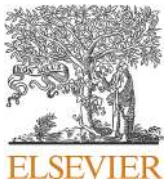
Neurocomputing 661 (2026) 131976



Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Training dynamics of GANs through the lens of persistent homology

Barbara Toniella Corradini [a,1,*], Ben Cullen [b,1], Caterina Gallegati [a,1], Sara Marziali [a,1], Giuseppe Alessio D'Inverno [c], Monica Bianchini [a], Franco Scarselli [a]

[a] Department of Information Engineering and Mathematics, University of Siena, Siena, 53100, Italy
[b] Department of Computer Science, University of Pisa, Pisa, 56127, Italy
[c] International School for Advanced Studies, Via Bonomea 265, Trieste, 34136, Italy

ARTICLE INFO

Communicated by W. Luo

Keywords:
Generative adversarial networks
Topological data analysis
Manifold analysis

ABSTRACT

Generative Adversarial Networks (GANs) aim to produce realistic samples by mapping a low-dimensional latent space with known distribution to a high-dimensional data space, exploiting an adversarial training mechanism. However, without an effective characterisation of the generative process, such models face significant challenges in training and architecture selection. In this work, we propose a Topological Data Analysis based approach, using persistent homology, which can provide such a characterisation, where topological information of a data manifold is summarised by its Persistent Diagram and the evolution of its topological features is tracked throughout training. Our approach is applied across multiple GAN architectures using two benchmark datasets, where we demonstrate that conventional metrics such as Fréchet Inception Distance and intrinsic dimension estimates cannot adequately capture the quality of generated samples. Instead, our results confirm that a topological description of the generative process within GANs successfully captures training convergence and mode collapse. Finally, the layer-wise topological analysis determines the role each layer plays in the generative process, and may provide future guidance for refinement of architectures. Code available at https://github.com/bcorrad/genfold25.git.

1. Introduction

Generative models are a class of unsupervised machine learning paradigms that aim to produce realistic and high-quality synthetic data. Such models are designed to learn a mapping $G_\theta : \mathcal{Z} \to \mathcal{X}$ from a low-dimensional latent space $\mathcal{Z}$ with known distribution p_z to a high-dimensional space $\mathcal{X}$ where the true data distribution p_x resides. Once this mapping is learned, new samples can be generated by sampling latent vectors $z \sim p_z$ and feeding them to the generator G_θ.

Among generative models, Generative Adversarial Networks (GANs) [17] have achieved state-of-the-art results in synthetic image generation through their adversarial training strategy: two networks—a *generator* and a *discriminator*—are trained simultaneously to compete with each other in a min-max game. During the training process, the goal of the generator is to fool the discriminator by producing synthetic images that resemble the training data, while the discriminator must learn to distinguish between real and synthetic images.

An appropriate characterisation of the generative process in GANs is important for controlling the learning procedure and designing optimal architectures. Providing a better understanding of the generative process can help address with several unsolved problems involving GANs [1]. For instance, how does the output space of the trained generator, or even its latent space, relate to the generated data distribution? What level of performance can be achieved by the GAN with respect to a specific training set? Is it possible to correctly identify the causes of mode collapse? Moreover, can we find some criterion to implement early stopping in GANs?

In this paper, we argue that the previous questions can be addressed by relying on the *manifold hypothesis* and *Topological Data Analysis* (TDA). Manifold hypothesis posits that certain classes of high-dimensional data, e.g., natural images, lie on or close to a low-dimensional sub-manifold embedded in the ambient space. This hypothesis has a long history in machine learning and was already implicitly adopted in both Principal Component Analysis [22,24,35] and Autoencoders [20]. Moreover, it has formed the basis of modern research in the field of *manifold learning* [23,30,41] and has often been used to explain the advantage of the high-level abstractions of data that are produced by deep learning

* Corresponding author.
Email address: barbara.corradini@unisi.it (B.T. Corradini).

[1] Authors contributed equally.

https://doi.org/10.1016/j.neucom.2025.131976
Received 30 June 2025; Received in revised form 22 October 2025; Accepted 29 October 2025
Available online 31 October 2025

architectures [4]. In the context of generative modelling, the manifold hypothesis implies that the generative task can be seen as twofold: (i) learning the low-dimensional structure that forms the support of the generated distribution, and (ii) learning the true distribution itself. Hence, if a GAN cannot successfully learn the low-dimensional structure, then it will not learn the true distribution [28].

This low-dimensional structure can be studied using various tools from *Topological Data Analysis* [11]. In particular, *persistent homology* [15] captures fundamental geometric and topological features, such as connected components, holes and voids, as they appear and persist over different geometric scales. By analysing the persistence of these features, an effective characterisation of the data manifold can be constructed, obtaining its multi-scale, global, intrinsic properties that are often overlooked by other methods. While several studies apply TDA to discriminative models [7,8,18,32], very few works use persistent homology for the analysis of generative ones [10,21], where a robust TDA-based approach is still lacking.

This work presents an empirical study in which we exploit TDA to analyse how a GAN learns the underlying low-dimensional structure of a dataset, and its implications with respect to the generative capabilities of the model. In particular, we investigate how a GAN learns by observing (i) changes in the topology of the generated manifold and (ii) transformations of the latent space manifold as it passes through the layers of the generator. Our approach suggests that TDA effectively identifies and characterises the learning dynamics as well as common issues observed in GANs. We also provide the trend of widely used evaluation metrics, such as the Fréchet Inception Distance (FID) [19] and the intrinsic dimension (ID) estimate [5], finding that these metrics are not always sufficiently informative to detect critical issues in GANs.

In particular, the following observations are derived.

- **About convergence of the generated manifold to the target**. If the homology of the generated manifold fails to converge within an acceptable margin to that of the target, the generator will not successfully learn the true distribution.
- **About FID score and ID.** Neither the FID score alone nor its combination with ID forms a sufficient metric to evaluate the quality of the generated distribution.
- **About mode collapse.** Mode collapse can be clearly identified as a sudden increase in the distance between the target and the generated manifold of the 0-homology group, i.e., the one identifying connected components.
- **About layer contribution.** We observe that different types of layers have a distinct impact on the manipulation of the generated manifold. This offers insights into how each layer impacts the learning dynamics.

Our paper is structured as follows. Section 2 summarises topology-based approaches to analyse different neural network architectures, as well as state-of-the-art applications in the domain of GANs. In Section 3, we present a brief introduction to persistent homology, used in this paper to characterise data manifolds. Section 4 outlines our methodology and provides a detailed description of each experiment. We present our results in Section 5 alongside conclusions and future directions in Section 6.

2. Related works

The first work to investigate the relationship between topological concepts and computational capabilities of neural network architectures is proposed in [7]. By establishing a topological complexity measure using Betti numbers, the authors derive theoretical upper and lower bounds on both the necessary width and depth of a neural network for implementing functions of varying complexity. A natural extension of this work is given in [18], which introduces an empirical analysis of the topological complexity of a dataset and the minimal network capacity required to classify it. The authors validate their methodology using synthetic datasets whose topology is a-priori known and then extend their empirical analysis to real datasets. Fully connected architectures of different depths and widths with ReLU activation functions are considered and compared in turn. In [32], the authors provide a qualitative analysis of how the topological complexity of data manifolds changes when passing through different layers of a Multi-Layer Perceptron. They demonstrate that in shallow architectures topological transformations are performed in the final layers, whereas in deep architectures they are spread out across all layers. A qualitative study of the generative process within Convolutional Neural Networks (as well as other generative model classes) is performed in [29]. The authors study the transformations of the internal representations of each structural block using homological descriptors known as *life-span sums*. In addition, *persistent homology fractal dimension* is used for estimating the ID of each embedding space. Nevertheless, only GAN learning dynamics are considered through the study of convergence of topological and geometric features, and an evaluation of the quality of the generated samples is missing.

On the other hand, very few works focus on evaluating the performance of GANs, as well as describing their generative process, through the lens of persistent homology. In [10], the authors study the evolution and convergence of the estimated intrinsic dimension and the first three Betti numbers of the generated manifold during training. However, the work does not investigate how different model architectures influence these topological and geometric properties, nor how such properties could be leveraged to evaluate the generative process. In [21], the authors propose a new metric known as *topology distance*, which compares the 0-dimensional homology groups of real and generated manifolds. The main criticism of this approach is related to data representation: both manifolds are obtained using a pretrained feature extractor, which may introduce biases in the final evaluation. In [25], a probabilistic approach is proposed for comparing manifolds for the evaluation of GANs. By choosing subsets of *landmark points* to derive *witness complexes*, representations of manifolds are obtained in an approximate stochastic manner. Discrete probability distributions are then computed using vector representations of their topological information. Finally, the authors estimate the topological similarity of the training and generated manifolds as a distance between the empirical mean of these distributions.

In [37], the authors employ persistent entropy to quantify topological complexity in generative models. Persistence diagrams are computed on cubical complexes built from image pixel intensities, yielding an image-level measure of structural diversity. Two theoretical results relate this entropy to the homological complexity and its mean value across samples, and the metric's distribution highlights issues such as mode collapse. However, their approach focuses on the topology of individual generated images and their averages, and does not establish a connection with the topology of the true and generated data manifold.

Our approach moves away from data representation issues by extracting encodings from the last layer of the GAN's generator, avoiding the use of auxiliary networks. In addition, we avoid introducing a-priori assumptions on the choice of points and parameters when computing simplicial complexes. Contrary to previous works, our analyses compare different descriptors of the generative process: persistence diagrams of the first three homology groups of a given manifold; the FID score between real and generated distributions; and the ID of both real and generated manifolds.

3. Preliminaries

This section introduces essential concepts from TDA for characterising the topology of a manifold using only a finite collection of data points. For a more comprehensive treatment, see Appendix A.

3.1. Homology

Simplicial complex. A *k-dimensional simplex*, or simply *k-simplex*, σ on a vertex set V is any subset consisting of $k+1$ elements contained in

V. A *simplicial complex* K on a vertex set V is a collection of non-empty subsets of V such that: (i) for every $v \in V$ then $\{v\} \in K$, and; (ii) if τ and σ are both simplices where $\tau \subset \sigma$ and $\sigma \in K$, then $\tau \in K$. The dimension of a simplicial complex K is given by $\dim(K) := \max\{\dim(\sigma) \mid \sigma \in K\}$.

Simplicial homology. Simplicial complexes are topological spaces that can be characterised by their homology groups, i.e., algebraic objects that are invariant under topological transformations known as *homeomorphisms*. These groups encode the presence of k-dimensional homological features of the topological space – e.g the number of connected components ($k = 0$), holes ($k = 1$), voids ($k = 2$) and other higher dimensional analogues. In order to compute the homology groups of a simplicial complex K, one constructs a sequence of vector spaces whose elements represent formal combinations of k-simplices. These are linked by linear maps called *boundary operators*, which encode how simplices are connected across dimensions. Studying the algebraic structure of these operators allows for the computation of the homological features of K.

Let K be a simplicial complex and $K^{(k)} = \{\sigma_1, \dots, \sigma_m\}$ be the collection of all k-simplices in K. Then the *k-th chain group of K* over a coefficient ring $\mathbb{F}$, denoted $\mathbf{C}_k(\mathbb{F}; K)$ or simply $\mathbf{C}_k$, is a vector space whose basis elements are the simplices $K^{(k)}$. Every *k-chain* $\gamma \in \mathbf{C}_k$ can be uniquely expressed as:

$$\gamma = \sum_{i=1}^{m} \gamma_\sigma \sigma_i \quad \text{where} \quad \gamma_\sigma \in \mathbb{F}.$$

The *k-th boundary operator*, ∂_k, acting on an arbitrary k-chain $\gamma \in \mathbf{C}_k$ is defined as:

$$\partial_k \gamma = \sum_{i=1}^{m} \sum_{j=0}^{k} \gamma_{\sigma_i} (\sigma_i)_{-j},$$

where $(\sigma_i)_{-j}$ is the $(k-1)$-simplex obtained by removing the vertex v_j from σ_i. Hence, $\partial_k : \mathbf{C}_k \to \mathbf{C}_{k-1}$ is a linear map with the property $\partial_k \circ \partial_{k+1} = 0$, i.e., *"a boundary of a boundary is trivial"*. For each dimension k the following holds:

$$\mathbf{B}_k := \mathrm{Img}\left(\partial_{k+1}\right) \subseteq \mathrm{Ker}\left(\partial_k\right) =: \mathbf{Z}_k,$$

where elements of $\mathbf{B}_k$ are called *k-boundaries* and elements of $\mathbf{Z}_k$ are called *k-cycles*.

With boundary operators and chain groups on the m-dimensional simplicial complex K, one can construct the following *chain complex*:

$$0 \xrightarrow{\partial_{m+1}} \mathbf{C}_m \xrightarrow{\partial_m} \mathbf{C}_{m-1} \xrightarrow{\partial_{m-1}} \dots \xrightarrow{\partial_1} \mathbf{C}_0 \xrightarrow{\partial_0} 0,$$

and define a quotient vector space called the *k-th homology group* of K as:

$$\mathbf{H}_k := \frac{\mathbf{Z}_k}{\mathbf{B}_k} = \frac{\mathrm{Ker}\left(\partial_k\right)}{\mathrm{Img}\left(\partial_{k+1}\right)} \quad \text{for} \quad k = 0, 1, \dots, m.$$

Hence, $\mathbf{H}_k$ represents the k-cycles that are not boundaries of higher dimensional chains and therefore capture "non-trivial" homological features within K. The rank of the k-the homology groups is known as the *k-th Betti number*:

$$\beta_k := \mathrm{rank}\left(\mathbf{H}_k\right) \quad \text{for} \quad k = 0, 1, \dots, m,$$

where β_k counts the number of k-dimensional voids in K.

3.2. Persistent homology

Let $\mathcal{P}$ be a point cloud that is either drawn from or close to a manifold $\mathcal{M}$ whose topological structure cannot be directly studied. Instead, one could consider a surrogate topological space, such as a simplicial complex, whose homology is equivalent to that of the manifold [33]. However, in the absence of certain a-priori knowledge about $\mathcal{M}$ – such as its condition number – it is not possible to guarantee this equivalence using a single complex. Alternatively, one can construct a sequence of nested simplicial complexes from $\mathcal{P}$ at varying geometric scales, forming what is known as a *filtration*. Studying the *persistent homology* of this filtration enables the inference of homological features of the underlying manifold.

Vietoris-Rips filtration. Given a metric δ on $\mathcal{P}$ and a geometric scale $\rho \geq 0$, the *Vietoris-Rips complex* $\mathrm{VR}_\rho(\mathcal{P})$ is given by the set of simplices:

$$\mathrm{VR}_\rho(\mathcal{P}) := \{\{u_0, \dots, u_k\} : \delta(u_i, u_j) \leq 2\rho \quad \text{for all } i, j \in [k], \text{ where } u_0, \dots, u_k \in \mathcal{P}\}.$$

For any sequence of scales $0 \leq \rho_1 \leq \rho_2 \leq \cdots \leq \rho_m$ the *Vietoris-Rips filtration* is given by the following chain of inclusions:

$$\mathrm{VR}_0(\mathcal{P}) \subseteq \mathrm{VR}_{\rho_1}(\mathcal{P}) \subseteq \mathrm{VR}_{\rho_2}(\mathcal{P}) \subseteq \cdots \subseteq \mathrm{VR}_{\rho_m}(\mathcal{P}),$$

where ρ_i is the *filtration parameter*.

