Score Matching Model for Unbounded Data Score

Dongjun Kim
KAIST
dongjoun57@kaist.ac.kr
Seungjae Shin
KAIST
tmdwo0910@kaist.ac.kr
Kyungwoo Song
University of Seoul
kyungwoo.song@uos.ac.kr
Wanmo Kang
KAIST
wanmo.kang@kaist.ac.kr
Il-Chul Moon
KAIST
icmoon@kaist.ac.kr

Abstract

Recent advance in diffusion models incorporates the Stochastic Differential Equation (SDE), which brings the state-of-the-art performance on image generation tasks. This paper improves such diffusion models by analyzing the model at the zero diffusion time. In real datasets, the score function diverges as the diffusion time ($t$) decreases to zero, and this observation leads an argument that the score estimation fails at $t = 0$ with any neural network structure. Subsequently, we introduce Unbounded Diffusion Model (UDM) that resolves the score diverging problem with an easily applicable modification to any diffusion models. Additionally, we introduce a new SDE that overcomes the theoretic and practical limitations of Variance Exploding SDE. On top of that, the introduced Soft Truncation method improves the sample quality by mitigating the loss scale issue that happens at $t = 0$. We further provide a theoretic result of the proposed method to uncover the behind mechanism of the diffusion models.

1 Introduction

Recent advances in the generative modeling enables to create highly realistic images. One direction of such modeling is likelihood-free models [1] based on the minimax training. The other direction is likelihood-based models, in which VAE [2], autoregressive models [3], and flow models [4] are involved. Diffusion models [5] are one of the most successful likelihood-based models where the generative process is modeled by the reverse diffusion process. The success of diffusion models achieves the state-of-the-art performance in image generation [6–8].

This paper improves such diffusion models by analyzing the model at the zero diffusion. When a diffusion model estimates the data score, we observe that the data score diverges as the diffusion time ($t$) decreases to zero. This unbounded data score leads us to prove that the score estimation fails at $t = 0$ even with any network network design. Subsequently, we add an easily applicable modification to the score function, and we name it by Unbounded Diffusion Model (UDM). We provide a theoretic result on UDM that guarantees the successful score estimation. Additionally, we investigate that either 1) Variance Exploding (VE) SDE loses the geometric progression as $t \to 0$ with its exact solution; or 2) VE-SDE loses the ground of modeling the reverse diffusion as the reverse SDE with its approximate solution [7]. This observation motivates us to introduce the Reciprocal VE-SDE (RVE-SDE) to keep the geometric progression even at $t = 0$ and to model the reverse process as the reverse SDE. Also, we introduce a Soft Truncation (ST) trick to improve the sample quality when we progress to $t = 0$. Afterwards, to analyze the ST-loss, we introduce a theoretic result on diffusion models with general weights that gives a new insight on the success of the diffusion models. In experiment, we achieve the state-of-the-art performances in terms of Fréchet Inception Distance (FID) and Negative Log-Likelihood (NLL) on various datasets.

Preprint. Under review.
Variance Preserving (VP) SDE describes the process of DDPM++ by \( \lambda \), where the perturbation kernel, \( p_\sigma(\tilde{x} | x) \), is driven by this Markov chain.

Recently, Song et al. [7] proposes the Continuous Diffusion Model (CDM) that generalizes the connection of the diffusion model with the score-based model. DDPM models the forward diffusion of NCSN is described by a Markov chain of \( x_i = \sqrt{1 - \beta_i} x_{i-1} + \sqrt{\beta_i} z_i \), and CDM reformulates the estimation of the diffusion posterior as the problem of estimating the score function. From this framework, the forward diffusion of NCSN is described by a Markov chain of \( x_i = x_{i-1} + \sqrt{\sigma_i^2 - \sigma_{i-1}^2} z_i \), and the perturbation kernel, \( p_\sigma(\tilde{x} | x) \), is driven by this Markov chain.

Recently, Song et al. [7] proposes the Continuous Diffusion Model (CDM) that generalizes the diffusion process from a discrete Markov chain to a continuous Markov chain driven by Stochastic Differential Equations (SDE): let a SDE is defined by

\[
\mathrm{d}x_t = f(x_t, t) \, \mathrm{d}t + g(t) \, \mathrm{d}w_t,
\]

where \( w_t \) is the standard Wiener process, \( f(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d \) is a drift term, and \( g(\cdot) : \mathbb{R} \to \mathbb{R} \) is a diffusion term. Then, the solution of the SDE is a stochastic process, \( x_t \) (or \( x(t) \)), indexed by time \( t \in [0, T] \), and Song et al. [7] interprets \( x_t \) as the diffused random variable at time \( t \). The continuous diffusion loss is

\[
\mathcal{L}(\theta; \lambda) = \frac{1}{2} \int_0^T \lambda(t) \mathbb{E}_{p_{\sigma}(x_0)} \mathbb{E}_{p_{\sigma_{max}}(x_t | x_0)} \left[ \| s_\theta(\tilde{x}_t, t) - \nabla_{\tilde{x}_t} \log p_{\sigma_{max}}(\tilde{x}_t, x_0) \|_2^2 \right] \, \mathrm{d}t,
\]

where the \( \lambda \) function is the weighting function. In CDM, it is a common practice of truncating the loss integration by \( [\epsilon, T] \) to escape from the training instability, and we denote such loss as \( \mathcal{L}(\theta; \lambda, \epsilon) \).

Variance Preserving (VP) SDE describes the process of DDPM++ by \( \mathrm{d}x_t = -\frac{1}{2} \beta(t) x_t \, \mathrm{d}t + \sqrt{\beta(t)} \, \mathrm{d}w_t \). Applying the Ito’s lemma to a transformation of \( y_t = e^{\frac{1}{2} \int_0^t \beta(s) \, \mathrm{d}s} x_t \), VP-SDE reduces to \( \mathrm{d}y_t = e^{\frac{1}{2} \int_0^t \beta(s) \, \mathrm{d}s} \sqrt{\beta(t)} \, \mathrm{d}w_t \), in which the transformed SDE describes VE-SDE of NCSN++. Therefore, VP-SDE and VE-SDE are equivalent representation of the linear SDEs, and we focus on VE-style SDEs of form \( \mathrm{d}x_t = g(t) \, \mathrm{d}w_t \) for the ease of interpretation in this paper. We emphasize that all below analyses are easily extendable to VP-SDE, and the discrete diffusion processes.

3 Motivation

3.1 Motivating Argument

This subsection investigates what happens if \( \epsilon \) is not sufficiently small enough, where \( \mathcal{L} \) is integrated on \( [\epsilon, T] \). The diffusion kernel of VE-SDE is \( p_{\sigma_{max}}(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma^2(t) I) \), where \( \sigma^2(t) = \ldots \)
Figure 2: Comparison of NCSN++ [7] and UDM with VE-SDE on a toy dataset. Both models estimate the score function on the exterior area of the unit ball well, but the estimated scores by NCSN++ is not outward-pointing inside the unit ball.

\[ \int_0^t g^2(s) \, ds \] is the variance of the diffusion kernel. Thus, the random variable \( \| x_t - x_0 \|^2 / \sigma^2(t) \) follows the \( \chi^2 \)-distribution of degree \( d \), and we observe that the average distance between \( x_t \) and \( x_0 \) is computed by \( E_{p_t(x_0)} E_{p_t(x)} || x_t - x_0 ||^2 = \sigma^2(t) \, d \). If \( \sigma_{\min} = \sigma(\epsilon) \), the diffusion model at best generates \( x_{\sigma_{\min}} \) by reverting the diffusion process, and the observation implies that any generated image \( (x_{\sigma_{\min}}) \) and its paired real image \( (x_0) \) will have the distance of \( \sigma_{\min}^2 \, d \) on average. Thus, the minimum perturbation noise, \( \sigma_{\min} \), is crucial on generating high fidelity images in particular on a high-dimensional dataset. Figure 1 shows that samples with \( \sigma \) (or \( t \)) not sufficiently small is visually noisy.

