Mapping the Multiverse of Latent Representations

Jeremy Wayland [1 2] **Corinna Coupette** [† 3 4] **Bastian Rieck** [† 1 2]

Abstract

Echoing recent calls to counter reliability and robustness concerns in machine learning via *multiverse analysis*, we present PRESTO, a principled framework for *mapping the multiverse* of machine-learning models that rely on *latent representations*. Although such models enjoy widespread adoption, the variability in their embeddings remains poorly understood, resulting in unnecessary complexity and untrustworthy representations. Our framework uses *persistent homology* to characterize the latent spaces arising from different combinations of diverse machine-learning methods, (hyper)parameter configurations, and datasets, allowing us to measure their pairwise *(dis)similarity* and statistically reason about their *distributions*. As we demonstrate both theoretically and empirically, our pipeline preserves desirable properties of collections of latent representations, and it can be leveraged to perform sensitivity analysis, detect anomalous embeddings, or efficiently and effectively navigate hyperparameter search spaces.

1. Introduction

Our ability to design and deploy new machine-learning models has far outpaced our understanding of their inner workings. The real-world successes of Variational Auto-Encoders (VAEs), Large Language Models (LLMs), and Graph Neural Networks (GNNs) notwithstanding, our benchmark-driven engineering approaches often come at the cost of an *inability to make formal predictions* about the capacity of a specific model to perform a specific task on a specific dataset. Thus, when observing a particular performance result, we are unsure to which extent it is impacted by (i) intrinsic problems with the data, such as poor data quality or data leakage, (ii) intrinsic problems with the model architecture, such as insufficient learning capacity, (iii) misfit between the data and the model architecture, (iv) unsuitable (parameter) choices for the training process, or (v) an under-explored hyperparameter landscape. These uncertainties contribute to a looming *reproducibility crisis* in machine learning that threatens to impede fundamental progress and reduce real-life impact (Gundersen et al., 2022; Haibe-Kains et al., 2020; Kapoor & Narayanan, 2023; McDermott et al., 2021).

Into the Multiverse. Ensuring robust, reliable, and reproducible results in machine-learning applications requires new conceptual frameworks and tools. As a starting point, we must acknowledge that all data work involves *many different choices* that may support *many different conclusions* (Simonsohn et al., 2020). In machine learning, such choices regularly include the model architecture and its hyperparameters (Feurer & Hutter, 2019), the dataset and its preprocessing (Muller & Strohmayer, 2022), as well as the technicalities of model training and evaluation (Sivaprasad et al., 2020). To rigorously assess machine-learning models, then, we should explicitly *embrace the variation* resulting from all reasonable combinations of reasonable choices, rather than keep individual choices hidden or implicit. This is the essence of *multiverse analysis* (Steegen et al., 2016), which was originally proposed to mitigate the perceived replication crisis in psychology (Simmons et al., 2011).

Representation Matters. In multiverse analysis, each set of mutually compatible choices (according to some specification) gives rise to a different analytical *universe*, and we assess the results of all universes in the same *multiverse* collectively to derive our conclusions. While the early applications of multiverse approaches in machine learning (Bell et al., 2022; Simson et al., 2023) have focused on variability in performance (*performance variability*), which afflicts all machine-learning models, the influential class of *latent-space models* (including VAEs, LLMs, and GNNs) also exhibits variability in latent representations (*representational variability*). In many cases, even relatively small (hyper)parameter changes can radically alter the embedding structure of latent-space models. As an example, consider Figure 1, which depicts

[†]These authors jointly directed this work. [1]Helmholtz Munich [2]Technical University of Munich [3]KTH Royal Institute of Technology [4]Max Planck Institute for Informatics. Correspondence to: Jeremy Wayland <jeremy.wayland@tum.de>, Corinna Coupette <coupette@mpi-inf.mpg.de>, Bastian Rieck <bastian.rieck@tum.de>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

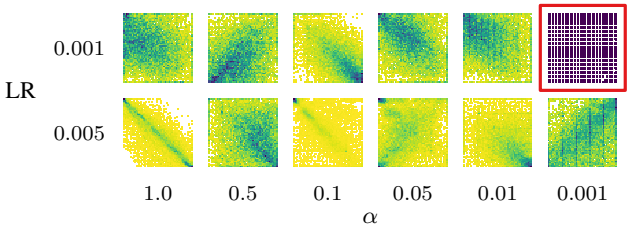


Figure 1. **Entangled disentanglement.** The embedding spaces of DAEs (Cha & Thiyagalingam, 2023) vary widely when we change the learning rate LR and the batch-normalization hyperparameter α of the model, and the latent structure of the XYC dataset (shapes varying in 2D coordinates and color) is properly disentangled only with the right parameter choices (□). PRESTO can *topologically* assess the (hyper)parameter sensitivity of latent-space models.

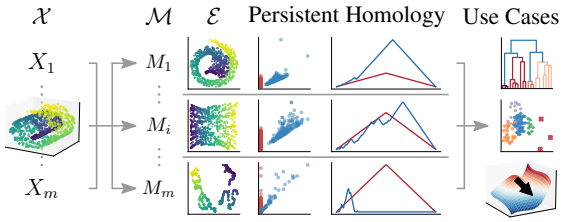


Figure 2. **The PRESTO pipeline.** For each model M_i in our multiverse $\mathcal{M}$, PRESTO computes the persistent homology associated with the embedding of a dataset X_i generated by M_i, yielding a set of embeddings $\mathcal{E}$. Thus enabled to compare the latent spaces of different models via the landscape distance of their persistence landscapes, with PRESTO, we can cluster, compress, detect outliers in, and analyze the sensitivity of (hyper)parameter configurations.

two-dimensional representations of the XYC dataset as generated by Disentangling Auto-Encoders (DAEs) with different learning rates and batch-normalization parameters. Although DAEs were recently designed by Cha & Thiyagalingam (2023) precisely to learn disentangled representations, and the XYC dataset was introduced to highlight the power of DAEs, the latent structure of the XYC dataset is disentangled only with the right parameter choices (□). More generally, representational variability in latent-space models remains poorly understood.

Variability Matters. While the *performance* variability of latent-space models is clearly connected to their *reliability*, the *representational* variability of such models is directly linked to their *interpretability* and *robustness*. First, if models differing only in their (hyper)parameters yield similar performance based on very dissimilar latent spaces, we cannot use these latent spaces to understand the models (impairing their interpretability). Second, if a small change in the (hyper)parameter or training-data configuration induces a large change in the latent-space structure of a model, the model associated with the original configuration may not capture the essence of the task, even if the model appears to be competitive when assessed based on performance-driven evaluation (indicating a lack of structural robustness). Therefore, representational variability not only complements performance variability in the analysis of latent-space models, but *ceteris paribus*, models with lower representational variability should be preferred over models with higher representational variability. Hence, understanding representational variability in latent-space models is crucial to ensure their overall alignment with responsible-machine-learning goals.

Our Contributions. Motivated by the need for responsible latent-space models, and encouraged by the promise of topological approaches to representational variability (Barannikov et al., 2021; 2022; Zhou et al., 2021), in this work, we use topology to map the multiverse of machine-learning models that rely on latent representations. In particular, we ask two guiding questions.

(Q1) ***Exploring* representational variability.** *How do the latent representations of machine-learning models vary across different choices of model architectures, (hyper)parameters, and datasets?*

(Q2) ***Exploiting* representational variability.** *How can we use representational variability to efficiently train and select robust and reliable machine-learning models?*

To address these questions, we make five contributions.

(C1) We introduce PRESTO, a **topological multiverse framework** to describe and *directly* compare both *individual* latent spaces and *collections* of latent spaces, as summarized in Figure 2.

(C2) We capture **essential features of latent spaces** via persistence diagrams and landscapes, allowing us to measure the pairwise *(dis)similarity* of embeddings and statistically reason about their *distributions*.

(C3) We prove **theoretical stability guarantees** for topological representations of latent spaces under projection.

(C4) We develop **scalable practical tools** to measure *representational* (hyper)parameter sensitivity, identify anomalous embeddings, compress (hyper)parameter search spaces, and accelerate model selection.

(C5) We demonstrate the utility of our tools via **extensive experiments** in numerous latent-space multiverses.

Our work improves our understanding of representational variability in latent-space models, and it offers a *structure-driven* alternative to existing performance-driven approaches in the responsible-machine-learning toolbox.

Structure. Having given some background on persistent homology in Section 2, we introduce PRESTO, our multiverse framework for exploring and exploiting representational variability in latent-space models in Section 3. After discussing related work in Section 4, we gauge the practical

utility of our framework through extensive experiments in Section 5, before concluding with a discussion in Section 6. Extensive supplements are provided in Appendices A to E.

2. Background

This section briefly introduces *persistent homology* (PH), the machinery for capturing essential features of data that forms the basis of our framework. Additional definitions and theorems can be found in Appendix A, while Appendix D contains background information on *latent-space models*. Persistent homology (Barannikov, 1994; Edelsbrunner & Harer, 2010) is a framework used to analyze topological characteristics of data at multiple spatial scales. It systematically quantifies the evolution of both *topological* and *geometric* features, as tracked by a *filtration*, i.e., a consistent ordering of the elements of a space. Filtrations typically arise by approximating data with *simplicial complexes*, i.e., generalized graphs, using metrics like the L^2 distance; we will work with computationally efficient α-complexes (Edelsbrunner et al., 1983). Thus, given a topological space X and a filtration $\{X_t\}_{t\in\mathbb{R}}$, with each X_t being a subcomplex of X, persistent homology computes a sequence of homology groups $\{H_h(X_t)\}_{h\geq 0}$ for each t. These groups capture h-dimensional topological features such as *connected components*, *cycles*, or *voids* of X_t at multiple resolutions, and they are *invariant* to spatial transformations like translation, rotation, and uniform scaling (when working with normalized distances). Moreover, persistent homology varies continuously under continuous transformations of the space.

Persistent-homology computations are typically summarized using *persistence diagrams*, which provide a condensed representation for tracking topological features across multiple scales. Formally, a persistence diagram $\mathrm{dgm} = \{(b_i, d_i)\}_i$ is a multiset of intervals, where b_i represents the 'birth time' and d_i represents the 'death time' of a given h-dimensional topological feature, i.e., $b_i, d_i \in \mathbb{R} \cup \{\infty\}$ with $b_i \leq d_i$. As the space Dgm of persistence diagrams is cumbersome to work with and does not afford efficiently-computable metrics, there are alternative representations of persistent homology, such as *persistence landscapes* (PL), which map persistence diagrams into a Banach space by transforming them into piecewise-linear functions $L_h\colon \mathrm{Dgm} \to \mathbb{R}$ (Bubenik, 2015). This transformation allows us to compare topological descriptors using computationally efficient metrics, and its calculation requires neither discretization nor additional parameter choices. For our work, the *stability* of persistent homology and persistence landscapes under perturbations and transformations of the data is particularly relevant: These descriptors are well-behaved under structure-preserving embeddings (Krishnamoorthy & May, 2023; Sheehy, 2014), and they capture *both* geometric and topological properties of data (Bubenik et al., 2020). Thus, we select persistent homology and persistence landscapes as our lens for assessing representational variation.

3. Topological Multiverse Analysis

Having established the necessary background, we now introduce PRESTO, our multiverse framework for exploring and exploiting representational variability in latent-space models via persistent homology. To define a multiverse of latent representations, we distinguish three categories of choices, namely, (1) *algorithmic choices* (i.e., model architecture and hyperparameters), (2) *implementation choices* (such as optimizer, learning rate, number of epochs, and random seeds), and (3) *data choices* (i.e., dataset and preprocessing). The mutually compatible options in each category give rise to three sets of valid choices, i.e., *algorithmic choices* $\alpha \in \mathcal{A}$, *implementation choices* $\iota \in \mathcal{I}$, and *data choices* $\delta \in \mathcal{D}$. We will also think of $\mathcal{A}$, $\mathcal{I}$, and $\mathcal{D}$ as sets of *vectors*, such that we can refer to $(\alpha, \iota, \delta) \in \mathcal{A} \times \mathcal{I} \times \mathcal{D}$ by its parameter vector $\theta := (\alpha, \iota, \delta)'$ of cardinality $c := |\theta|$. Thus, we arrive at the notion of a *latent-space multiverse*.

Definition 3.1 (Latent-Space Multiverse)**.** Given *algorithmic choices* $\mathcal{A}$, *implementation choices* $\mathcal{I}$, and *data choices* $\mathcal{D}$, a **latent-space multiverse** $\mathcal{M}$ is a subset of $\mathcal{A} \times \mathcal{I} \times \mathcal{D}$. Each element $\theta \in \mathcal{M}$ is a *universe* with an associated *model* $M^\theta\colon \mathbb{R}^* \to \mathbb{R}^d$, where d is the desired embedding dimension (part of θ), $*$ denotes a flexible input dimension, and we drop $^\theta$ for conciseness.

In this work, we are interested in *finite* latent-space multiverses, generated by *discrete* subsets of $\mathcal{A}$, $\mathcal{I}$, and $\mathcal{D}$.

3.1. PRESTO Pipeline

Given a finite latent-space multiverse $\mathcal{M} \subseteq \mathcal{A} \times \mathcal{I} \times \mathcal{D}$, to *explore* representational variability, we would like to compare the topologies of individual latent spaces and statistically assess their distribution in $\mathcal{M}$, which will also prove useful for *exploiting* representational variability in practical applications. Unfortunately, working with the topologies of high-dimensional latent spaces directly, e.g., by comparing them via the (normalized) bottleneck distance of persistence diagrams derived from Vietoris–Rips complexes, is *computationally prohibitive*. Furthermore, as persistence diagrams do not live in a *Banach space*, we cannot reason about their distributions. Hence, we instead propose the following *scalable* framework for topological multiverse analysis, which we call PRESTO (PRojected Embedding Similarity via Topological Overlays).[1] For each

[1]Although we advocate for the pipeline presented below, its individual steps can be easily modified, e.g., to use other topological descriptors. See Appendix B.2 for a detailed discussion.

model $M \in \mathcal{M}$ and a dataset X (which can differ from the training data of M), PRESTO performs four steps.[2]

(S1) **Embed data.** Compute the d-dimensional *embedding* $E := M(X)$. *Optionally,* (a) approximate the *diameter* of E, and (b) *normalize* E by this approximation.

(S2) **Project embeddings.** Project E down to $k \ll d$ dimensions. This can be done either *deterministically*, e.g., via Principal Component Analysis (PCA), or by generating a set of π *random projections* $\mathcal{P}$.[3]

(S3) **Construct persistence diagrams.** Calculate the *persistence diagram* dgm (or dgm_i for each projection $E_i \in \mathcal{P}$) based on its h-dimensional α-*complex*.

(S4) **Compute persistence landscapes.** Vectorize the persistence diagram dgm into a *persistence landscape* $L(M)$ (or dgm_i into L_i with $L(M) := \sum_{i \in [\pi]} L_i / \pi$).

Here, S1a and S1b replace exact normalization, and S2 and S3 replace persistence-diagram computation based on Vietoris-Rips complexes constructed in (potentially) high dimensions, all of which are computationally costly. S4 guarantees that we operate in a Banach space, and averaging here increases robustness if we work with random projections. Since PRESTO is based on persistent homology, it immediately benefits from PH's well-studied properties (see Section 2). Finally, our pipeline is *data-agnostic*, i.e., we can use any dataset X to study variability in $\mathcal{M}$ without requiring access to the training data—and we can even use PRESTO to study variability in *datasets* (see Section 5).

3.2. PRESTO Primitives

By performing S1 to S4 for each $M \in \mathcal{M}$, we obtain a set $\mathcal{L}$ of persistence landscapes, which allows us to achieve two fundamental tasks. First, we can measure the distance between two latent spaces via the PRESTO *distance* (PD).[4]

Definition 3.2 (PRESTO Distance [PD])**.** Given persistence landscapes $L(M_i)$ and $L(M_j)$, the **PRESTO^p_h distance** up to topological dimension h between M_i and M_j is

$$\text{PD}^p_h(M_i, M_j) := \sum_{x=0}^{h} d_{L^p}(L_x(M_i), L_x(M_j)) \ , \quad (1)$$

where d_{L^p} is the landscape distance based on the L^p norm.

Second, we can assess the variance in a set of latent spaces via the PRESTO *variance* (PV).

Definition 3.3 (PRESTO Variance [PV])**.** Given a set $\mathcal{L}$ of persistence landscapes of cardinality $N := |\mathcal{L}|$, the **PRESTO^p_h variance** up to topological dimension h of $\mathcal{L}$ is

$$\text{PV}^p_h(\mathcal{L}) := \frac{1}{N} \sum_{x=0}^{h} \sum_{L \in \mathcal{L}^x} \left(\|L\|_p - \mu_{\mathcal{L}^x} \right)^2 \ , \quad (2)$$

where $\|L\|_p$ is the L^p-based landscape norm, $\mathcal{L}^x$ denotes the landscape parts associated with x-dimensional topological features, and $\mu_{\mathcal{L}^x}$ is the mean of landscape norms in $\mathcal{L}^x$.

Although the modifications to an exact pipeline made by our PRESTO framework induce some changes in our representations, they retain the essential features of our original latent spaces. As a result, the error we introduce into our measurements in Eqs. (1) and (2) is bounded both theoretically and empirically, as we show in Sections 3.4 and 5.

3.3. PRESTO Applications

As we demonstrate in our experiments (Section 5), our multiverse framework and PRESTO primitives are useful for exploring and exploiting the representational variability of latent-space models in several different settings. In particular, they can help us (1) evaluate (hyper)parameter sensitivity, (2) detect anomalous embeddings, and (3) cluster and compress (hyper)parameter search spaces.