Persistent homology. The inclusion maps that define the filtration induce linear maps between the k-th homology groups of each complex. Hence, one can track the geometric scales at which each homological feature appears (its *birth*, ρ_b) and disappears (its *death*, ρ_d) within the filtration, representing the feature by its *persistence interval* (ρ_b, ρ_d). The length of this interval is the *persistence* of the corresponding feature and reflects its significance – features with greater persistence are typically considered more representative of the homological structure of the underlying manifold. In contrast, features with short persistence are often interpreted as topological noise within the filtration.

Persistence diagram. A graphical representation of the persistent homology of a filtration is given by its *persistence diagram* (PD). A PD is a multiset of points in $\mathbb{R}^2$, where the persistence interval of the i-th non-trivial feature, $(\rho_{i,b}, \rho_{i,d})$, is a point that lies above the diagonal $\Delta = \{(c, c) \mid c \in \mathbb{R}\}$. Δ is also included in the PD, where each of its points has infinite multiplicity. Features with a short persistence lie close to Δ. An example of a PD is shown in Fig. 1.

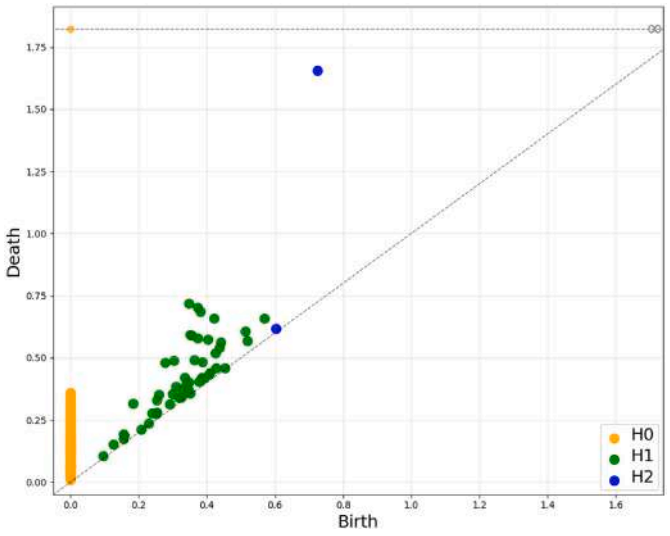


Fig. 1. A persistence diagram showing the evolution of topological structures across different scales for the first three homology dimensions. The H_0 features (orange) represent connected components, H_1 (green) represents loops, and H_2 (blue) represents voids. According to theory, only one connected component has infinite persistence, represented by a point on the top border of the diagram.

It is possible to compare the homological features of two PDs, D_1 and D_2, using various distance metrics. A popular choice is the p-th Wasserstein distance:

$$\mathcal{W}_p(D_1, D_2) = \inf_{\phi: D_1 \to D_2} \left(\sum_{a \in D_1} || a - \phi(a) ||_q^p \right)^{\frac{1}{p}}, \quad (1)$$

where $\phi : D_1 \to D_2$ is a bijection known as a *matching* and $q \in \mathbb{N}\backslash\{0\} \cup \infty$; in this work, we use the Euclidean distance by setting $q = 2$. Each matching has a cost based on the distance required to move a point $a \in D_1$ to a point $b = \phi(a) \in D_2$. Hence, the Wasserstein distance between two diagrams is the minimum-cost matching among all possible ϕ.

4. Material and methods

The ability of a GAN to replicate a given dataset strictly depends on the architectural features and the complexity of the dataset's manifold it aims to reproduce [18]. If the architecture of a GAN has sufficient expressive capacity to model a manifold with homological complexity similar to that of the real data, we expect that training can produce effective results. Inspired by those works exploring discriminative models using TDA, in this study we explore foundational GANs to infer a relationship between the generation capabilities and the topology of the training manifold. We pose the following question: *is it reasonable to relate GAN failures and, more generally, the quality of training to the evolution of the manifold throughout the training process?*

This section presents details of the experimental setup used to evaluate the performance and capabilities of selected GAN architectures. The experiments investigate each model's ability to replicate and generate realistic data across widely used datasets, such as MNIST [13], CIFAR-10 [26] and CelebA [27], as briefly described in Section 4.1. In Section 4.2, the implementation details are presented, providing: the hyperparameters of each GAN (see Appendix B for further details), the training configurations, and details for the FID, ID and persistent homology calculation. Each dataset presents different challenges in terms of data complexity, resolution, and class variety, providing a robust basis for assessing the expressive capacity of each model.

4.1. Datasets

MNIST. The MNIST dataset [13] is a collection of 70,000 grayscale images of handwritten digits (zero to nine), each of size 28×28 pixels. It is split into 60,000 training images and 10,000 test images. MNIST is widely used in image generation due to the low resolution of the images and their simple semantic and graphical content. Hence it is a useful benchmark for understanding the basic expressive power of a generative model.

CIFAR-10. The CIFAR-10 dataset [26] is a ten classes benchmark composed of $60,000$ RGB images of size 32×32 pixels. It is generally considered more complex when compared to MNIST, with higher variability in terms of objects, textures, and colour schemes. This dataset challenges a model's ability to capture and reproduce nuanced features, including colour dynamics and object structures. Hence, it is used to assess the performance of generative models on more diverse and complex data.

CelebA. The CelebA dataset [27] is a large-scale face attributes dataset containing over $200,000$ RGB images of celebrity faces, each annotated with 40 binary attribute labels. The images are usually resized to 64×64 or 128×128 in generative modeling tasks. CelebA poses a greater challenge compared to object-centric datasets like CIFAR-10 due to its focus on fine-grained facial features, variations in expression, pose, and illumination. It is widely used to evaluate the capacity of generative models to synthesize realistic and identity-preserving human faces while capturing subtle attribute-level variations.

4.2. Implementation details

GAN architectures. We perform experiments across four different GAN architectures: two linear models (a standard GAN and a WGAN-GP with a linear structure) and two convolutional models (a DCGAN and a WGAN-GP with a convolutional structure). Each model is trained for 400 epochs with a batch size of 32 and generates RGB images with a resolution 32×32 pixels. The input for each model is a noise vector with 100 components. For optimisation, we apply the Adam optimiser with learning rates and beta parameters tailored to each architecture. The implementation details of each architecture are provided in Appendix B.

Training configurations. Each architecture is trained on MNIST and CIFAR-10 datasets, using a single-class configuration (the digit 3 and the *cat* class, respectively) and a dual-class configuration (digits 3 and 7, and *cat* and *horse* classes, respectively). In the multi-class case, we consider only two classes in order to balance interpretability of results with the complexity of the target manifold. We additionally train models on the CelebA dataset, with all images rescaled to a resolution of 32×32 pixels.

Evaluation of generated samples using FID. The *Fréchet Inception Distance* score [19] is a widely-used metric for assessing the quality of images produced by a generative model. Specifically, it calculates the distance between the statistics of the feature representations of both generated and real images, as encoded by a pretrained Inception network. In each experiment, we compute the FID score for each epoch to quantitatively evaluate how closely the generated images resemble the target dataset during training. In order to ensure a fair comparison across epochs and architectures, the FID score is calculated using a sample of generated images, whose number matches the size of the target dataset.

Evaluation of dataset complexity by intrinsic dimension. We implement a maximum likelihood estimation approach to approximate the *intrinsic dimension* of data [5,36]. Due to computational constraints, the ID of each dataset is estimated on a representative fraction of the samples, determined by the dataset's complexity. We show in Appendix F that this subset is sufficient to capture the data characteristics without a significant loss of generality.

Persistent homology calculation. We characterise the shape of generated and target manifolds by computing their persistent homology as PDs using the Ripser library [38].

To reduce the computational complexity of persistent diagram calculations while preserving the essential topological features of the data, we employed a greedy permutation algorithm [9] to subsample the point cloud. This method, implemented in the Ripser library, selects a subset of points that are maximally dispersed using a "furthest point first" strategy [3], ensuring representative coverage of the original dataset. This approach is commonly used in TDA to enable scalable computation without significant loss of structural information.

We retain the first $n = 100$ points in the greedy ordering. This choice represents a practical trade-off: it reduces the time complexity of PD calculation while still providing a reasonably informative topological representation (see Appendix D for further details). While including more points would yield a finer-grained filtration, doing so would make our computations prohibitive based on the available hardware.

We evaluate the distance between PDs using the 2-Wasserstein distance, as defined in Eq. (1), and denote it for notational convenience as $\mathcal{W}(\cdot,\cdot)$, omitting the explicit reference to the parameter p. Instead, we refer to $\mathcal{W}_i$ as the 2-Wasserstein distance between the homology features of dimension $i = 0, 1, 2$ in the PDs. This distance is leveraged in three different scenarios:

- **Training stability.** We compare consecutive PDs of the generated data across epochs by evaluating the Wasserstein distance between them. In this scenario, PDs are computed for each epoch using the

GAN's final feature layer. For completeness, we include the PD of the initial noise vector in the comparison. By monitoring this distance over consecutive epochs, we gain insights into how stable the training of a given model is: large variations imply that the homology of the generated manifold may have changed drastically and hence that learning has become unstable; small variations imply stable training or that training may have stalled altogether.

- **Convergence of the generated manifold to the target.** An epoch-wise comparison of the generated manifold, however, is not sufficient for understanding how capable a GAN is in learning the underlying structure of the target manifold. For this reason, we propose comparing the PD of the generated manifold at each epoch with that of the target by computing the corresponding Wasserstein distance. Monitoring whether or not this quantity converges to either zero or a non-zero value may indicate the capacity of a GAN to learn the underlying structure of the target manifold.
- **Topological changes across layers.** We compute the mean and standard deviation of the Wasserstein distance between PDs of consecutive layers using all training epochs. This analysis may provide guidance for refinement, as well as selection, of model architectures that better capture the underlying structure of the target data.

5. Results and discussion

In this section, we present and discuss the results of our analysis, highlighting key findings and their implications for GAN behaviour and training dynamics. Specifically, we track the evolution of topological features of the generated manifold during training, as well as the transformations of the latent space manifold across layers. To this end, we evaluate multiple classes of GANs trained on subsets of widely-used datasets: MNIST, CIFAR-10 and CelebA.

By training and analysing the models across these datasets, we seek to identify and characterise several well-documented issues common to GANs, including mode collapse and training instability. Typically, these phenomena are assessed through qualitative examination of the generated samples, as they manifest visually in the output images. To deepen our analysis, we also examine trends of statistical (FID) and geometrical (ID) evaluation metrics for assessing generated images.

In the following, the notable case studies are presented to show how we visualise the results and derive the corresponding discussion. Additional experiments which mirror the same behaviors described in the section, are reported in Appendix C.

5.1. Convergence to the target manifold reflects proper training dynamics

The training dynamics of Linear WGAN-GP (Class 3 MNIST) show that the training is stable across all epochs (Fig. 2c), and that the generated manifold converges to that of the target (Fig. 2d). Moreover, $\mathcal{W}_0$ seems to play a critical role in shaping the manifold, showing the most prominent changes. On the other hand $\mathcal{W}_1$ and $\mathcal{W}_2$ distances remain relatively low and stable, suggesting limited contributions from loops or higher-dimensional features.

The ID of the generated manifold is initially less than that of the target, after which it steadily increases approaching true value (Fig. 2b). We first observe (Fig. 2a) a dip in the FID progression followed by a slowly decreasing trend that seems not to converge. This behaviour can be explained by looking at the generated images (Fig. 2f) in the following way:

1. After epoch 0, the model generates samples with noisy pixels, leading to very high FID and ID scores (Fig. 2a and b). In addition, large initial values are reported for $\mathcal{W}_0$, confirming that the topology of the generated manifold does not match that of the target (Fig. 2d);
2. The model then refines edges, removes noise and artifacts, improving FID and ID scores while reducing $\mathcal{W}_0$ between the generated and the target manifold. Additionally, the decreasing topological distance between generated manifolds across consecutive epochs indicates stable training progression;
3. The Wasserstein distance progression through layers (Fig. 2e) reveals that linear layers have the greatest impact on the value of $\mathcal{W}_0$, highlighting their role in shaping lower-dimensional topological features of the latent space, while normalisation layers likely focus on refining the geometry of the manifold.

We observe an overall analogous convergent behaviour in the training of Linear WGAN-GP (Classes 3 and 7 MNIST), as reported in Appendix C.

5.2. Non-converging $\mathcal{W}_0$ and low ID reveal limitations in model complexity and diversity

Taking into account the scale of the y-axis, the Wasserstein distances for consecutive epochs in the training dynamics of DCGAN (Class 3 CIFAR-10) suggest training stability (Fig. C.11a). However, the consistently high $\mathcal{W}_0$ distance between generated and target manifolds (Fig. C.11b) reveals that the DCGAN also struggles to learn the target manifold's topological structure. The downward trend of the ID estimate (Fig. C.10b) suggests that the model generates overly simplified images, whereas the lack of FID convergence (Fig. C.10a) likely corresponds to the appearance of artifacts such as faded textures and saturation in the generated samples (Fig. C.11d). At epoch 380, a sudden increase in $\mathcal{W}_0$ indicates a worsening in training stability as the model struggles to learn the correct topology of the target manifold (Fig. C.11a). Peaks in $\mathcal{W}_0$ distances in transposed convolutional layers (Fig. C.11c) suggest that their contribution is greatest in shaping the manifold, while activation and batch normalisation layers focus on refinement.

This experiment provides evidence for two key claims: (i) the non-converging $\mathcal{W}_0$ distance suggests that the model cannot capture the target dataset's complexity, and (ii) the low value for ID indicates that the model generates overly simplified images.

The Linear WGAN-GP on CelebA exhibits significant challenges. The Wasserstein distances for consecutive epochs indicates training stability (Fig. 5c). However, inspection of Fig. 5d shows the model's inability to converge to the target manifold's topological structure. Furthermore, the ID progression (Fig. 5b) does not show clear convergence, and despite a decreasing FID score (Fig. 5a), the generated samples (Fig. 5f) consistently display artifacts and low visual fidelity. This suggests that the model struggles to capture the complexity of the CelebA dataset, leading to simplified and unrealistic outputs.

We observe an overall analogous non-convergent behaviour in the training of vanilla GAN (Class 3 MNIST) and architectures trained on CelebA, as reported in Appendix C.

DCGAN on CIFAR-10 (Class 3)

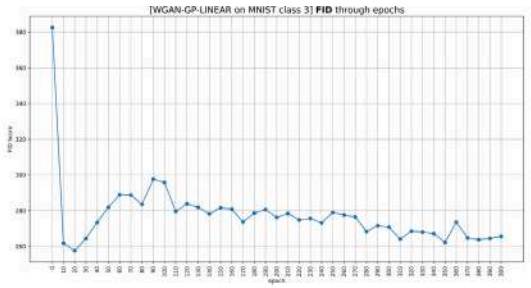
See Figs. 3 and 4.

Linear WGAN-GP on CelebA

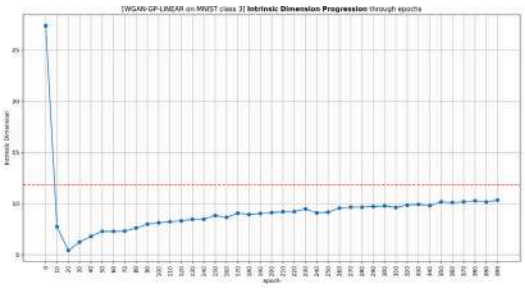
See Fig. 5.

