Do Heavy Tails Help Diffusion? On the Subtle Trade-off Between Initialization and Training

Hamza Cherkaoui[1]* **Hélène Halconruy**[1,2]* **Antonio Ocello**[3]

[1] SAMOVAR, Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France
[2] Modal'X, Université Paris Nanterre, Nanterre, France
[3] CREST, CNRS, Ecole polytechnique, Groupe ENSAE-ENSAI, ENSAE Paris, Institut Polytechnique de Paris, 91120 Palaiseau, France
{hamza.cherkaoui,helene.halconruy}@telecom-sudparis.eu
antonio.ocello@ensae.fr

Abstract

Recent works have proposed incorporating heavy-tailed (HT) noise into diffusion- and flow-based generative models, with the goals of better recovering the tails of target distributions and improving generative diversity. This motivation is intuitive: if the data are heavy-tailed, HT noise may appear better matched than light-tailed (LT) Gaussian noise. However, replacing Gaussian noise by HT noise also changes the underlying estimation problem. In this paper, we revisit this paradigm through a combined theoretical and empirical study, establishing sampling-error bounds for two representative diffusion models driven by HT and LT noise. We show that HT diffusion models can achieve exponentially fast initialization, but accumulated training errors may offset this advantage. Experiments on synthetic and real-world datasets support our theoretical findings. Our results call into question a growing design trend in generative modeling and challenge the use of HT noise to improve rare-region exploration.

1 Introduction

Do heavy tails help? At first glance, the answer seems obvious. When the target distribution exhibits heavy-tailed (HT) behavior, replacing Gaussian noise with HT alternatives in diffusion models appears both natural and theoretically justified. Indeed, a growing body of literature adopts this principle, aiming to improve tail recovery by aligning the model with the data. In this work, we challenge this paradigm. We show that HT diffusion models can achieve exponentially fast initialization, but training errors accumulate along the reverse chain and may outweigh this advantage in finite-sample regimes.

Modern generative methods, such as diffusion and flow-based approaches, can be viewed as mechanisms that transport a simple reference distribution, typically Gaussian, into a complex target distribution through a continuous-time transformation. In diffusion models (see, *e.g.*, Sohl-Dickstein et al., 2015; Song and Ermon, 2019; Ho et al., 2020; Song et al., 2021, 2020), this is achieved by defining a forward noising process that progressively corrupts the data, and a reverse-time dynamics driven by a learned score function that reconstructs samples. Flow-based models (Lipman et al., 2023; Boffi and Vanden-Eijnden, 2023; Albergo et al., 2025; Liu et al., 2023) instead parameterize this transport through continuous-time dynamics governed by neural vector fields. Despite their apparent differences, both frameworks rely on accurately estimating the score or vector field associated with

*Equal contribution.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

intermediate distributions along the path. This estimation problem is central: the quality of the learned model ultimately depends on how well these quantities can be approximated from finite data.

Such approaches have rapidly gained traction across a wide range of disciplines, including finance (Sattarov et al., 2023; Guo et al., 2025), weather forecasting (Landry et al., 2026; Couairon et al., 2026), and medical imaging (Bedin et al., 2025). In these applications, the goal is not only to generate high-quality samples but also to faithfully represent the full complexity of the underlying data distribution, especially rare and extreme events, ranging from catastrophic risks in finance and insurance to fine-grained and underrepresented structures in high-dimensional data. This is especially important in domains such as meteorology, insurance, and risk modeling, where rare or extreme events play a central role. Failing to model such regions can lead to poor risk assessment, lack of robustness, or loss of diversity.

Representing the tails of a distribution poses a fundamental challenge: by construction, these regions are sparsely observed in finite datasets, making them difficult to learn. As a result, the ability to recover such regimes depends critically on how well the underlying learning procedure extrapolates beyond high-density regions. This has naturally motivated recent efforts to design generative mechanisms that better explore and capture the tails of the data distribution during both training and sampling.

Heavy-tailed extensions of diffusion and flow models. Recent work has proposed replacing the Gaussian ingredients of diffusion and flow-based generative models, either in the reference distribution or the noising process, to better accommodate *heavy-tailed target distributions*. Among these contributions, Denoising Lévy Probabilistic Models (DLPM) replace the Gaussian increments of DDPM (Ho et al., 2020) with isotropic α-stable noise and adapt the denoising objective to learn the reverse dynamics (Shariatian et al., 2025). We refer to Appendix B for a detailed discussion of related approaches. In this paper, we revisit this paradigm from a statistical estimation perspective, examining whether matching the noise to the tails of the data translates into improved learning and sampling performance.

Contributions. At a high level, these contributions reflect the growing interest in HT generative modeling, but they also leave open a broader question:

Do heavy tails help, or do they make learning harder?

A twofold approach. Since existing approaches are often studied under different assumptions, metrics, and experimental settings, their respective strengths and limitations remain poorly understood. In particular, it is still unclear whether HT mechanisms genuinely improve sampling performance once the statistical difficulty of learning the reverse dynamics is taken into account. We address this question from two *complementary* perspectives:

• *From a theoretical viewpoint*, we compare Gaussian variance-exploding (VE) diffusion and DLPM for generating α-stable distributions. We derive non-asymptotic bounds in total variation distance, decomposing sampling error into initialization and training errors. Our analysis reveals a fundamental *trade-off*. While DLPM achieves substantially smaller initialization error due to its forward dynamics rather than heavy tails, this gain may be offset by larger training errors. In particular, approximation and estimation errors accumulate over the full horizon T for DLPM, whereas the Gaussian VE diffusion bound is primarily controlled by the early-stopping parameter t_0. This suggests that the statistical difficulty of learning the reverse dynamics may dominate the initialization advantage of HT models in finite-sample regimes.

• *From an empirical viewpoint*, we compare DLPM against DDPM, DDIM, t-EDM, and linear Gaussian flow matching. We assess global distributional fit using MMD^2 and tail coverage using the Tail Coverage Error (TCE), defined later in the paper. We also investigate rare-class recovery on imbalanced image datasets to examine whether HT diffusion models provide benefits in this setting, as suggested by Shariatian et al. (2025). Finally, sample-size learning curves and zero-network controls help distinguish the effects of training data availability from those of the sampling dynamics themselves.

Main insight. Overall, our theoretical and empirical findings challenge the presumed advantage of HT noising for HT generation. Although HT initialization reduces initialization error, this gain can be outweighed by the difficulty of learning the reverse dynamics, particularly with limited training data.

Our results show that **tail alignment does not guarantee better generation**, and that LT alternatives can outperform HT models.

2 Sampling error decomposition

We now introduce a sampling-error decomposition that isolates the main sources of discrepancy between the generated distribution and the target. Specifically, we decompose the global error into two main contributions: an **initialization error** and a **training error**, the latter further splitting into **approximation** and **finite-sample estimation** components , with non-asymptotic bounds for each term.

This decomposition serves two purposes. First, it yields a *transparent interpretation* of how different components of the pipeline contribute to the overall error. Second, it enables a *quantitative comparison* between two representative formulations: a variance-exploding Gaussian diffusion (VE-GD) on the LT side (see subsection 2.1), and DLPM on the HT side. To make this comparison concrete, we measure discrepancies in **total variation (TV) distance**, which provides a strong, distribution-level notion of error: it directly controls the worst-case difference in expectations over bounded observables and is sensitive to mass in the tails, making it particularly well-suited to distinguish HT and LT behaviors.

Notations. We denote by $\mathbf{I}_d$ the $d \times d$ identity matrix and by $\mathcal{P}\left(\mathbb{R}^d\right)$ the set of probability distributions μ on $\mathbb{R}^d$. We write $X \overset{\mathcal{L}}{=} Y$ when the random variables X and Y have the same distribution. We denote by $\mathcal{C}^k\left(\mathbb{R}^{d_0};\mathbb{R}^{d_1}\right)$ the space of k-times continuously differentiable functions from $\mathbb{R}^{d_0}$ to $\mathbb{R}^{d_1}$ and by $W^{\beta,2}(\mathbb{R}^{d_0})$ the Sobolev space of functions on $\mathbb{R}^d$ possessing square-integrable weak derivatives up to order β (in the fractional sense when $\beta \notin \mathbb{N}$). When $d_1 = 1$, we simplify the notation to $\mathcal{C}^k\left(\mathbb{R}^{d_0}\right)$. We write $a \lesssim b$ if there exists a constant $C > 0$ such that $a \leq Cb$, and $a \asymp b$ if $a \lesssim b$ and $b \lesssim a$. When there is no ambiguity, $\|\cdot\|$ denotes the Euclidean norm on $\mathbb{R}^d$, and $\|\cdot\|_{\mathrm{F}}$ denotes the Frobenius norm for matrices. We denote by $\mathcal{P}(\mathbb{R}^d) \ni \mu \mapsto \widehat{\mu}$ the Fourier transform of a function or distribution (see, *e.g.*, Folland, 1999).

2.1 Sampling error of VE-GD

Gaussian forward dynamics and terminology. Gaussian score-based diffusion models (Song et al., 2020) can be built on different forward dynamics. We consider the *variance-exploding* (VE) formulation, in which Gaussian noise is added to the data without rescaling, so that the noise variance grows with time. Our analysis relies on Yu and Yu (2026), who provide explicit statistical guarantees for score estimation and sampling with heavy-tailed targets for a VE diffusion with constant diffusion coefficient; we therefore adopt this setting: the forward process $(X_t)_{t\in[0,T]}$ solves

$$\mathrm{d}X_t = \mathrm{d}\mathrm{B}_t\,, \qquad X_0 \sim \pi_{\mathrm{data}}\,, \tag{1}$$

with $(\mathrm{B}_t)_{t\in[0,T]}$ a standard Brownian motion. We refer to this Gaussian VE model as*VE-GD*; all statements below refer to the dynamics Equation 1. By linearity, with $Z \sim \mathcal{N}(0, I_d)$ independent of X_0 and ϕ_t the density of $\mathcal{N}(0, tI_d)$,

$$X_t \overset{\mathcal{L}}{=} X_0 + \sqrt{t}\, Z\,, \qquad p_t = \pi_{\mathrm{data}} * \phi_t\,. \tag{2}$$

Following the classical theory of diffusion time reversal (Anderson, 1982; Haussmann and Pardoux, 1986; Cattiaux et al., 2023), there exists a process $(\overleftarrow{X}_t)_{t\in[0,T]}$ such that $\mathcal{L}\big((X_t)_{t\in[0,T]}\big) = \mathcal{L}\big((\overleftarrow{X}_{T-t})_{t\in[0,T]}\big)$. A fundamental result of Haussmann and Pardoux (1986) shows that, under mild regularity assumptions, this time-reversed process is itself governed by an SDE whose drift depends explicitly on the score function of the forward marginals, namely $(t, x) \mapsto \nabla \log p_t(x)$. This characterization forms the theoretical basis of score-based generative methods, as it allows the reverse dynamics to be simulated through score estimation. In the present setting, the reverse-time dynamics associated with Equation 1 takes the form

$$\mathrm{d}\overleftarrow{X}_t = \nabla \log p_{T-t}(\overleftarrow{X}_t)\,\mathrm{d}t + \mathrm{d}\widetilde{\mathrm{B}}_t\,,$$

with $(\widetilde{\mathrm{B}}_t)_{t\geq 0}$ a Brownian motion.

HT assumption. Throughout the paper, we assume that the target distribution satisfies the following polynomial tail condition.

Assumption 2.1 (Tail behavior) The target density π_{data} has polynomial tail index $\gamma > 1$, *i.e.*,

$$\mathbb{P}_{X_0 \sim \pi_{\mathrm{data}}}(|X_0| > r) \sim r^{-\gamma} \quad \text{as } r \to \infty\,.$$

The Gaussian smoothing in Equation 2 regularizes the density at small scales while preserving polynomial tail behavior: if π_{data} has tail index γ, then typically (see, *e.g.*, Theorem 1.2 in Nolan, 2020) we also have

$$\mathbb{P}(|X_t| > r) \sim r^{-\gamma} \quad \text{as } r \to \infty\,.$$

Score approximation error. In practice, the score function $\nabla \log p_t$ is learned from data using neural network parameterizations $s_\theta(x,t)$ trained via score matching objectives (Hyvärinen and Dayan, 2005; Vincent, 2011; Efron, 2011). This approximation induces a statistical error that propagates along the reverse-time dynamics. To quantify both the score estimation error and the bias introduced by Gaussian smoothing, we rely on the framework of Yu and Yu (2026).

Beyond the tail assumptions described above, where the parameter γ governs the tail behavior of the distribution, we need a regularity condition that controls its smoothness at small scales. Suppose that the target density π_{data} satisfies the following Sobolev regularity condition.

Assumption 2.2 For some $\beta \in (0,2]$ and $L > 0$,

$$\pi_{\mathrm{data}} \in W^{\beta,2}(\mathbb{R}^d)\,, \qquad \int_{\mathbb{R}^d} (1+|\xi|^2)^\beta |\widehat{\pi_{\mathrm{data}}}(\xi)|^2 \,\mathrm{d}\xi \le L^2\,.$$

Sampling error decomposition for VE-GD. Under the previous tail and regularity assumptions, the following theorem quantifies the different sources of error arising in the generative process of score-based models. More precisely, it makes explicit the respective contributions of initialization, approximation, and statistical estimation, and provides non-asymptotic bounds for each component in total variation distance.

