PLLay: Efficient Topological Layer based on Persistence Landscapes

–

Topological Data Analysis

Théo Dumont [1,2] Florent Michel [1,3] Hugo Simon [1,4]

[1]ENS Paris-Saclay, France

[2]Mines Paris, France [3]Ecole Polytechnique, France [4]Télécom Paris, France

December 14th, 2021

Table of content

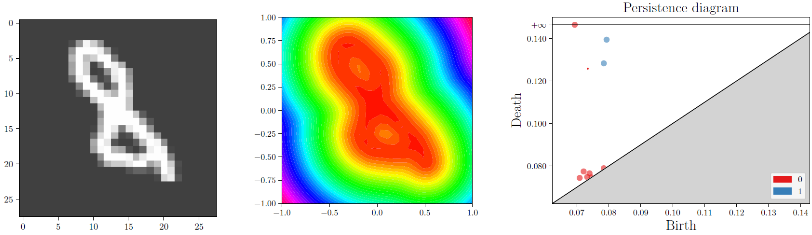


Figure 1: Digit 8 in MNIST with associated contour plot and persistence diagram induced by a specific filtration

- topology differentiates sets in topological spaces in a robust and *meaningful* geometric way
- **goal:** enhancing capacity of deep learning models using insights on the global “shape” of the data structure via *persistent homology*
- **how:** developing a layer that feeds the topological features of the data structure in an arbitrary network
- **difficulty:** combining persistent homology with statistics and machine learning

Related work

- topological **loss functions** (Poulenard et al. 2018[1], Hofer et al. 2019[2], Moor et al. 2020[3])
- topological **layers** (Hofer et al. 2017[4], Carrière et al. 2020[5])
- underlying goal: **generalization** (Carriere et al. 2021[6])

[1] Adrien Poulenard et al. (2018). "Topological function optimization for continuous shape matching". In: *Computer Graphics Forum*. Vol. 37. 5. Wiley Online Library, pp. 13–25.

[2] Christoph Hofer et al. (2019). "Connectivity-optimized representation learning via persistent homology". In: *International Conference on Machine Learning*. PMLR, pp. 2751–2760.

[3] Michael Moor et al. (2020). "Topological autoencoders". In: *International conference on machine learning*. PMLR, pp. 7045–7054.

[4] Christoph Hofer et al. (2017). "Deep learning with topological signatures". In: *arXiv preprint arXiv:1707.04041*.

[5] Mathieu Carrière et al. (2020). "Perslay: A neural network layer for persistence diagrams and new graph topological signatures". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2786–2796.

[6] Mathieu Carriere et al. (2021). "Optimizing persistent homology based functions". In: *International Conference on Machine Learning*. PMLR, pp. 1294–1303.

Related work

- topological **loss functions** (Poulenard et al. 2018[1], Hofer et al. 2019[2], Moor et al. 2020[3])
- topological **layers** (Hofer et al. 2017[4], Carrière et al. 2020[5])
- underlying goal: **generalization** (Carriere et al. 2021[6])

Limitations

1. rely on specific map or filtration
2. lack of stability results
3. differentiability not granted

---

[1] Adrien Poulenard et al. (2018). "Topological function optimization for continuous shape matching". In: *Computer Graphics Forum*. Vol. 37. 5. Wiley Online Library, pp. 13–25.

[2] Christoph Hofer et al. (2019). "Connectivity-optimized representation learning via persistent homology". In: *International Conference on Machine Learning*. PMLR, pp. 2751–2760.

[3] Michael Moor et al. (2020). "Topological autoencoders". In: *International conference on machine learning*. PMLR, pp. 7045–7054.

[4] Christoph Hofer et al. (2017). "Deep learning with topological signatures". In: *arXiv preprint arXiv:1707.04041*.

[5] Mathieu Carrière et al. (2020). "Perslay: A neural network layer for persistence diagrams and new graph topological signatures". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2786–2796.

[6] Mathieu Carriere et al. (2021). "Optimizing persistent homology based functions". In: *International Conference on Machine Learning*. PMLR, pp. 1294–1303.

