

# Deep Generative Models: Complexity, Dimensionality, and Approximation

**Kevin Wang**

*Department of Biostatistics*

*University of North Carolina at Chapel Hill*

KCWANG@UNC.EDU

**Hongqian Niu**

*Department of Biostatistics*

*University of North Carolina at Chapel Hill*

HONG.NIU@UNC.EDU

**Yixin Wang**

*Department of Statistics*

*University of Michigan*

YIXINW@UMICH.EDU

**Didong Li\***

*Department of Biostatistics*

*University of North Carolina at Chapel Hill*

DIDONGLI@UNC.EDU

**Editor:** Arindam Banerjee

## Abstract

Generative networks have shown remarkable success in learning complex data distributions, particularly in generating high-dimensional data from lower-dimensional inputs. While this capability is well-documented empirically, its theoretical underpinning remains unclear. One common theoretical explanation appeals to the widely accepted manifold hypothesis, which suggests that many real-world datasets, such as images and signals, often possess intrinsic low-dimensional geometric structures. Under this manifold hypothesis, it is widely believed that to approximate a distribution on a  $d$ -dimensional Riemannian manifold, the latent dimension needs to be at least  $d$  or  $d + 1$ . In this work, we show that this requirement on the latent dimension is not necessary by demonstrating that generative networks can approximate distributions on  $d$ -dimensional Riemannian manifolds from inputs of any arbitrary dimension, even lower than  $d$ , taking inspiration from the concept of space-filling curves. This approach, in turn, leads to a super-exponential complexity bound of the deep neural networks through expanded neurons. Our findings thus challenge the conventional belief on the relationship between input dimensionality and the ability of generative networks to model data distributions. This novel insight not only corroborates the practical effectiveness of generative networks in handling complex data structures, but also underscores a critical trade-off between approximation error, dimensionality, and model complexity.

**Keywords:** Approximation theory, generative adversarial networks, manifold hypothesis, space-filling curve.

---

\*. To whom correspondence should be addressed

## 1. Introduction

Generative models, such as generative adversarial networks (GANs, Goodfellow et al., 2014) and variational auto-encoders (VAEs, Kingma and Welling, 2013), have become a central topic in machine learning. These models are versatile, addressing various problems ranging from image generation (Bao et al., 2017; Han et al., 2018), style transfer (Karras et al., 2019), anomaly detection (Xia et al., 2022), to data augmentation (Tran et al., 2021), achieving increasingly accurate results across disciplines. Despite these advances, GANs, in particular, face challenges such as mode collapse, vanishing gradients, and training instability, particularly when distributions are not continuous or have disjoint supports (Arjovsky and Bottou, 2017).

While traditional GANs and VAEs primarily relied on the Kullback–Leibler divergence (KL) to measure distances between input and output distributions, more recent approaches have greatly improved training performance. The shift from KL to Wasserstein distance (Villani et al., 2009) is a notable example, as exemplified in the Wasserstein GAN (WGAN, Arjovsky et al., 2017), and further extended to Wasserstein VAE (WVAE, Ambrogioni et al., 2018) and Wasserstein Auto Encoders (Tolstikhin et al., 2017). Beyond this, there have been substantial enhancements in model architecture, loss functions, and regularization techniques. Advances such as conditional GANs (cGAN, Mirza and Osindero, 2014) for targeted image generation, auxiliary classifier GANs (AC-GANs, Odena et al., 2017), and self-attention GANs (SA-GANs, Zhang et al., 2019) have enriched the versatility and effectiveness of GANs. Similarly, VAEs have seen improvements with techniques such as hierarchical latent variables (Vahdat and Kautz, 2020; Sønderby et al., 2016) and incorporation of normalizing flows (Kingma et al., 2016), refining their ability to model complex distributions. These developments collectively contribute to the generation of high-quality, high-dimensional data from simpler, lower-dimensional distributions, with notable impact in fields such as photorealistic image generation (Wang et al., 2018; Sarkar et al., 2021).

The theoretical advancements in deep generative models, paralleling their practical applications, have been substantial. In particular, the depth and width of network architectures in these models have been theoretically shown to significantly impact their approximation capabilities, as detailed in the works focused on neural network expressiveness (Eldan and Shamir, 2016). Furthermore, the challenges of mode collapse in GANs and the propensity of VAEs to produce overly smooth outputs have catalyzed research into improving their architectural designs and training methodologies (Goodfellow et al., 2014; Kingma and Welling, 2013). Collectively, these theoretical advancements have not only deepened our understanding of the mechanisms behind deep generative models but also guided the development of more refined and capable generative architectures.

However, the theoretical investigation of the manifold hypothesis in deep generative models, particularly its implications in approximation theory, remains less developed. The manifold hypothesis, widely accepted as a crucial reason behind the success of these models, posits that real-world high-dimensional data often reside on lower-dimensional manifolds (Bronstein et al., 2017). This concept is critical because it suggests a fundamental reason why models such as GANs and VAEs, which operate in low-dimensional spaces, are able to capture complex data distributions effectively. Although empirical evidence sup-

ports this hypothesis, a deeper theoretical understanding of how these models approximate and learn distributions on lower-dimensional manifolds within high-dimensional spaces is crucial. Further theoretical work in this area is essential, not only to validate the manifold hypothesis but also to enhance the design and efficiency of generative models in handling complex, high-dimensional data.

Building on the previous discussion of the manifold hypothesis, Dahal et al. (2022) offers a noteworthy theoretical contribution in the field of deep generative models. Their work effectively addresses fundamental questions in approximation theory, demonstrating that distributions on a  $d$ -dimensional Riemannian manifold can be approximated by a deep generative network through the pushforward measure of an easily sampled distribution in  $d+1$  dimensions, such as a uniform distribution in a cube  $[0, 1]^{d+1}$ . Importantly, they provide an upper bound on the complexity of these models, revealing that the required number of layers and neurons is determined by both the approximation error and the manifold dimension  $d$ . This insight offers a direct theoretical justification for the models’ abilities to learn diverse probability distributions and suggesting avenues for efficiency improvements in practical applications.

However, the research by Dahal et al. (2022) also brings to light some important, yet unanswered questions. Primarily, the necessity of using a  $d + 1$  dimensional input space to approximate a  $d$  dimensional manifold warrants further investigation. This approach, integral for assembling local patches, raises the possibility of achieving similar outcomes with an input dimension that matches  $d$ . Furthermore, in real-world applications where  $d$  is not readily known, the choice of input dimension becomes a critical decision. If the chosen dimensionality is either greater or lesser than the actual  $d$ , how would it impact the model’s performance and efficiency? Answering these questions is crucial for advancing the development of deep generative models, potentially leading to more adaptable and efficient solutions for complex data representation.

In this work, we aim to understand how the input dimension relates to the manifold’s intrinsic dimension in deep generative models. We adapt the space-filling curve theory to demonstrate a novel aspect of deep generative models: their ability to learn a target data distribution from easy-to-sample distributions of arbitrary dimensions. This includes the capacity to approximate these distributions even from an input as low as a one-dimensional uniform distribution on the unit interval  $[0, 1]$  (see Figure 1 for examples of curves filling 2-dimensional manifolds). Intuitively, if the input dimension is smaller than the true dimension, the network can learn to “fill out” the true manifold with an increasingly complex series of structures. However, in order to accurately approximate the higher dimensional structure, the approximating manifold must fold onto itself in increasingly irregular ways to fill the space.

In addition, our work establishes the critical interplay among the input dimension, the true dimension of the manifold, and the approximation error. In particular, our findings highlight a “trade-off triangle” in deep generative models, where it is unattainable to achieve low (underestimated) dimensionality, low approximation error, and low network complexity simultaneously, where network complexity is defined to be network width (number of neurons) in this paper. Specifically, when the input dimension is underestimated and the approximation error is low, we encounter one corner of this triangle: a significant increase in model complexity. This complexity, evident in the escalated number of neurons, grows

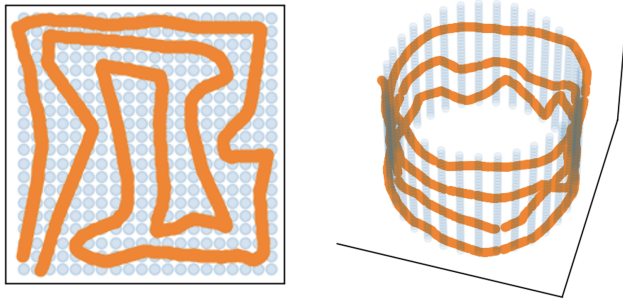


Figure 1: Two cases demonstrating the idea of how sufficiently large neural networks can learn distributions of higher dimension than their input sampling distributions by filling out the space. Depicted here are 1-dimensional partial space-filling curves beginning to “fill out” a 2-dimensional unit square (left) and a 2-dimensional cylindrical surface (right). The blue points represent the training sample from the target distribution used to fit the neural network, while the orange points represent new points generated by the trained network.

super-exponentially with approximation error, underscoring the complex trade-offs and interdependencies inherent in accurately gauging the intrinsic dimension of the data.

This triangle metaphorically highlights the necessity of balancing these aspects when designing generative models, a central theme of our contributions. Our contribution also includes empirical studies to support these findings, which are overlooked in existing literature (Dahal et al., 2022).

Notably, our proof techniques for the underestimated dimension case are novel, requiring in-depth exploration of mathematical literature in Monge-Ampère equations from the 1990s (Caffarelli, 1990a,b, 1991, 1992a,b, 1996), beyond the scope of existing literature (Dahal et al., 2022; Villani et al., 2009).

The structure of our paper is organized as follows: Section 2 lays the groundwork with preliminaries. Section 3 discusses our main theoretical contributions. In Section 4, we present simulations on toy cases for illustrative and visualization purposes. Section 5 offers a sketch of our proof, focusing primarily on the workflow, and is followed by a Discussion Section. Complete proofs and additional experimental details are provided in the Appendix.

## 2. Preliminaries

This section covers the essential concepts underpinning our research, including generative models, optimal transport, Riemannian manifolds, and existing approximation theory for generative models on Riemannian manifolds. This groundwork is crucial for appreciating the theoretical advances we present in later sections and situates our research within the broader context of generative modeling.

### 2.1 Generative Models

Generative models aim to learn the distribution of a dataset through sampling procedures. We consider our data following a distribution  $Q$  residing in an ambient space  $\mathbb{R}^D$  and an input distribution  $\rho$  over a simpler space  $\mathcal{S}$ , such as the uniform distribution on the



hypercube  $[0, 1]^d$ . The goal is to construct a generator  $g : \mathcal{S} \rightarrow \mathbb{R}^D$  that minimizes a discrepancy function between the generated (pushforward) distribution  $g_{\#}(\rho)$  and the target distribution  $Q$ , expressed as

$$\min_{g \in \mathcal{G}} \text{Discrepancy}(g_{\#}(\rho), Q).$$

In this context, the function class  $\mathcal{G}$  often comprises deep neural networks. A lower discrepancy signifies a closer match, meaning that the data generated from the simpler input space  $\mathcal{S}$  closely resemble the true data represented by distribution  $Q$ . The essence of generative models is to produce outputs that are “similar” to real data, aligning the generated samples with the actual distribution. A larger class  $\mathcal{G}$  usually yields a smaller discrepancy, and therefore, better performance. However, this enhancement comes at the cost of increased computational resources. As a result, understanding the balance between the size of  $\mathcal{G}$  and the discrepancy is of the utmost importance.

A practical challenge is that  $Q$  is often unknown, but instead, we only observe samples  $x_1, \dots, x_n \sim Q$  that are commonly believed to be independent and identically distributed (iid) following  $Q$ . As a result, we can replace  $Q$  by its empirical distribution  $Q_n := \frac{1}{n} \sum_{i=1}^n \delta_{x_i}$  where  $\delta_x$  is the Dirac measure at  $x$ . The practical objective of generative modeling is to minimize the discrepancy between the learned distribution and this empirical distribution:

$$\min_{g \in \mathcal{G}} \text{Discrepancy}(g_{\#}(\rho), Q_n),$$

where the minimizer is usually denoted by  $\hat{g}_n$ .

## 2.2 Optimal Transport

In the study of generative models, the choice of discrepancy function is of fundamental importance. The Wasserstein distance, a central concept in optimal transport theory (Villani et al., 2009), is commonly employed due to its effectiveness in measuring the ‘cost’ of transforming one distribution into another. This aligns well with the objectives of generative modeling.

**Definition 1** *The Wasserstein- $p$  distance between two distributions  $Q$  and  $\nu$  in domain  $\mathcal{M}$ , is defined as*

$$W_p(Q, \nu) = \left( \inf_{\gamma \in \Gamma(Q, \nu)} \mathbb{E}_{(x, y) \sim \gamma} (c(x, y)^p) \right)^{\frac{1}{p}},$$

where  $\Gamma(Q, \nu)$  is the set of all couplings of  $Q$  and  $\nu$ , containing all joint distributions over  $\mathcal{M} \times \mathcal{M}$  with marginals  $Q$  and  $\nu$ , and  $c : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}_{\geq 0}$  is called the cost function.

We focus on the Wasserstein-1 distance with  $c(x, y) = \|x - y\|$ , also known as the earth mover’s distance. This specific case is especially relevant for our work in generative models, offering a robust framework for evaluating the similarity between generated and target distributions. Wasserstein-1 distance additionally admits a dual form that makes it particularly suitable for computation, which is a property that Wasserstein- $p$  in general lacks. However, our theory on convergence is extendable to Wasserstein- $p$  distance in general, which we investigate in Corollary 9.

### 2.3 Space-filling Curves

In this paper, for a set  $S \subset \mathbb{R}^D$ , we define a space-filling curve to be a continuous curve  $\mathcal{M}_\epsilon \subset \mathbb{R}^D$  such that every point in  $S$  is within  $\epsilon$  of some point in  $\mathcal{M}_\epsilon$ . Consequently, we can define a space-filling manifold to be a manifold in  $\mathbb{R}^D$  satisfying the same properties:

**Definition 2** *A manifold  $\mathcal{M}_\epsilon \subset \mathbb{R}^D$  is said to be  $\epsilon$ -space-filling manifold of  $S \subset \mathbb{R}^D$  if*

$$d(S, \mathcal{M}_\epsilon) = \sup_{x \in S} \inf_{y \in \mathcal{M}_\epsilon} \|x - y\| < \epsilon.$$

*$\mathcal{M}_\epsilon$  is called a space-filling curve if it's a one-dimensional manifold.*

When  $S$  itself is a manifold, the existence of such  $\epsilon$ -space-filling manifolds is shown in Lemma 10. We note that this definition of the space-filling curve/manifold differs from the standard definition in topology that demands the curve be surjective on the larger space  $S$ . These classical space-filling curves are typically taken as the limit of certain classes of curves:

$$\mathcal{M}_{\text{space-filling}} = \lim_{\epsilon_i \rightarrow 0} \mathcal{M}_{\epsilon_i},$$

where the sequence  $\mathcal{M}_{\epsilon_i}$  is carefully chosen to preserve desired properties (such as non self-intersection) in the limit. However, for our purposes, we do not need the limit, only particular choices of  $\mathcal{M}_\epsilon$  satisfying  $d(S, \mathcal{M}_\epsilon) < \epsilon$ , and define space-filling manifolds thusly. We use the term “space-filling” for intuition purposes, and do not demand a completely space-filling property from our manifolds.

