GeomCA: Geometric Evaluation of Data Representations

Petra Poklukar [1]   **Anastasia Varava** [1]   **Danica Kragic** [1]

Abstract

Evaluating the quality of learned representations without relying on a downstream task remains one of the challenges in representation learning. In this work, we present Geometric Component Analysis (GeomCA) algorithm that evaluates representation spaces based on their geometric and topological properties. GeomCA can be applied to representations of any dimension, independently of the model that generated them. We demonstrate its applicability by analyzing representations obtained from a variety of scenarios, such as contrastive learning models, generative models and supervised learning models.

1. Introduction

Efficient data representations have been shown to improve machine learning models in numerous domains such as supervised and transfer learning (Oneto et al., 2020; Wang et al., 2020), density estimation (Kirichenko et al., 2020), reinforcement learning (Ghosh & Bellemare, 2020), to name a few. Significant progress has been made on learning representations with different structures, for example disentangled (Pfau et al., 2020; Kim & Mnih, 2018), in the form of a specific manifold or curvature (Arvanitidis et al., 2018; 2020; Moor et al., 2020; Schönenberger et al., 2020), or with particular similarity as learned by contrastive learning algorithms (Le-Khac et al., 2020). In general, data representations are desirable not only due to their low dimensionality but also because they enable measuring meaningful distances, which is especially critical in noisy data or visual data such as images and videos.

The quality of learned representations is typically determined by their performance on a specific downstream task. For example, disentanglement is determined by the accuracy of classifiers trained to predict the underlying factors of variation present in the dataset (Higgins et al., 2017; Kim & Mnih, 2018; Locatello et al., 2019). In reinforcement

[1]KTH Royal Institute of Technology, Stockholm, Sweden. Correspondence to: Petra Poklukar <poklukar@kth.se>.

learning and robotics, usefulness of representations is evaluated on the performance of the policy (Ghadirzadeh et al., 2020; Laskin et al., 2020) and the robotics task (Lippi et al., 2020), respectively. However, such evaluation favors representations that are tied to the downstream task, making them difficult to be generalized across variety of tasks.

A more general way to evaluate representations is to analyze how well their *global structure*, i.e., their geometric and topological properties, reflect the underlying structure of the data manifold. This direction has been recently explored in the context of Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) where it is challenging to define an appropriate downstream task for evaluation. Recently proposed methods, such as Improved Precision and Recall (IPR) (Kynkäänniemi et al., 2019) and Geometry Score (GS) (Khrulkov & Oseledets, 2018), have shown success in detecting failure cases of GANs but provide little insight into the actual *structure* of learned representations, thus hindering further investigation of local areas of the representation space where the potential failures arise.

Figure 1. Example of two images of different class label in the ImageNet dataset belonging to the same connected component in the VGG16 representation space.

In this work, we present a method for evaluating the structure of both the entire representation space and any of its connected components. We achieve this by comparing two discrete sets representing the true data manifold: the reference representation set R and the evaluation representation set E. Depending on the application, R and E can consist of raw data points or their features obtained from a neural network. Intuitively, if the structure of evaluated representations E is well aligned with the structure of the reference

representations R, then E well represents the underlying data manifold.

Our method, called Geometric Component Analysis (GeomCA), uses graphs to extract the structure of R and E, and analyzes the degree of their alignment. In contrast to other closely related methods such GS and IPR, we analyze the alignment not only on a *global* level, but also on a *local* level by analyzing the location and correspondence between the connected components of R and E. We demonstrate that GeomCA can detect outliers and connected components in E that are not present in R, as well as identify the co-ordinates of individual points from any component, thus enabling their visualization.

We apply GeomCA to different practical setups. First, we consider a contrastive learning scenario and evaluate the structural similarity between the encodings belonging to different classes of the training and validation datasets (Section 4). Second, we evaluate generative models by comparing the connected components of the training and generated datasets (Section 5). Finally, we apply GeomCA to investigate if features extracted by a supervised model are separated according to their respective classes (Section 6). For instance, Figure 1 shows two images belonging to different classes in the ImageNet dataset (Deng et al., 2009) that are close to each other in the feature space of VGG16 (Liu & Deng, 2015).

In summary, our contributions are: *(i)* we present GeomCA for assessing the quality of data representations by leveraging their geometric and topological properties (Sections 2, 3), and *(ii)* we experimentally demonstrate valuable insights provided by GeomCA on representations obtained from various models and scenarios (Sections 4, 5, 6).

2. Method

In this section, we introduce the proposed GeomCA algorithm. We present its intuitive idea in Section 2.1, its technical details in Section 2.2 and its improvements over the existing closely related methods in Section 2.3.

2.1. Intuitive Idea

The basic idea of the GeomCA algorithm is to compare the global properties (topology) and local properties (geometry) of two sets of representations, R and E, representing the underlying true data manifold $\mathcal{M}$. We say that E is a good representation of $\mathcal{M}$ if it is well aligned with the reference representation R. We aim to detect areas where R and E are coherent and quantify their alignment, as well as detect isolated individual representations (outliers) or groups of points from only one of the sets R or E. In summary, we wish to answer the following two questions:

(Q1) Do R and E have the same number of connected components, and are their sizes comparable? [Topology]

(Q2) How much do the connected components of R and E overlap? [Geometry]

We find the connected components of R and E by building ε-*threshold graphs*, or in short, ε-graphs. In an ε-graph, two points are connected by an edge if they are less than the given threshold ε apart[1]. This allows us to immediately answer (Q1): we build ε-graphs $\mathcal{G}^R, \mathcal{G}^E$ on R and E, respectively, and compare the number of their connected components as well as their sizes. Examples of ε-graphs for $\varepsilon_1 < \varepsilon_2 < \varepsilon_3$ are visualized in Figure 2. In the left panel, R (in blue) and E (in orange) both have 6 connected components composed of only one point each. When increasing ε, edges among them start emerging (colored with respective color of the set). In the middle panel, R has four connected components of size 1 and one of size 2, while E has three connected components of size 1, 2 and 3.