Table of content

Definition (DTM)

The Distance To Measure[a] (DTM) $d_{\mu,m_0} \colon \mathbb{R}^d \to \mathbb{R}$ for a probability distribution μ and parameter $m_0 \in \left]0,1\right[$ is defined by

$$d_{\mu,m_0}(x) = \left(\frac{1}{m_0}\int_0^{m_0} (\delta_{\mu,m}(x))^2 \mathrm{d}m\right)^{1/2} = \frac{1}{\sqrt{m_0}} W_2\left(m_0\delta_x ; \mathrm{Sub}_{m_0}(\mu)\right)$$

where $\delta_{\mu,m}(x) = \inf\{r > 0 \colon \mu(B(x,r)) > m\}$ and $\mathrm{Sub}_{m_0}(\mu)$ is the set of submeasures of μ with total mass m_0.

[a] Frédéric Chazal et al. (2011). "Geometric inference for probability measures". In: *Foundations of Computational Mathematics* 11.6, pp. 733–751.

- In practice, μ is an empirical measure as a valued point cloud.
- In the case of uniformly valued point cloud, DTM is proportional to the RMS of distances to the $k = \lfloor m_0 n \rfloor$ nearest neighbours.
- DTM sublevelset filtration can be seen as an alternative to kernel estimator suplevelset filtration (m_0 analogous to smoothing parameter).

- Persistence landscapes embeds persistence diagrams in a **vector space**
  - well-defined **linear combinations**, crucial for integrating to classical **neural networks**
  - well-defined expectancy allows **statistical analysis**
- Persistence landscapes are **stable** w.r.t. persistence diagrams equipped with bottleneck distance, so it benefits from stability results for diagrams.

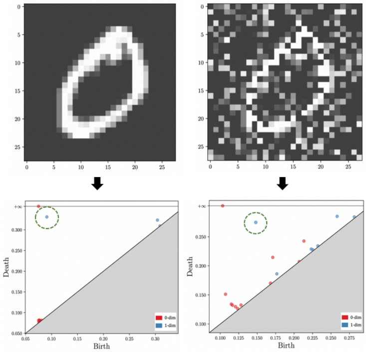


Figure 2: Diagram robustness induces landscape robustness

Algorithm 1 Implementation of Persistence Landscape Layer (PLLay)

Input: Point cloud X (optionally valued by Y), Number of landscapes K, Number of samples M, Output size H

1. compute a differentiable filtration, *e.g.* DTM sublevelset filtration
2. compute persistence diagram $\mathcal{D}$
3. compute the K first persistence landscapes of $\mathcal{D}$ on M sampling points: $\Lambda_{k,m} = \lambda_k(t_m)$
4. compute H weighted averages and apply element-wise a parametric and differentiable activation function: $g_\theta(W\Lambda)$, where $W \in \mathcal{M}_{H,K}(\mathbb{R}^+)$, $\sum_k W_{h,k} = 1$ and $g_\theta \colon \mathbb{R}^M \to \mathbb{R}$

Output: $h_{\text{top}}(X) \in \mathbb{R}^H$, which admits derivatives w.r.t. θ, W, t, X and Y

- Persistence Landscape Layer (PLLay) can be inserted anywhere in a deep learning architecture
- filtration agnostic, and accepts a huge variety of activation functions g_θ
- parameterizable complexity through K, M and H
- parameters W and θ are learned with classical automatic differentiation tools

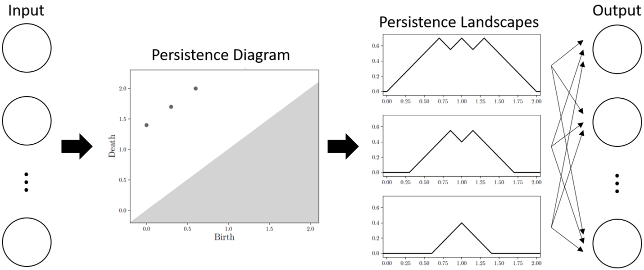


Figure 3: Illustration of PLLay. 1) the information of persistence diagram is encoded into persistence landscape (vectorized function), 2) a deep learning model determines which components of the landscape (*e.g.*, particular hills or valleys) are important for a given task during training.

Table of content

Theorem (differentiability w.r.t. input)

Let f be the filtration function. Let ξ be a map from each birth-death point $(b_i, d_i) \in \mathcal{D}_X$ to a pair of simplices (β_i, δ_i). Suppose that ξ is locally constant at X, and $f(\beta_i)$ and $f(\delta_i)$ are differentiable with respect to X_j's. Then, h_{top} is differentiable with respect to X and for all j,

$$\frac{\partial h_{top}}{\partial X_j} = F\left(\left(\frac{\partial f}{\partial X_j}, \frac{\partial f}{\partial X_j}, \frac{\partial g_\theta}{\partial x_l}, \frac{\partial \lambda_k}{\partial b_i}, \frac{\partial \lambda_k}{\partial d_i}\right)_{i,k,l}\right),$$

with F a simple linear combination.