Sensitivity Analysis. Following the reasoning sketched in Section 1, choices amplifying representational variability are both analytically interesting and practically problematic. Thus motivated to study (hyper)parameter sensitivity in latent-space multiverses $\mathcal{M}$, our goal here is to quantify the structural variation in the embedding space when introducing controlled variation in $\theta \in \mathcal{M}$. We seek to assess the *local* and *global* sensitivity in $\mathcal{M}$, as well as the sensitivity at *individual* coordinates θ in $\mathcal{M}$. To this end, we introduce our PRESTO *sensitivity* scores (PS).

Definition 3.4 (PRESTO Sensitivity [PS])**.** Given a multiverse $\mathcal{M}$, fix a model dimension i, and define an equivalence relation $\sim_i$ such that $\theta' \sim_i \theta'' \Leftrightarrow \theta'_j = \theta''_j$ for all $\theta', \theta'' \in \mathcal{M}$ and $j \neq i$, yielding q_i equivalence classes $\mathcal{Q}_i$.

The ***individual*** **PRESTO^p_h sensitivity** of equivalence class $Q \in \mathcal{Q}_i$ in $\mathcal{M}$ is

$$\text{PS}^p_h(Q \mid \mathcal{M}) := \sqrt{\text{PV}^p_h(\mathcal{L}[Q])} \ , \quad (3)$$

where $\mathcal{L}[Q] \subset \mathcal{L}$ is the set of landscapes associated with models in equivalence class Q. Aggregating over all equivalence classes in $\mathcal{Q}_i$, we obtain the ***local*** **PRESTO^p_h sensitivity** of $\mathcal{M}$ in model dimension i as

$$\text{PS}^p_h(\mathcal{M} \mid i) := \sqrt{\frac{1}{q_i} \sum_{Q \in \mathcal{Q}_i} \text{PV}^p_h(\mathcal{L}[Q])} \ . \quad (4)$$

[2] While we keep the dataset X fixed in our exposition for notational simplicity, as we illustrate experimentally in Section 5, we can also use our multiverse approach to assess how changing X affects our reconstruction of a (set of) latent space(s).

[3] While the deterministic approach is particularly suited for comparing *different* latent spaces, leveraging randomness allows us to study variability within *individual* latent spaces.

[4] In the following, to avoid ambiguity in our PRESTO-related definitions, we use a superscript p to denote the choice of L^p norm, and a subscript h to denote the maximum dimension of topological features considered. Otherwise, we drop these superscripts and subscripts for simplicity when they are not decisive.

Finally, aggregating over all $c = |\theta|$ dimensions of models in $\mathcal{M}$ yields the ***global* PRESTO^p_h sensitivity** of $\mathcal{M}$, i.e.,

$$\mathrm{PS}^p_h(\mathcal{M}) \coloneqq \sqrt{\frac{1}{c} \sum_{i \in [c]} \frac{1}{q_i} \sum_{Q \in \mathcal{Q}_i} \mathrm{PV}^p_h(\mathcal{L}[Q])} \ . \tag{5}$$

Note that when $\mathcal{M}$ varies only in one dimension, the individual, local, and global PRESTO sensitivities are identical, such that we can simply speak of the PRESTO *sensitivity*.

Outlier Detection. Since models with anomalous latent spaces are, by definition, not robust, we should understand for which sets of choices they occur and avoid working with them in practice. To detect such anomalous latent spaces in a multiverse $\mathcal{M}$ with associated landscapes $\mathcal{L}$, we can exploit their Banach-space structure, which allows us to use the PV (Definition 3.3), along with standard statistical approaches, to identify landscapes with anomalous norms.

Clustering and Compression. To identify interesting structure in a collection of latent spaces (arising, e.g., from a grid search), we can *cluster* the collection, represented by a multiverse $\mathcal{M}$ with associated landscapes $\mathcal{L}$, based on the PRESTO distance (Definition 3.2), using *any clustering method* based on pairwise distances. Reducing the costs of exhaustive (hyper)parameter searches, however, requires us to lower the number of configurations considered in detail. As we demonstrate experimentally, if two latent spaces are topologically close in our *target* setting (i.e., a search space we would like to *avoid exploring exhaustively*), they may also be close in a *proxy* setting (i.e., a search space we *will explore (or have explored) exhaustively*), such as when training on a related dataset. This permits us to reuse knowledge generated from proxy settings for our target setting, motivating the task of search-space compression. Given results from a proxy setting P, to *compress* the search space in our target setting $\mathcal{M}$, we define a threshold ϵ and select *representatives* $\mathcal{R} \subseteq \mathcal{M}$ such that for each $M_i \in \mathcal{M}$, there exists a representative $M_j \in \mathcal{R}$ with $\mathrm{PD}_P(M_i, M_j) \leq \epsilon$, where PD_P denotes the PRESTO distance in proxy setting P. Appendix C.2 provides more details on how to pick suitable representatives in practice.

3.4. PRESTO Stability

To ensure scalability, PRESTO computes topological descriptors on low-dimensional projections of embeddings, rather than working in a high-dimensional latent space. Therefore, we would like to ascertain that the distortion introduced by these projections remains bounded. To achieve this, we require the notion of a *multiverse metric space* (MMS). We will work with the L^2 landscape norm and consider topological features up to dimension 2 (i.e., $p = h = 2$), dropping the superscript p and the subscript h for notational conciseness, and deferring all proofs to Appendix A.

Definition 3.5 (Multiverse Metric Space $\mathfrak{M}$ [MMS])**.** For a multiverse $\mathcal{M}$ with associated embeddings $\mathcal{E}$, we define the topological distance of embeddings in $\mathcal{E}$ as

$$d_T(E_i, E_j) \coloneqq d(\mathrm{dgm}(E_i), \mathrm{dgm}(E_j)) \ , \tag{6}$$

where d can be any distance between persistence representations (e.g., diagrams or landscapes). A **multiverse metric space** is the tuple $(\mathcal{E}, d_T) =: \mathfrak{M}$.

When working with k-dimensional embedding *projections*, we operate in a *projected multiverse metric space* (PMMS).

Definition 3.6 (Projected Multiverse Metric Space $\mathfrak{M}^k$ [PMMS])**.** Given an MMS $\mathfrak{M} = (\mathcal{E}, d_T)$, fix the projection dimension $k \in \mathbb{N}$, s.t. $k \leq d_i$ for $E_i \in \mathcal{E}$. For a projector $f \colon \mathbb{R}^* \to \mathbb{R}^k$, let $\mathcal{P} \coloneqq \{f(E) \mid E \in \mathcal{E}\}$ be the set of *projected* embeddings. Using d_T from Eq. (6), a **projected multiverse metric space** is defined as $\mathfrak{M}^k \coloneqq (\mathcal{P}, d_T)$.

Relating an MMS $\mathfrak{M}$ to its k-dimensional counterpart $\mathfrak{M}^k$, we arrive at the notion of *topological loss*, i.e., the decrease in topological fidelity due to our projection.

Definition 3.7 (Topological Loss)**.** Given an MMS $\mathfrak{M}$, a projector $f \colon \mathbb{R}^* \to \mathbb{R}^k$, and an associated PMMS $\mathfrak{M}^k$, with $P_i \coloneqq f(E_i)$, the **topological loss** ℓ^k of $\mathfrak{M}^k$ is the maximum distance between elements of $\mathcal{E}$ and $\mathcal{P}$ measured by d_T, i.e.,

$$\ell^k \coloneqq \max_{E_i \in \mathcal{E}} d_T(E_i, P_i) \ . \tag{7}$$

This topological loss bounds the pairwise-distance perturbation of our metric space under projection.

Theorem 3.8 (Metric-Space Preservation under Projection)**.** *Given an MMS $\mathfrak{M}$ and an associated PMMS $\mathfrak{M}^k$ with topological loss ℓ^k, we can bound the pairwise-distance perturbation under projection as $\mathfrak{M}^k[i,j] \leq \mathfrak{M}[i,j] + 2\ell^k$.*

Consequently, as the topological loss *increases*, our precision in distinguishing embeddings *decreases* in a controlled manner, and we can further bound the PRESTO variance under projection as follows.

Theorem 3.9 (PRESTO Variance under Projection)**.** *Consider an MMS $\mathfrak{M} = (\mathcal{E}, d_T)$ with the landscape distance $d_T(E_i, E_j) \coloneqq d(L(E_i), L(E_j))$ and associated persistence landscapes $\mathcal{L}_{\mathfrak{M}}$. Further, let $\mathfrak{M}^k$ be a PMMS with a topological loss ℓ^k. Then we can bound the maximal change in* any *persistence landscape norm as*

$$\|L(E_i)\| - \ell^k \leq \|L(P_i)\| \leq \|L(E_i)\| + \ell^k \ . \tag{8}$$

Given the PRESTO *variance of $\mathfrak{M}$, PV($\mathcal{L}_{\mathfrak{M}}$), we can bound the* PRESTO *variance of $\mathfrak{M}^k$, i.e., PV($\mathcal{L}_{\mathfrak{M}^k}$), as*

$$|PV(\mathcal{L}_{\mathfrak{M}}) - PV(\mathcal{L}_{\mathfrak{M}^k})| \leq \frac{4\ell^k}{N}\sigma_i + \frac{\left(2\ell^k\right)^2}{N} \ , \tag{9}$$

where $\sigma_i \coloneqq \sum_{i=1}^{N} (\|L(E_i)\| - \mu_{\mathcal{E}})$.

As a result, PRESTO is stable as long as we can control the approximation error induced by the choice of projector function. To this end, Appendix A provides explicit bounds for several common classes of projection functions.

3.5. PRESTO Complexity

While scalability is a common concern in computational topology, our framework is specifically designed for scalability. We present a detailed complexity analysis in Appendix C.1.1, showing that, overall, PRESTO's computations are approximately linear in the number of samples in X. In Appendices C.1.2 and C.1.3, we also validate PRESTO's time complexity empirically, demonstrating that PRESTO distances can be computed faster than related (dis)similarity measures.

4. Related Work

PRESTO connects two strands of literature, i.e., *topological approaches* and *multiverse approaches* in machine learning. Since our framework additionally draws on numerous other fields, we provide an extended discussion in Appendix E.

Topological Approaches in Machine Learning. Topological approaches have been used to analyze and control representational variation, leading to regularization terms that preserve topological characteristics (Moor et al., 2020; Trofimov et al., 2023; Waibel et al., 2022), scores for assessing disentanglement (Zhou et al., 2021) or quality (Rieck & Leitte, 2015; 2016), as well as methods for learning disentangled representations (Balabin et al., 2023), studying neural networks (Klabunde et al., 2023; Kostenok et al., 2023; Purvine et al., 2023; Rieck et al., 2019), measuring generative quality (Kim et al., 2023), and enabling zero-shot training (Moschella et al., 2023). Drawing on topological concepts to analyze differences between latent spaces, the *manifold topology divergence* and the *representation topology divergence* (RTD) (Barannikov et al., 2021; 2022) are closest to our work. However, these methods focus on *pairwise* comparisons of *aligned* data, exhibit unfavorable scaling behavior, and do not enjoy theoretical fidelity guarantees. By contrast, as the first method for studying representational variability that leverages persistence landscapes, PRESTO enables a *multiverse* analysis in terms of models, (hyper)parameters, and datasets, readily handles *unaligned* embeddings, and quantifies all results in terms of *distance metrics*, thus improving their interpretability.

Multiverse Approaches in Machine Learning. Multiverse analysis (Steegen et al., 2016) aims to reduce arbitrariness and increase transparency in data analysis via the joint consideration of multiple reasonable analytical scenarios. Precursors of a multiverse perspective in machine learning have assessed hyperparameter choices (Kumar & Poole, 2020; Sivaprasad et al., 2020; Smith, 2018; Zhang et al., 2019), studied the causal structure of latent representations (Leeb et al., 2022), or compared different models (Diedrichsen & Kriegeskorte, 2017; Diedrichsen et al., 2020; Vittadello & Stumpf, 2021). By contrast, PRESTO provides a unified framework for the structural analysis of latent spaces across models, (hyper)parameters and datasets. While Bell et al. (2022) consider a model multiverse, their work differs in its goals and its methods. In particular, they focus on the *performance-driven exploration* of *continuous* search spaces via tools from *statistics*, while we pursue the *structure-driven characterization* of *discrete* search spaces using tools from *topology*.

5. Experiments

In our experiments, we ask three questions:

(Q0) **PRESTO's distinctive properties.**
How does PRESTO *relate to existing measures of representational (dis)similarity and variability?*

(Q1) **PRESTO for *exploring* representational variability.**
How can PRESTO *help us understand representational variability across different choices of model architectures, (hyper)parameters, and datasets?*

(Q2) **PRESTO for *exploiting* representational variability.**
How can PRESTO *help us to efficiently train and select robust and reliable machine-learning models?*

To address our guiding questions, we generate multiverses for two types of generative models, i.e., *variational autoencoders* as generators of *images*, and *transformers* as generators of *natural language*. We are particularly interested in the impact of *algorithmic* choices $\mathcal{A}$, *implementation* choices $\mathcal{I}$, and *data* choices $\mathcal{D}$ on the generated representations. Further experiments (including on dimensionality reduction), a multiverse analysis of the choices involved in the PRESTO pipeline, and more details on all results reported here can be found in Appendix B.[5]

VAE Multiverses. In brief, our VAE experiments study representational variation in the following dimensions.

- $\mathcal{A}$. We consider three VAE architectures: (1) β-VAE (Higgins et al., 2017), (2) INFOVAE (Zhao et al., 2019), and (3) WAE (Tolstikhin et al., 2018). For each architecture, we investigate the interplay between hyperparameter choices and latent-space structure when navigating trade-offs between reconstruction bias and KL-divergence weight in conjunction with loss variations and kernel parameters.
- $\mathcal{I}$. Specifically for β-VAE, in Appendix B.1.2, we explore

[5]Code: https://github.com/aidos-lab/Presto. Reproducibility package: https://doi.org/10.5281/zenodo.11355446. Unless otherwise noted, all experiments use normalization.

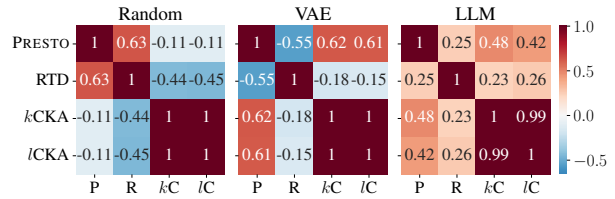


Figure 3. **Comparing PRESTO distances with other measures.** We show the Pearson correlations between PRESTO, RTD, kCKA, and lCKA, on random data (left), VAE embeddings (center), and LLM embeddings (right). PRESTO captures representational variation differently from existing methods.

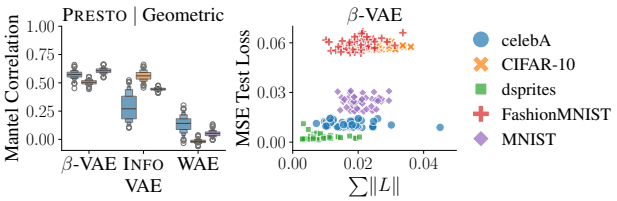


Figure 4. **Comparing PRESTO, latent-space geometry, and model performance.** We show the distribution of correlations between PRESTO distances and geometric latent-space distances in the VAE multiverse, estimating geometric distances based on the Pearson distance between Euclidean metric spaces of aligned random samples of size 512 over 256 random draws (left), as well as the relationship between landscape norms and model performance for β-VAE (right). PRESTO captures geometric similarity between latent spaces and is orthogonal to performance.

the relation between β and five implementation choices: (1) batch size b, (2) hidden dimensions h, (3) learning rate l, (4) sample size s, (5) and train-test split t.

$\mathcal{D}$. We train on five datasets: (1) *celebA*, (2) *CIFAR-10*, (3) *dsprites*, (4) *FashionMNIST*, and (5) *MNIST*.

For a detailed description, see Appendices B.1.1 and B.1.2.

Transformer Multiverses. Based on access to pretrained language models only, we focus our transformer experiments on algorithmic and data choices. In Appendices B.2.2 to B.2.4, we additionally use transformers to study the impact of implementation choices in the PRESTO pipeline.

$\mathcal{A}$. We consider six transformer models: (1) ADA, (2) MISTRAL, (3) DISTILROBERTA, (4) MINILM, (5) MPNET, and (6) QA-DISTILBERT. The first two models are large language models from OpenAI and MistralAI, respectively, whereas the other four models are sentence transformers taken from the *sentence-transformers* library (Reimers & Gurevych, 2019).

$\mathcal{I}$. To analyze the implementation multiverse of PRESTO pipeline choices, we introduce variation in the (1) number of samples s embedded, (2) number of projection components k considered, as well as in (3) embedding-projection method (PCA vs. random projections) and number of random projections π.

$\mathcal{D}$. We probe each trained model by embedding abstracts from four summarization datasets, i.e., (1) *arXiv*, (2) *bbc*, (3) *cnn*, and (4) *patents*, all of which are available via *HuggingFace*.

See Appendix B.2.1 for a detailed description.

5.1. PRESTO's Distinctive Properties

With our experimental setup in place, we turn to our zeroth guiding question: understanding PRESTO's distinctive properties. To begin, we compare PRESTO distances with other measures of representational (dis)similarity on the basic task of *pairwise* comparisons between *aligned* embeddings (a limitation imposed by competitor methods). Summarizing the correlation between PRESTO and RTD, both topology-based *dissimilarity* measures, as well as RBF-kernel and linear Centered Kernel Alignment (kCKA and lCKA), both *similarity* measures, in Figure 3, we observe that there is no consistent relationship between PRESTO distances, RTDs, and CKA scores. This indicates that PRESTO captures variation in latent-space structure differently from existing methods, which appears desirable, given the known limitations of existing approaches (cf. Davari et al., 2023).

