HAL
open science

Representation Topology Divergence: A Method for Comparing Neural Network Representations

Serguei Barannikov, Ilya Trofimov, Nikita Balabin, Evgeny Burnaev

To cite this version:

Serguei Barannikov, Ilya Trofimov, Nikita Balabin, Evgeny Burnaev. Representation Topology Divergence: A Method for Comparing Neural Network Representations. Proceedings of Machine Learning Research, 2022, Proceedings of the 39th International Conference on Machine Learning (ICML 2022), 162, pp.1607-1626. hal-03821864

HAL Id: hal-03821864

https://hal.science/hal-03821864v1

Submitted on 20 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Representation Topology Divergence: a Method for Comparing Neural Network Representations

Serguei Barannikov [1 2] **Ilya Trofimov** [1] **Nikita Balabin** [1] **Evgeny Burnaev** [1 3]

Abstract

Comparison of data representations is a complex multi-aspect problem that has no complete solution yet. We propose a method for comparing two data representations. We introduce the Representation Topology Divergence (RTD) which measures the dissimilarity in multi-scale topology between two point clouds of equal size with a one-to-one correspondence between points. The data point clouds are allowed to lie in different ambient spaces. The RTD is one of the few practical methods based on Topological Data Analysis (TDA) applicable to real machine learning datasets. Experiments show the proposed RTD agrees with the intuitive assessment of data representation similarity and is sensitive to its topological structure. We apply RTD to gain insights into neural network representations in computer vision and NLP domains for various problems: training dynamics analysis, data distribution shift, transfer learning, ensemble learning.

1. Introduction

Representations of objects are the essential component learnt by deep neural networks. In opposite to the distance in the original space, the similarity of representations is proved to be semantically meaningful. Despite the significant practical success of deep neural networks, many aspects of their behavior are poorly understood. Only a few methods study neural representations without relying on their quality on a specific downstream task. In this work, we focus on the comparison of representations from neural networks.

Comparison of representations is an ill-posed problem without a "ground truth" answer. Early studies were based on variants of Canonical Correlation Analysis (CCA): SVCCA, (Raghu et al., 2017), PWCCA (Morcos et al., 2018). However, CCA-like measures define similarity too loosely since they are invariant under any invertible linear transformation. The Centered Kernel Alignment (CKA), (Kornblith et al., 2019) is the statistical test to measure the independence of two sets of variables. (Kornblith et al., 2019) proved CKA to be more consistent with the intuitive similarity of representations. Particularly, neural networks learn similar representations from different seeds as evaluated by CKA. Another line of work is concerned with the alignment between groups of neurons (Li et al., 2015), (Wang et al., 2018). The similarity of representations is also a topic of study in neuroscience (Edelman, 1998; Kriegeskorte et al., 2008; Connolly et al., 2012).

Representational similarity metrics like CKA and CCA were used to gain insights on representations obtained in meta-learning (Raghu et al., 2020), to compare representations from different layers of language models (Voita et al., 2019), and to study the effect of fine-tuning (Wu et al., 2020). Finally, (Nguyen et al., 2021) used CKA to study the phenomenon of a "block structure" emerging in wide and deep networks in computer vision and compare their representations.

In this paper, we take a topological perspective on the comparison of neural network representations. We propose the *Representation Topology Divergence (RTD)* score, which measures dissimilarity between two point clouds of equal size with a one-to-one correspondence between points. Point clouds are allowed to lie in different ambient spaces. Existing geometrical and topological methods are dedicated to other problems: they are either too general and do not incorporate the one-to-one correspondence requirement (Khrulkov & Oseledets, 2018), (Tsitsulin et al., 2020), or they restrict point clouds to lie in the same ambient space (Kynkäänniemi et al., 2019), (Barannikov et al., 2021). Most of these methods are applied to the evaluation of GANs. Recently, (Moor et al., 2020) proposed a loss term to compare the topology of data in original and latent spaces and applied the term as a part of the Topological Autoencoder.

In this work, we make the following contributions:

[1]Skolkovo Institute of Science and Technology, Moscow, Russia [2]CNRS, Université Paris Cité, France [3]Artificial Intelligence Research Institute (AIRI), Moscow, Russia Correspondence to: Serguei Barannikov <S.Barannikov@skoltech.ru>.

Proceedings of the 39th International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

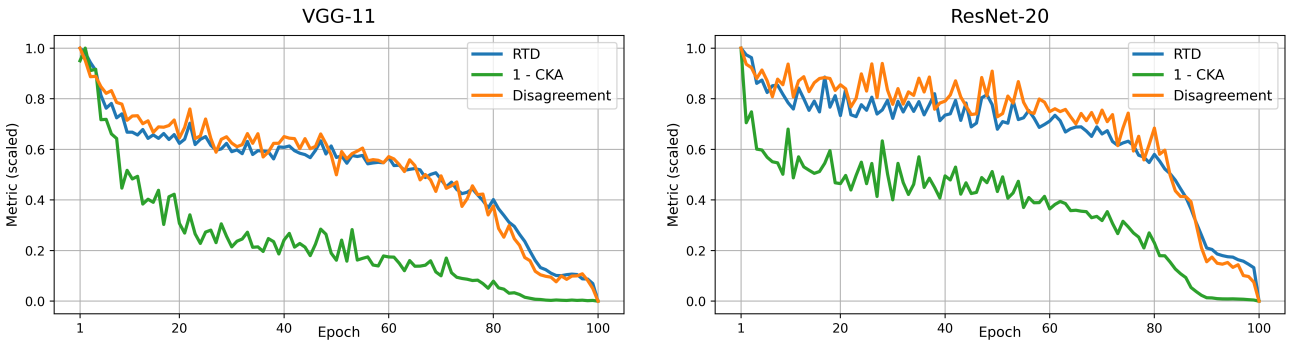


Figure 1: Comparison of representations after the ith epoch and the final one done by RTD, $1-$CKA, and disagreement of predictions. All the measures are normalized by division to their maximal values. Strikingly, RTD highly correlates with the disagreement of models' predictions.

1. We propose a topologically-inspired approach for comparison of neural network representations;
2. We introduce the *R-Cross-Barcode*$(P, \tilde{P})$, a tool based on Topological Data Analysis (TDA), which measures the differences in the multi-scale topology of two point clouds $P, \tilde{P}$ with a one-to-one correspondence between points;
3. Based on the *R-Cross-Barcode*$(P, \tilde{P})$, we define the *Representation Topology Divergence (RTD)*, the quantity measuring the multi-scale topological dissimilarity between two representations;
4. Our computational experiments show that RTD agrees with an intuitive notion of neural network representations similarity. In contrast to most existing approaches, RTD is sensitive to differences in topological structures (clusters, voids, cavities, tunnels, etc.) of the representations and enjoys a very good correlation with disagreement of models predictions. We apply RTD to compare representations in computer vision and NLP domains and various problems: training dynamics analysis, data distribution shift, transfer learning, ensemble learning, and disentanglement. Experiments show that RTD outperforms CKA, IMD, and SVCCA.

The source code is publicly available:
`https://github.com/IlyaTrofimov/RTD`.

2. Comparing Neural Network Representations

Our starting point is the geometric perspective on representation learning through the lens of the manifold hypothesis (Goodfellow et al., 2016), according to which real-world data presented in a high-dimensional space are expected to concentrate in the vicinity of a manifold of much lower dimension. The low-dimensional manifold $M_{\mathcal{P}}$ underlying the given data representation $\mathcal{P}$ can be accessed in general only through discrete sets of samples. The standard approach to recover the manifold $M_{\mathcal{P}}$ is to take a sample P and to approximate $M_{\mathcal{P}}$ by a set of simplexes with vertices from P. Commonly, to select the simplexes approximating $M_{\mathcal{P}}$ one has to fix a threshold $\alpha > 0$ and consider the simplexes with edge lengths not exceeding α (Niyogi et al., 2008; Belkin & Niyogi, 2001). It is difficult to guess the correct value of the threshold, and hence a reasonable approach is to study all thresholds at once.

Given two representations, we consider two corresponding graphs with distance-like weights and compare the difference in the multiscale topology of the two graphs.

Let $\mathcal{P}, \tilde{\mathcal{P}}$ be two representations giving two embeddings of the same data $\mathcal{V}$. The two embeddings $\mathcal{P}, \tilde{\mathcal{P}}$ belong in general to different ambient spaces and have the natural one-to-one correspondence between points in $\mathcal{P}$ and $\tilde{\mathcal{P}}$. Given a sample of data $V \subseteq \mathcal{V}$, the two representations $P = \mathcal{P}(V)$, $\tilde{P} = \tilde{\mathcal{P}}(V)$ define two weighted graphs $\mathcal{G}^{w}$, $\mathcal{G}^{\tilde{w}}$ with the same vertex set V. The weights w_{AB}, $\tilde{w}_{AB}$ of an edge AB are given by the distances $w_{AB} = \mathrm{dist}(P(A), P(B))$, $\tilde{w}_{AB} = \mathrm{dist}(\tilde{P}(A), \tilde{P}(B))$.

The simplicial approximation to the manifold $M_{\mathcal{P}}$ at threshold α consists of simplexes whose edges in $\mathcal{G}^{w}$ have weights not exceeding α. Let $\mathcal{G}^{w \leq \alpha}$ denote the graph with the vertex set V and the edges with weights not exceeding α. To compare the simplicial approximations to the manifolds $M_{\mathcal{P}}$ and $M_{\tilde{\mathcal{P}}}$ described by the graphs $\mathcal{G}^{w \leq \alpha}$ and $\mathcal{G}^{\tilde{w} \leq \alpha}$, we compare each of the two simplicial approximations with the union of simplices formed by edges present in at least one of the two graphs. The graph $\mathcal{G}^{\min(w, \tilde{w}) \leq \alpha}$ contains an edge between vertices A and B iff the distance between the points A and B is smaller than α in at least one of the representations P, $\tilde{P}$. The set of edges of the graph $\mathcal{G}^{\min(w, \tilde{w}) \leq \alpha}$ is the union of sets of edges of $\mathcal{G}^{w \leq \alpha}$ and $\mathcal{G}^{\tilde{w} \leq \alpha}$. The similarity of manifolds $M_{\mathcal{P}}$ and $M_{\tilde{\mathcal{P}}}$ can be measured by the degrees of

similarities of the graph $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ with the graph $\mathcal{G}^{w\leq\alpha}$ and the graph $\mathcal{G}^{\tilde{w}\leq\alpha}$.

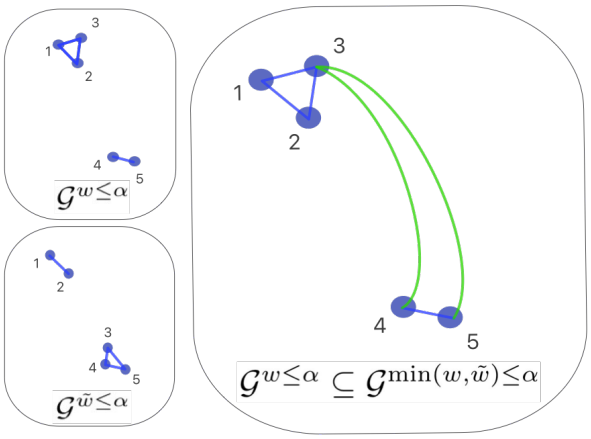


Figure 2: Graphs $\mathcal{G}^{w\leq\alpha}$, $\mathcal{G}^{\tilde{w}\leq\alpha}$ and $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ with edges not in $\mathcal{G}^{w\leq\alpha}$ colored in green.