5.3. Manifold topology supports the evaluation of training beyond FID and ID

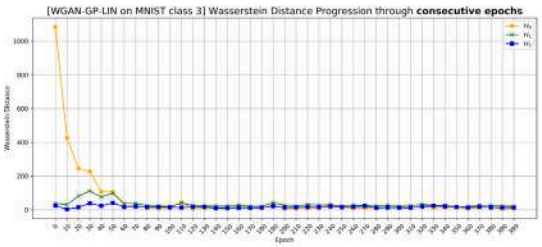
From the training of DCGAN (Class 3 MNIST), we observe that training is stable until epoch 330, where there is a sharp increase in $\mathcal{W}_0$ (Fig. C.13a). The same change is observed with respect to the target manifold (Fig. C.13b), where a plateau is recorded until the end of training. The ID (Fig. C.12b) shows a decreasing trend, reaching the target value and then diverging. Examining the generated samples (Fig. C.13d), they display a high black-white contrast (epoch 50) when the ID estimate matches that of the target, but appear increasingly faded by epoch 310. Finally, at the plateau (Figs. C.12a and C.13b, epoch 320 onwards),



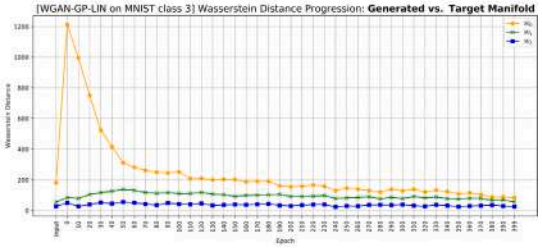
2(a) **FID score progression.** The blue line represents FID values at each epoch. We observe an overall decreasing behaviour of the metric.



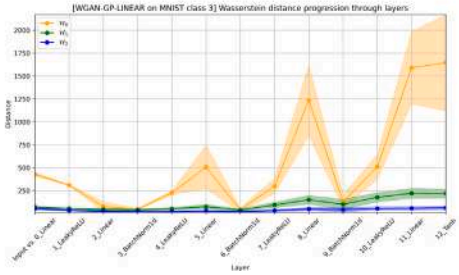
2(b) **ID progression.** The blue line shows the ID computed over epochs. A convergent behaviour from below is observed after an initial underestimation.



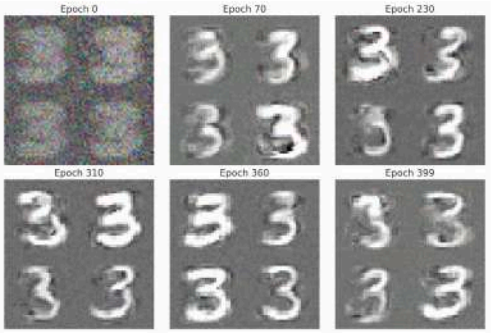
2(c) **Training stability.** Comparison of PDs for consecutive epochs. A convergent behaviour is observed for all the three homology classes.



2(d) **Convergence to the target manifold.** Comparison of PDs corresponding to the generated and target manifold. A convergent behaviour is observed for all the three homology classes.



2(e) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.



2(f) **Generated samples.** We report samples generated during training at different epochs.

Fig. 2. Case-study of a good training. Top: (a) FID score and (b) ID progression evaluated on the generated manifold. Bottom: Wasserstein distance between PDs of the generated manifold for (c) consecutive epochs, (d) with respect to the target manifold, and (e) through layers across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue). Training performed on a Linear WGAN-GP on MNIST dataset (class 3) for 400 epochs.

the model experiences mode collapse, generating only a single mode of images. The convolutional layers seem to be primarily involved in modifying the topology of the latent space manifold, which could be related to the change in the number of channels that they introduce. Hence, such layers may potentially influence the overall shape and complexity of the generated manifold.

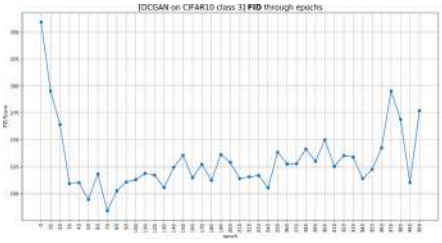
During the first half of training of DCGAN (Classes 3 and 7 MNIST), we observe a stable (Fig. C.15b) and convergent behaviour (Fig. C.15a). However, around epoch 210, the model drastically changes the topology of the generated manifold and then plateaus. The beginning of a new plateau corresponds to a sudden change in the training dynamics, as well as a new phase of mode collapse. Generated samples (Fig. C.15d) display repetitive patterns and a loss of diversity. This trend is repeated several times until the final epoch. This pattern is reflected in the anomalies observed in both the FID score and the ID progression (Fig. C.14a and b). As already noted, convolutional layers are primarily involved in the topological changes of the latent space manifold. However, when training involves multiple classes, the system's complexity increases significantly, as evidenced by the larger layer-wise distances compared to the single-class scenario (Fig. C.15c).

DCGAN on MNIST (Classes 3 and 7)

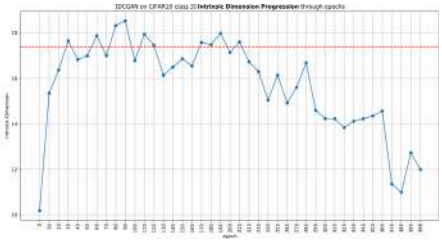
See Figs. 6 and 7.

5.4. Topology reveals training instability hidden by FID and ID in GAN training

Both FID (Fig. C.20a) and ID (Fig. C.20b) progression in Convolutional WGAN-GP (Class 3 CIFAR-10) training seem to imply

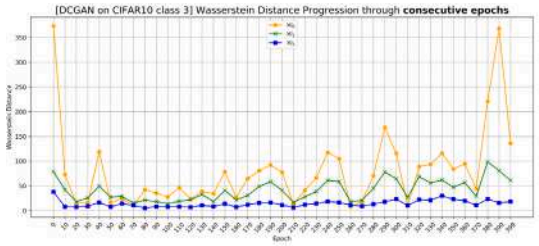


3(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

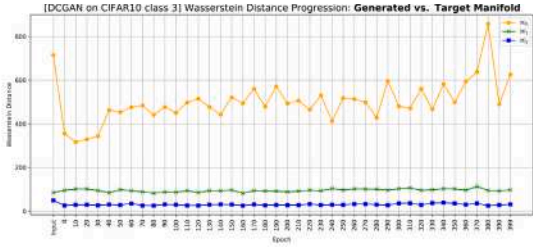


3(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

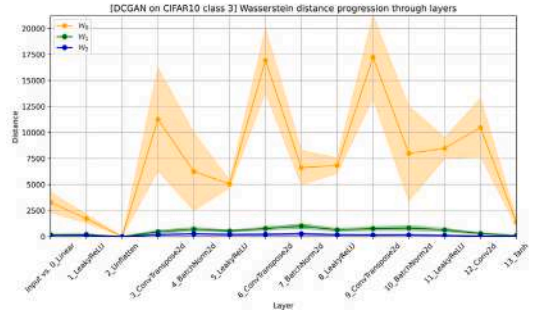
Fig. 3. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



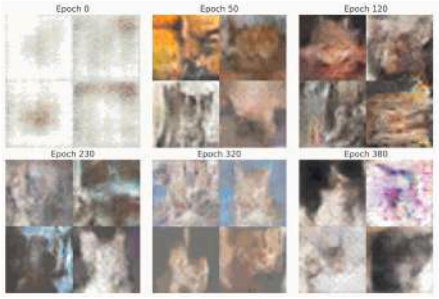
4(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



4(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



4(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.



4(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. 4. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

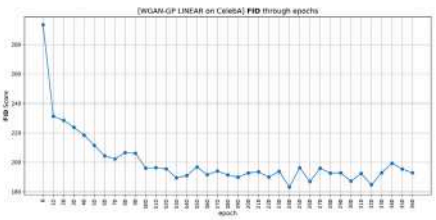
that the model discovers a good parameter configuration after a small number of epochs, even if the ID estimate is slightly above that of the target. However, it is clear that training is unstable (Fig. C.21a) resulting in the generation of data manifolds whose topology does not match that of the target (Fig. C.21b). Inspecting the generated samples, it is clear that the model has failed to learn the target dataset (Fig. C.21d). We observe that the convolutional and upsampling layers contribute most to transforming the topology of the latent space manifold (Fig. C.21c).

Convolutional WGAN-GP on CIFAR-10 (Class 3)

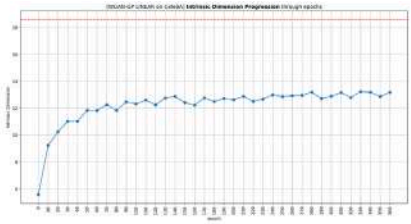
See Figs. 8 and 9.

5.5. Topology-informed insights into layer contributions during training

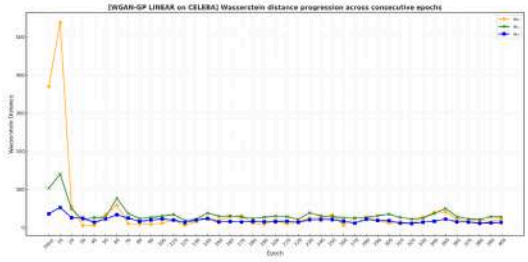
From the experiment performed on Linear WGAN-GP (Classes 3 and 7 MNIST), we observe stable training (Fig. C.19a) and effective convergence of the generated manifold to the target (Fig. C.19b). The ID progression (Fig. C.18b) gradually aligns with the target, while the steadily decreasing FID score (Fig. C.18a) reflects convergence. These observations, combined with the qualitative results (Fig. C.19d), confirm that the model trains effectively, producing stable and realistic samples while successfully capturing the target manifold's topological structure. Significant variations in $\mathcal{W}_0$ within the linear layers emphasise their critical role in global manifold transformations. In contrast, activation and normalisation layers focus on refining the geometry of the latent space manifold.



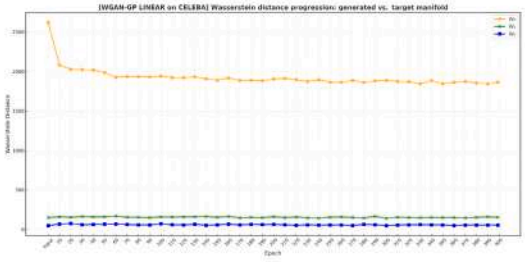
5(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.



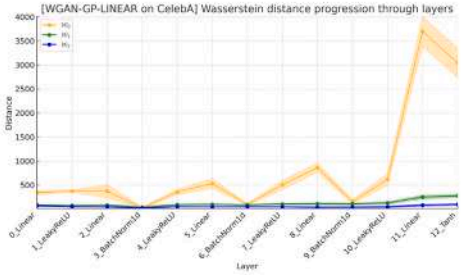
5(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.



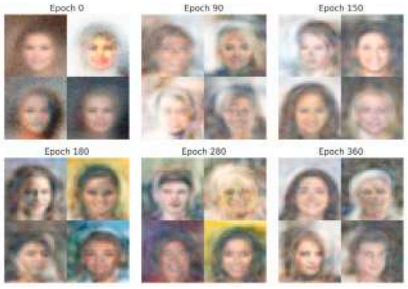
5(c) **Training stability.** Comparison of PDs for consecutive epochs. A convergent behaviour is observed for all the three homology classes.



5(d) **Convergence to the target manifold.** Comparison of PDs corresponding to the generated and target manifold. A convergent behaviour is observed for all the three homology classes.

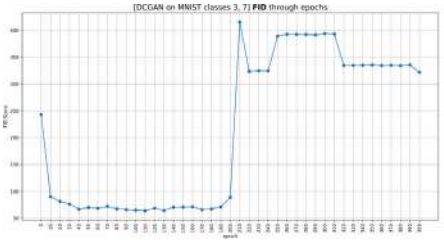


5(e) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

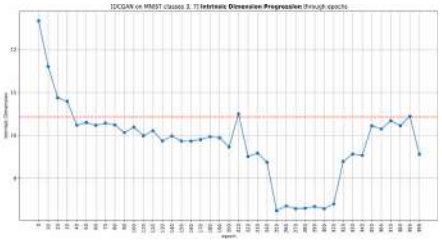


5(f) **Generated samples.** We report samples generated during training at different epochs.

Fig. 5. Case-study of non-convergent behaviour. Top: (a) FID score and (b) ID progression evaluated on the generated manifold. Bottom: Wasserstein distance between PDs of the generated manifold for (c) consecutive epochs, (d) with respect to the target manifold, and (e) through layers across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue). Training performed on a Linear WGAN-GP on MNIST dataset (class 3) for 400 epochs.

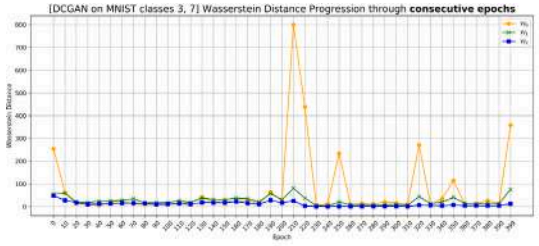


6(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

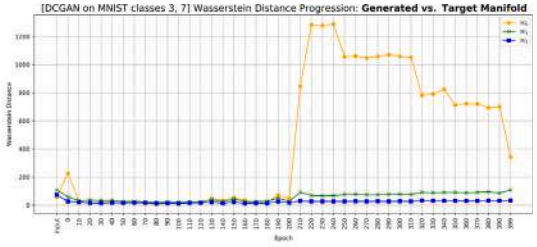


6(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

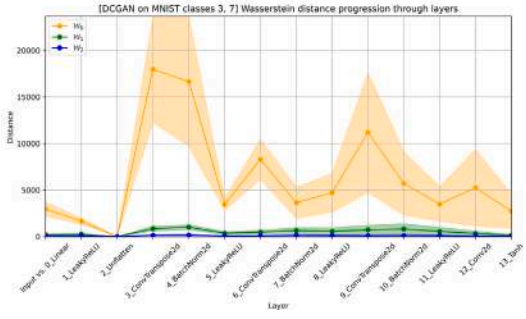
Fig. 6. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



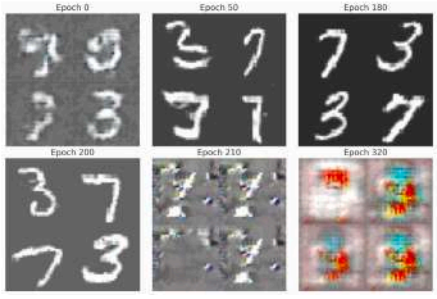
7(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



7(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.

