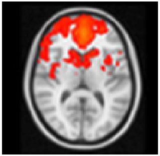
Vector Representations of Persistence Diagrams

Oleg Kachan
HSE University

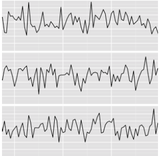
21 March 2024

Some bits of Topological Data Analysis

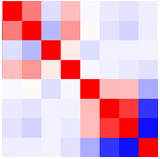
Weighted Graphs Analysis



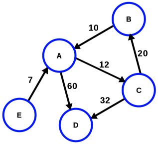
fMRI imaging



time series



correlation matrix



weighted graph

$\mathbf{R}_{ij}$
$r_{ij} \in [-1, 1]$

$\mathbf{A}_{ij}$ - adj. matrix
$a_{ij} = 1 - |r_{ij}|$
$a_{ij} \in [0, 1]$

Clique Complex

Let G be a weighted graph, with adjacency matrix $\mathbf{A}_{ij}$, then *clique complex* $K(G)$ is obtained by:

- get sparse graph by thresholding adjacency matrix $\mathbf{A}_{ij} \leq \varepsilon$

$$\mathbf{A}'_{ij} = \begin{cases} a_{ij} & a_{ij} \leq \varepsilon \\ 0 & otherwise \end{cases}$$

- to every k-clique of graph with adjacency matrix $\mathbf{A}'_{ij}$ associate $(k-1)$-simplex in the simplicial complex $K(G)$

Filtration is induced by weights on graph edges.

Vietoris-Rips Complex

Consider a finite metric space (point cloud) X. A *Vietoris-Rips complex* of X at radius ε is defined:

$$VR_{\varepsilon}(X) = \{\sigma \in X \mid d(x, x') \leq 2\varepsilon,\ \forall (x, x') \in \sigma\}$$

That is $k+1$ points form a k-simplex if they are all pairwise 2ε-distant.

Filtration is induced by the *distance to a set* function $d_X(y) := d(x, y)$, where $x \in X$ and $y \in \mathbb{R}^n$ for all x and y.

Cubical Complex

Given a 2D digital image X, a pixel in an image corresponds to a 2-simplex in a cubical complex $\mathrm{Cube}(X)$.

Cubical complex[1] is a simplicial complex, i.e. a family of sets closed under inclusion, consisting of simplices which are products of elementary intervals of $\mathbb{R}$.

Filtration is induced by a function of pixel intensity.

1. Kaczynski, Mischaikow, and Mrozek, *Computational Homology*, 2006.

Persistent Homology as a Mapping

Consider a pair (X, f), where X is a *topological space* and $f : X \to \mathbb{R}$ is a *filter function*.

Persistent homology is a mapping to the space of *persistence modules* (a collection of vector spaces connected by linear maps) over a field $\mathbb{F}$, that quantify the homology of *sublevel sets* $X_t = \{x \in X \mid f(x) \leq t\}$ of the function f on X, where t takes values from a totally ordered *index set* $(R, <)$:

$$(\mathcal{X}, f) \to H^f_{\bullet;\mathbb{F}}(\mathcal{X})$$

Decomposition of Persistence Module

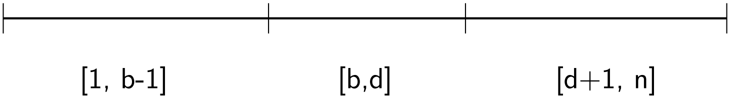
A persistent module $H_k^{f,\mathbb{F}}$ can be decomposed into a direct sum of indecomposable interval modules $\mathbb{I}_{\mathbb{F}}[b_i, d_i]$, where b_i is the *birth* of k-th homological class and d_i is its *death*.

$$H_k^{f,\mathbb{F}} \cong \bigoplus_{i=1}^{n=\beta_k} \mathbb{I}_{\mathbb{F}}[b_i, d_i]$$

Interval Module

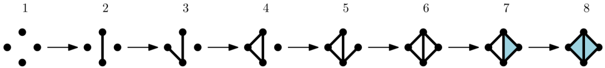
A subset $I \subseteq R$ is an *interval* if it is non-empty and $r \leq s \leq t$ with $r, t \in I$ implies $s \in I$. An *interval module* $\mathbb{I}_{\mathbb{F}}$ is given by $V_t = \mathbb{F}$ for $t \in I$, $V_t = 0$ for $t \notin I$, and $\rho_{ts} = \mathrm{id}$ for $s, t \in I$ with $s \leq t$.

$$0 \xrightarrow{0} \dots \xrightarrow{0} 0 \xrightarrow{0} \mathbb{F} \xrightarrow{1} \dots \xrightarrow{1} \mathbb{F} \xrightarrow{0} 0 \xrightarrow{0} \dots \xrightarrow{0} 0$$



Example of Decomposition

Filtration [2]:



$$0 \longrightarrow 0 \longrightarrow 0 \longrightarrow \mathbb{F} \longrightarrow \mathbb{F} \longrightarrow \mathbb{F}^2 \rightarrow \mathbb{F} \longrightarrow 0$$

Decomposition of persistence module:

$$0 \longrightarrow 0 \longrightarrow 0 \longrightarrow \mathbb{F} \xrightarrow{1} \mathbb{F} \xrightarrow{1} \mathbb{F} \xrightarrow{1} \mathbb{F} \longrightarrow 0 \quad [c]$$

$$0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow 0 \longrightarrow \mathbb{F} \longrightarrow 0 \longrightarrow 0 \quad [c']$$

$$H_1 \cong \bigoplus_{i=1}^{n=\beta_1} \mathbb{I}[b_i, d_i]$$

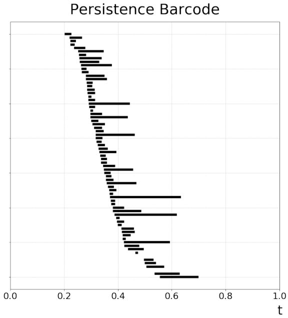
$$= \mathbb{I}[4,7] \oplus \mathbb{I}[6,6]$$

