Witness Autoencoder: Shaping the Latent Space with Witness Complexes

Simon Schönenberger
ETH Zürich
schsimo@ethz.ch

Anastasiia Varava
KTH
varava@kth.se

Vladislav Polianskii
KTH
vpol@kth.se

Jen Jen Chung
ETH Zürich
chungj@ethz.ch

Roland Siegwart
ETH Zürich
rsiegwart@ethz.ch

Danica Kragic
KTH
dani@kth.se

Abstract

We present a *Witness Autoencoder* (W-AE) – an autoencoder that captures geodesic distances of the data in the latent space. Our algorithm uses witness complexes to compute geodesic distance approximations on a mini-batch level, and leverages topological information from the entire dataset while performing batch-wise approximations. This way, our method allows to capture the global structure of the data even with a small batch size, which is beneficial for large-scale real-world data. We show that our method captures the structure of the manifold more accurately than the recently introduced topological autoencoder (TopoAE).

1 Introduction

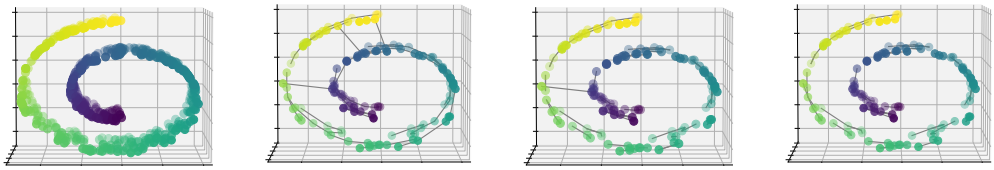
Representation learning aims to identify the underlying structure of data to facilitate the extraction of useful information [1]. Many representation learning methods are built around the *manifold hypothesis*, which states that high-dimensional real world data (e.g. images, text) lie on a low-dimensional *manifold* [1].

Currently, autoencoders (AEs) are widely used for non-linear dimensionality reduction in various applications, mainly due to the expressiveness of neural networks and the encoder-decoder architecture. However, one of the key issues of AEs is that their latent spaces do not necessarily reflect the geometric and topological structure of the true data manifold – i.e., they are not guaranteed to preserve relative distances between points and the topological structure of the data. Preserving this structure is beneficial not only for interpretability of the latent space, but also for generalization capabilities [2, 3] and robustness to adversarial attacks [4].

Most geometric manifold learning methods (e.g. ISOMAP [5], UMAP [6], t-SNE [7]) rely on constructing a neighborhood graph such as k-NN or ϵ-NN[1] to approximate *geodesic distances*, i.e. distances measured along the manifold. The choice of parameters k and ϵ is challenging for two reasons [8]: *(a)* Choosing k, or ϵ, too small results in a disconnected graph, which can lead to disjoint regions in the embedding of regions that are connected on the actual manifold. *(b)* Choosing k, or ϵ, too large leads to capturing erroneous distances, i.e. distances that deviate from the geodesic distances of the manifold, which are commonly referred to as *short-circuit errors* [9] (see fig. 1(c)). For ISOMAP, Balasubramanian and Schwartz [9] showed that already a single short-circuit error could lead to an undesired embedding. These challenges become even more severe if the data is not

[1] k-NN (k nearest neighbors), i.e. we consider the k nearest neighbor for each datum. ϵ-NN considers all neighbors within a ball with radius ϵ around each datum.

Topological Data Analysis and Beyond Workshop at the 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.



(a) Swiss roll dataset (b) VR persistence pairs (c) 1-NN (d) 1-NN-WC (ours)

Figure 1: Different graphs constructed on the Swiss roll dataset (*left*). The graphs constructed from edges of VR 0-order persistence pairings and 1-NN fail to approximate the geodesic distances, i.e. there are *short-circuit errors*. 1-NN-WC approximates the geodesic distances well. ($n_{bs} = 128, |W| = 2048$)

uniformly distributed over the manifold, i.e. if there exist low density regions, which in practice is often the case. UMAP addresses this issue by defining a metric such that the data is approximately uniformly distributed over the manifold [6]. One practical limitation of UMAP is the fact that it cannot be directly used to encode new data without reconstructing the embedding.

Recently introduced *Topological Autoencoders* (TopoAE) [10] and *Markov-Lipschitz deep learning* (MLDL) [11] build upon the idea of aligning (geodesic) distances between input and latent spaces. TopoAE aligns the distances between edges of 0-order persistence pairings, computed from a *Vietoris-Rips* (VR) complex. MLDL aligns distances between k-NN, though additional geometric constraints are discussed in [11] as well. Like most deep learning algorithms, TopoAE and MLDL use mini-batch training, and thus the neighborhood graph is constructed on a mini-batch level. We show that this can lead to short-circuit errors, since low density regions are more likely to occur in small mini-batches.