### 2.4 Riemannian Manifold

Although data often reside in high-dimensional space  $\mathbb{R}^D$ , there is substantial evidence suggesting that they lie on some low-dimensional manifolds (Bronstein et al., 2017). This concept underlies many manifold-based generative models, such as VAEs and GANs.

In the same manner as most existing work in the literature, we assume that the data are distributed on a  $d$ -dimensional orientable compact Riemannian manifold  $\mathcal{M}$ , isometrically embedded in the ambient space  $\mathbb{R}^D$ , with Riemannian metric  $g$ . A manifold is a locally Euclidean space, with each local neighbor, known as a local chart, diffeomorphic to Euclidean space  $\mathbb{R}^d$  (Boothby, 1986). The Riemannian metric  $g$  defines a smoothly varying inner product (metric) in each tangent space  $T_x \mathcal{M}$ . The geodesic distance  $d_{\mathcal{M}}(x, y)$  is defined as the length of the shortest path connecting  $x, y \in \mathcal{M}$ . In addition, there exists a well-defined  $d$ -form, known as the Riemannian volume form  $d\text{Vol}_{\mathcal{M}}$ , which often serves as the analog of the Lebesgue measure on Riemannian manifold. This allows us to define density functions of probability distributions with respect to the volume measure. For more details, see Do Carmo and Flaherty Francis (1992).

### 2.5 Existing Work

In previous research on generative models, there has been a focus on low-dimensional data structures, assuming that high-dimensional data are parametrized by low-dimensional latent parameters. This approach treats manifolds as globally homeomorphic to Euclidean space,

implying a single-chart manifold model (Luise et al., 2020; Schreuder et al., 2021; Block et al., 2022; Chae et al., 2023). However, this assumption presents limitations in dealing with the complexity inherent in general manifolds with multiple charts.

In contrast, Yang et al. (2022) and Huang et al. (2022) demonstrate that GANs can approximate any data distribution from a one-dimensional continuous distribution. This method does not assume a global chart; however, it heavily relies on GANs memorizing empirical data distributions, posing limitations in generating novel samples.

To overcome the limitations of the above approaches, Dahal et al. (2022) represents a significant development in this context. By avoiding the single-chart assumption and the need for memorizing data, they construct an oracle transport map suitable for general manifolds with multiple charts. This advancement in approximating distributions on Riemannian manifolds with neural network pushforwards paves the way for more sophisticated and effective generative models. Their study relies on two mild assumptions:

**Assumption 3**  $\mathcal{M}$  is a  $d$ -dimensional compact Riemannian manifold isometrically embedded in  $\mathbb{R}^D$ . As a consequence, there exists  $B > 0$  such that  $\|x\|_\infty \leq B, \forall x \in \mathcal{M}$ .

**Assumption 4**  $Q$  is supported on  $\mathcal{M}$  and has a density function  $q$  with respect to the volume measure  $\text{Vol}_{\mathcal{M}}$  with a positive lower bound  $c$ :  $0 < c \leq q(x), \forall x \in \mathcal{M}$ .

We consider the ReLU-type neural networks

$$g(x) = W_L \sigma(W_{L-1} \cdots \sigma(W_1 x + b_1) + \cdots + b_{L-1}) + b_L,$$

where  $\sigma$  is the ReLU activation function (Fukushima, 1969),  $W$  is the weight matrix,  $b$  is the bias vector. The function class, denoted by  $\mathcal{G}_{NN}(m, L, p, \kappa)$ , contains neural networks with ReLU activation function, input dimension  $m$ , maximum depth  $L$ , maximum width  $p$ , and bounded weights and biases by  $\kappa$ :

$$\mathcal{G}_{NN}(m, L, p, \kappa) := \{g = (g_1, \dots, g_D) : \mathbb{R}^m \rightarrow \mathbb{R}^D : \\ g_j \text{ is in form (2.5) with at most } L \text{ layers and max width } p, \|W_i\|_\infty \leq \kappa, \|b_i\|_\infty \leq \kappa\}.$$

Under the above assumptions, the two main theorems in Dahal et al. (2022) are as follows:

Firstly, Lemma 5 establishes that certain distributions on compact Riemannian manifolds can be approximated using a deep neural network and a simple input distribution. Crucially, the dimension of the input space is precisely one more than the manifold dimension. Furthermore, Lemma 5 provides detailed complexity bounds on the size of the network to achieve a specified level of accuracy measured by the Wasserstein loss.

**Lemma 5 (Theorem 1 of Dahal et al. 2022)** *Let  $\rho = \text{Unif}(0, 1)^{d+1}$ , then there exists a constant  $0 < \alpha < 1$  that is independent of  $D$  such that for any  $0 < \epsilon < 1$ , there exists a deep neural network  $g \in \mathcal{G}_{NN}(d+1, L, p, \kappa)$  with  $L = O(\log(\frac{1}{\epsilon}))$ ,  $p = O(D\epsilon^{-\frac{d}{\alpha}})$ ,  $\kappa = B$ , that satisfies  $W_1(g_\#(\rho), Q) < \epsilon$ .*

Next, Lemma 6 serves as the empirical counterpart of Lemma 5, addressing scenarios in which only finite samples following the distribution  $Q$  are observed. It demonstrates the

theoretical possibility of identifying such generators using finite samples instead of direct access to the true distribution  $Q$ . In addition, this lemma links the network size to the sample size rather than the approximation error.

**Lemma 6 (Theorem 2 of Dahal et al. 2022)** *Under the same assumption as in Lemma 5, let  $x_1, \dots, x_n \stackrel{iid}{\sim} Q$  be  $n$  iid samples from  $Q$ , then for any  $\delta > 0$ , set  $\epsilon = n^{-\frac{1}{d+\delta}}$  in Lemma 5 so that the network class  $\mathcal{G}_{NN}(d+1, L, p, \kappa)$  has parameters  $L = O\left(\log\left(n^{\frac{1}{d+\delta}}\right)\right)$ ,  $p = O\left(Dn^{\frac{d}{\alpha(d+\delta)}}\right)$ ,  $\kappa = B$ . Then the empirical risk minimizer  $\hat{g}_n$  has rate  $\mathbb{E}[W_1(\hat{g}_n(\rho), Q)] \leq Cn^{-\frac{1}{d+\delta}}$ , where  $C$  is a constant independent of  $n$  and  $D$ .*

These results shed light on how distributions on the manifold can be approximated by a deep neural network's pushforward of a low-dimensional easy-to-sample distribution.

However, this approach has several limitations. First, the intrinsic dimension  $d$  of the manifold  $\mathcal{M}$  is almost never known. Although there is an immense literature on estimating  $d$  (Levina and Bickel, 2004; Zheng et al., 2022; Horvat and Pfister, 2022; Brown et al., 2022), it has been proven to be an almost insurmountable problem due to its complexity (Fefferman et al., 2016). Second, even if  $d$  is known, the choice of input dimension  $d+1$  is unnatural, as it exceeds the true dimension. Technically, the extra dimension results from the need to connect local neighborhoods using a pasting algorithm. But this seems superfluous from an intrinsic perspective, considering partition of unity could potentially serve the same purpose. Third, in practice, the input dimension is often treated as a tuning parameter, or simply set based on historical experience or recommendations without rigorous optimization. It remains unclear how these decisions on input dimension affect approximation performance, especially when  $d$  is underestimated.

To address these issues, our study examines the scenario where the input dimension is arbitrary, potentially as minimal as one, within the same framework. We will summarize our primary findings in the subsequent section.

### 3. Main Theory

We begin with Dahal et al. (2022), which presents a method for approximating distributions on compact Riemannian manifolds using neural networks with an input dimension of  $d+1$ . This raises a natural question: *can the input dimension be reduced to  $d$ , the manifold's intrinsic dimension, or even lower, without compromising the approximation's effectiveness?* In this work, we answer this question affirmatively, leveraging concepts of space-filling curves and insights from a series of work on Monge-Ampère equations by Caffarelli (Caffarelli, 1990a,b, 1991, 1992a,b, 1996). Our theorem demonstrates the feasibility of using any input dimension  $m \geq 1$ , which can be either smaller or larger than  $d$ , even down to 1. However, reducing the input dimension below the manifold's dimension introduces a super-exponential increase in complexity. Finally, we highlight that our approach adheres to the same foundational assumptions as Dahal et al. (2022).

Our theorem, central to this discussion, asserts the approximation power of these neural networks under varying input dimensions. It establishes the conditions under which the networks can effectively approximate distributions on Riemannian manifolds, taking into account the dimensionality of the input and its impact on the network's complexity.

**Theorem 7 (Approximation Power of Deep Generative Models)** *Under Assumptions 3 and 4, and with  $\rho$  as the uniform measure on  $[0, 1]^m$ , for any  $\epsilon > 0$ , there exists a deep neural network  $g \in \mathcal{G}_{NN}(m, L, p, \kappa)$  such that  $L = O\left(\log\left(\frac{1}{\epsilon}\right)\right)$ ,  $p = \begin{cases} O\left(D\epsilon^{\frac{-m}{\alpha(m, \epsilon)}}\right) & m \leq d \\ O\left(D\epsilon^{\frac{-d}{\alpha}}\right) & m > d \end{cases}$ ,  $\kappa = \max\{B, 1\}$ , and  $W_1(g_{\#}(\rho), Q) < \epsilon$ . Furthermore, when  $m \leq d$ ,  $\lim_{\epsilon \rightarrow 0} \alpha(m, \epsilon) = 0$ , leading to a super-exponential increase in the width  $p$ .*

The purpose of this theorem is two-fold. Firstly, it demonstrates that neural networks can approximate distributions on a manifold, with the flexibility to use an input distribution uniformly distributed over a hypercube  $[0, 1]^m$  of any dimension. This means that the choice of  $m$  can be adapted as needed, whether it's smaller or larger than the manifold dimension  $d$ . We additionally note that the choice of our generator to be the uniform distribution on the unit hypercube is non-restrictive. These results can easily be extended to any input distribution with a density that is upper bounded away from infinity and lower bounded from zero on the unit cube, such as a truncated normal distribution or uniform distributions.

Secondly, the theorem provides a quantitative link between the complexity of the network and the relationship between the input dimension  $m$  and the manifold dimension  $d$ . Notably, when  $m > d$ , the rates coincide with those found in Dahal et al. (2022). However, the situation becomes particularly intriguing when the input dimension  $m$  is smaller than the target dimension  $d$ . In this scenario, the width of the neural network no longer increases at a polynomial rate of  $\frac{d}{\alpha}$ , but at a super-exponential rate of  $\frac{m}{\alpha(m, \epsilon)}$  where  $\frac{m}{\alpha(m, \epsilon)} \xrightarrow{\epsilon \rightarrow 0} \infty$ . This phenomenon implies that, while it is theoretically feasible to approximate a broad class of distributions with low-dimensional inputs, choosing an appropriate input dimension is critical to avoid excessively complex networks; it is often more advantageous to overestimate  $d$  than to underestimate it.

We now turn our attention to the empirical aspect of our theory, examining the statistical guarantees provided by our model in the presence of iid samples  $x_1, \dots, x_n \sim Q$ .

**Theorem 8 (Statistical Guarantees of Deep Generative Models)** *Given iid samples  $x_1, \dots, x_n \sim Q$  with empirical distribution  $Q_n$ , let  $\delta > 0$ ,  $L = O\left(\log\left(n^{\frac{1}{d+\delta}}\right)\right)$ ,*

$$p = \begin{cases} O\left(Dn^{\frac{m}{(m+\delta)\alpha(m, n, d, \delta)}}\right) & m \leq d \\ O\left(Dn^{\frac{d}{(d+\delta)\alpha}}\right) & m > d, \end{cases}$$

$\kappa = \max\{B, 1\}$ , then the empirical risk minimizer  $\hat{g}_n \in \mathcal{G}_{NN}(m, L, p, \kappa)$  satisfies

$$\mathbb{E}[W_1(\hat{g}_{n\#}(\rho), Q)] \leq (1 + 2C_\delta)n^{-\frac{1}{d+\delta}},$$

where  $C_\delta$  is a constant, independent of  $n$ . Furthermore, when  $m \leq d$ ,  $\lim_{n \rightarrow \infty} \alpha(m, n, d, \delta) = 0$ , leading to a super-exponential increase in the width  $p$ .

Similar to the population version, this theorem demonstrates that deep neural networks can approximate manifold distributions from  $n$  iid observations, with the approximation error diminishing at a rate of  $\frac{1}{d+\delta}$ . However, the complexity of the network, particularly

the width, increases alongside  $n$ . Although a larger  $n$  yields a smaller approximation error, when  $m \leq d$ , the width  $p$  grows super-exponentially with  $n$ .

These two theorems collectively underscore a “trade-off triangle” of model complexity, input dimension, and approximation accuracy. In the next section, we illustrate these concepts through simulations using  $m = 1, 2, 3$  as case studies for proof of concept and visualization.

Finally, we also show that Theorem 7 and Theorem 8 can be partially extended to Wasserstein- $p$  distances in the following corollary.

**Corollary 9** *Under the same setup as Theorem 7, for any  $\epsilon > 0$ , there exists a deep neural network  $g \in \mathcal{G}_{NN}(m, L, p, \kappa)$  such that  $W_p(g_{\#}(\rho), Q) < \epsilon$ . Moreover, under the same setup as Theorem 8, the empirical risk minimizer  $\hat{g}_n \in \mathcal{G}_{NN}(m, L, p, \kappa)$  satisfies  $\mathbb{E}[W_p(\hat{g}_{n\#}(\rho), Q)] \leq (1 + 2C_{\delta})n^{-\frac{1}{d+\delta}}$ .*

Note that the bounds for  $L, p, \kappa$  are not provided in this case; a more detailed discussion is presented in the proof in Appendix B.6.