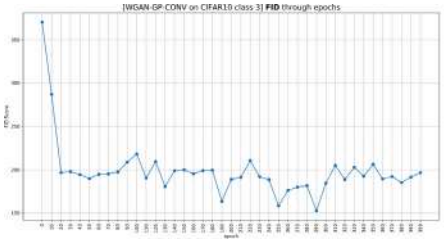


7(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

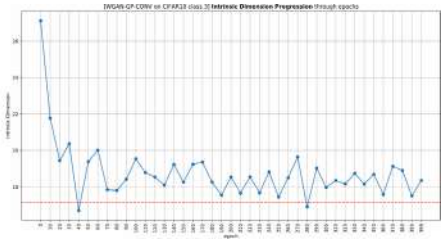


7(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. 7. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

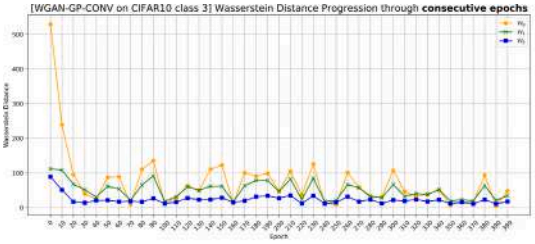


8(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

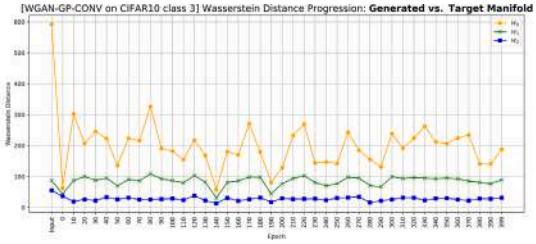


8(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

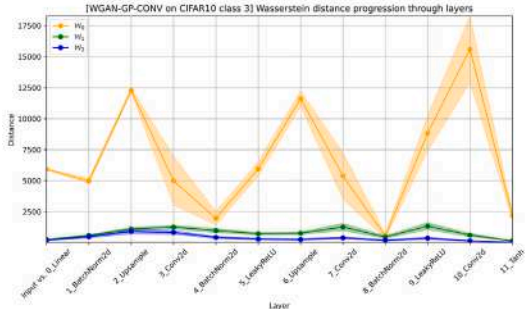
Fig. 8. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



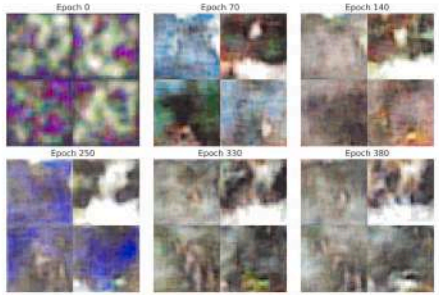
9(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



9(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



9(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.



9(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. 9. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

6. Conclusion

In this work, we provide strong empirical evidence for the critical role that the manifold support of the true data distribution plays in the overall behaviour of GAN models. Leveraging the manifold hypothesis, we motivate a TDA-based approach for characterising and studying the generative process within these models. In particular, this characterisation captures both training instability and mode collapse which are often observed within GAN architectures but typically identified only through visual inspection of generated samples.

We used persistent homology as a tool for characterising the structure of the generated and true data manifolds, providing topological signatures of both in the form of PDs. We tracked the topological evolution of these manifolds during training (epoch wise) by computing the Wasserstein distance between (i) PDs of the generated and true data manifold, and (ii) consecutive PDs of the training data manifold. Furthermore, we explored the role that different layers play in transforming the topology of the latent manifold across various GAN architectures.

We demonstrated that calculating the Wasserstein distance between the PDs of the generated and target manifold across epochs can be an effective tool in determining whether or not a model has sufficient capacity for correctly learning the support of the true data distribution. This epoch-wise analysis allowed us to identify the presence of mode collapse, which is otherwise often deduced visually using generated samples.

By comparing the PDs of the generated manifold over consecutive epochs, we were able to track the evolution of its topology during training. In particular, we showed that sufficiently large spikes in $\mathcal{W}_0$ values corresponded to episodes of training instability, where the model attempted to drastically change the structure of the generated manifold. This resulted in worsening topological distance from the target and severe degradation of the corresponding generated samples.

Our approach demonstrated that traditional metrics for determining sample quality, such as FID and estimated ID, are not sufficiently informative as they are unable to capture important properties that are otherwise encoded in the homology of the data manifold. Moreover, all experiments demonstrated that the homology group H_0 played the most important role in determining training stability, convergence in topology of the generated manifold to the true one, and existence of mode collapse. This is to be expected since H_0 (the number of connected components) often corresponds to structures with larger geometry than those of higher dimensions.

Our layer-wise analysis provided interesting insights into the contribution of different layers in transforming the latent space manifold across various architectures. We noted that the vanilla GAN architecture concentrates topological transformations in its final layers, mirroring what was observed in MLPs for classification tasks [32]. On the other hand, we observed that WGAN-GP spreads topological transformations across both shallow and deep layers, resulting in a more controlled topological transformation of the latent manifold. Convolutional models are more difficult to interpret. It is clear that convolutional layers play an important role in modifying the latent manifold's topology, where the number of channels changes with each convolution operation. However, determining their precise role in transforming the overall topology of the latent manifold remains difficult due to the inherent complexity of the operation. Nevertheless, these insights imply a strong dependency between the specific layers in a given architecture and its capacity to produce generated manifolds of a given topological structure. An in-depth, dedicated study on this topic may prove useful for understanding the minimal architectural requirements for a GAN.

Finally, methods for computing the persistent homology of a set of points often require large computational resources. In particular, standard TDA libraries such as GHUDI, Ripser (as implemented here), and giotto-tda are computationally prohibitive for large data sets, as highlighted in [34]. Recently, the authors of [39] introduced a scalable topological regulariser for GANs based on *Principal Persistent Measures* (PPMs). These are computed over multiple small subsamples of the original dataset, enabling efficient homological computations through parallelisation. While PPMs generalise persistence diagrams, the authors emphasise that they are not intended as replacements. Adopting a similar strategy could provide a computationally efficient topological perspective on how GANs learn and offer additional insights into their generative processes.

This work aims to provide a strong motivation for the inclusion and exploration of TDA-based methods for the improvement and efficiency of generative model training, such as an effective early stopping criterion or guidelines for architecture refinement based on the generative task.

CRediT authorship contribution statement

Barbara Toniella Corradini: Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Investigation, Conceptualization. **Ben Cullen:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Caterina Gallegati:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Sara Marziali:** Methodology, Investigation, Formal analysis. **Giuseppe Alessio D'Inverno:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization. **Monica Bianchini:** Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Conceptualization. **Franco Scarselli:** Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT solely to improve the clarity and fluency of the language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study was co-funded by the European Union - Next Generation EU, in the context of The National Recovery and Resilience Plan - Investment 1.5 Ecosystems of Innovation, Project Tuscany Health Ecosystem (THE), Spoke 3 - Advanced technologies, methods and materials for human health and well-being. ECS00000017, CUP: B63C22000680007.

C.G. acknowledges the support provided by the European Union - NextGenerationEU - Mission 4, Component 1, Investment 4.1 under the PNRR, as part of the Ph.D. scholarship co-financed by Ministerial Decree No. 118/2023.

G.A.D. acknowledges the support provided by the European Union - NextGenerationEU, in the framework of the iNEST - Interconnected Nord-Est Innovation Ecosystem (iNEST ECS00000043 – CUP G93C22000610007) project and its CC5 Young Researchers initiative. The views and opinions expressed are solely those of the authors and do not necessarily reflect those of the European Union, nor can the

European Union be held responsible for them. In addition, G.A.D. would like to acknowledge INdAM–GNCS.

Appendix A. Topological data analysis

This appendix provides a more in-depth discussion of essential concepts introduced in Section 3. In particular, it presents the essential theory of persistent homology that allows for the study of topological features at multiple scales. See the following for a comprehensive introduction to TDA [11], persistent homology [15], and computational topology [16].

A.1. Simplicial complexes

Simplex. A *k-dimensional simplex*—or simply a *k-simplex*—σ on a vertex set V is any subset consisting of $k+1$ elements contained in V. For example, a 0-simplex consists of a singleton $\{v\}$, a 1-simplex is a set of two vertices $\{v_0, v_1\}$ that represents an edge, a 2-simplex is a set of three vertices $\{v_0, v_1, v_2\}$ that represents the face of a triangle, and so on.

Simplicial complex. A *simplicial complex* K on a vertex set V is a collection of non-empty subsets of V such that: (i) for every *vertex* $v \in V$, the singleton $\{v\}$ is an element of K, and; (ii) if τ and σ are both simplices such that $\tau \subset \sigma$ and $\sigma \in K$ then necessarily $\tau \in K$. The dimension of a simplicial complex K is given by $\dim(K) := \max\{\dim(\sigma) | \sigma \in K\}$. A simplex τ is a *face* of a simplex σ, denoted $\tau \leq \sigma$, if every vertex in τ is also a vertex in σ. An orientation $o : K_0 \to \mathbb{N}$ on K is a function that assigns unique natural numbers to the vertices K_0 of the simplex K. Then, every k-simplex $\sigma \in K$ can be uniquely written as an ordered $k+1$-tuple $(v_0, \dots, v_k) \in K_0 \times \cdots \times K_0$, known as *an ordered k-simplex*, where $o(v_0) < o(v_1) < \cdots < o(v_k)$.

Geometric realisations of simplicial complexes. The above definitions only refer to *abstract* simplices and simplicial complexes. As will be seen in the following, one would like to construct geometric versions of these objects from points in $\mathbb{R}^d$. Let $v_0, \dots, v_k \in \mathbb{R}^d$ be a collection of $k+1$ affinely independent points. Then, the *geometric realisation* of the k-simplex $\sigma := \{v_0, \dots, v_k\}$ is given by the convex hull:

$$\left\{ x \in \mathbb{R}^d \mid x = \sum_{i=0}^{k} \lambda_i v_i \text{ where } \sum_{i=0}^{k} \lambda_i = 1 \right\}. \tag{A.1}$$

Hence, the geometric realisation of an abstract 0, 1, and 2-simplex in $\mathbb{R}^d$ is a point, a line segment, and a solid triangle, respectively. The geometric realisation of a m-dimensional simplicial complex for $m \leq d$ consists of the geometric realisations of all its constituent simplices in $\mathbb{R}^d$ glued together.

A.2. Simplicial homology

Algebraic topology aims to characterise topological spaces by assigning them algebraic objects such as groups, chains, and other constructs. Two topological spaces can then be compared using these algebraic objects, determining whether or not they are topologically equivalent to one another. *Homology* is one such method, where a topological space is characterised by its *homology groups* $\mathbf{H}_k$. These groups encode the presence of k-dimensional topological features such as the connected components ($k=0$), holes ($k=1$), voids ($k=2$), and higher dimensional analogues of the space.

In order to construct the homology groups of a simplicial complex K, one first constructs a chain of vector spaces where each vector space can be mapped to the next via a linear map known as a *boundary operator*. To simplify the following discussion, only vector spaces over the coefficient ring $\mathbb{F}_2 = \{0, 1\}$ are considered. For a more extensive treatment of homology theory, see [40].

Chain group. Let K be a simplicial complex and $K^{(k)} = \{\sigma_1, \dots, \sigma_m\}$ be the collection of all k-simplexes in K. Then the *k-th chain group of K over $\mathbb{F}_2$*, denoted $\mathbf{C}_k(\mathbb{F}_2; K)$, is a vector space whose basis elements are the simplexes $K^{(k)}$ and whose coefficient ring is $\mathbb{F}_2$. Every element $\gamma \in \mathbf{C}_k(\mathbb{F}_2; K)$—known as a *$k$-chain*—is a unique linear combination of elements in $K^{(k)}$ and can be expressed as:

$$\gamma = \sum_{i=1}^{m} \gamma_\sigma \sigma_i \quad \text{where} \quad \gamma_\sigma \in \mathbb{F}_2. \tag{A.2}$$

For simplicity, $\mathbf{C}_k(\mathbb{F}_2; K)$ will be denoted as $\mathbf{C}_k(K)$ from now on.

Boundary operator. Let K be an ordered simplicial complex and $\mathbf{C}_k(K)$ be its k-th chain group. The *k-th boundary operator* acting on a basis k-chain $\sigma = (v_0, \dots, v_k)$ is defined in the following way:

$$\partial_k^K \sigma = \sum_{i=0}^{k} (v_0, \dots, v_{i-1}, \hat{v}_i, v_{i+1}, \dots, v_k) = \sum_{i=0}^{k} \sigma_{-i}, \tag{A.3}$$

where $\hat{v}_i$ indicates the i-th vertex of σ is removed to form a $(k-1)$-simplex σ_{-i} known as the *i-th face of σ*. Now, taking any arbitrary k-chain $\gamma \in \mathbf{C}_k(K)$, one has:

$$\partial_k^K \gamma = \sum_{i=1}^{m} \gamma_{\sigma_i} \partial_k^K \sigma_i = \sum_{i=1}^{m} \sum_{j=0}^{k} \gamma_{\sigma_i} (\sigma_i)_{-j}. \tag{A.4}$$

Hence, $\partial_k^K : \mathbf{C}_k(K) \to \mathbf{C}_{k-1}(K)$ is a linear map with the property $\partial_k \circ \partial_{k+1} = 0$, i.e., *"a boundary of a boundary is trivial"*. This property suggests that, for each dimension k, $\mathbf{B}_k(K) := \text{Img}\left(\partial_{k+1}^K\right) \subseteq \text{Ker}\left(\partial_k^K\right) =: \mathbf{Z}_k(K)$, where elements of $\mathbf{B}_k(K)$ are called *k-boundaries* and elements of $\mathbf{Z}_k(K)$ are called *k-cycles*. Both $\mathbf{B}_k(K)$ and $\mathbf{Z}_k(K)$ form subspaces of $\mathbf{C}_k(K)$.

Homology group. With boundary operators and chain groups on the m-dimensional simplicial complex K, one can construct the following *chain complex*:

$$0 \xrightarrow{\partial_{m+1}^K} \mathbf{C}_m(K) \xrightarrow{\partial_m^K} \mathbf{C}_{m-1}(K) \xrightarrow{\partial_{m-1}^K} \dots \xrightarrow{\partial_2^K} \mathbf{C}_1(K) \xrightarrow{\partial_1^K} \mathbf{C}_0(K) \xrightarrow{\partial_0^K} 0,$$

and define a quotient vector space called the *k-th homology group of K*—denoted $\mathbf{H}_k(K)$—as:

$$\mathbf{H}_k(K) := \frac{\mathbf{Z}_k(K)}{\mathbf{B}_k(K)} = \frac{\text{Ker}\left(\partial_k^K\right)}{\text{Img}\left(\partial_{k+1}^K\right)} \quad \text{for} \quad k = 0, 1, \dots, m. \tag{A.5}$$

Hence, $\mathbf{H}_k(K)$ represents the k-cycles that are not boundaries of higher dimensional chains and therefore capture *"non-trivial"*

topological features within K. The rank of $\mathbf{H}_k(K)$ corresponds to the number of such topological features present in K and is known as the *k-th Betti number*:

$$\beta_k(K) := \operatorname{rank}\left(\mathbf{H}_k(K)\right) \quad \text{for} \quad k = 0, 1, \dots, m, \tag{A.6}$$

where $\beta_k(K)$ counts the number of k-dimensional voids present in K.

