Can Push-forward Generative Models Fit Multimodal Distributions?

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many generative models synthesize data by transforming a standard Gaussian random variable using a deterministic neural network. Among these models are the Variational Autoencoders and the Generative Adversarial Networks. In this work, we call them "push-forward" models and study their expressivity. We formally demonstrate that the Lipschitz constant of these generative networks has to be large in order to fit multimodal distributions. More precisely, we show that the total variation distance and the Kullback-Leibler divergence between the generated and the data distribution are bounded from below by a constant depending on the mode separation and the Lipschitz constant. Since constraining the Lipschitz constants of neural networks is a common way to stabilize generative models, there is a provable trade-off between the ability of push-forward models to approximate multimodal distributions and the stability of their training. We validate our findings on one-dimensional and image datasets and empirically show that models which do not optimize directly the push-forward map itself, such as diffusion models, do not suffer of such limitations.

1 Introduction

Generative modeling has become over the last years one of the most popular research topics in computer vision. From a mathematical perspective, the goal of generative modeling can be seen as predicting new synthetic samples from an unknown probability distribution ν on $\mathbb{R}^d$ given the information of m *true* samples x_i drawn from ν (the data distribution). A general approach to solve this problem is to define a parametric family of probability distributions $(\nu_\theta)_{\theta\in\Theta}$ and solve the problem

$$\min_{\theta\in\Theta} d(\frac{1}{m}\sum_i \delta_{x_i}, \nu_\theta) \ ,$$

where d is a similarity measure between probability distributions (d can either be a divergence or a distance) and δ_x is the delta distribution in x. Beside their direct application (Sandfort et al., 2019; Antoniou et al., 2018), generative models have been used in numerous applications in various machine learning subfields, such as solving inverse problems (Ravuri et al., 2021; Ledig et al., 2017) or machine translation (Isola et al., 2017; Yang et al., 2018). However, most generative modeling methods still lack theoretical understanding and it remains often unclear whether the method approaches correctly the probability distribution ν or only generates samples that appear to have been drawn from ν without fully recovering the underlying structure of the distribution. In this work, we focus on the particular class of *push-forward generative models*. Those models have in common that for any $\theta \in \Theta$, the parametric distribution ν_θ approaching ν is of the form

$$\nu_\theta = g_{\theta\#}\mu_p \ ,$$

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

where $\mu_p = \mathrm{N}(0, \mathrm{Id}_p)$ is the Gaussian standard distribution in dimension p, $\#$ is the push-forward operator [1], and $g_\theta : \mathbb{R}^p \to \mathbb{R}^d$ is a deterministic neural network of parameter θ. This class includes two of the most popular generative models: the *Variational Auto-Encoders* (VAEs) (Kingma and Welling, 2014) and the *Generative Adversarial Networks* (GANs) (Goodfellow et al., 2014).

Deep neural networks are most of the time Lipschitz mappings by design, since their activation functions are generally Lipschitz. In the literature, constraining the Lipschitz constant of a neural network is widely used as a way to increase its robustness (Virmaux and Scaman, 2018; Fazlyab et al., 2019), in particular to adversarial attacks (Goodfellow et al., 2015). Common approaches to bound Lipschitz constants of neural networks are spectral normalization (Miyato et al., 2018), adding a gradient penalization in the loss (Gulrajani et al., 2017; Mohajerin Esfahani and Kuhn, 2018), or Jacobian regularization (Pennington et al., 2017). These approaches have been widely used to stabilize the training of GANs, where Lipschitz constraints have been first imposed on discrimators (Arjovsky et al., 2017; Kodali et al., 2017; Fedus et al., 2018), while recent state-of-the-art architectures such as BigGAN (Brock et al., 2018), SAGAN (Zhang et al., 2019) or StyleGAN2 (Karras et al., 2020) also impose similar constraints on the generators through spectral normalization (Brock et al., 2018; Zhang et al., 2019), or Jacobian regularization (Karras et al., 2020). In contrast to GANs, the recent study of Kumar and Poole (2020) shows that the decoder Jacobian in VAEs is implicitely regularized, which limits naturally its Lipschitz constant.

Recently, Dhariwal and Nichol (2021) trained an unconditional *Score-based Generative Model* (SGM) (Song and Ermon, 2019; Ho et al., 2020) on ImageNet (Russakovsky et al., 2015) and achieved state-of-the-art generation. To the best of our knowledge, there is no push-forward generative models capable of reaching this kind of performance on such a complex dataset without explicitly adding any conditional label information in the model, see (Brock et al., 2018) for instance. SGMs (also known as *diffusion models*) proceed as follows: first, noise is progressively added to the data distribution until we reach a standard Gaussian distribution. Then this forward dynamics is reversed leveraging recent advances in deep learning and tools from score-matching (Hyvärinen, 2005; Vincent, 2011). We refer to Song et al. (2020) for an introduction on SGMs. In those models, the parametric generated distribution is also of the form $\nu_\theta = g_{\theta\#}\mu_p$, where g_θ is the whole reverse diffusion dynamic (which can be seen as a composition of deterministic Lipschitz mappings) and $p = d(N+1)$ with N being the total number of steps in the dynamic. However, those models are not push-forward generative models in the strict sense of the term, since the push-forward mapping is not a simple neural network anymore. An important difference is that optimization is not directly performed on the push-forward mapping itself but on an auxiliary function (the score). We therefore categorize them as *indirect push-forward generative models* in this work.