Theorem 2.1 (Sampling error decomposition for VE-GD) Suppose that Assumption 2.1 and Assumption 2.2 hold. Let $(p_t)_{t\in[0,T]}$ be the VE-GD forward marginals defined by Equation 2. Denote $\widehat{Y}$ the practical reverse process initialized from $\phi_T = \mathcal{N}(0, TI_d)$, driven by a learned score s_θ. Then, for any early stopping time $t_0 \in (0,T)$, the sampling error satisfies

$$\mathrm{TV}\big(\pi_{\mathrm{data}}, \mathcal{L}(\widehat{Y}_{T-t_0})\big) \lesssim \underbrace{t_0^{\frac{\beta(\gamma+1)}{d+2(\gamma+1)+2\beta}} + \mathrm{polylog}(n)\, n^{-\frac{\gamma+1}{2(d+\gamma+1)}}\, t_0^{-\frac{d(\gamma+1)}{4(d+\gamma+1)}}}_{\text{training error (approximation+estimation)}} + \underbrace{T^{-1/2}}_{\text{initialization error}} \quad . \tag{3}$$

The three terms in the bound admit a clear interpretation. The *first term* corresponds to the approximation error induced by Gaussian smoothing and early stopping, and reflects a bias-variance trade-off governed jointly by the Sobolev regularity β and the tail index γ. The *second term* captures the estimation error, which depends on the sample size n and deteriorates as t_0 decreases, due to the increasing difficulty of estimating the score at small noise levels. Finally, the *third term* is the initialization error, which arises from the mismatch between the true terminal distribution and the Gaussian prior, and decreases at a polynomial rate in the diffusion horizon T. This rate is a property of the VE dynamics Equation 1, in which X_0 is never rescaled and p_T approaches ϕ_T only through the growth of the noise variance.

The training contribution can be written in the generic form

$$t_0^a + \mathrm{polylog}(n)\, n^{-c}\, t_0^{-b}\,,$$

where

$$a = \frac{\beta(\gamma+1)}{d+2(\gamma+1)+2\beta}\,, \qquad b = \frac{d(\gamma+1)}{4(d+\gamma+1)}\,, \qquad c = \frac{\gamma+1}{2(d+\gamma+1)}\,.$$

In this decomposition, the first quantity corresponds to the approximation error (also interpretable as an early-stopping bias) while the second one captures the statistical estimation error arising from

score learning. Together, these two quantities exhibit the classical *bias-variance tradeoff*: choosing smaller values of t_0 reduces the approximation bias but simultaneously increases the statistical error due to the singular behavior of the score function as $t \to 0$.

Optimizing the tradeoff between these two contributions leads to the following optimal early-stopping scale: $t_0 \asymp n^{-\frac{c}{a+b}}$, giving the optimized training rate

$$\mathrm{polylog}(n)\, n^{-\frac{ca}{a+b}} = \mathrm{polylog}(n)\, n^{-\frac{2\beta(\gamma+1)}{4\beta(\gamma+1)+6\beta d+2d(\gamma+1)+d^2}} .$$

Hence, the VE-GD training error follows a nonparametric convergence rate in the sense of classical minimax nonparametric estimation theory; see, *e.g.*, Tsybakov (2008). As expected, the rate deteriorates with the ambient dimension d, while it improves with the Sobolev smoothness parameter β. Moreover, the dependence on the tail parameter γ shows that heavier tails lead to slower statistical convergence: larger γ (lighter tails) improve the rate, whereas small γ significantly degrade it. This reflects the additional difficulty of estimating score functions for heavy-tailed targets near small diffusion times.

Structural lower bounds. Although a matching minimax lower bound for the overall sampling error appears out of reach, each component of Equation 3 is unavoidable on an appropriate subclass of problems. *Initialization:* This term is deterministic and does not depend on n: it is a structural mismatch between the terminal marginal p_T and the prior ϕ_T, which cannot be reduced by collecting more data. For a translated target $\pi_{\mathrm{data}} = \delta_\mu * \pi$, the discrepancy $\mathrm{TV}(p_T, \phi_T)$ is of order $\|\mu\|\ T^{-1/2}$, so the polynomial rate $T^{-1/2}$ cannot be improved uniformly over the class within the VE dynamics. *Approximation:* One can construct target distributions whose Gaussian-smoothing bias at time t_0 is of the same order as the early-stopping term t_0^a. *Estimation:* Minimax lower bounds for score estimation of heavy-tailed targets are available in Yu and Yu (2026). These lower bounds hold term by term; they do not yield a lower bound for the sum of the three contributions.

2.2 Sampling error of DLPM

α-stable distributions. A natural way to adapt diffusion models to the HT setting consists in replacing the Gaussian noising process by stable distributions. This class of distributions arises as the limiting laws in the generalized central limit theorem (Gnedenko and Kolmogorov, 1968). In dimension one, an α-stable random variable is characterized by a tail index $\alpha \in (0, 2]$, together with location, scale, and skewness parameters. The case $\alpha = 2$ corresponds to the Gaussian distribution, whereas $\alpha < 2$ yields heavy-tailed laws satisfying

$$\mathbb{P}(|X| > r) \sim r^{-\alpha} \qquad \text{as } r \to \infty ,$$

and therefore $\mathbb{E}[|X|^p] < \infty$ if and only if $p < \alpha$ (Nolan, 2020). This tail behavior is directly connected to Assumption 2.1 of π_{data}: the underlying intuition is that α should be comparable to the tail index γ of π_{data}, so that the noising process and the target share a similar tail heaviness. In particular, smaller values of α correspond to heavier tails and weaker moment properties.

In the multivariate setting, several extensions are possible. We focus on isotropic, or rotationally invariant, α-stable distributions, whose characteristic function is given by

$$\mathbb{E}\big[\exp(iu^\top X)\big] = \exp\big(i\mu^\top u - \sigma^\alpha \|u\|^\alpha\big) , \qquad u \in \mathbb{R}^d .$$

We denote this law by $S_\alpha^{\mathbf{i}}(\mu, \sigma \mathbf{I}_d)$. For $\alpha = 2$, it coincides with a multivariate Gaussian, while for $\alpha < 2$ it provides a canonical heavy-tailed analogue used in stable-noise generative processes.

DLPM. We now turn to DLPM, taken as the canonical representative of generative models in the HT setting. In contrast with VE-GD, the forward noising process is driven by isotropic α-stable Lévy increments. We denote these increments by $A = (A_1, \ldots, A_T)$, where each A_t is α-stable. Conditionally on $A = a$, the forward path law is denoted by $k^{(\alpha)}_{0:T|a}$.

The learned reverse dynamics is modeled as a time-inhomogeneous Markov chain:

$$\overleftarrow{q}^{\theta}_{0:T|a} = \overleftarrow{q}^{\theta}_{T} \prod_{t=1}^{T} \overleftarrow{q}^{\theta}_{t-1|t,a} . \tag{4}$$

Exploiting the additive and scaling properties of α-stable laws, the forward recursion admits an explicit affine representation. In particular, the terminal variable can be written as

$$Y_T = a_T Y_0 + b_T S_\alpha \,, \tag{5}$$

where $S_\alpha \sim S_\alpha^{\mathbf{i}}(0, \mathbf{I}_d)$ is independent of Y_0, and (a_T, b_T) are deterministic coefficients determined by the noise schedule.

Sampling error bound for DLPM. To analyze the sampling error, we impose regularity conditions on the reverse object and on the forward signal-to-noise ratio.

Assumption 2.3 (Hölder regularity of the reverse object) Let $s_\alpha^\star : [0,T] \times \mathbb{R}^d \to \mathbb{R}^d$ be the true reverse object. We assume that there exists $\beta(\alpha) > 0$ and a constant $C_{\alpha,\beta} < \infty$ such that, for all $t \in [0,T]$,

$$s_\alpha^\star(t,\cdot) \in \mathcal{C}^{\beta(\alpha)}(\mathbb{R}^d;\mathbb{R}^d)\,, \qquad \sup_{t\in[0,T]} \|s_\alpha^\star(t,\cdot)\|_{\mathcal{C}^{\beta(\alpha)}} \le C_{\alpha,\beta}\,.$$

Assumption 2.4 (Exponential terminal signal-to-noise decay) Let $\rho_T := a_T/b_T$, where a_T and b_T are defined in Equation 5. We assume that there exist constants $C_\rho > 0$ and $c > 0$, independent of T, such that

$$|\rho_T| \le C_\rho e^{-cT}\,.$$

Equivalently, the contribution of the initial data Y_0 to the terminal variable Y_T decays exponentially fast relative to the accumulated α-stable noise.

The next theorem establishes a non-asymptotic decomposition of the DLPM sampling error into initialization, approximation, and statistical estimation components, mirroring the structure obtained previously for VE-GD.

Theorem 2.2 (Sampling error bound for DLPM) Let $k_{0|A}^{(\alpha)}$ denote the target conditional law at time 0, and let $\overleftarrow{q}^{\widehat{\theta}_n}_{0|A}$ be the learned reverse model based on n samples, where the parameter $\widehat{\theta}_n$ is chosen in a class of neural networks with m parameters. Under Assumption 2.3 and Assumption 2.4, for any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\mathbb{E}_A\left[\mathrm{TV}\left(k_{0|A}^{(\alpha)}, \overleftarrow{q}^{\widehat{\theta}_n}_{0|A}\right)\right] \lesssim \underbrace{T m^{-\beta(\alpha)/d} + T\sqrt{\frac{\mathrm{Comp} + \log(1/\delta)}{n}}}_{\text{training error (approximation+estimation)}} + \underbrace{e^{-cT}}_{\text{initialization error}}\,. \tag{6}$$

The three terms in the bound admit a clear interpretation. The first term is the *approximation* error, governed by the Hölder regularity $\beta(\alpha)$ of the reverse object. The second term captures the *estimation* error, which depends on the sample size n and the complexity Comp of the model class. The third term corresponds to the *initialization* error, which decays exponentially fast thanks to the vanishing signal-to-noise ratio at terminal time. The training contribution can be written as

$$T m^{-\beta(\alpha)/d} + T\sqrt{\frac{\mathrm{Comp} + \log(1/\delta)}{n}}\,.$$

Assuming $\mathrm{Comp} \asymp m$, balancing the approximation and estimation terms yields $m \asymp n^{\frac{d}{2\beta(\alpha)+d}}$, and therefore the optimized training rate

$$T n^{-\frac{\beta(\alpha)}{2\beta(\alpha)+d}}\,.$$

This coincides with the classical nonparametric rate for estimating a $\beta(\alpha)$-smooth object in dimension d; see, e.g., Tsybakov (2008). As expected, the rate deteriorates with the dimension d and improves with the regularity $\beta(\alpha)$. In contrast with the VE-GD bound in Equation 3, the initialization error decreases exponentially fast in T, rather than polynomially, which is an excellent initialization rate. In our analysis, this rate follows from Assumption 2.4, under which the weight of Y_0 in Y_T vanishes exponentially.

Structural lower bounds. As for VE-GD, the initialization, approximation, and estimation terms in Equation 6 are individually unavoidable on suitable subclasses. The initialization error is independent of n, while classical approximation and minimax arguments yield lower bounds for the local reverse-object errors. Moreover, the linear dependence on T is sharp: local kernel errors can accumulate across reverse steps without cancellation, yielding a global error of order T times the local error. Thus, without additional assumptions, the terms in Theorem 2.2 cannot be improved individually, although this does not establish a minimax lower bound for their sum. Complete arguments, including a proposition establishing the sharpness of the T-accumulation, are provided in subsection H.1.

2.3 Trade-off between the initialization error and the training error

What the comparison does and does not say. At first sight, DLPM appears more favorable for heavy-tailed data, with an exponentially decaying initialization error e^{-cT}, compared to the polynomial rate $T^{-1/2}$ of VE-GD: this is a genuine advantage of DLPM over the VE formulation considered here. The question is whether it survives the cost of training.

Theorem 2.1 and Theorem 2.2 provide a common *initialization–training* decomposition, allowing a comparison of the mechanisms governing both bounds, without establishing an ordering of actual sampling errors. The analyses retain model-specific regularity assumptions: Sobolev regularity for the data distribution in VE-GD, and Hölder regularity for the learned reverse object in DLPM. They also differ in their time structure: Theorem 2.1 concerns the continuous-time VE dynamics, whereas Theorem 2.2 concerns a discrete-time chain of T Markov kernels.

Together with the structural lower bounds, this yields the following *mechanistic* conclusion: *DLPM buys faster initialization at the price of a training cost that grows linearly with T.* Since increasing T enables the initialization gain, the training cost may dominate in finite-sample regimes. Further discussion of the assumptions and limitations of this comparison is provided in subsection H.2.

Each bound identifies its model's dominant bottleneck. For VE-GD, the dominant bottleneck is the training error, through the bias-variance trade-off near $t = 0$ governed by the early-stopping parameter t_0: the initialization error decays polynomially with T and becomes secondary once T is moderately large, while the training error is not amplified by T. For DLPM, the initialization error decreases with T whereas the training error increases linearly with it, so that better initialization is paid for in training. Within the DLPM bound alone, balancing e^{-cT} against $Tn^{-\beta(\alpha)/(2\beta(\alpha)+d)}$ yields an optimal horizon $T^\star \asymp c^{-1}\frac{\beta(\alpha)}{2\beta(\alpha)+d}\log n$ (up to lower-order terms), beyond which the training error dominates. *In finite-sample regimes, this crossover may occur before the initialization advantage becomes significant.*

Two independent sources for the training error gap. The difference between the training errors of the two models stems from two orthogonal mechanisms.

- *T-amplification, independent of the regularity of π_{data}.* Even when $\beta(\alpha) = \beta$, the DLPM training error scales linearly with T, whereas the VE-GD training error does not. By Proposition H.1, this dependence is intrinsic to the reverse chain. The VE-GD bound, established in continuous time, does not exhibit this factor; its training error is governed by early stopping at t_0.
- *Regularity of π_{data}, independent of T.* As discussed above, the two models learn different objects, endowed with different regularities: the VE-GD rate depends on the Sobolev regularity β of π_{data}, while the DLPM rate depends on the Hölder regularity $\beta(\alpha)$ of the reverse object $s^\star_\alpha$, with no assumed relationship between them. If $\beta(\alpha) < \beta$, DLPM incurs an additional regularity penalty.

