# REPRESENTATION TOPOLOGY DIVERGENCE: A METHOD FOR COMPARING NEURAL NETWORK REPRESENTATIONS.

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Comparison of data representations is a complex multi-aspect problem that has not enjoyed a complete solution yet. We propose a method for comparing two data representations. We introduce the Representation Topology Divergence (RTD) measuring 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 TDA-based practical methods 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 on neural networks representations in computer vision and NLP domains for various problems: training dynamics analysis, data distribution shift, transfer learning, ensemble learning, disentanglement assessment.

## 1 INTRODUCTION

Representations of objects are the essential component learnt by deep neural networks. In opposite to the distance in the original space, similarity of representations are proved to be semantically meaningful. Despite of the significant practical success of deep neural networks many aspect of their behaviour are poorly understood. Only few methods study learned 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). Hovewer, CCA-like measures define similarity too loosely since they are invariant to 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 it to be more consistent with the intuitive similarity of representations. Particularly, neural networks learn similar representation from different seeds as evaluated by CKA. Another line of work studies alignment between groups of neurons (Li et al., 2015), (Wang et al., 2018). The similarity of representation is also a topic of a study in neuroscience (Edelman, 1998; Kriegeskorte et al., 2008; Connolly et al., 2012).

Representations' comparison 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), 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 representations' comparison. We propose the *Representation Topology Divergence (RTD)* score which measures a dissimilarity between two point clouds of equal size with 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 doesn't incorporate the requirement of one-to-one correspondence (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). Such

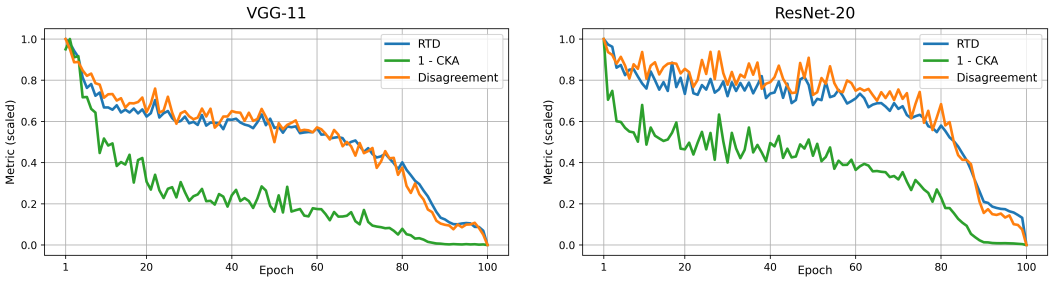



Figure 1: Comparison of representations after the i-th epoch and the final one done by RTD, 1−CKA, and disagreement of predictions. All the measures are normalized by division to their maximal values. Strikingly, RTD highly correlates with the disagreement of models' predictions.

methods (except for (Tsitsulin et al., 2020)) are mostly applied to the evaluation of GANs, where point clouds of real and generated objects are matched. Recently, (Moor et al., 2020) proposed a loss term to compare a topology of data in original and latent spaces (with natural one-to-one correspondence) and applied it as a part of the Topological Autoencoder.

In this work, we make the following contributions:

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 differences in multi-scale topology of two point clouds $P, \tilde{P}$ with one-to-one correspondence between points;
3. Based on the *R-Cross-Barcode*$(P, \tilde{P})$, we define the *Representation Topology Divergence (RTD)*, the scalar measuring the multi-scale topological dissimilarity between two representations;
4. By doing computational experiments, we show that RTD agrees with an intuitive notion of neural network representations similarity. In contrast with most existing approaches, RTD score is sensitive to cluster and other topological structures of the representations and enjoys 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, disentanglement. We have also compared RTD with CKA, IMD and SVCCA.

## 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 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 generally 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 simplices with vertices from $P$. A common approach to select the simplices approximating $M_{\mathcal{P}}$ is to fix a threshold $\alpha > 0$ and consider the simplices with edge lengths not exceeding $\alpha$ (Niyogi et al., 2008; Belkin & Niyogi, 2001). It is difficult in general to guess the correct value of the threshold, hence a reasonable viewpoint is to study all thresholds at once, see e.g. (Chazal & Michel, 2017). This can be accomplished by means of the mathematical tool, called barcode, that quantifies the evolution of manifold topology features over multiple scales.