## 4. Simulation

In this section, we provide empirical evidence on three toy examples for visualization purposes of these deep generative networks’ abilities to learn space-filling curve approximations. We demonstrate the ability of the standard ReLU network to map the uniform distribution on the unit hypercubes of various dimensions  $[0, 1]^m$  to a variety of target distributions, where  $m$  is the dimension of the input distribution. Note that here we are using a standard feedforward neural network with a Wasserstein-1 loss function approximation provided in the Python Optimal Transport package (POT, Flamary et al., 2021). This loss function takes the role of the “critic” network in a typical GAN model, hence our fitted feedforward network under this Wasserstein-1 loss function is analogous to the “generator” in a Wasserstein GAN (Arjovsky et al., 2017). We also report an empirical “fill distance” which we measure by taking a set of samples  $x_1, \dots, x_n$  from the true manifold, another set of samples  $y_1, \dots, y_N$  from the generated manifold, and calculating  $\frac{1}{n} \sum_{i=1}^n \min_{y_j} \{d(x_i, y_j)\}$ . This approximates the mean distance over all points in the data manifold to the closest point on the approximating curve. Hence, this is a measurement of the distance between the manifold supports of distributions, while the Wasserstein loss (used to train the neural networks) offers a measurement of the distance between distributions.

In Simulation 1, we show that the problem of finding a generator from the two dimensional square to the two dimensional square is computationally feasible (as in the required complexity is relatively small). Additionally, we show that the same problem with only a one dimensional input dimension is relatively harder, and also show the trajectory of training iterations to visualize how the neural network “learns” the distribution with an under-dimensioned input. In Simulation 2, we showcase the mapping of the same uniform distribution on  $[0, 1]$  to the uniform distribution on a 2-dimensional cylinder  $S^2 \subset \mathbb{R}^3$  ( $m = 1, d = 2, D = 3$ ). We also showcase that the same target distribution can be learned with much lower complexity when we match the input dimension to the manifold dimension, that is  $m = 2 = d$ . In Simulation 3, we take our target distribution to be the 3-dimensional

uniform distribution on a unit cube  $[0, 1]^3$  ( $d = D = 3$ ). As in Simulations 1 and 2, we train ReLU Networks using uniform distributions on the 1, 2, and 3 dimensional hypercubes ( $m = 1, 2, 3$ ) and compare performance. Full implementation details and additional simulations including a study of network complexity and approximation error, the use of higher input dimension than target manifold dimension ( $m = d + 1$ ) for each of the three simulation cases, and the case of non-uniform input or target distributions, along with a link to a GitHub repository containing all code, can be found in Appendix C.

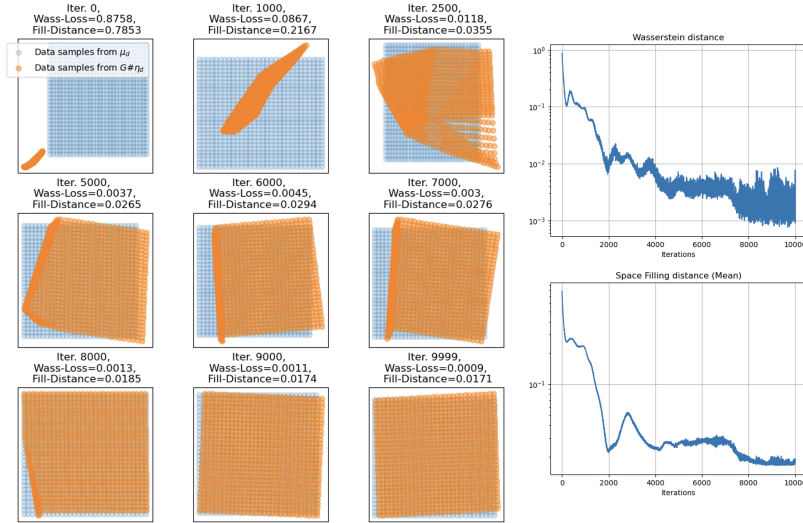


Figure 2: Simulation 1: Training results (left) of a small 2 hidden layer, 10 node each, fully connected network mapping the 2-D uniform input to a 2-D uniform target data manifold. The orange surface is generated by the neural network and is “filling in” the data manifold (blue). Wasserstein loss (top right) and fill distance (bottom right) between generated and observed data per training iteration for a total of 10,000 iterations.

#### 4.1 Simulation 1

In Simulation 1, we let the data distribution  $Q$  be a uniform distribution over the unit square  $[0, 1]^2$  and consider the cases where the input distribution  $\rho$  to the neural network is a uniform distribution over the unit square  $[0, 1]^2 \in \mathbb{R}^2$  and a uniform distribution over the unit interval  $[0, 1] \in \mathbb{R}$ . In this case, the manifold dimension  $d = 2$  is equal to the dimension of the ambient space  $D = 2$ . We first consider the case where the input and target dimensions are equal.

Aligned with our theory, the required complexity in this case is expected to be significantly lower than when the target data dimension is underestimated. In Figure 2, we show that training a fully connected feed forward network with just 2 hidden layers with 10 nodes each for 10,000 iterations under Wasserstein loss, again from the Python Optimal Transport (OT) package, achieves low Wasserstein loss as well as low average empirical fill distance. Note that in this case, during training the input is taken to be a random sample on the unit square at each iteration, while the target data is a uniform grid on the unit square, so that the input sample is not identical to the target data at any iteration.

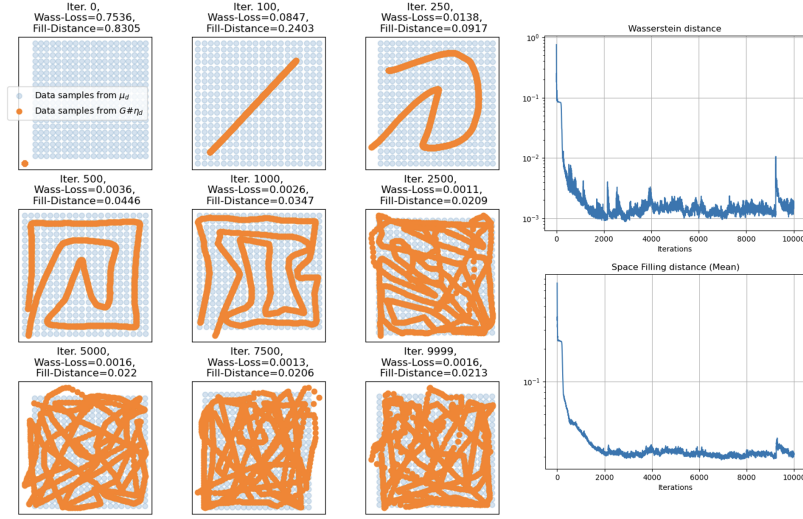


Figure 3: Simulation 1: Training results of a 5 hidden layer, 200 node each, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a uniform distribution on the interval  $[0, 1]$  to a uniform distribution on the unit square  $[0, 1]^2$ . It can be seen that as the loss decreases, the fitted curve fills more of the square as expected.

Next we re-examine the target distribution from the case that the input dimension  $\rho$  is a uniform distribution over the unit interval  $[0, 1]$ . Figure 3 demonstrates how a standard feedforward, fully-connected neural network, using ReLU activation and trained with Wasserstein loss, effectively maps the unit interval into an increasingly complex, space-filling curve in a two-dimensional space. Note that now a much larger network size of 5 hidden layers with 200 nodes each is required to reach a similar level of approximation to the equal input and target dimensions case. For further simulations regarding the effect of different network sizes see Appendix C.1, and for the  $m = d + 1$  case see Appendix C.2.

## 4.2 Simulation 2

Analogous to Simulation 1, in Simulation 2 we consider a situation where the data distribution  $Q$  is a uniform distribution over a cylinder embedded in  $\mathbb{R}^3$  and again consider the cases where the input distribution  $\rho$  to the neural network is a uniform distribution over the unit square  $[0, 1]^2 \in \mathbb{R}^2$  and the unit interval  $[0, 1] \in \mathbb{R}$ .

We first consider when the input distribution (uniform) dimension matches the target data manifold dimension (also uniform), where  $m = d = 2$ . We show that, aligned with our theory, the required complexity in this case is observed to be much lower than when the target data dimension is underestimated. In Figure 4, we show the results of training a fully connected feed forward network with a total of 3 hidden layers with 25 nodes each for 8,000 iterations under Wasserstein loss and that the network is able to achieve low Wasserstein loss as well as low average fill distance.

In contrast, we consider the same uniform distribution on a cylinder embedded in  $\mathbb{R}^3$  but with an input distribution of only one dimension. Note again that in this example we



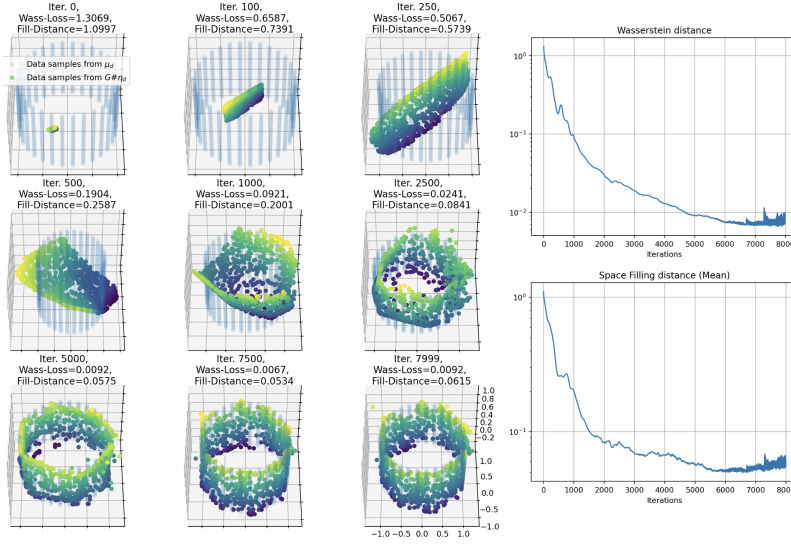


Figure 4: Simulation 2: Training results of a small 3 hidden layer, 25 node each, fully connected network for 8,000 iterations mapping a 2-D uniform input to a 2-D uniform target data manifold on a cylinder embedded in  $\mathbb{R}^3$ . Here the multicolored surface is generated by the neural network (colored by  $z$ -axis height) and is “filling in” the data manifold (blue).

have the manifold  $\mathcal{M}$  with dimension  $d = 2$ , embedded in an ambient Euclidean space with dimension  $D = 3$ . We show again that a feedforward, fully-connected neural network under Wasserstein loss and ReLU nonlinearity is still able to learn space-filling curves properly on the manifold of lower dimension than the ambient space (albeit with a much larger network size of 7 hidden layers with 250 nodes each). Again, the neural network maps the input interval into an increasingly complex space-filling curve on the cylinder as the number of training iterations increases and the loss decreases respectively as seen in Figure 5. For full implementation details and code, as well as the  $m = d + 1$  case, see Appendix C.

### 4.3 Simulation 3

In Simulation 3, we consider a uniform distribution on a unit cube  $[0, 1]^3$  in  $\mathbb{R}^3$ , while now taking as input one, two, and three-dimensional uniform distributions on  $[0, 1]^m$  for  $m = 1, 2, 3$  respectively. We show again that a feedforward, fully-connected neural network under Wasserstein loss and ReLU nonlinearity is still able to learn “space-filling sheets” on the manifold of higher dimension than the input.

We again start with the case where the input dimension matches the target dimension ( $m = 3$ ). In Figure 6, we show the results of training a neural network with 3 hidden layers of 128 neurons each. It can be seen that the network achieves both low Wasserstein loss as well as empirical fill distance in Figure 6. Note that in this case the input data is a randomly re-sampled each training iteration, while the target remains a fixed grid on the unit cube, such that the input and target data are not identical at any iteration.

Now we reduce the input dimension by 1, mapping a uniform distribution of dimension  $m = 2$  to a uniform distribution in a higher target dimension  $d = 3$ . We observe in Figure 7

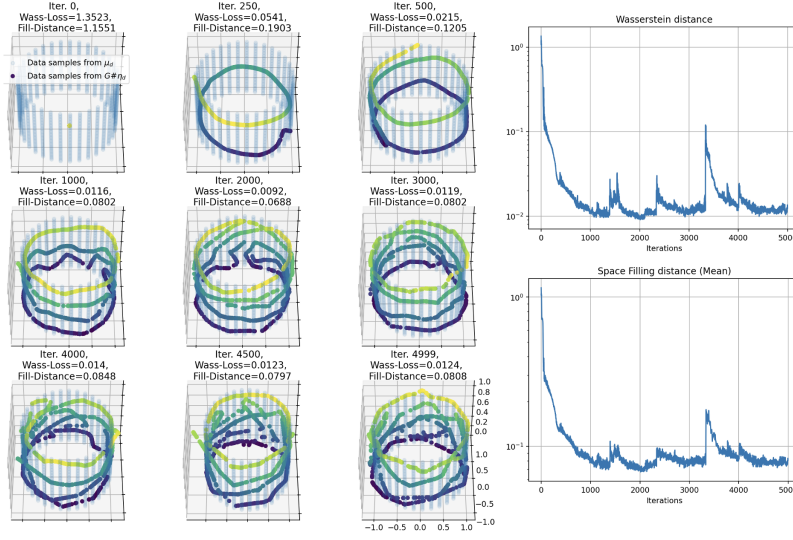


Figure 5: Simulation 2: Training results of a 7 hidden layer, 250 node each, fully connected network over 5,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a 1-D uniform distribution on the interval  $[0, 1]$  to a 2-D uniform distribution on a cylinder embedded in  $\mathbb{R}^3$ . As the loss decreases over the iterations, the fitted curve fills more of the surface.

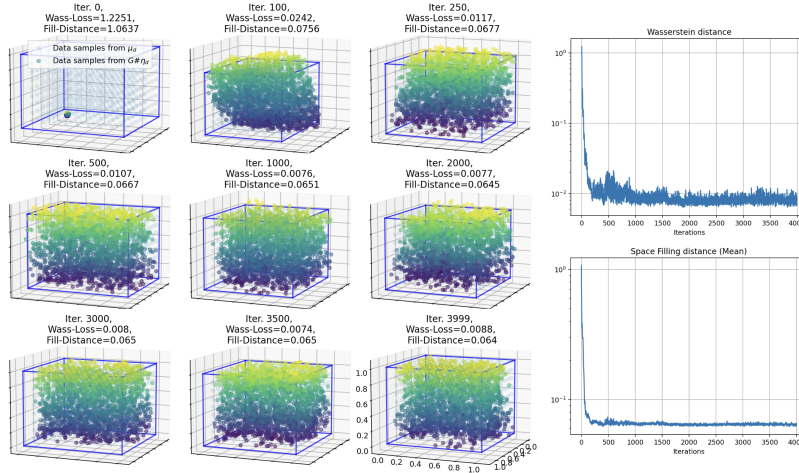


Figure 6: Simulation 3: Training results of a 3 hidden layer, 128 node each, fully connected network over 4,000 iterations mapping the 3-D uniform input to a 3-D uniform target data manifold on the unit cube. Here the multicolored curve is generated by the neural network (colored by  $z$ -axis height) and is “filling in” the data manifold (blue).

that a larger neural network with 4 hidden layers of 256 nodes each achieves similar loss to the smaller model used when  $m = d = 3$ .

