TDA and Machine Learning

TDA Journal Club, 27.11.2019, Sebastian Damrich

Outline

1. Persistent Homology as input to Machine Learning Methods

   Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252.

2. Learnt Topological Features

   Hofer, C. et al. (2017). Deep learning with topological signatures. In Advances in Neural Information Processing Systems (pp. 1634-1644).

3. Differentiable Persistent Homology

   Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200.

4. Neural Persistence

   Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR

1. PH as input for Machine Learning

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252.

Typical Machine Learning algorithms take distances, similarities (kernels) or features as input

Persistent Homology is represented as diagram or a barcode → transformation necessary

Data → Persistent Homology → Kernel / Vectorised Features → Machine Learning

Persistent Homology offers global features, often Machine Learning focuses on local features

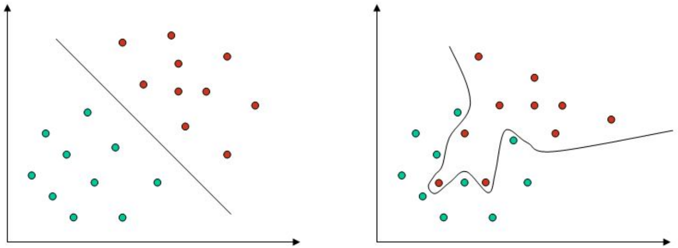
Distances

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252

Wasserstein / Bottleneck distances between Persistence Diagrams

- e.g. k-Nearest Neighbour classification (majority vote of k labelled points nearest to query)
- used to compute kernels

Support Vector Machines



Source: https://upload.wikimedia.org/wikipedia/de/a/a0/Diskriminanzfunktion.png

Classification method

Main steps:

- Optionally: Map data into high dimensional Hilbert space with map F
- Learns a hyperplane separating the classes during training
- Classify new points according to side of the hyperplane

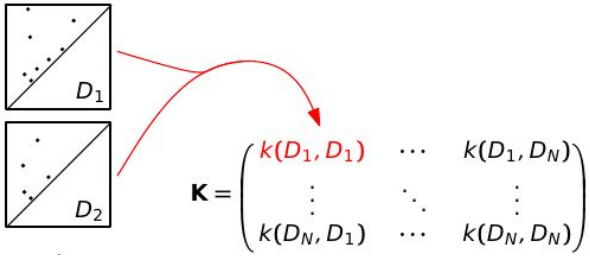
Kernel trick

- Only ever needs inner product between data points
- $\rightarrow k(x_i, x_j) = \langle F(x_i), F(x_j) \rangle$

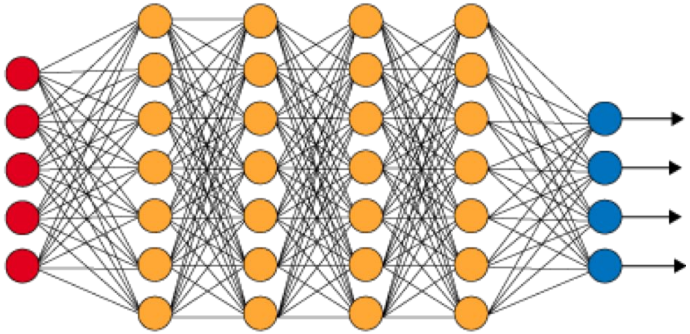
Kernels from Persistence Diagrams

Persistence scale-space kernel Reininghaus, J. et al. (2015). A stable multi-scale kernel for topological machine learning. CVPR

- Convolve Pers Diag (and its negative mirror by the diagonal) with Gaussian of scale σ
  - → implicitly maps input data to infinite dimensional space
- Inner product of two such smoothed Pers Diags gives similarity
- 1-Wasserstein stable
- Application: shape classification with SVM
- Slow to compute



Deep Neural Networks



Source: https://dvl.in.tum.de/teaching/i2dl-ws18/

State-of-the-art Machine Learning method

Each layer is linear function parametrised with "weights" followed by non-linear function

Typically non-linear function are ReLUs, i.e. max(x, 0)

Train via gradient descent minimising a loss function

Output layer is softmax for classification

Needs vectorised input

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Softmax function,
source: https://en.wikipedia.org/wiki/Softmax_function

Vectorised features from Pers. Diagrams

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252

Hand-crafted:

e.g. birth/death/persistence of the (second/third) longest Betti 0/1/2 bar.