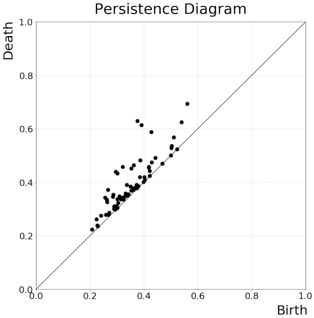
2. Image courtesy of Clement Maria

Persistence Barcode and Diagram

For each dimension k, a k-th *persistence barcode* $Bar_k(X)$ is a collection of intervals $\{\mathbb{I}[b_i, d_i]_k\}$, describing the decomposition of a persistent module.

A k-th *persistence diagram* $Dgm_k(X)$ is a natural bijection from a collection of intervals to a multiset of points $\{(b_i, d_i)_k \mid i \in I\}$ on extended Euclidean plane $\bar{\mathbb{R}}^2 := \{\mathbb{R}^2 \cup +\infty\}$.





Machine Learning with Topological Signatures

A persistence diagram being a multiset of points is not suited for machine learning algorithms generally expecting a vector of fixed dimension as an input. Although there are a number of distances[3] and kernels[4] defined on the space of persistence diagrams, we will focus on their finite vector representations.

3. Edelsbrunner and Harer, *Computational Topology: An Introduction*, 2010.
4. Carriere, Cuturi, and Oudot, "Sliced wasserstein kernel for persistence diagrams," 2017; Le and Yamada, "Persistence fisher kernel: A riemannian manifold kernel for persistence diagrams," 2018.

Machine Learning with Topological Signatures

Desired properties of representation[5]:

- is a vector in $\mathbb{R}^n$,
- is stable with respect to input noise,
- is efficient to compute,
- maintains an interpretable connection to the original PD,
- allows one to adjust the relative importance of points in different regions of the PD

5. Adams et al., "Persistence images: A stable Vector Representation of Persistent Homology," 2017.

Software

	Lang.	Algos.	Coeffs.	Filtrations	Gens.
JavaPlex	Java	H_* H^* Z_*	Q, F_p	VR, W	Yes
Perseus	C++	H_* Z_*	F_2	VR, Cub	No
Dionysus	C++	H_* H^* Z_*	F_2, F_p	VR, α, *Cech*	Yes
PHAT/DIPHA	C++	H_* H^*	F_2	VR, Cub, f.comp.	No
Gudhi	C++	H^*	F_2	VR, α, W, *Cub*	No
Ripser/Flagser	C++	H^*	F_p	VR, dir. flag	Yes
Eirene	Julia	H_*	F_2	VR, f.comp.	No
Simplicial.jl	Julia	H_*	F_2	VR, directed	No

Table: Software for computation of persistent homology[6]

Algorithms: H_* - homology, H^* - cohomology, Z_* - zigzag homology
Filtrations: VR - Vietoris-Rips complex, α - alpha complex, Cech - Cech complex, W - witness complex, Cube - cubical complex, f.comp. - general complex with user-defined filtration, *Gens.* stands for (co)homology generators

6. Otter et al., "A roadmap for the computation of persistent homology," 2017.

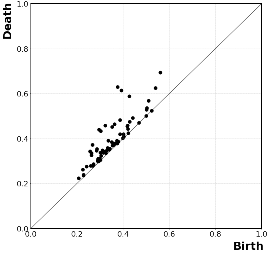
Vector Representations of Persistence Diagrams

Coordinates

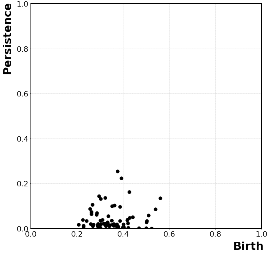
Consider an interval $(b_i, d_i)_k$ of dimension k, one can obtain a sets of quantities, summarizing different properties of interval:

Persistence:	*Midlife*:	*Mult. life*:
$p_i = d_i - b_i$	$ml_i = (b_i + d_i)/2$	$mul_i = d_i/b_i$

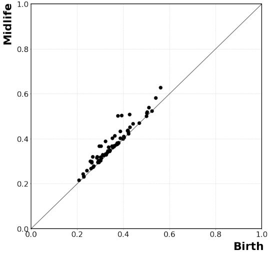
PDs in Different Coordinates



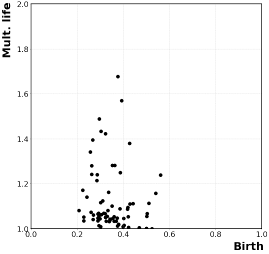
Original: x = b, y = d



Persistence: y = d - b

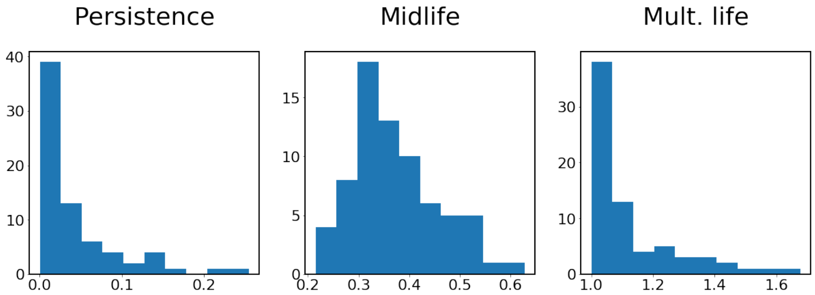


Midlife: y = (b + d) / 2



Mult.life: y = d / b

Empirical Distributions



Persistence (normalized):