We present a new way to construct neighborhood graphs from mini-batches, leading to improved approximations of geodesic distances. Furthermore, we present a novel loss term for autoencoders to enforce structure preservation in the latent space, which is closely related to the ones presented in [10, 11]. We make the following theoretical contributions: *(i)* we design a method for the construction of neighborhood graphs based on witness complexes that improve geodesic distance approximations on a mini-batch level; *(ii)* we propose a new autoencoder loss term that encourages alignment of the geodesic distances in both spaces. We demonstrate that, similarly to UMAP [6], our method is able to preserve geodesic distances of the dataset (i.e. unroll the Swiss roll), by using witness complexes. At the same time, our method has the advantage of using a decoder-encoder architecture, which allows it to easily embed new data. Compared to TopoAE, our method approximates the geodesic distances in the data space more accurately (see fig. 1 and fig. 7) and leads to a better distance preservation in latent space (see fig. 2 and appendix A).

2 Proposed method

Preliminaries. We start by introducing the notation. Let $\mathcal{X} \subset \mathbb{R}^D$ be the input space, $\mathcal{Z} \subset \mathbb{R}^d$ be the latent space, $\mathcal{D} = \{x_i\}_{i=1}^n,\ x_i \in \mathcal{X}$ a dataset and $X \subseteq \mathcal{D}$ a mini-batch of size n_{bs}. Further, let $f_\theta : \mathcal{X} \to \mathcal{Z}$ (encoder) and $g_\phi : \mathcal{Z} \to \mathcal{X}$ (decoder) be two non-linear functions parametrized by neural networks, that together represent an autoencoder. Let $\delta(\cdot)$ be a distance measure and $\mathbf{A}^X$ a pair-wise distance matrix with entries $a_{j,i} = a_{i,j} = \delta(x_i, x_j)$. Further let $\pi = \{(i,j)_l\}_{l=1}^m$ be a set of index pairings describing the edges that occur in the graph. Given $\mathbf{A}^X$ and π we will define $\mathbf{A}^X[\pi] \in \mathbb{R}^{|\pi|}$ to be a vector consisting of the edge lengths of graph π. We define $l : \mathbb{R}^n \to \mathbb{R}^n$ as $l(x) = \mathbb{1}_{\{x \geq 0\}} x$.

Topological autoencoder. The main contribution of TopoAE [10] is a topological regularization term $\mathcal{L}_t$, sucht that the total loss term of the autoencoder becomes,

$$\mathcal{L}(x) := \mathcal{L}_r(x, g_\phi(f_\theta(x))) + \lambda \mathcal{L}_t. \tag{1}$$

The topological regularization term $\mathcal{L}_t$ aligns "topologically relevant distances" from both spaces [10]. This is achieved by aligning the distances between edges of 0-order persistence pairings of a VR-filtration from both spaces, i.e. π^X (π^Z) is defined by a simplicial complex containing all edges of

0-order persistence pairings[2] of X ($Z = f_\theta(X)$). $\mathcal{L}_t$ is bidirectional and defined as,

$$\mathcal{L}_t := \underbrace{\frac{1}{2}||\mathbf{A}^X[\pi^X] - \mathbf{A}^Z[\pi^X]||^2}_{\mathcal{L}_{\mathcal{X}\to\mathcal{Z}}} + \underbrace{\frac{1}{2}||\mathbf{A}^Z[\pi^Z] - \mathbf{A}^X[\pi^Z]||^2}_{\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}}. \tag{2}$$

$\mathcal{L}_{\mathcal{X}\to\mathcal{Z}}$ encourages preservation of distances from the input space ($\mathcal{X}$) in the latent space ($\mathcal{Z}$). The role of $\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}$ is less obvious, but crucial for the TopoAE. Recall that we align the distances to preserve topological features, i.e. at convergence it would ideally hold that $\pi^X = \pi^Z$. Intuitively, if we find a pair in Z that does not appear in X, it means that these two points are too close. $\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}$ corrects that by pushing them apart. We will use that insight when we motivate our new loss term.

VR and Witness complexes. Commonly used Vietoris-Rips (VR) complexes approximate a topological space from the entire set of available data. Witness complexes [12] are constructed only from a subset of all available points, but capture the global structure of the data. A small set of *landmark* points (L) is chosen from a dataset, while all the points act as "witnesses" (W) and determine which simplices occur in the witness complex. Formally, a 1-simplex $\sigma = \{u_1, u_2\}$ is added to the witness complex at (filtration) value R iff,

$$\exists w \in W, s.t.\ max(\delta(u_1, w), \delta(u_2, w)) \leq R, \quad u_1, u_2 \in L. \tag{3}$$

In appendix C we provide a more precise definition of both VR and witness complexes.

2.1 Witness autoencoder (W-AE)

Witnessing a neighborhood graph. We leverage witness complexes to improve the batch-wise approximation of geodesic distances in the following way: we set the points in each mini-batch as the landmark points (i.e. $L = X$) and use the entire dataset as witnesses (i.e. $W = \mathcal{D}$). This way, despite performing gradient descent on mini-batches, we can leverage topological information from the entire dataset.

