Information Theoretic Learning for Diffusion Models with Warm Start

Yirong Shen
Imperial College London
ys6922@ic.ac.uk

Lu Gan
Brunel University of London
lu.gan@brunel.ac.uk

Cong Ling
Imperial College London
c.ling@imperial.ac.uk

Abstract

Generative models that maximize model likelihood have gained traction in many practical settings. Among them, perturbation-based approaches underpin many state-of-the-art likelihood estimation models, yet they often face slow convergence and limited theoretical understanding. In this paper, we derive a tighter likelihood bound for noise-driven models to improve both the accuracy and efficiency of maximum likelihood learning. Our key insight extends the classical Kullback–Leibler (KL) divergence–Fisher information relationship to arbitrary noise perturbations, going beyond the Gaussian assumption and enabling structured noise distributions. This formulation allows flexible use of randomized noise distributions that naturally account for sensor artifacts, quantization effects, and data distribution smoothing, while remaining compatible with standard diffusion training. Treating the diffusion process as a Gaussian channel, we further express the mismatched entropy between data and model, showing that the proposed objective upper-bounds the negative log-likelihood (NLL). In experiments, our models achieve competitive NLL on CIFAR-10 and state-of-the-art results on ImageNet across multiple resolutions, all without data augmentation, and the framework extends naturally to discrete data.

1 Introduction

Likelihood serves as a fundamental metric for evaluating density estimation and generative models. A tight negative log-likelihood (NLL) bound not only indicates a model's capacity to capture the fine-grained structure of the data distribution but also facilitates a range of downstream applications, including data compression [26, 29, 30, 33, 73, 75, 87], anomaly detection [7], out-of-distribution detection [84], semi-supervised learning [15], classifier [4, 86], image generation [89], transfer learning [57], density ratio estimation [6, 10], language models [23] and adversarial purification [71].

Rapid advancements in deep generative modelling [74] have led to various families of models achieving strong likelihood estimation performance, including energy-based models [21], normalizing flows [69, 90], variational autoencoders [40, 41, 55], diffusion models [28, 54, 62], cascaded models [46], and autoregressive models [9, 32]. A common underlying structure among many state-of-the-art models [40, 46, 62, 69, 90] is the transformation of data into noise via distinct functional mappings. These mappings, despite their differing mathematical forms, can be viewed as variants of diffusion models operating under the same noisy process, a Gaussian channel [14, 64, 82, 83].

Gaussian diffusion models can be broadly categorized into variance-preserving (VP) [28, 67] and variance-exploding (VE) [43, 72] processes by the variance of injected noise. These two paradigms differ in both the construction of the *forward* process and the formulation of likelihood. VP models are typically treated as Bayesian latent-variable models and trained via variational inference using the evidence lower bound (ELBO) [28, 40, 55, 62, 90], while VE models are interpreted as information-theoretic (IT) channels [25, 78, 82] that allow for direct likelihood estimation via estimation-theoretic tools [24, 43, 69]. However, existing likelihood estimation methods for VE models have not matched the performance of VP-based approaches. This raises the natural hypothesis

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

that likelihood performance may be highly sensitive to noise variance. Furthermore, while previous IT bounds [43, 44, 69, 81] may be slightly looser than ELBO, they often enjoy faster convergence and greater interpretability [3, 16, 45, 69, 83], leaving open the question of whether IT-based bounds can also be improved as competitive likelihood estimators with enjoying the faster speed and robustness.

Addressing this question requires extending existing *theoretical* frameworks. The extant Shannon–Fisher connections for diffusion models have largely assumed idealized isotropic Gaussian corruption [51, 69], yet real-world data rarely align with such simplified assumptions. Imaging data commonly feature Poisson-Gaussian sensor noise [20]; dequantization [27, 31, 74, 90] and data smoothing [52] often modeled by uniform or symmetric noise (*e.g.* a Laplacian kernel) addition to improve tail coverage, sharp transitions and robustness. Additionally, recent generative models deliberately introduce Poisson [85], heavy-tailed α-stable perturbations [88] or structured noise with the data distribution [62, 65], emphasizing the practical needs for more generalized frameworks.

In specific, when a continuous-density model is fitted directly to discrete data, the likelihood evaluation becomes singular and severely degrades performance. The conventional remedies, uniform or variational dequantization [27, 69, 74], inject auxiliary noise but suffer two drawbacks: they require an additional training phase, which is hard to train to the optimal, and, in general, introduce a pronounced training-evaluation gap that inflates the NLL performance [90] via the mismatched noise.

In this work, to eliminate both sources of discrepancy and instabilities in maximum likelihood learning with diffusion models, we propose *variance-aware likelihood bounds* via *arbitrary isotropic warm-up noise* perturbation. Our main contributions and findings are summarized as followed:

- We first prove Theorems 1 and Proposition 2, showing that for any isotropic noise, the Fisher information measures are asymptotically equivalent to their Shannon counterparts, thereby establishing a maximum likelihood learning framework well beyond the Gaussian setting.
- Building on this insight, Theorem 2 and Proposition 1 provide tightened analytic bounds on the likelihood of diffusion models with only minor architectural changes, specifically, a logarithm signal-to-noise ratio based parameterization and an additional low-variance noise regime, which together eliminate the train–test gap and stabilize optimization near $t = 0$.
- Empirically, ablation studies confirm the effectiveness of our bounds. Using an efficient importance-sampling scheme, our method achieves **2.50 bits/dim** on CIFAR-10 and new state-of-the-art results of **3.01**, **2.91**, and **2.59 bits/dim** on ImageNet-32, -64, and -128, respectively, while requiring only *0.3M training iterations* without any data augmentation.

2 Background

2.1 Incremental Gaussian Channel and Maximum Likelihood Estimation

Let the *source* dataset with N datapoints be denoted by $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^N$[1]. Assume every datapoint $\mathbf{x}$ is an i.i.d. sample drawn from an unknown distribution $p(\mathbf{x})$ supported on the *source space* $\mathcal{X} \subset \mathbb{R}^D$. Diffusion models [28, 40] naturally induce what is known as an *incremental signal-to-noise ratio (SNR) channel* [25, 93], which generates a sequence of progressively noisier $\mathbf{X}$ towards a pure noise:

$$\mathbf{Y}_t = \alpha_t \mathbf{X} + \sigma_t \mathbf{N}, \tag{1}$$

where $\alpha_t, \sigma_t^2 \in \mathbb{R}^+$ are smooth, non-negative, scalar-valued functions with finite derivatives with respect to time t over the fixed time horizon $t \in [0, 1]$. The noise term $\mathbf{N}$ consists of independent standard Gaussian entries. Intuitively, the ratio α_t^2/σ_t^2 can be interpreted as the signal-to-noise ratio (SNR) at time t, denoted as $\mathrm{SNR}(t) := \alpha_t^2/\sigma_t^2$. By enforcing $\mathrm{SNR}(t)$ to be strictly monotonically decreasing over the time interval $t \in [0, 1]$, the output $\mathbf{Y}_t$ asymptotically approaches a well-defined, analytically tractable stationary distribution $\pi(\mathbf{x})$ as $t \to 1$.

To recover and estimate the underlying data distribution $p(\mathbf{x})$ from noisy observations $\mathbf{Y}_t$, one must solve a density estimation problem under this forward noisy channel. From an information-theoretic perspective, the mismatched Gaussian channel [78] models the scenario where the true input distribution $p(\mathbf{x})$ is unknown, and one instead uses a hypothesis distribution $q(\mathbf{x})$ for estimation.

[1] In this paper, random objects are denoted by uppercase letters, and their realizations by lowercase letters. The expectation $\mathbb{E}(\cdot)$ is taken over the joint distribution of the random variables inside the brackets.

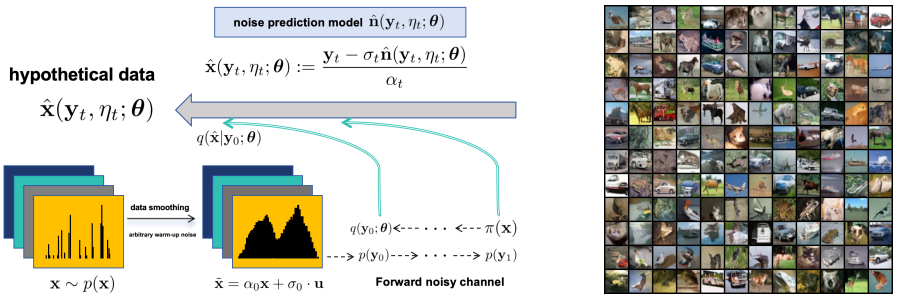

Figure 1: Toy example illustrating our method (left) and samples generated on CIFAR-10 (right). We apply an identical *warm-up* channel to both the data $p(\mathbf{x})$ and the model distribution $q(\mathbf{x}; \boldsymbol{\theta})$: injecting arbitrary noise $\boldsymbol{\Psi}$ produces smoothed data $\bar{\mathbf{x}}$ and a correspondingly perturbed model $\bar{q}_\theta$. This results in *two* variance-regime mismatched channels, a low-variance arbitrary-perturbation channel and a high-variance Gaussian noise regime, that share identical channel dynamics but differ in their priors, under which training aligns the Gaussian-perturbed distributions and learns the denoising map.

While classical methods for Gaussian channels remain applicable, their analytic solutions are typically intractable because it requires sampling from the posterior distribution of the noisy channel. A more tractable alternative involves leveraging the connection between relative entropy and Fisher divergence. Specifically, [51, 69] demonstrated that the weighted score matching (or equivalently, Fisher divergence) objectives could approximate maximum likelihood training of diffusion models[2].

2.2 Likelihood of Diffusion Models

Average log-likelihood is widely recognized as the default metric for evaluating generative models. Previous work has largely prioritized perceptual quality, emphasizing coarse scale patterns and global consistency of generated images, with common metrics such as the Fréchet Inception Distance (FID) and the Inception Score (IS). In contrast, we optimise for likelihood of the model, a criterion that is inherently sensitive to fine-scale details and the exact values of individual pixels.

To evaluate the likelihood in diffusion models, we define a model distribution $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$, typically parameterized by a neural network with parameters $\boldsymbol{\theta} \in \boldsymbol{\Theta}$, which aims to approximate the true data distribution $p(\mathbf{x})$. Given a sample $\mathbf{x} \sim p(\mathbf{x})$ and noise $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I})$, a noisy observation $\mathbf{y}_t$ is generated via the forward channel described in Equation (1). Under the mismatched channel framework, the marginal distribution over $\mathbf{y}_t$ induced by the model sample $\hat{\mathbf{x}} \sim q(\hat{\mathbf{x}}; \boldsymbol{\theta})$ is given by:

$$q(\mathbf{y}_t; \boldsymbol{\theta}) = \int_{\mathbb{R}^D} q(\hat{\mathbf{x}}; \boldsymbol{\theta})\, p(\mathbf{y}_t|\mathbf{x})\, d\hat{\mathbf{x}}. \tag{2}$$

In VP diffusion modeling, the stationary distribution at $t = 1$ is defined to be the stationary distribution $\pi(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$, which serves as the starting point for the sampling process. Since our primary focus is density estimation and probabilistic modeling, we defer implementation and details of sampling process to the Appendix. A.3 and optimization algorithms to future work.

Following [68], we parameterize the model distribution using a noise prediction network. Specifically, the model $\hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})$ is trained to predict the Gaussian noise $\mathbf{n}$ that was added during the forward process, where $\eta_t := -\log \mathrm{SNR}(t)$ defines the noise schedule in log-SNR space. A predicted hypothetical data $\hat{\mathbf{x}}$ is then obtained as:

$$\hat{\mathbf{x}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta}) := \frac{\mathbf{y}_t - \sigma_t \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})}{\alpha_t}. \tag{3}$$

Theoretically, the training objective of likelihood-based diffusion models is to minimize the KL divergence (see Def. 3) between the true data distribution $p(\mathbf{x})$ and the continuous model distribution $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$, i.e., $D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta}))$. However, as the noise variance $\sigma_t^2 \to 0$, the SNR diverges, leading to numerical instability during both training and sampling [38]. To address this, practical

[2] In likelihood-based generative modeling, this formulation is equivalent to maximizing the expected log-likelihood. In contrast, methods that prioritize sample quality typically optimize the 2-Wasserstein distance.

Algorithm 1: Training

Pre-processing: Wart Start
for *each* $\mathbf{x}$ *in* $\mathcal{X}$ **do**
  $\mathbf{u} \sim \Psi$; // Ψ: arbitrary noise
  $\tilde{\mathbf{x}} \leftarrow \alpha_0 \cdot \mathbf{x} + \sigma(0) \cdot \mathbf{u}$; // store $\tilde{\mathbf{x}}$

Training:
repeat
  $\tilde{\mathbf{x}} \sim \mathcal{X}, \mathbf{n} \sim \mathcal{N}(0, \mathbf{I})$;
  $t \sim \mathcal{U}(0,1), (\alpha_t^2, \sigma_t^2) \leftarrow t$;
  $\mathbf{y}_t \leftarrow \alpha_t \cdot \tilde{\mathbf{x}} + \sigma_t \cdot \mathbf{n}$;
  Take gradient descent step on
    $\nabla_{\boldsymbol{\theta}} \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, t; \boldsymbol{\theta})\|^2$;
until *converged*;

Algorithm 2: Likelihood Evaluation

Input: Trained model $f_{\boldsymbol{\theta}}$, test data $\mathbf{x}$
Output: Estimated loss $\mathcal{L}(\boldsymbol{\theta})$, NLL $\ell(\mathbf{x}; \boldsymbol{\theta})$
for *each* $\mathbf{x}$ *in* $\mathcal{X}$ **do**
  $\mathbf{u} \sim \Psi$; // Ψ: arbitrary noise
  $\tilde{\mathbf{x}} \leftarrow \alpha_0 \cdot \mathbf{x} + \sigma(0) \cdot \mathbf{u}$; // store $\tilde{\mathbf{x}}$

Evaluation:
  $\eta \sim \rho(\eta), (\alpha_\eta^2, \sigma_\eta^2) \leftarrow \eta$;
  $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I}), \quad \hat{\mathbf{n}} \leftarrow f(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})$;
  $\mathcal{L}(\sigma_\eta^2; \boldsymbol{\theta}) \leftarrow 0.5 \cdot Z \, \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|^2$;
    // Z is a normalizing constant
$\ell(\mathbf{x}; \boldsymbol{\theta}) \leftarrow \mathcal{H}(p(\mathbf{y}_1|\mathbf{x}), \pi) + \mathcal{L}(\sigma_t^2; \boldsymbol{\theta})$
return $\ell(\mathbf{x}; \boldsymbol{\theta})$;

diffusion models typically start the forward process at a small positive time $t = \epsilon > 0$ instead of $t = 0$ for improved stability. Yet, this small time offset introduces additional perturbation, and the corresponding training objective becomes $D_{\mathrm{KL}}(p(\mathbf{y}_\epsilon) \| q(\mathbf{y}_\epsilon; \boldsymbol{\theta}))$, which differs from the evaluation objective $D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta}))$. This discrepancy causes a mismatch between the training *expected log-likelihood* ($\mathbb{E}_{p(\mathbf{y}_\epsilon)}[\log q(\mathbf{y}_\epsilon; \boldsymbol{\theta})]$) and testing *expected log-likelihood* ($\mathbb{E}_{p(\mathbf{x})}[\log q(\hat{\mathbf{x}}; \boldsymbol{\theta})]$).

2.3 Dequantization for Density Estimation

When modeling real-world data, care must be taken to ensure that the reported likelihood values are meaningful [74]. Since such data is typically discrete, using continuous density models directly can lead to arbitrarily high likelihoods due to singularities. To mitigate this issue, it is now standard practice to add real-valued noise to integer-valued inputs, a process known as *dequantization* [17, 69, 76]. For example, in the case of 8-bit image data, the input values in $\{0, 1, \ldots, 255\}$ are typically perturbed by uniform noise, yielding $\mathbf{v} = \mathbf{x} + \mathbf{u}$ where $\mathbf{u} \sim \mathcal{U}[0,1)^D$. With this transformation, training a continuous density model on the uniformly dequantized data $\mathbf{v}$ can be interpreted as maximizing a lower bound on the log-likelihood of a discrete model defined over the original quantized inputs [27, 74]. However, this introduces a training–test gap in diffusion models: during training, the model $q(\mathbf{v}; \boldsymbol{\theta})$ is fitted to $p(\mathbf{y}_0)$, which corresponds to a Gaussian distribution centered at each discrete data point; during evaluation, however, q is tested on uniformly dequantized data. Although this improves numerical stability in practice [38, 69], the mismatch between training and evaluation degrades likelihood performance [90]. Moreover, while variational methods [27] allow other forms of noise injection, they introduce an additional optimization stage that is computationally expensive and often difficult to train to optimality.

3 Variance-Aware Likelihood Estimation of Diffusion models

In this section, we present an information-theoretical framework for variance-aware likelihood estimation in diffusion models. While retaining the standard score matching objective [34, 79], our method introduces a tighter, pointwise upper bound on the negative log-likelihood. We also incorporate a modified forward process and an importance sampling scheme to reduce the variance of the Monte Carlo estimator. Notations and definitions are deferred to the Appendices. A.1 and A.2.

In specific, we analyze the impact of noise variance (schedule) on likelihood estimation through the lens of Fisher divergence via thermodynamic integration [22, 56] along the entire noise variance-regime space, which includes a low-variance arbitrary noise regime ($0 \leq \sigma_t^2 < \sigma_0^2$) and a high-variance Gasussian channel ($\sigma_0^2 \leq \sigma_t^2 \leq \sigma_1^2$), providing a formal connection between score matching objectives to KL divergence under arbitrary isotropic noise perturbations. This analysis leads to an exact expression of the mismatched entropy $\mathcal{H}(p(\mathbf{x}), q(\mathbf{x}; \boldsymbol{\theta}))$, which improves optimization process, numerical stability and remains compatible with probabilistic modeling under non-Gaussian noise setting. Our method toy example pipeline is illustrated in Fig. 1 and Algorithms 1 and 2.

3.1 Relationship between Score Matching and KL Divergence

Beyond the pathology discussed in Section 2.2, real-world scenarios rarely align with the idealized Gaussian assumption. Sensor imperfections introduce structured noise such as Poisson, uniform

quantization errors, or impulsive salt-and-pepper patterns. Hence, extending KL–Fisher relations beyond Gaussian perturbations is not merely a theoretical extension, but addresses a pressing practical need in robust generative modeling. Motivated by these practical considerations, we investigate how training objectives that incorporate noise-perturbed distributions relate to the classical maximum likelihood principle. Specifically, we highlight the role of Fisher divergence (see Def. 5) in characterizing the first-order sensitivity of KL divergence under small additive noise. By doing so, we generalize prior results [50, 51, 69] from Gaussian to arbitrary isotropic noise distributions. Importantly, our derived relation ensures that score-matching losses remain consistent with the first-order KL term, thus preserving the maximum-likelihood interpretation for a broader class of models, including Poisson-flow [85] and Lévy-based diffusion frameworks [88].

Theorem 1 (Score Matching as the Small-Noise Limit of KL Divergence)**.** *Let $\mathbf{X} \sim p(\mathbf{x})$ be an arbitrary distributed random vector on $\mathbb{R}^D$, and let $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$ be a parametric model with $\hat{\mathbf{X}} \sim q(\hat{\mathbf{x}}; \boldsymbol{\theta})$. Define the perturbed observation*

$$\tilde{\mathbf{X}} := \alpha_t \mathbf{X} + \sigma_t \boldsymbol{\Psi},$$

where $\boldsymbol{\Psi}$ is a random vector independent of $\mathbf{X}$, satisfying $\mathbb{E}[\boldsymbol{\Psi}] = 0$ and $\mathrm{Cov}(\boldsymbol{\Psi}) = \mathbf{I}$. Let $p_{\sigma_t^2}$ and $q_{\sigma_t^2}$ denote the densities of $\tilde{\mathbf{X}}$ under p and q with noise variance σ_t^2, respectively. Suppose that the KL divergence $D_{\mathrm{KL}}(p_{\sigma_t^2} \| q_{\sigma_t^2})$ is finite for sufficiently small σ_t^2. Then the following limit holds:

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p_{\sigma_t^2} \| q_{\sigma_t^2}) \Big|_{\sigma_t^2 \to 0^+} = -\frac{1}{2} \int_{\mathbb{R}^D} p(\mathbf{x}) \left\| \nabla \log p(\mathbf{x}) - \nabla \log q(\hat{\mathbf{x}}; \boldsymbol{\theta}) \right\|^2 d\mathbf{x}, \tag{4}$$

i.e.,

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) \Big|_{\sigma_t^2 \to 0^+} = -\frac{1}{2} I\left(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})\right),$$

where $I(\cdot\|\cdot)$ denotes the Fisher divergence (equivalently, score matching objective) between p and q.

Proof. See Appendix. A.5.1. □

To further investigate the effect of additive noise on training objectives, we consider a second-order expansion of the KL divergence in terms of the noise variance σ_t^2. When both the true and model distributions are perturbed by small additive arbitrary noise, the KL divergence between them satisfies:

$$D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) + \frac{\sigma_t^2}{2} I(p\|q) + o(\sigma_t^2). \tag{5}$$

While maximum likelihood aims to minimize the KL divergence directly, according to Theorem. 1, score matching, defined through the Fisher divergence [34], does not directly minimize the KL divergence explicitly. The two objectives coincide only in the limit $\sigma_t^2 \to 0$ where score matching captures the first-order sensitivity of KL divergence to additive noise. When σ_t^2 is not infinitesimal, the Fisher term may dominate, potentially leading to biased or unstable solutions. This result significantly generalizes the observation that score matching seeks to eliminate its derivative in the scale space at $t = 0$ into arbitrary noise settings. In contrast, it is known that variational methods are known to be highly sensitive to noisy training data (see Appendix. B.6 for more details), which may give rise to many false extreme values, whereas score matching tends to be more robust to small perturbation in training data, suggesting that it seeks parameters which lead to models robust to noisy setting.

Therefore, the generalized result in Theorem 1, formalizing its role as a local approximation for likelihood-based training beyond the Gaussian setting [51]. To build upon this foundation, we next consider the case where the forward process begins at a small but strictly positive variance, satisfying $0 < \sigma_0^2 \ll 1$. This setting allows us to integrate from σ_0^2 rather than from zero, yielding a tractable and numerically stable lower bound while preserving consistency with the asymptotic result established above. We formalize this construction in the next section.

3.2 Bounding the Mismatched Entropy with Thermodynamic Diffusion

Building on Section 3.1, we can now consider a practical setting where the forward diffusion process (1) begins at a small but strictly positive noise level $0 < \sigma_0^2 \ll 1$. In this regime, we derive the following approximation on the mismatched entropy (see Def. 4) based on denoising score matching objectives [79] via thermodynamic integration [22, 56] along the interval $[\sigma_0^2, \sigma_1^2]$.

