A Geometric Explanation of the Likelihood OOD Detection Paradox

Hamidreza Kamkari [1 2 3] **Brendan Leigh Ross** [1] **Jesse C. Cresswell** [1] **Anthony L. Caterini** [1]
Rahul G. Krishnan [2 3] **Gabriel Loaiza-Ganem** [1]

Abstract

Likelihood-based deep generative models (DGMs) commonly exhibit a puzzling behaviour: when trained on a relatively complex dataset, they assign higher likelihood values to out-of-distribution (OOD) data from simpler sources. Adding to the mystery, OOD samples are never generated by these DGMs despite having higher likelihoods. This two-pronged paradox has yet to be conclusively explained, making likelihood-based OOD detection unreliable. Our primary observation is that high-likelihood regions will not be generated if they contain minimal probability mass. We demonstrate how this seeming contradiction of large densities yet low probability mass can occur around data confined to low-dimensional manifolds. We also show that this scenario can be identified through local intrinsic dimension (LID) estimation, and propose a method for OOD detection which pairs the likelihoods and LID estimates obtained from a *pre-trained* DGM. Our method can be applied to normalizing flows and score-based diffusion models, and obtains results which match or surpass state-of-the-art OOD detection benchmarks using the same DGM backbones. Our code is available at `https://github.com/layer6ai-labs/dgm_ood_detection`.

1. Introduction

Out-of-distribution (OOD) detection (Quiñonero-Candela et al., 2008; Rabanser et al., 2019; Ginsberg et al., 2023) is crucial for ensuring the safety and reliability of machine learning models given their deep integration into real-world applications ranging from finance (Sirignano & Cont, 2019) to medical diagnostics (Esteva et al., 2017). In areas as critical as autonomous driving (Bojarski et al., 2016) and medical imaging (Litjens et al., 2017; Adnan et al., 2022), these models, while proficient with in-distribution data, may give overconfident or plainly incorrect outputs when faced with OOD samples (Wei et al., 2022).

We focus on OOD detection using likelihood-based deep generative models (DGMs), which aim to learn the density that generated the observed data. Maximum-likelihood and related objectives operate by increasing model likelihoods or appropriate surrogates on training data, and since probability densities must be normalized, one might expect lower likelihoods for OOD points. Likelihood-based DGMs such as normalizing flows (NFs) (Dinh et al., 2017; Kingma & Dhariwal, 2018; Durkan et al., 2019) and diffusion models (DMs) (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song et al., 2021b;a) have proven to be powerful DGMs that can render photo-realistic images. Given these successes, it seems reasonable to attempt OOD detection by thresholding the likelihood of a query datum under a trained model.

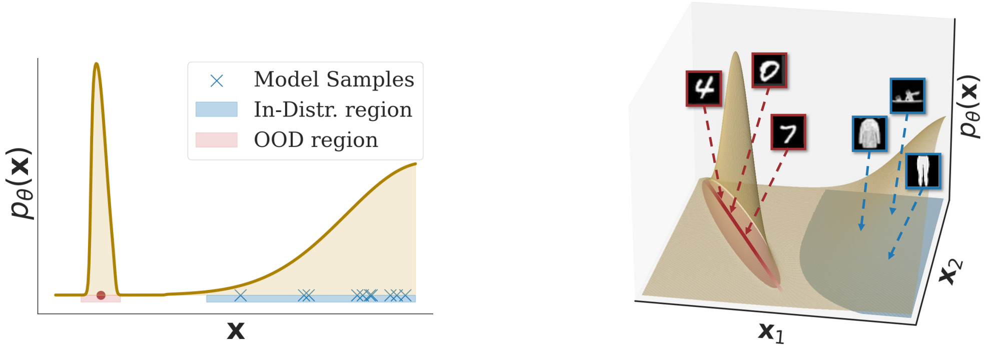
Surprisingly, likelihood-based DGMs trained on more complex datasets assign higher likelihoods to OOD datapoints taken from simpler datasets (Choi et al., 2018; Nalisnick et al., 2019a; Havtorn et al., 2021). This becomes even more puzzling in light of the facts that (i) said DGMs are trained to assign high likelihoods to in-distribution data without having been exposed to OOD data, and (ii) they only generate samples which are visually much more similar to the training data. In this work, we explore the following explanation for how both these observations can simultaneously be true (Zhang et al., 2021):

OOD datapoints can be assigned higher likelihoods while not being generated if they belong to regions of low probability mass.

Figure 1 illustrates that regions assigned high density by a model may integrate to very little probability mass. Our key insight is that when OOD data is "simpler" in the sense that it concentrates on a manifold of lower dimension than in-distribution data, the phenomenon depicted in Figure 1 becomes completely consistent with empirical observations. Based on this insight, we develop a new understanding of

[1] Layer 6 AI [2] University of Toronto [3] Vector Institute. Correspondence to: Hamidreza Kamkari, Brendan Leigh Ross, Jesse C. Cresswell, Anthony L. Caterini, Gabriel Loaiza-Ganem <{hamid, brendan, jesse, anthony, gabriel}@layer6.ai>, Rahul G. Krishnan <rahulgk@cs.toronto.edu>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).



(a)

(b)

Figure 1: **(a)** A 1D density which is highly peaked in the OOD region (red) assigns high likelihood, but low probability mass to OOD data. **(b)** An analogous sketch for a 2D density concentrated around a 1D OOD manifold (red line), illustrated with FMNIST as in-distribution and MNIST as OOD. The model density has become sharply peaked around the manifold of "simpler" data which has low intrinsic dimension, which is nonetheless assigned lower probability mass as it has negligible volume.

the eponymous paradox, leading us to the realization that estimating the intrinsic dimension of the involved manifolds provides a simple and effective way to perform OOD detection *using only a pre-trained likelihood-based DGM*.

Contributions We (i) develop an OOD detection method which classifies a datum as in-distribution only if it has high likelihood and is in a region with non-negligible probability mass – as measured by a large local intrinsic dimension (LID) estimate of the DGM's learned manifold; (ii) empirically verify our explanation for the OOD paradox for both NFs and DMs; (iii) achieve or match state-of-the-art OOD detection performance among methods using the same DGM backbone as us.

2. Background

Normalizing Flows In this study, we target DGMs that produce a density model p_θ, parameterized by θ, which can be easily evaluated. Among them, NFs readily provide probability densities through the change of variables formula, making them suitable for studying pathologies in the likelihood function. A NF is a diffeomorphic mapping $f_\theta : \mathcal{Z} \to \mathcal{X}$ from a latent space $\mathcal{Z} = \mathbb{R}^d$ to data space $\mathcal{X} = \mathbb{R}^d$, which transforms a simple distribution p_Z on $\mathcal{Z}$, typically an isotropic Gaussian, into a complicated data distribution p_θ on $\mathcal{X}$. The change of variables formula allows one to evaluate the likelihood of a datum $\mathbf{x} \in \mathcal{X}$ as

$$\log p_\theta(\mathbf{x}) = \log p_Z(\mathbf{z}) - \log |\det \boldsymbol{J}(\mathbf{z})| , \qquad (1)$$

where $\mathbf{z} = f_\theta^{-1}(\mathbf{x})$ and $\boldsymbol{J}(\mathbf{z}) = \nabla_\mathbf{z} f_\theta(\mathbf{z}) \in \mathbb{R}^{d \times d}$. NFs are constructed such that $\det \boldsymbol{J}(\mathbf{z})$ can be efficiently evaluated, and are trained through maximum-likelihood, $\max_\theta \mathbb{E}_{\mathbf{x}\sim p_0}[\log p_\theta(\mathbf{x})]$, where p_0 is the true data-generating distribution. Sampling $\mathbf{x} \sim p_\theta$ is achieved by transforming a sample $\mathbf{z} \sim p_Z$ through f_θ, i.e. $\mathbf{x}{=}f_\theta(\mathbf{z})$.

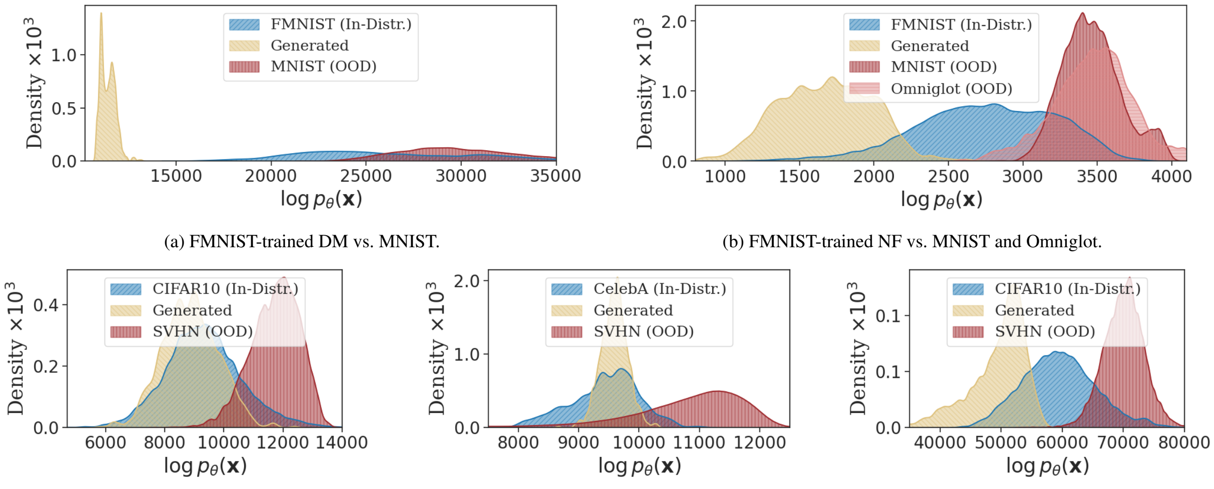
Diffusion Models DMs are popular and also admit likelihood evaluation. Various formulations of DMs exist; here we use score-based models (Song et al., 2021b). DMs first define an Itô stochastic differential equation (SDE):

$$\mathrm{d}\mathbf{x}_t = h(\mathbf{x}_t, t)\mathrm{d}t + g(t)\mathrm{d}\mathbf{w}_t, \quad \mathbf{x}_0 \sim p_0, \qquad (2)$$

where $h : \mathcal{X} \times [0, T] \to \mathcal{X}$, $g : [0, T] \to \mathbb{R}$, and $T > 0$ are hyperparameters, and where $\mathbf{w}_t$ denotes a d-dimensional Brownian motion. This SDE prescribes how to transform data $\mathbf{x}_0$ into noisy data $\mathbf{x}_t$, whose distribution we denote as p_t, the intuition being that p_T is extremely close to "pure noise". Equation 2 can be reversed in time in the sense that $\mathbf{y}_t = \mathbf{x}_{T-t}$ obeys the SDE

$$\begin{aligned} \mathrm{d}\mathbf{y}_t = &\big(g(T-t)^2 s_{T-t}(\mathbf{y}_t) - h(\mathbf{y}_t, T-t)\big)\,\mathrm{d}t \\ &+ g(T-t)\mathrm{d}\bar{\mathbf{w}}_t, \quad \mathbf{y}_0 \sim p_T, \end{aligned} \qquad (3)$$

where $s_{T-t}(\mathbf{y}_t) = \nabla_{\mathbf{y}_t} \log p_{T-t}(\mathbf{y}_t)$ denotes the (Stein) score, and $\bar{\mathbf{w}}_t$ is another Brownian motion. Solving Equation 3 would result in samples from p_0 at time T, but both the score and p_T are unknown. DMs use a neural network $s_\theta : \mathcal{X} \times [0, T] \to \mathcal{X}$ whose goal is to learn the true score. This is achieved through the denoising score matching objective (Vincent, 2011), which Song et al. (2021a) showed can be interpreted as likelihood-based for an appropriate hyperparameter choice. Sampling from a trained DM is achieved by approximately solving Equation 3: $s_\theta(\mathbf{y}_t, T{-}t)$ replaces $s_{T-t}(\mathbf{y}_t)$, and a Gaussian distribution $\hat{p}_T$ with an appropriately chosen covariance replaces p_T. This procedure implicitly defines the density p_θ of a DM. Song et al. (2021b) show that DMs can be interpreted as continuous NFs (Chen et al., 2018), transforming samples from $\hat{p}_T$ into (approximate) samples from p_0. In turn, this enables evaluating p_θ through a corresponding change of variable formula analogous to Equation 1.



(a) FMNIST-trained DM vs. MNIST.

(b) FMNIST-trained NF vs. MNIST and Omniglot.

(c) CIFAR10-trained NF vs. SVHN.

(d) CelebA-trained NF vs. SVHN.

(e) CIFAR10-trained DM vs. SVHN.

Figure 2: **(a)** A FMNIST-trained DM assigns higher likelihoods to MNIST. **(b)** A NF trained on FMNIST shows notably lower likelihoods on its own generated samples than on OOD data. **(c-e)** Analogous pathologies on RGB datasets, both for DMs and NFs.

Likelihood Pathologies in OOD Detection Choi et al. (2018) and Nalisnick et al. (2019a) first uncovered unintuitive behaviour that pervasively affects likelihood-based DGMs. For instance, NFs trained on relatively complex datasets like CIFAR10 (Krizhevsky & Hinton, 2009) and FMNIST (Xiao et al., 2017) often yield high likelihoods when tested on simpler ones like SVHN (Netzer et al., 2011) and MNIST (LeCun et al., 1998), respectively, despite the latter datasets not having been seen in the training process. While this issue is not exclusive to images (Ren et al., 2019), our experiments, shown in Figure 2, confirm these previous findings for models trained on image data. Additionally, we find that this pathological behaviour is not limited to these well-known cases, but extends to numerous dataset pairs and generated samples (see Appendix A for details).

Local Intrinsic Dimension According to the manifold hypothesis, natural data lies around low-dimensional submanifolds of $\mathcal{X} = \mathbb{R}^d$ (Bengio et al., 2013; Pope et al., 2021), where d is the *ambient* dimension of the data space. The *local intrinsic dimension* (LID) of $\mathbf{x} \in \mathcal{X}$ with respect to these data submanifolds is the dimension of the submanifold that contains $\mathbf{x}$. For example, if the ambient space is $\mathbb{R}^2$ and the data manifold is the 1D unit circle S^1, then any point $\mathbf{x} \in S^1$ will have a local intrinsic dimension of 1. Often, datasets concentrate on multiple non-overlapping submanifolds of different dimensionalities (Brown et al., 2023), in which case the LID will vary between datapoints.

Density models p_θ implicitly attempt to learn these manifolds by accumulating probability mass around them. As a consequence, even when defined on the full d-dimensional space $\mathcal{X}$, trained densities p_θ implicitly encode low-dimensional manifold structure. We refer to the manifold implied by p_θ as $\mathcal{M}_\theta$, which informally corresponds to regions of high density. When referring to LID with respect to the implied manifold $\mathcal{M}_\theta$, we will write $\text{LID}_\theta(\mathbf{x})$; it will be of interest to estimate $\text{LID}_\theta(\mathbf{x})$ for in- and out-of-distribution query points $\mathbf{x}$.

Sample-based methods to estimate intrinsic dimension exist (Fukunaga & Olsen, 1971; Levina & Bickel, 2004; Johnsson et al., 2014; Facco et al., 2017; Bac et al., 2021). Unfortunately, most of these are inadequate for our purposes, either because they estimate global (i.e. averaged or aggregated) intrinsic dimension instead of $\text{LID}_\theta(\mathbf{x})$, or because they require observed samples around $\mathbf{x}$ to produce the estimate. Since we will want $\text{LID}_\theta(\mathbf{x})$ for OOD points $\mathbf{x}$, the latter methods would require access to samples from p_θ which fall in the OOD region, which are of course unavailable. The key to circumvent this issue is to move away from sample-based estimators and instead rely directly on the given DGM.