To understand what exactly is captured by PRESTO, in the left panel of Figure 4, (see also Appendix B.1.4), we correlate *unnormalized* PRESTO distance matrices for different VAE hyperparameter multiverses with estimates of *geometric* distances between pairs of latent spaces (i.e., Pearson distances between random, aligned metric subspaces), displaying the distribution of correlations over multiple random draws. We expect to see *some* correlation between PRESTO distances and geometric distances because our framework is based on PH, which also captures some geometric properties (Bubenik et al., 2020). In line with this expectation, we see that PRESTO distances are correlated with *geometric* distances, albeit to different extents across models and datasets. In the right panel of Figure 4, we further examine the relationship between landscape norms, the foundation of PRESTO *variances*, and the performance of models in the β-VAE hyperparameter multiverse. We observe that while larger landscape norms are, overall, associated with larger losses, models with similar performance exhibit substantial variability in their landscape norms. This suggests that the variability captured by PRESTO is orthogonal to variability in performance (an impression further corroborated by an extended experiment in Appendix B.1.5), underscoring PRESTO's capacity to complement performance-based metrics, shed light on variability in *similarly-performing* models, and promote *representational stability* as a target in model evaluation.

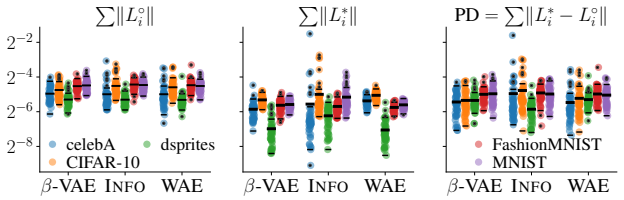


Figure 5. **Landscape-norm distributions in our VAE hyperparameter multiverse.** We show the distribution of landscape norms after initialization (left) and training (center), as well as the distribution of PRESTO distances between the landscape at initialization and the landscape after training (right). Thick lines indicate means, thin black lines indicate interquartile range, and black dots indicate outliers. Training differentially affects landscape norms across models and datasets.

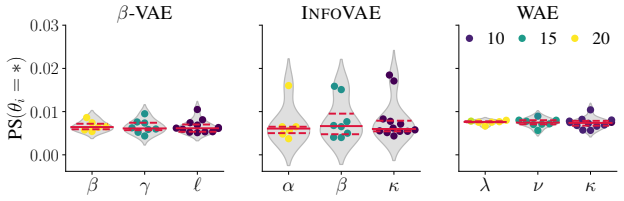


Figure 6. **Distributions of individual PRESTO sensitivity in our VAE hyperparameter multiverse.** We show the distribution of individual PRESTO sensitivity scores for all equivalence classes of models that vary a particular parameter θ_i while keeping the others constant. Marker colors indicate the number of observations per equivalence class, the solid red line marks the median, and dashed red lines indicate interquartile range. INFOVAE exhibits the largest variability in hyperparameter sensitivities.

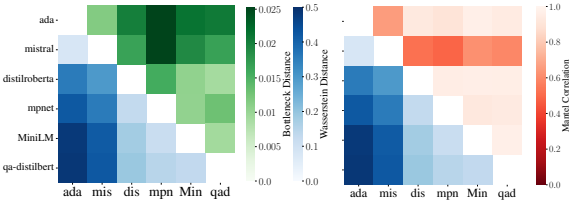


Figure 7. **Comparing MMSs across transformer models.** We show the (dis)similarity between the transformer multiverses associated with each of our models, as measured by the Wasserstein distance (lower triangles), the bottleneck distance (upper triangle left), or the Mantel correlation (upper triangle right). Topological distances provide a more nuanced perspective than permutation-based correlation assessments, and they clearly distinguish the MMSs of large language models from those of smaller models.

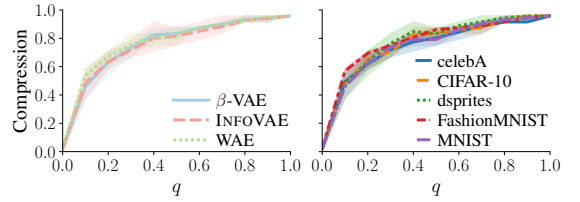


Figure 8. **Compressing the VAE hyperparameter multiverse.** We show the compression of the hyperparameter multiverse split by models (left) and datasets (right) achieved by restricting the hyperparameter search to set-cover representatives which guarantee that each universe has a representative at distance no larger than the qth quantile of the distribution of pairwise distances in the multiverse. With PRESTO, we can halve the size of the hyperparameter search space while ensuring low topological distortion.

5.2. Exploring Representational Variability

Reassured by PRESTO's distinctive properties, we now leverage our framework to explore representational variability in our VAE and transformer multiverses. We find that landscape norms are approximately normally distributed, such that they permit standard statistical approaches to outlier detection. As shown in Figure 5, training affects landscape norms differentially, depending on model and dataset choices. We observe that INFOVAE exhibits a larger fraction of anomalous configurations than β-VAE and WAE, which motivates us to explore individual PRESTO sensitivities for the main hyperparameters of our VAE models. Studying the distribution these sensitivities, depicted in Figure 6, reveals that INFOVAE has the largest variability in hyperparameter sensitivities—i.e., for INFOVAE, the effect of changing a hyperparameter depends more strongly on the position in the hyperparameter space than for β-VAE and WAE. We conclude that INFOVAE is less representationally stable than its contenders, which is confirmed by an analysis of the local and global PRESTO sensitivities of all models in Appendix B.1.3.

Turning to our transformer multiverses, and further demonstrating the exploratory power of PRESTO, in Figure 7, we use our topological tools to directly compare the multiverse metric spaces (MMSs) of our transformer models. We see that we can distinguish large language models from smaller language models based on topological comparisons between their embedding multiverses. Furthermore, topological comparisons between multiverse metric spaces have higher discriminatory power than comparisons based on Mantel correlation (Mantel, 1967).

5.3. Exploiting Representational Variability

In our explorations of representational variability, we have seen that PRESTO supports sensitivity analysis and outlier detection for latent-space models. Encouraged by these findings, we further investigate how our framework can leverage representational variability (and the lack thereof) to facilitate the efficient selection of representationally robust and reliable latent-space models in practice. As shown in Figure 8, with PRESTO, we can compress a VAE hyperparameter search space by 50%, ensuring

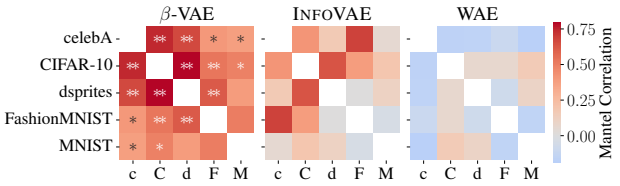


Figure 9. **Reusing hyperparameter knowledge.** We show the Mantel correlation (color) between the hyperparameter multiverses of our three VAE models when trained on five different datasets, annotating Bonferroni-corrected p-values at the 95% ($*$) and the 99% ($**$) significance level. The topological reusability of hyperparameter knowledge strongly depends on the chosen model.

that each coordinate is matched to a structurally similar representative in the compressed space. This suggests that there exist opportunities for environmentally conscious, yet empirically sound hyperparameter selection based on topological insights into latent-space structure (see Appendix B.1.6 for an additional experiment exploring such opportunities).

In Figure 9, we perform MMS comparisons with PRESTO to assess when the widespread custom of reusing hyperparameter knowledge across datasets is topologically justified. We find that among our VAE models, only β-VAE exhibits the cross-dataset latent-space consistency required for such a transfer. Theoretically sound knowledge transferability promotes sustainable yet rigorous machine-learning practices, and PRESTO appears as a promising tool to evaluate the cross-dataset representational consistency of latent-space models that is required to ensure it.

Finally, Appendix B.4 shows how PRESTO can be leveraged in the context of data analysis, e.g., when reasoning about the results of *non-linear dimensionality-reduction* methods.

6. Discussion and Conclusion

We introduced PRESTO, a *topological multiverse framework* to describe and relate (collections of) latent representations. PRESTO flexibly and scalably compares spaces varying in cardinality and dimension, surpassing existing work in generality while still capturing salient topological signal and benefiting from theoretical stability guarantees. By offering novel topological diagnostics for *distributions* of latent spaces, PRESTO unlocks a *structure-driven* alternative for studying representational variability in generative models, complementing performance-driven approaches. Drawing on the notion of *multiverse metric spaces*, we used PRESTO to develop scalable practical tools for efficiently evaluating and selecting latent-space models, including VAEs and transformers, across a wide range of configurations.

Limitations. PRESTO allows us to measure the sensitivity of latent-space models to changes in algorithmic, implementation, and data choices, and we can identify outliers among latent spaces. To the best of our knowledge, no suitable baselines currently exist for these purposes, since existing methods are based on model performance. Hence, exploring alternative approaches to sensitivity analysis and outlier detection for latent-space models based on their internal representations constitutes a crucial avenue for future work.

Moreover, each step in our PRESTO framework offers a number of choices. For example, we can opt to work with *normalized* or *unnormalized* embeddings, and choose *deterministic* or *random* projections. Based on preliminary experiments, our intuition is that normalization emphasizes *topological* variability, whereas computations based on unnormalized embeddings chiefly capture *geometric* variability. Similarly, random projections seem particularly suitable for studying variability *within* individual latent spaces, whereas deterministic projections appear to excel at comparisons *between* different latent spaces. However, how the representational variabilities observed under each choice (or combination of choices) are related, and how they can be interpreted, merits a separate in-depth investigation.

Finally, while PRESTO improves our understanding and handling of representational variability in latent-space models, we have only scratched the surface regarding its applications. Thus, we envisage (1) leveraging PRESTO to study latent representations beyond the generative domain, such as *graph embeddings* and *internal neural-network layers* (see Appendix B.3 for preliminary experiments on the latter), (2) extending PRESTO's reach to other areas of responsible and efficient model selection, such as *representational biases* and *zero-shot stitching*, and (3) integrating PRESTO's hyperparameter-compression and sensitivity-scoring methods into *machine-learning-operations tools*. Moreover, while our initial experiments on hyperparameter-search-space compression and hyperparameter-knowledge reuse seem promising, additional research is necessary to understand *when* and *how* we can leverage insights from one setting to inform hyperparameter-search and hyperparameter-selection strategies in other settings.

Overall, we believe that multiverse approaches are essential in the development of responsible machine-learning practices, and that PRESTO constitutes an important step toward establishing those approaches in the community.

Impact Statement

In this paper, we introduce a topological framework for understanding representational variability in latent spaces, along with scalable practical tools to efficiently select robust and reliable machine-learning models. Thus, our work directly contributes to the advancement of responsible-machine-learning goals.

Reproducibility Statement

We make all code, data, and results publicly available. Reproducibility materials are available at https://doi.org/10.5281/zenodo.11355446, and our code is maintained at https://github.com/aidos-lab/Presto.

Acknowledgments

C.C. is supported by *Digital Futures* at KTH Royal Institute of Technology. B.R. is supported by the Bavarian state government with funds from the *Hightech Agenda Bavaria*.

References

Adams, H., Emerson, T., Kirby, M., Neville, R., Peterson, C., Shipman, P. D., Chepushtanova, S., Hanson, E. M., Motta, F. C., and Ziegelmeier, L. Persistence images: A stable vector representation of persistent homology. *Journal of Machine Learning Research*, 18:8:1–8:35, 2017.

Balabin, N., Voronkova, D., Trofimov, I., Burnaev, E., and Barannikov, S. Disentanglement learning via topology, 2023. arXiv:2308.12696.

Barannikov, S., Trofimov, I., Sotnikov, G., Trimbach, E., Korotin, A., Filippov, A., and Burnaev, E. Manifold topology divergence: A framework for comparing data manifolds. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 7294–7305. Curran Associates, Inc., 2021.

Barannikov, S., Trofimov, I., Balabin, N., and Burnaev, E. Representation topology divergence: A method for comparing neural network representations. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 1607–1626. PMLR, 2022.

Barannikov, S. A. The framed Morse complex and its invariants. *Advances in Soviet Mathematics*, 21:93–115, 1994.

Bell, S. J., Kampman, O., Dodge, J., and Lawrence, N. Modeling the machine learning multiverse. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 18416–18429. Curran Associates, Inc., 2022.

Brown, B. C., Caterini, A. L., Ross, B. L., Cresswell, J. C., and Loaiza-Ganem, G. Verifying the union of manifolds hypothesis for image data. In *International Conference on Learning Representations*, 2023.

Bubenik, P. Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research*, 16:77–102, 2015.

Bubenik, P. and Dłotko, P. A persistence landscapes toolbox for topological statistics. *Journal of Symbolic Computation*, 78:91–114, 2017.

Bubenik, P., Hull, M., Patel, D., and Whittle, B. Persistent homology detects curvature. *Inverse Problems*, 36(2): 025008, February 2020.

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. Understanding disentangling in β-VAE, 2018. arXiv:1804.03599.

Cha, J. and Thiyagalingam, J. Orthogonality-Enforced Latent Space in Autoencoders: An Approach to Learning Disentangled Representations. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202, pp. 3913–3948. PMLR, 2023.

Chazal, F., de Silva, V., and Oudot, S. Persistence stability for geometric complexes. *Geometriae Dedicata*, 173(1): 193–214, 2014a.

Chazal, F., Fasy, B. T., Lecci, F., Rinaldo, A., and Wasserman, L. Stochastic convergence of persistence landscapes and silhouettes. In *Proceedings of the ThirtiethAannual Symposium on Computational Geometry*, pp. 474–483, 2014b.

Chazal, F., Fasy, B., Lecci, F., Michel, B., Rinaldo, A., and Wasserman, L. A. Subsampling methods for persistent homology. In Bach, F. R. and Blei, D. M. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 2143–2151, 2015.

Chen, C., Ni, X., Bai, Q., and Wang, Y. A topological regularizer for classifiers via persistent homology. In Chaudhuri, K. and Sugiyama, M. (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2573–2582. PMLR, 2019.

Cohen-Steiner, D., Edelsbrunner, H., and Harer, J. Stability of persistence diagrams. *Discrete & Computational Geometry*, 37(1):103–120, 2007.

Davari, M., Horoi, S., Natik, A., Lajoie, G., Wolf, G., and Belilovsky, E. Reliability of CKA as a similarity measure in deep learning. In *International Conference on Learning Representations*, 2023.

Diedrichsen, J. and Kriegeskorte, N. Representational models: A common framework for understanding encoding, pattern-component, and representational-similarity analysis. *PLOS Computational Biology*, 13 (4):1–33, 2017.

Diedrichsen, J., Berlot, E., Mur, M., Schütt, H. H., Shahbazi, M., and Kriegeskorte, N. Comparing representational geometries using whitened unbiased-distance-matrix similarity, 2020. arXiv:2007.02789.

Edelsbrunner, H. and Harer, J. *Computational Topology: An Introduction*. American Mathematical Society, Providence, RI, USA, 2010.

Edelsbrunner, H., Kirkpatrick, D., and Seidel, R. On the shape of a set of points in the plane. *IEEE Transactions on Information Theory*, 29(4):551–559, 1983.

Feurer, M. and Hutter, F. Hyperparameter optimization. In Hutter, F., Kotthoff, L., and Vanschoren, J. (eds.), *Automated Machine Learning: Methods, Systems, Challenges*, pp. 3–33. Springer, Cham, Switzerland, 2019.

Freeman, C. D. and Bruna, J. Topology and geometry of half-rectified network optimization. In *5th International Conference on Learning Representations*, 2017.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks, 2014. arXiv:1406.2661.

Gundersen, O. E., Coakley, K., Kirkpatrick, C., and Gil, Y. Sources of irreproducibility in machine learning: A review, 2022. arXiv:2204.07610.

Haibe-Kains, B., Adam, G. A., Hosny, A., Khodakarami, F., Massive Analysis Quality Control (MAQC) Society Board of Directors, Shraddha, T., Kusko, R., Sansone, S.-A., Tong, W., Wolfinger, R. D., Mason, C. E., Jones, W., Dopazo, J., Furlanello, C., Waldron, L., Wang, B., McIntosh, C., Goldenberg, A., Kundaje, A., Greene, C. S., Broderick, T., Hoffman, M. M., Leek, J. T., Korthauer, K., Huber, W., Brazma, A., Pineau, J., Tibshirani, R., Hastie, T., Ioannidis, J. P. A., Quackenbush, J., and Aerts, H. J. W. L. Transparency and reproducibility in artificial intelligence. *Nature*, 586(7829):E14–E16, 2020.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. β-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*, 2017.

Horoi, S., Huang, J., Rieck, B., Lajoie, G., Wolf, G., and Krishnaswamy, S. Exploring the geometry and topology of neural network loss landscapes. In Bouadi, T., Fromont, E., and Hüllermeier, E. (eds.), *Advances in Intelligent Data Analysis XX*, pp. 171–184, Cham, Switzerland, 2022. Springer. doi: 10.1007/978-3-031-01333-1_14.

Joyce, J. M. Kullback-Leibler Divergence. In Lovric, M. (ed.), *International Encyclopedia of Statistical Science*, pp. 720–722. Springer, Heidelberg, Germany, 2011.

Kapoor, S. and Narayanan, A. Leakage and the reproducibility crisis in machine-learning-based science. *Patterns*, 4(9):100804, 2023.

Karp, R. M. Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W. (eds.), *Proceedings of a Symposium on the Complexity of Computer Computations*, pp. 85–103, Boston, MA, USA, 1972. Springer.