Our conclusion does not require $\beta(\alpha) < \beta$: even when the regularities coincide, the linear factor T remains and is unavoidable.

For the two formulations considered, our results suggest a trade-off: DLPM can improve initialization while making the reverse dynamics more challenging to learn.

The theory provides *intuition* for why DLPM may be harder to train in finite-sample regimes, which we put to the test through the benchmarks in section 3.

3 Benchmark

We evaluate established heavy-tailed (HT) and light-tailed (LT) diffusion models on synthetic heavy-tailed and real-world datasets to empirically assess our theoretical findings. All experiments are implemented in PyTorch and conducted on NVIDIA V100 GPUs. The code is publicly available here. We compare DDPM-V (Ho et al., 2020), DDIM-V (Song et al., 2021), DLPM (Shariatian et al., 2025), t-EDM (Pandey et al., 2025), and Linear Gaussian flow (Lipman et al., 2023; Liu et al., 2023) on global fit, tail coverage, and rare-class recovery. DDIM-V and DDPM-V share the same model weights, follow the variance-preserving (VP) formulation, and employ V-prediction. DLPM and t-EDM zero-network controls retain the sampling dynamics with zeroed network parameters. Model details and references are given in subsection F.2.

Data and training. Table 1 covers two 150-dimensional stable targets, wet HRRR precipitation, positive wildfire sizes, and class-imbalanced CIFAR images. HRRR preprocessing removes exactly all-zero vectors, retaining tiny nonzero observations. CIFAR tests class-frequency imbalance rather than numerical heavy tails. Sources and exact splits appear in subsection F.1.

Data	d	Train	Val.	Eval.	$p_{\min}$
Stable iso.	150	25k	1k	5M	10^{-5}
Stable mix.	150	25k	1k	5M	10^{-5}
HRRR-E	150	20k	1k	79.5k	10^{-3}
HRRR-W	150	20k	1k	71.7k	10^{-3}
Wildfire	1	25k	1k	1M	10^{-4}
CIFAR100-LT	3,072	15.7k	1.96k	1.96k	-

Table 1: Dataset budgets (rounded). Eval. counts reference and generated samples per method; MMD^2 uses smaller subsets. The maximum quantile is $1 - p_{\min}$. Exact counts and preprocessing: Appendix F.

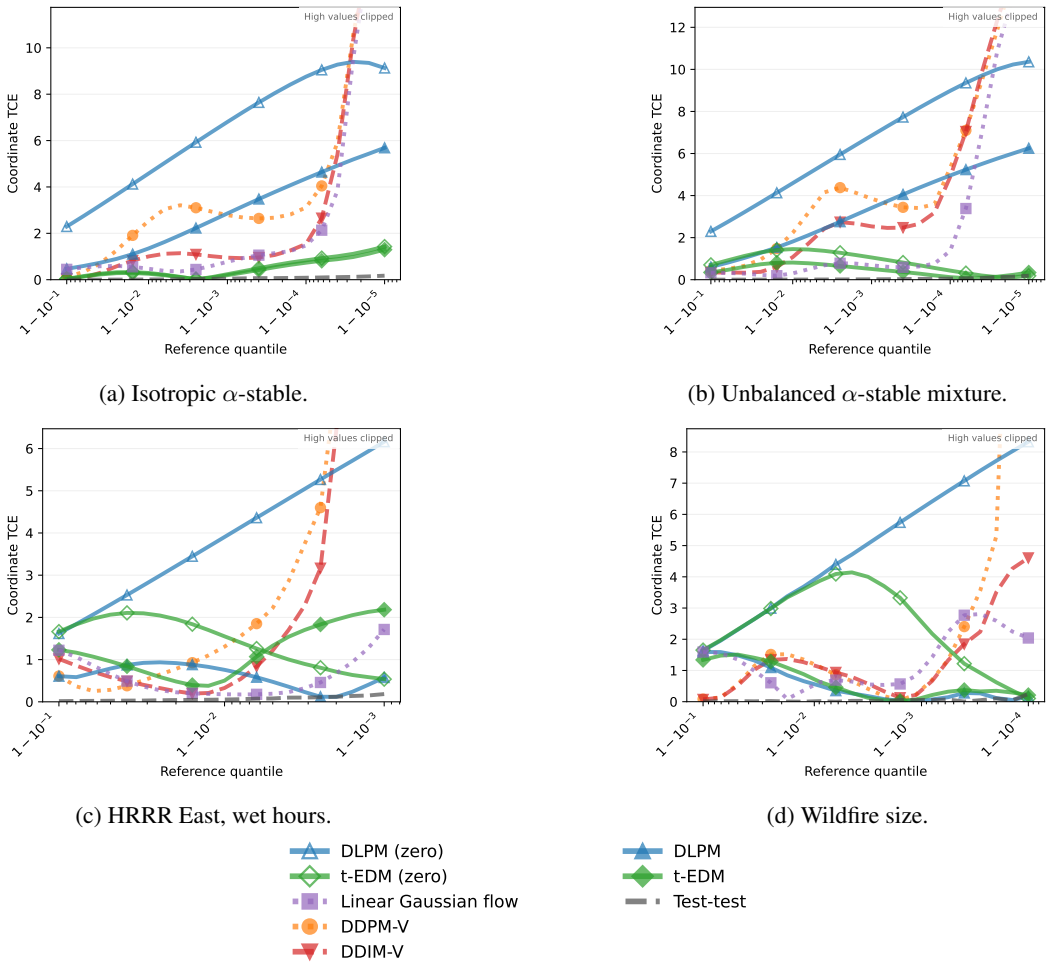
Vector models share a width-240, depth-4 MLP; CIFAR models share a two-layer transformer. Pilots use 16 epochs and validation selection, with three trials for vector data and one for CIFAR. Final training uses one seed, AdamW with cosine decay, and up to 720 epochs with early stopping. Detailed budgets are in Appendix F.

Evaluation. We report coordinate tail-coverage error (TCE), squared RBF-kernel MMD, and CIFAR rare-label recovery/label TV. Synthetic TCE uses coordinate magnitudes; real-vector TCE uses raw coordinates, consistently for reference and generated samples. Test-test is a finite-sample baseline. Definitions and evaluator checks are in Appendix E.

3.1 Synthetic and Tabular Results

In Figure 1, we first observe that t-EDM barely departs from its zero-network baseline on the synthetic datasets (a) and (b). This suggests that its performance is largely driven by the inductive bias of its sampling dynamics, rather than by learning the data distribution. Moreover, when training does affect tail calibration, it does not necessarily improve it, as illustrated in (c). DLPM exhibits a similar pattern, although it consistently departs from its zero-network baseline. Despite learning from the data, DLPM shows poor tail calibration on the synthetic datasets (a) and (b), while achieving substantially better tail recovery on HRRR East (c). The wildfire experiment (d) is particularly informative: in one dimension, with the same 25k training samples, both DLPM and t-EDM clearly improve upon their zero-network controls and recover meaningful tail behavior. This suggests that their difficulties in higher dimensions may be driven, at least in part, by finite-sample estimation rather than model design alone, see subsection G.2. These observations also highlight a limitation of TCE: it measures the recovery of extreme probability mass without necessarily capturing whether the underlying distributional structure has been learned. A heavy-tailed prior may therefore achieve favorable TCE values without yielding a faithful generative model.

Turning to LT approaches, we observe competitive tail recovery across datasets, at least up to the $1 - 10^{-2}$ quantile, and even better performance at more extreme quantiles, such as $1 - 10^{-4}$, in the synthetic cases (a) and (b). This illustrates that better tail alignment in HT models does not necessarily translate into improved generation. Consistent with Theorem 2.2 and Theorem 2.1, learning errors can play a decisive role in sampling performance, particularly in sparsely observed regions such as the tails.



(a) Isotropic α-stable.

(b) Unbalanced α-stable mixture.

(c) HRRR East, wet hours.

(d) Wildfire size.

Figure 1: Coordinate TCE versus reference quantile. Open markers denote zero networks; gray is test-test. High values are clipped where marked. Tail ranges follow Table 1.

Finally, examining global distributional fidelity through Figure 2, we observe that zero-network baselines fail to capture the underlying data structure. This further emphasizes that favorable tail calibration alone does not guarantee successful learning of the target distribution. Indeed, neither TCE nor MMD^2 fully captures the extent to which the generative model has learned the relevant structure. We observe comparable, if not better, recovery of the bulk structure by the LT models. Notably, linear Gaussian flow matching stands out for its strong performance relative to both HT and other LT alternatives.

3.2 Rare-Class Recovery on CIFAR100-LT

Heavy-tailed noise has also been proposed to improve robustness to class imbalance and mitigate mode collapse in generative modeling (Shariatian et al., 2025). We therefore complement the numerical heavy-tail benchmarks with an image-generation experiment assessing minority-class recovery. We construct CIFAR100-LT by subsampling the CIFAR-100 training images with exponentially decreasing class sizes and an imbalance factor of 10: the largest class contains ten times as many images as the smallest. Detailed preprocessing and evaluation settings appear in Appendix F.

In Table 2, DDPM-V achieves the highest rare-label recovery, while DDIM-V has the lowest label TV. DLPM and t-EDM obtain substantially lower quality-filtered rare-label recovery and larger label TV, indicating poorer recovery of minority-class mass under this evaluation protocol. Class-frequency recovery alone does not establish perceptual image quality or diagnose mode collapse. Nevertheless,

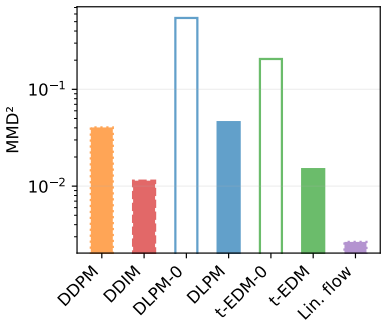


Figure 2: MMD^2 on HRRR East, on a log scale; lower is better. "-0" denotes a zero-network control.

Model	Rare ↑ (%)	TV ↓ (pp)
Reference target	100.0	0.0
DDPM-V	**56.3**	59.4
DDIM-V	46.5	**56.2**
DLPM	11.3	82.4
t-EDM	5.6	80.1
Linear Gaussian flow	21.1	64.6

Table 2: Rare recovery uses validated, quality-filtered rare labels relative to accepted real-test rare mass. TV compares predicted class frequencies.

this experiment highlights that minority-class recovery and numerical tail calibration capture distinct aspects of generative coverage.

Overall, our empirical results suggest that both DLPM and t-EDM struggle to learn the underlying data distribution, particularly in high-dimensional, finite-sample regimes, despite relying on HT noise that better matches the target tails. Neither model consistently outperforms LT alternatives, consistent with our theoretical insight that improved tail alignment does not necessarily translate into better generation when statistical learning remains the main bottleneck. Additional results, including sample-size comparisons and a comparison with the released DLPM implementation, are presented in Appendix G.

4 Conclusion

In this paper, we revisited heavy-tailed (HT) noising in diffusion-based generative modeling, examining whether matching the tail behavior of the noising process to that of the target distribution improves generation. Our theoretical analysis decomposes the sampling error into initialization and training contributions, further distinguishing approximation and finite-sample estimation effects. This reveals a *subtle trade-off*: compared with Gaussian VE diffusion, DLPM can reduce initialization error at an exponential rate, but may increase the difficulty of learning the reverse dynamics. In particular, approximation and estimation errors can accumulate along the reverse process, potentially offsetting the initialization gain. Our empirical benchmark supports this interpretation: across synthetic and real-world HT datasets, and using metrics assessing both distributional fidelity and tail calibration, HT noising does not consistently improve generative performance. In several regimes, light-tailed (LT) alternatives remain competitive or outperform DLPM.

Together, these findings suggest that **better tail alignment does not necessarily translate into better generation, particularly when training data are limited**. Indeed, a **central challenge in modeling extreme events is the scarcity of representative tail observations**. This motivates two complementary research directions, depending on whether additional data can be collected.

Improving training when additional data are available: Even with more observations, HT distributions pose optimization challenges due to potentially infinite moments. Promising directions include robust losses and estimators, gradient clipping, and data transformations for HT models, as well as tail-aware regularization and adaptive noise schedules for LT models. However, improved training alone cannot compensate for the absence of representative tail observations.

Correcting generation when additional data are unavailable: When tail observations are scarce, accurate generation in extreme regions requires additional prior information or structural assumptions. A promising direction is inference-time correction of pretrained models through initialization, sampling schemes, or *corrective guidance of the reverse dynamics*. Our analysis suggests that LT models offer a suitable starting point, combining favorable learning properties with guidance toward underrepresented regions. Although its effectiveness remains unproven, this strategy could improve tail coverage while preserving bulk fidelity without retraining the model.

Acknowledgments and Disclosure of Funding

We warmly thank Sylvain Le Corff and Gabriel Victorino Cardoso for fruitful discussions, insightful comments, and valuable suggestions that helped improve this work.

The work of Hamza Cherkaoui and Hélène Halconruy was supported by the French National Research Agency (ANR) under grant ANR-24-CE40-3341 (project DECATTLON). The work of Antonio Ocello was supported by Hi! PARIS and received government funding managed by the Agence Nationale de la Recherche under the France 2030 program, references (ANR-23-IACL-0005) and it was partially supported by the Labex Ecodec (reference project ANR-11-LABEX-0047). This work was granted access to the HPC resources of IDRIS under the allocation 2025-AD011016818, provided by GENCI.

References

Michael Albergo, Nicholas M Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *Journal of Machine Learning Research*, 26(209):1–80, 2025.