Proposition 1 (Thermodynamic Decomposition of Mismatched Entropy)**.** *Consider the signal model* (1)*, suppose* $p(\tilde{\mathbf{x}})$ *and* $q(\tilde{\mathbf{x}};\boldsymbol{\theta})$ *have continuous second-order derivatives and finite second moments. Denote by* $p(\mathbf{y}_1)$ *and* $\pi(\mathbf{x})$ *the output signals of channel at time* $t = 1$ *when inputs are* $p(\mathbf{x})$ *and* $q(\hat{\mathbf{x}};\boldsymbol{\theta})$ *respectively. Assume* $\pi(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$*, which is independent of* $\boldsymbol{\theta}$*. Let* $p(\mathbf{y}_t|\mathbf{x})$ *denote the channel* (1) *for any* $t \in [0, 1]$*, then for arbitrary datapoint* $\mathbf{x}$ *and small* σ_0^2 *with* $0 < \sigma_0^2 \ll 1$*:*

$$\mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}};\boldsymbol{\theta})) = \mathcal{H}(p(\mathbf{y}_1), \pi(\mathbf{x})) + \mathcal{J}_{DSM}(\boldsymbol{\theta}; \sigma_t^2(\cdot)) - \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}\|\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x})\|^2 d\sigma_t^2 + o(\sigma_0^2). \tag{6}$$

Here, the denoising score matching objective is defined as

$$\mathcal{J}_{DSM}(\boldsymbol{\theta}; \sigma^2(\cdot)) := \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}\|\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x}) - \hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta})\|_2^2 \, d\sigma_t^2, \tag{7}$$

and $\hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta}) = \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t;\boldsymbol{\theta}) := -\hat{\mathbf{n}}(\mathbf{y}_t, \eta_t;\boldsymbol{\theta})/\sigma_t$ *is a score network estimator.*

Proof. Detailed proof defers in Appendix. A.5.2. □

The Proposition 1 preserves the theoretical guarantees of the Fisher–KL expansion while yielding a tractable and computable loss. The resulting bound remains asymptotically tight as $\sigma_0^2 \to 0$, up to an additive residual term of order $o(\sigma_0^2)$. The resulting bound remains asymptotically tight as Proposition 1 can be interpreted through the lens of the cost of mismatch in statistical inference [35]. In this setting, the true data distribution is p, but the decoder employs an estimator optimized for a mismatched model $q_{\boldsymbol{\theta}}$. The resulting mismatched entropy $\mathcal{H}(p, q_{\boldsymbol{\theta}})$ quantifies the average code length incurred under this model discrepancy. It shows that this quantity decomposes into three parts: a mismatched output distribution loss, a score approximation error arising from the use of $\hat{\boldsymbol{s}}(\cdot;\boldsymbol{\theta})$ instead of the true score, and an irreducible term linked to the Fisher information of the channel.

3.3 Bounding the log-likelihood on Individual Datapoints

In many applications that benefit from likelihood optimization [29, 33], it is desirable to evaluate the log-likelihood of individual data points, which can be highly sensitive to fine-scale variations and the precise values of pixel intensities [40]. To account for this sensitivity, we derive a pointwize lower bound that remains tractable and stable under finite noise injection.

Theorem 2. *Let* $p(\mathbf{y}_t|\mathbf{x}) = \mathcal{N}(\alpha_t\mathbf{x}, \sigma_t^2\mathbf{I})$ *denote the Gaussian channel at any time* $t \in [0, 1]$*. With the same notations and conditions in Proposition 1, we have*

$$-\log q(\hat{\mathbf{x}};\boldsymbol{\theta}) \leq \mathcal{H}(p(\mathbf{y}_1|\mathbf{x}), \pi(\mathbf{x})) + \mathcal{L}_{DSM}(\sigma_t^2;\boldsymbol{\theta}), \tag{8}$$

in which $\mathcal{L}_{DSM}(\sigma_t^2;\boldsymbol{\theta})$ *is defined as*

$$\mathcal{L}_{DSM}(\sigma_t^2;\boldsymbol{\theta}) := \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}_{p(\mathbf{y}_t|\mathbf{x})}\|\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x}) - \hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta})\|_2^2 \, d\sigma_t^2, \tag{9}$$

and $\mathcal{H}(p(\mathbf{y}_1|\mathbf{x}), \pi(\mathbf{x}))$ *is given by*

$$\mathcal{H}(p(\mathbf{y}_1|\mathbf{x}), \pi(\mathbf{x})) = D_{KL}(p(\mathbf{y}_1|\mathbf{x})\|\pi(\mathbf{x})) + \mathcal{H}(p(\mathbf{y}_1|\mathbf{x})). \tag{10}$$

Proof. Detailed proof defers in the Appendix. A.5.3. □

3.4 Variance Reduction with Importance Sampling

A key challenge in evaluating diffusion models lies in accurately and efficiently estimating the loss integral of Equation (9), especially under a chosen noise variance schedule σ_t^2. In Appendix. B.2 we show that during training, the noise schedule acts as an importance sampling distribution for loss estimation, and therefore plays a crucial role in efficient optimization.

Among such, popular choices include linear [28], cosine [54], and learnable parameterizations [40, 62], with sigmoid-based schedules $\sigma_t^2 = \text{sigmoid}(\eta(t))$ remaining prevalent in state-of-the-art models. Accordingly, we employ a Monte Carlo estimator of this loss for evaluation and optimization.

To further improve the efficiency and reduce variance in Monte Carlo estimation, we propose two importance sampling (IS) strategies (see Appendix. B.3 for details).

Formally, the diffusion training objective can be expressed as an integral over the noise variance. It is shown in Appendix. B.1 that our denoising score matching loss (8) could be simplified to a noise prediction model $\hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})$ that directly infers the noise $\mathbf{n}$ that was used to generate $\mathbf{y}_t$:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \frac{1}{2} \int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}_{n \sim \mathcal{N}(0,\mathbf{I})} \left[\sigma_t^{-2} \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2 \right] \, d\sigma_t^2. \tag{11}$$

In practice, the evaluation of the integral is time-consuming, and Monte-Carlo methods are used to unbiasedly estimate it. In particular, the variance of the estimator has a direct impact on the optimization process, influencing both its stability and convergence speed. To mitigate this issue, it can be beneficial to decouple the integration variable used for loss estimation from the time variable employed during training. To this end, we introduce the negative log-SNR variable η and rewrite the training objective (11) as an expectation over this reparameterized variable, as shown followed. By interpreting the integral above as an expectation over the η, the Monte Carlo estimation highlights the role of $\rho(\eta)$ as an IS distribution and $w(\eta)$ as a noise variance-dependent weighting function:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \frac{1}{2} \mathbb{E}_{\mathbf{n} \sim \mathcal{N}(0,\mathbf{I}), \eta \sim \rho(\eta)} \left[\frac{w(\eta)}{\rho(\eta)} \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2 \right]. \tag{12}$$

Equivalently, optimizing $\rho(\eta)$ can be viewed as learning a monotone mapping $\eta(t) : [0, 1] \to [\eta_0, \eta_1]$, which corresponds to the inverse cumulative distribution function (CDF) of $\rho(\eta)$. Thus, the Monte Carlo estimator achieves reduced variance and admits closed-form inverse-CDF sampling, which motivates the following intuitively designed proposals.

Here, we sample $\eta \sim \rho(\eta)$ with a continuous distribution $\rho(\eta) \propto w(\eta)$ that renders the weight in (12) time-invariant, so that the noise prediction error remains scale-consistent across all $\eta \in [\eta_0, \eta_1]$ values without any amplification or shrinkage during the estimation. This is analogous to the likelihood-weighted and flow matching methods which conduct uniform sampling for all η [39, 90].

Furthermore, to further minimize the variance of the Monte Carlo estimator, we can optimize the sampling distribution $\rho(\eta)$ directly [40, 90]. Specifically, we parameterize $\eta(t)$ as a monotonic neural network and adjust it to minimize the variance of the diffusion loss (see Appendix. B.3.2). While this *learned* proposal can effectively reduce Monte Carlo approximation variance, it introduces additional optimization overhead and potential instability. Empirically, we find that the hand-crafted proposal achieves better convergence speed and comparable NLL results, without requiring additional training objectives or neural components. As such, we adopt our designed IS as our default strategy.

3.5 Training-Free Dequantization with Warm-up Noise

Real-world datasets usually contain discrete input, such as images or texts, and must be dequantized when training continuous-density models. As discussed in Section 2.3, and more details in Appendix. G, while dequantization makes diffusion models applicable to discrete data, the training–test discrepancy originates from the mismatch in noise injection between training and inference/testing. To reduce this discrepancy, we introduce an arbitrary isotropic *warm-up* noise $\mathbf{u} \sim \boldsymbol{\Psi}$ and inject it into training, inference and testing, that fit seamlessly into continuous-density diffusion models, and preserves the original maximum-likelihood objective without retraining.

By Theorem 1, the generalized KL–Fisher identity holds for any isotropic noise, which means that dequantization schemes previously restricted to uniform and Gaussian noise can now be extended to, for instance, logistic and Laplacian perturbations. However, existing average log-likelihood bounds [27, 74] are derived under mismatched entropy, defined as the sum of the KL divergence and the differential entropy. This motivates the generalization of the differential entropy formulation to arbitrary noise distributions, as follows, a result known as the de Bruijn identity in information theory.

Proposition 2 (de Bruijn identity with arbitrary noise [60])**.** *Let $\tilde{\mathbf{X}} = \alpha_t \mathbf{X} + \sigma_t \boldsymbol{\Psi}$, where $\mathbf{X}, \boldsymbol{\Psi} \in \mathbb{R}^D$ are independent random vectors, with $\boldsymbol{\Psi}$ satisfying $\mathbb{E}[\boldsymbol{\Psi}] = 0$ and $\mathrm{Cov}(\boldsymbol{\Psi}) = \mathbf{I}$. Assume the probability density function $p(\mathbf{x})$ of $\mathbf{X}$ is twice continuously differentiable and decays sufficiently fast at infinity, and that the Fisher information $\mathcal{J}(\mathbf{X})$ exists and is finite. Then,*

$$\frac{d}{d\sigma_t^2} \mathcal{H}\big(p(\tilde{\mathbf{x}})\big)\Big|_{\sigma_t^2 \to 0^+} = \frac{1}{2} \mathcal{J}\big(p(\mathbf{x})\big), \tag{13}$$

where $\mathcal{H}(\cdot)$ denotes the differential Shannon entropy (see Def. 1 and $\mathcal{J}(\cdot)$ denotes the Fisher information (see Def. 2).

The key implication of (13) is that the derivative is independent of the detailed statistics of the noise distribution. This suggests that the differential entropy forms a smooth manifold $\sigma_t \mathbf{\Psi}$, whose local geometry resembles an isotropic quadratic bowl. Combining Proposition 2 with the Theorem 1 gives a characterization of the gap between the true discrete cross-entropy and its continuous surrogate. Such dequantization noise can provide a tighter likelihood bound yet with no additional training costs. Our detailed empirical settings of dequantization noise could be found in Appendix. 2.3.

3.6 Numerical Stability

All theoretical developments in this work assume that the forward diffusion process begins at a small but strictly positive variance level $\sigma_0^2 > 0$. This choice avoids the pathological behaviour of the $\sigma_0^2 \to 0$ limit, which often leads to numerical instability in both training and evaluation [38, 40, 69]. Accordingly, we restrict the variance schedule to the interval $[\sigma_0^2, \sigma_1^2]$ throughout all derivations and implementations. Truncating the lower limit of integration to σ_0^2 introduces an approximation error of order $o(\sigma_0^2)$, as discussed in Section 3.1, which is negligible in both theory and practice. This ensures that our likelihood-weighted score matching objective remains a consistent estimator of the log-likelihood, while simultaneously improving numerical stability. Empirically, we observe that this design remains robust across diverse noise distributions, further confirming the practical reliability.

4 Related Works

Most prior analyses of variance schedules [1, 49, 54] in diffusion models focus on their role in reverse-time sampling [2, 40, 47, 55, 62, 68, 91, 92] (e.g., by approximating posteriors or solving reverse-time ODEs [50, 90]). In contrast, our work examines how the forward variance schedule influences both robustness and likelihood estimation. Proposition 2 further generalizes the classical de Bruijn identity [12] from Gaussian to arbitrary noise perturbations. Whereas prior information-theoretic studies [18, 80, 94] have characterized the connection of KL and Fisher divergences under Gaussian smoothing [50, 51, 69, 93], we extend these relationships to non-Gaussian noise families. Finally, [43] studies relative-entropy interpretations of denoising objectives and connects them to mean-square error, but does not address mismatched estimation [35, 78] in the sense of our framework.

5 Experiments

In this section, we present our training procedure and experiment settings, and our ablation studies to demonstrate how our techniques improve the likelihood of diffusion models. We report negative log-likelihood (NLL, in bits/dim) and Fréchet Inception Distance (FID) scores for all experiments, and compare convergence speed across models. Then we evaluate our model on a lossless compression benchmark against several competitive baselines in neural compression literature. Note that we focus here on pushing the state-of-the-art in density estimation, and while we report FID for completeness, we defer sample quality optimization to future work. Additional results can be found in Appendix D.

Datasets and Implementation We evaluate on CIFAR-10, anti-aliased ImageNet-32 dataset, ImageNet-64 and -128. No data augmentation is applied. We adopt the same architecture and hyperparameters as the VDM model [40], including a U-Net consisting of convolutional ResNet blocks without any downsampling. Unlike embeddings of diffusion time t, we embed the log-SNR value $\eta(t)$ or its reverse CDF time embedding instead. This modification better reflects the underlying noise scale and leads to improved likelihood estimation with importance sampling trick in practice.

Selection of Noise Schedules We consider two representative choices: 1. Variance-Preserving (VP): $\alpha_\eta^2 + \sigma_\eta^2 = 1$, ensuring unit marginal variance throughout the forward process. 2. Straight-Path (SP): $\alpha_\eta + \sigma_\eta = 1$, corresponding to linear interpolation in data space as in [1, 49]. We omit VE schedules, since in our preliminary runs the best VE model attained only 3.27 bits/dim on CIFAR-10, substantially worse than VP and SP. Moreover, to further illustrate the effects of the noise variance, we consider to evaluate various variance functions σ_t^2 shown in Appendix. B.4.

Table 1: Negative log-likelihood (NLL) in bits per dimension (BPD), FID, and number of training iterations (Iter., in million) on CIFAR-10 and ImageNet-32 datasets. "/" indicates results not reported or not applicable. *Denotes results obtained on the original ImageNet-32 release. Boldface denotes the best performance within each column, and blue text marks the second-best.

Models	CIFAR-10			ImageNet-32		
	NLL ↓	FID ↓	Iter.	NLL ↓	FID ↓	Iter.
(The models with IT methods)						
ScoreODE (second order) [50]	3.44	**2.37**	1.3	4.06*	/	1.3
ScoreODE (third order) [50]	3.38	2.95	1.3	4.04*	/	1.3
ScoreFlow [69]	2.80	5.34	1.6	3.79*	11.20*	1.6
Flow Matching [49]	2.99	6.35	0.391	3.53	**5.02**	0.25
Stoch. Interp. [1]	2.99	10.27	0.5	3.48*	8.49	0.6
i-DODE (SP with IS) [90]	2.56	11.20	6.2	3.44 / 3.69*	10.31	2.25 / 2.5*
(The models with variational methods)						
VDM [40]	2.65	7.60	10	3.72*	/	2
DiffEnc [55]	2.62	11.20	8	3.46	/	8
MuLAN [62]	2.55	17.62	8	3.67*	13.19	2
BSI [48]	2.64	/	10	3.44	/	10
W-PCDM [46] (VDM [40] weight)	**2.35**	6.23	2	3.32	/	10
W-PCDM [46] (EDM [37] weight)	10.31	2.42	2	/	/	/
Ours (SP with IS)	2.49	/	0.3	3.02	/	0.3
Ours (VP with IS)	2.50	10.18	0.3	**3.01**	14.76	0.3

Table 2: Comparison between our proposed model and other competitive models in the literature in terms of expected negative log likelihood on the test set computed as bits per dimension (BPD). Results from existing models are taken from the literature. "/" indicates results not reported or not applicable. Boldface denotes the best performance within each column, and blue text marks the second-best.

Model	**Type**	**ImageNet-32**	**ImageNet-64**	**ImageNet-128**
PixelCNN [77]	Autoregressive	3.83	3.57	/
Glow [42]	Flow	4.02	3.81	/
FLOW++ [27]	Flow	3.86	3.69	/
Sparse Transformer [9]	Autoregressive	/	3.44	/
Very deep VAE [8]	Autoencoder	3.80	3.52	/
Improved DDPM [54]	Diffusion	/	3.54	/
Routing Transformer [61]	Autoregressive	/	3.43	/
Flow Matching [49]	Flow	3.53	3.31	2.90
VDM [40]	Autoencoder	3.72	3.40	/
LP-PCDM [46]	Diffusion	3.52	3.12	2.91
W-PCDM [46]	Diffusion	3.32	2.95	2.64
Ours	Diffusion	**3.01**	**2.91**	**2.59**

5.1 Likelihood and Samples

Table 1 summarizes experimental results on CIFAR-10 and ImageNet-32 (more details in Appendix. B.5). At the request of one of the reviewers we also ran our model on additional data sets of higher resolution images baseline (Table. 2). On four V100 GPUs with CIFAR-10, our model trains slightly faster (2.34 vs. 2.04 iterations/sec) compared to VDM, due to the removal of the additional networks. Typically, achieving benchmark likelihood estimation performance requires several million training iterations, and the training process usually takes a week, or even a month or longer. Table 2 shows that we obtain state-of-the-art NLL on ImageNet-32/64/128 under directly comparable settings, while reducing training cost from millions of iterations in prior work to about 300 thousand.

Ablation Analysis and Additional Experiments Due to the high training cost, we conduct ablation studies only on CIFAR-10. In Table 7, we report both NLL and FID under different noise variance endpoints. We ablate the effect of the different IS weightings and reverse CDF embedding by comparing it to standard sinusoidal time encoding in Appendix. B.5. All models are trained for 300K steps with identical settings, except for the variance schedule. For fair comparison, we follow the VDM protocol [40] and evaluate models using the ELBO-based lower bound (see Appendix. B.6).

Our proposed bounds achieve state-of-the-art density estimation on ImageNet datasets and match the best reported results on CIFAR-10. We find that increasing the warm-up noise improves NLL but

Table 3: Comparison of NLL on CIFAR-10 and ImageNet-32 with different warm-up noise.

Likelihood Estimation Bound	CIFAR-10				ImageNet-32			
	Gaussian	Laplace	Logistic	Uniform	Gaussian	Laplace	Logistic	Uniform
ELBO	2.69	2.70	2.71	2.71	3.53	3.53	3.54	3.55
Our (SP + IS)	2.49	2.49	2.50	2.52	3.01	3.00	3.02	3.09
Our (VP + IS)	2.50	2.51	2.51	2.53	3.00	3.01	3.03	3.08

slightly degrades FID. This trade-off arises because larger noise suppresses pixel-level fluctuations and stabilizes training, while too little noise sufficiently regularize the likelihood objective, leading to worse likelihood estimates, despite improved sample quality. We provide details in the Appendix. C.

We also observe that the gap between ELBO and our IT-based bound enlarges with increasing noise levels. This discrepancy arises from the reconstruction loss used in ELBO [40, 55], which assumes conditional independence of pixels given the latent variable $\mathbf{y}_0$. As noise increases, the conditional distribution $p(\mathbf{y}_0|\mathbf{x})$ becomes more diffuse around its mode, thereby amplifying the approximation error. These findings highlight a fundamental intuition between likelihood accuracy and perceptual quality under different noise configurations. We leave the further study of this direction in the future.

5.2 Examining the Warm-up Noise Injection

We further examine the effect of different noise distributions, Gaussian, Laplace, logistic, and Uniform, each scaled to equal variance. As shown in Table 3, Gaussian noise performs best, followed by Laplace and logistic, while Uniform lags notably behind. This supports our intuition that heavier-tailed, exponential-family noises yield more stable training and improved likelihood estimation. Detailed theoretical discussion and additional analyses, including the connection to differential entropy, Fisher information and manifold hypothesis, are provided in Appendix G.

5.3 Lossless Progressive Coding

As shown in prior work [28, 40], likelihood-based generative models can be viewed as latent-variable models for neural lossless compression. We adopt this perspective and implement a Bits-Back Coding scheme [27] using our proposed model as the latent component. On CIFAR-10, our method achieves shorter average code lengths, measured in bits per dimension, compared to several strong baselines (see Table 4). We leave this avenue of research for further work.

Table 4: Lossless compression performance on CIFAR-10 in bits/dim.

Model	Compression Rate (bits/dim)
FLIF [66]	4.14
LBB [27]	3.12
IDF [33]	3.26
VDM [40]	2.72
ARDM [32]	2.71
W-PCDM [46]	**2.37**
Ours	2.57

6 Conclusion

Our generalized KL–Fisher relationship transforms noise injection from a theoretical consideration into a widely applicable practical strategy. This principled framework validates existing approaches employing non-Gaussian perturbations and offers new theoretical tools for tackling real-world generative modeling challenges. Our analysis shows that, in the small-noise regime, the score matching objective asymptotically approximates maximum likelihood. Minimizing this objective yields consistent improvements in likelihood across diverse noise schedules, variance settings, and datasets. When critically combined with importance sampling, our approach achieves on-par likelihood on CIFAR-10 and state-of-the-art likelihood on ImageNet datasets. Our results also motivate future exploration of information-theoretic objectives in generative modeling.

Limitations Our method improves likelihood estimation but does not construct a generative diffusion process under alternative noise. As a result, our method is limited to likelihood evaluation and cannot be directly used for sampling or generation. Dequantization bound, diffusion/drift coefficient and variance configurations are not fully explored. Due to resource limitations, we didn't explore tuning of hyperparameters and network architectures, which are left for future work. We leave full discussion and future extensions to the Appendix.

References

[1] Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *The Eleventh International Conference on Learning Representations*, 2023.

[2] Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-DPM: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. In *International Conference on Learning Representations*, 2022.

[3] Valentin De Bortoli. Convergence of denoising diffusion models under the manifold hypothesis. *Transactions on Machine Learning Research*, 2022. Expert Certification.

[4] Huanran Chen, Yinpeng Dong, Shitong Shao, Zhongkai Hao, Xiao Yang, Hang Su, and Jun Zhu. Diffusion models are certifiably robust classifiers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[5] Minhua Chen and John Lafferty. Mismatched estimation and relative entropy in vector Gaussian channels. In *2013 IEEE International Symposium on Information Theory*, pages 2845–2849. IEEE, 2013.

[6] Wei Chen, Shigui Li, Jiacheng Li, Junmei Yang, John Paisley, and Delu Zeng. Dequantified Diffusion-Schrödinger bridge for density ratio estimation. In *Forty-second International Conference on Machine Learning*, 2025.

[7] Zhaomin Chen, Chai Kiat Yeo, Bu Sung Lee, and Chiew Tong Lau. Autoencoder-based network anomaly detection. In *2018 Wireless telecommunications symposium (WTS)*, pages 1–5. IEEE, 2018.

[8] Rewon Child. Very deep VAEs generalize autoregressive models and can outperform them on images. In *International Conference on Learning Representations*, 2021.

[9] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[10] Kristy Choi, Chenlin Meng, Yang Song, and Stefano Ermon. Density ratio estimation via infinitesimal classification. In *International Conference on Artificial Intelligence and Statistics*, pages 2552–2573. PMLR, 2022.

[11] Jeremy Cohen, Elan Rosenfeld, and Zico Kolter. Certified adversarial robustness via randomized smoothing. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 1310–1320. PMLR, 09–15 Jun 2019.