2.1. Topological features for a pair of weighted graphs

One way to measure the discrepancy between the graphs $\mathcal{G}^{w\leq\alpha}$ and $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ is to count the graph $\mathcal{G}^{w\leq\alpha}$ connected components merged together in the graph $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$. We show an example of this situation in Figure 2 right, see also Figure 10, where three graphs $\mathcal{G}^{w\leq\alpha}$, $\mathcal{G}^{\tilde{w}\leq\alpha}$ and $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ are shown, with edges of the graph $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ not in $\mathcal{G}^{w\leq\alpha}$ colored in green. Each merging is represented by a class of green paths in $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ joining two blue clusters. The significance of the discrepancy constituted by the green path is measured by the difference $\alpha_d - \alpha_b$ in the smallest thresholds α_b, α_d at which the two clusters are merged in $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha_b}$ and $\mathcal{G}^{w\leq\alpha_d}$. Homology is the tool that permits counting such topological features, because of the space limit we gather the definitions and necessary properties of homology in Appendix A, see also (Hatcher, 2005). The number of these simplest topological features is the dimension of the kernel of linear map $H_0(\mathcal{G}^{w\leq\alpha}) \to H_0(\mathcal{G}^{\min(w,\tilde{w})\leq\alpha})$, as basis elements of the vector space H_0 correspond to the graph connected components. It may also happen that a non-trivial merging happens between two distant parts of the same $\mathcal{G}^{w\leq\alpha}$ cluster or between two $\mathcal{G}^{w\leq\alpha}$ clusters already connected via a chain of merging, as on Figure 9. The number of these features is the dimension of the cokernel of the map $H_1(\mathcal{G}^{w\leq\alpha}) \to H_1(\mathcal{G}^{\min(w,\tilde{w})\leq\alpha})$. Hence the number of non-trivial mergings is the sum of the two numbers. We are interested in these numbers for all possible thresholds α. When the threshold α is increased then more green and blue edges appear, and also certain green edges become blue. Using an auxiliary graph and the barcodes algorithm, we calculate the numbers of such topological features for all values of α at once.

2.2. R-Cross-Barcode

Recall that the Vietoris-Rips complex of a graph $\mathcal{G}$ equipped with edge weights' matrix m is the collection of $k-$simplexes, $k \geq 0$, which are $(k+1)-$element subsets of the set of vertices of $\mathcal{G}$, with the filtration threshold of a simplex defined by the maximal weight on the edges:

$$R_\alpha(\mathcal{G}^m) = \{\{A_{i_0}, \dots, A_{i_k}\}, A_i \in \mathrm{Vert}(\mathcal{G}) | m_{A_iA_j} \leq \alpha\}$$

Our simplicial approximation to the manifold $M_{\mathcal{P}}$ at threshold α is the union of all simplexes from the simplicial complex $R_\alpha(\mathcal{G}^w)$, and similarly the approximation to $M_{\tilde{\mathcal{P}}}$ is the union of all simplexes from $R_\alpha(\mathcal{G}^{\tilde{w}})$.

The dissimilarity between the filtered simplicial complexes $R_\alpha(\mathcal{G}^w)$ and $R_\alpha(\mathcal{G}^{\tilde{w}})$ can be quantified using the homological methods. The relevant tools here are homology, barcodes and homology exact sequences. We describe our construction below and, because of space limitations, we sketch further explanation of the construction in Appendix, Section A.2.

Concretely, to compare the multi-scale topology of the two weighted graphs $\mathcal{G}^w$ and $\mathcal{G}^{\tilde{w}}$ we introduce the weighted graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ with doubled set of vertices and with the edge weights defined as follows. For convenience, fix a numbering of vertices $\mathrm{Vert}(\mathcal{G}) = \{A_1, \dots, A_N\}$. For each vertex $A \in \mathrm{Vert}(\mathcal{G})$ we add the extra vertex A' together with A to $\hat{\mathcal{G}}$, plus the unique additional vertex O, and define the distance-like edge weights in $\hat{\mathcal{G}}^{w,\tilde{w}}$ as:

$$d_{A'_iA'_j} = \min(w_{A_iA_j}, \tilde{w}_{A_iA_j}),\ d_{A_iA'_j} = d_{A_iA_j} = w_{A_iA_j},$$
$$d_{A_iA'_i} = d_{OA_i} = 0,\ d_{A_jA'_i} = d_{OA'_i} = +\infty \qquad (1)$$

where $i < j$ and $O \in \mathrm{Vert}(\hat{\mathcal{G}}^{w,\tilde{w}})$ is the additional vertex. In practice, for the calculation of RTD described below, the distance matrix can be taken in a slightly simpler form $m = \begin{pmatrix} 0 & (w_+)^\intercal \\ w_+ & \min(w,\tilde{w}) \end{pmatrix}$, where w and $\tilde{w}$ are the edge weight matrices of $\mathcal{G}^w$ and $\mathcal{G}^{\tilde{w}}$, and w_+, respectively $(w_+)^\intercal$, is the matrix w with upper-(respectively, lower-)triangular part replaced by $+\infty$.

Next, we construct the Vietoris-Rips filtered simplicial complex of the graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ and take its barcode. The doubling of vertices in $\hat{\mathcal{G}}^{w,\tilde{w}}$ creates triangles OA_iA_j, $A_iA_jA'_j$, $A_iA'_iA'_j$ at the threshold $\alpha = w_{A_iA_j}$. These triangles "kill" the edge $A'_iA'_j$ becoming blue at this threshold. Intuitively, the $i-$th barcode of $R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}})$ records the i-dimensional topological features that are born in $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ but are not yet born near the same place in $\mathcal{G}^{w\leq\alpha}$ and the $(i-1)-$dimensional topological features that are dead in $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ but are not yet dead at the same place in $\mathcal{G}^{w\leq\alpha}$, see Theorem 2.1 below.

Definition. The *R-Cross-Barcode*$_i(P, \tilde{P})$ is the set of intervals recording the "births" and "deaths" of i-dimensional

Algorithm 1 R-Cross-Barcode$_i(P, \tilde{P})$

Input: w, $\tilde{w}$: matrices of pairwise distances within point clouds P, $\tilde{P}$
Require: vr(m): function computing filtered complex from pairwise distances matrix m
Require: B(C, i): function computing persistence intervals of filtered complex C in dimension i
$w, \tilde{w} \leftarrow w, \tilde{w}$ divided by their 0.9 quantiles
$m \leftarrow \begin{pmatrix} w & (w_+)^\intercal & 0 \\ w_+ & \min(w, \tilde{w}) & +\infty \\ 0 & +\infty & 0 \end{pmatrix}$
R-Cross-Barcode$_i \leftarrow$ B(vr$(m), i)$
Return: intervals list *R-Cross-Barcode*$_i(P, \tilde{P})$ representing "births" and "deaths" of topological discrepancies between P and $\tilde{P}$.

Algorithm 2 *RTD*$(\mathcal{P}, \tilde{\mathcal{P}})$, see section 2.4 for details, suggested default values: $b = 500$, $n = 10$

Input: $\mathcal{P} \in \mathbb{R}^{|\mathcal{V}| \times D}$, $\tilde{\mathcal{P}} \in \mathbb{R}^{|\mathcal{V}| \times \tilde{D}}$: data representations
for $j = 1$ **to** n **do**
  $V_j \leftarrow$ random choice $(\mathcal{V}, b)$
  $P_j, \tilde{P}_j \leftarrow \mathcal{P}(V_j), \tilde{\mathcal{P}}(V_j)$
  $\mathcal{B}_j \leftarrow$ *R-Cross-Barcode*$_1(P_j, \tilde{P}_j)$ intervals' list calculated by Algorithm 1
  $rtd_j \leftarrow$ sum of lengths of all intervals in $\mathcal{B}_j$
end for
RTD$_1(\mathcal{P}, \tilde{\mathcal{P}}) \leftarrow$ mean(rtd)
Return: number ***RTD***$_1(\mathcal{P}, \tilde{\mathcal{P}})$ representing discrepancy between the representations $\mathcal{P}, \tilde{\mathcal{P}}$

topological features in the filtered simplicial complex $R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}})$.

The *R-Cross-Barcode*$_*(P, \tilde{P})$ (for *Representations' Cross-Barcode*) records the differences in the multiscale topology of the two embeddings. The topological features with longer lifespans indicate in general the essential features.

Theorem 2.1. *Basic properties of R-Cross-Barcode*$_*(P, \tilde{P})$*:*

- *if $P(A) = \tilde{P}(A)$ for any object $A \in V$, then R-Cross-Barcode*$_*(P, \tilde{P}) = \varnothing$*;*
- *if all distances within $\tilde{P}(V)$ are zero i.e. all objects are represented by the same point in $\tilde{P}$, then for all $k \geq 0$: R-Cross-Barcode*$_{k+1}(P, \tilde{P})$ *= Barcode*$_k(P)$ *the standard barcode of the point cloud P;*
- *for any value of threshold α, the following sequence of natural linear maps of homology groups*

$$\xrightarrow{r_{3i+3}} H_i(R_\alpha(\mathcal{G}^w)) \xrightarrow{r_{3i+2}} H_i(R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})) \xrightarrow{r_{3i+1}} \xrightarrow{r_{3i+1}} H_i(R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}})) \xrightarrow{r_{3i}} H_{i-1}(R_\alpha(\mathcal{G}^w)) \xrightarrow{r_{3i-1}} \xrightarrow{r_{3i-1}} \dots \xrightarrow{r_1} H_0(R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})) \xrightarrow{r_0} 0 \quad (2)$$

is exact, i.e. for any j the kernel of the map r_j is the image of the map r_{j+1}.

The proof of the first two properties is immediate and the third property follows from the properties of distinguished triangles of complexes, see Appendix A for more details. The exactness of the sequence (2) for $j = 1, 2, 3$ implies that the calculation of the topological features from Section 2.1 for all α is reduced to the calculation of $H_1(R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}}))$ for all α, i.e. to the calculation of *R-Cross-Barcode*$_1(P, \tilde{P})$.

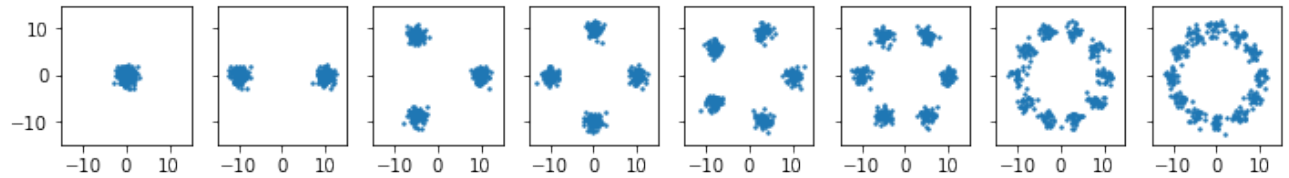
2.3. Representation Topology Divergence.