To construct a neighborhood graph for each mini-batch X based on a witness complexes, we use the smallest value R at which an edge (1-simplex) occurs as the pairwise distance,

$$\tilde{a}_{i,j} = \tilde{a}_{j,i} = \underset{x\in\mathcal{D}}{\arg\min}\, max\Big(\delta(u_i, x), \delta(u_j, x)\Big),\ \forall u_i, u_j \in X. \tag{4}$$

In the following we will refer to this method as *k-NN-WC*. In fig. 1(d) such a graph for $k = 1, n_{bs} = 128,\ |W| = 2048$ can be seen. Table 1 presents quantitative results on the observed number of short-circuit errors for 0-order persistence pairings of a VR filtration, k-NN-WC and k-NN, and shows that k-NN-WC can significantly reduce the occurrence of short-circuit errors.

Witness autoencoder. We propose witness autoencoder (W-AE). W-AE constructs k-NN-WC to get the pairings π_k^X. In the latent space it uses k-NN[3] to get the pairings π_k^Z. We define $\mathcal{L}_t$ as,

$$\mathcal{L}_t(k, \nu) := \underbrace{\frac{1}{2}||\mathbf{A}^X[\pi_k^X] - \mathbf{A}^Z[\pi_k^X]||^2}_{\mathcal{L}_{\mathcal{X}\to\mathcal{Z}}} + \underbrace{\frac{1}{2}||l(\nu\mathbf{A}^X[\pi_k^Z - \pi_k^X] - \mathbf{A}^Z[\pi_k^Z - \pi_k^X])||^2}_{\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}}. \tag{5}$$

Our $\mathcal{L}_{\mathcal{X}\to\mathcal{Z}}$ is similar to the one used in TopoAE, with the important difference of using k-NN-WC. For $\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}$ apart from using k-NN instead of the minimal spanning tree, we would like to point out three major differences: *(i)* we only consider pairs that appear in Z but not in X, which we refer to as *wrong pairs*; *(ii)* we introduce a hyper-parameter $\nu \geq 1$. For $\nu > 1$, $\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}$ actively pushes apart wrong pairs, by setting them further apart in $\mathcal{Z}$ than they actually are in $\mathcal{X}$; *(iii)* if for any pair it holds that $\nu\delta(x_i, x_j) \leq \delta(f_\theta(x_i), f_\theta(x_j))$ we do not *pull* the points together (this is because of $l(x) = \mathbb{1}_{\{x\geq 0\}}x$). In our experiments, we observed that this facilitates the convergence of π^Z to π^X. Our $\mathcal{L}_{\mathcal{Z}\to\mathcal{X}}$ is similar to $\mathcal{L}_{push}$ introduced in [11]. However, $\mathcal{L}_{push}$ pushes apart <u>all</u> non-neighbors (w.r.t $\mathcal{X}$) that are at distance below a fixed threshold in $\mathcal{Z}$ (hyperparameter B in [11]). In contrast, we place wrong pairs at a multiple of the actual distance in $\mathcal{X}$, based on the idea that non-nearby points on the manifold have distances greater than linear approximations [13]. Intuitively ν controls how "aggressively" the algorithm pushes wrong pairs apart.

[2]Intuitively, a graph constructed from the edges provided by 0-order persistence pairings is a minimum spanning tree connecting the points of a mini-batch (see appendix C or [10] for a formal explanation).

[3]Approximating geodesic distances in the latent space is less important, since we do manifold learning. Further, the latent space is constantly changing during training, which makes it difficult to apply k-NN-WC.

Table 1: Observed number of mini-batches out of 100 containing short-circuit errors for 0-order persistence pairings of a VR filtration (VR), k-NN and k-NN-WC ($|W| = 2048$).

n_{bs}	Method	Neighbors (k)			
		N/A	1	2	4
64	VR	100	—	—	—
	k-NN	—	77	100	100
	k-NN-WC	—	5	11	72
128	VR	61	—	—	—
	k-NN	—	35	76	100
	k-NN-WC	—	0	2	10
256	VR	6	—	—	—
	k-NN	—	3	11	77
	k-NN-WC	—	0	0	0

Table 2: Quantitative evaluation of latent representation. For each criterion the winner is marked in bold and underlined and the runner-up in bold.

Method	n_{bs}	$MSE_{\mathcal{M},\mathcal{Z}}$	Cont	Trust
W-AE	64	0.029	0.99966	0.99967
	128	0.011	0.99981	0.99998
	256	**0.001**	**0.99995**	**0.99995**
TopoAE	64	0.060	0.99973	0.99969
	128	0.061	0.99977	0.99970
	256	**0.010**	**0.99992**	**0.99992**
UMAP	-	0.024	0.99928	0.99698
t-SNE	-	0.027	0.99855	0.99837

3 Experimental study on Swiss roll dataset

In the following we provide an overview of our experimental study on W-AE. In appendix A an exhaustive overview of the results of this experimental study, definitions for all metrics used and a description of our model selection can be found. All experiments were performed on the Swiss roll dataset. Furthermore, we present in appendix B another example for k-NN-WC.

Architecture & training. We used an AE with two hidden layers for f_θ and g_ϕ consisting of 32 ReLu units each. Further, we normalized $\mathbf{A}^X$ and $\mathbf{A}^Z$. For optimization we used Adam [14], learning rates $\in [0.001, 0.1]$, $n_{bs} \in [64, 512]$, $\nu \in [1, 1.25]$, $k \in [1, 8]$ and trained for 1000 epochs with early stopping. We fixed the mini-batches over all epochs[4].