[12] M. Costa. A new entropy power inequality. *IEEE Transactions on Information Theory*, 31(6):751–760, 1985.

[13] M. Costa and T. Cover. On the similarity of the entropy power inequality and the brunn-minkowski inequality (corresp.). *IEEE Transactions on Information Theory*, 30(6):837–839, 1984.

[14] Thomas M Cover. *Elements of information theory*. John Wiley & Sons, 1999.

[15] Zihang Dai, Zhilin Yang, Fan Yang, William W. Cohen, and Ruslan Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6513–6523, Red Hook, NY, USA, 2017. Curran Associates Inc.

[16] Valentin De Bortoli, Emile Mathieu, Michael Hutchinson, James Thornton, Yee Whye Teh, and Arnaud Doucet. Riemannian score-based generative modelling. *Advances in neural information processing systems*, 35:2406–2422, 2022.

[17] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *International Conference on Learning Representations*, 2017.

[18] Luyao Fan, Jiayang Zou, Jiayang Gao, and Jia Wang. Differential properties of information in jump-diffusion channels. *arXiv preprint arXiv:2501.05708*, 2025.

[19] R. A. Fisher. Theory of statistical estimation. *Mathematical Proceedings of the Cambridge Philosophical Society*, 22(5):700–725, 1925.

[20] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008.

[21] Ruiqi Gao, Erik Nijkamp, Diederik P Kingma, Zhen Xu, Andrew M Dai, and Ying Nian Wu. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7518–7528, 2020.

[22] Andrew Gelman and Xiao-Li Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, pages 163–185, 1998.

[23] Ishaan Gulrajani and Tatsunori B Hashimoto. Likelihood-based diffusion language models. *Advances in Neural Information Processing Systems*, 36:16693–16715, 2023.

[24] Dongning Guo. Relative entropy and score function: New information-estimation relationships through arbitrary additive perturbation. In *2009 IEEE International Symposium on Information Theory*, pages 814–818, 2009.

[25] Dongning Guo, S. Shamai, and S. Verdu. Mutual information and minimum mean-square error in Gaussian channels. *IEEE Transactions on Information Theory*, 51(4):1261–1282, 2005.

[26] Leonhard Helminger, Abdelaziz Djelouah, Markus Gross, and Christopher Schroers. Lossy image compression with normalizing flows. In *Neural Compression: From Information Theory to Applications – Workshop @ ICLR 2021*, 2021.

[27] Jonathan Ho, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2722–2730. PMLR, 09–15 Jun 2019.

[28] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[29] Jonathan Ho, Evan Lohn, and Pieter Abbeel. Compression with flows via local bits-back coding. *Advances in Neural Information Processing Systems*, 32, 2019.

[30] Yung-Han Ho, Chih-Chun Chan, Wen-Hsiao Peng, Hsueh-Ming Hang, and Marek Domański. Anfic: Image compression using augmented normalizing flows. *IEEE Open Journal of Circuits and Systems*, 2:613–626, 2021.

[31] Emiel Hoogeboom, Taco Cohen, and Jakub Mikolaj Tomczak. Learning discrete distributions by dequantization. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021.

[32] Emiel Hoogeboom, Alexey A. Gritsenko, Jasmijn Bastings, Ben Poole, Rianne van den Berg, and Tim Salimans. Autoregressive diffusion models. In *International Conference on Learning Representations*, 2022.

[33] Emiel Hoogeboom, Jorn Peters, Rianne Van Den Berg, and Max Welling. Integer discrete flows and lossless compression. *Advances in Neural Information Processing Systems*, 32, 2019.

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

[35] Jiantao Jiao, Kartik Venkat, and Tsachy Weissman. Relations between information and estimation in discrete-time Lévy channels. *IEEE Transactions on Information Theory*, 63(6):3579–3594, 2017.

[36] Norman L Johnson, Samuel Kotz, and Narayanaswamy Balakrishnan. *Continuous univariate distributions, volume 2*, volume 2. John wiley & sons, 1995.

[37] Tero Karras, Miika Aittala, Samuli Laine, and Timo Aila. Elucidating the design space of diffusion-based generative models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc.

[38] Dongjun Kim, Seungjae Shin, Kyungwoo Song, Wanmo Kang, and Il-Chul Moon. Soft truncation: A universal training technique of score-based diffusion model for high precision score estimation. In *International Conference on Machine Learning*, pages 11201–11228. PMLR, 2022.

[39] Diederik Kingma and Ruiqi Gao. Understanding diffusion objectives as the ELBO with simple data augmentation. *Advances in Neural Information Processing Systems*, 36:65484–65516, 2023.

[40] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021.

[41] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[42] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1×1 convolutions. *Advances in neural information processing systems*, 31, 2018.

[43] Xianghao Kong, Rob Brekelmans, and Greg Ver Steeg. Information-theoretic diffusion. In *The Eleventh International Conference on Learning Representations*, 2023.

[44] Luis A. Lastras-Montaño. Information theoretic lower bounds on negative log likelihood. In *International Conference on Learning Representations*, 2019.

[45] 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.

[46] Henry Li, Ronen Basri, and Yuval Kluger. Likelihood training of cascaded diffusion models via hierarchical volume-preserving maps. In *The Twelfth International Conference on Learning Representations*, 2024.

[47] Shigui Li, Wei Chen, and Delu Zeng. Evodiff: Entropy-aware variance optimized diffusion inference. *arXiv preprint arXiv:2509.26096*, 2025.

[48] Marten Lienen, Marcel Kollovieh, and Stephan Günnemann. Generative modeling with bayesian sample inference. *arXiv preprint arXiv:2502.07580*, 2025.

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

[50] Cheng Lu, Kaiwen Zheng, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Maximum likelihood training for score-based diffusion odes by high order denoising score matching. In *International Conference on Machine Learning*, pages 14429–14460. PMLR, 2022.

[51] Siwei Lyu. Interpretation and generalization of score matching. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, page 359–366, Arlington, Virginia, USA, 2009. AUAI Press.

[52] Chenlin Meng, Jiaming Song, Yang Song, Shengjia Zhao, and Stefano Ermon. Improved autoregressive modeling with distribution smoothing. In *International Conference on Learning Representations*, 2021.

[53] Krishna R Narayanan and Arun R Srinivasa. On the thermodynamic temperature of a general distribution. *arXiv preprint arXiv:0711.1460*, 2007.

[54] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.

[55] Beatrix Miranda Ginn Nielsen, Anders Christensen, Andrea Dittadi, and Ole Winther. Diffenc: Variational diffusion with a learned encoder. In *The Twelfth International Conference on Learning Representations*, 2024.

[56] Yosihiko Ogata. A monte carlo method for high dimensional integration. *Numerische Mathematik*, 55:137–157, 1989.

[57] Yidong Ouyang, Liyan Xie, Hongyuan Zha, and Guang Cheng. Transfer learning for diffusion models. *Advances in Neural Information Processing Systems*, 37:136962–136989, 2024.

[58] Daniel P Palomar and Sergio Verdú. Gradient of mutual information in linear vector gaussian channels. *IEEE Transactions on Information Theory*, 52(1):141–154, 2005.

[59] Miquel Payaro and Daniel P. Palomar. Hessian and concavity of mutual information, differential entropy, and entropy power in linear vector gaussian channels. *IEEE Transactions on Information Theory*, 55(8):3613–3628, 2009.

[60] Olivier Rioul. Information theoretic proofs of entropy power inequalities. *IEEE transactions on information theory*, 57(1):33–55, 2010.

[61] Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68, 2021.

[62] Subham Sahoo, Aaron Gokaslan, Christopher M De Sa, and Volodymyr Kuleshov. Diffusion models with learned adaptive noise. *Advances in Neural Information Processing Systems*, 37:105730–105779, 2024.

[63] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

[64] Yirong Shen, Matthias Seeger, and Andrew Ng. Fast Gaussian process regression using KD-trees. *Advances in neural information processing systems*, 18, 2005.

[65] Jyotirmai Singh, Samar Khanna, and James Burgess. Squeezed diffusion models. *arXiv preprint arXiv:2508.14871*, 2025.

[66] Jon Sneyers and Pieter Wuille. FLIF: Free lossless image format based on MANIAC compression. In *2016 IEEE international conference on image processing (ICIP)*, pages 66–70. IEEE, 2016.

[67] Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, page 2256–2265. JMLR.org, 2015.

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

[69] Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34:1415–1428, 2021.

[70] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020.

[71] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. In *International Conference on Learning Representations*, 2018.

[72] 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*, 2021.

[73] Lucas Theis, Tim Salimans, Matthew D Hoffman, and Fabian Mentzer. Lossy compression with Gaussian diffusion. *arXiv preprint arXiv:2206.08889*, 2022.

[74] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[75] J Townsend, T Bird, and D Barber. Practical lossless compression with latent variables using bits back coding. In *7th International Conference on Learning Representations, ICLR 2019*, volume 7. International Conference on Learning Representations (ICLR), 2019.

[76] Benigno Uria, Iain Murray, and Hugo Larochelle. Rnade: the real-valued neural autoregressive density-estimator. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2175–2183, Red Hook, NY, USA, 2013. Curran Associates Inc.

[77] Aäron Van Den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International conference on machine learning*, pages 1747–1756. PMLR, 2016.

[78] Sergio Verdú. Mismatched estimation and relative entropy. *IEEE Transactions on Information Theory*, 56(8):3712–3720, 2010.

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

[80] Ben Wan, Tianyi Zheng, Zhaoyu Chen, and Jia Wang. Enhancing the accuracy of generative adversarial networks with Fokker–Planck equations. *Neurocomputing*, 638:130158, 2025.

[81] Ben Wan, Tianyi Zheng, Zhaoyu Chen, Yuxiao Wang, and Jia Wang. Pruning for sparse diffusion models based on gradient flow. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2025.

[82] Andre Wibisono, Varun Jog, and Po-Ling Loh. Information and estimation in Fokker-Planck channels. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 2673–2677, 2017.

[83] Andre Wibisono, Yihong Wu, and Kaylee Yingxi Yang. Optimal score estimation via empirical Bayes smoothing. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 4958–4991. PMLR, 2024.

[84] Zhisheng Xiao, Qing Yan, and Yali Amit. Likelihood regret: an out-of-distribution detection score for variational auto-encoder. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[85] Yilun Xu, Ziming Liu, Max Tegmark, and Tommi S. Jaakkola. Poisson flow generative models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[86] Shahar Yadin, Noam Elata, and Tomer Michaeli. Classification diffusion models: Revitalizing density ratio estimation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[87] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

[88] Eunbi Yoon, Keehun Park, Sungwoong Kim, and Sungbin Lim. Score-based generative models with Lévy processes. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[89] Kaiwen Zheng, Yongxin Chen, Huayu Chen, Guande He, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Direct discriminative optimization: Your likelihood-based visual generative model is secretly a GAN discriminator. In *Forty-second International Conference on Machine Learning*, 2025.

[90] Kaiwen Zheng, Cheng Lu, Jianfei Chen, and Jun Zhu. Improved techniques for maximum likelihood estimation for diffusion ODEs. In *International Conference on Machine Learning*, pages 42363–42389. PMLR, 2023.

[91] Tianyi Zheng, Cong Geng, Peng-Tao Jiang, Ben Wan, Hao Zhang, Jinwei Chen, Jia Wang, and Bo Li. Non-uniform timestep sampling: Towards faster diffusion model training. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7036–7045, 2024.

[92] Tianyi Zheng, Jiayang Zou, Peng-Tao Jiang, Hao Zhang, Jinwei Chen, Jia Wang, and Bo Li. Bidirectional Beta-tuned diffusion model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[93] Jiayang Zou, Luyao Fan, Jiayang Gao, and Jia Wang. Convexity of mutual information along the Fokker-Planck flow. *arXiv preprint arXiv:2501.05094*, 2025.

[94] Jiayang Zou, Luyao Fan, Jiayang Gao, and Jia Wang. A revisit to rate-distortion theory via optimal weak transport. *arXiv preprint arXiv:2501.09362*, 2025.

Contents of the Appendix

A Preliminaries and Reviews

We summarize the key notations and assumptions used in our theorems. The data distribution is $p(\mathbf{x})$, and the model is $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is restricted to a parameter space $\boldsymbol{\Theta}$. The vector Gaussian channel follows $\mathbf{Y}_t = \alpha_t \mathbf{X} + \sigma_t \mathbf{N}$, where $\sigma_t : \mathbb{R} \to \mathbb{R}$ controls the time-dependent coefficient, $t \in [0, 1]$ represents the time horizon, and $\mathbf{N} \sim \mathcal{N}(\mathbf{n}; 0, \mathbf{I})$ is Gaussian noise. The input $\mathbf{x} \sim p(\mathbf{x})$, and the output $\mathbf{y}_t \sim p(\mathbf{y}_t)$, with both as column vectors of appropriate dimensions.

A.1 Notations

In this paper, we are working on the Euclidean space $\mathbb{R}^D$ for some $D \geq 1$. We denote the ℓ_2-inner product between vectors $\mathbf{u} = (u_1, \ldots, u_d), \mathbf{v} = (v_1, \ldots, v_d) \in \mathbb{R}^D$ as $\mathbf{u}^\top \mathbf{v} = \langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^D u_i v_i$.

For a symmetric matrix $\mathbf{A} \in \mathbb{R}^{D \times D}$, the notation $\mathbf{A} \succeq 0$ means A is positive semidefinite, i.e., $\mathbf{u}^\top \mathbf{A} \mathbf{u} \geq 0$ for all $\mathbf{u} \in \mathbb{R}^R$. For symmetric matrices $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{D \times D}$, the notation $\mathbf{A} \succeq \mathbf{B}$ means $\mathbf{A} - \mathbf{B} \succeq 0$ is positive semidefinite. Throughout, let $\mathbf{I} \in \mathbb{R}^{D \times D}$ denote the identity matrix.

For a differentiable function $f : \mathbb{R}^D \to \mathbb{R}$, let $\nabla f(\mathbf{x}) \in \mathbb{R}^D$ denote the gradient vector at $\mathbf{x} \in \mathbb{R}^D$ of the partial derivatives: $(\nabla f(\mathbf{x}))_i = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}_i}$. Let $\nabla^2 f(\mathbf{x}) \in \mathbb{R}^{D \times D}$ be the Hessian matrix of second partial derivatives: $(\nabla^2 f(\mathbf{x}))_{i,j} = \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}_i \partial \mathbf{x}_j}$. Let $\Delta f(\mathbf{x}) = \mathrm{Tr}(\nabla^2 f(\mathbf{x}))$ be the Laplacian. We use $\mathcal{C}$ to denote all continuous functions, and let $\mathcal{C}^k$ denote the family of functions with continuous k-th order derivatives.

For a vector field $\mathbf{v} : \mathbb{R}^D \to \mathbb{R}^D$ with $\mathbf{v}(\mathbf{x}) = (\mathbf{v}_1(\mathbf{x}), \ldots, \mathbf{v}_d(\mathbf{x})) \in \mathbb{R}^D$, let $\nabla \mathbf{v} : \mathbb{R}^D \to \mathbb{R}^{D \times D}$ be the Jacobian matrix of mixed partial derivatives: $(\nabla \mathbf{v}(\mathbf{x}))_{i,j} = \frac{\partial \mathbf{v}_i(\mathbf{x})}{\partial \mathbf{x}_j}$. Let $\nabla \cdot \mathbf{v} : \mathbb{R}^D \to \mathbb{R}$ be the divergence of $\mathbf{v}$, defined by

$$(\nabla \cdot \mathbf{v})(\mathbf{x}) = \sum_{i=1}^D \frac{\partial \mathbf{v}_i(\mathbf{x})}{\partial \mathbf{x}_i} = \mathrm{Tr}(\nabla \mathbf{v}(\mathbf{x})).$$

Let $V_r \subset \mathbb{R}^D$ be the region (an D-dimensional ball) bounded by the closed, piecewise-smooth, oriented surface S_r, which is the *D-sphere* of radius r centered at the origin. At any point $\mathbf{y} \in S_r$, the symbol $\mathbf{e}_{S_r}(\mathbf{y})$ denotes the outward-pointing unit normal vector to S_r. Under the notation $d\mathbf{s}_r = \|d\mathbf{s}_r\| \mathbf{e}_{S_r}(\mathbf{y})$.

A.2 Definitions

Let $\mathcal{P}(\mathbb{R}^D)$ denote the space of probability distributions ρ over $\mathbb{R}^D$ which are absolutely continuous with respect to the Lebesgue measure and have a finite second moment $\mathbb{E}_\rho[\|\mathbf{X}\|^2] < \infty$. We identify a probability distribution $\rho \in \mathcal{P}(\mathbb{R}^D)$ with its probability density function with respect to the Lebesgue measure, which we also denote by ρ: $\mathbb{R}^D \to \mathbb{R}$, so $\rho(x) > 0$ and $\int_{\mathbb{R}^D} \rho(x) dx = 1$.

We say ρ is absolutely continuous with respect to another distribution ν, denoted by $\rho \ll \nu$, if $\nu(\mathbf{A}) = 0$ implies $\rho(\mathbf{A}) = 0$ for any $\mathbf{A} \subseteq \mathbb{R}^D$; if ρ and ν both have density functions, then $\rho \ll \nu$ means $\nu(\mathbf{x}) = 0$ implies $\rho(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathbb{R}^D$.

Definition 1. *Let $\mathcal{H}(\cdot) : \mathcal{P}(\mathbb{R}^D) \to \mathbb{R}$ be the differential Shannon entropy:*

$$\mathcal{H}(\rho) = -\mathbb{E}_\rho[\log \rho] = -\int_{\mathbb{R}^D} \rho(\mathbf{x}) \log \rho(\mathbf{x}) \, d\mathbf{x}. \tag{14}$$

Definition 2. *Let $\mathcal{J}(\cdot) : \mathcal{P}(\mathbb{R}^D) \to \mathbb{R}$ be the Fisher information:*

$$\mathcal{J}(\rho) = \mathbb{E}_\rho \left[\|\nabla \log \rho\|^2 \right] = -\mathbb{E}_\rho[\Delta \log \rho], \tag{15}$$

and we define $\mathcal{J}(\rho) = +\infty$ if ρ does not have a differentiable density. The second equality in the definition of $\mathcal{J}(\rho)$ above follows by integration by parts.

Note that the expression in (15) is a special case (with respect to a translation parameter) which does not involve an explicit parameter as in its most general definition.[3]

[3]The parameterized Fisher information matrix is defined with respect to a parameter $\boldsymbol{\theta} \in \boldsymbol{\Theta}$ by $\mathcal{J}(\rho) \equiv \mathbb{E}_\rho[\nabla_{\boldsymbol{\theta}} \log \rho(\mathbf{x}; \boldsymbol{\theta}) \, \nabla_{\boldsymbol{\theta}}^\top \log \rho(\mathbf{x}; \boldsymbol{\theta})]$.

Definition 3. *For probability distributions $\rho \ll \nu$ on $\mathbb{R}^D$, the Kullback-Leibler (KL) divergence or the relative entropy of ρ with respect to ν is defined by:*

$$D_{KL}(\rho\|\nu) = \mathbb{E}_\rho\left[\log\frac{\rho}{\nu}\right] = \int_{\mathbb{R}^D} \rho(\mathbf{x}) \log\frac{\rho(\mathbf{x})}{\nu(\mathbf{x})}\, d\mathbf{x}. \tag{16}$$

Definition 4. *For probability distributions $\rho \ll \nu$ on $\mathbb{R}^D$, the mismatched entropy or the cross entropy of ρ with respect to ν is defined by:*

$$\mathcal{H}(\rho,\nu) = -\mathbb{E}_\rho\left[\log\nu\right] = \int_{\mathbb{R}^D} \rho(\mathbf{x}) \frac{1}{\log\nu(\mathbf{x})} d\mathbf{x}. \tag{17}$$

Definition 5. *If ρ and ν have differentiable density functions, then the relative Fisher information of ρ with respect to ν is defined by:*

$$I(\rho\|\nu) = \mathbb{E}_\rho\left[\left\|\nabla\log\frac{\rho}{\nu}\right\|^2\right] = \int_{\mathbb{R}^D} \rho(\mathbf{x}) \left\|\nabla\log\frac{\rho(\mathbf{x})}{\nu(\mathbf{x})}\right\|^2 d\mathbf{x}. \tag{18}$$

We recall that the KL divergence, $D_{\mathrm{KL}}(\cdot\|\cdot)$, corresponds to the Bregman divergence of the negative entropy. Similarly, the relative Fisher information (RFI), $I(\cdot\|\cdot)$, can be viewed as the Bregman divergence of the Fisher information. Moreover, half of the RFI is equivalent to score matching, as defined in [34].

A.3 Standard Diffusion Models

Consider a Gaussian diffusion process [28], which starts from clean data $\mathbf{x}$ and defines a sequence of progressively noisier versions, denoted by channel outputs $\mathbf{y}_t$, where t runs from 0 (least noisy) to 1 (most noisy). In the sampling process, given T, we uniformly discretise the time interval into T timesteps, each of width $1/T$. Let $t(i) = i/T$ denote the current timestep and $s(i) = (i-1)/T$ the preceding one [40].

Forward Process. The forward process is defined by a conditional Gaussian distribution:

$$p(\mathbf{y}_t|\mathbf{x}) = \mathcal{N}(\alpha_t\mathbf{x}, \sigma_t^2\mathbf{I}),$$

and, by the Markov property,

$$p(\mathbf{y}_t|\mathbf{y}_s) = \mathcal{N}(\alpha_{t|s}\mathbf{y}_s, \sigma_{t|s}^2\mathbf{I}),$$

where

$$\alpha_{t|s} = \frac{\alpha_t}{\alpha_s}, \quad \sigma_{t|s}^2 = \sigma_t^2 - \frac{\alpha_{t|s}^2}{\sigma_s^2}.$$

Reverse Process. As established in prior work [28, 40, 68], the conditional distribution $p(\mathbf{y}_s|\mathbf{y}_t,\mathbf{x})$ is also Gaussian:

$$p(\mathbf{y}_s|\mathbf{y}_t,\mathbf{x}) = \mathcal{N}\left(\frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2}\mathbf{y}_t + \frac{\sigma_{t|s}^2\alpha_s}{\sigma_t^2}\mathbf{x},\ \frac{\sigma_s^2\sigma_{t|s}^2}{\sigma_t^2}\mathbf{I}\right).$$

As the ground truth $\mathbf{x}$ is not available during the reverse process, it is approximated by a neural network $\hat{\mathbf{x}}(\mathbf{y}_t, t; \boldsymbol{\theta})$, parameterised by $\boldsymbol{\theta}$. The learned reverse kernel becomes:

$$p(\mathbf{y}_s|\mathbf{y}_t;\boldsymbol{\theta}) = \mathcal{N}\left(\frac{\alpha_{t|s}\sigma_s^2}{\sigma_t^2}\mathbf{y}_t + \frac{\sigma_{t|s}^2\alpha_s}{\sigma_t^2}\hat{\mathbf{x}}(\mathbf{y}_t,t;\boldsymbol{\theta}),\ \frac{\sigma_s^2\sigma_{t|s}^2}{\sigma_t^2}\mathbf{I}\right).$$

A.4 Lemmas

In this section, we introduce all Lemmas in order to prove the Theorems in the next section.

