The Emergence of Reproducibility and Consistency in Diffusion Models

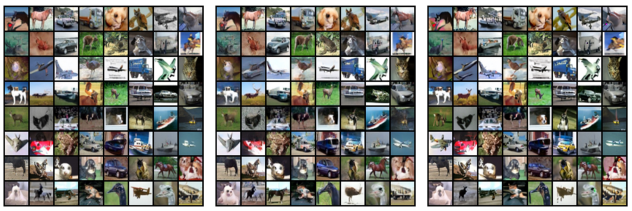
Huijie Zhang [*1] **Jinfan Zhou** [*1] **Yifu Lu** [1] **Minzhe Guo** [1] **Peng Wang** [1] **Liyue Shen** [1] **Qing Qu** [1]

Abstract

In this work, we investigate an intriguing and prevalent phenomenon of diffusion models which we term as "consistent model reproducibility": given the same starting noise input and a deterministic sampler, different diffusion models often yield remarkably similar outputs. We confirm this phenomenon through comprehensive experiments, implying that different diffusion models consistently reach the same data distribution and score function regardless of diffusion model frameworks, model architectures, or training procedures. More strikingly, our further investigation implies that diffusion models are learning *distinct distributions* influenced by the training data size. This is evident in two distinct training regimes: (i) "memorization regime," where the diffusion model overfits to the training data distribution, and (ii) "generalization regime," where the model learns the underlying data distribution. Our study also finds that this valuable property generalizes to many variants of diffusion models, including those for conditional generation and solving inverse problems. Lastly, we discuss how our findings connect to existing research and highlight the practical implications of our discoveries.

1. Introduction

Recently, diffusion models have emerged as a powerful new family of deep generative models with remarkable performance in many applications, including image generation (Ho et al., 2020; Song et al., 2020b; Rombach et al., 2022a) , image-to-image translation (Su et al., 2022; Saharia et al., 2022a; Zhao et al., 2022), text-to-image synthesis (Rombach et al., 2022a; Ramesh et al., 2021; Nichol et al., 2021), and solving inverse problem (Chung et al., 2022b; Song et al., 2022; Chung et al., 2022a; Song et al., 2023a; Li et al., 2024). These models learn an unknown data distribution generated from the Gaussian noise distribution through a process that imitates the non-equilibrium thermodynamic diffusion process (Ho et al., 2020; Song et al., 2020b). In the forward diffusion process, the noise is continuously injected into training samples; while in the reverse diffusion process, a model is learned to remove the noise from noisy samples parametrized by a noise-predictor neural network. Then guided by the trained model, new samples (e.g., images) from the target data distribution can be generated by transforming random noise instances through step-by-step denoising following the reverse diffusion process. Despite the remarkable data generation capabilities, the fundamental mechanisms driving their performance are largely underexplored.



(a) DDPMv4 (b) CT (c) U-ViT

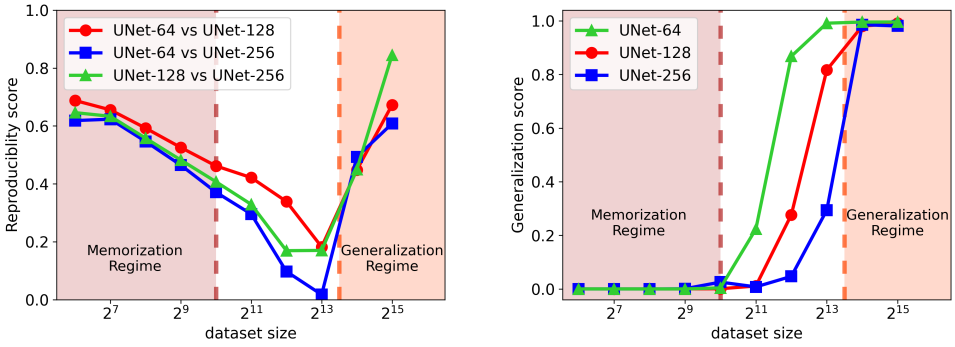
Figure 1. **Visualization of generation samples from different diffusion models.** We utilized denoising diffusion probabilistic models (DDPM) (Ho et al., 2020; Song et al., 2020a), consistency model (CT) (Song et al., 2023b), U-ViT (Bao et al., 2023) trained on CIFAR-10 (Krizhevsky et al., 2009) dataset. Samples in the corresponding row and column are generated from the same initial noise with a deterministic ODE sampler.

In this work, we study an intriguing while prevalent phenomenon that sets diffusion models apart from most other generative models. We refer to this phenomenon as "*consistent model reproducibility*". More precisely, as illustrated in Figure 1, when different diffusion models are trained on the same dataset, and sampled from the *same* noises when using a deterministic ODE sampler.[1]

[*]Equal contribution [1]Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, Location, Ann Arbor, MI 48109-2122, USA. Correspondence to: Huijie Zhang <huijiezh@umich.edu>, Qing Qu <qingqu@umich.edu>.

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

[1]We employ a deterministic sampler to ensure model reproducibility, but stochastic samplers can also achieve reproducibility when they generate consistent noise across different models.



(a) Reproducibilty (b) Generalizability

Figure 2. **"Memorization" and "Generalization" regimes for unconditional diffusion models.** We utilize DDPMv4 and train them on the CIFAR-10 dataset, adjusting both the model's size and the size of the training dataset. In terms of model size, we experiment with UNet-64, UNet-128, and UNet-256, where, for instance, UNet-64 indicates a UNet structure with an embedding dimension of 64. As for the dataset size, we select images from the CIFAR dataset, ranging from 2^6 to 2^{15}. Under each dataset size, different models are trained from the same subset of images. The figure on the left displays the reproducibility score as we compare various models across different dataset sizes, while the figure on the right illustrates the generalizability score of the models as the dataset size changes.

Different diffusion models consistently converge to ***nearly identical*** *image contents,*

which is *irrespective* of network architectures, training and sampling procedures, and perturbation kernels. This phenomenon implies that different diffusion models are learning nearly identical mapping and distributions, as further discussed in Section 3.

More interestingly, through studying the reproducibility under different regimes of training data size, we further find that diffusion models are learning *different types* of data distributions depending on the size of training data. As illustrated in Figure 2, this is corroborated by our findings that the consistent model reproducibility emerges in two distinct regimes: (*i*) "***Memorization regime***": the model has the capacity to memorize the training data but no ability to generate new samples. The co-existence of reproducibility and memorization implies that the diffusion model is learning the empirical multi-delta distribution of the training samples. (*ii*) "***Generalization regime***": the model regains reproducibility while it gains the ability to produce new data. The co-emergence of reproducibility and generalizability indicates that the diffusion model is learning the underlying distribution of the data.

Summary of contributions. In summary, we briefly highlight our contributions below:

- **A comprehensive study of model reproducibility.** We present the first comprehensive and systematic study of the reproducibility in diffusion models. Our findings are consistent under various network architectures, noise perturbation kernels, training and sampling settings.
- **Two regimes of model reproducibility and distribution learning.** Our analysis reveals that reproducibility manifests in two regimes. We demonstrate that diffusion models learn different types of distributions (i.e., empirical vs. underlying distribution) in different regimes.
- **Model reproducibility beyond unconditional diffusion models.** Under various different settings, we show that reproducibility manifest in different but structured ways, including conditional diffusion models, inverse problem solving, fine-tuning.

Theoretical and practical implications of our work. Theoretically, understanding the question will shed light on how the mapping function is learned and constructed between the noise and data distributions. As a deep learning problem with a highly non-convex objective function, the diffusion model reproducibility reflects its robust optimization landscape. Theoretically understanding this robustness could potentially lead to the development of more interpretable deep learning algorithms. In practice, a deeper understanding of model reproducibility could have the potential to (1) improve training efficiency (2) address data privacy issues with generative models (3) yield more interpretable and controllable data generative processes. We discuss this in detail in Section 6.