Sam Allen, Jonathan Koh, Johan Segers, and Johanna Ziegel. Tail calibration of probabilistic forecasts. *Journal of the American Statistical Association*, 2025.

Brian DO Anderson. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Adrian Baule. Generative modelling with jump-diffusions, 2026.

Lisa Bedin, Yazid Janati, Gabriel Victorino Cardoso, Josselin Duchateau, Rémi Dubois, and Eric Moulines. Reconstructing ecg from indirect signals: a denoising diffusion approach. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2299):20240330, 06 2025.

Nicholas M Boffi and Eric Vanden-Eijnden. Probability flow solution of the fokker–planck equation. *Machine Learning: Science and Technology*, 4(3):035012, 2023.

Patrick Cattiaux, Giovanni Conforti, Ivan Gentil, and Christian Léonard. Time reversal of diffusion processes under a finite entropy condition. *Annales de l'Institut Henri Poincaré (B) Probabilités et Statistiques*, 59(4):1844–1881, 2023.

Sitan Chen, Sinho Chewi, Jerry Li, Yuanzhi Li, Adil Salim, and Anru Zhang. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. In *The Eleventh International Conference on Learning Representations*, 2023.

Giovanni Conforti, Alain Durmus, and Marta Gentiloni Silveri. KL convergence guarantees for score diffusion models under minimal data assumptions, 2023.

Guillaume Couairon, Renu Singh, Anastase Charantonis, Christian Lessig, and Claire Monteleoni. Archesweathergen: Skillful and compute-efficient probabilistic weather forecasting with machine learning. *Science Advances*, 12(17):eadx2372, 2026.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in neural information processing systems*, 34:17695–17709, 2021.

David C. Dowell, Curtis R. Alexander, Eric P. James, Stephen S. Weygandt, Stanley G. Benjamin, Geoffrey S. Manikin, Benjamin T. Blake, John M. Brown, Joseph B. Olson, Ming Hu, Tatiana G. Smirnova, Terra Ladwig, Jaymes S. Kenyon, Ravan Ahmadov, David D. Turner, Jeffrey D. Duda, and Trevor I. Alcott. The high-resolution rapid refresh (HRRR): An hourly updating convection-allowing forecast model. part I: Motivation and system description. *Weather and Forecasting*, 37(8):1371–1395, 2022.

dynamical.org. NOAA HRRR analysis. https://dynamical.org/catalog/noaa-hrrr-analysis/, 2026. NOAA NWS NCEP data archive, accessed November 2026.

Bradley Efron. Tweedie's formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.

Tiziano Fassina, Gabriel Cardoso, Sylvan Le Corff, and Thomas Romary. Initialization-aware score-based diffusion sampling, 2026.

Gerald B Folland. *Real analysis: modern techniques and their applications*. John Wiley & Sons, 1999.

Xuefeng Gao, Hoang M Nguyen, and Lingjiong Zhu. Wasserstein convergence guarantees for a general class of score-based generative models. *Journal of machine learning research*, 26(43): 1–54, 2025.

Marta Gentiloni Silveri and Antonio Ocello. Beyond log-concavity and score regularity: Improved convergence bounds for score-based generative models in w2-distance, 2025.

Marta Gentiloni Silveri, Giovanni Conforti, and Alain Durmus. Theoretical guarantees in kl for diffusion flow matching. *Advances in Neural Information Processing Systems*, 37:138432–138473, 2024.

Boris Vladimirovich Gnedenko and Andreĭ Nikolaevich Kolmogorov. *Limit distributions for sums of independent random variables*, volume 2420. Addison-wesley, 1968.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Yunrui Guan, Krishnakumar Balasubramanian, and Shiqian Ma. Mirror flow matching with heavy-tailed priors for generative modeling on convex domains, 2025.

Zhengyi Guo, Jiatu Li, Wenpin Tang, and David D Yao. Diffusion generative models meet compressed sensing, with applications to imaging and finance. *arXiv preprint arXiv:2509.03898*, 2025.

U. G. Haussmann and E. Pardoux. Time reversal of diffusions. *The Annals of Probability*, 14(4): 1188–1205, 1986.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020.

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

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35: 26565–26577, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

David Landry, Claire Monteleoni, and Anastase Charantonis. Generating ensembles of spatially coherent in situ forecasts using flow matching. *Quarterly Journal of the Royal Meteorological Society*, 152(775):e70055, 2026.

Holden Lee, Jianfeng Lu, and Yixin Tan. Convergence for score-based generative modeling with polynomial complexity. *Advances in Neural Information Processing Systems*, 35:22870–22882, 2022.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *International Conference on Learning Representations*, 2023.

John P. Nolan. *Univariate Stable Distributions*. Springer Series in Operations Research and Financial Engineering. Springer, 2020.

Kushagra Pandey, Jaideep Pathak, Yilun Xu, Stephan Mandt, Michael Pritchard, Arash Vahdat, and Morteza Mardani. Heavy-tailed diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Ken-Iti Sato. *Lévy Processes and Infinitely Divisible Distributions*, volume 68 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, Cambridge, 1999.

Timur Sattarov, Marco Schreyer, and Damian Borth. Findiff: Diffusion models for financial tabular data generation. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pages 64–72, 2023.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875–1897, 2020.

Dario Shariatian, Umut Simsekli, and Alain Durmus. Heavy-tailed diffusion with denoising lévy probabilistic models. In *ICLR 2025-International Conference on Learning Representations*, 2025.

Karen C. Short. Spatial wildfire occurrence data for the united states, 1992–2024 [FPA_FOD_20260615]. Forest Service Research Data Archive, 2026. 7th Edition.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. pmlr, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020.

Arthur Stéphanovitch. Lipschitz regularity in flow matching and diffusion models: sharp sampling rates and functional inequalities. *arXiv preprint arXiv:2604.06065*, 2026.

Arthur Stéphanovitch, Eddie Aamari, and Clément Levrard. Generalization bounds for score-based generative models: a synthetic proof. *arXiv preprint arXiv:2507.04794*, 2025.

Stanislas Strasman, Antonio Ocello, Claire Boyer, Sylvain Le Corff, and Vincent Lemaire. An analysis of the noise schedule for score-based generative models. *Transactions on Machine Learning Research*, 2025a.

Stanislas Strasman, Sobihan Surendran, Claire Boyer, Sylvain Le Corff, Vincent Lemaire, and Antonio Ocello. Wasserstein convergence of critically damped langevin diffusions. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025b.

Alexandre B Tsybakov. Nonparametric estimators. In *Introduction to Nonparametric Estimation*, pages 1–76. Springer, 2008.

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

Yifeng Yu and Lu Yu. Diffusion models with heavy-tailed targets: Score estimation and sampling guarantees, 2026.

Table of contents

Appendix A	Notations	14
Appendix B	Related work	15
Appendix C	Proofs of Theorem 2.1, Theorem 2.2 and Proposition H.1	16
Appendix D	Useful technical results	21
Appendix E	Metrics	22
Appendix F	Experimental Details	23
Appendix G	Additional Empirical Results	25
Appendix H	Additional Remarks and Discussion	27

A Notations

Symbol	Description
$\mathbf{1}(\cdot)$	Indicator function.
I_d	Identity matrix of size $d \times d$.
$a \lesssim b$	There exists a universal constant $c > 0$ *s.t.*, $a \le cb$.
$h \asymp g$	There exist universal constants $c_1, c_2 > 0$ *s.t.*, $c_1 g \le h \le c_2 g$.
$f = O(g)$	There exists a constant $c > 0$ *s.t.*, $f \le cg$ for sufficiently large problem size.
$f = \widetilde{O}(g)$	Same as $O(g)$ up to polylogarithmic factors.
$\mathrm{polylog}(n)$	polylogarithmic factor in n, $\mathrm{polylog}(n) = (\log n)^k$, $k > 0$.
$\mathbb{R}^d$	Euclidean space of dimension d.
$\|\cdot\|$	Euclidean norm.
$\nabla \log p$	Score function of a density p.
$p * q$	Convolution of two densities.
$\mathcal{L}(X)$	Law of the random variable X.
$\mathbb{E}[\cdot]$, $\mathbb{P}(\cdot)$	Expectation and probability.
$\mathrm{TV}(\mu, \nu)$	Total variation distance between probability measures μ and ν.
$\mathrm{KL}(\mu \,\|\, \nu)$	Kullback–Leibler divergence between μ and ν.
$W^{\beta,2}(\mathbb{R}^d)$	Sobolev space of order β with L^2 integrability.
$\mathcal{C}^\beta(\mathbb{R}^d; \mathbb{R}^d)$	Hölder space of regularity β for vector-valued functions.
$\widehat{\cdot}$	Empirical or estimated quantity.
π_{data}, $\mu_\star$	Target data distribution.
$\gamma > 1$	Tail index of the data distribution.
$\alpha \in (0, 2]$	Tail index of the forward α-stable Lévy noise.
$S_\gamma^{\mathbf{i}}(\mu, \sigma I_d)$	Isotropic γ-stable distribution with location μ and scale σ.
β, L	Regularity exponent and associated constant.
T	Diffusion horizon.
$t_0 \in (0, T)$	Early-stopping time used in the DDPM analysis.
$X_t = X_0 + \sqrt{t}\, Z$	Gaussian forward noising process, with $Z \sim \mathcal{N}(0, I_d)$.
ϕ_t	Density of $\mathcal{N}(0, tI_d)$.
$p_t = \pi_{\mathrm{data}} * \phi_t$	Marginal law of X_t at time t.
$\overleftarrow{X}_t$	Reverse-time diffusion process.
$\widetilde{B}_t$	Brownian motion driving the reverse SDE.
$s_\theta(x, t)$	Score network approximating $\nabla \log p_t(x)$.
$\widetilde{Y}$	Reverse process initialized at the true terminal law.
$\widehat{Y}$	Practical reverse process initialized from the prescribed prior and driven by s_θ.

Symbol	Description
$A = (A_1, \dots, A_T)$	Sequence of α-stable Lévy increments.
S_α	Isotropic α-stable random variable.
a_T, b_T	Schedule coefficients in $Y_T = a_T Y_0 + b_T S_\alpha$.
$\rho_T = a_T / b_T$	Terminal signal-to-noise ratio.
$k^{(\alpha)}_{0:T\|a}$, K_a	Conditional and marginal forward path laws.
$\overleftarrow{q}^{\theta}_{0:T\|a}$, Q^{θ}_a	Conditional and marginal learned reverse path laws.
$\overleftarrow{q}^{\theta}_T$	Terminal prior of the learned reverse chain.
$s^\star_\alpha(t, \cdot)$	True reverse object targeted by the network.
$Y_0, \dots, Y_T$	State variables of the DLPM forward chain.
$Z = (A, Y_{0:T})$	Generic training variable.
Θ, Θ_m	Parameter space and ReLU-network class with m parameters.
$\theta^\star_{\alpha,r}$	Population minimizer of $\mathcal{L}^{\mathrm{TV}}_\alpha$.
$\widehat{\theta}_n$	Empirical minimizer of $\widehat{\mathcal{L}}^{\mathrm{TV}}_{n,\alpha}$.
$\ell^{\mathrm{TV}}_\theta(Z)$	Pointwise TV loss summed over the reverse chain.
$\mathcal{L}^{\mathrm{TV},\alpha}_{t-1}(\theta, a)$	Local TV loss at step t conditionally on $A = a$.
$\mathcal{L}^{\mathrm{TV}}_\alpha(\theta)$	Population TV risk.
$\widehat{\mathcal{L}}^{\mathrm{TV}}_{n,\alpha}(\theta)$	Empirical TV risk over n samples.
$\mathcal{F}^{\mathrm{TV}}$, $\mathcal{F}^{\mathrm{TV}}_t$	Global and local TV-loss classes.
$\mathfrak{R}_n(\mathcal{F})$	Empirical Rademacher complexity of $\mathcal{F}$.
ε_i	I.i.d. Rademacher random variables.
Comp	Generic complexity parameter of the model class.
$\delta \in (0, 1)$	Failure probability.
$\mathcal{E}_{\mathrm{init}}(T)$, $\mathcal{E}^{\mathrm{TV}}_{\mathrm{init}}(T, \alpha)$	Initialization error.
$\mathcal{E}_{\mathrm{app}}(t_0)$, $\mathcal{E}^{\mathrm{TV}}_{\mathrm{app}}(\alpha)$	Approximation error.
$\mathcal{E}_{\mathrm{est}}(n, t_0)$, $\mathcal{E}^{\mathrm{TV}}_{\mathrm{est}}(n, \alpha)$	Estimation error.

B Related work

Heavy-tail generative models. Several recent works propose to replace the Gaussian ingredients underlying diffusion and flow-based generative methods with heavier-tailed alternatives. In particular, Pandey et al. (2025) introduce t-EDM and t-Flow, extending EDM (Karras et al., 2022) and flow-matching methods (Lipman et al., 2023; Liu et al., 2023), by replacing Gaussian priors with multivariate Student-t distributions. Guan et al. (2025) combine mirror flow matching with Student-t priors to stabilize constrained generation on convex domains. More generally, Baule (2026) extend score-based diffusion to jump-diffusion forward processes driven by Gaussian noise with additional Poisson jumps. Together, these works follow the same principle: improving tail modeling by replacing Gaussian mechanisms with heavier-tailed counterparts. Our work revisits this paradigm from a statistical estimation perspective.

In a complementary direction, Denoising Lévy Probabilistic Models (DLPM) extend the Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020) construction by replacing Gaussian increments in the forward Markov chain with isotropic α-stable noise at each step (Shariatian et al., 2025). By exploiting the stability of α-stable distributions under addition, the forward chain admits an α-stable stationary distribution. The authors then derive a reverse-time Markov chain using detailed balance and adapt the DDPM training machinery through a conditioning argument and a reparameterization trick, leading to a modified denoising objective conditioned on the HT increment injected at each time step.