### 3.2 Motivating Example

This subsection illustrates a toy example that explains reducing \( \epsilon \) down to zero may cause inexact score estimation at \( t \to 0 \). Consider a generative process on spherical coordinate as

\[
\begin{align*}
\rho &\sim \text{Gamma}(\bar{k}, \bar{\theta}), \\
\theta &\sim \text{Uniform}(0, 2\pi),
\end{align*}
\]

with \((\bar{k}, \bar{\theta}) = (100, 0.01)\). Then, the data on \( xy \)-coordinate is distributed as in Figure 2 (a) and the data score of this distribution is illustrated in Figure 2 (b). As the time decreases to zero, the score network, \( s_\theta(x, t) \), should estimate the data score, which diverges at the origin to the infinity. However, as any neural network is locally Lipschitz, the score network cannot estimate a vector field that diverges near the origin as \( t \to 0 \) unless the neural network parameters blow up (see Section 4.1). Empirically, Figure 2 shows that NCSN++ [7] fails to estimate the data score of the inner circle as \( t \to 0 \). On the contrary, the gradient dynamics of the unit ball is well estimated by UDM with VE-SDE as \( t \to 0 \).

### 4 Methodology

#### 4.1 Failure of Score Estimation Near \( t = 0 \)

This section analyzes the failure of the estimation on the score function near the origin. We point out that there is no score network, \( s_\theta(x, t) \), that estimates the score function as \( t \to 0 \). Define \( s_t : \mathbb{R}^d \to \mathbb{R}^d \) as \( s_t(x) = s_\theta(x, t) \), and suppose that \((s_t)_{t>0} \) converges to \( \nabla \log p_r(x) \) as \( t \to 0 \) for some parameter \( \theta \). Then, \( \lim_{t \to 0} s_t(0) = \lim_{t \to 0} \lim_{\epsilon \to 0} s_t(x) = \lim_{\epsilon \to 0} \lim_{t \to 0} s_t(x) = \lim_{\epsilon \to 0} \nabla \log p_r(x) \).

However, the leftmost term is bounded as \( \lim_{t \to 0} \| s_t(0) \|_2 \leq \infty \), but the rightmost term diverges by \( \lim_{\epsilon \to 0} \| \nabla \log p_r(x) \|_2 = \infty \). Therefore, \( s_t \) cannot converge to \( \nabla \log p_r(x) \) as \( t \to 0 \) for any parameters, and the neural network parameters should diverge in order to take \( s_t \) close to \( \nabla \log p_r(x) \).

Lemma 1 states that the above argument holds on a more general setting. See Appendix A for proof.

**Lemma 1.** Let \( \mathcal{H}_{[0,T]} = \{ s : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d, \; s \text{ is locally Lipschitz} \} \). Suppose a continuous vector field \( \mathbf{v} \) defined on a subset \( U \subset \mathbb{R}^d \) is unbounded, then there exists no \( s \in \mathcal{H}_{[0,T]} \) such that \( \lim_{t \to 0} s(x, t) = \mathbf{v}(x) \) a.e. on \( U \).

\[ ^1 \text{We compute } \int_{[0,1]} \| s_\theta(x, \sigma_{\min}) - \nabla_x \log p(x) \|_2^2 \, dx \text{ to measure the estimation accuracy because the score estimation } \text{should} \text{ estimate the data score in the whole ambient space in order to generate realistic images.} \]

\[ ^2 \text{The interchange of limit is satisfied by the Moore-Osgood Theorem.} \]

\[ ^3 \text{Because any neural network is locally uniform with respect to its arguments.} \]
The score network, \( s_\theta \), is an element in \( \mathcal{H}_{[0,T]} \) for any neural architecture, so Lemma 1 indicates that the vanilla score network cannot estimate an unbounded data score with any neural network structure.

### 4.2 Unbounded Continuous Diffusion Model

As any form of neural network is locally Lipschitz, a score network with its second argument as \( t \), \( s_\theta(x, t) \), cannot estimate the data score on the nearby area of diverging point because the locally Lipschitz property is violated on the area. However, we guarantee the existence of a score network that estimates the unbounded data score successfully in Proposition 1, if the second argument of \( s_\theta \) is no longer finite. In general, for any function \( \eta : t \mapsto \mathbb{R} \) that diverges to the infinity as \( t \to 0 \), Proposition 1 states that a locally Lipschitz function of \( s(x, \eta) \) is able to estimate the unbounded data score as \( \eta \to \infty \).

**Proposition 1.** Let \( \mathcal{H}_{[1,\infty)} = \{ s : \mathbb{R}^d \times [1, \infty) \to \mathbb{R}^d, \text{ s is locally Lipschitz} \} \). Suppose a continuous vector field \( \mathbf{v} \) defined on a \( d \)-dimensional open subset \( U \) of a compact manifold \( M \) is unbounded, and the projection of \( \mathbf{v} \) on each axis is locally integrable. Then, there exists \( s \in \mathcal{H}_{[1,\infty)} \) such that \( \lim_{\eta \to \infty} s(x, \eta) = \mathbf{v}(x) \) a.e. on \( U \).

Proposition 1 becomes the foundation of UDM, and the introduction of \( \eta \) is the key to enable the estimation of the unbounded data score. UDM suggests the time input to be processed by \( \eta(t) \), and UDM models the score network as \( s_\theta(x, \eta(t)) \), rather than \( s_\theta(x, t) \). With this modeling, the unbounded estimation is possible with \( \eta s \) such as \( \eta(t) = \frac{1}{t} \) or \( \eta(t) = \frac{1}{\sigma(t)} \). This treatment on time input is applicable to both continuous and discrete diffusion models. See Appendix A for the complete proof.

### 4.3 Reciprocal VE-SDE

As we derive that DDPM++ is reducible to NCSN++ in Section 2, this subsection inspects VE-SDE at \( t = 0 \), and we introduce a new VE-type SDE to overcome both theoretic and practical limitations of VE-SDE. To motivate the need of a new SDE, we observe the difference of VE-SDE on its discrete and continuous counterparts. The discrete VE-SDE was introduced in NCSN \([10, 11]\) that has the perturbation kernel by \( p_0(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma_t^2 I) \) with the geometrically progressive \( \sigma_t \)'s, i.e., \( \sigma_{t-1}/\sigma_t = \gamma \) for some constant \( \gamma \). On the other hand, the perturbation kernel of the continuous VE-SDE becomes \( p_0(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma^2(t) I) \), where \( \sigma^2(t) := \int_0^t g^2(s) \, ds = c[\sigma_{0+t}^2 - 1] \) does not obey the geometric progression at \( t = 0 \), for \( g(t) = \sqrt{2c \log \sigma_0 \sigma_0^t} \). Mathematically, \( \sigma^2(t) \) should satisfy \( \frac{d}{dt} \log \sigma^2(t) = \gamma \) to be geometric, but it turns out that \( \frac{d}{dt} \log \sigma^2(t) \to \infty \) as \( t \to 0 \). This indicates that the geometric progression of the discrete VE-SDE is broken on its continuous counterpart at \( t = 0 \). In order to take both geometric progression and numerical stability, Song et al. \([7]\) approximate the forward diffusion process as the stochastic process defined by a perturbation of \( p_0(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma^{2t} I) \).