The *R-Cross-Barcode*$_*(P, \tilde{P})$ is by itself, to our opinion, a precise and intuitive tool for understanding discrepancies between two representations. There are several numerical characteristics measuring the non-emptyness of *R-Cross-Barcode*. Based on experiments and on relation of sum of bars' lengths with Earth Moving Distance (Barannikov et al., 2021), we define the sum of lengths of the bars in *R-Cross-Barcode*$_i(P, \tilde{P})$, denoted $RTD_i(P, \tilde{P})$, as the scalar characterizing the degree of topological discrepancy between the representations $P, \tilde{P}$. We use most often the average of $RTD_1(P, \tilde{P})$ and $RTD_1(\tilde{P}, P)$, denoted ***RTD*** score, in our computations below.

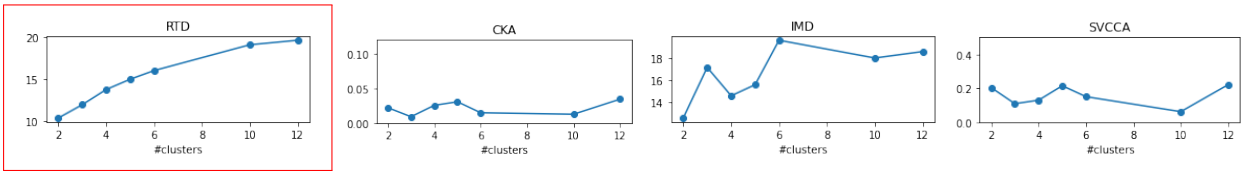
Proposition 2.2. *If $RTD_i(P, \tilde{P}) = RTD_i(\tilde{P}, P) = 0$ for all $i \geq 1$, then the barcodes of the weighted graphs $\mathcal{G}^w$ and $\mathcal{G}^{\tilde{w}}$ are the same in any degree. Moreover, in this case the topological features are located in the same places: the inclusions $R_\alpha(\mathcal{G}^w) \subseteq R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})$, $R_\alpha(\mathcal{G}^{\tilde{w}}) \subseteq R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})$ induce homology isomorphisms for any threshold α.*

2.4. Algorithm

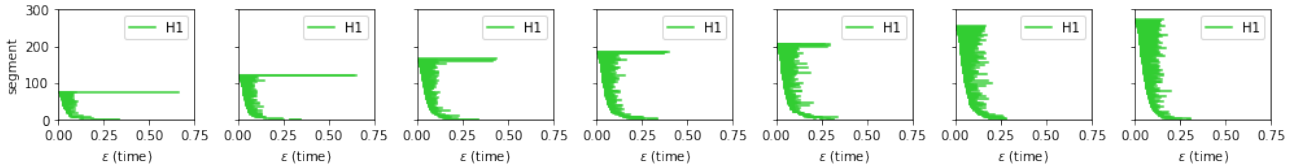
First we compute the *R-Cross-Barcode*$_1(P, \tilde{P})$ on two representations $P, \tilde{P}$ of a sample V. For this we calculate the matrices of pairwise distances w, $\tilde{w}$ within the point clouds P, $\tilde{P}$. We assume that the metrics in the ambient spaces of representations are normalized so that the two point clouds are of comparable size, namely their 0.9 quantile of pairwise distances coincide. This ensures that our score has scaling invariance, the reasonable property of a good representation similarity measure, as argued in e.g. (Kornblith et al., 2019). Next, the algorithm builds the Vietoris-Rips complex from the matrix m defined in Equation 1. Then the $1-$dimensional barcode, see (Barannikov, 2021; Chazal & Michel, 2017), of the built filtered simplicial complex is calculated. The last two steps can be done using scripts that are optimized for GPU acceleration (Zhang et al., 2020). Then we sum the lengths of bars in *R-Cross-Barcode*$_1(P, \tilde{P})$. To get the symmetric measure we usually take the half-sum



(a) Point clouds used in "clusters" experiment.



(b) Representations' comparison measures. Ideally, the measure should change monotonically with the increase of topological discrepancy.



(c) R-Cross-Barcode$(P, \tilde{P})$ for the "clusters" experiments. $\tilde{P}$ - is the point cloud having one cluster, P - 2, 3, 4, 5, 6, 10, 12 clusters.

Figure 3: RTD perfectly detects cluster structures, while rival measures fail. One cluster is compared with 2-12 clusters.

with the similar sum of bars in *R-Cross-Barcode*$_1(\tilde{P}, P)$. The computation is repeated a sufficient number of times to obtain the mean of the chosen characteristics. We have observed experimentally that about 10 times is usually sufficient for common datasets. The main steps of the computation are summarized in Algorithms 1 and 2.

Complexity. Algorithm 1 starts with computation of the two matrices of pairwise distances w, $\tilde{w}$ for a pair of representations of a sample V: $P \in \mathbb{R}^{b \times D}$, $\tilde{P} \in \mathbb{R}^{b \times \tilde{D}}$ involving $O(|V|^2(D + \tilde{D}))$ operations. Next, persistent intervals of the filtered complex must be computed. Given the distance matrix m, the complexity of their computation does not depend on the dimensions $D, \tilde{D}$ of the data representations. Generally, the barcode computation is at worst cubic in the number of simplexes involved. In practice, the computation is quite fast since the boundary matrix is typically sparse for real datasets. For R-Cross-Barcodes' calculation, we used GPU-optimized software. Thus, the computation of R-Cross-Barcode takes a similar time as in the previous step even on datasets of high dimensionality. Since only the dissimilarities in representation topology are calculated, the results are quite robust and a rather low number of iterations is needed to obtain accurate results.

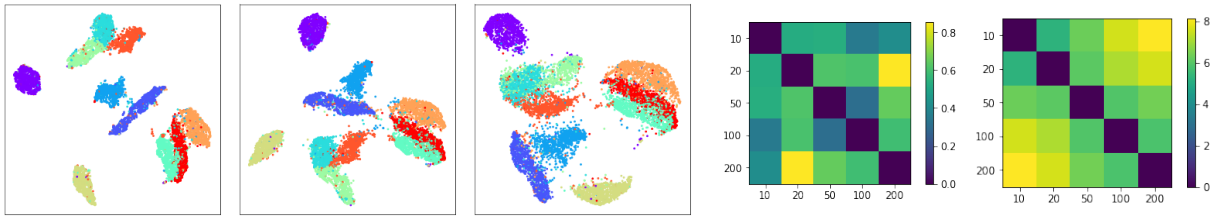
3. Experiments

In the experimental section, we study the ability of the proposed R-Cross-Barcodes and RTD to detect changes in topological structures with the use of synthetic point clouds; we demonstrate the superiority of RTD over CKA, SVCCA, IMD (Section 3.1). RTD meaningfully compares representations from UMAP with different parameters (Section 3.2). By comparing representations from various architectures (Section 3.3), layers, epochs, ensembles and after data distribution shift (Section 3.4) we show that RTD is in line with natural notion of representational similarity. A high correlation between RTD and disagreement of neural network predictions is an interesting empirical finding.

3.1. Experiments with synthetic point clouds

We start with small-scale experiments with synthetic point clouds: "clusters" and "rings". For the **"clusters" experiment** (Figure 3, top), the initial point cloud consists of 300 points randomly sampled from the 2-dimensional normal distribution having mean $(0, 0)$. Next, we split it into 2,3…12 parts (clusters) and move them to the circle of radius 10. Then, we compare the initial point cloud (having one cluster) with the split ones.

We compared these point clouds by calculating: RTD, CKA (Kornblith et al., 2019), IMD (Tsitsulin et al., 2020) and SVCCA (Raghu et al., 2017). We calculated linear CKA since (Kornblith et al., 2019) concluded that it provides the same performance as the RBF kernel, but does not require selecting a kernel width. For SVCCA, we calculated average correlation $\bar{\rho}$ for the truncation threshold 0.99, as recommended in (Raghu et al., 2017). The IMD score (Tsitsulin et al., 2020) was very noisy and we averaged it over 100 runs.



(a) 2D representations of MNIST with n_neighbors = 10, 50, 200

(b) 1-CKA

(c) RTD

Figure 4: Comparing representations of MNIST by UMAP with varying n_neighbors.

Figure 3b presents the results: RTD perfectly tracks the change of the topological complexity while the alternative measures mostly fail. The Kendall-tau rank correlations of the measures with a number of clusters are: RTD: 1.0, CKA: 0.23, IMD: 0.43, SVCCA: 0.14. We also note that RTD does not have any tunable parameters as SVCCA and does not require averaging over as many runs as IMD. Figure 3c shows the H_1 R-Cross-Barcodes calculated while comparing clusters. In accordance with the definition of RTD, H_0 barcodes are absent. The sum of the lengths of the segments increases with increasing differences in topology. Running times and all of the R-Cross-Barcodes are shown in Appendix C. Additional representation similarity measures were evaluated in Appendix I.

In the **"rings" experiment**, we compared synthetic point clouds consisting of a variable number of rings, see Figure 12a in Appendix D. Initially, there are 500 points uniformly distributed over the unit circle. Then, the points are moved onto circles with radii varying from 0.5 to 1.5. Finally, we compare the point cloud having 5 rings with other ones. Figure 12b in Appendix D present the results. RTD almost ideally reflects the change of the topological complexity while the alternative measures mostly fail. The Kendall-tau rank correlations of the measures with a number of rings are: RTD: 0.8, CKA: -0.2, IMD: 0.8, SVCCA: -0.2.

In the next sections, we compare RTD only with CKA, since it is the most popular method for comparing neural representations (Kornblith et al., 2019; Nguyen et al., 2021).

3.2. Comparing representations from UMAP

UMAP (McInnes et al., 2018) is the state-of-the-art method for visualizing high-dimensional datasets by obtaining their 2D/3D representations. We apply UMAP to the MNIST dataset to get 2D representations. We vary the number of neighbors in UMAP in the range $(10, 20, 50, 100, 200)$, see Figure 4a (all of the figures are in Appendix H). This parameter affects the cluster structure: for low values, the algorithm focuses on the local structure and clusters are crisp; for high values, the algorithm pays more attention to the global structure, and clusters were found to often overlap. Then, we perform the pairwise comparison of all the variants of 2D representations by RTD and CKA, see Figure 4. RTD reveals a nice monotonic pattern w.r.t. a number of neighbors, while values of CKA are quite chaotic.

3.3. Experiments with NAS-Bench-NLP

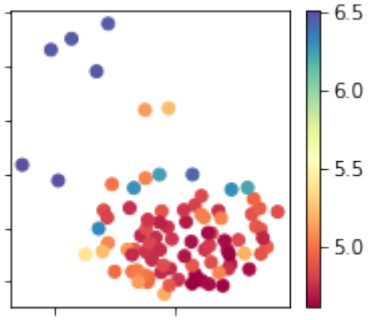


Figure 5: Multi-dimensional scaling of 90 architectures selected randomly from NAS-Bench-NLP. Color depicts log. perplexity.

Recently, neural architecture search has attracted a lot of attention in the machine learning community (Liu et al., 2019; Dong & Yang, 2019; Chen et al., 2021). NAS-Bench-NLP (Klyuchnikov et al., 2020) is a benchmark for neural architecture search which is a collection of 14,322 recurrent architectures; all of the architectures were trained on the PTB dataset. We took 90 randomly selected architectures and compared word embeddings by RTD: each architecture contains 400-dimensional embeddings of 10,000 words. Then, we evaluated all the pairwise similarities between embeddings[1] from the architectures and visualized them via multi-dimensional scaling, see Fig. 5, where color depicts a log. perplexity. According to common sense, architectures having similar embeddings have a similar log. perplexity. Also, we checked that RTD is approximately a metric for this particular case since it satisfies the triangle inequality for 97% of triplets of architectures from NAS-Bench-NLP.