Qualitative evaluation. The latent representation of the Swiss roll dataset constructed by TopoAE and W-AE for different mini-batch sizes can be seen in fig. 2. Compared to TopoAE our method succeeds at unrolling the Swiss roll for smaller mini-batches and achieves an embedding quality comparable to UMAP and t-SNE.

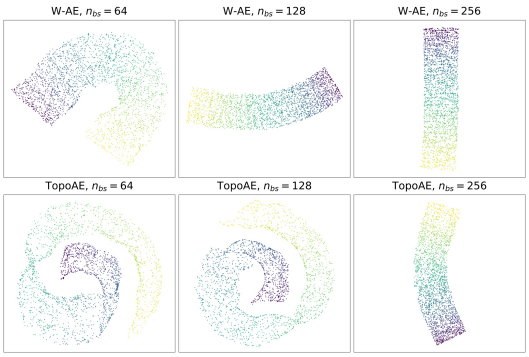


Figure 2: Latent representation obtained with TopoAE and W-AE of Swiss roll dataset for different mini-batch sizes.

Quantitative evaluation. Results for TopoAE, W-AE, UMAP and t-SNE can be seen in table 2[5]. W-AE outperforms its competitors w.r.t. *Trust* and *Cont* [15], except for $n_{bs} = 64$. $MSE_{\mathcal{M},\mathcal{Z}}$ measures the MSE between the *true* distance matrix of the manifold and the one computed from the resulting embedding[6]. W-AE achieves comparable results for $n_{bs} = 128$ to TopoAE for $n_{bs} = 256$. W-AE outperforms TopoAE for $n_{bs} = 256$, which is likely due to the new loss formulation.

4 Discussion

To summarize, we make two contributions: *(i)* with k-NN-WC we present a general method that is applicable to deep learning methods that rely on distance preservation (e.g. MLDL), and *(ii)* we provide a new loss term that facilitates convergence of π^Z to π^X.

In particular, providing *(i)* is essential to make deep learning methods that rely on distance preservation scalable to high-dimensional real-world data, since the underlying manifold is likely too complex to be approximated by k-NN on a mini-batch level for reasonable mini-batch sizes.

[4]This is an important detail, since reshuffling increases the likelihood that short-circuit errors appear.

[5]TopoAE and W-AE were evaluated on a test split, t-SNE and UMAP directly on the training data.

[6]We are able to compute this because we know the actual manifold, i.e. we can sample from it.

References

[1] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation Learning : A Review and New Perspectives. 35(8):1798–1828, 2013.

[2] Cem Anil, James Lucas, and Roger Grosse. Sorting out Lipschitz function approximation. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:432–452, 2019.

[3] Peter L. Bartlett, Dylan J. Foster, and Matus Telgarsky. Spectrally-normalized margin bounds for neural networks Peter. *Revista Militar*, (Nips), 2017.

[4] Yusuke Tsuzuku, Issei Sato, and Masashi Sugiyama. Lipschitz-margin training: Scalable certification of perturbation invariance for deep neural networks. *Advances in Neural Information Processing Systems*, 2018-Decem(NeurIPS):6541–6550, 2018. ISSN 10495258.

[5] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. ISSN 00368075. doi: 10.1126/science.290.5500.2319.

[6] Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. 2018. URL http://arxiv.org/abs/1802.03426.

[7] Geoffrey Hinton Laurens van der Maaten. Visualizing Data using t-SNE. *Annals of Operations Research*, 219(1):187–202, 2008. ISSN 15729338. doi: 10.1007/s10479-011-0841-3.

[8] Xianhua Zeng. Applications of Average Geodesic Distance in Manifold Learning. In Guoyin Wang, Tianrui Li, Jerzy W Grzymala-Busse, Duoqian Miao, Andrzej Skowron, and Yiyu Yao, editors, *Rough Sets and Knowledge Technology*, pages 540–547, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-79721-0.

[9] Mukund Balasubramanian and Eric L. Schwartz. The isomap algorithm and topological stability. *Science*, 295(5552):7, 2002. ISSN 00368075. doi: 10.1126/science.295.5552.7a.

[10] Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological Autoencoders. pages 1–18, 2019. URL http://arxiv.org/abs/1906.00722.

[11] Stan Z. Li, Zelin Zhang, and Lirong Wu. Markov-Lipschitz Deep Learning. 2020. URL http://arxiv.org/abs/2006.08256.

[12] Vin De Silva and Gunnar Carlsson. Topological estimation using witness complexes. (November 2014), 2004. doi: 10.2312/SPBG/SPBG04/157-166.

[13] Martin Jørgensen and Søren Hauberg. Isometric Gaussian Process Latent Variable Model for Dissimilarity Data. pages 1–10, 2020. URL http://arxiv.org/abs/2006.11741.

[14] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pages 1–15, 2015.

