HELMHOLTZ MUNICH
AIH Institute of AI for Health

Geometrical–Topological Loss Terms for Shape Analysis

Bastian Rieck (@Pseudomanifold)

Shape analysis is crucial, not only for biomedical applications.

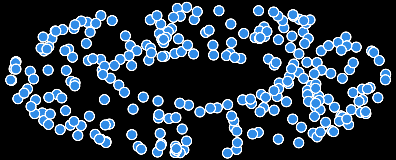
Data has shape, shape has meaning, and meaning drives understanding.
(Paraphrasing Gunnar Carlsson)

Shape analysis for meshes

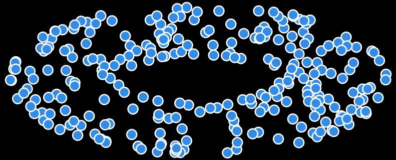
Shape analysis for meshes

HELMHOLTZ MUNICH

Reality is often messy...

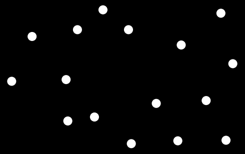


Reality is often messy...



Calculating simplicial complexes from data

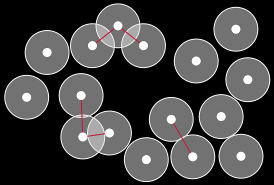
Vietoris–Rips complex



$$\mathcal{V}_\epsilon := \{\{x_1, x_2, \ldots\} \mid \operatorname{dist}\left(x_i, x_j\right) \leq \epsilon \text{ for all } i \neq j\}$$

Calculating simplicial complexes from data

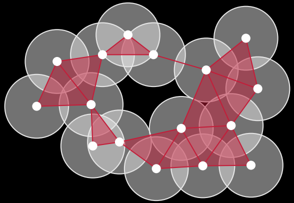
Vietoris–Rips complex



$$\mathcal{V}_\epsilon := \{\{x_1, x_2, \ldots\} \mid \mathrm{dist}\left(x_i, x_j\right) \leq \epsilon \text{ for all } i \neq j\}$$

Calculating simplicial complexes from data

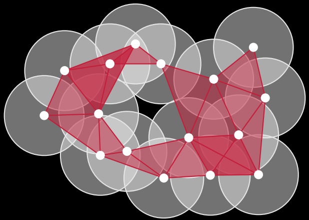
Vietoris–Rips complex



$$\mathcal{V}_\epsilon := \{\{x_1, x_2, \dots\} \mid \mathrm{dist}\left(x_i, x_j\right) \leq \epsilon \text{ for all } i \neq j\}$$

Calculating simplicial complexes from data

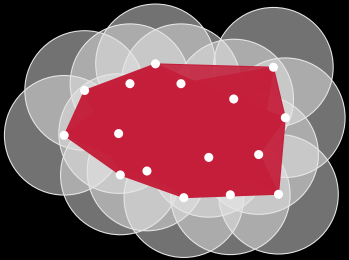
Vietoris–Rips complex



$$\mathcal{V}_\epsilon := \{\{x_1, x_2, \ldots\} \mid \operatorname{dist}\left(x_i, x_j\right) \leq \epsilon \text{ for all } i \neq j\}$$

Calculating simplicial complexes from data

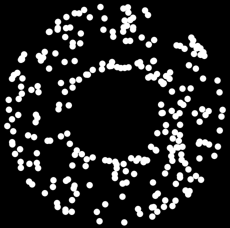
Vietoris–Rips complex



$$\mathcal{V}_\epsilon := \{\{x_1, x_2, \ldots\} \mid \mathrm{dist}\left(x_i, x_j\right) \leq \epsilon \text{ for all } i \neq j\}$$

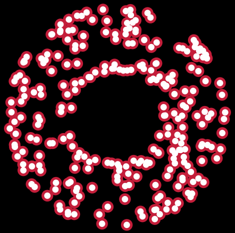
Persistent homology

Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on ‘when’ it was created and ‘when’ it was destroyed. Store these features in a *persistence diagram*.



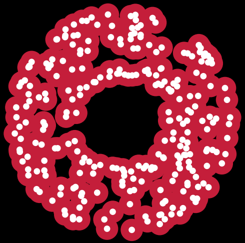
Persistent homology

Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on ‘when’ it was created and ‘when’ it was destroyed. Store these features in a *persistence diagram*.