$$p_i = d_i - b_i$$
$$\bar{p}_i = \frac{d_i - b_i}{\sum_i (d_i - b_i)}$$

Midlife (normalized):

$$ml_i = (b_i + d_i)/2$$
$$\bar{m}l_i = \frac{b_i + d_i}{\sum_i (b_i + d_i)}$$

Mult. life (normalized):

$$mul_i = d_i / b_i$$
$$m\bar{u}l_i = \frac{d_i / b_i}{\sum_i (d_i + b_i)}$$

Statistics

One can fix a quantity and consider a vector of statistics of its empirical distribution:

- min, max
- sum
- mean
- standard deviation, skewness, kurtosis
- median
- mean absolute deviation, 25-th and 75-th percentiles, interquartile range range
- entropy

Entropy is highest when the distribution is uniform, and lowest (exactly 0) when it is the Dirac delta function.

$$E_k(\mathbf{x}) = -\sum_{i=1}^{n} \mathbf{x}_i \log \mathbf{x}_i$$

PolynomialStatistics

Polynomials

Different authors consider polynomials[7]:

$$\sum_i b_i(d_i - b_i)$$

$$\sum_i (d_{max} - d_i)(d_i - b_i)$$

$$\sum_i b_i^2(d_i - b_i)^4$$

$$\sum_i (d_{max} - d_i)^2(d_i - b_i)^4$$

7. Adcock, Carlsson, and Carlsson, "The ring of algebraic functions on persistence bar codes," 2013.

and tropical[8] statistics:

$$\max_i d_i$$

$$\max_{i<j}(d_i + d_j)$$

$$\max_{i<j<k}(d_i + d_j + d_k)$$

$$\max_{i<j<k<l}(d_i + d_j + d_k + d_l)$$

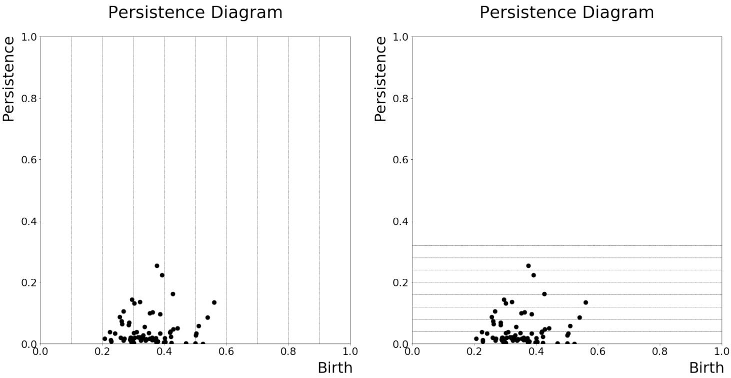
$$\sum_i d_i$$

$$\sum_i \min(28d_i, b_i)$$

$$\sum_i(\max_i(\min(28d_i, b_i) + d_i) - (\min(28d_i, b_i) + d_i))$$

8. Kališnik, "Tropical coordinates on the space of persistence barcodes," 2019.

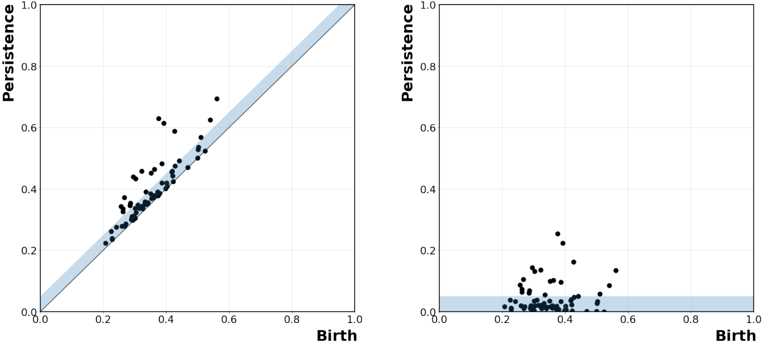
Histogram Binning



$$\phi = \text{ min, max, mean, median}$$

ε-Robustness

Set threshold ε below which data is considered as noise:

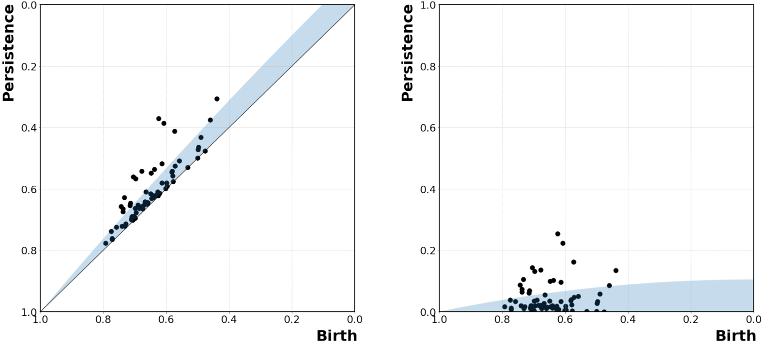


ε-Robustness as a Function of t

ε-Robustness can be generalized to depend on t. For example the error of correlation coefficient r does depend on its value:

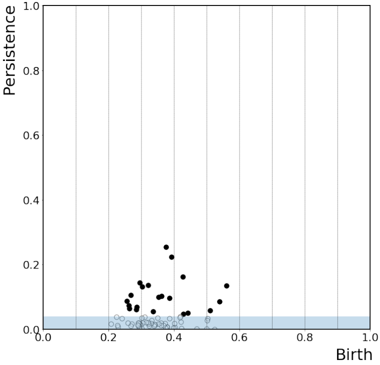
$$\mathrm{error}_r = \frac{r(z(r) + \frac{z_\alpha}{\sqrt{n-3}}) - r(z(r) - \frac{z_\alpha}{\sqrt{n-3}})}{2},$$

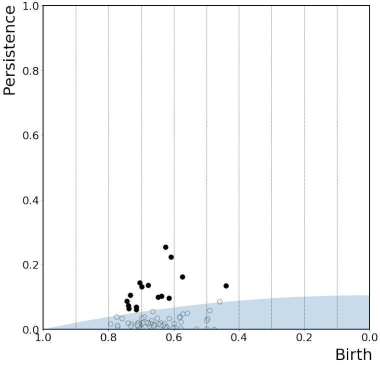
where $z(r) = \frac{1}{2} \ln \frac{1+r}{1-r}$ $\quad r(z) = -\frac{e^z - e^{-z}}{e^z + e^{-z}}$ $\quad z_\alpha(0.95) = 1.96$



ε-Robust Histogram Binning

For histogram binning, one would consider only statistics of bins of intervals with persistence greater than ε:





Persistence Curves

Persistent curve[9] is a function of persistent diagram at time t, given some aggregation and transform functions:

- Betti curve
- Euler characteristic curve
- Persistence curve
- Midlife curve
- Mult.life curve

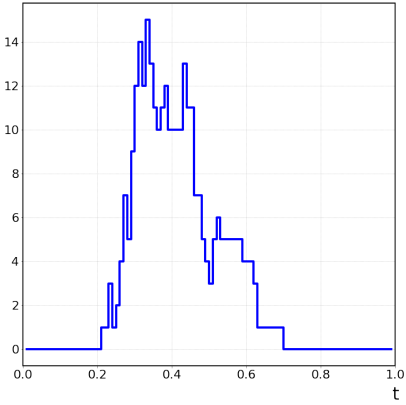
Entropy as a function of t:

- Persistence entropy curve
- Midlife entropy
- Mult.life entropy

9. Chung and Lawson, "Persistence Curves: A canonical framework for summarizing persistence diagrams," 2019.

Betti Curve

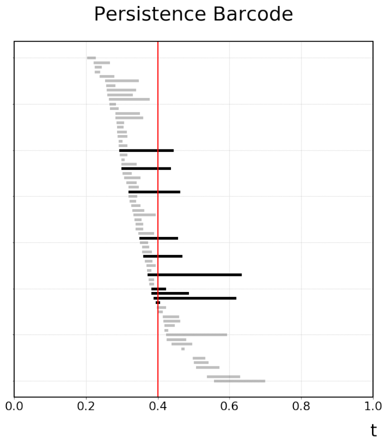
$$\beta_k(t) = \#\{(b_i, d_i)_k \mid b_i < t < d_i\}$$

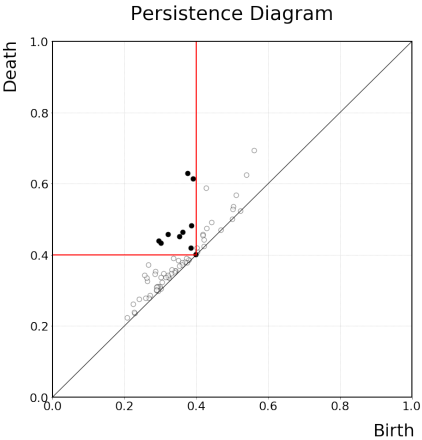


Betti Curve

Connection to Barcode and Diagram

$$\beta_k(t) = \#\{(b_i, d_i)_k \mid b_i < t < d_i\}$$





Euler Characteristic Curve

$$\chi(t) = \sum_{k=0}^{+\infty} (-1)^k \beta_k(t)$$

Persistence Curves

Other persistent curves

$$PC_k^{\phi}(t) = \left\{ \sum_i \phi(b_i, d_i)_k \mid b_i < t < d_i \right\}$$

Persistence:

$$\phi_p = \frac{d_i - b_i}{\sum_i^n (d_i - b_i)}$$

Midlife:

$$\phi_{p^+} = \frac{b_i + d_i}{\sum_i^n (b_i + d_i)}$$

Mult. life:

$$\phi_{p^*} = \frac{d_i / b_i}{\sum_i^n (d_i / b_i)}$$

Persistence Entropy

A *persistence entropy*[10], a statistic measuring the homogeneity of statistics of persistence intervals of dimension k at time t.

$$E_k(\mathbf{x})(t) = -\sum_{i=1}^{n} \mathbf{x}_i \log \mathbf{x}_i$$

$$\mathbf{x}_i(t) = \left\{ \phi(b_i, d_i)_k \;\middle|\; b_i \leq t \leq d_i \right\}$$

Persistence:

$$\phi_p = \frac{d_i - b_i}{\sum_i^n (d_i - b_i)}$$

Midlife:

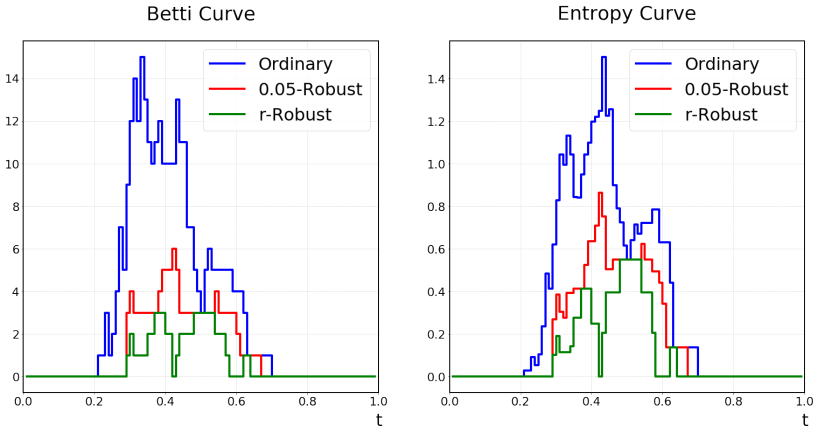
$$\phi_{p^+} = \frac{b_i + d_i}{\sum_i^n (b_i + d_i)}$$

Mult. life:

$$\phi_{p^*} = \frac{d_i / b_i}{\sum_i^n (d_i / b_i)}$$

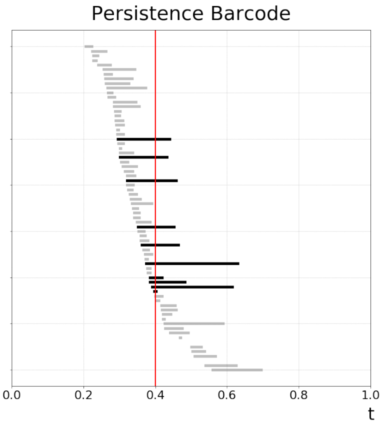
10. Atienza, Gonzalez-Diaz, and Soriano-Trigueros, "On the Stability of Persistent Entropy and New Summary Functions for TDA," 2019.

ε-Robust Persistence Curves

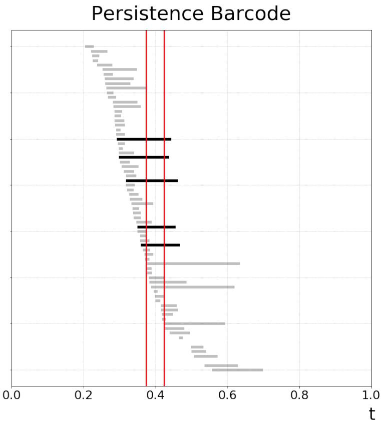


ε-Robust Persistence Curves

Connection to Barcodes



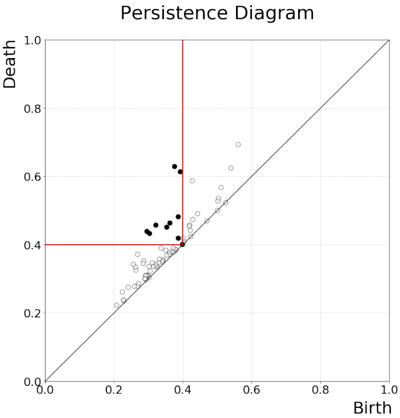
$t = 0.4, \varepsilon = 0$



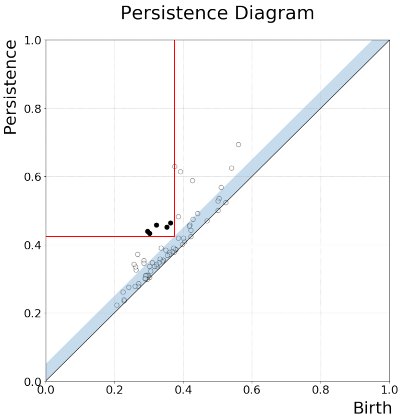
$t = 0.4, \varepsilon = 0.05$

ε-Robust Persistence Curves

Connection to Diagrams



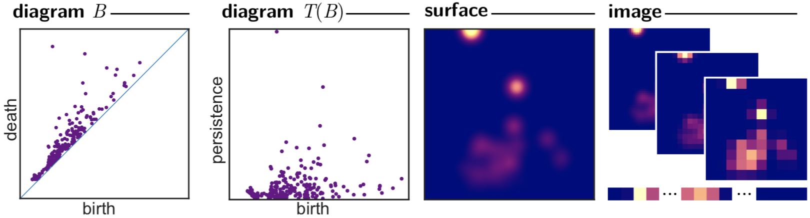
$t = 0.4, \varepsilon = 0$



$t = 0.4, \varepsilon = 0.05$

Persistence Image

A *persistence image*[11] views a persistence diagram as a 2-dimensional probability distribution and discretize its kernel density estimation by Gaussian kernel.



That is, at first, PD is transformed by $T : \mathbb{R}^2 \to \mathbb{R}^2$ to birth-persistence coordinates: $T(b, d) = (b, d - b)$.