Tempczyk et al. (2022) proposed such an estimator of LID. Unfortunately, it requires training multiple NFs, rendering it incompatible with our OOD detection goal of using a single pre-trained model. Meanwhile, Stanczuk et al. (2022) construct an estimator requiring a single *variance exploding* DM, i.e. they set h to zero in Equation 2. They argue that, given a query $\mathbf{x}$, a small enough $t_0 > 0$, and $\mathbf{x}'$ sufficiently close to $\mathbf{x}$, $s_\theta(\mathbf{x}', t_0)$ will lie on the normal space (at $\mathbf{x}$) of the manifold containing $\mathbf{x}$ – i.e. $s_\theta(\mathbf{x}', t_0)$ is orthogonal to the manifold. They propose using k independent runs of Equation 2, starting at $\mathbf{x}$ and evolving until time t_0, to obtain $\mathbf{x}^1_{t_0}, \dots, \mathbf{x}^k_{t_0}$ from which they construct the matrix $\mathbf{S}(\mathbf{x}) = [s_\theta(\mathbf{x}^1_{t_0}, t_0)| \cdots |s_\theta(\mathbf{x}^k_{t_0}, t_0)] \in \mathbb{R}^{d \times k}$. The rank of $\mathbf{S}(\mathbf{x})$ estimates the dimension of the normal space when k is large enough. In turn, LID can be estimated as

$$\text{LID}_\theta(\mathbf{x}) \approx d - \operatorname{rank} \mathbf{S}(\mathbf{x}). \tag{4}$$

3. Method

Intuitively, the fact that DGMs never generate OOD samples suggests that they contain the information needed to discern between OOD and in-distribution data – *even when they assign higher likelihoods to the former*. In this section we will show how to leverage LID to extract this information from a pre-trained model. Although conceptually our insights are agnostic to the type of model being used and thus apply to all likelihood-based DGMs, we will focus on NFs and DMs as LID and likelihoods can be readily computed for them.

3.1. LID, Volume, and Probability Mass

Before showing that $\text{LID}_\theta(\mathbf{x})$ is the key to the OOD detection paradox, we heuristically explain here how $\text{LID}_\theta(\mathbf{x})$ is related to the contiguous volume associated to the region around $\mathbf{x}$ by p_θ.

To demonstrate, consider a fitted three-dimensional model $p_\theta(\mathbf{x}) = w_{\text{in}}\mathcal{N}(\mathbf{x}; \mu_{\text{in}}, \Sigma_{\text{in}}) + w_{\text{out}}\mathcal{N}(\mathbf{x}; \mu_{\text{out}}, \Sigma_{\text{out}})$, where $\mathcal{N}(\mathbf{x}; \mu, \Sigma)$ is a Gaussian density with mean μ and covariance Σ evaluated at $\mathbf{x}$. Let $\|\mu_{\text{in}} - \mu_{\text{out}}\|_2 \gg 0$ so that the μ_{out} component is sufficiently far from μ_{in}, and let $w_{\text{out}} = \delta$ with $w_{\text{in}} = 1 - \delta$, where $1 \gg \delta > 0$ is vanishingly small so that data around μ_{out} has near-zero probability of being sampled. In these ways, μ_{in} and μ_{out} are analogous to in-distribution and pathological OOD data. Let the respective covariance matrices of the components be $\Sigma_{\text{in}} = \text{diag}(\sigma^2, \sigma^2, \varepsilon^2)$ and $\Sigma_{\text{out}} = \text{diag}(\varepsilon^2, \sigma^2, \varepsilon^2)$, where $\sigma \gg \varepsilon > 0$, with ε very close to zero. The impact of these covariance matrices is that the model distribution p_θ extends primarily in two dimensions around μ_{in} and in a single dimension around μ_{out}. Thus, numerically, $\text{LID}_\theta(\mu_{\text{in}}) = 2$ and $\text{LID}_\theta(\mu_{\text{out}}) = 1$.

We first note that $p_\theta(\mu_{\text{out}}) > p_\theta(\mu_{\text{in}})$ whenever $\varepsilon < \sigma\delta$, meaning it is possible here for p_θ to be pathologically high on OOD data. Since, informally, "probability mass = density × volume", a reasonable way to measure the volume assigned by p_θ around each of these modes is the ratio of probability mass to density. For $i \in \{\text{in}, \text{out}\}$, mode i has a respective mass of approximately w_i, so we can write

$$\text{vol}_\theta(\mu_i) \approx \frac{w_i}{p_\theta(\mu_i)} \approx \frac{1}{\mathcal{N}(\mu_i; \mu_i, \Sigma_i)} \propto \varepsilon^3 \left(\frac{\sigma}{\varepsilon}\right)^{\text{LID}_\theta(\mu_i)}, \quad (5)$$

where, since $\sigma > \varepsilon$, volume monotonically increases with LID. Though the models we use throughout this work are much more complex, we carry forward the same intuition that the volume $\text{vol}_\theta(\mathbf{x})$ assigned to the vicinity of $\mathbf{x}$ by p_θ increases with $\text{LID}_\theta(\mathbf{x})$. A small $\text{LID}_\theta(\mathbf{x})$ makes it possible in practice for $p_\theta(\mathbf{x})$ to be high, despite p_θ assigning negligible probability mass around $\mathbf{x}$.

LID and probability mass We now make the connection between LID and probability mass more concrete. For $R > 0$, we denote the d-dimensional Euclidean ball of radius R centred at $\mathbf{x}$ as $B_R(\mathbf{x})$. In Appendix B we argue that, for sufficiently negative scalars r,

$$\frac{\partial}{\partial r} \log \int_{B_{e^r\sqrt{d}}(\mathbf{x})} p_\theta(\mathbf{x}')\text{d}\mathbf{x}' \approx \text{LID}_\theta(\mathbf{x}) + C, \quad (6)$$

where C is a constant that depends neither on θ nor on $\mathbf{x}$. The integral on the left hand side corresponds to the probability assigned by p_θ to a small ball around $\mathbf{x}$. Thus, a large $\text{LID}_\theta(\mathbf{x})$ is equivalent to a rapid growth of the log probability mass that p_θ assigns to a neighbourhood of $\mathbf{x}$ as the size of the neighbourhood increases. In turn, we should expect the probability mass assigned around $\mathbf{x}$ to be large if and only if both $p_\theta(\mathbf{x})$ and the aforementioned rate of change are large as well, meaning that LID can indeed be informally understood as monotonically related to the probability mass. This view of LID provides the same intuition as the more informal "volume"-based interpretation, namely that probability mass being large is equivalent to both density and LID being large.

3.2. Detecting OOD Data with LID

We now discuss the situation illustrated in Figure 1. We begin by highlighting that there are three manifolds (or rather unions thereof) at play – $\mathcal{M}_{\text{in}}$, $\mathcal{M}_{\text{out}}$, and $\mathcal{M}_\theta$ – around which p_0, OOD data, and p_θ concentrate, respectively. We take the viewpoint that $\mathcal{M}_{\text{in}}$ and $\mathcal{M}_{\text{out}}$ do not overlap (i.e., $\mathcal{M}_{\text{in}} \cap \mathcal{M}_{\text{out}} = \emptyset$), otherwise OOD detection would be ill-posed (Le Lan & Dinh, 2021). The paradoxical nature of likelihood-based OOD detection can be summarized as follows: when $\mathbf{x} \in \mathcal{M}_{\text{out}}$ we should expect the ground truth $p_0(\mathbf{x}) \approx 0$ because $\mathcal{M}_{\text{in}} \cap \mathcal{M}_{\text{out}} = \emptyset$, and since p_θ was trained to approximate p_0, we should also expect $p_\theta(\mathbf{x}) \approx 0$. However, it is often observed that p_θ is *larger* on $\mathcal{M}_{\text{out}}$ than on $\mathcal{M}_{\text{in}}$; i.e., $\mathcal{M}_{\text{out}} \subset \mathcal{M}_\theta$. We now explain how this behaviour can occur by leveraging the notion of volume assigned by p_θ.

By the manifold hypothesis, $\mathcal{M}_{\text{out}}$ and $\mathcal{M}_{\text{in}}$ are low-dimensional, and they thus have zero (Lebesgue) volume in ambient space. However, here we are concerned with a different notion of "volume": the contiguous "volume" associated to a region around $\mathbf{x}$ by the full-dimensional density p_θ. We informally define this "volume" as a ratio of probability mass to density, as in Equation 5. In the OOD paradox, $p_\theta(\mathbf{x})$ is large for $\mathbf{x} \in \mathcal{M}_{\text{out}}$, yet samples are never drawn from $\mathcal{M}_{\text{out}}$, suggesting negligible probability mass has been assigned around the region. As a consequence, p_θ must have assigned a very small "volume" to the region around $\mathbf{x}$. This is made mathematically possible by the fact that $\mathcal{M}_{\text{out}}$ has a (Lebesgue) volume of zero, and thus p_θ can assign arbitrarily small "volume" to the region around $\mathcal{M}_{\text{out}}$, even when high densities are present. From this logic, we see that *the paradox is fully characterized by p_θ assigning high density to, but low "volume" around, the point $\mathbf{x} \in \mathcal{M}_{out}$.*

A priori, $\mathcal{M}_{\text{out}}$ being contained in the region over which p_θ happens to behave pathologically (i.e., $\mathcal{M}_{\text{out}} \subset \mathcal{M}_\theta$) might seem like an unbelievable coincidence. However, in the case of NFs, past work by Kirichenko et al. (2020) and Schirrmeister et al. (2020) has shown that the multi-scale convolutional architecture used by these models fixates on high-frequency local features and pixel-to-pixel correlations. Thus, when these features and correlations are present in OOD data, the corresponding likelihoods are inadvertently encouraged to become large through the model's implicit bias; we hypothesize other DGMs behave similarly (see Appendix A for an extended discussion). Our work is thus complementary to that of Kirichenko et al. (2020) and Schirrmeister et al. (2020): even when p_θ is a good model for the true data-generating distribution, we show that $p_\theta \approx p_0$ can be violated around a set $\mathcal{M}_\theta \setminus \mathcal{M}_{\text{in}}$ of small "volume", whereas they provide an explanation of why this set sometimes contains $\mathcal{M}_{\text{out}}$.

The connection between LID and "volume" also explains the directionality of the paradox; i.e., why it only arises when OOD data is simpler (in that it has lower intrinsic dimension) than in-distribution data. In the non-pathological case, when $\mathcal{M}_{\text{out}}$ is more complex (i.e., higher-dimensional) than $\mathcal{M}_{\text{in}}$, assigning large densities to $\mathcal{M}_{\text{out}}$ would necessarily correspond to a higher "volume" and hence high probability mass. High probability mass would imply that p_θ generates samples $\mathcal{M}_{\text{out}}$, which of course never occurs in practice. However, when $\mathcal{M}_{\text{out}}$ is lower-dimensional than $\mathcal{M}_{\text{in}}$, the model p_θ is able to assign lower $\text{LID}_\theta(\mathbf{x})$, and thus lower "volume", to $\mathcal{M}_{\text{out}}$. This allows it to simultaneously assign pathologically large densities and vanishingly small probability mass to $\mathcal{M}_{\text{out}}$. Only in this second case can p_θ closely approximate p_0 while also assigning high densities to $\mathcal{M}_{\text{out}}$.

It follows that if $\text{LID}_\theta(\mathbf{x})$ has a small value relative to in-distribution data, we can expect the probability mass that p_θ assigns around $\mathbf{x}$ to be negligible – even if $p_\theta(\mathbf{x})$ is large – suggesting that $\mathbf{x}$ should be classified as OOD.

3.3. Estimating LID

We have now justified the use of $\text{LID}_\theta(\mathbf{x})$ for OOD detection, yet this quantity cannot be evaluated, only estimated, which we now show how to do.

LID for NFs Consider a smooth map $f : \mathcal{Z} \to \mathcal{X}$ between two manifolds and a point $\mathbf{z} \in \mathcal{Z}$. If f has constant rank in an open neighbourhood around $\mathbf{z}$, then the intrinsic dimension of $\mathbf{x} = f(\mathbf{z})$ on its image is given by the rank of the differential of f at $\mathbf{z}$. When f is a NF, $\text{LID}_\theta(\mathbf{x})$ is thus formally given by $\operatorname{rank} \mathbf{J}(\mathbf{z})$ (Horvat & Pfister, 2022). Technically, NFs have full rank Jacobians by construction, making $\text{LID}_\theta(\mathbf{x}) = d$ for all $\mathbf{x}$. However, since NFs concentrate mass around the low-dimensional $\mathcal{M}_\theta$ they are not numerically invertible (Cornish et al., 2020; Behrmann et al., 2021), and so numerically they assign LIDs of less than d to most points. For a given NF and a query $\mathbf{x}$, we thus estimate the corresponding (numerical) LID as

$$\widehat{\text{LID}}_\theta^{\text{NF}}(\mathbf{x}) := \left|\left\{i \in [d] : \sigma_i^{\text{NF}}(\mathbf{x}) > \tau\right\}\right|, \qquad (7)$$

where $[d] = \{1, \dots, d\}$, $\sigma_i^{\text{NF}}(\mathbf{x})$ is the i-th singular value of $\mathbf{J}(\mathbf{z})$, and $\tau > 0$ is a threshold hyperparameter specifying which singular values are numerically equal to zero.

LID for DMs As previously mentioned, Stanczuk et al. (2022) developed an LID estimator for variance exploding DMs (Equation 4) which is based on $s_\theta(\mathbf{x}', t_0)$ orthogonally pointing towards $\mathcal{M}_\theta$. We found better performance with *variance preserving* DMs (Appendix D.3), where $h(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}$ and $g(t) = \sqrt{\beta(t)}$ for an affine function $\beta : [0, T] \to \mathbb{R}_{>0}$. In this case the direction of the drift in Equation 3 is not given by $s_{T-t}(\mathbf{y}_t)$ anymore, but by $s_{T-t}(\mathbf{y}_t) + \frac{1}{2}\mathbf{y}_t$ instead. Accordingly, we modify $\mathbf{S}(\mathbf{x}) \in \mathbb{R}^{d \times k}$, where $k > d$, as

$$\mathbf{S}(\mathbf{x}) = \left[s_\theta(\mathbf{x}_{t_0}^1, t_0) + \frac{\mathbf{x}_{t_0}^1}{2} \middle| \cdots \middle| s_\theta(\mathbf{x}_{t_0}^k, t_0) + \frac{\mathbf{x}_{t_0}^k}{2} \right], \qquad (8)$$

whose columns we now expect to point orthogonally towards $\mathcal{M}_\theta$.[1] Similarly to NFs, $\operatorname{rank} \mathbf{S}(\mathbf{x})$ can technically match d even though many of its singular values are almost zero. We thus estimate the (numerical) LID of DMs as

$$\widehat{\text{LID}}_\theta^{\text{DM}}(\mathbf{x}) := d - \left|\left\{i \in [d] : \sigma_i^{\text{DM}}(\mathbf{x}) > \tau\right\}\right|, \qquad (9)$$

where $\sigma_i^{\text{DM}}(\mathbf{x})$ is the i-th singular value of $\mathbf{S}(\mathbf{x})$.

Setting the threshold Both LID estimators presented above require setting τ to threshold singular values. We found that in practice, no single value of τ performed well across all datasets. To avoid having to manually tune this hyperparameter, we propose a way to set τ only using the available (in-distribution) training data. Specifically, we leverage local principal component analysis (LPCA) (Fukunaga & Olsen, 1971), which is a fast and simple LID estimator. Roughly, for a given in-distribution $\mathbf{x}$ and a provided dataset, LPCA uses the nearest neighbours of $\mathbf{x}$ in the dataset to construct a matrix whose rank approximates the LID at $\mathbf{x}$, and we calibrate τ to match the LPCA estimates. We reiterate that estimators based on nearest neighbours such as LPCA are not directly useful for identifying OOD data, since they require local samples around a query which

[1] While intuitive, adding $\mathbf{x}_{t_0}^j$ to the j-th column of $\mathbf{S}(\mathbf{x})$ is an ad-hoc modification to the estimator proposed by Stanczuk et al. (2022) to account for our use of variance preserving DMs. In practice this modification does not drastically affect the corresponding LID estimate, as numerically it is similar to adding the same constant vector to every column.

Algorithm 1 Dual threshold OOD detection, returns `True` if $\mathbf{x}$ is deemed OOD, and `False` if deemed in-distribution.

```
Require: $\log p_\theta(\mathbf{x}), \widehat{\text{LID}}_\theta(\mathbf{x}), \psi_\mathcal{L}, \psi_\text{LID}$
if $\log p_\theta(\mathbf{x}) < \psi_\mathcal{L}$ then
    return True                                  ▷ case $(i)$
if $\widehat{\text{LID}}_\theta(\mathbf{x}) < \psi_\text{LID}$ then
    return True                                  ▷ case $(ii)$
return False                                     ▷ case $(iii)$
```

are unavailable for OOD data and cannot be generated by p_θ. For a detailed evaluation of the proposed LID estimators, see Appendix C.