Persistent homology

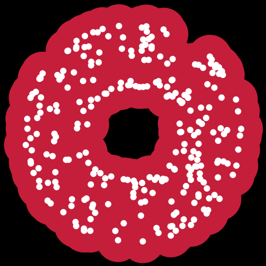
Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on 'when' it was created and 'when' it was destroyed. Store these features in a *persistence diagram*.



HELMHOLTZ MUNICH

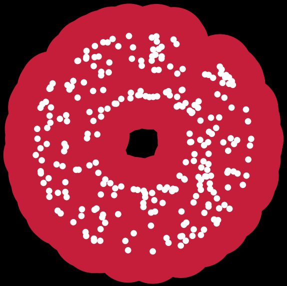
Persistent homology

Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on 'when' it was created and 'when' it was destroyed. Store these features in a *persistence diagram*.



Persistent homology

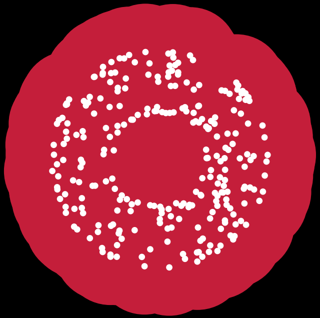
Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on 'when' it was created and 'when' it was destroyed. Store these features in a *persistence diagram*.



HELMHOLTZ MUNICH

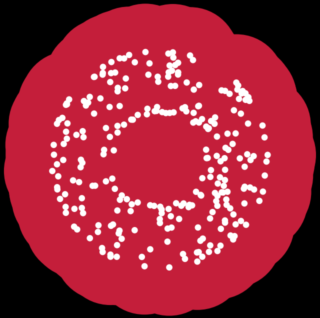
Persistent homology

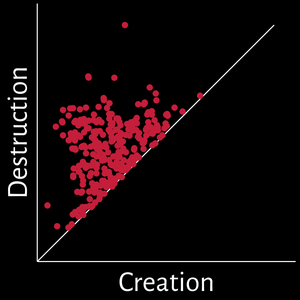
Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on 'when' it was created and 'when' it was destroyed. Store these features in a *persistence diagram*.



Persistent homology

Calculate simplicial complexes for *every* value of ϵ, while watching how topological features change. Assign each feature a duration, depending on 'when' it was created and 'when' it was destroyed. Store these features in a *persistence diagram*.





Distances between persistence diagrams

Bottleneck distance

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

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

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

Wasserstein distance

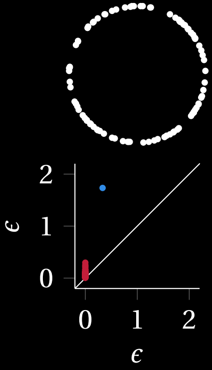
$$\mathrm{W}_p(\mathscr{D}_1, \mathscr{D}_2) := \left(\inf_{\eta\colon \mathscr{D}_1 \to \mathscr{D}_2} \sum_{x \in \mathscr{D}_1} \| x - \eta(x) \|_\infty^p \right)^{\frac{1}{p}}$$

HELMHOLTZ MUNICH

Stability theorem

Robustness to *small-scale* perturbations

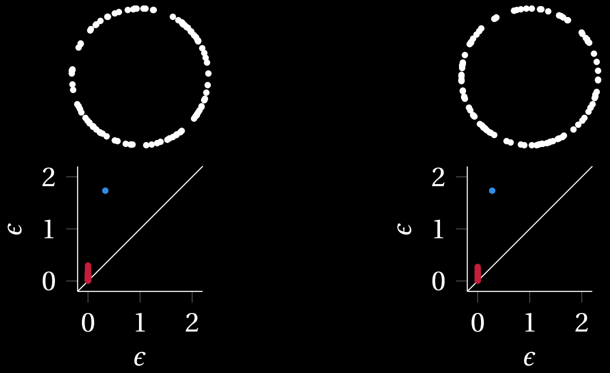
Let $\mathbb{M}$ be a triangulable space with continuous tame functions $f, g\colon \mathbb{M} \to \mathbb{R}$. Then the corresponding persistence diagrams satisfy $W_\infty\left(\mathcal{D}_f, \mathcal{D}_g\right) \leq \|f - g\|_\infty$.



