Topological Data Analysis for Machine Learning
Lecture 3: Topological Descriptors & How to Use Them

Bastian Rieck

Pseudomanifold

MLCB

D BSSE

ETH zürich

Preliminaries

Do you have feedback or any questions? Write to bastian.rieck@bsse.ethz.ch or reach out to @Pseudomanifold on Twitter. You can find the slides and additional information with links to more literature here:

https://topology.rocks/ecml_pkdd_2020

Recap

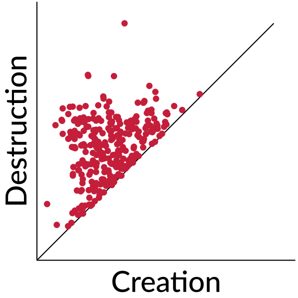
- There is a multi-scale generalisation of Betti numbers, called *persistent homology*.
- It is versatile and can be applied to point clouds or structured data.
- The resulting descriptors are called *persistence diagrams*.

In this lecture

The landscape of topological descriptors

What choices of topological descriptors do we have? What are their properties and respective advantages?

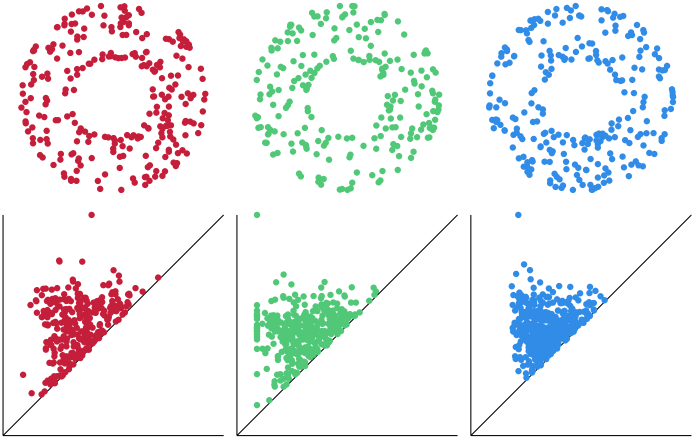
Persistence diagrams



- Points are tuples in $\mathbb{R} \times \mathbb{R} \cup \{\infty\}$.
- *Persistence* corresponds to distance to diagonal.
- Multiplicity of each point is not apparent!
- Space under diagonal is typically unused.

Properties of persistence diagrams

Stability (intuition)



Distances between persistence diagrams

Bottleneck distance

Given two persistence diagrams $\mathcal{D}$ and $\mathcal{D}'$, their *bottleneck* distance is defined as

$$W_\infty(\mathcal{D}, \mathcal{D}') := \inf_{\eta\colon \mathcal{D}\to\mathcal{D}'} \sup_{x\in\mathcal{D}} \|x - \eta(x)\|_\infty,$$

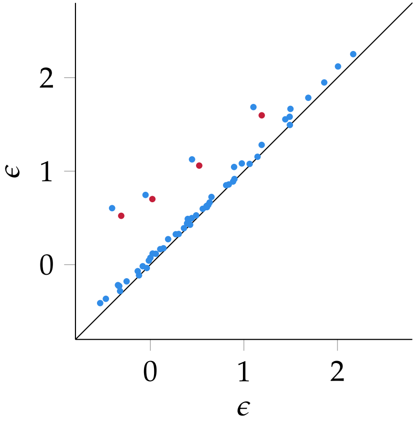
where $\eta\colon \mathcal{D} \to \mathcal{D}'$ denotes a bijection between the point sets of $\mathcal{D}$ and $\mathcal{D}'$ and $\|\cdot\|_\infty$ refers to the L_∞ distance between two points in $\mathbb{R}^2$.

Wasserstein distance

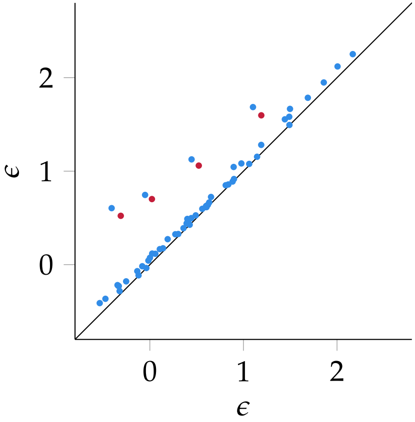
$$W_p(\mathcal{D}_1, \mathcal{D}_2) := \left(\inf_{\eta\colon \mathcal{D}_1\to\mathcal{D}_2} \sum_{x\in\mathcal{D}_1} \|x - \eta(x)\|_\infty^p\right)^{\frac{1}{p}}$$