Simplicial maps. In order to develop the theory of persistent homology, one last ingredient is needed: simplicial maps. A *simplicial map* $f : K \to L$ is a map that assigns the vertex set of one simplicial complex K to that of another L such that simplices in K are sent to simplices in L. If $K \subseteq L$, then the *inclusion map* $g : K \hookrightarrow L$ sends each simplex in K to the same simplex in L. A simplicial map $f : K \to L$ induces a linear map between the chain complexes $\mathbf{C}_k(K)$ and $\mathbf{C}_k(L)$:

$$f : \mathbf{C}_k(K) \to \mathbf{C}_k(L) \quad \text{where} \quad \gamma := \sum_{i=1}^{m} \gamma_\sigma \sigma_i \mapsto \sum_{i=1}^{m} \gamma_\sigma f(\sigma_i) =: f(\gamma), \tag{A.7}$$

which in turn induces a linear map $H_k(f) : \mathbf{H}_k(K) \to \mathbf{H}_k(L)$ between the k-th homology groups of K and L. Finally, given two simplicial maps $f_1 : K \to L$ and $f_2 : L \to M$, their composition $f_2 \circ f_1 : K \to M$ is also a simplicial map, and hence also induces a linear map $H_k(f_2 \circ f_1)$ between the homology groups of K and M.

A.3. Persistent homology

In many applications of TDA, a finite point cloud of data $\mathcal{P}$ is assumed to be either drawn directly from, or close to, a manifold $\mathcal{M} \subseteq \mathbb{R}^d$. The difficulty then lies in determining a simplicial complex whose homology meaningfully reflects that of $\mathcal{M}$. The authors in [33] proved the following proposition:

Proposition A.1 (Niyogi-Smale-Weinberger). *Let $\mathcal{M} \subseteq \mathbb{R}^d$ be a compact Riemannian manifold with condition number τ, and $\mathcal{P} := \{v_1, \dots, v_n\} \subseteq \mathbb{R}^d$ be $\epsilon/2$-dense in $\mathcal{M}$. Then, for any $\epsilon < \sqrt{3\tau/5}$, the union of balls $U := \bigcup_{v \in \mathcal{P}} B_\epsilon(v)$ deformation retracts to $\mathcal{M}$. Hence, the homology of U is equal to that of $\mathcal{M}$.*

The above proposition implies that under certain geometric and topological conditions of the manifold $\mathcal{M}$, there exists a geometric scale ϵ such that the homology of U and $\mathcal{M}$ are equivalent. Although a remarkable result, it is not possible to ensure these conditions on $\mathcal{M}$ nor on $\mathcal{P}$ *a-priori*. Hence, one must resort to a different method for estimating the homology of the unknown manifold.

Instead, *persistent homology* [14] is employed: a blended topological-geometric method that tracks the evolution of topological features of a point cloud over different geometric scales. This method requires the construction of a *filtration* on $\mathcal{P}$, i.e., a sequence of nested simplicial complexes whose vertices are the points in $\mathcal{P}$. For each simplicial complex in the filtration, its corresponding homology groups are computed and then tracked over the entire duration of the filtration using a series of inclusion maps. This allows one to study at what scales different topological features appear and for how long they persist. With this information, properties of the underlying manifold $\mathcal{M}$ can be inferred.

Vietoris-rips filtration. In order to construct a filtration, one first needs a way to build simplicial complexes from a point cloud of data $\mathcal{P} \subseteq \mathbb{R}^d$. Since many different constructions exist, the following discourse is restricted to a popular choice: the Vietoris-Rips complex. Given a point cloud $\mathcal{P}$, a metric δ on $\mathcal{P}$, and a scale $\rho \geq 0$, the *Vietoris-Rips complex* $\mathrm{VR}_\rho(\mathcal{P})$ is given by the set of simplices:

$$\mathrm{VR}_\rho(\mathcal{P}) := \left\{ \{u_0, \dots, u_k\} : \delta(u_i, u_j) \leq 2\rho \text{ for all } i, j \in [k], \text{ where } u_0, \dots, u_k \in \mathcal{P} \right\}. \tag{A.8}$$

Intuitively, the filtration starts by considering centred balls around each point u_i with radius equal to zero. Then by slowly expanding the radius until balls of at least two points intersect, its corresponding simplicial complex is added to the filtration. The reader may convince themselves that Eq. (A.8) is indeed a simplicial complex by noting that: (i) the isolated vertices are always included in $\mathrm{VR}_\rho(\mathcal{P})$ since $\delta(u_i, u_i) = 0$, and (ii) for any pair of simplices σ and τ where $\tau \leq \sigma$, $\sigma \in \mathrm{VR}_\rho(\mathcal{P})$ implies $\tau \in \mathrm{VR}_\rho(\mathcal{P})$.

For any sequence of scales $0 \leq \rho_1 \leq \rho_2 \leq \cdots \leq \rho_m$, we have the following chain of inclusions:

$$\mathrm{VR}_0(\mathcal{P}) \subseteq \mathrm{VR}_{\rho_1}(\mathcal{P}) \subseteq \mathrm{VR}_{\rho_2}(\mathcal{P}) \subseteq \cdots \subseteq \mathrm{VR}_{\rho_m}(\mathcal{P}). \tag{A.9}$$

Eq. (A.9) is known as a *Vietoris-Rips filtration of* $\mathcal{P}$ and will form the basis for the following presentation of persistent homology. Often ρ is referred to as the *filtration parameter* since it determines a Vietoris-Rips complex within the filtration.

Persistent homology. In order to compute the persistent homology for a point cloud $\mathcal{P}$, one first constructs a Vietoris-Rips filtration:

$$K_0 \subseteq K_1 \subseteq K_2 \subseteq \cdots \subseteq K_m \quad \text{where} \quad K_l = \mathrm{VR}_{\rho_l}(\mathcal{P}). \tag{A.10}$$

We let $g_j : K_j \hookrightarrow K_{j+1}$ be the inclusion map that sends the simplices in K_j to the same simplices in K_{j+1}. Since g_j is a simplicial map, it induces a linear map $H_k(g_j)$ between the homology groups $\mathbf{H}_k(K_j)$ and $\mathbf{H}_k(K_{j+1})$. The image of $H_k(g_j)$ contains precisely the elements of the homology group $\mathbf{H}_k(K_j)$ that are also present in $\mathbf{H}_k(K_{j+1})$.

Now, consider the composition of inclusion maps $g_{j,j+p} := g_{j+p} \circ \cdots \circ g_j$ for $j = 0, 1, \dots, m$ and $p = 1, 2, \dots, m - j$. It also induces a linear map between the homology groups $\mathbf{H}_k(K_j)$ and $\mathbf{H}_k(K_{j+p})$. Hence, the image of $H_k(g_{j,j+p})$ consists of all the elements of the homology group $\mathbf{H}_k(K_j)$ that *persist* in the interval (ρ_j, ρ_{j+p}). Similarly, the kernel of $H_k(g_{j,j+p})$ consists of all the elements in $\mathbf{H}_k(K_j)$ that *die* in the interval (ρ_j, ρ_{j+p}).

The following section presents how this result can then be used to track the persistence of each homological feature, across the entire filtration rather than just the interval $\rho \in (\rho_j, \rho_{j+p})$.

A.4. Measuring persistent homology

The induced linear maps $H_k(g_{j,j+p})$ contain crucial information for connecting the k-th homology groups of each of the complexes appearing in a filtration. Consider the following chain of vector spaces:

$$\mathbf{H}_k(K_0) \xrightarrow{H_k(g_0)} \mathbf{H}_k(K_1) \xrightarrow{H_k(g_1)} \cdots \xrightarrow{H_k(g_{m-2})} \mathbf{H}_k(K_{m-1}) \xrightarrow{H_k(g_{m-1})} \mathbf{H}_k(K_m). \tag{A.11}$$

For each pair of integers i, j, the sequence in Eq. (A.11) allows us to determine the associated *persistent homology group*, which is defined as the subset of $\mathbf{H}_k(K_j)$ given by

$$\mathbf{PH}_k(g_{i,j}) = \mathrm{Img}(g_{i,j}). \tag{A.12}$$

In particular, $\mathbf{PH}_k(g_{i,j})$ is the set of all homological features that appear in $\mathbf{H}_k(K_i)$ and also appear in $\mathbf{H}_k(K_j)$—that is, it contains all k-dimensional elements that persist over the interval (i, j).

However, we are not interested in the elements that persist over an interval (i, j); we would like to consider the elements that *appear* at filtration index i and then *disappear* after filtration index j. In particular, an element $v \in \mathbf{H}_k(K_i)$ is said to be *born* at filtration index i if $v \notin \mathrm{Img}(g_{i-1})$ and is said to *die* at filtration index $j \geq i$ which is the smallest j such that $v \in Ker(H_k(g_{i,j}))$. By convention, if such a j does not exist then the death index of v is considered to be $+\infty$.

The *persistence* of v is defined as death minus birth, i.e., $(j - i)$, and $\mathbf{PH}_k(g_{i,j})$ consists precisely of those homology classes in $\mathbf{H}_k(K_i)$ that continue to generate nontrivial homology in the larger complex $\mathbf{H}_k(K_j)$. From a geometrical point of view, they correspond to the equivalence classes of k-cycles that do not become k-boundaries in $\mathbf{H}_k(K_j)$.

Graphical representations. In order to provide a graphical representation of how homological features persist during a filtration, different methods exist. One such method is the *persistence diagram* (PD). A PD is a finite multiset of points in the cartesian plane $\mathbb{R}^2$, where each point (ρ_i, ρ_j) corresponds to a homological feature—trivial or otherwise—whose birth occurs at scale ρ_i and whose death occurs at scale ρ_j. The multiset $\Delta = \{(c, c) \mid c \in \mathbb{R}\}$ corresponds to the trivial elements along the diagonal of the PD, whereas off diagonal elements correspond to those of the homology groups $\mathbf{H}_k(K_i)$. Hence, points that are far from the diagonal represent features with greater persistence.

Other representations of the persistent homology of a filtration are *barcodes* and *Betti curves*. The first keeps track of birth and death of the features in the form of persistence intervals. Indeed, the plot draws these intervals by means of bars, whose lengths represent the lifespans of the corresponding features within a specific filtration range. The latter represents the evolution of Betti numbers of each homology group with respect to increasing the filtration parameter ρ.

Metrics. In theory, a PD is a finite multiset of points in $\mathbb{R}^2$ above the diagonal Δ, jointly with a countably infinite set of points along the diagonal. In practice, however, the most meaningful part of a PD is characterised by the points on the off-diagonal. Therefore, when considering a metric for these representations, it is important to bear in mind that a point near the diagonal $(c, c + \epsilon)$ represents a feature that has persisted for a short time ϵ. A diagram with this point of short persistence should be intuitively 'close' to the same diagram without that point, as if the feature had never appeared. Consequently, the distance between two diagrams is considered to be the minimum cost required to make their points coincide, matching off-diagonal to off-diagonal, or off-diagonal to the nearest point on the diagonal, with respect to a cost function for the correspondence [6]. Two closely related options for a metric on PDs are the Wasserstein distance and the bottleneck distance.

The *p-th Wasserstein distance*, $p \geq 1$, between two diagrams X, Y is defined as:

$$\mathcal{W}_p(X, Y) = \inf_{\phi: X \to Y} \left(\sum_{a \in X} \| a - \phi(a) \|_p^q \right)^{\frac{1}{q}}, \tag{A.13}$$

where the infimum is taken over all bijections ϕ between X and Y. For historical reasons, $q = \infty$ is used in the majority of applications. The limit case of p going to infinity, also known as the *bottleneck distance*, is defined as:

$$\mathcal{W}_\infty(X, Y) = \inf_{\phi: X \to Y} \sup_{a \in X} \| a - \phi(a) \|_\infty . \tag{A.14}$$

Appendix B. Implementation details

Table B.1 provides a more precise description of the implementation details for GAN architectures presented in Section 4.

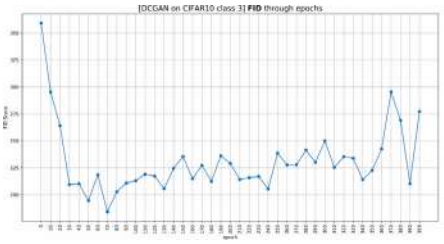
Table B.1
Architectural description of the four generative models evaluated in this work. All generators take as input a latent vector $z \in \mathbb{R}^d$. "ConvT" denotes transposed convolution, "BN" is Batch Normalisation, and "LReLU" is LeakyReLU with slope 0.2. Models differ in both architectural style (MLP vs convolutional) and training objective: standard GANs use binary cross-entropy loss, while WGAN-GP models use the Wasserstein loss with gradient penalty (GP). The presence of GP is indicated in the last column.

Model	Generator (G)	Discriminator / Critic (D)	Loss	GP
GAN (MLP)	$z \to$ Linear(128) $\to$ LReLU $\to$ Linear(256) $\to$ LReLU $\to$ Linear(512) $\to$ LReLU $\to$ Linear(img) $\to$ Tanh	$x \to$ Linear(512) $\to$ LReLU $\to$ Linear(256) $\to$ LReLU $\to$ Linear(1) $\to$ Sigmoid	BCE Loss	✗
DCGAN	$z \to$ Linear(256×7×7) $\to$ Reshape $\to$ ConvT(256→128) $\to$ BN $\to$ ReLU $\to$ ConvT(128→64) $\to$ BN $\to$ ReLU $\to$ ConvT(64→1) $\to$ Tanh	$x \to$ Conv(1→64) $\to$ LReLU $\to$ Conv(64→128) $\to$ BN $\to$ LReLU $\to$ Flatten $\to$ Linear(1) $\to$ Sigmoid	BCE Loss	✗
WGAN-GP (MLP)	$z \to$ Linear(128) $\to$ BN $\to$ LReLU $\to$ Linear(256) $\to$ BN $\to$ LReLU $\to$ Linear(512) $\to$ BN $\to$ LReLU $\to$ Linear(img) $\to$ Tanh	$x \to$ Linear(512) $\to$ LReLU $\to$ Linear(256) $\to$ LReLU $\to$ Linear(1)	Wasserstein + GP	✓
WGAN-GP (Conv)	$z \to$ Linear(256×7×7) $\to$ Reshape $\to$ ConvT(256→128) $\to$ BN $\to$ LReLU $\to$ ConvT(128→64) $\to$ BN $\to$ LReLU $\to$ ConvT(64→1) $\to$ Tanh	$x \to$ Conv(1→64) $\to$ LReLU $\to$ Conv(64→128) $\to$ BN $\to$ LReLU $\to$ Flatten $\to$ Linear(1)	Wasserstein + GP	✓

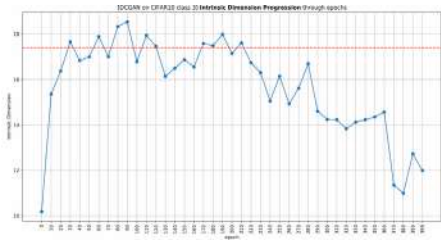
Appendix C. Results and discussion

DCGAN on CIFAR-10 (Class 3)

See Fig. C.10 and C.11.