2. Consistent Model Reproducibility

While the illustrations in Figure 1 and initial investigations in the seminal work (Song et al., 2020b) are motivating, this work provides a more comprehensive and systematic

study of model reproducibility in diffusion models.[2] We begin by proposing quantitative metrics to evaluate reproducibility as well as generalizability in diffusion models. Subsequently, we discover a strong relationship between the reproducibility and generalizability of diffusion models.

2.1. Measures of Reproducibility and Generalizability

Measure of model reproducbility. To study the reproducibility phenomenon in Figure 1 more quantitatively, we introduce the *reproducibility (RP) score* to measure the similarity of image pair generated from two different diffusion models starting from the *same noise*, which is drawn *i.i.d.* from the standard Gaussian distribution:

$$\text{RP Score} \;:=\; \mathbb{P}\left(\mathcal{M}_{\text{SSCD}}(\boldsymbol{x}_1, \boldsymbol{x}_2) > 0.6\right),$$

which measures the *probability* of a generated sample pair $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ from two different diffusion models to have *self-supervised copy detection* (SSCD) similarity $\mathcal{M}_{\text{SSCD}}$ larger than 0.6 (Pizzi et al., 2022; Somepalli et al., 2023b).[3] Higher RP score indicates stronger model reproducibility. In practice, we estimate *RP Score* by the empirical probability using 10K noise samples. The SSCD similarity is first introduced in (Pizzi et al., 2022) to measure the replication between image pair $(\boldsymbol{x}_1, \boldsymbol{x}_2)$, which is defined as follows:

$$\mathcal{M}_{\text{SSCD}}(\boldsymbol{x}_1, \boldsymbol{x}_2) = \frac{\text{SSCD}(\boldsymbol{x}_1) \cdot \text{SSCD}(\boldsymbol{x}_2)}{||\text{SSCD}(\boldsymbol{x}_1)||_2 \cdot ||\text{SSCD}(\boldsymbol{x}_2)||_2}$$

where $\text{SSCD}(\cdot)$ represents a neural descriptor for copy detection of images.

In addition, we also use the *mean-absolute-error (MAE) score* to measure the reproducibility, $\text{MAE Score} := \mathbb{P}\left(\text{MAE}(\boldsymbol{x}_1, \boldsymbol{x}_2) < 15.0\right)$, based upon similar setting with the RP score. $\text{MAE}(\cdot)$ is the operator that measures the mean absolute different of image pairs in the pixel value space ([0, 255]).

Measure of model generalizability. Moreover, we discover a strong relationship between model reproducibility and its generalizability, where the latter refers to the model's ability to produce *new samples* distinct from the ones in the training set. To assess the generalizability of diffusion models, we introduce the *generalization (GL) score* as follows:

$$\text{GL Score} := 1 - \mathbb{P}\left(\max_{i \in [N]} \left[\mathcal{M}_{\text{SSCD}}(\boldsymbol{x}, \boldsymbol{y}_i)\right] > 0.6\right),$$

which is defined based upon the *probability* of maximum $\mathcal{M}_{\text{SSCD}}$ over the training dataset larger than 0.6. Similar to RP score, we empirically sample 10K initial noises to estimate the probability. Intuitively, GL score measures the dissimilarity between the generated sample $\boldsymbol{x}$ and all N samples $\boldsymbol{y}_i$ from the training dataset $\{\boldsymbol{y}_i\}_{i=1}^N$. Higher GL score indicates stronger generalizability.

2.2. Model Reproducibility Manifests in Two Regimes

Based upon RP and MAE scores, we provide comprehensive quantitative studies (see Figure 4) to demonstrate the prevalence of model reproducibility in diffusion models. More interestingly, we discover that the reproducibility of the model arises either through memorization of the training data or by acquiring the ability to generalize. As highlighted in Figure 2, we show that

> *The model reproducibility manifests in two distinct* ***memorization*** *and* ***generalization*** *regimes*,

depending on the size of training data and model capacities. In the following, we discuss the two regimes in detail.

- **"Memorization regime"** characterizes the scenario where the reproducibility is due to the memorization of the training data distribution. As illustrated in the left region of Figure 2a, this regime occurs when the model has much larger capacity than the size of training data. Although the model possesses the ability to reproduce the same results starting from the same noise, the generated samples are only replications of the samples in the training data and the model lacks the ability to generate new samples; see the left region of Figure 2b. In this regime, the emergence of reproducibility is due to the fact that all diffusion models memorize the same multi-delta distribution of training samples. This can be verified by characterizing the closed-form solution of the score function under empirical multi-delta distribution (see Proposition 3.2), and by showing that practical diffusion models converge to such score function (see Figure 3). An in-depth study is provided in Section 3.1. It should noted that, given no generalizability, training diffusion models in this regime might hold limited practical interest.
- **"Generalization regime"** emerges when the diffusion model not only regains its reproducibility but also becomes capable of generating new samples distinct from the training data; see the right region of Figure 2b. This usually happens when the diffusion model is trained on large dataset without full capacity to memorize the whole dataset (Yoon et al., 2023); see the right region of Figure 2a. This is the regime in which diffusion models are commonly trained and employed in practice. As illustrated in Figure 2b, we revealed that there is a clear *transition* from the memorization regime to the generalization

[2] Recent seminal work (Song et al., 2020b) has observed a similar phenomenon (see also subsequent works (Song et al., 2023b; Karras et al., 2022)), but the study in (Song et al., 2020b) remains preliminary.

[3] As demonstrated in (Somepalli et al., 2023b), $\mathcal{M}_{\text{SSCD}} > 0.4$ already exhibits very strong visual similarities.

regime as the training samples increase. In the generalization regime, the model reproducbility co-emerges with the model's generalizability. We believe this is because all diffusion models are learning the same score function of the true underlying data distribution instead of the training data distribution. We provide an in-depth study in Section 3.2.

2.3. Reproducibility is Rare in Generative Models

We end this section by highlighting that only diffusion models appear to consistently exhibit model reproducibility. This property rarely exists in other generative models, with one exception as noted in (Khemakhem et al., 2020).[4] Detailed analysis of model reproducibility of Generative Adversarial Network (GAN) (Goodfellow et al., 2014) and Variational Autoencoder (VAE) (Kingma & Welling, 2013) are provided in Appendix B. In contrast to diffusion models, the observed lack of reproducibility in GANs and VAEs implies that they are not effectively trained to capture the underlying data distribution. This deficiency is a contributing factor to the occurrence of mode collapse in GANs (Arora & Zhang, 2017).

3. Analyzing Reproducibility in Two Regimes

If the diffusion models are learning the same data distribution $p(\boldsymbol{x}_0)$, the result of reproducibility among different diffusion models implies that they are approximating the same score function $\boldsymbol{s}(\boldsymbol{x}_t; t)$ of $p(\boldsymbol{x}_0)$, which can be derived from Tweedie's formula (Luo, 2022) as follows.