Lemma 1. *Let $\mathbf{X} \in \mathbb{R}^D$ be a random vector with density $p(\mathbf{x})$. Suppose $\boldsymbol{\Psi} \in \mathbb{R}^D$ is an arbitrary independent random vector with zero mean and identity matrix $\mathbf{I}$. Define $\tilde{\mathbf{X}} = \alpha_t\mathbf{X} + \sigma_t\,\boldsymbol{\Psi}$, and let $p(\tilde{\mathbf{x}})$ denote the density of $\tilde{\mathbf{X}}$. Then for every $\tilde{\mathbf{x}} \in \mathbb{R}^D$, as $\sigma_t^2 \to 0^+$, we have:*

$$\frac{d}{d\sigma_t^2}p(\tilde{\mathbf{x}})\bigg|_{\sigma_t^2=0^+} = \frac{1}{2}\Delta p(\mathbf{x}), \tag{19}$$

where $\Delta_{\mathbf{x}} = \sum_{j=1}^{D} \frac{\partial^2}{\partial (x_j)^2}$ *is the usual Laplacian in* $\mathbb{R}^D$.

Formula (19) allows the derivative w.r.t. the energy of the perturbation σ_t^2 to be transformed to the second derivative of the original pdf. In what followed we provide the proof for Lemma 1 which is slightly different than that in [53]. Note that Lemma 1 does not require the distribution of the perturbation to be symmetric as is required in [53].

Proof. Let $p(\tilde{\mathbf{x}})$ denote the probability density function of the random vector $\tilde{\mathbf{X}}$, and define its characteristic function as follows:

$$\phi(\mathbf{k}, \sigma_t^2) = \mathbb{E}\left[\exp\left(i\,\mathbf{k}^\top \tilde{\mathbf{X}}\right)\right], \quad \mathbf{k} \in \mathbb{R}^D. \tag{20}$$

Given that $\tilde{\mathbf{X}}$ is defined by

$$\tilde{\mathbf{X}} = \alpha_t \mathbf{X} + \sigma_t \mathbf{\Psi}, \tag{21}$$

where $\mathbf{X}$ and $\mathbf{\Psi}$ are independent, the characteristic function factorises as follows:

$$\begin{aligned}
\phi(\mathbf{k}, \sigma_t^2) &= \mathbb{E}\left[e^{i\mathbf{k}^\top(\alpha_t \mathbf{X} + \sigma_t \mathbf{\Psi})}\right] && (22)\\
&= \mathbb{E}\left[e^{i\mathbf{k}^\top \alpha_t \mathbf{X}}\right] \mathbb{E}\left[e^{i\mathbf{k}^\top \sigma_t \mathbf{\Psi}}\right] && (23)\\
&= \underbrace{\phi(\alpha_t \mathbf{k}, 0)}_{\text{characteristic function of } \mathbf{X}} \times \underbrace{\mathbb{E}\left[e^{i\mathbf{k}^\top \sigma_t \mathbf{\Psi}}\right]}_{\text{characteristic function of } \sigma_t \mathbf{\Psi}}. && (24)
\end{aligned}$$

Expanding the exponential function via Taylor's theorem gives

$$e^{i\mathbf{k}^\top \sigma_t \mathbf{\Psi}} = 1 + i\sigma_t(\mathbf{k}^\top \mathbf{\Psi}) - \frac{(\sigma_t \mathbf{k}^\top \mathbf{\Psi})^2}{2!} + O(\sigma_t^3). \tag{25}$$

Since $\mathbb{E}[\mathbf{\Psi}] = \mathbf{0}$ and $\mathrm{Cov}[\mathbf{\Psi}] = \mathbf{I}$, it follows that

$$\begin{aligned}
\mathbb{E}[\mathbf{k}^\top \mathbf{\Psi}] &= 0, && (26)\\
\mathbb{E}[(\mathbf{k}^\top \mathbf{\Psi})^2] &= \mathbf{k}^\top \mathbb{E}[\mathbf{\Psi}\mathbf{\Psi}^\top]\mathbf{k} = \|\mathbf{k}\|^2. && (27)
\end{aligned}$$

Taking expectations, we obtain

$$\mathbb{E}\left[e^{i\mathbf{k}^\top \sigma_t \mathbf{\Psi}}\right] = 1 - \frac{\sigma_t^2}{2}\|\mathbf{k}\|^2 + o(\sigma_t^2). \tag{28}$$

Thus, the characteristic function satisfies

$$\phi(\mathbf{k}, \sigma_t^2) = \phi(\alpha_t \mathbf{k}, 0)\left(1 - \frac{\sigma_t^2}{2}\|\mathbf{k}\|^2 + o(\sigma_t^2)\right). \tag{29}$$

Recalling the inverse Fourier transform, the probability density function is given by

$$p_{\sigma_t^2}(\tilde{\mathbf{x}}) = \frac{1}{(2\pi)^D} \int_{\mathbb{R}^D} e^{-i\mathbf{k}^\top \tilde{\mathbf{x}}} \phi(\mathbf{k}, \sigma_t^2) d\mathbf{k}. \tag{30}$$

Substituting the expansion of $\phi(\mathbf{k}, \sigma_t^2)$, we obtain

$$p_{\sigma_t^2}(\tilde{\mathbf{x}}) = \frac{1}{(2\pi)^D} \int_{\mathbb{R}^D} e^{-i\mathbf{k}^\top \tilde{\mathbf{x}}} \phi(\alpha_t \mathbf{k}, 0)\left(1 - \frac{\sigma_t^2}{2}\|\mathbf{k}\|^2 + o(\sigma_t^2)\right) d\mathbf{k}. \tag{31}$$

Since

$$p_0(\tilde{\mathbf{x}}) = \frac{1}{(2\pi)^D} \int_{\mathbb{R}^D} e^{-i\mathbf{k}^\top \tilde{\mathbf{x}}} \phi(\alpha_t \mathbf{k}, 0) d\mathbf{k}, \tag{32}$$

it follows that the difference $p_{\sigma_t^2}(\tilde{\mathbf{x}}) - p_0(\tilde{\mathbf{x}})$ corresponds to multiplying $\phi(\alpha_t \mathbf{k}, 0)$ by

$$-\frac{\sigma_t^2}{2}\|\mathbf{k}\|^2 + o(\sigma_t^2).$$

A standard result from Fourier analysis states that multiplication by $-\|\mathbf{k}\|^2$ in the Fourier domain corresponds to the application of the Laplacian Δ in the spatial domain, defined as

$$\Delta_{\mathbf{x}} = \sum_{j=1}^{D} \frac{\partial^2}{\partial x_j^2}.$$

Thus, for small σ_t^2, we obtain

$$p_{\sigma_t^2}(\tilde{\mathbf{x}}) = p_0(\tilde{\mathbf{x}}) + \frac{\sigma_t^2}{2}\Delta_{\tilde{\mathbf{x}}} p_0(\tilde{\mathbf{x}}) + o(\sigma_t^2). \tag{33}$$

Since $p_0(\tilde{\mathbf{x}})$ corresponds to $p(\mathbf{x})$ in the absence of noise, this completes the proof of Lemma 1. □

Lemma 2 (Vanishing boundary flux)**.** *For an arbitrary input distribution $p(\mathbf{x})$, an assumed input distribution $q(\hat{\mathbf{x}};\boldsymbol{\theta})$ and a vector Gaussian channel $p(\mathbf{y}_t|\mathbf{x}) = \mathcal{N}(\alpha_t\mathbf{x}, \sigma_t^2\mathbf{I})$ in (1), denote the corresponding output densities by $p(\mathbf{y}_t)$ and $q(\mathbf{y}_t;\boldsymbol{\theta})$ at time $t \in [0,1]$. For $r > 0$, let $B_r := \{\mathbf{y} \in \mathbb{R}^D : \|\mathbf{y}\| \le r\}$ be the closed D-ball and let $S_r := \partial B_r$ be the $(D-1)$-sphere with outward unit normal $\mathbf{e}_{S_r}(\mathbf{y}) = \mathbf{y}/\|\mathbf{y}\|$. Assume $p(\mathbf{y}_t)$ and $q(\mathbf{y}_t;\boldsymbol{\theta})$ are $\mathcal{C}^1$ with well-defined $\nabla_{\mathbf{y}}$ and $\Delta_{\mathbf{y}}$, and that $\mathbb{E}_p\big[\log q(\mathbf{y}_t;\boldsymbol{\theta})\big] < \infty$. Then the boundary integrals*

$$L_1 := \lim_{r\to\infty} \int_{S_r} \log q(\mathbf{y}_t;\boldsymbol{\theta})\, \nabla_{\mathbf{y}_t} p(\mathbf{y}_t) \cdot \mathbf{e}_{S_r}(\mathbf{y}_t)\, \mathrm{d}S = 0, \tag{34}$$

$$L_2 := \lim_{r\to\infty} \int_{S_r} \frac{1}{2}\, \frac{p(\mathbf{y}_t)}{q(\mathbf{y}_t;\boldsymbol{\theta})}\, \nabla_{\mathbf{y}_t} q(\mathbf{y}_t;\boldsymbol{\theta}) \cdot \mathbf{e}_{S_r}(\mathbf{y}_t)\, \mathrm{d}S = 0. \tag{35}$$

Proof. First, note that the limits L_1 and L_2 obviously exist, as each of these limits can be expressed as the sum of two converging integrals. We prove (34); the argument for (35) is analogous. For brevity, omit the subscript t and write $\mathbf{y}$ for $\mathbf{y}_t$. Let $\mathbf{e}_{S_r}(\mathbf{y})$ to be unit vector normal to S_r at the point $\mathbf{y}$. Under this notation $d\mathbf{s_r} = ||d\mathbf{s_r}||\mathbf{e}_{S_r}(\mathbf{y})$. We integrate over $r \ge 0$ the surface integral in (34) and apply Green's identity to find the relations. Set

$$f(r) := \int_{S_r} \log q(\mathbf{y})\, \nabla p(\mathbf{y}) \cdot d\mathbf{s_r}.$$

By the coarea formula (spherical coordinates),

$$\int_0^\infty f(r)\, \mathrm{d}r = \int_{\mathbb{R}^D} \nabla p(\mathbf{y}) \cdot \left(\log q(\mathbf{y})\, \mathbf{e}_{S_r}(\mathbf{y})\right)\, \mathrm{d}\mathbf{y}. \tag{36}$$

Using the product rule and the divergence theorem on B_r,

$$\begin{aligned}
&\int_{B_r} \nabla p(\mathbf{y}) \cdot \big(\log q(\mathbf{y})\, \mathbf{e}_{S_r}(\mathbf{y})\big)\, \mathrm{d}\mathbf{y} \\
&= \lim_{r\to\infty} \int_{S_r} p(\mathbf{y}) \log q(\mathbf{y})\, \mathbf{e}_{S_r}(\mathbf{y}) \cdot d\mathbf{s_r} \;-\; \int_{B_r} p(\mathbf{y})\, \nabla \cdot \big(\log q(\mathbf{y})\, \mathbf{e}_{S_r}(\mathbf{y})\big)\, \mathrm{d}\mathbf{y}.
\end{aligned} \tag{37}$$

Letting $r \to \infty$ we bound the two terms on the right-hand side.

First term. By the coarea formula,

$$\int_r^\infty \int_{S_r} \big|p(\mathbf{y}) \log q(\mathbf{y})\big|\, \mathrm{d}S\, \mathrm{d}r = \int_{\|\mathbf{y}\|>r} \big|p(\mathbf{y}) \log q(\mathbf{y})\big|\, \mathrm{d}\mathbf{y} \xrightarrow[r\to\infty]{} 0,$$

because $\mathbb{E}_p[|\log q(\mathbf{y})|] < \infty$. Hence $\int_{S_r} p(\mathbf{y}) \log q(\mathbf{y})\, \mathrm{d}S \to 0$ along the full sequence $r \to \infty$ (e.g., via a Cesàro argument).

Second term. Now we note that the absolute value of the divergence in the second term satisfies the relation

$$|\nabla \cdot (\log q(\mathbf{y})\mathbf{e}_{S_r}(\mathbf{y}))| = \frac{|\nabla q(\mathbf{y}) \cdot \mathbf{e}_{S_r}(\mathbf{y})|}{q(\mathbf{y})} \leq \frac{\|\nabla q(\mathbf{y})\|}{q(\mathbf{y})} \tag{38}$$

Hence, we have

$$\begin{aligned}
\frac{\|\nabla q(\mathbf{y})\|}{q(\mathbf{y})} &= \left(\sum_{i=1}^{n}\left[\int_{\mathbb{R}^D} \frac{q(\mathbf{x})}{q(\mathbf{y})}(2\pi\sigma_t^2)^{-\frac{D}{2}}\big(\frac{\|y_i-\alpha_t x_i\|}{\sigma_t^2}\big)\exp\big(-\frac{\|\mathbf{y}_t-\alpha_t\mathbf{x}\|^2}{2\sigma_t^2}\big)\,d\mathbf{x}.\right]^2\right)^{\frac{1}{2}} \\
&= \left(\sum_{i=1}^{n}\left[\mathbb{E}\left(\frac{Y_i-\alpha_t X_i}{\sigma_t^2}\Big|\mathbf{Y}=\mathbf{y}\right)\right]^2\right)^{\frac{1}{2}} \\
&\leq \left(\sum_{i=1}^{n}\left(\mathbb{E}\left(\left(\frac{Y_i-\alpha_t X_i}{\sigma_t^2}\right)^2\Big|\mathbf{Y}=\mathbf{y}\right)\right)\right)^{\frac{1}{2}} \\
&= \left(\mathbb{E}\left(\left(\frac{\|\mathbf{Y}-\alpha_t\mathbf{X}\|}{\sigma_t^2}\right)^2\Big|\mathbf{Y}=\mathbf{y}\right)\right)^{\frac{1}{2}}.
\end{aligned} \tag{39}$$

Integrating w.r.t. $p(\mathbf{y})\,\mathrm{d}\mathbf{y}$ and applying Jensen, we can write the chain of inequalities

$$\begin{aligned}
\int_{\mathbb{R}^n} p(\mathbf{y})\frac{\|\nabla q(\mathbf{y})\|}{q(\mathbf{y})}d\mathbf{y} &\leq \mathbb{E}_p\left\{\left(\mathbb{E}\left(\left(\frac{\|\mathbf{Y}-\alpha_t\mathbf{X}\|}{\sigma_t^2}\right)^2\Big|\mathbf{Y}=\mathbf{y}\right)\right)^{\frac{1}{2}}\right\} \\
&\leq \left\{\mathbb{E}_p\left(\mathbb{E}\left(\left(\frac{\|\mathbf{Y}-\alpha_t\mathbf{X}\|}{\sigma_t^2}\right)^2\Big|\mathbf{Y}=\mathbf{y}\right)\right)\right\}^{\frac{1}{2}} \\
&= \left[\mathbb{E}_p\left(\left(\frac{\|\mathbf{N}\|}{\sigma_t^2}\right)^2\right)\right]^{\frac{1}{2}} \\
&< \infty,
\end{aligned} \tag{40}$$

which means that

$$\mathbb{E}_p\big[\|\nabla \log q(\mathbf{y})\|\big] \leq \Big(\mathbb{E}_p\big[\|\mathbf{N}\|^2/\sigma_t^4\big]\Big)^{1/2} < \infty.$$

Therefore the right-hand side of (37) is finite, so

$$\int_0^\infty f(r)\,\mathrm{d}r < \infty.$$

Finally, f is locally absolutely continuous in r (by the smoothness of p and q and the coarea formula), hence $\lim_{r\to\infty} f(r)$ exists. Since $\int_0^\infty f(r)\,\mathrm{d}r < \infty$, this limit must equal 0. Thus $L_1 = 0$. The proof of $L_2 = 0$ follows by the same steps with $\frac{1}{2}\,\frac{p}{q}\nabla q$ in place of $\log q\,\nabla p$. □

Lemma 3. *Fix $t \in [0,1]$ and consider the Gaussian channel in* (1)*. Let $p(\mathbf{y}_t)$ and $q(\mathbf{y}_t;\boldsymbol{\theta})$ be the output densities induced by inputs $p(\mathbf{x})$ and $q(\mathbf{x};\boldsymbol{\theta})$, respectively. Assume $p(\cdot)$ and $q(\cdot;\boldsymbol{\theta})$ are sufficiently smooth and integrable so that differentiation under the integral sign is justified and the boundary terms in Lemma 2 vanish. Then the derivative of the mismatched output cross-entropy*

$$\mathcal{H}\big(p(\mathbf{y}_t), q(\mathbf{y}_t;\boldsymbol{\theta})\big) := -\int_{\mathbb{R}^D} p(\mathbf{y}_t)\,\log q(\mathbf{y}_t;\boldsymbol{\theta})\,\mathrm{d}\mathbf{y}_t$$

with respect to the noise variance σ_t^2 is

$$\frac{\mathrm{d}}{\mathrm{d}\sigma_t^2}\,\mathcal{H}\big(p(\mathbf{y}_t), q(\mathbf{y}_t;\boldsymbol{\theta})\big) = \frac{1}{2}\int_{\mathbb{R}^D}\left(\frac{\nabla q(\mathbf{y}_t;\boldsymbol{\theta})\cdot\nabla p(\mathbf{y}_t)}{q(\mathbf{y}_t;\boldsymbol{\theta})} \;+\; \nabla\left(\frac{p(\mathbf{y}_t)}{q(\mathbf{y}_t;\boldsymbol{\theta})}\right)\cdot\nabla q(\mathbf{y}_t;\boldsymbol{\theta})\right)\mathrm{d}\mathbf{y}_t, \tag{41}$$

where $\mathbf{a}\cdot\mathbf{b} = \langle\mathbf{a},\mathbf{b}\rangle$ denotes the Euclidean inner product.

Proof. By definition of cross-entropy in Def. 4,

$$\mathcal{H}\big(p(\mathbf{y}_t), q(\mathbf{y}_t;\boldsymbol{\theta})\big) = -\int_{\mathbb{R}^D} p(\mathbf{y}_t)\, \log q(\mathbf{y}_t;\boldsymbol{\theta})\, \mathrm{d}\mathbf{y}_t.$$

Differentiating w.r.t. σ_t^2 (justified by the assumed regularity) yields

$$\frac{\mathrm{d}}{\mathrm{d}\sigma_t^2}\Big(-\int p\log q\Big) = -\int \log q\, \frac{\partial p}{\partial \sigma_t^2}\, \mathrm{d}\mathbf{y}_t \;-\; \int p\, \frac{\partial}{\partial \sigma_t^2}\log q\, \mathrm{d}\mathbf{y}_t, \tag{42}$$

where we abbreviate $p = p(\mathbf{y}_t)$ and $q = q(\mathbf{y}_t;\boldsymbol{\theta})$.

Under Gaussian smoothing (heat equation),

$$\frac{\partial p}{\partial \sigma_t^2} = \frac{1}{2}\,\Delta_{\mathbf{y}_t} p, \qquad \frac{\partial q}{\partial \sigma_t^2} = \frac{1}{2}\,\Delta_{\mathbf{y}_t} q, \qquad \frac{\partial}{\partial \sigma_t^2}\log q = \frac{1}{q}\frac{\partial q}{\partial \sigma_t^2} = \frac{1}{2}\,\frac{\Delta_{\mathbf{y}_t} q}{q}.$$

We now recall Green's identity: If $\boldsymbol{\phi}(\mathbf{x})$ and $\boldsymbol{\psi}(\mathbf{x})$ are twice continuously differentiable functions in $\mathbb{R}^D$ and V is any set bounded by a piecewise smooth, closed, oriented surface S in $\mathbb{R}^D$, then

$$\int_V \boldsymbol{\phi}\nabla^2\boldsymbol{\psi}\, dV = \int_S \boldsymbol{\phi}\nabla\boldsymbol{\psi}\cdot d\mathbf{s} - \int_V \nabla\boldsymbol{\phi}\cdot\nabla\boldsymbol{\psi}\, dV, \tag{43}$$

where $\nabla\boldsymbol{\phi}$ denotes the gradient of $\boldsymbol{\phi}$, $d\mathbf{s}$ denotes the elementary area vector, and $\nabla\boldsymbol{\phi}\cdot d\mathbf{s}$ is the inner product of these two vectors. This identity plays the role of integration by parts in $\mathbb{R}^D$.

To apply Green's identity to (42), we let V_r be the D-sphere of radius r centered at the origin and having surface S_r. Then we use Green's identity on V_r and S_r with $\boldsymbol{\phi}(\mathbf{y}_t) = \log p(\mathbf{y}_t)$ and $\boldsymbol{\psi}(\mathbf{y}_t) = p(\mathbf{y}_t)$ and take the limit as $r \to \infty$. In Lemma 2 the surface integral over S_r is shown to vanish in the limit. Hence, by applying heat equation [12] and Green's identity for the first term on right hide side, we obtain:

$$-\int \log q(\mathbf{y}_t;\boldsymbol{\theta})\frac{d}{d\sigma_t^2}p(\mathbf{y}_t)\, d\mathbf{y}_t = -\frac{1}{2}\int \log q(\mathbf{y}_t;\boldsymbol{\theta})\Delta_{\mathbf{y}_t}p(\mathbf{y}_t)\, d\mathbf{y}_t \tag{44}$$

$$= \frac{1}{2}\int \nabla_{\mathbf{y}_t}\log q(\mathbf{y}_t;\boldsymbol{\theta})\cdot\nabla_{\mathbf{y}_t}p(\mathbf{y}_t)\, d\mathbf{y}_t \tag{45}$$

$$= \frac{1}{2}\int_{\mathbb{R}^D} \frac{\nabla q(\mathbf{y}_t;\boldsymbol{\theta})\cdot\nabla p(\mathbf{y}_t)}{q(\mathbf{y}_t;\boldsymbol{\theta})}\, d\mathbf{y}_t, \tag{46}$$

where we used integration by parts in high dimension and the vanishing boundary terms from Lemma 2. For the second term,

$$-\int p(\mathbf{y}_t)\frac{d}{d\sigma_t^2}\log q(\mathbf{y}_t;\boldsymbol{\theta})\, d\mathbf{y}_t = -\int p(\mathbf{y}_t)\frac{\nabla_{\sigma_t^2} q(\mathbf{y}_t;\boldsymbol{\theta})}{q(\mathbf{y}_t;\boldsymbol{\theta})}\, d\mathbf{y}_t \tag{47}$$

$$= -\frac{1}{2}\int \left(\frac{p(\mathbf{y}_t)}{q(\mathbf{y}_t;\boldsymbol{\theta})}\right)\Delta_{\mathbf{y}_t} q(\mathbf{y}_t;\boldsymbol{\theta})\, d\mathbf{y}_t \tag{48}$$

$$= \frac{1}{2}\int_{\mathbb{R}^D} \nabla\left(\frac{p(\mathbf{y}_t)}{q(\mathbf{y}_t)}\right)\cdot\nabla q(\mathbf{y}_t)\, d\mathbf{y}_t, \tag{49}$$

where we again applied Green's identity and used the vanishing of the boundary flux.