11. Adams et al., "Persistence images: A stable Vector Representation of Persistent Homology," 2017.

Persistence Image

Second, a smooth persistence surface $\rho_{PD_k} : \mathbb{R}^2 \to \mathbb{R}_+$ of $T(PD_k)$ defined as:

$$\rho(PD_k) = \sum_{u \in T(PD_k)} f(u) g_u(z)$$

is computed, where $f(u) : \mathbb{R} \to \mathbb{R}$ is a *weighting function*, weighting points with small persistence less:

$$w_b(t) = \begin{cases} 0, & \text{if } t \leq 0, \\ t/p_{max}, & \text{if } 0 < t < p_{max}, \\ 1, & \text{if } t \geq p_{max} \end{cases}$$

Persistence Image

$$g_u(z) = \frac{1}{2\pi\sigma} \exp -[(z_x - \mu_x)^2 + (z_y - \mu_y)^2]/2\sigma^2$$

is a Gaussian kernel. Third, a discretization over a grid of desired size is obtained by integration:

$$I(\rho_{PD_k}) = \int \int \rho_{PD_k}(z_x, z_y) dz_x dz_y.$$

Persistence Landscape

A persistence landscape[11] is an embedding of a persistence diagram in a space of continuous functions L^2. That is, for each birth-death point $p = b_i, d_i \in PD_k$, a continuous piecewise linear basis function $\Lambda_p(\varepsilon)$ is:

$$\Lambda_p(t) = \begin{cases} t - b, & t \in [b, \frac{b+d}{2}] \\ d - t, & t \in (\frac{b+d}{2}, d] \\ 0, & \text{otherwise} \end{cases}$$

Given a set of basis functions of a points, a *persistence landscape* is defined $\lambda_{PD_k}(t) = \mathrm{kmax}_{p \in PD_k} \Lambda_p(t)$, where $t \in [0, T], k \in \mathbb{N}$ and kmax is the k-th largest value in the set.

---

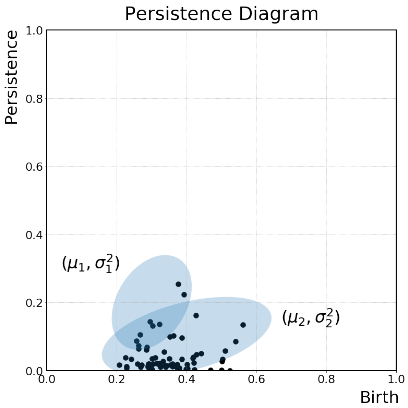
11. Bubenik and Dłotko, "A Persistence Landscapes Toolbox for Topological Statistics," 2017.

Learning task-specific vector representations

Learning with Topological Signatures

The first approach to learn vector representation of persistence diagram[12] evaluates the points against mixture of Gaussians with parameters (μ, σ) that are learned:

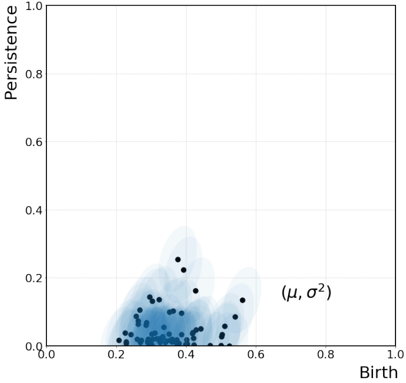
$$s_{\mu,\sigma^2} = \exp \sigma_b^2(-b - \mu_b) - \sigma_d^2(-d - \mu_d)$$



12. Hofer et al., "Deep Learning with Topological Signatures," 2017.

Learning Representations of Persistence Barcodes

In a recent work authors[13] further suggest three possible functions $\phi : \mathbb{R}^2 \to \mathbb{R}$, which roughly correspond to *Gaussian*, *spike* and *cone* functions centered on the diagram points.



13. Hofer, Kwitt, and Niethammer, "Learning representations of persistence barcodes," 2019.

PersLay: A Neural Network Layer for Persistence Diagrams

$$PersLay(Dg) := \text{op}(\{w(p) \cdot \phi(p)\}_{p \in Dg})$$

where:

- op is any *permutation invariant operation* (such as minimum, maximum, sum, k-th largest value),
- $w : \mathbb{R}^2 \to \mathbb{R}$ is a *weight function* for the persistence diagram points (can be linear, Gaussian mixture and grid), and
- $\phi : \mathbb{R}^2 \to \mathbb{R}^n$ is a *point transformation function*, mapping each point (b_i, d_i) of a persistence diagram to a vector.

W_{θ_1} and ϕ_{θ_2} are chosen from a class of differentiable functions, and parameters θ_1, θ_2 are optimized by backpropagation.[14]

14. Carriere et al., "PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures," 2019.

PersLay

Implemented point transformation functions

The *triangle point transformation* $\phi_\Lambda : \mathbb{R}^2 \to \mathbb{R}^n$, $p \mapsto [\Lambda_p(t_1), \ldots, \Lambda_p(t_n)]^T$, where the triangle function Λ_p associated to a point $p = (x, y) \in \mathbb{R}^2$ is $\Lambda_p : t \mapsto \max\{0, y - |t - x|\}$, with $q \in \mathbb{N}$ and $t_1, \ldots, t_n \in \mathbb{R}$.

The *indicator transformation* $\phi_\mathbb{I} : \mathbb{R}^2 \to \mathbb{R}^n$, $p \mapsto [\mathbb{I}_p(t_1), \ldots, \mathbb{I}_p(t_n)]^T$, where the indicator function $\mathbb{I}_p$ associated to a point $p = (x, y) \in \mathbb{R}^2$ is $\mathbb{I}_p : t \mapsto \mathbb{I}\{x < t < y\}$, with $q \in \mathbb{N}$ and $t_1, \ldots, t_n \in \mathbb{R}$.

The *Gaussian point transformation* $\phi_\Gamma : \mathbb{R}^2 \to \mathbb{R}^n$, $p \mapsto [\Gamma_p(t_1), \ldots, \Gamma_p(t_n)]^T$, where the Gaussian function Γ_p associated to a point $p = (x, y) \in \mathbb{R}^2$ is $\Gamma_p : t \mapsto \exp(-\|p - t\|_2^2 / 2\sigma^2)$ for a given $\sigma > 0$, $q \in \mathbb{N}$ and $t_1, \ldots, t_n \in \mathbb{R}^2$.