Stability theorem

Robustness to *small-scale* perturbations

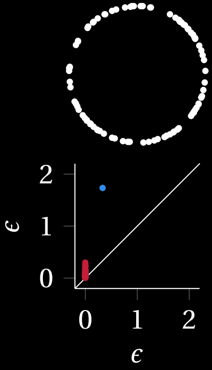
Let $\mathbb{M}$ be a triangulable space with continuous tame functions $f, g\colon \mathbb{M} \to \mathbb{R}$. Then the corresponding persistence diagrams satisfy $\mathrm{W}_\infty\left(\mathscr{D}_f, \mathscr{D}_g\right) \leq \|f - g\|_\infty$.

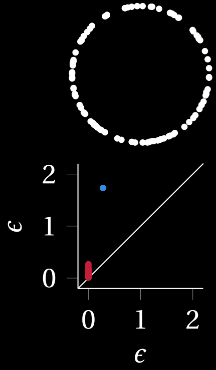


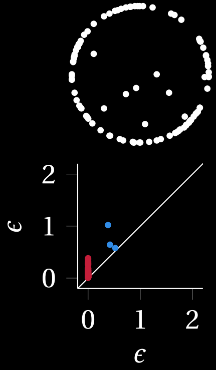
Stability theorem

Robustness to *small-scale* perturbations

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

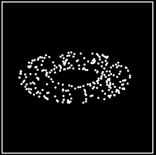






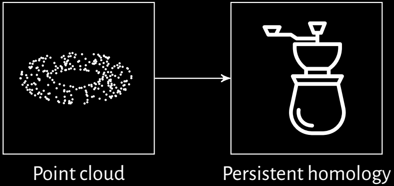
HELMHOLTZ MUNICH

A pipeline for topological machine learning



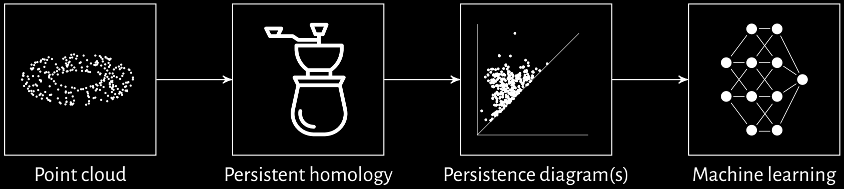
Point cloud

A pipeline for topological machine learning

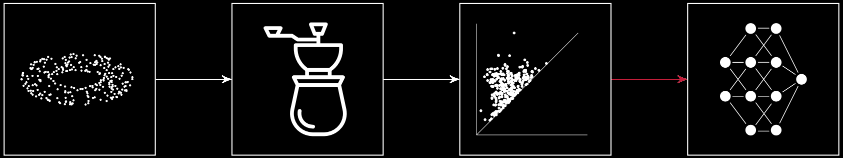


A pipeline for topological machine learning

A pipeline for topological machine learning



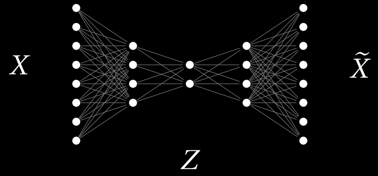
A pipeline for topological machine learning



- ✩ A. Poulenard, P. Skraba and M. Ovsjanikov, 'Topological Function Optimization for Continuous Shape Matching', *Computer Graphics Forum*, 2018.
- ✩ M. Moor*, M. Horn*, **B. Rieck**† and K. Borgwardt†, 'Topological Autoencoders', *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020, arXiv: `1906.00722 [cs.LG]`.
- ✩ M. Carrière, F. Chazal, M. Glisse, Y. Ike, H. Kannan and Y. Umeda, 'Optimizing persistent homology based functions', *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.

A very brief introduction to some machine learning techniques for representation learning

Autoencoders



Terminology

- $X \in \mathbb{R}^D$: input data
- $Z \in \mathbb{R}^d$: latent representation
- $\tilde{X} \in \mathbb{R}^D$: reconstructed data

Properties

- Typically, $D \gg d$.
- Use loss function $\mathcal{L}\left(X, \tilde{X}\right)$ to measure quality of reconstruction.