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

### 2.1 R-CROSS-BARCODE

Let $\mathcal{P}(\mathcal{V}), \tilde{\mathcal{P}}(\mathcal{V})$ be two representations giving two embeddings of the same data. The two embeddings $\mathcal{P}, \tilde{\mathcal{P}}$ belong in general to different ambient spaces and we 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 representation $P = \mathcal{P}(V)$ defines the weighted graph $\mathcal{G}^w$ with the vertex set $V$. The weight $w_{AB}$ of an edge $AB$ is given by the distance between points $P(A)$ and $P(B)$. Similarly the representation $\tilde{P} = \tilde{\mathcal{P}}(V)$ defines the weighted graph $\mathcal{G}^{\tilde{w}}$ on the same vertex set.

The simplicial approximation to the manifold $M_{\mathcal{P}}$ at threshold $\alpha$ consists of simplices whose edges in $\mathcal{G}^w$ have weights not exceeding $\alpha$. Let $\mathcal{G}^{w \leq \alpha}$ denote the graph with the vertex set $V$ and the edges with weights not exceeding $\alpha$. To compare the simplicial approximations to 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 embed both graphs into the graph $\mathcal{G}^{\min(w,\tilde{w}) \leq \alpha}$. The graph $\mathcal{G}^{\min(w,\tilde{w}) \leq \alpha}$ contains an edge between vertices $A$ and $B$ exactly when the distance between the points $A$ and $B$ is smaller than $\alpha$ in *at least one* of the representations $P$, $\tilde{P}$.

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

$$R_\alpha(\mathcal{G}^m) = \left\{ \{A_0, \ldots, A_k\}, A_i \in \text{Vert}(\mathcal{G}) \| m_{A_j A_l} \leq \alpha \right\}$$

Our simplicial approximations to the manifolds $M_{\mathcal{P}}$, $M_{\tilde{\mathcal{P}}}$ at threshold $\alpha$ are the unions of all simplices from the simplicial complexes $R_\alpha(\mathcal{G}^w)$, $R_\alpha(\mathcal{G}^{\tilde{w}})$.

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 the homology, the Whitehead theorem and the homology exact sequence. Because of the space limitations we sketch how this leads to our construction, described below, in the Appendix, Section B.

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 each vertex $A \in \text{Vert}(\mathcal{G})$ we add the extra vertex $A'$ together with $A$ to $\hat{\mathcal{G}}$ and define the distance-like edge weights in $\hat{\mathcal{G}}(w, \tilde{w})$ as:

$$d_{AB} = \min(w_{AB}, \tilde{w}_{AB}),\ d_{AB'} = d_{A'B} = w_{AB},\ d_{AA'} = 0,\ d_{A'B'} = 0 \tag{1}$$

where $B \in \text{Vert}(\mathcal{G})$, $A \neq B$.

In practice $\mathcal{G}$, $\hat{\mathcal{G}}$ are full graphs and the edge weights on the graph $\hat{\mathcal{G}}(w, \tilde{w})$ are given by $2N \times 2N$, $N = |V|$, symmetric matrix

$$m = \begin{pmatrix} 0 & w \\ w & \min(w, \tilde{w}) \end{pmatrix} \tag{2}$$

where $w$ and $\tilde{w}$ are the distance-like edge weights matrices of $\mathcal{G}^w$ and $\mathcal{G}^{\tilde{w}}$.

Next, we construct the Vietoris-Rips filtered simplicial complex from the graph $\hat{\mathcal{G}}(w, \tilde{w})$. Intuitively, the $i-$th barcode of $R_\alpha(\hat{\mathcal{G}}(w, \tilde{w}))$ records the $i-$dimensional topological features that are born in $R_\alpha(\mathcal{G}^{\tilde{w}})$ but are not yet born at the same place in $R_\alpha(\mathcal{G}^w)$ and the $(i+1)-$dimensional topological features that are dead in $R_\alpha(\mathcal{G}^{\tilde{w}})$ but are not yet dead at this place in $R_\alpha(\mathcal{G}^w)$, see Theorem 1 below.

**Definition**. The *R-Cross-Barcode*$_i(P, \tilde{P})$ is the set of intervals recording the "births" and "deaths" of i-dimensional 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 multiscale topology of the two embeddings. The topological features with longer lifespans indicate in general the essential features.