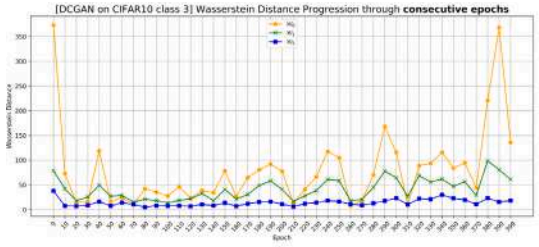


C.10(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

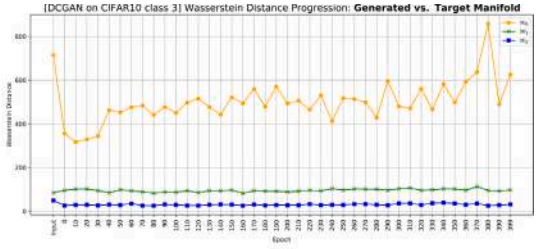


C.10(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

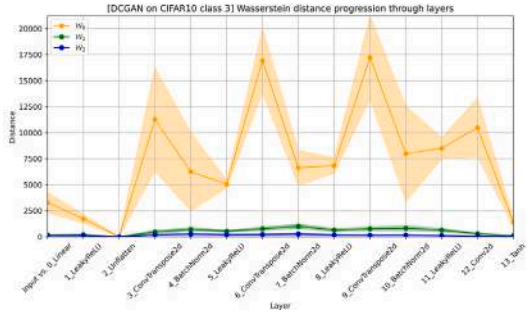
Fig. C.10. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



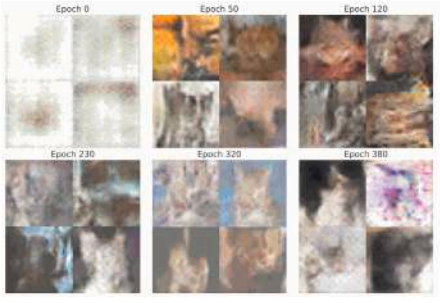
C.11(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.11(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.11(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

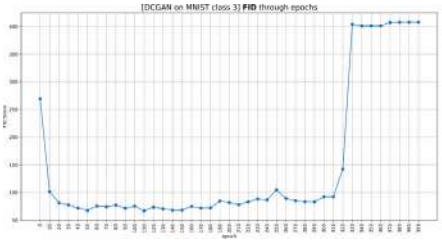


C.11(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. C.11. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

DCGAN on MNIST (Class 3)

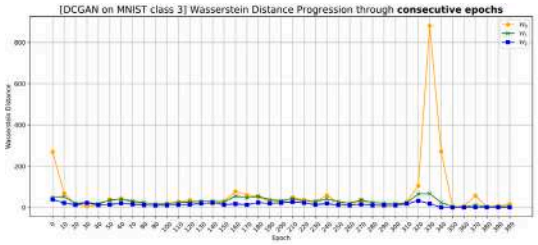
See Fig. C.12 and C.13.



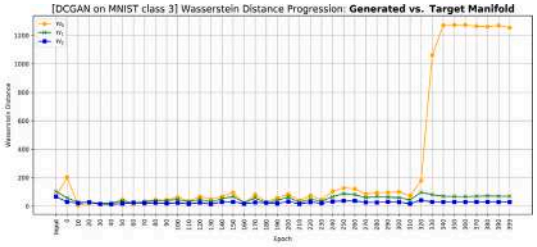
C.12(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

C.12(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

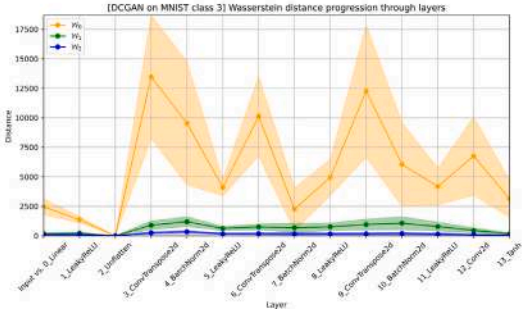
Fig. C.12. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



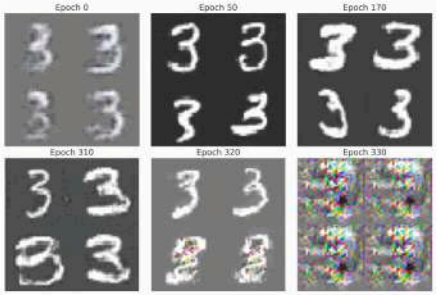
C.13(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.13(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.13(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

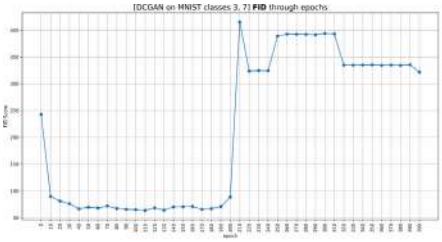


C.13(d) **Generated samples.** We report samples generated during training at different epochs.

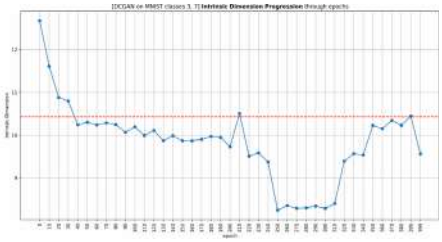
Fig. C.13. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

DCGAN on MNIST (Classes 3 and 7)

See Fig. C.14 and C.15.

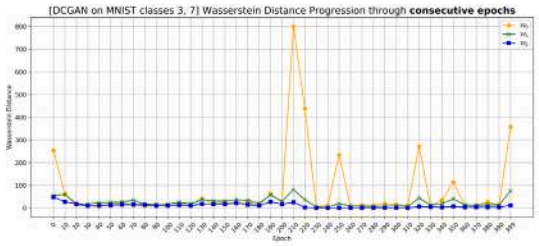


C.14(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

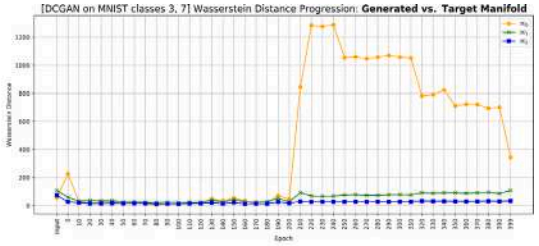


C.14(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

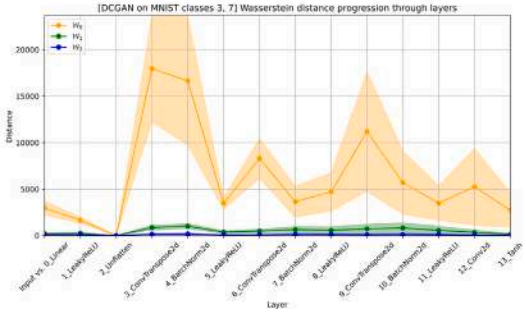
Fig. C.14. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



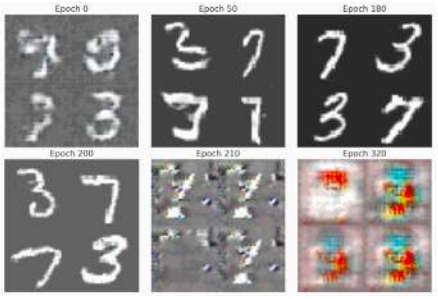
C.15(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.15(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.15(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

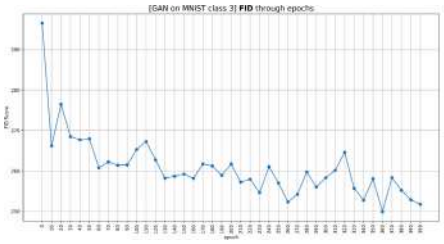


C.15(d) **Generated samples.** We report samples generated during training at different epochs.

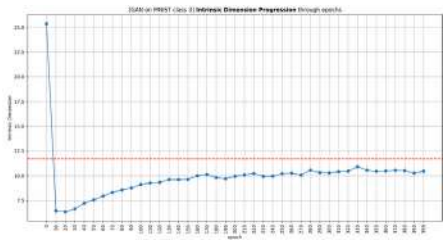
Fig. C.15. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

GAN on MNIST (Class 3)

See Fig. C.16 and C.17.

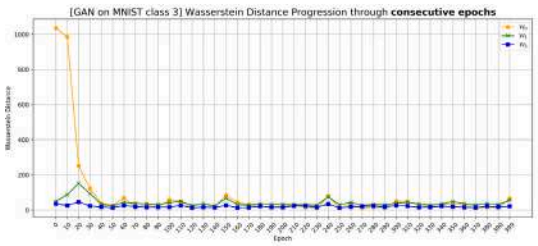


C.16(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

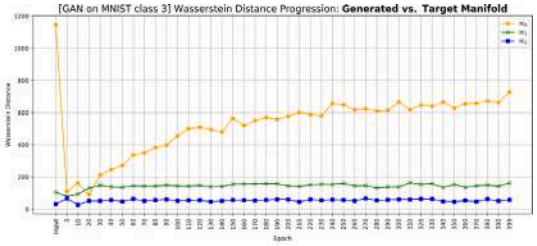


C.16(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

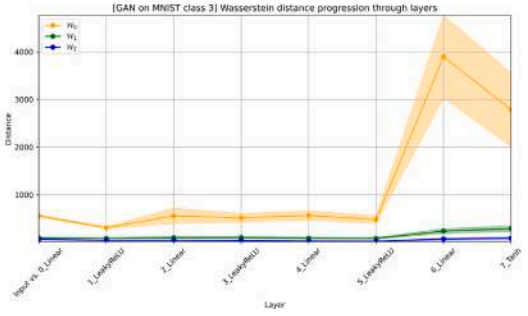
Fig. C.16. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



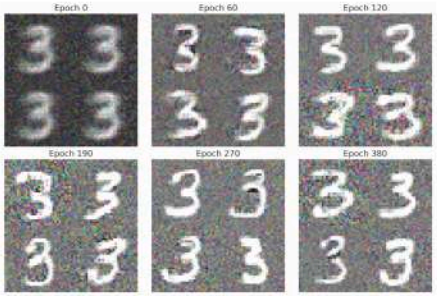
C.17(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.17(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.17(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

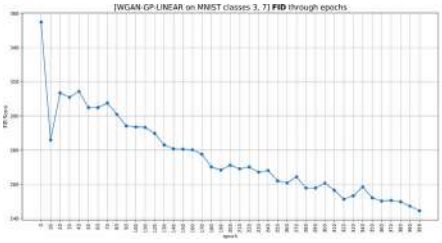


C.17(d) **Generated samples.** We report samples generated during training at different epochs.

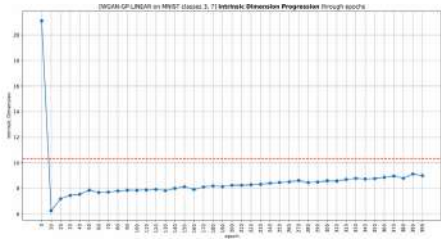
Fig. C.17. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

Linear WGAN-GP on MNIST (Classes 3 and 7)

See Fig. C.18 and C.19.

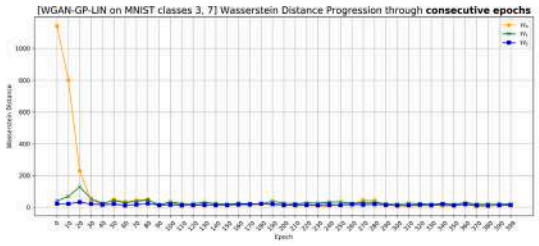


C.18(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

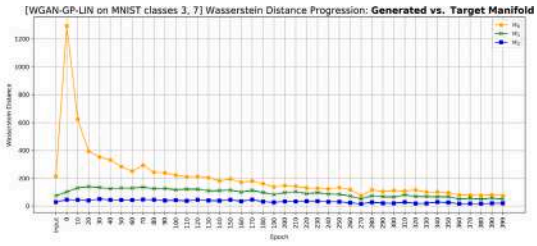


C.18(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

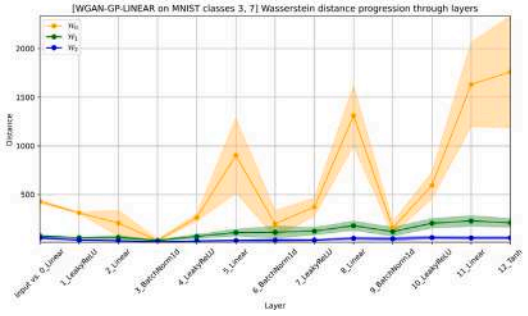
Fig. C.18. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



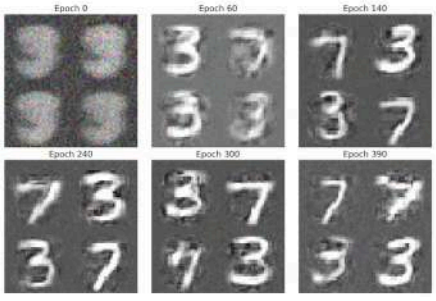
C.19(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.19(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.19(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

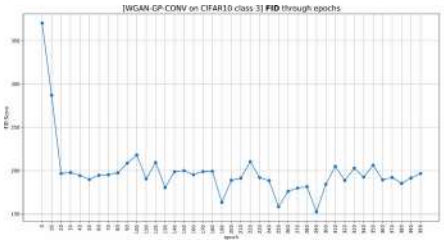


C.19(d) **Generated samples.** We report samples generated during training at different epochs.

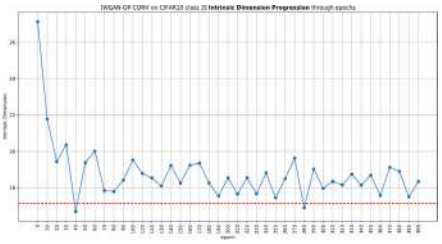
Fig. C.19. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

Convolutional WGAN-GP on CIFAR-10 (Class 3)

See Fig. C.20 and C.21.

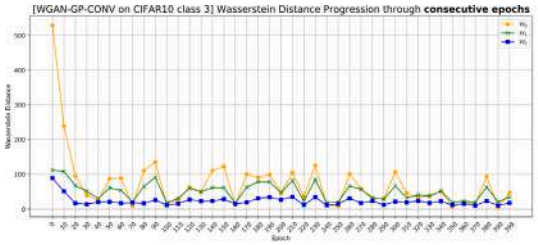


C.20(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

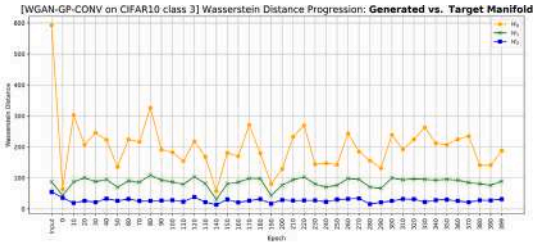


C.20(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

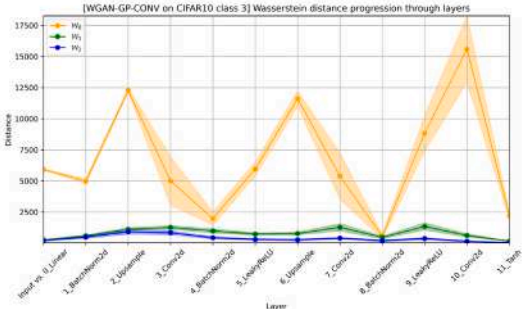
Fig. C.20. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



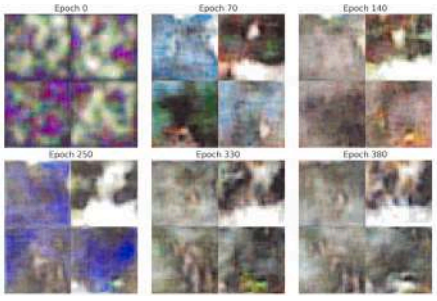
C.21(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.21(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.21(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

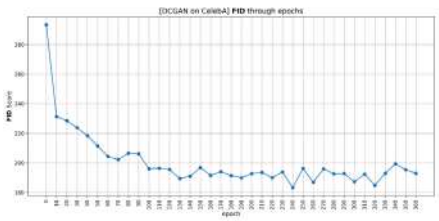


C.21(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. C.21. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

DCGAN on CelebA

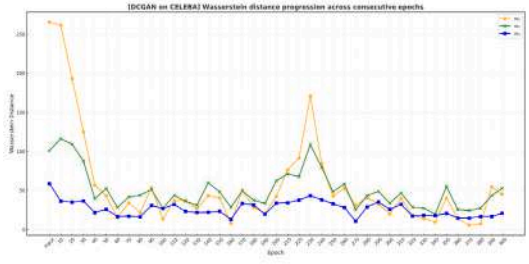
See Figs. C.22 and C.23.



