v \in \mathcal{X}_l$, as $\phi(v) = \mathbf{e}_s$ for $v \in \mathcal{X}_l$, $\mathbf{0}$ ($\in \mathbb{R}^s$) otherwise. Then, its "extension to any simplex in $\sigma \in K$ is given by

$$\Phi(\sigma) = \sum_{v \in \sigma} \phi(v)$$

Extension function

To address the problem of how to assign a label to an unlabeled point is done by introducing the extension function

Extension function

$\Psi : \mathcal{X}_u \to \mathbf{E}$ defined on a point $\mathbf{x} \in \mathcal{X}_u$ is

$$\Psi(\mathbf{x}) = \sum_{\sigma \in Lk_{K_i}(\{\mathbf{x}\})} w(\mathbf{x}, \sigma)\Phi(\sigma) = \sum_{\sigma \in St_{K_i}(\{\mathbf{x}\})} w(\mathbf{x}, \sigma \setminus \{\mathbf{x}\})\Phi(\sigma \setminus \{\mathbf{x}\}) = \sum_{j=1}^{s} a_j \mathbf{e}_j$$

with proper definition of weight function w, where Lk denotes the Link and St the Star (see cited the paper)

Finally: the label l_j corresponding to the highest coefficient a_j in the previous sum is the label of the point x.

cont

Remarks

- As for KNN, the label of $x \in \mathcal{X}_u$ is directly influenced by those of its neighbors. The method is a generalization of the KNN idea to the structure of simplicial complexes, where the concept of neighborhood is replaced by that of Lk (Link).
- To run the algorithm is essential to define the weight function w: points closer to the point $x \in \mathcal{X}_u$ influence more the prediction of its label. Here closer means w.r.t a distance or that along the filtration they live in some simplices born earlier.

Cont

The proposed weight is

$$w(x,\sigma) = \frac{\frac{1}{f(\sigma \cup \{x\})^2}}{\sum_{\mu \in St_{\mathcal{K}_i}(\{x\})} \frac{1}{f(\mu)^2}}$$

After some calculations (see the cited paper), we get the final expression for Ψ:

$$\Psi(\mathbf{x}) = \sum_{\sigma \in Lk_{K_i}(\{x\})} \frac{\Phi(\sigma)}{f(\sigma \cup \{x\})^2}$$

Local TDA

- To be able to use Global TDA with all datasets, we had to reduce the number of simplices or better take only the simplices with dimensions up to a certain value max_dim
- To determine the label of $x \in \mathcal{X}_u$ we suggest to consider only a cluster of K points centered on x and then apply Global TDA only to this small dataset, local dataset
- As for KNN, the local TDA depends on a particular parameter κ, that allows to make a zoom of the dataset restricting the number of points to consider for computations (in KNN, the k denotes the number of points that one decides to consider as significant and more influential for determining the final label).

Machine-learning pipeline

- Data are commonly divided into three groups:: **Training**, **Validation**, and **Test** Sets.
- The standard procedure consists of splitting the whole dataset into ratios (depending on some factors). Generally, a standard split is 60-80% for Training data, 10-20% for Validation data, and $10 - 20\%$ for Test data

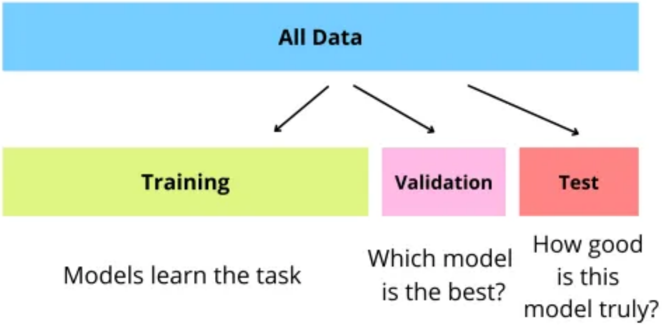


Figure: Example of Machine Learning Workflow

Class imbalance

- Class imbalance occurs when the number of samples in different classes is significantly unequal
- In classification from real-world scenarios, which usually has only two classes. Examples are: fraud, claim, spam detection, disease diagnosis that bring severe imbalance/biased datasets.
- In the binary case, once defined and set the model, one ends up with the confusion matrix that collects all information about the classification performances of the model itself, where Actual is the correct and real labels while Predicted collects values assigned by the model.

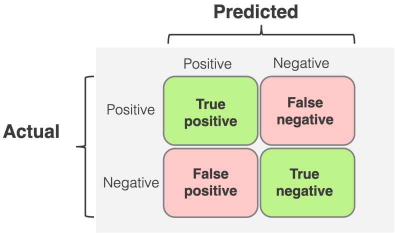


Figure: Confusion Matrix

Imbalance Ratio

In application, the 4 possible groups are denoted by the initials: TP, TN, FN, FP.

- A dataset is balanced if it has equal samples per class. A measure of prediction its quality is Accuracy,

$$\text{Accuracy} := \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$$

- A dataset is imbalanced when there is significant, or in some cases extreme disproportion among the number of examples of each class of the problem. The class or classes with abundant examples are called major or majority class, whereas the class with few examples is called minor or minority class.