Differences between the two distances

Bottleneck distance

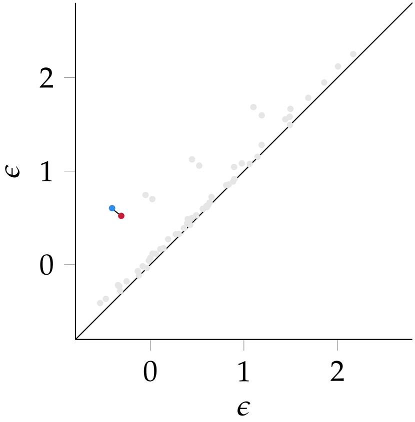


Wasserstein distance

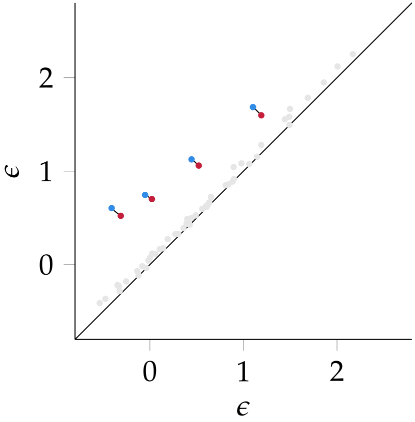


Differences between the two distances

Bottleneck distance



Wasserstein distance



Properties of persistence diagrams

Stability, formal definition

Tame functions

A function $f\colon \mathcal{M} \to \mathbb{R}$ is *tame* if it has a finite number of homological critical values and its homology groups are finite-dimensional.

Theorem

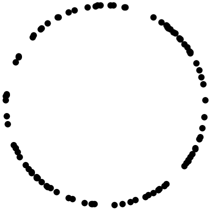
Let $\mathcal{M}$ be a triangulable space with continuous tame functions $f, g\colon \mathcal{M} \to \mathbb{R}$. Then the corresponding persistence diagrams $\mathcal{D}_f$ and $\mathcal{D}_g$ satisfy $W_\infty\left(\mathcal{D}_f, \mathcal{D}_g\right) \leq \|f - g\|_\infty$.

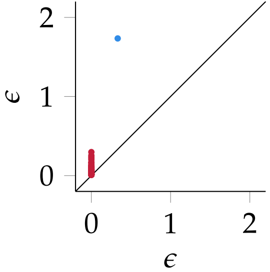
This theorem is due to Cohen-Steiner et al.[1] and laid the foundation for practical uses of persistent homology.

[1] D. Cohen-Steiner et al., ‘Stability of persistence diagrams’, *Discrete & Computational Geometry* 37.1, 2007, pp. 103–120

Properties of persistence diagrams

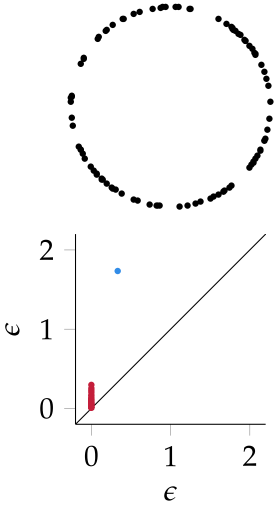
Stability only with respect to *small-scale* perturbations

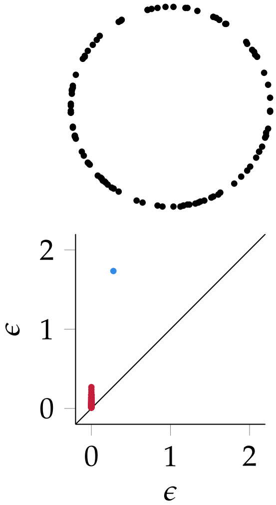




Properties of persistence diagrams

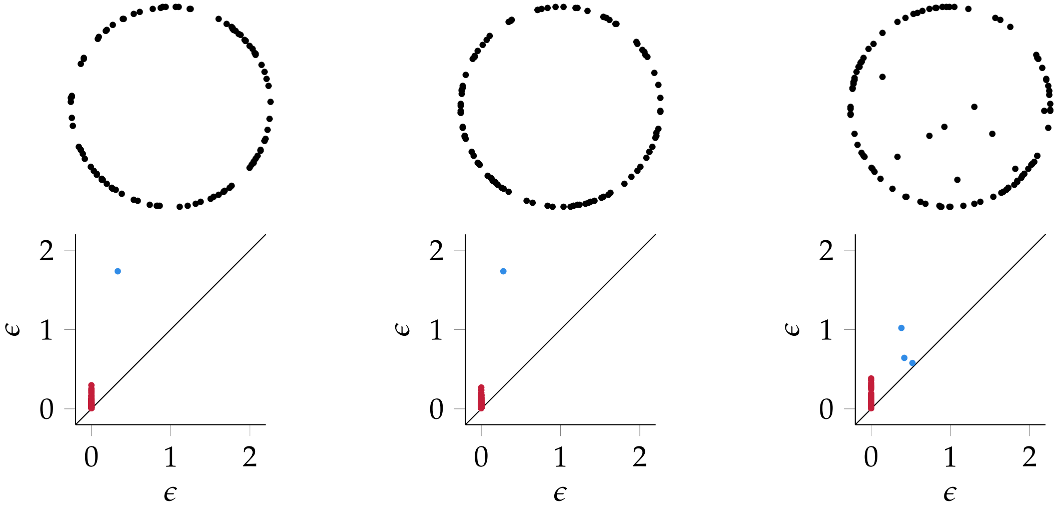
Stability only with respect to *small-scale* perturbations





Properties of persistence diagrams

Stability only with respect to *small-scale* perturbations



Interlude

Kernel theory

Kernel

Given a set $\mathcal{X}$, a function $k\colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a *kernel* if there is a Hilbert space $\mathcal{H}$ (an inner product space that is also a complete metric space) and a map $\Phi\colon \mathcal{X} \to \mathcal{H}$, such that $k(x, y) = \langle \Phi(x), \Phi(y) \rangle_{\mathcal{H}}$ for all $x, y \in \mathcal{X}$.

