Can Push-forward Generative Models Fit Multimodal Distributions?

Antoine Salmona
Centre Borelli,
ENS Paris Saclay, France

Agnès Desolneux
Centre Borelli, CNRS
ENS Paris Saclay, France

Julie Delon
MAP5, Université Paris Cité, France
Institut Universitaire de France (IUF)

Valentin De Bortoli
Center for Sciences of Data, CNRS
ENS Ulm, France

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 the recently introduced diffusion models do not suffer of such limitation.

1 Introduction

Generative modeling has become over the last years one of the most popular research topics in machine learning and 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

$$\min_{\theta\in\Theta} D(\tfrac{1}{m}\textstyle\sum_i \delta_{x_i}, \nu_\theta) \ ,$$

where D is a similarity measure between probability distributions and δ_x is the delta distribution at 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

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

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

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) [2] (Kingma and Welling, 2014) and the *Generative Adversarial Networks* (GANs) (Goodfellow et al., 2014). It also includes other models such as most of normalizing flows (Rezende and Mohamed, 2015).

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 implicitly regularized, which limits 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 model 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 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 (More details can be found in Appendix S2). 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. More precisely, 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 for $g_\#\mu_p$ to 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 that there is a fundamental trade-off for (direct) push-forward generative models between expressivity and stability of training. In Section 4, we illustrate these theoretical results on several experiments, showing the difficulties of GANs and VAEs to simulate multimodal distributions. We compare these models with SGMs and show experimentally that SGMs seem to be able to generate correctly multimodal distributions while keeping the Lipschitz constant of the score network relatively small, suggesting that these models do not suffer of such previously mentioned limitations. 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 $\mathbb{R}^d$, $f_\#\mu(\mathsf{A}) = \mu(f^{-1}(\mathsf{A}))$.

[2] In this work, the VAE model considered is the Gaussian-VAE since the data are real-valued. See Appendix S2 for details on why the generated distribution is of the form $g_{\theta\#}\mu_p$ in this model.

2 Related Works

Assessing the efficiency of push-forward models is a recurrent and important question in the literature. 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, Tanielian et al. (2020) prove an upper bound on the precision of vanilla GANs (the proportion of generated samples which could have been generated by the target distribution). 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.

In the context of normalizing flows, it has been shown that the invertibility constraint limits the expressivity of the model. Indeed, Cornish et al. (2020) show that distributions generated by invertible normalizing flows have a support which is necessarily homeomorphic to the support of the latent distribution. As an outcome, the Lipschitz constant of the inverse flow has to approach infinity to correctly approximate distributions lying on disconnected manifolds (Cornish et al., 2020; Hagemann and Neumayer, 2021; Behrmann et al., 2021). To improve the expressivity of normalizing flows, it has been proposed in Cornish et al. (2020) and Wu et al. (2020) to inject stochasticity in the model.

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 strictly 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 the mass of γ on the hypersurface $\partial\mathsf{A}$. Note that the support of γ and A can be sets of intrinsic 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 $r = 0$, dividing by r 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 intrinsic 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.

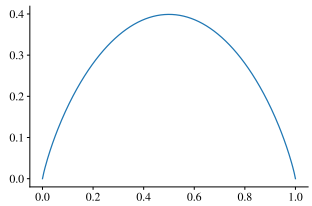

Figure 1: Graph of $\varphi \circ \Phi^{-1}$.

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 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* $\nu = \lambda \mathrm{N}(m_1, \sigma^2 \operatorname{Id}_d) + (1-\lambda)\mathrm{N}(m_2, \sigma^2 \operatorname{Id}_d)$ *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*

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

Sketch of proof. The proof of this result consists into 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 $\nu^+(\partial\mathsf{H})$. See Figure S1 for a visualization of $\partial\mathsf{H}$ in the univariate case. □

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* $\operatorname{supp}(\nu) = \mathbb{R}$. *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 *and* ν, *where* Φ_ν *is the cumulative distribution function of* ν. *Then we have* $\operatorname{Lip}(g) \geq \operatorname{Lip}(T_{\mathrm{OT}})$.

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 $\operatorname{Lip}(g)$. 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):

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

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 lower 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 and plugging it in the Donsker-Varadhan representation in order to get a lower bound 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. Similarly to Corollary 6, we also get an explicit lower bound in the case where ν is a mixture of two isotropic Gaussians. As for Corollary 6, we stick to the balanced case for simplicity.

Corollary 8. *Let $\nu = (1/2)\left[\mathrm{N}(m_1, \sigma^2\,\mathrm{Id}_d) + \mathrm{N}(m_2, \sigma^2\,\mathrm{Id}_d)\right]$ 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. We denote*

$$\lambda = g_{\#}\mu_p\left(\{(m_2-m_1)^T\,(x-(m_2+m_1)/2) \leq 0 \ : \ x\in\mathbb{R}^d\}\right) \ ,$$

and we suppose without loss of generality, that $\lambda \in (0, 1/2]$. Then,

$$d_{\mathrm{KL}}(g_{\#}\mu_p, \nu) \geq A\log\left(\frac{A}{B}\right) + (1-A)\log\left(\frac{1-A}{1-B}\right) \ ,$$

where

$$A = \int_{-\Phi^{-1}(1-\lambda)}^{\|m_2-m_1\|/4\sigma\mathrm{Lip}(g)-\Phi^{-1}(1-\lambda)} \varphi(t)\mathrm{d}t \ , \text{ and } B = (1/2)\int_{\|m_2-m_1\|(2\sigma-1)/4\sigma^2}^{\|m_2-m_1\|(2\sigma+1)/4\sigma^2} \varphi(t)\mathrm{d}t \ .$$

Observe that this time, $\mathrm{Lip}(g)$ is no longer the only dependency in g since the bound also depends on the proportion of the modes of $g_{\#}\mu_p$. However, it should be noted that when $g_{\#}\mu_p$ approximates correctly ν, λ is automatically close to $1/2$ and so $\Phi^{-1}(1-\lambda)$ is small in that case. To conclude, this section, we highlight the fact that, if our results are dimension free in theory, the dimension might be hidden in the distances between modes and the Lipschitz constant of g when working with real datasets. Indeed, the order of magnitude of the Euclidean distance between two samples x_i is likely to increases with the dimension d. As an outcome, the orders of magnitude of the distance between modes and so the Lipschitz constant that g must reach for correct generation probably increase with d also.

4 Experiments

In what follows, we illustrate the pratical implications of our results by training GANs, VAEs and SGMs on simple bi-modal distributions. More precisely, we show on one hand that generating multimodal distributions with GANs and VAEs is difficult since for those models, good generation necessarily involves generative networks with large Lipschitz constants. On the other hand, we show that SGMs seem to be able to generate multimodal distributions while keeping the Lipschitz constant of the score network relatively small 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 numbers of learnable parameters. All details on the experiments and architecture of the networks can be found in Appendix S5.

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.

Histograms of generated distributions. Figure 2 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 forget parts of their 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.

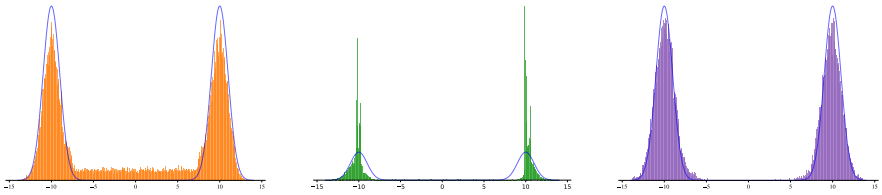

Figure 2: Histograms of distributions generated with VAE (left, in orange), GAN (middle, in green), and with SGM (right, in purple) for $m = 10$. The data distribution densities are plotted in blue.

Lipschitz constant and mass between modes. In Figure 3 (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 2, as the mapping learned by the VAE is not stiff enough to concentrate the push forward measure on the two modes. One possible explanation for the interpolating behavior of the VAE is that the Euclidean norm of the Jacobian of the VAE decoder is implicitly 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 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 push-forward mapping (the whole generation dynamic) is likely much larger than the Lipschitz constant of the neural network itself, and so the model is able to push-forward a Gaussian distribution into a multimodal distribution keeping a relatively small Lipschitz constant of the score network. Finally, in Figure 3 (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 this observation does not apply to SGMs since in this setting the network is applied multiple times.

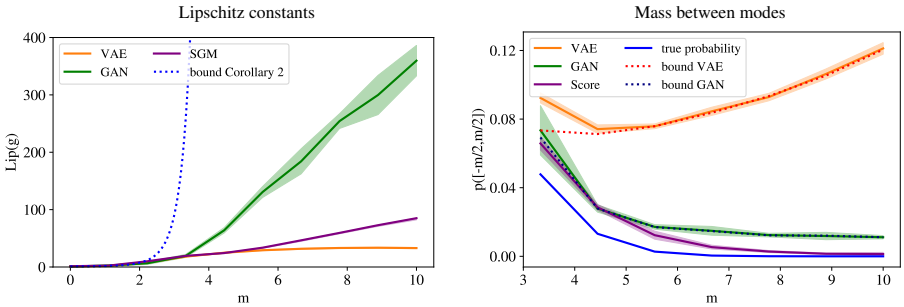

Figure 3: Left: evolution of the Lipschitz constants of the three different generative models trained on 50000 samples of $(1/2)[\mathrm{N}(-m, 1) + \mathrm{N}(m, 1)]$, in function of m. Right: evolution of the proportion of samples generated by the three models on the interval $[-m/2, m/2]$. We also 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.

