On the Universality of Volume-Preserving and Coupling-Based Normalizing Flows

Felix Draxler [1] **Stefan Wahl** [1] **Christoph Schnörr** [1] **Ullrich Köthe** [1]

Abstract

We present a novel theoretical framework for understanding the expressive power of normalizing flows. Despite their prevalence in scientific applications, a comprehensive understanding of flows remains elusive due to their restricted architectures. Existing theorems fall short as they require the use of arbitrarily ill-conditioned neural networks, limiting practical applicability. We propose a distributional universality theorem for well-conditioned coupling-based normalizing flows such as RealNVP (Dinh et al., 2017). In addition, we show that volume-preserving normalizing flows are not universal, what distribution they learn instead, and how to fix their expressivity. Our results support the general wisdom that affine and related couplings are expressive and in general outperform volume-preserving flows, bridging a gap between empirical results and theoretical understanding.

1. Introduction

Density estimation and generative modeling of complex distributions is a fundamental problem in statistics and machine learning, with applications ranging from computer vision (Kingma & Dhariwal, 2018) to thermodynamic systems (Noé et al., 2019; Albergo et al., 2019; Nicoli et al., 2021) and uncertainty quantification (Ardizzone et al., 2018b).

Normalizing flows are a class of generative models that learn a probability density $p_\theta(x)$ from samples $x \sim p(x)$ or from a potentially unnormalized ground truth density $\hat{p}(x) \propto p(x)$. They are implemented by transporting a simple multivariate base density such as the standard normal via a learned invertible function to the distribution of interest.

Constructing flexible and tractable invertible neural networks is nontrivial and a significant body of work has developed a plethora of architectures, see (Kobyzev et al., 2021) for an overview. The evidence for which architecture to choose in practice is mostly limited to empirical results, however. In this work, we prove rigorous results regarding the universality of the families of volume-preserving and coupling-based normalizing flows.

First, we consider volume-preserving flows such as (Dinh et al., 2015; Sorrenson et al., 2019; Toth et al., 2020). Volume-preservation can be useful in certain applications, such as disentanglement (Sorrenson et al., 2019) or learning distributions with controllable temperature $p(x|T) \propto (\hat{p}(x))^T$ (Dibak et al., 2022). However, we show that they are not universal and learn a biased distribution in practice. We also provide a simple solution to restore universality, by adding a single one-dimensional non-volume-preserving layer.

Second, we improve on the universality theory of coupling-based normalizing flows (not preserving volume). These flows are a particularly efficient variant of parameterizing invertible neural networks, with fast training and inference. Despite their seemingly strong architectural constraints, in practice even the simple affine coupling-based normalizing flow (Dinh et al., 2017) can learn high-dimensional distributions such as images (Kingma & Dhariwal, 2018).

Theoretical explanations for this architecture's ability to fit complex distributions are limited. Existing proofs make assumptions that are not valid in practice, as the involved constructions rely on ill-conditioned neural networks (Teshima et al., 2020a; Koehler et al., 2021) or construct a volume-preserving flow (Koehler et al., 2021). We introduce a new proof for the distributional universality of coupling-based normalizing flows that does not require ill-conditioned neural networks to converge. This proof is constructive, showing that training affine coupling blocks sequentially converges to the correct target (compare Figure 1).

In summary, we contribute:

- We show for the first time that volume-preserving flows are not universal, derive what distribution they converge to instead and provide simple fixes for their shortcomings in Section 4.

[1]Heidelberg University, Germany. Correspondence to: Felix Draxler <felix.draxler@iwr.uni-heidelberg.de>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Figure 1. Our universality proof constructs a normalizing flow by iteratively adding affine coupling blocks. We illustrate this by constructing such a flow from topologically challenging toy data. Starting with the input $p(x)$, each block first rotates the latent distribution $p_{n-1}(z)$ from the previous step (first column), then applies an affine coupling layer that transforms the active dimensions to zero mean and unit variance for each passive coordinate (second column). The resulting latent distribution $p_n(z)$ converges step by step (third column) to a standard normal distribution, until the learned additional layers essentially learn the identity (last row). The learned data distribution $p_n(x)$ converges in parallel (right).

- We show that whenever the target distribution is not perfectly learned, there is an affine coupling block that reduces the loss (Section 5.2).
- We use this result to give a new universality proof for coupling-based normalizing flows that is not volume-preserving, considers the full support of the distribution, and is not ill-conditioned in Section 5.3.

Our results validate insights previously observed only empirically: Affine coupling blocks are an effective foundation for normalizing flows, and volume-preserving flows have limited expressive power. We also show that the most recent distributional universality proof for affine coupling-based normalizing flows by Koehler et al. (2021) constructs such a volume-preserving flow in Section 5.1.

2. Related Work

Normalizing flows are a class of generative models based on invertible neural networks (Rezende & Mohamed, 2015). We focus on the analytical expressivity of volume-preserving and coupling-based normalizing flows, see Section 3.

Table 1. Our construction of a universal coupling flow overcomes important limitations of the previous work on arbitrary input $p(x)$ (Teshima et al., 2020a; Koehler et al., 2021): We use well-conditioned coupling blocks, consider convergence on the full space and allow variable volume change $|f'(x)| \neq \text{const}$, which is necessary for universality in KL divergence by Theorem 4.2.

	Teshima et al.	Koehler et al.	Thm. 5.4
Well-conditioned	✗	✗	✓
Variable $\lvert f'(x)\rvert$	✓	✗	✓
Global support	✗	✗	✓

That coupling-based normalizing flows work well in practice despite their restricted architecture has sparked the interest of several papers analyzing their distributional universality, i.e. the question whether they can approximate any target distribution to arbitrary precision (see Definition 3.1). Teshima et al. (2020a) showed that coupling flows are universal approximators for invertible functions, which results in distributional universality. Koehler et al. (2021) demonstrated that affine coupling-based normalizing flows can approximate any distribution with arbitrary precision using just three coupling blocks. However, these works rely on ill-conditioned coupling blocks and consider convergence only on a bounded subspace. Our work addresses these limitations and shows that training a normalizing flow layer by layer yields universality. In addition, we find that Koehler et al. (2021) effectively construct a volume-preserving flow, which we show not to be universal in KL divergence, the loss used in practice. This line of work is summarized in Table 1.

Some works show distributional universality of augmented affine coupling-based normalizing flows, which add at least one additional dimension usually filled with exact zeros (Huang et al., 2020; Koehler et al., 2021; Lyu et al., 2022). The problem with adding additional zeros is that the flow is not exactly invertible anymore in the data domain and usually loses tractability of the change of variables formula (Equation (1)). Lee et al. (2021) add i.i.d. Gaussians as additional dimensions, which again allows density estimation, but they only show how to approximate the limited class of log-concave distributions. Our universality proof does not rely on such a construction.

Other theoretical work on the expressivity of normalizing flows considers more expressive invertible neural networks, including SoS polynomial flows, Neural ODEs and Residual Neural Networks (Jaini et al., 2019; Zhang et al., 2020; Teshima et al., 2020b; Ishikawa et al., 2022). Another line of work found that the number of required coupling blocks

is independent of dimension D for Gaussian distributions compared to $O(D)$ Gaussianization blocks that lack couplings between dimensions (Koehler et al., 2021; Draxler et al., 2022; 2023).

Regarding volume-preserving flows, to the best of our knowledge there is no previous work showing their non-universality. Our results also complement the fact that Hamiltonian Monte Carlo (HMC) resamples momenta at every step in order to sample the correct target distribution (Neal, 2011).

3. Background

Normalizing Flows are a class of generative models that represent a distribution $p(x)$ with parameters by learning an invertible function $z = f(x)$ so that the latent codes $z \in \mathbb{R}^D$ obtained from the data $x \in \mathbb{R}^D$ are distributed like a standard normal distribution $p(z) = \mathcal{N}(z; 0, I)$. Via the change of variables formula, see (Kobyzev et al., 2020; Papamakarios et al., 2021; Kruse, 2023) for a review, this invertible function yields an explicit form for the density $p(x)$:

$$p(x) = p(z = f(x)) \left| \det f'(x) \right|, \tag{1}$$

where $f'(x) = \frac{\partial}{\partial x} f(x)$ is the Jacobian matrix of f at x and $|\det f'(x)|$ is its absolute determinant. Note that we consistently denote by $p(\cdot)$ a ground truth target distribution and by $p_\theta(\cdot)$ a learned approximation.

Equation (1) allows easily evaluating the model density at points of interest. Obtaining samples from $p(x)$ can be achieved by sampling from the latent standard normal and applying the inverse $f^{-1}(z)$ of the learned transformation:

$$x = f^{-1}(z) \quad p(x) \text{ for } z \sim p(z). \tag{2}$$

The change of variables formula (Equation (1)) can be used directly to train a normalizing flow. The corresponding loss minimizes the Kullback-Leibler divergence between the true data distribution $p(x)$ and the learned distribution, which can be optimized via a Monte-Carlo estimate of the involved expectation:

$$\begin{aligned} L &= D_{KL}(p(x) \| p_\theta(x)) && (3)\\ &= \mathbb{E}_{x \sim p(x)}[\log p(x) - \log p_\theta(x)] && (4)\\ &= \mathbb{E}_{x \sim p(x)}[-\log p_\theta(x)] + \text{const}. && (5) \end{aligned}$$

This last variant makes clear that minimizing this loss is exactly the same as maximizing the log-likelihood of the training data. For training, the expectation value is approximated using (batches of) training samples $x_{1,\dots,N}$.

In order for Equations (1) and (2) to be useful in practice, $f(x)$ must have (i) a tractable inverse $f^{-1}(z)$ for fast sampling, and (ii) a tractable Jacobian determinant $|\det f'(x)|$ for fast training while (iii) being expressive enough to model complicated distributions. These constraints are nontrivial to fulfill at the same time and significant work has been put into constructing such invertible neural networks, see (Kobyzev et al., 2021).

Coupling-Based Normalizing Flows are an invertible neural network design that lies in a sweet spot of scaling well to large dimensions yet remaining fast to sample from (Draxler et al., 2023) and exhibits a tractable Jacobian determinant. Its basic building block is the coupling layer, which consists of one invertible function $x_i = c(x_i; \theta_i)$ for each dimension, but with a twist: Only the second half of the dimensions $a = x_{D/2+1,\dots,D}$ (active) is changed in a coupling layer, and the parameters of this transformation $\theta = (b)$ are predicted by a function called the conditioner that takes the first half of dimensions $b = x_{1,\dots,D/2}$ (passive) as input:

$$x_i = (f_{\text{cpl}}(x))_i = \begin{cases} b_i & i \le D/2 \\ c(a_{i-D/2}; \theta_{i-D/2}(b)) & \text{else.} \end{cases} \tag{6}$$

In practice, the conditioner of each block is realized by a neural network $\theta_\cdot(b)$ with parameters θ. Calculating the inverse of the coupling layer is easy, as $a = x$ for the passive dimensions. This allows computing the parameters $\theta(b)$ necessary to invert the active half of dimensions:

$$x_i = (f_{\text{cpl}}^{-1}(x))_i = \begin{cases} b_i & i \le D/2 \\ c^{-1}(a_{i-D/2}; \theta_{i-D/2}(b)) & \text{else.} \end{cases} \tag{7}$$

Choosing the right one-dimensional invertible function $c(x; \theta)$ is the subject of active research, see our list in Appendix A.1 and Kobyzev et al. (2021). Many applications use affine-linear functions (Dinh et al., 2017):

$$c(x; s, t) = sx + t, \tag{8}$$

where $s > 0$ and t are predicted by the conditioner $(s, t)(b)$ as a function of the passive dimensions. Especially for smaller-dimensional problems it has proven useful to use more flexible c such as rational-quadratic splines (Durkan et al., 2019b). The positive and negative universality theorems in this paper apply to all coupling architectures we are aware of. At the same time, we give a direct reason for using more expressive couplings, as they can learn the same distributions with fewer layers in Section 5.4.

In order to be expressive, a coupling flow consists of a stack of coupling layers, each with a different active and passive subspace. Varying the subspaces is realized by an additional layer before each coupling which mixes dimensions by a rotation matrix $Q \in SO(D)$:

$$f_{\text{rot}}(x) = Qx, \qquad f_{\text{rot}}^{-1}(x) = Q^T x. \tag{9}$$

The matrix Q can either be chosen to be a hard permutation or any matrix with orthonormal columns, which mixes dimensions linearly. The latter can optionally be learned (Kingma & Dhariwal, 2018). From a representational perspective, these variants are interchangeable because any soft permutation can be represented by a constant number of coupling blocks with hard permutations (Koehler et al., 2021).

A rotation layer together with a coupling layer forms a coupling block:

$$f_{blk}(x) = (f_{cpl} \quad f_{rot})(x) = f_{cpl}(Qx): \tag{10}$$

In the Section 5, we are concerned with what distributions $p(x)$ a potentially deep concatenation of coupling blocks can represent. Taken together, the parameters of the conditioners of the coupling layers together with the rotation matrices Q make up the learnable parameters θ of the entire normalizing flow.

Volume-Preserving Normalizing Flows or sometimes incompressible flows are a variant of normalizing flows that have a constant Jacobian determinant $|f'_\theta(x)| = \text{const}$. This simplifies the change of variables formula in Equation (1) above, where $C = |f'(x)|$:

$$p_\theta(x) = p(z = f_\theta(x))\, C: \tag{11}$$

Volume-preserving flows have been demonstrated to have useful properties in certain applications such as disentanglement (Sorrenson et al., 2019), or temperature-scaling in Boltzmann generators (Dibak et al., 2022) or to preserve volume in physical state-space (Toth et al., 2020). However, we show that the volume-preserving change of variables in Equation (11) does not allow for universal normalizing flows regardless of the architecture in Section 4.

For one-dimensional functions, a constant volume change implies that $f(x) = Cx + t$ is linear. For multivariate functions, $f(x)$ can be nonlinear, only that any volume change in one dimension must be compensated by an inverse volume change in the remaining dimensions. Prominent implementations are nonlinear independent components estimation (NICE) (Dinh et al., 2015), general incompressible-flow networks (GIN) (Sorrenson et al., 2019) or Neural Hamiltonian Flows (NHF) (Toth et al., 2020). For example, GIN realizes volume-preserving coupling blocks by ensuring $\sum_{i=1}^{D/2} \log s_i(b) = \text{const}$. We list common volume-preserving constructions in Appendix A.2. Note that "volume-preserving" is strictly speaking a misnomer when $C \neq 1$, but the term is commonly used also in this more general case. In Lemma B.3, we show that non-unit volume change can be absorbed into a single scaling layer.