What is this good for?

Such a kernel can be used to assess the dissimilarity between two objects! The feature space $\mathcal{H}$ can be high-dimensional, thus simplifying classification.

A Stable Multi-Scale Kernel for Topological Machine Learning

This is the first kernel between persistence diagrams[2]; it is simple to implement and expressive.

Kernel and feature map definition

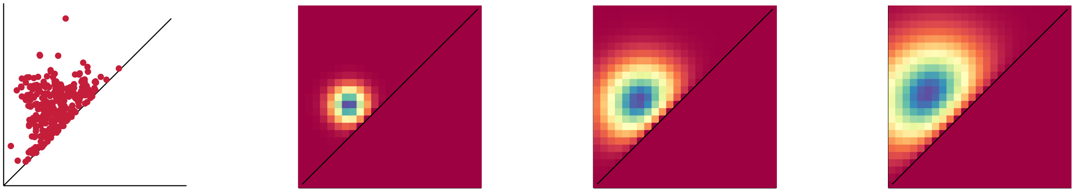
$$k(\mathcal{D}, \mathcal{D}') := \frac{1}{8\pi\sigma} \sum_{p \in \mathcal{D}, q \in \mathcal{D}'} \exp(-8^{-1}\sigma^{-1}\|p - q\|^2) - \exp(-8^{-1}\sigma^{-1}\|p - \overline{q}\|^2)$$

$$\Phi(x) := \frac{1}{4\pi\sigma} \sum_{p \in \mathcal{D}} \exp(-4^{-1}\sigma^{-1}\|x - p\|^2) - \exp(-4^{-1}\sigma^{-1}\|x - \overline{p}\|^2)$$

[2] J. Reininghaus et al., 'A stable multi-scale kernel for topological machine learning', *IEEE Conference on Computer Vision and Pattern Recognition* (CVPR), Red Hook, NY, USA, 2015, pp. 4741–4748

A Stable Multi-Scale Kernel for Topological Machine Learning

Feature map illustration



Persistence diagram $\sigma = 0.1$ $\sigma = 0.5$ $\sigma = 1.0$

More kernels & applications

Alternative formulations exist, based on sliced Wasserstein distance calculations[3], kernel embeddings[4], or Riemannian geometry[5].

Applications

- Kernel PCA for visualisation, dimensionality reduction, and feature generation
- Kernel SVM for classification
- Kernel SVR for regression

[3] M. Carrière et al., 'Sliced Wasserstein Kernel for Persistence Diagrams', vol. 70, Proceedings of Machine Learning Research, 2017, pp. 664–673

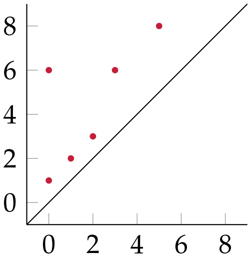
[4] G. Kusano et al., 'Kernel Method for Persistence Diagrams via Kernel Embedding and Weight Factor', *Journal of Machine Learning Research* 18.189, 2018, pp. 1–41

[5] T. Le and M. Yamada, 'Persistence Fisher Kernel: A Riemannian Manifold Kernel for Persistence Diagrams', *Advances in Neural Information Processing Systems 31*, 2018, pp. 10007–10018

Betti curves

A simplified representation of persistence diagrams

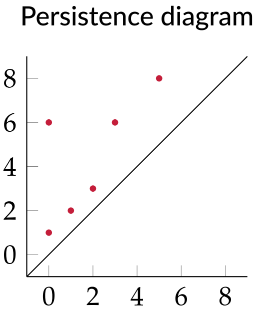
Persistence diagram

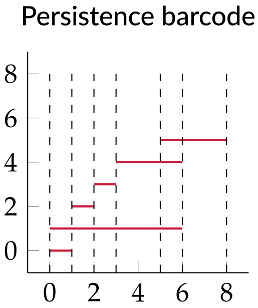


The Betti curve is a function mapping a persistence diagram to an integer-valued curve, i.e. each Betti curve is a function $\mathcal{B}\colon \mathbb{R} \to \mathbb{N}$.

Betti curves

A simplified representation of persistence diagrams

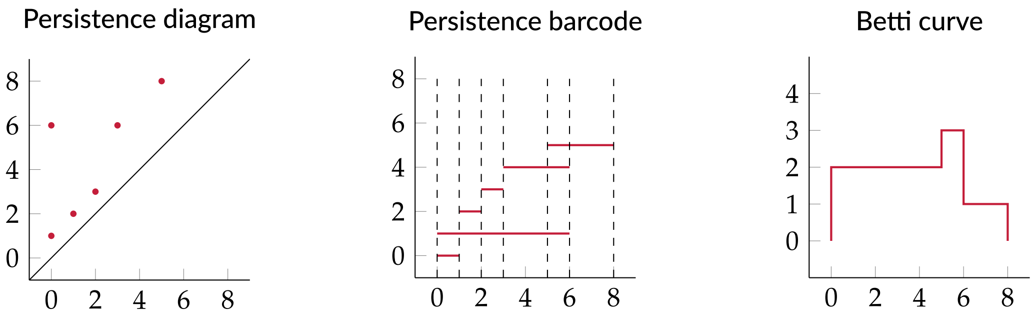




The Betti curve is a function mapping a persistence diagram to an integer-valued curve, i.e. each Betti curve is a function $\mathcal{B}\colon \mathbb{R} \to \mathbb{N}$.