The *persistence landscape*: $\phi = \phi_\Lambda$ with samples $t_1, \ldots, t_n \in \mathbb{R}$, op $=$ k-th largest value, $w = 1$ (a constant weight function).

The *Betti curve*: $\phi = \phi_{\mathbb{I}}$ with samples $t_1, \ldots, t_n \in \mathbb{R}$, op $= \mathrm{sum}$, $w = 1$ (a constant weight function).

The *persistence image*: $\phi = \phi_\Gamma$ with samples $t_1, \ldots, t_n \in \mathbb{R}^2$, op $= \mathrm{sum}$, arbitrary weight function w.

PersLay

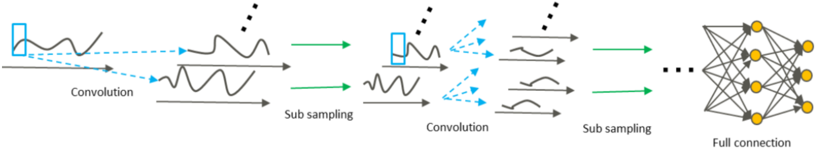
Results

	PD	PersLay
REDDIT5K	55.0	55.6(±0.3)
REDDIT12K	44.2	47.7(±0.2)
COLLAB	71.6	76.4(±0.4)
IMDB-B	68.8	71.2(±0.7)
IMDB-M	48.2	48.8(±0.6)
COX2 *	81.5	80.9(±1.0)
DHFR *	78.2	80.3(±0.8)
MUTAG *	85.1	89.8(±0.9)
PROTEINS *	72.2	74.8(±0.3)
NCI1 *	72.3	73.5(±0.3)
NCI109 *	67.0	69.5(±0.3)

Table: Classification performance

1-dimensional CNN

Umeda[15] considers 1-d CNN on Betti curves:



Datasets	Gyro sensor	EEG dataset	EMG dataset
	Accuracy		
method\validation	Leave one subject out [%]	10-fold[%]	Leave one subject out[%]
SVM+Betti sequence	63.5 ± 11.3	66.7 ± 5.6	49.6 ± 18.2
connected input 1-CNN+Betti sequence	79.8 ± 5.0	75.38 ± 5.7	74.4 ± 10.6
parallel 1-CNN+Betti sequence	86.1 ± 7.2	-	76.4 ± 7.2

Table: Classification performance

15. Umeda, "Time series classification via topological data analysis," 2017.

Deep Sets

Deep Sets[16] model $f : (\mathbb{R}^3)^N \to \mathbb{R}^d$ consists of

$$f(\{x_1, \ldots, x_N\}) = \rho\left(\sum_{i=1}^{n} \phi_\theta(x_i)\right), \quad (1)$$

- a MLP encoder $\phi_\theta : \mathbb{R}^3 \to \mathbb{R}^D$ mapping each diagram point $x_i = (b_i, d_i, h_i)$, with parameters θ shared between points,
- a permutation invariant pooling operation $(\cdot) : (\mathbb{R}^D)^N \to \mathbb{R}^D$ to obtain a representation of a diagram at whole (particularly for Deep Sets - sum pooling), and
- a decoder $\rho : \mathbb{R}^D \to \mathbb{R}^d$ which further transforms the diagram representation.

16. Zaheer et al., "Deep sets," 2017.

Transformers

Deep sets transforms individual points $\mathbf{x}_i$ in the diagram $\mathbf{X}$ independently via MLP. Self-attention transformer[17] makes each point $\mathbf{x}_i$ a nonlinear weighted combination of every point in the diagram

$$\Phi^{ATTN}_{\mathbf{W}_q,\mathbf{W}_k,\mathbf{W}_v}(\{x_1,\dots,x_n\}) = \sigma\left(\frac{(\mathbf{W}_q\mathbf{X})(\mathbf{W}_k\mathbf{X})^T}{\sqrt{D}}\right)\mathbf{W}_v\mathbf{X}, \qquad (2)$$

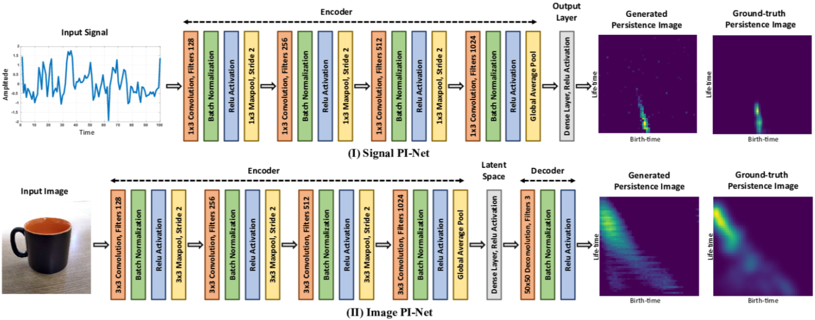
where $\Phi_{ATTN} : (\mathbb{R}^3)^N \to (\mathbb{R}^D)^N$.

17. Reinauer, Caorsi, and Berkouk, "Persformer: A transformer architecture for topological machine learning," 2021.

Learning the persistent homology map

PI-Net

PI-Net[18] learns persistence images directly from data, either 2D images or multivariate time series, after being trained on ground-truth persistence images of PDs extracted by the persistent homology algorithm.