Distributional Universality means that a certain class of generative models can represent any distribution $p(x)$. Due to the nature of neural networks, we cannot hope for our generative model to exactly (i.e. exact equality in the mathematical sense) represent $p(x)$. This becomes clear via an analogue in the context of regression: A neural network with ReLU activations always models piecewise linear functions, and as such it can never exactly regress a parabola $y = x^2$. However, for every finite value of > 0 and given more and more linear pieces, it can follow the parabola ever so closer, so that the average distance between x^2 and $f(x)$ vanishes: $\mathbb{E}_{x \sim p(x)}[|x^2 \quad f(x)|^2] < $. To characterize the expressivity of a class of neural networks, it is thus instructive to call a class of networks universal if the error between the model and any target can be reduced arbitrarily.

In terms of representing distributions $p(x)$, the following definition captures universality of a class of model distributions, similar to (Teshima et al., 2020a, Definition 3):

Definition 3.1. A set of probability distributions $\mathcal{P}$ is called a *distributional universal approximator* if for every possible target distribution $p(x)$ there is a sequence of distributions $p_n(x) \in \mathcal{P}$ such that $p_n(x) \overset{n\to\infty}{\to} p(x)$.

The formulation of universality as a convergent series is useful as it (i) captures that the distribution in question $p(x)$ may not lie in $\mathcal{P}$, and (ii) the series index n usually reflects a hyperparameter of the underlying model corresponding to computational requirements (for example, the depth of the network).

We have left the exact definition of the limit "$p_n(x) \overset{n\to\infty}{\to} p(x)$" open as we may want to consider different variations of convergence. The existing literature on affine coupling-based normalizing flows considers weak convergence (Teshima et al., 2020a) respectively convergence in Wasserstein distance (Koehler et al., 2021). Many metrics of convergence have been proposed, see Gibbs & Su (2002) for a systematic overview.

While we consistently state our assumptions, we usually restrict ourselves to data distributions with densities $p(x)$ that are bounded and continuous, and that have finite support and finite moments, which covers distributions of practical interest.

4. Non-Universality of Volume-Preserving Flows

In this section, we consider normalizing flows with constant Jacobian determinant $|f'_\theta(x)| = \text{const}$. We show that volume-preserving flows are not universal in KL divergence. We then propose how universality can be recovered.

From the volume-preserving change-of-variables in Equation (11) we can derive the KL divergence $D_{KL}(p(x) \| p_\theta(x))$

in the special case of $C = 1$:

$$L = \int p(x) \log \frac{p(x)}{p_z(x)} dx \tag{12}$$

$$= H[p(x)] - \int p(x) \log p(z = f(x)) dx. \tag{13}$$

Only the last term depends on f. To derive the minimizer, consider the data $p(x)$ and latent distribution $p(z)$ on a regular grid over $\mathbb{R}^D$ with some spacing $a > 0$. Then, define a volume-preserving flow with $C = 1$ that permutes the grid cells $B_i \mapsto B_{f(i)}$ (within the cells, keep the relative positions). Then, discretize the above integral on the grid by approximating the latent probability by the average density in each cell, that is $p(z) \approx \frac{1}{a^D} p(B_i : z \in B_i)$:

$$\int p(x) \log p(z = f(x)) dx \tag{14}$$

$$\sum_i p(x \in B_{s_x(i)}) \log(p(z \in B_{f(i)})/|a^D|). \tag{15}$$

This is minimized by a bijective $f : \mathbb{N} \to \mathbb{N}$ that permutes the grid cells such that the cell with the highest probability $p(x \in B_i)$ in the data space aligns with the cell with the highest (logarithmic) probability in latent space, and so on:

$$f(i) = s_z(s_x^{-1}(i)); \tag{16}$$

where $s_v(i)$ is a sorting of the grid cells, determined by probability mass $p(v \in B_i)$ for $v = x$ respectively $v = z$.

The following theorem makes the above argument continuous and determines the optimal volume change C:

Theorem 4.1. Given a continuous bounded input density $p(x)$. Then, for any volume-preserving flow $p_z(x)$ with a standard normal latent distribution, the achievable KL divergence is bounded from below:

$$D_{KL}(p(x)\|p_z(x)) \geq D_{KL}(p_*(z)\|\mathcal{N}(0, |\Sigma_{p_*(z)}|^{\frac{1}{D}} I)), \tag{17}$$

where $p_*(z)$ is constructed by decreasingly sorting the probability densities $p(x)$ from the origin with unit volume change, and $\Sigma_{p_*(z)}$ is its covariance matrix. The minimal loss is achieved for $C = |\Sigma_{p_*(z)}|^{\frac{1}{2D}}$.

The optimal $p_*(x)$ and its latent counterpart $p_*(z)$ are constructed by sorting both the data and latent space by density and progressively assigning regions of decreasing density to each other (see proof in Appendix B.1). Figure 2 shows how this optimal distribution $p_*(x)$ differs from the target $p(x)$ for a bimodal toy distribution in 2D.

The following result formalizes that volume-preserving flows are not universal:

Theorem 4.2. The family of normalizing flows with constant Jacobian determinant $|f'(x)| = \text{const}$ is not a universal distribution approximator under KL divergence.

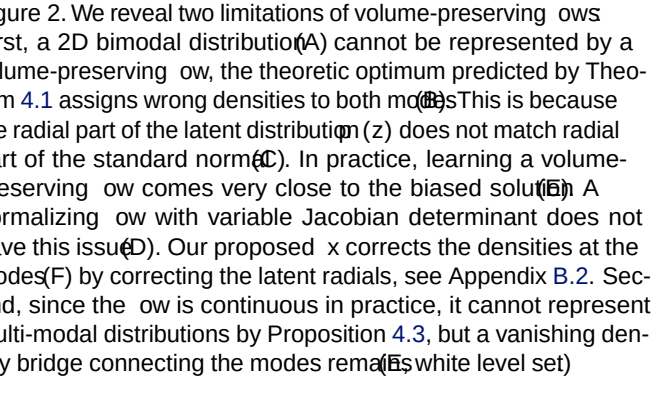

Figure 2. We reveal two limitations of volume-preserving flows: First, a 2D bimodal distribution (A) cannot be represented by a volume-preserving flow, the theoretic optimum predicted by Theorem 4.1 assigns wrong densities to both modes (B). This is because the radial part of the latent distribution $p_*(z)$ does not match radial part of the standard normal (C). In practice, learning a volume-preserving flow comes very close to the biased solution (E). A normalizing flow with variable Jacobian determinant does not have this issue (D). Our proposed fix corrects the densities at the modes (F) by correcting the latent radials, see Appendix B.2. Second, since the flow is continuous in practice, it cannot represent multi-modal distributions by Proposition 4.3, but a vanishing density bridge connecting the modes remains (E, white level set)

The proof constructs a concrete $p(x)$ and shows that the KL is bounded from below by a finite value for every possible value of C (see Appendix B.3).

The construction underlying Theorem 4.1 shows a clear path to construct a universal volume-preserving flow: The best achievable latent distribution $p_*(z)$ (the push-forward of $p(x)$ through f_*) is rotationally symmetric due to the sorting procedure. Now transform both $p_*(z)$ and the target standard normal $p(z)$ into hyperspherical coordinates (r, s). As both distributions are rotationally symmetric, only their radial parts $p_*(r)$ and $p(r)$ need to be matched, which can be achieved via the addition of a single one-dimensional non-volume-preserving transformation (see Appendix B.2). As this fix is one-dimensional, unique, and can be applied after training, we think that it is compatible with retaining beneficial properties of volume-preserving flows.

Figure 2 also shows another shortcoming of volume-preserving flows as they are practically implemented: There is a thin bridge of density between the modes with roughly constant height, so that the lower mode in the ground truth is not a local maximum of the learned density. The reason is that flows are implemented as continuous invertible functions (as opposed to Theorems 4.1 and 4.2, which only require invertibility). This makes the learned distribution

$p_\theta(x)$ inherit the mode structure of the latent $p(z)$:

Proposition 4.3. A normalizing flow $p_\theta(x)$ based on diffeomorphisms $f_\theta(x)$ with constant Jacobian determinant $|f_\theta'(x)| = \text{const}$ has the same number of modes as the latent distribution $p(z)$.

The proof in Appendix B.4 uses that diffeomorphisms map open sets to open sets, and thus the neighborhoods of density maxima in the latent space remain neighborhoods of density maxima in the data space. Note that the thin bridge connecting the modes can be made arbitrarily small by an expressive enough volume-preserving flow, so that the shortcoming in Proposition 4.3 does not manifest in a bias in the KL divergence in addition to Theorem 4.1.

Together, we identify a fundamental limitation for applications based on volume-preserving flows. It explains why RealNVP significantly outperforms NICE in practice (Dinh et al., 2017). Work using volume-preserving flows must take this limited expressivity and the resulting biases in the learned distributions into account. In Section 5.1, we show that this problem also applies to the most recent universality proof for coupling-based normalizing flows by Koehler et al. (2021).

5. Universality of Coupling-Based Normalizing Flows

In this section, we present our improved universality proof for non-volume-preserving coupling-based normalizing flows. It overcomes limitations of previous constructions and relies on the simple idea of iteratively training coupling blocks.

5.1. Problems with Existing Constructions

The existing proofs (Teshima et al., 2020a; Koehler et al., 2021) that affine and more expressive coupling flows are distributional universal approximators come with several limitations. In particular, their constructions use ill-conditioned coupling blocks as the translations $t(b)$ approximate step functions, as noted by Koehler et al. (2021). Also, they give guarantees only on a compact subspace $K \subset \mathbb{R}^D$, and Teshima et al. (2020a) use only one active dimension per coupling. This limits their practical applicability.

In addition, the flow constructed in Koehler et al. (2021) is volume-preserving and thus not universal in KL divergence by our Theorem 4.2. Their proof is technically accurate, but they only show convergence under Wasserstein distance W_2 in (Koehler et al., 2021, Theorem 1), which does not imply convergence in KL (see Appendix C.4).

It is easy to see that their flow is volume-preserving by looking at the scaling functions $s(b)$ in the three affine coupling layers (Equation (8)) they use. They read: $s^{(1)} = 0$ and $s^{(2)} = s^{(3)} = \varepsilon'$. This means that the overall flow has a Jacobian determinant $|f_\theta'| = (\varepsilon'^{0} \varepsilon'^{0})^{\frac{D}{2}}$. This volume change is independent of the input, making the flow volume-preserving.

Also note how the volume change is directly tied to the guaranteed Wasserstein distance, since they guarantee that $W_2(p_\theta(x), p(x)) < \varepsilon$, and the above scalings fulfill $\varepsilon' \to 0$. Thus, the volume change $|f_\theta'|^{\frac{3D}{2}}$ vanishes, and its inverse $|(f_\theta^{-1})'|^{\frac{3D}{2}}$ explodes as ε is reduced, rendering the flow ill-conditioned regardless of the distribution at hand. This is additional to the ill-condition of the translation terms $t(b)$ approximating step functions.

Together, this calls for a new universality guarantee that is based on a new coupling flow construction. Our new construction, presented in the following sections, uses a flow that is neither arbitrarily ill-conditioned nor volume-preserving. Also, it converges globally and considers vanilla affine coupling blocks.

5.2. Affine Coupling Blocks have a Unique Fixed Point

To construct our new universality theorem, we first analyze the effect of a single affine coupling on learning a target distribution $p(x)$. Our main result is that an affine coupling block always reduces the loss when it has not yet perfectly learned the target distribution.

To derive this, consider the pushforward of $p(x)$ through the flow $f_\theta(x)$, that is the latent distribution the flow actually creates by mapping $x \mapsto z = f_\theta(x)$:

$$p_\theta(z) = (f_\theta]\, p(x))(z) = p(x = f_\theta^{-1}(z))\,|(f_\theta^{-1})'(z)|. \tag{18}$$

Now rewrite the loss L in Equation (3) into a form which compares $p_\theta(z)$ to the target latent distribution $p(z)$:

$$\begin{aligned} L &= D_{KL}(p(x) \| p_\theta(x)) && (19)\\ &= \int p(x) \log \frac{p(x)}{p(z = f_\theta(x))\,|f_\theta'(x)|}\,dx && (20)\\ &= \int p(z)\,|f_\theta^{-1\prime}(z)| \log \frac{p(f_\theta^{-1}(z))\,|f_\theta^{-1\prime}(z)|}{p(z)}\,dz && (21)\\ &= D_{KL}(p_\theta(z) \| p(z)). && (22) \end{aligned}$$

This identity shows that the divergence between the model $p_\theta(x)$ and the data $p(x)$ can equally be measured in the latent space, via the KL divergence between the current latent distribution that the model generates as the pushforward $p_\theta(z)$ and the target latent distribution $p(z)$.

Let us now consider what happens if we append one more affine coupling block f_{blk, θ_+} to an existing normalizing flow $f_\theta(x)$, resulting in a flow which we call $p_{[\theta, \theta_+]}(x)$. Let us choose the parameters of the additional coupling block θ_+ such that it maximally reduces the loss without changing

the previous parameters:

$$\min_{+} D_{KL}(p_{[\ _{+}}(z)\|p(z)): \tag{23}$$

Let us formalize the above construction for later use:

De nition 5.1. Given a normalizing ow p (x) on a continuous probability distribution $p(x)$ with nite rst and second moment and $p(x) > 0$ everywhere. Then, we de ne the loss improvement by an af ne coupling block as:

$$\begin{aligned}&{}_{af\,ne}(p\ (z))\\ &:= D_{KL}(p\ (z)\|p(z)) \quad \min_{+} D_{KL}(p_{[\ _{+}}(z)\|p(z)); \end{aligned}\tag{24}$$

where $_{+} = (\ Q; '\ )$ parameterizes a single bi-Lipschitz af ne coupling block whose conditioner neural network has at least two hidden layers of nite width and ReLU activations and $p\ (z) = (\ f\ _{]}\ p)(z)$.

The coupling block is restricted to be bi-Lipschitz in order to be well-conditioned. This means that we can choose $L \geq 1$ such that $L^{-1} < \frac{\|f_{cpl}(x) - f_{cpl}(y)\|}{\|x - y\|} < L$, making both forward and inverse passes through each coupling well-conditioned. Choosing a smaller L will result in more coupling blocks, each more numerically stable. Note that we assume ReLU networks for mathematical convenience, but think that the de nition is equivalent to versions with different activation functions.