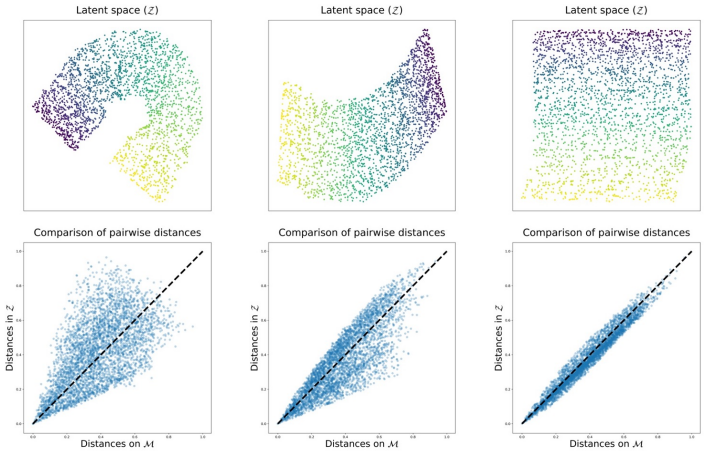
[15] Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC Bioinformatics*, 4:1–13, 2003. ISSN 14712105. doi: 10.1186/1471-2105-4-48.

[16] Denis Blackmore and Thomas J. Peters. Computational Topology. *Open Problems in Topology II*, pages 493–545, 2007. doi: 10.1016/B978-044452208-5/50049-1.

A Supplementaries on experimental study

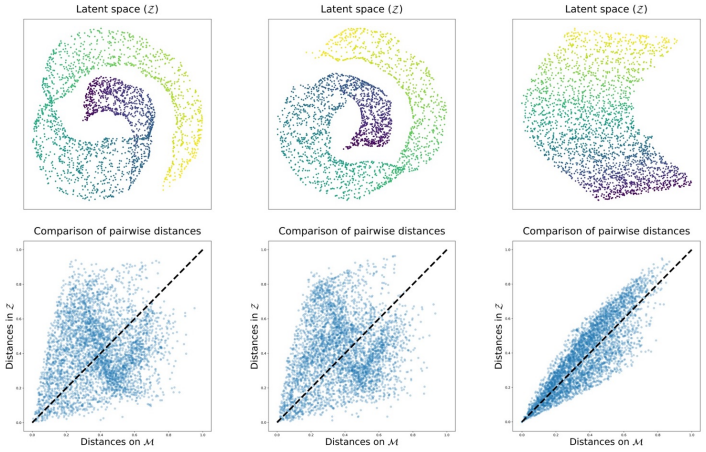
A.1 Pairwise distances between latent space and true manifold

In the following we present latent spaces of the Swiss roll dataset of all models compared in the study, and compare the pairwise distances between the latent space ($\mathcal{Z}$) and the true manifold ($\mathcal{M}$), which in the ideal case lie on the $45\,\mathrm{deg}$ line (i.e. $\delta_{\mathcal{Z}}(i,j) = \delta_{\mathcal{M}}(i,j)$). Note that local distance are reflected by the lower left corner, while global distances are in the top right corner. In general it is more difficult to approximate the global distances correctly. W-AE performs better on the whole range of distances. Quantitatively these results are reflected by $MSE_{\mathcal{M},\mathcal{Z}}$ and $\hat{\sigma}_k^{iso}$ (see table 3). The axis are normalized in the following plots.



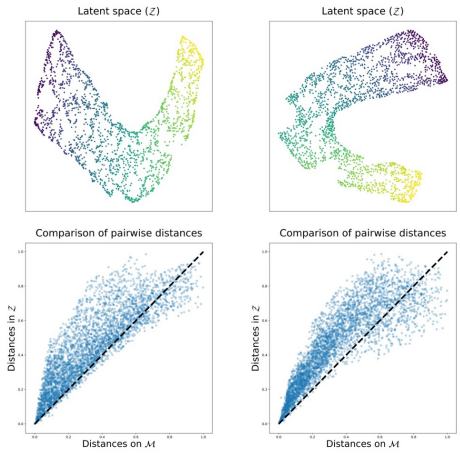
(a) W-AE, $n_{bs} = 64$ (b) W-AE, $n_{bs} = 128$ (c) W-AE, $n_{bs} = 256$

Figure 3: Latent space (top row) and pairwise distance comparison between true manifold ($\mathcal{M}$) and latent space ($\mathcal{Z}$) constructed from W-AE. The global distances on the manifold get approximated more accurately for larger batch sizes. Yet compared to TopoAE the approximation is already better for smaller batch sizes. (see fig. 4)



(a) TopoAE, $n_{bs} = 64$ (b) TopoAE, $n_{bs} = 128$ (c) TopoAE, $n_{bs} = 256$

Figure 4: Latent space (top row) and pairwise distance comparison between true manifold ($\mathcal{M}$) and latent space ($\mathcal{Z}$) constructed from TopoAE. The global distances on the manifold get approximated more accurately for larger batch sizes.



(a) UMAP (b) tSNE

Figure 5: Latent space (top row) and pairwise distance comparison between true manifold ($\mathcal{M}$) and latent space ($\mathcal{Z}$) constructed tSNE and UMAP.