Contributions of the paper. In this paper, we study the expressivity of direct and indirect push-forward generative models in relation to the Lipschitz constant of the push-forward mapping they learn. Our contribution is twofold: first, in Section 3, for a Lipschitz function g and a given multimodal probability distribution ν, we formally demonstrate that the Lipschitz constant of g must necessarily be large in order that $g_\#\mu_p$ approximate ν correctly, as it has been already intuitively observed in the literature (Lu et al., 2020; Luise et al., 2020; Khayatkhoei et al., 2018). As a direct consequence, we exhibit lower bounds on $d(g_\#\mu_p, \nu)$, where d is the total variation distance or the Kullback-Leibler divergence, with an explicit dependence on the Lipschitz constant $\mathrm{Lip}(g)$ of g, which highlights there is a fundamental trade-off for (direct) push-forward generative models between expressivity and stability of training. Second, in Section 4 we show empirically that SGMs seem to be able to circumvent the previously mentioned limitation of push-forward generative models by not directly optimizing the push-forward map and are therefore able to generate multimodal distributions while keeping their training stable. We also offer comparison with VAEs and GANs and show that our theoretical results can explains some of the already observed behaviors of these latter models when trained on multimodal distributions, such as interpolation between modes for VAEs and *mode collapse* (Arjovsky and Bottou, 2017; Metz et al., 2017) for GANs. All the proofs are postponed to the appendix.

[1] If μ is a measure on $\mathbb{R}^p$ and f is a mapping from $\mathbb{R}^p$ to $\mathbb{R}^d$, the push-forward measure $f_\#\mu$ is the measure on $\mathbb{R}^d$ such that for all Borel set A of $\mathbb{R}^p$, $f_\#\mu(\mathsf{A}) = \mu(f^{-1}(\mathsf{A}))$.

2 Related Works

Assessing the efficiency of push-forward models is a recurrent and important question in the literature. The authors of Sajjadi et al. (2018) and Kynkäänniemi et al. (2019) propose Precison and Recall metrics to assess GANs, aiming to measure simultaneously the mode collapse and the proportion of off-manifold generated samples. Using similar metrics, the authors of Tanielian et al. (2020) show that the precision (the proportion of generated samples which could have been generated by the target distribution) of standard GANs is always upper-bounded. To overcome this limitation, they simply propose to reject samples associated with large values of the generator Jacobian. The intuition behind this idea is that those samples lie in regions of the space where the discontinuous optimal generator would "jump" between modes and so are off-manifold.

Another line of research focuses on the fact that the model has access to only the empirical distribution $\nu_n = \frac{1}{n}\sum_i \delta_{x_i}$ and not to the true target distribution. For instance Nagarajan et al. (2018) study to what extend GANs only memorize the data. Gulrajani et al. (2018) highlight the fact that common GAN benchmarks prefer training set memorization to a model which imperfectly fits the true distribution but covers more of its support. Related to this, Stéphanovitch et al. (2022) study specifically the Wasserstein GAN case, where the latent distribution is uniform and construct an optimal generator which minimizes the Wasserstein distance of order 1 between the push-forward measure and the empirical distribution, thus deriving a lower bound on the 1-Wasserstein distance. In the same paper, and more related to our work, the authors study the asymptotic case of an infinite number of data and show that most of the time the minimal 1-Wasserstein distance between the push-forward measure and the target distribution remains stricly positive.

3 Push-forward measure and Lipschitz mappings

In this section, we study the properties of the push-forward measure $g_{\#}\mu_p$ when $\mu_p = \mathrm{N}(0, \mathrm{Id}_p)$ is the standard Gaussian distribution in dimension p and g is a Lipschitz mapping. First, for any probability measure γ on $\mathbb{R}^d$ and any Borel set A of $\mathbb{R}^d$, we define the γ-surface area of A by

$$\gamma^+(\partial\mathsf{A}) = \liminf_{\varepsilon\to 0^+}(\gamma(\mathsf{A}_\varepsilon) - \gamma(\mathsf{A}))/\varepsilon \ ,$$

where $\mathsf{A}_\varepsilon = \{x \in \mathbb{R}^d \ : \ \text{there exists } a \in \mathsf{A}, \ \|x - a\| \leq \varepsilon\}$ is the ε-extension of A and $\partial\mathsf{A}$ is the boundary of A. The γ - surface area can be interpreted as a measure (not in the strict mathematical sense of the term) of the mass of γ on the hypersurface $\partial\mathsf{A}$. Note that the support of γ and A can be sets of intrisic dimension smaller than d, which is most of the time the case when working with real data which are likely to live on low dimensional manifolds (Pope et al., 2020). The main theoretical result of this paper establishes some properties of push-forward measures depending on the regularity of the push-forward mapping.

Theorem 1. *Let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function with Lipschitz constant $\mathrm{Lip}(g)$. Then for any Borel set $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$,*

$$\mathrm{Lip}(g)(g_{\#}\mu_p)^+(\partial\mathsf{A}) \geq \varphi\left(\Phi^{-1}(g_{\#}\mu_p(\mathsf{A}))\right) \ , \tag{1}$$

where $\varphi(x) = (2\pi)^{-1/2}\exp[-x^2/2]$ and $\Phi(x) = \int_{-\infty}^{x}\varphi(t)\mathrm{d}t$. In addition, we have that for any $r \geq 0$

$$g_{\#}\mu_p(\mathsf{A}_r) \geq \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_{\#}\mu_p(\mathsf{A}))\right) \ . \tag{2}$$

Sketch of proof. The proof of this result consists in establishing lower-bounds on $(g_{\#}\mu_p)^+(\partial\mathsf{A})$ and $g_{\#}\mu_p(\mathsf{A}_r)$ which can be expressed as Gaussian integrals. We conclude upon combining this result and the Gaussian isoperimetric inequality, see Sudakov and Tsirelson (1978) $\mu_p^+(\partial\mathsf{A}) \geq \varphi(\Phi^{-1}(\mu_p(\mathsf{A})))$. □

Note that (2) implies (1) upon remarking that (2) is an equality for $r = 0$ and letting $r \to 0$. Theorem 1 recovers the Gaussian inequality in the case where g is the identity mapping and extends it to all Lipschitz mappings. As the Gaussian inequality, Theorem 1 is dimension free, in the sense that neither d, nor p, nor the intrisic dimension of $g(\mathbb{R}^p)$ play a role in the lower bounds. In the following section, we are going to use Theorem 1 to (i) give a lower bound on the Lipschitz constant so that push-forward generative models *exactly* match the data distribution, (ii) give a lower bound on the total variation and the Kullback-Leibler divergence between the push-forward and data distributions which depends on the Lipschitz constant of the model.