More recently, several works have explored an alternative paradigm that preserves the standard Gaussian noising process while adapting other components of the generative pipeline to HT settings. In particular, Yu and Yu (2026) study the statistical limits of Gaussian score-based diffusion models for HT target distributions, deriving score-estimation and sampling rates that explicitly depend on the tail behavior of the data. In a different direction, Fassina et al. (2026) analyze the role of the backward-process initialization and propose a learned initialization strategy that directly minimizes the initialization error without modifying the diffusion dynamics itself. Together, these works suggest

that improving HT generation may not necessarily require changing the noising process, but rather adapting the estimation and initialization procedures underlying diffusion sampling.

C Proofs of Theorem 2.1, Theorem 2.2 and Proposition H.1

Theoretical guarantees. Theoretical guarantees for score-based generative models have developed along several complementary directions. A first line of work studies convergence in strong distributional metrics such as α-divergences, Total Variation (TV), and Kullback–Leibler (KL) divergence, typically by controlling the error accumulated along the reverse-time dynamics (see, *e.g.*, De Bortoli et al., 2021; Lee et al., 2022; Chen et al., 2023; Gentiloni Silveri et al., 2024; Conforti et al., 2023; Stéphanovitch, 2026). Early results established TV guarantees under smoothness assumptions on the score, while more recent analyses obtained KL bounds under milder assumptions on the data distribution; through Pinsker's inequality, such bounds also imply convergence in TV.

A second line of work focuses on Wasserstein distances, which are particularly natural for sampling and estimation problems. Existing results provide $\mathcal{W}_1$ or $\mathcal{W}_2$ guarantees under assumptions such as manifold structure, bounded support, score regularity, or log-concavity, with recent refinements further clarifying the role of Lipschitz regularity and generalization in Wasserstein convergence (Gao et al., 2025; Strasman et al., 2025a,b; Gentiloni Silveri and Ocello, 2025; Stéphanovitch et al., 2025).

Our goal is to focus on the statistical *difficulty of the score estimation task* induced by the chosen noising process. For this reason, we work directly in TV distance, rather than deriving TV guarantees indirectly from KL divergence through Pinsker's inequality. This choice is particularly natural in the HT setting considered here: when the reference or noising distribution is heavy-tailed, KL-based comparisons with Gaussian counterparts may become infinite or fail to capture the relevant discrepancy. On the other hand, TV bounds remain well-defined and directly interpretable at the distributional level. Our analysis is therefore complementary to Fassina et al. (2026) as both the metric and the resulting error decomposition differ.

C.1 Gaussian VE diffusion (VE-GD)

Let now π_{data} satisfy Assumption 2.1. The corresponding light-tailed diffusion framework based on Gaussian noising processes is presented in subsection 2.1.

Error decomposition. Because $p_T = \pi_{\mathrm{data}} * \phi_T$ remains heavy-tailed, the KL divergence with a Gaussian prior may be infinite (this is the case when $\gamma \leq 2$, since p_T then has no finite second moment).

It is therefore natural to measure the error in total variation. For an early stopping time $t_0 \in (0, T)$, define $p_{t_0} = \pi_{\mathrm{data}} * \phi_{t_0}$. Let $\widetilde{Y}$ denote the reverse process initialized from p_T, and $\widehat{Y}$ the practical reverse process initialized from $\phi_T = \mathcal{N}(0, TI_d)$ and driven by s_θ. Then

$$\mathrm{TV}\big(\pi_{\mathrm{data}}, \mathcal{L}(\widehat{Y}_{T-t_0})\big) \leq \underbrace{\mathrm{TV}(\pi_{\mathrm{data}}, p_{t_0})}_{\mathcal{E}_{\mathrm{app}}(t_0)} + \underbrace{\mathrm{TV}\big(p_{t_0}, \mathcal{L}(\widetilde{Y}_{T-t_0})\big)}_{\mathcal{E}_{\mathrm{est}}(n,t_0)} + \underbrace{\mathrm{TV}\big(p_T, \phi_T\big)}_{\mathcal{E}_{\mathrm{init}}(T)} .$$

Approximation error $\mathcal{E}_{\mathrm{app}}(t_0)$. The term $\mathcal{E}_{\mathrm{app}}(t_0)$ quantifies the bias introduced by Gaussian smoothing. It reflects the discrepancy between the original density and its regularized version p_{t_0}. Under Sobolev regularity and polynomial tails, Yu and Yu (2026) show that

$$\mathcal{E}_{\mathrm{app}}(t_0) \lesssim t_0^{\frac{\beta(\gamma+1)}{d+2(\gamma+1)+2\beta}} .$$

This rate arises from a balance between local smoothing (controlled by β) and tail mass (controlled by γ).

Estimation error $\mathcal{E}_{\mathrm{est}}(n, t_0)$. The term $\mathcal{E}_{\mathrm{est}}(n, t_0)$ captures the error due to score estimation. It measures how well the learned score s_θ approximates $\nabla \log p_t$ along the reverse trajectory. It can be bounded in terms of the cumulative $\mathbb{L}^2(p_t)$-error:

$$\mathcal{E}_{\mathrm{est}}(n, t_0) \lesssim \left(\int_{t_0}^{T} \mathbb{E}\left[\int \|s_\theta(x,t) - \nabla \log p_t(x)\|^2 p_t(x)\, \mathrm{d}x \right] \mathrm{d}t \right)^{1/2} .$$

For heavy-tailed targets, this leads to the nonparametric rate (see Yu and Yu (2026), Theorem 2)

$$\mathcal{E}_{\mathrm{est}}(n, t_0) \lesssim \mathrm{polylog}(n)\, n^{-\frac{\gamma+1}{2(d+\gamma+1)}}\, t_0^{-\frac{d(\gamma+1)}{4(d+\gamma+1)}}\,.$$

Initialization error $\mathcal{E}_{\mathrm{init}}(T)$. The initialization error $\mathcal{E}_{\mathrm{init}}(T)$ reflects the mismatch between the true terminal distribution p_T and the Gaussian prior used in practice. Since $\gamma > 1$, the first moment is finite, and one obtains

$$\mathcal{E}_{\mathrm{init}}(T) \lesssim T^{-1/2}\,.$$

This term decreases with the diffusion horizon and is independent of the learning procedure.

Final bound. Combining the three contributions, we obtain

$$\mathrm{TV}\big(\pi_{\mathrm{data}}, \mathcal{L}(\widehat{Y}_{T-t_0})\big) \lesssim t_0^{\frac{\beta(\gamma+1)}{d+2(\gamma+1)+2\beta}} + \mathrm{polylog}(n)\, n^{-\frac{\gamma+1}{2(d+\gamma+1)}}\, t_0^{-\frac{d(\gamma+1)}{4(d+\gamma+1)}} + T^{-1/2}\,.$$

C.2 DLPM

The corresponding heavy-tailed generative setting based on α-stable noising processes is introduced in subsection 2.2.

Decomposition of the error. Since $k^{(\alpha)}_{0|a}$ and $\overleftarrow{q}^{\theta}_{0|a}$ are the Y_0-marginals of K_a and Q^{θ}_a, respectively, the data-processing inequality for total variation yields

$$\mathrm{TV}\big(k^{(\alpha)}_{0|a}, \overleftarrow{q}^{\theta}_{0|a}\big) \le \mathrm{TV}(K_a, Q^{\theta}_a). \tag{7}$$

Using contraction of total variation under Markov kernels and a telescoping argument for inhomogeneous chains, we obtain

$$\begin{aligned}
\mathrm{TV}(K_a, Q^{\theta}_a) &\le \mathrm{TV}\big(k^{(\alpha)}_{T|a}, \overleftarrow{q}^{\theta}_{T}\big) \\
&\quad + \sum_{t=1}^{T} \mathbb{E}_{K_a}\Big[\mathrm{TV}\big(k^{(\alpha)}_{t-1|t,a}(\cdot|Y_t, a), \overleftarrow{q}^{\theta}_{t-1|t,a}(\cdot|Y_t, a)\big)\Big] \\
&= \mathrm{TV}\big(k^{(\alpha)}_{T|a}, \overleftarrow{q}^{\theta}_{T}\big) + \sum_{t=1}^{T} \mathcal{L}^{\mathrm{TV},\alpha}_{t-1}(\theta, a)
\end{aligned} \tag{8}$$

where for each $t \in \{1, \dots, T\}$ and $a \in \mathbb{R}^T$, we define the local TV loss

$$\mathcal{L}^{\mathrm{TV},\alpha}_{t-1}(\theta, a) := \mathbb{E}_{K_a}\Big[\mathrm{TV}\big(k^{(\alpha)}_{t-1|t,a}(\cdot|Y_t, a), \overleftarrow{q}^{\theta}_{t-1|t,a}(\cdot|Y_t, a)\big)\Big]\,,$$

and the expectation is taken with respect to the true path law K_a. The population TV risk is then defined as

$$\mathcal{L}^{\mathrm{TV}}_{\alpha}(\theta) := \sum_{t=1}^{T} \mathbb{E}_A\Big[\mathcal{L}^{\mathrm{TV},\alpha}_{t-1}(\theta, A)\Big]\,.$$

Equivalently, introducing a generic data variable $Z := (A, Y_{0:T}) \sim K_A \otimes \mathbb{P}_A$. We can write

$$\mathcal{L}^{\mathrm{TV}}_{\alpha}(\theta) = \mathbb{E}\big[\ell^{\mathrm{TV}}_{\theta}(Z)\big], \qquad \ell^{\mathrm{TV}}_{\theta}(Z) := \sum_{t=1}^{T} \mathrm{TV}\big(k^{(\alpha)}_{t-1|t,A}(\cdot|Y_t, A), \overleftarrow{q}^{\theta}_{t-1|t,A}(\cdot|Y_t, A)\big)\,.$$

Given i.i.d. samples $Z_1, \dots, Z_n$, the empirical TV risk is

$$\widehat{\mathcal{L}}^{\mathrm{TV}}_{n,\alpha}(\theta) := \frac{1}{n}\sum_{i=1}^{n} \ell^{\mathrm{TV}}_{\theta}(Z_i)\,.$$

Their respective minimizers are given by

$$\theta^{\star}_{\alpha,r} \in \arg\min_{\theta\in\Theta} \mathcal{L}^{\mathrm{TV}}_{\alpha}(\theta) \quad \text{and} \quad \widehat{\theta}_n \in \arg\min_{\theta\in\Theta} \widehat{\mathcal{L}}^{\mathrm{TV}}_{n,\alpha}\,.$$

Then, plugging $\theta = \widehat{\theta}_n$ into the previous inequalities in Equation 7 and Equation 8 gives

$$\mathbb{E}_A\left[\mathrm{TV}\big(k^{(\alpha)}_{0|A}, \overleftarrow{q}^{\widehat{\theta}_n}_{0|A}\big)\right] \le \mathbb{E}_A\left[\mathrm{TV}(K_A, Q_A^{\widehat{\theta}_n})\right]$$
$$\le \underbrace{\mathbb{E}_A\big[\mathrm{TV}\big(k^{(\alpha)}_{T|A}, \overleftarrow{q}^{\widehat{\theta}_n}_{T}\big)\big]}_{\mathcal{E}^{\mathrm{TV}}_{\mathrm{init}}(T,\alpha)} + \underbrace{\mathcal{L}^{\mathrm{TV}}_\alpha(\theta^\star_{\alpha,r})}_{\mathcal{E}^{\mathrm{TV}}_{\mathrm{app}}(\alpha)} + \underbrace{\left[\mathcal{L}^{\mathrm{TV}}_\alpha(\widehat{\theta}_n) - \mathcal{L}^{\mathrm{TV}}_\alpha(\theta^\star_{\alpha,r})\right]}_{\mathcal{E}^{\mathrm{TV}}_{\mathrm{est}}(n,\alpha)}.$$

Approximation error $\mathcal{E}^{\mathrm{TV}}_{\mathrm{app}}(\alpha)$. In the DLPM setting, the forward dynamics involve convolution with an α-stable law. For any positive time $t > 0$, this induces a smoothing effect: the resulting density belongs to fractional Hölder (or Sobolev) spaces. Since the reverse object $s^\star_\alpha$ is constructed from this regularized density (e.g. via derivatives or score-type expressions), it inherits a positive degree of Hölder regularity. This motivates the following assumption.

Let Θ_m be a class of ReLU neural networks with m parameters and sufficient depth. A standard approximation result for ReLU neural networks states that for any $f \in \mathcal{C}^\beta([0,1]^d)$, there exists a network f_m with m parameters such that

$$\|f - f_m\|_{\mathbb{L}^\infty([0,1]^d)} \lesssim m^{-\beta/d},$$

up to logarithmic factors (see Schmidt-Hieber (2020)). Applying this result componentwise to $s^\star_\alpha(t,\cdot)$ yields a network s_θ such that

$$\sup_{t\in[0,T]} \|s_\theta(t,\cdot) - s^\star_\alpha(t,\cdot)\|_{\mathbb{L}^\infty} \lesssim m^{-\beta(\alpha)/d}.$$

Therefore, for any distribution of (Y_t, A),

$$\mathbb{E}\big[\|s_\theta(t,Y_t,A) - s^\star_\alpha(t,Y_t,Y_0,A)\|^2\big] \le \|s_\theta - s^\star_\alpha\|^2_{\mathbb{L}^\infty} \lesssim m^{-2\beta(\alpha)/d}.$$

Then,

$$\inf_{\theta\in\Theta_m} \left(\mathbb{E}\|s_\theta(t,Y_t,A) - s^\star_\alpha(t,Y_t,Y_0,A)\|^2\right)^{1/2} \lesssim m^{-\beta(\alpha)/d},$$

up to logarithmic factors. The implicit constant depends on $d, \beta(\alpha)$ and $C_{\alpha,\beta}$, but not on m.