With this approximate VE diffusion process, \( x_t \) will not converge to \( x_0 \) in distribution because \( c\sigma_0^{2t} \to c > 0 \) as \( t \to 0 \). Since any solution of SDE is time-continuous in distribution, the non-convergence of \( x_t \) to \( x_0 \) implies that there does not exist any SDE starting from \( t = 0 \) that describes the approximate process. Therefore, the reverse diffusion process of this approximate diffusion process is not modeled by a reverse SDE.

**Proposition 2.** There is no SDE that has the stochastic process \( \{x_t\}_{t \in [0,T]} \) defined by a perturbation kernel \( p_0(x_t | x_0) = \mathcal{N}(x_t; x_0, \sigma^{2t} I) \) as the solution.

This observation is catastrophic because VE-SDE loses its ground on stochastic calculus when computing the log-likelihood via either the Instantaneous Change of Variable \([7]\) or Feynmann-Kac/Girsanov theorems \([12, 13]\); or when sampling new images \([7, 14, 15]\) by solving the reverse SDE \([16]\) through numerical SDE solvers \([17]\). We emphasize that this theoretic analysis becomes a serious issue when it comes to \( t = 0 \), which was not handled in Song et al. \([7]\) with details.

From the motivation above, we introduce Reciprocal Variance Exploding (RVE) SDE whose variance, \( \sigma^2(t) \), is geometric by its nature. RVE-SDE is defined as the form of \( d\mathbf{x}_t = g(t) \, d\mathbf{w}_t \) with

\[
  g(t) := \begin{cases} \sqrt{2c \log \sigma_0^{-1} \sigma_0^{2t/4}} & \text{if } t > 0, \\ 0 & \text{if } t = 0. \end{cases}
\]
We take the continuous-time diffusion loss, \( \mathcal{L}(\theta; \lambda, \epsilon) := \frac{1}{2} \int_t^T (\lambda(t) - \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)) \, dt \), to train the score network, where \( \mathcal{L}_t(\theta) := \mathbb{E}_{p_{\theta}(\mathbf{x}_t)} \mathbb{E}_{p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)}[\|\nabla_{\mathbf{x}_t} \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)\|^2] \). When training the network, the Monte-Carlo estimation of \( \mathcal{L} \) induces the estimation variance, which originates from a pair of sources: the first source is the estimation on the integration over time, \( \int dt \); and the other source is the estimation on the denoising score loss, \( \mathcal{L}_t \). Recent papers \cite{19, 6, 20, 12, 13, 21} apply the importance sampling to minimize the estimation variance by changing the weight function of \( \mathcal{L}(\theta; \lambda, \epsilon) \) from \( \lambda(t) \) to a new weight function, \( \tilde{\lambda}(t) \). The prior works assumed \( \sigma^2(t) \mathcal{L}_t(\theta) \approx \mathbb{E}[\sigma^2(t)\|\nabla_{\mathbf{x}_t} \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)\|^2] = d \), and take \( \tilde{\lambda}(t) = \sigma^2(t) \) for the minimum variance estimator with respect to the integration. We additionally note that the variance of the second source also remains at the identical level by time with \( \tilde{\lambda}(t) = \sigma^2(t) \) because \( \text{Var}(\sigma^2(t)\|\nabla_{\mathbf{x}_t} \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)\|^2) = 2d, \forall t > 0 \), as described by the red dots in Figure 4-(a).

The loss \( \mathcal{L}(\theta; \lambda, \epsilon) \) is the continuous-time variational bound of the log-likelihood when \( \lambda(t) = g^2(t) \) \cite{12, 13}. With the introduced RVE-SDE, we observe that

\[
\mathcal{L}(\theta; g^2(t), \epsilon) = \frac{1}{2} \int_t^T g^2(t) \mathcal{L}_t(\theta) \, dt = (\log \sigma_0^{-1}) \int_t^T \sigma^2(\tau^{-1}) \mathcal{L}_{\tau^{-1}}(\theta) \, d\tau,
\]

by substituting the integrating variable to the reciprocated time because \( g^2(t) = 2 \log \sigma_0^{-1} \sigma^2(t) \) in RVE-SDE. Therefore, the importance sampling of RVE-SDE with \( \lambda(t) = g^2(t) \) is equivalent with drawing the Monte-Carlo diffusion time from a uniform distribution on \( [\tau, \frac{1}{\tau}] \). This equivalence provides the implementation ease that is an advantage of RVE-SDE.

As we push \( t \to 0 \), Figure 4-(a) illustrates a crucial observation that \( \sigma^2(t) \mathcal{L}_t(\theta) \) is not at the same scale because \( \mathcal{L}_t(\theta) \) cannot be simply approximated by \( \mathbb{E}[\|\nabla_{\mathbf{x}_t} \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)\|^2] \). The Monte-Carlo estimation of \( \sigma^2(t) \mathcal{L}_t(\theta) \) (blue) is blowing up when \( t \to 0 \) (or \( \sigma \to 0 \)), while \( \sigma^2(t)\|\nabla_{\mathbf{x}_t} \log p_{\theta}(\mathbf{x}_t | \mathbf{x}_0)\|^2 \) (red) is at the same scale regardless of \( t \). In consequence, the Monte-Carlo samples of \( \mathbf{x}_t \) with low \( t \)’s dominate the gradient signal of the diffusion loss.

The above observation on the imbalanced loss scale by time harms the sample fidelity and the diversity of diffusion models. When a sample is obtained by solving (denoising) the reverse SDE \cite{7} through a numerical SDE solver (or ancestral sampling \cite{22}) if the process is discrete \cite{5}), Figure 3 depicts that the overall sample quality is determined at the initial stage of the denoising process. Therefore, we argue that the diffusion with a large \( t \) attributes the global sample fidelity, and the diffusion with a small \( t \) improves the local sample fidelity, so we emphasize that the loss imbalance issue would damage the global sample quality.

To successfully train the score network without sacrificing the global fidelity, we detour the loss scale issue by the Soft Truncation (ST) trick that is taking different \( \epsilon \) for every mini-batch. In formula,

\[
\mathcal{L}_{ST}(\theta; g^2, \mathbb{P}) = \mathbb{E}_{\mathbb{P}(\epsilon)}[\mathcal{L}(\theta; g^2, \epsilon)],
\]

where \( \mathbb{P}(\epsilon) \) is the prior distribution of \( \epsilon \). In the mini-batch update, we optimize the Monte-Carlo estimation of \( \mathcal{L}(\theta; \lambda, \epsilon) \) with the sampled \( \epsilon \). Figure 4-(b) illustrates that the overall scale of the Monte-Carlo estimation with the hard truncation (blue), i.e., \( \mathbb{P}(\epsilon) \) has a mass one at \( \sigma^{-1}(\sigma_{\text{min}}) \), is driven by a few instances in a mini-batch. With softening the truncation, a mini-batch with high truncation of \( \epsilon = \sigma^{-1}(1) \) (purple) focuses on the score estimation at a large \( t \), and mini-batches with medium \( \epsilon = \sigma^{-1}(0.1) \) (green) or \( \sigma^{-1}(0.01) \) (red) focus on the middle region of \([0, T]\). In consequence,

4The existence and uniqueness of the solution for RVE-SDE is guaranteed by Theorem 5.2.1 in \cite{18} because the diffusion term, \( g \), is bounded on \([0, T]\).
While previous researches [12, 13] reveal the connection between weights, we define the fully generalized diffusion loss as which remains an open question so far. In order to analyze the continuous diffusion loss with general weights, we define the fully generalized diffusion loss as

\[ L(\theta; \lambda) = \int_0^T \mathbb{E}_p(x_0) \mathbb{E}_{p_{t0}(x_t|\xi_t)} \left[ k(x_t, t) \left\| \nabla_{x_t} \log p(x_t) \right\|^2_{\mathbb{R}^d} \right] dt, \]

where \( k \) is now a function of both data and time. The original diffusion loss, \( L(\theta; \lambda) \), can be reframed by this generalized loss with \( k(x_t, t) = \lambda(t) \), \( p = p_r \), and \( q = q_\theta \).