Cang, Z. et al. (2015). A topological approach for protein classification. Computational and Mathematical Biophysics, 3(1).

Summarising statistics:

e.g.
$$\mathcal{E}(p, q, i_0; \mathrm{PD}) = \sum_{i=i_0}^{\infty} |d_i - b_i|^p \left(\frac{d_i + b_i}{2}\right)^q$$

Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200.

Binned version of barcode

subdivide the filtration range into bins and count features within each bin (#births, #deaths, rank)

Application: Protein Classification

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252..

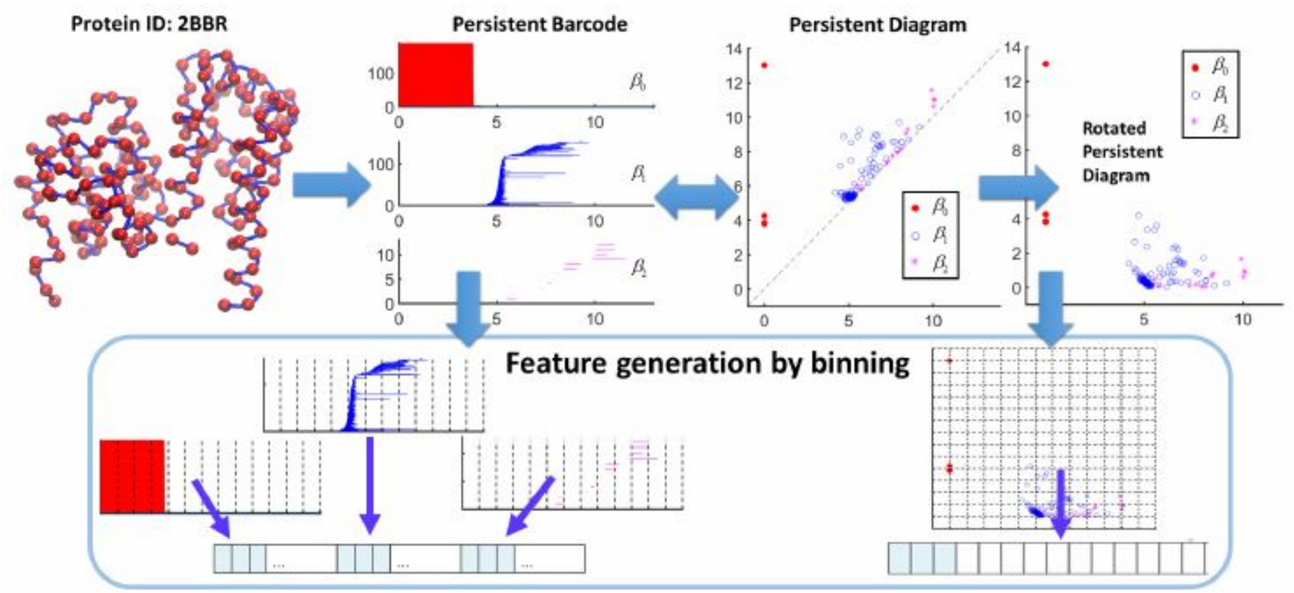
Aim: Train a classifier to distinguish 3 types for different proteins

Available data: molecular structure of labelled proteins

Topological features: 13 hand-crafted features / binned # deaths, # births and ranks
of Pers. Homology in dimensions 0-2 of Risp complex

Application: Protein Classification

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252..



Application: Protein Classification

Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252..