Summing the two contributions yields (41). □

Lemma 4. *Let $(X, \mathcal{S}, \mu_i)$ be probability spaces, and let $T : X \to Y$ be a measurable transformation inducing probability measures ν_i on $(Y, \mathcal{T})$ such that*

$$\nu_i(G) = \mu_i(T^{-1}(G)), \quad \forall G \in \mathcal{T}, \quad i = 1, 2.$$

Denote the Radon-Nikodym derivatives of ν_i with respect to a common reference measure γ as

$$g_i(y) = \frac{d\nu_i}{d\gamma}.$$

Then, the KL divergence remains invariant under the transformation T, i.e.,

$$D_{\mathrm{KL}}(\nu_1||\nu_2) = D_{\mathrm{KL}}(\mu_1||\mu_2),$$

where

$$D_{\mathrm{KL}}(\nu_1||\nu_2) = \int_Y g_1(y) \log \frac{g_1(y)}{g_2(y)} d\gamma(y).$$

Proof. Let λ be a reference measure on X such that μ_i has densities f_i with respect to λ, i.e.,

$$f_i(x) = \frac{d\mu_i}{d\lambda}(x).$$

By the change of variables under T, the density functions transform as

$$g_i(y) = \frac{d\nu_i}{d\gamma}(y) = \frac{d\mu_i}{d\lambda}(T^{-1}(y)) J_T^{-1}(y),$$

where $J_T(y) = \left|\det \frac{dT}{dx}\right|$ is the Jacobian determinant of T.

Substituting into the definition of KL divergence:

$$D_{\mathrm{KL}}(\nu_1||\nu_2) = \int_Y g_1(y) \log \frac{g_1(y)}{g_2(y)} d\gamma(y),$$

we expand:

$$= \int_Y \left(f_1(T^{-1}(y)) J_T^{-1}\right) \log \frac{f_1(T^{-1}(y)) J_T^{-1}}{f_2(T^{-1}(y)) J_T^{-1}} d\gamma(y).$$

Since the Jacobian terms cancel, we obtain:

$$= \int_Y f_1(T^{-1}(y)) \log \frac{f_1(T^{-1}(y))}{f_2(T^{-1}(y))} d\gamma(y).$$

By the change of variables $z = T^{-1}(y)$, we rewrite this as:

$$= \int_X f_1(z) \log \frac{f_1(z)}{f_2(z)} d\lambda(z) = D_{\mathrm{KL}}(\mu_1||\mu_2).$$

Thus, the KL divergence is invariant under the transformation T, completing the proof. □

A.5 Proof and Remark of Theorems

A.5.1 Proof of Theorem. 1

Theorem 1. *Let $\mathbf{X} \sim p(\mathbf{x})$ be an arbitrary distributed random vector on $\mathbb{R}^D$, and let $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$ be a parametric model with $\hat{\mathbf{X}} \sim q(\hat{\mathbf{x}}; \boldsymbol{\theta})$. Define the perturbed observation*

$$\tilde{\mathbf{X}} := \alpha_t \mathbf{X} + \sigma_t \boldsymbol{\Psi},$$

where $\boldsymbol{\Psi}$ is a random vector independent of $\mathbf{X}$, satisfying $\mathbb{E}[\boldsymbol{\Psi}] = 0$ and $\mathrm{Cov}(\boldsymbol{\Psi}) = \mathbf{I}$. Let $p_{\sigma_t^2}$ and $q_{\sigma_t^2}$ denote the densities of $\tilde{\mathbf{X}}$ under p and q with noise variance σ_t^2, respectively. Suppose that the KL divergence $D_{\mathrm{KL}}(p_{\sigma_t^2} \| q_{\sigma_t^2})$ is finite for sufficiently small σ_t^2. Then the following limit holds:

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p_{\sigma_t^2} \| q_{\sigma_t^2}) \Big|_{\sigma_t^2 \to 0^+} = -\frac{1}{2} \int_{\mathbb{R}^D} p(\mathbf{x}) \left\| \nabla \log p(\mathbf{x}) - \nabla \log q(\hat{\mathbf{x}}; \boldsymbol{\theta}) \right\|^2 d\mathbf{x}, \tag{50}$$

i.e.,

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) \Big|_{\sigma_t^2 \to 0^+} = -\frac{1}{2} I\left(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})\right),$$

where $I(\cdot\|\cdot)$ denotes the Fisher divergence (equivalently, score matching objective) between p and q.

Proof. According to the definition of relative entropy (See Def. 3), we have

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) = \frac{d}{d\sigma_t^2} \int_{\mathbb{R}^D} p(\tilde{\mathbf{x}}) \log(\frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}) \, d\tilde{\mathbf{x}} \tag{51}$$

$$= \int \frac{\partial p(\tilde{\mathbf{x}})}{\partial \sigma_t^2} \log(\frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}) + p(\tilde{\mathbf{x}}) \frac{\partial}{\partial \sigma_t^2} \log(\frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}) \, d\tilde{\mathbf{x}} \tag{52}$$

$$= \int \frac{\partial p(\tilde{\mathbf{x}})}{\partial \sigma_t^2} \log(\frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}) + p(\tilde{\mathbf{x}}) \left(\frac{\partial p(\tilde{\mathbf{x}})}{\partial \sigma_t^2} / p(\tilde{\mathbf{x}}) - \frac{\partial q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}{\partial \sigma_t^2} / q(\tilde{\mathbf{x}}; \boldsymbol{\theta}) \right) \, d\tilde{\mathbf{x}}. \tag{53}$$

Invoking Lemma 1 on (53) yields

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) \Big|_{\sigma_t^2 = 0^+} = \frac{1}{2} \int \left(p''(\tilde{\mathbf{x}}) \log(\frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})}) + p''(\tilde{\mathbf{x}}) - \frac{q''(\tilde{\mathbf{x}}; \boldsymbol{\theta}) p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})} \right) \, d\tilde{\mathbf{x}}. \tag{54}$$

For convenience, define:

$$\upsilon(\tilde{\mathbf{x}}) = \log \frac{p(\tilde{\mathbf{x}})}{q(\tilde{\mathbf{x}}; \boldsymbol{\theta})} = \log p(\tilde{\mathbf{x}}) - \log q(\tilde{\mathbf{x}}; \boldsymbol{\theta})$$

Recall the Green's identity:

$$\int_{\mathbb{R}^D} \nu \nabla^2 \mu \, d\tilde{\mathbf{x}} = \int_{\partial \mathbb{R}^D} \nu \frac{\partial \mu}{\partial n} dS - \int_{\mathbb{R}^D} (\nabla \nu \cdot \nabla \mu) \, d\tilde{\mathbf{x}}. \tag{55}$$

Setting $\nu = p$ and $\mu = \upsilon = \log \frac{p}{q}$, we obtain:

$$\int_{\mathbb{R}^D} p'' \upsilon \, d\tilde{\mathbf{x}} = \int_{\partial \mathbb{R}^D} p \frac{\partial \upsilon}{\partial n} dS - \int_{\mathbb{R}^D} (\nabla \upsilon \cdot \nabla p) \, d\tilde{\mathbf{x}}. \tag{56}$$

Since $p(\tilde{\mathbf{x}})$ has finite differential Shannon entropy, the boundary integral vanishes, leaving

$$\int_{\mathbb{R}^D} p'' \upsilon \, d\tilde{\mathbf{x}} = - \int_{\mathbb{R}^D} (\nabla \upsilon \cdot \nabla p) \, d\tilde{\mathbf{x}}. \tag{57}$$

For the term $q'' p / q$, we apply the same identity:

$$\int_{\mathbb{R}^D} \frac{q'' p}{q} d\tilde{\mathbf{x}} = - \int_{\mathbb{R}^D} (p \nabla \log q \cdot \nabla \log q) \, d\tilde{\mathbf{x}}. \tag{58}$$

Since $\int_{\mathbb{R}^D} p''(\tilde{\mathbf{x}}) d\tilde{\mathbf{x}} = 0$ under appropriate boundary conditions, it does not contribute. Similar technique was used in [5, 58, 59]. Using $\upsilon = \log p - \log q$, we compute:

$$\nabla \upsilon = \nabla \log p - \nabla \log q. \tag{59}$$

Thus,

$$\nabla p \cdot \nabla \upsilon = \nabla p \cdot (\nabla \log p - \nabla \log q). \tag{60}$$

Since $\nabla p = p \nabla \log p$, we substitute:

$$\nabla p \cdot \nabla \upsilon = p \nabla \log p \cdot (\nabla \log p - \nabla \log q). \tag{61}$$

Expanding,

$$\nabla \log p \cdot (\nabla \log p - \nabla \log q) = \|\nabla \log p\|^2 - \nabla \log p \cdot \nabla \log q. \tag{62}$$

Thus,

$$\int_{\mathbb{R}^D} p'' \upsilon \, d\tilde{\mathbf{x}} = - \int_{\mathbb{R}^D} p \left(\|\nabla \log p\|^2 - \nabla \log p \cdot \nabla \log q \right) d\tilde{\mathbf{x}}. \tag{63}$$

Using the result for $q'' p / q$:

$$\int_{\mathbb{R}^D} \frac{q'' p}{q} d\tilde{\mathbf{x}} = - \int_{\mathbb{R}^D} (p \nabla \log q \cdot \nabla \log q \,) \, d\tilde{\mathbf{x}}. \tag{64}$$

Summing both contributions,

$$\frac{1}{2} \int_{\mathbb{R}^D} \left(-p \left(\|\nabla \log p\|^2 - \nabla \log p \cdot \nabla \log q \right) - p \|\nabla \log q\|^2 + p \nabla \log p \cdot \nabla \log q \right) \, d\tilde{\mathbf{x}}. \tag{65}$$

Rearranging, we conclude:

$$\frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\tilde{\mathbf{x}}) \| q(\tilde{\mathbf{x}}; \boldsymbol{\theta})) \Big|_{\sigma_t^2 = 0^+} = -\frac{1}{2} \int_{\mathbb{R}^D} p(\mathbf{x}) \|\nabla \log p(\mathbf{x}) - \nabla \log q(\mathbf{x}; \boldsymbol{\theta})\|_2^2 \, d\mathbf{x}, \tag{66}$$

which completes the proof. □

Similar to the classical result in [51, 78], the relation in Theorem 1 holds because both sides quantify the error induced by a mismatch between the true distribution p and the prior q provided to the estimator. Naturally, when $p = q$, both sides vanish; otherwise, the derivative is strictly negative, indicating that perturbations reduce the relative entropy. This observation also yields the relative entropy version of the data processing inequality:

$$D_{\mathrm{KL}}(\rho\|\nu) \geq D_{\mathrm{KL}}(\bar{\rho}\|\bar{\nu}), \tag{67}$$

where

$$\bar{\rho} = \int W(\mathbf{y}|\mathbf{x})\, d\rho(\mathbf{x}), \qquad \bar{\nu} = \int W(\mathbf{y}|\mathbf{x})\, d\nu(\mathbf{x}).$$

Here, W denotes a noisy channel. This inequality asserts that the KL divergence between two distributions decreases under the action of a common channel, which is consistent with the data processing argument used in [69], where, the channel W corresponds to time-reversed Brownian motion, which can be viewed as a continuous analogue of the Gaussian channel [82]. Under the assumption that $\bar{\rho}(\mathbf{x}) = \bar{\nu}(\mathbf{x}) = \pi(\mathbf{x})$, the result further implies that the diffusion process smooths the discrepancy between inputs. Moreover, the neural network acts as a channel simulator, enabling efficient sampling by integrating its output into the neural ODE solvers [37].

A.5.2 Proof of Proposition. 1

Proposition 1. *Consider the signal model* (1)*, suppose $p(\tilde{\mathbf{x}})$ and $q(\tilde{\mathbf{x}};\boldsymbol{\theta})$ have continuous second-order derivatives and finite second moments. Denote by $p(\mathbf{y}_1)$ and $\pi(\mathbf{x})$ the output signals of channel at time $t = 1$ when inputs are $p(\mathbf{x})$ and $q(\hat{\mathbf{x}};\boldsymbol{\theta})$ respectively. Assume $\pi(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$, which is independent of $\boldsymbol{\theta}$. Let $p(\mathbf{y}_t|\mathbf{x})$ denote the channel* (1) *for any $t \in [0, 1]$, then for arbitrary datapoint $\mathbf{x}$ and small σ_0^2 with $0 < \sigma_0^2 \ll 1$:*

$$\mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}};\boldsymbol{\theta})) = \mathcal{H}(p(\mathbf{y}_1), \pi(\mathbf{x})) + \mathcal{J}_{\mathrm{DSM}}(\boldsymbol{\theta}; \sigma_t^2(\cdot)) - \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}\|\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x})\|^2 d\sigma_t^2 + o(\sigma_0^2). \tag{68}$$

Here, the denoising score matching objective is defined as

$$\mathcal{J}_{DSM}(\boldsymbol{\theta}; \sigma^2(\cdot)) := \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}\|\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x}) - \hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta})\|_2^2\, d\sigma_t^2, \tag{69}$$

and $\hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta}) = \nabla_{\mathbf{y}_t} \log q(\mathbf{y}_t;\boldsymbol{\theta}) := -\hat{\mathbf{n}}(\mathbf{y}_t, \eta_t;\boldsymbol{\theta})/\sigma_t$ is a score network estimator.

Proof. Recall the thermodynamic integration commonly used in statistical physics:

$$\int_0^{\Xi} \frac{d}{d\xi} f(\xi) d\xi = f(\Xi) - f(0). \tag{70}$$

Let $f(\sigma_t^2) := D_{\mathrm{KL}}\big(p(\mathbf{y})\|q(\mathbf{y};\boldsymbol{\theta})\big)$, where $\mathbf{Y}_t = \alpha_t \mathbf{X} + \sigma_t \mathbf{N}$, i.e.,

$$\int_0^{\Xi} \frac{d}{d\xi} f(\xi) d\xi = f(\Xi) - f(0), \tag{71}$$

becomes equivalent to

$$\int_0^{\sigma_1^2} \frac{d}{d\sigma_t^2} f(\sigma_t^2) d\sigma_t^2 = f(\sigma_1^2) - f(0), \tag{72}$$

which yields the following expression,

$$\int_0^{\sigma_1^2} \frac{d}{d\sigma_t^2} D_{\mathrm{KL}}(p(\mathbf{y})\|q(\mathbf{y};\boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\mathbf{y}_1)\|q(\mathbf{y}_1;\boldsymbol{\theta})) - D_{\mathrm{KL}}(p(\mathbf{x})\|q(\hat{\mathbf{x}};\boldsymbol{\theta})). \tag{73}$$

This occurs because the KL divergence remains invariant under measurable transformations shown in Lemma 4. In particular, probability density transformation serves as an illustrative example. Let $\mathbf{X} \sim \rho$ be a random vector on the measure space with a probability density function $p(\mathbf{x})$. We define

the transformation $\mathbf{X}' = T(\mathbf{X})$. Consequently, the probability density function of $\mathbf{X}'$, denoted as $p(\mathbf{x}')$, can be determined using:

$$p(\mathbf{x}') = p(T^{-1}(\mathbf{x}')) \left| \frac{d}{d\mathbf{x}'} T^{-1}(\mathbf{x}') \right|. \tag{74}$$

Furthermore, the KL divergence satisfies the following property:

$$D_{\mathrm{KL}}(\rho(\alpha\mathbf{x}) \| \nu(\alpha\mathbf{x})) = D_{\mathrm{KL}}(\rho(\mathbf{x}) \| \nu(\mathbf{x})) \quad \forall \alpha > 0, \tag{75}$$

which implies that the KL divergence depends solely on the relative shape of the two distributions rather than their absolute scale. Invoke the Lemma 3 and results in [51] on (73) yields

$$D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\mathbf{y}_1) \| q(\mathbf{y}_1; \boldsymbol{\theta})) + \frac{1}{2} \int_0^{\sigma_1^2} I(p(\mathbf{y}_t) \| q(\mathbf{y}_t; \boldsymbol{\theta})) d\sigma_t^2. \tag{76}$$

With the fact that $q(\mathbf{y}_1; \boldsymbol{\theta}) := \pi(\mathbf{x})$, which is independent of $\boldsymbol{\theta}$, we have

$$D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\mathbf{y}_1) \| \pi(\mathbf{x})) + \frac{1}{2} \int_0^{\sigma_1^2} I(p(\mathbf{y}_t) \| q(\mathbf{y}_t; \boldsymbol{\theta}))\, d\sigma_t^2. \tag{77}$$

The mismatched entropy between data and model becomes equivalent to

$$\mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\mathbf{x}) \| q(\hat{\mathbf{x}}; \boldsymbol{\theta})) + \mathcal{H}(p(\mathbf{x})). \tag{78}$$

Invoking (77) on (78) yields

$$\mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = D_{\mathrm{KL}}(p(\mathbf{y}_1) \| \pi(\mathbf{x})) + \frac{1}{2} \int_0^{\sigma_1^2} I(p(\mathbf{y}_t) \| q(\mathbf{y}_t; \boldsymbol{\theta}))\, d\sigma_t^2 + \mathcal{H}(p(\mathbf{x})). \tag{79}$$

Recall de Bruijin's identity [12]:

$$\frac{d}{d\sigma_t^2} \mathcal{H}(p(\mathbf{y}_t)) = \frac{1}{2} \mathcal{J}(p(\mathbf{y}_t)). \tag{80}$$

Using thermodynamic integration again, we have

$$\int_0^{\sigma_1^2} \frac{d}{d\sigma_t^2} \mathcal{H}(p(\mathbf{y}_t)) d\sigma_t^2 = \mathcal{H}(p(\mathbf{y}_1)) - \mathcal{H}(p(\alpha_0 \mathbf{x})), \tag{81}$$

which is equivalent to

$$\mathcal{H}(p(\alpha_0 \mathbf{x})) = \mathcal{H}(p(\mathbf{y}_1)) - \frac{1}{2} \int_0^{\sigma_1^2} \mathcal{J}(p(\mathbf{y}_t)) d\sigma_t^2. \tag{82}$$

Recall the property of differential entropy (see Theorem 8.6.4 (8.71) in [14])

$$\mathcal{H}(a\mathbf{X}) = \mathcal{H}(\mathbf{X}) + \log |\det(a)|, \tag{83}$$

and the connection with denoising score matching (see (11) in [79])

$$I(p(\mathbf{y}_t) \| q(\mathbf{y}_t; \boldsymbol{\theta})) = \mathbb{E}\left[\| \nabla \log p(\mathbf{y}_t | \mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t; \boldsymbol{\theta}) \|^2 \right] + \mathcal{J}(p(\mathbf{y}_t)) - \mathbb{E}\left[\| \nabla \log p(\mathbf{y}_t | \mathbf{x}) \|^2 \right]. \tag{84}$$

Finally, we have

$$\begin{aligned} \mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = \Bigg(& D_{\mathrm{KL}}(p(\mathbf{y}_1) \| \pi(\mathbf{x})) + \mathcal{H}(p(\mathbf{y}_1)) - \frac{1}{2} \int_0^{\sigma_1^2} \mathcal{J}(p(\mathbf{y}_t))\, d\sigma_t^2 - D \log |\alpha_0| \\ & + \frac{1}{2} \Bigg(\int_0^{\sigma_1^2} \mathbb{E}\left[\| \nabla \log p(\mathbf{y}_t | \mathbf{x}) - \hat{\boldsymbol{s}}(\mathbf{y}_t; \boldsymbol{\theta}) \|^2 \right] + \mathcal{J}(p(\mathbf{y}_t)) - \mathbb{E}\left[\| \nabla \log p(\mathbf{y}_t | \mathbf{x}) \|^2 \right] d\sigma_t^2 \Bigg) \Bigg) \end{aligned} \tag{85}$$

$$= \mathcal{H}(p(\mathbf{y}_1), \pi(\mathbf{x})) + \mathcal{J}_{\mathrm{DSM}}(\boldsymbol{\theta}; \sigma_t^2(\cdot)) - \frac{1}{2} \int_0^{\sigma_1^2} \mathbb{E} \| \nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t | \mathbf{x}) \|^2 d\sigma_t^2 - D \log |\alpha_0|. \tag{86}$$

Since the identity $\sigma_t^2 + \alpha_t^2 = 1$ holds for all t, setting the lower limit of integration such that $\sigma_0^2 = 0$ implies $\alpha_0^2 = 1$. Consequently, $|\alpha_0| = 1$, and thus: $\log |\alpha_0| = \log 1 = 0$. And invoking Theorem 1, we finally get

$$\mathcal{H}(p(\mathbf{x}), q(\hat{\mathbf{x}}; \boldsymbol{\theta})) = \mathcal{H}(p(\mathbf{y}_1), \pi(\mathbf{x})) + \mathcal{J}_{\mathrm{DSM}}(\boldsymbol{\theta}; \sigma_t^2(\cdot)) - \frac{1}{2} \int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E} \| \nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t | \mathbf{x}) \|^2 d\sigma_t^2 + o(\sigma_0^2). \tag{87}$$

□

A.5.3 Proof of Theorem 2

Theorem 2. *Let $p(\mathbf{y}_t|\mathbf{x}) = \mathcal{N}(\alpha_t\mathbf{x}, \sigma_t^2\mathbf{I})$ denote the Gaussian channel at any time $t \in [0,1]$. With the same notations and conditions in Proposition 1, we have*

$$-\log q(\hat{\mathbf{x}};\boldsymbol{\theta}) \leq \mathcal{H}(p(\mathbf{y}_1|\mathbf{x}),\pi(\mathbf{x})) + \mathcal{L}_{\text{DSM}}(\sigma_t^2;\boldsymbol{\theta}), \tag{88}$$

in which $\mathcal{L}_{DSM}(\sigma_t^2;\boldsymbol{\theta})$ is defined as

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2;\boldsymbol{\theta}) := \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}_{p(\mathbf{y}_t|\mathbf{x})}\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t;\boldsymbol{\theta})\|^2 d\sigma_t^2,$$

and $\mathcal{H}(p(\mathbf{y}_1|\mathbf{x}),\pi(\mathbf{x}))$ is given by

$$\mathcal{H}(p(\mathbf{y}_1|\mathbf{x}),\pi(\mathbf{x})) = D_{\text{KL}}(p(\mathbf{y}_1|\mathbf{x})\|q(\mathbf{y}_1)) + \mathcal{H}(p(\mathbf{y}_1|\mathbf{x})).$$

Proof. From Proposition 1 we have

$$\begin{aligned}
-\mathbb{E}_{p(\mathbf{x})}[\log q(\hat{\mathbf{x}};\boldsymbol{\theta})] &= -\mathbb{E}_{p(\mathbf{y}_1)}[\log\pi(\mathbf{x})] + \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\mathbb{E}_{p(\mathbf{y}_t|\mathbf{x})p(\mathbf{x})}\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t;\boldsymbol{\theta})\|^2 d\sigma_t^2 \\
&\quad - \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\mathbb{E}_{p(\mathbf{y}_t|\mathbf{x})p(\mathbf{x})}\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x})\|^2 d\sigma_t^2 \qquad (89) \\
&= -\int_{\mathbb{R}^D}\int_{\mathbb{R}^D} p(\mathbf{y}_1|\mathbf{x})p(\mathbf{x})d\mathbf{x}\log\pi(\mathbf{x})d\mathbf{y}_1 \\
&\quad - \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\iint_{\mathbb{R}^D} p(\mathbf{y}_t|\mathbf{x})p(\mathbf{x})\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x})\|^2 d\mathbf{x}d\mathbf{y}_t d\sigma_t^2 \\
&\quad + \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\iint_{\mathbb{R}^D} p(\mathbf{y}_t|\mathbf{x})p(\mathbf{x})\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t;\boldsymbol{\theta})\|^2 d\mathbf{x}d\mathbf{y}_t d\sigma_t^2 \qquad (90)
\end{aligned}$$

Given a fixed channel $p(\mathbf{y}_t|\mathbf{x})$, we can easily see that $\int_{\mathbb{R}^D} p(\mathbf{x})d\mathbf{x}$ in both sides of (90) can be canceled to get

$$\begin{aligned}
-\log q(\hat{\mathbf{x}};\boldsymbol{\theta}) &= -\int_{\mathbb{R}^D} p(\mathbf{y}_1|\mathbf{x})\log\pi(\mathbf{x})d\mathbf{y}_1 - \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\int_{\mathbb{R}^D} p(\mathbf{y}_t|\mathbf{x})\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x})\|^2 d\mathbf{y}_t d\sigma_t^2 \\
&\quad + \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\int_{\mathbb{R}^D} p(\mathbf{y}_t|\mathbf{x})\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t,\boldsymbol{\theta})\|^2 d\mathbf{y}_t d\sigma_t^2. \qquad (91)
\end{aligned}$$

The second term in (91) corresponds to one-half of the integrated Fisher information of the Gaussian distribution, which is strictly non-negative [19], we obtain:

$$-\log q(\hat{\mathbf{x}};\boldsymbol{\theta}) \leq \mathcal{H}(p(\mathbf{y}_1|\mathbf{x}),\pi(\mathbf{x})) + \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2}\int_{\mathbb{R}^D} p(\mathbf{y}_t|\mathbf{x})\|\nabla_{\mathbf{y}_t}\log p(\mathbf{y}_t|\mathbf{x}) - \boldsymbol{s}(\mathbf{y}_t,\boldsymbol{\theta})\|^2 d\mathbf{y}_t d\sigma_t^2, \tag{92}$$

which finishes the proof. □

A.5.4 Proof of Proposition 2

Proposition 2. *Let $\tilde{\mathbf{X}} = \alpha_t\mathbf{X} + \sigma_t\boldsymbol{\Psi}$, where $\mathbf{X}, \boldsymbol{\Psi} \in \mathbb{R}^D$ are independent random vectors, with $\boldsymbol{\Psi}$ satisfying $\mathbb{E}[\boldsymbol{\Psi}] = 0$ and $\text{Cov}(\boldsymbol{\Psi}) = \mathbf{I}$. Assume the probability density function $p(\mathbf{x})$ of $\mathbf{X}$ is twice continuously differentiable and decays sufficiently fast at infinity, and that the Fisher information $\mathcal{J}(\mathbf{X})$ exists and is finite. Then,*

$$\frac{d}{d\sigma_t^2}\mathcal{H}\big(p(\tilde{\mathbf{x}})\big)\Big|_{\sigma_t^2\to 0^+} = \frac{1}{2}\mathcal{J}\big(p(\mathbf{x})\big), \tag{93}$$

where $\mathcal{H}(\cdot)$ denotes the differential Shannon entropy and $\mathcal{J}(\cdot)$ denotes the Fisher information.