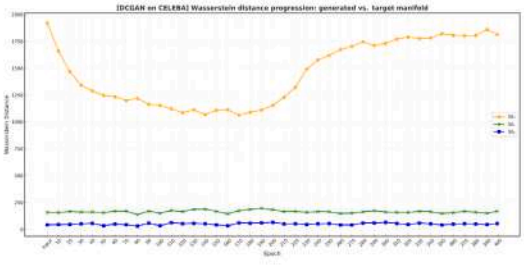
C.22(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

C.22(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

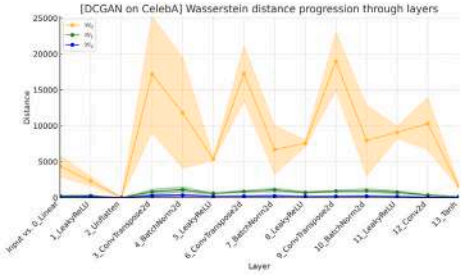
Fig. C.22. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



C.23(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.23(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.23(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

C.23(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. C.23. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

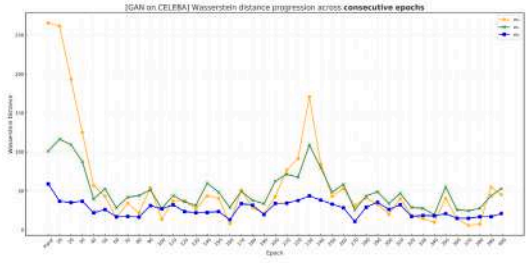
GAN on CelebA

See Figs. C.24 and C.25.

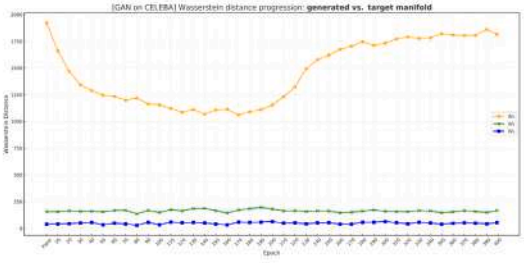
C.24(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

C.24(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

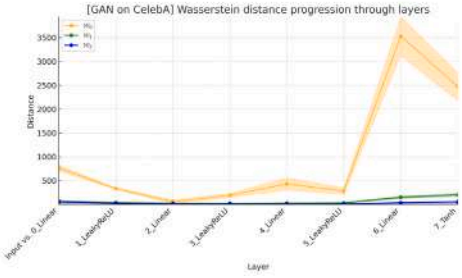
Fig. C.24. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



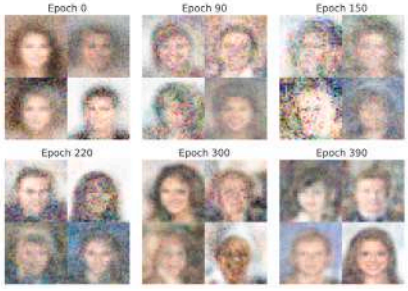
C.25(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.25(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.25(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

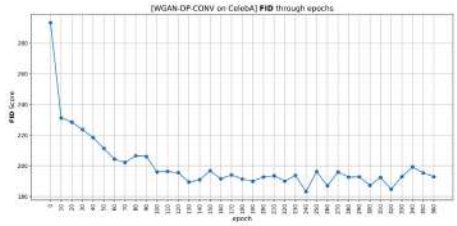


C.25(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. C.25. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

Convolutional WGAN-GP on CelebA

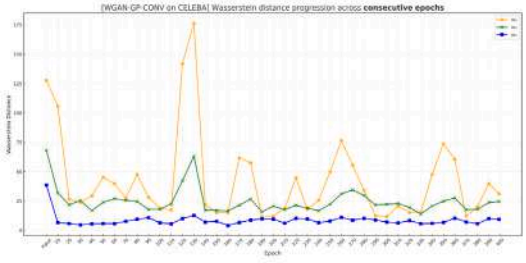
See Figs. C.26 and C.27



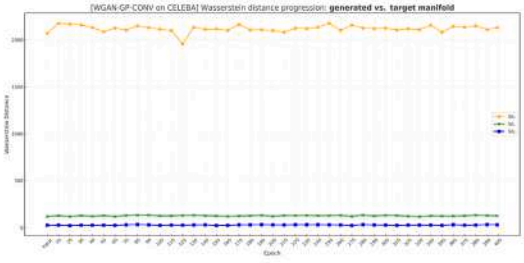
C.26(a) **FID score progression of generated samples for each epoch.** This metric quantifies the model's convergence and generative quality improvement.

C.26(b) **Intrinsic dimension progression of generated samples for each epoch.** The red dashed line indicates the intrinsic dimension measured on the target dataset. This metric reflects the model's ability to represent the data manifold.

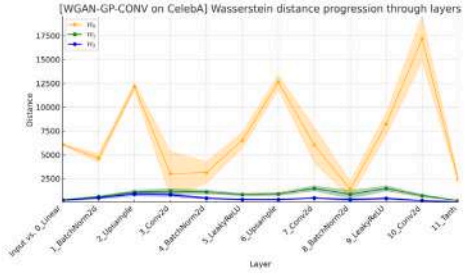
Fig. C.26. Generated image quality assessment. The FID score (a) and intrinsic dimension (b) progression provide insights into the model's ability to generate realistic images and capture data complexity. The model is trained over 400 epochs.



C.27(a) **About training stability.** Comparing the PD of each epoch with the previous one reveals how the model evolves its generated manifold.



C.27(b) **About converging to the target manifold.** Tracking the Wasserstein distance between generated and target manifolds can reveal the model's expressive power.



C.27(c) **Layer-wise manifold dynamics.** The mean and standard deviation of the Wasserstein distance between PDs of consecutive layers computed across all the training epochs.

C.27(d) **Generated samples.** We report samples generated during training at different epochs.

Fig. C.27. Topology-based analysis. The progression of Wasserstein distances across homology dimensions $\mathcal{W}_0$ (orange), $\mathcal{W}_1$ (green), and $\mathcal{W}_2$ (blue) reveals how closely the generated manifold aligns with target topological structures (a), offering insights into the stability and quality of the training process (b). The model is trained over 400 epochs.

Appendix D. On the choice of n for persistent homology calculation

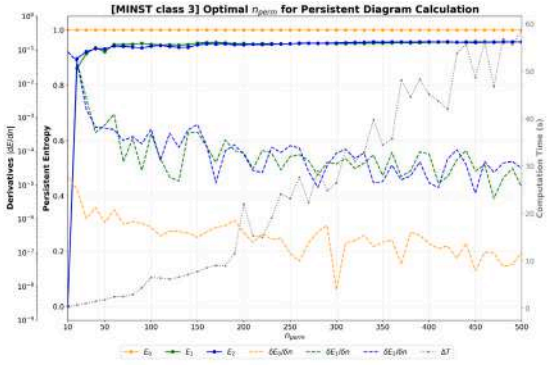
In this appendix we clarify how we select the number of points n used by the greedy permutation algorithm for computing persistence diagrams (PDs) (see [3] for Ripser algorithm details). Our goal is to identify an n that maximizes the informativeness of the resulting PDs while keeping the computation time within a reasonable budget.

To quantify informativeness we use the normalised *persistence entropy* [2,12,31] (PE) of each homology class E_i. We normalise each PE value as in [31] by dividing it by the sum of lifespans for the same homology class, so that values lie in $[0, 1]$. By construction, $E_i = 1$ corresponds to maximal entropy (all intervals of equal length $1/n$) and $E_i = 0$ to minimal entropy (a single long interval, with others negligible). Infinite-persistence features are excluded to avoid trivial zero-entropy dominance.

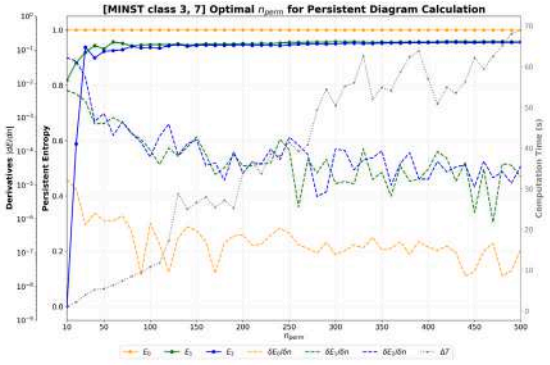
Since $E_i \in [0, 1]$ and our smallest feasible increment is $\delta n = 1$, the empirical derivative $\delta E_i/\delta n$ lies in $[0, 1]$. A derivative on the order of 10^{-3} therefore indicates only a very small change in normalised entropy as n increases, i.e., the addition of points contributes essentially redundant topological information. We use this derivative to detect the saturation point of informativeness with respect to n.

We carry out this analysis on the MNIST and CIFAR-10 training sets (both single- and two-class scenarios). For each dataset, we compute PE for each homology class i using between 10 and 500 points sampled with the greedy permutation (in steps of 10, i.e., $n = 10, 20, \ldots, 500$). For each n we also measure the corresponding PD computation time. Fig. D.28 shows: the normalised PE E_i as a function of n, the derivative $\delta E_i/\delta n$, and the computation time ΔT per PD.

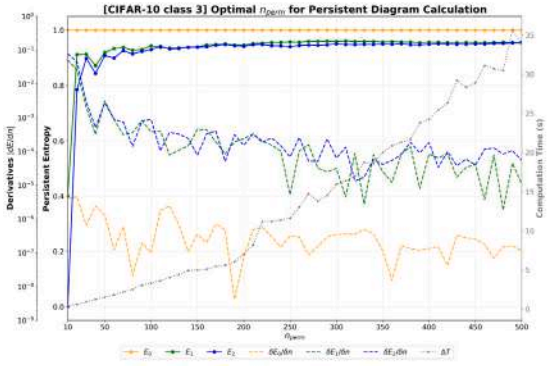
Across all datasets and homology classes, E_i increases with n initially but plateaus beyond $n \simeq 100$. Correspondingly, $\delta E_i/\delta n$ drops to $\mathcal{O}(10^{-3})$ or less, indicating that additional points beyond $n = 100$ add negligible new topological information. In contrast, computation time continues to grow sharply with n, with $n = 500$ already at the limit of our hardware capacity.



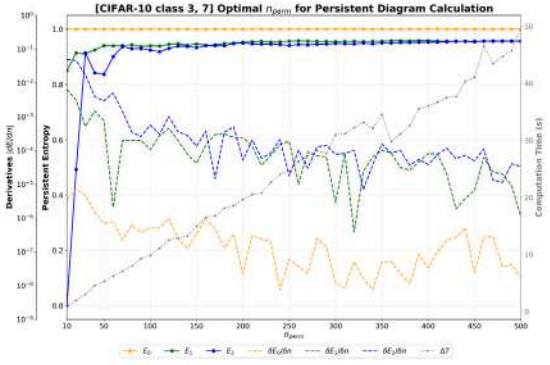
D.28(a) MNIST, class 3



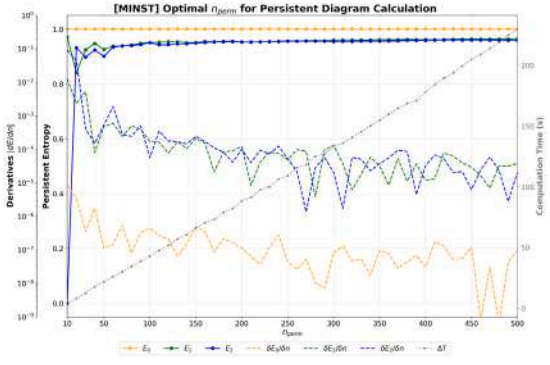
D.28(b) MNIST, classes 3 and 7



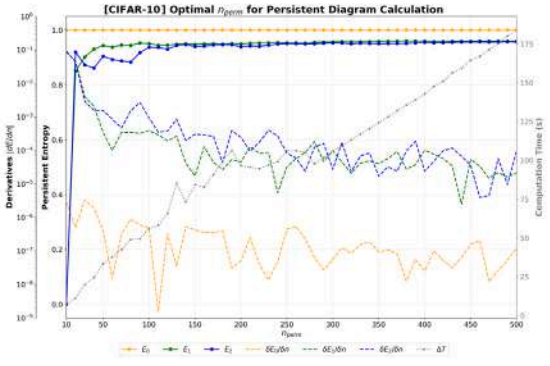
D.28(c) CIFAR10, class 3



D.28(d) CIFAR10, classes 3 and 7



D.28(e) MNIST, all classes



D.28(f) CIFAR10, all classes

Fig. D.28. Persistent entropy and its derivatives as a function of the number of used by the greedy permutations (n_{perm}) for different datasets and class selections. The computation time (ΔT) is also shown.

We therefore set $n = 100$ as a trade-off point balancing topological informativeness and computational feasibility. This value reflects (i) saturation in normalised PE, (ii) a derivative $\delta E_i/\delta n$ indicating trivial topological gain beyond this threshold, and (iii) manageable computation time on our available hardware. In other words, while larger n could in principle refine the PD slightly, the marginal gains in persistent entropy are negligible compared to the cost.

This procedure ensures that our PD calculations retain essentially all topological signals of the sampled data manifolds while remaining computationally practical.

D.1. Persistence entropy

Persistence entropy is a scalar statistic derived from a persistence diagram $D = \{(b_i, d_i)\}_{i=1}^{n}$, where each pair (b_i, d_i) encodes the birth and death of a topological feature identified via persistent homology. The lifespan of a feature is given by its *persistence* $p_i = d_i - b_i$, and the total persistence is defined as $S = \sum_{i=1}^{n} p_i$.

Assuming $S > 0$, we define the *normalised persistence values* as

$$\tilde{p}_i = \frac{p_i}{S}, \quad \text{for } i = 1, \dots, n,$$

which induces a discrete probability distribution over the lifespans. The *persistence entropy* is then defined in analogy to the Shannon entropy as follows:

$$E(D) = -\sum_{i=1}^{n} \tilde{p}_i \log \tilde{p}_i.$$

This measure quantifies the information content, or disorder, in the distribution of topological features.

In the presence of numerous low-persistence features, persistence entropy shows a *plateau behaviour* that indicates saturation in topological complexity: additional features cease to provide significant new information. Consequently, the plateauing of entropy serves as a practical indicator of structural noise and can be employed to guide model selection, thresholding, or scale sensitivity analysis in topological data pipelines.