[1] to speedup computation, we averaged the metrics for 10 random batches of 100 word embeddings. The average relative std. dev. of RTD was 8%.

Table 1: The correlation of metrics with Disagreement in the training dynamics experiment

	RTD	1−CKA
VGG-11	**0.976 ± 0.003**	0.818 ± 0.010
ResNet-20	**0.971 ± 0.001**	0.924 ± 0.008

3.4. Experiments with convolutional neural networks

To demonstrate the abilities of RTD to work with image representations, we train ResNet-20 (He et al., 2016) and VGG-11 (Simonyan & Zisserman, 2014) networks on CIFAR (Krizhevsky et al., 2009) datasets. In experiments, we compare RTD with CKA and disagreement of predictions. For a more intuitive comparison, we consider 1−CKA instead of CKA. As a measure of the difference in predictions, we use Disagreement (Kuncheva & Whitaker, 2003; Wen et al., 2020), the fraction of mismatched predictions calculated as $\frac{1}{N}\sum_{n=1}^{N}\left[f_{\theta_1}(x_n) \neq f_{\theta_2}(x_n)\right]$, where $f_\theta(x)$ denotes the class label predicted by the network for input x. As discussed in (Fort et al., 2019), the lower the accuracy of predictions, the higher its potential mismatch due to the possibility of the wrong answers being random, and then we normalize the Disagreement by $(1-a)$, where a is the mean accuracy of the predictions. To calculate the final metrics, we averaged the values for five random batches of 500 representations from the test dataset.

3.4.1. Training dynamics

In the first experiment, we analyze the training dynamics of neural networks. On each epoch, we collect the outputs of the convolutional part that extract the representations. To compare dynamics properly, we scaled the metrics by their maximum value. Fig. 1 shows the dynamics of the differences with the final representations. The results coincide with the intuition: the representations on each epoch become more similar to the final one. Moreover, RTD demonstrates the same behavior as disagreement of predictions. RTD better correlates with the Disagreement, see Table 1.

3.4.2. Layers

In the next experiment, we compare the outputs of layer blocks within the trained network. For VGG-11, the block has the form Conv→BN→Activation→(Pooling), and for ResNet-20, we take the output of the first Conv→BN→Activation block, and then the outputs of each residual block. In Figure 6, we see that both RTD and 1−CKA show similar results, including the slight difference between adjacent layers. We see that both metrics reveal the significant changes in the outputs of the ResNet-20 last block. In Figure 18, we performed similar experiment with ResNet-50 and ConvNeXt-tiny (Liu et al., 2022) architectures pre-trained on ImageNet-1k dataset (Deng et al., 2009).

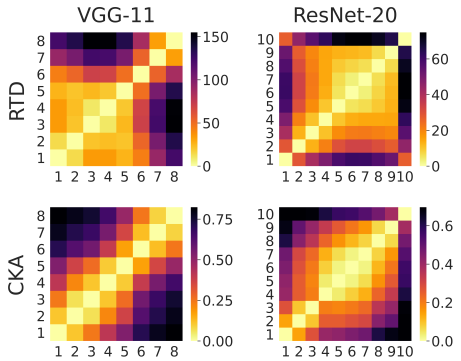


Figure 6: The representation differences between the layer blocks within trained networks. The columns correspond to the architecture, and the rows, to the metric.

Table 2: Analysis of ResNet-20 representations under different data distribution shifts. The correlation of metrics with Disagreement.

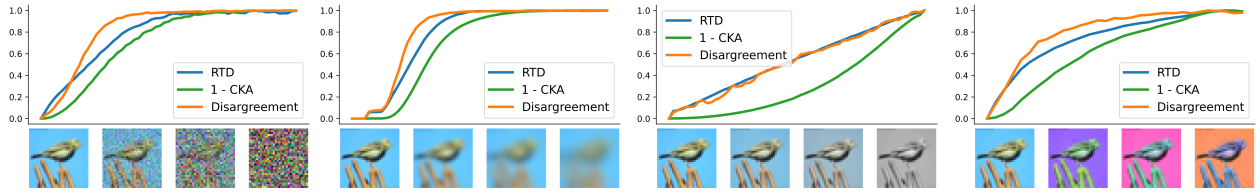
	RTD	1−CKA
Noise	**0.966 ± 0.001**	0.927 ± 0.006
Gaussian blur	**0.982 ± 0.004**	0.913 ± 0.011
Grayscale	**0.990 ± 0.004**	0.928 ± 0.040
Hue	**0.978 ± 0.008**	0.927 ± 0.017

3.4.3. Data Distribution Shift

Here, we apply the data distribution shift to test the RTD. As a shift, we consider different image transformations: noising, blurring, grayscaling, and hue changing. For each transformation, we analyze the metric dynamics as the strength of a transformation increases. Figure 7 confirms our sanity check of the monotony of RTD and other metrics with respect to data distribution shift. Moreover, Table 2 shows that RTD has a higher correlation with disagreement of predictions.

3.4.4. Ensembles

It is known that an ensemble of neural networks performs better than a single network and can estimate the uncertainty of the predictions. It is shown in (Lee et al., 2015; Opitz et al., 1996) that the diverse ensembles work better. Thus, measuring ensembles' diversity is important. The disagreement is a good example of such a measure. To show that RTD can measure the diversity as well as disagreement, we learn two types of ensembles: the classical ensemble, when we learn the networks from different random initializations, and the Fast Geometric Ensemble (FGE) (Garipov et al., 2018), which is known to have lower diversity. We learn four models for each type of ensemble and average the metrics among all pairs. The results in Table 3 confirm that RTD is capable of measuring the diversity on the same scale as the disagreement of predictions.



(a) Noise (b) Gaussian blur (c) Grayscale (d) Hue

Figure 7: Analysis of ResNet-20 representations under different data distribution shifts. The dynamics of scaled metrics with the monotonic transformations of images.

Table 3: The averaged metric among all pairs of ensemble members with a ResNet-20 architecture, and the relative difference between the types of ensemble.

	Class. Ensemble	FGE	Diff. %
RTD	15.27 ± 0.12	10.45 ± 0.32	**31.6**
1−CKA	0.094 ± 0.02	0.033 ± 0.003	64.9
Disagreement	0.915 ± 0.05	0.607 ± 0.03	**33.6**

Table 4: The correlation of metrics with Disagreement in the transfer learning experiment

	RTD	1−CKA
CIFAR-100	$\mathbf{0.98 \pm 0.01}$	0.93 ± 0.02
CIFAR-10	$\mathbf{0.91 \pm 0.01}$	0.89 ± 0.02

3.4.5. Transfer learning

Another possible application is the measure of changes in representations after transferring the pre-trained model to a new task. In this experiment, we conduct the transfer learning from CIFAR-100 to the CIFAR-10 dataset. We make full fine-tuning with the small learning rate for the convolutional part. In Fig. 8, we demonstrate the dynamics for both dataset representations. The results again coincide with the intuition about the difference during the learning steps, and here RTD has also a high correlation with Disagreement, see Table 4. Also, we note that RTD can be applied to the continual learning task, where catastrophic forgetting appears, and thus it is crucial to track the changes in network representations.

3.5. Additional experiments

We describe how RTD can be used to evaluate a disentanglement of generative models in Appendix G. Comparisons of BigGAN's internal representations by RTD agree with those of images by FID, see Appendix E.

4. Conclusions

In this paper, we have proposed a topologically-inspired approach to compare neural network representations. The most widely used methods for this problem are statistical: Canonical Correlation Analysis (CCA) and Centered Kernel Alignment (CKA). But the problem itself is a geometric one: the comparison of two neural representations of the same objects is de-facto the comparison of two points clouds from different spaces. The natural way is to compare their geometrical and topological features with due account of their localization — that is exactly what was done by the R-Cross-Barcode and RTD. We demonstrated that RTD coincides with the natural assessment of representations similarity. We used the RTD to gain insights into neural network representations in computer vision and NLP domains for various problems: training dynamics analysis, data distribution shift, transfer learning, ensemble learning, and disentanglement assessment.

RTD correlates strikingly well with the disagreement of models' predictions; this is an intriguing topic for further research. Finally, R-Cross-Barcode and RTD are general tools that are not limited only to the comparison of representations. They could be applied to other problems involving comparison of two point clouds with one-to-one correspondence, for example, in 3D computer vision.

Acknowledgements. The work was supported by the Analytical center under the RF Government (subsidy agreement 000000D730321P5Q0002, Grant No. 70-2021-00145 02.11.2021).

References

Barannikov, S. Framed Morse complexes and its invariants. *Adv. Soviet Math.*, 22:93–115, 1994.

Barannikov, S. Canonical Forms = Persistence Diagrams. Tutorial. In *European Workshop on Computational Geometry (EuroCG 2021)*, 2021.

Barannikov, S., Trofimov, I., Sotnikov, G., Trimbach, E., Korotin, A., Filippov, A., and Burnaev, E. Manifold

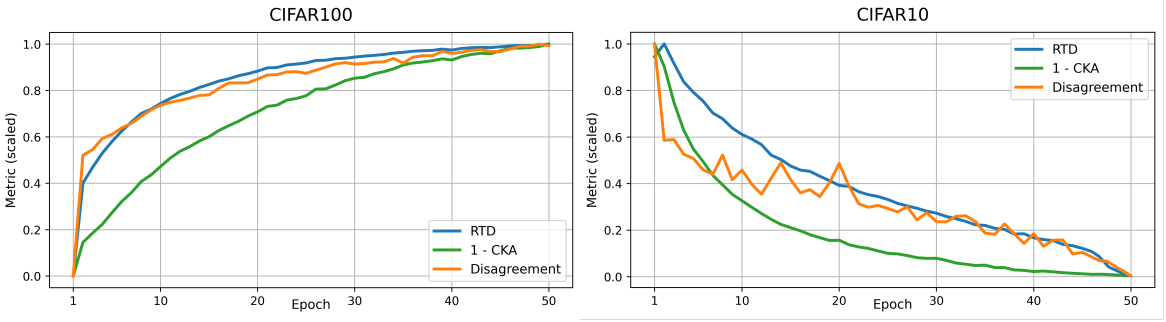


Figure 8: Scaled metrics demonstrating the difference between representations of CIFAR-100 and CIFAR-10 datasets during fine-tune process.

Topology Divergence: a framework for comparing data manifolds. In *Proceedings of the 35th International Conference on Neural Information Processing Systems, NeurIPS'21, arXiv:2106.04024*, 2021.

Belkin, M. and Niyogi, P. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, pp. 585–591, 2001.

Brock, A., Donahue, J., and Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Chazal, F. and Michel, B. An introduction to topological data analysis: fundamental and practical aspects for data scientists. *arXiv:1710.04019*, 2017.

Chen, W., Gong, X., and Wang, Z. Neural architecture search on imagenet in four gpu hours: A theoretically inspired perspective. *International Conference on Learning Representations*, 2021.