Considering this loss improvement, $_{af\,ne}(p\ (z))$ is a useful quantity, since we can show that is directly related to convergence of the ow:

Theorem 5.2. With the de nitions from De nition 5.1:

$$p\ (z) = N(z; 0; I\ ) \quad () \quad {}_{af\,ne}(p\ (z)) = 0: \tag{25}$$

This is a nontrivial result: One might have thought that it is possible to end up in distributions such that an af ne coupling block gets stuck and cannot improve on the loss. Instead, if adding another coupling layer has no effect, the latent distribution has converged to a standard normal. This unique xed point allows using $_{af\,ne}(p\ (z))$ as a convergence metric for our universality theorem in Section 5.3.

In the remainder of this section, we give a sketch of the proof of Theorem 5.2, with technical details moved to Appendix C.1.

We proceed as follows: First, we use an explicit form of the maximal loss improvement $_{af\,ne}(p\ (z))$ for in nitely expressive af ne coupling blocks (Draxler et al., 2020). Then, we show in Lemma 5.3 that convergence of these unrealistic networks is equivalent to convergence of nite ReLU networks. Finally, we show that $_{af\,ne}(p\ (z)) = 0$ implies $p\ (z) = N(z; 0; I\ )$. The other direction is trivial, since by $p\ (z) = N(0; I\ )$, no loss improvement is possible.

If we assume for a moment that neural networks can exactly represent arbitrary continuous functions, then this hypothetical maximal loss improvement was computed by Draxler et al. (2020, Theorem 1). A single af ne coupling block with a xed rotation layer Q, in order to maximally reduce the loss, will standardize the data by normalizing the rst two moments of the active half of dimensions $a = (Qx)_{D=2+1\ ;:::;D}$ conditioned on the passive half of dimensions $b = (Qx)_{1;:::;D}$. The moments before the coupling

$$E_{a_i\,|b}[a_i] = m_i(b); \qquad Var_{a_i\,|b}[a_i] = \ _i(b) \tag{26}$$

are mapped to:

$$E_{a_i\,|b}[a_i] = 0; \qquad Var_{a_i\,|b}[a_i] = 1: \tag{27}$$

This is achieved via the following af ne transformation, shifting the conditional mean to zero and scaling the conditional standard deviation to one:

$$a_i(a_i; b) = \frac{1}{\ _i(b)}(a_i \quad m_i(b)): \tag{28}$$

In terms of loss, this transformation can at most achieve the following loss improvement, with a contribution from each passive coordinate b:

$${}_{af\,ne}(p\ (z)) = \max_{Q} E_b[S(b)]; \tag{29}$$

where Q enters through $(b; a) = Qx$, and the expectation goes over the component-wise non-Standardness (S, Draxler et al., 2022):

$$S(b) = \sum_{i=1}^{D=2} D_{KL}(N(m_i(b); \ _i(b)\|N(0; 1)) \tag{30}$$

$$= \frac{1}{2}\sum_{i=1}^{D=2} E_b\Big[\underbrace{m_i^2(b)}_{(A)} + \underbrace{\ _i^2(b) \quad 1 \quad \log\ _i^2(b)}_{(B)}\Big]; \tag{31}$$

With the asterisk, we denote that this improvement cannot necessarily be reached in practice with nitely sized and well-conditioned neural networks. More expressive coupling functions can reduce the loss stronger, see Section 5.4.

What loss improvement can be achieved if we go back to nite neural networks? It turns out that $_{af\,ne}(p\ (z)) > 0$ is equivalent to the existence of a well-conditioned coupling block as in De nition 5.1 with $_{af\,ne}(p\ (z)) > 0$:

Lemma 5.3. Given a continuous probability density $p(z)$ on $z \in R^k$. Then,

$${}_{af\,ne}(p\ (z)) > 0 \quad () \quad {}_{af\,ne}(p\ (z)) > 0: \tag{32}$$

This says that the events $_{af\,ne}(p\ (z)) = 0$ and $_{af\,ne}(p\ (z)) = 0$ can be used interchangeably. The equivalence comes from the fact that if $_{af\,ne}(p\ (z)) > 0$, then

we can always construct a conditioner neural network that scales the conditional standard deviations closer to one or the conditional means closer to zero, reducing the loss. In the detailed proof in Appendix C.1.2 we also make use of a classical regression universal approximation theorem (Hornik, 1991) and ensure the additional coupling block is well-conditioned.

Finally, if the first two conditional moments of any latent distribution $p(z)$ are normalized for all rotations Q:

$$E_{a_i|b}[a_i] = m_i(b) = 0\,; \quad \mathrm{Var}_{a_i|b}[a_i] = \sigma_i(b) = 1\,; \quad (33)$$

then the distribution must be the standard normal distribution: $p(z) = \mathcal{N}(z; 0, I)$. Equation (31) enforces two characteristics of $p(z)$ that uniquely identify the standard normal distribution: (A) It must be rotationally symmetric, since $m_i(b) = 0$ for all Q holds only for rotationally symmetric distributions (Eaton, 1986). (B) This is term non-negative and zero only for $\sigma_i(b) = 1$ for all Q, which uniquely identifies the standard normal from all rotationally symmetric distributions (Bryc, 1995).

This concludes the proof sketch of Theorem 5.2 and we are now ready to present our universality result, employing $\Phi_{\mathrm{affine}}(p(z))$ as a convergence metric.

5.3. Affine Coupling Flows Universality

We now confirm that affine coupling flows are a distributional universal approximator in terms of the convergence metric we derived in Section 5.2:

Theorem 5.4. For every continuous $p(x)$ with finite first and second moment with infinite support, there is a sequence of normalizing flows $p_n(x)$ consisting of n L-bi-Lipschitz affine coupling blocks such that their latent distributions converge:

$$p_n(z) \overset{n\to\infty}{\rightharpoonup} \mathcal{N}(z; 0, I)\,; \quad (34)$$

in the sense that $\Phi_{\mathrm{affine}}(p_n(z)) \overset{n\to\infty}{\to} 0$.

This means that with increasing depth, the latent distribution of the flow converges to the standard normal. The use of $\Phi_{\mathrm{affine}}(p_n(z))$ as a convergence metric is justified by Theorem 5.2 that $p_n(z) = \mathcal{N}(0, I)$, $\Phi_{\mathrm{affine}}(p_n(z)) = 0$.

The proof of Theorem 5.4 explicitly constructs a normalizing flow by following an iterative scheme. We start with the data distribution as our original guess for the latent distribution: $p_0(z) = p(x = z)$. Then, we repeatedly append individual affine coupling blocks $f_{\mathrm{blk}}(x)$ consisting of a rotation Q and a coupling f_{cpl}, optimizing the new parameters to maximally reduce the loss as in Equation (23).

This series of coupling blocks converges: $\Phi_{\mathrm{affine}}(p(z))$ measures how much adding each affine coupling block reduces the loss, but the total loss that can be reduced by the concatenation of many blocks is bounded. Since improvements $\Phi_{\mathrm{affine}}(p(z))$ are also non-negative, they must converge to zero for the sum to be finite (Rudin, 1976, Theorems 3.14 and 3.23). By Theorem 5.2, the fixed point of this procedure is a standard normal distribution in the latent space. We give the full proof in Appendix C.2.

Figure 1 shows an example for how Theorem 5.4 constructs the coupling flow in order to learn a toy distribution. The affine coupling flow is able to learn the distribution well, despite its difficult topology. Empirically, this is also true in terms of KL divergence: Figure 5 in Appendix E.1 shows the relation between $\Phi_{\mathrm{affine}}(p(z))$ and the KL divergence for the flow, both of which decrease over the course of training.

Table 1 summarizes how our construction is closer to practice than previous work (Teshima et al., 2020a; Koehler et al., 2021): We only use well-conditioned bi-Lipschitz couplings, allow variable volume change $|g'(x)|$ (as evidenced by the rescaling term in Equation (28)) and consider the entire support of $p(x)$. We give further details on the sensitivity of $\Phi_{\mathrm{affine}}(p(z))$ to volume-preserving transformations in Appendix B.1.3.

Limitations: Despite these advances, there are some properties we hope can be improved in the future: First, our construction shows that we can build a deep enough flow with arbitrary precision, but we have not exploited that blocks can be jointly optimized. Thus, while our construction shows universality of end-to-end training, we expect a flow trained this way to require fewer blocks than our iterative proof for the same performance.

Secondly, it is unclear how the convergence metric $\Phi_{\mathrm{affine}}(\cdot)$ in Definition 5.1 is related to convergence in the loss used in practice, the KL divergence given in Equation (3). In practice, we find that constructing a coupling flow through iterative training converges in KL divergence (see Figure 5 in Appendix E.1), so we conjecture that our way of constructing a universal coupling flow converges in KL divergence. The reverse holds: We show in Corollary C.3 in Appendix C.3 that convergence in KL implies convergence under our new metric.

Finally, our proof gives no guarantee on the number of required coupling blocks to achieve a certain performance. Related work shows that the number of layers is constant with dimension for the special case of Gaussian data (Koehler et al., 2021; Draxler et al., 2022), but in practice is a hyperparameter that is to be tuned depending on the data and together with the complexity of the subnetworks. We hope that our contribution paves the way towards a full understanding of affine coupling-based normalizing flows.

Figure 3. Information-geometric view of couplings more expressive than affine: The conditional KL divergence $D_{KL}(p(a|b)\|N(0,I))$ can be split into two orthogonal KL divergences, the non-Standardness $S(b)$ sensitive to the first two moments, and the negentropy $J(b)$ sensitive to non-Gaussianity. Affine couplings only reduce $S(b)$, more expressive coupling also affect $J(b)$.

5.4. Expressive Coupling Flow Universality

The above Theorem 5.4 shows that affine couplings $c(a_i;\ ) = sa_i + t$ are sufficient for universal distribution approximation. As mentioned in Section 3, a plethora of more expressive coupling functions have been suggested, for example neural spline flows (Durkan et al., 2019b) that use monotone rational-quadratic splines as the coupling function. It turns out that by choosing the parameters in the right way, all coupling functions we are aware of can exactly represent an affine coupling, except for the volume-preserving variants, see Appendix A.1. For example, a rational quadratic spline can be parameterized as an affine function by using equidistant knots $(a_k; a_k)$ such that $a_k = sa_k + t$ and fixing the derivative at each knot to 1.

Thus, the universality of more expressive coupling functions follows immediately from Theorem 5.4, just like Ishikawa et al. (2022) extended their results from affine to more expressive couplings:

Corollary 5.5. For every continuous $p(x)$ with finite first and second moment with infinite support, there is a sequence of normalizing flows $f_n(x)$ consisting of n coupling blocks with coupling functions at least as expressive as affine couplings such that:

$$p_n(z) \overset{n\to\infty}{\to} N(z;0,I); \tag{35}$$

in the sense that $_{affine}(p_n(z)) \overset{n\to\infty}{\to} 0$.

Our proof of Theorem 5.4, constructed through layer-wise training, shows how more expressive coupling functions can outperform affine functions using the same number of blocks. Similar to the loss improvement for an affine coupling in Equation (24), let us compute the maximally possible loss improvement for an arbitrarily flexible coupling function:

$$_{universal} = \max_Q (E_b[J(b) + S(b)]) \quad _{affine}; \tag{36}$$

where the expectation again goes over the passive coordinate $b = (Qz)_{1,\dots,D-2}$ and $z \sim p(z)$.

Here, the loss improvement additional to the non-Standardness $S(b)$ as given in Equation (31) is the conditional negentropy $J(b) = \sum_{i=1}^{D-2} D_{KL}(p(a_i|b)\|N(m_i(b); _i(b)))$, which measures the deviation of each active dimension from a Gaussian distribution with matching mean and variance. An affine coupling function $c(a_i;\ ) = sa_i + t$ doesn't influence this term, due to its symmetrical effect on both sides of the KL in $J(p)$ (Draxler et al., 2022, Lemma 1). More expressive coupling blocks, however, are able to tap on this loss component if the conditional distributions $p(a_i|b)$ are non-Gaussian, see Figure 3 for an example. Note that while a single affine coupling does not affect $J(b)$, subsequent blocks can because the overall loss is redistributed over the loss terms.

The impact of this gain likely varies with the dataset. For instance, in images, the distribution of one color channel of one pixel conditioned on the other color channels in the entire image, often shows a simple unimodal pattern with low negentropy. This may explain why affine coupling blocks are enough to learn the distribution of images (Kingma & Dhariwal, 2018). We give additional technical details on Equation (36) and the subsequent arguments in Appendix D.

6. Conclusion

Our new universality proofs show an intriguing hierarchy of the universality of different coupling blocks:

1. Volume-preserving normalizing flows are not universal in KL divergence and thus cannot learn all target densities $p(x)$.
2. Affine coupling flows such as RealNVP (Dinh et al., 2017) are distributional universal approximators in terms of $_{affine}$ despite their seemingly restrictive architecture.
3. Coupling flows with more expressive coupling functions are also universal approximators, but they converge faster by tapping on an additional loss component in layer-wise training.

Our work theoretically grounds choosing coupling blocks for practical applications with normalizing flows, combined with their easy implementation and training and inference speed. We remove spurious constructions present in previous proofs and use a simple principle instead: Train a flow layer by layer.

Using volume-preserving flows may have negatively affected existing work. We show what distribution $p(x)$ they approximate instead of the true target $p(x)$ and propose how universality can be recovered by learning the actual latent distribution after training.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

This work is supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC-2181/1 - 390900948 (the Heidelberg STRUCTURES Cluster of Excellence). It is also supported by the Vector Stiftung in the project TRINN (P2019-0092) and the Carl-Zeiss-Stiftung. We thank Armand Rousselot and Peter Sorrenson for the fruitful discussions and feedback. We thank Vincent Souveton for the helpful comments regarding Neural Hamiltonian Flows and Hamiltonian Monte Carlo.

References

Albergo, M. S., Kanwar, G., and Shanahan, P. E. Flow-based generative models for Markov chain Monte Carlo in lattice field theory. Physical Review D, 100(3):034515, August 2019. ISSN 2470-0010, 2470-0029. doi: 10.1103/PhysRevD.100.034515.

Ardizzone, L., Bungert, T., Draxler, F., Köthe, U., Kruse, J., Schmier, R., and Sorrenson, P. Framework for Easily Invertible Architectures (FrEIA), 2018a.

Ardizzone, L., Kruse, J., Rother, C., and Köthe, U. Analyzing Inverse Problems with Invertible Neural Networks. In International Conference on Learning Representations, 2018b.