Each of the inner partial derivatives are easily computable, except $\partial f / \partial X_j$ which depends on f. In the case of the DTM filtration, we can express it in closed form.

Note: extension of Poulenard et al. 2018[7] for arbitrary filtrations.

[7] Adrien Poulenard et al. (2018). "Topological function optimization for continuous shape matching". In: *Computer Graphics Forum*. Vol. 37. 5. Wiley Online Library, pp. 13–25.

Stability

Robustness to perturbations of the persistence diagram:

Theorem (robustness to perturbations)

Let g_θ be $\|.\|_\infty$-Lipschitz i.e. there exists $L_g > 0$ such that $|g_\theta(x) - g_\theta(y)| \leq L_g \|x - y\|_\infty$ for all $x, y \in \mathbb{R}^n$. Then for two persistence diagrams $\mathcal{D}$ and $\mathcal{D}'$:

$$\|h_{top}(\mathcal{D}) - h_{top}(\mathcal{D}')\|_\infty \leq L_g d_B(\mathcal{D}, \mathcal{D}')$$

Stability w.r.t. input:

Theorem (stability)

Let P be a distribution and P_n be the empirical distribution induced from the input X. If $\mathcal{D}_P$ and $\mathcal{D}_X$ are the persistence diagrams for the DTM filtration, then:

$$\|h_{top}(\mathcal{D}_X) - h_{top}(\mathcal{D}_P)\|_\infty \leq L_g m_0^{-1/2} W_2(P_n, P)$$

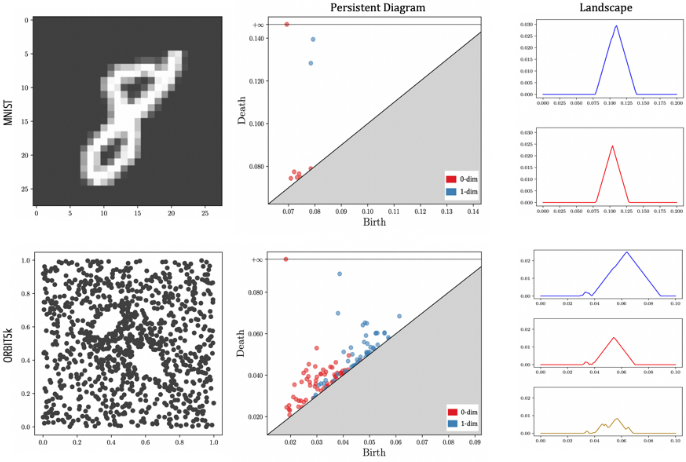


Figure 4: Classification experiments on MNIST and ORBIT5K perturbed with noise and corrupted pixels

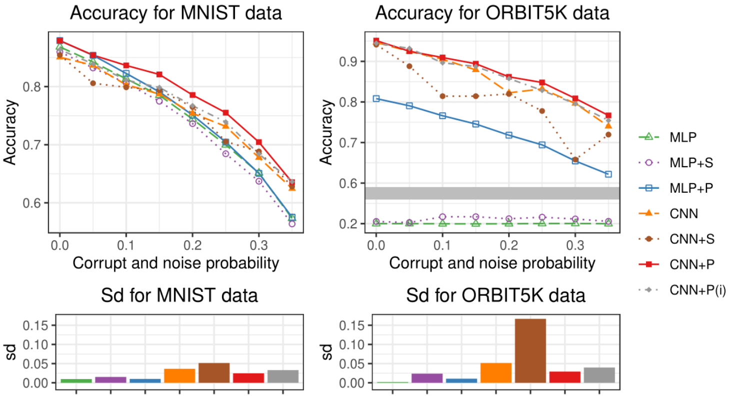


Figure 5: Test accuracy with respect to noise in `MNIST` and `ORBIT5K` experiments

- `MNIST`: better accuracy and more stable
- `ORBIT5K`: way better accuracy and clear reduction of the variance of the accuracy

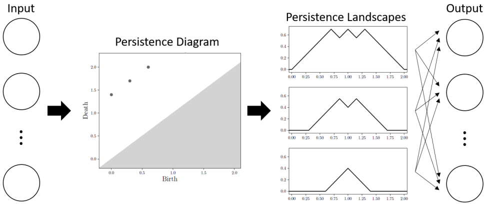
Numerical results

Model	Accuracy
PersLay (Carrière et al. 2020)	0.877 (± 0.010)
CNN + PLLay	0.950 (± 0.016)

Table 1: Comparison to PersLay on `ORBIT5K`

- Better accuracy than PersLay, the former state-of-the-art network for synthetic orbit classification coming from dynamical systems!