Proof. By the smoothing properties established in Lemma 1, the distribution $p(\tilde{\mathbf{x}})$ is differentiable with respect to σ_t^2. Since the integrand in equation (13) is both continuous and differentiable in σ_t^2, we may interchange the order of differentiation and integration to obtain:

$$\frac{d}{d\sigma_t^2}\mathcal{H}\big(p(\tilde{\mathbf{x}})\big) = -\int_{\mathbb{R}^D} \frac{d}{d\sigma_t^2} p(\tilde{\mathbf{x}}) \cdot (1 + \log p(\tilde{\mathbf{x}}))\, d\tilde{\mathbf{x}} \tag{94}$$

$$= -\frac{1}{2}\int_{\mathbb{R}^D} (\Delta_{\tilde{\mathbf{x}}} p(\tilde{\mathbf{x}})) \log p(\tilde{\mathbf{x}})\, d\tilde{\mathbf{x}}. \tag{95}$$

Following the same argument as in Theorem 1, this establishes the desired expression, which recovers the special case of isotropic additive noise discussed in [60]. □

B Improve the Likelihood Estimation Bounds

Our theoretical analysis suggests that different variance functions can lead to varying performance in likelihood estimation. To validate this empirically, we evaluate the effect of different noise schedules, variance functions and datasets on likelihood estimation performance.

However, in practice, exact numerical evaluation of the integral is generally intractable. A common approach is to use Monte Carlo method to estimate the variance integral in $\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta})$ during both training and evaluation. Moreover, reducing the variance of the Monte Carlo estimator for the continuous loss objective generally improves the efficiency of optimization. We next summarise our empirical observations on likelihood estimation across different settings.

B.1 Variance-Aware Likelihood Bounds

As mentioned in Section 3.3, denoising score matching [79] is typically used as the training objective for diffusion models, serving as a surrogate for approximating the likelihood function $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$. Specifically, recall that out channel (1) is defined as $p(\mathbf{y}_t|\mathbf{x}) = \mathcal{N}(\alpha_t \mathbf{x}, \sigma_t^2 \mathbf{I})$, such that $\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t|\mathbf{x}) = -\mathbf{n}/\sigma_t$. In the sense that with the score model $\hat{\boldsymbol{s}}(\mathbf{y}_t; \boldsymbol{\theta}) := -\hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})/\sigma_t$, the objective takes the following integral form, where the loss is evaluated under varying noise levels:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}_{p(\mathbf{y}_t|\mathbf{x})}\left[\sigma_t^{-2} \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2\right] d\sigma_t^2. \tag{96}$$

We begin by approximating the expectation in $\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta})$ via Monte Carlo estimation. This involves drawing samples $\mathbf{y}_t \sim p(\mathbf{y}_t|\mathbf{x})$ from a tractable Gaussian kernel, which enables efficient estimation of the denoising objective. To estimate the integral over the variance schedule, we apply a second Monte Carlo approximation by uniformly sampling σ_t^2 from the interval $[\sigma_0^2, \sigma_1^2]$. However, we empirically observe that such an estimator can significantly degrade the optimization process, potentially due to increased variance or poor convergence behaviour. To address this, we reparameterize the objective in terms of a smoother coordinate, namely, the negative log signal-to-noise ratio (log-SNR), which facilitates both numerical stability and analytical tractability.

Even under constrained noise schedules, such as the Variance Preserving (VP) formulation [28], which enforces $\alpha_t^2 + \sigma_t^2 = 1$, there remains freedom in how α_t, σ_t evolve over time t. To abstract away from specific parameterizations such as VP or VE schedules, we adopt a log-SNR parameterization as the default: $\eta_t := -\log \text{SNR}(t) = -\log \frac{\alpha_t^2}{\sigma_t^2}$. This reparameterization simplifies the analysis and unifies various diffusion formulations under a single coordinate system. Under this reparameterization, the change of variable from time t to negative log-SNR η follows:

$$\frac{d\eta_t}{dt} = \frac{1}{\sigma_t^2}\frac{d\sigma_t^2}{dt} - \frac{1}{\alpha_t^2}\frac{d\alpha_t^2}{dt}. \tag{97}$$

For the VP schedule, this simplifies to:

$$\frac{d\eta_t}{dt} = \frac{1}{\alpha_t^2 \sigma_t^2}\frac{d\sigma_t^2}{dt}. \tag{98}$$

By setting the noisy process to follow an EDM-like variance-exploding (VE) schedule [37], we let $\alpha_t^2 \equiv 1$ and parameterize the variance as $\sigma_t^2 := \exp(\eta_t)$. This configuration is equivalent to that

of NCSNv2 [70] when σ_t follows a geometric sequence interpolating between 0.01 and 50, i.e., $\eta(t) = 2\log(0.01) + 2\log(5000)\, t$. Under this specification, the objective in (96) reduces to the continuous-time loss proposed in [40]:

$$\mathcal{L}_\infty(\mathbf{x}) := \frac{1}{2}\mathbb{E}_{\mathbf{n}\sim\mathcal{N}(0,\mathbf{I}),t\sim\mathcal{U}(0,1)}[\eta'(t)\|\mathbf{n} - \hat{\mathbf{n}}_{\boldsymbol{\theta}}(\mathbf{y}_t; t)\|_2^2], \tag{99}$$

where $\eta'(t) = d\eta_t/dt$ acts as a natural weighting function. In the above case, this means that $\eta'(t) = 2\log[5000]$ and thus that weighted function is a constant. We leave the VE schedule optimization for future works.

Similarly, in the information-theoretic VE formulation [43], i.e., $\mathbf{z}_\gamma = \sqrt{\gamma}\,\mathbf{x} + \mathbf{n}$, our bound recovers the loss integrates over the signal-to-noise ratio γ:

$$\mathcal{L} := \frac{1}{2}\int_{\text{SNR}_{\text{min}}}^{\text{SNR}_{\text{max}}} \mathbb{E}_{p(\mathbf{z}_\gamma,\mathbf{x})}\left[\|\mathbf{x} - \hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_\gamma, \gamma)\|_2^2\right] d\gamma, \tag{100}$$

where instead of integrating w.r.t. time t, now integrate w.r.t. the signal-to-noise ratio γ, and where $\text{SNR}_{\text{max}} = \gamma(1)$, $\text{SNR}_{\text{min}} = \gamma(0)$.

In a more general form, the objective in (96) recovers the weighted continuous-time formulation of Eq.(66) [40], where the weighting function is defined as $w(t) := \alpha_t^2$; equivalently, this corresponds to scaling the signal coefficient by α_t across time:

$$\mathcal{L}_\infty^w(\mathbf{x}) := \frac{1}{2}\mathbb{E}_{\mathbf{n}\sim\mathcal{N}(0,\mathbf{I}),t\sim\mathcal{U}(0,1)}[\eta'(t)\alpha_t^2\|\mathbf{n} - \hat{\mathbf{n}}_{\boldsymbol{\theta}}(\mathbf{y}_t; t)\|_2^2], \tag{101}$$

Moreover, (96) can be further unified under the stochastic differential equation (SDE) framework [72]:

$$d\mathbf{x} = \boldsymbol{f}(\mathbf{x}, t)\, dt + g(t)\, d\mathbf{w}, \tag{102}$$

where $\boldsymbol{f}(\mathbf{x}, t) : \mathbb{R}^D \to \mathbb{R}^D$, $g(t) \in \mathbb{R}$ are manually designed noise schedules and $\mathbf{w} \in \mathbb{R}^D$ is a standard Wiener process. When the variance function satisfies $d\sigma_t^2/dt = g^2(t)$, the corresponding loss aligns with the likelihood-weighted score-matching objective [69]:

$$\mathcal{J}_{\text{SM}}(\boldsymbol{\theta}) := \int_0^T \frac{g^2(t)}{2\sigma_t^2}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}),\, \mathbf{n}\sim\mathcal{N}(0,\mathbf{I})}\left[\|\sigma_t\hat{\boldsymbol{s}}(\mathbf{y}_t;\boldsymbol{\theta}) + \mathbf{n}\|_2^2\right] dt. \tag{103}$$

The same principle extends to deterministic flow-based formulations, including continuous-time ODEs and higher-order probability flows [50, 69, 90] with specific design of the $g^2(t)$. In this case, the training objective becomes:

$$\mathcal{J}_{\text{SM}}^\eta(\boldsymbol{\theta}) := \frac{1}{2}\int_{\eta_0}^{\eta_1} \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x}),\, \mathbf{n}\sim\mathcal{N}(0,\mathbf{I})}\left[\|\sigma_t\hat{\boldsymbol{s}}(\mathbf{y}_\eta;\boldsymbol{\theta}) + \mathbf{n}\|_2^2\right] d\eta. \tag{104}$$

This reformulation highlights that the loss landscape is intimately linked to the choice of noise schedule, particularly the functional form of σ_t^2 as a function of log-SNR η. Since both the weighting in the integrand and the resulting likelihood bounds depend explicitly on σ_t^2, different scheduling strategies can lead to substantial differences in training dynamics, gradient variance, and overall model performance. In the following, we provide both theoretical and empirical analyses to quantify how the design of σ_t^2 influences estimation accuracy, stability, and convergence behaviour.

B.2 The Noise Schedule Matters

Many practical objectives, including the ones above, are special cases of the likelihood–weighting loss [39, 40, 69]. Let $\eta = g(t)$ be a differentiable, monotone noise parameterization of time $t \in [0, 1]$ with fixed endpoints $\eta_{\min} = g(0)$ and $\eta_{\max} = g(1)$. For a per-example loss, define

$$\mathcal{L}_w(\mathbf{x}) := \frac{1}{2}\int_{\eta_{\min}}^{\eta_{\max}} w(\eta)\, \mathbb{E}_{\mathbf{n}\sim\mathcal{N}(0,\mathbf{I})}\left[\left\|\mathbf{n} - \hat{\mathbf{n}}\big(\mathbf{y}_\eta, \eta; \boldsymbol{\theta}\big)\right\|_2^2\right] d\eta, \tag{105}$$

where $\mathbf{y}_\eta$ denotes the noisy input at noise level η (e.g., via a log-SNR parameterization). As shown in [40], the *integral* in (105) is invariant to smooth reparameterizations of η (i.e., to the choice of schedule) so long as the endpoints $\eta_{\min}, \eta_{\max}$ are kept fixed. This invariance extends to the weighted diffusion loss family [39] because the integrand is measured per unit η.

However, this invariance does *not* carry over to the Monte Carlo (MC) estimator used in evaluation. If we sample $t \sim \mathcal{U}(0,1)$ and $\mathbf{n} \sim \mathcal{N}(0, \mathbf{I})$, then by change of variables,

$$\mathcal{L}_w(\mathbf{x}) = \frac{1}{2}\, \mathbb{E}_{t\sim\mathcal{U}(0,1),\, \mathbf{n}\sim\mathcal{N}(0,\mathbf{I})}\Big[\, w\big(\eta(t)\big)\, \Big|\tfrac{d\eta}{dt}(t)\Big|\, \big\|\mathbf{n} - \hat{\mathbf{n}}\big(\mathbf{y}_{\eta(t)}, \eta(t); \boldsymbol{\theta}\big)\big\|_2^2 \Big]. \tag{106}$$

More generally, if we sample directly $\eta \sim \rho(\eta)$ over $[\eta_{\min}, \eta_{\max}]$ (with $\rho > 0$ a.e.), then

$$\mathcal{L}_w(\mathbf{x}) = \frac{1}{2}\, \mathbb{E}_{\eta\sim\rho,\, \mathbf{n}}\left[\, \frac{w(\eta)}{\rho(\eta)}\, \big\|\mathbf{n} - \hat{\mathbf{n}}\big(\mathbf{y}_\eta, \eta; \boldsymbol{\theta}\big)\big\|_2^2 \right], \tag{107}$$

revealing that the *noise schedule* induces an *importance-sampling distribution* $\rho(\eta)$ over η and a corresponding weighting function $w(\eta)$ related to $\sigma_t^2(\eta)$, simultaneously. Consequently, while the population loss is invariant to reparameterization, the *variance* of the MC estimator (and of its gradients) depends on ρ. Choosing ρ to better match the integrand can substantially reduce estimator variance and thus accelerate optimization. In particular, the variance-minimizing density satisfies the heuristic proportionality $\rho^\star(\eta) \propto \mathbb{E}_{\mathbf{x}\sim\mathcal{X}, \mathbf{n}\sim\mathcal{N}(0,\mathbf{I})}\Big[w(\eta)\, \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_\eta, \eta; \boldsymbol{\theta})\|_2^2\Big]$.

B.3 Variance Reduction with Importance Sampling

Monte Carlo approximation offers a computationally efficient alternative to exact integration, but typically introduces variance into the training objective. To mitigate this, we adopt both designed and learned importance sampling (IS) strategies, as outlined in Section 3.4. As mentioned in Section. B.2, the diffusion model is conducted for all η in $[\eta_0, \eta_1]$ through an integral. In practice, the evaluation of the integral is time-consuming, and Monte-Carlo methods are used to unbiasedly estimate the objective by uniformly sampling η. Thus, a continuous importance distribution $\rho(\eta)$ can be proposed for variance reduction. Denote

$$\mathcal{L}(\mathbf{x}, \mathbf{n}, \eta; \boldsymbol{\theta}) := \frac{\alpha_t^2 \|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_\eta, \eta; \boldsymbol{\theta})\|_2^2}{2},$$

then

$$\mathcal{L}_w(\mathbf{x}; \boldsymbol{\theta}) = \mathbb{E}_{\eta\sim\rho(\eta)}\mathbb{E}_{\mathbf{x},\mathbf{n}}\left[\frac{\mathcal{L}(\mathbf{x}, \mathbf{n}, \eta; \boldsymbol{\theta})}{\rho(\eta)}\right]. \tag{108}$$

We propose to use two types of importance sampling (IS), and empirically compare them for faster convergence.

B.3.1 Designed IS

In particular, we propose a continuous proposal distributions $\rho(\eta)$ over η, which is proportional α_η^2. Since we have explicit expressions for the density, we utilize inverse transform sampling to design a sampling procedure. Concretely, we take uniform samples of a number $t \in [0, 1]$, and solve the following equation about η_t:

$$\frac{1}{Z}\int_{\eta_0}^{\eta_1} \alpha_\eta^2\, d\eta = t, \quad Z = \int_{\eta_0}^{\eta_1} \alpha_\eta^2\, d\eta, \tag{109}$$

where we have defined maximum time $t = 1$, and Z is a normalizing constant.

Uniform Weighting in η-Space. As a canonical example, we consider uniform weighting in the log-SNR domain. In this case, the proposal distribution becomes $\rho(\eta) \propto \alpha_\eta^2$, yielding:

$$Z = \int_{\eta_0}^{\eta_1} \text{sigmoid}(-\eta)\, d\eta = \log\left(\frac{1 + e^{-\eta_0}}{1 + e^{-\eta_1}}\right).$$

Also, for squashed hyperbolic tangent, we have:

$$Z = \int_{\eta_0}^{\eta_1} \text{sigmoid}(-2\eta)\, d\eta = \frac{1}{2}\log\left(\frac{1 + e^{-2\eta_0}}{1 + e^{-2\eta_1}}\right).$$

To enable inverse transform sampling, we define the antiderivative $g(\eta) := \log(1 + e^{-\eta})$, and set $l_0 = g(\eta_0), l_1 = g(\eta_1)$. The corresponding cumulative distribution function (CDF) is then:

$$F(\eta) = \frac{l_0 - g(\eta)}{Z}, \quad \eta \in [\eta_0, \eta_1].$$

Sampling is performed by drawing $u \sim \mathcal{U}(0,1)$ and solving for η such that $F(\eta) = u$. Since $F(\eta)$ is smooth and strictly monotonic, this inverse can be computed via root-finding or a precomputed lookup table. The method is similar to the implementation in [69, 90].

Uniform weighting in t-space. As discussed in Section B.1, drawing t uniformly from the interval $[0,1]$ amounts to using the proposal density $\rho(\eta) = 1/\alpha^2_{\eta(t)}$. With this baseline choice the objective in Eq. (99) can be optimized in its native form: each Monte-Carlo sample is simply re-weighted by the factor $w(\eta) = \alpha_t^2\, \eta'(t)$, which bundles the forward-process weight α_t^2 and the Jacobian $\eta'(t)$. Although the resulting estimator exhibits higher variance than the importance-sampling schemes introduced earlier, it provides a clear reference point for measuring the effectiveness of less sophisticated designs.

B.3.2 Learned IS

The variance of the Monte-Carlo estimator depends on the learned network $\hat{\mathbf{n}}(\cdot;\boldsymbol{\theta})$. To minimize the variance, we can parameterize the IS with another network and treat the variance as an objective [40, 90]. Actually, learning $\rho(\eta)$ is equivalent to learning a monotone mapping. Thus, we can uniformly sample t, and regard the IS as change-of-variable from η to t:

$$\mathcal{L}_w(\mathbf{x};\boldsymbol{\psi},\boldsymbol{\theta}) = \mathbb{E}_{t\sim\mathcal{U}(0,1)}\mathbb{E}_{\mathbf{x},\mathbf{n}}\left[\eta'(t;\boldsymbol{\psi})\,\mathcal{L}(\mathbf{x},\mathbf{n},\eta;\boldsymbol{\theta})\right], \tag{110}$$

where $\eta'(t;\boldsymbol{\psi}) = d\eta(t;\boldsymbol{\psi})/dt$ and $\eta(t;\boldsymbol{\psi})$ is a monotonic neural network with parameters $\boldsymbol{\psi}$. Since the variance $\mathrm{Var}_{\mathbf{x},\mathbf{n},t}[\eta'(t;\boldsymbol{\psi})\,\mathcal{L}(\mathbf{x},\mathbf{n},\eta;\boldsymbol{\theta})] = \mathbb{E}_{\mathbf{x},\mathbf{n},t}[\big(\eta'(t;\boldsymbol{\psi})\,\mathcal{L}(\mathbf{x},\mathbf{n},\eta;\boldsymbol{\theta})\big)^2] - \big(\mathcal{L}_w(\mathbf{x};\boldsymbol{\psi},\boldsymbol{\theta})\big)^2$, where $\mathcal{L}_w(\mathbf{x};\boldsymbol{\psi},\boldsymbol{\theta})$ is proved [40] invariant to $\eta(t;\boldsymbol{\psi})$, we can minimize $\mathbb{E}_{\mathbf{x},\mathbf{n},t}[\big(\eta'(t;\boldsymbol{\psi})\,\mathcal{L}(\mathbf{x},\mathbf{n},\eta;\boldsymbol{\theta})\big)^2]$ for variance reduction. At this point, we parameterize $\eta(t;\boldsymbol{\psi})$ similar to [40]. where the network consists of 3 linear layers with weights that are restricted to be positive l_1, l_2, l_3, which are composed as $\tilde{\eta}(t;\boldsymbol{\psi}) = l_1(t) + l_3(\mathrm{sigmoid}(l_2(l_1(t))))$. The l_2 layer has 1024 outputs, where l_1 and l_3 have a single output.

We therefore postprocess the monotonic neural network as

$$\eta(t;\boldsymbol{\psi}) = \eta_0 + (\eta_1 - \eta_0)\frac{\tilde{\eta}(t;\boldsymbol{\psi}) - \tilde{\eta}(0;\boldsymbol{\psi})}{\tilde{\eta}(1;\boldsymbol{\psi}) - \tilde{\eta}(0;\boldsymbol{\psi})} \tag{111}$$

where the constants $\eta_0 = -\log(\mathrm{SNR}_{\mathrm{MAX}})$, $\eta_1 = -\log(\mathrm{SNR}_{\mathrm{MIN}})$ define the target SNR range. This construction ensures that $\eta(t;\boldsymbol{\psi})$ remains monotonic and bounded. By adjusting the network parameters $\boldsymbol{\psi}$ we adapt the time-to-noise mapping $t \mapsto \eta(t)$ to match regions with higher loss variance, thereby allocating more samples to informative regions of the diffusion trajectory.

Overhead. While this approach seeks the optimal IS, learning η adds a lightweight auxiliary network training steps and complex gradient operation through $\eta(t;\boldsymbol{\psi})$, but no extra score-network passes. Hence, we use it mainly to benchmark the optimality of hand-designed IS; more aggressive adaptive schemes following [39, 90] are compatible.