To answer (Q2), we additionally need to quantify the alignment of these connected components. Intuitively, this can be measured in terms of edges connecting R and E. In Figure 2, we visualized such edges in gray color. In the middle panel, the two R components in top right area are well aligned with the largest E component. This area increases for a larger ε shown in the right panel.
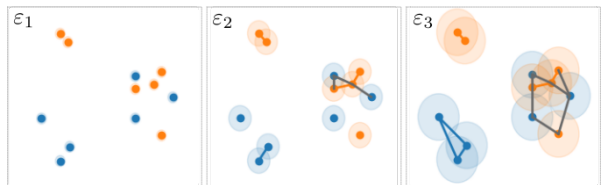

Figure 2. Examples of ε-graphs obtained for $0 < \varepsilon_1 < \varepsilon_2 < \varepsilon_3$ built on the sets R (blue) and E (orange).

We connect points from R and E using the same ε threshold as in $\mathcal{G}^R$ and $\mathcal{G}^E$. This is equivalent to building an ε-graph $\mathcal{G}^{R \cup E}$ on the union $R \cup E$. In this way, studying the alignment of $\mathcal{G}^R$ and $\mathcal{G}^E$ translates to studying the nature of the connected components of $\mathcal{G}^{R \cup E}$. If R and E are well aligned, all the connected components of $\mathcal{G}^{R \cup E}$ are "well mixed". This means, equally *well represented* by points from R and E which are, in turn, also *well geometrically positioned*. For example, in Figure 3 (a) and (b), both $\mathcal{G}^{R \cup E}$ graphs have two connected components that are equally well represented by R and E. However, in (b), R and E in the

---

[1] Vertex sets of graph-connected components in an ε-graph are equivalent to clusters obtained from the DBSCAN clustering algorithm (Ester et al., 1996).

outer component are not well mixed but rather concatenated. On contrary, in (c), the two components containing both R and E points are well geometrically aligned but none of the components are equally well represented by R and E.

In summary, we evaluate the topological and geometric properties of R and E by investigating the connected components of the ε-graph $\mathcal{G}^{R\cup E}$. We answer (Q1) by analyzing the nature of the vertices in each component, and (Q2) by analyzing the nature of the edges. This intuitive idea is the main driver behind GeomCA, which we rigorously define in the following section.
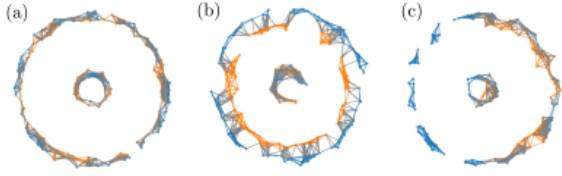

Figure 3. Examples of 2-dimensional points representing the set R (blue) and E (orange) arranged in components of different consistency and quality.

2.2. GeomCA Algorithm

Let $X = \{x_i\}_{i=1}^{n_X} \subset \mathbb{R}^M$ be a dataset of observations and let $Z = \{z_i\}_{i=1}^{n_X} \subset \mathbb{R}^N$ denote their representations obtained from any model $\mathbb{M}$, i.e., $Z = \mathbb{M}(X)$. In a machine learning setup, it is commonly assumed that $N \ll M$, although this is not necessary for GeomCA to work. Let $R = \{z_i\}_{i=1}^{n_R}$ and $E = \{z_i\}_{i=1}^{n_E}$ be two subsets of representations in Z for which we assume that $n_R + n_E \leq n_X$ and $R \neq E$. The latter assumption eliminates the case where R and E are perfectly aligned. While GeomCA provides most insight into representations when $R \cap E = \emptyset$, a non-empty intersection might be desirable in situations where it is important to investigate deviations from the intersection $R \cap E$.

As intuitively explained in Section 2.1, the idea of GeomCA is to analyze the alignment of R and E using ε-threshold graphs defined below.

Definition 2.1 *An ε-threshold graph, or ε-graph, on the set of points W with respect to the radius $\varepsilon > 0$ is a graph $\mathcal{G}_\varepsilon(W) = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V} = W$ and edges $\mathcal{E} = \{e_{ij} = (v_i, v_j) \in \mathcal{V} \times \mathcal{V} \,|\, d(v_i, v_j) < \varepsilon\}$.*

We built an ε-graph $\mathcal{G}_\varepsilon(R \cup E)$ on the union $R \cup E$. We discuss the choice of the radius ε in Section 3.1. In the remainder of this section, we will refer to the graph $\mathcal{G}_\varepsilon(R \cup E)$ simply as $\mathcal{G}$, and denote its connected components by $\mathcal{G}_i$ such that $\mathcal{G} := \sqcup_i \mathcal{G}_i$. Moreover, we define a restriction of a graph $\mathcal{H}$ to a subset W to be the subgraph $\mathcal{H}^W \subset \mathcal{H}$ with vertex and edge sets restricted to W. For example, a

connected component $\mathcal{G}_i$ restricted to the set R is a graph $\mathcal{G}_i^R$ obtained by removing all E points from the vertex set as well as all the edges from and to them from the edge set of $\mathcal{G}_i$. Lastly, we denote by $|\mathcal{H}|_\mathcal{V}$ and $|\mathcal{H}|_\mathcal{E}$ the cardinalities of the vertex set and edge set of a graph $\mathcal{H}$, respectively.

Our algorithm (summarized in Algorithm 1) consists of a *local evaluation* and a *global evaluation* phase. The former evaluates how well the connected components of $\mathcal{G}$ are represented by R and E, while the latter evaluates the alignment of R and E on the level of the entire graph $\mathcal{G}$. We describe each of these phases in detail in the following.