Aim: Train a classifier to distinguish 3 types for different proteins

Available data: molecular structure of labelled proteins

Topological features: 13 hand-crafted features / binned # deaths, # births and ranks
of pers. homology in dimensions 0-2 of Risp complex

Results:

Binned features yield better performance
SMV: 87% accuracy
NN: 91 % accuracy

2. Learnable Features

Hofer, C. et al. (2017). Deep learning with topological signatures. In Advances in Neural Information Processing Systems (pp. 1634-1644).

Main idea

Use learnt, not hand-crafted features in a natural setting for topological features selection

Place Gaussians in Persistence Diagram and compute score at feature points

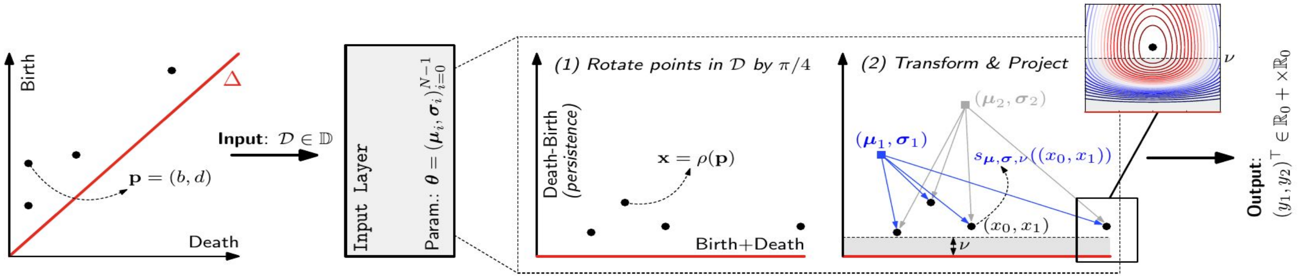


Illustration of point weighing with learnable Gaussians

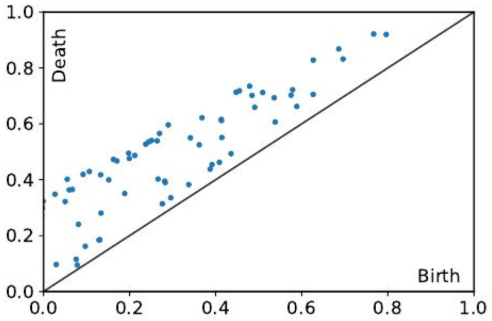
2. Learnable Features

Hofer, C. et al. (2017). Deep learning with topological signatures. In Advances in Neural Information Processing Systems (pp. 1634-1644).

Concatenate sums of scores of points in Pers. Diagram for different Gaussians
→ input to a Neural Network

Learnt features are 1-Wasserstein stable

Learn position and covariance of the Gaussians via gradient descent



Learnt locations of Gaussians

Formula for scoring function

$$s_{\boldsymbol{\mu},\boldsymbol{\sigma},\nu}\big((x_0, x_1)\big) = \begin{cases} e^{-\sigma_0^2(x_0-\mu_0)^2-\sigma_1^2(x_1-\mu_1)^2}, & x_1 \in [\nu, \infty) \\ e^{-\sigma_0^2(x_0-\mu_0)^2-\sigma_1^2(\ln(\frac{x_1}{\nu})+\nu-\mu_1)^2}, & x_1 \in (0, \nu) \\ 0, & x_1 = 0 \end{cases}$$

Weighing function. A Gaussian far from diagonal, logarithmic deformation close to diagonal

Application: Graph classification

Hofer, C. et al. (2017). Deep learning with topological signatures. In Advances in Neural Information Processing Systems (pp. 1634-1644).