Table of content



Strengths

- can be placed anywhere in network
- flexibility (filtration, g_θ)
- code available
  `https://github.com/jisuk1/pllay`
- W and θ are as easy to learn as in a classical MLP

Weaknesses

- DTM: m_0 is fixed and plays a huge role in the induced topology
- automatic filtering of small topological features through K: informative in certain cases

Perspectives

- learn sampling points of landscapes λ_k?

Why this paper?

- “*How can one make TDA and deep learning interact?*”
- recent paper
- flexibility and adaptability of the method
- *Topological insights in machine learning*, Kathryn Hess, EPFL (MATH & IA, 09/03/2021)

References

Carriere, Mathieu et al. (2021). "Optimizing persistent homology based functions". In: *International Conference on Machine Learning*. PMLR, pp. 1294–1303.

Carrière, Mathieu et al. (2020). "Perslay: A neural network layer for persistence diagrams and new graph topological signatures". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2786–2796.

Chazal, Frédéric et al. (2011). "Geometric inference for probability measures". In: *Foundations of Computational Mathematics* 11.6, pp. 733–751.

Hofer, Christoph et al. (2017). "Deep learning with topological signatures". In: *arXiv preprint arXiv:1707.04041*.

Hofer, Christoph et al. (2019). "Connectivity-optimized representation learning via persistent homology". In: *International Conference on Machine Learning*. PMLR, pp. 2751–2760.

Moor, Michael et al. (2020). "Topological autoencoders". In: *International conference on machine learning*. PMLR, pp. 7045–7054.

Poulenard, Adrien et al. (2018). "Topological function optimization for continuous shape matching". In: *Computer Graphics Forum*. Vol. 37. 5. Wiley Online Library, pp. 13–25.

Thank you!

Topological loss functions

- Poulenard et al. 2018[8]

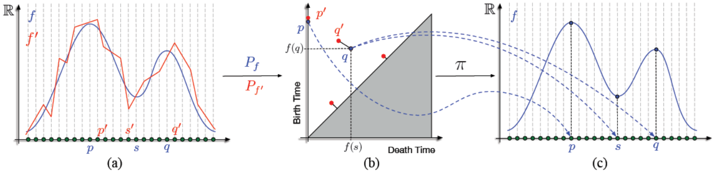


Figure 1: *(a) Two functions (f in blue and f' in red) defined on a 1-d interval. (b) Persistence diagrams of f, f'. (c) The map from the persistence diagram back to the underlying space for f.*

- Hofer et al. 2019[9]:

$$\mathcal{L}_\eta(S) = \sum_{t \in \dagger(S)} |\eta - \varepsilon_t|$$

- Moor et al. 2020[10]: reconstruction and topological loss functions

---

[8] Adrien Poulenard et al. (2018). "Topological function optimization for continuous shape matching". In: *Computer Graphics Forum*. Vol. 37. 5. Wiley Online Library, pp. 13–25.

[9] Christoph Hofer et al. (2019). "Connectivity-optimized representation learning via persistent homology". In: *International Conference on Machine Learning*. PMLR, pp. 2751–2760.

[10] Michael Moor et al. (2020). "Topological autoencoders". In: *International conference on machine learning*. PMLR, pp. 7045–7054.