Connolly, A. C., Guntupalli, J. S., Gors, J., Hanke, M., Halchenko, Y. O., Wu, Y.-C., Abdi, H., and Haxby, J. V. The representation of biological classes in the human brain. *Journal of Neuroscience*, 32(8):2608–2618, 2012.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Dong, X. and Yang, Y. Searching for a robust neural architecture in four gpu hours. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1761–1770, 2019.

Edelman, S. Representation is representation of similarities. *Behavioral and brain sciences*, 21(4):449–467, 1998.

Fort, S., Hu, H., and Lakshminarayanan, B. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D., and Wilson, A. G. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 8803–8812, 2018.

Gelfand, S. I. and Manin, Y. I. *Methods of homological algebra*. Springer Science & Business Media, 2002.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT press Cambridge, 2016.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*, pp. 63–77. Springer, 2005.

Hatcher, A. *Algebraic topology*. 2005.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local Nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Khrulkov, V. and Oseledets, I. Geometry score: A method for comparing generative adversarial networks. In *International Conference on Machine Learning*, pp. 2621–2629. PMLR, 2018.

Klyuchnikov, N., Trofimov, I., Artemova, E., Salnikov, M., Fedorov, M., and Burnaev, E. Nas-bench-nlp: neural architecture search benchmark for natural language processing. *arXiv preprint arXiv:2006.07116*, 2020.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529. PMLR, 2019.

Kriegeskorte, N., Mur, M., and Bandettini, P. A. Representational similarity analysis-connecting the branches of systems neuroscience. *Frontiers in systems neuroscience*, 2:4, 2008.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Kuncheva, L. I. and Whitaker, C. J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning*, 51(2):181–207, 2003.

Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., and Aila, T. Improved precision and recall metric for assessing generative models. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2019.

Le Peutrec, D., Nier, F., and Viterbo, C. Precise Arrhenius law for p-forms: The Witten Laplacian and Morse–Barannikov complex. *Annales Henri Poincaré*, 14(3): 567–610, Apr 2013. ISSN 1424-0661. doi: 10.1007/s00023-012-0193-9.

Lee, S., Purushwalkam, S., Cogswell, M., Crandall, D., and Batra, D. Why m heads are better than one: Training a diverse ensemble of deep networks. *arXiv preprint arXiv:1511.06314*, 2015.

Li, Y., Yosinski, J., Clune, J., Lipson, H., Hopcroft, J. E., et al. Convergent learning: Do different neural networks learn the same representations? In *FE@ NIPS*, pp. 196–212, 2015.

Liu, H., Simonyan, K., and Yang, Y. Darts: Differentiable architecture search. *International Conference on Learning Representations*, 2019.

Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11976–11986, 2022.

McInnes, L., Healy, J., and Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Moor, M., Horn, M., Rieck, B., and Borgwardt, K. Topological autoencoders. In *International Conference on Machine Learning*, pp. 7045–7054. PMLR, 2020.

Morcos, A. S., Raghu, M., and Bengio, S. Insights on representational similarity in neural networks with canonical correlation. *arXiv preprint arXiv:1806.05759*, 2018.

Nguyen, T., Raghu, M., and Kornblith, S. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. *International Conference on Learning Representations*, 2021.

Niyogi, P., Smale, S., and Weinberger, S. Finding the homology of submanifolds with high confidence from random samples. *Discrete & Computational Geometry*, 39(1-3): 419–441, 2008.

Opitz, D. W., Shavlik, J. W., et al. Generating accurate and diverse members of a neural-network ensemble. *Advances in neural information processing systems*, pp. 535–541, 1996.

Raghu, A., Raghu, M., Bengio, S., and Vinyals, O. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *International Conference on Learning Representations*, 2020.

Raghu, M., Gilmer, J., Yosinski, J., and Sohl-Dickstein, J. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *arXiv preprint arXiv:1706.05806*, 2017.

Simonyan, K. and Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Tsitsulin, A., Munkhoeva, M., Mottin, D., Karras, P., Bronstein, A., Oseledets, I., and Mueller, E. The shape of data: Intrinsic distance for data distributions. In *International Conference on Learning Representations*, 2020.

Voita, E., Sennrich, R., and Titov, I. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. *EMNLP*, 2019.

Wang, L., Hu, L., Gu, J., Wu, Y., Hu, Z., He, K., and Hopcroft, J. Towards understanding learning representations: To what extent do different neural networks learn the same representation. *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Wen, Y., Tran, D., and Ba, J. Batchensemble: An alternative approach to efficient ensemble and lifelong learning. *ArXiv*, abs/2002.06715, 2020.

Whitehead, G. W. *Elements of homotopy theory*, volume 61. Springer Science & Business Media, 1968.

Wu, J. M., Belinkov, Y., Sajjad, H., Durrani, N., Dalvi, F., and Glass, J. Similarity analysis of contextual word representation models. *Proceedings of ACL*, 2020.

Zhang, S., Xiao, M., and Wang, H. Gpu-accelerated computation of Vietoris-Rips persistence barcodes. In *36th International Symposium on Computational Geometry (SoCG 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

Zhou, S., Zelikman, E., Lu, F., Ng, A. Y., Carlsson, G., and Ermon, S. Evaluating the disentanglement of deep generative models through manifold topology. *preprint arXiv:2006.03680*, 2020.

Zomorodian, A. J. *Computing and comprehending topology: Persistence and hierarchical Morse complexes (Ph.D.Thesis)*. University of Illinois at Urbana-Champaign, 2001.

A. Background on Simplicial Complexes. Barcodes

The simplicial complex is a combinatorial object that can be thought of as a higher-dimensional generalization of a graph.

A simplex is defined via the set of its vertices. Given a finite set V, a $k-$simplex is a finite $(k+1)-$element subset in V. Simplicial complex S is a collection of $k-$simplexes, $k \geq 0$, which satisfies the natural condition that for each $\sigma \in S$, $\sigma' \subset \sigma$ implies $\sigma' \in S$. A simplicial complex consisting only of $0-$ and $1-$simplexes is a graph.

Denote via $C_k(S)$ the vector space over the field $\mathbb{Z}/2\mathbb{Z} = \{0, 1\}$ whose basis elements are $k-$simplexes from S. The boundary linear operator $\partial_k : C_k(S) \to C_{k-1}(S)$ is defined on $\sigma = \{A_0, \ldots, A_k\}$ as

$$\partial_k \sigma = \sum_{j=0}^{k} \{A_0, \ldots, A_{j-1}, A_{j+1}, \ldots, A_k\}.$$

The kth **homology** group $H_k(S)$ is the factor vector space $\ker \partial_k / \operatorname{im} \partial_{k+1}$. The elements $c \in \ker \partial_k$ are called cycles. The elements of $H_k(S)$ represent various $k-$dimensional topological features in S. A basis in $H_k(S)$ corresponds to a set of basic topological features.

For example, the vector space H_0 has the basis whose elements are in one-to-one correspondence with equivalence classes of vertices, connected by paths of $1-$simplices (edges), i.e. with connected components of S. The basis elements of the vector space H_1 correspond to basic equivalence classes of nontrivial closed paths of $1-$simplices. Two closed paths, also named $1-$cycles, are equivalent if they are connected by a chain of modifications by boundaries of triangles ($2-$simplices).

A map $S_1 \to S_2$, e.g. $\mathcal{G}^{w \leq \alpha} \to \mathcal{G}^{\min(w, \tilde{w}) \leq \alpha}$ see section 2.1, defines the maps $H_k(S_1) \to H_k(S_2)$. The kernel of the linear map $H_0(S_1) \to H_0(S_2)$ is spanned by the pairs of S_1 clusters merged together in S_2. The cokernel of the linear map $H_1(S_1) \to H_1(S_2)$ consists of $1-$cycles in S_2 which are not from S_1, i.e. it consists of equivalence classes of closed paths in S_2, which cannot be modified by boundaries of triangles into images of $1-$cycles from S_1.

In applications, the simplicial complexes are often built via consequential adding of simplexes one after another in increasing order of some numerical characteristics. Mathematically this corresponds to a filtration on the simplicial complex. It is defined as a family of simplicial complexes S_α, indexed by a finite set of real numbers, with nested collections of simplexes: for $\alpha_1 < \alpha_2$ all simplexes of S_{α_1} are also in S_{α_2}. An example of a filtered simplicial complex is the Vietoris-Rips simplicial complex from Section 2.2.

The inclusions $S_\alpha \subseteq S_\beta$ induce the maps on homology $H_k(S_\alpha) \to H_k(S_\beta)$. The evolution of cycles across the nested family of simplicial complexes S_{α_i} is described by the principal persistent homology theorem (Barannikov, 1994; Zomorodian, 2001; Le Peutrec et al., 2013), according to which for each dimension there exists a choice of a set of basic topological features across all nested simplicial complexes S_α so that each basic feature c appears in $H_k(S_\alpha)$ at specific time $\alpha = b_c$ and disappears at specific time $\alpha = d_c$. The barcode of the filtered complex is the record of the appearance, or "birth" time, and the disappearance, or "death" time, of all these basic topological features.

A.1. Exact sequence and topological features

A sequence of vector spaces and linear maps

$$A_5 \xrightarrow{r_4} A_4 \xrightarrow{r_3} A_3 \xrightarrow{r_2} A_2 \xrightarrow{r_1} A_1 \tag{3}$$

is exact at A_j if the kernel of the linear map r_{j-1} coincides with the image of the previous map r_j.

Proposition A.1. *If the sequence (3) is exact at A_2, A_3, A_4 then $A_3 \simeq \operatorname{Ker}(r_1) \oplus \operatorname{Coker}(r_4)$.*

Proof. Since $A/\operatorname{Ker}(r) \simeq \operatorname{Image}(r)$ for any linear map $r : A \to A'$, therefore $A_3 \simeq \operatorname{Image}(r_2) \oplus \operatorname{Ker}(r_2)$. If the sequence is exact at A_2, then $\operatorname{Image}(r_2) \simeq \operatorname{Ker}(r_1)$. Exactness at A_3, A_4 gives $\operatorname{Ker}(r_2) \simeq \operatorname{Image}(r_3)$, $\operatorname{Ker}(r_3) \simeq \operatorname{Image}(r_4)$. Then $\operatorname{Image}(r_3) \simeq A_4/\operatorname{Ker}(r_3)$ imply that $\operatorname{Ker}(r_2) \simeq A_4/\operatorname{Image}(r_4)$, which equals $\operatorname{Coker}(r_4)$, the cokernel of the linear map r_4. Hence $A_3 \simeq \operatorname{Ker}(r_1) \oplus \operatorname{Coker}(r_4)$. □

Therefore the exact sequence from Theorem 2.1 implies that the calculation of the topological features from Section 2.1 for all α is reduced to the calculation of $H_1(R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}}))$ for all α, i.e. to the calculation of *R-Cross-Barcode*$_1(P, \tilde{P})$.

A.2. Construction of R-Cross-Barcode

Here we gather some intuition behind the construction of the graph $\hat{\mathcal{G}}^{w,\tilde{w}}$ and the *R-Cross-Barcode*.