3.4. Putting it All Together

So far we have argued that LID can be used for OOD detection and have shown how to obtain estimates $\widehat{\text{LID}}_\theta(\mathbf{x})$ by using Equation 7 or Equation 9. In summary, three mutually exclusive cases can happen for a point $\mathbf{x}$: (i) $\log p_\theta(\mathbf{x})$ is small (relative to in-distribution data). (ii) $\log p_\theta(\mathbf{x})$ is large and $\widehat{\text{LID}}_\theta(\mathbf{x})$ is small. In both of these cases p_θ assigns negligible probability mass around $\mathbf{x}$, which in turn means $\mathbf{x}$ should be classified as OOD. (iii) $\log p_\theta(\mathbf{x})$ and $\widehat{\text{LID}}_\theta(\mathbf{x})$ are both large, in which case the likelihood spiking on $\mathbf{x}$ is not pathological, implying that $\mathbf{x}$ should be classified as in-distribution. This leads to our proposed dual threshold OOD detection method, described in Algorithm 1, where $\psi_\mathcal{L}$ and ψ_LID are the log-likelihood and LID thresholds, respectively. We highlight that our method differs from standard (single threshold) likelihood-based OOD detection only in that we classify case (ii) as OOD instead of in-distribution.

4. Related Work

A substantial amount of research into likelihood pathologies tries to explain the underlying causes of the OOD paradox. One particular line of research proposes probabilistic explanations: Choi et al. (2018) and Nalisnick et al. (2019b) put forth the "typical set" hypothesis, which has been contested in follow-up work. For example, Le Lan & Dinh (2021) argue that likelihood rankings not being invariant to data reparameterizations causes the paradox, whereas Caterini & Loaiza-Ganem (2021) claim it is the lower entropy of "simpler" distributions as compared to the higher entropy of more "complex" ones – which somewhat aligns with our work, although we use intrinsic dimension instead of entropy to quantify complexity.

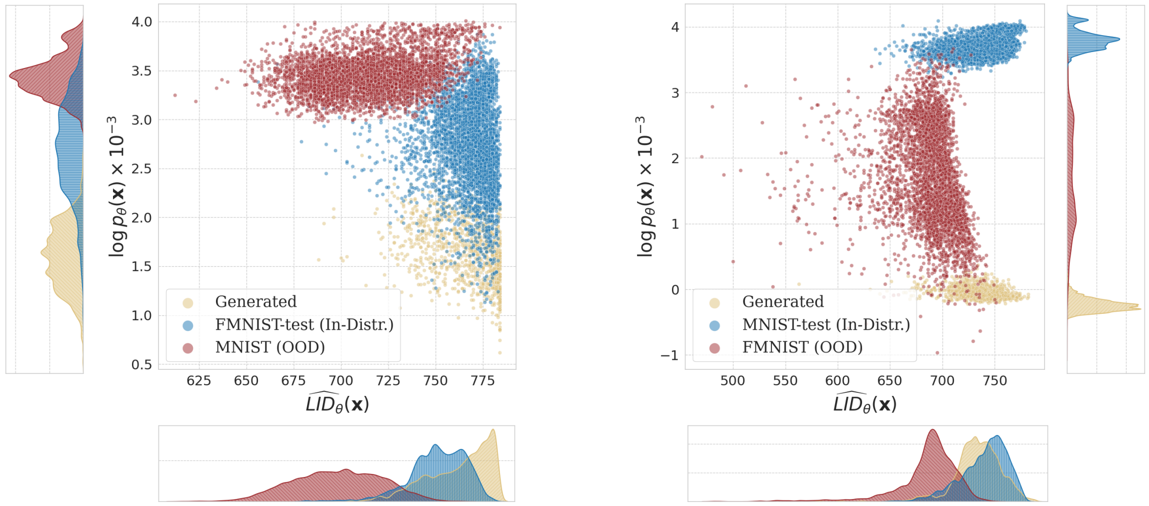
We diverge from these explanations in that they all assume, sometimes implicitly, that $\mathcal{M}_\text{in}$ and $\mathcal{M}_\text{out}$ overlap. We find it extremely plausible, for example, that the intersection between CIFAR10 and SVHN images is empty. In this sense, we are more in agreement with Zhang et al. (2021), who propose a similar explanation to ours based on probability mass. Nonetheless, we differ from the work of Zhang et al. (2021) in several key ways: (i) they blame poor model fit as the culprit, which is inconsistent with our results showing that likelihoods do not distinguish between OOD and generated samples; (ii) they do not establish a connection to LID; and (iii) they do not empirically verify their explanation since they do not propose a method to address the issue.

Another line of work aims to build DGMs which do not experience the OOD paradox (Li et al., 2022), sometimes at the cost of generation quality. For example, Grathwohl et al. (2020) and Liu et al. (2020) argue that the training procedure of energy-based models (EBMs) (Xie et al., 2016; Du & Mordatch, 2019) provides inductive biases which help OOD detection, and Yoon et al. (2021) construct an EBM specifically designed for this task. Kirichenko et al. (2020) and Loaiza-Ganem et al. (2022) first embed data into semantically rich latent spaces, and then employ dense neural network architectures, thus minimizing susceptibility to local high-frequency features. We differ from these works in that they attempt to build DGMs that are better at likelihood-based OOD detection, whereas we only leverage a pre-trained model.

Other works use "outside help" or auxiliary models. Some methods assume access to an OOD dataset (Nalisnick et al., 2019b), require class labels (Görnitz et al., 2013; Ruff et al., 2020; van Amersfoort et al., 2021; Liu et al., 2022), or leverage an image compression algorithm (Serrà et al., 2020). Some other works, while fully unsupervised, require training auxiliary models on distorted data (Ren et al., 2019), on the incoming test datapoint with regularization (Xiao et al., 2020), or on data summary statistics (Morningstar et al., 2021). A final line of work leverages DMs for OOD detection. Graham et al. (2023), Liu et al. (2023), and Choi et al. (2023) propose methods based on reconstruction errors. Goodier & Campbell (2023), who use the variational formulation of DMs (Sohl-Dickstein et al., 2015; Ho et al., 2020) rather than the score-based one, adopt a likelihood ratio approach which averages the DM loss across various noise levels. Again, we are different from these works in that we do not just care about empirical performance, but also about understanding why likelihoods alone fail – and leveraging this understanding for fully unsupervised OOD detection.

5. Experiments

Setup We compare datasets within two classes: (i) 28×28 greyscale images, including FMNIST, MNIST, Omniglot (Lake et al., 2015), and EMNIST (Cohen et al., 2017); and (ii) RGB images resized to $32 \times 32 \times 3$, comprising SVHN, CIFAR10 and CIFAR100 (Krizhevsky & Hinton, 2009),



(a)

(b)

Figure 3: LID estimates and likelihood scatterplots, along with corresponding marginals. **(a)** FMNIST-trained model, evaluated on FMNIST, MNIST, and generated samples. **(b)** MNIST-trained model, evaluated on FMNIST, MNIST, and generated samples.

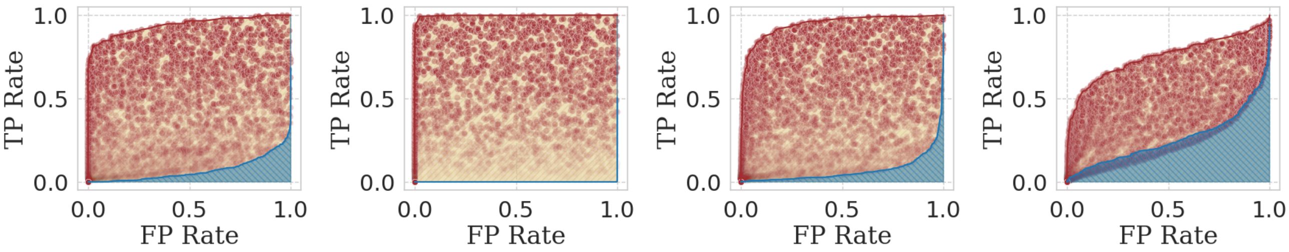
Tiny ImageNet (Le & Yang, 2015), and a simplified, cropped version of CelebA (Kist, 2021). We give experimental details on model training in Appendix D.1 and Appendix D.2. Due to space constraints, we report results for all dataset pairs in Appendix D.4.

Evaluation OOD detection methods use the area under the curve (AUC) of the receiver operator characteristic (ROC) curve for evaluation. The true positive rate (TPR) and false positive rate (FPR) from an OOD classifier correspond to points on the FPR-TPR plane. By sweeping over all possible threshold values, these points determine the ROC graph. For single threshold OOD classifiers, the graph provides points on a curve indicating the best achievable TPR for each FPR, the area under which is denoted as AUC-ROC. On the other hand, the ROC graph for dual threshold classifiers corresponds to points on a surface – not a curve – on the FPR-TPR plane. The upper boundary of this surface defines a curve, which also indicates the best achievable TPR for each FPR. Thus, in a slight abuse of terminology, we also denote the area under this curve as AUC-ROC, as it is directly comparable to that of single threshold classifiers. See Appendix D.5 for a thorough explanation of such ROC curves and how we compute their AUC.

LID with Likelihoods Isolates OOD Regions We compute log-likelihoods and LID estimates for NFs trained on FMNIST and MNIST, with results shown in Figure 3. The scatterplots with both likelihoods and LIDs show clear separation between in-distribution and OOD, despite the likelihood and LID marginals overlapping. Furthermore, the "directions" predicted by our method are correct: in the pathological case (FMNIST-trained), we see that while OOD points have higher likelihoods, they also have lower LIDs; whereas in the non-pathological case (MNIST-trained), likelihoods are lower for OOD data. These results highlight not only the importance of using LID estimates for OOD detection, but also that of combining them with likelihoods, as the two together provide a proxy for probability mass.

Visualizing the Benefits of Dual Thresholding The separation of OOD and in-distribution data shown in the scatterplots in Figure 3 confirms that likelihood/LID pairs contain the needed information for OOD detection. However, it remains to show that Algorithm 1 succeeds at this task (recall that we cannot simply train a classifier to differentiate between red and blue points in Figure 3 since the red OOD points are unavailable when designing the OOD detector). Figure 4 provides a visual comparison showcasing the ROC curves from our dual thresholding technique versus the ROC curves constructed by single threshold classifiers using only likelihoods. These results show a dramatic boost in AUC-ROC performance across four different pathological scenarios, highlighting the relevance of combining likelihoods with LIDs for OOD detection. For further experiments, please refer to the ablations in Appendix D.6.

Quantitative Comparisons In the top part of Table 1, we compare our method against several normalizing flow (NF) baselines, all of which are evaluated using the exact same pre-trained NF as our method. These baselines are: (i) naïvely labelling large log-likelihoods $\log p_\theta(\mathbf{x})$ as in-distribution, which strongly fails at identifying "simpler" distributions as OOD when trained on "complex" datasets; (ii) using $\|\frac{\partial}{\partial \mathbf{x}} \log p_\theta(\mathbf{x})\|_2$ as a proxy for local probability mass as proposed by Grathwohl et al. (2020), which



(a) **FMNIST vs. MNIST**: AUC-ROC boost ($0.070 \to 0.953$)

(b) **FMNIST-gen vs. Omniglot**: AUC-ROC boost ($0.000 \to 0.996$)

(c) **CIFAR10 vs. SVHN**: AUC-ROC boost ($0.060 \to 0.926$)

(d) **CIFAR10-gen vs. CelebA**: AUC-ROC boost ($0.258 \to 0.733$)

Figure 4: ROC visualizations for select pathological OOD tasks on NFs. The red dots correspond to the FPR-TPR pairs of our method obtained from different dual thresholds, the yellow areas correspond to the region under the associated Pareto frontier (i.e. the upper boundary of the red dots), while the blue areas represent the region below the ROC curve for single threshold likelihood-based classifiers. **(a)** FMNIST-trained model with MNIST as OOD; **(b)** as in (a) except we now discern between generated samples and MNIST. **(c)** CIFAR10-trained model with SVHN as OOD; **(d)** as in (c) except we now discern between generated samples and CelebA.

Table 1: AUC-ROC (higher is better). The top part of the table contains NF-based approaches, the middle rows contain DM-based approaches, and the last row shows an EBM-based one. **Notation**: * indicates tasks where likelihoods alone do not exhibit pathological behaviour; ‡ indicates methods that employ external information or auxiliary models. For the NF and DM methods, we bold the best performing approach among themselves, and the EBM model is bolded when it surpasses all others.

Trained on	MNIST *		FMNIST		CIFAR10		SVHN *	
OOD Dataset	FMNIST	Omniglot	MNIST	Omniglot	SVHN	CelebA	CIFAR10	CelebA
NF Likelihood	**1.000**	0.796	0.073	0.085	0.063	0.391	**0.987**	**0.996**
NF $\|\frac{\partial}{\partial \mathbf{x}} \log p_\theta(\mathbf{x})\|_2$	0.156	0.444	0.516	0.538	0.722	0.433	0.200	0.080
Complexity Correction‡	0.945	0.852	0.939	**0.935**	0.835	0.479	0.771	0.639
NF Likelihood Ratios‡	0.944	0.722	0.666	0.639	0.299	0.396	0.302	0.099
NF Dual Threshold (Ours)	**1.000**	**0.855**	**0.951**	0.864	**0.936**	**0.655**	**0.987**	**0.996**
DM Likelihood	0.996	**1.000**	0.240	0.952	0.064	0.360	**0.996**	**0.996**
DM $\|s_\theta(\mathbf{x}, 0)\|_2$	0.919	0.004	0.075	0.001	0.883	**0.716**	0.120	0.145
DM Reconstruction‡	**1.000**	0.999	**0.970**	**0.992**	0.876	0.630	0.984	0.995
DM Likelihood Ratios	0.224	0.296	0.781	0.388	0.829	0.553	0.326	0.357
DM Dual Threshold (Ours)	0.996	**1.000**	0.912	0.959	**0.944**	0.648	**0.996**	**0.996**
NAE	**1.000**	0.994	**0.995**	0.976	0.919	**0.887**	0.948	0.965

performs inconsistently across tasks; (iii) the complexity correction method of Serrà et al. (2020), which uses image compression information to adjust the inflated likelihood observed in OOD datapoints – despite this comparison being unfair in that the baseline accessed image compression algorithms in addition to the NF, we beat it across all tasks except one; (iv) the likelihood ratios approach of Ren et al. (2019), which is once again unfair as it employs an auxiliary likelihood-based reference model to compute ratios, yet we uniformly beat it across tasks.

Besides the strong empirical performance of our method with NFs, other aspects of the top part of Table 1 warrant attention. Both the complexity correction and likelihood ratio baselines lose performance over naïvely using likelihoods on non-pathological tasks, i.e. when models are trained on relatively "simple" data like MNIST or SVHN. Since likelihoods perform well at these tasks, they are often considered "easy" and thus omitted from comparisons. The fact that these baselines struggle at these tasks is a novel finding that suggests these methods "overfit" to the pathological tasks. See Appendix D.7 for a thorough discussion.

Additionally, we test our dual threshold method with diffusion models (DMs) in the middle rows of Table 1, and compare its performance against: (i) using only likelihoods, which fails at most pathological tasks; (ii) using the norm of the likelihood derivative, or equivalently, the norm of the score function (Grathwohl et al., 2020) – this also fails in pathological tasks; (iii) the reconstruction-error-based approach of Graham et al. (2023), which provides the strongest baseline and is very similar to Choi et al. (2023) – we find it noteworthy that we perform on par with this baseline, beating it at five out of eight tasks, despite the fact that it is not fully unsupervised as it relies on a pre-trained LPIPS encoder (Zhang et al., 2018); (iv) the likelihood ratios method of Goodier & Campbell (2023), which does not perform well.[2] We used the exact same DM for every comparison, and find that our method outperforms all fully unsupervised baselines that do not leverage outside data.

Overall, we believe it is remarkable that our dual threshold outperforms every baseline for NFs and performs on par with the strongest DM baseline, both pathological and non-pathological tasks, despite some baselines having access to additional information. We see these results as strong evidence supporting the understanding that we derived about the OOD paradox and its connection to LID. We also highlight that, as mentioned in Section 2, we identified cases of likelihoods behaving pathologically on generated samples. In Appendix D.4, we show that our dual threshold method also excels at detecting these scenarios.

The last row of Table 1 shows normalized autoencoders (NAEs) (Yoon et al., 2021). NAEs are EBMs specially tailored for OOD detection at the cost of generation quality, but to the best of our knowledge achieve state-of-the-art performance on fully unsupervised, likelihood-based OOD detection. Once again, we believe that the empirical results of our dual threshold method are remarkable: we achieve similar performance to NAEs on most tasks, even outperforming them on four, despite using a general purpose model p_θ, not one explicitly designed for OOD detection.