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 instability and mode collapse. This is why many state-of-the-art GANs apply spectral normalization (Miyato et al., 2018) on their generators. In Figure 4 (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 4 (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.

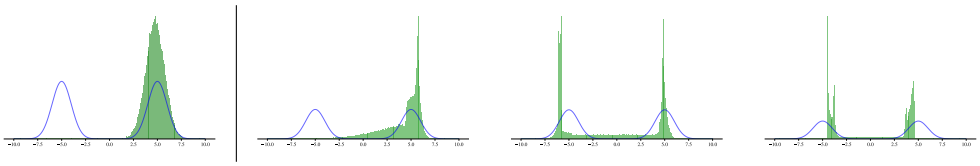

Figure 4: Histograms of distributions generated with GANs with spectral normalization applied on the generator (left), and with gradient penalty (right) for $\mathrm{Lip}(g) \approx L = 5$, $\mathrm{Lip}(g) \approx L = 15$ and $\mathrm{Lip}(g) \approx L = 25$. The data distribution densities are plotted in blue.

Influences of generator depth and time of training. In Figure 5, we study the effect of increasing the number of layers of the generative network as well as increasing the training time on the value of the Lipschitz constant of the VAE decoder and the GAN generator. 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 dramatically 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.

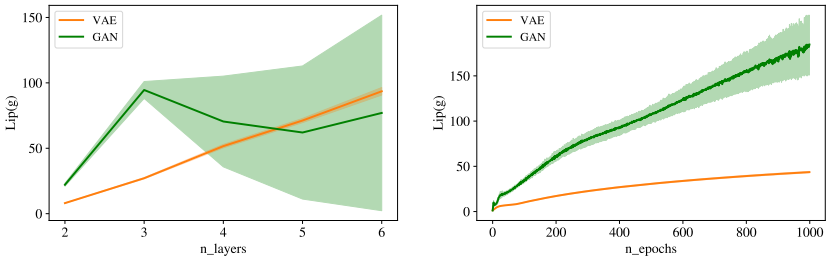

Figure 5: Evolution of the Lipschitz constant of the generative network with respect to its number of layers (left) and of the Lipschitz constant in function of the numbers of epochs (right). The experiments are averaged over 10 runs and the colored bands correspond to +/- the standard deviation.

Influence of generator architecture. Finally, we study in Figure 6 the impact of the architecture of the generative network (i.e. the VAE decoder and the GAN generator) on its Lipschitz constant as well as on the training stability of the model by comparing three different architectures: first, we use a simple feed-forward network as precedently, then we add additive skip-connections of type "resnet" (He et al., 2016) to the previous backbone, and last we add concatenation skip-connections of type "densenet" (Huang et al., 2017) instead of additive skip-connections. For both models, it seems that more expressive decoder architectures do not help to reach larger values of Lipschitz constant. However, one can observe that in the GAN setting, even if the model remains certainly too unstable for correct distribution generation, adding additive skip-connections seems to stabilize the training a little since the colored bands are narrower than for the two other models. This suggests that some generator architectures may be better than others at learning mappings with large Lipschitz constants while staying stable.

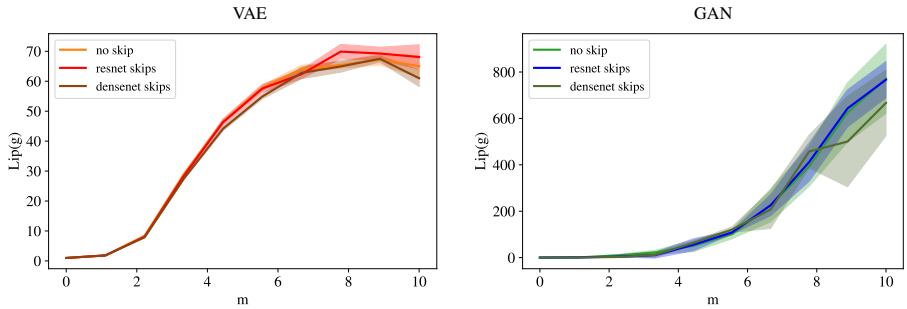

Figure 6: Evolution of the Lipschitz constant of the VAE decoder (left) and the GAN generator (right) trained on 50000 samples of $(1/2)[\mathrm{N}(-m,1)+\mathrm{N}(m,1)]$ for 3 different architectures of the generative network: simple feed-forward backbone, backbone with skip-connections of type "resnet", and backbone with skip-connections of type "densenet". Experiments are averaged over 5 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.

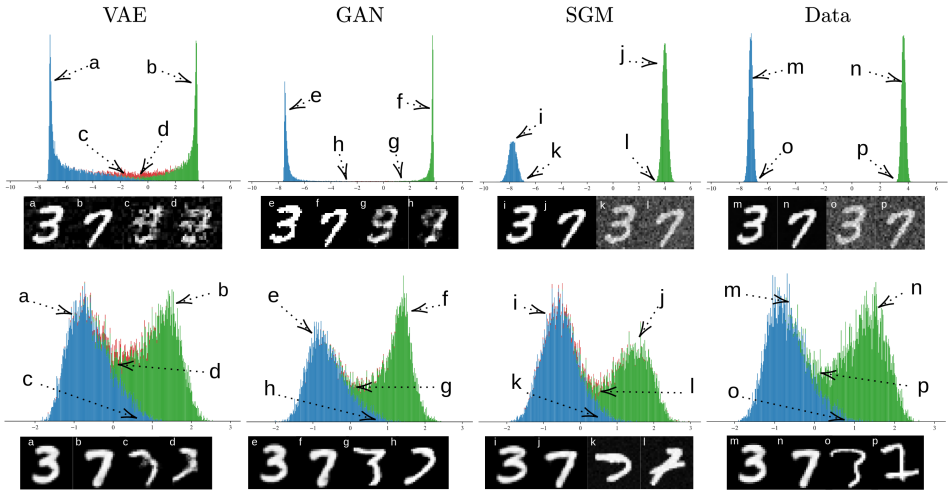

Figure 7: mixture of Gaussians (top): histograms of projections on the line passing through the mean of each Gaussian. Subset of MNIST (bottom): histograms of projections on the line passing through the barycenters of all the 3 and 7 in the deep Wasserstein embedding space. Bins of data are colored in 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 intrinsic 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 7 top. Moreover, GAN and VAE both fail to generate noisy versions of the images. As in the univariate case, the SGM is able to not interpolate between

modes and seem to retrieve the Gaussian structure of the modes. This suggests that while direct push-forward models fail at representing multimodal distributions, considering stacked models with noise 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).

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 7 (bottom). Note that the distribution does not exhibit strong multimodality features contrary to the mixture of Gaussians settings, see Figure 7. 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. This suggests that the structure of the generation dynamic in these models is particularly adapted to (indirectly) learn mappings with large Lipschitz constants. Their good performance on multimodal distributions might follow from the fact that these models do not optimize directly the push-forward mapping itself and/or that noise is injected at each step during the generation process. Hence, a future perspective of work would be to study what are the structural aspects of diffusion models that play a significant role in their expressivity.

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 S6), 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. In *International Conference on Artificial Neural Networks*, pages 594–603. Springer.

Arjovsky, M. and Bottou, L. (2017). Towards principled methods for training generative adversarial networks. *stat*, 1050:17.

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

Behrmann, J., Vicol, P., Wang, K.-C., Grosse, R., and Jacobsen, J.-H. (2021). Understanding and mitigating exploding inverses in invertible neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 1792–1800. 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*.

Cornish, R., Caterini, A., Deligiannidis, G., and Doucet, A. (2020). Relaxing bijectivity constraints with continuously indexed normalising flows. In *International Conference on Machine Learning*, pages 2133–2143. PMLR.

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.

Hagemann, P. and Neumayer, S. (2021). Stabilizing invertible neural networks using mixture models. *Inverse Problems*, 37(8):085002.

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. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 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. (2015). Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

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. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690.

Lim, J. H. and Ye, J. C. (2017). Geometric gan. *arXiv preprint arXiv:1705.02894*.

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 *International Conference on Machine Learning*, volume 30, page 3. PMLR.

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 *International Conference on Learning Representations*.

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. In *International Conference on Learning Representations*.

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. (2017). Hierarchical implicit models and likelihood-free variational inference. *Advances in Neural Information Processing Systems*, 30.

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, H., Köhler, J., and Noé, F. (2020). Stochastic normalizing flows. *Advances in Neural Information Processing Systems*, 33:5933–5944.

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.

S1 Organization of the supplementary

The supplementary is organized as follows. First, in Appendix S2, we give details on why the generated distribution in each aforementioned generative model is a push-forward of a standard Gaussian distribution. In Appendix S3, we give the full proofs of all the theoretical results of the paper. In Appendix S4, we show a generalization of Corollary 5 in the case there are more than two disconnected manifolds. In Appendix S5, we give details on the experiments. Finally, in Appendix S6, we provide additional experimental results and additional visualizations of histograms of generated distributions for the univariate case, and generated data for the experiments on MNIST.

S2 More details on push-forward generative models

In this section, we specify why in each aforementioned model the generated distribution is of the form $g_{\theta\#}\mu_p$ with $\mu_p = \mathrm{N}(0, \mathrm{Id}_p)$ being the standard Gaussian distribution in dimension p and $g_\theta : \mathbb{R}^p \to \mathbb{R}^d$ being a deterministic mapping of parameter θ.