Before we further discuss, let us define the divergence of the fractional orders by \( D_\alpha(p, q) = \mathbb{E}_{p(x)} \left[ \left\| (1 - \Delta)^{\frac{\alpha}{2}} \log \frac{p(x)}{q(x)} \right\|^2 \right] \). Then, this fractional order divergence expands the concept of integer order divergence that \( D_1 \) is equivalent to the original Fisher divergence, \( D_F(p, q) = \mathbb{E}_{p(x)} \left[ \left\| \nabla \log \frac{p(x)}{q(x)} \right\|^2 \right] \), by the Sobolev inequality [24, 25]; and \( D_0 \) upper bounds the square of the KL divergence by the Holder inequality [24]. With this concept, we provide the next theorem:

**Theorem 1.** If \( \frac{\lambda(t)}{g^2(t)} \) is a nondecreasing and nonnegative continuous function, there exists nonnegative weights \( k(x, t) \) and \( \zeta_\alpha(t) \) such that the followings are satisfied:

1. If \( q_t \to p_t \) for all \( t \in [0, T] \), then \( L(p, q; k) \to L(p, q; \lambda) \),
2. For any \( \alpha \in [0, 1] \), \( L(p, q; k) \geq \int_0^T \zeta_\alpha(t) D_\alpha(p_t, q_t) dt \).

Theorem 1 indicates that optimizing \( L_{ST}(\theta; g^2, \mathbb{P}) \) approximately bounds the fractional divergence, \( \int_0^T \zeta_\alpha(t) D_\alpha(p_t, q_{\theta,t}) dt \) because \( \lambda(t) = \left( \int_0^t \mathbb{P}(\epsilon) \mathbb{d}\epsilon \right) g^2(t) \) satisfies the condition of Theorem 1. In particular, when \( \alpha = 0 \), Theorem 1 means that \( L(\theta; g^2, \mathbb{P}) \) approximates the upper bound on the integration of KL divergences, \( \int_0^T \zeta_0(t) D_0(p_t, q_{\theta,t}) dt \geq \int_0^T \zeta_0(t) D_{KL}(p_t, q_{\theta,t}) dt \). In addition, when \( \alpha > 0 \), the effort of estimating the first derivative (score) automatically forces the model to estimate the fractional-ordered derivatives (fractional score). This gives a new insight on the diffusion models because the derivative of integer order is defined locally, but the fractional-ordered derivative is defined globally:

\[
(1 - \Delta)^{\frac{\alpha}{2}} \log p(x_n) = \int e^{2\pi i \xi_n} (1 + |\xi|^2)^{\frac{\alpha}{2}} \left( \int e^{-2\pi i \xi} \log p(x) \mathbb{d}x \right) \mathbb{d}\xi.
\]
Table 1: Performance comparisons on CIFAR-10, Imagenet 32 × 32, Imagenet 64 × 64, CelebA 64 × 64, CelebA-HQ 256 × 256, and STL-10 datasets. The boldfaced numbers present the best performance, and the underlined numbers present the second best performance.

| Model | CIFAR10 32 × 32 | Imagenet 32 × 32 | CelebA 64 × 64 | CelebA-HQ 256 × 256 | STL-10 48 × 48 |
|-------|-----------------|------------------|-----------------|----------------------|----------------|
|       | NLL (↑) | FID (↓) | IS (↑) | NLL (↑) | FID (↓) | IS (↑) |
| UDM (RVE) + ST | 3.04 | 2.33 | 10.11 | 3.59 | 3.32 | 1.93 | 2.78 | 7.16 | 7.71 | 13.43 |

Likelihood-based Models
- CR-NVAE [26] 2.51
- LSGM (FID) [21] 3.43
- DenseFlow-74-10 [4] 2.98
- Gamma Distribution DDIM [27] -
- VDM [8] 2.65
- NCSN++ cont. (deep, VE) [7] 3.43
- DDPM++ cont. (deep, sub-VP) [7] 2.99
- ScoreFlow (cont. norm. flow) [12] 2.74
- Improved DDPM (L_{max}) [19] 3.37

Likelihood-free Models
- StyleGAN2-ADA+Tuning [28] - 2.92 10.02
- Styleformer [29] - 2.82 9.94
- PGGAN [30] - - 8.8
- TransGAN [31] - 9.26 9.02

From this global property, the fractional derivatives of \( p_t \) and \( q_t \) at the training dataset, \( \{x_n\}_{n=1}^N \), largely deviate if the overall distributions out of \( \{x_n\}_{n=1}^N \) are mismatched. In consequence, the estimation of the fractional derivatives forces the model to count the unobserved region \( \mathbb{R}^d \{x_n\}_{n=1}^N \) by simply minimizing the loss computed on the observed data \( \{x_n\}_{n=1}^N \). Therefore, Theorem 1 provides an additional theoretic backbone of the diffusion model with ST-loss.

6 Experiments
6.1 Quantitative Performance

Table 1 compares our UDM with RVE-SDE and ST trick against the current best generative models on CIFAR-10 [32] 32 × 32, downsampled Imagenet [33] 32 × 32 and 64 × 64, CelebA [34] 64 × 64, CelebA-HQ [30] 256 × 256, and STL-10 [35] 48 × 48. We compute bits-per-dim, proportional to NLL, by applying the Instantaneous Change of Variable [36, 7] with the uniform dequantization [37, 38], rather than the variational dequantization [39] to sidestep the need of training an auxiliary flow network. Also, we use the clean-FID introduced in Parmar et al. [40] for computing the FID score. Training and evaluation details are available in Appendix B.

We establish the state-of-the-art model in Table 1. On CIFAR-10, we observe that our model improves the continuous diffusion model with VE-SDE (NCSN++ [7]) in NLL and IS at the expense of sacrificing FID. In particular, UDM surpasses the previous best IS performance (StyleGAN2-ADA+Tuning [28]) by performing 10.11 on CIFAR-10. UDM achieves the state-of-the-art NLL performance on the validation set of Imagenet32 and Imagenet64. In particular, we emphasize that our model surpasses both normalizing flow models (DenseFlow-74-10 [4]) and diffusion models (VDM [3]) on Imagenet. Additionally, UDM trained on CelebA64 is the best performer in terms of the FID performance, but CR-NVAE [26] outperforms our model in NLL, in which we emphasize that there is a chance of improving NLL with UDM by reducing the dequantization gap [42] with alternative dequantization schemes [39, 38]. UDM with high-dimensional CelebA-HQ 256 × 256 performs the state-of-the-art in FID out of the baseline models of the diffusion models (LSGM [21]) and the GAN models (PGGAN [30]). Finally, UDM on STL-10 dataset largely outperforms the baselines by reducing the state-of-the-art FID from 15.17 [29] to 7.71, and IS from 11.01 [29] to 13.43.