Local evaluation The goal of this phase is to analyze the connected components of $\mathcal{G}$. As mentioned in Section 2.1, we study their geometric properties with respect to the sets R and E. In particular, we study their (i) vertex heterogeneity determined by the ratio of representations from R and E contained in them, and (ii) heterogeneity of edges among these vertices. We refer to these geometric properties as *consistency* and *quality* of connected components, respectively, and rigorously define them in the following.

Definition 2.2 *Consistency c of a component $\mathcal{G}_i$ is defined as the ratio*

$$c(\mathcal{G}_i) = 1 - \frac{|\,|\mathcal{G}_i^R|_\mathcal{V} - |\mathcal{G}_i^E|_\mathcal{V}\,|}{|\mathcal{G}_i|_\mathcal{V}}. \tag{1}$$

A component $\mathcal{G}_i$ attains high consistency score $c(\mathcal{G}_i)$ if it contains equally many representations from R and E, i.e., if $|\mathcal{G}_i^R|_\mathcal{V} \approx |\mathcal{G}_i^E|_\mathcal{V}$. We call such component *consistent*. On contrary, $c(\mathcal{G}_i)$ is low if $\mathcal{G}_i$ is dominated either by points from R or E, in which case $\mathcal{G}_i$ is said to be *inconsistent*. In Figure 3, panels (a) and (b) contain examples of consistent components, while (c) shows inconsistent ones consisting only of points from R. However, as seen in (b), consistency itself is not a sufficient measure as it fails to detect cases where R and E are consistent but not well geometrically positioned. This is measured by component quality determined by the number of edges among representations from R and E as defined below.

Definition 2.3 *Quality of a component $\mathcal{G}_i$ is defined as the ratio*

$$q(\mathcal{G}_i) = \begin{cases} 1 - \frac{(|\mathcal{G}_i^R|_\mathcal{E} + |\mathcal{G}_i^E|_\mathcal{E})}{|\mathcal{G}_i|_\mathcal{E}} & if\ |\mathcal{G}_i|_\mathcal{E} \geq 1, \\ 0 & otherwise. \end{cases} \tag{2}$$

A component $\mathcal{G}_i$ attains high quality score, if it exhibits good connectivity among representations from R and E it contains, i.e., if both $|\mathcal{G}_i^R|_\mathcal{E}$ and $|\mathcal{G}_i^E|_\mathcal{E}$ are small. We call the edges connecting R and E *heterogeneous* and a component with high number of heterogeneous edges to be of *high quality*. On the other hand, if edges in $\mathcal{G}_i$ exist only among

points from one of the sets R or E, the component achieves low quality score and is said to be of *low quality* and its edges *homogeneous*. In Figure 3 (a) and (b), the connected components are consistent but only the ones in (a) are also of high quality (visualised by large number of gray edges). On contrary, the components in (c) are inconsistent but the largest component in fact has many heterogeneous edges, thus attaining high quality score.

Algorithm 1 GeomCA

Require: sets of representations R and E
Require: component consistency thresholds η_c
Require: component quality threshold η_q
Require: Distance threshold ε
$\quad \mathcal{G} \leftarrow \texttt{build\_epsilon\_graph}(R, E)$
$\quad$ **[Phase: Local evaluation]**
$\quad \mathcal{C} \leftarrow \texttt{get\_connected\_components}(\mathcal{G})$
$\quad \mathcal{Q}_{\text{local}} \leftarrow \texttt{zeros(len}(\mathcal{C})\texttt{, 2)}$
$\quad$ **for** $i = 0, \dots, \texttt{len}(\mathcal{C})$ **do**
$\quad\quad \mathcal{G}_i \leftarrow \mathcal{C}[i]$
$\quad\quad$ compute $c(\mathcal{G}_i)$ as in Definition 2.2
$\quad\quad$ compute $q(\mathcal{G}_i)$ as in Definition 2.3
$\quad\quad \mathcal{Q}_{\text{local}}[i, :] \leftarrow [c(\mathcal{G}_i), q(\mathcal{G}_i)]$
$\quad$ **end for**
$\quad$ **[Phase: Global evaluation]**
$\quad$ compute $c(\mathcal{G})$ and $q(\mathcal{G})$ as in Definition 2.4
$\quad$ compute $\mathcal{P}, \mathcal{R}$ with respect to η_c, η_q as in Definition 2.5
$\quad \mathcal{Q}_{\text{global}} \leftarrow [\mathcal{P}, \mathcal{R}, c(\mathcal{G}), q(\mathcal{G})]$
Return: $\mathcal{Q}_{\text{global}}, \mathcal{Q}_{\text{local}}$

Global evaluation The consistency and quality measures can be also used in several ways to obtain global scores over the entire ε-graph. First, we simply generalize Definitions 2.2 and 2.3 to $\mathcal{G}$.

Definition 2.4 *We define $c(\mathcal{G})$ as network consistency, and $q(\mathcal{G})$ as network quality.*

The global network consistency and quality are important measures to detect imbalances between the sets R and E. This is especially applicable in large-scale experiments where the sizes of R and E are reduced for computational purposes. We discuss such reduction in Section 3.2 and demonstrate the usefulness of network consistency and quality measures in these situations in Section 5.

Next, we exploit the components of certain consistency and quality to retrieve two more global scores: precision and recall. These are determined by the fraction of points from one set contained in specific components of $\mathcal{G}$.