Kim, P. J., Jang, Y., Kim, J., and Yoo, J. TopP&R: Robust support estimation approach for evaluating fidelity and diversity in generative models, 2023. arXiv:2306.08013.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes, 2013. arXiv:1312.6114.

Klabunde, M., Schumacher, T., Strohmaier, M., and Lemmerich, F. Similarity of neural network models: A survey of functional and representational measures, 2023. arXiv:2305.06329.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3519–3529. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/kornblith19a.html.

Kostenok, E., Cherniavskii, D., and Zaytsev, A. Uncertainty estimation of transformers' predictions via topological analysis of the attention matrices, 2023. arXiv:2308.11295.

Krishnamoorthy, B. and May, N. H. A Normalized Bottleneck Distance on Persistence Diagrams and Homology Preservation under Dimension Reduction, 2023. arXiv:2306.06727.

Kumar, A. and Poole, B. On implicit regularization in β-VAEs. In Daumé III, H. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5480–5490. PMLR, 2020.

Leeb, F., Bauer, S., Besserve, M., and Schölkopf, B. Exploring the latent space of autoencoders with interventional assays. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 21562–21574. Curran Associates, Inc., 2022.

Mantel, N. The detection of disease clustering and a generalized regression approach. *Cancer Research*, 27 (2_Part_1):209–220, 1967.

McDermott, M. B., Wang, S., Marinsek, N., Ranganath, R., Foschini, L., and Ghassemi, M. Reproducibility in machine learning for health research: Still a ways to go. *Science Translational Medicine*, 13(586):eabb1655, 2021.

McInnes, L., Healy, J., and Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, 2020. arXiv:1802.03426.

Moon, K. R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D. B., Chen, W. S., Yim, K., van den Elzen, A., Hirn, M. J., Coifman, R. R., Ivanova, N. B., Wolf, G., and Krishnaswamy, S. Visualizing structure and transitions in high-dimensional biological data. *Nature Biotechnology*, 37(12):1482–1492, 2019.

Moor, M., Horn, M., Rieck, B., and Borgwardt, K. Topological autoencoders. In Daumé III, H. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, number 119 in Proceedings of Machine Learning Research, pp. 7045–7054. PMLR, 2020.

Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., and Rodolà, E. Relative representations enable zero-shot latent space communication. In *International Conference on Learning Representations*, 2023.

Muller, M. and Strohmayer, A. Forgetting practices in the data sciences. In *CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2022. ACM.

Purvine, E., Brown, D., Jefferson, B., Joslyn, C., Praggastis, B., Rathore, A., Shapiro, M., Wang, B., and Zhou, Y. Experimental observations of the topology of convolutional neural network activations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(8): 9470–9479, 2023.

Reimers, N. and Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Inui, K., Jiang, J., Ng, V., and Wan, X. (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992. Association for Computational Linguistics, 2019.

Rieck, B. and Leitte, H. Persistent homology for the evaluation of dimensionality reduction schemes. *Computer Graphics Forum*, 34(3):431–440, 2015.

Rieck, B. and Leitte, H. Exploring and comparing clusterings of multivariate data sets using persistent homology. *Computer Graphics Forum*, 35(3):81–90, 2016.

Rieck, B., Togninalli, M., Bock, C., Moor, M., Horn, M., Gumbsch, T., and Borgwardt, K. Neural persistence: A complexity measure for deep neural networks using algebraic topology. In *International Conference on Learning Representations*, 2019.

Roweis, S. T. and Saul, L. K. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290 (5500):2323–2326, 2000.

Scoccola, L. and Perea, J. A. FibeRed: Fiberwise Dimensionality Reduction of Topologically Complex Data with Vector Bundles, 2023. arXiv:2206.06513.

Sheehy, D. R. The Persistent Homology of Distance Functions under Random Projection. In *Proceedings of the 30th Annual Symposium on Computational Geometry*, pp. 328–334, 2014.

Simmons, J. P., Nelson, L. D., and Simonsohn, U. False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, 22(11):1359–1366, 2011.

Simonsohn, U., Simmons, J. P., and Nelson, L. D. Specification curve analysis. *Nature Human Behaviour*, 4(11):1208–1214, 2020.

Simson, J., Pfisterer, F., and Kern, C. Using multiverse analysis to evaluate the influence of model design decisions on algorithmic fairness. In *HHAI 2023: Augmenting Human Intellect*, pp. 382–384. IOS Press, 2023.

Sivaprasad, P. T., Mai, F., Vogels, T., Jaggi, M., and Fleuret, F. Optimizer benchmarking needs to account for hyperparameter tuning. In Daumé III, H. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9036–9045. PMLR, 13–18 Jul 2020.

Smith, A. D., Catanzaro, M. J., Angeloro, G., Patel, N., and Bendich, P. Topological parallax: A geometric specification for deep perception models, 2023. arXiv:2306.11835.

Smith, L. N. A disciplined approach to neural network hyper-parameters: Part 1 – Learning rate, batch size, momentum, and weight decay, 2018. arXiv:1803.09820.

Steegen, S., Tuerlinckx, F., Gelman, A., and Vanpaemel, W. Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, 11(5):702–712, 2016.

Tenenbaum, J. B., de Silva, V., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Tolstikhin, I., Bousquet, O., Gelly, S., and Schoelkopf, B. Wasserstein auto-encoders. In *International Conference on Learning Representations*, 2018.

Trofimov, I., Cherniavskii, D., Tulchinskii, E., Balabin, N., Burnaev, E., and Barannikov, S. Learning topology-preserving data representations. In *International Conference on Learning Representations*, 2023.

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

van der Maaten, L. and Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86): 2579–2605, 2008.

van der Merwe, R., Newman, G., and Barnard, E. Manifold characteristics that predict downstream task performance, 2022. arXiv:2205.07477.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Vittadello, S. T. and Stumpf, M. P. H. Model comparison via simplicial complexes and persistent homology. *Royal Society Open Science*, 8(10):211361, 2021.

von Rohrscheidt, J. and Rieck, B. Topological Singularity Detection at Multiple Scales. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 35175–35197. PMLR, July 2023.

Waibel, D. J. E., Atwell, S., Meier, M., Marr, C., and Rieck, B. Capturing shape information with multi-scale topological loss terms for 3D reconstruction. In Wang, L., Dou, Q., Fletcher, P. T., Speidel, S., and Li, S. (eds.), *Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pp. 150–159, Cham, Switzerland, 2022. Springer.

Zhang, G., Li, L., Nado, Z., Martens, J., Sachdeva, S., Dahl, G., Shallue, C., and Grosse, R. B. Which algorithmic choices matter at which batch sizes? Insights from a noisy quadratic model. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Zhao, S., Song, J., and Ermon, S. InfoVAE: Balancing learning and inference in variational autoencoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 5885–5892, 2019.

Zhou, S., Zelikman, E., Lu, F., Ng, A. Y., Carlsson, G. E., and Ermon, S. Evaluating the disentanglement of deep generative models through manifold topology. In *International Conference on Learning Representations*, 2021.

Appendix

In this appendix, we provide the following supplementary materials.

A. **Extended Theory.** Definitions, proofs, and results omitted from the main text.
B. **Extended Experiments.** Additional details and experiments complementing the discussion in the main paper.
C. **Extended Methods.** Further details on properties and parts of the PRESTO pipeline.
D. **Extended Background.** More information on the latent-space models to which PRESTO can be applied.
E. **Extended Related Work.** Discussion of additional related work.

A. Extended Theory

In this section, we state the definitions, provide the proofs of our main theoretical results, and derive the additional results omitted from the main text.

Definition A.1 (Bottleneck Distance d_B)**.** Let X and Y be two finite metric spaces. The **bottleneck distance** between the persistence diagrams of X and Y is denoted by $d_B(\cdot,\cdot)$ and defined as

$$d_B(D_1, D_2) = \min_{\gamma\colon D_1 \to D_2} \max_{x \in D_1} \|x - \gamma(x)\|_\infty \ , \tag{10}$$

for a bijection $\gamma\colon D_1 \to D_2$.

Definition A.2 (Vietoris–Rips Complex)**.** The **Vietoris–Rips complex** of a metric space (X, d) at diameter r, denoted as $\mathbf{VR}(X, r)$, is defined as

$$\mathbf{VR}(X, r) = \{\sigma \subseteq X \mid \forall x, y \in \sigma, d(x, y) \leq r\} \ . \tag{11}$$

The elements σ of $\mathbf{VR}(X, r)$ are the simplices of the complex, and they represent subsets of X such that the pairwise distances between their elements are at most r.

Notation. Following the form and notation of the results in Krishnamoorthy & May, which are proved for Vietoris–Rips filtrations, we write $\mathrm{dgm}(X)$ for the Vietoris–Rips persistence diagram of a metric space X, and similarly $d_B(X, Y)$ for the bottleneck distance between two persistence diagrams $\mathrm{dgm}(X)$ and $\mathrm{dgm}(Y)$.

Definition A.3 (Normalized Bottleneck Distance d_N)**.** For metric spaces (X, d_X) and (Y, d_Y) with diameters $\mathrm{diam}(X), \mathrm{diam}(Y)$, the **normalized bottleneck distance** is defined as

$$d_N(X, Y) = d_B\left(\frac{X}{\mathrm{diam}(X)}, \frac{Y}{\mathrm{diam}(Y)}\right) \ . \tag{12}$$

Moreover, as proved by Krishnamoorthy & May, d_N has the favorable properties of scale invariance and stability.

Definition A.4 (Johnson–Lindenstrauss Projection)**.** Let (X, d_X) and (Y, d_Y) be metric spaces. A **Johnson-Lindenstrauss projection** (JL projection) is a linear map $f\colon X \to Y$ that satisfies the following inequality for some $0 < \varepsilon < 1$:

$$(1 - \varepsilon)\, d_X(u, v)^2 \leq d_Y(f(u), f(v))^2 \leq (1 + \varepsilon)\, d_X(u, v)^2 \ . \tag{13}$$

The JL projection provides a controlled distortion of pairwise distances, allowing for a significant reduction in dimensionality while approximately preserving the geometry of the original space. The celebrated JL Lemma guarantees the existence of such a projection f.

Lemma A.5 (Johnson–Lindenstrauss Persistent Homology Preservation)**.** *Let $X \subset \mathbb{R}^d$ and $\varepsilon \in (0, 1)$. If $f\colon \mathbb{R}^d \to \mathbb{R}^m$ is a JL linear projection, then*

$$d_N(X, f(X)) \leq \varepsilon \ ,$$

where n, the dimension of the projection, is assumed to be larger than $8\ln(|X|)/\varepsilon^2$.

JL-linear maps represent the most efficient methods to collapse extremely large latent spaces. In practice, Gaussian random projections are used, but even random orthogonal projections are sufficient to preserve the topology. However, to achieve even tighter bounds on the topology of the projection at even lower dimensions, we can invoke more sophisticated projection methods.

Multidimensional Scaling (MDS). Given an input metric space $X = \{x_1, \ldots, x_n\}$ with a metric d_X and desired reduced dimension m, MDS finds a centered data set $\tilde{X} = \{\tilde{x}_1, \ldots, \tilde{x}_n\} \subset \mathbb{R}^m$ such that $\sum_{i,j=1}^{n} \left(d(x_i, x_j) - \|\tilde{x}_i - \tilde{x}_j\|\right)^2$ is minimized. This projection is achieved in two steps:

1. Obtain a realization Φ of X in $\mathbb{R}^k$ for some k, i.e., $\Phi\colon X \to \Phi(X) \subset \mathbb{R}^k$ is an isometry.
2. Orthogonally project the realized data onto the first $m \leq d$ dominant eigenvectors of the covariance matrix $C(\Phi(X))$.

To find a realization of X in $\mathbb{R}^k$, use the fact that $-{}^1\!/_2 D_X^{\circ 2} = C_n G X C_n$, where GX is the Gram matrix of X, and $C_n = I_n - {}^1\!/_n \mathbf{1}_n$ is the centering matrix. Since GX is positive semidefinite, it has a unique root $Z = [z_1, \ldots, z_n] \in \mathbb{R}^{k \times n}$ such that $Z^T Z = GX$. The realization of X is given by $\Phi(X) = \{z_1, \ldots, z_n\}$. With the realization matrix Z, compute the *singular-value decomposition* $Z = U \Sigma V^T$, where $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \ldots \geq \sigma_n$. The eigenvectors of $C(Z) = Z Z^T = U \Sigma^2 U^T$ are the columns of U, and the eigenvalues λ_i of $C(Z)$ are σ_i^2. Take the first m columns of U to perform an orthogonal projection.

Definition A.6 (Metric Multidimensional Scaling [mMDS])**.** Let $(X, d_X) = \{x_1, \ldots, x_n\} \subset \mathbb{R}^d$ be a finite metric space with $GX = U \Sigma^2 U^T$ the corresponding Gram matrix, and $Z = [z_1, \ldots, z_n]$ a realization of X in $\mathbb{R}^k$ where $GX = Z^T Z$. To embed X into dimension $0 < m \leq d$, we project onto the first m dominant eigenvectors of GX, defined as $\tilde{U} = [u_1, \ldots, u_m]$. The mMDS reduction $P_X^{(m)}\colon X \to \mathbb{R}^m$ is the map

$$P_X^{(m)}(x_i) = \tilde{x}_i = \operatorname{proj}_{\operatorname{Im}(\tilde{U})}(z_i) \ . \tag{14}$$

Lemma A.7 (mMDS Homology Preservation)**.** *Let $X \subset \mathbb{R}^d$ and $0 < m \leq d$. Then $P_X^{(m)}\colon X \to \mathbb{R}^m$ preserves the homology of X according to the following bound on d_N:*

$$d_N(X, P_X^{(m)}) \leq \frac{2\sqrt{2}}{\operatorname{diam}(P_X^m(X))} \sqrt[4]{\frac{(\sum_1^m \lambda_i^2)(\sum_{m+1}^d \lambda_i^2)}{(\sum_1^d \lambda_i^2)}} \ ,$$

where λ_i is defined to be the ith eigenvalue of the covariance matrix $C(X)$.

Lemma A.8 (bi-Lipschitz Homology Preservation)**.** *For metric spaces $(X, d_X), (Y, d_Y)$, let $f\colon X \to Y$ be a k-bi-Lipschitz map. The change in d_N can be bounded as*

$$d_N(X, Y) = \left| \frac{k^2 - 1}{k} \right| \frac{\operatorname{diam}(X)}{\operatorname{diam}(Y)} \ . \tag{15}$$

Before we can prove general statements about the preservation of our topological distance under projections, we need to prove some supporting lemmas concerning the relationships between various topological distances.

Lemma A.9 (Bottleneck-Distance Bound)**.** *Let $p \in \mathbb{R}_{>0}$ and W_p denote the pth Wasserstein distance between persistence diagrams. Then the* bottleneck distance *d_B constitutes an upper bound for W_p, i.e., $W_p \leq d_B$.*

Proof. We prove a more general statement. For $p \leq q$, the function $\phi(x) := x^{\frac{q}{p}}$ is *convex*. Thus, for any function f and any probability measure μ, we have

$$\left(\int f^p \mathrm{d}\mu \right)^{\frac{1}{p}} = \phi\left(\left(\int f^p \mathrm{d}\mu \right)^{\frac{1}{p}} \right)^{\frac{1}{q}} \tag{16}$$

$$\leq \left(\int \phi\left(f^p\right) \mathrm{d}\mu \right)^{\frac{1}{q}} \qquad \text{(by Jensen's inequality)} \tag{17}$$

$$= \left(\int f^q \mathrm{d}\mu \right)^{\frac{1}{q}} . \tag{18}$$

As this result holds for *general* functions f, it particularly holds for the cost functions used in the calculation of the Wasserstein distance between persistence diagrams. Finally, we have $\lim_{p \to \infty} W_p = d_B$ (some works even use the suggestive notation W_∞ to denote the bottleneck distance), concluding the proof. □

Continuing in a similar vein, we turn to the analysis of the norms of *persistence landscapes*.

Lemma A.10 (Landscape-Norm Bound)**.** *Let $p \in \mathbb{R}_{>0}$ and $\| \cdot \|_p$ denote the pth persistence landscape norm (Bubenik, 2015, Section 2.4). Then the* infinity norm *$\| \cdot \|_\infty$ is an upper bound for $\| \cdot \|_p$, i.e., $\|\lambda\|_p \leq \|\lambda\|_\infty$ for all persistence landscapes λ.*

Proof. The persistence landscape norm is simply an L^p norm, calculated for a specific class of piecewise-linear functions, i.e., the persistence landscapes. We prove a more general result holding for *all* functions f that satisfy a certain integral property. Specifically, we assume that $f \in L^q$, i.e., the absolute value of f, raised to the qth power, has a finite integral. This holds for a large class of functions, and for *all* persistence landscapes in particular. Since $|f|^p \leq |f|^q$ for $p \leq q$, this implies that $f \in L^p$. Now let $a := q/p$ and $b := q/q-p$. Since $1/a + 1/b = 1$, the numbers a and b are *Hölder conjugates* and we may apply Hölder's inequality with $g = \mathbb{1}$, the function that is identically to 1 over the domain of f, yielding

$$\|f^p \cdot \mathbb{1}\|_1 \leq \|f^p\|_a \cdot \|\mathbb{1}\|_b \ . \tag{19}$$

The left-hand side is equivalent to $\|f\|_p^p$, while the right-hand side, following the definition of the norm, evaluates to

$$\|f^p\|_a = \int \left(|f^p|^{\frac{q}{p}} \mathrm{d}\mu\right)^{\frac{p}{q}} \tag{20}$$

$$= \|f\|_q^p. \tag{21}$$

By definition of the norm, the other factor satisfies $\|\mathbb{1}\|_b \geq 1$. Putting this together, we have

$$\|f\|_p^p \leq \|f\|_q^p \tag{22}$$

$$\therefore \|f\|_p \leq \|f\|_q \ . \tag{23}$$

As a consequence, the persistence-landscape norms are bounded similarly to the Wasserstein distances. Noticing that $\lim_{p\to\infty} \|f\|_p = \|f\|_\infty$ concludes the proof. □

The immediate consequence of Lemma A.9 and Lemma A.10 is that *any* bound calculation of our topological distances is, eventually, upper-bounded by the bottleneck distance between persistence diagrams. While such a bound is by its very nature potentially rather coarse, the bottleneck distance is suitable as the most extreme upper topological bound since it is known to be itself bounded by the geometrical variation between the two spaces from which the respective persistence diagrams arise. More precisely, we have

$$d_B(X, Y) \leq 2d_{\mathrm{GH}}(X, Y) \ , \tag{24}$$

where d_{GH} denotes the *Gromov–Hausdorff distance* between X and Y (Chazal et al., 2014a). Thus, any topological variation, regardless of the distance metric, is upper-bounded by the geometrical variation between two spaces.