3.1 Lower bounding the Lipschitz constant of push-forward mappings

Equation (1) implies that the Lipschitz constant of g must necessarily be large for $g_{\#}\mu_p$ to be multimodal. It provides indeed a lower bound on the Lipschitz constant of the mappings g which push μ_p into a given measure ν. In the extreme case where the support of ν is composed of disconnected manifolds, we retrieve that there doesn't exist any Lipschitz mapping which pushes μ_p into ν since it can be found Borel sets A with null ν -surface area but such that the right-hand term of (1) is strictly positive (which occurs when $0 < \nu(\mathsf{A}) < 1$). In the intermediate case where the support of ν is connected but ν is multimodal, the less mass ν has between modes, the larger must be the Lipschitz constant of the mappings which push μ_p into ν. Indeed, if ν has little mass between its modes, one can find sets A with arbitrarly small ν-surface area and such that $0 < \nu(\mathsf{A}) < 1$. As a toy example, we get an explicit bound on the Lipschitz constant of the mappings which push μ_p into a mixture of two isotropic Gaussians.

Corollary 2. *Let $\nu = \lambda \mathrm{N}(m_1, \sigma^2 \mathrm{Id}_d) + (1-\lambda)\mathrm{N}(m_2, \sigma^2 \mathrm{Id}_d)$ with $m_1, m_2 \in \mathbb{R}^d$, $\sigma > 0$ and $\lambda \in (0,1)$. Assume that there exists $g : \mathbb{R}^p \to \mathbb{R}^d$ Lipschitz such that $g_{\#}\mu_p = \nu$. Then*

$$\mathrm{Lip}(g) \geq \sigma \exp\left[\|m_2 - m_1\|^2/(8\sigma^2) - (\Phi^{-1}(\lambda))^2/2\right] .$$

Sketch of proof. The proof of this result consists in applying Inequality (1) of Theorem 1 on the half-space H such that $\partial\mathsf{H}$ is the equidistant line of the means of the two Gaussians, and then to explicit the value of $\nu^+(\partial\mathsf{H})$. □

Note that assuming there exists $g : \mathbb{R}^p \to \mathbb{R}^d$ such that $g_{\#}\mu_p = \nu$ implies $p \geq d$ since ν covers the whole ambient space and so g must be a surjective mapping. This bound is maximal in the balanced case when $\lambda = 1/2$ since $\Phi^{-1}(\lambda) = 0$ in that case. Otherwise, the more unbalanced the modes are, the smaller the bound is since the two terms in the exponential compensate each other more and more. Extending this corollary to mixtures of more than two Gaussians with different covariance matrices is technically difficult but we could expect a similar exponential growth in the square distance between modes since it depends mainly on the order of magnitude of the local minima of the distribution density. As a by product of Theorem 1, we also get the following result which shows that (in the one-dimensional case) the optimal transport map for the ℓ_2 cost minimizes the Lipschitz constant of the push-forward mapping.

Corollary 3. *Let ν be a probability measure on $\mathbb{R}$ with density w.r.t. the Lesbesgue measure and such that $\mathrm{supp}(\nu) = \mathbb{R}$. Assume that there exists $g : \mathbb{R}^p \to \mathbb{R}$ Lipschitz such that $\nu = g_{\#}\mu_p$. Let us denote $T_{\mathrm{OT}} = \Phi_\nu^{-1} \circ \Phi$ the Monge map between μ_1 et ν, where Φ_ν is the cumulative distribution function of ν. Then we have $\mathrm{Lip}(g) \geq \mathrm{Lip}(T_{\mathrm{OT}})$.*

To the best of our knowledge, extending this proposition to the case where $d > 1$ remains an open problem. We show now that Equation (2) of Theorem 1 allows to derive lower bounds on similarity measures between the push-forward measure and the target distribution.

3.2 Lower bounds on similarity measures between probability distributions

Equation (2) provides a bound on the minimal mass the push-forward measure $g_{\#}\mu_p$ can have on a given set when g is fixed with Lipschitz constant $\mathrm{Lip}(g)$. As a consequence, if ν is a distribution such that there exists sets on which ν has less mass than the minimal quantity that $g_{\#}\mu_p$ can reach on those sets given the value of $\mathrm{Lip}(g)$, then $g_{\#}\mu_p$ cannot be equal to ν, implying that most of similarity measures between $g_{\#}\mu_p$ and ν will be automatically strictly positive. In the following, we consider that g and ν are fixed and we derive lower bounds on the total variation distance and the Kullback-Leibler divergence between $g_{\#}\mu_p$ and ν. We recall that the total variation distance between two probability measures on $\mathbb{R}^d$, ν_0, ν_1 is given by

$$d_{\mathrm{TV}}(\nu_0, \nu_1) = \sup\{\nu_0(\mathsf{A}) - \nu_1(\mathsf{A}) \ : \ \mathsf{A} \in \mathcal{B}(\mathbb{R}^d)\} .$$

Similarly, we define the Kullback-Leibler divergence between two probability measures on $\mathbb{R}^d$, ν_0, ν_1, using the Donsker-Varadhan representation (Dupuis and Ellis, 2011, Lemma 1.4.3a):

$$d_{\mathrm{KL}}(\nu_0||\nu_1) = \sup\{\textstyle\int_{\mathbb{R}^d} f(x)\mathrm{d}\nu_0(x) - \log\left(\int_{\mathbb{R}^d} \exp[f(x)]\mathrm{d}\nu_1(x)\right) \ : \ f \in \mathfrak{B}(\mathbb{R}^d, \mathbb{R})\} ,$$

where $\mathfrak{B}(\mathbb{R}^d, \mathbb{R})$ denotes the set of all bounded mappings from $\mathbb{R}^d$ to $\mathbb{R}$. In the following, we will denote for any $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ and $r > 0$,

$$\alpha_g(\mathsf{A}, r) = \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_{\#}\mu_p(\mathsf{A}))\right) \ ,$$
$$\beta_g(\mathsf{A}, r) = \alpha_g(\mathsf{A}, r) - g_{\#}\mu_p(\mathsf{A}) \ ,$$

where $\alpha_g(\mathsf{A}, r)$ and $\beta_g(\mathsf{A}, r)$ are the lower bounds of $g_{\#}\mu_p(\mathsf{A}_r)$ and $g_{\#}\mu_p(\mathsf{A}_r \setminus \mathsf{A})$ provided by Theorem 1. We start by proving lower bounds on the total variation distance.