Betti curves

A simplified representation of persistence diagrams



The Betti curve is a function mapping a persistence diagram to an integer-valued curve, i.e. each Betti curve is a function $\mathcal{B}\colon \mathbb{R} \to \mathbb{N}$.

Properties of Betti curves

- Easy to calculate
- Simple representation, ‘living’ in the space of piecewise linear functions
- Vector space operations are possible (addition, scalar multiplication)
- Distances and kernels can be defined

Kernel

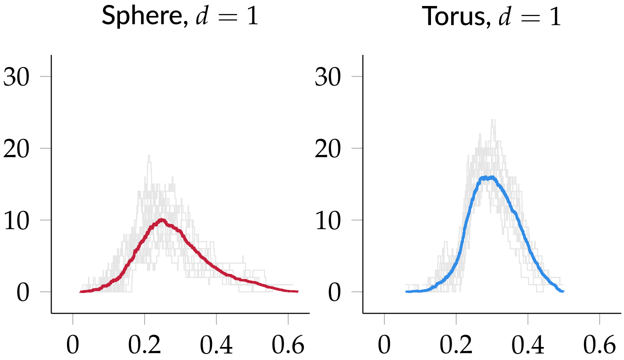
$$k_p(\mathcal{D}, \mathcal{D}') := -\left(\int_{\mathbb{R}} |\mathcal{B}_{\mathcal{D}}(x) - \mathcal{B}_{\mathcal{D}'}(x)|^p \mathrm{d}x \right)^{\frac{1}{p}}$$

More properties and formal descriptions are available in a preprint![6]

[6] B. Rieck et al., *Topological Machine Learning with Persistence Indicator Functions*, 2019, arXiv: 1907.13496 [math.AT], in press

Betti curves

Exploiting the vector space structure

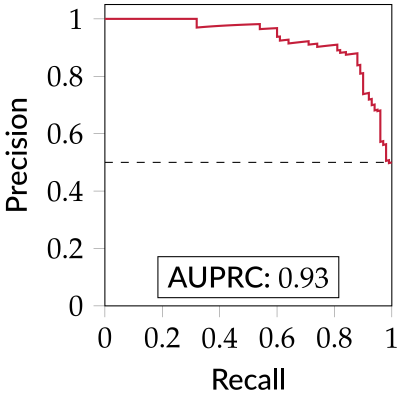


Permits hypothesis testing or comparing *means* of distributions!

Betti curves

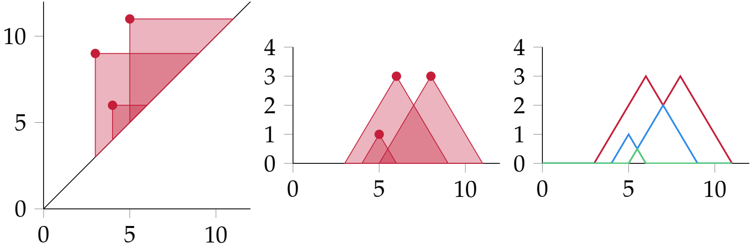
Classification scenario example

- Use REDDIT-BINARY data set (co-occurrence graphs)
- Calculate filtration based on *vertex degree*
- Calculate persistence diagrams for $d = 1$ (cycles)
- Given $p = 1$, use a kernel SVM for classification



Persistence landscapes

- Calculate rank of 'covered' topological features of a diagram
- 'Peel off' layers iteratively



This formulation is due to Peter Bubenik[7]; it has beneficial statistical properties, and *also* permits the efficient calculation of distances and kernels!

[7] P. Bubenik, 'Statistical Topological Data Analysis Using Persistence Landscapes', *Journal of Machine Learning Research* 16, 2015, pp. 77–102

Persistence landscapes

Properties and recent work

Efficient Topological Layer based on Persistent Landscapes

Kwangho Kim *[1], Jisu Kim †[2], Joon Sik Kim[1], Frédéric Chazal[2], and Larry Wasserman[1]

[1]Carnegie Mellon University, USA
[2]Inria Saclay, France

05 February, 2020

Abstract

We propose a novel topological layer for general deep learning models based on persistent landscapes, in which we can efficiently exploit underlying topological features of the input data structure. We use the robust DTM function and show differentiability with respect to layer inputs, for a general persistent homology with arbitrary filtration. Thus, our proposed layer can be placed anywhere in the network architecture and feed critical information on the topological features of input data into subsequent layers to improve the learnability of the networks toward a given task. A task-optimal structure of the topological layer is learned during training via backpropagation, without requiring any input featurization or data preprocessing. We provide a tight stability theorem, and show that the proposed layer is robust towards noise and outliers. We demonstrate the effectiveness of our approach by classification experiments on various datasets.