Table 3: Complete overview of the quantitative evaluation of the experimental study. For each criterion the winner is marked in bold and underlined and the runner-up in bold

Method	n_{bs}	$\mathcal{L}_{rec}$	$MSE_{\mathcal{M},\mathcal{Z}}$	Trust	Cont	$\hat{\sigma}_{45}^{iso}$	$KL_{0.1}$	$KL_{0.01}$
	64	0.274	0.029	0.99966	0.99967	0.20	0.044	0.023
W-AE	128	0.721	0.011	0.99982	0.99981	0.15	0.044	0.018
	256	**0.144**	**0.001**	**0.99995**	**0.99995**	**0.07**	0.046	**0.011**
	64	10.459	0.06	0.99973	0.99969	0.21	**0.004**	0.017
TopoAE	128	10.945	0.061	0.99977	0.99970	0.19	**0.002**	**0.014**
	256	**0.168**	**0.01**	**0.99992**	**0.99992**	**0.09**	0.042	0.022
UMAP	—	—	0.024	0.99928	0.99698	0.74	0.066	0.056
t-SNE	—	—	0.027	0.99855	0.99837	0.33	0.050	0.044

A.2 Quantitative evaluation

In table 3 an exhaustive overview over the quantitative results can be seen. The definitions for all metrics can be found in appendix A.4. We want to point out the following remarks:

- For the Swiss roll dataset, optimizing $\mathcal{L}_t$ is not at odds with optimizing $\mathcal{L}_r$. More experimental results are needed to verify if and to what extent that applies to other datasets.
- $MSE_{\mathcal{M},\mathcal{Z}}$ and $\hat{\sigma}_{45}^{iso}$ reflect the improved latent embedding that can be seen qualitatively most accurately. Our method preserves global distances on the manifold more accurately, as well as local, pairwise distances ($\hat{\sigma}_{k}^{iso}$).
- Overall our method outperforms its competitors (except for $n_{bs} = 64$) w.r.t. continuity and trustworthiness. We would like to point here that the choice of k (for continuity and trustworthiness) has a very strong effect on the outcome of the evaluation. Chosen too small, the error can be underestimated, chosen too large, non-neighbor points are included in the evaluation (i.e. short-circuit errors).
- We included KL_κ as presented in [10]. KL_κ heavily depends on κ. In our work we observed that choosing κ to be difficult, hence we doubt if it is a good measure to assess the embedding quality.

A.3 Training details

The presented results are obtained from a grid search. W-AE was trained for learning rates $\in [0.001, 0.1]$, $\lambda \in [512, 8192]$, $n_{bs} \in [64, 512]$, $\nu \in [1, 1.25]$, $k \in [1, 8]$. For selection we chose the 10 best models according to matched edges in data and latent space, from which we selected the one with the highest trustworthiness score. TopoAE was trained for learning rates $\in [0.001, 0.1]$, $\lambda \in [512, 8192]$, $n_{bs} \in [64, 512]$ and the best model selected according to the trustworthiness score. For a fair comparison we also compared TopoAE results for other metrics, however TopoAE never succeeded to unroll the Swiss roll for $n_{bs} \in \{64, 128\}$. UMAP was trained for $k \in [2, 40]$, $min_{dist} \in [0.05, 0.5]$, t-SNE for $perplexity \in [10, 100]$, and the best models selected according to the trustworthiness score. We ran all models with 10 different random initializations (affects the model initialization and sampling of the data).

A.4 Evaluation metrics

For quantitative evaluation of our work we used rank-based and distance-based criteria. We will first define the ones used in section 3 ($MSE_{\mathcal{M},\mathcal{Z}}$, *Trust*, *Cont*), and introduce additional ones ($\hat{\sigma}_k^{iso}$,KL_κ) thereafter.

$\mathbf{MSE}_{\mathcal{M},\mathcal{Z}}$ is defined as the mean squared error between the pairwise distance matrices from the actual manifold and latent space. We can measure this, because we work with toy datasets, i.e. we can sample directly from the manifold, transform the data into the dataspace and apply a manifold learning method. It is defined as,

$$MSE_{\mathcal{M},\mathcal{Z}} = \mathrm{Tr}((\mathbf{A}^{\mathcal{M}} - \mathbf{A}^{Z})^T(\mathbf{A}^{\mathcal{M}} - \mathbf{A}^{Z})). \tag{6}$$

Trustworthiness and continuity [15] measure how well the k-NN of a point are preserved when changing between spaces. Trustworthiness (Trust) captures that when going from data space to latent space and continuity (Cont) when going from latent space to data space:

$$Trust(k) := 1 - \frac{2}{Nk(2N - 3k - 1)} \sum_{i=1}^{N} \sum_{\substack{j \in \mathcal{N}_k(x_i) \\ j \notin \mathcal{N}_k(z_i)}} (rank(\mathbf{Z}, i, j) - k) \tag{7}$$

$$Cont(k) := 1 - \frac{2}{Nk(2N - 3k - 1)} \sum_{i=1}^{N} \sum_{\substack{j \in \mathcal{N}_k(z_i) \\ j \notin \mathcal{N}_k(x_i)}} (rank(\mathbf{X}, i, j) - k) \tag{8}$$

$$\tag{9}$$

As is common for rank based measures, we need to choose a k. Therefore we computed Cont and Trust for $k \in \{15, 30, 45\}$ and averaged.