6We downsize the dataset from 96 × 96 to 48 × 48 following the baselines [31, 29].
7We compute the NLL performance with the checkpoint released by Song et al. [7]. Other than this performance, all performances in Table 1 are the reported performances.
8For instance, an improved performance of UDM can be obtained from the variational Rényi max (VR-max) approximation with \( K = 50 \) [38] by having 1.86 as NLL on CelebA64, but we do not set VR-max as a default method because it does not guarantee the upper bound of NLL.
When computing the Inception Score (IS) [43], there is a minor discrepancy between likelihood-based models and likelihood-free models. IS-50k in diffusion models [7, 5] computes the score with 50k generated images once, while IS-5k in GAN models [28, 29] computes the score 10 trials with independently generated 5k samples and report the performance as the average of scores. We report IS-50k following Song et al. [7], but we note that IS-5k performs almost identical to IS-50k by performing 10.07 on CIFAR10 and 13.34 on STL-10 in terms of IS-5k.

One particular observation in Table 1 is the IS performance (13.34) of STL-10 that exceeds the number of classes (10) of the dataset. We follow Park and Kim [29], the current state-of-the-art model on STL-10, to train STL-10 with 105k images by aggregating the labeled (5k) and the unlabeled (100k) images [31]. Although the labeled images have 10 classes, the unlabeled images are sampled from a broader distribution of images, so the unlabeled dataset contains the other types of animals such as bears or rabbits, whose classes are different from the labeled classes. Therefore, a well-trained model for the STL-10 dataset with 105k images would perform the IS of which value exceeds the number of labeled classes.

6.2 Ablation Study

Table 2 shows that the suggested soft truncation trick indeed improves the image quality with a big margin on CIFAR-10 dataset. We have experiment the truncation methods with three priors: 1) the first case is the hard truncation (HT) with the delta prior that has measure one at $\sigma^{-1}(\sigma_{\text{min}})$, of which case corresponds to the previous diffusion models [7] with RVE-SDE; 2) the second case is the soft truncation (ST) with the prior being defined on a short region as $P(\epsilon) \propto \frac{1}{\epsilon^2} \mathbf{1}_{[\sigma^{-1}(\sigma_{\text{min}}), \sigma^{-1}(0.01)]}(\epsilon)$; and 3) the last case is with prior of long region as $P(\epsilon) \propto \frac{1}{\epsilon^2} \mathbf{1}_{[\sigma^{-1}(\sigma_{\text{min}}), \sigma^{-1}(\sigma_{\text{max}})]}(\epsilon)$. Table 2 presents that the soft truncation with a uniform prior boosts the sample quality with a small downgrade on NLL, so all experiments in the paper follow this uniform prior on $[\sigma^{-1}(\sigma_{\text{min}}), \sigma^{-1}(\sigma_{\text{max}})]$ as default.

Table 3 demonstrates that lowering the perturbation level ($\sigma$) improves the sample quality on the FFHQ dataset. In Table 3, we use a benchmark image quality assessment metric, Natural Image Quality Evaluator (NIQE) [9], to investigate the importance of the noise level on the image quality. It is common practice [44] in the GAN community to use PSNR and SSIM [45] for the image-wise quality assessment, but those metrics require the noise-free reference images to compare the structural similarity with, in which our setting does not permit such clean images. Instead, NIQE is a reference-free image quality metric that penalizes the blurry and noisy images.
Table 3 indicates that the image generation gets better as the perturbation level decreases in both image-wise (NIQE) and distributional (FID) quality metrics, where NIQE in Table 3 is the averaged NIQE of 1k images. However, empirically, lowering \( \sigma_{min} \) comes at the cost of training instability even with the soft truncation trick. As Figure 6, we introduce \( \sigma_{min} = 0.001 \) to be a default noise level up to the dataset of resolution \( 256 \times 256 \). All the results in this paper except FFHQ \( 1024 \times 1024 \) in Figure 5 and Table 3 are experimented with \( \sigma_{min} = 0.001 \). As described in Section 3.1, the average distance between the sample and the corresponding real image is proportional to the data dimensionality by a factor of the square of \( \sigma_{min}^2 \), so lowering the minimum noise level \( \sigma_{min} \) may generate better samples. We leave the optimal search for \( \sigma_{min} \) of resolutions higher than \( 256 \times 256 \) as further study.

7 Conclusion

This paper tackles the estimation of the exploding data score by introducing UDM based on theoretical analyses. UDM enables the computation of the exact log-likelihood with the aid of RVE-SDE. Also, this paper introduces a theorem that provides a lower bound of the proposed ST-loss as a divergence in Figure 6, we introduce \( \sigma_{min} = 0.001 \) to be a default noise level up to the dataset of resolution \( 256 \times 256 \). All the results in this paper except FFHQ \( 1024 \times 1024 \) in Figure 5 and Table 3 are experimented with \( \sigma_{min} = 0.001 \). As described in Section 3.1, the average distance between the sample and the corresponding real image is proportional to the data dimensionality by a factor of the square of \( \sigma_{min}^2 \), so lowering the minimum noise level \( \sigma_{min} \) may generate better samples. We leave the optimal search for \( \sigma_{min} \) of resolutions higher than \( 256 \times 256 \) as further study.

7.1 Limitation and Potential Negative Social Impact

The major limitation of the diffusion models lies on the training and the evaluation time. Potential risk from this work is the negative usage of the deep generative models, such as deepfake images. The diffusion models enable the high resolution virtual images, and we acknowledge that there is any chance of misuse for malicious purposes.

Acknowledgements

We appreciate to Daehan Park and Jaeyoung Byeon for their review of the proof on Theorem 1.

A Proofs

Lemma 1. Let \( \mathcal{H}_{[0,T]} = \{ s : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d, s \text{ is locally Lipschitz} \} \). Suppose a continuous vector field \( v \) defined on a subset \( U \) of a compact manifold \( M \) (i.e., \( v : U \subset M \to \mathbb{R}^d \) is unbounded, then there exists no \( s \in \mathcal{H}_{[0,T]} \) such that \( \lim_{t \to 0} s(x,t) = v(x) \) a.e. on \( U \).

Proof. Since \( U \) is an open subset of a compact manifold \( M \), \( \| x_1 - x_2 \| \leq \text{diam}(M) \) for all \( x_1, x_2 \in U \). Also, if \( t_1, t_2 \in [0,T], |t_1 - t_2| \) is bounded. Hence, the local Lipschitzness of \( s \) implies that there exists a positive \( K > 0 \) such that \( \| s(x_1,t_1) - s(x_2,t_2) \| \leq K \| x_1 - x_2 \| + |t_1 - t_2| \) for any \( x_1, x_2 \in U \) and \( t_1, t_2 \in [0,T] \). Therefore, for any \( s \in \mathcal{H}_{[0,T]} \), there exists \( C > 0 \) such that \( \| s(x,t) \| < C \) for all \( x \in U \) and \( t \in [0,T] \), which leads no \( s \) that satisfies \( s(x,t) = v(x) \) a.e. on \( U \) as \( t \to 0 \).