Bryc, W. The Normal Distribution, volume 100 of Lecture Notes in Statistics. Springer New York, New York, NY, 1995. ISBN 978-0-387-97990-8 978-1-4612-2560-7. doi: 10.1007/978-1-4612-2560-7.

Cambanis, S., Huang, S., and Simons, G. On the theory of elliptically contoured distributions. Journal of Multivariate Analysis, 11(3):368–385, September 1981. ISSN 0047259X. doi: 10.1016/0047-259X(81)90082-8.

Cardoso, J.-F. Dependence, Correlation and Gaussianity in Independent Component Analysis. Journal of Machine Learning Research, 4:1177–1203, 2003. ISSN 1532-4435.

Chen, S. and Gopinath, R. Gaussianization. In Leen, T., Dietterich, T., and Tresp, V. (eds.), Advances in Neural Information Processing Systems 13, 2000.

Dibak, M., Klein, L., Krämer, A., and Noé, F. Temperature steerable flows and Boltzmann generators. Phys. Rev. Res., 4(4):L042005, October 2022. doi: 10.1103/PhysRevResearch.4.L042005.

Dinh, L., Krueger, D., and Bengio, Y. NICE: Non-linear Independent Components Estimation. In International Conference on Learning Representations, Workshop Track, 2015.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. Density estimation using Real NVP. In International Conference on Learning Representations, 2017.

Draxler, F., Schwarz, J., Schorr, C., and Köthe, U. Characterizing the Role of a Single Coupling Layer in Affine Normalizing Flows. In German Conference on Pattern Recognition, 2020.

Draxler, F., Schörr, C., and Köthe, U. Whitening Convergence Rate of Coupling-based Normalizing Flows. In Advances in Neural Information Processing Systems, 2022.

Draxler, F., Kühmichel, L., Rousselot, A., Müller, J., Schnoerr, C., and Koethe, U. On the Convergence Rate of Gaussianization with Random Rotations. In International Conference on Machine Learning, 2023.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Cubic-Spline Flows. In International Conference on Machine Learning, Workshop Track, 2019a. doi: 10.48550/ARXIV.1906.02145.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural Spline Flows. In Advances in Neural Information Processing Systems, 2019b.

Eaton, M. L. A characterization of spherical distributions. Journal of Multivariate Analysis, 20(2):272–276, December 1986. ISSN 0047259X. doi: 10.1016/0047-259X(86)90083-7.

Gibbs, A. L. and Su, F. E. On Choosing and Bounding Probability Metrics. International Statistical Review / Revue Internationale de Statistique, 70(3):419–435, 2002. ISSN 03067734, 17515823.

Glorot, X. and Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In International Conference on Artificial Intelligence and Statistics, 2010.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard,

K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. Array programming with NumPy. Nature, 585(7825):357–362, 2020.

Ho, J., Chen, X., Srinivas, A., Duan, Y., and Abbeel, P. Flow++: Improving Flow-Based Generative Models with Variational Dequantization and Architecture Design. In International Conference on Machine Learning, 2019.

Hornik, K. Approximation capabilities of multilayer feedforward networks. Neural Networks, 4(2):251–257, 1991. ISSN 08936080. doi: 10.1016/0893-6080(91)90009-T.

Huang, C.-W., Krueger, D., Lacoste, A., and Courville, A. Neural Autoregressive Flows. In International Conference on Machine Learning, 2018.

Huang, C.-W., Dinh, L., and Courville, A. Augmented Normalizing Flows: Bridging the Gap Between Generative Flows and Latent Variable Models. In International Conference on Learning Representations, Workshop Track, 2020.

Hunter, J. D. Matplotlib: A 2D graphics environment. Computing in Science & Engineering, 9(3):90–95, 2007.

Ishikawa, I., Teshima, T., Tojo, K., Oono, K., Ikeda, M., and Sugiyama, M. Universal approximation property of invertible neural networks. arXiv preprint arXiv:2204.07415, 2022.

Jaini, P., Selby, K. A., and Yu, Y. Sum-of-Squares Polynomial Flow. In International Conference on Machine Learning, 2019.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization, January 2017.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. In Advances in Neural Information Processing Systems, 2018.

Kobyzev, I., Prince, S. J., and Brubaker, M. A. Normalizing Flows: An Introduction and Review of Current Methods. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(11):3964–3979, 2021.

Koehler, F., Mehta, V., and Risteski, A. Representational aspects of depth and conditioning in normalizing flows. In International Conference on Machine Learning, 2021.

Köthe, U. A review of change of variable formulas for generative modeling. arXiv preprint arXiv:2308.02665, 2023.

Lee, H., Pabbaraju, C., Sevekari, A. P., and Risteski, A. Universal approximation using well-conditioned normalizing flows. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), Advances in Neural Information Processing Systems, volume 34, pp. 12700–12711. Curran Associates, Inc., 2021.

Lyu, J., Chen, Z., Feng, C., Cun, W., Zhu, S., Geng, Y., Xu, Z., and Chen, Y. Universality of parametric Coupling Flows over parametric diffeomorphisms. arXiv preprint arXiv:2202.02906, 2022.

McKinney, W. Data Structures for Statistical Computing in Python. In van der Walt, S. and Jarrod Millman (eds.), 9th Python in Science Conference, 2010.

Müller, T., Mcwilliams, B., Rousselle, F., Gross, M., and Novák, J. Neural Importance Sampling. ACM Transactions on Graphics, 38(5):1–19, 2019. ISSN 0730-0301. doi: 10.1145/3341156.

Neal, R. M. MCMC Using Hamiltonian Dynamics. In Brooks, S., Gelman, A., Jones, G. L., and Meng, X.-L. (eds.), Handbook of Markov Chain Monte Carlo. Chapman and Hall/CRC, 2011.

Nicoli, K. A., Anders, C. J., Funcke, L., Hartung, T., Jansen, K., Kessel, P., Nakajima, S., and Stornati, P. Estimation of Thermodynamic Observables in Lattice Field Theories with Deep Generative Models. Physical Review Letters, 126(3):032001, January 2021. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.126.032001.

Noé, F., Olsson, S., Köhler, J., and Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. Science, 365(6457):eaaw1147, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems, 2019.

Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In International Conference on Machine Learning, 2015.

Rudin, W. Principles of Mathematical Analysis. International Series in Pure and Applied Mathematics. McGraw-Hill, New York St. Louis San Francisco [etc.], third ed edition, 1976. ISBN 978-0-07-054235-8.

Sorrenson, P., Rother, C., and Köthe, U. Disentanglement by nonlinear ICA with general incompressible-flow networks (GIN). In International Conference on Learning Representations, 2019.

Souveton, V., Guillin, A., Jasche, J., Lavaux, G., and Michel, M. Fixed-kinetic Neural Hamiltonian Flows for enhanced interpretability and reduced complexity. In International Conference on Artificial Intelligence and Statistics, 2024.

Teshima, T., Ishikawa, I., Tojo, K., Oono, K., Ikeda, M., and Sugiyama, M. Coupling-based Invertible Neural Networks Are Universal Diffeomorphism Approximators. In Advances in Neural Information Processing Systems, 2020a.

Teshima, T., Tojo, K., Ikeda, M., Ishikawa, I., and Oono, K. Universal Approximation Property of Neural Ordinary Differential Equations. In Advances in Neural Information Processing Systems, Workshop Track, 2020b.

The pandas development team. Pandas-dev/pandas: Pandas, February 2020.

Toth, P., Rezende, D. J., Jaegle, A., Racaniere, S., Botev, A., and Higgins, I. Hamiltonian generative networks. In International Conference on Learning Representations, 2020.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

Wehenkel, A. and Louppe, G. Unconstrained Monotonic Neural Networks. In Advances in Neural Information Processing Systems, 2019.

Zhang, H., Gao, X., Unterman, J., and Arodz, T. Approximation Capabilities of Neural ODEs and Invertible Residual Networks. In International Conference on Machine Learning, 2020.

Ziegler, Z. and Rush, A. Latent Normalizing Flows for Discrete Sequences. In International Conference on Machine Learning, 2019.

A. Architectures

A.1. Compatible Coupling Functions

The following lists all coupling functions $c(a; \theta)$ (see Equation (6) for its usage) we are aware of. Our universality guarantees Theorem 5.4 and Corollary 5.5 hold for all of them:

- Affine coupling flows as RealNVP (Dinh et al., 2017) and GLOW (Kingma & Dhariwal, 2018):

$$c(x; \theta) = sx + t. \tag{37}$$

  Here, $\theta = [s, t] \in \mathbb{R}_+ \times \mathbb{R}$. Note that NICE ($s = 1$, Dinh et al. (2015)) and GIN ($\prod_{i=1}^{D=2} s_i = 1$, (Sorrenson et al., 2019)) follow the same functional form, but are volume-preserving and thus not universal (see Section 4).

- Nonlinear squared flow (Ziegler & Rush, 2019):

$$c(x; \theta) = ax + b + \frac{c}{1 + (dx + h)^2}; \tag{38}$$

  for $\theta = [a, b, c, d, h] \in \mathbb{R}_+ \times \mathbb{R}^4$. Choose $c = 0$ to obtain an affine coupling.

- Flow++ (Ho et al., 2019):

$$c(x; \theta) = s \left(\sum_{j=1}^{K} \alpha_j \, \sigma\left(\frac{x - \beta_j}{\gamma_j} \right) + t \right). \tag{39}$$

  Here, $\theta = [s, t, (\alpha_j, \beta_j, \gamma_j)_{j=1}^K] \in \mathbb{R}_+ \times \mathbb{R} \times (\mathbb{R} \times \mathbb{R} \times \mathbb{R}_+)^K$ and σ is the logistic function. Choose all $\alpha_j = 0$ except for $\alpha_1 = 1$, all $\beta_j = 0$ and all $\gamma_j = 1$ to obtain an affine coupling.

- SOS polynomial flows (Jaini et al., 2019):

$$c(x; \theta) = \int_0^x \sum_{k=1}^{K} \left(\sum_{l=0}^{r} a_{k,l} \, u^l \right)^2 du + t. \tag{40}$$

  Here, $\theta = [t, (a_{k,l})_{k,l}] \in \mathbb{R} \times \mathbb{R}^{rK}$. Choose all $a_{k,l} = 0$ except for $a_{1,0} = s$ to obtain an affine coupling.

- Spline flows in all variants: Cubic (Durkan et al., 2019a), piecewise-linear, monotone quadratic (Müller et al., 2019), and rational quadratic (Durkan et al., 2019b) splines. A spline is parameterized by knots with optional derivative information depending on the spline type, and computes the corresponding spline function. Choose the spline knots as $y_i = sx_i + t$ for an affine coupling, choose the derivatives as $x_i' = s$ for an affine coupling.

- Neural autoregressive flow (Huang et al., 2018) use a feed-forward neural network to parameterize $c(x; \theta)$ by a feed-forward neural network. They show that a neural network is guaranteed to be bijective if all activation functions are strictly monotone and all weights positive. One can construct a ReLU network with a single linear region to obtain an affine coupling.

- Unconstrained monotonic neural networks (Wehenkel & Louppe, 2019) also use a feed-forward neural, but restrict it to have positive output. To obtain $c(x; \theta)$, this function is then numerically integrated with a learnable offset for $x = 0$. Choose a constant neural network to obtain an affine coupling.

A.2. Volume-Preserving Normalizing Flows

Below, we list the ways to construct volume-preserving flows we are aware of. Our non-universality results Theorems 4.1 and 4.2 and Proposition 4.3 hold for all of them:

- Nonlinear independent components estimation (NICE) (Dinh et al., 2015) is a coupling block:

$$c(x; \theta) = x + t. \tag{41}$$

  Here, $\theta = t \in \mathbb{R}$.

- General Incompressible-flow Networks (GIN) (Sorrenson et al., 2019) generalize NICE by allowing the individual dimensions to change volume, only the overall volume change is normalized:

$$c_i(x_i; \theta) = \frac{s_i}{\prod_{j=1}^{D-2} s_j} x_i + t_i; \tag{42}$$

  Here, $\theta = [s; t] \in \mathbb{R}_+^{D-2} \times \mathbb{R}^{D-2}$ is jointly predicted for all active dimensions and then normalized as above.

- Neural Hamiltonian Flows (Toth et al., 2020) parameterize a Neural ODE as a Hamiltonian system:

$$\frac{dq}{dt} = \frac{\partial H}{\partial p}, \qquad \frac{dp}{dt} = -\frac{\partial H}{\partial q} \tag{43}$$

  The Hamiltonian $H(p; q)$ is a real-valued function that is parameterized by a neural network. Its derivatives are obtained via automatic differentiation. The variant Fixed-kinetic Neural Hamiltonian Flows (Souveton et al., 2024) fixes the kinetic term of the Hamiltonian to $K(p) = \frac{1}{2} p^T M^{-1} p$, where the positive definite matrix M is learned, and learns the potential $V(q)$ via a neural network to obtain $H(p; q) = K(p) + V(q)$. The solution to the above ODE is volume-preserving on $x = (p; q)$.

Note that some works employing volume-preserving flows such as Dibak et al. (2022); Souveton et al. (2024) consider augmented flows (Huang et al., 2020), where additional noise dimensions are padded $p(a|x)$ to the data distribution of interest $p(x)$. Then, the flow learns the joint distribution $p(x; a) = p(x)p(a|x)$. Depending on how $p(a|x)$ is constructed, this can positively or negatively impact the expressivity of the considered volume-preserving flow. For example, if $p(a|x) = p(a)$, then the joint distribution $p(x; a)$ has at least the same number of modes as $p(x)$, but the learned joint distribution $p_\theta(x; a)$ can only have a single mode by Proposition 4.3, inducing a bias. To derive the universality in terms of KL, apply Theorem 4.1 to the joint distribution at hand. On the positive side, Souveton et al. (2024) find that $p(a|x) = \mathcal{N}(\mu(x); \sigma(x)^2 I)$ brings the obtained $p_\theta(x) = \int p_\theta(x; a)\,da$ closer to the target. This can be seen having an independent augmentation $a \sim \mathcal{N}(0; I) \perp x$ plus a single RealNVP coupling shifting and scaling the augmented dimensions. This effectively breaks the volume-preservation of the flow in the joint space. It is unclear, however, whether this removes all biases from the volume-preserving flow.