Local isometry is measured by $\hat{\sigma}_k^{iso}$, which we define as the standard deviation of the set of length ratios,

$$l_{i,j}^X = \frac{\delta(f_\theta(x_i), f_\theta(x_j))}{\delta(x_i, x_j)}, \quad \forall x_j \in \mathcal{N}_k(x_i), x_i \in X \tag{10}$$

For comparability we normalized the ratios, i.e. scale the mean to 1, thus if all $l_{i,j}^X \approx 1$, f_θ is locally isometric.

Local KL divergence measures the *Kullback-Leiber divergence* between density distributions in the data and latent space as introduced in [10]. The density estimate is defined as,

$$f_\kappa^{\mathcal{X}}(x) := \sum_{y \in X} exp\Big(-\kappa^{-1}\delta(x, y)^2 \Big), \ \kappa \in \mathbb{R}^+. \tag{11}$$

δ is the Euclidian distance normalized for each point. κ is a length scale parameter [10], i.e. large κ captures more of the global structure, while a small κ captures more of the local structure. The local KL divergence is then defined as,

$$KL_\kappa := D_{KL}(f_\kappa^{\mathcal{X}} || f_\kappa^{\mathcal{Z}}), \tag{12}$$

where $D_{KL}(\cdot||\cdot)$ denotes the KL divergence itself.

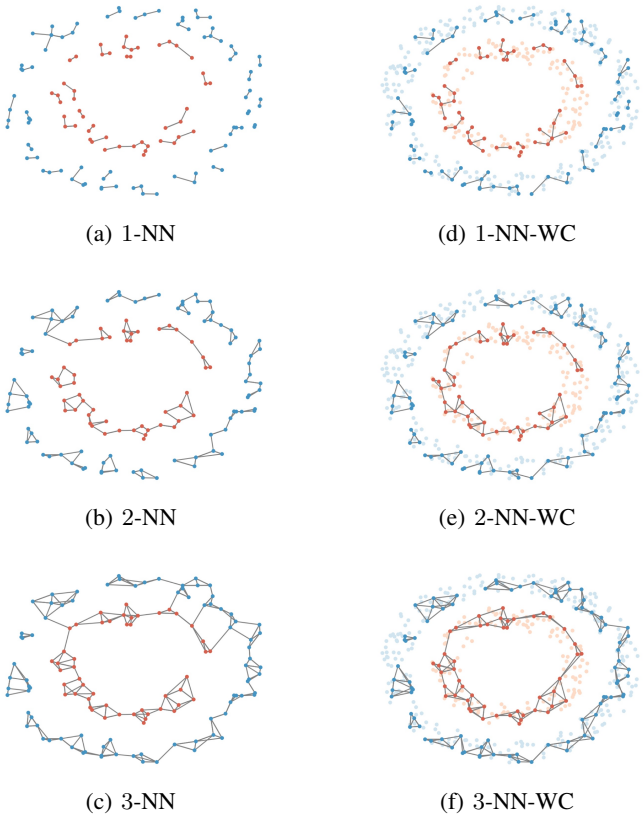
B Toy dataset: two concentric annuli

To illustrate the strength of k-NN-WC we show neighborhood graphs on the toy dataset *concentric circles* in the following. Figure 6(a) show the dataset and fig. 6(b) shows a subsample of it which represents a mini-batch with $n_i = 48$ (inner circle) and $n_o = 82$ (outer circle) [7].

(a) 1-NN (b) 1-NN-WC

Figure 6: Concentric circle toy dataset (fig. 6(a)) and a subsample of it (fig. 6(b)).

Figure 7 shows neighborhood graphs for $k \in \{1, 2, 3\}$ constructed with k-NN and k-NN-WC. As for the Swiss roll, short-circuit errors can be observed, i.e. for $k > 1$, k-NN starts to approximate non-geodesic distances (see fig. 7(b)), while k-NN-WC approximates the geodesic distances well also for higher k.



(a) 1-NN (d) 1-NN-WC

(b) 2-NN (e) 2-NN-WC

(c) 3-NN (f) 3-NN-WC

Figure 7: Neighborhood graphs constructed with k-NN and k-NN-WC for $k \in \{1, 2, 3\}$ from the mini-batch which can be seen in fig. 6(b) ($n_o = 82,\ n_i = 48$). For the witnesses we sampled a new set of points ($n_o^w = 244,\ n_i^w = 140$), that is represented by the lucent points.

[7] The ratio n_o/n_i corresponds to the ratio of the area of the two annuli, s.t. that they have the same sampling density.

C Background: Topological data analysis

In this section we will cover some basic concepts from topological data analysis (TDA). It is by no means a complete introduction. For that the reader is referred to the excellent book by Blackmore and Peters [16] on computational topology, from where we took most of the definitions that follow.