Why autoencoders?

- Encoder–decoder architecture (we learn the *identity* function).
- 'Middle' layer serves as *bottleneck* or *latent representation*.
- Latent representations can be used for visualisation in lower dimensions, interpolating between data, clustering, and many other tasks.

A simple autoencoder

- Encoder: linear transformation $\mathbb{R}^D \to \mathbb{R}^2$
- Decoder: linear transformation $\mathbb{R}^2 \to \mathbb{R}^D$
- Loss function: mean squared error, $\mathscr{L}\left(X, \tilde{X}\right) := \|X - \tilde{X}\|_2^2$

A simple autoencoder

Some reconstructions

0 epochs

A simple autoencoder

Some reconstructions

10 epochs

A simple autoencoder

Some reconstructions

20 epochs

A simple autoencoder

Some reconstructions

30 epochs

A simple autoencoder

Some reconstructions

40 epochs

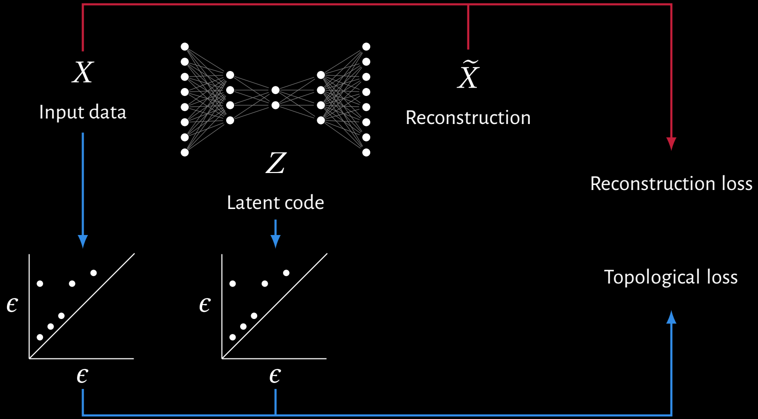
A simple autoencoder

Some reconstructions

50 epochs

Topological autoencoders

Overview



Topological autoencoders

Gradient calculation intuition

Suppose we calculate a Vietoris–Rips complex based on some distances $\mathbf{A}$:

$$\begin{bmatrix} 0 & 1 & 9 & 10 \\ 1 & 0 & 7 & 8 \\ 9 & 7 & 0 & 3 \\ 10 & 8 & 3 & 0 \end{bmatrix}$$

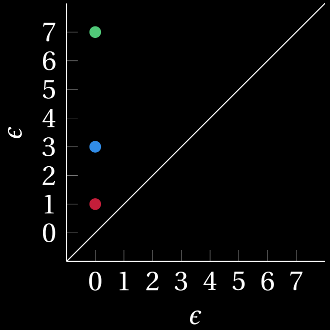
Every point in the persistence diagram can be mapped to *one* distance in the distance matrix! The diagram changes continuously as a function of this matrix.

Topological autoencoders

Gradient calculation intuition

Suppose we calculate a Vietoris–Rips complex based on some distances $\mathbf{A}$:

$$\begin{bmatrix} 0 & 1 & 9 & 10 \\ 1 & 0 & 7 & 8 \\ 9 & 7 & 0 & 3 \\ 10 & 8 & 3 & 0 \end{bmatrix}$$



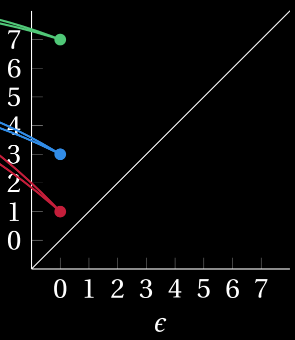
Every point in the persistence diagram can be mapped to *one* distance in the distance matrix! The diagram changes continuously as a function of this matrix.

Topological autoencoders

Gradient calculation intuition

Suppose we calculate a Vietoris–Rips complex based on some distances $\mathbf{A}$:

$$\begin{bmatrix} 0 & 1 & 9 & 10 \\ 1 & 0 & 7 & 8 \\ 9 & 7 & 0 & 3 \\ 10 & 8 & 3 & 0 \end{bmatrix}$$



Every point in the persistence diagram can be mapped to *one* distance in the distance matrix! The diagram changes continuously as a function of this matrix.