Theorem 3.8 (Metric-Space Preservation under Projection)**.** *Given an MMS $\mathfrak{M}$ and an associated PMMS $\mathfrak{M}^k$ with topological loss ℓ^k, we can bound the pairwise-distance perturbation under projection as $\mathfrak{M}^k[i,j] \leq \mathfrak{M}[i,j] + 2\ell^k$.*

Proof. Following the preceding discussion, as well as Lemma A.9 and Lemma A.10, it is sufficient to phrase the desired inequality in terms of the bottleneck distance between two spaces. Let E_i, E_j denote the respective embeddings, and let f be a projector. Starting from the left-hand side, and using the triangle inequality, we get

$$d_B\left(f\left(E_i\right), f\left(E_j\right)\right) \leq d_B\left(f\left(E_i\right), E_i\right) + d_B\left(E_i, f\left(E_j\right)\right) \tag{25}$$

$$\leq d_B\left(f\left(E_i\right), E_i\right) + d_B\left(E_i, E_j\right) + d_B\left(E_j, f\left(E_j\right)\right) \tag{26}$$

$$\leq \mathfrak{M}[i,j] + d_B\left(f\left(E_i\right), E_i\right) + d_B\left(E_j, f\left(E_j\right)\right). \tag{27}$$

The last two terms on the right-hand side depend on the selected projector function. By Eq. (7), we see that their sum is upper-bounded by $2\ell^k$, concluding the proof. □

Theorem 3.8 might not be entirely satisfying because ℓ^k still has an underlying dependency on the projector. However, there are no general bounds available unless we choose a specific projector. Notice that for many classes of projectors, including metric MDS, PCA, and random projections, suitable bounds for ℓ^k itself may be obtained.

Theorem 3.9 (PRESTO Variance under Projection)**.** *Consider an MMS $\mathfrak{M} = (\mathcal{E}, d_T)$ with the landscape distance $d_T(E_i, E_j) := d(L(E_i), L(E_j))$ and associated persistence landscapes $\mathcal{L}_{\mathfrak{M}}$. Further, let $\mathfrak{M}^k$ be a PMMS with a topological loss ℓ^k. Then we can bound the maximal change in* any *persistence landscape norm as*

$$\|L(E_i)\| - \ell^k \leq \|L(P_i)\| \leq \|L(E_i)\| + \ell^k \ . \tag{8}$$

Given the PRESTO *variance of* $\mathfrak{M}$*,* $PV(\mathcal{L}_{\mathfrak{M}})$*, we can bound the* PRESTO *variance of* $\mathfrak{M}^k$*, i.e.,* $PV(\mathcal{L}_{\mathfrak{M}^k})$*, as*

$$|PV(\mathcal{L}_{\mathfrak{M}}) - PV(\mathcal{L}_{\mathfrak{M}^k})| \leq \frac{4\ell^k}{N}\sigma_i + \frac{\left(2\ell^k\right)^2}{N} \ , \tag{9}$$

where $\sigma_i \coloneqq \sum_{i=1}^{N}(\|L(E_i)\| - \mu_{\mathcal{E}})$.

Proof. For the first part of the statement, we notice that $\|L(E_i)\| = d(L(E_i), \emptyset)$, i.e., the distance of a given persistence landscape to the empty landscape. Thus,

$$\begin{aligned}
\|L(E_i)\| &\leq d(L(E_i), L(P_i)) + d(L(P_i), \emptyset) && \text{(triangle inequality)} && (28)\\
&= d(L(E_i), L(P_i)) + \|L(P_i)\| && \text{(by definition of the norm)} && (29)\\
&\leq \|L(P_i)\| + \ell^k && \text{(by definition of } \ell^k\text{).} && (30)
\end{aligned}$$

Applying the same argument to $\|L(P_i)\|$ concludes the proof of this part. As for the second part, we recall that we want to bound $|\mathrm{PV}(\mathcal{L}_{\mathfrak{M}}) - \mathrm{PV}(\mathcal{L}_{\mathfrak{M}^k})|$. To this end, we need to find a suitable bound for the second term, i.e., the term that deals with *projected* embeddings. Here, we note that each summand is of the form $(\|L(P_i)\| - \mu_{\mathcal{P}})^2$, for which we can obtain a bound using Eq. (8) as

$$\begin{aligned}
\|L(P_i)\| - \mu_{\mathcal{P}} &\leq \|L(E_i)\| + \ell^k - \mu_{\mathcal{P}} && (31)\\
&\leq \|L(E_i)\| + \ell^k - \mu_{\mathcal{E}} + \ell^k \ , && (32)
\end{aligned}$$

with the second inequality arising from the lower bound of Eq. (8). $|\mathrm{PV}(\mathcal{L}_{\mathfrak{M}}) - \mathrm{PV}(\mathcal{L}_{\mathfrak{M}^k})|$. Putting this together and calculating the differences per term, we get

$$|\mathrm{PV}(\mathcal{L}_{\mathfrak{M}}) - \mathrm{PV}(\mathcal{L}_{\mathfrak{M}^k})| \leq \frac{4\ell^k}{N}\sum_{i=1}^{N}(\|L(E_i)\| - \mu_{\mathcal{E}}) + \frac{\left(2\ell^k\right)^2}{N} \ . \tag{33}$$

□

B. Extended Experiments

In this section, we provide additional details and experiments complementing the discussion in Section 5, presenting

1. extended experiments in our **VAE hyperparameter multiverse** and our **β-VAE implementation multiverse**,
2. extended experiments in our **transformer multiverse**, including a multiverse analysis of the **PRESTO pipeline**,
3. preliminary experiments exploring PRESTO's use as a dissimilarity measure for **neural-network representations**, and
4. a multiverse analysis of **non-linear dimensionality-reduction** methods.

We make all our code, data, and results available at https://doi.org/10.5281/zenodo.11355446. Our code is maintained at https://github.com/aidos-lab/Presto.

B.1. Extended Experiments in VAE Multiverses

Here, we provide further details on the configuration of our VAE multiverses, as well as additional experiments complementing our discussion in the main text.

B.1.1. THE VAE HYPERPARAMETER MULTIVERSE

As previewed in the main text, our VAE hyperparameter multiverse investigates the hyperparameter space for three commonly cited autoencoder architectures, namely (1) β-VAE (Higgins et al., 2017), (2) INFOVAE (Zhao et al., 2019), and (3) WAE (Tolstikhin et al., 2018), covering hyperparameters ranges that appear commonly in the literature, as well as widely used open-source implementations. The explicit values and brief descriptions of the grid searches that determine our multiverse are detailed in Table 1.

We select three hyperparameters for each architecture, combining their searches to generate 24 unique configurations. Each model was trained using a random $[0.6, 0.3, 0.1]$ train/validation/test split for each of our five datasets. Furthermore, to

VAE	θ_i	Values	Description
β-VAE	β	$1, 4, 16, 64$	Recon Bias
	γ	$500, 750, 1\,000$	KLD Bias
	ℓ	B, H	Loss Variations
INFOVAE	β	$1, 5, 10$	Recon Bias
	α	$-5, -2, -0.5, 0$	KLD Bias
	κ	imq, rbf	MMD Kernel
WAE	λ	$10, 20, 50, 100$	MMD Prior Bias
	ν	$1, 2, 3$	Kernel Width
	κ	imq, rbf	MMD Kernel

Table 1. **VAE Hyperparameter Multiverse.** For each of three VAEs, we explore the product of all varied-parameter values across five datasets for two sample sizes (100% and 50% of the training set), for a total of $3 \cdot 5 \cdot 4 \cdot 2 \cdot 3 \cdot 2 = 720$ configurations. In particular, we get 48 configurations *per architecture*, which we use to evaluate sensitivity in Tables 5 and 6, detect outliers in Figure 5, and compare multiverses in Figure 9. For β-VAE loss variations, H is the original implementation from Higgins et al. (2017), and B stems from Burgess et al. (2018).

Parameters	Datasets				
	celebA	*CIFAR-10*	*dsprites*	*FashionMNIST*	*MNIST*
Hidden Dimensions	$(32, 64, 128, 256, 512)$	$(32, 64, 128)$	$(8, 16)$	$(32, 64)$	$(32, 64)$
Latent Dimension	50	25	25	10	10
Batch Size	128	128	128	64	64

Table 2. **Default implementation choices in the VAE hyperparameter multiverse**. We show our fixed implementation parameters, over which we vary the algorithmic parameters listed in Table 1. Each model was trained using an ADAM optimizer with a learning rate of 0.001 over 30 epochs.

understand the variability in latent structure under different cardinalities, we train models with 100% (0.6) and 50% (0.3) training-set sizes, keeping validation- and test-set cardinalities fixed.

Although our multiverse approach supports varying algorithmic, implementation, and data choices, we found it prudent to design an environment that demonstrates the latent variability of algorithmic choices. Naturally, this required fixing *some* parameters across runs (e.g., our train/test/split ratios). See Table 2 for the values of our fixed implementation choices.

B.1.2. THE β-VAE IMPLEMENTATION MULTIVERSE

Complementing our VAE hyperparameter multiverse, we design a multiverse to explore how implementation choices, i.e., (1) batch size b, (2) hidden dimensions h, (3) learning rate l, (4) sample size s, (5) and train-test split t can affect the latent representations of variational autoencoders. We focus on β-VAE, varying the aforementioned parameters. Our exact choices are detailed in Table 3 and Table 4.

β	Parameter	Values
$\{2, 16, 64\}$	Batch Sizes (b)	$8, 16, 32, 64, 128, 256$
	Learning Rates (l)	$0.002, 0.004, 0.008, 0.016, 0.032, 0.064$
	Training Sample Sizes (s)	$0.5, 0.6, 0.7, 0.8, 0.9, 1$

Table 3. **β-VAE Implementation Multiverse.** For β-VAE, we explore the relation between the β hyperparameter and various implementation choices. This table contains our choices for batch size (b), learning rate (l), and sample size (s). Our choices for train-test split (t) and hidden dimensions (h), are explained in Table 4 and Appendix B.1.2, respectively.

Hidden Layers	*celebA*	*CIFAR-10*	*dsprites*	*FashionMNIST*	*MNIST*
2	$(16, 32)$ $(32, 64)$ $(64, 128)$ $(128, 256)$	$(8, 16)$ $(16, 32)$ $(32, 64)$ $(64, 128)$	$(8, 16)$ $(16, 32)$ $(32, 64)$ $(64, 128)$	$(8, 16)$ $(16, 32)$ $(32, 64)$ $(64, 128)$	$(8, 16)$ $(16, 32)$ $(32, 64)$ $(64, 128)$
3	$(32, 64, 128)$	$(8, 16, 32)$ $(32, 64, 128)$	$(16, 32, 64)$	$(16, 32, 64)$	
4			$(16, 32, 64, 128)$	$(16, 32, 64, 128)$	$(16, 32, 64, 128)$
5	$(32, 64, 128, 256, 512)$				

Table 4. **Implementation choices in the β-VAE implementation multiverse: Hidden layers for our datasets**. Given the varying complexity of each dataset in our β-VAE implementation multiverse, we vary the complexity of the layer as seen in the literature and popular PYTORCH autoencoder frameworks.

Our train-test splits iterate over five random train/validation/test splits at a fixed size of $[0.6, 0.3, 0.1]$. Moreover, acknowledging interplay between algorithmic and implementation choices, we train each implementation configuration over 3 different choices for $\beta \in \{2, 16, 64\}$, for the datasets that we also considered in the hyperparameter multiverse, i.e., (1) *celebA*, (2) *CIFAR-10*, (3) *dsprites*, (4) *FashionMNIST*, and (5) *MNIST*.

B.1.3. PRESTO SENSITIVITIES

Our sensitivity scores are based on the variance in the structure of a *distribution* of latent spaces, as measured by PRESTO. Though this is not restricted to multiverse analysis (i.e., the variation in landscape norms can be applied to any distribution of embeddings), the multiverse treatment of different algorithmic, implementation, and data choices results naturally in a collection of latent representations. Thus, we tailor our scores specifically to understanding variation within a multiverse. As stated in the main text, we define three different variations of PRESTO sensitivity (PS) in Definition 3.4 that make use of Definition 3.3. We repeat these definitions here for convenience.

Given a multiverse $\mathcal{M}$, fix a model dimension i, and define an equivalence relation $\sim_i$ such that $\theta' \sim_i \theta'' \Leftrightarrow \theta'_j = \theta''_j$ for all $\theta', \theta'' \in \mathcal{M}$ and $j \neq i$, yielding q_i equivalence classes $\mathcal{Q}_i$.

The ***individual* PRESTO^p sensitivity** of equivalence class $Q \in \mathcal{Q}_i$ in $\mathcal{M}$ is

$$\mathrm{PS}_h^p(Q \mid \mathcal{M}) \coloneqq \sqrt{\mathrm{PV}_h^p(\mathcal{L}[Q])}\ ,$$

where $\mathcal{L}[Q] \subset \mathcal{L}$ is the set of landscapes associated with models in equivalence class Q. Aggregating over all equivalence classes in $\mathcal{Q}_i$, we obtain the ***local* PRESTO^p sensitivity** of $\mathcal{M}$ in model dimension i as

$$\mathrm{PS}_h^p(\mathcal{M} \mid i) \coloneqq \sqrt{\frac{1}{q_i} \sum_{Q \in \mathcal{Q}_i} \mathrm{PV}_h^p(\mathcal{L}[Q])}\ .$$

Finally, aggregating over all $c = |\theta|$ dimensions of models in $\mathcal{M}$ yields the ***global* PRESTO^p sensitivity** of $\mathcal{M}$, i.e.,

$$\mathrm{PS}_h^p(\mathcal{M}) \coloneqq \sqrt{\frac{1}{c} \sum_{i \in [c]} \frac{1}{q_i} \sum_{Q \in \mathcal{Q}_i} \mathrm{PV}_h^p(\mathcal{L}[Q])}\ .$$

Recall from Definition 3.3 that p and h represent the p-norm for landscapes and the homology dimension, respectively. In our experiments, we consider up homology features up to second order ($h = 2$), which preserves the scalability of our pipeline while still capturing descriptive higher-dimensional topological information. Additionally, we default to $p = 2$, understanding nicely the theoretical trade-offs for different p-norms (cf. Lemma A.10).

VAE	θ_i	$\mu_{\|\mathcal{L}\|}$	$\sigma_{\|\mathcal{L}\|}$	μ_{PS}	σ_{PS}	$\sigma_{\|\mathcal{L}\|}/\mu_{\|\mathcal{L}\|}$	$\sigma_{\text{PS}}/\mu_{\text{PS}}$	$\mu_{\text{PS}}/\mu_{\|\mathcal{L}\|}$	$\sigma_{\text{PS}}/\sigma_{\|\mathcal{L}\|}$
β-VAE	β	0.0171	0.0012	0.0067	0.0012	0.0714	0.1762	0.3899	0.9615
	γ	0.0171	0.0017	0.0065	0.0016	0.1021	0.2453	0.3807	0.9148
	ℓ	0.0171	0.0015	0.0065	0.0015	0.0893	0.2364	0.3819	1.0105
INFOVAE	α	0.0182	0.0063	0.0072	0.0044	0.3475	0.6176	0.3953	0.7026
	β	0.0182	0.0040	0.0081	0.0047	0.2191	0.5815	0.4461	1.1841
	κ	0.0182	0.0043	0.0081	0.0047	0.2344	0.5820	0.4421	1.0978
WAE	λ	0.0185	0.0008	0.0075	0.0005	0.0405	0.0610	0.4063	0.6111
	ν	0.0185	0.0011	0.0075	0.0010	0.0573	0.1306	0.4022	0.9168
	κ	0.0185	0.0013	0.0074	0.0013	0.0690	0.1705	0.3978	0.9824

Table 5. **Local PRESTO sensitivity in the VAE hyperparameter multiverse.** Along with the average and standard deviation of the landscape norms for the latent representations associated with each architecture ($\mu_{\|\mathcal{L}\|}, \sigma_{\|\mathcal{L}\|}$), we display the local sensitivities for each of the main parameters searched (see Table 1 for our parameter list). Recall that local PS is an average over different equivalence classes that partition $\mathcal{M}$, denoted (μ_{PS}). Given the standard partition sizes in this multiverse (equal number of hyperparameter configurations per architecture), we can also take standard deviations (σ_{PS}) and other derivatives, further describing the relative variability between architectures. We see that different VAE architectures have varying levels of sensitivities and robustness across their hyperparameter spaces, which should be investigated thoroughly when using them as generative models.