Keywords: topological data analysis, deep learning, persistent diagram, persistent homology, topological feature, stability theorem

*kwanghk@cmu.edu
†jisu.kim@inria.fr

- The landscape can be *sampled* at regular intervals to obtain a fixed-size feature vector.
- Built-in hierarchy!
- Bijective mapping (no information lost).
- Stability theorems hold.
- Recently: usage as neural network layer!

Other functional summaries

Template functions

Approximating Continuous Functions on Persistence Diagrams Using Template Functions

Jose A. Perea joperea@msu.edu
Elizabeth Munch muncheli@egr.msu.edu
Department of Computational Mathematics, Science, and Engineering; and Department of Mathematics

Firas A. Khasawneh khasawn3@egr.msu.edu
Department of Mechanical Engineering

Michigan State University
East Lansing, MI 48824, USA

Abstract

The persistence diagram is an increasingly useful tool from Topological Data Analysis, but its use alongside typical machine learning techniques requires mathematical finesse. The most success to date has come from methods that map persistence diagrams into $\mathbb{R}^n$, in a way which maximizes the structure preserved. This process is commonly referred to as *featurization*. In this paper, we describe a mathematical framework for featurization using template functions. These functions are general as they are only required to be continuous and compactly supported. We discuss two realizations: tent functions, which emphasize the local contributions of points in a persistence diagram, and interpolating polynomials, which capture global pairwise interactions. We combine the resulting features with classification and regression algorithms on several examples including shape data and the Rossler system. Our results show that using template functions yields high accuracy rates that match and often exceed those of existing featurization methods. One counter-intuitive observation is that in most cases using interpolating polynomials, where each point contributes globally to the feature vector, yields significantly better results than using tent functions, where the contribution of each point is localized. Along the way, we provide a complete characterization of compactness in the space of persistence diagrams.

Keywords: Topological Data Analysis, Persistent Homology, Machine Learning, Featurization, Bottleneck Distance

1. Introduction

Many machine learning tasks can be reduced to the following problem: Approximate a continuous function defined on a topological space, the "ground truth," given the function values (or approximations thereof) on some subset of the points. This task has been well studied for data sitting in Euclidean space; however, more work is necessary to extend these ideas to arbitrary topological spaces. In this paper, we focus on the task of classification and regression on the space of persistence diagrams endowed with the bottleneck distance, $(\mathcal{D}, d_B)$. These objects arise in the field of Topological Data Analysis (TDA) as signatures

1

- Evaluate template (*tent function*) on persistence diagram.
- This incorporates more than just point information!

Let g be a template function operating on persistence pairs, then we obtain a simple embedding based on summation:

$$
\begin{aligned}
f\colon \mathbb{R} \times \mathbb{R} \cup \{\infty\} &\to \mathbb{R} \\
\mathcal{D} &\mapsto \sum_{x \in \mathcal{D}} g(x)
\end{aligned}
$$

Obtain a feature vector by using *multiple* template functions!

Histogram-based vectorisation

Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19)

Persistence Bag-of-Words for Topological Data Analysis

Bartosz Zieliński[1], **Michał Lipiński**[1], **Mateusz Juda**[1], **Matthias Zeppelzauer**[2] and **Paweł Dłotko**[3]

[1]The Institute of Computer Science and Computer Mathematics, Faculty of Mathematics and Computer Science, Jagiellonian University

[2]Media Computing Group, Institute of Creative Media Technologies, St. Pölten University of Applied Sciences,

[3]Department of Mathematics and Swansea Academy of Advanced Computing, Swansea University

{bartosz.zielinski, michal.lipinski, mateusz.juda}@uj.edu.pl, m.zeppelzauer@fhstp.ac.at, p.t.dlotko@swansea.ac.uk

Abstract

Persistent homology (PH) is a rigorous mathematical theory that provides a robust descriptor of data in the form of persistence diagrams (PDs). PDs exhibit, however, complex structure and are difficult to integrate in today's machine learning workflows. This paper introduces persistence bag-of-words: a novel and stable vectorized representation of PDs that enables the seamless integration with machine learning. Comprehensive experiments show that the new representation achieves state-of-the-art performance and beyond in much less time than alternative approaches.

1 Introduction

Topological data analysis (TDA) provides a powerful framework for the structural analysis of high-dimensional data. A main tool of TDA is Persistent Homology (PH) [Edelsbrunner and Harer, 2010], which currently gains increasing importance in data science [Ferri, 2017]. It has been applied to a number of disciplines including, biology [Gameiro *et al.*, 2014], material science [Lee *et al.*, 2017], analysis of financial markets [Gidea and Katz, 2018]. Persistence homology is also used as a novel measure of GANs (Generative Adversarial Networks) performance [Khrulkov and Oseledets, 2018], and as a complexity measure for neural network architectures [Rieck *et al.*, 2018]. PH can be efficiently computed using various currently available tools [Bauer *et al.*, 2017; Dey *et al.*, 2019; Maria *et al.*, 2014]. A basic introduction to PH is given in the supplementary material (SM in the following)[1].