Definition 2.5 *Let $\eta_c, \eta_q \in [0, 1]$ be real numbers. Let*

$$\mathcal{S}(\eta_c, \eta_q) = \bigcup_{\substack{q(\mathcal{G}_i) > \eta_q, \\ c(\mathcal{G}_i) > \eta_c}} \mathcal{G}_i \tag{3}$$

be the union of the connected components $\mathcal{G}_i$ with the minimum consistency and quality scores determined by η_c and η_q, respectively. Let $\mathcal{S}(\eta_c, \eta_q)^R$ and $\mathcal{S}(\eta_c, \eta_q)^E$ denote the restrictions of $\mathcal{S}(\eta_c, \eta_q)$ to the sets R and E, respectively. We define precision $\mathcal{P}$ and recall $\mathcal{R}$ with respect to η_c, η_q as

$$\mathcal{P} = \frac{|\mathcal{S}^E|_{\mathcal{V}}}{|\mathcal{G}^E|_{\mathcal{V}}} \quad \mathcal{R} = \frac{|\mathcal{S}^R|_{\mathcal{V}}}{|\mathcal{G}^R|_{\mathcal{V}}}, \tag{4}$$

respectively, where we omitted the explicit dependency on η_c, η_q for simplicity.

The thresholds η_c and η_q determine the level of alignment between the sets R and E that we wish to consider, and therefore enable to easily focus our analysis on the connected components of the desired quality and consistency. A high value of η_c requires the components to be consistent while a high value of η_q additionally requires the components to have large number of heterogeneous edges. The effect of these thresholds is demonstrated in Appendix A.1.

2.3. Comparison with Closely Related Methods

Our method is in spirit closest to Geometry Score (GS) (Khrulkov & Oseledets, 2018), and in implementation to Improved Precision and Recall score (IPR) (Kynkäänniemi et al., 2019). Both of these methods were developed for evaluation of generative models and therefore also use a reference set R consisting of training data, and an evaluation set E consisting of the generated data. GS first estimates the manifolds described by R and E using Witness complexes, and then compares their topological properties using persistent homology. The comparison is based on Relative Living Times (RTL) of homology derived from persistence barcodes in a probabilistic form. Because of Witness complexes, GS relies on repetitive subsampling to obtain a stable estimate. GeomCA instead compares topological properties of R and E by analyzing an ε-graph which is equivalent to the 1-skeleton of a Vietoris-Rips graph at the given threshold ε. In contrast to GS, GeomCA exploits all the samples and does not require subsampling. Compared to GS, GeomCA is much simpler to tune as it depends only one hyperparameter ε with an intuitive interpretation.

In IPR, the R and E manifolds are approximated using spheres around each point with radius determined by their k-nearest neighbours. The hyperparameter k can result in large volumes in sparse areas, which authors resolve with manual pruning. GeomCA could be interpreted as using spheres of fixed radius ε, except that we do not endow the graph with any volume. In order to run IPR, the sets R and E need to have the same size, which is not requirement for neither GS or GeomCA.

In contrast to both methods, GeomCA not only extracts the connected component but also enables detailed analysis of

their structure by investigating the corresponding vertices and edges. Moreover, GeomCA enables flexibility to evaluate components of specific size, consistency or quality. In addition to the detailed local evaluation of the components, our refined metrics also provide insights into the global structure of the representation space.

3. Implementation Details and Experimental Design

In this section, we provide additional implementation details as well as an overview of our experiments.

3.1. Selecting distance threshold ε

The structure of the ε-graph $\mathcal{G}$ depends on the hyperparameter ε determining the maximum length of its edges. Extracting the true underlying value of ε is a non-trivial task, especially in higher dimensional representation spaces. If ε is too small, each point is contained in its own component, while ε too large connects all the points into one single component. The true ε that results in the approximation of $\mathcal{M}$ reflecting the correct topology of the space lies between these two extreme choices. A more precise estimate could be determined by topological algorithms, such as persistent homology (Zomorodian & Carlsson, 2004). However, due to computational and scalability issues of this approach, we instead resort to a simple practically applicable heuristic and estimate ε empirically by examining the distances in the reference set R. We randomly sample $2k$ representations from R and calculate their pairwise distances $D = \{d(z_i, z_j) | i = 1, \ldots k, j = k + 1, \ldots, 2k\}$. We then set ε as pth percentile of the set D and denote it by $\varepsilon = \varepsilon(p)$. In our experiments, we chose small p in scenarios where we expect the distances among certain points to have low variance. For example, in contrastive learning, we expect points considered similar to have small distances (Section 4). On the other hand, in high-dimensional representation spaces, we expect the estimated distances to naturally have a larger variance which is why we chose a larger p (Sections 5 and 6). We leave the improvements on the choice of ε as future work.

3.2. Reducing the number of representations

As seen from Definition 2.1, the construction of ε-graph involves calculation of pairwise distances among points in $R \cup E$. Such calculation can become a computational burden when analyzing large sets of representations. One way to reduce the number of representations in R and E without losing the topological information is to perform *geometric sparsification* defined below.

Definition 3.1 *A geometric sparsification of a set W with respect to a sparsificaltion distance $\delta > 0$ is a subset $W' \subset W$ such that $d(w_i, w_j) > \delta$ for every $w_i, w_j \in W'$, $i \neq j$.*

The sparsificaiton parameter δ determines the extend of data reduction, where a larger δ results in a sparser point cloud. We perform geometric sparsification on the sets R and E separately, such that we can construct the ε-graph more efficiently using the obtained sparse sets R' and E'. The reason for separate sparsification of R and E is to detect potential differences in their topology. Intuitively, if R and E reflect the structure of the same representation space, then so should the sparsified sets.