**Theorem 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 R-Cross-Barcode*$_*(P, \tilde{P})$ *= Barcode*$_*(P)$ *the standard barcode of the point cloud $P$;*

**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 $i$

$m \leftarrow \begin{pmatrix} 0 & w \\ w & \min(w, \tilde{w}) \end{pmatrix}$

*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 2.3 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*$(\mathcal{P}, \tilde{\mathcal{P}}) \leftarrow$ mean$(rtd)$

**Return:** number ***RTD***$(\mathcal{P}, \tilde{\mathcal{P}})$ representing discrepancy between the representations $\mathcal{P}, \tilde{\mathcal{P}}$

- *for any value of threshold $\alpha$, the following sequence of natural linear maps of homology groups*

$$\ldots \to H_i(R_\alpha(\mathcal{G}^w)) \to H_i(R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})) \to H_i(R_\alpha(\hat{\mathcal{G}}(w, \tilde{w}))) \to \\ \to H_{i-1}(R_\alpha(\mathcal{G}^w)) \to H_{i-1}(R_\alpha(\mathcal{G}^{\min(w,\tilde{w})})) \to \ldots \quad (3)$$

*is exact; recall that it means that the kernel of any map is the image of the previous map*

The proof of the first two properties is immediate and the third property follows from the exactness of the corresponding sequence of simplicial complexes, see Appendix for more details.

### 2.2 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 its relation with Earth Moving Distance, see (Barannikov et al., 2021), we have defined 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 oftenly the average of $RTD_1(P, \tilde{P})$ and $RTD_1(\tilde{P}, P)$, denoted ***RTD*** score, in our computations below.

**Proposition 1.** *If $RTD_i(P, \tilde{P}) = RTD_i(\tilde{P}, P) = 0$ for all $i \geq 0$ 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 for any threshold $\alpha$ the isomorphisms in homology.*

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

*Complexity.* Algorithm 1 starts with the 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 doesn't depend on the dimensions $D, \tilde{D}$ of the data representations. Generally, the barcode computation is at worst cubic in the number of simplices involved. In practice, the calculation is quite fast since the boundary matrix is typically sparse for real datasets. For R-Cross-Barcodes' calculation we used the GPU-optimized software. Thus, the computation of R-Cross-Barcode takes similar time as in the previous step even on datasets of big 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 the proposed R-Cross-Barcodes and RTD to detect changes of topological structures using synthetic point clouds; we demonstrate the superiority of RTD over CKA, SVCCA, IMD (Section 3.1). 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 neural representations' similarity. High correlation of RTD and disagreement of networks' predictions is an interesting empirical finding. Additionally, empirical evidence suggests that RTD is useful for evaluating disentanglement of representations (Section 3.2).

### 3.1 Experiments with synthetic point clouds

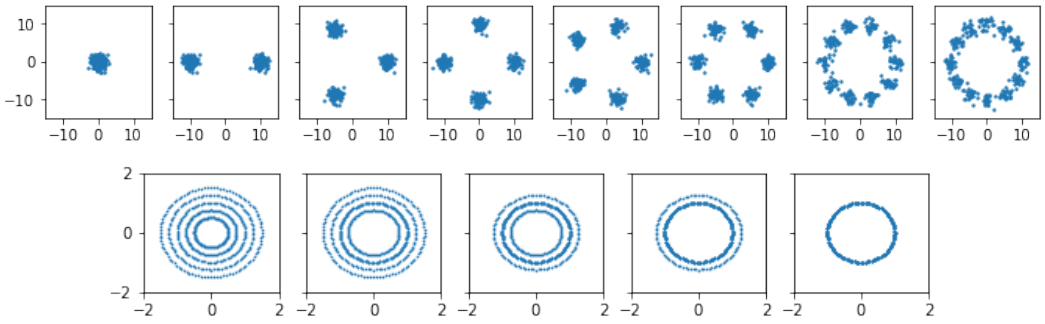
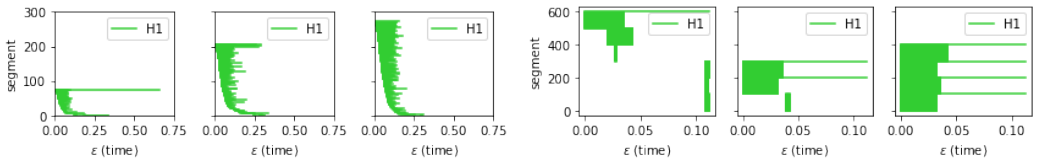


Figure 2: Point clouds used in "clusters" (top) and "rings" (bottom) experiments.