Proposition 1. Let \( \mathcal{H}_{[1,\infty)} = \{ s : \mathbb{R}^d \times [1,\infty) \to \mathbb{R}^d, s \text{ is locally Lipschitz} \} \). Suppose a continuous vector field \( v \) defined on a d-dimensional open subset \( U \) of a compact manifold \( M \) is unbounded, and the projection of \( v \) on each axis is locally integrable. Then, there exists \( s \in \mathcal{H}_{[1,\infty)} \) such that \( \lim_{\eta \to \infty} s(x,\eta) = v(x) \) a.e. on \( U \).

Proof. Let \( h \) be a standard mollifier function. If \( h_t(x) = t^{-n} h(x/t) \), then \( v_t := h_t * v \) converges to \( v \) a.e. on \( U \) as \( t \to 0 \) (Theorem 7-(ii) of Appendix C in [24]). Therefore, if we define \( s(x,\eta) := v_{1/\eta}(x) \) on the domain of \( v_{1/\eta}(x) \) and \( s(x,\eta) := 0 \) elsewhere, then \( s(x,\eta) \to v(x) \) a.e. on \( U \) as \( \eta \to \infty \).
A building block of the proof for Theorem 1 is Theorem 2 of Durkan and Song [46]: suppose \( f \) is a function that satisfies \( f(1) = 0 \), and the \( f \)-divergence is given by \( D_f(p|q) = \int q(x)f\left( \frac{p(x)}{q(x)} \right) \, dx \).

**Lemma 2.** [Theorem 2 of [46]] Assuming \( \log p_t(x) \) and \( \log q_t(x) \) are smooth functions which have at most polynomial growth at the infinity and \( p_T = \pi \), where \( \pi \) is the prior distribution, we have

\[
D_f(p_\epsilon, q_\epsilon) = \int_0^T \mathbb{E}_p \left[ g^2(t) f'' \left( \frac{p(x)}{q(x)} \right) \right] \left\| \nabla \log p_t(x) - \nabla \log q_t(x) \right\|_2^2 dt.
\]

**Remark 1.** Note that Lemma 2 does not require \( f \) to be convex. All \( f \) should satisfy are the second differentiability and \( f(1) = 0 \).

**Lemma 3.** [Theorem 1 of [12]] Assuming that the conditions of Theorem 1 in [12] satisfied and let \( p_T = \pi \), where \( \pi \) is the prior distribution. Then, up to a constant,

\[
D_{KL}(p_\epsilon, q_\epsilon) \leq \mathcal{L}(\theta; g^2, \epsilon).
\]

**Remark 2.** Although the original papers [46] and [12] only prove the Lemmas 2 and 3 when \( \epsilon = 0 \), the direct extension of the original proofs yield the general result with \( \epsilon \geq 0 \).

**Theorem 1.** If \( \frac{\lambda(t)}{g^2(t)} \) is a nondecreasing and nonnegative continuous function, there exists nonnegative weights \( k(x, t) \) and \( \zeta_\alpha(t) \) such that the followings are satisfied:

1. If \( q_t \to p_t \) for all \( t \in [0, T] \), then \( \mathcal{L}(p_t, q_t) \to \mathcal{L}(p_t, \lambda) \),

2. For any \( \alpha \in [0, 1] \), \( \mathcal{L}(p_t, q_t) \geq \int_0^T \zeta_\alpha(t) D_{KL}(p_t, q_t) \, dt \).

**Remark 3.** The ST-trick with \( \mathbb{P}(\epsilon) \propto \frac{1}{\epsilon} \mathbbm{1}_{[\xi, \tau]}(\epsilon) \) for \( \xi < \tau \) satisfies the condition of Theorem 1 because

\[
\frac{\lambda(t)}{g^2(t)} = \int_0^t \mathbb{P}(\epsilon) \, d\epsilon = \begin{cases} \left( \frac{1}{\xi} - \frac{1}{\tau} \right) / \left( \frac{1}{\xi} - \frac{1}{\tau} \right) & \text{if } t > \tau \\ \frac{1}{\tau} - \frac{1}{\xi} & \text{if } t \in [\xi, \tau] \\ 0 & \text{if } t < \xi \end{cases}
\]

**Remark 4.** The coefficient \( \zeta_\alpha(t) \) is nonzero for all \( \alpha \in [0, 1] \) only if 1) \( \frac{\lambda(t)}{g^2(t)} > 0 \) and 2) \( \frac{d}{dt} \left( \frac{\lambda(t)}{g^2(t)} \right) > 0 \). Thus, with the prior \( \mathbb{P}(\epsilon) \propto \frac{1}{\epsilon} \mathbbm{1}_{[\xi, \tau]}(\epsilon) \), \( \zeta_\alpha(t) = 0 \) for \( t \not\in [\xi, \tau] \). If \( \xi = \sigma^-1(\sigma_{min}) \) and \( \tau = \sigma^-1(\sigma_{max}) = T \), then \( \zeta_\alpha(t) > 0 \) for \( t \in [\sigma^-1(\sigma_{min}), T] \).

**Proof.** Let us define \( k(x_t, t) := u(t) + b(x_t, t) \int_0^t v(s) \, ds \), where \( u(t) = \frac{1}{2} \left( \lambda(t) + \lambda(0) \frac{g^2(t)}{g^2(0)} \right) \), \( v(t) = \frac{1}{2} \frac{d}{dt} \left( \frac{\lambda(t)}{g^2(t)} \right) \), and \( b(x_t, t) = g^2(t) \left( \log q_t(x_t) + 1 \right) \). Then, as \( q_t \to p_t \), \( \log q_t(x_t) \to 0 \), and \( b(x_t, t) \to g^2(t) \). Hence, we have

\[
k(x_t, t) \to \frac{1}{2} \left( \lambda(t) + \lambda(0) \frac{g^2(t)}{g^2(0)} \right) + \frac{1}{2} g^2(t) \int_0^t \frac{d}{ds} \left( \frac{\lambda(t)}{g^2(t)} \right) \, ds = \lambda(t).
\]
Therefore, we found $k$ such that $\mathcal{L}(p, q; k) \rightarrow \mathcal{L}(p, q; \lambda)$. With our $k$ setting, we prove the inequality by decomposing $\mathcal{L}(p, q; k)$ as

$$
\mathcal{L}(p, q; k) = \frac{1}{2} \int_0^T \mathbb{E}_{p_t(x_t)} \left[ k(x_t, t) \| \nabla_x \log \frac{p_t(x_t)}{q_t(x_t)} \|_2^2 \right] dt \\
= \frac{1}{2} \int_0^T u(t) \mathbb{E}_{p_t(x_t)} \left[ \left\| \nabla_x \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2^2 \right] + \frac{1}{2} \int_0^T \left( \int_0^t (v(s) ds) \right) \mathbb{E}_{p_t(x_t)} \left[ b(x_t, t) \left\| \nabla_x \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2^2 \right] dt \\
= (A) + (B).
$$

To the further derivation, we observe

$$
\frac{1}{2} \int_0^T v(t) \left( \int_0^T A(s) ds \right) dt = \frac{1}{2} \int_0^T \left( \int_0^t (v(s) ds) \right) A(t) dt,
$$

for any function $A$ by the Fubini theorem. Then, the right-hand-side of Eq. 1 becomes the second term of the above decomposition if we plug in $A$ with

$$
A(t) = \mathbb{E}_{p_t(x_t)} \left[ g^2(t) \left( \log \frac{p_t(x_t)}{q_t(x_t)} + 1 \right) \left\| \nabla_x \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2^2 \right].
$$