We emphasize that this is an optional pre-processing step added for computational efficiency and can be disregarded if sufficiently powerful hardware is available. Note that the process affects the introduced consistency score c as it reduces the number of points in sets R and E but *does not change their topology* precisely because it takes into account the geometric position of the points. The choice of the sparsification parameter δ is closely related to the choice of the distance threshold ε, and should be chosen from the interval $[0, \varepsilon]$. Intuitively, if $\delta = \varepsilon$, a component in the ε-graph $\mathcal{G}$ is created only when points from R and E are well mixed. This means that between every pair of points from one set (e.g., R) there necessarily needs to exists a point from the other set (e.g., E) that is less than ε apart from both of the points from the first set (e.g., R). On the other hand, if $\delta < \varepsilon$, we still allow points from each of the sets to get connected without having a "witness" from the other set.

3.3. Experiment Overview

We implemented GeomCA described in Algorithm 1 using GUDHI library (The GUDHI Project, 2020) which supports efficient computation of geometric sparsification, and Networkx library (Hagberg et al., 2008) for building and analyzing ε-graphs. Our code is available on GitHub[2]. We applied GeomCA on three different scenarios and evaluated

- similarity of representations obtained from two contrastive learning methods, Siamese (Hadsell et al., 2006) and SimCLR (Chen et al., 2020),

- quality and diversity of images generated by a StyleGAN (Karras et al., 2019), and

- separability of representations obtained from a pretrained VGG16 (Liu & Deng, 2015) model on ImageNet (Deng et al., 2009).

We compared the results with IPR and GS methods using hyperparameters described in Appendix A. We denote the

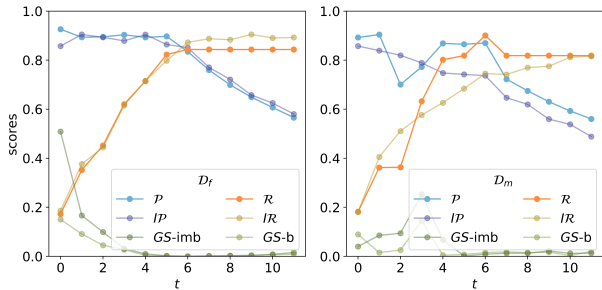---

[2] https://github.com/petrapoklukar/GeomCA

Figure 5. Precision $\mathcal{P}$ and recall $\mathcal{R}$ scores obtained on representations from Siamese network trained on $\mathcal{D}_f$ (left) and $\mathcal{D}_m$ (right) when varying mode truncation level t. We compare the results with $I\mathcal{P}$ and $I\mathcal{R}$ scores, as well as GS computed on balanced (*b*) and imbalanced (*imb*) sets R and E_t (multiplied by 10 on the right).

IPR precision and recall by $I\mathcal{P}$, $I\mathcal{R}$, respectively, and mark GS scores with *b* if R and E were of the same size (balanced) and *imb* in the opposite case (imbalanced). In all experiments, the components analyzed in the local evaluation were sorted by their size in decreasing order such that $\mathcal{G}_0$ always denotes the largest component in the graph.

4. Experiment 1: Contrastive Learning

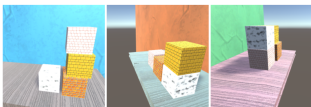We evaluated two models for learning contrastive representations, Siamese and SimCLR, on an image dataset introduced by (Chamzas et al., 2020). Images, shown in Figure 4, consist of four boxes placed in 12 possible arrangements recorded from front, left and right camera views in different scene color configurations. In this experiment, we used two of their datasets: (i) $\mathcal{D}_f$ containing front view images and (ii) $\mathcal{D}_m$ additionally containing images recorded from the left and right views. Each dataset consists of 5000 training images and 5000 test images not used during training. We always constructed R and E from 12-dimensional representations of training and test images, respectively.

Figure 4. Examples of box images recorded from front, right and left views (left to right) contained in $\mathcal{D}_f$ and $\mathcal{D}_m$.

Mode Truncation Experiment In the first experiment, we applied GeomCA to investigate mode collapses and mode discoveries, two possible scenarios occurring during training of deep neural networks. We constructed R from representations corresponding to the first 7 classes (arrangements of boxes), $c_0, \ldots, c_6$, and defined the sets E_t to contain images

from the first t classes, $c_0, \ldots, c_t$, for $t = 0, \ldots, 11$ (see Appendix A.1 for exact sizes of these sets). Therefore, E_t imitate mode collapse for $t < 7$ and mode discovery for $t > 7$. Since contrastive learning models should encode similar classes closeby, we used a small $\varepsilon = \varepsilon(1)$. Moreover, we used $\delta = \frac{\varepsilon}{2}$ to allow the homogeneous clusters also forming a component (see discussion in Sections 3.1 and 3.2), and chose $\eta_c = 0.75, \eta_q = 0.45$ in order to analyze only consistent components of high quality.

In Figure 5, we show precision and recall scores, $\mathcal{P}, \mathcal{R}$, obtained on $R \cup E_t$ for each t. The representations were obtained from two Siamese models trained on $\mathcal{D}_f$ (left) and $\mathcal{D}_m$ (right). We observe that the scores correctly reflect the number of modes covered by each E_t, where recall (in orange) increases but precision (in blue) decreases with increasing t. At $t = 6$, where E_6 perfectly matches R, both $\mathcal{P}, \mathcal{R}$ are high. We observe that the scores correlate well with the IPR scores (visualised in purple and yellow), but not with GS which fails to detect mode discovery cases. Note that IPR scores require R and E to have the same size and were obtained by randomly sampling $\min(|R|, |E|)$ samples for each of the sets. This is not needed for GeomCA which can handle even heavily imbalanced sets, for example, as obtained for $t = 0$ or $t = 11$. We applied GS on both imbalanced (dark green) and balanced (light green) sets, following the authors' recommendation, which yielded similar results. Moreover, GeomCA and IPR correctly identify the modes even on the harder dataset $\mathcal{D}_m$ (right panel), where GS is unsuccessful also for the mode collapse cases.