Lemma 3.1. *Suppose the distribution learned by diffusion model is $p(\boldsymbol{x}_0)$ and the perturbation kernel $p_t(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\boldsymbol{I})$ with perturbation parameters s_t, σ_t. The ideal score function has the following form*

$$\begin{aligned}\boldsymbol{s}(\boldsymbol{x}_t; t) &= \frac{1}{s_t^2\sigma_t^2}\left(\mathbb{E}_{\boldsymbol{x}_t\sim p_t(\boldsymbol{x}_t)}[\boldsymbol{x}_0|\boldsymbol{x}_t] - \boldsymbol{x}_t\right) \\ &= \frac{1}{s_t^2\sigma_t^2}\left(s_t\frac{\mathbb{E}_{\boldsymbol{x}_0\sim p(\boldsymbol{x}_0)}[\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\boldsymbol{I})\cdot\boldsymbol{x}_0]}{\mathbb{E}_{\boldsymbol{x}_0\sim p(\boldsymbol{x}_0)}[\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\boldsymbol{I})]} - \boldsymbol{x}_t\right).\end{aligned}$$

Furthermore, let $f_{\boldsymbol{s}} : \mathcal{E} \mapsto \mathcal{I}$ be the mapping from the noise space $\mathcal{E}$ to the image space $\mathcal{I}$, by using a deterministic ODE sampler and the score function $\boldsymbol{s}(\boldsymbol{x}_t; t)$. The reproducibility of diffusion models is result from the learned mapping $f_{\boldsymbol{s}}$ is reproducibility. Therefore, to understand the phenomenon in the memorization and generalization regimes, it boils down to understand two questions:

- What data distribution $p(\boldsymbol{x}_0)$ are the diffusion models learning in each regime?
- How well do diffusion models approximate the score function $\boldsymbol{s}(\boldsymbol{x}_t; t)$ of the corresponding distribution $p(\boldsymbol{x}_0)$?

In the following, we study both questions for the memorization regime in Section 3.1 and the generalization regime in Section 3.2, respectively.

3.1. Reproducibility in Memorization Regime

Through a combination of theoretical and experimental study, we show that in the memorization regime,

> *reproducibility is a result of memorizing the* ***training distribution*** $p(\boldsymbol{x}_0) = \frac{1}{N}\sum_{i=1}^{N}\delta(\boldsymbol{x}_0 - \boldsymbol{y}_i)$,

here $p(\boldsymbol{x}_0)$ denotes the multi-delta distribution of the training samples $\{\boldsymbol{y}_i\}_{i=1}^N$ and $\delta(\cdot)$ denotes the Dirac delta function. In the following, we corroborate our claim by (i) deriving the optimal score function of $p(\boldsymbol{x}_0)$ in Proposition 3.2, and by (ii) showing that practical diffusion models converge to the optimal score function in the small data regime; see Figure 4.

Proposition 3.2. *Given a training dataset $\{\boldsymbol{y}_i\}_{i=1}^N$ of N-samples, consider the same setting of Lemma 3.1 with $p(\boldsymbol{x}_0)$ following the empirical multi-delta distribution $p(\boldsymbol{x}_0) = \frac{1}{N}\sum_{i=1}^{N}\delta(\boldsymbol{x}_0 - \boldsymbol{y}_i)$. In this setting, we can show that the score function can be characterized as*

$$\boldsymbol{s}_{emp}(\boldsymbol{x}_t; t) = -\frac{1}{s_t^2\sigma_t^2}\left[\boldsymbol{x}_t - s_t\frac{\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})\boldsymbol{y}_i}{\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})}\right]$$

The proof for Proposition 3.2 can be found in the Appendix C, building upon previous findings from (Karras et al., 2022; Yi et al., 2023). From Proposition 3.2, we can see that the score function $\boldsymbol{s}_{\text{emp}}(\boldsymbol{x}_t; t)$ is purely determined by the given training dataset $\{\boldsymbol{y}_i\}_{i=1}^N$ and perturbation parameters s_t, σ_t.

Moreover, by comparing the reproducibility between the theoretical noise-to-image mapping $f_{\boldsymbol{s}_{\text{emp}}}$ and different practically trained diffusion models, our experiments in Figure 3 (left) demonstrate that the trained networks have a very *high similarity* compared with the theoretical solution when the training data size is small enough. In the meanwhile, the training loss in Figure 3 (right) also converges to the minimum value in this case, which is proven in Appendix C. As such, in the memorization regime when the model has a much larger capacity than the training data, the reproducibility among different diffusion models and the theoretical mapping implies that all diffusion models are approximating the same score function of the empirical multi-delta distribution of the training data. In this regime, the diffusion model lacks the ability to generate new samples.

[4] (Khemakhem et al., 2020) demonstrates that VAE is uniquely identifiable encoding given a factorized prior distribution over the latent variables.

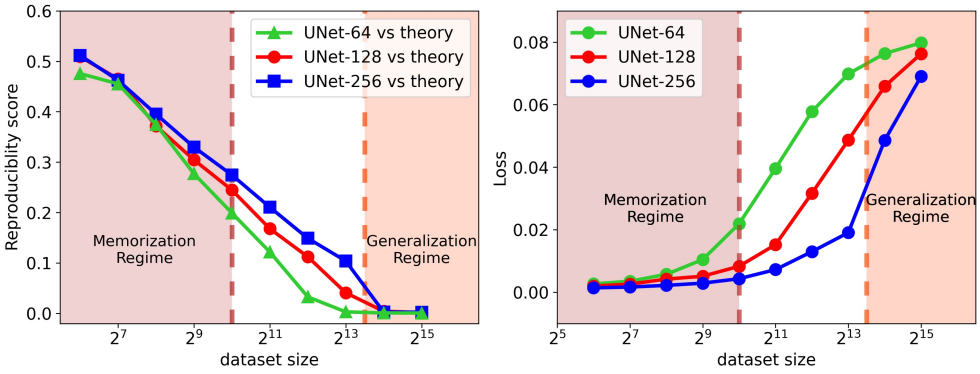


Figure 3. **Convergence of the optimal denoiser (left) and training loss (right) w.r.t. the training data size.** We employ DDPMv4 and conduct training on the CIFAR-10 dataset. During this process, we make modifications to both the model's capacity and the size of the training dataset, maintaining the same configuration as depicted in Figure 2. The left figure illustrates the reproducibility score between each diffusion model and the theoretically unique identifiable encoding as outlined in Proposition 3.2, the right figure illustrates the training loss for these models when trained till converge.

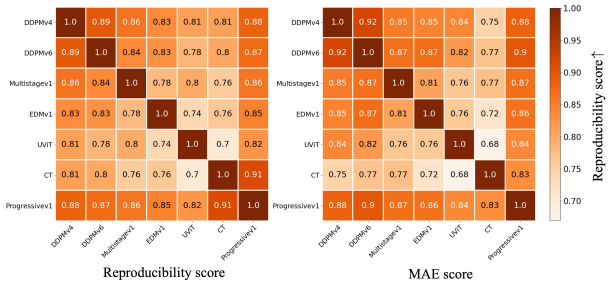


Figure 4. **Similarity among different unconditional diffusion model settings in generalization regime.** We visualize the quantitative results based upon seven different unconditional diffusion models (DDPMv4, DDPMv6 (Ho et al., 2020; Song et al., 2020a), Multistagev1 (Zhang et al., 2023), EDMv1 (Karras et al., 2022), UViT (Bao et al., 2023), CT (Song et al., 2023b), Progressivev1 (Salimans & Ho, 2022)) based upon reproducibility score (left) and MAE score (right) (defined in Section 2.1). About more detailed settings and a more comprehensive comparison could be found in Appendix A.

3.2. Reproducibility in Generalization Regime

Second, we study reproducibility in the generalization regime, which is the typical training setting for most practical diffusion models. Within this regime, we first focus on examining the learning of score function through model reproducibility. Based upon preliminary studies using simple models, we show that in the generalization regime,

> *reproducibility is a byproduct of diffusion model learning the* ***ground-truth distribution*** $p(\mathbf{x}_0)$.

Following this, we conduct a thorough investigation into the reproducibility of various pre-trained diffusion models used in real-world applications.