S2.1 Direct push-forward models

In GANs, the generated distribution is trivially of the form $g_{\theta\#}\mu_p$, where g_θ is the generator and p is the dimension of the latent space. This is also the case in most of normalizing flow models, where g_θ is a composition of neural flows and p is automatically equal to d since the networks must be invertible. In the Gaussian-VAE model described in Kingma and Welling (2014), the conditional probability $p_\theta(x|z)$, with $z \in \mathbb{R}^p$ and $x \in \mathbb{R}^d$ being respectively the latent and observable variables, is of the form $p_\theta(x|z) = \mathrm{N}(f_\theta(z), h_\theta^2(z)\,\mathrm{Id}_d)$, where $f_\theta(z)$ and $g_\theta)$ are the outputs of the decoder. This is often simplified in practice to $p_\theta(x|z) = \mathrm{N}(f_\theta(z), c^2\,\mathrm{Id}_d)$ with $c > 0$ being an hyperparameter of the model. Thus the generated distribution in Gaussian-VAE is of the form $g_{\#}\mu_p$, where $g_\theta : \mathbb{R}^{p+d} \to \mathbb{R}^d$ is the neural network defined as

$$g_\theta(z, z') = \begin{cases} f_\theta(z) + h_\theta^2(z)z' & \text{if } p_\theta(x|z) = \mathrm{N}(f_\theta(z), h_\theta^2(z)\,\mathrm{Id}_d) \\ f_\theta(z) + c^2 z' & \text{if } p_\theta(x|z) = \mathrm{N}(f_\theta(z), c^2\,\mathrm{Id}_d)\ . \end{cases}$$

Moreover, if relaxing the conditional probability to $p_\theta(x|z) = \mathrm{N}(f_\theta(z), h_\theta^2(z)\,\mathrm{Id}_d)$ or $p_\theta(x|z) = \mathrm{N}(f_\theta(z), c^2\,\mathrm{Id}_d)$ instead of $\delta_{f_\theta(z)}$ is crucial for the training of VAEs, this is not particularly relevant during inference since the relaxed conditional distribution is simply a noisy version of the push-forward measure $f_{\theta\#}\mu_p$. For this reason, the inference in VAEs is often done by simply sampling from the single output $f_\theta(z)$ of the decoder, with $z \sim \mu_p$, and so the generated distribution is trivially of the form $f_{\theta\#}\mu_p$ in that case.

S2.2 Score-based generative models

In diffusion models, the generation process is an Euler-Maruyama discretization of the reverse-time denoising diffusion (Song et al., 2020)

$$\mathrm{d}X_t = \big(f(X_t, t) - g^2(t)\nabla \log p_t(X_t)\big)\,\mathrm{d}t + g(t)\mathrm{d}B_t\ ,$$

where $(X_t)_{t\in[0,T]}$ is a random process on $\mathbb{R}^d$, $(B_t)_{t\in[0,T]}$ is a Brownian motion, and where $f(.,t) : \mathbb{R}^d \to \mathbb{R}^d$ is a vector-valued function called the drift operator, $g : \mathbb{R} \to \mathbb{R}$ is a real-valued function called the diffusion coefficient and $\nabla \log p_t(X_t)$ is the score of the marginal law of X_t, which is approximated by a neural network $s_\theta(X_t, t)$. This discretization yields for instance, for an appropriate choice of f and g, to the annealed Langevin dynamic (Song and Ermon, 2019):

$$\begin{cases} x_0 = z_0 \quad (z_0 \sim \mathrm{N}(0, \mathrm{Id}_d)) \\ x_{k+1} = x_k + (\alpha_k/2)s_\theta(x_k, \sigma_k) + \sqrt{\alpha_k}z_{k+1} \quad (z_{k+1} \sim \mathrm{N}(0, \mathrm{Id}_d))\ , \end{cases}$$

where $\alpha_k = \varepsilon\sigma_k^2/\sigma_0^2$ with $\varepsilon > 0$ being an hyperparameter of the model and $(\sigma_k)_{k\geq 0}$ is such that it exists K (another hyperparameter of the model) such that $(\sigma_{Ki})_{i\geq 0}$ is a geometric progression, and for all $k \geq 0$, $\sigma_k = \sigma_{K\lfloor k/K\rfloor}$. Denoting for $k \geq 1$, $h_\theta^k : \mathbb{R}^{2d} \to \mathbb{R}^d$ the function defined as

$$h_\theta^k(x, y) = x + (\alpha_k/2)s_\theta(x, \sigma_k) + \sqrt{\alpha_k}y\ ,$$

it follows that when the data are generated with a Langevin dynamic of N iterations, the generated distribution is of the form $g_{\theta\#}\mu_p$ with $p = d(N+1)$ and where $g_\theta : \mathbb{R}^{d(N+1)} \to \mathbb{R}^d$ is the function defined as

$$g_\theta(z_0, z_1, z_2, \dots, z_{N-1}, z_N) = h_\theta^N \left(h_\theta^{N-1}\left(\dots(h_\theta^2\left(h_\theta^1(z_0, z_1), z_2\right), \dots), z_{N-1}\right), z_N\right) .$$

S3 Proofs of the theoretical results

S3.1 Proof of Theorem 1

We start by recalling the Gaussian isoperimetric inequality Sudakov and Tsirelson (1978).

Lemma S9. *Let* $\mathsf{A} \in \mathcal{B}(\mathbb{R}^p)$ *and* $\mu_p = \mathrm{N}(0, \mathrm{Id}_p)$. *Then we have*

$$\mu_p^+(\partial \mathsf{A}) \geq \varphi(\Phi^{-1}(\mu_p(\mathsf{A}))) ,$$

where $\varphi(x) = (2\pi)^{-1/2}\exp[-x^2/2]$ *and* $\Phi(x) = \int_{-\infty}^x \varphi(t)\mathrm{d}t$. *Equivalently, for all* $r \geq 0$

$$\mu_p(\mathsf{A}_r) \geq \Phi(r + \Phi^{-1}(\mu_p(\mathsf{A}))) .$$

In particular, using Lemma S9, one can show that among all sets of given Gaussian measure μ_p, half-spaces have the minimal μ_p-surface area. We are now ready to turn to the proof of Theorem 1.

Proof of Theorem 1. Let $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ such that $g_\#\mu_p(\mathsf{A}) > 0$ (note that if $g_\#\mu_p(\mathsf{A}) = 0$ then the result is trivial). First, we show that for any $\varepsilon > 0$, $g((g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)}) \subset \mathsf{A}_\varepsilon$. Let x be in $g((g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)})$. There exists $z_1 \in (g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)}$ such that $g(z_1) = x$. There also exists $z_2 \in g^{-1}(\mathsf{A})$ such that

$$\|z_1 - z_2\| \leq \varepsilon/\mathrm{Lip}(g) .$$

Hence, we have that

$$\|x - a\| \leq \mathrm{Lip}(g)\|z_1 - z_2\| \leq \varepsilon ,$$

where $a = g(z_2)$. Since $z_2 \in g^{-1}(\mathsf{A})$, $a \in \mathsf{A}$, and therefore $x \in \mathsf{A}_\varepsilon$. Using this result, the fact that $g_\#\mu_p(\mathsf{B}) = \mu_p(g^{-1}(\mathsf{B}))$ and $\mathsf{B} \subset g^{-1}(g(\mathsf{B}))$ for any $\mathsf{B} \in \mathcal{B}(\mathbb{R}^d)$, we have

$$\begin{aligned}\liminf_{\varepsilon\to 0^+}\{g_\#\mu_p(\mathsf{A}_\varepsilon) - g_\#\mu_p(\mathsf{A})\}/\varepsilon &\geq \liminf_{\varepsilon\to 0^+}\{g_\#\mu_p(g((g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)})) - g_\#\mu_p(\mathsf{A})\}/\varepsilon \\ &\geq \liminf_{\varepsilon\to 0^+}\{\mu_p((g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)}) - \mu_p(g^{-1}(\mathsf{A}))\}/\varepsilon . \qquad \text{(S1)}\end{aligned}$$

Using Lemma S9, we have

$$\mathrm{Lip}(g)\liminf_{\varepsilon\to 0^+}\{(\mu_p((g^{-1}(\mathsf{A}))_{\varepsilon/\mathrm{Lip}(g)}) - \mu_p(g^{-1}(\mathsf{A})))\}/\varepsilon \geq \varphi(\Phi^{-1}(\mu_p(g^{-1}(\mathsf{A})))) ,$$

Combining this result and (S1), we get that

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

In addition, using Lemma S9, we have for all $r \geq 0$

$$\mu_p((g^{-1}(\mathsf{A}))_{r/\mathrm{Lip}(g)}) \geq \Phi(r/\mathrm{Lip}(g) + \Phi^{-1}(\mu_p(g^{-1}(\mathsf{A})))) .$$

Using this result and that $g((g^{-1}(\mathsf{A}))_{r/\mathrm{Lip}(g)}) \subset \mathsf{A}_r$, we have for any $r \geq 0$