6. Conclusions, Limitations, and Future Work

In this paper we studied the OOD detection paradox, where likelihood-based DGMs assign high likelihoods to OOD points from "simpler" datasets, but do not generate them. We proposed a geometric explanation of how the paradox can arise as a consequence of models assigning low probability mass around these OOD points when they have small intrinsic dimensions. We then leveraged LID estimators for our dual threshold OOD detection method. Having decidedly outperformed the use of likelihoods by themselves, our results strongly support our geometric explanation. We believe that extending the utility of this geometric viewpoint and of LID beyond OOD detection is an extremely interesting path for follow-up work.

We highlight that the LID estimator of Stanczuk et al. (2022), which we heavily relied upon for DMs, obtains very different estimates on image datasets than previously established LID estimators. Our dual threshold technique works despite this discrepancy likely because only LID rankings are relevant for OOD detection. Further improving DM-based LID estimators is a promising avenue to boost performance.

Finally, while our ideas are widely applicable to any density model, the current incarnation of our method is limited in that it only applies to NFs and DMs, as estimating LID is more tractable for these models. Extending our method to DGMs whose LID might be estimated as the rank of an appropriate matrix (like the Jacobian of a decoder), such as variational autoencoders (Kingma & Welling, 2014; Rezende et al., 2014; Vahdat & Kautz, 2020), injective NFs (Brehmer & Cranmer, 2020; Caterini et al., 2021; Ross & Cresswell, 2021), or DMs on latent space (Vahdat et al., 2021; Rombach et al., 2022) is also likely to work. Nonetheless, we see extending our method to EBMs, which achieve state-of-the-art likelihood-based OOD detection, to be a particularly promising direction for future research.

Impact Statement

The goal of this work is to advance the understanding and methodology in the detection of out-of-distribution data using a pre-trained deep generative model. Our contribution is primarily of a theoretical nature and therefore has limited broader societal impacts on its own. However, the generality of our work lends itself to use with several existing classes of models. Therefore, we emphasize that further evaluation on diverse datasets and data modalities is required before our method is deployed as the sole tool to detect pathologies in data.

Acknowledgements

We thank Valentin Villecroze and George Stein for useful feedback on a previous version of this manuscript. RGK acknowledges support from a Tier II Canada Research Chair and a Canada CIFAR AI Chair.

References

Adnan, M., Kalra, S., Cresswell, J. C., Taylor, G. W., and Tizhoosh, H. R. Federated learning and differential privacy for medical image analysis. *Scientific Reports*, 12 (1):1953, 2022.

Bac, J., Mirkes, E. M., Gorban, A. N., Tyukin, I., and Zinovyev, A. Scikit-Dimension: a python package for intrinsic dimension estimation. *Entropy*, 23(10):1368, 2021.

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

[2] Note that Graham et al. (2023) applied their method on latent diffusion models (Rombach et al., 2022), which, as discussed in Section 4, makes OOD detection easier. We also point out that Goodier & Campbell (2023) used the variational formulation of DMs (Ho et al., 2020), and we adapted their method to score-based models. These discrepancies explain the differences between the numbers in Table 1 and those reported in these papers.

Bengio, Y., Courville, A., and Vincent, P. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35 (8):1798–1828, 2013.

Blum, A., Hopcroft, J., and Kannan, R. *Foundations of Data Science*. Cambridge University Press, 2020.

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., Zhang, X., Zhao, J., and Zieba, K. End to end learning for self-driving cars. *arXiv:1604.07316*, 2016.

Brehmer, J. and Cranmer, K. Flows for simultaneous manifold learning and density estimation. In *Advances in Neural Information Processing Systems*, 2020.

Brown, B. C., Caterini, A. L., Ross, B. L., Cresswell, J. C., and Loaiza-Ganem, G. Verifying the union of manifolds hypothesis for image data. In *International Conference on Learning Representations*, 2023.

Caterini, A. L. and Loaiza-Ganem, G. Entropic issues in likelihood-based OOD detection. In *I (Still) Can't Believe It's Not Better! Workshop at NeurIPS 2021*, pp. 21–26, 2021.

Caterini, A. L., Loaiza-Ganem, G., Pleiss, G., and Cunningham, J. P. Rectangular flows for manifold learning. In *Advances in Neural Information Processing Systems*, 2021.

Chen, R. T., Rubanova, Y., Bettencourt, J., and Duvenaud, D. K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, 2018.

Choi, H., Jang, E., and Alemi, A. A. WAIC, but Why? Generative ensembles for robust anomaly detection. *arXiv:1810.01392*, 2018.

Choi, S., Lee, H., Lee, H., and Lee, M. Projection regret: Reducing background bias for novelty detection via diffusion models. In *Advances in Neural Information Processing Systems*, 2023.

Cohen, G., Afshar, S., Tapson, J., and Van Schaik, A. EMNIST: Extending MNIST to handwritten letters. In *International Joint Conference on Neural Networks (IJCNN)*, pp. 2921–2926, 2017.

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

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

Du, Y. and Mordatch, I. Implicit generation and modeling with energy based models. In *Advances in Neural Information Processing Systems*, 2019.

Durkan, C., Bekasov, A., Murray, I., and Papamakarios, G. Neural spline flows. In *Advances in neural information processing systems*, 2019.

Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

Facco, E., d'Errico, M., Rodriguez, A., and Laio, A. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. *Scientific Reports*, 7(1): 12140, 2017.

Fukunaga, K. and Olsen, D. R. An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers*, 100(2):176–183, 1971.

Ginsberg, T., Liang, Z., and Krishnan, R. G. A learning based hypothesis test for harmful covariate shift. In *International Conference on Learning Representations*, 2023.

Goodier, J. and Campbell, N. D. Likelihood-based out-of-distribution detection with denoising diffusion probabilistic models. In *British Machine Vision Conference (BMVC)*, 2023.

Görnitz, N., Kloft, M., Rieck, K., and Brefeld, U. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013.

Graham, M. S., Pinaya, W. H., Tudosiu, P.-D., Nachev, P., Ourselin, S., and Cardoso, J. Denoising diffusion models for out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2947–2956, 2023.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. Your classifier is secretly an energy based model and you should treat it like one. In *International Conference on Learning Representations*, 2020.

Havtorn, J. D., Frellsen, J., Hauberg, S., and Maaløe, L. Hierarchical VAEs know what they don't know. In *International Conference on Machine Learning*, pp. 4117–4128, 2021.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Horvat, C. and Pfister, J.-P. Intrinsic dimensionality estimation using normalizing flows. In *Advances in Neural Information Processing Systems*, 2022.

Hutchinson, M. F. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 18(3):1059–1076, 1989.

Johnsson, K., Soneson, C., and Fontes, M. Low bias local intrinsic dimension estimation from expected simplex skewness. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(1):196–202, 2014.

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

Kingma, D. P. and Welling, M. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

Kirichenko, P., Izmailov, P., and Wilson, A. G. Why normalizing flows fail to detect out-of-distribution data. In *Advances in Neural Information Processing Systems*, 2020.

Kist, A. M. CelebA Dataset cropped with Haar-Cascade face detector, 2021. URL https://doi.org/10.5281/zenodo.5561092.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. *Technical Report*, 2009.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Le, Y. and Yang, X. Tiny ImageNet visual recognition challenge. *Technical Report*, 2015.

Le Lan, C. and Dinh, L. Perfect density models cannot guarantee anomaly detection. *Entropy*, 23(12):1690, 2021.

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

Levina, E. and Bickel, P. Maximum likelihood estimation of intrinsic dimension. In *Advances in Neural Information Processing Systems*, 2004.

Li, Y., Wang, C., Xia, X., Liu, T., Xin, M., and An, B. Out-of-distribution detection with an adaptive likelihood ratio on informative hierarchical VAE. In *Advances in Neural Information Processing Systems*, 2022.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., Van Der Laak, J. A., Van Ginneken, B., and Sánchez, C. I. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42: 60–88, 2017.

Liu, J. Z., Padhy, S., Ren, J., Lin, Z., Wen, Y., Jerfel, G., Nado, Z., Snoek, J., Tran, D., and Lakshminarayanan, B. A simple approach to improve single-model deep uncertainty via distance-awareness. *Journal of Machine Learning Research*, 23(42):1–63, 2022.

Liu, R. and Zhu, Y. On the consistent estimation of optimal Receiver Operating Characteristic (ROC) curve. In *Advances in Neural Information Processing Systems*, 2022.

Liu, W., Wang, X., Owens, J., and Li, Y. Energy-based out-of-distribution detection. In *Advances in Neural Information Processing Systems*, volume 33, pp. 21464–21475, 2020.

Liu, Z., Zhou, J. P., Wang, Y., and Weinberger, K. Q. Unsupervised out-of-distribution detection with diffusion inpainting. In *International Conference on Machine Learning*, pp. 22528–22538, 2023.

Loaiza-Ganem, G., Ross, B. L., Cresswell, J. C., and Caterini, A. L. Diagnosing and fixing manifold overfitting in deep generative models. *Transactions on Machine Learning Research*, 2022.

Lu, Y., Wang, Z., and Bal, G. Mathematical analysis of singularities in the diffusion model under the submanifold assumption. *arXiv:2301.07882*, 2023.

Morningstar, W., Ham, C., Gallagher, A., Lakshminarayanan, B., Alemi, A., and Dillon, J. Density of states estimation for out of distribution detection. In *International Conference on Artificial Intelligence and Statistics*, pp. 3232–3240, 2021.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019a.

Nalisnick, E., Matsukawa, A., Teh, Y. W., and Lakshminarayanan, B. Detecting out-of-distribution inputs to deep generative models using typicality. *arXiv:1906.02994*, 2019b.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Pidstrigach, J. Score-based generative models detect manifolds. In *Advances in Neural Information Processing Systems*, 2022.

Pope, P., Zhu, C., Abdelkader, A., Goldblum, M., and Goldstein, T. The intrinsic dimension of images and its impact on learning. In *International Conference on Learning Representations*, 2021.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. *Dataset Shift in Machine Learning*. MIT Press, 2008.

Rabanser, S., Günnemann, S., and Lipton, Z. Failing loudly: An empirical study of methods for detecting dataset shift. In *Advances in Neural Information Processing Systems*, 2019.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., Depristo, M., Dillon, J., and Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, 2019.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pp. 1278–1286, 2014.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Ross, B. L. and Cresswell, J. C. Tractable density estimation on learned manifolds with conformal embedding flows. In *Advances in Neural Information Processing Systems*, 2021.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K.-R., and Kloft, M. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2020.

Saremi, S. and Hyvärinen, A. Neural empirical bayes. *Journal of Machine Learning Research*, 20(181):1–23, 2019.

Schirrmeister, R., Zhou, Y., Ball, T., and Zhang, D. Understanding anomaly detection with deep invertible networks through hierarchies of distributions and features. In *Advances in Neural Information Processing Systems*, 2020.

Serrà, J., Álvarez, D., Gómez, V., Slizovskaia, O., Núñez, J. F., and Luque, J. Input complexity and out-of-distribution detection with likelihood-based generative models. In *International Conference on Learning Representations*, 2020.

Sirignano, J. and Cont, R. Universal features of price formation in financial markets: Perspectives from deep learning. *Quantitative Finance*, 19(9):1449–1459, 2019.

Sneyers, J. and Wuille, P. FLIF: Free lossless image format based on MANIAC compression. In *International Conference on Image Processing*, pp. 66–70, 2016.

Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., and Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265, 2015.

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

Song, Y., Durkan, C., Murray, I., and Ermon, S. Maximum likelihood training of score-based diffusion models. In *Advances in Neural Information Processing Systems*, 2021a.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021b.

Stanczuk, J., Batzolis, G., Deveney, T., and Schönlieb, C.-B. Your diffusion model secretly knows the dimension of the data manifold. *arXiv:2212.12611*, 2022.

Tempczyk, P., Michaluk, R., Garncarek, L., Spurek, P., Tabor, J., and Golinski, A. LIDL: Local intrinsic dimension estimation using approximate likelihood. In *International Conference on Machine Learning*, pp. 21205–21231, 2022.

Vahdat, A. and Kautz, J. NVAE: A deep hierarchical variational autoencoder. In *Advances in Neural Information Processing Systems*, 2020.

Vahdat, A., Kreis, K., and Kautz, J. Score-based generative modeling in latent space. In *Advances in Neural Information Processing Systems*, 2021.

van Amersfoort, J., Smith, L., Jesson, A., Key, O., and Gal, Y. On feature collapse and deep kernel learning for single forward pass uncertainty. *arXiv:2102.11409*, 2021.

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

von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., and Wolf, T. Diffusers: State-of-the-art diffusion models. `https://github.com/huggingface/diffusers`, 2022.

Wegner, S.-A. Lecture notes on high-dimensional data. *arXiv:2101.05841*, 2021.

Wei, H., Xie, R., Cheng, H., Feng, L., An, B., and Li, Y. Mitigating neural network overconfidence with logit normalization. In *International Conference on Machine Learning*, pp. 23631–23644, 2022.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv:1708.07747*, 2017.

Xiao, Z., Yan, Q., and Yali, A. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. In *Advances in Neural Information Processing Systems*, 2020.

Xie, J., Lu, Y., Zhu, S.-C., and Wu, Y. A theory of generative ConvNet. In *International Conference on Machine Learning*, pp. 2635–2644, 2016.

Yoon, S., Noh, Y.-K., and Park, F. Autoencoding under normalization constraints. In *International Conference on Machine Learning*, pp. 12087–12097, 2021.

Zhang, L., Goldstein, M., and Ranganath, R. Understanding failures in out-of-distribution detection with deep generative models. In *International Conference on Machine Learning*, pp. 12427–12436, 2021.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018.

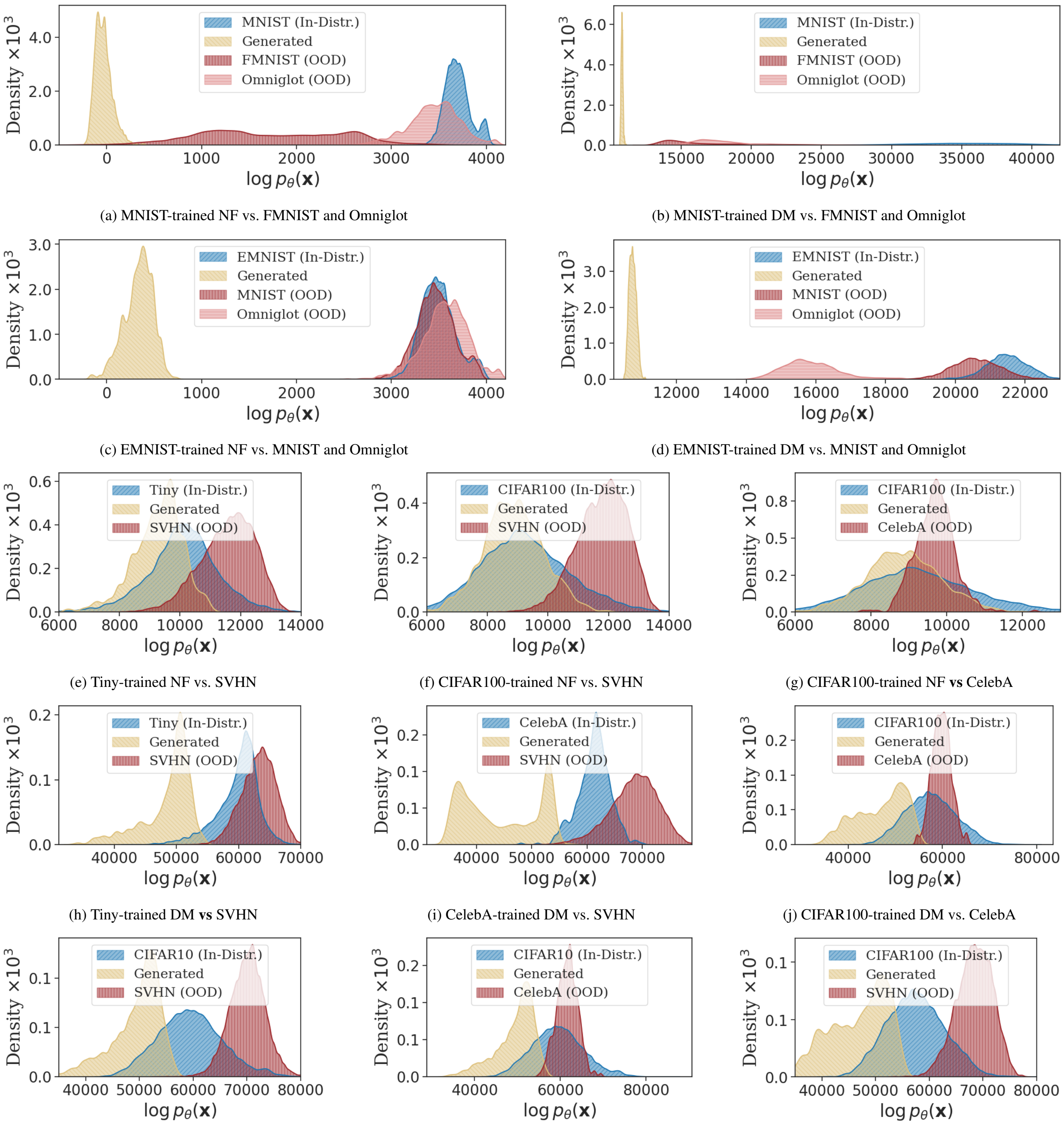
A. Diagnosing Pathologies in Normalizing Flows and Diffusion Models

In this section, we list the full extent of the pathologies we identified in our experiments. The first class is the standard one, in which models assign equal or higher likelihoods to OOD data than to in-distribution data; furthermore, we observe a new class of pathologies where the model assigns low likelihoods to its own generated samples compared to OOD data. In addition to Figure 2, which shows that FMNIST vs. MNIST, CIFAR-10 vs. SVHN, and CelebA vs. SVHN are pathological, Figure 5 depicts pathological behaviour for EMNIST vs. MNIST, EMNIST vs. Omniglot, Tiny ImageNet vs. SVHN, CelebA vs. SVHN, CIFAR100 vs. SVHN, and CIFAR100 vs. CelebA.