3.2.1. Reproducibility & Distribution Learning

However, analysis of the estimation accuracy under the true natural image distribution is exceedingly challenging. Instead, we illustrate through empirical evidence that diffusion models have the capacity to learn the underlying distribution by utilizing data samples generated from two *given* distributions: (*i*) a mixture of Gaussian distribution and (*ii*) pre-trained diffusion models.

Case 1: Learning score functions of a mixture of Gaussians. We first consider learning diffusion models based upon the following *mixture of low-rank Gaussian* (MoG) distribution [5]:

$$p(\boldsymbol{x}_0) = \frac{1}{C}\sum_{i\in[C]} \mathcal{N}(\boldsymbol{x}_0; \mathbf{0}, \boldsymbol{\Sigma}_i) \text{ with } \boldsymbol{\Sigma}_i = \boldsymbol{U}_i\boldsymbol{U}_i^\top, \quad (1)$$

where C is the number of classes, and $\boldsymbol{U}_i^* \in \mathbb{R}^{d\times r}$ is the low-rank basis for the ith class with $r \ll d$. In this case, by invoking Lemma 3.1, we can show that the corresponding score function has the following form.

Proposition 3.3. *Under the same setting of Lemma 3.1 with $p(\boldsymbol{x}_0)$ following the MoG distribution introduced in equation 1, we can show that the optimal score function is:*

$$\boldsymbol{s}_{\text{MoG}}(\boldsymbol{x}_t, t) = \sum_{i\in[C]} \frac{\pi_i(\boldsymbol{x}_t, t)}{s_t^2\sigma_t^2}\left(-\boldsymbol{x}_t + \frac{1}{1+\sigma_t^2}\boldsymbol{U}_i\boldsymbol{U}_i^\top\boldsymbol{x}_t\right),$$

with $\pi_i(\boldsymbol{x}_t, t) = \frac{\mathcal{N}(\boldsymbol{x}_t; \mathbf{0}, s_t^2\boldsymbol{U}_i\boldsymbol{U}_i^\top + s_t^2\sigma_t^2\boldsymbol{I}_d)}{\sum_{i\in[C]}\mathcal{N}(\boldsymbol{x}_t; \mathbf{0}, s_t^2\boldsymbol{U}_i\boldsymbol{U}_i^\top + s_t^2\sigma_t^2\boldsymbol{I}_d)}$.

[5] As shown in (Wang & Vastola, 2023), the learned data distribution could be approximated as the Mixture of Gaussian distribution.

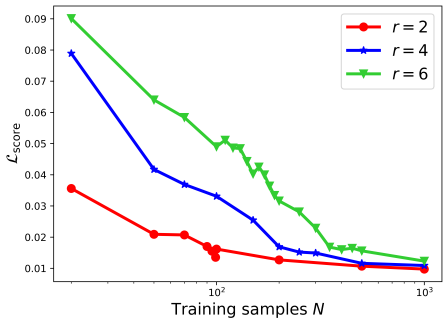


Figure 5. **Score matching accuracy.** We train the same diffusion model with varying numbers of training samples N and subspace dimension r from the Mixture of Gaussian distribution defined in Equation (1) and plot the metric $\mathcal{L}_{\text{score}}$ in different colors for each r. The detailed experimental settings are in Appendix D.1.

The proof can be found in Appendix C. To test whether practical diffusion models converge to the optimal score function $\boldsymbol{s}_{\text{MoG}}(\boldsymbol{x}_t, t)$, we train the diffusion models $\boldsymbol{s}_{\boldsymbol{\theta}}$ by using N data points $\{\boldsymbol{y}_i\}_{i=1}^N \subseteq \mathbb{R}^n$ drawn from the MoG distribution in equation 1. We measure the distance between $\boldsymbol{s}_{\text{MoG}}(\boldsymbol{x}_t, t)$ and $\boldsymbol{s}_{\boldsymbol{\theta}}$ by $\mathcal{L}_{\text{score}}$:

$$\mathcal{L}_{\text{score}} := \mathbb{E}_{\substack{t\sim\mathcal{U}(0,1),\boldsymbol{x}_0\sim p(\boldsymbol{x}_0)\\ \boldsymbol{x}_t\sim p_t(\boldsymbol{x}_t|\boldsymbol{x}_0)}}\left[\|\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t) - \boldsymbol{s}_{\text{MoG}}(\boldsymbol{x}_t, t)\|_2\right],$$

where the expectation is calculated for t uniformly sampled from $[0, 1]$, $\boldsymbol{x}_0$ sampled from the MoG distribution $p(\boldsymbol{x}_0)$ and x_t sampled from the noise perturbation kernel $p_t(\boldsymbol{x}_t|\boldsymbol{x}_0)$ given t and $\boldsymbol{x}_0$. From experiment results shown in Figure 5, we observe that $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ converges to $\boldsymbol{s}_{\text{MoG}}(\boldsymbol{x}_t, t)$ as N increases given different r. Therefore, under this setting of MoG distribution, the diffusion model could converge towards the score function $\boldsymbol{s}_{\text{MoG}}$ given enough training samples (in the generalization regime).

Case 2: Learning score functions from pre-trained diffusion models. Second, suppose the underlying image distribution $p(\boldsymbol{x}_0)$ can be characterized by the noise-to-image mapping $f_{\boldsymbol{s}_{\boldsymbol{\theta}_1}}(\boldsymbol{\epsilon}), \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{0}, s_t^2\sigma_t^2\boldsymbol{I}_d)$ of a pretrained diffusion model in generalization regime $\boldsymbol{s}_{\boldsymbol{\theta}_1}$. We sample N data points from $p(\boldsymbol{x}_0)$ to generate a training dataset $\{\boldsymbol{y}_i\}_{i=1}^N$, based upon which we train another diffusion model $\boldsymbol{s}_{\boldsymbol{\theta}_2}$ with sufficient large N (in the generalization regime). We then calculate the reproducibility of the two models following the same metric as in Section 2.1.

Experimentally, we find that the two models have a **high RP Score=0.80**, which indicates that the diffusion model $f_{\boldsymbol{s}_{\boldsymbol{\theta}_2}}$ could converge to the underlying distribution, which is the same data distribution as $f_{\boldsymbol{s}_{\boldsymbol{\theta}_1}}$, and at the same time they have the same noise-to-image mapping. The detailed experiment settings are in Appendix D.2.

3.2.2. Prevalence of Reproducibility

Finally, we conclude this section by showing the prevalence of reproducibility in the generalization regime, which is irrespective of *network architectures, training and sampling procedures*, and *perturbation kernels*. Specifically, in Figure 4, we visualize the *similarity matrix* for seven different popular diffusion models, where each element of the matrix measures pairwise similarities of two different diffusion models based upon RP score (left) and MAE score (right). All the models are trained with the CIFAR-10 dataset (Krizhevsky et al., 2009). Experimental details and more comprehensive studies can be found in Appendix A.