(a) "clusters" experiment: 1 cluster vs. 2, 6, 12 clusters (b) "rings" experiment: 5 rings vs. 4, 3, 1 rings

Figure 3: Examples of R-Cross-Barcodes for experiments with synthetic point clouds.

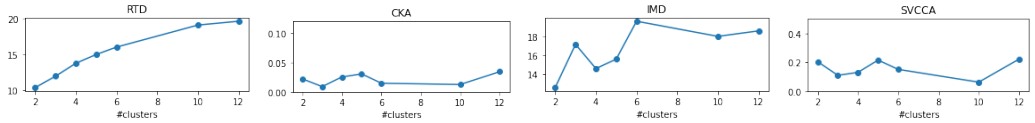



Figure 4: Representations' comparison measures for the "clusters" point clouds. Ideally, the evaluation measure should monotonically change with the increase of topological complexity.

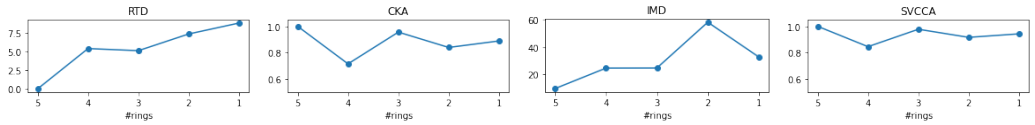



Figure 5: Representations' comparison measures for the "rings" point clouds. Ideally, the evaluation measure should monotonically change with the increase of topological complexity.

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

In the "rings" experiment, we compare synthetic point clouds consisting of a variable number of rings, see Figure 2, bottom. Initially, there are 500 points uniformly distributed over the unit circle. Then, the points are moved onto circles with varying radius, from $0.5$ to $1.5$. Finally, we compare the leftmost point cloud having 5 rings with other ones.

In both of the experimental settings there is a one-to-one correspondence between points in the point clouds. We compared these point clouds by calculating: RTD (proposed), 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 with RBF kernel but doesn't require to select a kernel width. For SVCCA, we calculated average correlation $\bar{\rho}$ for the truncation threshold 0.99, as recommended by the authors (Raghu et al., 2017). The IMD score (Tsitsulin et al., 2020) was very noisy and we averaged it over 100 runs.

Figures 4, 5 present the results. RTD measure 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 clusters are: RTD: 1.0, CKA: 0.23, IMD: 0.43, SVCCA: 0.14; for the number of rings: RTD: 0.9, CKA: -0.2, IMD: 0.9, SVCCA: -0.2. We also note than RTD score doesn't have any tunable parameters as SVCCA and doesn't require averaging over as many runs as IMD. Figure 3 shows some of the $H_1$ R-Cross-Barcodes calculated while comparing clusters and rings. In accordance with the definition, $H_0$ barcodes are absent. The sum of lengths of segments increases as differences of topology increases. All of the R-Cross-Barcodes are in Appendix D.

### 3.2 Experiments with disentanglement

Learning disentangled representations is a fundamental problem for improving generalization, robustness, and interpretability of generative models. Zhou et al. (2020) proposed to evaluate the disentanglement of generative models by comparing 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 $X_v = g(Z\,|_{z_i=v})$ for different values of $v$. If the direction $z_i$ corresponds to an interpretable factor, then $X_v$ must be topologically similar for different $v$.

Table 1: 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 | $162.8 \pm 18.6$ |

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 $\delta$. 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`[1] 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 axes-aligned vectors and orthogonal to

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

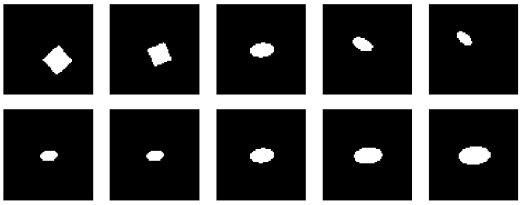


Figure 6: dSprites generated across directions in the latent space, top: random direction, bottom: axis-aligned direction, corresponds to an interpretable factor of variation.

random vectors, see Table 1. Except for the first axis, the topological dissimilarity estimated by RTD is significantly less then 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 disentanglement of image representations for more complex datasets.