VAE	$\mu_{\|\mathcal{L}\|}$	$\sigma_{\|\mathcal{L}\|}$	μ_{PS}	σ_{PS}	$\sigma_{\|\mathcal{L}\|}/\mu_{\|\mathcal{L}\|}$	$\sigma_{\text{PS}}/\mu_{\text{PS}}$	$\mu_{\text{PS}}/\mu_{\|\mathcal{L}\|}$	$\sigma_{\text{PS}}/\sigma_{\|\mathcal{L}\|}$
β-VAE	0.0171	0.0015	0.0066	0.0014	0.0876	0.2193	0.3842	0.9623
INFOVAE	0.0182	0.0049	0.0078	0.0046	0.2670	0.5937	0.4278	0.9948
WAE	0.0185	0.0011	0.0075	0.0009	0.0556	0.1207	0.4021	0.8368

Table 6. **Global PRESTO sensitivity in the VAE hyperparameter multiverse.** Here, we aggregate the local sensitivities into a global sensitivity score, analyzing the representational variability across 48 unique latent representations for each VAE architecture in the multiverse. Again, we provide additional derivative statistics to give context for the scale of these scores. These values echo our findings described in the main paper: INFOVAE has the highest representational variability in the VAE hyperparameter multiverse.

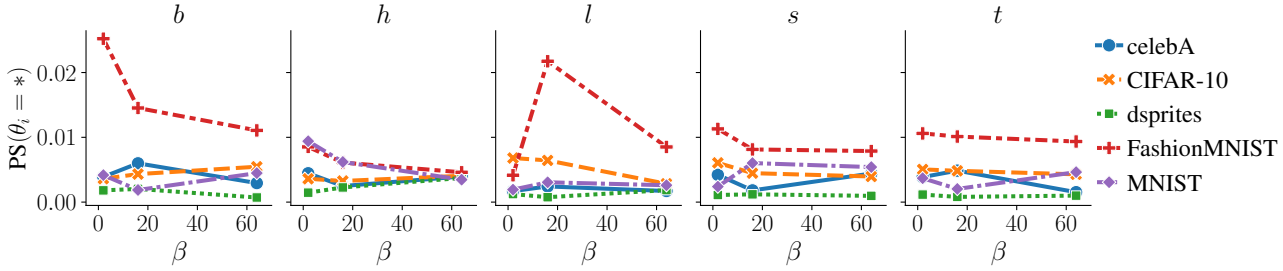
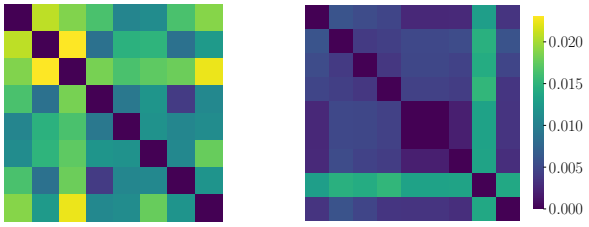


Figure 10. **PRESTO sensitivity in the β-VAE implementation multiverse.** We show the PRESTO sensitivity scores for the β-VAE as a function of β when varying (from left to right) batch size b, hidden dimensions h, learning rate l, sample size s, and the train-test split t. There is no consistent relationship between the choice of β and the sensitivity of the latent space to implementation-parameter choices.

Against this background, we provide overviews of local and global PRESTO sensitivities and related statistics in the VAE hyperparameter multiverse in Tables 5 and 6, along with an analysis of PRESTO sensitivities in the β-VAE implementation multiverse in Figure 10. The overview tables confirm our impression from the main paper that INFOVAE has the highest representational variability among our VAE models, whereas Figure 10 highlights the complex interplay between hyperparameter and implementation choices.

Parameter	Value
Dataset	*MNIST*
Batch Size	64
Model	β-VAE
*Seeds**	$0, 2^1, 2^2, 2^3, 2^4, 2^5, 2^6, 2^7$
Latent Dimension	5
Hidden Dimensions	(8, 16)
β	4
γ	1000.0
Loss Type	H
Optimizer	Adam
Learning Rate	0.001

Table 7. **Model configurations for experiment with random initializations.** To investigate how training affects variability in latent-space models, we train eight β-VAE models on a fixed training set of *MNIST*. The models only differ in their random initializations (*Seeds**).



(a) Untrained Models (b) Trained Models

Figure 11. **Examining latent representations pre- and post-training.** We juxtapose the pairwise PRESTO distances of untrained latent representations that arise from identical β-VAE model configurations trained on *MNIST* with different random seeds (a) with the pairwise PRESTO distances of latent representations arising from these configurations after training (b). We find that training decreases the structural variability among randomly seeded models, implying that models agree about key topological features. Moreover, PRESTO is able to detect anomalous seeds, i.e., seeds resulting in latent representations that, when trained, constitute structural outliers.

Extending our exploration of how training affects variability in latent-space models (cf. Figure 5), we further investigate the sensitivity of random initializations in a small multiverse of β-VAE models trained on the *MNIST* dataset. In particular, we fix all parameters except for the random seed that initializes the β-VAE, leading to eight models in the multiverse (see Table 7 for the full configurations). We use PRESTO to assess how much structural variability is induced by random initializations alone, and compare this to the variability we observe in the latent spaces of our trained models. To this end, we compute the PRESTO sensitivity over eight untrained embeddings and eight trained embeddings based on a fixed training set from *MNIST*. We find that the sensitivity of the initial embeddings is higher (PS $= 0.0188$) than that of the trained embeddings (PS $= 0.0076$).

To complement this analysis, we additionally compute the multiverse metric spaces (MMSs) for the multiverses arising from our initialized and trained embeddings, visualizing the pairwise PRESTO distances between the universes in each MMS in Figure 11. Again, we find that the topological variation of the initial embeddings is *higher* than that of their trained counterparts, indicating a convergence to similar topological features over training. PRESTO's invariance to scaling (when using normalization) allows us to analyze this convergence despite geometric differences in coordinate systems between the trained latent representations. We also observe that PRESTO reveals a seed resulting in an anomalous latent representations (here: seed 2^6), highlighting the importance of tools like PRESTO that can understand distributions of latent representations and reassess the impact of choices like random initialization that are often overlooked in practice.

B.1.4. PRESTO'S RELATION TO GEOMETRIC AND GENERATIVE SIMILARITY

Having revisited PRESTO sensitivities, we now expand our discussion around Figure 4, where we asked what geometric and generative signal exactly is picked up by our framework. We begin by describing our experimental setup in more detail and then move on to discuss PRESTO's relation to various notions of geometric and generative similarity.

Experimental Setup. Let $E = M(X)$ for $M \in \mathcal{M}$ be a latent space in our VAE multiverse, where X is a training set of images. Fix a subspace cardinality (N), such that we can choose a random sample from the training set $S \subseteq X$ such that $|S| = N$. We then map S into the latent space, obtaining $M(S) \subseteq E$, which will be our objects of comparison, i.e., $\mathcal{S} \coloneqq \{M(S) \mid M \in \mathcal{M}\}$, where $|\mathcal{S}| = n_d$ is the number of random draws. This results in a set of random latent subspaces, used to compute the correlations displayed in Figure 4.

Geometric Similarity. To study the similarity in geometric structure between latent spaces, we can use a number of tools from representation learning. Recognizing the plethora of well-studied approaches in the field, and acknowledging the myriad interesting tools we could use to further investigate this phenomenon in future work, for the purposes of our current exposition, we take relatively simple approach. We opt to endow each $M(S) \in \mathcal{S}$ with a metric, such that we can compare metric spaces between $M(S), M'(S) \in \mathcal{S}^2$ using the Pearson distance between their matrix representations. Note that these are indeed *metric subspaces* of the original embeddings. By aggregating the observed behavior over a large number of random draws, we obtain a computable baseline for assessing the geometric capabilities of PRESTO (working *without* normalization). Given that the α-complexes leveraged by PRESTO default to using Euclidean distances between points, we also use this to produce a (pairwise distance) matrix representation of the elements of $\mathcal{S}$. The results are displayed in the left panel of Figure 4. Additionally, given the utility of *cosine similarity* in the study of latent spaces, we compute the Pearson distance between the pairwise cosine similarity matrices for $M(S), M'(S)$, to assess PRESTO's relation to a different geometric signal in the right panel of Figure 12.

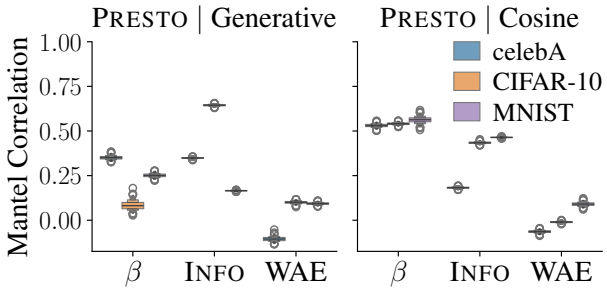


Figure 12. **PRESTO's correlation with generated images and cosine distances.** We assess PRESTO's ability to detect generative (dis)similarity between models by measuring its correlation with batch MSE loss for random subsets of images, generated by perturbing aligned latent coordinates (left), as well as its correlation with the Pearson distance between random subspaces represented by their pairwise cosine similarity matrices (right). Some VAE architectures show potential to have unsupervised generative properties described by the topological and geometric properties of their latent spaces.

Generative Similarity. As established in Figure 4, with further details provided in Appendix B.1.5, PRESTO is in many ways *orthogonal* to performance. This leads us to an interesting question: Are there other properties of a model's latent space, also orthogonal to performance, that can describe a the model's properties as a generator? While this merits its own extensive study, for the purposes of this work, we are interested in designing an experiment that could relate a notion of *unsupervised* generative similarity to PRESTO. Using the experimental design described above, namely $\mathcal{S}$, we suggest generating *comparable* images that were unseen during training using the following pipeline. For each $M(S) \in \mathcal{S}$: (1) Compute the centroid C of the original latent space E. (2) For $v \in M(S)$, compute $\tilde{v} = (1-t)v + tC$, where $t \in \mathbb{R}$ is a (small) perturbation parameter. (3) Use the decoder associated with M to generate a new image $I_{\tilde{v}}$.

This establishes a set of generated images $\tilde{I} \coloneqq \{I_{\tilde{v}} : v \in M(S)\}$ that is robust to multiverse considerations—as various latent spaces are encoded into very different scales and coordinate systems, directly sampling from the latent space becomes impractical. In contrast, we provide an unsupervised approach for generating principled comparisons between $M(S), M'(S)$ in the pixel space by comparing the batch Mean-Squared Error (MSE) between $\tilde{I}$ and $\tilde{I}'$. We display our results for measuring the correlation between PRESTO and the *generative distance matrices* over different random draws of $\mathcal{S}$, such that each cell in the matrix $M[i,j]$ that compares E_i to E_j is computed by $MSE(\tilde{I}_i, \tilde{I}_j)$, in the left panel of Figure 12.

B.1.5. PRESTO'S RELATION TO PERFORMANCE

Continuing the corresponding discussion in the main paper, we now further analyze the relationship between PRESTO, an inherently structural measure, and performance. Figure 4 from the main text highlights that, in many of our evaluations, the latent spaces from the most stable and reliable VAE architecture (β-VAE) show no correlation between PRESTO and performance. In a similar vein, Figure 13 depicts the relationship between landscape norms and performance (MSE reconstruction loss on the test set) for WAE and INFOVAE. p vs. landscape norm. Like β-VAE, landscape norms for WAE appear to be orthogonal to performance, whereas the conclusion for INFOVAE (our overall weakest performer) is not nearly as clear and merits additional exploration in future work. Across all model architectures, however, PRESTO demonstrates the capacity to characterize differences between the latent spaces of models with *similar performance*.

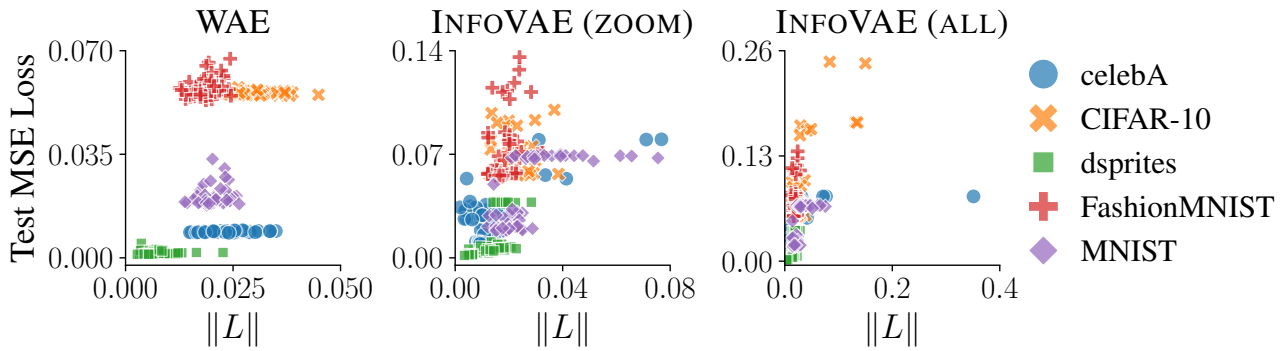


Figure 13. **Relationships between landscape norms and performance**. To assess the relationship between generative-model performance and PRESTO, we compare landscape norms, a proxy for topological complexity, and test reconstruction loss (MSE errors) for the WAE and INFOVAE architectures. We observe that PRESTO can distinguish WAE models that perform similarly across multiple datasets, while INFOVAE's hyperparameter space, which is sensitive with respect to performance and representational variability, results in unnecessary complexity that should encourage additional care when applying this model to known and unknown tasks.

B.1.6. LOW-COMPLEXITY TRAINING

Finally, we examine PRESTO's ability to perform hyperparameter compression in *low-complexity* training environments. While we already established exciting opportunities for hyperparameter reuse across datasets using PRESTO (cf. Figure 9), low-complexity training constitutes another avenue for leveraging PRESTO's hyperparameter compression to dismantle the environmentally costly culture of brute-force performance-based hyperparameter optimization. In Figure 14, we investigate the stability of our compression routine when halving the size of the training set (i.e., reducing it to 50% of its original size).

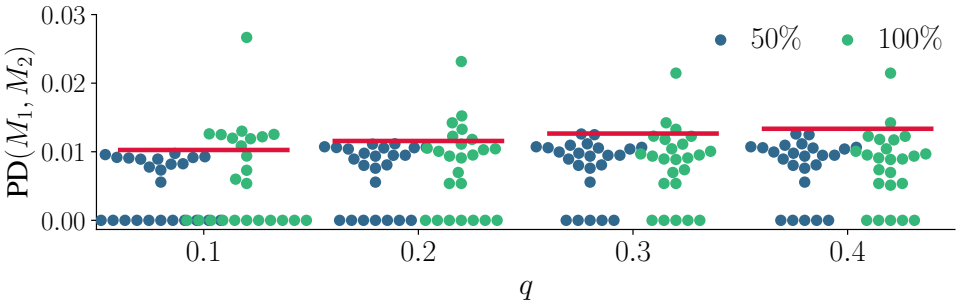


Figure 14. **Opportunities for low-complexity training.** We show the pairwise PRESTO distances between individual universes and their closest representatives in the β-VAE *FashionMNIST* hyperparameter multiverse, as assessed based on training with 50% (blue) or 100% (teal) of the training data, where the set of representatives is computed based on the 50%-multiverse. Red lines show the threshold value associated with the x-axis quantile q in the 50%-multiverse, and markers with $y = 0$ indicate self-representation. With PRESTO, we can compress the hyperparameter search space in a low-complexity training setting and perform high-complexity training only on a smaller set of representatives, with limited topological distortion.

B.2. Extended Experiments in Transformer Multiverses

Here, we provide further details on the configuration of our transformer multiverse, as well as additional experiments assessing the effect of choices in the PRESTO pipeline on our landscape norms, distances, and sensitivity scores.

B.2.1. THE TRANSFORMER MULTIVERSE

While our VAE multiverses focused on representational variability induced by hyperparameter and implementation choices, our transformer multiverse is designed to investigate both PRESTO's power as a black-box diagnostic and the impact of choices in the PRESTO pipeline on our measurements. To this end, we embed $2^{14} = 16\,384$ abstracts from four summarization datasets using six pretrained transformer models, covering different levels of textual technicality and language-model sophistication. Thus, we obtain 24 sets of $(2^{14} \times *)$-dimensional embeddings, where $* \in \{384, 768, 1\,024, 1\,536\}$ is the embedding dimensionality of the respective language model. From these embeddings, we generate projections onto $k \in [4]$ components via PCA or Gaussian random projections, using $s \in \{2^i \mid i \in \{10, \ldots, 14\}\}$ embedded samples to compute persistent homology, as well as $\pi \in \{2^i \mid i \in \{3, \ldots, 9\}\}$ projections to average landscapes when using random projections. We summarize the setup of our transformer multiverse in Table 8.

Decision θ_i	Values
Model M	ADA, DISTILROBERTA, MINILM, MISTRAL, MPNET, QA-DISTILBERT
Dataset X	*arXiv*, *bbc*, *cnn*, *patents*
Dataset sample size s	2^i for $i \in \{10, 11, 12, 13, 14\}$
Number of projection components k	$1, 2, 3, 4$
Projection method	PCA, Gaussian Random Projections
Number of Gaussian projections π	2^i for $i \in \{3, 4, 5, 6, 7, 8, 9\}$

Table 8. **Transformer Multiverse.** We work with $6 \cdot 4 = 24$ sets of embeddings, investigating each through the lens of multiple combinations of different choices involved in the PRESTO pipeline.