B. Proofs on Volume-Preserving Normalizing Flows

B.1. Minimizer of Volume-Preserving Normalizing Flows

Volume-preserving flows are not universal in KL divergence, see Theorem 4.2. In this section, we consider what distribution a volume-preserving flow converges to instead. We first construct the latent distribution a sufficiently rich volume-preserving flows converges to when trained with KL divergence (Appendix B.1.1) and then show that this actually minimizes the KL (Appendix B.1.2).

We also demonstrate in what sense d_{affine} is sensitive to volume-preserving transformations. We therefore show in Appendix B.1.3 that this flow converges under our convergence measure d_{affine} if and only if that volume-preserving flow converges to a standard normal in the latent space under KL divergence.

B.1.1. Rotationally Symmetric Distribution with Same Level Set Structure

Let us repeat the change-of-variables formula for a volume-preserving flow from Equation (11):

$$p_\theta(x) = p(z = f_\theta(x)) \cdot C; \tag{44}$$

where $C = |f'_\theta(x)|$ is constant with respect to x. Intuitively, this means that such a flow can permute the probability mass at all locations x, and apply a single global factor to scale all probability values by spreading out the distribution.

We now construct the best possible distribution learned by a volume-preserving flow to an arbitrary $p(x)$ in terms of KL divergence. We therefore split the input distribution $p(x)$ into its level sets:

$$L_v(p(\cdot)) := \{x \in \mathbb{R}^D : p(x) = v\}. \tag{45}$$

Acting on the input distribution, a volume-preserving flow yields a latent distribution $p(z)$ whose level set structure is closely related to that of the input distribution, in the following sense:

$$|L_v(p(x))| = \frac{|L_{v=C}(p(z))|}{C}. \tag{46}$$

Intuitively, this captures that a volume-preserving flow maps the input space to the latent space such that the level set in data space for the density value v is mapped to the level set in the latent space of level $v=C$ – and the $(D-1)$-dimensional volumes are scaled by the factor C. Here, we have assumed that the level sets are $(D-1)$-dimensional.

In the following, we use these level sets to construct the distribution $p(z)$ a volume-preserving flow converges to in the latent space. This allows us to specify the best solution a volume-preserving flow can converge to. We first consider a fixed C and then later solve the optimization over C separately.

Lemma B.1. Let $p(x)$ be a bounded continuous probability density with $(D-1)$-dimensional level sets almost everywhere. Then, a unique continuous probability density $p(z)$ with the following properties exists:

1. Its level sets have equal volume: $|L_v(p)| = |L_v(p)|$,
2. p is rotationally symmetric: $p(z) = p(Qz)$ for all $Q \in SO(D)$,
3. $p(z_1, 0, \dots, 0)$ is strictly monotonically decreasing in $z_1 < 1$.

Proof. We write $p(x) = p(r)p(\phi|r)$, where (r, ϕ) are hyper-spherical coordinates. Since $p(x)$ should be rotationally symmetric, the distribution of the solid angle ϕ is isotropic, and equal to one over the surface $A_{D-1}(r)$ of the $(D-1)$-dimensional hypersphere:

$$p(\phi|r) = \frac{1}{A_{D-1}(r)} = \frac{\Gamma(D/2)}{2\pi^{\frac{D}{2}} r^{D-1}} \tag{47}$$

This makes p rotationally symmetric and leaves us with constructing $p(r)$.

Define the superlevel sets $L_v^+(p)$ for $p(x)$ as follows:

$$L_v^+(p) = \{x \in \mathbb{R}^D : p(x) \geq v\}. \tag{48}$$

Their volume $|L_v^+(p)|$ as measured in $(D-1)$ dimensions is monotonically decreasing in v, and its derivative yields the volume of the level set:

$$\frac{\partial}{\partial v}|L_v^+(p)| = |L_v(p)|. \tag{49}$$

We now demand that

$$|L_v^+(p)| = |L_v^+(p')| \tag{50}$$

and integrate out v:

$$|L_v^+(p')| = \int 1[p(x) \geq v]dx \tag{51}$$

$$= \int d\Omega \int r^{D-1} 1[p(r) \geq v]dr \tag{52}$$

$$= A_{D-1}(1) \int r^{D-1} 1[p(r) \geq v]dr. \tag{53}$$

Since $p(r)$ should decrease monotonically with r, we can replace the indicator function by integral boundaries, where the upper limit depends on the target density value v. We identify $\max_x p(x)$ with $R = 0$:

$$|L_v^+(p')| = A_{D-1}(1) \int_0^{R(v)} r^{D-1} dr \tag{54}$$

$$= A_{D-1}(1) \frac{1}{D} R(v)^D \tag{55}$$

Rearranging yields $R(v)$ from $|L_v^+(p)|$:

$$R(v) = \left(\frac{D|L_v^+(p)|}{A_{D-1}(1)} \right)^{\frac{1}{D}} \tag{56}$$

As $R(v)$ is monotonous in $|L_v^+(p)|$, $|L_v^+(p)|$ is continuous and monotonous in v, and $R(v)$ is invertible, its inverse can be used to define $p_\perp(r)$:

$$p_\perp(r) = R^{-1}(r). \tag{57}$$

By choosing $R(v)$ to fulfill Equation (50), their derivatives also match:

$$|L_v(p)| = \frac{\partial}{\partial v}|L_v^+(p)| = \frac{\partial}{\partial v}|L_v^+(p_\perp)| = |L_v^+(p_\perp)|. \tag{58}$$

The other properties of f follow directly from the construction above. The density is unique (up to zero sets) since a rotationally symmetric distribution is uniquely defined by a one-dimensional ray (Eaton, 1986). □

We now show that the latent distribution $p_\perp(z)$ is actually attainable by a volume-preserving flow:

Lemma B.2. Under the assumptions of Lemma B.1. There exists a function $f: \mathbb{R}^D \to \mathbb{R}^D$ that is bijective, continuous and volume-preserving with unit volume change ($|f'(x)| = 1$) almost everywhere and $p_\perp(z) = p_\perp(x = z)$.

This means that there is a volume-preserving flow that exactly pushes $p_\perp(x)$ to its respective $p_\perp(x)$. Note that this volume-preserving flow has $|f'(x)| = 1$.

Proof. First, note that $p_\perp(z) = p_\perp(x = z)$ as constructed above can be achieved by a volume-preserving bijection. To see this, divide the space into the level sets $L_{p_\perp}(v)$ of $p_\perp(x)$. Since $p_\perp(x)$ is continuous, there is a countable sequence of thresholds v_i at which the number of connected components in the level sets $L_{p_\perp}(v)$ jumps: The first jump is at $v_{\max} = \max_x p_\perp(x)$, below which we find as many connected components as there are maximal modes $p_\perp(x)$. The number of connected components changes whenever there is a saddle point or maximum in $p_\perp(x)$. Between each subsequent pair of jumps (v_i, v_{i+1}), each connected component can be continuously assigned to a countable cluster number. This yields two tessellations of the entire space: One into level sets, and one into continuous connected components. To construct f, we first assign the highest points of $x \in \mathbb{R}^D$ to $f(x) = 0$. Then, we continuously arrange the finite number of components until the next jump in $(\max_x p_\perp(x), v_1)$ around the origin such that the resulting level sets are concentric circles. This construction pushes $p_\perp(x)$ to $p_\perp(x)$ and fulfills the above constraints. □

B.1.2. Best Volume-Preserving Normalizing Flow under KL Divergence

We are going to make use of the following identity that a volume-preserving flow with $|\tilde{f}'(x)| = 1$ with a standard normal in the latent space can be written as a volume-preserving flow with $|f'(x)| = 1$ and an alternative latent distribution $p(z) = \mathcal{N}(0; C^{-2}I)$:

Lemma B.3. Given a volume-preserving bijection f that is diffeomorphic almost everywhere, and with $|f'(x)| = C$ for some $C > 0$. Then, there exists a volume-preserving bijection $\tilde{f}$ with $|\tilde{f}'(x)| = 1$ such that:

$$((f^{-1})_\sharp \mathcal{N}(0; I))(x) = ((\tilde{f}^{-1})_\sharp \mathcal{N}(0; C^{-2}I))(x) \tag{59}$$

In other words, the global volume change of a volume-preserving flow can be absorbed into a single scaling last layer at the latent end of the flow.

Proof. Let $\tilde{f}(x) = C^{-1} f(x)$, which has $|\tilde{f}'(x)| = C/C = 1$ and write $\mathcal{N}(0; C^{-2}I)$ as the push-forward through $z \mapsto C^{-1}z$. □

Now we show our main Theorem 4.1 on volume-preserving flows:

Theorem 4.1. Given a continuous bounded input density $p(x)$. Then, for any volume-preserving flow $p_\nu(x)$ with a standard normal latent distribution, the achievable KL divergence is bounded from below:

$$D_{KL}(p(x)\|p_\nu(x)) \geq D_{KL}(p_\nu(z)\|\mathcal{N}(0; |\Sigma_{p_\nu(z)}|^{\frac{1}{D}} I)), \tag{60}$$

where $p_\nu(z)$ is constructed as in Lemma B.1, and $\Sigma_{p_\nu(z)}$ is its covariance matrix. The minimal loss is achieved for $C = |\Sigma_{p_\nu(z)}|^{\frac{1}{2D}}$.

Proof. By Equation (22), $D_{KL}(p(x)\|p_\nu(x)) = D_{KL}(p_\nu(z)\|\mathcal{N}(0; I))$. The variant in the latent space can be rewritten using the entropy of $p_\nu(z)$ as:

$$D_{KL}(p_\nu(z)\|\mathcal{N}(0; I)) = D_{KL}(p_\nu(z)\|\mathcal{N}(0; I)) = -H[p_\nu(x)] - \int p_\nu(z) \log \mathcal{N}(z; 0; I)\, dz. \tag{61}$$

The entropy of the latent distribution of a volume-preserving flow only depends on the volume change constant C, but not on the exact choice of f:

$$\begin{aligned} H[p_\nu(z)] &= -\int p_\nu(z) \log p_\nu(z)\, dz && (62)\\ &= -\int p(x) \log(p_\nu(z = f_\nu(x))\, C)\, dx && (63)\\ &= -\int p(x) \log p(x)\, dx - \log C && (64)\\ &= H[p(x)] - \log C. && (65) \end{aligned}$$

Inserting into Equation (61):

$$D_{KL}(p_\nu(z)\|\mathcal{N}(0; I)) = D_{KL}(p_\nu(z)\|\mathcal{N}(0; I)) = -H[p(x)] + \log C - \int p_\nu(z) \log \mathcal{N}(z; 0; I)\, dz. \tag{66}$$

Using Lemma B.3, rewrite the last term in Equation (66) as an integral over x:

$$-\int p_\nu(z) \log \mathcal{N}(z; 0; C^2 I)\, dz = -\int p(x) \log \mathcal{N}(\tilde{f}_\nu(x); 0; C^2 I)\, dx. \tag{67}$$

This reveals the KL is minimized by assigning the highest values of $p(x)$ to the highest values of $\mathcal{N}(0; C^2 I)$. Since the order of densities $\mathcal{N}(z; 0; C^2 I)$ is the same regardless of C, the assignment $\tilde{f}_\nu(x)$ does not depend on C, which can be estimated separately via

$$D_{KL}(p_\nu(z)\|\mathcal{N}(0; C^2 I)). \tag{68}$$

Adapting (Draxler et al., 2022, Proposition 1), the KL divergence Equation (68) can be decomposed as follows:

$$D_{KL}(p_\nu(z)\|\mathcal{N}(0; C^2 I)) = D_{KL}(p_\nu(z)\|\mathcal{N}(0; |\Sigma_z|^{\frac{1}{D}} I)) + D_{KL}(\mathcal{N}(0; |\Sigma_z| I)\|\mathcal{N}(0; C^2 I)), \tag{69}$$

where $|\Sigma_z|$ is the determinant of the covariance matrix of the latent codes: $\Sigma_z = \mathrm{Cov}_{z\sim p_\nu(z)}[z]$. The first term is invariant under scaling the latent codes, and the second term is minimal for $C^2 = |\Sigma_z|^{\frac{1}{D}}$.

□

The remaining loss $D_{KL}(p_\nu(z)\|\mathcal{N}(0; |\Sigma_{p_\nu(z)}|^{\frac{1}{D}} I))$ can be reduced to zero by switching to spherical coordinates and learning $p_\nu(r)$ via a non-volume-preserving one-dimensional distribution, see Appendix B.2.

B.1.3. Sensitivity of α_{affine} to Volume-Preserving Flows

We now confirm that if a volume-preserving flow is not universal under KL divergence, it is also not universal under α_{affine}.

Lemma B.4. For a family of normalizing flows with constant Jacobian determinant $|\det J_\theta(x)| = \text{const}$, such that $D_{KL}(p_\theta(z) \| p(z)) \to 0$, it holds that $\alpha_{affine} \to 0$ if and only if $D_{KL}(p_\theta(z) \| \mathcal{N}(0, I)) = 0$.

Proof. By Lemma 5.3, we can use $\alpha_{affine} > 0$ and $\alpha_{affine} > 0$ interchangeably. By its definition in Equation (31),

$$\alpha_{affine} = \max_Q \frac{1}{2} \sum_{i}^{D-2} \mathbb{E}_b \left[m_i^2(b) + \sigma_i^2(b) - 1 - \log \sigma_i^2(b) \right] \tag{70}$$

According to Theorem 4.1, the latent distribution $p_\theta(z) = p_\theta(z = x)$ minimizes the KL in the latent space: $D_{KL}(p_\theta(z) \| \mathcal{N}(0, I))$ (the exact assignment between data and latent codes is not unique, but all lead to the same latent estimate).

At this minimum, $p_\theta(z) = p_\theta(x = z)$. Since $p_\theta(x)$ is symmetric under rotations, it holds that, it holds that $m_i(b) = 0$ for $(a, b) = Qx$ in all rotations Q. However, since $\mathrm{Var}_{a \sim (Q_\# p_\theta)(a)}[a_i \mid b] \leq 1$ for all Q, it holds that

$$\alpha_{affine} = \frac{1}{2} \sum_{i}^{D-2} \mathbb{E}_b \left[\sigma_i^2(b) - 1 - \log \sigma_i^2(b) \right], \tag{71}$$

which evaluates to the same value regardless of Q, since the distribution is rotationally symmetric.