Data: ~ 12k graphs representing reddit threads for 11 topics
vertices = users, edge = there is a direct question-response interaction

Filtration by increasing vertex degree (edge weight = maximum degree of adjacent vertex)

Persistence Diagrams for dimensions 0 and 1

Baseline with SVM on feature vectors consisting of the N longest persistences

Results for graph classification

	N						Ours
	5	10	20	40	80	160	
MPEG-7	81.8	82.3	79.7	74.5	68.2	64.4	**91.8**
Animal	48.8	50.0	46.2	42.4	39.3	36.0	**69.5**
reddit-5k	37.1	38.2	39.7	42.1	43.8	45.2	**54.5**
reddit-12k	24.2	24.6	27.9	29.8	31.5	31.6	**44.5**

	reddit-5k	reddit-12k
GK [31]	41.0	31.8
DGK [31]	41.3	32.2
PSCN [24]	49.1	41.3
RF [4]	50.9	42.7
Ours (w/o essential)	49.1	38.5
Ours (w/ essential)	**54.5**	**44.5**

Left: Accuracy of SVM on vectorised topological features compared to learnt features
Right: Comparison to state-of-the-art, including graph convolutional neural network which use the vertex degree

3. Differentiable Persistent Homology

Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200

Main idea: Make computation of Persistent Homology differentiable

Find the simplices responsible for the birth / death of a bar $\pi_f(k) : \{b_i, d_i\}_{i \in \mathcal{I}_k} \to (\sigma, \tau)$

Find points that create the simplices $\omega_{\mathcal{R}}(\sigma) = \text{argmax}_{(v,w)\in\sigma} ||v - w||$

Loss function: $\mathcal{E}(p, q, i_0; \text{PD}) = \sum_{i=i_0}^{\infty} |d_i - b_i|^p (\frac{d_i + b_i}{2})^q$

Given gradient d loss / d b_i, backpropagate to gradient d loss / d v and d loss / d w

Implementation still on CPU ...

Applications of differentiable PH

Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200

Differentiable Persistent Homology can be used at every point in deep learning pipeline:

as loss, e.g. fine tune generative model to output a single connected component

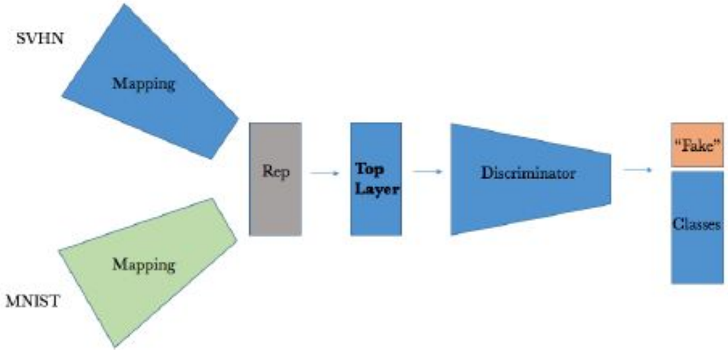
Left: Output of basic generative model $\mathcal{E}(1,0,2;\mathrm{PD}_0)$
Right: Output after fine-tuning with loss

Applications of differentiable PH

Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200

Differentiable Persistent Homology

as intermediate layer in neural network to allow beneficial preprocessing



Left: Schematic architecture for classifier network with intermediate Topological Layer
Right: Representation before applying the Topological Layer, digits have been thickened

4. Neural Persistence

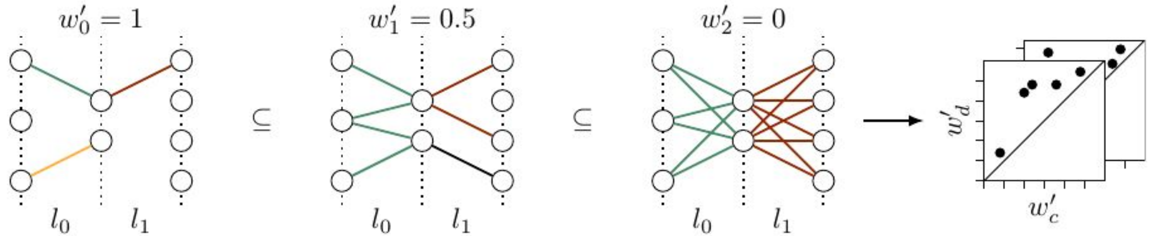
Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR

Apply Persistent Homology to Neural Network graph instead of data

Obtain complexity measure of the network

p-norm of vector of 0-dimensional persistences per layer on a filtration of decreasing absolute weight