IR definition

The Imbalance Ratio (IR) in binary datasets, is defined as the

$$IR := \frac{Card(major\ class)}{Card(minor\ class)}$$

Other metrics

- Binary scenario (2 classes)

$$\text{Recall} = \frac{TP}{TP+FN}$$

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{F1-score} = \frac{2 * \text{ Precision } * \text{ Recall}}{\text{Precision} + \text{Recall}}.$$

- Multiclass scenario (n classes) other metrics are used. For example, Balanced Accuracy, it considers the number of correct predictions per class, calledrecall, and then takes the average. More precisely

$$\text{Recall}_i = \frac{\text{test samples of class } i \text{ correctly classified}}{\text{all test samples of class i}}$$

$$\text{Balanced\_Accuracy} = \frac{\sum_{i=1}^{n} \text{Recall}_i}{n}.$$

Experiment on various datasets

Dataset	# samples	# classes	IR
CIRCLES	50	2	25:25
IRIS	150	3	50:50
WINE	178	3	71:48
MOON	200	2	100:100
SURGERY	470	2	400:70
CANCER	570	2	357:213
LIVER	580	2	413:167
DIAB. RET.	1080	2	540:540
RICE	3260	2	1630:1630

Table: Datasets for classification. In red the imbalanced datasets

Cont

Dataset	AVG	HAVG	MAX	MEDIAN	RANDOM
CIRCLES	0.504	**0.529**	**0.529**	0.488	0.488
IRIS	0.936	**0.961**	0.936	0.947	0.936
WINE	**0.967**	0.946	0.946	0.953	0.951
MOON	0.513	**0.564**	0.516	0.515	0.526
SURGERY	0.479	0.518	0.497	0.489	**0.525**
CANCER	0.944	0.946	**0.952**	0.942	0.949
LIVER	0.564	**0.598**	0.565	0.571	0.579
DIAB. RET.	**0.628**	0.614	0.607	0.612	0.611
RICE	0.904	0.900	**0.915**	0.906	0.909

Table: Accuracy or Balanced_Accuracy of Local TDA classifier related to different datasets (best values in **bold**)

Remark: the choice of the selector does not affect too much the model. The best selectors are HAVR and MAX

Comparison with classical data analysis methods

We take into account here the most three famous baseline methods, such as KNN, DT and SVM.

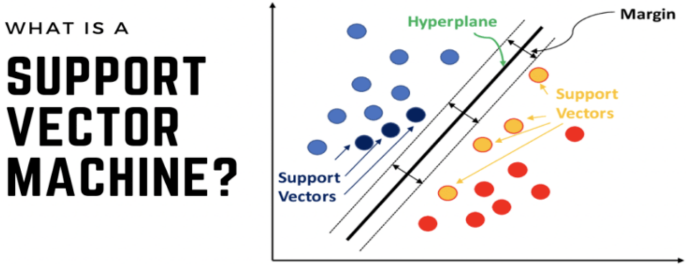
- **KNN**: the hyperparameter k represents the number of neighbors to consider at each iteration of the method: `n_neighbors` is taken among $\{1, 2, \ldots, 50\}$ and as method or `algorithm` used to compute the nearest neighbors we consider `ball_tree, kd_tree, brute`
- **DT**: for `criterion`, namely the function to measure the quality of a split, we take into account `gini, entropy, log_loss`
- **SVM**: we choose `kernel` among `linear, poly, rbf` that are equivalent to the linear kernel, the polynomial kernel of some degree, and Gaussian RBF with C=1 (the shape parameter), as commonly done.

Cont

Dataset	Global	Local	KKN	DT	SVM
CIRCLES	**0.529**	**0.529**	0.383	0.396	0.296
IRIS	0.961	0.961	0.95	0.912	**0.967**
WINE	0.967	0.967	0.976	0.904	**0.977**
MOON	0.539	**0.564**	0.531	0.548	0.462
SURGERY	0.504	0.525	0.482	0.498	**0.578**
CANCER	0.948	0.952	0.959	0.918	**0.962**
LIVER	0.592	0.598	0.568	0.599	**0.701**
DIAB. RET.	0.617	0.628	0.664	0.628	**0.696**
RICE	0.921	**0.935**	0.929	0.889	0.927

Table: Comparison in terms of Accuracy or Balanced_accuracy between methods and datasets (in **bold** global best score, in red the best one among topological methods)

Support Vector Machine (SVM)



In words: the SVM (binary) classification, larger the margin between the hyperplane and the closest point turns out to be, the higher the classification is shown by the model

Binary supervised learning

- Let $\Omega \subset \mathbb{R}^d$ and $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset \mathcal{X} \subset \Omega$ be the set of input data with $d, m \in \mathbb{N}$. We have a training set, composed of the couples $(\mathbf{x}_i, y_i)$ with $i = 1, \ldots, m$ and $y_i \in \mathcal{Y} = \{-1, 1\}$.
- The binary supervised learning task consists in finding a function $f : \Omega \longrightarrow \mathcal{Y}$, the model, such that it can predict, in a satisfactory way, the label of an unseen $\tilde{\mathbf{x}} \in \Omega \setminus \mathcal{X}$.