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

□

S3.2 Proof of Corollary 2

We prove the corollary when $\nu = \lambda \mathrm{N}(-m, \sigma^2 \operatorname{Id}_d) + (1-\lambda)\mathrm{N}(m, \sigma^2 \operatorname{Id}_d)$ since the problem can always be reduced to that case by translation and setting $m = (m_2 - m_1)/2$. Let H be defined by $\mathsf{H} = \{x \in \mathbb{R}^d | m^T x \geq 0\}$. Note that for any $x \in \partial\mathsf{H}$, $\|x - m\| = \|x + m\|$. Since the problem is invariant by rotation, we can consider without any loss of generality that $m = (\|m\|, 0, \dots, 0)$. In that case, we have $\nu = \nu_1 \otimes \mathrm{N}(0, \sigma^2 \operatorname{Id}_{d-1})$, where $\nu_1 = \lambda \mathrm{N}(-\|m\|, \sigma^2) + (1-\lambda)\mathrm{N}(\|m\|, \sigma^2)$, and $\otimes$ is the tensor product between measures. In this case, we have that $\mathsf{H} = \{x_1 \geq 0\} \times \mathbb{R}^{d-1}$. Therefore, we have

$$\begin{aligned}\nu^+(\partial\mathsf{H}) &= \liminf_{\varepsilon \to 0^+} \{(\textstyle\int_{\mathsf{H}_\varepsilon} p_\nu(x)\mathrm{d}x - \int_{\mathsf{H}} p_\nu(x)\mathrm{d}x)\}/\varepsilon \ , \\ &= \liminf_{\varepsilon \to 0^+} \{(\textstyle\int_{-\varepsilon}^{+\infty} \int_{\mathbb{R}^{d-1}} p_{\nu_1}(x_1)h(y)\mathrm{d}x_1\mathrm{d}y - \int_0^{+\infty} \int_{\mathbb{R}^{d-1}} p_{\nu_1}(x_1)h(y)\mathrm{d}x_1\mathrm{d}y)\}/\varepsilon \ ,\end{aligned}$$

where p_ν and p_{ν_1} are the respective densities of ν and ν_1, and h is the density of $\mathrm{N}(0, \sigma^2 I_{d-1})$. It follows that

$$\begin{aligned}\nu^+(\partial\mathsf{H}) &= \liminf_{\varepsilon \to 0^+} (1/\varepsilon) \textstyle\int_{-\varepsilon}^0 p_{\nu_1}(x_1)(\int_{\mathbb{R}^{d-1}} h(y)\mathrm{d}y)\mathrm{d}x_1 \\ &= \liminf_{\varepsilon \to 0^+} (1/\varepsilon) \textstyle\int_{-\varepsilon}^0 p_{\nu_1}(x_1)\mathrm{d}x_1 = p_{\nu_1}(0) = (2\pi\sigma^2)^{-1/2} \exp[-\|m\|^2/(2\sigma^2)] \ .\end{aligned}$$

Applying Theorem 1, we get that

$$\operatorname{Lip}(g) \geq \varphi(\Phi^{-1}(\nu(\mathsf{H})))/\nu^+(\partial\mathsf{H}) \ .$$

Furthermore, one can derive that

$$\begin{aligned}\nu(\mathsf{H}) &= \lambda(1 - \Phi(m/\sigma)) + \Phi(m/\sigma)(1-\lambda) \\ &= \lambda(1 - 2\Phi(m/\sigma)) + \Phi(m/\sigma) \ .\end{aligned}$$

Observing that $\lambda - \nu(\mathsf{H})$ is an increasing function of λ and $\lambda - \nu(\mathsf{H}) = 0$ if $\lambda = 1/2$, we get that $\lambda \leq \nu(\mathsf{H})$ if $\lambda \leq 1/2$ and $\lambda \geq \nu(\mathsf{H})$ if $\lambda \geq 1/2$. Since $\varphi \circ \Phi^{-1}$ reaches its maximum in $1/2$, it follows that for any $\lambda \in (0,1)$ we have

$$\varphi(\Phi^{-1}(\nu(\mathsf{H}))) \geq \varphi(\Phi^{-1}(\lambda)) \ ,$$

and thus

$$\begin{aligned}\operatorname{Lip}(g) &\geq (2\pi)^{1/2}\sigma\varphi(\Phi^{-1}(\lambda)) \exp[\|m\|^2/(2\sigma^2)] \\ &\geq \sigma \exp[\|m\|^2/(2\sigma^2) - (\Phi^{-1}(\lambda))^2/2] \ ,\end{aligned}$$

which concludes the proof.

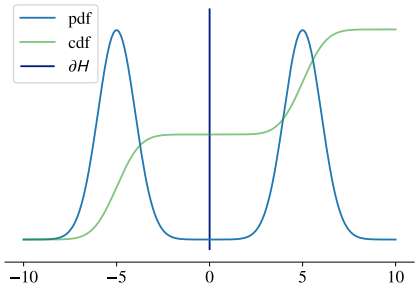

Figure S1: The hypersurface $\partial\mathsf{H}$ in the univariate case.

S3.3 Proof of Corollary 3

Since ν admits a density p_ν with respect to the Lesbegue measure, it follows that Φ_ν is absolutely continuous and therefore differentiable almost everywhere w.r.t. the Lebesgue measure using the Lebesgue differentiation theorem. Moreover, since $\operatorname{supp}(\nu) = \mathbb{R}$, it follows that $\Phi_\nu : \mathbb{R} \to (0,1)$ is

increasing and therefore is bijective, and so $T_{\mathrm{OT}} = \Phi_\nu^{-1} \circ \Phi$ is also differentiable almost everywhere w.r.t. the Lebesgue measure and bijective, with inverse $T_{\mathrm{OT}}^{-1} = \Phi^{-1} \circ \Phi_\nu$, using (Peyré and Cuturi, 2019, Remark 2.29). Therefore, for any $x \in \mathbb{R}$ we have

$$\begin{aligned} T'_{\mathrm{OT}}(x) &= \varphi(x)/p_\nu(T_{\mathrm{OT}}(x)) \\ &= \varphi(\Phi^{-1}(\Phi_\nu(T_{\mathrm{OT}}(x)))/p_\nu(T_{\mathrm{OT}}(x)) \ . \end{aligned}$$

Let $y \in \mathbb{R}$. Using Theorem 1 with $\mathsf{A} = (-\infty, y]$ we get that for any $g : \mathbb{R}^p \to \mathbb{R}$, Lipschitz such that $g_\# \mu_p = \nu$,

$$\mathrm{Lip}(g) \geq \sup_{y \in \mathbb{R}} \varphi(\Phi^{-1}(\Phi_\nu(y))/p_\nu(y) \ ,$$

and so, since T_{OT} is bijective

$$\mathrm{Lip}(g) \geq \sup_{x \in \mathbb{R}} |T'_{\mathrm{OT}}(x)| \ ,$$

which concludes the proof.

S3.4 Proof of Theorem 4

Let $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ and let $r > 0$. We have on one hand

$$\begin{aligned} |g_\# \mu_p(\mathsf{A}_r \setminus \mathsf{A})| &\leq |g_\# \mu_p(\mathsf{A}_r \setminus \mathsf{A}) - \nu(\mathsf{A}_r \setminus \mathsf{A})| + |\nu(\mathsf{A}_r \setminus \mathsf{A})| \\ &\leq d_{\mathrm{TV}}(g_\# \mu_p, \nu) + \nu(\mathsf{A}_r \setminus \mathsf{A}) \ . \end{aligned}$$

Using Theorem 1, we get

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

and so

$$d_{\mathrm{TV}}(g_\# \mu_p, \nu) \geq \alpha_g(\mathsf{A}, r) - g_\# \mu_p(\mathsf{A}) - \nu(\mathsf{A}_r \setminus \mathsf{A}) \ ,$$

where $\alpha_g(\mathsf{A}, r) = \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_\# \mu_p(\mathsf{A}))\right)$. On the other hand, we have

$$\begin{aligned} |g_\# \mu_p(\mathsf{A}_r)| &\leq |g_\# \mu_p(\mathsf{A}_r) - \nu(\mathsf{A}_r)| + |\nu(\mathsf{A}_r)| \\ &\leq d_{\mathrm{TV}}(g_\# \mu_p, \nu) + \nu(\mathsf{A}_r \setminus \mathsf{A}) + \nu(\mathsf{A}) \ . \end{aligned}$$

Using Theorem 1, we get

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

and so

$$d_{\mathrm{TV}}(g_\# \mu_p, \nu) \geq \alpha_g(\mathsf{A}, r) - \nu(\mathsf{A}) - \nu(\mathsf{A}_r \setminus \mathsf{A}) \ ,$$

which concludes the proof.

S3.5 Proof of Corollary 5

To prove Corollary 5, we will need the following lemma:

Lemma S10. *Let $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$ and $r > 0$. We denote $\mathsf{B} = (\mathsf{A}_r)^c$. Then*

$$\mathsf{B}_r \subset \bar{\mathsf{A}^c} \ ,$$

where $\bar{\mathsf{A}^c}$ denotes the closure of the complementary of A.

Proof. Let $x \in \mathsf{B}_r$. There exists $b \in \mathsf{B}$ such that $\|x - b\| \leq r$. Moreover, since $\mathsf{B} = (\mathsf{A}_r)^c$, it follows that for all $a \in \mathsf{A}$,

$$\|b - a\| > r \ .$$

Then

$$r < \|b - x\| + \|x - a\| \ ,$$

and so, it follows that for all $a \in \mathsf{A}$,

$$\|x - a\| > 0 \ .$$

Thus $x \in \bar{\mathsf{A}^c}$. □

Now we are ready to turn to the proof of Corollary 5.

