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

Qi Zhao and Yusu Wang
The Ohio State University, Department of Computer Science and Engineering

Introduction

Persistent homology summarizes topological features of a domain X in a summary called the *persistence diagrams*. 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.

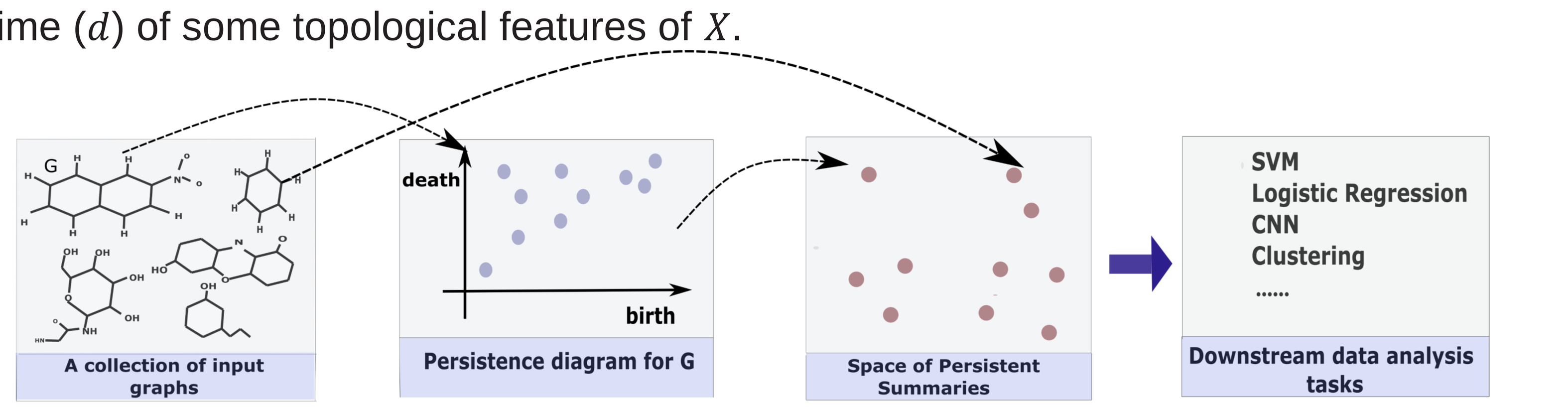

Figure 1: A persistence-based data analysis framework

Vector representations of persistence diagrams:

○ Persistence Images (PI)
○ Persistence Weighted Gaussian kernel (PWGK)
○ Sliced Wasserstein kernel (SW)
……

Our contributions

○ Different from previous approaches applying pre-set weight functions on persistence points, we propose a metric learning method to learn weight functions from labeled data.
○ We propose a new Weighted Persistence Image kernel (WKPI) for persistence summaries based on PI, and proof its positive semi-definiteness and stability.
○ We apply the learned kernel to the challenging graph classification task, achieve or outperform the state-of-art.

Persistence-based framework

Given a shape X and a descriptor function $f: X \to \mathbb{R}$:
Sublevel-set: $X_{\leq a} \coloneqq \{x \in X \mid f(x) \leq a\}$.
Filtration: $X_{\leq a_1} \subseteq X_{\leq a_2} \subseteq \cdots \subseteq X_{\leq a_n}$, where $a_1 < a_2 < \cdots a_n$ are n real values.

Inspect X through the filtration, when a topological feature appears in $X_{\leq a_i}$ and disappears upon entering $X_{\leq a_j}$, captures it as a *persistence point* $p = (a_i, a_j)$ where a_i denotes its birth while a_j is the death. A persistent diagram consists of all persistence points and the diagonal in the birth-death plane. Please read [1] for details.
Given a graph $G = (V, E)$, and a descriptor function f defined on V or E. The 0-dimensional persistence diagrams encodes connected components' creation and destruction, and the cycles features are recorded by the 1-dimensional extended persistence diagrams [2].

Persistence Images: A vectorization of persistence diagrams for machine learning tasks, relevant to persistence points locations. See [3] for details.