Theorem 4. *Let ν be a probability measure on $\mathbb{R}^d$ and let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function. Then,*

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \sup\{\alpha_g(\mathsf{A}, r) - \min\{g_{\#}\mu_p(\mathsf{A}), \nu(\mathsf{A})\} - \nu(\mathsf{A}_r \setminus \mathsf{A}) \ : \ \mathsf{A} \in \mathcal{B}(\mathbb{R}^d), r > 0\} \ . \quad (3)$$

Sketch of Proof. The proof of this result consists in bounding from below the total variation distance on one hand by $|g_{\#}\mu_p(\mathsf{A}_r \setminus \mathsf{A}) - \nu(\mathsf{A}_r \setminus \mathsf{A})|$ and on the other hand by $|g_{\#}\mu_p(\mathsf{A}_r) - \nu(\mathsf{A}_r)|$ for a given $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ and a given $r > 0$, and then applying Theorem 1. □

Observe that (3) always holds but the right-hand term may become negative if the Lipschitz constant of g is large enough. The main idea behind this bound is to find a set A and a real $r > 0$ such that ν has a lot of mass on A but almost no mass on $\mathsf{A}_r \setminus \mathsf{A}$. For instance, if ν is a distribution on two disconnected manifolds M_1 and M_2, an optimal choice for A would either be M_1 or M_2 and the optimal r would be the distance between the two manifolds. Using Theorem 4, one can derive smaller but more explicit bounds only depending on ν and the Lipschitz constant of g. As a first example, we derive an explicit lower bound in the case where ν is a bi-modal distribution on two disconnected manifolds.

Corollary 5. *Let ν be a measure on $\mathbb{R}^d$ on two disconnected manifolds M_1 and M_2 such that $\nu(\mathsf{M}_1) = \lambda$ and $\nu(\mathsf{M}_2) = 1 - \lambda$, with $\lambda \in [1/2, 1)$, and let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function. Then,*

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \int_{\Phi^{-1}(\lambda)}^{d(\mathsf{M}_1, \mathsf{M}_2)/2\mathrm{Lip}(g) + \Phi^{-1}(\lambda)} \varphi(t)dt \ ,$$

where $d(\mathsf{M}_1, \mathsf{M}_2) = \inf\{\|m_1 - m_2\| \ : \ m_1 \in \mathsf{M}_1, m_2 \in \mathsf{M}_2\}$.

As a second example, we also get an explicit lower bound in the case where ν is a mixture of two isotropic Gaussians. For simplicity we stick to the balanced case.

Corollary 6. *Let $\nu = (1/2)[\mathrm{N}(m_1, \sigma^2 \mathrm{Id}_d) + \mathrm{N}(m_2, \sigma^2 \mathrm{Id}_d)]$ with $m_1, m_2 \in \mathbb{R}^d$ and $\sigma \geq 0$. Let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function. Then,*

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \int_0^{\|m_2 - m_1\|/4\sigma\mathrm{Lip}(g)} \varphi(t)\mathrm{d}t - (1/2) \int_{\|m_2 - m_1\|(2\sigma - 1)/4\sigma^2}^{\|m_2 - m_1\|(2\sigma + 1)/4\sigma^2} \varphi(t)\mathrm{d}t \ .$$

In both corollaries, the lower bound tends to $1/2$ when the distance between the modes tends to infinity, meaning that $g_{\#}\mu_p$ is far from well approaching ν. Note that the lower bound exhibited in Corollary 5 is always strictly positive regardless of the value of the Lipschitz constant of g. One can also observe that this latter bound is maximal in the balanced case, when $\lambda = 1/2$, since the standard normal distribution concentrates its mass around 0. Finally, we end this section by deriving a similar lower bound on the Kullback-Leibler divergence between $g_{\#}\mu_p$ and ν. We consider the Kullback-Leibler divergence since this is a measure of similarity between measures which is bounded and is very sensitive to the mismatch of supports between the generated and the data distributions.

Theorem 7. *Let ν be a probability measure on $\mathbb{R}^d$ and let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function. Then,*

$$d_{\mathrm{KL}}(g_{\#}\mu_p || \nu) \geq \sup\{\beta_g(\mathsf{A}, r) \log\left(\frac{\beta_g(\mathsf{A}, r)}{\nu(\mathsf{A}_r \setminus \mathsf{A})}\right) + (1 - \beta_g(\mathsf{A}, r)) \log\left(\frac{1 - \beta_g(\mathsf{A}, r)}{1 - \nu(\mathsf{A}_r \setminus \mathsf{A})}\right) \ : \ \mathsf{A} \in \mathcal{B}(\mathbb{R}^d), r > 0\} \ .$$

Sketch of Proof. The proof of this result consists in setting $f = \zeta \chi_{\mathsf{A}_r \setminus \mathsf{A}}$ with $\zeta > 0$ and where χ_{A} is the characteristic function of the set A in order to get a lower bound of the Kullback-Leibler divergence depending on the probability $g_{\#}\mu_p(\mathsf{A}_r \setminus \mathsf{A})$ for a given A, a given r and a given ζ. Then we apply Theorem 1 and we derive the optimal value of ζ. □

As above, this bound always holds but the right-hand term becomes negative if $\mathrm{Lip}(g)$ is large enough. As for Theorem 4, the main idea is to find a set A and a real r such that ν has a lot of mass on A, but ν has almost no mass on $\mathsf{A}_r \setminus \mathsf{A}$. Observe that if $\nu(\mathsf{A}_r \setminus \mathsf{A})$ tends to 0, the left-hand term of the bound tends to infinity. This is coherent with the behavior of the Kullback-Leibler divergence. To conclude this section, we highlight the fact that, if our results are dimension free in theory, the dimension is likely to be hidden in the distances between modes and the Lipschitz constant of g when working with real datasets.