As we can see from Figure 4, there is a very consistent model reproducible phenomenon for comparing any two models. For even the most dissimilar models, the RP and MAE scores are notably high at 0.7 and 0.68, respectively. Specifically, we observe the following:

- **Different network architectures.** We evaluate (i) U-Net (Ronneberger et al., 2015) based architecture: DDPM (Ho et al., 2020), DDPM++ (Song et al., 2020b), Multistage (Zhang et al., 2023), EDM (Karras et al., 2022), Consistency Training (CT) and Distillation (CD) (Song et al., 2023b), and (ii) Transformer (Vaswani et al., 2017) based architecture: DiT (Peebles & Xie, 2022) and U-ViT (Bao et al., 2023). This phenomenon remains consistent regardless of the specific architecture employed.
- **Different training procedures.** We consider discrete (Ho et al., 2020) and continuous (Song et al., 2020b) settings, training from scratch or distillation (Salimans & Ho, 2022; Song et al., 2023b) for the diffusion model. When we compare CT (consistency training) and EDMv1, even when we use different training losses, they both converge to similar noise-to-image mappings. Additionally, comparing DDPMv1 and Progressivev1 reveals that both training from scratch and distillation approaches lead to the same results.
- **Different sampling procedures.** For sampling, we only use *deterministic* samplers, such as DPM-Solver (Lu et al., 2022), Heun-Solver (Karras et al., 2022), DDIM (Song et al., 2020a) etc. For example, DDPMv4 utilizes DPM-solver, EDMv1 employs a 2nd order heun-solver, and CT utilizes consistency sampling, yet they all exhibit very high model reproducibility.
- **Different perturbation kernels.** For the data corruption process, we compared Variance Preserving (VP) (Ho et al., 2020), Variance Exploding (VE), and sub Variance Preserving (sub-VP) (Song et al., 2020b) perturbation methods for noise perturbation stochastic differential equations. We scale the initial noise using the standard deviation specific to the terminated Gaussian distribution

of each perturbation kernel to ensure a fair comparison, details can be found in Appendix A. Our observations indicate that the choice of perturbation methods (VP, sub-VP, and VE) has a limited impact on reproducibility when comparing DDPMv4, DDPMv6, and EDMv1.

4. Beyond Unconditional Diffusion Models

In this section, we explore the concept of model reproducibility in a broader context, extending beyond unconditional diffusion models. We demonstrate that model reproducibility manifests **more generally** across various scenarios, including conditional diffusion models, diffusion models for inverse problems, and the fine-tuning of diffusion models. Due to space constraints, we defer the results of fine-tuning diffusion models to Appendix H.

4.1. Conditional Diffusion Models

Conditional diffusion model, introduced by (Ho & Salimans, 2022; Dhariwal & Nichol, 2021), gained its popularity in many applications such as text-to-image generation (Rombach et al., 2022a; Ramesh et al., 2021; Nichol et al., 2021). These models achieve a superior degree of control and enhanced quality in output generation through the integration of rich class embeddings within the denoising function. Interestingly, we find that *model reproducibility of conditional models exhibits in a structured way and is strongly related to unconditional counterparts.*

Specifically, our experiments in Figure 6 demonstrate that (*i*) model reproducibility exists among different conditional diffusion models, and (*ii*) model reproducibility presents between conditional and unconditional diffusion models *only* if the type (or class) of content generated by the unconditional models matches that of the conditional models. More results can be found in Appendix E.

To support our claims, we define the *conditional reproducibility score* between different conditional diffusion models by RP_{cond} Score $:= \mathbb{P}\left(\mathcal{M}_{\text{SSCD}}(\boldsymbol{x}_1^c, \boldsymbol{x}_2^c) > 0.6 \mid c \in \mathcal{C}\right)$ to evaluate similarity between outputs of different conditional diffusion models, based on the likelihood of their similarity exceeding a threshold from the same initial noise and conditioned on the class $c \in \mathcal{C}$. We also introduce a between reproducibility score $\text{RP}_{between}$ Score $:= \mathbb{P}\left(\max_{c\in\mathcal{C}}[\mathcal{M}_{\text{SSCD}}(\boldsymbol{x}_1, \boldsymbol{x}_2^c)] > 0.6\right)$, for an unconditional generation $\boldsymbol{x}_1$ and conditional generation $\boldsymbol{x}_2^c$ originating from an identical noise, to assess the similarity between unconditional output $\boldsymbol{x}_1$ and conditional output $\boldsymbol{x}_2^c$.

Results in Figure 6 (a) (b) show that samples from different conditional models (EDM-cond, UViT-cond, MultistageEDM-cond) are similar when conditioned on the same class and noise, supporting Claim (i). On the other hand, a high $\text{RP}_{between}$ Score and visual similarities between unconditional and conditional samples, as seen Figure 6 (c), support Claim (ii).

Furthermore, beside the CIFAR-10 dataset, we also demonstrate the conditional reproducibility on large-scale datasets such as ImageNet (Deng et al., 2009) in Figure 7 and large-scale diffusion models such as Stable Diffusion (Rombach et al., 2022a) in Appendix F.

4.2. Diffusion Models for Inverse Problems

Recently, diffusion models have also been demonstrated as rich, structural priors to solve a broad spectrum of inverse problems (Song et al., 2023a; Chung et al., 2022a; Song et al., 2021; Chung et al., 2022b),[6] including but not limited to image super-resolution, de-blurring, and inpainting. Motivated by these promising results, our illustration is based upon solving the image inpainting problem using a modified deterministic variant of diffusion posterior sampling (DPS) (Chung et al., 2022a), showcasing that for solving inverse problem using diffusion models: *model reproducibility holds only within the same type of network architectures.*

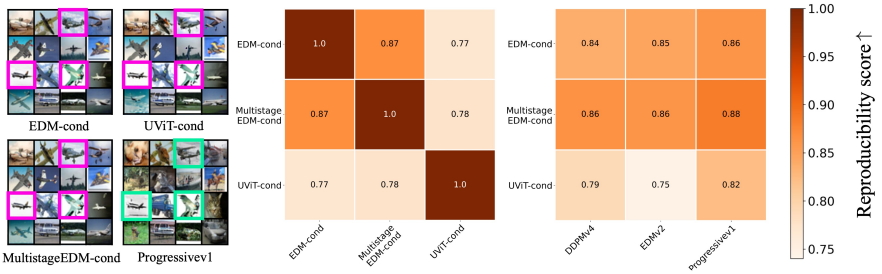
Our claim is supported by the experimental results in Figure 8. Specifically, Figure 8 (a) virtualizes the samples generated from different diffusion models, and Figure 8 (b) presents the similarity matrix of model reproducibility between different models, i.e., U-Net based (DDPMv1, DDPMv2, DDPMv3, DDPMv4, Multistagev1) and Transformer based (DiT, U-ViT) architectures. We note a strong degree of model reproducibility *among* architectures of the same type (e.g., U-Net vs. Transformer), but the model reproducibility score exhibits a notable decrease when any U-Net model is compared with any Transformer-based model.

We conjecture that the lack of reproducibility across network architectures is due to the following reasons: (*i*) DPS introduces the gradient term $\frac{\partial \boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)}{\partial \boldsymbol{x}_t}$ during the sampling, and this extra term might break the reproducibility for different type of architectures. (*ii*) the reproducibility between different types of architectures might not hold for out-of-distribution data generation, whereas the data $\boldsymbol{x}_t$ passed into the learned score function $\boldsymbol{s}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)$ is out-of-distribution for solving inverse problems. We leave these for future study.

5. Related Works

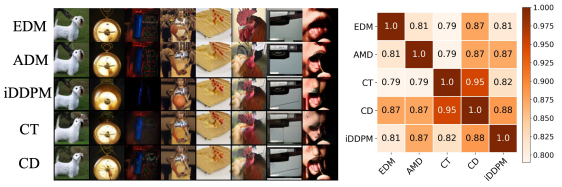
Convergence analysis of diffusion models. Numerous theoretical studies have investigated the diffusion model's convergence towards the underlying distribution. Most of these studies, including (Li et al., 2023a; Chen et al., 2022; Benton et al., 2023; Block et al., 2020; Lee et al., 2022;

[6] Here, the problem is often to reconstruct an unknown signal $\boldsymbol{u}$ from the measurements $\boldsymbol{z}$ of the form $\boldsymbol{z} = \mathcal{A}(\boldsymbol{u}) + \boldsymbol{\eta}$, where $\mathcal{A}$ denotes some (given) sensing operator and $\boldsymbol{\eta}$ is the noise.