The common output representation of PH are persistence diagrams (PDs) which are multisets of points in $\mathbb{R}^2$. Due to their variable size, PDs are not easy to integrate within common data analysis, statistics and machine learning workflows. To alleviate this problem, a number of kernel functions and vectorization methods for PDs have been introduced. Kernel-based approaches have a strong theoretical background but in practice they often become inefficient when the number of training samples is large. As the entire kernel matrix must usually be computed explicitly (like in case of SVMs), this leads to roughly quadratic complexity in computation time and memory with respect to the size of the training set. Furthermore, such approaches are limited to kernelized methods, such as SVM and kernel PCA. Vectorized representations in contrast are compatible with a much wider range of methods and do not suffer from complexity constraints of kernels. Since they require a spatial quantization of the PD they might suffer from a loss in precision compared to kernels, especially since PDs are sparsely and unevenly populated structures.

In this work, we present a novel spatially adaptive and thus more accurate representation of PDs, which aims at combining the large representational power of kernel-based approaches with the general applicability of vectorized representations. To this end, we extend the popular bag-of-words (BoW) encoding (originating from text and image retrieval) to TDA to cope with the inherent sparsity of PDs [McCallum and Nigam, 1998; Sivic and Zisserman, 2003]. The proposed adaptation of BoW gives a universally applicable fixed-sized feature vector of low-dimension. It is, under mild conditions, stable with respect to a standard metric in PDs. Experiments demonstrate that our new representation achieves state-of-the-art performance and even outperforms numerous competitive methods while requiring orders of magnitude less time and being more compact. Due to the run-time efficiency of our approach it may in future enable the application of TDA for larger-scale data than possible today.

The paper is structured as follows. Section 2 reviews related approaches. In Section 3 we introduce persistence bag-of-words and prove its stability. Sections 5 and 6 present experimental setup, results and discussion. Please consider the SM^a for additional information.

2 Background and Related Work

Different kernel based and vectorized representations have been introduced to make PDs compatible with statistical anal-

[1]Supplementary material: http://www.ii.uj.edu.pl/~zielinsb/papers/2019_ijcai_supplement.pdf

4489

- Cluster persistence diagram
- Learn representatives
- Learn 'bag-of-word' (BOW) representation
- Use quantised BOW representation as feature vector

Parameters are not easy to pick and there is no 'intuitive' description of the resulting representation. This can be overcome, however!

Persistence images

Multi-scale descriptors

Algorithm

Use $\Psi\colon \mathbb{R}^2 \to \mathbb{R}$ to turn a diagram $\mathcal{D}$ into a surface via $\Psi(z) := \sum_{x,y \in \mathcal{D}} \mathrm{w}(x,y)\Phi(x,y,z)$, where $\mathrm{w}(\cdot)$ is a fixed piecewise linear weight function and $\Phi(\cdot)$ denotes a probability distribution, which is typically chosen to be a normalised symmetric Gaussian. By discretising Ψ (using an $r \times r$ grid), a persistence diagram is transformed into a *persistence image*.[8]

[8] H. Adams et al., 'Persistence Images: A Stable Vector Representation of Persistent Homology', *Journal of Machine Learning Research* 18.8, 2017, pp. 1–35

Persistence images

Properties

Journal of Machine Learning Research 18 (2017) 1-35 Submitted 7/16; Published 2/17

Persistence Images: A Stable Vector Representation of Persistent Homology

Henry Adams ADAMS@MATH.COLOSTATE.EDU
Tegan Emerson EMERSON@MATH.COLOSTATE.EDU
Michael Kirby KIRBY@MATH.COLOSTATE.EDU
Rachel Neville NEVILLE@MATH.COLOSTATE.EDU
Chris Peterson PETERSON@MATH.COLOSTATE.EDU
Patrick Shipman SHIPMAN@MATH.COLOSTATE.EDU
Department of Mathematics
Colorado State University
1874 Campus Delivery
Fort Collins, CO 80523-1874

Sofya Chepushtanova SOFYA.CHEPUSHTANOVA@WILKES.EDU
Department of Mathematics and Computer Science
Wilkes University
84 West South Street
Wilkes-Barre, PA 18766, USA

Eric Hanson ERIC.HANSON@TCU.EDU
Department of Mathematics
Texas Christian University
Box 298900
Fort Worth, TX 76129

Francis Motta MOTTA@MATH.DUKE.EDU
Department of Mathematics
Duke University
Durham, NC 27708, USA

Lori Ziegelmeier LZIEGEL1@MACALESTER.EDU
Department of Mathematics, Statistics, and Computer Science
Macalester College
1600 Grand Avenue
Saint Paul, MN 55105, USA

Editor: Michael Mahoney

Abstract

Many data sets can be viewed as a noisy sampling of an underlying space, and tools from topological data analysis can characterize this structure for the purpose of knowledge discovery. One such tool is persistent homology, which provides a multiscale description of the homological features within a data set. A useful representation of this homological information is a *persistence diagram* (PD). Efforts have been made to map PDs into spaces with additional structure valuable to machine learning tasks. We convert a PD to a finite-dimensional vector representation which we call a *persistence image* (PI), and prove the stability of this transformation with respect to small perturbations in the inputs. The

- Beneficial stability properties
- Intuitive description in terms of density estimates
- Resolution and smoothing parameter are hard to choose
- Representation is not sparse (quadratic scaling with r!)
- Easy to use in a classification setting, though!