4 Experiments

In what follows, we illustrate the pratical implications of our results by training GANs, VAEs and SGMs on simple bi-modal distribution. More precisely, we show on one hand that generating multimodal distributions with GANs and VAEs is hard since it necessarily involves generative networks with large Lipschitz constants. On the other hand, we show that SGMs seem to be able to implicitly learn push-forward mappings with large Lipschitz constants and thus do not suffer of the same limitation. First, we focus on the univariate case where we can easily assess the Lipschitz constants of the networks. Then we illustrate our results in higher dimensions by training the three models on datasets derived from MNIST (LeCun et al., 1998). In all our experiments, we use the same architecture for the VAE decoder and the GAN generator in order to offer rigorous comparisons of the different models. For score-based modeling, we use architectures with similar number of learnable parameters. All details on the experiments and architecture of the networks can be found in Appendix S4.

4.1 Univariate case

First, we train a VAE and a GAN with one-dimensional latent spaces on 50000 independent samples drawn from a balanced mixture of two univariate Gaussians $\nu = (1/2)[\mathrm{N}(-m, 1) + \mathrm{N}(m, 1)]$ for different values of $m > 0$. We also train a SGM on the same samples.

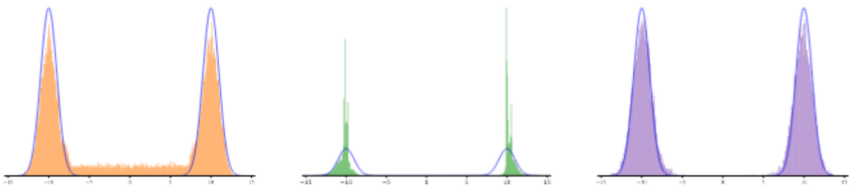

Figure 1: Histograms of distributions generated with VAE (left, in orange), GAN (middle, in green), and with SGM (right, in purple) for $m = 10$. The data distribution densities are plotted in blue.

Histograms of generated distributions. Figure 1 shows histograms of generated distributions for $m = 10$ with the three different models. VAE models seem to generate Gaussians modes but interpolate significantly between them, while GANs do not interpolate but fail to retrieve the structure of the target distribution and forgets parts of its support, which is known as *mode collapse* and is a common pitfall of such models (Arjovsky and Bottou, 2017; Metz et al., 2017). On the same task, SGMs do not suffer from such shortcomings. In the following section, we will link the interpolation/mode-collapsing properties of these models with their Lipschitz constants.

Lipschitz constant and mass between modes. In Figure 2 (right), we observe that the GAN generator reaches much larger Lipschitz constants than the VAE decoder. This explains the difference of behaviors between GAN and VAE observed in Figure 1, as the learned mapping is not stiff enough to concentrate the push forward measure on the two modes in the case of VAE. One possible explanation for the interpolating behavior of the VAE is to note that the Euclidean norm of the Jacobian of the VAE decoder is implicitely regularized during training, as it has been demonstrated in Kumar and Poole (2020). Both GAN and VAE saturate the constraint on $g_{\#}\mu_p([-m/2, m/2])$ provided by Theorem 1, meaning that the generative networks minimize the amount of mass between modes as much as their Lipschitz constants allow it. Finally, we can observe that the score network is able to keep a relatively small Lipschitz constant compared to the GAN, while managing to interpolate

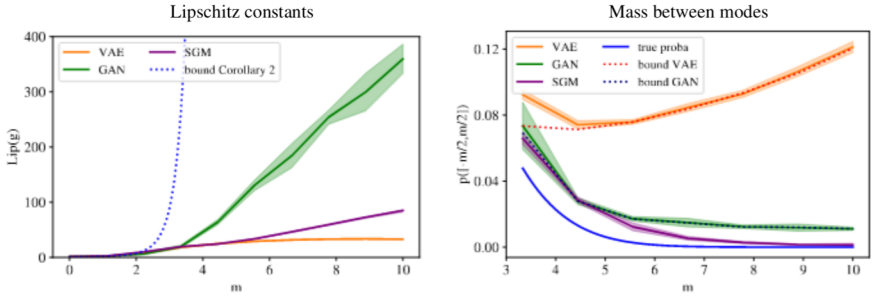

Figure 2: Left: evolution of the Lipschitz constants of the three different generative models trained on 50000 samples of $(1/2)[\mathrm{N}(-m,1)+\mathrm{N}(m,1)]$, in function of m. Right: evolution of the proportion of samples generated by the three models on the interval $[-m/2, m/2]$. We also show on this graph the lower bounds predicted by Theorem 1 for the VAE and the GAN, as well as the true probability $\nu([-m/2, m/2])$. Experiments are averaged over 10 runs and the colored bands correspond to +/- the standard deviation.

less than the latter. A probable explanation for this follows from the fact that the score network is used multiple time during inference. Hence, the Lipschitz constant of the whole generation dynamic is likely to be much larger than the Lipschitz constant of the neural network itself, and so the model is able to push-forward a Gaussian distribution into multimodal distributions keeping a relatively small Lipschitz constant of the score network. Finally, in Figure 2 (left), we observe that when m increases, the Lipschitz constant of the VAE decoder and the GAN generator becomes rapidly much smaller than the value of the lower bound provided by Corollary 2. This means that for m large enough it is not possible to close the gap between the data distribution and the push-forward distribution. We highlight that our results do not cover SGMs since in this setting the network is applied multiple times.