Estimation error $\mathcal{E}^{\mathrm{TV}}_{\mathrm{est}}(n,\alpha)$. Define the loss class

$$\mathcal{F}^{\mathrm{TV}} = \{\ell^{\mathrm{TV}}_\theta : \theta \in \Theta\}.$$

Since the TV loss is uniformly bounded, the class $\mathcal{F}^{\mathrm{TV}}$ is bounded as well. Applying the Rademacher complexity bound of Theorem D.1, we obtain that, with probability at least $1-\delta$,

$$\mathcal{E}^{\mathrm{TV}}_{\mathrm{est}}(n,\alpha) \lesssim \mathfrak{R}_n(\mathcal{F}^{\mathrm{TV}}) + T\sqrt{\frac{\log(1/\delta)}{n}}.$$

To control the Rademacher complexity, we use

$$\ell^{\mathrm{TV}}_\theta(Z) = \sum_{t=1}^T \ell^{\mathrm{TV}}_{t,\theta}(Z), \quad \text{with} \quad \ell^{\mathrm{TV}}_{t,\theta}(Z) := \mathrm{TV}\left(k^{(\alpha)}_{t-1|t,A}(\cdot \mid Y_t, A), \overleftarrow{q}^{\theta}_{t-1|t,A}(\cdot \mid Y_t, A)\right).$$

By subadditivity of Rademacher complexity,

$$\mathfrak{R}_n(\mathcal{F}^{\mathrm{TV}}) \le \sum_{t=1}^T \mathfrak{R}_n(\mathcal{F}^{\mathrm{TV}}_t),$$

where $\mathcal{F}^{\mathrm{TV}}_t := \{\ell^{\mathrm{TV}}_{t,\theta} : \theta \in \Theta\}$. Assume that each class $\mathcal{F}^{\mathrm{TV}}_t$ satisfies

$$\mathfrak{R}_n(\mathcal{F}^{\mathrm{TV}}_t) \lesssim \sqrt{\frac{\mathrm{Comp}}{n}},$$

for some complexity parameter Comp depending on the model class (e.g. number of parameters, norm constraints, or network depth).

Then

$$\mathfrak{R}_n(\mathcal{F}^{\mathrm{TV}}) \lesssim T\sqrt{\frac{\mathrm{Comp}}{n}},$$

and therefore

$$\mathcal{E}_{\mathrm{est}}^{\mathrm{TV}}(n,\alpha) \lesssim T\sqrt{\frac{\mathrm{Comp}}{n}} + T\sqrt{\frac{\log(1/\delta)}{n}}\,.$$

Combining terms yields

$$\mathcal{E}_{\mathrm{est}}^{\mathrm{TV}}(n,\alpha) \lesssim T\sqrt{\frac{\mathrm{Comp}+\log(1/\delta)}{n}}\,.$$

Initialization error $\mathcal{E}_{\mathrm{init}}^{\mathrm{TV}}(T,\alpha)$. We assume that the terminal prior is chosen as

$$\overleftarrow{q}_T^{\,\theta} = \mathcal{L}(b_T S_\alpha)\,,$$

which matches the dominant noise component of Y_T in Equation 5. Introducing $\rho_T := a_T/b_T$, we may rewrite

$$Y_T = b_T(\rho_T Y_0 + S_\alpha)\,.$$

Since total variation is invariant under invertible linear transformations, it follows that

$$\mathrm{TV}\big(\mathcal{L}(Y_T), \mathcal{L}(b_T S_\alpha)\big) = \mathrm{TV}\big(\mathcal{L}(\rho_T Y_0 + S_\alpha), \mathcal{L}(S_\alpha)\big).$$

Conditioning on Y_0 and using convexity of total variation yields

$$\mathcal{E}_{\mathrm{init}}^{\mathrm{TV}}(\alpha) = \mathbb{E}\left[\mathrm{TV}\left(p_\alpha(\cdot - \rho_T Y_0), p_\alpha\right)\right],$$

where p_α denotes the density of S_α.

To control this quantity, we use a first-order Taylor expansion. For any $y \in \mathbb{R}^d$,

$$p_\alpha(x-\rho y) - p_\alpha(x) = -\int_0^1 \rho\, y \cdot \nabla p_\alpha(x - s\rho y)\,\mathrm{d}s,$$

which implies

$$\mathrm{TV}\big(p_\alpha(\cdot - \rho y), p_\alpha\big) \le \frac{|\rho|\,\|y\|}{2}\int_{\mathbb{R}^d} \|\nabla p_\alpha(x)\|\,\mathrm{d}x.$$

The integrability condition

$$\int_{\mathbb{R}^d} \|\nabla p_\alpha(x)\|\,\mathrm{d}x < \infty$$

holds for α-stable densities (which are smooth with polynomially decaying derivatives; see Sato (1999)). Under Assumption 2.4, the initialization bound

$$\mathcal{E}_{\mathrm{init}}^{\mathrm{TV}}(\alpha) \lesssim |\rho_T|\,\mathbb{E}\|Y_0\|$$

implies, since $\mathbb{E}\|Y_0\| < \infty$ (Y_0 has γ-stable tails with $\gamma \in (1,2)$),

$$\mathcal{E}_{\mathrm{init}}^{\mathrm{TV}}(\alpha) = \mathrm{O}_{\alpha,\gamma}(e^{-cT})\,.$$

Final bound. Combining the initialization, approximation, and estimation bounds, we obtain, with probability at least $1-\delta$,

$$\mathbb{E}_A\left[\mathrm{TV}\left(k_{0|A}^{(\alpha)}, \overleftarrow{q}_{0|A}^{\,\widehat{\theta}_n}\right)\right] \lesssim e^{-cT} + T\,m^{-\beta(\alpha)/d} + T\sqrt{\frac{\mathrm{Comp}+\log(1/\delta)}{n}}.$$

C.3 Proof of Proposition H.1

Construction. Set $\Omega = \{0, 1, \dots, T\} \cup \{*\}$ and $\mu_T = \delta_T$. For each step $t \in \{1, \dots, T\}$, define the true kernel

$$Q_t(x, \cdot) = \begin{cases} \delta_{x-1} & \text{if } x \in \{1, \dots, T\}\,, \\ \delta_x & \text{if } x \in \{0, *\}\,. \end{cases}$$

Each Q_t decrements the state by one, the states 0 and $*$ being absorbing. Starting from $\mu_T = \delta_T$, the true chain moves deterministically along $T \to T-1 \to \cdots \to 0$, so that $\mu_T Q_T \cdots Q_1 = \delta_0$. In particular, the true chain is at state t exactly when the kernel of step t is applied. The learned kernels are obtained by introducing a single *error channel* at each step:

$$\widehat{Q}_t(x, \cdot) = \begin{cases} (1-\varepsilon)\,\delta_{t-1} + \varepsilon\,\delta_* & \text{if } x = t\,, \\ Q_t(x, \cdot) & \text{if } x \neq t\,. \end{cases}$$

The kernels $\widehat{Q}_t$ and Q_t coincide except at state $x = t$, where $\widehat{Q}_t$ sends a fraction ε of the mass to the absorbing state $*$. Hence

$$\sup_{x \in \Omega} \mathrm{TV}\big(Q_t(x, \cdot), \widehat{Q}_t(x, \cdot)\big) = \mathrm{TV}\big(\delta_{t-1}, (1-\varepsilon)\delta_{t-1} + \varepsilon\delta_*\big) = \varepsilon\,,$$

which proves the first claim.

Propagation of the error. We show by induction that, for every $k \in \{0, \dots, T\}$,

$$\mu_T \widehat{Q}_T \cdots \widehat{Q}_{T-k+1} = (1-\varepsilon)^k\,\delta_{T-k} + \big(1 - (1-\varepsilon)^k\big)\,\delta_*\,, \tag{9}$$

with the convention that the empty product is the identity. For $k = 0$, Equation 9 reduces to $\mu_T = \delta_T$. Assume that Equation 9 holds for some $k < T$ and apply $\widehat{Q}_{T-k}$. The mass located at $*$ remains there since $*$ is absorbing, while the mass $(1-\varepsilon)^k$ located at $T-k$ meets the error channel of step $T-k$ and is mapped to $(1-\varepsilon)\delta_{T-k-1} + \varepsilon\delta_*$. Therefore,

$$\begin{aligned} \mu_T \widehat{Q}_T \cdots \widehat{Q}_{T-k} &= (1-\varepsilon)^{k+1}\,\delta_{T-k-1} + \big(1 - (1-\varepsilon)^k + \varepsilon(1-\varepsilon)^k\big)\,\delta_* \\ &= (1-\varepsilon)^{k+1}\,\delta_{T-k-1} + \big(1 - (1-\varepsilon)^{k+1}\big)\,\delta_*\,, \end{aligned}$$

which completes the induction. Taking $k = T$ gives

$$\mu_T \widehat{Q}_T \cdots \widehat{Q}_1 = (1-\varepsilon)^T\,\delta_0 + \big(1 - (1-\varepsilon)^T\big)\,\delta_*\,.$$

Since δ_0 and δ_* are mutually singular,

$$\mathrm{TV}\big(\mu_T Q_T \cdots Q_1,\ \mu_T \widehat{Q}_T \cdots \widehat{Q}_1\big) = 1 - (1-\varepsilon)^T\,.$$

The error channels are disjoint, in the sense that each of them is activated exactly once along the true trajectory, and every discrepancy pushes mass in the same direction, towards $*$. The local errors therefore accumulate without cancellation.

Lower bound. Using $1 - \varepsilon \le e^{-\varepsilon}$, we get $1 - (1-\varepsilon)^T \ge 1 - e^{-T\varepsilon}$. If $T\varepsilon \le 1$, the concavity of $x \mapsto 1 - e^{-x}$ gives $1 - e^{-x} \ge (1 - e^{-1})x \ge x/2$ on $[0, 1]$, hence $1 - e^{-T\varepsilon} \ge T\varepsilon/2$. If $T\varepsilon > 1$, then $1 - e^{-T\varepsilon} > 1 - e^{-1} > 1/2$. In both cases,

$$1 - (1-\varepsilon)^T \ge \tfrac{1}{2}\min\{1, T\varepsilon\}\,.$$

Conversely, the telescoping argument of Equation 8 gives $1 - (1-\varepsilon)^T \le T\varepsilon$, so that the linear accumulation is sharp up to a universal constant. □

C.4 Comparison of the VE-GD and DLPM bounds.

We now compare the two bounds through a common decomposition into *initialization*, *approximation*, and *estimation* errors, in order to highlight the respective strengths and limitations of VE-GD and DLPM.

VE-GD bound.

$$\mathrm{TV}\big(\pi_{\mathrm{data}}, \mathcal{L}(\widehat{Y}_{T-t_0})\big) \lesssim \underbrace{T^{-1/2}}_{\text{initialization error}} + \underbrace{t_0^{\frac{\beta(\gamma+1)}{d+2(\gamma+1)+2\beta}} + \mathrm{polylog}(n)\, n^{-\frac{\gamma+1}{2(d+\gamma+1)}}\, t_0^{-\frac{d(\gamma+1)}{4(d+\gamma+1)}}}_{\text{training error}}\,.$$

DLPM bound.

$$\mathbb{E}_A\left[\mathrm{TV}\left(k_{0|A}^{(\alpha)}, \overleftarrow{q}_{0|A}^{\widehat{\theta}_n}\right)\right] \lesssim \underbrace{e^{-cT}}_{\text{initialization error}} + \underbrace{T\, m^{-\beta(\alpha)/d} + T\sqrt{\frac{\mathrm{Comp} + \log(1/\delta)}{n}}}_{\text{training error}}.$$

We now compare each source of error in turn.

Initialization. The initialization term measures how well the terminal distribution matches the chosen prior. In VE-GD, this term decays only polynomially as $T^{-1/2}$, because in the VE dynamics Equation 1 the initial condition is never rescaled: p_T is a mixture of $\mathcal{N}(x, TI_d)$, $x \sim \pi_{\mathrm{data}}$, whose discrepancy with ϕ_T only vanishes through the growth of the variance. In contrast, DLPM achieves an exponential decay e^{-cT}, because its noise schedule makes the weight of Y_0 in Y_T vanish exponentially relative to the injected noise (Assumption 2.4). This exponential rate is a substantial advantage of DLPM over the VE-GD LT model. Our comparison concerns these two representative formulations, Gaussian VE diffusion and stable-noise DLPM.

Training error (approximation + estimation). The training error can be decomposed into two components: an *approximation error*, which measures how well the model class can represent the reverse dynamics, and an *estimation error*, which reflects the statistical difficulty of learning from finite data. We first consider the approximation component. In DLPM, this error scales as $Tm^{-\beta(\alpha)/d}$, reflecting the accumulation of local approximation errors along the T reverse steps. In practice, T must be large to ensure that the forward process sufficiently mixes (so that the terminal distribution is close to the prior). As a consequence, even moderate local approximation errors can accumulate and dominate the bound.

By contrast, the VE-GD bound, established in continuous time, does not exhibit this accumulation. The approximation error is controlled through the early-stopping parameter t_0, which balances bias and variance, and does not scale with T.

A similar phenomenon occurs for the estimation component. In DLPM, the loss is a sum of local contributions along the reverse chain, which leads to an estimation error of the form $T\sqrt{(\mathrm{Comp} + \log(1/\delta))/n}$ reflecting the accumulation of statistical fluctuations across the T steps. Since each step involves learning a conditional distribution, the overall estimation error grows linearly with T. By contrast, the VE-GD bound does not exhibit such accumulation. The estimation error is controlled globally through the early-stopping parameter t_0, leading to a rate of the form $n^{-\frac{\gamma+1}{2(d+\gamma+1)}}$, without any multiplicative dependence on T. As a result, even though this rate may deteriorate with the tail index γ, it does not suffer from the same amplification mechanism , so that the corresponding bound does not grow with T.