The Support Vectors Algorithm, is an optimization approach proposed

Scholkopf B.; Smola A.J. **Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond**. *The MIT Press* **2002**, ISBN: 978-026-225-693-3

The original formulation

The SVM optimization problem is given by

$$\max_{\alpha \in \mathbb{R}^m} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle$$

$$\text{s. to} \quad \sum_{i=1}^{m} \alpha_i y_i = 0$$

$$0 \le \alpha_i \le \frac{C}{m} \; \forall i = 1, \ldots, m$$

$\alpha_i > 0$ are called Support Vectors.
Remark: if data are NOT linearly separable, it is better to introduce **kernels**.

Through the kernel trick

If $\mathcal{X}$ is a general set (not a subset of $\mathbb{R}^d$), without any structure, the previous theory holds with kernels.

Q & A

Q: how an unseen pattern $\mathbf{x}$ is "similar" to one in our training pattern? A: we introduce a kernel κ

$$\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$$

$$(x, \bar{x}) \mapsto k(x, \bar{x})$$

that is a function that returns a real number characterizing the similarity between x and $\bar{x}$. A simple case is the dot product,

$$\langle x, \bar{x} \rangle = \sum_{i=1}^{m} x_i \bar{x}_i .$$

- We need to represent the patterns as vectors in some space $\mathcal{H}$ with a dot product. The map $\Phi : \mathcal{X} \to \mathcal{H}, \;\; x \mapsto \mathbf{x}$ where $\mathbf{x}$ denotes the vector. The space $\mathcal{H}$ (an Hilbert space) is called the feature space.

- Φ can be a nonlinear map and of the form

$$\Phi(x) = \kappa(\cdot, x)$$

with $\mathcal{H}$ the RKHS related to kernel κ. Thus $\mathcal{H}$ is explicitly equal to the space of functions $\mathbb{R}^{\mathcal{X}} = \{f : \mathcal{X} \to \mathbb{R}\}$ and the relation between this one and $\mathcal{X}$ is given by

$$\Phi : \mathcal{X} \to \mathbb{R}^{\mathcal{X}}$$
$$x \mapsto \kappa(\cdot, x).$$

- We assume κ is positive definite.
- Through Φ we embed patterns into a vector space,feature space, the we would like to be RKHS

$$\mathcal{F} = \left\{ f \mid f = \sum_{i=1}^{m} \alpha_i \kappa(\cdot, x_i),\ m \in N,\ x_i \in \mathcal{X}, \alpha_i \in \mathbb{R}, \right\}.$$

This space is equipped with an inner product defined as

$$\begin{aligned} f(x) &= \sum_{i=1}^{m} \alpha_i \kappa(x, x_i), \;\; g(x) = \sum_{j=1}^{m} \beta_i \kappa(x, \bar{x}_j) \\ \langle f(x), g(x) \rangle &:= \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \beta_j \kappa(x_i, \bar{x}_j) \end{aligned}$$

with $f, g \in \mathcal{F}$, $x_1, \ldots, x_m$ and $\bar{x}_1, \ldots, \bar{x}_m$ two sets of patterns chosen in $\mathcal{X}$

Theorem

A function κ defined on $\mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a reproducing kernel if and only if there exists a Hilbert space $\mathcal{H}$ and a mapping $\Phi : \mathcal{X} \to \mathcal{H}$, such that for all $x, \bar{x} \in \mathcal{X}$

$$\kappa(x, \bar{x}) = \langle \Phi(x), \Phi(\bar{x}) \rangle_{\mathcal{H}}$$

This formula states the equivalence between a kernel evaluation and a dot product of feature maps referred to as Kernel Trick in the machine learning literature.

SVM formulation using kernels

Hence, by the feature map, the kernel is defined as $\kappa(x, \bar{x}) := \langle \Phi(x), \Phi(\bar{x}) \rangle_{\mathcal{H}}$ and the new SVM problem becomes

New SVM optimization problem

$$\max_{\alpha \in \mathbb{R}^m} \quad \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j \kappa(x_i, x_j)$$

$$\text{s. to} \quad \sum_{i=1}^{m} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq \frac{C}{m} \ \forall i = 1, \ldots, m$$

Remark: if x_i are Persistent Diagrams, κ is called a Persistence Kernel

Section 6

Persistence Kernels

Persistent Scale Space Kernel (PSSK)

Idea: Compute feature map as the solution of a PDE

The PDE for Persistence Scale Space Kernel

Let $\Omega_{ad} = \{\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2 : x_2 \geqslant x_1\}$ and let $\delta_{\mathbf{x}}$ denote a Dirac delta centered at $\mathbf{x}$. For a given persistence diagram D, we consider the solution $u : \Omega_{ad} \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}$ such that $(\mathbf{x}, t) \mapsto u(\mathbf{x}, t)$ of the heat equation:

$$\begin{aligned} \Delta_{\mathbf{x}} u = \partial_t u \quad & \text{in } \Omega_{ad} \times \mathbb{R}_{\geqslant 0} \\ u = 0 \quad & \text{on } \partial\Omega_{ad} \times \mathbb{R}_{\geqslant 0} \\ u = \sum_{\mathbf{y} \in D} \delta_{\mathbf{y}} \quad & \text{on } \Omega_{ad} \times 0 \end{aligned}$$