The Vietoris-Rips complex $R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})$ is the union of simplexes whose edges connect data points with distance less than α in at least one of representations $\mathcal{P}, \tilde{\mathcal{P}}$. An inclusion of simple simplicial complexes $S \subset R$ is an equivalence in homotopy category, if and only if the induced map on homology is an isomorphism (Whitehead, 1968). The maps on homology induced by the inclusions of filtered simplicial complexes

$$R_\alpha(\mathcal{G}^w) \subseteq R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}),\ R_\alpha(\mathcal{G}^{\tilde{w}}) \subseteq R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}) \tag{4}$$

should therefore be as close as possible to isomorphisms, in order that the approximations at threshold α to the manifolds $M_{\mathcal{P}}$ and $M_{\tilde{\mathcal{P}}}$ have essentially the same geometric features located at the same places. It follows from the exact sequence from Theorem 2.1 that the R-Cross-Barcode$_*(P, \tilde{P})$ is exactly the list of topological features describing the failure of the maps induced on homology by inclusions (4) to be isomorphisms.

Introduce the weighted graph $\hat{\mathcal{G}}^w$ with doubled set of vertices and with the edge weights defined as follows. We fix the numbering of vertices $\text{Vert}(\mathcal{G}) = \{A_1, \dots, A_N\}$. Let us add the extra vertex A' together with A to $\hat{\mathcal{G}}^w$ for each vertex $A \in \text{Vert}(\mathcal{G})$, plus the two additional vertexes O, O', and define the distance-like edge weights in $\hat{\mathcal{G}}^w$ as:

$$\begin{gathered} d_{A_iA_j} = d_{A_iA'_j} = w_{A_iA_j}, \\ d_{A'_iA'_j} = d_{A_iA'_i} = d_{O'A'_i} = d_{OA_i} = 0,\ d_{A'_jA_i} = d_{O'A_i} = d_{OA'_i} = d_{OO'} = +\infty \end{gathered} \tag{5}$$

where $i < j$ and $O, O' \in \text{Vert}(\hat{\mathcal{G}}^w)$ are the two additional vertexes.

The suspension $C[-1]$ of chain complex C denotes the same chain complex with degree shifted by 1, $C[-1]_n = C_{n-1}$, so that the the nth chains of $R_\alpha(\mathcal{G}^w)[-1]$ are linear combinations of $(n-1)-$dimensional simplexes from $R_\alpha(\mathcal{G}^w)$. We denote via $A_{i_1} \dots A_{i_n}[-1]$ the element from $C_n(R_\alpha(\mathcal{G}^w)[-1])$ corresponding to the simplex $A_{i_1} \dots A_{i_n}$.

A chain map f between two chain complexes (C, d_C) and (B, d_B) is a sequence of linear maps $f_n : C_n \to B_n$ that commutes with the boundary operators: $d_{B,n} \circ f_n = f_{n-1} \circ d_{C,n}$. The cone of a chain map f is the chain complex $\text{Cone}(f) = C[-1] \oplus B$ with differential $d_{\text{Cone}(f)} = \begin{pmatrix} d_{C[-1]} & 0 \\ f[-1] & d_B \end{pmatrix}$. A homotopy equivalence is a pair of chain maps $f : C \to B$, $g : B \to C$, and a pair of maps $h_{C,n} : C_n \to C_{n+1}$, $h_{B,n} : B_n \to B_{n+1}$, such that $g \circ f = \text{Id} + [h_C, d_C]$ and $f \circ g = \text{Id} + [h_B, d_B]$. We assume that the Vietoris-Rips complexes are augmented with $C_{-1} = \mathbb{Z}/2\mathbb{Z}$ and $\partial_0\{A_i\} = 1$.

The proof of the exact homology sequence from Theorem 2.1 follows from the following two propositions.

Proposition A.2. *There are homotopy equivalences of chain complexes:*

$$R_\alpha(\mathcal{G}^w)[-1] \sim R_\alpha(\hat{\mathcal{G}}^w) \tag{6}$$

$$\text{Cone}\left(R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})\right) \sim R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}}). \tag{7}$$

Proof. The simplexes of the chain complex $R_\alpha(\hat{\mathcal{G}}^w)$ are of four types: $A_{i_1} \dots A_{i_k} A'_{i_k} \dots A'_{i_n}$, $A_{i_1} \dots A_{i_k} A'_{i_{k+1}} \dots A'_{i_n}$, $OA_{i_1} \dots A_{i_n}$ and $O'A'_{i_1} \dots A'_{i_n}$ where $A_{i_k} \in \text{Vert}(\mathcal{G})$, $i_0 < \dots < i_k < i_{k+1} < \dots < i_n$, with edge weights satisfying $w_{A_{i_r}A_{i_s}} < \alpha$ for $r \le k$. Define the map $\phi : R_\alpha(\mathcal{G}^w)[-1] \to R_\alpha(\hat{\mathcal{G}}^w)$

$$\phi : A_{i_1} \dots A_{i_n}[-1] \mapsto OA_{i_1} \dots A_{i_n} + O'A'_{i_1} \dots A'_{i_n} + \sum_{k=1}^{n} A_{i_1} \dots A_{i_k} A'_{i_k} \dots A'_{i_n} \tag{8}$$

The map ϕ together with the map $\tilde{\phi} : R_\alpha(\hat{\mathcal{G}}^w) \to R_\alpha(\mathcal{G}^w)[-1]$

$$\tilde{\phi} : OA_{i_1} \dots A_{i_n} \mapsto A_{i_1} \dots A_{i_n}[-1],\ \tilde{\phi}(\Delta) = 0 \text{ for any other simplex } \Delta, \tag{9}$$

gives a homotopy equivalence, $\tilde{\phi} \circ \phi = \text{Id}$, $\phi \circ \tilde{\phi} = \text{Id} + [h, \partial]$, where the homotopy h is given by

$$\begin{gathered} h : A_{i_1} \dots A_{i_k} A'_{i_{k+1}} \dots A'_{i_n} \mapsto \sum_{l=1}^{k} A_{i_1} \dots A_{i_l} A'_{i_l} \dots A'_{i_n} + O'A'_{i_1} \dots A'_{i_n} \\ h : A'_{i_1} \dots A'_{i_n} \mapsto O'A'_{i_1} \dots A'_{i_n},\ h(\Delta) = 0 \text{ for any other simplex } \Delta. \end{gathered} \tag{10}$$

Simplexes of the chain complex $R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}})$ are of three types. The first type: $A_{i_1}\dots A_{i_k}A'_{i_k}\dots A'_{i_n}$ with edge weights satisfying $w_{A_{i_r}A_{i_s}} < \alpha$ for $r \leq k$ and $\min(w_{A_{i_r}A_{i_s}}, \tilde{w}_{A_{i_r}A_{i_s}}) < \alpha$ for $r,s > k$; the second type: $A_{i_1}\dots A_{i_{k-1}}A'_{i_k}\dots A'_{i_n}$ with edge weights satisfying $w_{A_{i_r}A_{i_s}} < \alpha$ for $r < k$, and $\min(w_{A_{i_r}A_{i_s}}, \tilde{w}_{A_{i_r}A_{i_s}}) < \alpha$ for $r,s \geq k$; and the third type: $OA_{i_1}\dots A_{i_n}$ with edge weights satisfying $w_{A_{i_r}A_{i_s}} < \alpha$ for all r,s. Define the map $\psi : \mathrm{Cone}\left(R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})\right) \to R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}})$

$$\psi : A_{i_1}\dots A_{i_n}[-1] \mapsto OA_{i_1}\dots A_{i_n} + \sum_{k=1}^{n} A_{i_1}\dots A_{i_k}A'_{i_k}\dots A'_{i_n} \tag{11}$$

for $A_{i_1}\dots A_{i_n}[-1] \in R_\alpha(\mathcal{G}^w)[-1]$,

$$\psi : A_{i_1}\dots A_{i_n} \mapsto A'_{i_1}\dots A'_{i_n} \tag{12}$$

for $A_{i_1}\dots A_{i_n} \in R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})$. The map ψ together with the map $\tilde{\psi} : R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}}) \to \mathrm{Cone}\left(R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})\right)$

$$\begin{gathered}\tilde{\psi} : OA_{i_1}\dots A_{i_n} \mapsto A_{i_1}\dots A_{i_n}[-1],\ A_{i_1}\dots A_{i_n}[-1] \in R_\alpha(\mathcal{G}^w)[-1],\\ A'_{i_1}\dots A'_{i_n} \mapsto A_{i_1}\dots A_{i_n},\ A_{i_1}\dots A_{i_n} \in R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}),\\ \tilde{\psi}(\Delta) = 0 \text{ for any other simplex } \Delta,\end{gathered} \tag{13}$$

gives a homotopy equivalence, $\tilde{\psi}\circ\psi = \mathrm{Id}$, $\psi\circ\tilde{\psi} = \mathrm{Id} + [H,\partial]$, where the homotopy H is given by

$$\begin{gathered}H : A_{i_1}\dots A_{i_k}A'_{i_{k+1}}\dots A'_{i_n} \mapsto \sum_{l=1}^{k} A_{i_1}\dots A_{i_l}A'_{i_l}\dots A'_{i_n},\ 1 \leq k \leq n\\ H(\Delta) = 0 \text{ for any other simplex } \Delta.\end{gathered} \tag{14}$$

□

The long exact sequences such as (2) arise from distinguished triangles in the homotopy category of chain complexes. A distinguished triangle is a diagram isomorphic in this category to a diagram $A \xrightarrow{f} B \to \mathrm{Cone}(f) \to A[-1]$.

Proposition A.3. *The embeddings of graphs $\mathcal{G}^{w\leq\alpha} \subseteq \mathcal{G}^{\min(w,\tilde{w})\leq\alpha} \subset \hat{\mathcal{G}}^{w,\tilde{w}\leq\alpha}$ give distinguished triangles, see (Gelfand & Manin, 2002), in the homotopy category of chain complexes:*

$$R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}) \to R_\alpha(\hat{\mathcal{G}}^{w,\tilde{w}}) \to R_\alpha(\mathcal{G}^w)[-1]. \tag{15}$$

Proof. Taken together the homotopy equivalences (8)-(14) define an isomorphism of (15) with the distinguished triangle

$$R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}) \to \mathrm{Cone}\left(R_\alpha(\mathcal{G}^w) \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})\right) \to R_\alpha(\mathcal{G}^w)[-1]. \tag{16}$$

□

Comparison with Cross-Barcode and Geometry Score. The Cross-Barcode from (Barannikov et al., 2021) compares two data manifolds lying in the same ambient space. It does not use the information that can be provided by a one-to-one correspondence between points of the two data clouds. To compare the locations of topological features the Cross-Barcode from loc.cit. uses instead the proximity information inferred from the pairwise distances between points from different clouds lying in the same ambient space. Geometry score from (Khrulkov & Oseledets, 2018) is based on a comparison of standard barcodes for each cloud and is insensitive to the location of topological features, for example, it does not detect any difference when similar topological features are located geometrically in distant places of the two clouds.

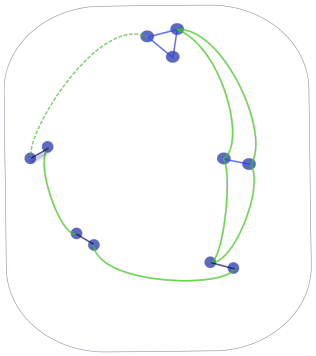


Figure 9: Merging between clusters already connected via a chain of mergings.