### 3.3 Experiments with NAS-Bench-NLP

Recently, neural architecture search attracted a lot of attention in the machine learning community (Liu et al., 2019; Dong & Yang, 2019; Chen et al., 2021). Klyuchnikov et al. (2020) developed a benchmark for neural architecture search which is a collection of 14'322 recurrent architectures; all of them were trained on the PTB dataset. We took 90 random architectures and compared word embeddings: each architecture contains 400-dimensional embeddings of 10'000 words. Then, we evaluated all the pairwise similarities between embeddings[2] from the architectures and visualized them via multi-dimensional scaling, see fig. 7, where color depicts a log. perplexity. Accordingly to a common sense, architectures having similar embeddings have similar log. perplexity.

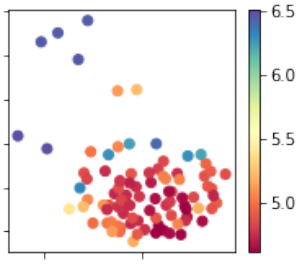



Figure 7: Multi-dimensional scaling of 90 random architectures from NAS-Bench-NLP. Color depicts log. perplexity.

### 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 the experiments, we compare RTD with CKA and disagreement of predictions. For 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 values for five random batches of 500 representations from the test dataset.

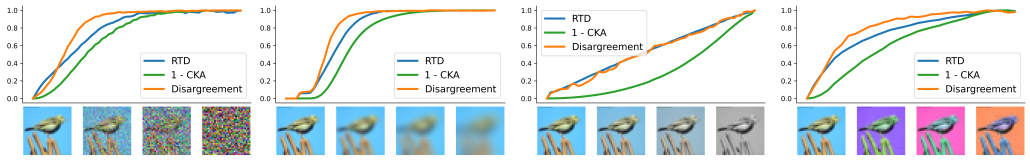
#### 3.4.1 Training dynamics

As 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 differences with the final representations.

Table 2: The correlation of metrics with Disagreement in the training dynamics experiment

| | RTD | $1-$CKA |
|---|---|---|
| VGG-11 | **0.99** | 0.83 |
| ResNet-20 | **0.98** | 0.93 |

[2]to speedup the computation, we averaged metrics for 10 random batches of 100 word embedding.




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

Figure 9: Analysis of ResNet-20 representations under different data distribution shifts. The dynamics of scaled metrics with the monotonic transformations of images.

| | RTD | 1−CKA |
|---|---|---|
| Noise | **0.968** | 0.928 |
| Gaussian blur | **0.982** | 0.915 |
| Grayscale | **0.998** | 0.937 |
| Hue | **0.982** | 0.916 |

Table 3: Analysis of ResNet-20 representations under different data distribution shifts. The correlation of metrics with Disagreement.

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 2.

### 3.4.2 Layers

As 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 fig. 8, we see that both RTD and 1−CKA show similar results, including the slight difference between adjacent layers. Also, we can see that both metrics reveal the significant changes in the outputs of the ResNet-20 last block.

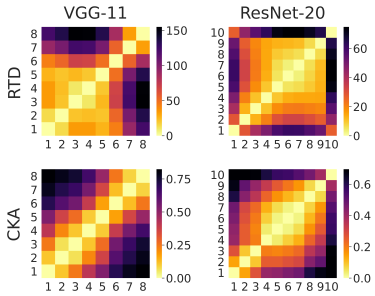



Figure 8: The representation differences between the layer blocks within trained networks. The columns correspond to the architecture, and the rows correspond to the metric.

### 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 a strength of transformation increases. Fig. 9 confirms our sanity check about the monotony of RTD and other metrics with respect to data distribution shift. Moreover, Table 3 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 showed 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 one, when we learn the networks from different random initializations, and the Fast Geometric Ensemble (FGE) (Garipov et al., 2018), which is known to have the lower diversity. We learn four models for each type of ensemble and average the metrics among all pairs. The results in Table 4 confirm that RTD can measure the diversity on the same scale as the disagreement of predictions.

| | Classical Ensemble | FGE | Difference, % |
|---|---|---|---|
| RTD | $15.27 \pm 0.12$ | $10.45 \pm 0.32$ | **31.6** |
| 1−CKA | $0.094 \pm 0.02$ | $0.033 \pm 0.003$ | 64.9 |
| Disagreement | $0.915 \pm 0.05$ | $0.607 \pm 0.03$ | **33.6** |

Table 4: The averaged metric among all pairs of ensemble members with a ResNet-20 architecture, and the relative difference between the types of ensemble.