However, using local evaluation, GeomCA can provide a more detailed insight into the sets R and E_t. For example, in the right panel of Figure 5 we observe a drop in $\mathcal{P}, \mathcal{R}$ scores at $t = 2$. We investigated this by analyzing the quality of the components $\mathcal{G}_i$ of $\mathcal{G}(R \cup E_2)$ containing at least 100 representations, i.e, $|\mathcal{G}_i|_\mathcal{V} > 100$. The resulting scores are visualized in the left panel of Figure 6, where the markers of the components were scaled with their size, and colored with blue if they contain only points from R and gray if they additionally contain points from E. We clearly see that the drop in $\mathcal{P}, \mathcal{R}$ scores originates from the large heterogeneous component $\mathcal{G}_2$ with $q(\mathcal{G}_2)$ just below the chosen threshold $\eta_q = 0.45$. Moreover, we see that the model fails to fully separate all 7 classes since we can observe only 6 large components (x-axis). This is even more evident when performing the same analysis on $\mathcal{G}(R \cup E_3)$ visualized in the right panel of Figure 6. Here, we observe only 4 large components and a significant growth in size of the first component which now contains 53% of $R \cup E_3$ instead of 24 % of $R \cup E_2$ as for $t = 2$. Note that such detailed insights cannot be obtained by the existing frameworks such as IPR and GS.
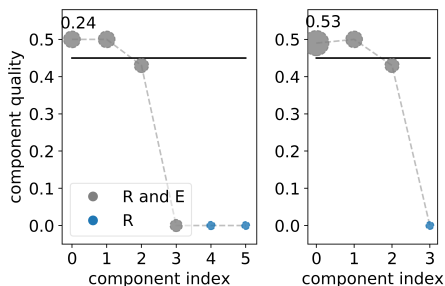
Figure 6. Quality of the components (y-axis) containing more than 100 points obtained at $t = 2$ (left) and $t = 3$ (right) from the Siamese network on $\mathcal{D}_m$. The gray line denotes the threshold $\eta_q = 0.45$. Components' markers are scaled with their size. Gray denotes heterogeneous components, while blue denotes homogeneous components consisting only of R.

In Appendix A.1, we demonstrate how the component consistency and quality thresholds η_c, η_q can be flexibly adjusted to evaluate only components of certain minimum quality.

Evaluating class separability In Figure 5 (left), we have seen that R and E_6 obtained from $\mathcal{D}_f$ by the Siamese model are well aligned. In this experiment, we applied GeomCA to both Siamese and SimCLR models and investigated the extend of the separation that these models achieve among the 7 classes contained in the sets R and E_6. In an ideal case, we would observe exactly 7 *consistent* components of *high-quality*. Moreover, if the clusters are far apart from each other, the result should be robust to the choice of the distance threshold ε. However, as discussed in Section 3.1, for a too small ε, there should be no such components, while for a too large ε we should observe only one large component. To eliminate the effect of sparsification and ensure perfect consistency, we randomly sampled 250 points from each of the classes c_t and ran GeomCA without any further reduction of points.

In Figure 7, we plot the number of components with $|\mathcal{G}_i|_\mathcal{V} > 100$ obtained by Siamese (in green) and SimCLR models (in red) when varying $\varepsilon \in \{0.05, 0.1, \ldots, 0.9\}$. We clearly observe that only Siamese model well separates the classes since we observe 7 components for a large range of ε choices. Surprisingly, SimCLR is much more sensitive to the choice of ε, Moreover, we observe that Siamese network achieves higher network quality than SimCLR (visualized in Figure 10, Appendix A.1),

5. Experiment 2: Generative Models

GeomCA algorithm can also be used to evaluate the quality and diversity of samples generated by generative models.
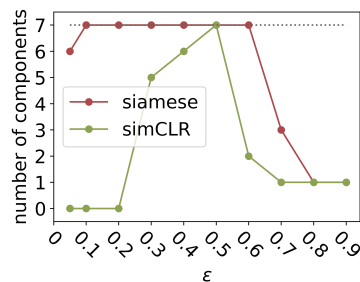

Figure 7. Number of components containing more than 100 points (y-axis) obtained from Siamese and SimCLR models when varying the distance threshold ε.

We used a StyleGAN trained on FFHQ dataset (Karras et al., 2019) and replicated the truncation experiment as performed in (Kynkäänniemi et al., 2019). Here, the latent vectors generating images are during testing sampled from a truncated normal distribution such that the values which fall outside a given range are resampled to fall inside that range (Brock et al., 2019). The level of truncation is controlled by the parameter ψ determining a tradeoff between perceptual quality and variation of images. We generated 50000 images and obtained their 4096-dimensional representations from a pretrained VGG16 model. These composed the set E, while we created R from 50000 representations of the training data. Due to large dimensionality of the representations, we chose $\varepsilon = \varepsilon(10)$, and $\eta_c, \eta_q = 0$. Since the generated representations E are in an ideal case well aligned with R, we chose $\delta = \varepsilon$.
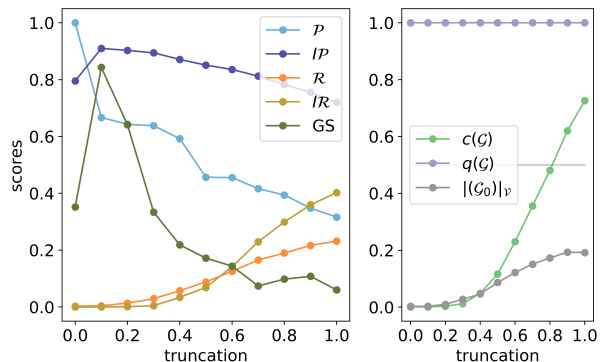

Figure 8. Results of the StyleGAN truncation experiment. Left: GeomCA precision and recall $\mathcal{P}, \mathcal{R}$ compared with IPR and GS (multiplied by 10) scores. Right: network consistency $c(\mathcal{G})$ and quality $q(\mathcal{G})$ as well as the size $|\mathcal{G}_0|_\mathcal{V}$ of the only component containing more than 100 points (scaled by the number of all points in $\mathcal{G}$).