Regarding the second class of pathologies, we observe a stark difference between the likelihoods of generated samples and the in-distribution ones. To demonstrate this, we also visualize the likelihoods of generated samples in Figure 2 and Figure 5. Notably, generated sample likelihoods are almost always smaller than both in-distribution as well as OOD samples. To the best of our knowledge, generated samples having lower likelihoods than OOD data is a new class of pathologies not previously discussed in the literature. This raises new unexplained phenomena, even in cases such as MNIST- and Omniglot-trained models which were previously thought to be non-pathological (Nalisnick et al., 2019a).

The rationale of Schirrmeister et al. (2020) might provide an explanation for why these new pathologies occur in NFs. They claim that the multi-scale NF architectures used for modelling images pick up on high-frequency features, such as sharp edges, which are prevalent in any natural dataset; this prompts the latent variables corresponding to shallow scales to *sharply* center around zero (i.e. with a very small variance), and in turn the likelihood of these latent variables strongly inflates the total likelihood, regardless of whether the original datapoint is OOD or not. When passing a generated sample (that appears semantically similar to in-distribution data) inversely through an NF, we get shallow-scale latent variables that have a standard deviation of 1 by design. Therefore, compared to OOD data selected from natural images, these latent variables are not as sharply concentrated around 0, and hence produce relatively smaller likelihoods. That said, none of the explanations in related works such as those by Kirichenko et al. (2020) or Schirrmeister et al. (2020) directly provide any conclusive insight into the inductive biases that make DMs behave this way. Among related work on DMs, Goodier & Campbell (2023) offers a potential explanation for the pathology, suggesting that the score network prioritizes high-frequency features in earlier timesteps—a characteristic also common in OOD data; however, the evidence they present does not adequately address the occurrence of the pathology on generated samples, which is a novel observation of our work.

We emphasize that these explanations clarify why NF or DM likelihoods for OOD data points can behave pathologically, but not why low likelihood data is generated in the first place. Therefore they do not contradict, but rather complement, our explanation. Similar to in-distribution data, generated samples have comparatively higher probability masses, potentially even surpassing that of in-distribution data. Consequently, in scenarios where the likelihood is pathological for OOD data, the LID is anticipated to be small; our experiments in Table 8 further substantiate this observation by getting consistent performance across OOD detection tasks. Nevertheless, studying the inductive biases in DMs that lead to this new pathological behaviour of likelihoods on generated data is an interesting open question requiring future research.



(a) MNIST-trained NF vs. FMNIST and Omniglot

(b) MNIST-trained DM vs. FMNIST and Omniglot

(c) EMNIST-trained NF vs. MNIST and Omniglot

(d) EMNIST-trained DM vs. MNIST and Omniglot

(e) Tiny-trained NF vs. SVHN

(f) CIFAR100-trained NF vs. SVHN

(g) CIFAR100-trained NF **vs** CelebA

(h) Tiny-trained DM **vs** SVHN

(i) CelebA-trained DM vs. SVHN

(j) CIFAR100-trained DM vs. CelebA

(k) CIFAR10-trained DM vs. SVHN

(l) CIFAR10-trained DM vs. CelebA

(m) CIFAR100-trained DM vs. SVHN

Figure 5: Overview of likelihood pathologies: **(a-d)** Models trained on EMNIST or MNIST assign the highest likelihoods to in-distribution data as expected, but obtain strikingly low likelihoods on generated samples – even lower than for OOD data. **(e-m)** Pathologies on RGB datasets where both the in-distribution samples and the generated samples are assigned likelihoods smaller than that of OOD datapoints.

B. A Mathematical Link between Probability Mass and Local Intrinsic Dimension

Here we make the association between probability mass and LID more mathematically concrete. This connection is enabled by a surprising result linking Gaussian convolutions and LID (Loaiza-Ganem et al., 2022; Tempczyk et al., 2022). Intuitively, adding high-dimensional but low-variance Gaussian noise corrupts p_θ more easily when p_θ concentrates around low-dimensional regions (see Figure 1 from Tempczyk et al. (2022)). Comparing p_θ convolved with noise for different noise levels allows one to infer LID from the rate at which p_θ is corrupted as the noise increases. More formally, defining the convolution between a model density p_θ and a Gaussian with log standard deviation r as

$$\rho_r(\mathbf{x}) \coloneqq [p_\theta(\cdot) * \mathcal{N}(\,\cdot\,; \mathbf{0}, e^{2r}\boldsymbol{I}_d)](\mathbf{x}) = \int p_\theta(\mathbf{x}-\mathbf{x}')\mathcal{N}(\mathbf{x}'; \mathbf{0}, e^{2r}\boldsymbol{I}_d)\mathrm{d}\mathbf{x}', \tag{10}$$

Tempczyk et al. (2022) showed that under mild regularity conditions, for sufficiently negative r (i.e. low variance noise),

$$\log \rho_r(\mathbf{x}) = r(\mathrm{LID}_\theta(\mathbf{x}) - d) + \mathcal{O}(1). \tag{11}$$

Equation 11 suggests that, for sufficiently negative r, the rate of change of $\log \rho_r(\mathbf{x})$ with respect to r can be used to estimate LID, since

$$\frac{\partial}{\partial r} \log \rho_r(\mathbf{x}) \approx \mathrm{LID}_\theta(\mathbf{x}) - d. \tag{12}$$

We will now link the above quantity to the probability mass that p_θ assigns around $\mathbf{x}$. Let $B_R(\mathbf{x}) \coloneqq \{\mathbf{x}' \in \mathcal{X} : \|\mathbf{x}' - \mathbf{x}\|_2^2 \leq R^2\}$ be an ℓ_2 ball of radius R around $\mathbf{x}$. The probability that p_θ assigns to this ball is

$$\mathbb{P}_\theta\left(\mathbf{x}' \in B_R(\mathbf{x})\right) = \int_{B_R(\mathbf{x})} p_\theta(\mathbf{x}')\mathrm{d}\mathbf{x}' = \mathrm{vol}(B_R(\mathbf{0})) \cdot \left[p_\theta(\cdot) * \mathcal{U}(\,\cdot\,; B_R(\mathbf{0}))\right](\mathbf{x}), \tag{13}$$

where vol(B) denotes the d-dimensional Lebesgue measure of B (i.e. its volume), and $\mathcal{U}(\,\cdot\,; B)$ is the density of the uniform distribution on B – we note that vol$(B_R(\mathbf{x}))$ is not strictly equivalent to the volume defined in Equation 5, as the latter depends on p_θ and "ignores directions along which p_θ is negligible". We now leverage the standard and well-known result that in high dimensions, the uniform distribution on the ball is approximately Gaussian, $\mathcal{U}(\,\cdot\,; B_{e^r\sqrt{d}}(\mathbf{0})) \approx \mathcal{N}(\,\cdot\,; \mathbf{0}, e^{2r}\boldsymbol{I}_d)$.[3] This result combined with Equation 10 suggests that, if we take $R = e^r\sqrt{d}$, we can approximate the log probability mass in Equation 13 as

$$\log \mathbb{P}_\theta\left(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x})\right) \approx \log \mathrm{vol}(B_{e^r\sqrt{d}}(\mathbf{0})) + \log \rho_r(\mathbf{x}). \tag{14}$$

Differentiating with respect to r then yields

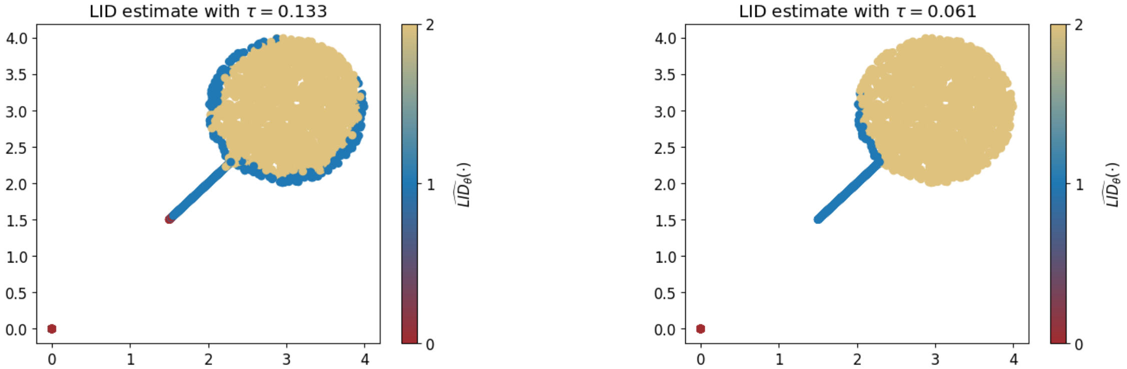
$$\frac{\partial}{\partial r} \log \mathbb{P}_\theta\left(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x})\right) \approx \mathrm{LID}_\theta(\mathbf{x}) + \left[\frac{\partial}{\partial r} \log \mathrm{vol}(B_{e^r\sqrt{d}}(\mathbf{0})) - d\right], \tag{15}$$

which establishes a clear relationship between (log) probability mass and LID. Note that the second term on the right hand side of Equation 15 is just a fixed function of r which does not depend on the model p_θ nor on $\mathbf{x}$, so that thresholding this quantity is equivalent to thresholding $\mathrm{LID}_\theta(\mathbf{x})$ (whenever the model and r are kept fixed). In other words, small/large values of $\mathrm{LID}_\theta(\mathbf{x})$ are equivalent to small/large values of the rate of change of the log probability mass around $\mathbf{x}$, i.e. $\frac{\partial}{\partial r} \log \mathbb{P}_\theta(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x}))$. In turn, our dual threshold algorithm described in Subsection 3.4 can be interpreted as thresholding on an estimate of $\frac{\partial}{\partial r} \log \mathbb{P}_\theta(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x}))$ rather than on $\widehat{\mathrm{LID}}_\theta(\mathbf{x})$, which matches the understanding that we derived in Section 3: even if $\log p_\theta(\mathbf{x})$ is large, if $\log \mathbb{P}_\theta(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x}))$ increases slowly as a function of r (i.e. $\mathrm{LID}_\theta(\mathbf{x})$ is small, this corresponds to case (ii) in Subsection 3.4), we can sensibly expect p_θ to assign lower probability mass to a small ball around $\mathbf{x}$ as compared to the case where both $\log p_\theta(\mathbf{x})$ is large and $\log \mathbb{P}_\theta(\mathbf{x}' \in B_{e^r\sqrt{d}}(\mathbf{x}))$ increases quickly (which now corresponds to case (iii) in Subsection 3.4). We can thus understand our dual threshold method as attempting to classify points as in-distribution when they have (relatively) large probability mass around them.

[3] Readers unfamiliar with this can see Saremi & Hyvärinen (2019) for a related derivation in a machine learning context. Note that this derivation shows Gaussians are approximately uniform on the boundary of the ball, but another classic result is that, in high dimensions, the majority of the mass of the ball lies near its boundary (see e.g. Wegner (2021)), so that uniform distributions on the ball or its boundary are also approximately equal. Similarly, a textbook derivation shows that when the dimensionality d is large, almost all the probability mass of a standard Gaussian is concentrated in an annulus at radius $\sqrt{d}$ (Blum et al., 2020).

Table 2: NF-based OOD detection performance (AUC-ROC) for various τ thresholds compared to using dataset-specific values of τ based on LPCA estimates of LID (higher is better).

	FMNIST vs. MNIST	MNIST vs. FMNIST	CIFAR10 vs. SVHN	SVHN vs. CIFAR10
$\tau = 10^{-10}$	$\mathbf{0.961}$	1.000	0.730	0.987
$\tau = 4.5 \times 10^{-5}$	0.957	1.000	0.737	0.987
Using LPCA	0.951	1.000	$\mathbf{0.936}$	0.987



(a) LID estimates for a DM.

(b) LID estimates for an NF.

Figure 6: LID estimates using our LPCA approach to set thresholds. Colours indicate $\widehat{\text{LID}}_\theta$ for a: **(a)** DM; **(b)** NF.

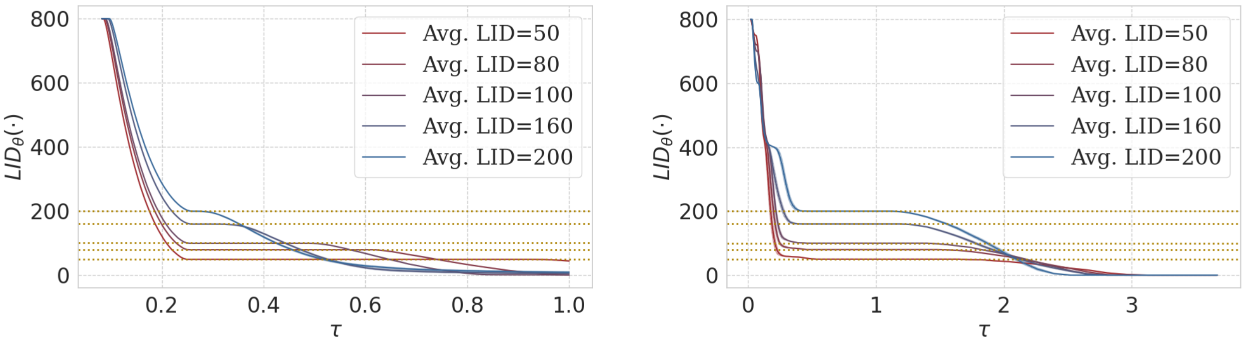
C. LID Estimation and Setting the Threshold

While setting a constant threshold τ can yield effective OOD detection results — as demonstrated in Figure 3 where we set it to an infinitesimal value of $\tau = 10^{-10}$ — choosing the right value of τ remains crucial for good performance across tasks. From a numerical perspective setting τ to an excessively small value would result in the LID estimator always predicting the ambient dimension, while setting it to an excessively large value will result in our estimator predicting 0. While Horvat & Pfister (2022) and Stanczuk et al. (2022) offer thorough methods for setting the threshold τ and estimating intrinsic dimension, our focus is primarily on effective LID estimation for OOD detection. Consequently, we adopt a straightforward and rapid approach for LID estimation that behaves well for our intents and purposes.

One sensible way of setting τ is to calibrate it based on another model-free estimator of LID using the training data. In particular, we perform local principal component analysis (LPCA) which is a model-free method for LID estimation. LPCA is similar to the LID estimator in Horvat & Pfister (2022) which also uses the concept of local linearizations. We use the `scikit-dimension` (Bac et al., 2021) implementation and use the algorithm introduced by Fukunaga & Olsen (1971) with `alphaFO` set to 0.001 to estimate the average LID of our training data. Then τ is set so that LID_θ estimates of the training dataset match the LPCA average.