Hence, the second decomposition is reduced to the left-hand-side of Eq. 1, which is

$$
(B) = \frac{1}{2} \int_0^T v(t) \int_t^T \mathbb{E}_{p_s(x_s)} \left[ g^2(s) \left( \log \frac{p_s(x_s)}{q_s(x_s)} + 1 \right) \left\| \nabla_x \log \frac{p_s(x_s)}{q_s(x_s)} \right\|_2^2 \right] ds dt.
$$

Now, let us recall Lemma 2 for $f(r) := r (\log r)^2$. Since the second derivative is $f''(r) = \frac{2}{3} (\log r + 1)$, we have

$$
D_f(p_t, q_t) = \mathbb{E}_{p_t(x_t)} \left[ \left\| \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2 \right] \\
= \int_t^T \mathbb{E}_{p_s(x_s)} \left[ g^2(s) \left( \log \frac{p_s(x_s)}{q_s(x_s)} + 1 \right) \left\| \nabla_x \log \frac{p_s(x_s)}{q_s(x_s)} \right\|_2 \right] ds,
$$

so the second decomposition becomes

$$
(B) = \frac{1}{2} \int_0^T v(t) \mathbb{E}_{p_t(x_t)} \left[ \left\| \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2 \right] dt.
$$

Therefore, we have

$$
\mathcal{L}(p, q; k) = \frac{1}{2} \int_0^T u(t) \mathbb{E}_{p_t(x_t)} \left[ \left\| \nabla_x \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2 \right] + v(t) \mathbb{E}_{p_t(x_t)} \left[ \left\| \log \frac{p_t(x_t)}{q_t(x_t)} \right\|_2 \right] dt.
$$

To prove the inequality from the above decomposition, let us define $p_{t,n}(x_t) = 1_{\|x_t\|_2 \leq n} p_t(x_t) + 1_{\|x_t\|_2 > n} p_t(x_t)$. Then, the sequence $\{p_{t,n}\}_{n \in \mathbb{N}}$ converges to $p_t$ pointwisely. Also, $p_{t,n} \in A_2(\mathbb{R}^d)$ [47, 48] for any $t, n$. By applying (3.14) of [25] with coefficients $w_1 = w_0 = w_1' = p_{t,n}$, $p = p_0 = p = 2$, $\gamma_0 = 0$, $\gamma_1 = 1$, $\epsilon = 1$, and $\theta = \alpha$, there exists a function $\zeta_\alpha(t)$ for any $\alpha \in (0, 1)$ such that the following inequality holds:

$$
\frac{1}{2} \int_0^T v(t) \mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| \nabla_x \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_2 \right] + v(t) \mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_2 \right] \\
= \lim_{n \to \infty} \frac{1}{2} \int_0^T v(t) \mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| \nabla_x \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_2 \right] + v(t) \mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_2 \right] \\
\geq \lim_{n \to \infty} \zeta_\alpha(t) \left\| \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_{W^{\alpha,2}} + v(t) \mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| \log \frac{p_{t,n}(x_t)}{q_t(x_t)} \right\|_2 \right] (2)
$$

where $\|f\|_{W^{\alpha,2}(p_{t,n})}$ is a weighted Sobolev norm of $\mathbb{E}_{p_{t,n}(x_t)} \left[ \left\| (1 - \Delta)^{\frac{\alpha}{2}} f(x_t) \right\|_2 \right]^{1/2}$. We select $\zeta_0(t) = v(t)$ and $\zeta_1(t) = u(t)$ at the extreme cases.
B Experimental Details

We remain c and σ₀ of RVE-SDE unrevealed in the main paper. To choose the hyperparameters, it satisfies the following system of equations as a minimum requirement:

\[
\begin{align*}
\sigma_{RVE}(\epsilon) &= c \sigma_0^2 = \sigma_{\text{min}}, \\
\sigma_{RVE}(T) &= c \sigma_0^2 = \sigma_{\text{max}},
\end{align*}
\]

in which the solution is given as

\[
\begin{align*}
c &= \frac{\sigma_{\text{min}}^2 \sigma_{\text{max}}^2}{\epsilon^2} \\
\sigma_0 &= \left( \frac{\sigma_{\text{min}}}{\sigma_{\text{max}}} \right)^{\frac{T}{\epsilon^2}},
\end{align*}
\]

where \( \epsilon = 10^{-5}\) and \( T = 1 \). Here, we put \( \sigma_{\text{max}} \) as suggested in Song et al. [7].

For the selection of \( \eta \)-function, we empirically observe that the score network ignores the signal of \( \sigma \) in its response if we put \( \frac{1}{\sigma} \) in the score network as \( \frac{1}{\sigma} \) is degenerate, i.e., it is near zero, when \( \sigma \) is large enough. Observing that the global sample fidelity depends on the estimation accuracy on high \( t \), \( \eta(t) = \frac{1}{\sigma(t)} \) is not desirable for its poor estimation accuracy on high \( t \).

On the other hand, Song and Ermon [11] assumes the score parametrization of \( \frac{s_\theta(x)}{\sigma} \) from the empirical observation that the conditional score network \( s_\theta(x, t) \) is reciprocal to \( \sigma(t) \). However, it is apparent in Figure 4(a) that the reciprocity of \( s_\theta(x, t) \) to \( \sigma \) is violated as \( \sigma \to 0 \). In addition, Jolicoeur-Martineau et al. [14] provides the formula of the optimal score function to be \( s^*(x, t) = \frac{\mathbb{E}_{x\sim N(0, \sigma^2)} [N(0, \sigma^2) - x]}{\sigma^2(t)} \), where the numerator attains no closed-form solution. Therefore, as \( \sigma \to 0 \), conditioning \( \frac{1}{\sigma} \) to the score network rather than \( \log \sigma \) [7] is clearly advantageous because it is only left for the the score network with \( \frac{1}{\sigma} \) to find the order of \( \sigma \).

Summing the above observations, we select \( \eta \) to be a mixture of \( \log \sigma \) (advantageous over \( \frac{1}{\sigma} \) on high \( t \)) and \( \frac{1}{\sigma} \) (advantageous over \( \log \sigma \) on small \( t \)) as

\[
\eta(t) = \begin{cases} 
 \log \sigma(t) & \text{if } \sigma \geq 0.01 \\
 -\frac{d_1}{\sigma(t) + d_2} + d_3 & \text{if } \sigma < 0.01,
\end{cases}
\]

where \( d_1 \) and \( d_2 \) are the coefficients that makes \( \eta(\sigma(t)) \) a continuously differentiable function in terms of \( \sigma \), and \( d_2 = 0.0001 \) is a hyperparameter. The \( \eta \)-function on perturbation noises less than 0.01 is reciprocal to \( \sigma(t) \), and this choice enhances the neural network converging to the data score faster than just using the original design of \( \log \sigma \) [7]. Figure 7 describes \( \eta(\sigma(t)) \) as \( \sigma_{\text{min}} \) varies.

On datasets of resolution \( 32 \times 32 \), we use the batch size of 128, which consumes about 48Gb GPU memory. On STL-10 with resolution \( 48 \times 48 \), we use the batch size of 196, and on datasets of resolution \( 64 \times 64 \), we experiment with 256 batch size. The batch size for the datasets of resolution \( 256 \times 256 \) is 40, which takes nearly 120Gb of GPU memory. On the dataset of \( 1024 \times 1024 \) resolution, we use the batch size of 16, which takes around 120Gb of GPU memory. We use five NVIDIA RTX-3090 GPU machines to train the model exceeding 48Gb, and we use a pair of NVIDIA RTX-3090 GPU machines to train the model that consumes less than 48Gb. We set the signal-to-noise ratio as 0.16 on \( 32 \times 32 \) datasets, 0.17 on \( 48 \times 48 \) and \( 64 \times 64 \) datasets, 0.075 on \( 256 \times 256 \) sized.
datasets, and 0.15 on $1024 \times 1024$. On datasets less than $256 \times 256$ resolution, we iterate 1,000 steps for the predictor-corrector sampler, while we apply 2,000 steps on the high-dimensional datasets.