PSSK

The feature map $\Phi_\sigma : \mathfrak{D} \to \mathrm{L}^2(\Omega_{ad})$ at scale $\sigma > 0$ of a persistent diagram $D \in \mathfrak{D}$ is defined as $\Phi_\sigma(D) = u\big|_{t=\sigma}$. This map yields the Persistence Scale Space Kernel[2] (**PSSK**) K_σ on $\mathfrak{D}$ as:

$$K_\sigma(D, E) = \langle \Phi_\sigma(D), \Phi_\sigma(E) \rangle_{\mathrm{L}^2(\Omega_{ad})}.$$

$$\boxed{K_\sigma(D, E) = \frac{1}{8\pi\sigma} \sum_{\mathbf{y}\in D, \mathbf{z}\in E} \exp\left(-\frac{\|\mathbf{y}-\mathbf{z}\|^2}{8\sigma}\right) - \exp\left(-\frac{\|\mathbf{y}-\bar{\mathbf{z}}\|^2}{8\sigma}\right)}$$

where $\mathbf{z} = (a, b)$, $\bar{\mathbf{z}} = (b, a)$, for any $D, E \in \mathfrak{D}$.

[2] J. Reininghaus, S. Huber, U. Bauer, R. Kwitt *A Stable Multi-Scale Kernel for Topological Machine Learning*, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4741-4748 (2015)

Persistent Weighted Gaussian Kernel (PWGK)

Idea: Replace each PD with a measure

- Consider a strictly positive definite Gaussian kernel, e.g. $\kappa_G(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}$, $\sigma > 0$ and its RKHS space $\mathcal{H}_{\kappa_G}$
- Let $\Omega \subset \mathbb{R}^d$, $M_b(\Omega)$ the space of finite signed Radon measures
- Let $E_{\kappa_G} : M_b(\Omega) \to \mathcal{H}_{\kappa_G}, \mu \mapsto \int_\Omega \kappa_G(\cdot, x) d\mu(x)$

For any persistence diagram $D \in \mathfrak{D}$, if $\mu_D^w = \sum_{x \in D} w(x)\delta_x$, where $w(x) > 0$ for all $x \in D$

$$\boxed{E_{\kappa_G}(\mu_D^w) = \sum_{x \in D} w(x)\kappa_G(\cdot, x).}$$

PWGK

The Persistence Weight Gaussian Kernel[3] (**PWGK**) is defined as

$$K_G^w(D,E) = \exp\left(-\frac{1}{2\tau^2} \| E_{\kappa_G}(\mu_D^w) - E_{\kappa_G}(\mu_E^w) \|^2_{\mathcal{H}_{\kappa_G}} \right), \; \tau > 0$$

for any $D, E \in \mathfrak{D}$.

[3] G. Kusano, K. Fukumizu, Y. Hiraoka, *Kernel method for persistence diagrams via kernel embedding and weight factor*, The Journal of Machine Learning Research vol. 18(1) (2017), pp. 6947-6987

Sliced Wasserstein Kernel (SWK)

Let consider μ and ν two nonnegative measures on $\mathbb{R}$ such that $\mu(\mathbb{R}) = r = |\mu|$ and $\nu(\mathbb{R}) = r = |\nu|$, let consider the 1-Wasserstein distance for nonnegative measures

$$\mathcal{W}(\mu,\nu) = \inf_{P\in\Pi(\mu,\nu)} \int\int_{\mathbb{R}\times\mathbb{R}} |x-y|dP(x,y)$$

with $\Pi(\mu,\nu)$ is the set of measures in $\mathbb{R}^2$ with marginals μ and ν

Definiton [Sliced Wasserstein distance]

Given $\theta \in \mathbb{R}^2$ with $\|\theta\|_2 = 1$, let $L(\theta)$ denote the line $\{\lambda\theta|\lambda\in\mathbb{R}\}$ and let $\pi_\theta : \mathbb{R}^2 \to L(\theta)$ be the orthogonal projection onto $L(\theta)$. Let $D, E \in \mathfrak{D}$ and let $\mu_D^\theta := \sum_{p\in D}\delta_{\pi_\theta(p)}$ and $\mu_{D\Delta}^\theta := \sum_{p\in D}\delta_{\pi_\theta\circ\pi_\Delta(p)}$ and similarly for μ_E^θ and $\mu_{E\Delta}^\theta$ where π_Δ is the orthogonal projection onto the diagonal. Then, the Sliced Wasserstein distance is

$$SW(D,E) = \frac{1}{2\pi}\int_{\mathbb{S}^1} \mathcal{W}(\mu_D^\theta + \mu_{E\Delta}^\theta, \mu_E^\theta + \mu_{D\Delta}^\theta)d\theta$$

with $\mathbb{S}^1$ the unit circle

SWK

Thus, the Sliced Wasserstein Kernel[4] (**SWK**) is defined as

$$\boxed{K_{SW}(D,E) := \exp\left(-\frac{SW(D,E)}{2\sigma^2}\right),\ \sigma > 0}$$

for any $D, E \in \mathfrak{D}$.