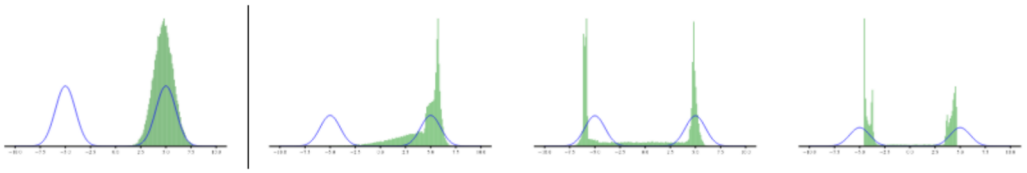

Figure 3: histograms of distributions generated with GANs with spectral normalization applied on generator (left), and with gradient penalty (right) for $\mathrm{Lip}(g) \approx L = 5$, $\mathrm{Lip}(g) \approx L = 15$ and $\mathrm{Lip}(g) \approx L = 25$. The data distribution densities are plotted in blue.

Stability of GAN and mode collapse. Odena et al. (2018) suggested that the magnitude of the norm of the generator jacobian may be causally related to unstability and mode collapse. This is why many state-of-the-art GANs apply spectral normalization (Miyato et al., 2018) on their generators. In Figure 3 (left), we show that this technique cannot be used when training GANs on multimodal distributions: since spectral normalization constraints the Lipschitz constant of the generator to be smaller than 1, the GAN is trained towards concentrating in one of the modes of the distribution over interpolating massively between them. This has been referred to as *mode dropping* by Khayatkhoei et al. (2018). To complete this analysis, we also train the GAN adding an additional gradient penalty term $10/L^2 \max_{z\sim\mathrm{N}(0,\mathrm{Id}_p)}(\|\nabla_z g_\theta(z)\|_2^2 - L)^2$, in the generator loss function, similarly to WP-GAN (Gulrajani et al., 2017), where L is an hyperparameter corresponding to the targeted Lipschitz constant. As expected, we can observe in Figure 3 (right), that when $\mathrm{Lip}(g)$ increases, the GAN begin to generate both modes but becomes also more and more prone to mode collapse. This illustrates the fundamental trade-off between expressivity and robustness in push-forward generative models.

Influences of model size and time of training. Finally, we study the effect of the architecture and the training time on the value of the Lipschitz constant of the VAE decoder and the GAN generator in Figure 4. In the VAE setting, the Lipschitz constant increases linearly with the depth of the decoder. This is not the case in the GAN setting, where increasing the size of the model seems to dramaticaly affect its stability. For both models, the Lipschitz constants of the generative network grow with

the number of epochs. Yet this growth seems to be logarithmic for the VAE and the GAN seems to becomes more unstable as the number of epochs increases.

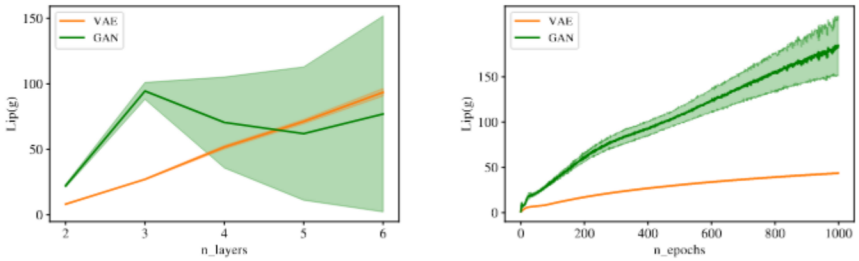

Figure 4: evolution of the Lipschitz constant of the generative network with respect to its number of layers (left) and of the Lipschitz constant in function of the numbers of epochs (right). The experiments are averaged over 10 runs and the colored bands correspond to +/- the standard deviation.

4.2 Experiments on MNIST

We train a VAE, a GAN and a SGM on two datasets derived from MNIST (LeCun et al., 1998): first, two images of two different digits (3 and 7) are chosen and 10000 noisy versions of theses images are drawn with a noise amount of $\sigma = 0.15$, forming a dataset of $n = 20002$ independent samples drawn from a balanced mixture of two Gaussian distributions in dimension $784 = 28 \times 28$. Second, we train the models on the subset of all 3 and 7 of MNIST. We emphasize that our goal is not reach state-of-the-art performance on this problem but rather to illustrate our theoretical results in a moderate dimensional setting.

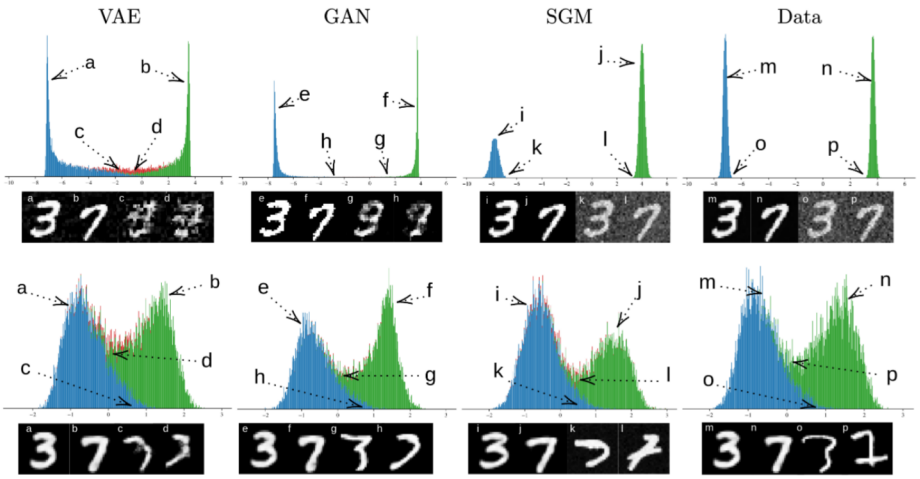

Figure 5: mixture of Gaussians (top): histograms of projections on the line passing through the mean of each Gaussian. Subset of MNIST (bottom): histograms of projections on the line passing through the barycenters of all the 3 and 7 in the deep Wasserstein embedding space. Bins of data are colored in blue if they are classified as 3, in green if classified as 7, and in red if classified as another digit.