18. Som et al., “PI-Net: A Deep Learning Approach to Extract Topological Persistence Images,” 2019.

PI-Net

Concatenating features obtained from AlexNet and Network in Network with ground-truth and learned topological features results an improvement in image classification on CIFAR10 and SVHN datasets.

Method	CIFAR10		SVHN	
	Mean±SD	p-Value	Mean±SD	p-Value
Alexnet	80.49±0.30	-	93.08±0.17	-
Alexnet + PI	80.52±0.38	0.8932	93.72±0.10	0.0001
Alexnet + Image PI-Net	**81.25±0.49**	0.0182	**93.83±0.11**	<0.0001
Alexnet + Image PI-Net FA	**81.23±0.42**	0.0125	**93.92±0.13**	<0.0001
Alexnet + Image PI-Net FS	**81.80±0.24**	0.0001	**93.94±0.13**	<0.0001
NIN	84.93±0.13	-	95.83±0.07	-
NIN + PI	85.29±0.30	0.0392	95.75±0.08	0.1309
NIN + Image PI-Net	**86.61±0.19**	<0.0001	**96.04±0.04**	0.0004
NIN + Image PI-Net FA	**86.62±0.39**	<0.0001	**95.97±0.05**	0.0066
NIN + Image PI-Net FS	**86.61±0.40**	<0.0001	**96.06±0.04**	0.0002

Table: Classification performance

PI-Net

Authors report a decrease up of two orders of magnitude in the computation time and conclude that it makes real-time TDA applications possible.

Method	Mean±SD (10^{-3} seconds)	
	CIFAR10 (50,000 images)	**SVHN (73,257 images)**
Conventional TDA - CPU	146.50±3.83	105.03±3.57
Image PI-Net - GPU	**2.52±0.02**	**2.19±0.02**

Table: Computation speed

Another paper on learning the persistent homology map[19].

19. Montúfar, Otter, and Wang, "Can neural networks learn persistent homology features?," 2020.

References I

Adams, Henry, Tegan Emerson, Michael Kirby, Rachel Neville, Chris Peterson, Patrick Shipman, Sofya Chepushtanova, Eric Hanson, Francis Motta, and Lori Ziegelmeier. "Persistence images: A stable Vector Representation of Persistent Homology." *The Journal of Machine Learning Research* 18, no. 1 (2017): 218–252.

Adcock, Aaron, Erik Carlsson, and Gunnar Carlsson. "The ring of algebraic functions on persistence bar codes." *arXiv preprint arXiv:1304.0530*, 2013.

Atienza, Nieves, Rocío Gonzalez-Diaz, and M Soriano-Trigueros. "On the Stability of Persistent Entropy and New Summary Functions for TDA." *arXiv preprint arXiv:1803.08304*, 2019.

Bubenik, Peter, and Paweł Dłotko. "A Persistence Landscapes Toolbox for Topological Statistics." *Journal of Symbolic Computation* 78 (2017): 91–114.

References II

Carriere, Mathieu, Marco Cuturi, and Steve Oudot. "Sliced wasserstein kernel for persistence diagrams." In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, 664–673. 2017.

Carriere, Mathieu, Frederic Chazal, Yuichi Ike, Theo Lacombe, Martin Royer, and Yuhei Umeda. "PersLay: A Neural Network Layer for Persistence Diagrams and New Graph Topological Signatures." *stat* 1050 (2019): 17.

Chung, Yu-Min, and Austin Lawson. "Persistence Curves: A canonical framework for summarizing persistence diagrams." *arXiv preprint arXiv:1904.07768*, 2019.

Edelsbrunner, Herbert, and John Harer. *Computational Topology: An Introduction*. American Mathematical Soc., 2010.

Hofer, C., R. Kwitt, M. Niethammer, and A. Uhl. "Deep Learning with Topological Signatures." In *NIPS*. 2017.

References III

Hofer, Christoph D, Roland Kwitt, and Marc Niethammer. "Learning representations of persistence barcodes." *Journal of Machine Learning Research* 20, no. 126 (2019): 1–45.

Kaczynski, Tomasz, Konstantin Mischaikow, and Marian Mrozek. *Computational Homology*. Vol. 157. Springer Science & Business Media, 2006.

Kališnik, Sara. "Tropical coordinates on the space of persistence barcodes." *Foundations of Computational Mathematics* 19, no. 1 (2019): 101–129.

Le, Tam, and Makoto Yamada. "Persistence fisher kernel: A riemannian manifold kernel for persistence diagrams." In *Advances in Neural Information Processing Systems*, 10007–10018. 2018.

References IV

Montúfar, Guido, Nina Otter, and Yuguang Wang. "Can neural networks learn persistent homology features?" *arXiv preprint arXiv:2011.14688*, 2020.

Otter, Nina, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington. "A roadmap for the computation of persistent homology." *EPJ Data Science* 6, no. 1 (2017): 17.

Reinauer, Raphael, Matteo Caorsi, and Nicolas Berkouk. "Persformer: A transformer architecture for topological machine learning." *arXiv preprint arXiv:2112.15210*, 2021.

Som, Anirudh, Hongjun Choi, Karthikeyan Natesan Ramamurthy, Matthew Buman, and Pavan Turaga. "PI-Net: A Deep Learning Approach to Extract Topological Persistence Images." *arXiv preprint arXiv:1906.01769*, 2019.

References V

Umeda, Yuhei. "Time series classification via topological data analysis." *Information and Media Technologies* 12 (2017): 228–239.

Zaheer, Manzil, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. "Deep sets." In *Advances in neural information processing systems*, 3391–3401. 2017.