[4] M. Carriere, M. Cuturi, S. Oudot, *Sliced Wasserstein kernel for persistent diagrams*, International Conference on Machine Learning, PMLR 2017, pp.664-673

Persisten Fisher Kernel (PFK)

Idea: Replace each PD with a probability distribution
So if $D \in \mathfrak{D}$

$$\rho_D(x) := \frac{1}{Z} \sum_{u \in D} N(x; u, \sigma I)$$

where N is a gaussian function, $Z = \int \sum_{u \in D} N(x; u, \sigma I) dx$ and I is the identity matrix.
The probability simplex is $\mathbb{P} = \{\rho | \int \rho(x) dx = 1, \rho(x) \geq 0\}$.

Definition [Fisher Information Metric for probability distributions]

Given two element in $\rho_i, \rho_j \in \mathbb{P}$, the Fisher Information Metric is

$$d_{\mathbb{P}}(\rho_i, \rho_j) = \arccos\left(\int \sqrt{\rho_i(x) \rho_j(x)} dx \right).$$

PFK

Definition [Fisher Information Metric for PD]

Let D, E be two finite and bounded persistence diagrams. The Fisher Information Metric between D and E, is defined as

$$d_{FIM}(D,E) := d_{\mathbb{P}}(\rho_{D\cup E_\Delta}, \rho_{E\cup D_\Delta})$$

where $D_\Delta := \{\Pi_\Delta(u) | u \in D\}$ and Π_Δ is the projection on the diagonal $\Delta = \{(a,a) | a \geq 0\}$.

The Persistence Fisher Kernel[5] (**PFK**) is then defined as

$$\boxed{K_F(D,E) := \exp(-t d_{FIM}(D,E)),\ t > 0, \text{ for any } D, E \in \mathfrak{D}.}$$

[5] T. Le, M. Yamada, *Persistence fisher kernel: A Riemannian manifold kernel for persistence diagrams*, arXiv preprint arXiv:1802.03569 (2018)

Persistent Image (PI)

If $D \in \mathfrak{D}$ we introduce a change of coordinates, $T : \mathbb{R}^2 \to \mathbb{R}^2$ given by $T(x, y) = (x, y - x)$ and let $T(D)$ be the transformed multiset in first-persistence coordinates. Let g_u be the 2-dimensional Gaussian with mean u and variance σ^2, defined as

$$g_u(x, y) = \frac{1}{2\pi\sigma^2} e^{-[(x-u_x)^2+(y-u_y)^2]/2\sigma^2},$$

Fix a weight function $f : \mathbb{R}^2 \to \mathbb{R}$, $f \geq 0$.
For instance, $f(x, y) = w_b(y)$

$$w_b(t) = \begin{cases} 0 & \text{if } t \leq 0, \\ \frac{t}{b} & \text{if } 0 < t < b, \\ 1 & \text{if } t \geq b. \end{cases}$$

cont

Definition [Persistent Surface]

Given $D \in \mathfrak{D}$, the corresponding persistence surface $\rho_D : \mathbb{R}^2 \to \mathbb{R}$ is the function

$$\rho_D(x, y) = \sum_{u \in T(D)} f(u) g_u(x, y).$$

If we divide the plane in a grid with n^2 pixels $(P_{i,j})_{i,j=1,\dots,n}$

Persistent Image

Given $D \in \mathfrak{D}$, its Persistence Image is the collection of pixels

$$PI(\rho_D)_{i,j} = \int\int_{P_{i,j}} \rho_D(x, y) dxdy.$$

PI

Thus, through persistence image, each persistence diagram is turned into a vector $PIV \in \mathbb{R}^{n^2}$ that is $PIV(D)_{i+n(j-1)} = PI(D)_{i,j}$, then it is possible to introduce the following Persistent Image Kernel (**PI**)

$$\boxed{K_{PI}(D,E) = < PIV(D), PIV(E) >_{\mathbb{R}^{n^2}} .}$$

Numerical tests

1. Simplicial complexes and persistence diagrams, by Python libraries: `gudhi`, `ripser`, `giotto-tda` and `persim` .
2. SVM by the `Scikit` library of Python.
3. We performed a random splitting (70%/30%) for training and testing and applied a 10-fold cross-validation on the training set for the hyperparameters tuning. Then we averaged the results over 10 runs.
4. For PFK, we precomputed the Gram matrices using a Matlab (Matlab R2023b) routine because it is faster than the Python one. The values for C belong to $\{0.001, 0.01, 0.1, 1, 10, 100\}$.
   - **PSSK**: $\sigma \in \{0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10\}$
   - **PWGK**: $\tau \in \{0.001, 0.01, 0.1, 1, 10, 100\}$, $\rho \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$, $p \in \{1, 5, 10, 50, 100\}$, $C_w \in \{0.001, 0.01, 0.1, 1\}$ and we chose the Gaussian one.
   - **SWK**: $\eta \in \{0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10\}$
   - **PFK**: $\sigma \in \{0.001, 0.01, 0.1, 1, 10\}$ and $t \in \{0.1, 1, 10, 100, 1000\}$
   - **PI**: $\sigma \in \{0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1, 1, 10\}$ and number of pixel 0.1.