(a) Visualization (b) Conditional diffusion models (c) Conditional and unconditional diffusion models

Figure 6. **Model reproducibility for conditional diffusion model in the generalization regime.** In this study, we employ conditional diffusion models, specifically U-Net-based (EDM-cond, MultistageEDM-cond) and transformer based (UViT-cond), which we train on the CIFAR-10 dataset using class labels as conditions. Additionally, we select unconditional diffusion models, namely Progressivev1, DDPMv4, and EDMv2, as introduced in Section 3.2.2. Figure (a) showcases sample generations from both unconditional and conditional diffusion models (with the "plane" serving as the condition for the latter). Notably, samples within the same row and column originate from the same initial noise. The reproducibility scores between the conditional diffusion models are presented in (b), and between unconditional and conditional diffusion models in (c).



(a) Visualization. (b) RP_{cond} Score

Figure 7. **Model reproducibility for conditional diffusion model generations on ImageNet dataset.** In this experiment, we choose the conditional diffusion model (EDM, ADM (Dhariwal & Nichol, 2021), CD, CT, iDDPM (Nichol & Dhariwal, 2021)) trained on the ImageNet dataset. 10K image pairs are generated to estimate the RP_{cond} Score. Due to the complexity of the ImageNet dataset, we set the threshold for the SSCD metric as 0.4 instead of 0.6 here, following the setting in (Somepalli et al., 2023b).

Yingxi Yang & Wibisono, 2022), have established convergence by assuming an L^2-accurate score estimation. Others have explored convergence without relying on this assumption. Nonetheless, these studies rely on strong simplification regarding network architectures (Li et al., 2023b; Chen et al., 2023) and data distributions (Chen et al., 2023). Our paper provides an empirical complement to existing theoretical analyses. In contrast, our paper focuses on the learned distribution and score function under various practical diffusion model settings. The empirical findings not only broadens the understanding of diffusion models in realistic settings but also bridges the gap between theory and practice.

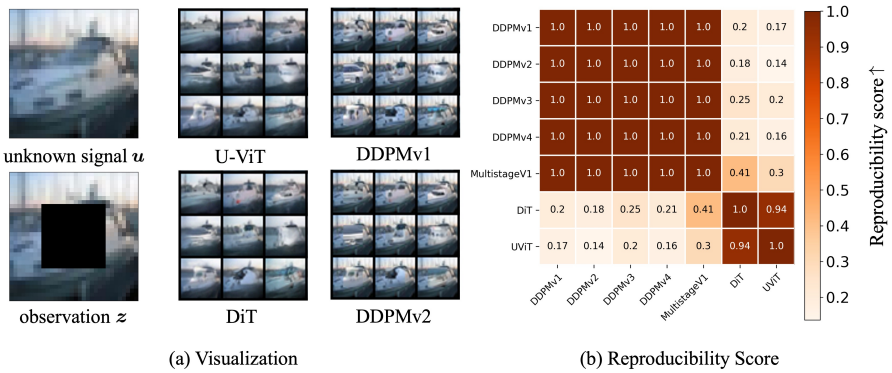
Understanding memorization & generalization. Recently, Yoon et al. (2023) categorized the training regimes of diffusion models into memorization and generalization, concluding that diffusion models tend to generalize when they fail to memorize the training data. In the memorization regime, Yi et al. (2023); Gu et al. (2023) demonstrated that training diffusion models converges towards an optimal denoiser. In contrast, in the generalization regime, Pidstrigach (2022) linked generalization in simple settings to avoiding overfitting, while Kadkhodaie et al. (2023) showed that the generalization capabilities of diffusion models arise from an implicit bias towards geometry-adaptive harmonic bases. Furthermore, Somepalli et al. (2023a;b); Carlini et al. (2023) revealed that diffusion models can still replicate training samples even in the generalization regime, leading to significant privacy concerns.

In comparison, our work takes a step further to delve into the problem. By examining the largely overlooked reproducibility phenomenon, our work is the first to show that diffusion models learn distinct distributions in different regimes: in the memorization regime, diffusion models learn the empirical distribution, while in the generalization regime, they learn the underlying distribution. Moreover, our research provides the first empirical evidence that diffusion models can overcome the curse of dimensionality when learning the underlying distribution, enabling effective generalization even with a limited number of training samples. Finally, our analysis also extends to conditional diffusion models and diffusion models for inverse problems, which have not been addressed in previous studies.

6. Conclusions and Implications

This study focuses on an important phenomenon in diffusion models, which we term "consistent model reproducibility". We believe this intriguing phenomenon could significantly impact future research on diffusion models. Below, we outline several promising directions:

Improving training efficiency. The potential of this work to improve the training efficiency of diffusion models lies in



(a) Visualization (b) Reproducibility Score

Figure 8. **Model reproducibility for solving inverse problems in the generalization regime.** In this investigation, we employ various unconditional diffusion models, as introduced in Section 3.2.2, which were initially trained on the CIFAR-10 dataset. Our approach involves utilizing a modified deterministic variant of diffusion posterior sampling (DPS), as detailed in Appendix G. Specifically, we focus on the task of image inpainting. Figure (a) presents both the observation $\boldsymbol{z}$, unknown signal $\boldsymbol{u}$, and generations from different diffusion models. Notably, samples within the same row and column originate from the same initial noise. The reproducibility scores for different diffusion models under the DPS algorithm are quantitatively analyzed in (b).

leveraging the distinct relationship between noise and image spaces. Recent research (Zhang et al., 2024) illustrates this by delineating the training of diffusion models into three stages, each employing networks of varying sizes. This approach capitalizes on the reproducibility phenomenon, indicating that adequately parameterized networks learn the same score function. Consequently, by appropriately adjusting parameter sizes for each stage, empirical evidence shows that the proposed method surpasses existing techniques, particularly in improving training efficiency in the generalization regime. These findings imply that incorporating reproducibility as a guiding principle in training diffusion models holds significant promise for future research endeavors.

Black-box model privacy. Several commercial, large-scale diffusion models, e.g. Imagen (Saharia et al., 2022b), DALL-E (Betker et al., 2023), are designed as black-box systems, raising significant privacy concerns due to the property of reproducibility. Our analysis, in the Case 2 of Section 3.2.1, indicates that one can replicate the mapping from a trained diffusion model $f_{\boldsymbol{s}_{\boldsymbol{\theta}}}$ by training a new score function $\boldsymbol{s}_{\boldsymbol{\theta}'}$ from generated data by $f_{\boldsymbol{s}_{\boldsymbol{\theta}}}$ (through the open-source API). Furthermore, given the white-box duplication $f_{\boldsymbol{s}_{\boldsymbol{\theta}'}}$, gradient-based adversarial attacking (Guo et al., 2021) and training data privacy (Carlini et al., 2023) would arise as more exacerbated problems.

Controllable data generation. Given the unique mapping learned by the diffusion model, we could control image distribution by manipulating the noise distribution. More specifically, in text-driven image generation, image distribution could be manipulated for adversarial attacking (Zou et al., 2023), robust defending (Zhu et al., 2023), copyright protection (Somepalli et al., 2023b;a). In solving inverse problems, one recent paper (Liu et al., 2023a) manipulated the noise distribution for more efficient sampling. Beyond, the image distribution could also be designed to reduce the uncertainty and variance in our signal reconstruction (Jalal et al., 2021; Chung & Ye, 2022; Luo et al., 2023a).