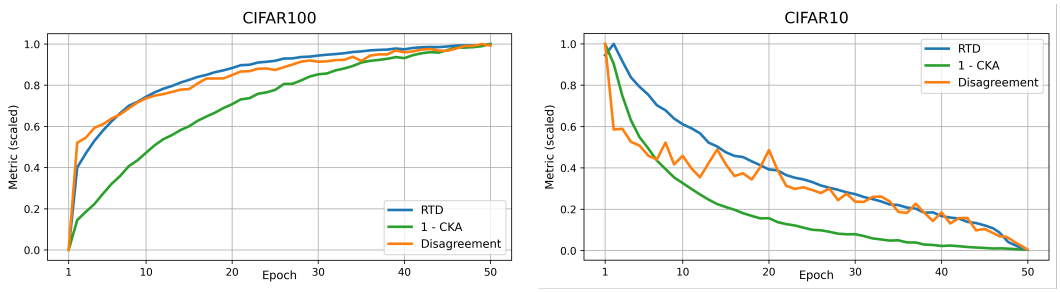



Figure 10: Scaled metrics demonstrating the difference between representations of CIFAR-100 and CIFAR-10 datasets during fine-tune process.

### 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 CIFAR-10 dataset. We make full fine-tuning with the small learning rate for the convolutional part. In fig. 10, 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 high correletion with Disagreement, see Table 5. 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 of network representations.

## 4 Conclusions

Table 5: The correlation of metrics with Disagreement in the transfer learning experiment

| | RTD | 1−CKA |
|---|---|---|
| CIFAR-100 | **0.99** | 0.93 |
| CIFAR-10 | **0.92** | 0.89 |

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 geometrical one: comparison of two neural representations of the same objects is de-facto comparison of two points clouds from different spaces having one-to-one correspondence. The natural way is to compare their geometrical and topological features, taking into account their localization, that is exactly 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 on neural networks representations in computer vision and NLP domains for various problems: training dynamics analysis, data distribution shift, transfer learning, ensemble learning, disentanglement assessment.

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

## REFERENCES


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

Serguei Barannikov, Ilya Trofimov, Grigorii Sotnikov, Ekaterina Trimbach, Alexander Korotin, Alexander Filippov, and Evgeny Burnaev. Manifold Topology Divergence: a framework for comparing data manifolds. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems, NeurIPS'21, arXiv:2106.04024*, 2021.

Mikhail Belkin and Partha Niyogi. 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.

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

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

Andrew C Connolly, J Swaroop Guntupalli, Jason Gors, Michael Hanke, Yaroslav O Halchenko, Yu-Chien Wu, Hervé Abdi, and James V Haxby. The representation of biological classes in the human brain. *Journal of Neuroscience*, 32(8):2608–2618, 2012.

Xuanyi Dong and Yi Yang. 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.

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

Stanislav Fort, Huiyi Hu, and Balaji Lakshminarayanan. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. 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.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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

Nikita Klyuchnikov, Ilya Trofimov, Ekaterina Artemova, Mikhail Salnikov, Maxim Fedorov, and Evgeny Burnaev. Nas-bench-nlp: neural architecture search benchmark for natural language processing. *arXiv preprint arXiv:2006.07116*, 2020.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pp. 3519–3529. PMLR, 2019.

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

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

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

Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2019.

Stefan Lee, Senthil Purushwalkam, Michael Cogswell, David Crandall, and Dhruv Batra. Why m heads are better than one: Training a diverse ensemble of deep networks. *arXiv preprint arXiv:1511.06314*, 2015.

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

Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *International Conference on Learning Representations*, 2019.

Michael Moor, Max Horn, Bastian Rieck, and Karsten Borgwardt. Topological autoencoders. In *International Conference on Machine Learning*, pp. 7045–7054. PMLR, 2020.

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

Thao Nguyen, Maithra Raghu, and Simon Kornblith. 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.

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

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

Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. *International Conference on Learning Representations*, 2020.

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

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

Anton Tsitsulin, Marina Munkhoeva, Davide Mottin, Panagiotis Karras, Alex Bronstein, Ivan Oseledets, and Emmanuel Mueller. The shape of data: Intrinsic distance for data distributions. In *International Conference on Learning Representations*, 2020.