Topological layers

- Hofer et al. 2017[11]

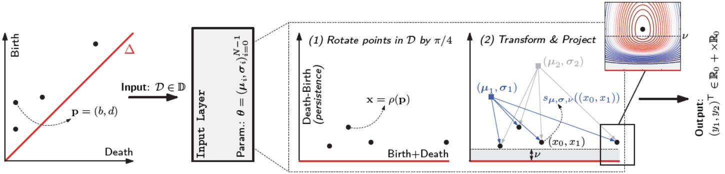


Figure 1: Illustration of the proposed network *input layer* for topological signatures. Each signature, in the form of a persistence diagram $\mathcal{D} \in \mathbb{D}$ (*left*), is projected w.r.t. a collection of *structure elements*. The layer's learnable parameters $\boldsymbol{\theta}$ are the locations $\boldsymbol{\mu}_i$ and the scales $\boldsymbol{\sigma}_i$ of these elements; $\nu \in \mathbb{R}^+$ is set a-priori and meant to discount the impact of points with low persistence (and, in many cases, of low discriminative power). The layer output $\mathbf{y}$ is a concatenation of the projections. In this illustration, $N = 2$ and hence $\mathbf{y} = (y_1, y_2)^\top$.

- Carrière et al. 2020[12]

$$\mathrm{PERSLAY}(\mathrm{Dg}) := \mathrm{op}\left(\{w(p) \cdot \varphi(p)\}_{p \in \mathrm{Dg}}\right)$$

[11] Christoph Hofer et al. (2017). "Deep learning with topological signatures". In: *arXiv preprint arXiv:1707.04041*.

[12] Mathieu Carrière et al. (2020). "Perslay: A neural network layer for persistence diagrams and new graph topological signatures". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 2786–2796.

Theorem (closed form for ∂h_{top})

Let f be the filtration function. Let ξ be a map from each birth-death point $(b_i, d_i) \in \mathcal{D}_X$ to a pair of simplices (β_i, δ_i). Suppose that ξ is locally constant at X, and $f(\beta_i)$ and $f(\delta_i)$ are differentiable with respect to X_j's. Then, h_{top} is differentiable with respect to X and

$$\frac{\partial h_{top}}{\partial X_j} = \sum_i \frac{\partial f\left(\beta_i\right)}{\partial X_j} \sum_{l=1}^{m} \frac{\partial g_\theta}{\partial x_l} \sum_{k=1}^{K_{\max}} \omega_k \frac{\partial \lambda_k(l\nu)}{\partial b_i} + \sum_i \frac{\partial f\left(\delta_i\right)}{\partial X_j} \sum_{l=1}^{m} \frac{\partial g_\theta}{\partial x_l} \sum_{k=1}^{K_{\max}} \omega_k \frac{\partial \lambda_k(l\nu)}{\partial d_i}.$$

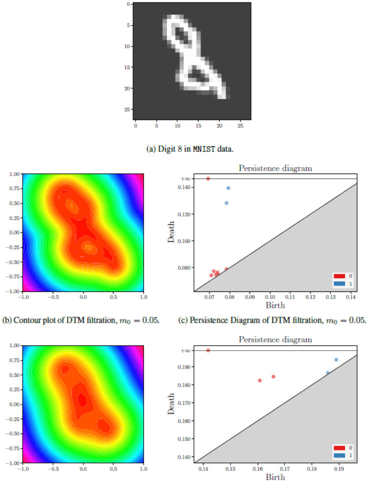
When f is the DTM filtration:

Theorem (closed form for ∂DTM)

When X_j's and ς satisfy that $\sum_{X_i \in N_k(y)} \varpi_i \left\|X_i - y_l\right\|^r$ are different for each $y_l \in \varsigma$, then $f(\varsigma)$ is differentiable with respect to X_j and

$$\frac{\partial f(\varsigma)}{\partial X_j} = \frac{\varpi_j' \left\|X_j - y\right\|^{r-2} \left(X_j - y\right) I\left(X_j \in N_k(y)\right)}{\left(\hat{d}_{m_0}(y)\right)^{r-1} m_0 \sum_{i=1}^{n} \varpi_i}$$

where I is an indicator function and $y = \arg\max_{z \in \varsigma} \hat{d}_{m_0}(z)$. In particular, f is differentiable a.e. with respect to Lebesgue measure on X.



(a) Digit 8 in MNIST data.

(b) Contour plot of DTM filtration, $m_0 = 0.05$. (c) Persistence Diagram of DTM filtration, $m_0 = 0.05$.

(d) Contour plot of DTM filtration, $m_0 = 0.2$. (e) Persistence Diagram of DTM filtration, $m_0 = 0.2$.

Figure 6: When $m_0 = 0.05$, DTM filtration aggregates more locally, and the 1st persistent homology extracts two loop structures of the digit 8. When $m_0 = 0.2$, DTM filtration aggregates the digit 8 more globally, and the 0th persistent homology extracts three connected component structures of the digit 8.