Acknowledgement

HJZ, YFL, PW, and QQ acknowledge support from NSF CAREER CCF-2143904, NSF CCF-2212066, NSF CCF-2212326, NSF IIS 2312842, ONR N00014-22-1-2529, an AWS AI Award, and a gift grant from KLA. LYS and QQ acknowledge support from MICDE Catalyst Grant, and LYS also acknowledges the support from the MIDAS PODS Grant. Results presented in this paper were obtained using CloudBank, which is supported by the NSF under Award #1925001, and the authors acknowledge efficient cloud management framework SkyPilot (Yang et al., 2023) for computing. The authors acknowledge valuable discussions with Prof. Jeffrey Fessler (U. Michigan), Prof. Saiprasad Ravishankar (MSU), Prof. Rongrong Wang (MSU), Prof. Weijie Su (Upenn), Dr. Ruiqi Gao (Google DeepMind), Mr. Bowen Song (U. Michigan), Mr. Xiao Li (U. Michigan), Mr. Zekai Zhang (U. Tsinghua), Dr. Ismail R. Alkhouri (U. Michigan and MSU)

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. On potential negative social impact is discussed in Section 6. Given the reproducibility, commercial black-box diffusion models are susceptible to replication, adversarial attacks, and leaks of training data.

References

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Arora, S. and Zhang, Y. Do gans actually learn the distribution? an empirical study. *arXiv preprint arXiv:1706.08224*, 2017.

Bao, F., Nie, S., Xue, K., Cao, Y., Li, C., Su, H., and Zhu, J. All are worth words: A vit backbone for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22669–22679, 2023.

Benton, J., De Bortoli, V., Doucet, A., and Deligiannidis, G. Linear convergence bounds for diffusion models via stochastic localization. *arXiv preprint arXiv:2308.03686*, 2023.

Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2:3, 2023.

Block, A., Mroueh, Y., and Rakhlin, A. Generative modeling with denoising auto-encoders and langevin sampling. *arXiv preprint arXiv:2002.00107*, 2020.

Brock, A., Donahue, J., and Simonyan, K. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramer, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. In *32nd USENIX Security Symposium (USENIX Security 23)*, pp. 5253–5270, 2023.

Chen, M., Huang, K., Zhao, T., and Wang, M. Score approximation, estimation and distribution recovery of diffusion models on low-dimensional data. *arXiv preprint arXiv:2302.07194*, 2023.

Chen, S., Chewi, S., Li, J., Li, Y., Salim, A., and Zhang, A. R. Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions. *arXiv preprint arXiv:2209.11215*, 2022.

Chung, H. and Ye, J. C. Score-based diffusion models for accelerated mri. *Medical Image Analysis*, 80:102479, 2022.

Chung, H., Kim, J., Mccann, M. T., Klasky, M. L., and Ye, J. C. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022a.

Chung, H., Sim, B., Ryu, D., and Ye, J. C. Improving diffusion models for inverse problems using manifold constraints. *Advances in Neural Information Processing Systems*, 35:25683–25696, 2022b.

De Bortoli, V., Thornton, J., Heng, J., and Doucet, A. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.

Delbracio, M. and Milanfar, P. Inversion by direct iteration: An alternative to denoising diffusion for image restoration. *arXiv preprint arXiv:2303.11435*, 2023.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A. H., Chechik, G., and Cohen-Or, D. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Gu, X., Du, C., Pang, T., Li, C., Lin, M., and Wang, Y. On memorization in diffusion models. *arXiv preprint arXiv:2310.02664*, 2023.

Guo, C., Sablayrolles, A., Jégou, H., and Kiela, D. Gradient-based adversarial attacks against text transformers. *arXiv preprint arXiv:2104.13733*, 2021.

Han, L., Li, Y., Zhang, H., Milanfar, P., Metaxas, D., and Yang, F. Svdiff: Compact parameter space for diffusion fine-tuning. *arXiv preprint arXiv:2303.11305*, 2023.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jalal, A., Arvinte, M., Daras, G., Price, E., Dimakis, A. G., and Tamir, J. Robust compressed sensing mri with deep generative priors. *Advances in Neural Information Processing Systems*, 34:14938–14954, 2021.

Kadkhodaie, Z., Guth, F., Simoncelli, E. P., and Mallat, S. Generalization in diffusion models arises from geometry-adaptive harmonic representation. *arXiv preprint arXiv:2310.02557*, 2023.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35: 26565–26577, 2022.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Khrulkov, V., Ryzhakov, G., Chertkov, A., and Oseledets, I. Understanding ddpm latent codes through optimal transport. *arXiv preprint arXiv:2202.07477*, 2022.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

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

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

Li, G., Wei, Y., Chen, Y., and Chi, Y. Towards faster non-asymptotic convergence for diffusion-based generative models. *arXiv preprint arXiv:2306.09251*, 2023a.

Li, P., Li, Z., Zhang, H., and Bian, J. On the generalization properties of diffusion models. *arXiv preprint arXiv:2311.01797*, 2023b.

Li, X., Kwon, S. M., Alkhouri, I. R., Ravishanka, S., and Qu, Q. Decoupled data consistency with diffusion purification for image restoration. *arXiv preprint arXiv:2403.06054*, 2024.

Liu, G., Sun, H., Li, J., Yin, F., and Yang, Y. Accelerating diffusion models for inverse problems through shortcut sampling. *arXiv preprint arXiv:2305.16965*, 2023a.

Liu, G.-H., Vahdat, A., Huang, D.-A., Theodorou, E. A., Nie, W., and Anandkumar, A. I^2sb: Image-to-image schr\odinger bridge. *arXiv preprint arXiv:2302.05872*, 2023b.

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., and Zhu, J. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

Luo, C. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

Luo, G., Blumenthal, M., Heide, M., and Uecker, M. Bayesian mri reconstruction with joint uncertainty estimation using diffusion models. *Magnetic Resonance in Medicine*, 90(1):295–311, 2023a.

Luo, Z., Gustafsson, F. K., Zhao, Z., Sjölund, J., and Schön, T. B. Image restoration with mean-reverting stochastic differential equations. *arXiv preprint arXiv:2301.11699*, 2023b.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

Moon, T., Choi, M., Lee, G., Ha, J.-W., and Lee, J. Fine-tuning diffusion models with limited data. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022. URL https://openreview.net/forum?id=0J6afk9DqrR.

Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., and Chen, M. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022.

Pidstrigach, J. Score-based generative models detect manifolds. *Advances in Neural Information Processing Systems*, 35:35852–35865, 2022.

Pizzi, E., Roy, S. D., Ravindra, S. N., Goyal, P., and Douze, M. A self-supervised descriptor for image copy detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14532–14542, 2022.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022a.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. stable-diffusion. https://github.com/CompVis/stable-diffusion, 2022b.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241. Springer, 2015.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., and Aberman, K. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22500–22510, 2023.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., and Norouzi, M. Palette: Image-to-image diffusion models. In *ACM SIGGRAPH 2022 Conference Proceedings*, pp. 1–10, 2022a.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022b.

Salimans, T. and Ho, J. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35: 25278–25294, 2022.

Shi, Y., De Bortoli, V., Campbell, A., and Doucet, A. Diffusion schr\" odinger bridge matching. *arXiv preprint arXiv:2303.16852*, 2023.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Diffusion art or digital forgery? investigating data replication in diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6048–6058, 2023a.

Somepalli, G., Singla, V., Goldblum, M., Geiping, J., and Goldstein, T. Understanding and mitigating copying in diffusion models. *arXiv preprint arXiv:2305.20086*, 2023b.

Song, B., Kwon, S. M., Zhang, Z., Hu, X., Qu, Q., and Shen, L. Solving inverse problems with latent diffusion models via hard data consistency. *arXiv preprint arXiv:2307.08123*, 2023a.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Song, J., Vahdat, A., Mardani, M., and Kautz, J. Pseudoinverse-guided diffusion models for inverse problems. In *International Conference on Learning Representations*, 2022.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Song, Y., Shen, L., Xing, L., and Ermon, S. Solving inverse problems in medical imaging with score-based generative models. *arXiv preprint arXiv:2111.08005*, 2021.