Topological autoencoders

Loss term

$$\mathscr{L}_t := \mathscr{L}_{\mathscr{X}\to\mathscr{Z}} + \mathscr{L}_{\mathscr{Z}\to\mathscr{X}}$$

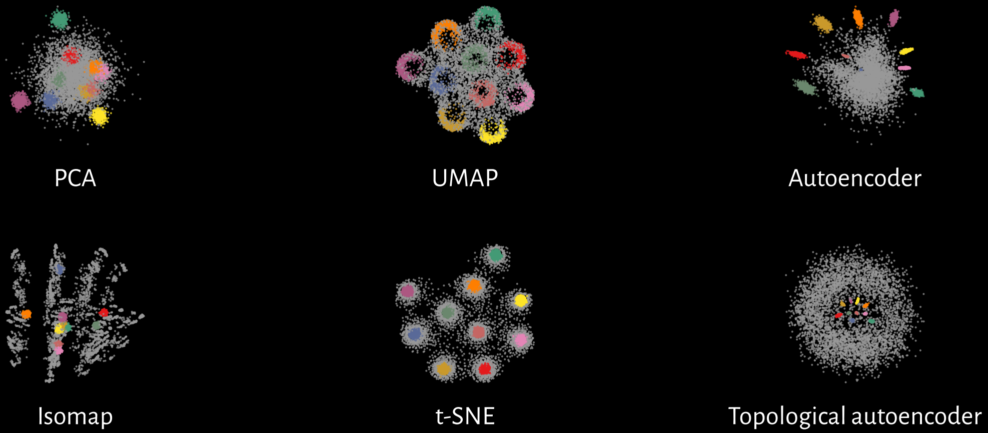
$$\mathscr{L}_{\mathscr{X}\to\mathscr{Z}} := \frac{1}{2}\left\|\mathbf{A}^X\left[\pi^X\right] - \mathbf{A}^Z\left[\pi^X\right]\right\|^2 \qquad \mathscr{L}_{\mathscr{Z}\to\mathscr{X}} := \frac{1}{2}\left\|\mathbf{A}^Z\left[\pi^Z\right] - \mathbf{A}^X\left[\pi^Z\right]\right\|^2$$

- $\mathscr{X}$: input space
- $\mathscr{Z}$: latent space
- $\mathbf{A}^X$: distances in input mini-batch
- $\mathbf{A}^Z$: distances in latent mini-batch
- π^X: persistence pairing of input mini-batch
- π^Z: persistence pairing of latent mini-batch

The loss is *bi-directional*!

Qualitative evaluation

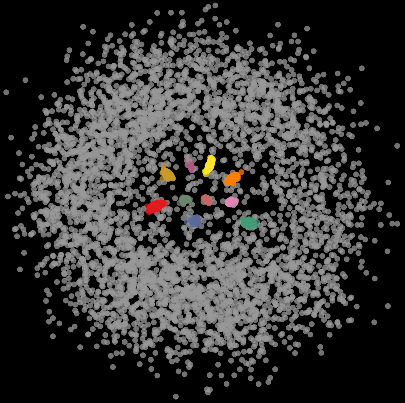
'Spheres' data set



Qualitative evaluation

'Spheres' data set; zooming in...

Autoencoder



Topological autoencoder

A new evaluation metric

Use *distance to a measure* density estimator, i.e.

$$f_\sigma^{\mathscr{X}}(x) := \sum_{y \in \mathscr{X}} \exp\left(-\sigma^{-1} \operatorname{dist}(x, y)^2\right),$$

where dist denotes a metric such as the Euclidean distance. This is well-defined on mini-batches and on the full input data set.

Given σ, we evaluate $\mathrm{KL}_\sigma := \mathrm{KL}\left(f_\sigma^X \parallel f_\sigma^Z\right)$, which measures the similarity between the two density distributions.