Extensions of persistence images

Learning weights

Learning metrics for persistence-based summaries and applications for graph classification

Qi Zhao
zhao.2017@osu.edu

Yusu Wang
yusu@cse.ohio-state.edu

Computer Science and Engineering Department
The Ohio State University
Columbus, OH 43221

Abstract

Recently a new feature representation framework based on a topological tool called persistent homology (and its persistence diagram summary) has gained much momentum. A series of methods have been developed to map a persistence diagram to a vector representation so as to facilitate the downstream use of machine learning tools. In these approaches, the importance (weight) of different persistence features are usually *pre-set*. However often in practice, the choice of the weight-function should depend on the nature of the specific data at hand. It is thus highly desirable to *learn* a best weight-function (and thus metric for persistence diagrams) from labelled data. We study this problem and develop a new weighted kernel, called *WKPI*, for persistence summaries, as well as an optimization framework to learn the weight (and thus kernel). We apply the learned kernel to the challenging task of graph classification, and show that our WKPI-based classification framework obtains similar or (sometimes significantly) better results than **the best results** from a range of previous graph classification frameworks on benchmark datasets.

1 Introduction

In recent years a new data analysis methodology based on a topological tool called persistent homology has started to attract momentum. The persistent homology is one of the most important developments in the field of topological data analysis, and there have been fundamental developments both on the theoretical front (e.g, [23, 10, 13, 8, 14, 5]), and on algorithms / implementations (e.g, [43, 4, 15, 20, 29, 3]). On the high level, given a domain X with a function $f : X \to \mathbb{R}$ on it, the persistent homology summarizes "features" of X across multiple scales simultaneously in a single summary called the *persistence diagram* (see the second picture in Figure 1). A persistence diagram consists of a multiset of points in the plane, where each point $p = (b, d)$ intuitively corresponds to the birth-time (b) and death-time (d) of some (topological) features of X w.r.t. f. Hence it provides a concise representation of X, capturing *multi-scale features* of it simultaneously. Furthermore, the persistent homology framework can be applied to complex data (e.g, 3D shapes, or graphs), and different summaries could be constructed by putting different descriptor functions on input data.

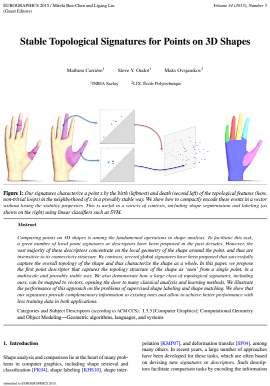
Due to these reasons, a new persistence-based feature vectorization and data analysis framework (Figure 1) has become popular. Specifically, given a collection of objects, say a set of graphs modeling chemical compounds, one can first convert each shape to a persistence-based representation. The input data can now be viewed as a set of points in a persistence-based feature space. Equipping this space with appropriate distance or kernel, one can then perform downstream data analysis tasks (e.g, clustering).

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

- Obtain persistence images from graph filtration
- *Learn* a weight function on the persistence image
- Calculate weighted distance between images
- Use this as a *kernel* in an SVM

Other vectorisation methods

Extracting signatures


EUROGRAPHICS 2015 / Mirela Ben-Chen and Ligang Liu
(Guest Editors)

Volume 34 (2015), Number 5

Stable Topological Signatures for Points on 3D Shapes

Mathieu Carrière[1] Steve Y. Oudot[1] Maks Ovsjanikov[2]

[1]INRIA Saclay [2]LIX, École Polytechnique

Figure 1: *Our signatures characterize a point x by the birth (leftmost) and death (second left) of the topological features (here, non-trivial loops) in the neighborhood of x in a provably stable way. We show how to compactly encode these events in a vector without losing the stability properties. This is useful in a variety of contexts, including shape segmentation and labeling (as shown on the right) using linear classifiers such as SVM.*

Abstract

Comparing points on 3D shapes is among the fundamental operations in shape analysis. To facilitate this task, a great number of local point signatures or descriptors have been proposed in the past decades. However, the vast majority of these descriptors concentrate on the local geometry of the shape around the point, and thus are insensitive to its connectivity structure. By contrast, several global *signatures have been proposed that successfully capture the overall topology of the shape and thus characterize the shape as a whole. In this paper, we propose the first point descriptor that captures the topology structure of the shape as 'seen' from a single point, in a multiscale and provably stable way. We also demonstrate how a large class of topological signatures, including ours, can be mapped to vectors, opening the door to many classical analysis and learning methods. We illustrate the performance of this approach on the problems of supervised shape labeling and shape matching. We show that our signatures provide complementary information to existing ones and allow to achieve better performance with less training data in both applications.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

1. Introduction

Shape analysis and comparison lie at the heart of many problems in computer graphics, including shape retrieval and classification [FK04], shape labeling [KHS10], shape interpolation [KMP07], and deformation transfer [SP04], among many others. In recent years, a large number of approaches have been developed for these tasks, which are often based on devising new *signatures* or *descriptors*. Such descriptors facilitate comparison tasks by encoding the information

submitted to EUROGRAPHICS 2015.

Given two points x, y in a persistence diagram, calculate

$$m(x,y) := \min\{\|x-y\|_\infty, d_\Delta(x), d_\Delta(y)\},$$

where $d_\Delta(x)$ denotes the L_∞ distance to the diagonal. Sort all $m(x,y)$ in descending order and pick k of them (padding with zeroes) to obtain a fixed-size feature vector representation. Very effective, but the computation scales quadratically in the number of entries of a persistence diagram!