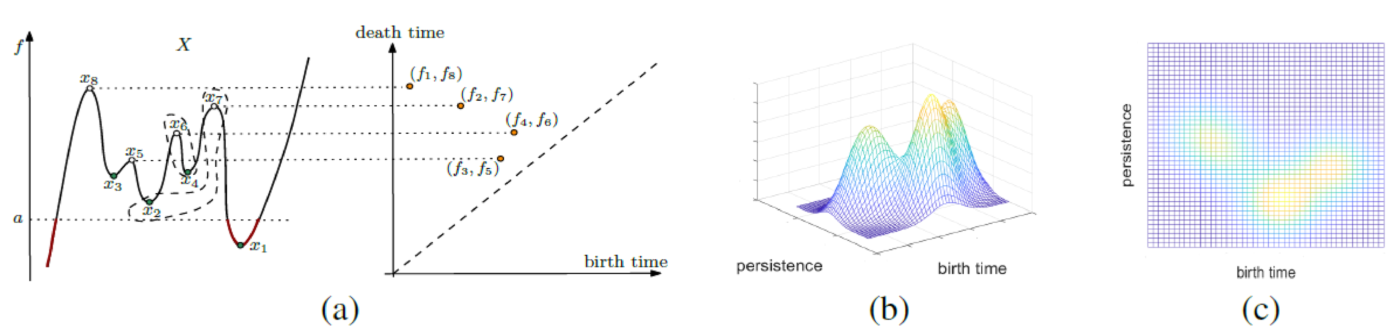

Figure 2: (a) The generation of a persistence diagram. (b) and (c) show the graph of the corresponding persistence surface and image.

Metric learning frameworks

Given a set of n objects classified into k classes, fix the grid $\mathcal{P}$ to generate persistence images. Let p_s be the center of the s-th pixel $\boldsymbol{p}_s$ in $\mathcal{P}$ for $s \in \{1, 2, \dots, N\}$.

Definition 1: *Let* $w: \mathbb{R}^2 \to \mathbb{R}$ *be a non-negative weight function. Given two persistence images* PI *and* PI', *the* weighted persistence image kernel (WKPI) *is defined as*

$$k_w(PI, PI') \coloneqq \sum_{S=1}^{N} w(p_s) e^{-\frac{(PI(s) - PI'(s))^2}{2\sigma^2}}$$

Lemma 2: *The WKPI kernel is positive semi-definite.*

Definition 3*: Given two persistence diagrams* A *and* B, *let* PI_A *and* PI_B *be their persistence images. Given a weight function* $w: \mathbb{R}^2 \to \mathbb{R}$, *the WKPI-distance is:*

$$D_w(A, B) \coloneqq \sqrt{k_w(PI_A, PI_A) + k_w(PI_B, PI_B) - 2k_w(PI_A, PI_B)}$$

This is the pseudo-metric induced by the inner product on Hilbert space.

Theorem 4: (Stability) *Given a weight function* $w: \mathbb{R}^2 \to \mathbb{R}$, $c_w = ||w||_\infty = sup_{z \in \mathbb{R}^2} w(z)$. *Given two persistence diagrams* A *and* B, *with persistence images* PI_A *and* PI_B, *we have* $D_w(A, B) \leq \sqrt{\frac{20 c_w}{\pi}} \frac{1}{\sigma \tau} d_{W,1}(A, B)$, *where* σ *is the width of the Gaussian used to define our WKPI kernel, and* $d_{W,1}(,)$ *is the 1-Wasserstein distance between persistence diagrams.*

Optimization problem for metric-learning

Given a set of graphs $\Xi = \{X_1, X_2, \dots, X_n\}$ already classified to k classes $C_1, C_2, \dots, C_k$, let $\{\Lambda_1, \Lambda_2, \dots, \Lambda_n\}$ be the resulting set of persistence diagrams. Given a weight function w, the in-class distances for C_t and total distance from objects in C_t to all objects in Ξ are

$$cost_w(t, t) = \sum_{i, j \in C_t} D_W^2(\Lambda_i, \Lambda_j); \qquad cost_w(t, \cdot) = \sum_{i \in C_t, j \in \{1, 2, \dots, n\}} D_W^2(\Lambda_i, \Lambda_j)$$

Given a weight function $w: \mathbb{R}^2 \longrightarrow \mathbb{R}$, the total-cost of induced *WKPI-distance*:

$$TC(w) \coloneqq \sum_{t=1}^{k} \frac{cost(t, t)}{cost(t, \cdot)}$$

Aim to find the weight function w^* from a certain function class $\mathcal{F}$ such that:

$$w^* = argmin_{w \in \mathcal{F}} TC(w)$$

In our experiments below, $\mathcal{F}$ is set as mixture Gaussian distribution functions, it becomes a parametric optimization problem, and can be solved by (stochastic) gradient descent algorithms.