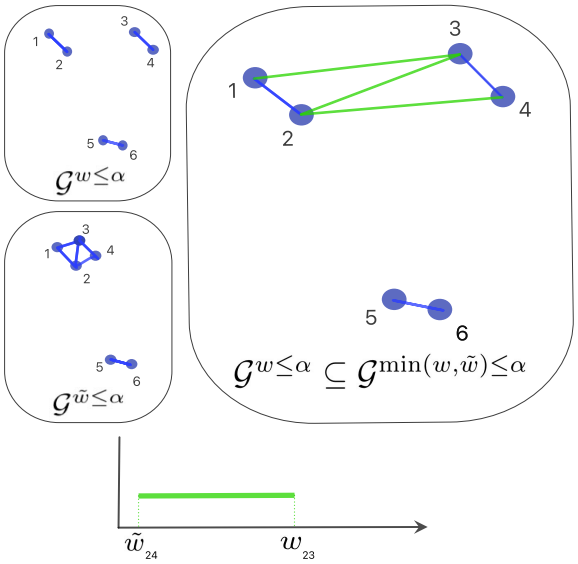


Figure 10: Merging of three clusters into two clusters. Graphs $\mathcal{G}^{w\leq\alpha}$, $\mathcal{G}^{\tilde{w}\leq\alpha}$ and $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ are shown. Edges of $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$ not in $\mathcal{G}^{w\leq\alpha}$ are colored in green. In this example there are exactly four different weights $(13), (14), (23), (24)$ in the graphs $\mathcal{G}^{w\leq\alpha}$ and $\mathcal{G}^{\min(w,\tilde{w})\leq\alpha}$. The unique topological feature in *R-Cross-Barcode*$_1(P, \tilde{P})$ in this case is born at the threshold $\tilde{w}_{24}$ when the difference in the cluster structures of the two graphs arises, as the points 2 and 4 are in the same cluster at this threshold in $\mathcal{G}^{\min(w,\tilde{w})}$ and not in $\mathcal{G}^{w}$. This feature dies at the threshold w_{23} since the clusters containing 2 and 4 are merged at this threshold in $\mathcal{G}^{w}$.

B. Discussion of CKA

Given two series of equal size $x_i \in \mathbb{R}^{n_x}$, $y_i \in \mathbb{R}^{n_y}$, $i = 1 \dots n$ the CKA (Kornblith et al., 2019) is defined as

$$\text{CKA}(K, L) = \frac{\text{HSIC}(K, L)}{\sqrt{\text{HSIC}(K, K)\text{HSIC}(L, L)}}$$

where HSIC(K, L) is a Hilbert-Schmidt Independence Criterion (Gretton et al., 2005), $K_{i,j} = k(x_i, x_j)$, $L_{i,j} = l(y_i, y_j)$, $L = E - n^{-1}$ where $k(\cdot,\cdot)$, $l(\cdot,\cdot)$ are kernels. HSIC itself an empirical estimate of the Hilbert-Schmidt norm of the cross-covariance operator. HSIC is equivalent to maximum mean discrepancy between the joint distribution $P(X, Y)$ and the product of the marginal distributions $P(X)P(Y)$; HSIC = 0 implies independence of X and Y if the associated kernel is universal.

However, CKA is sometimes applied to measure similarity between representations from different layers of a neural network. In this case $Y = f(X)$. X and Y are tightly dependent and the joint distribution can always be factorized as

$P(X, Y) = P(Y|X)P(X)$. Thus, the application of CKA to the comparison of representation from different layers is questionable.

C. Details on experiments with synthetic point clouds

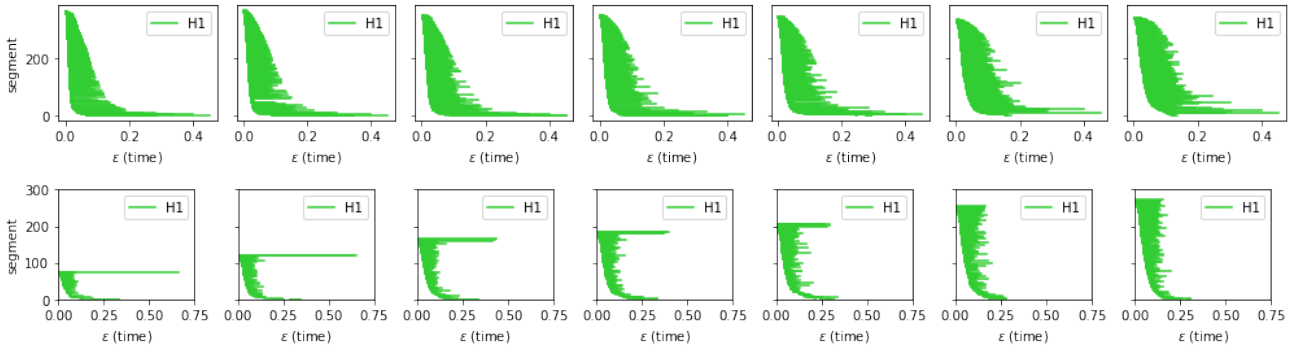
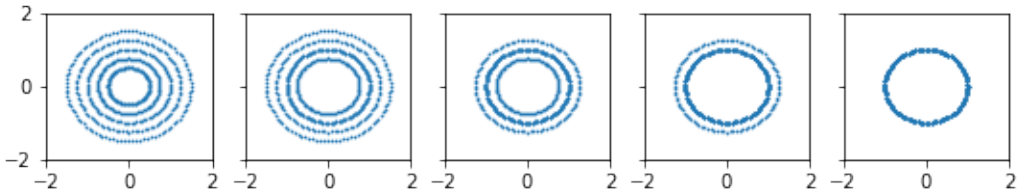


Figure 11: R-Cross-Barcodes for the "clusters" experiments. Top: R-Cross-Barcode$(\tilde{P}, P)$, Bottom: R-Cross-Barcode$(P, \tilde{P})$; $\tilde{P}$ is the point cloud having one cluster; P- 2, 3, 4, 5, 6, 10, 12 clusters.

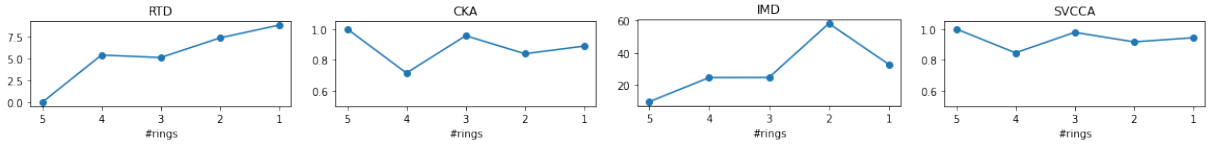
Runtime comparison. Here we present the total wall time of the experiments with synthetic point clouds:

"Clusters experiment": RTD: 19.7 s, CKA: 0.07 s, IMD: 83 s, SVCCA: 0.03 s.
"Rings experiment": RTD: 144 s, CKA: 0.7 s, IMD: 91 s, SVCCA: 0.6 s.

D. Details on the "rings" experiment



(a) Point clouds used in "rings" experiment.



(b) Representations' comparison measures. Ideally, the measure should change monotonically with the increase of topological discrepancy.

Figure 12: RTD perfectly detects changes in topology, while rival measures fail. Five rings are compared with 5,4,3,2,1 rings.

E. Experiment with BigGAN

In this experiment, we applied RTD and CKA for comparison of internal representations in BigGAN (Brock et al., 2018) [2].

Initially, we generated a set of $k = 100$ random latent codes $Z_0 = \{z_{0,j}\}_{j=1}^{k}$ and derived sets $Z_1, \ldots, Z_n$ by adding to Z_0 a Gaussian noise of increasing strength $z_{i,j} = z_{0,j} + \epsilon_{i,j}$, where $\epsilon_{i,j} \sim N(0, \sigma_i)$. The noise standard deviation σ_i grows from 0.001 to 0.25 by a logarithmic scale and the difference between Z_0 and Z_i tends to increase when i increases.

[2] we used the pretrained model from
https://github.com/lukemelas/pytorch-pretrained-gans

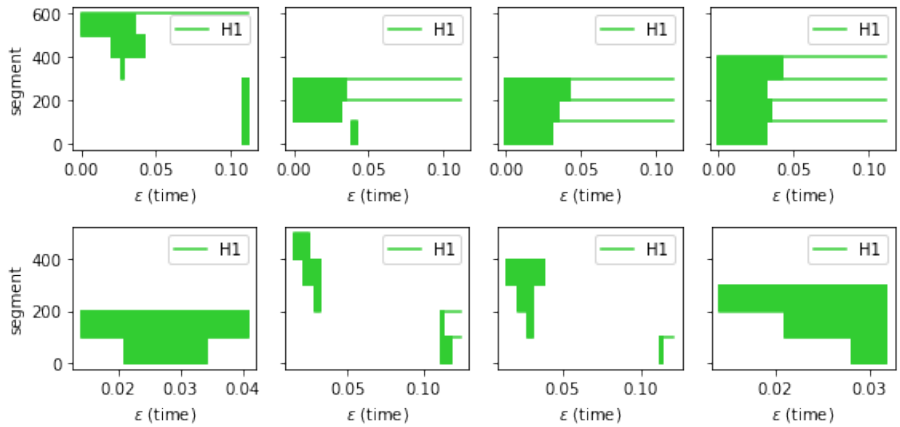


Figure 13: R-Cross-Barcodes for the "rings" experiments. Top: R-Cross-Barcode$(P, \tilde{P})$, Bottom: R-Cross-Barcode$(\tilde{P}, P)$. P - is the point cloud having 5 rings, $\tilde{P}$ - 4, 3, 2, 1 rings.

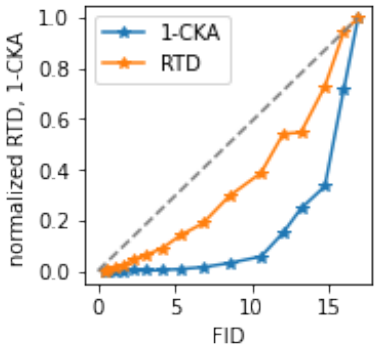


Figure 14: Comparison of normalized RTD, CKA (computed for sets of internal representations) vs. FID (computed for sets of images).

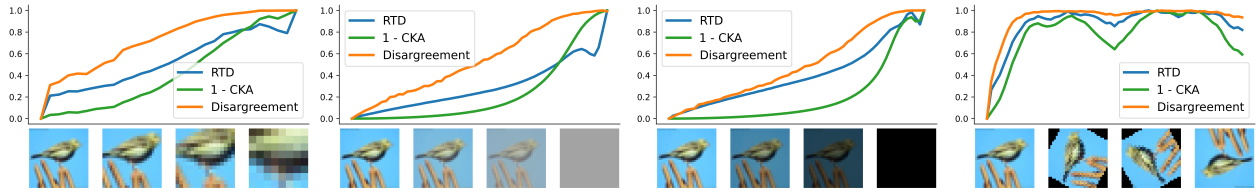
Then, we pass sets of latent codes $Z_0, \ldots, Z_n$ together with vector encoding of the "husky" class through the BigGAN and save internal representations R_i for one of the top layers (results were quite similar for other layers). Also, we get sets of images $I_0, \ldots, I_k$. To compare these sets we used the state-of-the-art measure FID (Heusel et al., 2017) which is often applied for GAN evaluation.