Summary. Overall, while DLPM benefits from a significantly improved initialization, its *training error* (defined as the sum of approximation and estimation errors) scales linearly with T. Since T must be large to ensure good initialization, these errors accumulate along the reverse chain and can dominate the overall bound. As a result, even though DLPM enjoys a faster initialization rate than Gaussian VE diffusion, VE-GD may be preferable in practice when the training error is the main bottleneck.

D Useful technical results

We recall a standard uniform deviation bound for bounded function classes. Let $\mathcal{F}$ be a class of functions. Define the empirical Rademacher complexity

$$\mathfrak{R}_n(\mathcal{F}) = \mathbb{E}_{\varepsilon, Z}\left[\sup_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} \varepsilon_i f(Z_i)\right], \tag{10}$$

where $(\varepsilon_i)_{i=1}^n$ are i.i.d. Rademacher variables independent of the sample $(Z_i)_{i=1}^n$.

Theorem D.1 (Uniform deviation bound for bounded classes) Assume that the class $\mathcal{F}$ is uniformly bounded, namely there exists $B > 0$ such that

$$\sup_{f\in\mathcal{F}} \|f\|_\infty \le B\,.$$

Then, with probability at least $1-\delta$,

$$\sup_{f\in\mathcal{F}}\left|\mathbb{E}[f(Z)]-\frac{1}{n}\sum_{i=1}^{n}f(Z_i)\right|\le 2\,\mathfrak{R}_n(\mathcal{F})+CB\sqrt{\frac{\log(1/\delta)}{n}},$$

for some universal constant $C>0$.

E Metrics

We evaluate vector distributions in their original coordinates after undoing any training standardization. Lower scores are better unless stated otherwise. Main vector results are single-final-run estimates, not means or standard deviations over repeated final training runs.

E.1 Coordinate Tail Coverage

Let $x_1,\dots,x_n$ be reference observations and $y_1,\dots,y_m$ generated observations in $\mathbb{R}^d$. Define $g(x)=|x|$ coordinatewise for the synthetic benchmarks and $g(x)=x$ for HRRR and wildfire. For each reference quantile q and coordinate j, compute

$$u_j(q)=\widehat{Q}_q\big(\{g(x_i)_j\}_{i=1}^n\big),$$
$$r_j(q)=\frac{1}{n}\sum_{i=1}^{n}\mathbf{1}\{g(x_i)_j>u_j(q)\},\qquad s_j(q)=\frac{1}{m}\sum_{i=1}^{m}\mathbf{1}\{g(y_i)_j>u_j(q)\}.$$

The coordinate-TCE curve is

$$C(q)=\frac{1}{d}\sum_{j=1}^{d}\left|\log\frac{s_j(q)+\varepsilon}{r_j(q)+\varepsilon}\right|,\qquad \varepsilon=10^{-12}. \tag{11}$$

Thus reference and generated samples receive the same transformation. The denominator is the empirical reference exceedance mass, rather than exactly $1-q$. This targets tail calibration directly (Allen et al., 2025); an identical reference/generated sample has zero error.

Each curve uses 21 logarithmically spaced exceedance probabilities $p=1-q$, from 10^{-1} to $p_{\min}$. The minimum is 10^{-5} for the synthetic benchmarks, 10^{-3} for HRRR, and 10^{-4} for wildfire. The reference quantile axis is displayed on a logit scale. The test-test baseline compares two disjoint halves of the reference sample using the same metric; it is a finite-sample diagnostic, not a confidence interval. Displayed curves may clip high values for readability; the scores themselves are not clipped.

E.2 Squared RBF-Kernel MMD

We report the biased empirical squared maximum mean discrepancy (Gretton et al., 2012), including diagonal kernel terms:

$$\widehat{\mathrm{MMD}}^2=\frac{1}{n^2}\sum_{i,i'}k(x_i,x_{i'})+\frac{1}{m^2}\sum_{\ell,\ell'}k(y_\ell,y_{\ell'})-\frac{2}{nm}\sum_{i,\ell}k(x_i,y_\ell), \tag{12}$$

where $k(x,y)=\exp(-\gamma\|x-y\|_2^2)$. The bandwidth is fixed from reference pairs only, with γ equal to one half divided by the median positive squared reference distance, and is shared across methods within a dataset. MMD^2 uses 1000 observations per sample for synthetic and wildfire data, and 512 for HRRR. These subsets are smaller than the full TCE evaluation samples. MMD^2 assesses global distributional fit and complements the tail-specific curves; its bar plots use logarithmic vertical axes.

E.3 CIFAR Rare-Label Recovery and Label TV

A balanced ResNet-18 classifier, with a 3×3 input convolution and no initial max pooling, assigns generated images to CIFAR classes. Its training pool uses the original CIFAR training images while excluding the generative benchmark's validation and evaluation images. The official CIFAR test images are split into two alternating halves: one selects classifier weights and checks the final acceptance precision, while the other calibrates the acceptance rule. Classifier training uses SGD with

learning rate 0.1, momentum 0.9, weight decay 5×10^{-4}, cosine scheduling, batch size 256, and up to 80 epochs with patience 12. Training augmentation includes random crops and horizontal flips.

The rare set $\mathcal{R}$ contains the 20 least frequent classes in the generative training split, with class index breaking frequency ties. A rare prediction is accepted only if it meets a confidence threshold and a same-class feature-support threshold, using normalized classifier embeddings and a balanced real-image feature bank. Support thresholds use the fifth percentile of calibration similarity within each rare class. Confidence calibration targets at least 90% weighted precision among accepted rare predictions. Recovery is reported only when held-out rare-class accuracy is at least 60%, checked acceptance precision is at least 90%, and at least 30 real rare predictions are accepted.

If a_g and a_r are accepted rare counts among N_g generated and N_r real evaluation images, rare recovery is

$$\text{Recovery}(\%) = 100\,\frac{a_g/N_g}{a_r/N_r}. \tag{13}$$

It is a ratio of accepted rare-label masses, not classifier accuracy. If $\widehat{h}_g(c)$ and $\widehat{h}_r(c)$ are predicted class frequencies, label TV in percentage points is

$$\text{TV}(\text{pp}) = 50 \sum_{c=1}^{100} |\widehat{h}_g(c) - \widehat{h}_r(c)|. \tag{14}$$

The current evaluator has 75.1% overall and 81.7% rare-class accuracy on real evaluation images. Its confidence cutoff is approximately 0.986, checked acceptance precision is 90.2%, and 71 of 1958 real images receive accepted rare labels. These checks support label-based comparisons but do not establish overall perceptual image quality.

E.4 Tail-Quantile MSLE for the DLPM Comparison

The additional DLPM experiment uses the tail-quantile MSLE associated with the protocol of Shariatian et al. (2025). For absolute-coordinate quantile functions Q_j^X and Q_j^Y, we compute

$$\text{MSLE}_{0.95} = \frac{1}{d} \sum_{j=1}^{d} \int_{0.95}^{1} \left[\log Q_j^X(u) - \log Q_j^Y(u)\right]^2 \mathrm{d}u. \tag{15}$$

The integral uses 1000 midpoint quantile levels, with quantiles clipped below at 10^{-12} before taking logarithms. Because it operates on coordinate magnitudes, this definition handles signed data. It differs from pointwise squared differences of $\log(1+x)$ and is separate from the main TCE evaluation.

F Experimental Details

Experiments use single-GPU Slurm jobs with one NVIDIA V100 and eight CPU cores. Each main training job runs the four trained models sequentially. Epoch budgets below are maxima: early stopping can shorten training, and the best validation weights are restored.

F.1 Datasets and Preprocessing

Isotropic α-stable. We generate centered symmetric isotropic α-stable samples in $d = 150$, with $\alpha = 1.7$, using a Gaussian scale-mixture representation. The fixed seeded split contains $25{,}000$ training, $1{,}000$ validation, and $10{,}000$ test-loss samples. Distributional evaluation draws an independent reference sample of 5M observations and 5M generated observations per method. No standardization is applied.

Unbalanced α-stable mixture. This 150-dimensional target has 16 modes in a rank-6 latent subspace, power-law mode probabilities with imbalance parameter $\tau = 1.2$, mean scale 7.5, base scale 0.55, anisotropy 1.0, and stable local noise with $\alpha = 1.7$. The structure seed is zero. Splits, evaluation counts, and standardization follow the isotropic benchmark.

HRRR East and West. We use the NOAA HRRR analysis archive processed by dynamical.org (Dowell et al., 2022; dynamical.org, 2026). Each hourly vector contains precipitation at 150 fixed U.S. cities, selected east or west of longitude -100° and mapped to distinct nearest HRRR grid cells. Precipitation rates are converted to mm/hour. The requested period is October 6, 2014 through September 30, 2026; incomplete aligned hours are excluded. Both regions contain 103,018 complete hours. We remove 2,473 all-zero East vectors and 10,367 all-zero West vectors, leaving 100,545 and 92,651 wet hours. No positive threshold is imposed: tiny nonzero observations remain. A seeded random split reserves 20,000 training and 1,000 validation hours per region. All remaining wet hours are evaluated: 79,545 East and 71,651 West.

Wildfire size. We sample distinct positive-size U.S. fire records from the seventh edition of FPA-FOD, covering 1992–2024 (Short, 2026). Each observation is final fire size in acres. The seeded split contains 25,000 training, 1,000 validation, and 1M evaluation observations.

CIFAR100-LT. We construct a class-imbalanced subset of the CIFAR-100 training images, with imbalance factor 10 and image shape $3 \times 32 \times 32$ (Krizhevsky, 2009). Ten percent is reserved for each of validation and evaluation, yielding $15{,}657/1{,}958/1{,}958$ images. Pilots use a fixed seeded subset of 4096 training images; final training uses the full training split. Evaluation pairs all 1,958 real evaluation images with 1,958 generated images per method. The configured generation cap is 5000, but evaluation never exceeds the available real reference count. The separate classifier pools are described in subsection E.3.

HRRR and wildfire coordinates are standardized using training means and standard deviations only; zero standard deviations are replaced by one. The inverse transformation is applied before distributional evaluation. Synthetic coordinates remain unstandardized. CIFAR pixels are mapped to $[-1, 1]$. All main splits and final training runs use seed zero.

F.2 Models and Architectures

Table 4 lists DDPM-V (Ho et al., 2020), DDIM-V (Song et al., 2021), DLPM (Shariatian et al., 2025), t-EDM, Linear Gaussian flow (Lipman et al., 2023; Liu et al., 2023), and two zero-network controls. DDPM-V and DLPM use velocity (V) and noise prediction, respectively. DDIM-V and DDPM-V share the same model weights, follow the variance-preserving (VP) formulation. t-EDM uses the Student-t construction of Pandey et al. (2025) with EDM preconditioning (Karras et al., 2022). Linear Gaussian flow uses $x_t = (1-t)x_0 + tx_1$ and a learned velocity field. The zero controls set every network parameter to zero while retaining the sampler, schedule, and preconditioning; they are not simply independent samples drawn directly from the initial prior. DDIM uses the trained DDPM network, with $\eta = 0$, without separate training. Step counts do not imply identical numbers of network evaluations across solvers.

Model	Noise / path	Weights	Sampler	Steps
DDPM-V	Gaussian	trained	DDPM	128
DDIM-V	Gaussian	shared DDPM	DDIM, $\eta = 0$	128
DLPM	stable, $\alpha = 1.8$	trained	native DLPM	128
t-EDM	Student-t, $\nu = 2.2$	trained	stochastic Heun	64
Linear Gaussian flow	linear Gaussian	trained	Heun	128
DLPM (zero)	stable, $\alpha = 1.8$	zeroed	native DLPM	128
t-EDM (zero)	Student-t, $\nu = 2.2$	zeroed	stochastic Heun	64

Table 4: Main-model settings. DDPM and flow noise scales are selected by the pilot. t-EDM fixes $\sigma_{\min} = 0.005$, $\sigma_{\max} = 5$, and $\sigma_{\text{data}} = 1$.

F.3 Pilot Selection and Final Training

One optimization setting is selected per dataset and trained model, using the mean validation objective over pilot trials. Model-specific objective values are not compared across methods. The pilot budget reported here is 16 epochs, with three trials per candidate for vector data and one for CIFAR. Vector pilots use the full training split; CIFAR pilots use 4096 images. The pilot and final stages use the same backbone.

Data	Shared backbone
All vector experiments	Time-conditioned MLP: width 240, depth 4, time embedding dimension 128, normalization enabled, dropout zero. The same architecture is used in pilots, final training, learning curves, and the DLPM implementation comparison.
CIFAR100-LT	Transformer: two layers, four attention heads of dimension 32 (hidden width 128), patch size 4, continuous timestep conditioning, GELU approximation, adaptive layer normalization, and dropout zero.

Table 5: Neural-network architectures. Vector input/output dimension matches the dataset; image input/output shape is $3 \times 32 \times 32$.

Data	Learning-rate grid	$\sigma_{\max}$ grid	Candidates / trials
Synthetic	$\{10^{-4}, 2\times10^{-4}, 5\times10^{-4}\}$	$\{2, 5, 8\}$	9/3 or 3/3
HRRR, wildfire	$\{2\times10^{-4}, 5\times10^{-4}\}$	$\{2, 5\}$	4/3 or 2/3
CIFAR100-LT	$\{5\times10^{-5}, 10^{-4}, 2\times10^{-4}\}$	$\{2, 5, 8\}$	9/1 or 3/1

Table 6: Pilot grids. DDPM-V and Linear Gaussian flow tune learning rate and $\sigma_{\max}$; DLPM and t-EDM tune learning rate only. Candidate/trial counts are given in that order.