All datasets, as well as our four smaller transformer models, are available on HuggingFace.

(1) **Datasets.**
  (a) *arXiv*: abstracts of all arXiv articles up to the end of 2021;
  (b) *bbc*: summaries of BBC news articles;
  (c) *cnn*: summaries of news articles from CNN and DailyMail; and
  (d) *patents*: abstracts of U.S. patent applications.
  We embed the first 2^{14} samples from the designated *training* sets of these datasets with each of our models.

(2) **Models.**
  (a) DISTILROBERTA: general-purpose model, embedding dimension 768, maximum sequence length 512 word pieces;
  (b) MINILM: general-purpose model, embedding dimension 384, maximum sequence length 256 word pieces;
  (c) MPNET: general-purpose model, embedding dimension 768, maximum sequence length 384 word pieces; and
  (d) QA-DISTILBERT: QA-specialized model, embedding dimension 768, maximum sequence length 512 word pieces.
  These models are provided as pretrained by the *sentence-transformers* library (Reimers & Gurevych, 2019). They come with *normalized* embeddings, and we *truncate* the texts to be embedded in the (rare) event that they exceed a model's maximum sequence length.

To obtain embeddings from our two large language models, ADA (embedding dimension: 1 536) and MISTRAL (embedding dimension: 1 024), we query the corresponding APIs of their providers (OpenAI and MistralAI, respectively).

B.2.2. LANDSCAPE NORMS

To start, we investigate how our choice of *projector* interacts with our chosen *sample size* s, keeping the number of projection components fixed at $k = 3$. In particular, we are interested in how our landscape norms change as we vary these parameters, since landscape norms underlie both our PRESTO *distances* and our PRESTO *sensitivities*. Inspecting the distributions of landscape norms when working with Gaussian random projections in Figure 15, we see that norms are approximately normally distributed, and that *larger sample sizes* are associated with *smaller landscape-norm means* as well as *smaller landscape-norm variance*.

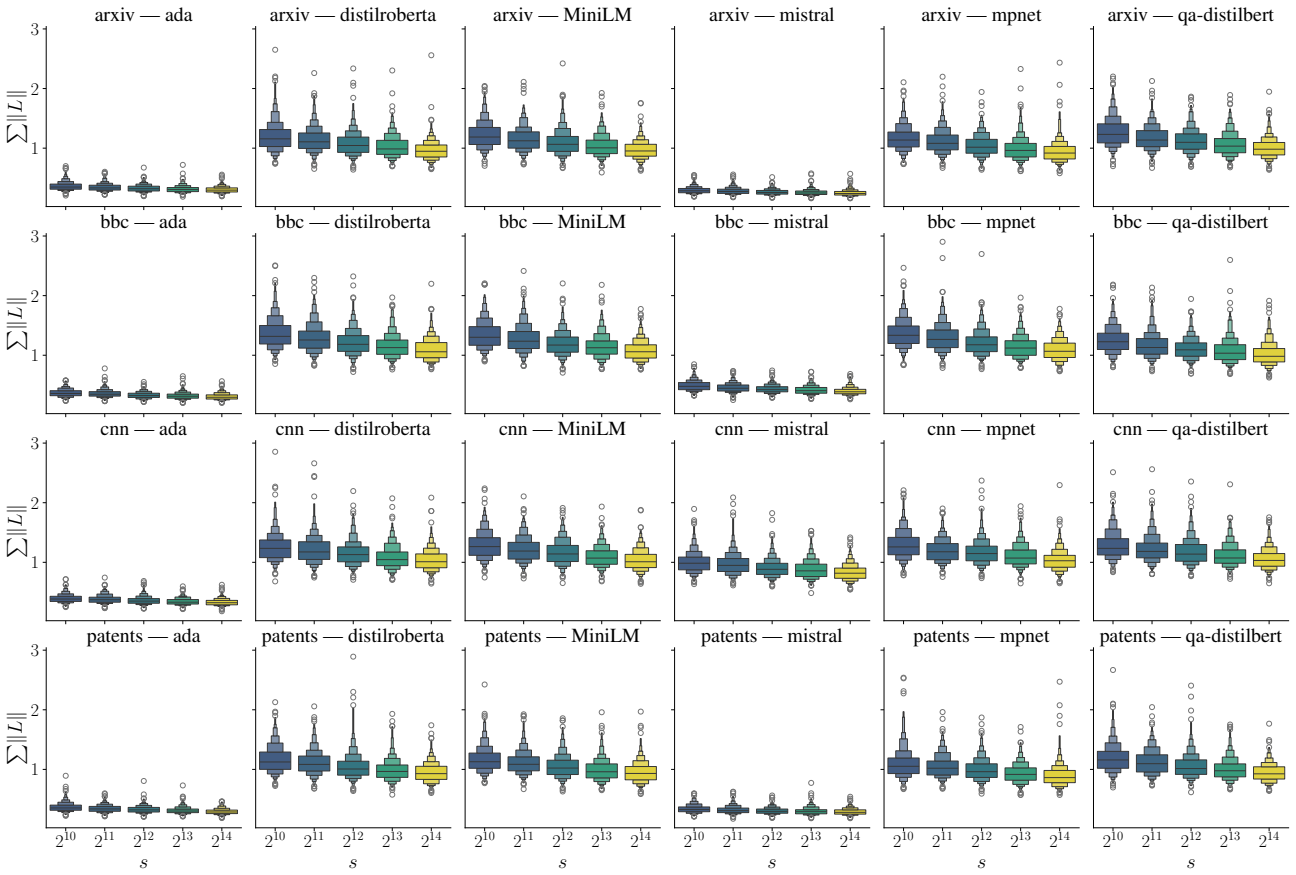


Figure 15. **Distribution of landscape norms when projecting via Gaussian random projections and varying the sample size s.** We show the distribution of landscape norms when working with $\pi = 2^9 = 512$ Gaussian random projections and varying the sample size s of the embedded dataset. Larger sample sizes are associated with smaller landscape norms, and the landscape norm distributions of large language models are clearly distinguishable from those of smaller transformer models.

Next, we compare the distributions of norms under Gaussian random projections with the fixed norms we obtain under PCA. We observe that although the norms live on different scales, with PCA-based norms being much smaller than the norms of Gaussian random projections, the change pattern when varying the sample size s is qualitatively similar, and the distinction between large language models and smaller transformer models is equally evident. Hence, while Gaussian random projections allow us explore representational variability *within* individual latent spaces, when comparing *between* latent spaces, we may use PCA as well. Thus, we fix PCA as a projector for the remainder of our experiments.

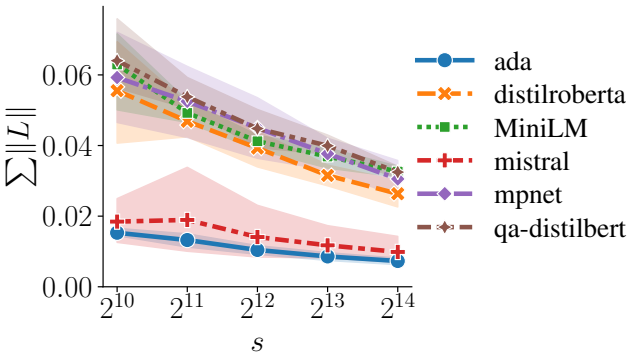


Figure 16. **Landscape norms when projecting with PCA and varying the sample size s.** We show the landscape norms of PCA-based projections as a function of the sample size s of the embedded dataset. While landscape norms obtained via PCA live on a scale which is different from that of landscape norms obtained via Gaussian Random projections, their distinguishing power and behavior under sample-size variation are qualitatively similar.

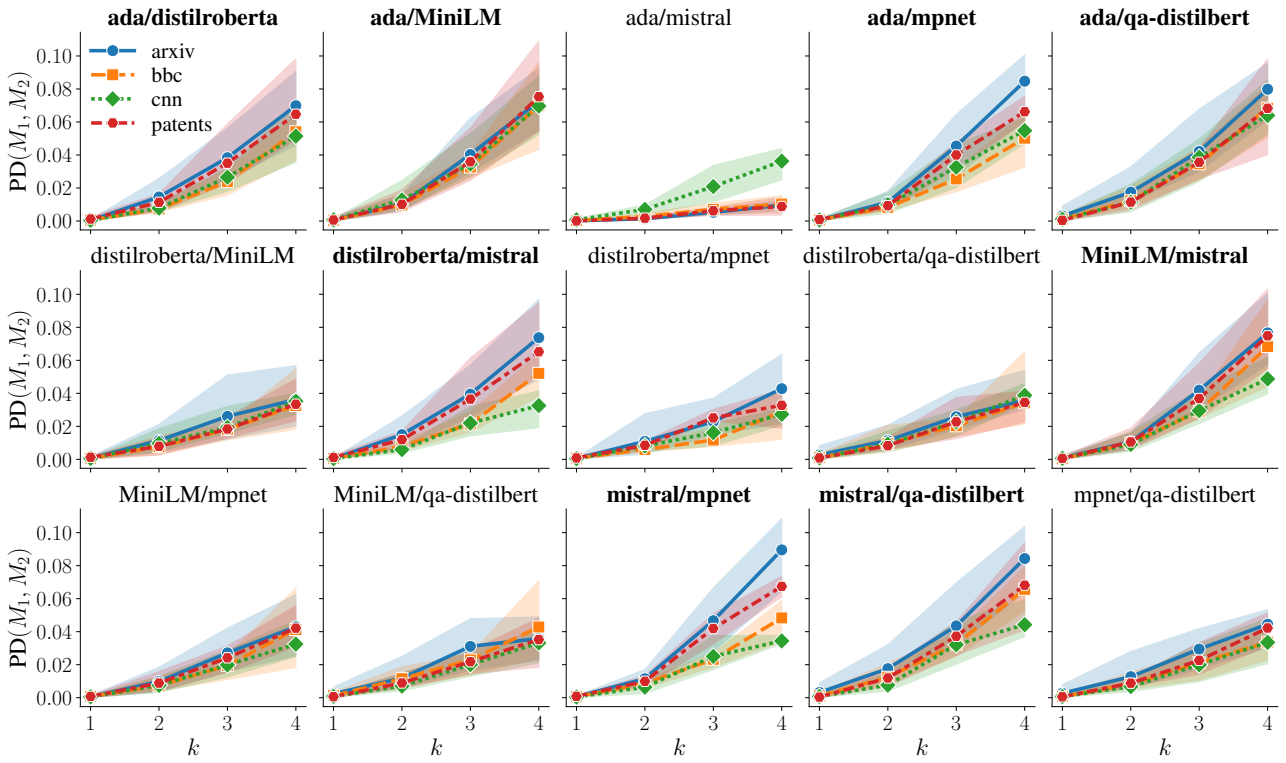


Figure 17. **PRESTO distances between transformer models.** We show the PRESTO distance between transformer models as measured on different datasets (lines) as a function of the number of projection components k, where the bands indicate the range of measurements across varying dataset sample sizes s (from 2^{10} to 2^{14}), and we typeset comparisons involving exactly one large language model in bold. For $k > 2$ projection components, PRESTO distances distinguish large language models from smaller models based on technical datasets.

B.2.3. PRESTO DISTANCES

In Figure 7, we showed the pairwise distances between the multiverse metric spaces associated with our transformer models. For a more fine-grained perspective, in Figure 17, we additionally depict the range of pairwise PRESTO distances between individual models as we vary the number of projection components k and the number of samples s considered. We see that as we increase the number of projection components, PRESTO's capacity to distinguish large language models from smaller transformer models increases. Furthermore, in many comparisons involving exactly one large language model, for $k > 2$, the PRESTO distance between the compared models is larger for our technical datasets (*arXiv*, *patents*) than for our news datasets (*bbc*, *cnn*).

B.2.4. PRESTO SENSITIVITIES

Complementing our assessment of PRESTO sensitivities in the VAE hyperparameter and implementation multiverses (Figures 6 and 10), we now investigate the local PRESTO sensitivities of models and datasets in our transformer multiverse as a function of the number of samples s and the number of projection components k. When exploring sensitivity to *model variation* in Figure 18, we observe two trends: PRESTO sensitivities become *smaller* as we increase the number of samples s, and they become *larger* as we increase the number of projection components k. These trends persist when we turn to *dataset variation*, depicted in Figure 19. Since a smaller number of samples provides a rougher picture of a model's latent space, and a larger number of projection components allows us to capture more variation, these patterns are to be expected, which further increases our confidence in PRESTO. Finally, Figure 19 reveals that the PRESTO sensitivity when varying *datasets* differs widely between *models*—and notably, PRESTO's dataset sensitivity separates our two large language models, ADA and MISTRAL (cf. Figure 19, middle panel).

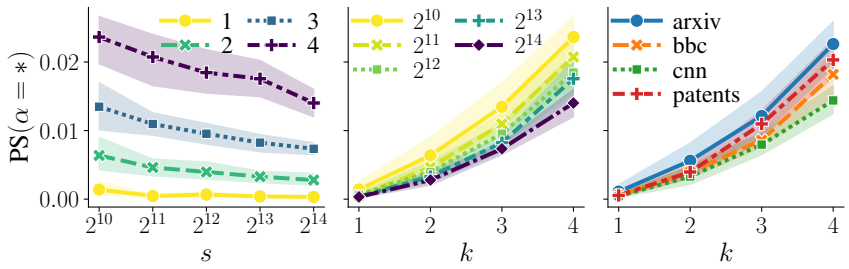


Figure 18. **Local PRESTO sensitivity of *models* in the transformer multiverse.** In the left and middle panels, we show the local PRESTO sensitivity scores of the model choice α (lines), along with the range of individual PRESTO sensitivities across our *models* (bands), as a function of the number of samples s and the number of projection components k. In the right panel, we show the local PRESTO sensitivity scores of the model choice α (lines), along with the range of individual PRESTO sensitivities across *sample sizes* s (bands), as a function of the number of projection components k and the dataset δ. The smaller the number of samples, and the larger the number of projection components, the higher the relevance of the model choice.

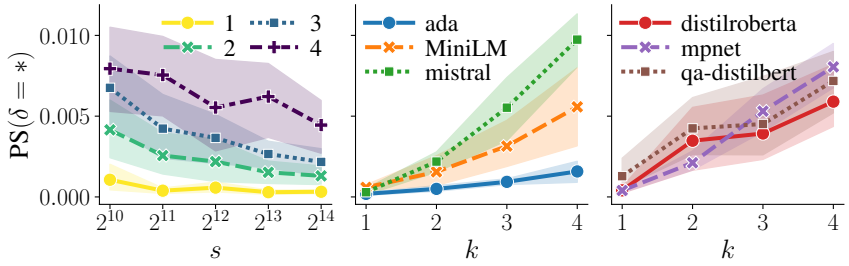


Figure 19. **Local PRESTO sensitivity of *datasets* in the transformer multiverse.** In the left panel, we show the local PRESTO sensitivity scores of the dataset choice δ (lines), along with the range of individual PRESTO sensitivities across our *datasets* (bands), as a function of the number of samples s and the number of projection components k. In the middle and right panels, we show the local PRESTO sensitivity scores of the dataset choice δ (lines), along with the range of individual PRESTO sensitivities across our *number of samples* s (bands), as a function of the number of projection components k and the transformer model α. Both the local PRESTO sensitivity and the variance in individual PRESTO sensitivities are highly dependent on the chosen transformer model.

B.3. PRESTO as a Dissimilarity Measure for Neural-Network Representations

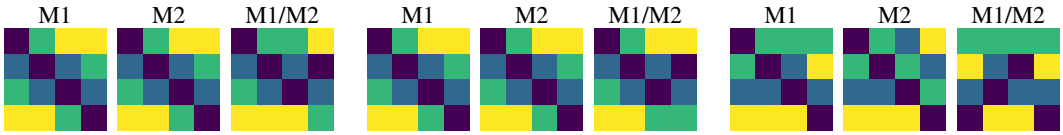
PRESTO is designed for comparisons between general latent spaces, not neural representations specifically, and we deliberately deferred an in-depth treatment of PRESTO for neural-network forensics to future work. To gauge the potential of PRESTO as a dissimilarity measure in this context, we compare PRESTO with two variants of CKA (Kornblith et al., 2019). When measured by CKA, representations in neighboring neural-network layers are consistently judged as more similar to each other than to representations in more distant neural-network layers. Our preliminary experiments using the MNIST-trained neural-network representations made publicly available in the CKA repository, summarized in Table 9, indicate that PRESTO captures this trend as well. In Figure 20, we additionally depict the similarity ranks of PRESTO (transformed into a similarity measure) as well as lCKA and kCKA for relationships between different layers in the same model (*intra-model relationships*) as well as relationships between the same or different layers of differently seeded models (*inter-model relationships*). However, we also emphasize that there is no reason to expect, a priori, that the relationships between the internal representations of a neural-network model should be quantitatively the same across differently (hyper)parameterized neural-network models. Rather, this is an interesting hypothesis that PRESTO will allow us to test going forward.

B.4. Multiverse Analysis of Non-Linear Dimensionality-Reduction Methods

Manifold-learning techniques are essential tools in various applied sciences, including computational biology and medicine, where algorithms such as UMAP, T-SNE, and PHATE (McInnes et al., 2020; Moon et al., 2019; van der Maaten & Hinton, 2008) are commonly used to generate low-dimensional embeddings of complex datasets, imbuing these computationally tractable representations with geometric and topological structure learned from the data manifold.