Finally, we reduce the input dimension one more time for the case when the input dimension is  $m = 1$  and the data manifold is uniform in  $d = 3$  dimensions. We observe in Figure 8

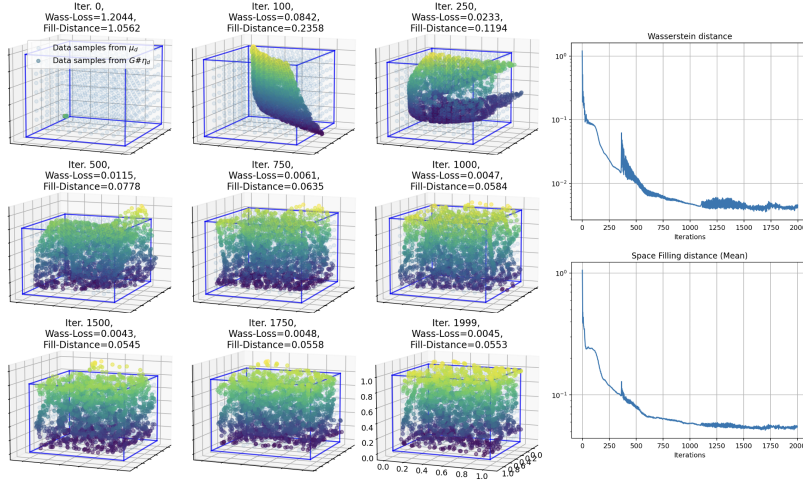


Figure 7: Simulation 3: Training results of a 4 hidden layer, 256 node each, fully connected network over 2,000 iterations mapping a uniform distribution on the unit square  $[0,1]^2$  to a uniform distribution on  $[0,1]^3$ , colored by height ( $z$ -axis). As the loss decreases over the iterations, the fitted surface fills more of the space in the cube.

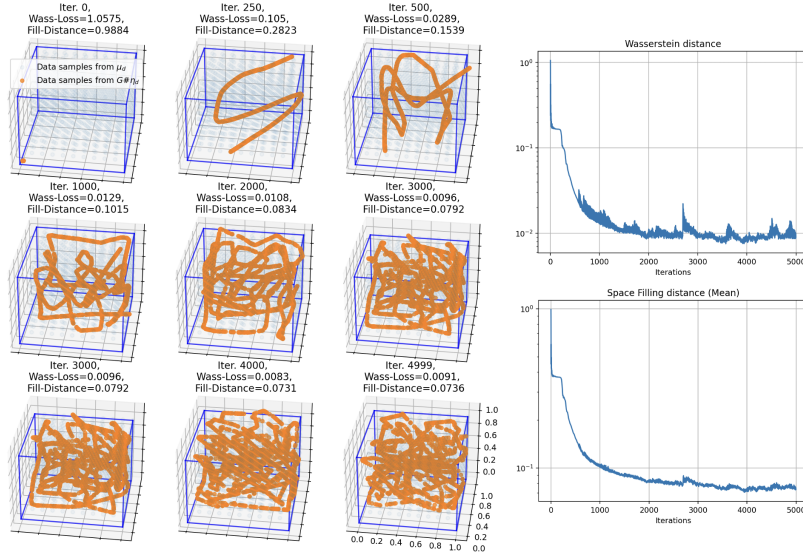


Figure 8: Simulation 3: Training results of a 5 hidden layer, 256 node each, fully connected network over 5,000 iterations mapping the 1-D uniform input on the interval  $[0,1]$  to a 3-D uniform target data manifold on the unit cube. Here the orange curve is generated by the neural network and is “filling in” the data manifold (blue).

that again a larger network with 5 hidden layers of 256 neurons each is required to achieve similar loss to the  $m = 2$  and  $m = 3$  cases. Together, the three simulations underscore the networks’ ability to adaptively map lower-dimensional inputs onto higher-dimensional man-

ifolds in a generalizable manner, effectively demonstrating our main theorems in practice. Again, see Appendix C for full implementation details and extension to  $m = d + 1$ .

## 5. Sketch of the Proof

In this section, we provide a proof sketch of our main theorems, highlighting key steps while deferring technical details to the Appendix. We start with the proof of Theorem 7.

### 5.1 Sketch Proof of Theorem 7

The proof is divided into three cases, depending on the input dimension  $m$ : when  $m > d$ , when  $2 \leq m \leq d$ , and when  $m = 1$ . We start with the simplest case:  $m > d$ .

#### 5.1.1 CASE 1: $m > d$

In this case, since there are  $m - d$  redundant input variables, we introduce a layer  $g_1 : \mathbb{R}^m \rightarrow \mathbb{R}^{d+1}$  that effectively ignores these redundant input variables:

$$g_1(x_1, \dots, x_m) = (x_1, \dots, x_{d+1}).$$

$g_1$  is a one-layer ReLU neural network with zero bias, and a weight matrix  $W = \begin{bmatrix} \mathbf{I}_{(d+1) \times (d+1)} & \mathbf{0}_{(d+1) \times (m-d-1)} \end{bmatrix}$ . This step simplifies the network's input to  $d + 1$  dimensions, aligning with the settings of Lemma 5. We then define the pushforward measure of  $\rho$  via  $g_1$ , defined on  $\mathbb{R}^{d+1}$ , by  $g_1$  as  $\rho' = g_{1\#}(\rho)$ . By Lemma 5, for any  $\epsilon > 0$ , there exists a neural network  $g' \in \mathcal{G}_{NN}(d + 1, L', p', \kappa)$  such that

$$W_1(g'_\#(\rho'), Q) < \epsilon,$$

where  $L' = O(\log(\frac{1}{\epsilon}))$ ,  $p' = O(D\epsilon^{-d/\alpha})$ ,  $\kappa = B$ .

Now we let  $g := g' \circ g_1$ , so we have

$$W_1(g_\#(\rho), Q) = W_1((g' \circ g_1)_\#(\rho), Q) = W_1(g'_\#(g_{1\#}(\rho)), Q) = W_1(g'_\#(\rho'), Q) < \epsilon.$$

Finally, observe that  $g \in \mathcal{G}_{NN}(m, L, p, \kappa)$ , where  $L = L' + 1 = O(\log(\frac{1}{\epsilon}))$ ,  $p = \max\{p', m\} = O(D\epsilon^{-d/\alpha})$ ,  $\kappa = \max\{B, 1\}$ , which finishes the proof of this case.

#### 5.1.2 CASE 2: $2 \leq m \leq d$

In this case, where the input dimension lies between 2 and  $d$ , our strategy comprises several steps. First, we find a  $m - 1$  dimensional compact Riemannian manifold  $\mathcal{M}'_\epsilon$  approximating  $\mathcal{M}$ . Then, we project the original distribution  $Q$  onto  $\mathcal{M}'_\epsilon$  to obtain  $Q'_\epsilon$ , an approximation of  $Q$ . Next, we smooth  $Q'_\epsilon$  into  $Q''_\epsilon$  to satisfy Assumption 4. Finally, we show that  $Q''_\epsilon$  can be approximated by a deep generative model, as detailed step by step below.

Firstly, the existence of such a  $m - 1$  dimensional manifold  $\mathcal{M}'_\epsilon$  is guaranteed by the following Lemma 10 leveraging the idea of a space-filling manifold, with proofs in Appendix B.1.

**Lemma 10** *Let  $\mathcal{M}$  be a  $d$  dimensional Riemannian manifold isometrically embedded in  $\mathbb{R}^D$ . Then, for any  $1 \leq q \leq d$  and any  $\epsilon > 0$ , there exists a  $q$ -dimensional manifold isometrically embedded in  $\mathbb{R}^D$ , denoted by  $\mathcal{M}'_\epsilon$  satisfying*

$$d(\mathcal{M}, \mathcal{M}'_\epsilon) = \sup_{x \in \mathcal{M}} \inf_{y \in \mathcal{M}'_\epsilon} \|x - y\| < \epsilon.$$

Secondly, we define  $Q'_\epsilon := \pi_{\epsilon\#}(Q_\epsilon)$ , where  $\pi_\epsilon : \mathcal{M} \rightarrow \mathcal{M}'_\epsilon$  is the orthogonal projection onto  $\mathcal{M}'_\epsilon$ . The Wasserstein distance  $W_1(Q, Q')$  is then bounded by  $W_1(Q, Q'_\epsilon) \leq d(\mathcal{M}, \mathcal{M}'_\epsilon) < \epsilon$ .

Thirdly, since  $Q'_\epsilon$  may not satisfy Assumption 4, we cannot directly apply Lemma 5. To address this, we find another distribution on  $\mathcal{M}'_\epsilon$ , denoted by  $Q''_\epsilon$  that satisfies Assumption 4, with  $W_1(Q'_\epsilon, Q''_\epsilon) < \epsilon$ . The existence of such  $Q''_\epsilon$  is supported by the following Lemma 11, with proofs given in Appendix B.2.

**Lemma 11** *Let  $\mathcal{M}'$  be a  $m - 1$  dimensional Riemannian manifold isometrically embedded in  $\mathbb{R}^D$ , and let  $Q'$  be a distribution on  $\mathcal{M}'$ . Then, for any  $\epsilon > 0$ , there exists a distribution  $Q''$  on  $\mathcal{M}'$  satisfying Assumption 4, with*

$$W_1(Q', Q'') < \epsilon.$$

Finally, by Lemma 5, there exists a network  $g \in \mathcal{G}_{NN}(m, L, p, \kappa)$  such that  $W_1(g_\#(\rho), Q''_\epsilon) < \epsilon$ , where

$$L = O\left(\log\left(\frac{1}{\epsilon}\right)\right), \quad p = O\left(D\epsilon^{-\frac{m}{\alpha(m, \epsilon)}}\right), \quad \kappa = B.$$

By the triangular inequality, we have

$$W_1(g_\#(\rho), Q) \leq W_1(g_\#(\rho), Q''_\epsilon) + W_1(Q''_\epsilon, Q'_\epsilon) + W_1(Q'_\epsilon, Q) \leq 3\epsilon.$$

The remaining step is to show that  $\alpha(m, \epsilon) \rightarrow 0$  as  $\epsilon \rightarrow 0$  for any  $m$ , which leads to the super-exponential complexity. However, this proof is technical, so we summarize it in Lemma 12 and defer the proofs to Appendix B.3.

**Lemma 12**  $\alpha(m, \epsilon) \rightarrow 0$  as  $\epsilon \rightarrow 0$  for any  $2 \leq m \leq d$ .

### 5.1.3 CASE 3: $m = 1$

For the case where  $m = 1$ , the approach for  $2 \leq m \leq d$  is not directly applicable, as Lemma 5 requires the input dimension to be at least two. Thus, we adopt a different strategy to construct the approximation manifold. Notably, in this case, the input space is a unit interval  $[0, 1]$ , with  $\rho = \text{Unif}(0, 1)$ .

First, we apply Lemma 10 again, with  $q = 1$ , to obtain an approximation manifold  $\mathcal{M}'_\epsilon$ , a one-dimensional space-filling curve. Subsequently, we define  $Q'_\epsilon = \pi_{\epsilon\#}(Q)$ , where  $\pi_\epsilon : \mathcal{M} \rightarrow \mathcal{M}'_\epsilon$  is the orthogonal project. This yields  $Q'_\epsilon$  as a distribution on  $\mathcal{M}'_\epsilon$  satisfying  $W_1(Q, Q'_\epsilon) \leq \epsilon$ .

Lemma 13 guarantees the construction of a smooth parameterization for the curve  $\mathcal{M}'_\epsilon$ , given by  $\eta_\epsilon : [0, 1] \rightarrow \mathcal{M}'_\epsilon$ , with  $W_1(\eta_{\epsilon\#}(\rho), Q'_\epsilon) \leq \epsilon$ . The proof is in Appendix B.4.

**Lemma 13** *Let  $\mathcal{M}'_\epsilon$  be a parametric curve of dimension 1, and let  $Q'_\epsilon$  be a distribution on  $\mathcal{M}'_\epsilon$ , then there exists a reparameterization  $\eta_\epsilon : [0, 1] \rightarrow \mathcal{M}'_\epsilon$  such that*

$$W_1(\eta_{\epsilon\#}(\rho), Q'_\epsilon) \leq \epsilon.$$

Next, Lemma 14 allows us to approximate  $\eta$  using a neural network  $g$ .

**Lemma 14 (Dahal et al. (2022) Lemma 3)** *If  $\eta(t)$  is  $\alpha$  Hölder continuous and bounded, then for any  $\epsilon > 0$  there is a ReLU network  $g$  such that*

$$\int_{[0,1]} \|g(t) - \eta(t)\| dt \leq \epsilon,$$

and  $g \in \mathcal{G}_{NN}(1, L, p, \kappa)$ , where  $L = O(\log(\frac{1}{\epsilon}))$ ,  $p = O(\epsilon^{-\frac{1}{\alpha}})$ , and  $\kappa = B$ .

Then we obtain  $g$  with  $W_1(\eta_{\epsilon\#}(\rho), g_{\#}(\rho)) \leq \epsilon$  and

$$W_1(Q, g_{\#}(\rho)) \leq W_1(Q, Q'_{\epsilon}) + W_1(Q'_{\epsilon}, \eta_{\epsilon\#}(\rho)) + W_1(\eta_{\epsilon\#}(\rho), g_{\#}(\rho)) < 3\epsilon.$$

Finally, we note that  $\alpha = \alpha(\epsilon)$  depends on  $\epsilon$  through  $\eta_{\epsilon}$ , and we claim that  $\alpha(\epsilon) \xrightarrow{\epsilon \rightarrow 0} 0$ . This is summarized as Lemma 15 with proofs in Appendix B.5, which completes the proof of this  $m = 1$  case.

**Lemma 15**  $\alpha(\epsilon) \xrightarrow{\epsilon \rightarrow 0} 0$ .

## 5.2 Sketch Proof of Theorem 8

Let  $x_1, \dots, x_n$  be iid samples from  $Q$ , and let  $Q_n$  denote the associated empirical distribution. By Theorem 7, for any  $\epsilon > 0$ , there exists  $g \in \mathcal{G}_{NN}(m, L, p, \kappa)$  such that  $W_1(Q, g_{\#}(\rho)) < \epsilon$ , where  $L = O(\log(\frac{1}{\epsilon}))$ ,  $p = O(D\epsilon^{\frac{-m}{\alpha(m, \epsilon)}})$ , and  $\kappa = B$ .