To increase efficiency, we select a random set of 80 data points from our training set as representative samples. We then employ a binary search to fine-tune τ. During each iteration of the binary search, we compare the average LID_θ of our subsamples with the intrinsic dimension determined by LPCA. If the average LID_θ is lower, we increase τ; otherwise, we decrease it. We initially set τ's binary search range between 0 and 10^{10}, representing a wide range of plausible thresholds. Binary search is then executed in 50 steps to accurately ascertain a value of τ. Table 2 represents three distinct scenarios to assess how to set τ optimally for NF models. In the first two rows, τ is held constant across datasets, while in the third, τ is dynamically adjusted to each dataset based on the above approach. Although there is a minor performance drop in the FMNIST vs. MNIST comparison, this is offset by a notable enhancement in the CIFAR10 vs. SVHN case. This significant improvement further justifies our preference for this method of obtaining the threshold rather than setting it to a fixed value as a hyperparameter.

To assess the LID estimates across points with varying dimensionalities, we utilize a 2D lollipop dataset (Tempczyk et al., 2022) depicted in Figure 6. This dataset comprises points uniformly sampled from three distinct submanifolds: (i) the "candy" portion with an intrinsic dimension of 2; (ii) the "stick" with an intrinsic dimension of 1; and (iii) an isolated point



(a) LID estimates for a DM.

(b) LID estimates for an NF.

Figure 7: Average LID estimates for datapoints in 800-dimensional datasets with low intrinsic dimension across various threshold values. The dotted lines correspond to the the LID estimate obtained from LPCA. Results are shown for: **(a)** DMs; **(b)** NFs.

with an intrinsic dimension of 0. Utilizing LPCA, which accurately estimates the average LID, our LPCA-based method correctly sets the threshold τ. As shown in Figure 6, the LID estimates are generally precise on both NFs and DMs, with deviations only near the boundaries of the submanifolds, which is arguably appropriate behaviour.

In addition, to test our estimator in high-dimensional environments, we examine five distinct datasets, each with 800 ambient dimensions but varying intrinsic dimensionalities of 50, 80, 100, 160, and 200. For each dataset, we initially generate $10{,}000$ samples from an isotropic Gaussian in the respective intrinsic dimension and then replicate individual elements to expand the ambient dimensionality to 800. We then train both NFs and DMs on these datasets, adjusting the threshold values τ to derive the results presented in Figure 7. A key observation is that the model LID estimates stabilize at the actual intrinsic dimension for a range of thresholds τ. Moreover, LPCA effectively aligns with this stabilization point, enabling our binary search method to precisely estimate the optimal threshold τ – thus giving us confidence in our LID estimates.

Table 3: Essential hyperparameter settings for the normalizing flow models.

Property	Model Configuration
Learning rate	1×10^{-3}
Gradient Clipping	Value based (max = 1.0)
Scheduler	`ExponentialLR` (with a factor of 0.99)
Optimizer	`AdamW`
Weight decay	5×10^{-5}
Batch size	128
Epochs	400
Transform blocks	Actnorm → (1×1) Convolution → Coupling
Number of multiscale levels	7 levels
Coupling layer backbone	ResNet (channel size $= 64$, # blocks = 2, dropout = 0.2)
Masking scheme	Checkerboard
Latent Space	Standard isotropic Gaussian
Data pre-processing	Dequantization & Logit scaling
Data shape	$28 \times 28 \times 1$ for grayscale and $32 \times 32 \times 3$ for RGB

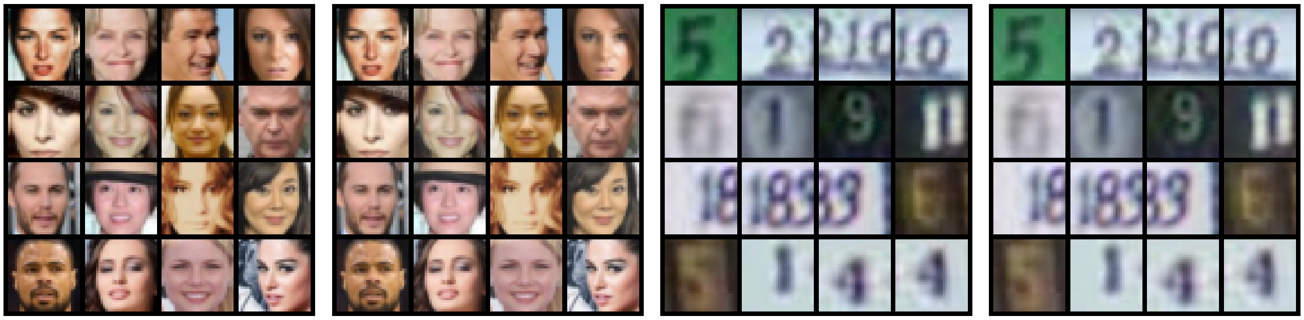
D. Experimental Details and Additional Experiments

D.1. Hyperparameter Setting for Normalizing Flows

We trained both Glow (Kingma & Dhariwal, 2018) and RQ-NSFs (Durkan et al., 2019) on our datasets, with the hyperparameters detailed in Table 3. Specifically, while Glow utilized an affine coupling layer, we adopted RQ-NSF's piecewise rational quadratic coupling with two bins and linear tails capped at 1. In Figure 9 and Figure 10, we highlight failure cases of the Glow architecture. The artifacts, particularly in CelebA, Tiny ImageNet, and Omniglot samples, stem from the affine coupling layers' unfavourable numerical properties. In contrast, the RQ-NSF architectures showed no such issues, leading us to adopt them for subsequent experiments.

In the context of OOD detection, expressive architectures sometimes face issues of numerical non-invertibility and exploding inverses, particularly with OOD samples (Behrmann et al., 2021). While expressive NFs adeptly fit data manifolds, their mapping from a full-dimensional space to a lower-dimensional one can cause non-invertibility, especially in OOD datapoints. Behrmann et al. (2021) specifically identified non-invertibility examples in Glow models on OOD data. Contrarily, the RQ-NSFs we trained according to the hyperparameter setup in Table 3 demonstrated full reconstruction on OOD data, as depicted in Figure 8. This is another reason why we chose RQ-NSFs.

We used an NVIDIA Tesla V100 SXM2 with 7 hours of GPU time to train each of the models.



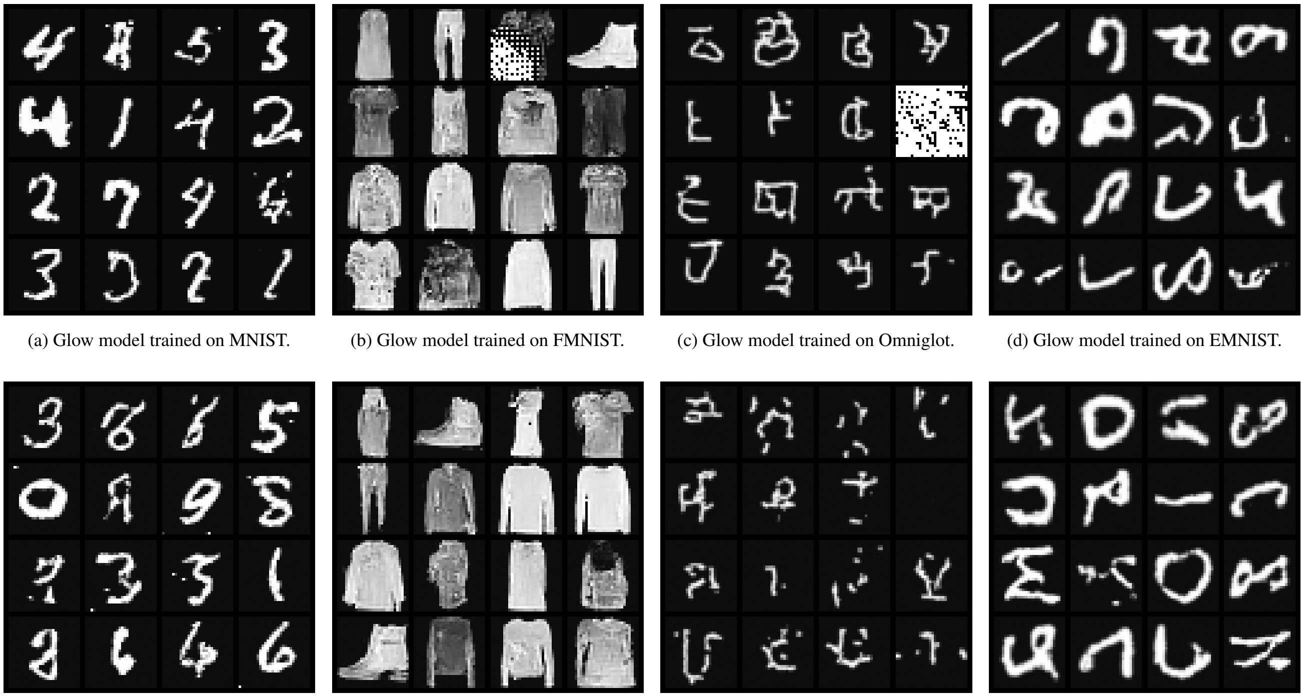
(a) Original samples from the test split.

(b) Reconstructed samples from the test split.

(c) Original samples from the OOD dataset.

(d) Reconstructed samples from the OOD dataset.

Figure 8: Numerical invertibility: **(a-b)** A random batch of samples and their reconstructions from the test split of an RQ-NSF model trained on CelebA. **(c-d)** A random batch of samples and their reconstructions from the OOD dataset, SVHN.



(a) Glow model trained on MNIST.

(b) Glow model trained on FMNIST.

(c) Glow model trained on Omniglot.

(d) Glow model trained on EMNIST.

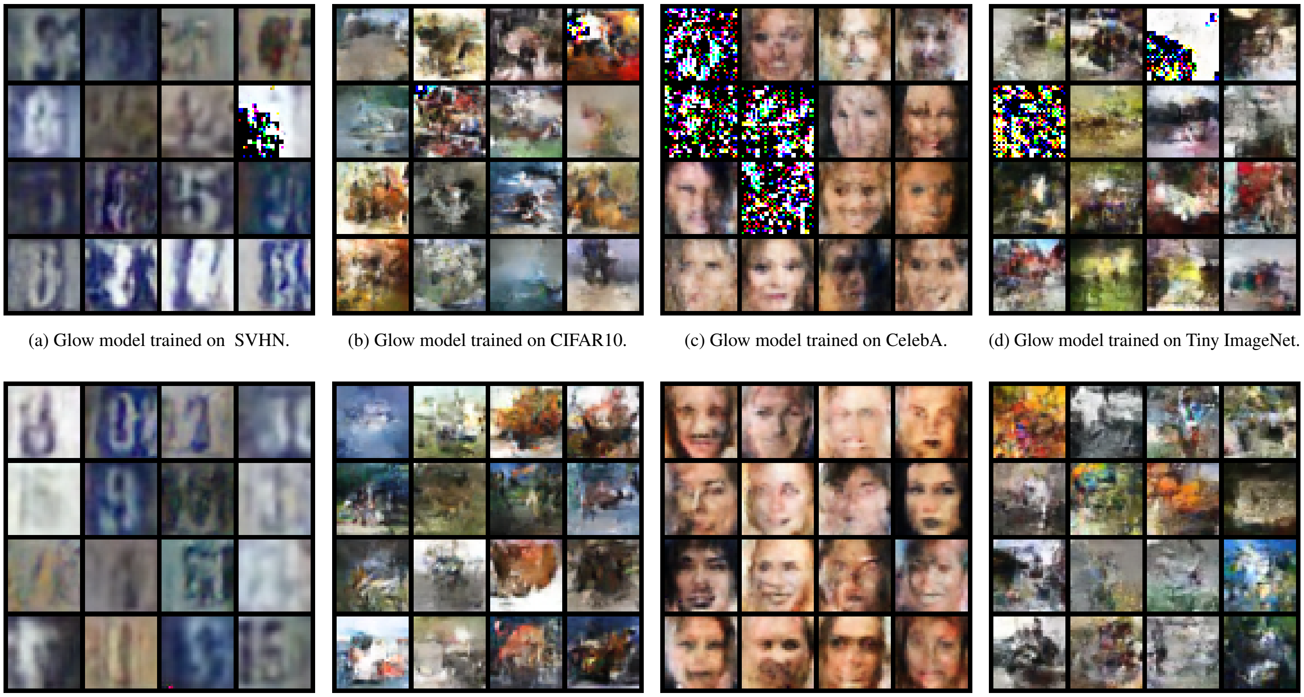
(e) RQ-NSF model trained on MNIST.

(f) RQ-NSF model trained on FMNIST.

(g) RQ-NSF model trained on Omniglot.

(h) RQ-NSF model trained on EMNIST.

Figure 9: Samples generated from models trained on the grayscale collection: due to numerical properties of affine coupling layers, Glow models tend to produce artifacts in their generated data.



(a) Glow model trained on SVHN.

(b) Glow model trained on CIFAR10.

(c) Glow model trained on CelebA.

(d) Glow model trained on Tiny ImageNet.

(e) RQ-NSF model trained on SVHN.

(f) RQ-NSF model trained on CIFAR10.

(g) RQ-NSF model trained on CelebA.

(h) RQ-NSF model trained on Tiny ImageNet.

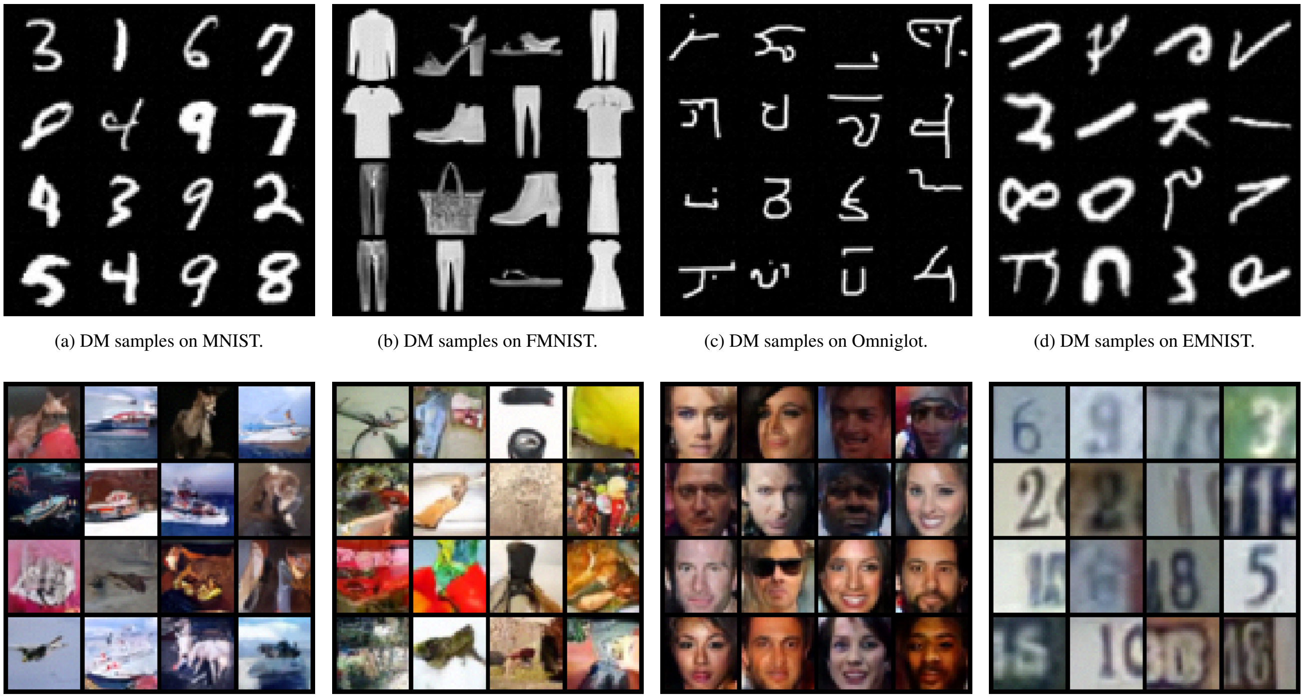
Figure 10: Samples generated from models trained on the RGB collection: the artifacts in Glow models are apparent.

Table 4: Essential hyperparameter settings for the Diffusion models.