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

Liwei Wang, Lunjia Hu, Jiayuan Gu, Yue Wu, Zhiqiang Hu, Kun He, and John Hopcroft. Towards understanding learning representations: To what extent do different neural networks learn the same representation. *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: An alternative approach to efficient ensemble and lifelong learning. *ArXiv*, abs/2002.06715, 2020.

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

John M Wu, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. Similarity analysis of contextual word representation models. *Proceedings of ACL*, 2020.

Sharon Zhou, Eric Zelikman, Fred Lu, Andrew Y. Ng, Gunnar Carlsson, and Stefano Ermon. Evaluating the disentanglement of deep generative models through manifold topology. *preprint arXiv:2006.03680*, 2020.

# 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 $k-$simplices, $k \geq 0$, 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 $1-$simplices is a graph.

Denote via $C_k(S)$ the vector space over the field $\mathbb{Z}/2\mathbb{Z}$ whose basis elements are $k-$simplices 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 $k-$th **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.

In applications the simplicial complexes are often built via consequential adding of simplices one after another in increasing order of some numerical characteristics. Mathematically this corresponds to the filtration on the simplicial complex. It is defined as a family of simplicial complexes $S_\alpha$, indexed by a finite set of real numbers, with nested collections of simplices: for $\alpha_1 < \alpha_2$ all simplices of $S_{\alpha_1}$ are also in $S_{\alpha_2}$.

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 $S_{\alpha_i}$ is described by the principal persistent homology theorem (Chazal & Michel, 2017; Barannikov, 1994), according to which for each dimension there exists a choice of a set of basic topological features across all nested simplicial complexes $S_\alpha$ 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 appearence, or "birth" time, and disappearence, or "death" time, of all these basic topological features.

## A.1 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. 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).

Therefore 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 be as close as possible to isomorphisms, in order that the appoximations to manifolds $M_{\mathcal{P}}$ and $M_{\tilde{\mathcal{P}}}$ have similar geometrical and topological features located at the same place.

It follows from the exact sequence from Theorem 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.

# B Properties of *R-Cross-Barcode*$_*(P, \tilde{P})$.

The proof of the exact homology sequence from Theorem 1 follows from the following proposition.

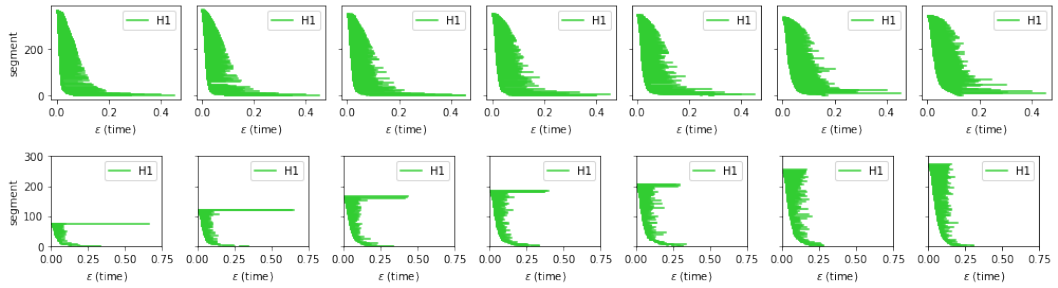



Figure 11: R-Cross-Barcodes for the "clusters" experiments. Top: R-Cross-Barcode$(P, \tilde{P})$, Bottom: R-Cross-Barcode$(\tilde{P}, P)$. $P$ - is the point cloud having one cluster, $\tilde{P}$ - 2, 3, 4, 5, 6, 10, 12 clusters.

**Proposition 2.** *The embedding of weighted graphs $\mathcal{G}^{\min(w,\tilde{w})} \to \hat{\mathcal{G}}(w, \tilde{w}))$ gives the exact sequence of chain complexes*

$$0 \to R_\alpha(\mathcal{G}^{\min(w,\tilde{w})}) \to R_\alpha(\hat{\mathcal{G}}_0(w, \tilde{w})) \to R'_\alpha(\mathcal{G}^w) \to 0. \tag{5}$$

*where $R_\alpha(\hat{\mathcal{G}}_0(w, \tilde{w}))$ has the same homology as $R_\alpha(\hat{\mathcal{G}}(w, \tilde{w}))$ and $R'_\alpha(\mathcal{G}^w)$ has the same homology as $R_\alpha(\mathcal{G}^w)$ shifted by $+1$.*

The proof follows from the straightforward construction of a homotopy to identity map.