For the empirical minimizer  $\hat{g}_n$ , we can break down the Wasserstein distance  $W_1(\hat{g}_{n\#}(\rho), Q)$  as follows:

$$\begin{aligned} W_1(\hat{g}_{n\#}(\rho), Q) &\leq W_1(\hat{g}_{n\#}(\rho), Q_n) + W_1(Q_n, Q) \leq W_1(g_{\#}(\rho), Q_n) + W_1(Q_n, Q) \\ &\leq W_1(g_{\#}(\rho), Q) + W_1(Q, Q_n) + W_1(Q_n, Q) \leq \epsilon + 2W_1(Q, Q_n). \end{aligned}$$

Given Assumptions 3 and 4, Lemma 5 of Dahal et al. (2022) implies that for any  $\delta > 0$ , a constant  $C_{\delta}$  exists such that

$$\mathbb{E}[W_1(Q, Q_n)] \leq C_{\delta} n^{-\frac{1}{d+\delta}},$$

where  $C_{\delta}$  is a constant independent of  $n$ . Setting  $\epsilon = n^{-\frac{1}{d+\delta}}$ , we have

$$\mathbb{E}[W_1(\hat{g}_{\epsilon\#}(\rho), Q)] \leq (1 + 2C_{\delta}) n^{-\frac{1}{d+\delta}}.$$

Thus, we have achieved the claimed expected Wasserstein distance with desired complexity of  $\hat{g}_n$ :  $L = O(\log(n^{\frac{1}{d+\delta}}))$ ,  $O(Dn^{\frac{m}{(m+\delta)\alpha(m, n, d, \delta)}})$ , and  $\kappa = B$ . In particular, when  $m > d$ ,  $\alpha(m, n, d, \delta)$  remains constant ( $\alpha$ ), while when  $m \leq d$ ,  $\lim_{n \rightarrow \infty} \alpha(m, n, d, \delta) = \lim_{\epsilon \rightarrow 0} \alpha(m, \epsilon) = 0$  as per Theorem 7.

## 6. Discussion

Using the theory of space-filling curves, in this work we establish key relationships between the input dimension, the true dimension of the data manifold, and the approximation error of fitting deep generative networks. Specifically, we have shown that deep generative networks can learn the manifold structure in a generalizable manner regardless of the input dimension, even when the input dimension is as low as one. Furthermore, we quantify the complexity trade-off among network size, input dimension, and approximation error, showing that underestimating the input dimension below the manifold dimension introduces a super-exponential increase in the width of the network. Furthermore, these results did not require any additional assumptions compared to previous work (Dahal et al., 2022). Toy simulation studies on two- and three-dimensional manifolds provide empirical evidence of GANs learning such space-filling curves from various input dimensions, offering additional explanation to the success of GANs and similar models to learn distributions without explicit need for dimensionality estimation in practice.

Based on these results, there are multiple potential directions for future work. First, although we have derived statistical guarantees for existence, there is no guarantee on being able to practically fit such models, especially when the input dimension is heavily underestimated. Second, while we have established upper bounds on the complexity necessary to achieve certain approximation errors, identifying lower bounds poses a formidable challenge and remains an open question. The absence of explicit lower bounds in existing literature highlights the complexity and novelty of this problem area. Establishing these bounds would not only deepen our understanding of the behavior of generative networks but also potentially guide the development of more efficient network architectures. Third, exploring the phase transition in model complexity versus approximation error as a potential method for estimating the intrinsic dimension of data manifolds could provide practical insights into dimensionality reduction and the effective training of generative models. Finally, although we have shown that neural networks are capable of learning distributions exhibiting low dimensional manifold structures, there remains situations in which the data do not lie on a manifold of a single intrinsic dimension throughout, as is the case in the union of manifolds hypothesis (Brown et al., 2022). Such extension would further elucidate the ability of these deep generative models to empirically approximate arbitrary datasets in practice.

## Acknowledgment

DL acknowledges Wenjing Liao for a helpful discussion. KW was supported by National Institute of Health (NIH) grant R01 LM014407; HN was supported by NIH R01 CA253450; YW was supported in part by the Office of Naval Research under grant N00014-23-1-2590, the National Science Foundation under grant No. 2231174, No. 2310831, No. 2428059, No. 2435696, No. 2440954, and a Michigan Institute for Data Science Propelling Original Data Science (PODS) grant; DL was supported by NIH R01 LM014407, NIH P50 CA058223, NIH P42 ES031007.

## Appendix A. Code Availability

All code for experiments and generating paper figures can be found in the GitHub repository at <https://github.com/hong-niu/dgm24>. All experiments were run on a single machine with an Nvidia RTX 4080 GPU with 16 GB of memory and adapted from the Python Optimal Transport (POT) package Flamary et al. (2021) for computing the Wasserstein distance.

## Appendix B. Additional Proof Details

In this section, we prove five lemmas used in the sketch of the proofs of Theorem 7 and Theorem 8, namely, Lemma 10, Lemma 11, Lemma 12, Lemma 13, and Lemma 15. Note that Lemma 14 is directly from Lemma 3 of Dahal et al. (2022), so we skip its proof.

### B.1 Proof of Lemma 10

We first prove Lemma 10, which shows that for a compact  $d$  dimensional manifold  $\mathcal{M}$ , there exists a smaller dimensional manifold  $\mathcal{M}'_\epsilon$  that approximates the manifold  $\mathcal{M}$ .

**Proof** Since  $\mathbb{R}^D$  is a separable metric space,  $\mathcal{M} \subset \mathbb{R}^D$  is also separable. Let  $E \subset \mathcal{M}$  be a dense countable subset of  $\mathcal{M}$ , enumerated  $E = \{e_1, e_2, \dots\}$ . Because  $E$  is dense on  $\mathcal{M}$ , we have

$$\lim_{n \rightarrow \infty} \sup_{x \in \mathcal{M}} \inf_{y \in \{e_1, \dots, e_n\}} \|x - y\|_2^2 = 0.$$

Suppose that we can find a sequence of manifolds  $\mathcal{M}_n$  satisfying for all  $n \in \mathbb{N}$ ,

$$E_n := \{e_1, \dots, e_n\} \subseteq \mathcal{M}_n.$$

We have

$$0 \leq \sup_{x \in \mathcal{M}} \inf_{y \in \mathcal{M}_n} \|x - y\|_2^2 \leq \sup_{x \in \mathcal{M}} \inf_{y \in \{e_1, \dots, e_n\}} \|x - y\|_2^2.$$

Consequently:

$$\lim_{n \rightarrow \infty} \sup_{x \in \mathcal{M}} \inf_{y \in \mathcal{M}_n} \|x - y\|_2^2 = 0.$$

Therefore, for any  $\epsilon > 0$ , there exists a large enough  $n$  such that

$$\sup_{x \in \mathcal{M}} \inf_{y \in \mathcal{M}_n} \|x - y\|_2^2 < \epsilon.$$

It suffices to show that for any finite set  $E_n$ , there exists a Riemannian Manifold  $\mathcal{M}_n$  interpolating  $E_n$ , that is,  $E_n \subseteq \mathcal{M}_n$ .

We know there exists a  $q$  dimensional subspace, represented by the associated projection matrix  $P$ , such that, for all  $i \neq j$ ,  $Pe_i \neq Pe_j$ . We can choose a basis of this subspace to be the first  $q$  coordinates through a change of basis. For convenience, we thus claim without loss of generality that the projection onto the first  $q$  coordinates  $\pi_q(x_1, \dots, x_n) = (x_1, \dots, x_q)$  is injective on  $E_n$  so that

$$\pi_q(e_i) \neq \pi_q(e_j) \text{ when } i \neq j.$$

Then there exists infinitely many interpolants,  $f$ , satisfying  $y_i = f(\pi_q(e_i))$ .



Included in our options are smooth interpolants, such as polynomial interpolants, splines, variational minimizing surfaces, RBF interpolations, Backus-Gilbert (Wendland, 2004), etc. We choose the cubic polyharmonic/thin plate spline due to its simplicity (Duchon, 1977):

$$f = \min_{f \in W_2^m(\pi_q(M)), y_i = f(\pi_q(e_i))} J_m^d(f),$$

where  $J_m^d(f) = \sum_{\alpha_1 + \dots + \alpha_d = m} \frac{m!}{\alpha_1! \dots \alpha_d!} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \left( \frac{\partial^m f}{\partial x_1^{\alpha_1} \dots \partial x_d^{\alpha_d}} \right) \prod_{j=1}^d dx_j$ .

This interpolant is a smooth function on a compact domain, with a bounded derivative and its image is a smooth manifold. Further, we consider the function:

$$F : \pi_q(\mathcal{M}) \rightarrow \mathbb{R}^n, (x_1, \dots, x_q) \mapsto (x_1, \dots, x_q, f_{q+1}(x_1, \dots, x_q), \dots, f_n(x_1, \dots, x_q)).$$

Note that the image  $F(\pi_q(\mathcal{M}))$  is a Riemannian manifold that interpolates  $F(\pi_q(e_i)) = e_i$ . Thus,  $E \subseteq F(\pi_q(\mathcal{M}))$  and

$$d(F(\pi_q(e_i)), \mathcal{M}) \leq d_1(E, \mathcal{M}) \leq \epsilon.$$

Hence, for any  $d$  dimensional Riemannian manifold  $\mathcal{M}$  and any  $\epsilon > 0$ , there exists a  $q$  dimensional Riemannian manifold  $\mathcal{M}'$ , such that

$$d(\mathcal{M}, \mathcal{M}'_\epsilon) = \sup_{x \in \mathcal{M}} \inf_{y \in \mathcal{M}'_\epsilon} \|x - y\| < \epsilon.$$

■

## B.2 Proof of Lemma 11

We now prove Lemma 11, which shows that for any distribution on a compact manifold, we may find a sufficiently close distribution that satisfies Assumption 4.

### Proof

The objective of this lemma is to perturb  $Q'$  slightly, creating a perturbed distribution  $Q''$  that satisfies Assumption 4. The proof involves the following five steps:

1. **Decomposition of the manifold:** Partition the manifold  $\mathcal{M}'_\epsilon$  into a finite union of balls, with each as a coordinate chart derived from some Euclidean ball.
2. **Local pull back distribution:** Pull back the local distribution  $Q'$  through the exponential map.
3. **Smooth local distributions:** Smooth each local pull-back distribution to ensure closeness to the original pull-back.
4. **Pushforward and reweighting:** Push these local distributions forward to  $\mathcal{M}'$  using their exponential maps. Paste them together to form a new distribution, and re-weight it to account for overlaps.

5. **Density adjustment:** Add a small uniform density to ensure a positive lower bound on density and show that this new distribution is sufficiently close to the original distribution and satisfies Assumptions 4.

Since  $\mathcal{M}'$  is compact, the injectivity radius, denoted by  $\iota$ , is strictly positive (Do Carmo and Flaherty Francis, 1992). Fix  $0 < r < \iota$  and denote  $U_x = \exp_x(B_x(0, r))$ , where  $\exp_x$  is the exponential map at  $x$  and  $B_x(0, r)$  is the Euclidean ball in the tangent space  $T_x\mathcal{M}'$ . Again, due to the compactness, there exists a finite covering  $\{\exp_{x_j}(B_{x_j}(0, r)) : j = 1, 2, \dots, N\}$ .

Let  $Q'_{x_j}$  be the restriction of  $Q'$  onto  $U_{x_j}$ , that is  $Q'_{x_j}(S) = \int_S \mathbb{1}(x \in U_{x_j}) dQ'(x)$ . Since  $r < \iota$ , the exponential map  $\exp_{x_j}$  is invertible on  $U_{x_j}$ , so the pullback of  $Q'_{x_j}$  through  $\exp_{x_j}(\cdot)$ , denoted by  $P_{x_j}$ , a measure on  $B_{x_j}(0, r)$ , is well-defined.

However, the pull-back measure  $P_{x_j}$  may not be smooth enough. To address this, for  $\delta > 0$ , we consider the function

$$H_\delta(x, y) = C_\delta(x) \mathbb{1}\{\|x - y\| < \delta\},$$

where  $C_\delta(x)$  is the normalizing constant  $\frac{1}{C_\delta(x)} = \int_{B_{x_j}(0, r)} \mathbb{1}(\|x - y\| < \delta) dy$ .

Then denote the locally smoothed measure by  $P_{x_j}^\delta$ , with density:

$$p_{x_j}^\delta(y) = \int_{B_{x_j}(0, r)} H_\delta(x, y) dP_{x_j}.$$

Since the transport cost between  $(P_{x_j}, P_{x_j}^\delta)$  is bounded by  $\sup_{H_\delta(x, y)=1} \|x - y\| \leq \delta$ , we have  $W_1(P_{x_j}, P_{x_j}^\delta) \leq \delta$ .

Pushing  $P_{x_j}^\delta$  forward through the exponential map onto  $\mathcal{M}'$  yields a locally smoothed distribution on  $U_{x_j}$ ,  $\exp_{x_j\#}(P_{x_j}^\delta)$ . We have:

$$\begin{aligned} W_1(\exp_{x_j\#}(P_{x_j}), \exp_{x_j\#}(P_{x_j}^\delta)) &= \inf_{\gamma \in \Gamma(\exp_{x_j\#}(P_{x_j}), \exp_{x_j\#}(P_{x_j}^\delta))} \int_{U_{x_j} \times U_{x_j}} \|x - y\| d\gamma \\ &= \inf_{\gamma' \in \Gamma(P_{x_j}, P_{x_j}^\delta)} \int_{B_{x_j}(0, r) \times B_{x_j}(0, r)} \|\exp_{x_j}(x) - \exp_{x_j}(y)\| d\gamma' \\ &\leq L_{x_j} \inf_{\gamma \in \Gamma(P_{x_j}, P_{x_j}^\delta)} \int_{B_{x_j}(0, r) \times B_{x_j}(0, r)} \|x - y\| d\gamma, \end{aligned}$$

where  $\Gamma$  is the coupling and  $L_{x_j}$  is the Lipschitz constant of  $\exp_{x_j}$  on  $\overline{B_{x_j}(0, r)}$  (Do Carmo and Flaherty Francis, 1992). Note that as  $\delta$  approaches 0,  $\exp_{x_j}(P_{x_j}^\delta)$  converges in distribution to  $Q'_{x_j}$ .