Proof of Corollary 5. We set $r = d(\mathsf{M}_1, \mathsf{M}_2)/2$ and $\mathsf{A} = (\mathsf{M}_1)_r$. Using Theorem 4, we have

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \alpha_g(\mathsf{A}, r) - \min\{g_{\#}\mu_p(\mathsf{A}), \nu(\mathsf{A})\} - \nu(\mathsf{A}_r \setminus \mathsf{A}) \ .$$

First we suppose that $g_{\#}\mu_p(\mathsf{A}) \geq \nu(\mathsf{A})$: since Φ is a non-decreasing function, it follows that

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

Moreover $\min\{g_{\#}\mu_p(\mathsf{A}), \nu(\mathsf{A})\} = \nu(\mathsf{A}) = \lambda = \Phi(\Phi^{-1}(\lambda))$ and so it follows

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

since ν has no mass on $\mathsf{A}_r \setminus \mathsf{A}$. Now we suppose that $g_{\#}\mu_p(\mathsf{A}) \leq \nu(\mathsf{A})$: we then set $\mathsf{B} = \mathsf{A}^c$. Since $g_{\#}\mu_p(\mathsf{A}) \leq \nu(\mathsf{A})$, we have $g_{\#}\mu_p(\mathsf{B}) \geq \nu(\mathsf{B})$. Applying Theorem 4, and the same reasoning as before we get

$$\begin{aligned} d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) &\geq \alpha_g(\mathsf{B}, r) - \min\{g_{\#}\mu_p(\mathsf{B}), \nu(\mathsf{B})\} - \nu(\mathsf{B}_r \setminus \mathsf{B}) \\ &\geq \Phi\left(d(\mathsf{M}_1, \mathsf{M}_2)/(2\mathrm{Lip}(g)) + \Phi^{-1}(1-\lambda)\right) - \Phi(\Phi^{-1}((1-\lambda)) - \nu(\mathsf{B}_r \setminus \mathsf{B}) \ . \end{aligned}$$

Using Lemma S10, we get that $\nu(\mathsf{B}_r \setminus \mathsf{B}) \leq \nu(\bar{\mathsf{A}^c} \setminus (\mathsf{A}_r)^c)$ but $\nu(\bar{\mathsf{A}^c} \setminus (\mathsf{A}_r)^c) = 0$ since ν has no mass on $\bar{\mathsf{A}^c} \setminus (\mathsf{A}_r)^c$ except on its boundary and so its follows that

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

since $\Phi^{-1}(1-\lambda) = -\Phi^{-1}(\lambda)$. Since $\lambda \geq 1/2$, it follows that $\Phi^{-1}(\lambda) \geq 0$ and so

$$\int_{-\Phi^{-1}(\lambda)}^{r/\mathrm{Lip}(g)-\Phi^{-1}(\lambda)} \varphi(t)dt \geq \int_{\Phi^{-1}(\lambda)}^{r/\mathrm{Lip}(g)+\Phi^{-1}(\lambda)} \varphi(t)dt \ ,$$

which concludes the proof. □

S3.6 Proof of Corollary 6

As previously, we prove the corollary when $\nu = (1/2)[\mathrm{N}(-m, \sigma^2 \mathrm{Id}_d) + \mathrm{N}(m, \sigma^2 \mathrm{Id}_d)]$ since the problem can always be reduced to that case by translation and setting $m = (m_2 - m_1)/2$. Since the problem is invariant by rotation, we can assume without any loss of generality that $m = (\|m\|, 0, \ldots, 0)$. Let H be the half-space of $\mathbb{R}^d$ defined by $\mathsf{H} = (-\infty, 0] \times \mathbb{R}^{d-1}$ and we set $r = \|m\|/2\sigma$. First we suppose that $g_{\#}\mu_p(\mathsf{H}) \geq \nu(\mathsf{H})$: using Theorem 4, we get that

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \alpha_g(\mathsf{H}, r) - \min\{g_{\#}\mu_p(\mathsf{H}), \nu(\mathsf{H})\} - \nu(\mathsf{H}_r \setminus \mathsf{H}) \ ,$$

with $\mathsf{H}_r = (-\infty, \|m\|/2\sigma] \times \mathbb{R}^{d-1}$. On one hand we have that $\nu = \nu_1 \otimes \mathrm{N}(0, \sigma^2 \mathrm{Id}_{d-1})$, where $\nu_1 = (1/2)[\mathrm{N}(-\|m\|, \sigma^2) + \mathrm{N}(\|m\|, \sigma^2)]$ and so $\nu(\mathsf{H}_r \setminus \mathsf{H}) = \nu_1([0, \|m\|/2\sigma])$. On the other hand we have that $\min\{g_{\#}\mu_p(\mathsf{H}), \nu(\mathsf{H})\} = \nu(\mathsf{H})$ and $g_{\#}\mu_p(\mathsf{H}) \geq 1/2$ since $g_{\#}\mu_p(\mathsf{H}) \geq \nu(\mathsf{H})$. Hence it follows that

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \Phi(r/\mathrm{Lip}(g)) - 1/2 - \nu_1([0, \|m\|/2\sigma]) \ .$$

Now we suppose that $g_{\#}\mu_p(\mathsf{H}) \leq \nu(\mathsf{H})$: we then set $\mathsf{H}_2 = (0, +\infty] \times \mathbb{R}^{d-1}$. Since $g_{\#}\mu_p(\mathsf{H}) \leq 1/2$, we get that $g_{\#}\mu_p(\mathsf{H}_2) \geq 1/2$ and so $g_{\#}\mu_p(\mathsf{H}_2) \geq \nu(\mathsf{H}_2)$. Hence we retrieve the previous case and so it follows that

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \Phi(r/\mathrm{Lip}(g)) - 1/2 - \nu_1([-\|m\|/2\sigma, 0]) \ .$$

Since $\nu_1([-\|m\|/2\sigma, 0]) = \nu_1([0, \|m\|/2\sigma])$, we get in both cases

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \Phi(r/\mathrm{Lip}(g)) - 1/2 - \nu_1([0, \|m\|/2\sigma]) \ .$$

Now we derive the value of $\nu_1([0, \|m\|/2\sigma])$:

$$\begin{aligned} \nu_1([0, \|m\|/2\sigma]) &= (1/2) \int_0^{m/2\sigma} (2\pi\sigma^2)^{-1/2} \exp[-(x+m)^2/2\sigma^2]dx \\ &\quad + (1/2) \int_0^{m/2\sigma} (2\pi\sigma^2)^{-1/2} \exp[-(x-m)^2/2\sigma^2]dx \\ &= (1/2) \int_{-m/2\sigma}^{m/2\sigma} (2\pi\sigma^2)^{-1/2} \exp[-(x+m)^2/2\sigma^2]dx \\ &= (1/2) \int_{\|m\|(2\sigma-1)/2\sigma^2}^{\|m\|(2\sigma+1)/2\sigma^2} \varphi(x)dx \ , \end{aligned}$$

which concludes the proof.

S3.7 Proof of Theorem 7

Let $\mathsf{A} \in \mathcal{B}(\mathbb{R}^d)$, $r > 0$ and $\zeta > 0$. We set for any $x \in \mathbb{R}^d$ $f(x) = \zeta \chi_{\mathsf{A}_r \setminus \mathsf{A}}(x)$, where χ_{A} denotes the characteristic function of the set A. Since f is bounded, it follows that

$$\begin{aligned} d_{KL}(g_\# \mu_p || \nu) &\geq \int_{\mathbb{R}^d} f(x) d g_\# \mu_p(x) - \log\left(\int_{\mathbb{R}^d} e^{f(x)} d\nu(x)\right) \\ &\geq \zeta g_\# \mu_p(\mathsf{A}_r \setminus \mathsf{A}) - \log\left(1 + (e^\zeta - 1)\nu(\mathsf{A}_r \setminus \mathsf{A})\right) . \end{aligned}$$

Using Theorem 1, we get

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

Thus we get

$$d_{\mathrm{KL}}(g_\# \mu_p || \nu) \geq \sup\{J(\zeta, \mathsf{A}, r) \,:\, \zeta \in \mathbb{R}, \mathsf{A} \in \mathcal{B}(\mathbb{R}^d), r > 0\} ,$$

where the functional J is defined by

$$\begin{aligned} J(\zeta, \mathsf{A}, r) = \zeta &\left(\Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_\# \mu_p(\mathsf{A}))\right) - g_\# \mu_p(\mathsf{A})\right) \\ &- \log\left(1 + (e^\zeta - 1)\nu(\mathsf{A}_r \setminus \mathsf{A})\right) . \end{aligned}$$

Differentiating J with respect to ζ, we get that

$$\nabla_\zeta J(\zeta, \mathsf{A}, r) = \beta_g(\mathsf{A}, r) - (e^\zeta \nu(\mathsf{A}_r \setminus \mathsf{A}))/(1 + (e^\zeta - 1)\nu(\mathsf{A}_r \setminus \mathsf{A})) ,$$

where $\beta_g(\mathsf{A}, r) = \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_\# \mu_p(\mathsf{A}))\right) - g_\# \mu_p(\mathsf{A})$. Applying the first order condition, we get that:

$$\zeta^* = \log[\beta_g(\mathsf{A}, r)(1 - \nu(\mathsf{A}_r \setminus \mathsf{A}))] - \log[\nu(\mathsf{A}_r \setminus \mathsf{A})(1 - \beta_g(\mathsf{A}, r))] .$$

By re-injecting the value of ζ^*, we get

$$\begin{aligned} \zeta^* \beta_g(\mathsf{A}, r) - \log\left(1 + (e^{\zeta^*} - 1)\nu(\mathsf{A}_r \setminus \mathsf{A})\right) &= \beta_g(\mathsf{A}, r) \log\left(\tfrac{\beta_g(\mathsf{A}, r)(1 - \nu(\mathsf{A}_r \setminus \mathsf{A}))}{\nu(\mathsf{A}_r \setminus \mathsf{A})(1 - \beta_g(\mathsf{A}, r))}\right) \\ &\quad - \log\left(\tfrac{1 - \nu(\mathsf{A}_r \setminus \mathsf{A})}{1 - \beta_g(\mathsf{A}, r)}\right) \\ &= \beta_g(\mathsf{A}, r) \log\left(\tfrac{\beta_g(\mathsf{A}, r)}{\nu(\mathsf{A}_r \setminus \mathsf{A})}\right) \\ &\quad + (1 - \beta_g(\mathsf{A}, r)) \log\left(\tfrac{1 - \beta_g(\mathsf{A}, r)}{1 - \nu(\mathsf{A}_r \setminus \mathsf{A})}\right) , \end{aligned}$$

which concludes the proof.