Property	Model Configuration
Learning rate	5×10^{-5}
Gradient Clipping	Value based (max = 1.0)
Optimizer	`Adam`
Batch size	128
Epochs	200
Score-matching weighting	Likelihood weighting $\lambda(t) \coloneqq \mathrm{Var}(\mathbf{x}_{T-t} \mid \mathbf{x}_0)$
SDE dynamics	Variance preserving with $\beta(t) \coloneqq 0.1 + 20t$
Maximum time	$T = 1$
UNet # channels	$(2 \times 128) \to (2 \times 256) \to (2 \times 512)$
Attention	The penultimate UNet block performs spatial self-attention
Down/Up-sampling Blocks	ResNet
Data pre-processing	Dequantization & Scaling pixels between $[0, 1]$
Data shape	Resize everything to 32×32

D.2. Hyperparameter Setting for Diffusion Models

We utilized UNet backbones from the `diffusers` library (von Platen et al., 2022) for training our diffusion models. While the library provided the foundational architecture, we manually implemented additional functionalities such as LID estimation and log likelihood computations. The key hyperparameters guiding our training process are detailed in Table 4. Samples generated by these models, as illustrated in Figure 11, demonstrate a markedly superior quality compared to those produced by NFs. We used an NVIDIA Tesla V100 SXM2 with on average 4 hours of GPU time to train each of the models.



(a) DM samples on MNIST.

(b) DM samples on FMNIST.

(c) DM samples on Omniglot.

(d) DM samples on EMNIST.

(e) DM samples on CIFAR10.

(f) DM samples on Tiny ImageNet.

(g) DM samples on CelebA.

(h) DM samples on SVHN.

Figure 11: Samples generated from DMs trained on all the datasets: the number of diffusion steps to generate these samples are 1000.

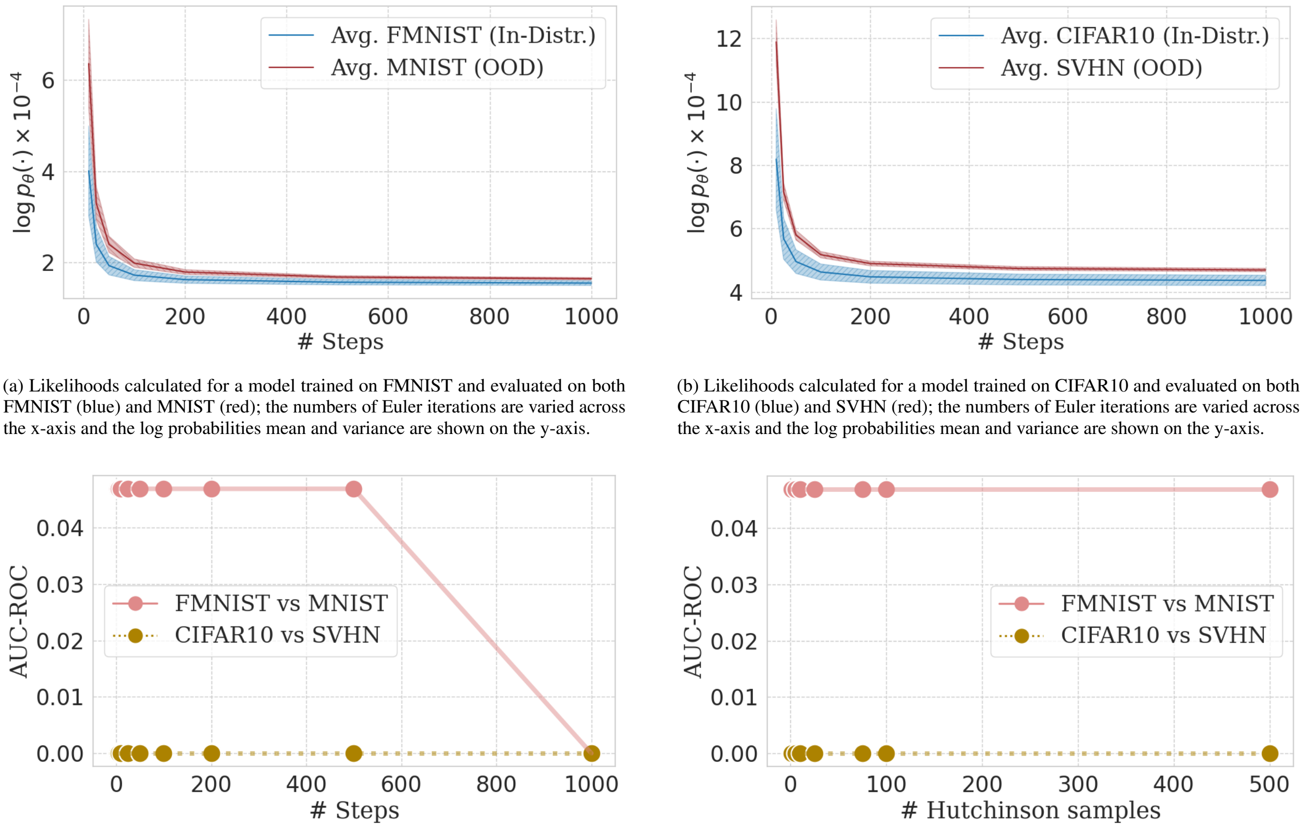
Table 5: AUC-ROC (higher is better) at A (vs.) B tasks. Ablation between variance preserving and exploding variations of DMs.

Trained on	MNIST		FMNIST		CIFAR10		SVHN	
OOD Dataset	FMNIST	Omniglot	MNIST	Omniglot	SVHN	CelebA	CIFAR10	CelebA
Variance Exploding DM Likelihood	**0.999**	**1.000**	0.211	0.953	0.088	0.485	**0.990**	**0.998**
Variance Exploding DM Dual Threshold	**0.999**	**1.000**	0.899	0.954	0.855	0.485	**0.990**	**0.998**
Variance Preserving DM Likelihood	0.996	1.000	0.240	0.952	0.064	0.360	0.996	0.996
Variance Preserving DM Dual Threshold	0.996	1.000	**0.912**	**0.959**	**0.944**	**0.648**	0.996	0.996

Computing Likelihoods To calculate the likelihood, one needs to solve an appropriate ordinary differential equation (for more details, please refer to Song et al. (2021b)). We use a standard Euler numerical solver and take 25 iterations to compute the likelihood estimates. Moreover, at each step of the solver, one needs to compute the trace of the score's Jacobian utilizing the Hutchinson trace estimator (Hutchinson, 1989). We set the number of samples for trace estimation to 25. These hyperparameters are chosen to ensure tractability of estimating likelihoods across entire datasets of size 2^{15}. In addition to that, we performed an extensive hyperparameter sweep to make sure that the pathology occurs even when likelihoods are computed more accurately with a stronger hyperparameter setting — i.e. with more steps in the differential equation solver or with more Hutchinson samples to estimate the trace. For further evaluation, we have picked two pathological OOD detection scenarios: FMNIST vs. MNIST and CIFAR10 vs. SVHN where likelihoods inflate on OOD data. In Figure 12a and Figure 12b we hold the number of Hutchinson samples at a constant 500 and vary the number of steps in the Euler solver. As the number of steps increases, the likelihood estimate concentrates more accurately around the true likelihood value. However, even with a larger number of steps, the ordering of OOD likelihoods versus their in-distribution counterparts does not change. We performed a similar experiment where we held the number of steps at a constant 500 and varied the numbers of Hutchinson samples; we observed no substantial change in the mean and standard deviation of the likelihoods beyond 25 samples. Finally, we also compute AUC-ROC for single threshold classifiers on likelihood values for these tasks to quantify the pathology. In Figure 12d, we maintain a constant step count of 25 while varying the number of Hutchinson samples. This adjustment reveals negligible variations in the AUC metric across both evaluated tasks. Similarly, Figure 12c documents our experiment where we fix the Hutchinson samples at 25 and modify the step count. Here, not only do we observe minimal fluctuations in the AUC-ROC metric with increasing steps, but we also note a deterioration in performance for the FMNIST vs. MNIST, indicating the pathology even becomes worse as the configuration for likelihoods becomes more accurate. These results rule out the possibility that the pathological behaviour of likelihoods for OOD detection in DMs was caused by poor density evaluation.

D.3. Ablations on Variance Exploding Diffusion Models

Stanczuk et al. (2022) propose their LID estimation technique for variance exploding DMs, we trained a series of variance exploding DMs with the same hyperparameter setting as Table 4 but with an appropriate noise scheduler and assessed their performance in OOD detection, as detailed in Table 5. The results demonstrate that combining variance preserving LID with likelihoods yields more consistent outcomes for OOD detection, leading us to favour this type of DM. We hypothesize this is due to either one (or a combination) of the following reasons that render the likelihood estimates inaccurate: (i) Variance preserving DMs are designed in such a way that p_T is very close to an isotropic Gaussian. In practice $\hat{p}_T$ is thus chosen as an isotropic Gaussian for these models. On the other hand, for variance exploding DMs, p_T does not converge, and $\hat{p}_T$ is simply set to a Gaussian with large variance. We thus hypothesize that the greater mismatch between p_T and $\hat{p}_T$ in the variance exploding setting might in turn render likelihoods less reliable, thereby adversely affecting the performance of OOD detection. (ii) The input to the score networks in variance exploding DMs is heterogeneous, meaning that at time t the input $\mathbf{x}_t$ to the score network $s_\theta(\mathbf{x}_t, t)$ might be either extremely large or small in scale. In particular, for $t \gg 0$, $\mathbf{x}_t$ would be extremely noisy with large variance, and as $t \to 0^+$, $\mathbf{x}_t$ would be on the scale of the image data. This results in an unstable score network that can negatively impact the likelihood estimates. It is important to highlight that Song & Ermon (2019) employ a technique named `CondInstanceNorm++` to address this challenge. Unfortunately, this method has not been incorporated into the existing `diffusers` diffusion architectures that we have used. Despite this omission, our variance preserving DMs yield satisfactory outcomes without necessitating such additional complexities.



(a) Likelihoods calculated for a model trained on FMNIST and evaluated on both FMNIST (blue) and MNIST (red); the numbers of Euler iterations are varied across the x-axis and the log probabilities mean and variance are shown on the y-axis.

(b) Likelihoods calculated for a model trained on CIFAR10 and evaluated on both CIFAR10 (blue) and SVHN (red); the numbers of Euler iterations are varied across the x-axis and the log probabilities mean and variance are shown on the y-axis.

(c) AUC-ROC evaluated by a single threshold OOD detector on likelihoods for two pathological tasks; the number of Euler iterations are varied across the x-axis and the AUC-ROC is shown on the y-axis.

(d) AUC-ROC evaluated by a single threshold OOD detector on likelihoods for two pathological tasks; the number of samples for Hutchinson trace estimation is varied across the x-axis and the AUC-ROC is shown on the y-axis.

Figure 12: The pathological behaviour holds even for strong hyperparameter settings for likelihood evaluation: **(a-b)** increasing the number of Euler steps produces more accurate likelihood estimates but does not change the ordering of the in-distribution and OOD estimates; **(c-d)** increasing the number of Euler steps and trace estimation samples does not improve the performance of a likelihood-based OOD detector.

D.4. Extra Dataset Pairs and Results on the Generated Samples Pathologies

Table 6 and Table 7 compare naïvely detecting OOD based on likelihoods against our dual threshold method across all tasks for NFs and DMs, respectively. In these tables, we use the suffixes "-train", "-test", and "-gen" when talking about datasets to specify if we are referring to the train set, test set, or generated samples, respectively. For datasets A and B, "A vs. B" indicates the OOD detection task that aims to distinguish A-test from B-test using the model p_θ pre-trained on A-train. When we write "A-gen vs. B", A-test is replaced by A-gen, but p_θ is still pre-trained on A-train. Note that even for A-gen vs. B tasks we calibrate τ using the training dataset. For fairness across all tasks, for any dataset, a random set of $2^{15} = 32768$ sampled datapoints with replacement has been considered. We can see an extremely consistent improvement, highlighting the relevance of dual thresholding.

The tasks of the form A-gen vs. B are especially relevant in our study as there is no discrepancy between the in-distribution and model manifolds by design, and thus any practical concerns about poor model fit and misalignment between generated and in-distribution samples would be addressed; this in turn tests our hypothesis in a more isolated manner. Furthermore, in Table 8 we have benchmarked our method against the baselines of Table 1 but replaced tasks A vs. B with A-gen vs. B. For all the A-gen vs. B tasks, we use the same τ as we used for the corresponding A vs. B task, i.e. we do not recalibrate τ. Overall, our dual threshold method outperforms *all* the baselines on NFs, but falls short in some comparisons against DM baselines. We hypothesize this discrepancy is because the LID estimates from Stanczuk et al. (2022) are not sufficiently accurate.

Table 6: AUC-ROC Results for NF Experiments: Comparing likelihood-only and dual-threshold methods (higher is better). The table is split into greyscale tasks (top) and RGB tasks (bottom). Entries showing over 10% improvement when comparing dual thresholding to the likelihood-only counterpart are boldfaced.

OOD Task Type	A-gen vs. B		A vs. B	
Dataset Pair A and B	(AUC-ROC) Likelihood	(AUC-ROC) Dual Threshold	(AUC-ROC) Likelihood	(AUC-ROC) Dual Threshold
FMNIST and MNIST	0.000	**0.999**	0.073	**0.951**
FMNIST and EMNIST	0.001	**0.956**	0.394	**0.596**
FMNIST and Omniglot	0.000	**0.996**	0.085	**0.864**
Omniglot and FMNIST	0.153	**1.000**	1.000	1.000
Omniglot and MNIST	0.016	**1.000**	1.000	1.000
Omniglot and EMNIST	0.000	**1.000**	0.984	0.984
EMNIST and FMNIST	0.038	**1.000**	0.998	0.998
EMNIST and MNIST	0.000	**1.000**	0.540	**0.806**
EMNIST and Omniglot	0.000	**1.000**	0.389	**0.824**
MNIST and FMNIST	0.007	**0.999**	1.000	1.000
MNIST and EMNIST	0.000	**1.000**	0.987	0.988
MNIST and Omniglot	0.000	**1.000**	0.796	0.855
Tiny and CelebA	0.639	0.660	0.821	0.821
Tiny and SVHN	0.030	**0.961**	0.154	**0.913**
Tiny and CIFAR100	0.694	**0.799**	0.805	0.831
Tiny and CIFAR10	0.694	**0.787**	0.805	0.831
CelebA and Tiny	0.938	0.960	0.906	0.928
CelebA and SVHN	0.148	**0.935**	0.146	**0.949**
CelebA and CIFAR100	0.945	0.968	0.921	0.942
CelebA and CIFAR10	0.948	0.967	0.921	0.939
SVHN and Tiny	0.972	0.972	0.989	0.989
SVHN and CelebA	0.984	0.984	0.996	0.996
SVHN and CIFAR100	0.967	0.967	0.986	0.986
SVHN and CIFAR10	0.970	0.970	0.987	0.987
CIFAR100 and Tiny	0.386	**0.448**	0.477	0.479
CIFAR100 and CelebA	0.226	**0.646**	0.370	**0.638**
CIFAR100 and SVHN	0.014	**0.941**	0.072	**0.933**
CIFAR100 and CIFAR10	0.403	**0.486**	0.490	0.491
CIFAR10 and Tiny	0.376	**0.535**	0.485	0.491
CIFAR10 and CelebA	0.239	**0.724**	0.391	**0.655**
CIFAR10 and SVHN	0.014	**0.950**	0.063	**0.936**
CIFAR10 and CIFAR100	0.426	**0.602**	0.521	0.562

Table 7: AUC-ROC Results for DM Experiments: Comparing likelihood-only and dual-threshold methods (higher is better). The table is split into greyscale tasks (top) and RGB tasks (bottom). Entries showing over 10% improvement when comparing dual thresholding to the likelihood-only counterpart are boldfaced.