Now we recombine these locally smoothed measures back into a new measure:

$$Q_{\delta, 0} = \sum_{j=1}^N \exp_{x_j\#}(P_{x_j}^\delta).$$

Then let  $N(x)$  to be the number of  $U_{x_j}$  containing  $x$ , that is  $N(x) = \sum_{j=1}^N \mathbb{1}(x \in U_{x_j})$  so that:

$$\lim_{\delta \rightarrow 0} Q_{\delta,0} = \lim_{\delta \rightarrow 0} \sum_{c=1}^N \exp_{x_{j\sharp}}(P_{x_j}^\delta) = \int_{\mathcal{M}'} N(x) dQ'(x).$$

To compensate for overlap in the covering, we adjust  $Q_{\delta,0}$  to define a new measure

$$Q_\delta(A) = \int_A \frac{1}{N(x)} dQ_{\delta,0}.$$

It is clear that as  $\delta \rightarrow 0$ ,  $Q_\delta \rightarrow_d Q'$ . Due to the compactness of  $\mathcal{M}'$ , the cost function is bounded and thus

$$\lim_{\delta \rightarrow 0} W_1(Q', Q_\delta) \rightarrow 0.$$

So far, however, the density of  $Q_\delta$ , denoted by  $q_\delta$  may not be positively lower bounded, as required by Assumption 4. To fill in the final gap, we conduct the following surgery to  $Q_\delta$ .

By the compactness of  $\mathcal{M}'$ , the uniform density, denoted by  $u(x)$  is well defined on  $\mathcal{M}'$ . Let  $\theta > 0$  be an arbitrary small number, we define  $q'' = (1 - \theta)q_\delta + \theta u$ . Let  $Q''$  be the corresponding distribution, we observe that 1):  $Q''$  satisfies Assumption 4 for any  $\delta$  and  $\theta$ ; 2)  $Q''$  converges in distribution to  $Q'$  when  $\delta, \theta \rightarrow 0$ . That is, for any  $\epsilon$ , there exists  $\delta > 0$  and  $\theta > 0$  such that  $W_1(Q', Q_\delta) < \frac{\epsilon}{2}$  and  $W_1(Q'', Q_\delta) < \frac{\epsilon}{2}$ , so that

$$W_1(Q'', Q') \leq W_1(Q'', Q_\delta) + W_1(Q_\delta, Q') < \epsilon,$$

which closes the proof. ■

### B.3 Proof of Lemma 12

We now prove Lemma 12, which shows that when the input dimension is smaller than the true dimension, the quantity  $\alpha(m, \epsilon)$  approaches 0 as  $\epsilon$  approaches 0.

**Proof** In existing literature, such as Dahal et al. (2022),  $\alpha$  is often treated as a constant, since the manifold under consideration is fixed. However, in our framework, both the approximating space-filling manifold  $\mathcal{M}'_\epsilon$  and the approximating distribution  $Q'_\epsilon$  change with  $\epsilon$ . In this situation, it becomes imperative to keep track of the “constant”  $\alpha$  while the manifold  $\mathcal{M}'_\epsilon$  and  $Q'_\epsilon$  are evolving. This task is extremely challenging for several reasons. First, much of the existing literature, including Villani et al. (2009); Dahal et al. (2022), treats  $\alpha$  as a constant, without a in-depth discussion. Second, even when we scrutinize their proofs, they typically establish only the existence of  $\alpha$ , without any explicit construction. Third, the intricacies of  $\alpha$  are hidden within the study of Monge-Ampère equations of the potentials of the density functions of  $Q'_\epsilon$ , a distinct, although related, field of study.

Our analysis begins with the recent contributions of Dahal et al. (2022), who claim that there exists a constant  $\alpha$  so that the complexity  $p = O(D\epsilon^{-d/\alpha})$ . The foundation of their construction of optimal transport relies on Theorem 12.50 from Villani et al. (2009), also known as Caffarelli’s regularity theory. Within this theorem, the constant  $\alpha$ , denoted by  $\beta$

by Villani, is introduced ambiguously as “for some  $\beta \in (0, 1)$ ”, without further elucidation provided. To address this, we examined a series of papers by Caffarelli (Caffarelli, 1990a,b, 1991, 1992a,b, 1996), where we discovered that the constant  $\alpha$  depends on the infimum of the density function of  $Q'_\epsilon$ , denoted by  $q'_\epsilon$ . In essence, as  $\epsilon \rightarrow 0$ ,  $\inf q'_\epsilon \rightarrow 0$ , leading to the conclusion that  $\alpha(m, \epsilon) \rightarrow 0$ . We prove this claim step by step.

1. Prove  $\text{Vol}(\mathcal{M}'_\epsilon) \xrightarrow{\epsilon \rightarrow 0} \infty$ .

First we prove that as  $\epsilon \rightarrow 0$ , the volume of the space-filling manifold  $\mathcal{M}'_\epsilon$  goes to infinity. Note that this volume is the Riemannian volume of the  $m - 1$  dimensional manifold  $\mathcal{M}'_\epsilon$ , not the volume in the ambient space. By Lemma 10, the tube around  $\mathcal{M}'_\epsilon$  with radius  $2\epsilon$ , denoted by  $T_{2\epsilon}(\mathcal{M}'_\epsilon) := \{x \in \mathbb{R}^D : d(x, \mathcal{M}'_\epsilon) < 2\epsilon\}$ , contains  $\mathcal{M}$ . Then, by Weyl’s Tube Theorem (Weyl, 1939; Gray, 2003), the volume of the tube is given by the following series expansion:

$$\begin{aligned} \text{Vol}(T_{2\epsilon}(\mathcal{M}'_\epsilon)) = & \frac{(\pi 2\epsilon^2)^{\frac{D-(m-1)}{2}}}{(\frac{1}{2}(D-(m-1)))!} \sum_{l=0}^{[(m-1)/2]} \frac{k_{2l}(\mathcal{M}'_\epsilon)(2\epsilon)^{2l}}{(D-(m-1)+2)(D-(m-1)+4)\cdots(D-(m-1)+2l)}, \end{aligned}$$

where  $k_{2l}(\mathcal{M}'_\epsilon)$  is the integrated mean curvature of  $\mathcal{M}'_\epsilon$  and  $k_0(\mathcal{M}'_\epsilon) = \text{Vol}(\mathcal{M}'_\epsilon)$  as a special case when  $l = 0$ . Focusing on the leading term, we have

$$\text{Vol}(T_{2\epsilon}(\mathcal{M}'_\epsilon)) = \frac{(\pi 2\epsilon^2)^{\frac{D-(m-1)}{2}}}{(\frac{1}{2}(D-(m-1)))!} (\text{Vol}(\mathcal{M}'_\epsilon) + O(\epsilon^2)). \quad (1)$$

Similarly, we estimate the volume of the tube around  $\mathcal{M}$ ,  $T_\epsilon(\mathcal{M})$  as:

$$\text{Vol}(T_\epsilon(\mathcal{M})) = \frac{(\pi\epsilon^2)^{\frac{D-d}{2}}}{(\frac{1}{2}(D-d))!} (\text{Vol}(\mathcal{M}) + O(\epsilon^2)). \quad (2)$$

Now we show that  $T_{2\epsilon}(\mathcal{M}'_\epsilon) \supset T_\epsilon(\mathcal{M})$ . For any  $x \in T_\epsilon(\mathcal{M})$ , there exists  $y \in \mathcal{M}$  such that  $\|x - y\| < \epsilon$ . By the construction of  $\mathcal{M}'_\epsilon$ , there exists  $z \in \mathcal{M}'_\epsilon$  such that  $\|y - z\| < \epsilon$ , as a result,

$$d(x, \mathcal{M}'_\epsilon) \leq \|x - z\| \leq \|x - y\| + \|y - z\| \leq 2\epsilon,$$

which implies  $x \in T_{2\epsilon}(\mathcal{M}'_\epsilon)$  and hence  $T_{2\epsilon}(\mathcal{M}'_\epsilon) \supset T_\epsilon(\mathcal{M})$ .

Linking the two volumes in Equation (1) and Equation (2), we conclude that

$$\begin{aligned} & \frac{(\pi 2\epsilon^2)^{\frac{D-(m-1)}{2}}}{(\frac{1}{2}(D-(m-1)))!} (\text{Vol}(\mathcal{M}'_\epsilon) + O(\epsilon^2)) \geq \frac{(\pi\epsilon^2)^{\frac{D-d}{2}}}{(\frac{1}{2}(D-d))!} (\text{Vol}(\mathcal{M}) + O(\epsilon^2)) \\ \implies & \text{Vol}(\mathcal{M}'_\epsilon) \gtrsim C \epsilon^{\frac{m-1-d}{2}} \xrightarrow{\epsilon \rightarrow 0} \infty, \end{aligned}$$

where the last limit is due to  $m - 1 - d < 0$ .

2. Prove  $\inf q'_\epsilon \xrightarrow{\epsilon \rightarrow 0} 0$ .

Since  $\mathcal{M}'_\epsilon$  is compact and  $q'_\epsilon$  integrates to one, we have

$$1 = \int_{\mathcal{M}'_\epsilon} q'_\epsilon d\text{Vol}_{\mathcal{M}'_\epsilon} \geq \text{Vol}(\mathcal{M}'_\epsilon) \cdot \inf q'_\epsilon.$$

As a result, we conclude that  $\inf q'_\epsilon \xrightarrow{\epsilon \rightarrow 0} 0$  since  $\text{Vol}(\mathcal{M}'_\epsilon) \xrightarrow{\epsilon \rightarrow 0} \infty$ .

3. Prove  $\alpha(m, \epsilon) \xrightarrow{\epsilon \rightarrow 0} 0$ .

As stated in Theorem 12.50 of Villani et al. (2009), the optimal transport mapping, denoted by  $F_\epsilon$ , is represented by its potential  $F_\epsilon \in C^{1,\alpha(m,\epsilon)}$  such that  $F_\epsilon = \nabla \psi_\epsilon \in C^{\alpha(m,\epsilon)}$ . Moreover, we know that  $q'_\epsilon(\nabla \psi_\epsilon) \det D_{ij} \psi_\epsilon = q$  from Equation (1) of Caffarelli (1992b), where  $D_{ij}$  represents the Hessian matrix. The existence of such  $\alpha(m, \epsilon)$  was first given by Caffarelli (1991), which relies on the key assumption that there exists constants  $\lambda_1, \lambda_2$  such that

$$0 < \lambda_{1,\epsilon} \leq \det D_{ij} \psi_\epsilon \leq \lambda_{2,\epsilon} < \infty. \quad (3)$$

Then, the key quantity we are interested, is defined as  $\alpha(m, \epsilon) = -\log_2 \delta(\epsilon)$ , where  $\delta(\epsilon) < 1$  is the constant such that  $h_{\epsilon,1/2}(x - x_0) < \delta(\epsilon) < h_{\epsilon,1}(x - x_0)$ , where  $h_{\epsilon,\beta}$  is the cone generated by  $x_0$  and the level surface  $\psi_\epsilon = \beta$  (see Lemma 2 of Caffarelli (1991) for more details). The existence of such  $\delta(\epsilon)$  is given by Lemma 2 of Caffarelli (1991). Since  $\inf q'_\epsilon \xrightarrow{\epsilon \rightarrow 0} 0$ , we know that  $\lambda_{2,\epsilon} = \sup \det D_{ij} \psi_\epsilon \rightarrow \infty$ , which violates the key assumption in Caffarelli (1991), leading to the fact that  $\psi_\epsilon$  has infinite derivative and hence infinite local changes and inseparable level sets, so that  $\delta(\epsilon) \xrightarrow{\epsilon \rightarrow 0} 1$ . As a result  $\alpha(m, \epsilon) = -\log_2 \delta(\epsilon) \xrightarrow{\epsilon \rightarrow 0} 0$ , as desired. ■

#### B.4 Proof of Lemma 13

We now prove Lemma 13, which states that an approximating curve for a manifold may be reparameterized by a differentiable function such that the pushforwards of the uniform distribution is close to the true distribution.

**Proof** We first introduce the arc-length parameterization as  $\xi(t) : [0, L_\epsilon] \rightarrow \mathcal{M}'_\epsilon$ , which makes  $\xi$  invertible with a constant speed  $\|\xi'(t)\| \equiv 1$ . For simplicity, we assume the total length  $L_\epsilon = 1$  in this proof. Let  $P_\xi$  be the pull back of  $Q'_\epsilon$  onto  $[0, 1]$  through  $\xi$ , that is,  $\xi_\#(P_\xi) = Q'_\epsilon$ . Then for  $\delta > 0$ , define

$$H_\delta(x, y) = C_\delta(x) \mathbf{1}\{\|x - y\| < \delta\},$$

where  $C_\delta(x)$  is the normalizing constant:

$$\frac{1}{C_\delta(x)} = \int_{[0,1]} \mathbf{1}\{\|x - y\| < \delta\} dy.$$

Then we define the following density function

$$f_\delta(x) := \int_{[0,1]} H_\delta(x, y) dP_\xi \leq \int_{[0,1]} \sup_{x \in [0,1]} \|C_\delta(x)\| dP_\xi = \sup_{x \in [0,1]} \|C_\delta(x)\| = \frac{1}{\delta}.$$

Denote the corresponding distribution as  $F_\delta$ . Let  $X \sim P_\xi$  and define  $Y \mid X = x$  to be uniform on the interval  $[x - \delta, x + \delta] \cap [0, 1]$ , so that  $(X, Y)$  is a coupling in  $\Gamma(P_\xi, F_\delta)$ . The cost of such a transport is bounded by

$$\mathbb{E}_{(X,Y)} \|X - Y\| = \mathbb{E}_X [\mathbb{E}_Y [\|X - Y\| \mid X]] \leq \mathbb{E}_X [\mathbb{E}_Y [\delta \mid X]] = \delta.$$

By the definition of the Wasserstein distance,  $W_1(F_\delta, P_\xi) < \delta$ . Subsequently, we can bound

$$\begin{aligned} W_1(\xi_\#(F_\delta), Q'_\epsilon) &= \inf_{\gamma \in \Gamma(\xi_\#(F_\delta), Q'_\epsilon)} \int_{\mathcal{M}'_\epsilon \times \mathcal{M}'_\epsilon} \|x - y\| d\gamma = \inf_{\gamma \in \Gamma(\xi_\#(F_\delta), \xi_\#(P_\xi))} \int_{\mathcal{M}'_\epsilon \times \mathcal{M}'_\epsilon} \|x - y\| d\gamma \\ &= \inf_{\gamma \in \Gamma(F_\delta, P_\xi)} \int_{[0,1] \times [0,1]} \|\xi(x) - \xi(y)\| d\gamma \leq \inf_{\gamma \in \Gamma(F_\delta, P_\xi)} \int_{[0,1] \times [0,1]} \|x - y\| d\gamma \\ &= W_1(F_\delta, P_\xi) \leq \delta. \end{aligned}$$

If the density  $f_\delta(x)$  is lower bounded away from 0, then the Cumulative Distribution Function (CDF) of  $f_\delta$ , denoted by  $C_\delta(x)$ , is invertible. Then we use the inverse transform to approximate  $Q'_\epsilon$ :

$$W_1((\xi \circ C_\delta^{-1})_\#(\rho), Q'_\epsilon) \leq \delta.$$

Thus,  $\eta_\epsilon(t) := \xi(F_\delta^{-1}(t))$  is our desired parametrization. Here, we highlight its dependence on  $\epsilon$  through the domain  $\mathcal{M}'_\epsilon$ , which is crucial for Lemma 15.