S3.8 Proof of Corollary 8

As previously, we prove the corollary when $\nu = (1/2)[\mathrm{N}(-m, \sigma^2 \mathrm{Id}_d) + \mathrm{N}(m, \sigma^2 \mathrm{Id}_d)]$ since the problem can always be reduced to that case by translation and setting $m = (m_2 - m_1)/2$. Since the problem is invariant by rotation, we can assume without any loss of generality that $m = (\|m\|, 0, \ldots, 0)$. Furthermore, observe that the half-space $\{(m_2 - m_1)^T (x - (m_1 + m_2)/2) \leq 0 \,:\, x \in \mathbb{R}^d\}$ becomes $(-\infty, 0] \times \mathbb{R}^{d-1}$ in that case, and that the condition $\lambda \in (0, 1/2]$ is indeed non-restrictive since the problem is invariant by rotation. We set as before $\mathsf{H} = (-\infty, 0] \times \mathbb{R}^{d-1}$ and $r = \|m\|/2\sigma$.

Applying Theorem 7, we get

$$d_{\mathrm{KL}}(g_\# \mu_p || \nu) \geq \beta_g(\mathsf{H}, r) \log\left(\tfrac{\beta_g(\mathsf{H}, r)}{\nu(\mathsf{H}_r \setminus \mathsf{H})}\right) + (1 - \beta_g(\mathsf{H}, r)) \log\left(\tfrac{1 - \beta_g(\mathsf{H}, r)}{1 - \nu(\mathsf{H}_r \setminus \mathsf{H})}\right) .$$

On one hand, we get

$$\begin{aligned} \beta_g(\mathsf{H}, r) &= \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_\# \mu_p(\mathsf{H})\right) - g_\# \mu_p(\mathsf{H}) \\ &= \Phi\left(r/\mathrm{Lip}(g) + \Phi^{-1}(g_\# \mu_p(\mathsf{H})\right) - \Phi\left(\Phi^{-1}(g_\# \mu_p(\mathsf{H})\right) \\ &= \int_{\Phi^{-1}(\lambda)}^{\|m\|/2\sigma \mathrm{Lip}(g) + \Phi^{-1}(\lambda)} \varphi(t) \mathrm{d}t \\ &= \int_{-\Phi^{-1}(1-\lambda)}^{\|m\|/2\sigma \mathrm{Lip}(g) - \Phi^{-1}(1-\lambda)} \varphi(t) \mathrm{d}t , \end{aligned}$$

noting $\lambda = g_{\#}\mu_p(\mathsf{H})$. We replaced $\Phi^{-1}(\lambda)$ by $-\Phi^{-1}(1-\lambda)$ in order to emphasize that $\Phi^{-1}(\lambda) \leq 0$ since $\lambda \leq 1/2$. Observe that if we supposed $\lambda \geq 1/2$, we would have $\beta_g(\mathsf{H}^c, r) \geq \beta_g(\mathsf{H}, r)$ and so the bound that we would have found by reasoning on H would have been sub-optimal. On the other hand, observing as before that $\nu = \nu_1 \otimes \mathrm{N}(0, \sigma^2 \operatorname{Id}_{d-1})$, where $\nu_1 = (1/2)[\mathrm{N}(-\|m\|, \sigma^2) + \mathrm{N}(\|m\|, \sigma^2)]$, we get that

$$\begin{aligned}\nu(\mathsf{H}_r \setminus \mathsf{H}) &= \nu_1([0, \|m\|/2\sigma]) \\ &= (1/2) \int_{\|m\|(2\sigma-1)/2\sigma^2}^{\|m\|(2\sigma+1)/2\sigma^2} \varphi(t)\mathrm{d}t \ ,\end{aligned}$$

which concludes the proof.

S4 Additional theoretical result

In this section we derive a generalization of Corollary 5 when ν is a distribution whose support is composed of more than two disconnected manifolds.

Corollary S11. *Let ν be a measure on $\mathbb{R}^d$ on N disconnected manifolds $(\mathsf{M}_1, \dots, \mathsf{M}_N)$, and let $g : \mathbb{R}^p \to \mathbb{R}^d$ be a Lipschitz function. Then,*

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \max_{I \subset [\![1,N]\!]} \int_{\Phi^{-1}(\lambda)}^{d(\bigsqcup_{i\in I} \mathsf{M}_i, \bigsqcup_{j\in[\![1,N]\!]\setminus I} \mathsf{M}_j)/2\mathrm{Lip}(g)+\Phi^{-1}(\lambda)} \varphi(t)dt \ ,$$

where for $\mathsf{A}, \mathsf{B} \in \mathcal{B}(\mathbb{R}^d)$, $d(\mathsf{A}, \mathsf{B}) = \inf\{\|a-b\| \ : \ a \in \mathsf{A}, b \in \mathsf{B}\}$, and $\lambda = \nu\left(\bigsqcup_{i\in I} \mathsf{M}_i\right)$ if $\nu\left(\bigsqcup_{i\in I} \mathsf{M}_i\right) \geq 1/2$ and $\lambda = 1 - \nu\left(\bigsqcup_{i\in I} \mathsf{M}_i\right)$ otherwise.

Proof. Let $I \subset [\![1, N]\!]$. First, we suppose that $\nu\left(\bigsqcup_{i\in I} \mathsf{M}_i\right) \geq 1/2$. Since ν can be seen as a bi-modal distribution on the two disconnected sets $\bigsqcup_{i\in I} \mathsf{M}_i$ and $\bigsqcup_{j\in[\![1,N]\!]\setminus I} \mathsf{M}_j$, we can apply Corollary 5. Thus we get

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) \geq \int_{\Phi^{-1}(\lambda)}^{d(\bigsqcup_{i\in I} \mathsf{M}_i, \bigsqcup_{j\in[\![1,N]\!]\setminus I} \mathsf{M}_j)/2\mathrm{Lip}(g)+\Phi^{-1}(\lambda)} \varphi(t)dt \ .$$

If $\nu\left(\bigsqcup_{i\in I} \mathsf{M}_i\right) \leq 1/2$, we can still apply Corollary 5 by interchanging the roles of $\bigsqcup_{i\in I} \mathsf{M}_i$ and $\bigsqcup_{j\in[\![1,N]\!]\setminus I} \mathsf{M}_j$, thus we get also Inequality (S4) in that case, which concludes the proof. □

S5 Experimental details

We detail our experiments in dimension 1 in Appendix S5.1. In Appendix S5.2, we give details on our experiment on the synthetic mixture of two Gaussians derived from MNIST. Finally, we detail the experiment on the subset of all 3 and 7 of MNIST in Appendix S5.3. We trained our models using 2 NVIDIA Titan Xp from the proprietary server of our institution with an estimated total training time of approximately 175 GPU hours. Code is available here [3].

S5.1 Univariate case

In the univariate case we use a simple 3-layer Multi Layer Perceptron (MLP) of shape $(1, 128, 256, 1)$ as decoder for the VAE and as generator for the GAN. The network has a total of 33537 learnable parameters. The score network uses also a a 3-layer MLP block, this time of shape $(1, 96, 196, 1)$, in which at each layer is injected the noise information transformed by a positional encoding (Vaswani et al., 2017) and then by another MLP block size $(16, 32, 64)$, see Figure S2. The score network has a total of 34665 learnable parameters. In all three models, we use LeakyReLU (Maas et al., 2013) as

[3] https://github.com/AntoineSalmona/Push-forward-Generative-Models

non-linearity with a negative slope of 0.2. The three models are trained during 400 epochs with a batch size of 1000 using ADAM (Kingma and Ba, 2015) with a momentum of 0.9 and a learning rate of 10^{-4}. In the following, we give more specific details for each model.

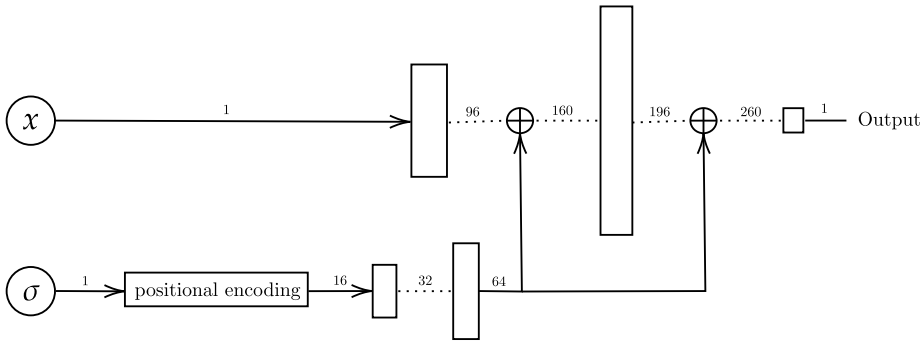

Figure S2: Architecture of the score network used for the univariate experiments. The "positional encoding" block applies the sine transform described in Vaswani et al. (2017). $\bigoplus$ corresponds to concatenation, the vertical blocks correspond to the fully connected layers and the numbers over the arrows correspond to the size of the vectors.