In the left panel of Figure 8, we visualize GeomCA $\mathcal{P}$ and $\mathcal{R}$, IPR and GS (multiplied by 10) scores obtained at each truncation level ψ. We observe that all methods reflect the applied truncation, with some deviations for GS at $\psi = 0$. Comparing GeomCA and IPR, we observe fairly consistent recall but more variation in the precision scores. Therefore, we further investigated the network consistency $c(\mathcal{G})$ and network quality $q(\mathcal{G})$. The results, visualized in the right panel of Figure 8, show that the network consistency (green) is lower than 0.5 for $\psi \leq 0.8$. This is the effect of the geometric sparsification, which in fact removes the majority of E points. For example, the sparsified E contains only 2 points for $\psi = 0.0, 0.1$. As in Section 3.2, we argue that this provides valuable insights into the structure of the generated points. If these reflected the structure of the training points R, the sparsification would return sparsified sets R and E of approximately the same size. Since this is not the case even for $\psi = 1.0$, we argue that the model fails to fully learn the true distribution of the training data, which is also reflected in our low precision scores $\mathcal{P}$. Note that the network has perfect quality regardless the value of ψ due to $\delta = \varepsilon$. As discussed in Section 3.2, this requires every point in R to be 'witnessed' by a point in E, which give rise to heterogeneous edges in the network.

Investigation of homogeneous edges requires to choose $\delta < \varepsilon$ and potentially increasing ε itself. In Appendix A.2, we provide further experiments when varying both ε and δ and show that GeomCA correctly reflects the structure of the space in all cases. We also use this large scale experiment to perform both time complexity and robustness analysis for varying number of samples considered in the sets R and E.

6. Experiment 3: VGG16 Model

The FFHQ representations in the StyleGAN evaluation in Section 5 are obtained from a VGG16 model pretrained on the ImageNet dataset. In a detailed analysis, we always observed only one connected component containing more than 100 points, the size of which grew with the truncation ψ as shown in the right panel of Figure 8 (in gray and labeled with $|\mathcal{G}_0|_\mathcal{V}$). However, since VGG16 is a supervised learning model, we would expect it to be able to separate representations at least to some extent. To determine whether this inseparability originates from the VGG16 model or the nature of the FFHQ dataset, which contains images of faces, we also applied GeomCA to VGG16 representations of the ImageNet dataset.

We performed a simple experiment and defined the sets R and E to contain 5 different classes of the ImageNet dataset each. In version 1, we manually chose classes representing kitchen utilities for R, and dogs for E such that R and E contain semantically different representation (see Appendix A.3 for exact labels and sizes). In version 2, the 5 classes

Table 1. GeomCA scores obtained on VGG16 representations from ImageNet experiment in version 1 (Kitchen utilities vs dogs) and version 2 (random) compared with IPR and GS scores.

	KITCHEN VS. DOGS	RANDOM
$c(\mathcal{G}), q(\mathcal{G})$	0.75, 1.00	0.98, 1.00
$\mathcal{P}$	0.0042	0.0423
$\mathcal{R}$	0.0130	0.0391
$|\mathcal{G}^R|_\mathcal{V}$	1688	1688
$|\mathcal{G}^E|_\mathcal{V}$	2839	1630
$|\mathcal{G}_0|_\mathcal{V}$	18	77
# NON TRIVIAL $\mathcal{G}_i$	7	25
$I\mathcal{P}, I\mathcal{R}$	0.78, 0.97	0.95 , 0.98
GS	0.0018	0.0004

for R and E were chosen at random. If VGG16 is able to separate the classes, then we do not expect to obtain components with high consistency and quality in version 1, while few small ones can emerge in version 2 due to the random choice. Moreover, if the sets in version 1 reflect differences in semantic information of classes, this could potentially be seen in the imbalances after the sparsification process, while this should not be significant in version 2.

As in Section 5, we estimated $\varepsilon = \varepsilon(10)$ and chose $\delta = \varepsilon$. The results of the global and local GeomCA evaluation as well as IPR and GS scores are shown in Table 1. The graph $\mathcal{G}$ in version 1 achieves 75% consistency, which is indeed the result of sparsification. This indicates that the designed sets R and E contain different semantic information. Moreover, since $\mathcal{P}$ and $\mathcal{R}$ are both low, we conclude that there is little overlap between the sets. This can also be seen from the fact that we obtain only 7 non trivial components containing more than one point, where the largest component $\mathcal{G}_0$ contains only 18 elements. On contrary, we observe high $c(\mathcal{G})$ and slightly larger $\mathcal{P}, \mathcal{R}$ scores in version 2, which indicates that there are few areas where R and E are well aligned. This can also be seen from larger number of non trivial components as well as more points in the largest component. Note that $q(\mathcal{G}) = 1$ due to $\delta = \varepsilon$. Therefore, we hypothesise that VGG16 achieves a certain level of separation of ImageNet training classes. We emphasise that it is difficult to draw the same conclusion from either IPR or GS as they fail to provide such detailed insight.

To gain further insights into separability capabilities of the model, we visualize images of representations contained in the obtained components. In Figure 9, we visualize in red a component containing two representations, and in blue an E outlier, both from version 1 (top row) and version 2 (bottom row). In both cases, red components show examples of erroneous merges based on human labels, which are not surprising due to the striking similarity between the images. In the top row, both images contain a silver pot, while the

Figure 9. Examples of ImageNet images corresponding to the representations from version 1 (top) and version 2 (bottom) obtained from a pretrained VGG16. Images with red stroke are taken from components of size 2, where left column images belong to representations of R and right ones to E. Images with blue stroke correspond to outliers. See the text for discussion.

right one also shows a dog. In the bottom row, the crabs in the images belong to two different species that are arguably hard to differentiate but both are placed in a human hand. Another example from version 2 is shown in Figure 1 where images have similar background but contain different object in the center. In the case of outliers, it is rather hard to spot the dog in the lower corner of the top row image, while the sliding window in the bottom one, which is the image label, seems to be of secondary focus after the animals.