If  $f_\delta$  is not lower bounded away from 0, we can adopt the same technique in step 5 in the proof of Lemma 11, where we perturb it by a small uniform distribution with density  $u$ :  $f'_\delta(x) = (1 - \theta)f_\delta(x) + u$ , where  $\theta > 0$ . Denoting the CDF of  $f'_\delta$  as  $C'_\delta$ , we define  $\eta_\epsilon(t) := \xi(C'^{-1}_\delta(t))$ . As a result, there exists  $\delta, \theta > 0$  such that

$$W_1(\eta_\#(\rho), Q'_\epsilon) = W_1((\xi \circ C'^{-1}_\delta)_\#(\rho), Q'_\epsilon) < \epsilon.$$

■

## B.5 Proof of Lemma 15

We now explain the proof of Lemma 15, which shows that  $\lim_{\epsilon \rightarrow 0} \alpha(m, \epsilon) = 0$  when  $m = 1$ .

**Proof** The proof follows the same structure as the proof of Lemma 12, with exactly the same 3 steps. In fact, the arguments presented in the proof of Lemma 12 are applicable for any dimension  $m$ . ■

### B.6 Proof of Corollary 9

In Dahal et al. (2022), the authors first prove the existence of an oracle map  $g$  that perfectly transports the input distribution to the target distribution, that is  $Q = g_{\#}(\rho)$ . They then approximate this function  $g$  via a neural network  $f$  such that  $\|f - g\|_{L_1(\rho)} \leq \epsilon$ , with explicit complexity bound on  $f$ . Then, in Lemma 2 of their paper, they show that one can further bound the Wasserstein-1 distance by  $W_1(f_{\#}(\rho), g_{\#}(\rho)) \leq C\|f - g\|_{L_1(\rho)} \leq C\epsilon$ .

To generalize the setting to Wasserstein- $p$  distances, we need two components. First we extend Lemma 2 of Dahal et al. (2022) to bound Wasserstein- $p$  distances by  $\|f - g\|_{L_p(\rho)}$ . Then we find a suitable neural network  $f$  to control  $\|f - g\|_{L_p(\rho)}$ . We start with the Wasserstein- $p$  distance:

$$W_p(f_{\#}(\rho), g_{\#}(\rho)) = \inf_{\pi \sim \Gamma(f_{\#}(\rho), g_{\#}(\rho))} \mathbb{E}_{(x,y) \sim \pi} (\|x - y\|_2^p)^{1/p}.$$

Because the Wasserstein- $p$  distance is the infimum over all couplings, we may find an upper bound by using any choice of coupling. The most obvious choice of coupling is  $(f(z), g(z)) \sim (f_{\#}(\rho), g_{\#}(\rho))$  where  $z \sim \rho$ , which yields the following upper bound:

$$\begin{aligned} W_p(f_{\#}(\rho), g_{\#}(\rho))^p &= \inf_{\pi \in \Gamma(f_{\#}(\rho), g_{\#}(\rho))} \mathbb{E}_{(x,y) \sim \pi} \|x - y\|_2^p \\ &\leq \mathbb{E}_{z \sim \rho} \|f(z) - g(z)\|_2^p \\ &\leq \int \|f(x) - g(x)\|_2^p d\rho \\ &= \|f - g\|_{L_p(\rho)}^p. \end{aligned}$$

According to Dahal et al. (2022), the oracle  $g$  may be constructed so as to be  $\alpha$  Hölder continuous, and thus continuous. The Universal Approximation Theorem of neural networks Hornik (1989) states that if  $g$  is continuous, we may construct a neural network  $f$  such that  $\|f - g\|_{\infty} \leq \epsilon$ . Note that  $\|f - g\|_{L_p(\rho)} \leq \|f - g\|_{\infty}$ , so we have

$$W_p(f_{\#}(\rho), g_{\#}(\rho)) \leq \|f - g\|_{L_p(\rho)} \leq \|f - g\|_{\infty} \leq \epsilon,$$

since  $\rho$  is a probability measure.

For the empirical result, we follow the same line of argument as in the Wasserstein-1 case. Define  $Q_n$  to be the empirical distribution and  $\hat{g}_{n\#}$  to be the empirical risk minimizer. Because  $Q$  is bounded, we know it has a finite  $p$ -th moment, so we have

$$\begin{aligned} W_p(\hat{g}_{n\#}(\rho), Q) &\leq W_p(\hat{g}_{n\#}(\rho), Q_n) + W_p(Q_n, Q) \leq W_p(g_{\#}(\rho), Q_n) + W_p(Q_n, Q) \\ &\leq W_p(g_{\#}(\rho), Q) + W_p(Q, Q_n) + W_p(Q_n, Q) \leq \epsilon + 2W_p(Q, Q_n). \end{aligned}$$

From Weed and Bach (2019), we know that if the Minkowski dimension of  $Q$  is  $\gamma$ , then for all  $\delta > 0$ , there exists  $C_{\delta} > 0$  such that  $\mathbb{E}(W_p(Q, Q_n)) \leq C_{\delta} n^{\frac{1}{\gamma+\delta}}$ . By assumption,  $Q$  is supported on a compact  $d$  dimensional Riemannian manifold and consequently its Minkowski dimension is  $d$ . Thus, by choosing  $\epsilon = n^{-\frac{1}{d+\delta}}$  as we did before, we obtain  $\mathbb{E}[W_p(\hat{g}_{n\#}(\rho), Q)] \leq (1 + 2C_{\delta})n^{-\frac{1}{d+\delta}}$ .

While we have proved the existence of an approximating deep neural network for the Wasserstein- $p$  distance, providing specific bounds on the network’s complexity analogous to those established for the Wasserstein-1 distance in Theorem 7 and Theorem 8 remains unresolved. In the case of  $W_1$ , the complexity bounds are intricately tied to the construction detailed in Lemma 2 of Dahal et al. (2022). Extending these bounds to  $W_p$  distances would require developing a similar foundational lemma tailored to  $W_p$  settings, a task that involves substantial theoretical innovation and is not immediately straightforward. Another potential approach involves leveraging results from Ohn and Kim (2019), which offer complexity bounds in the  $L_p$  norm; however, this approach introduces a new challenge as the bound on network weights,  $\kappa$ , becomes dependent on  $\epsilon$ . Removing the dependency on  $\epsilon$  in the bound for network weights is not straightforward, requiring further investigation. Whether we develop constructions similar to Lemma 2 of Dahal et al. (2022) for  $W_p$  distances, or apply theories like those in Ohn and Kim (2019), the challenges are nontrivial. We consider this an important direction for future research.

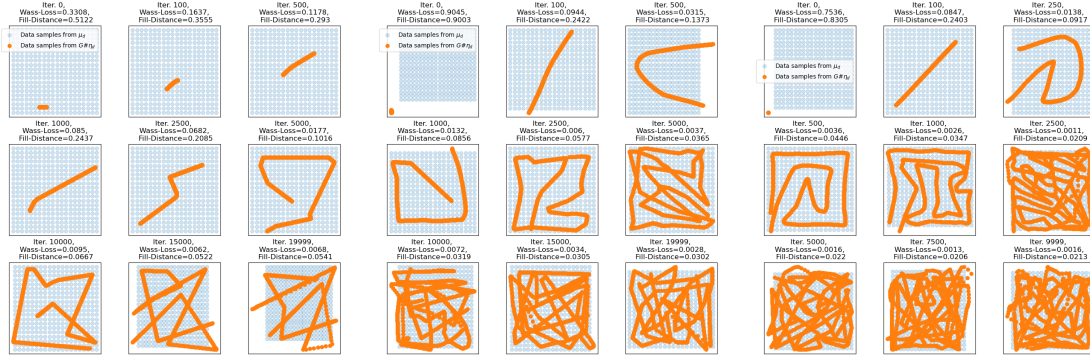
## Appendix C. Additional Experiments

Below we provide additional simulations elucidating the behavior of the neural networks within the same toy scenarios based on the simulations of the main paper to further support our theoretical findings. In Section C.1, we present an additional experiment regarding neural network complexity and approximation error in the context of Simulation 1. In Section C.2, we demonstrate consistent findings with the known theory extending each of the 3 simulations of the main paper to the input  $m = d + 1$  case as proven previously in (Dahal et al., 2022). In each case we find that a smaller or equivalently sized neural network can learn comparatively close approximations to the  $m = d$  cases presented in the main paper. Finally, in Section C.3, we show that using a normally distributed input, rather than uniform, leads again to equally strong approximation performance against a normally distributed target output. Again, complete implementation notebooks generating each of the paper figures can be found at: <https://github.com/hong-niu/dgm24>.

### C.1 Extensions to Simulation 1 (Unit Square), $m = 1, d = 2$

In this section, we provide an additional study of complexity size and approximation accuracy when varying the size of the network. We repeat the same setup as Simulation 1 of the main paper as a representative example, where the input dimension  $m = 1$  and the target data dimension  $d = 2$ , with the following changes. We present the results of three different sized networks a) 2 hidden layers, 10 nodes each, b) 3 hidden layers, 100 nodes each, and c) 5 hidden layers, 200 nodes each, and also significantly increase the training epochs to a maximum of 20,000 iterations for each experiment. In each case, the input distribution is uniform in 1-dimension, and the data distribution is uniform in 2-dimensions. We observe that the approximation improves with increasing network complexity, and that aligned with our theory, a model as small as network a) with two hidden layers is sufficient for fitting the data manifold when the input dimension is equal to the data, but such network is insufficient for approximating the manifold to the same degree when the input dimension is underestimated.





a) 2 layers, 10 nodes each    b) 3 layers, 100 nodes each    c) 5 layers, 200 nodes each

Figure 9: Training results of 3 different network sizes on mapping the 1-dimensional uniform input to a higher 2-dimensional uniform target data manifold. Here the orange curve is generated by the neural network and is “filling in” the data manifold (blue). Networks a) and b) were trained to 20,000 iterations, while network c) only required 10,000 iterations to converge. We observe that as the network complexity increases, the model is able to better fit the target distribution with both lower Wasserstein loss and empirical fill distance against the true data. Note that the smallest network a) is sufficient when the input and target data dimension are equal, as was in Section 4.1, but is unable to achieve the same degree of approximation after 20,000 training iterations now that the data dimension has been underestimated, aligning with expectations from our presented theory.

Again consistent with our theory, we find that a much larger network c) with 5 hidden layers of 200 nodes each is required to achieve a closer Wasserstein loss and fill distance to the results of Section 4.1 now that the data dimension has been strictly underestimated, with a final Wasserstein loss of 0.0016 and fill distance of 0.0213. Furthermore, network c) improves on the medium sized network in network b) which achieves a final loss of 0.0028 and fill distance of 0.0302. We also note that the largest network c) also consistently achieves lower losses throughout training than the middle sized network b), despite being trained for 10,000 fewer iterations.

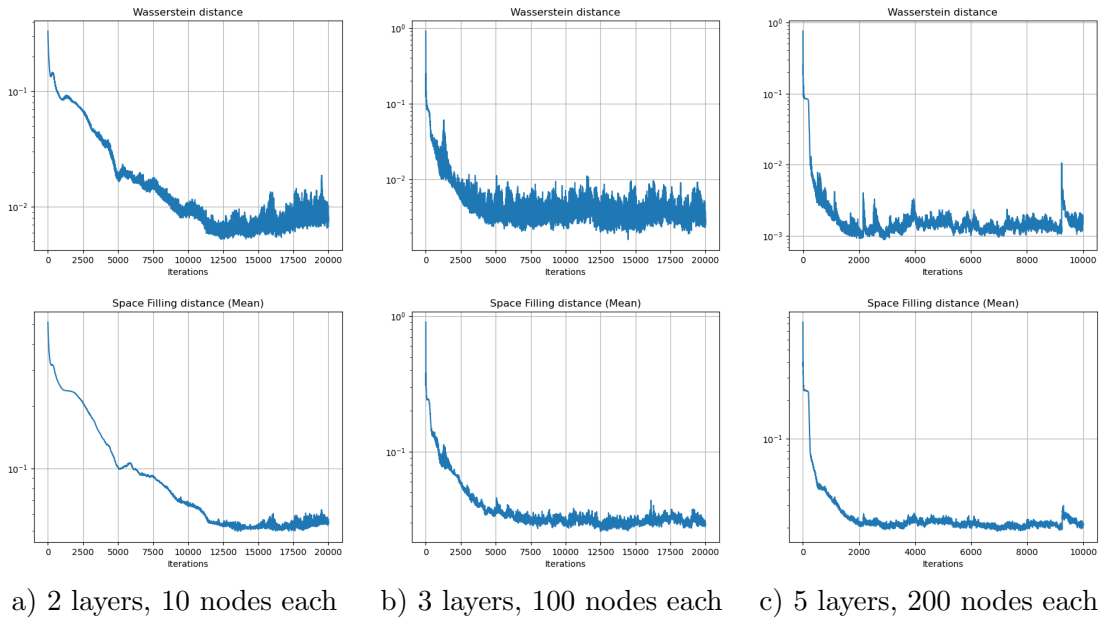


Figure 10: [Top Row] Wasserstein loss per training iteration for each of the three networks presented in Section C.1. [Bottom Row] Fill distance between generated curve and data per iteration for each of the three networks presented in Section C.1.

## C.2 Extensions to the Case for $m = d + 1$

Next we show that in each of the cases of Simulation 1, 2 and 3 from the main paper, when the input dimension  $m$  is 1 greater than the dimension of the target manifold  $d$ , a neural network can be trained to approximate the manifold as well as the  $m = d$  case as is consistent with previous theory in (Dahal et al., 2022). We note that we were able to find such a neural network using fewer or equivalently many neurons than that of the  $m = d$  case in each of the three cases.

In Figure 11, we show that a smaller neural network than the 2-D to 2-D analogue of Simulation 1 is able to learn a mapping between a 3-D uniform input distribution to a smaller 2-D target distribution. We note that in this case, the number of training samples used to train the network (900) is identical to the 2-D to 2-D case in Simulation 1 of the main paper (900), however the number of visualization points for the 3-D uniform input has been increased to maintain the density of the input grid as the 2-D case.