Other vectorisation methods

Summary statistics

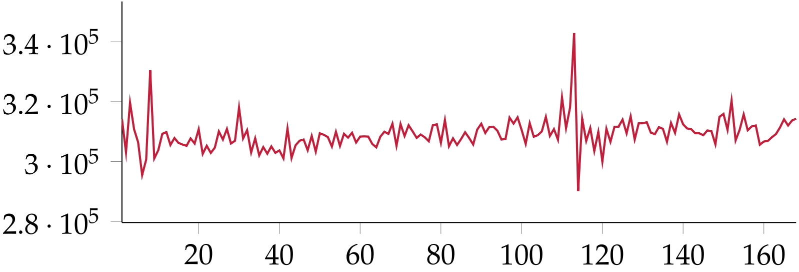
Norms of a persistence diagram

$$\|\mathcal{D}\|_\infty := \max_{x,y\in\mathcal{D}} \operatorname{pers}(x,y)^p \quad \text{and} \quad \|\mathcal{D}\|_p := \sqrt[p]{\sum_{x,y\in\mathcal{D}} \operatorname{pers}(x,y)^p},$$

These norms are stable and highly useful in obtaining simple descriptions of time-varying persistence diagrams!

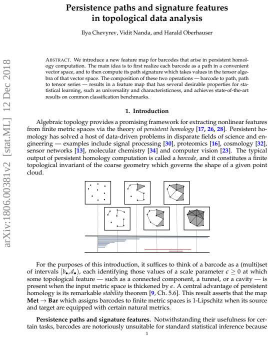
Example

Total persistence of a time series of persistence diagrams



Multiple curves can be easily compared with each other—making this an excellent *proxy* for more complicated distance calculations.

Generic vectorisation based on signatures



arXiv:1806.00381v2 [stat.ML] 12 Dec 2018

Persistence paths and signature features in topological data analysis

Ilya Chevyrev, Vidit Nanda, and Harald Oberhauser

Abstract. We introduce a new feature map for barcodes that arise in persistent homology computation. The main idea is to first realize each barcode as a path in a convenient vector space, and to then compute its path signature which takes values in the tensor algebra of that vector space. The composition of these two operations — barcode to path, path to tensor series — results in a feature map that has several desirable properties for statistical learning, such as universality and characteristicness, and achieves state-of-the-art results on common classification benchmarks.

1. Introduction

Algebraic topology provides a promising framework for extracting nonlinear features from finite metric spaces via the theory of *persistent homology* [17, 26, 28]. Persistent homology has solved a host of data-driven problems in disparate fields of science and engineering — examples include signal processing [30], proteomics [16], cosmology [32], sensor networks [13], molecular chemistry [34] and computer vision [23]. The typical output of persistent homology computation is called a *barcode*, and it constitutes a finite topological invariant of the coarse geometry which governs the shape of a given point cloud.

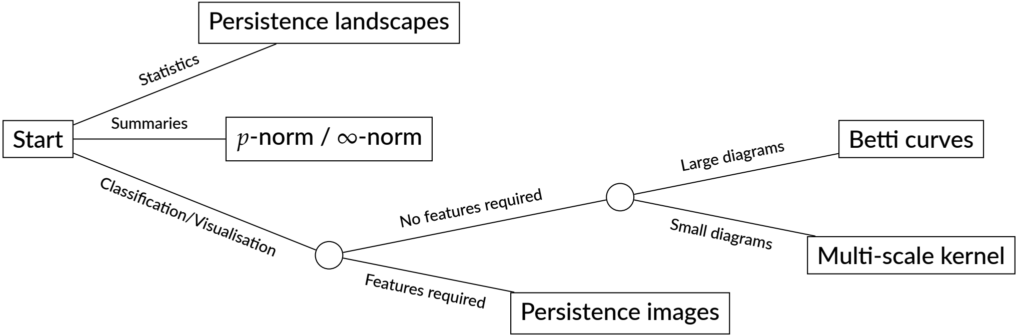
For the purposes of this introduction, it suffices to think of a barcode as a (multi)set of intervals $[b_\bullet, d_\bullet)$, each identifying those values of a scale parameter $\epsilon \geq 0$ at which some topological feature — such as a connected component, a tunnel, or a cavity — is present when the input metric space is thickened by ϵ. A central advantage of persistent homology is its remarkable *stability* theorem [9, Ch. 5.6]. This result asserts that the map **Met** $\to$ **Bar** which assigns barcodes to finite metric spaces is 1-Lipschitz when its source and target are equipped with certain natural metrics.

Persistence paths and signature features. Notwithstanding their usefulness for certain tasks, barcodes are notoriously unsuitable for standard statistical inference because

1

- Different representations can also give rise to *paths*.
- Use *path signature* (a universal non-linearity on paths of bounded variation) to compare them.
- Path signatures have several beneficial properties, one of them being stability!
- Promising results, but computationally 'heavy'.

Which method to use in practice?



Take-away messages

- The original persistence diagram is cumbersome to work with due to its multiset structure.
- Hence, there are numerous topological descriptors for different usage scenarios.
- Two large classes of methods exist, kernel-based and feature-based (although some kernels also give rise to finite-dimensional features).

https://topology.rocks/ecml_pkdd_2020