Throughout the experiments, we use the reverse diffusion [7] for the predictor algorithm, and the Langevin dynamics [49] for the corrector algorithm. For the detailed settings, we release our code at https://github.com/Kim-Dongjun/UDM.

C Further Experimental Results

Figure 8 shows how image is created from the trained model, and Figures from 9 to 14 present randomly generated samples of the trained model. Also, Tables 4 and 5 present the additional experimental results in term of FID on LSUN Bedroom and FFHQ $256 \times 256$.

References

[1] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4401–4410, 2019.

[2] Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. arXiv preprint arXiv:2007.03898, 2020.

[3] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Łukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In International Conference on Machine Learning, pages 4055–4064. PMLR, 2018.
Figure 9: Random samples on CIFAR10.

[4] Matej Grčić, Ivan Grubišić, and Siniša Šegvić. Densely connected normalizing flows. *arXiv preprint arXiv:2106.04627*, 2021.

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020.

[6] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. *arXiv preprint arXiv:2105.05233*, 2021.

[7] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[8] Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *arXiv preprint arXiv:2107.00630*, 2021.

[9] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a “completely blind” image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012.

[10] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *arXiv preprint arXiv:1907.05600*, 2019.
[11] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *arXiv preprint arXiv:2006.09011*, 2020.

[12] Yang Song, Conor Durkan, Iain Murray, and Stefano Ermon. Maximum likelihood training of score-based diffusion models. *arXiv e-prints*, pages arXiv–2101, 2021.

[13] Chin-Wei Huang, Jae Hyun Lim, and Aaron Courville. A variational perspective on diffusion-based generative models and score matching. *arXiv preprint arXiv:2106.02808*, 2021.

[14] Alexia Jolicoeur-Martineau, Rémi Piché-Taillefer, Rémi Tachet des Combes, and Ioannis Mitliagkas. Adversarial score matching and improved sampling for image generation. *arXiv preprint arXiv:2009.05475*, 2020.

[15] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021.

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

[17] Peter E Kloeden and Eckhard Platen. Stochastic differential equations. In *Numerical Solution of Stochastic Differential Equations*, pages 103–160. Springer, 1992.
Figure 11: Random samples on LSUN Church.

[18] Bernt Oksendal. *Stochastic differential equations: an introduction with applications*. Springer Science & Business Media, 2013.

[19] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *arXiv preprint arXiv:2102.09672*, 2021.

[20] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv preprint arXiv:2106.15282*, 2021.

[21] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *arXiv preprint arXiv:2106.05931*, 2021.

[22] Christopher M Bishop. Pattern recognition. *Machine learning*, 128(9), 2006.

[23] Elias M Stein and Rami Shakarchi. *Real analysis: measure theory, integration, and Hilbert spaces*. Princeton University Press, 2009.

[24] Lawrence C Evans. Partial differential equations. *Graduate studies in mathematics*, 19(2), 1998.
Figure 12: Random samples on FFHQ 256.

[25] Daehan Park. Weighted maximal $l_{q}(l_{p})$-regularity theory for time-fractional diffusion-wave equations with variable coefficients. *arXiv preprint arXiv:2103.13673*, 2021.

[26] Samarth Sinha and Adj B Dieng. Consistency regularization for variational auto-encoders. *arXiv preprint arXiv:2105.14859*, 2021.

[27] Eliya Nachmani, Robin San Roman, and Lior Wolf. Non gaussian denoising diffusion models. *arXiv preprint arXiv:2106.07582*, 2021.

[28] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *arXiv preprint arXiv:2006.06676*, 2020.

[29] Jeeseung Park and Younggeun Kim. Styleformer: Transformer based generative adversarial networks with style vector. *arXiv preprint arXiv:2106.07023*, 2021.

[30] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[31] Yifan Jiang, Shiyu Chang, and Zhangyang Wang. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*, 2021.
Figure 13: Random samples on CelebA-HQ 256.

[32] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[33] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In International Conference on Machine Learning, pages 1747–1756. PMLR, 2016.

[34] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaou Tang. Deep learning face attributes in the wild. In Proceedings of the IEEE international conference on computer vision, pages 3730–3738, 2015.

[35] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Proceedings of the fourteenth international conference on artificial intelligence and statistics, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[36] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. arXiv preprint arXiv:1806.07366, 2018.

[37] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. arXiv preprint arXiv:1511.01844, 2015.

[38] Emiel Hoogeboom, Taco S Cohen, and Jakub M Tomczak. Learning discrete distributions by dequantization. arXiv preprint arXiv:2001.11235, 2020.
Figure 14: Random samples on FFHQ 1024

[39] Jonathan Ho, Xi Chen, Aravind Srinivas, Yan Duan, and Pieter Abbeel. Flow++: Improving flow-based generative models with variational dequantization and architecture design. In International Conference on Machine Learning, pages 2722–2730. PMLR, 2019.

[40] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu. On buggy resizing libraries and surprising subtleties in fid calculation. arXiv preprint arXiv:2104.11222, 2021.

[41] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems, 30, 2017.

[42] Didrik Nielsen and Ole Winther. Closing the dequantization gap: Pixelcnn as a single-layer flow. arXiv preprint arXiv:2002.02547, 2020.

[43] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. Advances in neural information processing systems, 29: 2234–2242, 2016.

[44] Kancharla Parimala and Sumohana Channappayya. Quality aware generative adversarial networks. Advances in neural information processing systems, 32:2948–2958, 2019.
[45] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612, 2004.

[46] Conor Durkan and Yang Song. On maximum likelihood training of score-based generative models. *arXiv preprint arXiv:2101.09258*, 2021.

[47] Loukas Grafakos. *Classical fourier analysis*, volume 2. Springer, 2008.

[48] Loukas Grafakos. *Modern fourier analysis*, volume 250. Springer, 2009.

[49] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688. Citeseer, 2011.

[50] Ivan Skorokhodov, Savva Ignatyev, and Mohamed Elhoseiny. Adversarial generation of continuous images. *arXiv preprint arXiv:2011.12026*, 2020.

[51] Chieh Hubert Lin, Chia-Che Chang, Yu-Sheng Chen, Da-Cheng Juan, Wei Wei, and Hwann-Tzong Chen. Coco-gan: Generation by parts via conditional coordinating. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4512–4521, 2019.

[52] Ceyuan Yang, Yujun Shen, Yinghao Xu, and Bolei Zhou. Data-efficient instance generation from instance discrimination. *arXiv preprint arXiv:2106.04566*, 2021.

[53] Ji Lin, Richard Zhang, Frieder Ganz, Song Han, and Jun-Yan Zhu. Anycost gans for interactive image synthesis and editing. *arXiv preprint arXiv:2103.03243*, 2021.

[54] Ivan Anokhin, Kirill Demochkin, Taras Khakhulin, Gleb Sterkin, Victor Lempitsky, and Denis Korzenkov. Image generators with conditionally-independent pixel synthesis. *arXiv preprint arXiv:2011.13775*, 2020.

[55] Edgar Schonfeld, Bernt Schiele, and Anna Khoreva. A u-net based discriminator for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8207–8216, 2020.