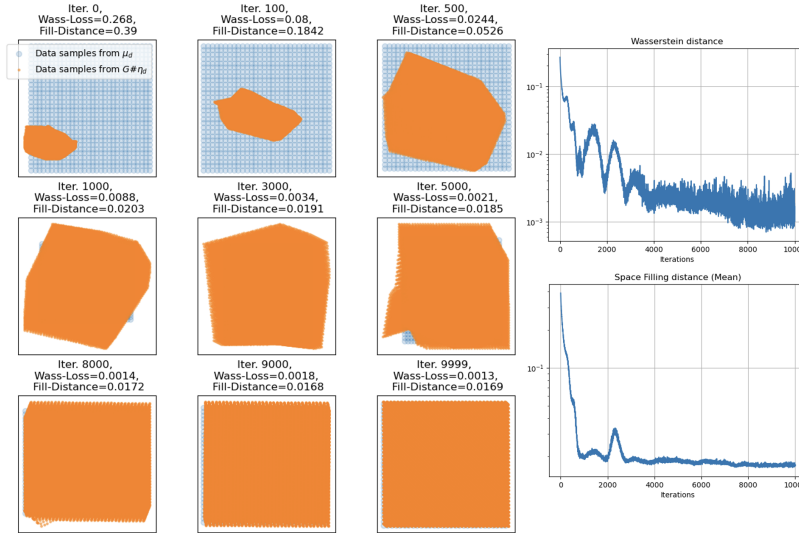


Figure 11: Training results of a 1 hidden layer, 10 node, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a uniform distribution on the cube  $[0, 1]^3$  to a uniform distribution on the unit square  $[0, 1]^2$ . It can be seen that as the loss decreases, the fitted curve fills more of the square as expected, and as well as the 2-D to 2-D analogue in Simulation 1.

Next, we repeat for  $m = d + 1$  input dimension analogues of Simulation 2 in Figure 12, and for Simulation 3 in Figure 13. Again, the number of training samples used in both of these simulations is equal to the respective  $m = d$  cases of the main paper, while the neural networks used in the  $m = d + 1$  cases actually use fewer total neurons to learn similarly accurate approximations.

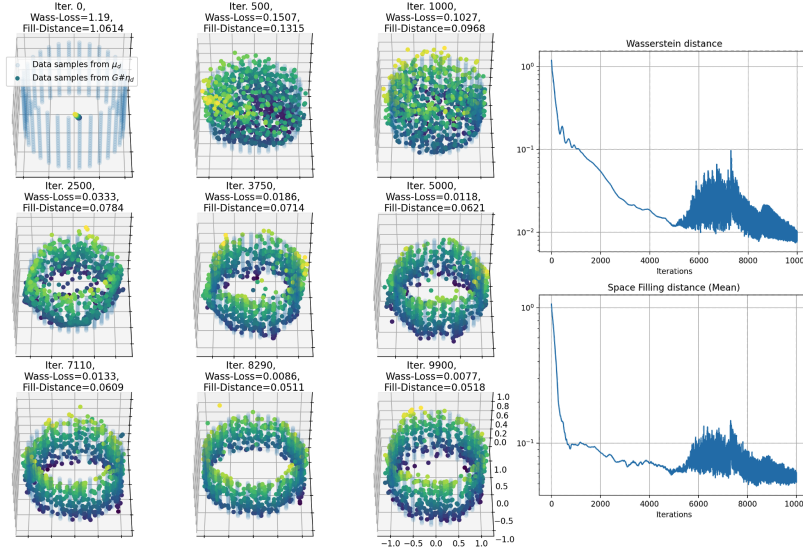


Figure 12: Training results of a 3 hidden layer, 20 node each, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a uniform distribution on the unit cube  $[0, 1]^3$  to a 2-D uniform distribution on a cylinder. It can be seen that as the loss decreases, the fitted curve fills more of the cylindrical surface as expected, and as well as the 2-D to 2-D analogue in Simulation 2.

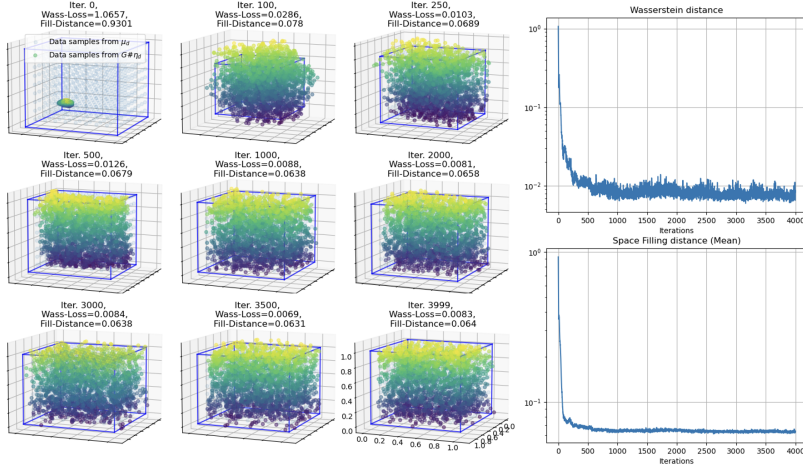


Figure 13: Training results of a 2 hidden layer, 128 node each, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a uniform distribution on  $[0, 1]^4$  to a uniform distribution on the unit cube  $[0, 1]^3$ . It can be seen that as the loss decreases, the fitted curve fills more of the cube as expected, and as well as the 3-D to 3-D analogue in Simulation 3.

### C.3 Extensions to Non-Uniform Distributions

Here, we briefly consider the case when the distributions are no longer uniform. Here we present two simulations where the target manifold is a 2-D normally distributed sample, and show that keeping the neural network architecture identical and only varying using either a 1-D uniform or 1-D normal distribution as the input results in equally accurate approximations of the target manifold in terms of Wasserstein distance and fill distance as is consistent with our theoretical assumptions.

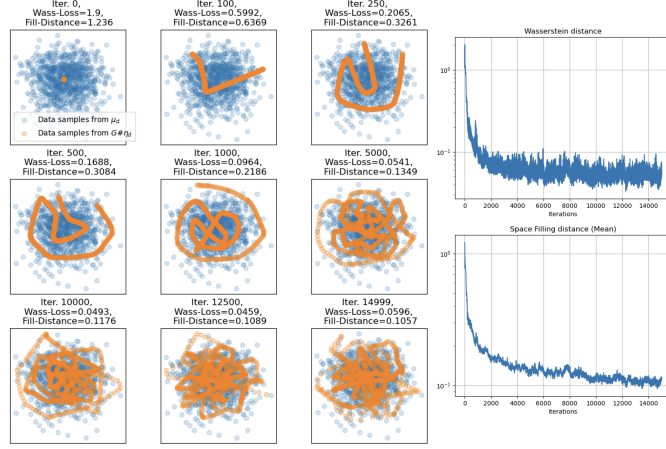


Figure 14: Training results of a 5 hidden layer, 200 node each, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a uniform distribution on the interval  $[0, 1]$  to a 2-D normally distributed target sample.

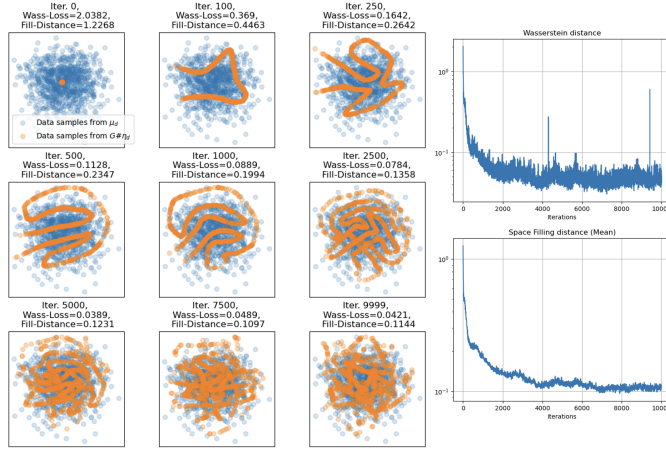


Figure 15: Training results of a 5 hidden layer, 200 node each, fully connected network for 10,000 iterations. Training trajectory (left), Wasserstein loss (top right), and fill distance (bottom right) of mapping a 1-D normally distributed input to a 2-D normally distributed target sample.

## References

- Luca Ambrogioni, Umut Güçlü, Yağmur Güçlütürk, Max Hinne, Marcel A van Gerven, and Eric Maris. Wasserstein variational inference. *Advances in Neural Information Processing Systems*, 31, 2018.
- Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.
- Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017.
- Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. CVAE-GAN: fine-grained image generation through asymmetric training. In *Proceedings of the IEEE international conference on computer vision*, pages 2745–2754, 2017.
- Adam Block, Zeyu Jia, Yury Polyanskiy, and Alexander Rakhlin. Intrinsic dimension estimation using Wasserstein distance. *The Journal of Machine Learning Research*, 23(1):14124–14160, 2022.
- William M Boothby. *An introduction to differentiable manifolds and Riemannian geometry*. Academic press, 1986.
- Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.
- Bradley CA Brown, Anthony L Caterini, Brendan Leigh Ross, Jesse C Cresswell, and Gabriel Loaiza-Ganem. The union of manifolds hypothesis. In *NeurIPS 2022 Workshop on Symmetry and Geometry in Neural Representations*, 2022.
- Luis A Caffarelli. Interior  $W^{2,p}$  estimates for solutions of the Monge-Ampère equation. *Annals of Mathematics*, pages 135–150, 1990a.
- Luis A Caffarelli. A localization property of viscosity solutions to the Monge-Ampère equation and their strict convexity. *Annals of mathematics*, 131(1):129–134, 1990b.
- Luis A Caffarelli. Some regularity properties of solutions of Monge Ampère equation. *Communications on pure and applied mathematics*, 44(8-9):965–969, 1991.
- Luis A Caffarelli. Boundary regularity of maps with convex potentials. *Communications on pure and applied mathematics*, 45(9):1141–1151, 1992a.
- Luis A Caffarelli. The regularity of mappings with a convex potential. *Journal of the American Mathematical Society*, 5(1):99–104, 1992b.
- Luis A Caffarelli. Boundary regularity of maps with convex potentials–ii. *Annals of mathematics*, 144(3):453–496, 1996.

- Minwoo Chae, Dongha Kim, Yongdai Kim, and Lizhen Lin. A likelihood approach to nonparametric estimation of a singular distribution using deep generative models. *Journal of Machine Learning Research*, 24(77):1–42, 2023.
- Biraj Dahal, Alexander Havrilla, Minshuo Chen, Tuo Zhao, and Wenjing Liao. On deep generative models for approximation and estimation of distributions on manifolds. *Advances in Neural Information Processing Systems*, 35:10615–10628, 2022.
- Manfredo Perdigao Do Carmo and J Flaherty Francis. *Riemannian geometry*, volume 6. Springer, 1992.
- Jean Duchon. Splines minimizing rotation-invariant semi-norms in Sobolev spaces. In *Constructive Theory of Functions of Several Variables: Proceedings of a Conference Held at Oberwolfach April 25–May 1, 1976*, pages 85–100. Springer, 1977.
- Ronen Eldan and Ohad Shamir. The power of depth for feedforward neural networks. In *Conference on learning theory*, pages 907–940. PMLR, 2016.
- Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *Journal of the American Mathematical Society*, 29(4):983–1049, 2016.
- Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boissunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. Pot: Python optimal transport. *The Journal of Machine Learning Research*, 22(1):3571–3578, 2021.
- Kunihiko Fukushima. Visual feature extraction by a multilayered network of analog threshold elements. *IEEE Transactions on Systems Science and Cybernetics*, 5(4):322–333, 1969.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- Alfred Gray. *Tubes*, volume 221. Springer Science & Business Media, 2003.
- Changhee Han, Hideaki Hayashi, Leonardo Rundo, Ryosuke Araki, Wataru Shimoda, Shinichi Muramatsu, Yujiro Furukawa, Giancarlo Mauri, and Hideki Nakayama. GAN-based synthetic brain MR image generation. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 734–738. IEEE, 2018.
- Kurt Hornik. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.
- Christian Horvat and Jean-Pascal Pfister. Intrinsic dimensionality estimation using normalizing flows. *Advances in Neural Information Processing Systems*, 35:12225–12236, 2022.
- Jian Huang, Yuling Jiao, Zhen Li, Shiao Liu, Yang Wang, and Yunfei Yang. An error analysis of generative adversarial networks for learning distributions. *The Journal of Machine Learning Research*, 23(1):5047–5089, 2022.

- Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.
- Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.
- Elizaveta Levina and Peter Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems*, 17, 2004.
- Giulia Luise, Massimiliano Pontil, and Carlo Ciliberto. Generalization properties of optimal transport GANs with latent distribution learning. *arXiv preprint arXiv:2007.14641*, 2020.
- Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.
- Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier GANs. In *International conference on machine learning*, pages 2642–2651. PMLR, 2017.
- Ilsang Ohn and Yongdai Kim. Smooth function approximation by deep neural networks with general activation functions. *Entropy*, 21(7):627, 2019.
- Kripasindhu Sarkar, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. HumanGAN: A generative model of human images. In *2021 International Conference on 3D Vision (3DV)*, pages 258–267. IEEE, 2021.
- Nicolas Schreuder, Victor-Emmanuel Brunel, and Arnak Dalalyan. Statistical guarantees for generative models without domination. In *Algorithmic Learning Theory*, pages 1051–1071. PMLR, 2021.
- Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. *Advances in neural information processing systems*, 29, 2016.
- Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017.
- Ngoc-Trung Tran, Viet-Hung Tran, Ngoc-Bao Nguyen, Trung-Kien Nguyen, and Ngai-Man Cheung. On data augmentation for GAN training. *IEEE Transactions on Image Processing*, 30:1882–1897, 2021.
- Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. *Advances in neural information processing systems*, 33:19667–19679, 2020.
- Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.



- Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8798–8807, 2018.
- Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. 2019.
- Holger Wendland. Scattered data approximation: Conditionally positive definite functions. 2004.
- Hermann Weyl. On the volume of tubes. *American Journal of Mathematics*, 61(2):461–472, 1939.
- Xuan Xia, Xizhou Pan, Nan Li, Xing He, Lin Ma, Xiaoguang Zhang, and Ning Ding. GAN-based anomaly detection: A review. *Neurocomputing*, 493:497–535, 2022.
- Yunfei Yang, Zhen Li, and Yang Wang. On the capacity of deep generative networks for approximating distributions. *Neural networks*, 145:144–154, 2022.
- Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019.
- Yijia Zheng, Tong He, Yixuan Qiu, and David Wipf. Learning manifold dimensions with conditional variational autoencoders. 2022.