Quantitative evaluation

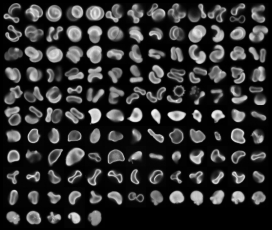
Method	$\mathrm{KL}_{0.01}$	$\mathrm{KL}_{0.1}$	KL_{1}	ℓ-MRRE	ℓ-Cont	ℓ-Trust	ℓ-RMSE	MSE (data)
Isomap	0.181	*0.420*	*0.008 81*	*0.246*	*0.790*	*0.676*	10.4	
PCA	0.332	0.651	0.015 30	0.294	0.747	0.626	11.8	0.9610
t-SNE	*0.152*	0.527	0.012 71	**0.217**	0.773	**0.679**	**8.1**	
UMAP	0.157	0.613	0.016 58	0.250	0.752	0.635	*9.3*	
AE	0.566	0.746	0.016 64	0.349	0.607	0.588	13.3	**0.8155**
TopoAE	**0.085**	**0.326**	**0.006 94**	0.272	**0.822**	0.658	13.5	*0.8681*

Application: Predicting the Shape of Cells

Cell shape prediction

- Use confocal fluorescence microscopy to obtain images of cells.
- What is the 3D shape of a cell?
- Morphological analysis is crucial for certain pathologies!

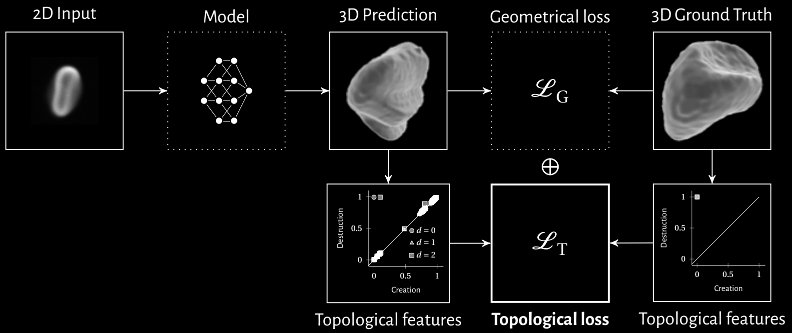
  When used properly, RBC [red blood cell] morphology can be a ***key tool*** *for laboratory hematology professionals to recommend appropriate clinical and laboratory follow-up and to select the best tests for definitive diagnosis.* *(J. Ford, 'Red blood cell morphology', International Journal of Laboratory Hematology, 2013.)*



HELMHOLTZ MUNICH

SHAPR

Overview

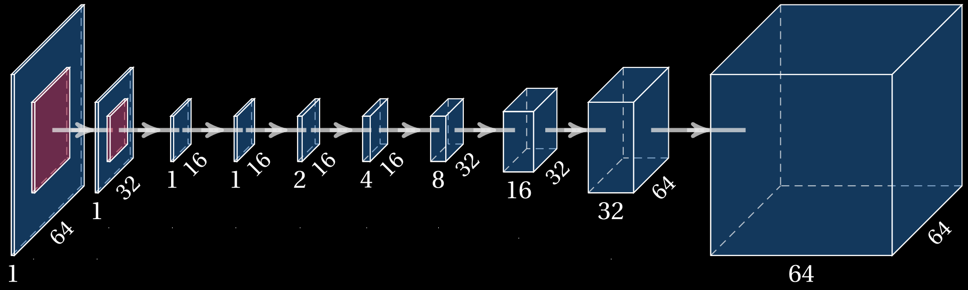


We are trying to solve a complicated *inverse problem*, going from 2D to 3D. This is an ill-defined problem with a large number of potential solutions.

D. J. E. Waibel, S. Atwell, M. Meier, C. Marr and **B. Rieck**, 'Capturing Shape Information with Multi-Scale Topological Loss Terms for 3D Reconstruction', *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2022, arXiv: `2203.01703 [cs.CV]`, in press.

SHAPR

Architecture



We are learning a *likelihood function* $f\colon \mathbb{R}^3 \to \mathbb{R}$. Formally, f 'lives' on a voxel grid, assigning each voxel x a value that indicates the likelihood of x being part of the 'true' volume.