While this minimum may not be exactly achieved by a continuous volume-preserving flow, a sufficiently rich architecture is able to achieve universality $D_{KL}(p_\theta(z) \| \mathcal{N}(0, I)) \to D_{KL}(p(z) \| \mathcal{N}(0, I))$. Since the KL divergence implies the convergence of expectation values (Gibbs & Su, 2002), it holds that $\alpha_{affine} \to \frac{1}{2} \sum_i^{D-2} \mathbb{E}_b \left[\sigma_i^2(b) - 1 - \log \sigma_i^2(b) \right] \geq 0$. Equality holds if and only if $D_{KL}(p_\theta(z) \| \mathcal{N}(0, I)) = 0$. □

Note that the same argument also applies to Wasserstein distance and weak convergence, so this does not indicate that α_{affine} is more informative about convergence than these convergence metrics.

B.2. Fixing Volume-Preserving Flows by Learning the Latent Radial Distribution

By Theorem 4.1, the global minimizer of a volume-preserving flow is given by $p_\theta(z)$ in the latent space, as characterized by Lemma B.1. Both $p_\theta(z)$ and the standard normal distribution $p(z) = \mathcal{N}(0, C^2 I)$ are rotationally symmetric, so it is useful to make a change of variables to hyperspherical coordinates (r, ϕ):

$$p_\theta(z) = p_\theta(r(z)) \, p_\theta(\phi(z) \mid r(z)) \left| \frac{d(r, \phi)}{dz} \right|, \tag{72}$$

$$p(z) = p(r(z)) \, p(\phi(z) \mid r(z)) \left| \frac{d(r, \phi)}{dz} \right|. \tag{73}$$

The rotational symmetry implies that:

$$p_\theta(\phi \mid r) = p(\phi \mid r) = \frac{1}{A_{D-1}(r)}, \tag{74}$$

where $A_{D-1}(r)$ is the surface of the $(D-1)$-dimensional sphere of radius r.

This means that we only need to match $p_\theta(r)$ and $p(r)$. This can be achieved by a one-dimensional transformation of r.

Fixing the latent distribution can be done after training, as already training the volume-preserving flow with a standard normal in the latent space sorts the data such that $p(x_a) \geq p(x_b)$ implies that $r_a \leq r_b$, that is points of higher ground truth density are mapped to points closer to the origin. If we now replace the latent distribution with another distribution that fulfills the same constraint by fitting $p(r)$, the minimizer with this latent distribution remains the same. Thus, we can train with a standard normal and later fix the one-dimensional distribution.

B.3. Proof of Theorem 4.2

Theorem 4.2. The family of normalizing flows with constant Jacobian determinant $|\det f'(x)| = \text{const}$ is not a universal distribution approximator under KL divergence.

Proof. In this section, we want to present a two-dimensional example, for which no normalizing flow with constant Jacobian determinant can be constructed such that the KL-divergence between the data distribution and the distribution defined by the normalizing flow is zero.

$$p(x,y) = \begin{cases} 0.9 & \text{if } (x,y) \in [-0.5,0.5] \times [-0.5,0.5] \\ 0.9 - k \cdot (|x| - 0.5) & \text{if } |x| \in [0.5, \frac{0.9}{k} + 0.5] \wedge |y| \in [0,|x|] \\ 0.9 - k \cdot (|y| - 0.5) & \text{if } |y| \in [0.5, \frac{0.9}{k} + 0.5] \wedge |x| \in [0,|y|] \\ 0 & \text{else} \end{cases} \tag{75}$$

The data distribution $p(x,y)$ which has to be approximated by the model is defined in Equation (75). This data distribution has a constant value of 0.9 in a box centered around the origin with a side length of one. This region of constant density is skirted by a margin where the density decreases linearly to zero. Outside the decreasing region, the density is zero. The linear decline is governed by the constant k in Equation (75) which has to be chosen such that the density integrates to one. Since our example only requires the region of constant density but not the decaying tails of it, the exact functional form of the decaying regions are not relevant as long as they lead to a properly normalized distribution. Equation (75) only provides a possible definition of such a density.

To approximate this data distribution, a normalizing flow as defined in Section 3 is considered. In this example, we focus on normalizing flows with constant Jacobian determinant. To simplify notation, we define $1 = |\det f'(x)| = \text{const}$.

$$A = \{(x,y) \in \mathbb{R}^2 : 0.9 < p(x,y)\} \tag{76}$$

$$B = [-0.5,0.5] \times [-0.5,0.5] \tag{77}$$

$$A = B \setminus A \tag{78}$$

We choose $= 0.1$ and use this constant to define the set A (see Equation (76)). In addition we define B which is the region of the data space, where the data distribution has a constant value of 0.9 (see Equation (77)). A is the complement of A on B (see Equation (78)).

The aim of this example, is to find lower bounds for the KL-divergence between the data distribution and the distribution defined by the normalizing flow. To find these bounds we use Pinsker's inequality (Gibbs & Su, 2002) which links the total variation distance to the Kullback-Leibler divergence:

$$d_{TV}(p,\tilde{p}) = \sup_{A \text{ measurable}} |P(A) - \tilde{P}(A)| \leq \sqrt{\frac{1}{2} D_{KL}(p||\tilde{p})} \tag{79}$$

It is worth mentioning that constructing one measurable event for which $|P(A) - \tilde{P}(A)| > 0$ provides a lower bound for the total variation distance and therefore for the KL divergence.

To construct such an event, we consider two distinct cases, which consider different choice for the normalizing flow, characterized by the value of the absolute Jacobian determinant.

Case 1: $A = \emptyset$

This case arises if the absolute Jacobian determinant is so small, that the distribution defined by the normalizing flow never exceeds the limit defining A or if it is chosen so large, that the volume of A vanishes.

In this case, we find $A = B$ and $|A| = 1$ where $|A|$ denotes the volume of the data space occupied by A. Using the fact that the data distribution has a constant value of 0.9 in B and that $\tilde{p} < 0.9$ in $A = B$,

$$P(A) - \tilde{P}(A) = 0.9 - \tilde{P}(A) \tag{80}$$

$$> 0.9 - 1 + (0.9 -) \tag{81}$$

$$= > 0 \tag{82}$$

Using Equation (82) as a lower bound for the total variation distance Equation (82) we can apply Equation (79) to find Equation (85) as a lower bound for the KL divergence.

$$D_{KL}(p||p') \geq 2\,\delta_{TV}(p;p')^2 \tag{83}$$
$$\geq 2\,\varepsilon^2 \tag{84}$$
$$= 0.02 \tag{85}$$

Case 2: $A \neq \emptyset$;

Inserting the definition of $p(x;y)$ as given in Equation (1) into the definition of A (see Equation (76)) and rewriting the condition defining the set yields Equation (86).

$$A = \{(x;y) \in \mathbb{R}^2 : \frac{0.9}{J} < p(z = f(x;y))\} \tag{86}$$

This defines a set C in the latent space which is defined in Equation (87).

$$C = \{z \in \mathbb{R}^2 : \frac{0.9}{J} < p(z)\} \tag{87}$$

Since the normalizing flows considered in this example have a constant Jacobian, the volume of A in the data space is directly linked to the volume of C in the latent space via Equation (88).

$$|A| = \frac{1}{J}\,|C| \tag{88}$$

The definition of C Equation (87) shows, that C is a circle around the origin of the latent space. To determine the volume of C we compute the radius of this circle. This is done by inserting the definition of the latent distribution, which is a two-dimensional standard normal distribution, into the condition defining C (see Equation (87)). This yields Equation (89). Since the latent distribution is rotational invariant, one can simply look at it as a function of the distance r from the origin. Solving the for r leads to Equation (90).

$$\frac{0.9}{J} < \frac{1}{2\pi}\exp\left(-\frac{r^2}{2}\right) \tag{89}$$
$$\Leftrightarrow\quad r = \sqrt{-2\log\left(\frac{2\pi(0.9)}{J}\right)} \tag{90}$$

Inserting Equation (90) into the formula for the area of a circle and using Equation (88), yields Equation (92) as an expression for the volume of A. The lower bound for the volume of A arises from finding the local maximum (which is also the global maximum) of Equation (92) with respect to the absolute Jacobian determinant J.

$$|A| = \frac{1}{J}\,\pi r^2 \tag{91}$$
$$= \frac{1}{J}\,2\pi\log\left(\frac{J}{2\pi(0.9)}\right) \tag{92}$$
$$\frac{1}{e\,(0.9\pi)} \tag{93}$$
$$\tag{94}$$

As in the previous case, we now compute $P(A) - P(A')$.

$$P(A) \quad P(A) = |A| \quad 0.9 \quad P(A) \tag{95}$$

$$|A| \quad 0.9 \quad |A| \quad (0.9 \quad) \tag{96}$$

$$= |A| \tag{97}$$

$$(1 \quad |A|) \tag{98}$$

$$1 \quad \frac{1}{e \quad (0.9 \quad)} \tag{99}$$

Using Equation (83) and Equation (99) as a lower bound for the total variation distance and inserting our choice for yields for Equation (100) as a lower bound for the KL divergence between the data distribution and the distribution de ned by the normalizing ow.

$$D_{KL}(p||p) \quad 2 \quad 1 \quad \frac{1}{e \quad (0.9 \quad)}^{2} \tag{100}$$

$$0.0058 \tag{101}$$

We can conclude that we have derived lower bounds for the KL divergence between the data distribution and the distribution de ned by the normalizing ow, which cannot be undercut by any normalizing ow with a constant absolute Jacobian determinant. Therefore, we have proven that the class of normalizing ows with constant (absolute) Jacobian determinant cannot approximate arbitrary continuous distributions if one uses the KL divergence as a convergence measure.□

B.4. Proof of Proposition 4.3

De nition B.5. Given a probability density $p(x)$ and a connected set $M \quad R^D$. Then, M is called a mode of $p(x)$ if

$$p(x) = p(y) \quad 8x; y \ 2 \ M; \tag{102}$$

and there is a neighborhood U of M such that:

$$p(x) > p(y) \quad 8x \ 2 \ M; y \ 2 \ U \ n \ M: \tag{103}$$

With this de nition of a mode, let us characterize the correspondence between modes of $p(x)$ and $p(z)$ for a volume-preserving ow:

Lemma B.6. Given a latent probability density $p(z)$, a diffeomorphism $f : R^D \ ! \ R^D$ with constant Jacobian determinant $|f'(x)| = const$ and a mode $M \quad R^D$. Then, $f(M)$ is a mode of $p(x)$.

Proof. We show that $f(M)$ ful lls De nition B.5. First, for every $x; y \ 2 \ f(M)$: The pre-images of $x; y$ are unique in M as f is bijective, that is $f^{-1}(x); f^{-1}(y) \ 2 \ M$. As M is a mode:

$$p(f^{-1}(x)) = p(f^{-1}(y)): \tag{104}$$

We follow:

$$(f_] p)(x) = p(f^{-1}(x))|J| = p(f^{-1}(y))|J| = (f_] p)(y); \tag{105}$$

where we have used the change-of-variables formula for bijections and $|J| = const$.

Let U be a neighborhood of M such that Equation (103) is ful lled. As f is continuous, there is a neighborhood V of $f(M)$ such that $V \quad f(U)$. Consider $x \ 2 \ f(M); y \ 2 \ V \ n \ f(M)$. As M is a mode:

$$p(f^{-1}(x)) > p(f^{-1}(y)): \tag{106}$$

Multiplying both sides by $|J|$, we nd:

$$(f_] p)(x) = p(f^{-1}(x))|J| > p(f^{-1}(y))|J| = (f_] p)(y): \tag{107}$$

Thus, $f(M)$ is a mode of $(f_] p)(x)$ by De nition B.5. □

This makes us ready for the proof:

Proposition 4.3. A normalizing flow $p(x)$ with constant Jacobian determinant $|f'(x)| = \text{const}$ has the same number of modes as the latent distribution $p(z)$.

Proof. By Lemma B.6, every mode of $p(x)$ implies a mode of $(f_\sharp p)(x)$. Also, every mode of $(f_\sharp p)(x)$ implies a mode of $(f^{-1}_\sharp f_\sharp p)(x) = p(x)$. Therefore, there is a one-to-one correspondence of modes between $p(x)$ and $(f_\sharp p)(x)$. □

C. Proofs on Affine Coupling Flows

C.1. Proof of Unique Fixed Point of Affine Coupling Blocks

C.1.1. Underlying Results from Previous Work

Here, we restate the results from the literature that our main proof is based on:

First, (Eaton, 1986) show that if for some vector-valued random variable X and every pair of orthogonal projections the mean of one projection conditioned on the other is zero, X follows a spherical distribution:

Theorem C.1 (Eaton (1986)). Suppose the random vector $X \in \mathbb{R}^D$ has a finite mean vector. Assume that for each vector $v \neq 0$ and for each vector u perpendicular to v (i.e. $u \cdot v = 0$):

$$\mathbb{E}[u \cdot X \,|\, v \cdot X] = 0. \tag{108}$$

Then X is spherical and conversely.

Secondly, Cambanis et al. (1981, Corollary 8a) identifies the Gaussian from all elliptically contoured (which includes spherical) distributions. We write it in the form of Bryc (1995, Theorem 4.1.4):

Theorem C.2 (Bryc (1995)). Let $p(x)$ be radially symmetric with $\mathbb{E}[\|x\|^\alpha] < \infty$ for some $\alpha > 0$. If

$$\mathbb{E}[\|x_{1,\dots,m}\|^\alpha \,|\, x_{m+1,\dots,n}] = \text{const}, \tag{109}$$

for some $1 \leq m < n$, then $p(x)$ is Gaussian.

Finally, Draxler et al. (2020, Theorem 1) show that the explicit form of the maximally achievable loss improvement by an affine coupling block $\Delta_{\text{affine}}(p(z))$ if the data is rotated by a fixed rotation layer Q is given by (omitting the dependence on $p(z)$ to avoid clutter):

$$\Delta_{\text{affine}}(Q) = D_{KL}(p_0(z) \| p(z)) - \min_{s,t} D_{KL}(p_{s,t|Q}(z) \| p(z)) \tag{110}$$

$$= \tfrac{1}{2}\mathbb{E}_b\left[m_i(b)^2 + \sigma_i(b)^2 - 1 - \log \sigma_i(b)^2\right]. \tag{111}$$

Here, s, t are the scaling and translation in an affine coupling block (see Equation (8)), and we optimize over continuous functions for now. By $p_{s,t|Q}(z)$ we denote the latent distribution achieved if $(a, b) = s(b) \odot a + t(b)$ is applied to $p_0(a, b)$, the rotated version of the incoming data $p(z)$. The symbols $m_i(b), \sigma_i(b)^2$ are conditional moments of the active dimensions a_i conditioned on the passive dimensions b:

$$m_i(b) = \mathbb{E}_{a_i|b}[a_i], \qquad \sigma_i(b) = \mathbb{E}_{a_i|b}[a_i^2] - m_i(b)^2. \tag{112}$$

These conditional moments are continuous functions if $p(x)$ is a continuous distribution and $p(b) > 0$ for all passive $b \in \mathbb{R}^{D-2}$. The improvement in Equation (111) is achieved by the affine coupling block with the following subnetwork:

$$s_i(b) = \frac{1}{\sigma_i(b)}, \qquad t_i(b) = \frac{m_i(b)}{\sigma_i(b)}. \tag{113}$$

Note that $s(b)$ and $t(b)$ are continuous functions and not actual neural networks. In the next section, we show that a similar statement on practically realizable neural networks that is sufficient for our universality.