Mixture of Gaussians. For this experiment, we set the dimension of the latent space in the GAN and the VAE to $784 = 28 \times 28$ since it is the intrisic dimension of the support of the data distribution. In order to visualize the interpolation between modes, we project the data on the line passing through the mean of each Gaussian, i.e. the two original clean images, and we plot histograms of the one-dimensional projections. In order to understand which bins of data in the histograms correspond to which digit, we train a classifier and we assign a color in function of which digit the data have been classified as. Results can be found in Figure 5 top. Moreover, they both fail to generate noisy versions of the images. As in the univariate case, the score-based model is able to not interpolate

between modes and seem to retrieve the Gaussian structure of the modes. This highlights that while push-forward models fail at representing multimodal distributions, considering stacked models with stochastic input at each step (as in SGM) might help to close the gap between the generated and the data distributions. However SGM does not manage to retrieve the right modes proportions. This is a well-known shortcoming of score-based models which has been studied in (Wenliang and Kanagawa, 2020). We observed that increasing the size of the score network seems to be a way to reduce this issue (see Appendix S5).

Subset of MNIST. Finally we train the three different models on the subset of MNIST composed of all 3 and 7 (no Gaussian noise was added). We choose a latent dimension of 20 for the VAE and the GAN. Since the Euclidean distance is not a meaningful metric to compare the different digits of MNIST, we use the deep Wasserstein embedding proposed by Courty et al. (2018): an autoencoder is learned in a supervised fashion such that the Euclidean distance in the latent space approximates the Wasserstein distance between pairs of images of MNIST. In the learned Wasserstein space, we project data on the line passing through the Euclidean barycenters of all 3 and 7 and plot histograms of projections, using the same classifier as before. Results can be found in Figure 5 (bottom). Note that the distribution does not exhibit strong multimodality features contrary to the mixture of Gaussians settings, see Figure 5. As before, the VAE interpolate between modes, the GAN manages to not interpolate but generate a narrower histogram, and the score-based model does not interpolate and seems to recover the structure of the distribution, but doesn't retrieve the right modes proportions. However, we emphasize that all these models seem to perform better than on the previous dataset. A possible explanation of this is that the modes are less separated than in the Gaussian mixtures and therefore the model is easier to train.

5 Discussion

In this work, given a Lipschitz mapping g and a measure ν, we derived lower bounds on the total variation distance and the Kullback-Leibler divergence between the push-forward measure $g_{\#}\mu_p$ and ν depending on the Lipschitz constant of the mapping g. These bounds indicate how the mass between the modes of the push-forward measure depends on the Lipschitz constant of the push-forward mapping. They highlight the trade-off between the ability of VAEs and GANs to fit multimodal distributions and the stability of their training.

A common assumption in the imaging literature, validated empirically by Pope et al. (2020), is that distributions of natural images live on low dimensional manifolds. Understanding whether these distributions are composed of separated modes or not remains, to the best of our knowledge, an open problem. To that extent, the fact that unsupervised push-forward generative models perform well on datasets such as CelebA (Liu et al., 2015) could possibly be, in regard of our work, an indicator that the data distributions of those datasets are unimodal, or at least not composed of well separated modes.

Several techniques have been proposed in the literature to fit data distributions on disconnected manifolds. Most of them consist in overparametrizing the model, either by using stacked generative networks (Khayatkhoei et al., 2018; Mehr et al., 2019) or by learning a more complex latent distribution than the standard Gaussian (Gurumurthy et al., 2017; Rezende and Mohamed, 2015; Kingma et al., 2016; Luise et al., 2020). Other methods consist in rejecting a posteriori samples associated to large values of the Jacobian generator (Tanielian et al., 2020; Issenhuth et al., 2020). In this work, we empirically showed that score-based models seemed to be able to fit separated manifolds without model overparametrization or additional posterior sample rejection scheme, by not learning directly the push-forward mapping. It is possible that their good performances at implicitly learning push-forward maps with large Lipschitz constants also follows from the structure of the generation dynamic which consists in stacked networks with addition of noise input at each step. Hence, a future perspective of work would be to study if certain structures of the push-forward map are better than others to learn mappings with large Lipschitz constants while staying stable.

A possible limitation of this work is that the bounds derived on the Kullback-Leibler divergence and total variation distance are not tight (see Appendix S5), mainly because they take no account of the fact that when interpolating, $g_{\#}\mu_p$ has automatically less mass than ν on the modes since a significant amount of its total mass is between them. In future work, we plan to tighten the gap between our bounds and the true distance.

References

Antoniou, A., Storkey, A., and Edwards, H. (2018). Augmenting image classifiers using data augmentation generative adversarial networks. *International Conference on Artificial Neural Networks*, 27:594–603.

Arjovsky, M. and Bottou, L. (2017). Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877.

Brock, A., Donahue, J., and Simonyan, K. (2018). Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Courty, N., Flamary, R., and Ducoffe, M. (2018). Learning wasserstein embeddings. In *International Conference on Learning Representations*.

Dhariwal, P. and Nichol, A. (2021). Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34.

Dupuis, P. and Ellis, R. S. (2011). *A weak convergence approach to the theory of large deviations*. John Wiley & Sons.

Elfwing, S., Uchibe, E., and Doya, K. (2018). Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*, 107:3–11.

Fazlyab, M., Robey, A., Hassani, H., Morari, M., and Pappas, G. (2019). Efficient and accurate estimation of Lipschitz constants for deep neural networks. *Advances in Neural Information Processing Systems*, 32.

Fedus, W., Rosca, M., Lakshminarayanan, B., Dai, A. M., Mohamed, S., and Goodfellow, I. (2018). Many paths to equilibrium: Gans do not need to decrease a divergence at every step. In *International Conference on Learning Representations*.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. *stat*, 1050:20.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. *Advances in neural information processing systems*, 30.

Gulrajani, I., Raffel, C., and Metz, L. (2018). Towards gan benchmarks which require generalization. In *International Conference on Learning Representations*.