Pilot early-stopping patience is 32 epochs for vector data and 4 for CIFAR, with 16 warmup updates for CIFAR pilots. The synthetic 10k held-out split is used for test loss, not as the 5M distributional reference.

G Additional Empirical Results

G.1 Additional Vector-Distribution Results

Figure 3 completes the coordinate-TCE comparison with HRRR West. The remaining MMD^2 bar plots are collected in Figure 4.

G.2 Sample-Size Dependence

We compare DDPM-V and DLPM on isotropic α-stable targets with data tail index 1.7, in dimensions $d \in \{5, 30, 50\}$. We report training sizes $n \in \{10, 100, 1000, 10000\}$, averaged over five trials. Within each dimension/trial pair, training subsets are nested, and validation and reference pools use independent random streams. Validation uses 1000 samples and distributional evaluation generates 100,000 samples per model. All methods use the common MLP of Table 5; DLPM fixes noise index 1.8, while DDPM-V uses $\sigma_{\max} = 5$. Sampling uses 128 steps.

Each trained model receives exactly 10,000 optimizer updates, with replacement minibatches of size 1024, rather than a fixed epoch count. AdamW uses learning rate 5×10^{-4}, weight decay 10^{-6}, gradient clipping at norm 10, 400 warmup updates, and cosine decay to 5% of the initial rate. The initialization and minibatch-index streams are reset for paired comparisons across training sizes. We evaluate global fit using MMD^2 on 1000 reference and 1000 generated observations, following subsection E.2. Table 8 reports the five-trial means.

Data	Batch	Epochs	Patience	Weight decay	Warmup	Clip norm
Synthetic	1024	720	100	10^{-6}	400	10
HRRR, wildfire	512	720	100	10^{-6}	100	10
CIFAR100-LT	256	720	24	10^{-4}	50	5

Table 7: Final training settings. Epochs are maxima, patience is in epochs, and warmup is in optimizer updates. AdamW uses the pilot-selected learning rate and a cosine schedule ending at 5% of that rate. Each main model is trained once.

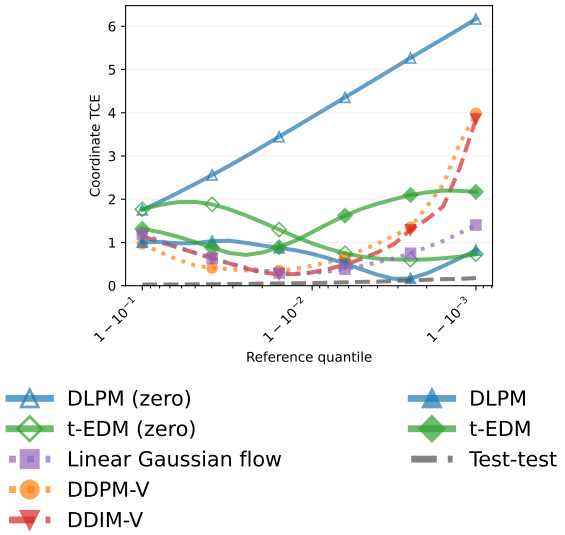
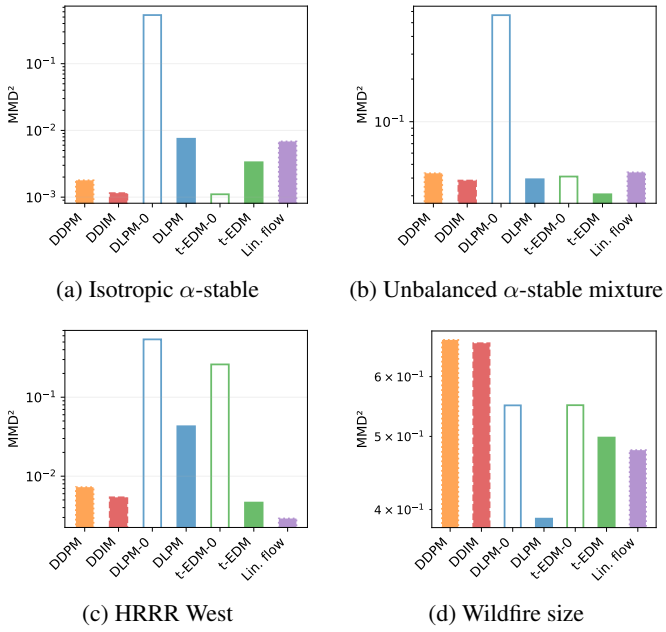


Figure 3: Coordinate TCE on HRRR West, using the same conventions as Figure 1.



(a) Isotropic α-stable

(b) Unbalanced α-stable mixture

(c) HRRR West

(d) Wildfire size

Figure 4: Remaining MMD^2 results. Lower is better; vertical axes are logarithmic. The suffix "0" denotes a zero-network control.

Train	$d = 5$		$d = 30$		$d = 50$	
	DDPM-V	DLPM	DDPM-V	DLPM	DDPM-V	DLPM
10	5.850×10^{-2}	3.119×10^{-1}	1.355×10^{-1}	2.812×10^{-1}	9.815×10^{-2}	2.871×10^{-1}
100	1.464×10^{-2}	1.833×10^{-1}	7.386×10^{-2}	2.638×10^{-1}	9.168×10^{-2}	2.458×10^{-1}
1k	1.459×10^{-3}	2.136×10^{-3}	2.406×10^{-2}	2.604×10^{-1}	5.065×10^{-2}	2.537×10^{-1}
10k	2.275×10^{-3}	6.656×10^{-4}	2.308×10^{-3}	1.033×10^{-3}	1.142×10^{-3}	1.984×10^{-2}

Table 8: Mean MMD^2 over five trials versus training size, for $d \in \{5, 30, 50\}$. Lower is better.

DDPM-V achieves lower mean MMD^2 across all dimensions for training sizes up to $n = 1000$, suggesting that DLPM faces greater statistical learning difficulties in data-scarce regimes. However, at $n = 10000$, DLPM outperforms DDPM-V for $d = 5$ and $d = 30$, while DDPM-V remains better for $d = 50$. This highlights the role of task difficulty: DLPM becomes more competitive as the sample size increases and the dimension decreases. These observations support our theoretical intuition that DLPM can face additional learning challenges, which may be mitigated by sufficient data or lower dimensionality.

G.3 DLPM empirical claim reproduction

Additionally, we reproduce the experimental setting of Table 1 in Shariatian et al. (2025), which reports improved tail coverage and robustness for DLPM on heavy-tailed data. To assess reproducibility, we compare our independent DLPM implementation, a thin wrapper around the authors' original implementation, and their published results, using the MSLE metric adopted in the original study. Despite these complementary implementations, we are unable to recover the reported performance ranking. As shown in Table 9, this discrepancy raises questions about the reproducibility of the original empirical findings.

$\text{MSLE}_{0.95}$	$\alpha = 1.5$	$\alpha = 1.7$	$\alpha = 1.9$	$\alpha = 2.0$
DLPM (our code)	0.203 ± 0.386	0.002 ± 0.001	0.004 ± 0.003	0.007 ± 0.009
DLPM (author code wrapped)	0.079 ± 0.071	0.005 ± 0.008	0.003 ± 0.002	0.009 ± 0.007
DLPM (paper results)	0.160 ± 0.128	0.071 ± 0.028	0.132 ± 0.101	0.798 ± 0.601

Table 9: Shariatian et al. (2025) protocol comparing our DLPM and the wrapped author implementation. Lower $\text{MSLE}_{0.95}$ is better.

H Additional Remarks and Discussion

We collect here additional remarks and discussions that provide further insight into our analysis.

H.1 Structural lower bounds.

As for VE-GD, each component of Equation 6 is unavoidable on an appropriate subclass of problems. *Initialization:* Writing $Y_T = b_T(S_\alpha + \rho_T Y_0)$, a deterministic translated initial condition $Y_0 = y$ gives $\text{TV}\big(\mathcal{L}(Y_T), \mathcal{L}(b_T S_\alpha)\big) \gtrsim |\rho_T| \, \|y\|$ for $|\rho_T|$ small, which is exponentially small in T when $|\rho_T| \asymp e^{-cT}$. As for VE-GD, this term is independent of n and is not a statistical error. *Approximation:* Classical approximation theory for Hölder classes yields the lower bound $m^{-\beta(\alpha)/d}$ for the local approximation error of the reverse object (Tsybakov, 2008). *Estimation:* Assouad and Fano arguments give the classical minimax rate $n^{-\beta(\alpha)/(2\beta(\alpha)+d)}$ for estimating the reverse object (Tsybakov, 2008). It remains to propagate these local errors to the generated distribution. The factor T in Equation 6 enters through the telescoping decomposition of the reverse chain (see subsection C.2), which bounds the global error by the sum of the T local kernel errors. The next proposition shows that this linear accumulation cannot be improved in general.

Proposition H.1 (Tightness of the T-accumulation) Let $\varepsilon \in (0, 1)$ and $T \geq 1$. There exist a measurable space Ω, an initial distribution μ_T on Ω, true reverse Markov kernels $(Q_t)_{t=1}^T$ and learned kernels $(\widehat{Q}_t)_{t=1}^T$ on Ω such that

$$\sup_{x \in \Omega} \text{TV}\big(Q_t(x, \cdot), \widehat{Q}_t(x, \cdot)\big) \leq \varepsilon \quad \text{for all } t \in \{1, \dots, T\},$$

and

$$\text{TV}\big(\mu_T Q_T \cdots Q_1, \; \mu_T \widehat{Q}_T \cdots \widehat{Q}_1\big) = 1 - (1 - \varepsilon)^T \geq \tfrac{1}{2} \min\{1, T\varepsilon\}.$$

The proof, given in subsection C.3, introduces at each reverse step a single "error channel" of mass ε. Each channel is activated exactly once along the trajectory and all errors push mass in the

same direction, so they accumulate without cancellation. Since the telescoping argument gives the matching upper bound $T\varepsilon$, the linear dependence on T is sharp up to a universal constant. Taking $\varepsilon \asymp m^{-\beta(\alpha)/d}$ (approximation) or $\varepsilon \asymp \sqrt{\mathrm{Comp}/n}$ (estimation) recovers the T-linear scaling of Theorem 2.2: without additional contractivity or cancellation properties of the reverse chain, the factor T is intrinsic to the reverse-chain structure rather than a proof artifact. As for VE-GD, these lower bounds hold term by term and do not constitute a complete minimax lower bound for the sum.

H.2 What the comparison does and does not say.

At first sight, the DLPM bound appears more favorable, in agreement with the intuitive principle that heavy-tailed data should be generated using heavy-tailed noise: its initialization error decays exponentially fast, as e^{-cT}, whereas the VE-GD initialization error only decays polynomially, as $T^{-1/2}$. This exponential rate is a genuine strength of DLPM; in our analysis, it follows from Assumption 2.4 on the noise schedule. Note that the comparison below concerns two representative formulations, Gaussian VE diffusion and DLPM, and not light-tailed versus heavy-tailed diffusion models in general. These formulations also differ in their time structure. Theorem 2.1 is stated for the continuous-time VE dynamics Equation 1 and its time reversal, so that the bound contains no time-discretization error and T is a diffusion horizon. Theorem 2.2 is stated for the discrete-time DLPM chain Equation 4, made of T reverse Markov kernels, so that T is both the horizon and the number of reverse steps; the linear factor T in its training error counts the local kernel errors accumulated along the chain.

Theorem 2.1 and Theorem 2.2 provide a common framework for comparing the sampling guarantees of the two models through the same *initialization-training* decomposition. This shared structure makes it possible to identify how model-specific terms contribute to the error bounds, although it does not establish an ordering of the actual sampling errors. The analyses account for the distinct objects learned by each model and retain the regularity assumptions of their original formulations, making explicit the conditions under which each guarantee holds. On one hand, VE-GD learns the score $\nabla \log p_t$ of Gaussian-smoothed marginals, so that Sobolev regularity of π_{data} is the natural assumption, Gaussian convolution transferring it to the smoothed marginals. On the other hand, DLPM learns the conditional reverse kernels of the α-stable chain; since no quantitative transfer from a Sobolev assumption on π_{data} to this reverse object is known, we impose a standard Hölder assumption directly on it. The two assumptions play parallel methodological roles, each tailored to the object learned by its respective model. Together with the structural lower bounds above, which show that each term is attained on a suitable subclass of problems, these results provide a principled basis for comparing the mechanisms governing the two error bounds. This yields the following *mechanistic* conclusion:

DLPM buys faster initialization at the price of a training cost that grows linearly with T.

Since a large T is precisely what enables the initialization gain, this cost may dominate in finite-sample regimes.

H.3 Understanding DLPM Training Instability

In our experiments, we observe that DLPM can exhibit greater run-to-run variability. Several factors may contribute to this behavior, including heavy-tailed perturbations, potentially more variable stochastic gradients, and the difficulty of estimating reverse dynamics in low-density regions. Our theoretical analysis also shows that local approximation and estimation errors can accumulate along the reverse chain, potentially amplifying these difficulties. However, our experiments do not isolate the respective contributions of these mechanisms. A systematic investigation of these mechanisms remains an interesting direction for future work.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We provide both a theoretical and an empirical sections.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the results obtained.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions setting are clearly mentioned.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The empirical setting is clearly described.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The datasets used are public and the code is provided.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All settings are detailed.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The number of trials for the empirical contributions was properly calibrated to reduce the results variance.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computer resources is given in the supplementary.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: For instance, the anonymity was carefully preserved.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: NA

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: NA

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.