Bandiziol C.; De Marchi S.: Persistence Symmetric Kernels for Classification: A Comparative Study. *Symmetry* (2024), *16, 1236*

Results

Data sets: SHREC14 (human models of different body shapes and 20 poses) DYN SYS (Dynamical System of 2 Odes), MNIST (images of handwritten digits), BZR (collection of chemical compounds), DISTAL (?)

Kernel	DYN SYS	MNIST	BZR	DISTAL
PSSK	0.829	0.729	0.557	0.658
PWGK	0.819	0.754	0.655	0.696
SWK	**0.841**	**0.802**	**0.712**	**0.723**
PFK	0.784	0.734	0.682	0.676
PI	0.777	0.760	0.585	0.662

Table: Accuracy or balanced accuracy related to several datasets

Kernel	SHREC14	BZR	DISTAL
PSSK	582	11731	49481
PWGK	1841	3751	43152
SWK	**209**	**321**	**1418**
PFK	319	814	11405
PI	266	640	1825

Table: Computational costs (seconds) marked in **bold** the best values

Intrinsic Dimension (ID): main definitions and concepts

References follows the draft

Adams H.; Aminian M.; Farnell E.; Kirby M.; Peterson C.; Mirth J.; Neville R.; Shipman P.; Shonkwiler C. A Fractal Dimension for Measures via Persistent Homology. (eds) Topological Data Analysis *Abel Symposia Springer* **2020**, *15*, ISBN: 978-3-030-43407-6

Definitons

Definition of ID

The intrinsic dimension (shortly, ID) is the minimum number of local coordinates needed to describe the data

Other way

A dataset $\Omega \subset \mathbb{R}^D$ is said to have Intrinsic Dimension (ID) equal to d if its elements lie entirely, without loss of information, within a d - dimensional manifold $\mathcal{M}$ of $\mathbb{R}^D$, where $d < D$.

- Knowing the value of the ID is critical to ensure the reliability low-dimensional data visualization and the validity of dimensionality reduction as a data preprocessing step.
- In addition, the ID is often a very useful metric per se, allowing the analyst to capture key information about the geometry of the data compare data and models and track temporal variations of complexity.

The ID is generally not known a priori: we can get ID estimates directly from the data.

Classical estimators method

- Projective. Example, the PCA (project to the space spanned by the first significant d eigenvectors of the covariance matrix)
- Geometric-statistical. Example: the Correlation Dimension, the number of point within $B(r;x)$ scales as $N_r \approx r^d$, with d the ID.

$\hookrightarrow$ Both methods have limitations: large number of points when ID is high or fail in presence of highly non-uniformly/non-isotropic distributuions $\hookleftarrow$

We look for TOPOLOGICAL approaches for ID estimators

The ID is generally not known a priori: we can get ID estimates directly from the data.

Classical estimators method

- Projective. Example, the PCA (project to the space spanned by the first significant d eigenvectors of the covariance matrix)
- Geometric-statistical. Example: the Correlation Dimension, the number of point within $B(r;x)$ scales as $N_r \approx r^d$, with d the ID.

$\hookrightarrow$ Both methods have limitations: large number of points when ID is high or fail in presence of highly non-uniformly/non-isotropic distributuions $\hookleftarrow$

We look for TOPOLOGICAL approaches for ID estimators

Most famous fractal-based ID estimators

Main idea

Distances between points lying on a fractal or a low dimensional manifold follow scaling laws that depend on the ID of the set.

Box-Counting Dimension

Let X_N be a subset of $\mathbb{R}^D$, considered as a metric space, and let $N_\epsilon \ (\propto \epsilon^{d_{BC}})$ denote the infimum of the number of closed balls of radius ϵ required to cover X_N. Then the Box-Counting Dimension of Ω is

$$d_{BC} := \lim_{\epsilon \to 0} \frac{\log(N_\epsilon)}{\log(1/\epsilon)}$$

provided this limit exists.
If X_N is a I.I.D. (Indipendent and Identically Distributed) set of N points from a regular metric μ of $\mathbb{R}^D$ then $\lim_{N\to\infty} d_{BC} = d$.

Most famous fractal-based ID estimators

Main idea

Distances between points lying on a fractal or a low dimensional manifold follow scaling laws that depend on the ID of the set.

Box-Counting Dimension

Let X_N be a subset of $\mathbb{R}^D$, considered as a metric space, and let N_ϵ $(\propto \epsilon^{d_{BC}})$ denote the infimum of the number of closed balls of radius ϵ required to cover X_N. Then the Box-Counting Dimension of Ω is

$$d_{BC} := \lim_{\epsilon \to 0} \frac{\log(N_\epsilon)}{\log(1/\epsilon)}$$

provided this limit exists.
If X_N is a I.I.D. (Indipendent and Identically Distributed) set of N points from a regular metric μ of $\mathbb{R}^D$ then $\lim_{N\to\infty} d_{BC} = d$.