It is natural to assume that the difference between sets of internal representations R_0 and R_i should have a good correlation with the difference between sets of images I_0 and I_i. To check this hypothesis, we calculated RTD(R_0, R_i), CKA(R_0, R_i) and compared them with FID(I_0, I_i), for $i = 1, \ldots, n$. Figure 14 shows the results. We conclude that RTD enjoys higher correlation with FID: 0.97, while the correlation of CKA and FID is lower: 0.79.

F. Details on experiments with convolutional networks

Metrics to correlate		Noise	Gaussian Blur	Grayscale	Hue
Disagreement	RTD	**0.966 ± 0.001**	**0.982 ± 0.004**	**0.990 ± 0.004**	**0.978 ± 0.008**
	1−CKA	0.927 ± 0.006	0.913 ± 0.011	0.928 ± 0.040	0.927 ± 0.017
Error rate	RTD	**0.982 ± 0.002**	0.963 ± 0.007	0.856 ± 0.052	0.935 ± 0.030
	1−CKA	0.966 ± 0.007	**0.999 ± 0.001**	**0.958 ± 0.018**	**0.944 ± 0.033**

Table 5: Analysis of ResNet-20 representations under different data distribution shifts. The correlation of RTD and 1−CKA with Disagreement and Error rate.



(a) Zoom (b) Contrast (c) Brightness (d) Rotation

Figure 15: Analysis of ResNet-20 representations under different data distribution shifts. The dynamics of scaled metrics with the monotonic application of various types of image transformations.

Metrics to correlate		Zoom	Brightness	Contrast	Rotation
Disagreement	RTD	**0.950 ± 0.006**	**0.975 ± 0.002**	**0.936 ± 0.010**	**0.955 ± 0.015**
	1−CKA	0.886 ± 0.010	0.854 ± 0.024	0.851 ± 0.021	0.857 ± 0.020
Error rate	RTD	0.946 ± 0.006	0.921 ± 0.011	0.937 ± 0.005	0.940 ± 0.009
	1−CKA	**0.994 ± 0.002**	**0.997 ± 0.001**	**0.998 ± 0.001**	**0.981 ± 0.005**

Table 6: Analysis of ResNet-20 representations under different data distribution shifts. The correlation of RTD and 1−CKA with Disagreement and Error rate.

G. Experiments with disentanglement

Learning disentangled representations is a fundamental problem for improving the generalization, robustness, and interpretability of generative models. (Zhou et al., 2020) proposed to evaluate the disentanglement of generative models by comparing the topology of data manifold slices. Let Z be a latent space, X - a space of objects, $g : Z \to X$ - a generator. (Zhou et al., 2020) compares slices $X_v = g(Z \mid_{z_i=v})$ for different values of v. If the direction z_i corresponds to an interpretable factor, then X_v must be topologically similar for different v.

We use the following experimental design. $Z_{v,n} = \{z \in Z \mid (z, n) = v\}$ - a slice in a latent space orthogonal to a unit vector n. We take a finite random sample $Z_1 \subset Z_{v,n}$ and a shifted sample $Z_2 = \{z_i + \delta n\}_{i=1}^{|Z_1|}$ for small δ. By definition, Z_1 and Z_2 have natural point-wise mapping and we can estimate homological similarity of $g(Z_1)$ and $g(Z_2)$ by RTD.

In this experiment, we use `dSprites`[3] for the evaluation of disentanglement. `dSprites` is a dataset of procedurally generated 2D shapes from 5 ground truth independent latent factors: shape, scale, rotation, x-position, and y-position of a sprite. Thus, the latent space is disentangled and fully factorized. Particularly, we compare the slices orthogonal to axis-aligned vectors and orthogonal to random vectors, see Table 10. Except for the first axis, the topological dissimilarity estimated by RTD is significantly less than for a random direction. The first axis corresponds to a categorical factor - shape for which the aforementioned approach is arguably not applicable. The `dSprites` dataset is quite simple and RTD was calculated for point clouds in the pixel space. However, the same technique can be straightforwardly applied to evaluate the disentanglement of image representations for more complex datasets.

	RTD	1−CKA
Disagreement	**0.98 ± 0.01**	0.93 ± 0.02
Error rate	0.9 ± 0.03	**0.99 ± 0.01**

(a) CIFAR-100

	RTD	1−CKA
Disagreement	**0.91 ± 0.01**	0.89 ± 0.02
Error rate	0.60 ± 0.02	**0.73 ± 0.01**

(b) CIFAR-10

Table 7: The correlation of metric dynamics when transferring the ResNet-20 network from CIFAR-100 to CIFAR-10 dataset.

	VGG-11	ResNet-20
Number of epochs	100	
Optimizer	SGD, momentum=0.9	
Learning rate (initial)	0.1	
Scheduler	<50%: 0.1 50-90%: 0.1-0.001 (linear) >90%: 0.001	
Batch size	128	

Table 8: Details on learning the neural networks from random initialization on CIFAR datasets.

	Encoder part	Classifier part
Number of epochs	50	
Optimizer	SGD, momentum=0.9	
Learning rate (initial)	0.001	0.1
Scheduler	None	<50%: 0.1 50-90%: 0.1-0.001 (linear) >90%: 0.001
Batch size	128	

Table 9: Details on fine-tuning the ResNet-20 from CIFAR-100 to CIFAR-10 dataset.

H. Details on dimensionality reduction of MNIST with UMAP

Visual inspection of Figure 17 reveals apparent incoherences of CKA. Denote by $U(n)$ representations obtained by UMAP with the number of neighbors n. According to CKA (Figure 4b), $U(10)$ is closer to $U(200)$ than to $U(20)$; also $U(200)$ is closer to $U(10)$ than to $U(100)$.

I. Additional experiments

For the "clusters" experiments, we did additional comparisons of point clouds with alternative similarity measures. Firstly, we calculated CKA with the RBF kernel for 3 bandwidths equal to 0.2, 0.4, 0.8 of median pairwise distances (as proposed in (Kornblith et al., 2019)). The performance as measured by Kendall-tau correlation with the true ordering was 0.23, 0.04, 0.14 - not better than for CKA with the linear kernel. Secondly, we applied the topological loss term from (Moor et al., 2020). The performance as measured by Kendall-tau correlation with the true ordering was poor: -0.52.

J. Internal similarity of Neural Network layers

Here we compare the outputs of layer blocks within the trained network. We consider ResNet-50 and ConvNeXt-tiny (Liu et al., 2022) architectures pre-trained on ImageNet-1k dataset (Deng et al., 2009). We calculate RTD, CKA and SVCCA within outputs after each Bottleneck Residual Block or ConvNeXt's block respectively. In Fig. 18, we plot similarity

[3] https://github.com/deepmind/dsprites-dataset

Table 10: Evaluation of the disentanglement for various directions in the latent space of `dSprites`.

axis	**RTD**
axis 1	148.1
axis 2	71.3
axis 3	53.4
axis 4	41.2
axis 5	40.5
random	162.8 ± 18.6

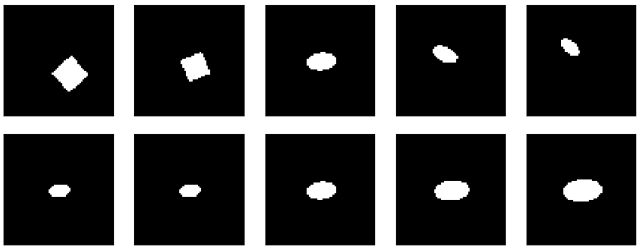


Figure 16: dSprites generated across directions in the latent space, top: random direction, bottom: axis-aligned direction, corresponds to an interpretable factor of variation.

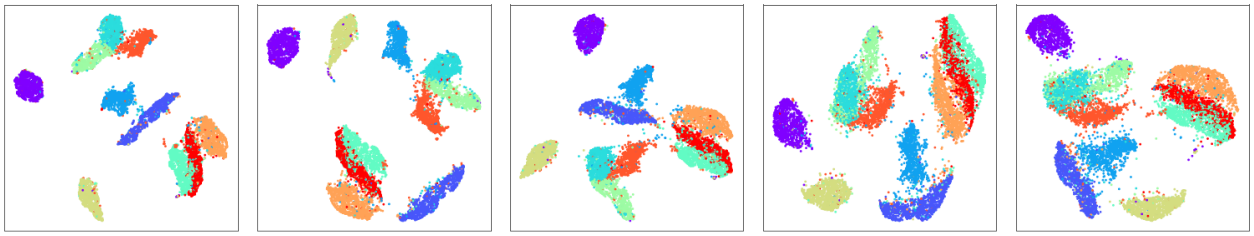


Figure 17: 2D representations of MNIST produced by UMAP, n_neighbors $\in (10, 20, 50, 100, 200)$

between layers within each architecture. We observe that RTD catches architecture's block structure better than CKA, SVCCA. The ResNet-50 architecture has sequence of blocks in form [3, 4, 6, 3] and it can be seen that RTD highlights it with sub-squares of corresponding sizes.

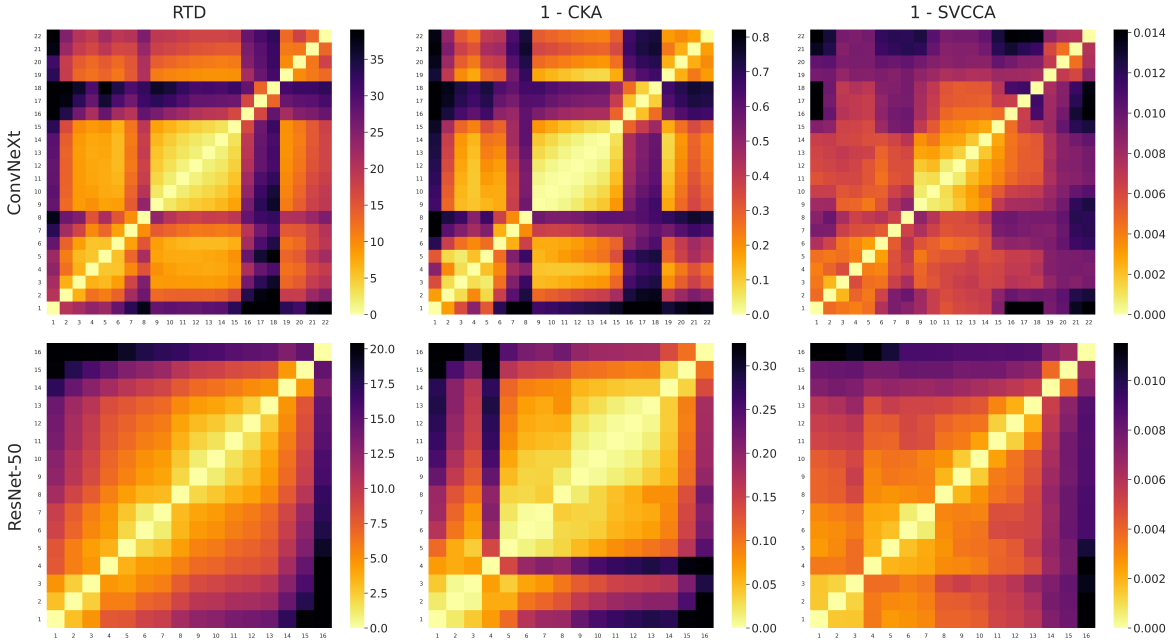


Figure 18: The representation differences between the layer blocks within trained networks, ImageNet-1k dataset. The columns correspond to the metrics, and the rows – to the architectures.