*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 and can not use the information provided by one-to-one correspondence between points. It uses instead the proximity information inferred from the pairwise distances between points from different clouds lying in the same ambient space. Geometry score is based on comparison of standard barcodes for each cloud and for example does not detect any difference when similar topological features are located geometrically in very distant places of the two clouds.

## C 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 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 comparison of representation from different layers is questionable.

## D Details on experiments with point clouds

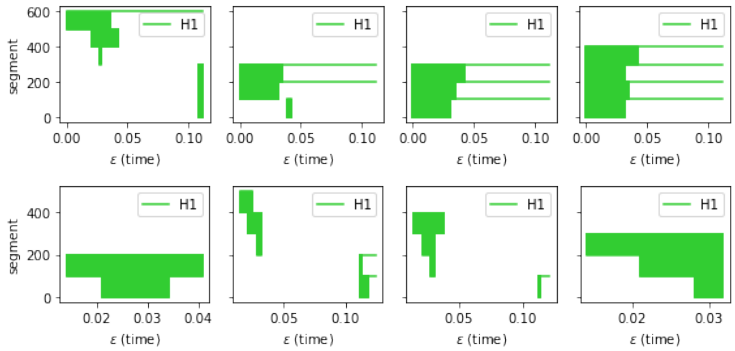
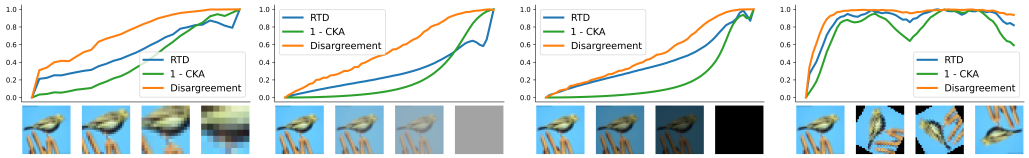



Figure 12: R-Cross-Barcodes for the "rings" experiments. Top: R-Cross-Barcode$(P, \tilde{P})$, Bottom: R-Cross-Barcode$(\tilde{P}, P)$. $P$ - is the point cloud having 5 rings, $\tilde{P}$ - 4, 3, 2, 1 rings.

| Metrics to correlate | | Noise | Gaussian Blur | Grayscale | Hue |
|---|---|---|---|---|---|
| Disagreement | RTD | **0.968** | **0.982** | **0.998** | **0.982** |
| | 1−CKA | 0.928 | 0.915 | 0.937 | 0.916 |
| Error rate | RTD | **0.984** | 0.974 | 0.875 | 0.975 |
| | 1−CKA | 0.967 | **0.999** | **0.978** | **0.987** |

Table 6: 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 13: Analysis of ResNet-20 representations under different data distribution shifts. The dynamics of scaled metrics with monotonic application of various types of image transformations.

| Metrics to correlate | | Zoom | Brightness | Contrast | Rotation |
|---|---|---|---|---|---|
| Disagreement | RTD | **0.956** | **0.978** | **0.939** | **0.962** |
| | 1−CKA | 0.891 | 0.857 | 0.853 | 0.861 |
| Error rate | RTD | 0.945 | 0.922 | 0.938 | 0.945 |
| | 1−CKA | **0.995** | **0.998** | **0.998** | **0.984** |

Table 7: Analysis of ResNet-20 representations under different data distribution shifts. The correlation of RTD and 1−CKA with Disagreement and Error rate.

| | RTD | 1−CKA |
|---|---|---|
| Disagreement | **0.99** | 0.93 |
| Error rate | 0.91 | **0.99** |

(a) CIFAR-100

| | RTD | 1−CKA |
|---|---|---|
| Disagreement | **0.92** | 0.89 |
| Error rate | 0.60 | **0.73** |

(b) CIFAR-10

Table 8: 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<br>50-90%: 0.1-0.001 (linear)<br>>90%: 0.001 | |
| Batch size | 128 | |

Table 9: 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<br>50-90%: 0.1-0.001 (linear)<br>>90%: 0.001 |
| Batch size | 128 | |

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