Correlation dimension

Main idea

It describes how the number of points within a certain distance (or radius) scales increasing it. Mathematically it is described using the correlation integral of a probability measure μ, i.e. the mean that the states at different times are close. Given a threshold $\epsilon > 0$

$$C(\epsilon) := \lim_{N \to \infty} \frac{1}{N^2} f \qquad (5)$$

where f is the number of pair (i, j) whose distance $\|x_i - x_j\| < \epsilon$ (usually described by the Heaviside step function $H(x) = 0$ for $x < 0$, $H(x) = 1$ for $x \geq 0$ (see below)).

As $\epsilon \to 0$, the correlation integral scales as

$$C(\epsilon) \approx \epsilon^{d_C} \qquad (6)$$

with d_C also known as the correlation exponent.

Cont

The correlation dimension can be estimated from finite sets of points, sufficiently large and evenly distributed, i.e an I. I. D. set,

Let $\mathcal{X}_N$ be an I.I.D. sample of N points from μ. Let us count the number of pairs of points within distance ϵ, the (5) can be taken

$$\tilde{C}(N,\epsilon) := \frac{1}{N^2} \sum_{\substack{x_i, x_j \in \mathcal{X}_N \\ x_i \neq x_j}} H(\epsilon - \|x_i - x_j\|). \tag{7}$$

We then have

$$C(\epsilon) = \lim_{N\to\infty} \tilde{C}(N,\epsilon) \tag{8}$$

Pratical approach

In applications, given a finite (large) set of points χ_N, the CD can be estimated as the slope of the log-log plot of $\tilde{C}(N,\epsilon)$ versus ϵ in the limit of small ϵ.

ID estimators using PH

We assume to have a metric space X or a Manifold $\mathcal{M}$ embedded in some $\mathbb{R}^D$ equipped with a probability measure μ. Let $\mathcal{X}_N$ denote a set of N points sampled from X ($\mathcal{M}$) according to μ.

For any $\alpha > 0$, we define the power-weighted sum

$$\boxed{E_\alpha^i(\mathcal{X}_n) := \sum_{I \in PH_i(\mathcal{X}_n)} |I|^\alpha} \tag{9}$$

where $PH_i(\mathcal{X}_N)$, $i = 0, 1, 2, \ldots, D$ indicate the collections of topological features of dimensions $0, 1, 2, \ldots, D$ and $|I|$ denotes the persistence (or lifetime) of the topological feature I.

i-dim. Persistent Homology Fractal Dimension (PHFD)

i-dim. PHFD

Let X be a metric space equipped with a probability measure μ, let $\mathcal{X}_N \subset X$ be a random sample of n points from X distributed according to μ, and let $E_1^i(\mathcal{X}_N)$ as above. The i-dimensional Persistent Homology Fractal Dimension of μ is given by

$$dim_{PH}^i(\mu) = \inf_{d>0}\{\exists C(i,\mu,d) : E_1^i(\mathcal{X}_N) \leq C\, N^{(d-1)/d} \text{with prob. 1 as } N \to +\infty\}.$$

Conjecture: For all $0 \leq i < d$, there is a constant $C \geq 0$ (depending on μ, k, and i) such that

$$E_1^i(\mathcal{X}_N) = CN^{(d-1)/d} \tag{10}$$

with probability 1 as $N \to +\infty$.

Adams H.; Aminian M. et al. A Fractal Dimension for Measures via Persistent Homology. (eds) Topological Data Analysis *Abel Symposia Springer* **2020** (i-dim. PHFD)

Cont

Assuming the validity of this conjecture, taking the logarithm in (10), we get

$$\log(E_1^i(\mathcal{X}_N)) = \log(C) + \frac{d-1}{d}\log(N)\,, \tag{11}$$

which suggests that we can estimate D as the slope of the regression line as function of $\log(N)$, that is from $(d-1)/d$

Persistent Homology (PH) dimension

Anoher estimator is the PH dimension

PH dimension

Let X be a bounded subset of a metric space and μ a measure defined on X. For each $i \in \mathbb{N}$ and $\alpha > 0$, we define the Persistent Homology dimension (PH dim) as

$$dim_{PH_i^\alpha}(\mu) = \frac{\alpha}{1 - \beta}$$

where

$$\beta = \limsup_{N \to +\infty} \frac{\log(\mathbb{E}(E_\alpha^i(X_N))}{\log(N)}$$