Variational autoencoder. We use the vanilla VAE model as described in Kingma and Welling (2014). In the following, we denote θ and ϕ the respective parameters of the decoder and the encoder. The decoder f_ϕ is composed of an MLP block of size $(1, 256, 128)$ followed by two parallel fully connected layers of shape $(128, 1)$ which gives two outputs $f_{1\phi}(x)$ and $f_{2\phi}(x)$. Then the input z of the decoder g_θ is obtained by the so-called reparametrization trick, which consists in sampling $z \sim q_\phi^{z|x}$, where $q_\phi^{z|x} = \mathrm{N}(f_{1\phi}(x), \exp[f_{2\phi}(x)])$. During training, the model minimizes the following loss function:

$$\mathcal{L}_{\mathrm{VAE}}(\theta, \phi) = \mathbb{E}_{x\sim\nu}[\mathrm{ELBO}_{\theta,\phi}(x, q_\phi^{z|x}, p_\theta^{x|z})] \ ,$$

where $p_\theta^{x|z} = \mathrm{N}(g_\theta(z), c^2 \, \mathrm{Id}_d)$ and ELBO is the Evidence Lower Bound (Blei et al., 2017), defined as follows:

$$\mathrm{ELBO}_{\theta,\phi}(x, q_\phi^{z|x}, p_\theta^{x|z}) = \mathbb{E}_{z\sim q_\phi^{z|x}}[\log(p_\theta(x|z))] - d_{\mathrm{KL}}(q_\phi^{z|x} || \mathrm{N}(0, \mathrm{Id}_p)) \ .$$

The standard deviation c in $p_\theta^{x|z}$ is an hyperparameter of the model. For our experiments, we observed that $c = 0.1$ gave good results.

Generative adversarial network. As for the VAE, we use the vanilla GAN model as described in Goodfellow et al. (2014). The discriminator is 4-layer MLP of shape $(1, 512, 256, 128, 1)$ with spectral normalization (Miyato et al., 2018) in order to reduce as much as possible mode collapse. We train the model using the vanilla adversarial loss, that the discriminator d_ϕ tries to maximize and that the generator g_θ tries to minimize

$$\mathcal{L}_{\mathrm{GAN}}(\theta, \phi) = \mathbb{E}_{x\sim\nu}[\log(d_\phi(x)] + \mathbb{E}_{z\sim\mathrm{N}(0,\mathrm{Id}_p)}[\log(1 - d_\phi(g_\theta(z)))] \ .$$

We also tried with the hinge version of the adversarial loss, as proposed in Lim and Ye (2017) and Tran et al. (2017) and we obtained similar results.

Score-based generative modeling. Our diffusion model is similar to the model introduced by Song and Ermon (2019). The neural network s_θ learns to approximate, for a given x and a given σ, the score $\nabla_x p_\nu(x, \sigma)$ of the data distribution convoluted with a Gaussian distribution of standard deviation σ. This is done by first defining a geometrical progression $\{\sigma_i\}_{i=1}^L$ where $L = 10$ and where the ratio is chosen such that $\sigma_L \approx 0.01$, and then minimizing the Fischer divergence (Vincent, 2011)

$$\mathcal{L}_{\mathrm{SGM}}(\theta) = \mathbb{E}_{\sigma\sim 1/L \sum \delta_{\sigma_i}} \left[\sigma^2 \mathbb{E}_{x\sim\nu} \left[\mathbb{E}_{y\sim\mathrm{N}(x,\sigma^2\,\mathrm{Id}_d)} \left[\left\| s_\theta(y, \sigma) + (y - x)/\sigma^2 \right\|^2 \right] \right] \right] \ .$$

Then, in order to generate data, we use an annealed Langevin dynamic scheme as defined in Song and Ermon (2019). In the Langevin dynamic, we set the step size to 2×10^{-5} and the number of step for each value of σ to 100 as in Song and Ermon (2019).

Influence of generator depth. For this experiment, we increase the number of layers of the VAE decoder and the GAN generator from 2 to 6. At each new layer, we double the number of neurons at the previous layer. For instance, the generative network with 2 layers is thus an MLP of shape $(1, 128, 1)$ and the one with 6 layers is an MLP of shape $(1, 128, 256, 512, 1024, 2048, 1)$. Specifically to the GAN model, we also increase the number of layers in the discriminator in order to keep the dynamic between this latter and the generator balanced. As in the 3-layers case, the discriminator is one layer deeper than the generator. For instance, the discriminator associated to the generator with 2 layers is an MLP of shape $(1, 256, 128, 1)$.

Influence of generator architecture. For this experiment, we use a feed-forward MLP of shape $(1, 256, 256, 256, 1)$ as backbone. Then we add two additive pre-activation skip-connections of type "resnet" between the first and the second hidden layers and between the second and the third hidden layers. Finally, we replace the two previous additive skip-connections of type "resnet" by concatenation pre-activation skip-connections of type "densenet".

S5.2 Synthetic mixture of Gaussians on MNIST

Models details. We adapt our three models to MNIST, changing mainly the networks architectures and making small modifications that we describe in what follows. We base the architecture of the GAN and the VAE on DCGAN (Radford et al., 2015), using the generator as decoder and the discriminator as encoder for our VAE. This is done by doubling the last layer of the discriminator in order that the VAE encoder has two outputs as in the univariate case. For the GAN model, we replaced the convolutional discriminator by a simple MLP of shape $(784, 512, 256, 128, 1)$ because the dynamic between the generator and the discriminator seemed unbalanced otherwise. We also update our GAN model using some features of SAGAN (Zhang et al., 2019): applying spectral normalization on the discriminator and using the unconditional hinge version of the adversarial loss (Lim and Ye, 2017; Tran et al., 2017):

$$\mathcal{L}_{\text{GAN}}^{d_\phi} = -\mathbb{E}_{x\sim\nu}[\min\{0, -1 + d_\phi(x)\}] - \mathbb{E}_{z\sim\text{N}(0,\text{Id}_p)}[\min\{0, -1 - d_\phi(g_{\theta(z)}\}] \ ,$$
$$\mathcal{L}_{\text{GAN}}^{g_\theta} = -\mathbb{E}_{z\sim\text{N}(0,\text{Id}_p)}[d_\phi(g_\theta(z))] \ .$$

Such loss function is equivalent to minimize the Kullback-Leibler divergence between the generated distribution and the data distribution. The VAE decoder and the GAN generator have 1713088 learnable parameters. For the score network architecture, we use the vanilla U-Net architecture (Ronneberger et al., 2015) in which we double the number of channels at each layer, we add group normalization (Wu and He, 2018) after each convolution and we replace the ReLU non-linearies by SiLU (Elfwing et al., 2018). As in the univariate case, we use positional encoding (Vaswani et al., 2017) followed by a MLP block of shape $(1, 16, 32)$ to incorporate the noise information at each layer. The score network has 1607392 learnable parameters. For inference, we use the same Langevin dynamic scheme as above with the same hyperparameters as in the univariate case. The three models are trained during 100 epochs with a batch size of 128 using ADAM with a momentum of 0.9 and a learning rate of 2×10^{-4}.

Additional details. The histograms of projection on the line passing through the mean of each Gaussians are obtained using 20000 generated samples. To assign a color to each bin of the histograms, we train a simple MLP of shape $(784, 1024, 50, 10)$ as classifier on MNIST. The classifier is trained during 10 epochs using again ADAM with a momentum of 0.9 and a learning rate of 2×10^{-4} and reaches an accuracy of 0.98 on the test set.

S5.3 Subset of MNIST

Models details. Since the dataset is more complex than before, we use bigger models. For the score network, we use the architecture defined in Ho et al. (2020), in which we set the number of channels to 64 instead to 128 and we remove the self attention layers (Wang et al., 2018) for computational resource purposes. The score network has 6072065 learnable parameters. Again, we use an annealed Langevin dynamic scheme for inference with the same hyperparameters as before. For the VAE and the GAN, we use the same architecture as before, using this time the convolutional discriminator of DCGAN, and quadrupling the number of channels at each layer. This is mainly done in order to scale the generator/decoder to the score network. Hence the VAE decoder/GAN generator has 7151104

learnable parameters. We train all three models during 600 epochs with a batch size of 128 using ADAM with a momentum of 0.9 and a learning rate of 2×10^{-4}.

Additional details. We use the deep Wasserstein embedding proposed by Courty et al. (2018) in order to visualize histograms of projection in the Wasserstein space. We use the exact same network architecture and the same training procedure that in Courty et al. (2018): first, one million pairs of digits of MNIST are chosen randomly, in which 700000 are kept for the training set, 200000 for the test set, and 100000 for the validation set. We normalize each image in order to consider it as a two-dimensional distribution and we compute the 1-Wasserstein distance for each pair. Then, we train an autoencoder in a supervised manner in a way that the images at output of the autoencoder are close to the images in input, and that the euclidean distance between two vectors in the latent space is close to the 1-Wassertein distance between the two corresponding images of MNIST. As in Courty et al. (2018), the latent Wasserstein space is of dimension 50 and the autoencoder is trained during 100 epochs with a batch size of 100 and with an early stopping criterion. Again, we use ADAM with a momentum of 0.9 and a learning rate of 10^{-3}. We use the same classifier as before to assign color to each bin of the histograms. Finally, the histograms of projection on the line passing through the deep Wasserstein barycenters of all 3 and 7 are obtained using 20000 generated samples.

S6 Additional experimental results

In the following, we provide additional experimental results. First, we compare estimates of the bounds of Theorem 4, Corollary 6, and Theorem 7 to estimates of the total variation distance and the Kullback-Leibler divergence in the univariate case. Then we study the possible correlation between the size of the score network and the tendency of the score-based model to generate unbalanced modes. Finally, we provide additional visualizations of histograms of generated distributions for the univariate case and generated samples for the experiments on MNIST.