OOD Task Type	A-gen vs. B		A vs. B	
Dataset Pair A and B	(AUC-ROC) Likelihood	(AUC-ROC) Dual Threshold	(AUC-ROC) Likelihood	(AUC-ROC) Dual Threshold
MNIST and EMNIST	0.000	**1.000**	0.846	0.846
MNIST and FMNIST	0.000	**0.999**	0.996	0.996
MNIST and Omniglot	0.000	**0.980**	1.000	1.000
EMNIST and MNIST	0.000	**1.000**	0.830	0.830
EMNIST and FMNIST	0.000	**0.991**	0.999	0.999
EMNIST and Omniglot	0.000	**1.000**	1.000	1.000
FMNIST and MNIST	0.000	**1.000**	0.240	**0.912**
FMNIST and EMNIST	0.000	**1.000**	0.339	**0.568**
FMNIST and Omniglot	0.000	**1.000**	0.952	0.959
Omniglot and MNIST	0.000	**0.971**	0.995	0.995
Omniglot and EMNIST	0.000	**0.952**	1.000	1.000
Omniglot and FMNIST	0.000	**0.979**	1.000	1.000
SVHN and Tiny	0.768	**0.891**	0.996	0.996
SVHN and CIFAR10	0.774	**0.833**	0.996	0.996
SVHN and CelebA	0.608	**0.802**	0.996	0.996
SVHN and CIFAR100	0.773	**0.834**	0.996	0.996
Tiny and SVHN	0.000	**0.996**	0.219	**0.951**
Tiny and CIFAR10	0.172	**0.708**	0.882	0.908
Tiny and CelebA	0.012	**0.919**	0.895	0.895
Tiny and CIFAR100	0.190	**0.701**	0.880	0.910
CIFAR10 and SVHN	0.000	**0.987**	0.064	**0.944**
CIFAR10 and Tiny	0.065	**0.675**	0.452	0.458
CIFAR10 and CelebA	0.000	**0.847**	0.360	**0.648**
CIFAR10 and CIFAR100	0.120	**0.688**	0.528	0.560
CelebA and SVHN	0.001	**0.985**	0.087	**0.747**
CelebA and Tiny	0.130	**0.716**	0.844	0.845
CelebA and CIFAR10	0.164	**0.702**	0.877	0.878
CelebA and CIFAR100	0.176	**0.700**	0.876	0.878
CIFAR100 and SVHN	0.000	**0.994**	0.045	**0.945**
CIFAR100 and Tiny	0.039	**0.766**	0.416	**0.465**
CIFAR100 and CIFAR10	0.052	**0.791**	0.470	0.504
CIFAR100 and CelebA	0.000	**0.902**	0.340	**0.663**

Table 8: AUC-ROC (higher is better) at A-gen vs. B task; due to the extensive computation time required for the DM baselines, the tasks are executed on subsamples of size 512. **Notation**: * tasks where likelihoods alone do not exhibit pathological behaviour, ‡ methods that employ external information or auxiliary models. For each task, we bold the best performing model.

Trained on	MNIST *		FMNIST		CIFAR10		SVHN *	
OOD Dataset	FMNIST	Omniglot	MNIST	Omniglot	SVHN	CelebA	CIFAR10	CelebA
NF Likelihood	0.007	0.000	0.000	0.000	0.014	0.239	**0.970**	**0.984**
NF $\|\frac{\partial}{\partial \mathbf{x}} \log p_\theta(\mathbf{x}_0)\|_2$	0.997	0.997	0.993	0.993	0.712	0.379	0.195	0.077
Complexity Correction‡	0.026	0.000	0.044	0.001	0.678	0.243	0.714	0.451
NF Likelihood Ratios‡	0.998	**1.000**	**1.000**	**1.000**	0.299	0.396	0.302	0.099
NF Dual Threshold (Ours)	**0.999**	**1.000**	0.999	0.996	**0.950**	**0.724**	**0.970**	**0.984**
DM Likelihood	0.843	0.925	0.000	0.777	0.000	0.001	0.806	0.625
DM Reconstruction	**1.000**	**1.000**	0.994	**0.998**	0.817	0.610	**0.927**	**0.930**
DM Likelihood Ratios‡	0.985	0.991	**0.971**	**0.998**	0.959	**0.835**	0.441	0.474
DM Dual Threshold (Ours)	0.843	0.931	0.880	0.938	**0.966**	0.797	0.806	0.625

Table 9: Ablation study on the necessity of dual threshold on AUC-ROC (higher is better).

Method	FMNIST vs. MNIST	MNIST vs. FMNIST	CIFAR10 vs. SVHN	SVHN vs. CIFAR10
NF $\text{LID}_\theta(\mathbf{x})$	0.951	0.006	0.936	0.014
NF $\Vert\frac{\partial}{\partial \mathbf{x}} \log p_\theta(\mathbf{x})\Vert_2$ and $\log p_\theta(\mathbf{x})$	0.516	0.983	0.722	0.962
NF $\text{LID}_\theta(\mathbf{x})$ and $\log p_\theta(\mathbf{x})$	0.951	1.000	0.936	0.987
DM $\text{LID}_\theta(\mathbf{x})$	0.912	0.004	0.944	0.005
DM $\Vert s_\theta(\mathbf{x}, 0)\Vert_2$ and $\log p_\theta(\mathbf{x})$	0.240	0.996	0.883	0.994
DM $\text{LID}_\theta(\mathbf{x})$ and $\log p_\theta(\mathbf{x})$	0.912	0.996	0.944	0.996

D.5. Evaluation Metric Details

Formal Definition of the ROC Curve for Dual Threshold Classifiers When the ROC graph does not follow a curve-like structure – as is the case with dual threshold classifiers – *optimal ROC curves* are used to generalize traditional ROC curves (Liu & Zhu, 2022). The optimal ROC curve is obtained by first establishing a partial order on the ROC graph; a classifier is "better" than another if it has both a smaller FPR and a larger TPR. Furthermore, the Pareto frontier of a partial order is the set of all maximal elements, and in turn, optimal ROC curves are defined by interpolating along the Pareto frontier of this partial order. For illustration, Figure 4 shows the frontier alongside the optimal ROC curve as the upper boundary of the ROC graph using red lines. In OOD detection, we consider a finite dataset of N samples, each being assigned a likelihood and LID, constituting the sets $\{\log p_\theta(\mathbf{x}^{(n)})\}_{n=1}^N$ and $\{\widehat{\text{LID}}_\theta(\mathbf{x}^{(n)})\}_{n=1}^N$, respectively. These values can be seen as "features" and the labels associated with a datapoint are binary: whether the corresponding datapoint is in-distribution or OOD. With that in mind, the possible achievable combinations of FPR-TPR pairs are finite, making the Pareto frontier a set of disjoint points rather than a continuous curve. Therefore, we obtain the optimal ROC curve by step-interpolating the Pareto frontier of FPR-TPR pairs. The area under the optimal ROC curve holds various interpretations, making it an appropriate generalization of the traditional AUC-ROC in many model specification scenarios. For an in-depth exploration, we direct readers to Liu & Zhu (2022).

Computing the Optimal ROC Curve To numerically compute the optimal ROC curve, one must first define a set of dual threshold classifiers as:

$$\Psi := \{(\psi_{\mathcal{L}} \pm \varepsilon, \psi_{\text{LID}} \pm \varepsilon) : \psi_{\mathcal{L}} \in \{\log p_\theta(\mathbf{x}^{(n)})\}_{n=1}^N, \psi_{\text{LID}} \in \{\widehat{\text{LID}}_\theta(\mathbf{x}^{(n)})\}_{n=1}^N\}, \tag{16}$$

where in practice we set $\varepsilon = 10^{-10}$. Since the cardinality of Ψ is $\mathcal{O}(N^2)$, computing all FPR-TPR pairs may not be feasible. To address this, we select a subset of Ψ and calculate the FPR-TPR pairs for this subset. We then compute their Pareto frontier, and in turn, the AUC-ROC. Although this method may underestimate the true AUC-ROC, we observe that the estimated AUC-ROC rapidly converges to the true value as the subset size increases. For the results presented in our tables, we have used a subset of Ψ consisting of 5×10^5 classifiers.

D.6. Extra Ablations

Throughout our paper, we have argued in favour of our dual threshold method, which combines likelihoods and LID estimates. To highlight that our strong performance is not just based on dual thresholding itself, we carry out an ablation where we use dual thresholding, but on likelihood and gradient norm $\Vert\frac{\partial}{\partial \mathbf{x}} \log p_\theta(\mathbf{x})\Vert_2$ pairs (or $\Vert s_\theta(\mathbf{x}, 0)\Vert_2$ for DMs). In this case, gradient norms are a proxy for how peaked a density is, in place of LID estimates. Table 9 shows the results for both NFs and DMs, highlighting that LID estimates are much more useful. The table also shows that using single thresholds with LID estimates is also not enough to reliably detect OOD points. In the case of DMs, for the results presented in both Table 9 and Table 1, we compute $s_\theta(\mathbf{x}, \epsilon)$ using a value of $\epsilon = 10^{-4}$. This approach is adopted to ensure numerical stability, a key consideration given that score-based diffusion models are known to become numerically unstable with extremely small timesteps (Pidstrigach, 2022; Lu et al., 2023).

D.7. Critical Analysis of OOD Baselines

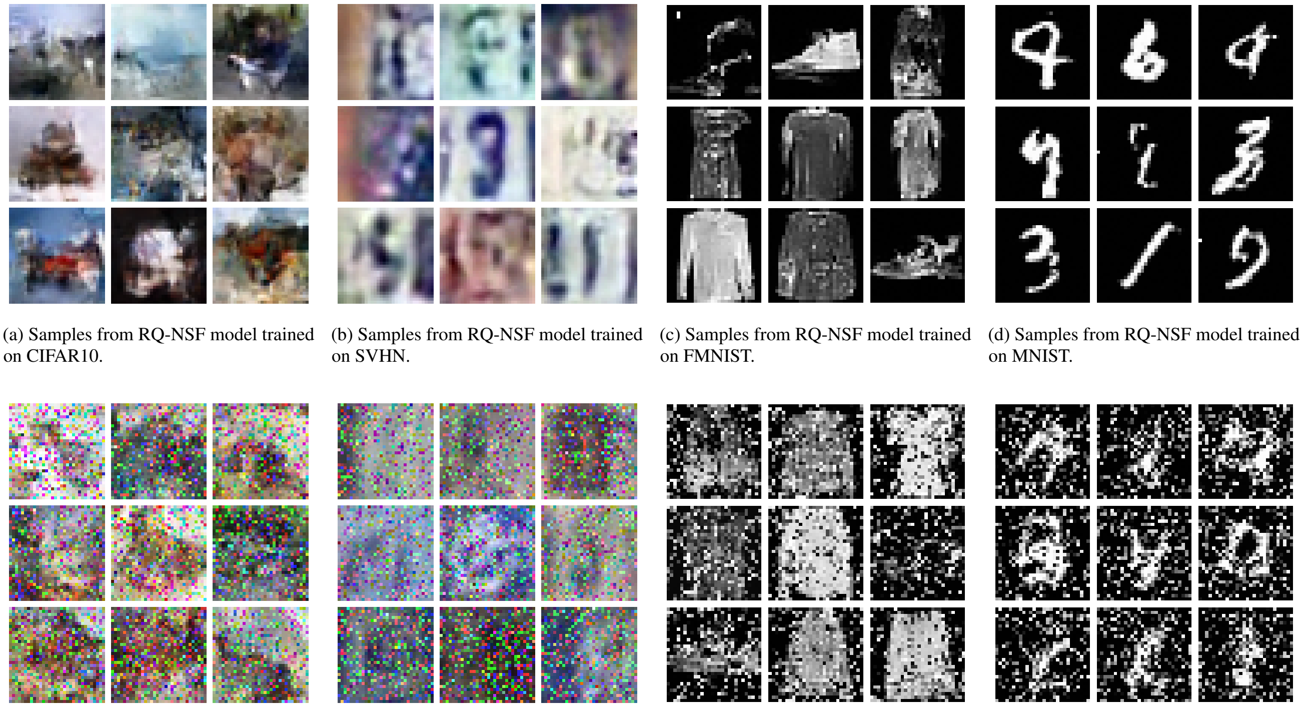
As we outlined in Section 5, when benchmarking against the complexity correction and likelihood ratio methods, we observed notable underperformance in non-pathological directions. Both methods aim to correct inflated likelihoods encountered in pathological OOD scenarios by assigning a score to each datapoint, which is obtained by adding a complexity term to the likelihood (Serrà et al., 2020), or subtracting a reference likelihood obtained from a model trained on augmented data (Ren et al., 2019). This score then becomes the foundation for their OOD detection through single thresholding. However, as we will demonstrate in this section, these techniques often necessitate an artificial hyperparameter setup to combine these metrics together, making it less than ideal.

Formally, both of these studies aim to find a score $\xi(\mathbf{x})$ to correct the inflated likelihood term $\log p_\theta(\mathbf{x})$, by adding a metric $m(\mathbf{x})$ as follows:

$$\xi(\mathbf{x}) = \log p_\theta(\mathbf{x}) + \lambda \cdot m(\mathbf{x}). \tag{17}$$

In Serrà et al. (2020), $\lambda = 1$ and $m(\mathbf{x})$ is the bit count derived by compressing $\mathbf{x}$ using three distinct image compression algorithms and selecting the least bit count from the trio (an ensemble approach as they describe). The algorithms include standard `cv2`, PNG, JPEG2000, and FLIF (Sneyers & Wuille, 2016). Moreover, we did not find any official implementation for the complexity correction method; however, since their algorithm was fairly straightforward, we re-implemented it according to their paper and it is readily reproducible in our experiments. On the other hand, Ren et al. (2019) propose training a reference likelihood model with the same architecture as the original model; however, on perturbed data. We employ another RQ-NSF, samples of which are depicted in the bottom row of Figure 13. Ren et al. (2019) claim that their reference model only learns background statistics that are unimportant to the semantics we care for in OOD detection; hence, subtracting the reference likelihood $m(\mathbf{x})$ can effectively correct for these confounding statistics that potentially inflate our original likelihoods. That said, they employ a hyperparameter tuning process on λ to ensure best model performance.

As illustrated in Figure 14, we sweep over values of λ and compare our method against all these models. While certain λ values enhance OOD detection in pathological scenarios, they falter in non-pathological contexts. In contrast, our dual thresholding remains robust irrespective of the scenario's nature. This observation underscores a significant gap in the OOD detection literature. While several methods address the OOD detection pathologies, many are overly specialized, performing well predominantly in the pathological direction. The results we report in Table 1 correspond to the best values of λ.



(a) Samples from RQ-NSF model trained on CIFAR10.

(b) Samples from RQ-NSF model trained on SVHN.

(c) Samples from RQ-NSF model trained on FMNIST.

(d) Samples from RQ-NSF model trained on MNIST.

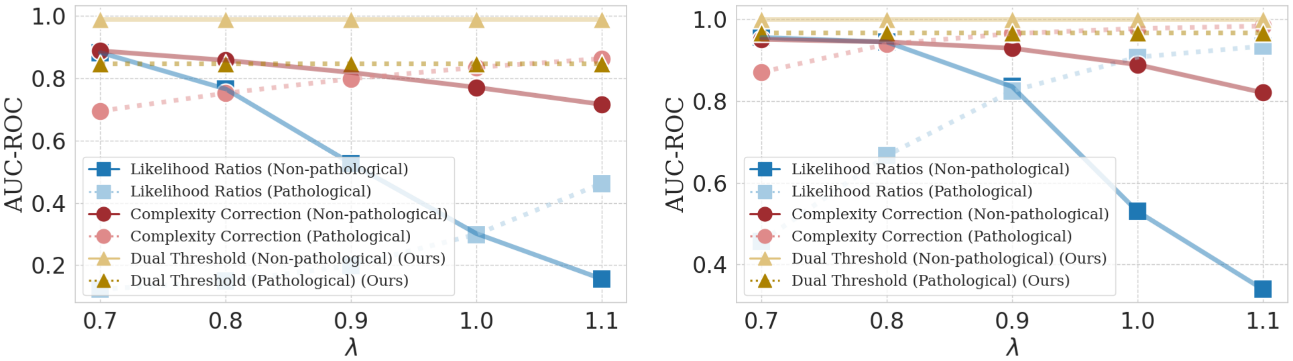
(e) Samples from background model trained on CIFAR10.

(f) Samples from background model trained on SVHN.

(g) Samples from background model trained on FMNIST.

(h) Samples from background model trained on MNIST.

Figure 13: Samples generated from normal and background models that are trained using the RQ-NSF hyperparameters provided in Table 3. The background models are trained on perturbed data, using the scheme presented by Ren et al. (2019).



(a) Performance comparison of different methods on two pathological and non-pathological OOD detection tasks obtained from the FMNIST and MNIST pair.

(b) Performance comparison of different methods on two pathological and non-pathological OOD detection tasks obtained from the CIFAR10 and SVHN pair.

Figure 14: Comparing our dual thresholding approach against all the different single score thresholding baselines by sweeping over different values of λ in Equation 17. The tasks that are considered are either: pathological such as **(a)** FMNIST vs. MNIST or **(b)** CIFAR10 vs. SVHN; or non-pathological such as **(a)** MNIST vs. FMNIST or **(b)** SVHN vs. CIFAR10.