Jaquette J.; Schweinhart B. Fractal dimension estimation with persistent homology: A comparative study. *Communications in Nonlinear Science and Numerical Simulation* **2020**, *84, 105163*

i-dim. α PHFD

Inspired by these two definitions, we have combined them in i-dim. α PHFD

i-dim. α PHFD

Let X be a metric space equipped with a probability measure μ, let $\mathcal{X}_N \subset X$ be a random sample of n points from X distributed according to μ, and let $E^i_\alpha(X_N)$ as above. The i-dimensional α Persistent Homology Fractal Dimension (i-dim. α PHFD) of μ is given by

$$dim^{i,\alpha}_{PH}(\mu) = \inf_{d>0}\{\exists C(i,\mu,d) : E^i_\alpha(X_N) \leq CN^{(d-\alpha)/d} \text{ with prob. 1 as N} \to +\infty\}.$$

Numerical tests

Tuning the value of α

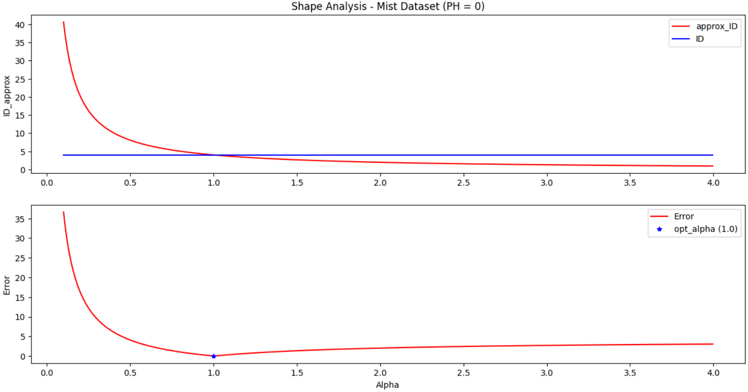


Figure: Shape Analysis dataset Mist with PH_0

Datasets

- Benchmark datasets

Dataset	d	Description
Helix	2	2-dimensional helix in $\mathbb{R}^3$
Swiss	2	Swiss-Roll in $\mathbb{R}^3$
Sphere	3	3-dimensional sphere linearly embedded in $\mathbb{R}^4$
NonLinear	4	Nonlinear Manifold in $\mathbb{R}^8$
Affine3d5d	3	Affine space in $\mathbb{R}^5$
Mist	4	Conc. figure, mistakable with a 3-dim. one in $\mathbb{R}^6$
CurvedManifold	12	Nonlinear (highly curved) manifold in $\mathbb{R}^{72}$
NonLinear6d36d	6	Nonlinear manifold in $\mathbb{R}^{36}$

- Fractals: Sierpiski triangle (4000 points) and Ikeda attractor (500 points)
- Neural activity stimulation (see below): Fdgo (25200), Context (10400), Reactgo_filtered (5200) points in $\mathbb{R}^{256}$

Neural Activity Datasets

- Starting from the analysis of activity trajectories of particular recurrent neural networks (RNNs), the aim is to mirror the brain functionality related to basic tasks (stimuli-response mapping on primates) using an NN.

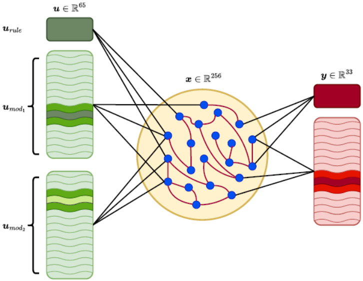


Figure: Scheme of the source of data

The figure represents the idea of how to use an RNN. We have considered only 3 stimuli, whose data are stored in .csv file with the name Fdgo, Context, and Reactgo_filtered.

Results I

Here d denotes the ID to approximate

Dataset	d	Corr. dim.
Helix	2	1.99
Swiss	2	1.98
Sphere	3	2.98
NonLinear	4	3.87
Affine3d5d	3	3.01
Mist	4	3.54
CurvedManifold	12	11.66
NonLinear6d36d	6	5.82
Sierpinski Triangle	1.585	1.585
Ikeda	unk	1.68
Fdgo	unk	1.07
Contextdm1	unk	1.14
Reactgo	unk	2.15

Table: Correlation Dimension of all datasets

Results II

Dataset	d	PH_0	PH_1
Helix	2	2.01	2.38
Swiss	2	1.93	2.16
Sphere	3	2.90	3.14
NonLinear	4	3.98	6.45
Affine3d5d	3	2.84	2.91
Mist	4	4.01	6.11
CurvedManifold	12	12.73	-
NonLinear6d36d	6	5.96	9.80
Serpinski	1.58	1.61	1.87
Ikeda Attractor	1.71	2.12	2.13
Reactgo	unk	2.47	2.54
Fdgo	unk	2.14	2.17
Contextdm1	unk	3.07	3.03

Table: α-PHFD for $0, 1$-dim. for all datasets with the "optimal α

Remark

In general, considering PH_0, the estimator performs better than on estimating PH_1.

Python implementation

Some information about software details:

- we use our code written in python
- the persistence diagrams are computed with free library available as ripser, persim and gudhi
- we consider a K-Fold CV averaged over 10 runs (random 70/30 training/testing splits)

Python implementations are available on GitHub repositories by Cinzia Bandiziol:

https://github.com/cinziabandiziol/TDA_classification
https://github.com/cinziabandiziol/persistence_kernels
https://github.com/cinziabandiziol/Topological_ID_Estimator