SHAPR

Loss function

$$\mathcal{L}_{\mathrm{G}}(f,f') := \frac{2\,\mathcal{L}_{\mathrm{Dice}}(f,f') + \mathcal{L}_{\mathrm{BCE}(f,f')}}{2}$$

$$\mathcal{L}_{\mathrm{Dice}}(f,f') := \frac{2\left|\mathrm{Vol}_f \cap \mathrm{Vol}_{f'}\right|}{\left|\mathrm{Vol}_f\right| + \left|\mathrm{Vol}_{f'}\right|} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}$$

Intuition

Compare *geometry* of the resulting volumes on a per-voxel basis. Is the reconstructed volume well-aligned with the ground truth one?

SHAPR goes topological

$$\mathscr{L}_{\mathsf{T}}(f, f', q) := \sum_{i=0}^{d} \mathrm{W}_q\left(\mathscr{D}_f^{(i)}, \mathscr{D}_{f'}^{(i)}\right) + \mathrm{pers}\left(\mathscr{D}_{f'}^{(i)}\right)$$

SHAPR goes topological

$$\mathscr{L}_{\mathsf{T}}(f, f', q) := \sum_{i=0}^{d} \mathrm{W}_q\left(\mathscr{D}_f^{(i)}, \mathscr{D}_{f'}^{(i)}\right) + \mathrm{pers}\left(\mathscr{D}_{f'}^{(i)}\right)$$

Loss components

- Aligning the ground truth likelihood f and the predicted likelihood function f'.
- Reducing the geometrical–topological variation of the predicted likelihood function f'.

SHAPR goes topological

$$\mathcal{L}_{\mathsf{T}}(f, f', q) := \sum_{i=0}^{d} \mathrm{W}_q\left(\mathcal{D}_f^{(i)}, \mathcal{D}_{f'}^{(i)}\right) + \mathrm{pers}\left(\mathcal{D}_{f'}^{(i)}\right)$$

Loss components

- Aligning the ground truth likelihood f and the predicted likelihood function f'.
- Reducing the geometrical–topological variation of the predicted likelihood function f'.

We obtain a *combined loss* by choosing $\lambda \in \mathbb{R}_{>0}$ and calculating:

$$\mathcal{L} := \mathcal{L}_{\mathsf{G}} + \lambda \mathcal{L}_{\mathsf{T}}$$

Quantitative results

		Red blood cell ($n = 825$)		Nuclei ($n = 887$)	
Metric	$\mathscr{L}_T$	Median	$\mu \pm \sigma$	Median	$\mu \pm \sigma$
1-IoU	✗	0.48	0.49 ± 0.12	0.62	0.62 ± 0.11
	✓	**0.46**	**0.47 ± 0.10**	**0.61**	**0.61 ± 0.11**
Volume	✗	0.31	0.35 ± 0.31	0.34	0.48 ± 0.47
	✓	**0.21**	**0.25 ± 0.24**	**0.32**	**0.43 ± 0.42**
Surface area	✗	0.20	0.24 ± 0.20	0.21	0.27 ± 0.25
	✓	**0.13**	**0.18 ± 0.16**	**0.18**	**0.25 ± 0.24**
Surface roughness	✗	0.35	0.36 ± 0.24	**0.17**	**0.18 ± 0.12**
	✓	**0.24**	**0.29 ± 0.22**	0.18	0.19 ± 0.13

Summary

- Topology can provide useful inductive biases for shape reconstruction tasks.
- Persistence diagrams encode geometrical *and* topological properties of data.
- Integration into 'standard' machine learning models is possible!

Publications

- F. Hensel, M. Moor and **B. Rieck**, 'A Survey of Topological Machine Learning Methods', *Frontiers in Artificial Intelligence*, 2021.
- M. Moor*, M. Horn*, **B. Rieck**† and K. Borgwardt†, 'Topological Autoencoders', *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020, arXiv: `1906.00722 [cs.LG]`.
- D. J. E. Waibel, S. Atwell, M. Meier, C. Marr and **B. Rieck**, 'Capturing Shape Information with Multi-Scale Topological Loss Terms for 3D Reconstruction', *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, 2022, arXiv: `2203.01703 [cs.CV]`, in press.

Software

`https://github.com/aidos-lab/pytorch-topological`

❤ Acknowledgements

My co-authors, in particular Carsten, Dominik, Felix, Karsten, Max, and Michael.