C.1.2. Relation to practical neural networks (Lemma 5.3)

Before moving to the proof of Theorem 5.2, we show the helper statement Lemma 5.3. For reference, let us repeat the definition of the loss improvement by an affine coupling block $\Delta_{\text{affine}}(p(z))$:

Definition 5.1. Given a normalizing flow $p(x)$ on a continuous probability distribution $p(x)$ with finite first and second moment and $p(x) > 0$ everywhere. Then, we define the loss improvement by an affine coupling block as:

$$\Delta_{\text{affine}}(p(z)) := D_{KL}(p(z)\|p(z)) - \min_{\phi} D_{KL}(p_{[\phi]}(z)\|p(z)); \tag{114}$$

where $\phi = (Q, \theta)$ parameterizes a single bi-Lipschitz affine coupling block whose conditioner neural network has at least two hidden layers of finite width and ReLU activations and $p_{[\phi]}(z) = (f_{[\phi]}p)(z)$.

To relate Equations (111) and (113) to actually realizable networks, which can exactly follow the arbitrary continuous functions $s_i(b), t_i(b)$, the following statement asserts that the fix point of adding coupling layers with infinitely expressive conditioner functions is the same as actually realizable and well-conditioned coupling blocks:

Lemma 5.3. Given a continuous probability density $p(z)$ on $z \in \mathbb{R}^k$. Then,

$$\Delta_{\text{affine}}(p(z)) > 0 \tag{115}$$

if and only if:

$$\Delta_{\text{affine}}(p(z)) > 0. \tag{116}$$

Proof. First, note that $\Delta_{\text{affine}}(Q) \geq \Delta_{\text{affine}}(Q) \geq 0$ since no practically realizable coupling block can achieve better than Equation (111). Thus, if $\Delta_{\text{affine}}(Q) = 0$, so is $\Delta_{\text{affine}}(Q) = 0$.

For the reverse direction, we fix $Q = I$, and otherwise consider a rotated version of p. Also, without loss of generality, we consider one single active dimension i in the following, but the construction can then be repeated for each other active dimension.

If we apply any affine coupling layer $f_{\text{cpl};\theta}(a; b) = s_\theta(b)a + t_\theta(b)$, the loss change by this layer can be computed from the theoretical maximal improvement $\Delta_{\text{affine}}(Q)$ before and after adding this layer $\tilde{\Delta}_{\text{affine}}(I)$:

$$\Delta_{\text{affine}}(I) = \Delta_{\text{affine}}(I) - \tilde{\Delta}_{\text{affine}}(I) = \tfrac{1}{2}E_b\left[m_i(b)^2 + \sigma_i(b)^2 - 1 - \log \sigma_i(b)^2\right] - \tfrac{1}{2}E_b\left[\tilde{m}_i(b)^2 + \tilde{\sigma}_i(b)^2 - 1 - \log \tilde{\sigma}_i(b)^2\right]. \tag{117}$$

The moments after the affine coupling layers read:

$$\tilde{m}_i(b) = s_\theta(b)m_i(b) + t_\theta(b); \qquad \tilde{\sigma}_i(b) = s_\theta(b)\,\sigma_i(b). \tag{118}$$

Case 1: $E_b[\sigma_i(b)^2 - 1 - \log \sigma_i(b)^2] > 0$:

Then, without loss of generality, by continuity and positivity of $\sigma_i(b)$ and consequential continuity of $\sigma_i(b)$ in b, there is a convex open set $A \subseteq \mathbb{R}^{D-2}$ with non-zero measure $p(A) > 0$ where $\sigma_i(b) > 1$. If $\sigma_i(b) < 1$ everywhere, apply the following argument flipped around $\sigma_i(b) = 1$.

Denote by $\sigma_{\max} = \max_{b \in A} \sigma_i(b)$. Then, by continuity of $\sigma_i(b)$ there exists $B \subseteq A$ so that $\sigma_i(b) > (\sigma_{\max} + 1)/2 + 1 =: \sigma_{\max}/2$ for all $b \in B$. Let $C \subseteq B$ be a multidimensional interval $[l_1, r_1] \times \cdots \times [l_{D-2}, r_{D-2}]$ with $p(C) > 0$ inside B.

Now, we construct a ReLU neural network with two hidden layers with the following property, where $E \subseteq C$ are specified later with $p(F) > p(E) > 0$:

$$\begin{cases} f_\theta(x) = \frac{1}{\sigma_{\max}/2} & x \in E \\ \frac{1}{\sigma_{\max}/2} \leq f_\theta(x) < 1 & x \in D \setminus E \\ f_\theta(x) = 0 & \text{else.} \end{cases} \tag{119}$$

To do so, we make four neurons for each dimension $i = 1, \ldots, D-2$:

$$\text{ReLU}(x_i - l_i), \text{ReLU}(x_i - l_i - \varepsilon), \text{ReLU}(x_i - r_i), \text{ReLU}(x_i - r_i + \varepsilon); \tag{120}$$

where $0 < < \min_i (r_i \quad l_i)=4$. If we add these four neurons with weights $1; \ 1; \ 1; 1$, we find the following piecewise function:

$$\begin{cases} 0 & x \quad l_i \\ x \quad l_i & l_i < x < l_i + \\ & l_i + \quad x \quad r_i \\ r_i \quad x & r_i \quad < x < r_i \\ 0 & r_i \quad x: \end{cases} \tag{121}$$

If we repeat this for each dimension and add together all neurons with the corresponding weights into a single neuron in the second layer, then only inside $D = (l_1 + ; r_1 \quad) \quad (l_{D=2} + ; r_{D=2} \quad)$ C the weighted sum would equal $D = 2$. By choosing as above, this region has nonzero volume. We thus equip the single neuron in the second layer with a bias of $D=2 +$ for some $<$, so that it is constant with value inside $E = (l_1 + \quad ; r_1 \quad +) \quad (l_{D=2} + \quad + ; r_{D=2} \quad)$ D and smoothly interpolates to zero in the rest of D.

For the output neuron of our network, we choose weight $(w_{max=2} \quad 1)=$ and bias 1. By inserting the above construction, we find the network specified in Equation (119).

Now, for all $b \ 2 \ D$,

$$1 < \tilde{} _i (b) < \quad _i (b); \tag{122}$$

so that

$$m_i (b)^2 + \tilde{} _i (b)^2 \quad 1 \quad \log \tilde{} _i (b)^2 < m_i (b)^2 + \quad _i (b)^2 \quad 1 \quad \log \quad _i (b)^2: \tag{123}$$

Thus, parameters exist that improve on the loss. (Note that this construction can be made more effective in practice by identifying the sets where $> \ 1$ resp. $< \ 1$ and then building neural networks that output one or scale towards $s(b) = 1$ everywhere. Because we are only interested in identifying improvement, the above construction is sufficient.)

Now, regarding t , we focus on $E_b[m_i (b)^2] > 0$ (otherwise choose $t = 0$ as a constant, which corresponds to a ReLU network with all weights and biases set to zero):

$$E_b[m_i (b)^2] > E_b[(s' (b) m_i (b) + t' (b))^2]: \tag{124}$$

By (Hornik, 1991, Theorem 1) there always is a t that fulfills this relation.

Case 2: $E_b[\ _i (b)^2 \quad 1 \quad \log \ _i (b)^2] = 0$. Then, choose the neural network $s(b) = 1$ as a constant. As $_{affine} > 0$, $E_{b \ p(a;b)}[m_i (b)^2] > 0$ and we can use the same argument for the existence case as before.

It is left to show that a L -bi-Lipschitz coupling block can be constructed. To achieve this, replace the action of the coupling block $a_i = s(b) a_i + t(b)$ by $a_i = (s(b) a_i + t(b)) + (1 \quad) a_i$. Since $s(b)$ and $t(b)$ above were constructed to move the data in the right direction, we obtain a finite loss improvement, since > 0, $_{affine} > 0$. The Jacobian J of the restricted coupling block is $jf'_{cpl} j = jf'_{original} j + (1 \quad)I$. Since the eigenvectors are unchanged, all eigenvalues $^{(i)}_{original}$ of $jf'_{original} j$ are modified to $^{(i)} = \ ^{(i)}_{original} + (1 \quad)$. This moves all eigenvalues closer to 1. Choose > 0 such that $\min_i \ ^{(i)} \ L^{-1}$ and $\max_i \ ^{(i)} \ L$ to achieve L -bi-Lipschitzness. □

C.1.3. Proof of Theorem 5.2

The following theorem based on Lemma 5.3 shows that $_{affine}(p \ (z))$ is a useful measure of convergence to the standard normal distribution:

Theorem 5.2. With the definitions from Definition 5.1:

$$p \ (z) = \ N (z; 0; I) \ () \quad _{affine}(p \ (z)) = 0 : \tag{125}$$

Proof. The forward direction is trivial: $p(z) = \ N (0; I)$ and therefore $D_{KL}(p(z) \| N (0; I)) = 0$. As adding an identity layer is a viable solution to Equation (23), there is a s with $D_{KL}(p \ (z) \| N (0; I)) = 0$, and thus $_{affine}(p \ (z)) = 0$.

For the reverse direction, start with $_{affine}(p \ (z)) = 0$. Then, by Lemma 5.3, also $_{affine}(p \ (z)) = 0$.

Figure 4. The normalizing flow we construct in our proof is remarkably simple: We iteratively add coupling blocks, optimizing the parameters of the new block while keeping previous parameters fixed. Theorem 5.2 shows that if adding another blocks shows no improvement in the loss, the flow has converged to a standard normal distribution in the latent space. Since the total loss that can be removed is finite, the flow converges.

The maximally achievable loss improvement for any rotation Q is then given by Equation (31):

$$\ell_{\text{affine}}(p(z)) = \max_Q \frac{1}{2}\sum_{i=1}^{D-2} \mathbb{E}_b\left[m_i(b)^2 + \sigma_i(b)^2 - 1 - \log \sigma_i(b)^2\right] = 0. \tag{126}$$

It holds that both $x^2 \geq 0$ and $x^2 - 1 - \log x^2 \geq 0$. Thus, the following two summands are zero:

$$0 = \tfrac{1}{2}\mathbb{E}_b\left[m_i(b)^2\right], \tag{127}$$

$$0 = \tfrac{1}{2}\mathbb{E}_b\left[\sigma_i(b)^2 - 1 - \log \sigma_i(b)^2\right]. \tag{128}$$

This holds for all Q since the maximum over Q is zero.

By continuity of $p(b)$ and $m_1(b)$ in p, this implies for all b:

$$\mathbb{E}_{a_1|b}[a_1] = 0. \tag{129}$$

Fix b_1 and marginalize out the remaining dimensions $b_{2,\dots,D-2}$ to compute the mean of a_1 conditioned on b_1:

$$m_{a_1|b} = \mathbb{E}_{a_1|b_1}[a_1] = \mathbb{E}_{b_{1,\dots,D-2}}[\mathbb{E}_{a_1|b}[a_1]] = \mathbb{E}_{b_{1,\dots,D-2}}[0] = 0. \tag{130}$$

As a_1 and b_1 are arbitrary orthogonal directions since the above is valid for any Q, we can employ Theorem C.1 to follow that $p(x)$ is spherically symmetric.

We are left with showing that for a spherical $p(x)$, if for all Q there is no improvement $\ell_{\text{affine}}(Q)$, then $p(x) = \mathcal{N}(0, I)$.

Without loss of generality, we can fix $Q = I$, as $(Q_\sharp p)(x) = p(x)$ for all Q. We write $x = (p, a)$.

As $\ell_{\text{affine}} = 0$, we can follow $\sigma_i(b) = 1$ like above. This implies that:

$$\mathbb{E}_{a|b}[\|a\|^2] = \sum_{i=1}^{D-2}(m_i(b)^2 + \sigma_i(b)^2) = D-2. \tag{131}$$

In particular, this is independent of b and we can thus apply Theorem C.2 with $\sigma = 2$.

Finally, $m(b) = 0$ and $\sigma_i(b) = 1$ for all Q imply that $p(x) = \mathcal{N}(0, I)$. □

C.2. Proof of Theorem 5.4

Theorem 5.4. For every continuous $p(x)$ with finite first and second moment with in finite support, there is a sequence of normalizing flows $f_n(x)$ consisting of n affine coupling blocks such that:

$$p_n(z) \xrightarrow{n\to\infty} \mathcal{N}(z; 0, I), \tag{132}$$

in the sense that $\varepsilon_{\text{af ne}}(p_n(z)) \to 0$.

Proof. The proof idea of iteratively adding new layers which are trained without changing previous layers is visualized in Figure 4.

Let us consider a coupling-based normalizing flow of depth n and call the corresponding latent distributions $p_n(z)$, where $n = 0$ corresponds to the initial data distribution $p(x)$. Denote by $L_n = D_{KL}(p_n(z)\|p(z))$ the corresponding loss. Then, if we add another layer to the flow, we achieve a difference in loss of: $\varepsilon_{\text{af ne}}(p_n(z)) = L_n - L_{n+1}$.

Without loss of generality, we may assume that the rotation layer of each block can be chosen freely. Otherwise, add 48 coupling blocks with fixed rotations that together exactly represent the Q we want, as shown by Koehler et al. (2021, Theorem 2).