S6.1 Bounds on TV distance and KL divergence in the univariate case

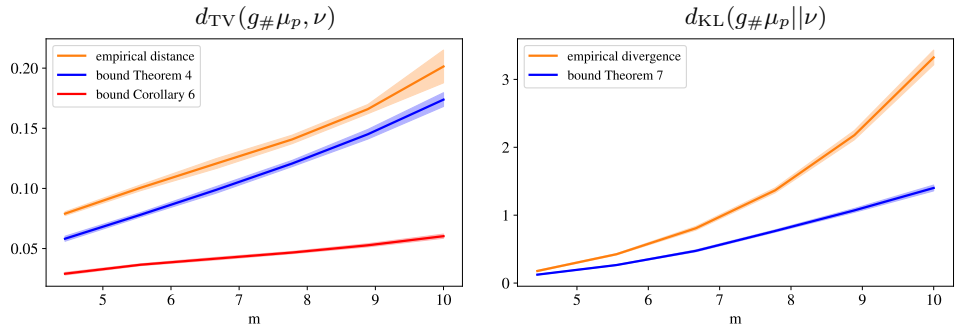

Figure S3: total variation distance (left) and Kullback-Leibler divergence (right) for the VAE (in orange) and estimates of the respective lower bounds from Theorem 4 and Theorem 7 in blue. The lower bound of Corollary 6 is also plotted in red for the total variation. The experiments are averaged over 10 runs and the colored bands correspond to +/- the standard deviation.

In this experiment, we compare estimates of the bounds of Theorem 4, Corollary 6, and Theorem 7 to estimates of the total variation distance and the Kullback-Leibler divergence. We only provide results for the VAE since the bounds are not interesting for the GAN since they are consequences of interpolation between modes due to a small Lipschitz constant of the generative network. Yet this latter in the GAN case achieves a large Lipschitz constant so does not interpolate significantly. To estimate empirically the total variation distance and the Kullback-Leibler divergence, we used their respective analytical formula

$$d_{\mathrm{TV}}(g_{\#}\mu_p, \nu) = (1/2)\int_{\mathbb{R}} |p_{g_{\#}\mu_p}(x) - p_\nu(x)| dx \ ,$$
$$d_{\mathrm{KL}}(g_{\#}\mu_p || \nu) = \int_{\mathbb{R}} p_{g_{\#}\mu_p}(x) \log\left(p_{g_{\#}\mu_p}(x)/p_\nu(x)\right) dx \ ,$$

where $p_{g_{\#}\mu_p}$ and p_ν are the respective densities of $g_{\#}\mu_p$ and ν. In order to estimate the lower bounds of Theorem 4 and Theorem 7, we set A of the form $(-\infty, -r/2]$ and we perform a grid search on r.

In Figure S3, we can observe that the estimates of the bounds provided by Theorem 4 and Theorem 7 are not tights. This is possibly because we selected a sub-optimal A but it most likely follows from the fact that the bounds don't take into account that $g_{\#}\mu_p$ has automatically less mass on the modes than ν since a significant amount of its total mass is between them. One can also observe that the explicit lower bound of Corollary 6 is much smaller than the bound of Theorem 4. This can be explained by the facts that $\|m\|/2\sigma$ is probably a sub-optimal choice of r and that the bound of Corollary 6 minimizes the interpolation between modes over all the mappings with Lipschitz constant $\mathrm{Lip}(g)$, regardless whether these mappings approximate well ν on its modes or not. Since there is less interpolation if the modes are unbalanced (see Section 3.2), it is likely that the mappings g such that $g_{\#}\mu_p$ is unbalanced are affecting the value of this bound in a bad way.

S6.2 Additional examples

S6.2.1 Univariate histograms

We provide additional visualizations of histograms of generated data with the three models for various values of m.

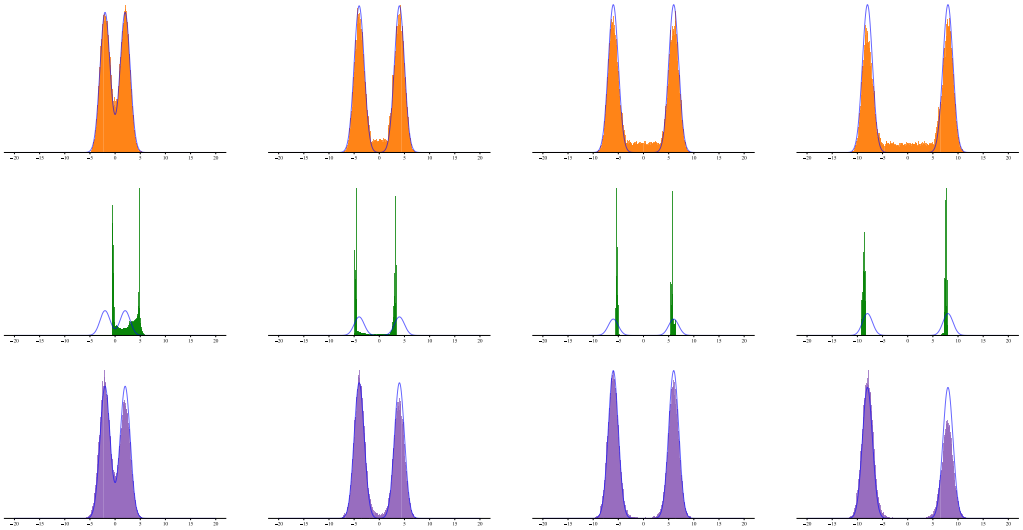

Figure S4: Histograms of distributions generated with VAE (top, in orange), GAN (middle, in green), and with SGM (bottom, in purple) for $m = 2$, $m = 4$, $m = 6$ and $m = 8$. biblographie stylebiblographie stylebiblographie style The data distribution densities are plotted in blue.

We can observe that the score-based model already generates unbalanced modes, but the phenomenon is globally less visible than in higher dimensions. Secondly, we provide additional visualizations of histograms of generated data with GANs trained with an additional gradient penalty term in the generator loss for various values of $L \approx \mathrm{Lip}(g)$.

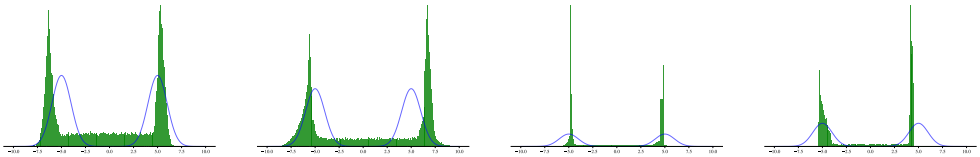

Figure S5: histograms of distributions generated with GANs with with gradient penalty for $\mathrm{Lip}(g) \approx L = 11$, $\mathrm{Lip}(g) \approx L = 15$, $\mathrm{Lip}(g) \approx L = 19$ and $\mathrm{Lip}(g) \approx L = 23$. The data distribution densities are plotted in blue.

S6.2.2 Visualization of generated data

Finally, we show randomly chosen generated samples with VAE, GAN and SGM on the synthetic mixture of Gaussian on MNIST and the subset of all 3 and 7 of MNIST.

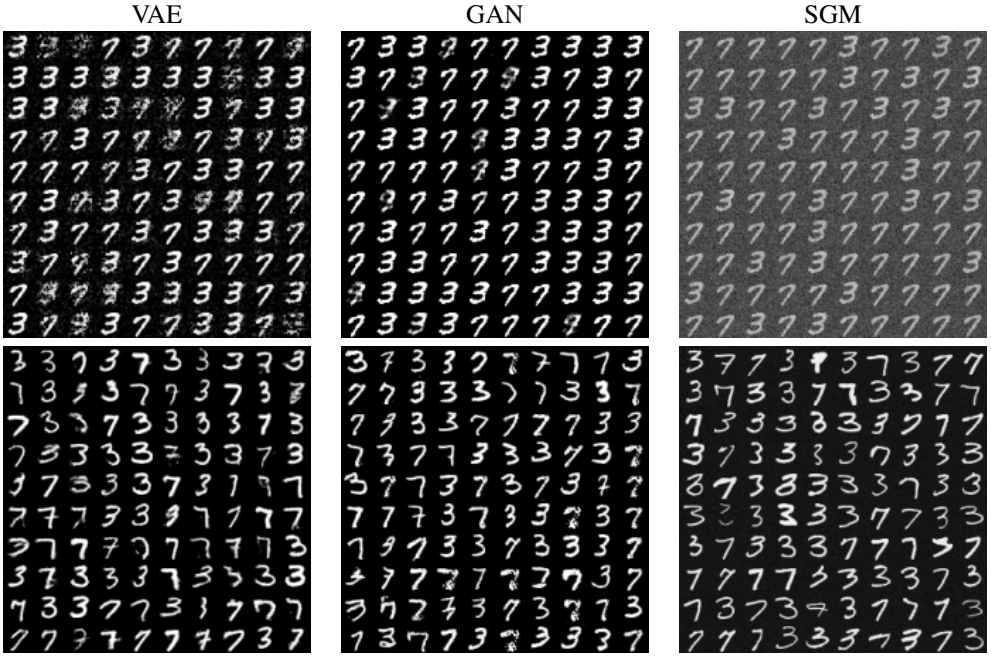

Figure S6: Generated samples with VAE, GAN and SGM on the synthetic mixture of Gaussian on MNIST (top) and the subset of all 3 and 7 of MNIST (bottom). The samples have been randomly chosen.