B.4 Log-SNR-Timed Channel with Different Variance Functions

Given the central role of σ_t^2 in shaping both the weighting of the loss and the structure of the forward process, it is natural to ask how different functional forms of $\sigma_t^2(\eta)$ affect the overall behaviour of the model. In particular, we investigate how alternative variance functions under the log-SNR parameterization influence the resulting likelihood bound, optimization dynamics, and numerical stability. This section provides both analytical insights and empirical comparisons across several representative schedules. As state in Section B.1, using η timing and denoising score matching estimator, the variance weighted objectives are reformulated as:

$$\mathcal{L}_{\mathrm{DSM}}(\sigma_t^2;\boldsymbol{\theta}) = \frac{1}{2}\int_{\sigma_0^2}^{\sigma_1^2} \mathbb{E}_{\mathbf{n}}\left[\sigma_t^{-2}\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t,\eta_t;\boldsymbol{\theta})\|_2^2\right]\, d\sigma_t^2. \tag{112}$$

Due to the η-timed schedule properties, we replace the time subscript with η, α_η and σ_η are deterministic functions of η without any hyperparameters.

Table 5: Specification of related values and objectives under VP schedule. With $s(\eta) = \text{sigmoid}(\eta)$, $s_- = \text{sigmoid}(-\eta)$, $\kappa = s^{a/2}$.

Formula	Generalized Logistic Sigmoid	Squashed Hyperbolic tangent
α_η	$\sqrt{1 - \left(\frac{1+\tanh(\eta/2)}{2}\right)^a}$	$\sqrt{\frac{1}{1+\exp(2\eta)}}$
σ_η	$\kappa(\eta)$	$\sqrt{\frac{1}{1+\exp(-2\eta)}}$
$\mathcal{L}_{\text{DSM}}$	$\frac{a}{2}\int_{\eta_0}^{\eta_1} s_-(\eta)\,\mathbb{E}_{\mathbf{n}}[\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta; \boldsymbol{\theta})\|_2^2]\, d\eta$	$\int_{\eta_0}^{\eta_1} s_-(2\eta)\,\mathbb{E}_{\mathbf{n}}\left[\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta; \boldsymbol{\theta})\|_2^2\right]\, \mathrm{d}\eta$

Table 6: Specification of related values and objectives under SP schedule. With $s(\eta) = \text{sigmoid}(\eta)$, $s_- = \text{sigmoid}(-\eta)$.

Formula	Generalized Logistic Sigmoid	Squashed Hyperbolic tangent
α_η	$1 - \sqrt{\text{sigmoid}(\eta)^a}$	$1 - \sqrt{\frac{1}{1+\exp(-2\eta)}}$
σ_η	$\sqrt{\text{sigmoid}(\eta)^a}$	$\sqrt{\frac{1}{1+\exp(-2\eta)}}$
$\mathcal{L}_{\text{DSM}}$	$\frac{a}{2}\int_{\eta_0}^{\eta_1} s_-(\eta)\,\mathbb{E}_{\mathbf{n}}[\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta; \boldsymbol{\theta})\|_2^2]\, d\eta$	$\int_{\eta_0}^{\eta_1} s_-(2\eta)\,\mathbb{E}_{\mathbf{n}}\left[\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta; \boldsymbol{\theta})\|_2^2\right]\, \mathrm{d}\eta$

Logistic Sigmoid. We begin with the benchmark variance function $\sigma_\eta^2 = \text{sigmoid}(\eta) := \frac{1}{1+e^{-\eta}}$, which is widely adopted in VP-based diffusion models [40]. This schedule maps variance smoothly into the unit interval, with most of the variation concentrated around $\sigma_\eta^2 = 0$ [90]. Its symmetry and boundedness make it a natural default choice, but also raise questions about its flexibility in capturing tail behaviour and class imbalance. We have the analytic form of the derivative of $d\sigma_\eta^2 = (1-\sigma_\eta^2)\sigma_\eta^2 d\eta$:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \frac{1}{2}\int_{\eta_0}^{\eta_1} \mathbb{E}_{\mathbf{n}}\left[(1-\sigma_\eta^2)\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2\right]\, d\eta. \tag{113}$$

Generalized Logistic Sigmoid. Then we consider the type I generalized logistic sigmoid function [36] $\sigma_\eta^2 = \text{sigmoid}^a(\eta) = (\frac{1}{1+e^{-\eta}})^a$ with a is strict positive coefficient. We have $d\sigma_\eta^2 = a(1 - \text{sigmoid}(\eta))\sigma_\eta^2 d\eta$:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \frac{a}{2}\int_{\eta_0}^{\eta_1} \mathbb{E}_{\mathbf{n}}\left[\text{sigmoid}(-\eta)\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2\right]\, d\eta. \tag{114}$$

Squashed Hyperbolic tangent. Then we consider the Tanh squash function $\sigma_\eta^2 = \frac{1}{2}(\tanh(\eta) + 1)$, we have the analytic form of the derivative of $d\sigma_\eta^2 = 2(1-\sigma_\eta^2)\sigma_\eta^2 d\eta$:

$$\mathcal{L}_{\text{DSM}}(\sigma_t^2; \boldsymbol{\theta}) = \int_{\eta_0}^{\eta_1} \mathbb{E}_{\mathbf{n}}\left[(1-\sigma_\eta^2)\|\mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t, \eta_t; \boldsymbol{\theta})\|_2^2\right]\, d\eta. \tag{115}$$

Thus, we can derive their specific objectives and equivalent predictors using the formula for general noise schedules. We summarize them in Table 5 and Table 6.

B.5 Empirical Results

Tables 8 and 9 report various configurations of the noise variance function and its endpoints in the VP setting. Empirically, increasing the noise variance at the initial stage of the diffusion process yields consistently improved likelihood bounds.

Table 7: Comparison of NLLs and FID on CIFAR-10 under different noise variance settings.

(a) NLL bounds with different noise variances

Likelihood Bounds	$\eta_0 = -8.7$	$\eta_0 = -13.3$
ELBO	3.99	2.75
Ours (SP with IS)	2.49	2.79
Ours (VP with IS)	2.50	2.78

(b) FID of CIFAR-10 with different losses

Loss	$\eta_0 = -8.7$	$\eta_0 = -13.3$
ELBO	14.60	11.9
Ours (VP)	10.18	9.42

Table 8: Performance comparison across generalized sigmoid schedules with varying exponent a, initial log-SNR η_0, and importance sampling strategies on CIFAR-10. Bold denotes the best NLL and FID within each group.

Schedule a	η_0	**Sampling Strategy**	**NLL (↓)**	**FID (↓)**
$a = 1$	-13.3	Uniform in t	2.81	9.79
		IS: Learned Importance	**2.75**	**9.12**
		IS: Uniform in η	2.78	9.42
$a = 1$	-8.7	Uniform in t	2.62	14.9
		IS: Learned Importance	**2.50**	**10.05**
		IS: Uniform in η	2.50	10.18
$a = 2$	-13.3	Uniform in t	3.62	/
		IS: Learned Importance	**3.47**	/
		IS: Uniform in η	3.51	/
$a = 2$	-8.7	Uniform in t	3.16	/
		IS: Learned Importance	3.05	/
		IS: Uniform in η	**3.04**	/
$a = 0.5$	-13.3	Uniform in t	2.54	/
		IS: Learned Importance	**2.45**	/
		IS: Uniform in η	2.47	/
$a = 0.5$	-8.7	Uniform in t	2.61	/
		IS: Learned Importance	**2.31**	/
		IS: Uniform in η	2.32	/

We begin with the benchmark configuration adopted in [40], where $\sigma_t^2 = \mathrm{sigmoid}(\eta(t))$ and the initial negative log-SNR value is set to $\eta_0 = -13.3$. Under this setting, the model achieves an NLL of 2.81 bits/dim after 300K iterations, which is comparable to the ELBO result of 2.75 reported on CIFAR-10 with the same number of iterations.

When using $a = 0.5$, i.e., $\sigma_t^2 = \mathrm{sigmoid}(\eta)^{1/2}$, the results remain promising. However, for $\eta_0 = -8.7$, we note that $\mathrm{sigmoid}(-8.7)^{1/2} \approx 1.29 \times 10^{-2}$, which falls outside the regime where the assumption $\sigma_0^2 \ll 1$ holds strictly, thus violating the theoretical conditions underlying our analysis. Nevertheless, this configuration can be regarded as a practical compromise, yielding strong empirical performance despite the theoretical violation.

We further observe that increasing the warm-up noise variance generally leads to degraded FID scores. This motivates a more systematic investigation of such extreme configurations, as summarized in Tables 9, 10, and 11. Empirically, the likelihood performance appears insensitive to the detailed shape of the variance schedule, but depends primarily on the endpoint values σ_0^2 and σ_1^2. These findings suggest that careful tuning of the endpoint variances can have a larger effect than modifying the overall schedule shape, particularly for likelihood-oriented objectives, an observation consistent with prior analyses in [28, 38, 40].

Table 9: Performance comparison across squashed Hyperbolic tangent schedules with log-SNR endpoints η_0, η_1, and importance sampling strategies on CIFAR-10 with VP. Bold denotes the best NLL within each group.

η_0	η_1	**Sampling Strategy**	**NLL (↓)**
-13.3	5	Uniform in t	3.83
		IS: Learned Importance	**3.77**
		IS: Uniform in η	3.79
-8.7	5	Uniform in t	3.35
		IS: Learned Importance	**3.15**
		IS: Uniform in η	3.18
-4.33	2.5	Uniform in t	2.61
		IS: Learned Importance	**2.51**
		IS: Uniform in η	2.53

Table 10: Performance comparison across generalized sigmoid schedules with varying exponent a, initial log-SNR η_0, and importance sampling strategies on CIFAR-10 with SP. Bold denotes the best NLL and FID within each group.

a	η_0	**Sampling Strategy**	**NLL (↓)**
$a = 1$	-13.3	Uniform in t	2.82
		IS: Learned Importance	**2.77**
		IS: Uniform in η	2.79
$a = 1$	-8.7	Uniform in t	2.62
		IS: Learned Importance	**2.49**
		IS: Uniform in η	2.50
$a = 0.5$	-13.3	Uniform in t	2.53
		IS: Learned Importance	2.47
		IS: Uniform in η	**2.45**
$a = 0.5$	-8.7	Uniform in t	2.61
		IS: Learned Importance	**2.31**
		IS: Uniform in η	2.33

B.6 Likelihood Estimation Comparison for IT-bound and ELBO on CIFAR-10

Tables 12 and 13 report the average loss components of models trained on CIFAR-10. We observe that all IT-based models employing a standard sigmoid variance schedule achieve likelihood estimates that match or surpass those of the ELBO baseline. Notably, when combined with importance sampling, the IT bound consistently outperforms the ELBO across all configurations. However, when using uniform sampling in t-space, the IT-bound appears to trade off a small amount of likelihood for faster convergence, likely due to reduced variance in gradient estimation. We further observe that increasing the warm-up noise variance leads to a notable rise in the total ELBO. This increase is largely driven by the reconstruction term,

$$-\mathbb{E}_{p(\mathbf{y}_0|\mathbf{x})}[\log q(\hat{\mathbf{x}}|\mathbf{y}_0)], \tag{116}$$

which becomes more pronounced as $p(\mathbf{y}_0|\mathbf{x})$ flattens under higher noise levels, making accurate reconstruction more challenging.

Intuitively, the loss structure is grounded in the auto-encoder paradigm [41], where one commonly assumes a factorised posterior $q(\hat{\mathbf{x}}|\mathbf{y}_0)$, implying conditional independence across the elements of $\hat{\mathbf{x}}$. In the context of image data, this assumption translates into the belief that pixel values are

Table 11: Performance comparison across squashed Hyperbolic tangent schedules with log-SNR endpoints η_0, η_1, and importance sampling strategies on CIFAR-10 with SP. Bold denotes the best NLL within each group.

η_0	η_1	**Sampling Strategy**	**NLL (↓)**
-13.3	5	Uniform in t	3.82
		IS: Learned Importance	3.76
		IS: Uniform in η	**3.73**
-8.7	5	Uniform in t	3.32
		IS: Learned Importance	**3.14**
		IS: Uniform in η	3.18
-4.33	2.5	Uniform in t	2.60
		IS: Learned Importance	**2.51**
		IS: Uniform in η	2.51

Table 12: Decomposition of ELBO and Information-Theoretic Bound (IT) with initial endpoint $\eta_0 = -13.3$. Values reported on CIFAR-10 in 310K iterations (↓ lower is better).

Term	**Description**	**Value (bits/dim)**
	Evidence Lower Bound (ELBO)	*Total = 2.794*
Term 1	KL divergence between $p(\mathbf{y}_1\|\mathbf{x})$ and prior $\pi(\mathbf{x})$	0.0012358
Term 2	Reconstruction loss ($-\mathbb{E}_{p(\mathbf{y}_0\|\mathbf{x})}[\log q(\mathbf{x}\|\mathbf{y}_0)]$)	0.0103869
Term 3	Diffusion loss ($\sum \mathbb{E}_{p(\mathbf{y}_{t(i)}\|\mathbf{x})} D_{\mathrm{KL}}[p(\mathbf{y}_{s(i)}\|\mathbf{y}_{t(i)}, \mathbf{x}) \| q(\mathbf{y}_{s(i)}\|\mathbf{y}_{t(i)}; \boldsymbol{\theta})]$)	2.7836967
	Information-Theoretic Bound (IT)	*Total = 2.805*
Term 1	Integrated denoising score matching term	0.7622061
Term 2	Mismatched entropy between $p(\mathbf{y}_1\|\mathbf{x})$ and prior $\pi(\mathbf{x})$)	2.0434874

conditionally independent given the latent code $\mathbf{y}_0$, and each pixel depends only on its corresponding latent component. It follows the form:

$$q(\hat{x}_i|y_{0,i}) \propto p(y_{0,i}|x_i) = \mathcal{N}(y_{0,i}; \alpha_0 x_i, \sigma_0^2), \tag{117}$$

where we normalize over all possible values of x_i. However, we argue that this assumption is overly restrictive and does not hold in practice. Within the auto-encoding framework, $q(\hat{\mathbf{x}}|\mathbf{y}_0)$ can be interpreted as a decoder tasked with reconstructing the data. The approximation may hold when the SNR at $t = 0$ is sufficiently high. In this case, the conditional distribution $p(\mathbf{y}_0|\mathbf{x})$ becomes sharply peaked around $\mathbf{y}_0 = \alpha_0 \mathbf{x}$, effectively imposing strong constraints on the reconstruction loss. Specifically, it induces high sensitivity to noise: small deviations in the warm-up noise level can significantly affect the fidelity of reconstructions. Moreover, since likelihood estimation is inherently sensitive to the probability of individual pixel values and fine-grained image details, this modelling choice represents a practical compromise rather than a principled solution. In effect, the assumption simplifies computation at the expense of capturing complex dependencies among pixels, which can be critical for accurate reconstruction and reliable likelihood evaluation.

Increasing the warm-up noise smooths the sharp concentration of $p(\mathbf{y}_0|\mathbf{x})$, easing reconstruction by reducing the signal-to-noise ratio. In the VP setting, this implies α_0 deviates more from 1, yielding more stable behaviour. Crucially, Theorem 1 ensures robustness under arbitrary noise, allowing principled tuning of the starting noise to balance stability and accuracy. See Appendix. G for more related discussions.

Furthermore, we observe that small perturbations may fail to regularise extreme pixel values introduced during dequantisation, preventing effective mapping into a smooth continuous domain.

Table 13: Decomposition of ELBO and Information-Theoretic Bound (IT) with initial endpoint $\eta_0 = -8.7$. Values reported on CIFAR-10 in 310K iterations (↓ lower is better).

Term	**Description**	**Value (bits/dim)**
	Evidence Lower Bound (ELBO)	*Total = 3.99*
Term 1	KL divergence between $p(\mathbf{y}_1\|\mathbf{x})$ and prior $\pi(\mathbf{x})$	0.0012358
Term 2	Reconstruction loss ($-\mathbb{E}_{p(\mathbf{y}_0\|\mathbf{x})}[\log q(\mathbf{x}\|\mathbf{y}_0)]$)	2.7219771
Term 3	Diffusion loss ($\sum \mathbb{E}_{p(\mathbf{y}_{t(i)}\|\mathbf{x})} D_{\mathrm{KL}}[p(\mathbf{y}_{s(i)}\|\mathbf{y}_{t(i)}, \mathbf{x}) \| q(\mathbf{y}_{s(i)}\|\mathbf{y}_{t(i)}; \boldsymbol{\theta})]$)	1.2691765
	Information-Theoretic Bound (IT)	*Total = 2.51*
Term 1	Integrated denoising score matching term	0.4663643
Term 2	Mismatched entropy between $p(\mathbf{y}_1\|\mathbf{x})$ and prior $\pi(\mathbf{x})$)	2.0435017

Table 14: Likelihood in bits per dimension (BPD) and sample quality (FID scores) on CIFAR-10 and ImageNet-32, for vanilla VDM, MuLAN and ours. "†" indicates the result from [62] for 10K samples generated using an adaptive-step ODE solver.

Model	**CIFAR-10**			**ImageNet-32**		
	Steps	VLB (↓)	FID (↓)	Steps	VLB (↓)	FID (↓)
VDM [40]	10M	2.65	**7.6**	2M	3.72	14.26†
+ MuLAN [62]	2M	2.65	18.54	2M	3.71	**13.19**
Ours ($\eta_0 = -13.3$)	0.3M	2.78	9.42	0.3M	3.28	13.80
Ours ($\eta_0 = -8.7$)	0.3M	**2.50**	10.18	0.3M	**3.01**	14.76

Slightly increasing the initial noise helps suppress such outliers, resulting in improved likelihood behaviour and more stable training. Moreover, in score-based models with noise prediction, a very small starting noise can impair residual estimation near $t = 0$, degrading likelihood due to poor signal-noise separation. We conjecture that adopting a velocity-based parameterisation [55, 90] ($\mathbf{v}$-network) may alleviate this issue.

C Samples Quality and FID

Just as selecting appropriate training and optimization strategies is necessary to achieve strong performance in a given application, so too is the choice of evaluation metric pivotal for drawing valid conclusions. We must stress that the primary focus of this paper is on maximizing the likelihood learning metric, specifically, negative log-likelihood measured in bits-per-dimension (BPD; lower is better), rather than on optimizing Fréchet Inception Distance (FID) or Inception Score (IS), for the following reasons.

Model samples undoubtedly serve as a valuable diagnostic tool, often enabling us to form an intuition about why a model may underperform and how it might be improved. From this standpoint, a generative model ought to produce samples that are indistinguishable from those in the training set, whilst encompassing its full variability. To quantify these properties, a variety of metrics, such as the IS and the FID, have been proposed. However, both qualitative and quantitative assessments based on model samples can be misleading with respect to a model's density-estimation capabilities, as well as its effectiveness in probabilistic modelling tasks beyond image synthesis [74]. Consequently, average log-likelihood remains the de facto standard for quantifying generative image-modeling performance. For many sophisticated models, the average log-likelihood is challenging to compute or even approximate. Indeed, it is possible for a model with sub-optimal log-likelihood to generate visually impressive samples, or conversely, for a model with excellent log-likelihood to produce poor samples, an observation that underlines the lack of a direct relationship between FID and negative log-likelihood (NLL).

Table 15: Comparison of the mean FID scores with standard error for our model on CIFAR-10 with sigmoid noise schedule after 0.3M steps. We provide both FID scores on 10K and 50K samples and with respect to both train and test set.

Model	FID 10K train	FID 10K test	FID 50K train	FID 50K test
Ours ($\eta_0 = -8.7$)	11.85 ± 0.2	12.91 ± 0.2	10.18 ± 0.2	10.90 ± 0.2
Ours ($\eta_0 = -13.3$)	10.16 ± 0.2	11.53 ± 0.3	9.41 ± 0.2	9.50 ± 0.2
DiffEnc [55]	14.6 ± 0.8	18.5 ± 0.7	11.1 ± 0.8	15.0 ± 0.7

Table 16: Comparison of the mean FID scores with standard error for our model on ImageNet-32 with sigmoid noise schedule after 0.3M steps. We provide both FID scores on 10K and 50K samples and with respect to both train and test set.

Model	FID 10K train	FID 10K test	FID 50K train	FID 50K test
Ours ($\eta_0 = -8.7$)	16.90 ± 0.2	17.91 ± 0.2	14.72 ± 0.2	14.76 ± 0.2
Ours ($\eta_0 = -13.3$)	15.76 ± 0.2	16.15 ± 0.3	13.21 ± 0.2	13.80 ± 0.2

From an information-theoretic standpoint, it is well known that maximizing the log-likelihood of a probabilistic model is equivalent to minimizing the KL divergence from the data distribution to the model distribution. By contrast, FID operates by fitting multivariate Gaussian distributions to the embeddings of real and generated images and then measuring their discrepancy via the Fréchet distance (equivalently, the 2-Wasserstein or Earth Mover's distance). Clearly, the mathematical formulations of these two metrics diverge fundamentally: one corresponds to a mismatched estimation problem under a KL-based criterion, while the other embodies an optimal-transport task.

Moreover, FID conflates both fidelity to the real data distribution and the diversity of generated samples into a single score, and its absolute value is highly sensitive to myriad factors, ranging from the number of samples and the particular checkpoint of the feature extractor network to low-level image-processing choices. Consequently, the visual appeal of generated images, as quantified by FID, correlates only imperfectly with a model's log-likelihood performance. In our work, we concentrate on advancing the state of the art in likelihood estimation; although we report FID scores for completeness, we leave the optimization of sample quality to future research.

Although our model was not explicitly optimized for perceptual sample quality, we report FID scores, a standard metric for visual realism, for both our model and VDM on CIFAR-10 and ImageNet-32 (Tables 14, 15, and 16). From these results, we observe that both models achieve comparable FID scores across datasets. Importantly, the reported values vary substantially depending on the number of generated samples and whether FID is computed against the training or test set. As expected, using more samples improves FID, and evaluation against the training set consistently yields better scores, likely due to closer distributional alignment.

Furthermore, among models with similar likelihood performance, such as i-DODE [90], MuLAN [62] and DiffEnc [55], our method not only achieves the best negative log-likelihood but also retains the lowest FID despite requiring substantially fewer training iterations. Conversely, methods like W-PCDM [46] that explicitly optimize for FID exhibit a marked reduction in likelihood performance.

Consistent with prior observations [69, 28, 54], we find that models achieving better log-likelihood often exhibit slightly worse FID scores. Nevertheless, we emphasize that this degradation in FID is minor, and qualitatively, the generated samples from both models are visually indistinguishable (see Figs. 2, 3, 4 and 5).

D Experimental Settings

Datasets We perform all experiments on CIFAR-10 and ImageNet datasets. CIFAR-10 contains 50,000 training and 10,000 test images. The ImageNet variant includes 1,281,149 training and 49,999 test images. Among the two known versions of ImageNet32, we adopt the newer, anti-aliased version [49], which facilitates likelihood training and remains publicly available. The older version used in [69, 40] is no longer accessible. Furthermore, it is notable that ImageNet contains some personal sensitive information and may cause privacy concern [69].

Model Architectures Our model architecture closely follows the design of Variational Diffusion Models (VDMs) [40]. Specifically, we adopt the original U-Net backbone from VDM for pixel-space diffusion without modification. Our diffusion model is parameterized in terms of the η-timed normalized noise predictor. This architecture is optimized for likelihood-based training and includes key design choices such as the removal of internal downsampling and upsampling, and the use of Fourier feature embeddings to improve fine-scale detail prediction. Consistent with VDM's dataset-dependent configurations, we use a U-Net of depth 32 with 128 channels for CIFAR-10, and 256 channels for ImageNet-32. Our model for ImageNet-64 and -128 uses double the depth at 64 ResNet layers in both the forward and backward direction in the U-Net. It also uses a constant number of channels of 256. All models apply a dropout rate of 0.1 in intermediate layers.