7. Conclusion and discussion

We presented GeomCA algorithm for evaluating topological and geometrical properties of representation spaces. The intuition behind GeomCA is that if two given sets of representations R and E contain observations from the same data manifold, then they are necessarily well aligned. We measure this alignment by analyzing the connected component of an ε-threshold graph built on their union $R \cup E$. For each component, we determine its *consistency* by measuring the ratio of points from R and E contained in it, and *quality* by measuring the ratio of heterogeneous edges connecting points from R and E. Moreover, we aggregate these scores into four global measures, *precision, recall, network consistency* and *network quality*. We demonstrate the usefulness of the proposed global and local measures in several different scenarios such as evaluation of separability of representations obtained from both contrastive learning or supervised learning algorithms as well as in the evaluation of trained generative models.

References

Arvanitidis, G., Hansen, L. K., and Hauberg, S. Latent space oddity: on the curvature of deep generative models. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SJzRZ-WCZ.

Arvanitidis, G., Hauberg, S., and Schölkopf, B. Geometrically enriched latent spaces. *arXiv preprint arXiv:2008.00565*, 2020.

Brock, A., Donahue, J., and Simonyan, K. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1xsqj09Fm.

Chamzas, C., Lippi, M., Welle, M. C., Varava, A., Marino, A., Kavraki, L. E., and Kragic, D. State representations in robotics: Identifying relevant factors of variation using weak supervision. In *Robot Learning Workshop, Neurips*, 2020.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/chen20j.html.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. IEEE, 2009.

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pp. 226–231. AAAI Press, 1996.

Ghadirzadeh, A., Poklukar, P., Kyrki, V., Kragic, D., and Björkman, M. Data-efficient visuomotor policy training using reinforcement learning and generative models. *arXiv preprint arXiv:2007.13134*, 2020.

Ghosh, D. and Bellemare, M. G. Representations for stable off-policy reinforcement learning. In III, H. D. and Singh,

A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3556–3565. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/ghosh20b.html.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems*, volume 27, pp. 2672–2680. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf.

Hadsell, R., Chopra, S., and LeCun, Y. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 1735–1742, 2006. doi: 10.1109/CVPR.2006.100.

Hagberg, A. A., Schult, D. A., and Swart, P. J. Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J. (eds.), *Proceedings of the 7th Python in Science Conference*, pp. 11 – 15, Pasadena, CA USA, 2008.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Sy2fzU9gl.

Karras, T., Laine, S., and Aila, T. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Khrulkov, V. and Oseledets, I. Geometry score: A method for comparing generative adversarial networks. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2621–2629, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/khrulkov18a.html.

Kim, H. and Mnih, A. Disentangling by factorising. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2649–2658, Stockholmsmässan, Stockholm Sweden, 10–15

Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/kim18b.html.

Kirichenko, P., Izmailov, P., and Wilson, A. G. Why normalizing flows fail to detect out-of-distribution data. *Advances in Neural Information Processing Systems*, 33, 2020.

Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., and Aila, T. Improved precision and recall metric for assessing generative models. 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, pp. 3927–3936. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/0234c510bc6d908b28c70ff313743079-Paper.pdf.

Laskin, M., Srinivas, A., and Abbeel, P. CURL: Contrastive unsupervised representations for reinforcement learning. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5639–5650. PMLR, 13–18 Jul 2020. URL http://proceedings.mlr.press/v119/laskin20a.html.

Le-Khac, P. H., Healy, G., and Smeaton, A. F. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020. doi: 10.1109/ACCESS.2020.3031549.

Lippi, M., Poklukar, P., Welle, M. C., Varava, A., Yin, H., Marino, A., and Kragic, D. Latent space roadmap for visual action planning of deformable and rigid object manipulation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

Liu, S. and Deng, W. Very deep convolutional neural network based image classification using small training sample size. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 730–734, 2015. doi: 10.1109/ACPR.2015.7486599.

Locatello, F., Bauer, S., Lucic, M., Raetsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning*, pp. 4114–4124. PMLR, 2019.

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

Oneto, L., Donini, M., Luise, G., Ciliberto, C., Maurer, A., and Pontil, M. Exploiting mmd and sinkhorn divergences

for fair and transferable representation learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Pfau, D., Higgins, I., Botev, A., and Racanière, S. Disentangling by subspace diffusion. *arXiv preprint arXiv:2006.12982*, 2020.

Schönenberger, S. T., Varava, A., Polianskii, V., Chung, J. J., Kragic, D., and Siegwart, R. Witness autoencoder: Shaping the latent space with witness complexes. In *NeurIPS 2020 Workshop on Topological Data Analysis and Beyond*, 2020. URL https://openreview.net/forum?id=1gQfXt_U5a-.

The GUDHI Project. *GUDHI User and Reference Manual*. GUDHI Editorial Board, 3.4.0 edition, 2020. URL https://gudhi.inria.fr/doc/3.4.0/.

Wang, F., Liu, H., Guo, D., and Sun, F. Unsupervised representation learning by invariancepropagation. *Advances in Neural Information Processing Systems*, 33, 2020.

Zomorodian, A. and Carlsson, G. Computing persistent homology. In *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, SCG '04, pp. 347–356, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138857. doi: 10.1145/997817.997870. URL https://doi.org/10.1145/997817.997870.