Experiments

Neuron tree dataset

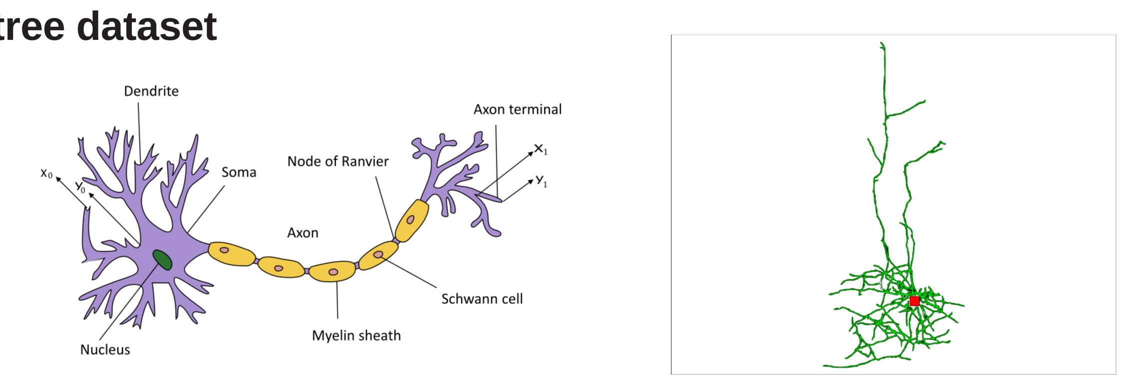

Figure 3: A neuron cell (left) downloaded from Wikipedia and an example of a neuron tree (right) from the dataset

Given a neuron tree, set the descriptor function as $f: T \longrightarrow \mathbb{R}$, where $f(x)$ is the geodesic distance from x to the root of T along the tree. Baselines are persistence-based approaches, including Persistence Weighted Gaussian kernel (PWGK), PI, SW and the PWGK with the trained weight function by optimizing the same metric cost function (trainPWGK).

Datasets	PWGK	SW	PI	trainPWGK	WKPI-kM	WKPI-kC
Neuron-binary	80.5±0.4	85.3±0.7	83.7±0.3	84.6±2.4	89.6±2.2	86.4±2.4
Neuron-multi	45.1±0.3	57.6±0.6	44.2±0.3	49.7±2.4	56.6±2.7	59.3±2.3

Table 1: Classification accuracies on two different neuron tree dataset. –kM and –kC are two different initialization methods of choosing mixture Gaussian centers: the k-means or k-centers or all persistence points in training dataset.

Benchmarks of graph classification

(1) Graph datasets derived from small chemical components or protein molecules.
(2) Graph datasets representing the relations or networks in Reddit or IMDB.
Baselines are 3 graph kernels: RetGK, WL, P-WL-UC; and 2 GNNs: PSCN, GIN

Datasets	RetGK	WL	P-WL-UC	PSCN	GIN	WKPI-kM	WKPI-kC
NCI1	84.5±0.2	85.4±0.3	85.6±0.3	76.3±1.7	82.7±1.6	87.5±0.5	84.5±0.5
PTC	62.5±1.6	55.4±1.5	63.5±1.6	62.3±5.7	66.6±6.9	62.7±2.7	68.1±2.4
PROTEIN	75.8±0.6	71.2±0.8	75.9±0.8	75.0±2.5	76.2±2.6	78.5±0.4	75.2±0.4
DD	81.6±0.3	78.6±0.4	78.5±0.4	76.2±2.6	-	82.0±0.5	80.3±0.4
MUTAG	90.3±1.1	84.4±1.5	85.2±0.3	89.0±4.4	90.0±8.8	85.8±2.5	88.3±2.6
IMDB-binary	71.8±1.0	70.8±0.5	73.0±1.0	71.0±2.3	75.1±5.1	70.7±1.1	75.1±1.1
REDDIT-5K	56.1±0.5	51.2±0.3	-	49.1±0.7	57.5±1.5	59.1±0.5	59.5±0.6

Table 2: Classification accuracies on benchmark graph datasets.

Reference:

[1] H. Edelsbrunner and J. Harer. *Computational Topology: an Introduction*. American Mathematical Society, 2010
[2] D.Cohen-Steiner, H. Edelsbrunner and J. Harer. Extending persistence using Poincare and Lefschetz duality. *Foundations of computational MatheMatics*, 9(1):79-103 2009.
[3] H. Adams, T. Emerson, M. Kirby, R. Neville, C. Peterson, P. Shipman, S. Chepushtanova, E. Hanson, F.Motta, and L. Ziegelmeier. Persistence images: a stable vector representation of persistent homology. *Journal of Machine Learning Research*, 2017

THE OHIO STATE UNIVERSITY