C.1 Simplicial complex

Let us start by introducing the concept of *k-simplicies* and *faces*,

Definition C.1 (k-simplex σ [16]) *Let $u_0, u_1, ..., u_k$ be points in $\mathbb{R}^d$. A k-simplex σ is the convex hull of $k+1$ affinely independent points, i.e. $\sigma = conv\{u_0, u_1, ..., u_k\}$. Its dimension is defined as $dim(\sigma) = k$.*

Definition C.2 (Face of σ [16]) *Let σ be a k-simplex, then we define τ as a face of σ if it is a non-empty subset of σ. Furthermore we say it is proper if the subset is not the entire set.*

If we take a collection of such k-simplicies, i.e. a set of k-simplicies, it is – under certain prerequisites – a simplicial complex,

Definition C.3 (Simplicial complex K [16]) *We define a simplicial complex K as a collection of simplicies, that satisfies the following conditions:*

1. *Every face of every simplex in K belongs to K, i.e. $\tau \in K,\ \forall \tau \subseteq \sigma,\ \forall \sigma \in K$*
2. *The intersetion of any two simplices in K is either empty or a face of both.*

One of the simplest and most common ways to construct simplicial complexes is the Vietoris-Rips (VR) complex,

Definition C.4 (Vietoris-Rips complex [16]) *Let S be a finite set of points in $\mathbb{R}^d$ and $B_x(r) = x + r\mathbb{B}^d$ be the closed ball with center $x \in \mathbb{R}^d$ and radius $r \in \mathbb{R}$. Then the Vietoris-Rips complex is defined as:*

$$VR(r) = \left\{ \sigma \subseteq S \mid \exists u, u' \in \sigma \ s.t.\ B_u(r) \cap B_{u'}(r) \neq \emptyset \right\} \tag{13}$$

Vietoris-Rips complexes construct a simplicial complex from the entire dataset. For large datasets this gets computationally infeasible, yet the topology of a space can normally be captured by a smaller subset already. Witness complexes that we use in our work, build around that idea. They construct a simplicial complex only from a subset of vertices available, and use the remaining points to determine when a simplex occurs in the filtration. Formally we can describe the nested family of witness complexes as,

Definition C.5 (Nested family of witness complexes [12]) *Let $\langle \mathcal{X}, \delta \rangle$ be a metric space, $X \subset \mathcal{X}$ be a dataset, $L = \{l_0, ..., l_n\} \subseteq X$ be a set of landmark points and $R \in \mathbb{R}^+$. Then the k-simplex $\sigma = \{u_1, ..., u_k\}$ with $u_i \in L$ belongs to $W(X, L; R)$ iff all its faces belong to $W(X, L; R)$ and there is a witness $x \in X$, such that:*

$$max(\delta(u_i, x) : u_i \in \{u_1, ..., u_k\}) \leq R \tag{14}$$

C.2 Persistent homology

To give a precise definition for persistent homology groups goes beyond the scope of this appendix. Therefore we will first define filtrations, then give an intuitive explanation of homology groups and finally define persistence pairings, which are needed in the context of TopoAE and W-AE.

The construction of most simplicial complexes depends on a hyperparameter, i.e. a scale at which we wish to construct them. For the Vietoris-Rips complex from definition C.4 this parameter is represented by r and for the witness complex from definition C.5 by R. A priori it is impossible to choose that parameter in a suitable way. Therefore it is common to analyze the growing family of simplicial complexes, which are defined by a filtration,

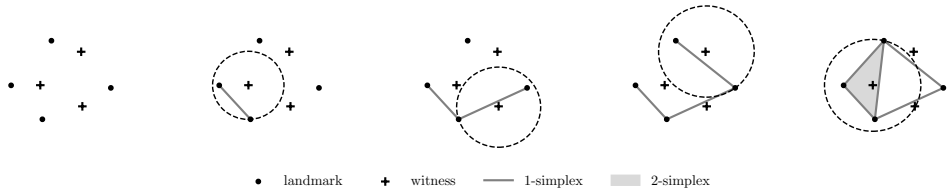


Figure 8: Witness complex filtration: Whenever a simplex gets witnessed by a witness, it is added to the simplicial complex. the circle indicates how far a witness can "see", i.e. the filtration radius R. We only marked the circle around the witness that actually witnesses the simplex that got added at the corresponding step.

Definition C.6 (Filtration [16]) *Let K be a simplicial complex, then we call the sequence of (growing) subcomplexes of K a filtration,*

$$\emptyset = K_0 \subseteq K_1 \subseteq ... \subseteq K_{n-1} \subseteq K_n \subseteq K \tag{15}$$

Given such a filtration, persistence homology studies the creation and destruction of homology classes. Homology classes can be said to describe topological features, i.e. the 0-homology class describes connected components, the 1-homology class describes tunnels, and the 2-homology classes describe voids. Each such class gets created at a certain point in the filtration and destroyed at a later point, this is normally referred to as birth and death in TDA. Since a simplex is involved in the birth and death of every homology class, we can create persistence pairings,

Definition C.7 (Persistence pairings [16]) *Let the filtration be $K_0 \subset K_1 \subset \cdot \subset K_n$ such that $K_0 = \emptyset$ and $K_{i+1} \setminus K_i = \sigma_i$, i.e. we add at every step $i \in [1, n]$ in the filtration exactly one simplex σ_i. Further let the homology class γ be created at step i, i.e. when adding σ_i, and be destroyed at step j, i.e. when adding σ_j. Then we call the pairing (i, j) the persistence pairing, because σ_i created γ and σ_j destroyed γ.*