Gurumurthy, S., Kiran Sarvadevabhatla, R., and Venkatesh Babu, R. (2017). Deligan: Generative adversarial networks for diverse and limited data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 166–174.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Ho, J., Jain, A., and Abbeel, P. (2020). Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851.

Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

Hyvärinen, A. (2005). Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4).

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. *Conference On Computer Vision and Pattern Recognition*, 31:1125–1134.

Issenhuth, T., Tanielian, U., Picard, D., and Mary, J. (2020). Learning disconnected manifolds: Avoiding the no gan's land by latent rejection.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., and Aila, T. (2020). Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119.

Khayatkhoei, M., Elgammal, A., and Singh, M. (2018). Disconnected manifold learning for generative adversarial networks. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 7354–7364.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. (2016). Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29.

Kingma, D. P. and Welling, M. (2014). Auto-Encoding Variational Bayes. *Stat*, 1050:1.

Kodali, N., Abernethy, J., Hays, J., and Kira, Z. (2017). On convergence and stability of gans. *arXiv preprint arXiv:1705.07215*.

Kumar, A. and Poole, B. (2020). On implicit regularization in β-vaes. In *International Conference on Machine Learning*, pages 5480–5490. PMLR.

Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., and Aila, T. (2019). Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. *Conference on Computer Vision and Pattern Recognition*, 31:4681–4690.

Lim, J. H. and Ye, J. C. (2017). Geometric gan. *arXiv e-prints*, pages arXiv–1705.

Liu, Z., Luo, P., Wang, X., and Tang, X. (2015). Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738.

Lu, C., Chen, J., Li, C., Wang, Q., and Zhu, J. (2020). Implicit normalizing flows. In *International Conference on Learning Representations*.

Luise, G., Pontil, M., and Ciliberto, C. (2020). Generalization properties of optimal transport gans with latent distribution learning. *arXiv preprint arXiv:2007.14641*.

Maas, A. L., Hannun, A. Y., Ng, A. Y., et al. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3. Citeseer.

Mehr, E., Jourdan, A., Thome, N., Cord, M., and Guitteny, V. (2019). Disconet: Shapes learning on disconnected manifolds for 3d editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3474–3483.

Metz, L., Poole, B., Pfau, D., and Sohl-Dickstein, J. (2017). Unrolled generative adversarial networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. *International Conference on Learning Representations*, 6.

Mohajerin Esfahani, P. and Kuhn, D. (2018). Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *Mathematical Programming*, 171(1):115–166.

Nagarajan, V., Raffel, C., and Goodfellow, I. J. (2018). Theoretical insights into memorization in gans. In *Neural Information Processing Systems Workshop*, volume 1.

Odena, A., Buckman, J., Olsson, C., Brown, T., Olah, C., Raffel, C., and Goodfellow, I. (2018). Is generator conditioning causally related to gan performance? In *International conference on machine learning*, pages 3849–3858. PMLR.

Pennington, J., Schoenholz, S., and Ganguli, S. (2017). Resurrecting the sigmoid in deep learning through dynamical isometry: theory and practice. *Advances in neural information processing systems*, 30.

Peyré, G. and Cuturi, M. (2019). Computational optimal transport: with applications to data science. *Foundations and Trends in Machine Learning*, 11(5-6):355–607.

Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T. (2020). The intrinsic dimension of images and its impact on learning. In *International Conference on Learning Representations*.

Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., Madge, S., et al. (2021). Skilful precipitation nowcasting using deep generative models of radar. *Nature*, 597(7878):672–677.

Rezende, D. and Mohamed, S. (2015). Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252.

Sajjadi, M. S., Bachem, O., Lucic, M., Bousquet, O., and Gelly, S. (2018). Assessing generative models via precision and recall. *Advances in Neural Information Processing Systems*, 31.

Sandfort, V., Yan, K., Pickhardt, P. J., and Summers, R. M. (2019). Data augmentation using generative adversarial networks (CycleGAN) to improve generalizability in ct segmentation tasks. *Scientific Reports*, 9(1):1–9.

Song, Y. and Ermon, S. (2019). Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. (2020). Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*.

Stéphanovitch, A., Tanielian, U., Cadre, B., Klutchnikoff, N., and Biau, G. (2022). Optimal 1-wasserstein distance for wgans. *arXiv preprint arXiv:2201.02824*.

Sudakov, V. N. and Tsirelson, B. S. (1978). Extremal properties of half-spaces for spherically invariant measures. *Journal of Soviet Mathematics*, 9(1):9–18.

Tanielian, U., Issenhuth, T., Dohmatob, E., and Mary, J. (2020). Learning disconnected manifolds: a no gan's land. In *International Conference on Machine Learning*, pages 9418–9427. PMLR.

Tran, D., Ranganath, R., and Blei, D. M. (2017). Deep and hierarchical implicit models. *arXiv preprint arXiv:1702.08896*, 7(3):13.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Vincent, P. (2011). A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674.

Virmaux, A. and Scaman, K. (2018). Lipschitz regularity of deep neural networks: analysis and efficient estimation. *Advances in Neural Information Processing Systems*, 31.

Wang, X., Girshick, R., Gupta, A., and He, K. (2018). Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803.

Wenliang, L. K. and Kanagawa, H. (2020). Blindness of score-based methods to isolated components and mixing proportions. *arXiv preprint arXiv:2008.10087*.

Wu, Y. and He, K. (2018). Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19.

Yang, Z., Chen, W., Wang, F., and Xu, B. (2018). Improving neural machine translation with conditional sequence generative adversarial nets. *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1:1346–1355.

Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. (2019). Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] the limitations of our work are discussed in Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [No] The contribution of this paper is a theoretical contribution on generative modeling with no new algorithm. Consequently, the potential negative societal impacts are the same as for other papers about generative modeling (for instance bots, deepfakes,...). Since our contribution is theoretical we do not discuss these issues.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] The full set of assumptions is indicated before each result.
   (b) Did you include complete proofs of all theoretical results? [Yes] the complete proofs of the results are included in Appendix S2.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code to reproduce the main experiments will be included in the supplementary material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix S4.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We include the standard deviation for Figure 2 and Figure 4.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix S4.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]