Normalise layer persistency by number of vertices and average over layers

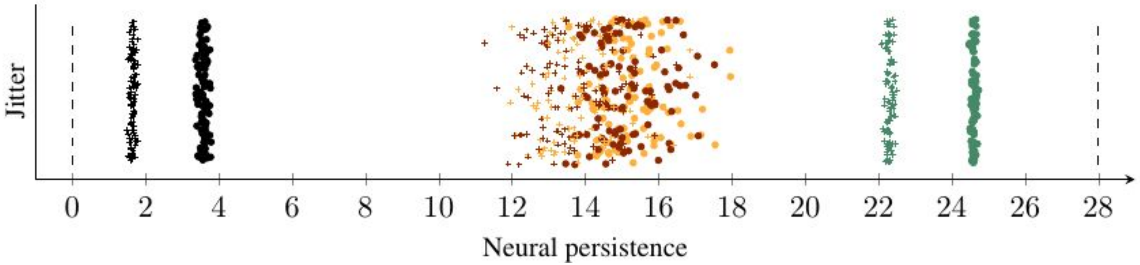


Filtration of simplicial complex associated to weighted network layer

Applications of Neural Persistence

Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR

Trained networks have higher Neural Persistence



Black dots uniform initialisation, red dots Gaussian initialisation, yellow dots diverging networks, green dots converged networks

Application: Early stopping

Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR

Early stopping necessary to prevent overfitting for Neural Networks

Early stopping based on evaluation loss:

- Stop training if evaluation loss does not decrease for some training iterations

Early stopping based on Neural Persistence:

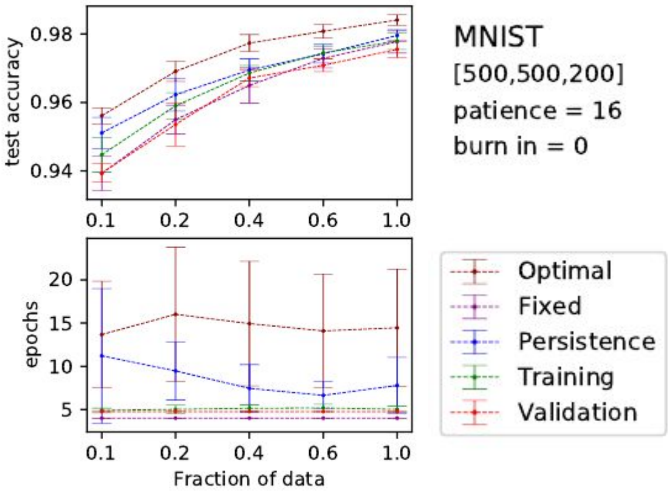
- Stop training if Neural Persistence does not increase for some training iterations
- Small loss in accuracy and earlier stopping compared to evaluation loss stopping (on same data)
- More available training data

Early Stopping with Neural Persistence

Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR

If data is scarce, the additional training data for Neural Persistence stopping results in beneficial later stopping times

But: Empirically does not work for CNNs!



Summary

1. Transform Persistent Homology information to a kernel or a featurised vector to feed it to Machine Learning pipelines
2. Learnt features outperform hand-selected ones
3. Differentiable Persistent Homology can be used in many scenarios of deep learning

→ Combine 2. and 3. for truly end-to-end learnable pipeline

4. Persistent Homology is used to analyse Neural Networks

References

1. Pun, C. S. et al. (2018). Persistent-Homology-based Machine Learning and its Applications--A Survey. arXiv preprint arXiv:1811.00252.
2. Reininghaus, J. et al. (2015). A stable multi-scale kernel for topological machine learning. CVPR
3. Hofer, C. et al. (2017). Deep learning with topological signatures. In Advances in Neural Information Processing Systems (pp. 1634-1644).

4. Brüel-Gabrielsson, R. et al. (2019). A Topology Layer for Machine Learning. arXiv preprint arXiv:1905.12200.

5. Rieck, B. et al. (2018). Neural persistence: A complexity measure for deep neural networks using algebraic topology. ICLR