We construct the blocks of the flow iteratively: Choose the rotation and subnetwork parameters θ_{n+1} of each additional block such that the block maximally reduces the loss, keeping the parameters of the previous blocks $\theta_{1,\dots,n}$ fixed. Then, $\varepsilon_{\text{af ne}}(p_n(z))$ attains the value given in Equation (24) (Equation (114)):

$$\varepsilon_{\text{af ne}}(p_n(z)) = L_n - L_{n+1} = D_{KL}(p_{\theta_{1,\dots,n}}(z)\|p(z)) - \min_{\theta_{n+1}} D_{KL}(p_{\theta_{1,\dots,n}[\theta_{n+1}}(z)\|p(z)) \geq 0; \tag{133}$$

Each layer contributes a non-negative improvement in the loss which can at most sum up to the initial loss:

$$\sum_{k=0}^{n-1} \varepsilon_{\text{af ne}}(p_k(z)) = L_0 - L_n \leq L_0 \quad \text{for all } n \geq 1; \tag{134}$$

and the inequality is due to $L_n \geq 0$. For a non-negative series that is bounded above, the terms of the series must converge to zero (Rudin, 1976, Theorems 3.14 and 3.23), which shows convergence in terms of Section 5.2:

$$\sum_{n=0}^{\infty} \varepsilon_{\text{af ne}}(p_n(z)) \leq L_0 < \infty) \varepsilon_{\text{af ne}}(p_n(z)) \to 0. \tag{135}$$

□

C.3. Relation to Convergence in KL

Corollary C.3. Given a series of probability distributions $p_n(z)$. Then, convergence in KL divergence

$$D_{KL}(p_n(z)\|\mathcal{N}(0;1)) \to 0 \tag{136}$$

implies convergence in the loss improvement by a single af ne coupling as in De nition 5.1:

$$\varepsilon_{\text{af ne}}(p_n(z)) \to 0. \tag{137}$$

Proof. By assumption, for every $\varepsilon > 0$ there exists $N \geq N$ such that:

$$D_{KL}(p_n(z)\|\mathcal{N}(0;1)) < \varepsilon \quad \forall n > N. \tag{138}$$

This implies convergence of $\varepsilon_{\text{af ne}}(p_n(z))$, by the following upper bound via the sum of all possible future improvements which is bounded from above by the total loss:

$$\varepsilon_{\text{af ne}}(p_n(z)) \leq \sum_{m=n}^{\infty} \varepsilon_{\text{af ne}}(p_m(z)) \leq D_{KL}(p_n(z)\|\mathcal{N}(0;1)) < \varepsilon \quad \forall n > N. \tag{139}$$

□

C.4. Convergence in Wasserstein but not in KL Divergence

In Section 5.1 we argued that convergence under Wasserstein distance $W_2(p, p_n) \to 0$ does not imply convergence under KL divergence $D_{KL}(p\|p_n) \to 0$. We illustrate that via an example:

Take a standard normal target $p(x) = \mathcal{N}(0, 1)$ in 1D and approximate by a mixture of n distributions:

$$p_n(x) = \sum_{i=1}^{n} p([i\,a_n, (i+1)\,a_n])\ \mathcal{U}(x - a_n\,i). \tag{140}$$

This mixture splits the input space into bins of width a_n, and positions a peak at the left of each bin, weighted by the amount of mass in the bin in the target distribution.

The optimal transport plan underlying the Wasserstein distance redistributes the weight from the left edge of each bin over the entire bin. This means that the total distance any point has to travel under the optimal transport plan is $\leq a_n$. Thus it holds that $W_2(p, p_n) \leq a_n$. If we choose $a_n \to 0$, so does $W_2(p, p_n) \to 0$.

The KL divergence can be lower bounded by the total variation via Pinsker's inequality (see also Equation (79)):

$$\mathrm{TV}(p, p_n) = \sup_{A \text{ measurable}} |P(A) - P_n(A)| \leq \sqrt{\frac{1}{2} D_{KL}(p\|p_n)} \tag{141}$$

The set of all bin interiors $I = \bigcup_{i=1}^{\infty} (i\,a_n, (i+1)\,a_n)$ is measurable. It holds that $P(I) = 1$ (since we only exclude the zero-set of bin edges to get I from $\mathbb{R}$). Also, $P_n(I) = 0$ since all the mass is concentrated at the bin edges, and so $1 - \mathrm{TV}(p, p_n) \leq \sqrt{\frac{1}{2} D_{KL}(p\|p_n)}$ regardless of a_n.

Thus $D_{KL}(p\|p_n) \geq 2 > 0$ regardless of n, but $W_2(p, p_n) \to 0$.

Intuitively, the construction in Koehler et al. (2021) is related to the mixture above. The vanishing scaling terms from latent to data space in their universality proof squeeze the distribution. The translation terms ensure that this squeezed distribution is distributed over the space such that the error in terms of Wasserstein distance is bounded by the grid length.

D. Benefits of More Expressive Coupling Blocks

To see what the best improvement for an infinite capacity coupling function can ever be, we make use of the following Pythagorean identity combined from variants in Chen & Gopinath (2000); Cardoso (2003); Draxler et al. (2022):

$$L = D_{KL}(p(z)\|\mathcal{N}(0, I)) = B + \mathop{\mathbb{E}}_{b \sim p(a;b)}[D(b) + J(b) + S(b)]. \tag{142}$$

The symbols $B, D(b), J(b), S(b)$ all denote KL divergences:

The first two terms remain unchanged under a coupling layer: The KL divergence to the standard normal in the passive dimensions $B = D_{KL}(p(b)\|\mathcal{N}(0, I_{D=2}))$, which are left unchanged. The dependence between active dimensions $D(b) = D_{KL}(p(a|b)\|p(a_1|b) \cdots p(a_{D=2}|b))$ measures the multivariate mutual information between active dimensions. It is unchanged because each dimension is treated conditionally independent of the others (Chen & Gopinath, 2000).

The remaining terms measure how far each dimension $p(a_i|b)$ differs from the standard normal: The negentropy measures the divergence to the Gaussian with the same first moments $p(a_i|b)$ in each dimension, summing to $J(b) = \sum_{i=1}^{D=2} D_{KL}(p(a_i|b)\|\mathcal{N}(m_i(b), \sigma_i(b)))$. Finally, the non-Standardness $S(b) = \sum_{i=1}^{D=2} D_{KL}(\mathcal{N}(m_i(b), \sigma_i(b))\|\mathcal{N}(0, 1))$ measures how far these 1d Gaussian are away from the standard normal distribution.

Note that the total loss L is invariant under a rotation of the data. The rotation does, however, affect how that loss is distributed into the different components in Equation (142).

If we restrict the coupling function to be affine-linear $f(a_i; \cdot) = s a_i + t$ (i.e. a RealNVP coupling), then this means that also $J(b)$ is left unchanged, essentially because $p(a_i|b)$ and $\mathcal{N}(m_i(b), \sigma_i(b))$ undergo the same transformation (Draxler et al., 2022, Lemma 1). Only a nonlinear coupling function $f(a_i; \cdot)$ can thus affect $J(b)$ and reduce it to $J(b) < J(b)$.

Taking the loss difference between two layers, we find Equation (36).

E. Experimental Details

We base our code on PyTorch (Paszke et al., 2019), Numpy (Harris et al., 2020), Matplotlib (Hunter, 2007) for plotting and Pandas (McKinney, 2010; The pandas development team, 2020) for data evaluation.

E.1. Layer-Wise Flow

In experiment on a toy dataset for Figure 1, we demonstrate that a coupling flow constructed layer by layer as in Equation (28) learns a target distribution. We proceed as follows:

We construct a data distribution on a circle as a Gaussian mixture of M Gaussians with means $m_i = (r\cos\varphi_i, r\sin\varphi_i)$, where $\varphi_i = 0, \frac{1}{M}2\pi, \ldots, \frac{M-1}{M}2\pi$ are equally spaced, and $r = 0.3$. The advantage of approximating the ring with this construction is that this yields a simple to evaluate data density, which we need for accurately plotting $p(x)$:

$$p(x) = \frac{1}{M}\sum_{i=1}^{M} \mathcal{N}(x; m_i, \sigma^2 I). \tag{143}$$

We then fit a total 100 layers in the following way: First, treat $p(x)$ as the initial guess for the latent distribution. Then, we build the affine coupling block that maximally reduces the loss using Equation (28). We therefore need to know the conditional mean $m(b)$ and standard deviation $\sigma(b)$ for each b. We approximate this from a finite number of samples N which are grouped by the passive coordinate into B bins so that N/B samples are in each bin. We then compute the empirical mean m_i and standard deviation σ_i over the active dimension in each bin $i = 1, \ldots, B$. According to Equation (28), we define $s_i = \frac{1}{\sigma_i}$ and $t_i = -\frac{1}{\sigma_i}m_i$ at the bin centers and interpolate between bins using a cubic spline. Outside the domain of the splines, we extrapolate with constants with the value of the closest bin. We do not directly optimize over Q, but choose the Q that reduces the loss most out of N_Q random 2d rotation matrices.

We limit the step size of each layer to avoid artifacts from finite training data, by mapping:

$$x \leftarrow \lambda x + (1-\lambda) f_{\text{blk}}(x). \tag{144}$$

In addition, we resample the training data from the ground truth distribution after every step to avoid overfitting. We do not explicitly control for the bi-Lipschitz constant of our coupling blocks because we do not encounter any numerical problems.

We choose $N = 2^{26}$, $B = 64$, $M = 20$, $\lambda = 0.5$, $N_Q = 10$. The resulting flow has $64 \cdot 2 \cdot 100 = 12{,}800$ learnable parameters. Figure 5 shows how the KL divergence vanishes for our layer-wise training, together with a fine-tuning.

E.2. Volume-Preserving Normalizing Flows

The target distribution is a two-dimensional Gaussian Mixture Model with two modes. The two modes have the same relative weight but different covariance matrices ($\Sigma_1 = I \cdot 0.2$, $\Sigma_2 = I \cdot 0.1$) and means ($m_1 = [-0.5, -0.5]$, $m_2 = [0.5, 0.5]$).

The normalizing flow with a constant Jacobian determinant consists of 15 GIN coupling blocks as introduced in Sorrenson et al. (2019). This type of coupling blocks has a Jacobian determinant of one. To allow for a global volume change, a layer with a learnable global scaling is added after the final coupling block. This learnable weight is initialized as one. For the normalizing flow with variable Jacobian determinant, the GIN coupling is modified by removing the normalization of the scaling factors in the affine couplings. This allows the normalizing flow to have variable Jacobian determinants. In this case, the global scaling block is omitted. To implement the normalizing flow, we use the FrEIA package (Ardizzone et al., 2018a) implementation of the GIN coupling blocks.

In both normalizing flows, the two subnetworks used to compute the parameters of the affine couplings are fully connected neural networks with two hidden layers and a hidden dimensionality of 128. ReLU activations are used. The weights of the linear layers of the subnetworks are initialized by applying the PyTorch implementation of the Xavier initialization (Glorot & Bengio, 2010). In addition, the weights and biases of the final layer of each subnetworks are set to zero.

The networks are trained using the Adam (Kingma & Ba, 2017) with PyTorch's default settings and an initial learning rate of $1 \cdot 10^{-3}$ which is reduced by a factor of ten after 5000, 10000 and 15000 training iterations. In total, the training ran for 25000 iterations. In each iteration, a batch of size 128 was drawn from the target distribution to compute the negative log likelihood objective. We use a standard normal distribution as the latent distribution.

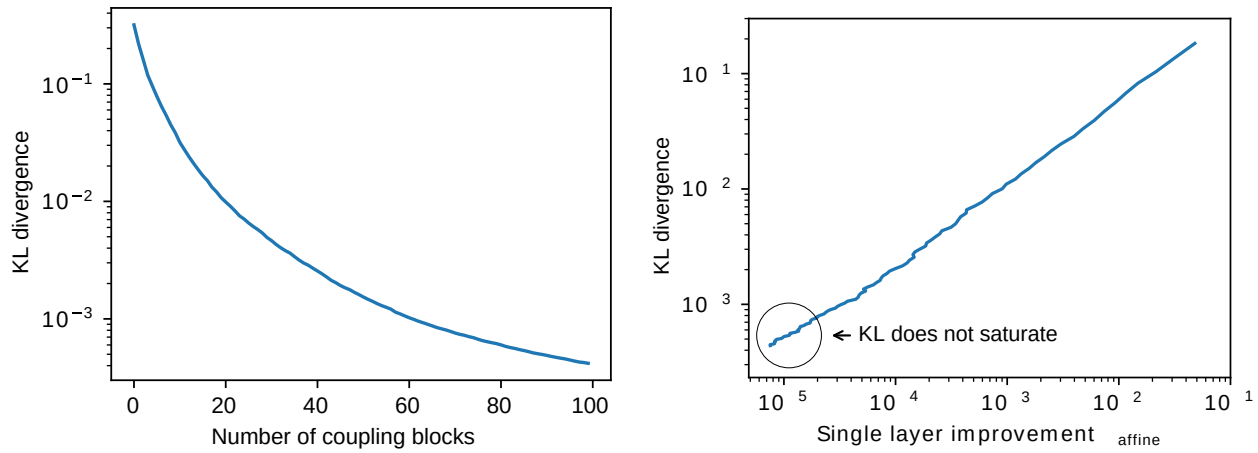

Figure 5. Empirically, the KL divergence decreases as more coupling blocks are added for the toy distribution considered in Figure 1 *(left)*. At the same time, there is a strong correlation between the loss improvement by a single coupling block $_{\text{affine}}(p_\theta(z))$ and the KL divergence *(right)*. Crucially, the KL divergence does not saturate as the loss improvements become smaller. The coupling flow considered is trained according to the greedy layer-wise training in our proof construction.

For obtaining the optimal distribution $p\ (x)$, we follow the grid procedure in Section 4 and compute the probabilities on a regular 400×400 grid of grid spacing 0.01. The covariance of $p\ (z)$ is computed for the latent scaling layer by sampling 4096 points from the mixture model, moving them according to the volume-preserving flow learned using the grid and computing their empirical covariance matrix. This yields essentially the same scaling as obtained from training the volume-preserving flow.

In order to learn $p\ (r)$ as in Appendix B.2, we sample $5 \cdot 10^8$ samples from ground truth data distribution. These samples are passed through the volume-preserving flow and afterwards the L_2 norm is applied to the latent codes to obtain the latent radii r. We subtract the smallest observed radius from all radii to ensure that the distribution we construct is supported for all $r \geq 0$ and fit a histogram with 4200 bins to the radii. To obtain a smoother distribution, we use the left bin edges of the histogram and the corresponding density values to fit a cubic spline using SciPy's `interpolate` package (Virtanen et al., 2020). We choose the partition function of the distribution $p_r(r)$ defined by the spline such that it integrates to one. For a given latent code z the latent density can be computed by evaluating p_r at $r = \|z\|_2$ and correcting for the volume at the given radius (see Equation (145)).

$$p(z) = \frac{1}{2\pi\|z\|_2} \cdot p_r(r = \|z\|_2) \tag{145}$$