Hardware For the ImageNet-64 and -128 experiments, we used a single GPU node with 8 A800s or 8 H20-NVLink. For the CIFAR-10 and ImageNet-32 experiments, the models were trained and evaluated on 4 GPUs spanning several GPUs types like V100, L20s, A40s, and 3090s with float32 precision.

Training We follow the same default training settings as [40]. For all our experiments, we use the Adam optimizer with learning rate 2×10^{-4}, exponential decay rates of $\beta_1 = 0.9$, $\beta_2 = 0.99$ and decoupled weight decay coefficient of 0.01. We also maintain an exponential moving average (EMA) of model parameters with an EMA rate of 0.9999 for evaluation.

For CIFAR-10, the training processes are conducted on a cluster of 4 GPU cards of NVIDIA V100s. We pretrain the model for 0.3 million iterations using a batch size of 128, which takes around 38 hours. Then we finetune the model for 1K iterations using a batch size of 256 and accumulate the gradient for every 4 batches. Note that in related works [49], experiments on ImageNet-32 (new version) are conducted at a larger batch size (512 or 1024), which may improve the results. For ImageNet-32, the training processes are conducted on 4 GPU cards of NVIDIA A800 (80GB). We pretrain the model for 0.3 million iterations using a batch size of 512, which takes around 3 days. Then we finetune the model for 1K iterations using a batch size of 1024 and accumulate the gradient for every 4 batches.

FID We report Fréchet Inception Distance (FID) scores computed on 50,000 generated samples, unless otherwise noted. This follows the standard setup used in [40], with ancestral sampling over 1,000 sampling timesteps. FID is evaluated against both the training and test sets for CIFAR-10 and ImageNet-32. While increasing the warm-up noise level may slightly increase FID scores, we find that the visual quality of generated samples remains comparable (see Figs. 2, 3, 4 and 5).

E Consistency Across Predictors and Corresponding Objectives

As discussed in [40], the diffusion model can be interpreted from three distinct perspectives: as a denoising process, a noise prediction model, and a score-based model. Similarly, our model admits four equivalent parameterizations, with the velocity-based $\mathbf{v}_t$-prediction [63, 90] included in addition to the three canonical forms, which is summarized in Table 17.

Remark Let $\mathbf{v}_t = \alpha_t \mathbf{n} - \sigma_t \mathbf{x}$ and define the instantaneous velocity as $\tilde{\mathbf{v}} = \dot{\alpha}_t \mathbf{x} + \dot{\sigma}_t \mathbf{n}$. We consider four types of predictors, each parameterized by $\boldsymbol{\theta}$, along with their corresponding matching objectives. Each loss is weighted by a positive time-dependent function $w(t)$:

- **Score predictor** $\hat{\boldsymbol{s}}(\mathbf{y}_t; \boldsymbol{\theta})$ with likelihood-weighted score matching loss [69, 72]:

$$\mathcal{J}_{\text{SM}}(w(t); \boldsymbol{\theta}) := \mathbb{E}_t \left[w(t)\, \mathbb{E}_{\mathbf{x},\mathbf{n}} \left[\| \nabla \log p(\mathbf{y}_t) - \hat{\boldsymbol{s}}(\mathbf{y}_t; \boldsymbol{\theta}) \|_2^2 \right] \right].$$

- **Noise predictor** $\hat{\mathbf{n}}(\mathbf{y}_t; \boldsymbol{\theta})$ with standard noise-matching loss [40, 28, 54]:

$$\mathcal{J}_{\text{NL}}(w(t); \boldsymbol{\theta}) := \mathbb{E}_t \left[w(t)\, \mathbb{E}_{\mathbf{x},\mathbf{n}} \left[\| \mathbf{n} - \hat{\mathbf{n}}(\mathbf{y}_t; \boldsymbol{\theta}) \|_2^2 \right] \right].$$

- **Data predictor** $\hat{\mathbf{x}}(\mathbf{y}_t; \boldsymbol{\theta})$ with reconstruction-based data-matching loss [68]:

$$\mathcal{J}_{\text{DL}}(w(t); \boldsymbol{\theta}) := \mathbb{E}_t \left[w(t)\, \mathbb{E}_{\mathbf{x},\mathbf{n}} \left[\| \mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}_t; \boldsymbol{\theta}) \|_2^2 \right] \right].$$

Table 17: Analytical relationships between optimal predictors under the Gaussian forward process.

Predictor Type	Symbol	Optimal Expression	Expressed via
Score	$\boldsymbol{s}^*(\mathbf{y}_t)$	$\nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t)$	Score matching
Noise	$\hat{\mathbf{n}}^*(\mathbf{y}_t)$	$-\sigma_t\, \boldsymbol{s}^*(\mathbf{y}_t)$	Noise prediction
Data	$\hat{\mathbf{x}}^*(\mathbf{y}_t)$	$\dfrac{\mathbf{y}_t - \sigma_t \hat{\mathbf{n}}^*(\mathbf{y}_t)}{\alpha_t}$	Denoising reconstruction
Velocity	$\hat{\tilde{\mathbf{v}}}^*(\mathbf{y}_t)$	$f(t)\mathbf{y}_t - \frac{1}{2}g^2(t)\boldsymbol{s}^*(\mathbf{y}_t, t)$	Flow parameterisation
Velocity (alt.)	$\hat{\tilde{\mathbf{v}}}^*(\mathbf{y}_t)$	$\dfrac{\dot{\alpha}_t}{\alpha_t}\mathbf{y}_t - \left(\dot{\sigma}_t - \dfrac{\dot{\alpha}_t \sigma_t}{\alpha_t}\right)\sigma_t\, \boldsymbol{s}^*(\mathbf{y}_t)$	Score-based ODE

- **Velocity predictor** $\tilde{\mathbf{v}}(\mathbf{y}_t; \boldsymbol{\theta})$ with flow-matching loss [50, 90]:

$$\mathcal{J}_{\text{FM}}(w(t); \boldsymbol{\theta}) := \mathbb{E}_t \left[w(t)\, \mathbb{E}_{\mathbf{x},\mathbf{n}} \left[\|\tilde{\mathbf{v}} - \tilde{\mathbf{v}}(\mathbf{y}_t; \boldsymbol{\theta})\|_2^2 \right] \right].$$

Under the Gaussian forward process, the optimal solutions to these objectives are analytically related, and yield equivalent predictors when appropriately reparameterized. Specifically, they are equivalent by the following relations:

$$\hat{\mathbf{n}}^*(\mathbf{y}_t; \boldsymbol{\theta}) = -\sigma_t \boldsymbol{s}^*(\mathbf{y}_t; \boldsymbol{\theta}) = -\sigma_t \nabla_{\mathbf{y}_t} \log p(\mathbf{y}_t) \tag{118}$$

$$\hat{\mathbf{x}}^*(\mathbf{y}_t; \boldsymbol{\theta}) = \frac{\mathbf{y}_t - \sigma_t \hat{\mathbf{n}}^*(\mathbf{y}_t; \boldsymbol{\theta})}{\alpha_t} \tag{119}$$

$$\hat{\mathbf{v}}_t^*(\mathbf{y}_t, \boldsymbol{\theta}) = \frac{\alpha_t^2 + \sigma_t^2}{\alpha_t} \hat{\mathbf{n}}^*(\mathbf{y}_t; \boldsymbol{\theta}) - \frac{\sigma_t}{\alpha_t} \mathbf{y}_t \tag{120}$$

$$\hat{\tilde{\mathbf{v}}}^*(\mathbf{y}_t; \boldsymbol{\theta}) = \frac{\dot{\alpha}_t}{\alpha_t} \mathbf{y}_t + \left(\dot{\sigma}_t - \frac{\dot{\alpha}_t \sigma_t}{\alpha_t} \right) \hat{\mathbf{n}}^*(\mathbf{y}_t; \boldsymbol{\theta}). \tag{121}$$

The predictor $\hat{\mathbf{v}}_t^*(\mathbf{y}_t, \boldsymbol{\theta})$ defined as a linear combination of noise and data components, represents a static velocity target in the latent space. In contrast, the instantaneous flow $\hat{\tilde{\mathbf{v}}}^*(\mathbf{y}_t; \boldsymbol{\theta}) = d\mathbf{y}_t/dt$ arises from differentiating the forward process with respect to time. The two are related via the temporal dynamics of α_t and σ_t, and coincide when the process is linear and velocity is time-invariant.

F Numerical Stability

Finite-precision arithmetic is fragile for terms of the form $1 - \varepsilon$. In our discrete-time objective, several intermediates are extremely close to one (e.g., cumulative coefficients and survival factors). With a naïve float32 implementation, these values can round to exactly 1, corrupting the computation and yielding incorrect losses/gradients. Prior discrete-time diffusion implementations [28] used float64 to sidestep such issues. In contrast, our formulation is numerically stable enough that float64 is unnecessary; standard float32 suffices.

A numerically problematic term, for example, is the sampling variance $\sigma_{t|s}^2$. It is straightforward [40] to verify that

$$\sigma_{t|s}^2 = -\operatorname{expm1}\big(\operatorname{softplus}(\gamma(s)) - \operatorname{softplus}(\gamma(t))\big), \tag{122}$$

where $\operatorname{expm1}(x) \equiv e^x - 1$ and $\operatorname{softplus}(x) \equiv \log(1 + e^x)$ are numerically stable primitives in common numerical computing packages. Evaluating $\sigma_{t|s}^2$ via (122) avoids catastrophic cancellation near 1 and keeps computations stable in float32.

G Randomized Distribution Smoothing and Dequantization

Modern generative models often lean on the *manifold hypothesis* [3, 52]: real-world high-dimensional data concentrate near a low-dimensional manifold. When the hypothesis holds exactly, the data distribution is singular with respect to the ambient Lebesgue measure and its density is not well

Table 18: Comparison of the NLL for ELBO and our bound on CIFAR-10 in training and testing with 0.3 million iterations.

Model	train	test
ELBO	2.75 ± 0.002	2.79 ± 0.002
Ours ($\eta_0 = -13.3$)	2.79 ± 0.002	2.80 ± 0.003
Ours ($\eta_0 = -8.7$)	2.49 ± 0.002	2.50 ± 0.003

defined. When it holds approximately, only points in a thin neighborhood of the manifold carry non-negligible mass; elsewhere the density is near zero. Consequently, any ambient-space density that tries to fit such data must exhibit sharp transitions (large first-order derivatives, *i.e.,* a large, possibly unbounded, Lipschitz constant), which is notoriously challenging for likelihood-based models.

Furthermore, while natural images are typically stored using 8-bit integers, they are often modeled using densities, *i.e.*, an image is treated as an instance of a continuous random variable. Since the discrete data distribution has differential entropy of negative infinity, this can lead to arbitrary high likelihoods even on test data. To avoid this case, it is becoming best practice to add real-valued noise to the integer pixel values to dequantize the data.

To this end, we propose to address such issues in the density estimation problem via a *warm-start* process. Inspired by the recent success of randomized smoothing techniques in adversarial defense and distribution smoothing [11, 52], we propose to apply randomized smoothing to diffusion generative modeling.

Randomized Distribution Smoothing Unlike [11] where randomized smoothing is applied to a model, and [52] where symmetric random noise is applied to the data distribution, we inject the arbitrary randomized smoothing into both data $p(\mathbf{x})$ and model $q(\hat{\mathbf{x}}; \boldsymbol{\theta})$. Specifically, we convolve an arbitrary isotropic noise distribution with the data distribution and model to obtain the new "smoother" distributions. By choosing an appropriate smoothing distribution, we aim to make warm start process easier than the original learning problem: smoothing facilitates learning in the first stage by making the input distribution fully supported without sharp transitions in the density function; generating a sample given a noisy one is easier than generating a sample from scratch.

Dequantization Mismatch Another representative issue in dequantization [69] method with diffusion models is training-evaluation mismatch. During training, each datapoint is treated as a narrow Gaussian (or logit-normal) [40, 55] centred on the original value, while evaluation typically occurs on data perturbed with uniform noise [74]. This inconsistency introduces a distributional shift between training and test likelihood evaluation. Variational dequantization [27], conditional autoregressive model [52] and soft truncation [38] alleviate this issue by learning the dequantization noise and finding the optimal ϵ, but incur substantial computational cost and convergence instability.

Remarks Our Theorem 1 and Proposition 2 offer a theoretical perspective on these issues without adding extra network structures. From an information-theoretic standpoint, adding small noise to discrete data corresponds to smoothing data, increasing entropy at a rate controlled by the Fisher information of the data distribution. For highly peaked distributions [40, 55], this smoothing is especially effective, substantially increasing entropy and reducing irregularities, thereby providing a better-conditioned target for model fitting. Proposition 2 formally characterizes this entropy increase, while preserving the original KL divergence in the small-noise limit. Table 18 shows the training and evaluation mismatch of ELBO and our methods.

Beyond smoothing effects, Theorem 1 provides a first-order expansion of the KL divergence with respect to small additive noise. Specifically, when training begins from a nonzero noise level σ_0^2, the initial KL objective is reduced by a factor proportional to $\frac{\sigma_0^2}{2} I(p\|q)$. Since this term is always non-negative and strictly positive when $p \neq q$, introducing initial noise simplifies the optimization landscape by suppressing fine-scale discrepancies that are otherwise difficult to capture at the outset. Intuitively, the model first learns to match broader statistical structure before refining finer details, stabilizing gradient flow and avoiding early overfitting to discrete artefacts.

Taken together, Theorem 1 and Proposition 2 clarify why choosing a small but nonzero initial noise level is beneficial. From a signal processing perspective, σ_0^2 trades off approximation accuracy against numerical tractability. If σ_0^2 is too large, the training target becomes overly blurred, requiring extra effort to recover fine details. If it is too small, the model must approximate a distribution with sharp discontinuities or high-frequency details from the outset, which can hinder learning and lead to poor convergence.

Truncated Normal Dequantization We present a training-free dequantization strategy example, recently adopted by [90], that naturally fits the diffusion framework and exemplifies our theoretical results. Let $\mathbf{X}_0 \in \{0, ..., 255\}^D$ denote 8-bit discrete data scaled to $[-1, 1]$. To define a discrete density, we use a continuous model $q(\cdot; \boldsymbol{\theta})$ evaluated on dequantized inputs:

$$Q(\mathbf{x}_0; \boldsymbol{\theta}) = \int_{\mathbf{u} \in [-\frac{1}{256}, \frac{1}{256}]^D} q(\mathbf{x}_0 + \mathbf{u}; \boldsymbol{\theta})\, d\mathbf{u}. \tag{123}$$

This matches the diffusion-based formulation if we write $\mathbf{y}_\epsilon = \alpha_\epsilon \mathbf{x}_0 + \sigma_\epsilon \tilde{\boldsymbol{\epsilon}}$ and choose $\tilde{\boldsymbol{\epsilon}} \sim \mathcal{TN}(0, \mathbf{I}, -\tau, \tau)$ with $\tau = \frac{\alpha_\epsilon}{256\sigma_\epsilon}$. Then $\mathbf{u} = \frac{\sigma_\epsilon}{\alpha_\epsilon}\tilde{\boldsymbol{\epsilon}} \in [-\frac{1}{256}, \frac{1}{256}]^D$ by construction.

Applying a change of variables and accounting for Jacobian terms, the variational lower bound becomes:

$$\log P_0(\mathbf{x}_0) \geq \mathbb{E}_{\tilde{\boldsymbol{\epsilon}} \sim \mathcal{TN}(0, \mathbf{I}, -\tau, \tau)} \left[\log q\Big(\mathbf{x} + \frac{\sigma_\epsilon}{\alpha_\epsilon} \tilde{\boldsymbol{\epsilon}} \Big) - \log p(\tilde{\boldsymbol{\epsilon}}) \right] + D \log \sigma_\epsilon. \tag{124}$$

Using the known entropy of the truncated Gaussian [90], the bound further simplifies to:

$$\log P_0(\mathbf{x}_0) \geq \mathbb{E}_{\hat{\boldsymbol{\epsilon}} \sim \mathcal{TN}(0, \mathbf{I}, -3, 3)} \left[\log q(\hat{\mathbf{x}}_\epsilon) \right] + \frac{D}{2} \log(2\pi e \sigma_\epsilon^2) - 0.01522 \times D. \tag{125}$$

Choosing the Warm-up Noise We further investigate the influence of different *warm-up* noise distributions, Gaussian, Laplace, logistic and Uniform, each scaled to have equal variance. As shown in Table 3, Gaussian noise yields the best performance, closely followed by Laplace and logistic, whereas Uniform significantly underperforms. This result aligns with our theoretical intuition that exponential-family noise distributions, characterized by heavier tails, stabilize training and enhance likelihood estimation. Introducing Laplace noise, previously unexplored in this context, allows us to explicitly examine the impact of heavier-tailed perturbations.

This observation is consistent with the fact that the Gaussian distribution minimizes Fisher information among all distributions with fixed differential entropy [13], and simultaneously maximizes differential entropy among all distributions with the same variance. We attribute the performance gap to tail behavior: Uniform noise has compact support and weakly perturbs extreme values, whereas Laplace and Gaussian assign higher probability mass to large deviations. This results in stronger regularization and more stable gradients. In particular, Laplace noise promotes robustness and sparsity, making it effective for high-dimensional or heavy-tailed data.

We consider a baseline that models the data with a mixture of logistic components. Although this parameterization is, in principle, expressive enough to represent a multimodal distribution, in practice the baseline fails to recover all modes. We attribute this gap to optimization/initialization difficulties that arise when the target density exhibits sharp transitions (i.e., a large Lipschitz constant). By contrast, our method is more robust: it captures the distinct modes even with as few as two mixture components. In this sense, we further generalize the framework of [52] by smoothing both data and model with a shared isotropic kernel and training via the score matching to stabilize optimization on high-Lipschitz targets.

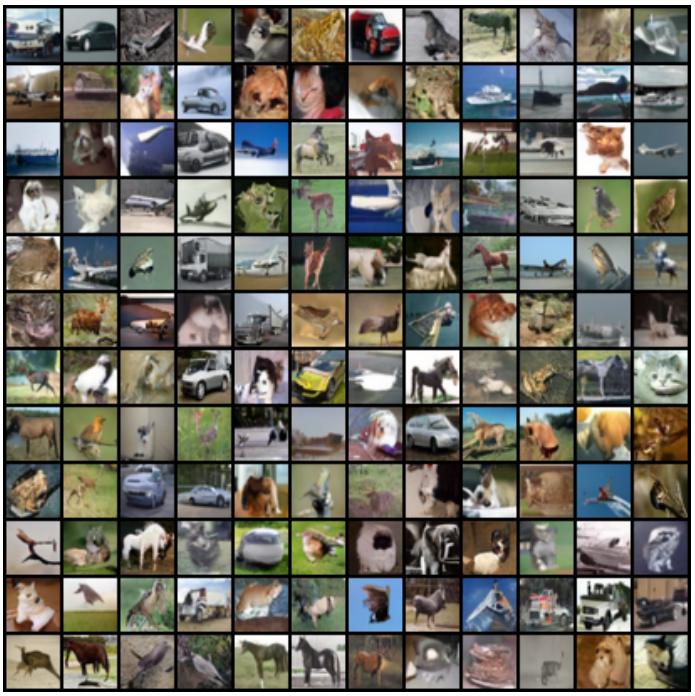

Figure 2: Random samples from our model trained on CIFAR-10 for 300000 parameter updates with EMA. The model was trained in VDM [40] endpoints, and sampled using 1000 sampling timestep.

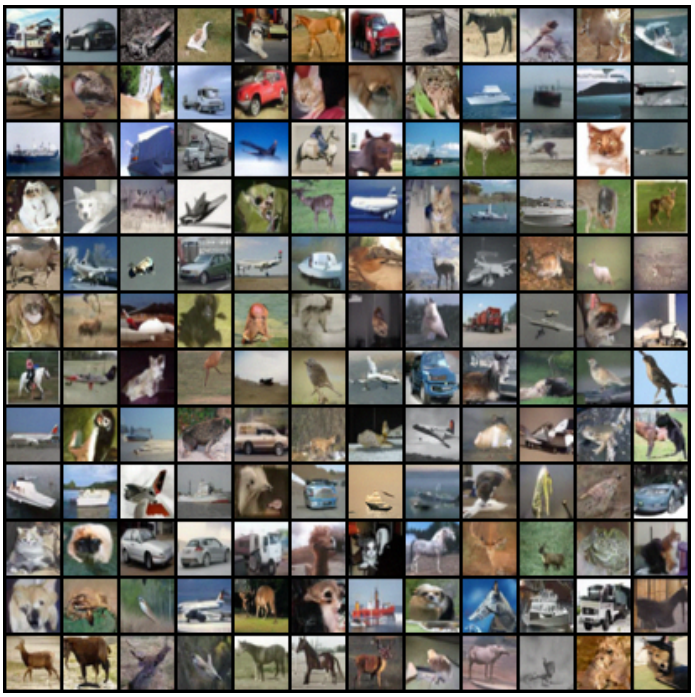

Figure 3: Random samples from our model trained on CIFAR-10 for 300000 parameter updates with EMA. The model was trained with our endpoints, and sampled using 1000 sampling timestep.

Figure 4: Random samples from our model trained on ImageNet32 for 300000 parameter updates. The model was trained in VDM [40] endpoints, and sampled using 1000 sampling timestep.

Figure 5: Random samples from our model trained on ImageNet32 for 300000 parameter updates. The model was trained with our endpoints, and sampled using 1000 sampling timestep.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

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

   Answer: [Yes]

   Justification: See our introduction for a list of claims including connection with relative entropy with score matching with arbitrary noise.

   Guidelines:

   - The answer NA 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 NA 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: Yes, our method improves likelihood estimation but does not construct a generative diffusion process under alternative noise. See the paper for more details.

Guidelines:

- The answer NA 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: Please see our detailed proofs.

   Guidelines:

   - The answer NA 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: Not only we do show all equations and train on standard datasets, we will open source the code.

   Guidelines:

- The answer NA 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: [No]

   Justification: We will open source after paper acceptance.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.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://nips.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, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes, we include all hyperparameters in the paper and will open source code.

   Guidelines:

   - The answer NA 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: **[TODO]**

   Justification: We will report the deviatations for NLL in Appendix.

   Guidelines:

   - The answer NA 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: We provide this in the paper.

   Guidelines:

   - The answer NA 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: Our paper is just a diffusion model useful for density estimation with standard datasets.

   Guidelines:

   - The answer NA 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: [NA]

    Justification: **[TODO]**

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We do not believe our method will have a high risk of abuse as our models are not perceptually SOTA, they only provide for density estimation.

Guidelines:

- The answer NA 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: [NA]

    Justification: We are using standard benchmark datasets.

    Guidelines:

    - The answer NA 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: [Yes]

    Justification: We provide the new official source of ImageNet32.

    Guidelines:

    - The answer NA 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: [NA]

Justification:

Guidelines:

- The answer NA 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: [NA]

    Justification:

    Guidelines:

    - The answer NA 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 rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA 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 (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.