	Type of Correlation		
Comparison	Pearson	Spearman	Kendall
PRESTO vs. lCKA	-0.88 ($p = 0.02$)	-0.89 ($p = 0.02$)	-0.73 ($p = 0.06$)
PRESTO vs. kCKA	-0.75 ($p = 0.09$)	-0.89 ($p = 0.02$)	-0.73 ($p = 0.06$)

Table 9. **PRESTO's relationship to CKA variants.** We display the correlations between PRESTO (a dissimilarity measure) and lCKA resp. kCKA (similarity measures) for neural-network models trained on MNIST using different seeds.



(a) lCKA (b) kCKA (c) PRESTO

Figure 20. **Comparing neural activations.** We replicate an experiment by Kornblith et al. (2019), who assess the similarity of neural activations in 4-layer CNNs trained on MNIST, comparing PRESTO (turned into a similarity measure) with similarities produced by lCKA and kCKA. For each method, we show the similarity ranks for different layers in the same model (left and middle subpanels) as well as for different layers across two different models (right subpanels). Here, similarity ranks are computed column-wise, and darker colors indicate higher ranks (i.e., if in the comparison between model 1 [M1] and model 2 (M2), the square in position $[i, j]$ is colored in dark blue, layer j of M1 is most similar to layer i of M2).

These methods also exhibit representational variability that can be measured with PRESTO, especially when varying their hyperparameters. Among the most critical parameters determining the structure of a low-dimensional representation is the parameter controlling the *locality* of the dimensionality-reduction method, which manifests in various variants of a *number-of-neighbors* parameter (named differently for different algorithms). In Table 10, we use PRESTO's sensitivity scores based on the distribution of landscapes that arise from varying this locality parameter across different synthetic and real-world datasets.

	Model				
Dataset	Isomap	LLE	Phate	t-SNE	UMAP
Breast Cancer	13.41	4.59	1.52	0.21	0.34
Diabetes	0.22	51.19	0.37	0.06	0.32
Digits	0.35	0.94	0.24	0.24	1.14
Iris	0.46	9.01	0.52	0.74	1.28
Moons	0.00	0.00	1.67	1.34	0.22
Swiss Roll	1.56	0.26	1.28	0.42	0.60

Table 10. **Sensitivity of dimensionality-reduction methods.** We show the PRESTO sensitivity scores of the *number-of-neighbors* parameter for five dimensionality-reduction algorithms on six datasets. Popular dimensionality-reduction vary widely in hyperparameter sensitivity.

To further demonstrate PRESTO's utility in the manifold-learning space, in Figure 21, we cluster the latent representations arising from different combinations of dimensionality-reduction algorithms, hyperparameters, and datasets based on the multiverse metric space constructed from pairwise PRESTO distances between embeddings. We are confident that PRESTO-based tools will be useful for practitioners in the applied sciences by (1) allowing them to assess the *sensitivity* of their algorithms and datasets with respect to their hyperparameter choices, and (2) helping them condense the multiverse of representations into a manageable set of structurally distinct representatives via hyperparameter *compression*.

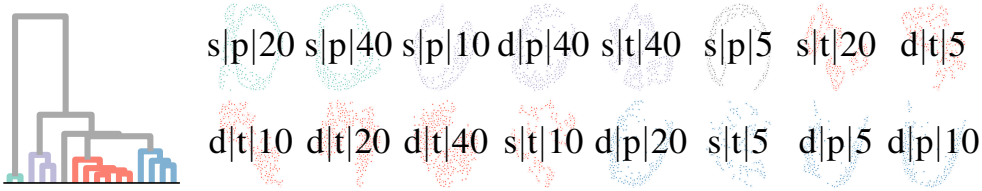


Figure 21. **Clustering embeddings with PRESTO.** We show the dendrogram of a complete-linkage clustering of 2-dimensional embeddings based on PRESTO distances (left), as well as the clustered embeddings, colored by their cluster when cutting the dendrogram at distance 1 (right). Annotations are of shape dataset|method|n, where dataset $\in$ {swiss roll (s), diabetes (d)}, method $\in$ {phate (p), t-SNE (t)}, and $n \in \{5, 10, 20, 40\}$ is the nearest-neighbors parameter of both methods. With PRESTO, we can compare potentially unaligned embeddings originating from different datasets and methods.

C. Extended Methods

In this section, we provide details on the computational complexity of topological multiverse analysis and detail how we can pick good representatives for (hyperparameter-)search-space compression.

C.1. PRESTO Complexity

C.1.1. THEORETICAL ANALYSIS

The individual PRESTO steps of the PRESTO pipeline exhibit the following computational complexities.

(S1) **Embed data.** The complexity of computing $E = M(X) \in \mathbb{R}^{s \times *}$ for $X \in \mathbb{R}^{s \times *}$ depends only on X and M, and hence, is independent of our specific design choices.

(S2) **Project embeddings.** Given embedding $E \in \mathbb{R}^{s \times d}$, computing k-dimensional PCA takes $\mathcal{O}(ksd)$ time via truncated SVD, while generating π random projections of E onto k dimensions takes $\mathcal{O}(\pi ksd)$ time.

(S3) **Construct persistence diagrams.** When constructing an h-dimensional α-complex from s points, the bottleneck is computing the Delaunay triangulation, which takes worst-case expected time $\mathcal{O}(s^{\lceil h/2 \rceil + 1})$ but often runs in $\mathcal{O}(s \log s)$ time in practice.

(S4) **Compute persistence landscapes.** If done exactly, computing a persistence landscape from s birth-death pairs takes $\mathcal{O}(s^2)$ time. However, at the cost of a small perturbation in our persistence diagrams, we can round birth and death times to lie on a grid of constant size, such that we can obtain persistence landscapes in $\mathcal{O}(s \log s)$ time (Bubenik & Dłotko, 2017). When working with random projections, averaging π *exact* persistence landscapes takes $\mathcal{O}(s^2 \pi \log \pi)$ time, whereas averages of *approximate* landscapes can be computed in $\mathcal{O}(s\pi)$ time (Bubenik & Dłotko, 2017).

Hence, for a constant number of latent dimensions d and (if applicable) projections π, the topology-based steps of the PRESTO pipeline can be performed in $\widetilde{\mathcal{O}}(s)$ time, i.e., our computations are approximately linear in the number of samples in X. This also holds for the computation of our PRESTO *primitives*, i.e., the PRESTO *distance* (PD) and the PRESTO *variance* (PV). Overall, we obtain a scalable toolkit for the topological analysis of representational variability in latent-space models.

C.1.2. EMPIRICAL ANALYSIS

We supplement our theoretical analysis with PRESTO's empirical running times, detailed in Table 11. Note that these running times are based on a single-CPU implementation. Various optimization and parallelization strategies for persistent-homology calculations and diameter approximations exist, and PRESTO can benefit from them directly.

	d		
s	128	256	512
2^{12}	0.30 ± 0.03	0.30 ± 0.01	0.34 ± 0.04
2^{14}	1.16 ± 0.02	1.20 ± 0.04	1.26 ± 0.03
2^{16}	5.36 ± 0.28	5.28 ± 0.11	7.26 ± 0.88
2^{18}	25.86 ± 1.31	28.43 ± 2.04	36.26 ± 9.70

(a) PRESTO without Normalization

	d		
s	128	256	512
2^{12}	0.59 ± 0.04	0.69 ± 0.03	0.92 ± 0.02
2^{14}	2.35 ± 0.04	2.84 ± 0.06	4.01 ± 0.14
2^{16}	10.45 ± 0.34	12.32 ± 0.41	38.20 ± 0.95
2^{18}	67.54 ± 2.27	154.47 ± 36.00	156.71 ± 45.48

(b) PRESTO with Normalization

Table 11. **Empirical running times of PRESTO.** We report the average running times (seconds) of computing PRESTO distances across random embeddings of varying sizes on a single CPU. We compute 10 different pairs of randomly seeded embeddings for each size (s, d). We project the embeddings using PCA into 2 dimensions and fit our landscapes using Alpha Complexes for homology dimensions 0 and 1.

C.1.3. RUNTIME COMPARISON WITH OTHER METHODS

Unlike other methods, PRESTO can compare latent spaces with hundreds of thousands of samples embedded into large latent dimensions on commodity hardware. In the age of large LLM embeddings, our method allows users with limited computational resources to run PRESTO quickly and without using paralyzing amounts of memory. In Table 12, we compare the running times of computing PRESTO distances with the running times of computing several other (dis)similarity measures. Here, "Pairwise" refers to computing basic pairwise distances in the high-dimensional space, "VR" refers to constructing a Vietoris-Rips complex based on these distances, and "IMD" (Intrinsic Multi-scale Distance) refers to the method by Tsitsulin

s	d	PRESTO (no norm.)	PRESTO (norm.)	CKA	Pairwise	VR	IMD
2^{12}	128	0.308	0.721	22.568	0.23	3.113	2.692
2^{12}	256	0.302	1.149	24.643	0.446	5.558	4.772
2^{12}	512	0.313	2.184	27.689	0.692	9.641	6.633
2^{14}	128	1.236	2.816	2 601.56	9.614	170.41	33.968
2^{14}	256	1.197	4.652	2 646.33	10.649	220.507	64.061
2^{14}	512	1.248	11.096	2 445.32	18.081	272.997	101.336

Table 12. **Comparative running times.** We compare PRESTO to other representational similarity methods, reporting the run times (seconds) for PRESTO (with and without normalization), CKA, Pairwise Distances (`scikit-learn`), Vietoris Rips (`guhdi`), and IMD on a pair of random embeddings of varying sizes on a single CPU. PRESTO times are averaged over 10 pairs of embeddings.

et al. (2020). We do not include running times for Representation Topological Divergence (RTD) (Trofimov et al., 2023) because it is designed specifically for GPUs. Furthermore, we note that PRESTO's computational complexity in large spaces hinges on the computation of alpha complexes, persistence landscapes, and diameter approximations (when normalizing spaces). These have the potential to be accelerated and parallelized nicely (Chazal et al., 2014b; 2015). We leave the integrations and analyses of these optimizations to future work and will reflect updates in our open-source implementation.

C.2. PRESTO Compression

When using PRESTO for search-space compression, our goal is to select a small set of *representatives* $\mathcal{R}$ (e.g., hyperparameter vectors) to explore in detail such that each universe in the original search space has a representative in the compressed search space at topological distance no larger than ϵ. To obtain an efficient set of such representatives (i.e., a small set of configurations that together satisfy our topologically-dense-sampling criterion), we can take two approaches. First, we can *cluster* a multiverse based on the pairwise topological distances between its universes using a method that bounds intra-cluster distances (e.g., using agglomerative complete-linkage clustering and cutting the dendrogram at height ϵ) and pick one representative from each cluster. Alternatively, to bound the size of the representative set as a function of its minimum size, we can interpret topologically dense sampling as a *set-cover problem* (or equivalently, a *hitting-set problem*), where our universes are both the elements (to be represented themselves) and the candidate sets (representing themselves and others). While minimum set cover is NP-hard (Karp, 1972), the simple *greedy approximation algorithm* that picks the candidate universe capable of representing the largest number of unrepresented universes can guarantee that our set of representatives has size at most $|\mathcal{R}| \leq \mathcal{H}(m)c^*$, where $m \coloneqq |\mathcal{M}|$ is the size of our multiverse, $\mathcal{H}(m) \in \mathcal{O}(\log m)$ denotes the m-th harmonic number, and c^* is the minimum number of representatives needed to cover all configurations in $\mathcal{M}$.

D. Extended Background

This section provides some background information on *latent-space models*, the objects of our variability assessment. We provide further details on two categories of latent-space models that are particularly relevant to our work: *generative models* and *representation-learning algorithms*. However, our framework can be applied to *any* model that uses embeddings.

D.1. Generative Models

Generative models, such as those developed by Vaswani et al. (2017) or Goodfellow et al. (2014), are at the forefront of deep-learning research, enabling the synthesis of new data as well as complex data transformations such as *style transfer*. They rely on *embeddings*, i.e., learned low-dimensional representations of data, to drive their generative capabilities. The geometric relationships within the context of the latent space are learned by the model during training, becoming a cornerstone of the model's characteristics as a generator. Here, we discuss *variational autoencoders* (VAEs) as the class of generative models featuring most prominently in our experiments. Originally developed by (Kingma & Welling, 2013), VAEs are probabilistic models that learn a generative distribution $p(x, z) = p(z)p(x|z)$, where $p(z)$ represents a prior distribution over the latent variable z, and $p(x|z)$ is the likelihood function responsible for generating the data x given z. VAEs are trained with the objective of maximizing a variational lower bound $L_{\text{VAE}}(x)$ on the log-likelihood $\log p(x)$, subject to the inequality $\log p(x) \geq L_{\text{VAE}}(x)$. The expression for the variational lower bound of a VAE is

$$L_{\text{VAE}}(x) = \mathbb{E}_{q(z|x)}[\log p(x|z)] - \text{KL}(q(z|x)||p(z)) \ , \tag{34}$$

where $q(z|x)$ stands for an approximate posterior and KL denotes the Kullback–Leibler divergence (Joyce, 2011). Here, the term $\mathbb{E}_{q(z|x)}[\log p(x|z)]$ represents the expected log-likelihood of x given z under the approximate posterior $q(z|x)$, while $\mathrm{KL}(q(z|x)||p(z))$ quantifies the divergence between $q(z|x)$ and the prior $p(z)$. The ultimate objective during the training of VAEs is to maximize the expected lower bound $\mathbb{E}_{p_d(x)}[L_{\mathrm{VAE}}(x)]$, where $p_d(x)$ is the data distribution.

D.2. Representation-Learning Algorithms

Representation-learning algorithms leverage salient features of the input data to obtain structure-preserving representations. Many (though not all) representation-learning algorithms are based on the *manifold hypothesis*, which posits that data are sampled from an (unknown) low-dimensional latent manifold. Even in light of a compelling and ongoing line of research that questions the integrity of the manifold hypothesis for certain datasets (Brown et al., 2023; Scoccola & Perea, 2023; von Rohrscheidt & Rieck, 2023), the need to embed data into low-dimensional latent spaces remains a key aspect for both data preprocessing and the development of generative models. Though linear methods like Principal Component Analysis (PCA) are incredibly useful in their own right, nowadays, significant emphasis is placed on *non-linear dimensionality-reduction algorithms* (NLDR algorithms) as the most prominent methods for preserving salient geometric relationships. These algorithms are maps $f_{\mathrm{red}}\colon \mathbb{R}^D \to \mathbb{R}^d$, typically with $d \ll D$, that make use of local geometric information in a high-dimensional space to estimate properties of the underlying manifold. When mapping into the latent space, such algorithms often aim to maintain *pairwise geodesic distances* between neighboring points. Some of the most widely used NLDR algorithms include `UMAP` by (McInnes et al., 2020), `PHATE` by (Moon et al., 2019), `t-SNE` by (van der Maaten & Hinton, 2008), `Isomap` by (Tenenbaum et al., 2000), and `LLE` by (Roweis & Saul, 2000).

Although the precise behavior of the mapping is unique to individual implementations, many algorithms rely on estimating local properties of the data via k-nearest-neighbor graphs, treating k as the *locality scale*, also known as the `n-neighbors` parameter. In combination with the other hyperparameters (if any), this helps the algorithm obtain a final representation of the crucial geometric relationships in the data. For unsupervised tasks, this raises a non-trivial decision: *What is the correct scale at which to probe a particular dataset?* In the absence of labels, the answer to this question is highly context-specific: The most insightful geometric relationships cannot be known *a priori*. Here, we can use PRESTO to understand the multiverse of representations that arise from the various algorithmic choices, implementation choices, and data choices involved in non-linear dimensionality reduction. As demonstrated for VAEs and transformers in the main text, PRESTO can describe the distribution of embeddings arising from multiverse considerations and quantify representational variability in NLDR. See Appendix B.4 for supplementary experiments on hyperparameter sensitivity in NLDR algorithms and clustering of embeddings across different algorithmic, hyperparameter, and data choices.

E. Extended Related Work

In addition to the related work mentioned in the main paper, which directly deals with representational variability in one way or the other, our analyses and definitions draw upon a wealth of additional research in *topological data analysis*. Here, seminal works by Bubenik (2015) and Adams et al. (2017) introduce *persistence landscapes* and *persistence images*, respectively, opening the door toward more efficient topological descriptors that can be gainfully deployed in a machine-learning setting. We focus on *persistence landscapes* in this paper since they do not require any additional parameter choices, but our pipeline remains valid for *persistence images*. Persistence landscapes have the advantage that more of their statistical behavior has been studied (Bubenik & Dłotko, 2017), and recent work even shows that they capture certain geometric properties of spaces (Bubenik et al., 2020). This consolidates their appeal as an expressive shape descriptor of data.

Beyond research in topological data analysis, several works address the backbone of our pipeline, i.e., *persistent homology*, directly. Cohen-Steiner et al. (2007) prove the seminal stability theorem upon which most of the follow-up work is based, and Chazal et al. (2015) establish the foundation for understanding the behavior of topological descriptors (including persistence landscapes) under subsampling. Furthermore, Chazal et al. (2014a) show that all geometric constructions like the Vietoris–Rips complex lead to stable outcomes in the sense that geometric variation always provides an upper bound on topological variation. We will subsequently make use of this seminal result to motivate the stability and choice of metrics.

Finally, topological approaches have also shown their utility in the context of studying individual neural-network models. Of particular interest in current research are the investigation of representational similarities (Klabunde et al., 2023) or the analysis of particular parts of a larger model, such as attention matrices (Smith et al., 2023). This strand of research is motivated by insights into how understanding geometrical-topological characteristics of data and models can lead to improvements in certain tasks (Chen et al., 2019; van der Merwe et al., 2022)—or, specifically, how the topology of data can be used to characterize the *loss landscape* of a model (Freeman & Bruna, 2017; Horoi et al., 2022).