Appendix E. Computation time versus estimation accuracy

We analyse the problem of choosing the number of samples n to balance computation time and estimation accuracy for intrinsic dimension (ID). We evaluate the scaling of ID estimation across multiple datasets to analyse the trade-off between computational cost and statistical accuracy. Our analysis considers the training sets of MNIST (60,000 images), CIFAR-10 (50,000 images), and CelebA (over 160,000 images), for both full datasets and restricted subsets, i.e., single classes and groups of classes. For each experimental condition, we repeated the computation three times and report the averaged results for both the estimated ID and the runtime. Figs. E.29 and E.30 display the resulting curves, showing the average intrinsic dimension (blue) and the corresponding computation time (grey) as a function of the number of images considered.

Discussion. The results show that ID estimation exhibits strong diminishing returns: additional samples reduce estimation error at a slow rate, while the runtime grows super-linearly. CelebA is particularly demanding, with halving the ID estimation error requiring more than an order of magnitude increase in time. This suggests that, in practice, selecting a moderate fraction of the dataset (20–30 % for MNIST or CIFAR-10, 15 % for CelebA) is a reasonable compromise between estimation stability and computational efficiency.

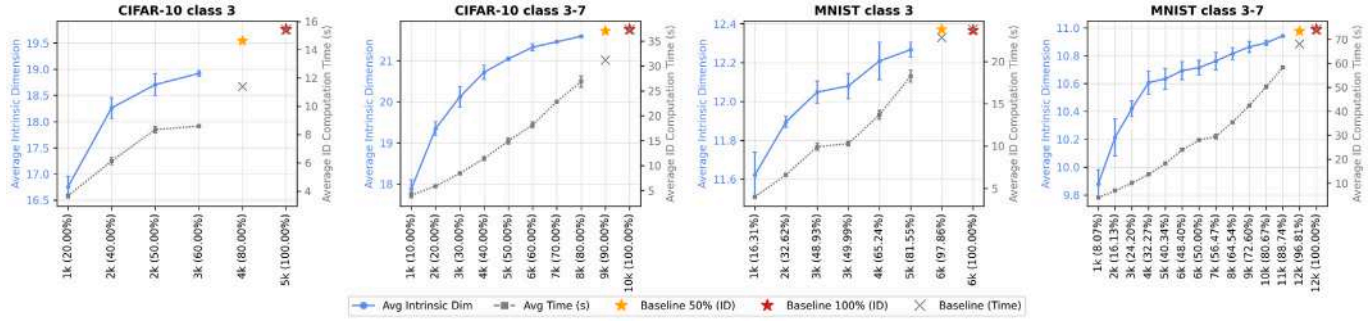


Fig. E.29. Comparative analysis of intrinsic dimension and computation time for specific subsets of CIFAR-10 and MNIST reveals that increasing the number of images leads to higher intrinsic dimensionality and longer computation times. The highlighted baseline points (stars for intrinsic dimension, crosses for time) at 50 % and 100 % dataset sizes provide clear reference points for evaluating model scalability and performance.

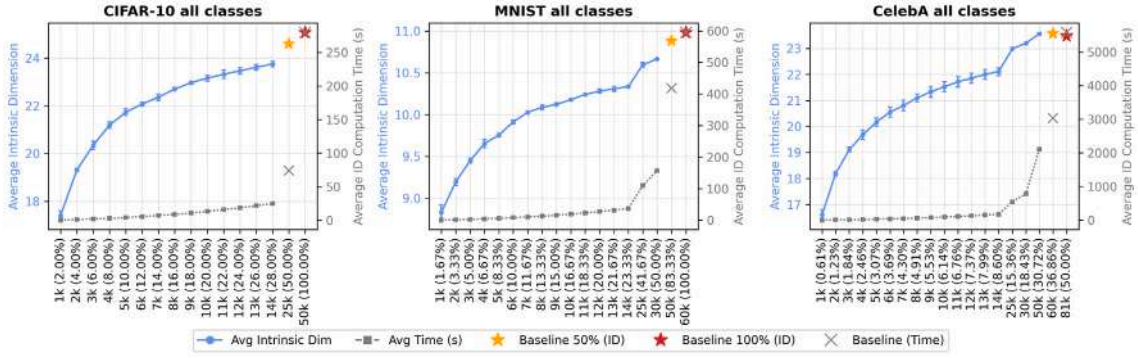


Fig. E.30. Average intrinsic dimension and computation time are plotted against the number of images (and dataset percentage) for CIFAR-10, MNIST, and CelebA. The intrinsic dimension shows a gradual increase, indicating richer feature representation with more data. Computation time scales roughly linearly, with baseline markers (orange star/gray cross at 50 %, red star/gray cross at 100 %) emphasizing the trade-off between dataset size and resource requirements.

Appendix F. Computation time versus estimation accuracy

We analyse the problem of choosing the number of samples n to balance computation time and estimation accuracy for intrinsic dimension (ID). We evaluate the scaling of ID estimation across multiple datasets to analyse the trade-off between computational cost and statistical accuracy. Our analysis considers the training sets of MNIST (60,000 images), CIFAR-10 (50,000 images), and CelebA (over 160,000 images), for both full datasets and restricted subsets, i.e., single classes and groups of classes. For each experimental condition, we repeated the computation three times and reported the averaged results for both the estimated ID and the runtime. Figs. E.29 and E.30 display the resulting curves, showing the average intrinsic dimension (blue) and the corresponding computation time (grey) as a function of the number of images considered.

Discussion. The results show that ID estimation exhibits strong diminishing returns: additional samples reduce estimation error at a slow rate, while the runtime grows super-linearly. CelebA is particularly demanding, with halving the ID estimation error requiring more than an order of magnitude increase in time. This suggests that, in practice, selecting a moderate fraction of the dataset (20–30 % for MNIST or CIFAR-10, 15 % for CelebA) is a reasonable compromise between estimation stability and computational efficiency.

Data availability

Benchmarks are already available.

References

[1] Z. Ahmad, Z.U.A. Jaffri, M. Chen, S. Bao, Understanding gans: fundamentals, variants, training challenges, applications, and open problems, Multimed. Tools Appl. 84 (2025) 10347–10423.

[2] N. Atienza, R. Gonzalez-Diaz, M. Rucco, Persistent entropy for separating topological features from noise in vietoris-rips complexes, J. Intell. Inf. Syst. 52 (2019) 637–655.

[3] U. Bauer, Ripser: efficient computation of vietoris–rips persistence barcodes, J. Appl. Comput. Topol. 5 (2021) 391–423.

[4] Y. Bengio, et al, Learning deep architectures for ai, Found. Trends Mach. Learn. 2 (2009) 1–127.

[5] R. Bennett, The intrinsic dimensionality of signal collections, IEEE Trans. Inf. Theory 15 (1969) 517–525.

[6] J.J. Berwald, J.M. Gottlieb, E. Munch, Computing wasserstein distance for persistence diagrams on a quantum computer, arXiv preprint arXiv:1809.06433, 2018.

[7] M. Bianchini, F. Scarselli, On the complexity of neural network classifiers: a comparison between shallow and deep architectures, IEEE Trans. Neural Netw. Learn. Syst. 25 (2014) 1553–1565.

[8] M.S. Bucarelli, G.A. D'Inverno, M. Bianchini, F. Scarselli, F. Silvestri, A topological description of loss surfaces based on betti numbers, Neural Netw. (2024) 106465.

[9] N.J. Cavanna, M. Jahanseir, D.R. Sheehy, A geometric perspective on sparse filtrations, in: Canadian Conference on Computational Geometry, 2015, pp. 116–121.

[10] A. Chaudhuri, A. Simmons, M. Abdelrazek, Quantifying manifolds: do the manifolds learned by generative adversarial networks converge to the real data manifold?, in: Australasian Joint Conference on Artificial Intelligence, Springer, 2024, pp. 202–213.

[11] F. Chazal, B. Michel, An introduction to topological data analysis: fundamental and practical aspects for data scientists, Front. Artif. Intell. 4 (2021) 667963.

[12] H. Chintakunta, T. Gentimis, R. Gonzalez-Diaz, M.J. Jimenez, H. Krim, An entropy-based persistence barcode, Pattern Recognit. 48 (2015) 391–401.

[13] L. Deng, The mnist database of handwritten digit images for machine learning research, IEEE Signal Process. Mag. 29 (2012) 141–142.

[14] Edelsbrunner, Edelsbrunner, Edelsbrunner, Topological persistence and simplification, Discret. Comput. Geom. 28 (2002) 511–533.

[15] H. Edelsbrunner, J. Harer, et al, Persistent homology-a survey, Contemp. Math. 453 (2008) 257–282.

[16] H. Edelsbrunner, J.L. Harer, Computational Topology: An Introduction, American Mathematical Society, 2022.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, Adv. Neural Inf. Process. Syst. 27 (2014).

[18] W.H. Guss, R. Salakhutdinov, On characterizing the capacity of neural networks using algebraic topology, arXiv preprint arXiv:1802.04443, 2018.

[19] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, Adv. Neural Inf. Process. Syst. 30 (2017).

[20] G.E. Hinton, R. Zemel, Autoencoders, minimum description length and helmholtz free energy, Adv. Neural Inf. Process. Syst. 6 (1993).

[21] D. Horak, S. Yu, G. Salimi-Khorshidi, Topology distance: a topology-based approach for evaluating generative adversarial networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2021, pp. 7721–7728.

[22] H. Hotelling, Analysis of a complex of statistical variables into principal components., J. Educ. Psychol. 24 (1933) 417.

[23] A.J. Izenman, Introduction to manifold learning, Wiley Interdiscip. Rev. Comput. Stat. 4 (2012) 439–446.

[24] I.T. Jolliffe, Principal Component Analysis for Special Types of Data, Springer, 2002.

[25] V. Khrulkov, I. Oseledets, Geometry score: a method for comparing generative adversarial networks, in: International Conference on Machine Learning, PMLR, 2018, pp. 2621–2629.

[26] A. Krizhevsky, V. Nair, G. Hinton, Cifar-10 (Canadian Institute for Advanced Research), http://www.cs.toronto.edu/kriz/cifar.html

[27] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: Proceedings of International Conference on Computer Vision (ICCV), 2015.

[28] G. Loaiza-Ganem, B.L. Ross, R. Hosseinzadeh, A.L. Caterini, J.C. Cresswell, Deep generative models through the lens of the manifold hypothesis: a survey and new connections, arXiv:2404.02954, 2024.

[29] G. Magai, Deep neural networks architectures from the perspective of manifold learning, in: 2023 IEEE 6th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), IEEE, 2023, pp. 1021–1031.

[30] M. Meilă, H. Zhang, Manifold learning: what, how, and why, Annu. Rev. Stat. Its Appl. 11 (2024).

[31] A. Myers, E. Munch, F. Khasawneh, Persistent homology of complex networks for dynamic state detection, Phys. Rev. E 100 (2019) 022314, https://doi.org/10.1103/PhysRevE.100.022314

[32] G. Naitzat, A. Zhitnikov, L.H. Lim, Topology of deep neural networks, J. Mach. Learn. Res. 21 (2020) 1–40.

[33] P. Niyogi, S. Smale, S. Weinberger, Finding the homology of submanifolds with high confidence from random samples, Discrete Comput. Geom. 39 (2008) 419–441.

[34] T. Papamarkou, T. Birdal, M. Bronstein, G. Carlsson, J. Curry, Y. Gao, M. Hajij, R. Kwitt, P. Liò, P. Di Lorenzo, et al, Position paper: challenges and opportunities in topological deep learning.

[35] K. Pearson, Liii. On lines and planes of closest fit to systems of points in space, Lond. Edinb. Dubl. Philos. Mag. J. Sci. 2 (1901) 559–572.

[36] P. Pope, C. Zhu, A. Abdelkader, M. Goldblum, T. Goldstein, The intrinsic dimension of images and its impact on learning, in: International Conference on Learning Representations, 2021, https://openreview.net/forum?id=XJk19XzGq2J

[37] S. Suresh, V. Abrol, Evaluating generative models via cubical homology based persistent entropy, in: Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2, 2025, pp. 2768–2777.

[38] C. Tralie, N. Saul, R. Bar-On, Ripser.Py: a lean persistent homology library for python, J. Open Source Softw. 3 (2018) 925, https://doi.org/10.21105/joss.00925

[39] W.H. Tung, D. Lee, H. Yan, Towards scalable topological regularizers, in: The Thirteenth International Conference on Learning Representations, 2025, https://openreview.net/forum?id=FjZcwQJX8D

[40] J.W. Vick, Homology Theory: An Introduction to Algebraic Topology, vol. 145, Springer Science & Business Media, 2012.

[41] J. Wang, Z. Zhang, H. Zha, Adaptive manifold learning, Adv. Neural Inf. Process. Syst. 17 (2004).

Author biography

Barbara Toniella Corradini is a Postdoctoral Researcher at the University of Siena, Italy. She received her M.Sc. degree in Computer and Automation Engineering at the University of Siena and her Ph.D. in Smart Computing from the University of Florence. Her research interests include properties and applications of generative models. She has also served as a reviewer for international journals in the field of artificial intelligence and machine learning.

Ben W. G. Cullen is pursuing a PhD in Computer Science at the University of Pisa. His research interests include geometric and algebraic methods for investigating the mathematical foundations of deep learning. He holds an integrated Master's degree in Theoretical Physics (MPhys) from the University of Edinburgh and a Master of Science (M.Sc) in Applied Mathematics, awarded cum laude, from the University of Siena, Italy.

Caterina Gallegati is a Ph.D. student in Information and Engineering Science at the University of Siena. She received a Bachelor's Degree in Mathematics and Applications in 2021, and a Master's Degree cum Laude in Applied Mathematics in 2023. Her research focuses on the mathematical foundations of Deep Learning, with particular interest in generative models and the problem of learning data manifolds.

Sara Marziali is a Ph.D. student in Information and Engineering Science at the University of Siena. She received a Bachelor's Degree in Mathematics in 2019, and a Master's Degree cum Laude in Applied Mathematics in 2022. Her research focuses Applied Algebraic Geometry and Discrete Geometry, Machine Learning and Bioinformatics.

Giuseppe Alessio D'Inverno received his M.Sc in Applied Mathematics and the Ph.D. degree in Information Engineering and Science from University of Siena. He is currently a Postdoctoral Researcher in Numerical Analysis in the MathLab group at the International School for Advanced Studies (SISSA) in Trieste. His research interests span from learning theory, with a focus on approximation, to surrogate modeling by means of deep learning.

Monica Bianchini is Associate Professor at the Information Engineering and Mathematics Department of the University of Siena. Her research interests are in the field of machine learning, particularly on neural networks for graph data and deep learning, approximation theory, bioinformatics and computer vision. She has authored more than 140 papers, and edited books and special issues on international journals. Prof. Bianchini is an Associate Editor for several journals and also a permanent member of the Editorial Board of some top international conferences on machine/deep learning.

Franco Scarselli received the Laurea degree with honors in Computer Science from the University of Pisa in 1989 and the Ph.D. degree in Computer Science and Automation Engineering from the University of Florence in 1994. After the Ph.D. he has been supported by foundations of private and public companies and by a postdoc of the University of Florence. In 1999, he moved to the University of Siena, where he was initially a research associate and, currently, an ordinary professor at the department of Information Engineering and Mathematics.