Song, Y., Dhariwal, P., Chen, M., and Sutskever, I. Consistency models. 2023b.

Su, X., Song, J., Meng, C., and Ermon, S. Dual diffusion implicit bridges for image-to-image translation. *arXiv preprint arXiv:2203.08382*, 2022.

Tomczak, J. and Welling, M. Vae with a vampprior. In *International Conference on Artificial Intelligence and Statistics*, pp. 1214–1223. PMLR, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wang, B. and Vastola, J. J. The hidden linear structure in score-based models and its application. *arXiv preprint arXiv:2311.10892*, 2023.

Yang, Z., Wu, Z., Luo, M., Chiang, W.-L., Bhardwaj, R., Kwon, W., Zhuang, S., Luan, F. S., Mittal, G., Shenker, S., et al. {SkyPilot}: An intercloud broker for sky computing. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*, pp. 437–455, 2023.

Yi, M., Sun, J., and Li, Z. On the generalization of diffusion model. *arXiv preprint arXiv:2305.14712*, 2023.

Yingxi Yang, K. and Wibisono, A. Convergence of the inexact langevin algorithm and score-based generative models in kl divergence. *arXiv e-prints*, pp. arXiv–2211, 2022.

Yoon, T., Choi, J. Y., Kwon, S., and Ryu, E. K. Diffusion probabilistic models generalize when they fail to memorize. In *ICML 2023 Workshop on Structured Probabilistic Inference & Generative Modeling*, 2023.

Zhang, H., Lu, Y., Alkhouri, I., Ravishankar, S., Song, D., and Qu, Q. Improving efficiency of diffusion models via multi-stage framework and tailored multi-decoder architectures. *arXiv preprint arXiv:2312.09181*, 2023.

Zhang, H., Lu, Y., Alkhouri, I., Ravishankar, S., Song, D., and Qu, Q. Improving training efficiency of diffusion models via multi-stage framework and tailored multi-decoder architectures. In *Conference on Computer Vision and Pattern Recognition 2024*, 2024. URL https://openreview.net/forum?id=YtptmpZQOg.

Zhao, M., Bao, F., Li, C., and Zhu, J. Egsde: Unpaired image-to-image translation via energy-guided stochastic differential equations. *Advances in Neural Information Processing Systems*, 35:3609–3623, 2022.

Zhu, K., Wang, J., Zhou, J., Wang, Z., Chen, H., Wang, Y., Yang, L., Ye, W., Gong, N. Z., Zhang, Y., et al. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. *arXiv preprint arXiv:2306.04528*, 2023.

Zou, A., Wang, Z., Kolter, J. Z., and Fredrikson, M. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

Appendix

We include more comprehensive experiment settings, quantitative results, and detailed discussion of the unconditional diffusion model in Appendix A, Quantitative analysis of other generative models is in Appendix B, theoretical proof in Appendix C, experiment setting for Section 3.2.1 in Appendix D, experiment settings for conditional diffusion model in Appendix E, stable diffusion in Appendix F, diffusion model for solving inverse problems in Appendix G, fine-tuning diffusion model in Appendix H.

A. Unconditional Diffusion Model

Expanded experiment setting More detailed settings of the diffusion model we selected are listed in Table 1. With the exception of DiT and UViT, which we implemented and trained ourselves, all selected diffusion model architectures utilize the author-released models.

Architectural Relationships For DDPMv1, DDPMv2, and DDPMv7, we adopt the DDPM architecture initially proposed by (Ho et al., 2020), but we implement it using the codebase provided by (Song et al., 2020b). DDPMv3 and DDPMv8, on the other hand, employ DDPM++, an enhanced version of DDPM introduced by (Song et al., 2020b). DDPM++ incorporates BigGAN-style upsampling and downsampling techniques, following the work of (Brock et al., 2018). DDPMv4, DDPMv5, and DDPMv6 adopt DDPM++(deep), which shares similarities with DDPM++ but boasts a greater number of network parameters. Moving to Multistagev1, Multistagev2, and Multistagev3, these models derive from the Multistage architecture, a variant of the U-Net architecture found in DDPM++(deep). For EDMv1, EDMv2, CT, and CD, the EDM architecture is identical to DDPM++, but they differ in their training parameterizations compared to other DDPM++-based architectures. Finally, UViT and DiT are transformer-based architectures.

Distillation Relationships CD, Progressivev1, Progressivev2, and Progressivev3 are all diffusion models trained using distillation techniques. CD employs EDM as its teacher model, while Progressivev1, Progressivev2, and Progressivev3 share DDPMv3 as their teacher model. It's worth noting that these models employ a progressive distillation strategy, with slight variations in their respective teacher models, as elaborated in (Salimans & Ho, 2022).

Initial Noise Consistency However, it is important to note a nuanced difference related to the noise perturbation kernels. Specifically, for VP and subVP noise perturbation kernels, we define the noise space as $\mathcal{E} = \mathcal{N}(\mathbf{0}, \boldsymbol{I})$, whereas the VE noise perturbation kernel introduces a distinct noise space with $\mathcal{E} = \mathcal{N}(\mathbf{0}, \boldsymbol{\sigma}_{\text{max}}^2 \cdot I)$, where σ_{max} is predefined. So during the experiment, we sample 10K initial noise $\boldsymbol{\epsilon}_{\text{vp, subvp}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$ for the sample generation of diffusion models with VP and subVP noise perturbation kernel. For diffusion models with VE noise perturbation kernel, the initial noise is scaled as $\boldsymbol{\epsilon}_{\text{ve}} = \sigma_{\text{max}} \boldsymbol{\epsilon}_{\text{vp, subvp}}$.

Additionally, it's worth mentioning that for all 8x8 image grids shown in the Figure 1, 11, 14, 15, 16, 17, 22 no matter for the unconditional diffusion model, conditional diffusion model, diffusion model for the inverse problem, or fine-tuning diffusion model, we consistently employ the same 8x8 initial noise configuration. The same setting applies to 10k initial noises for reproducibility score. This specific design is for more consistent results between different variants of diffusion models (e.g., we could clearly find the relationship between the unconditional diffusion model and conditional diffusion model by comparing Figure 11 and Figure 16, 17).

Further discussion In Figure 11, we provide additional visualizations, offering a more comprehensive perspective on our findings. For a deeper understanding of our results, we present extensive quantitative data in Figure 10 and Figure 9. Building upon the conclusions drawn in Section 3.2.2, we delve into the consistency of model reproducibility across discrete and continuous timestep settings. To illustrate, we compare DDPMv1 and DDPMv2, demonstrating that model reproducibility remains steadfast across these variations.Moreover, it's worth noting that while all reproducibility scores

Table 1. **Comprehensive unconditional reproducibility experiment settings**

Name	Architecture	SDE	Sampler	Continuous	Distillation
DDPMv1	DDPM	VP	DPM-Solver	✓	✗
DDPMv2	DDPM	VP	DPM-Solver	✗	✗
DDPMv3	DDPM++	VP	DPM-Solver	✓	✗
DDPMv4	DDPM++(deep)	VP	DPM-Solver	✓	✗
DDPMv5	DDPM++(deep)	VP	ODE	✓	✗
DDPMv6	DDPM++(deep)	sub-VP	ODE	✓	✗
DDPMv7	DDPM	sub-VP	ODE	✓	✗
DDPMv8	DDPM++	sub-VP	ODE	✓	✗
Multistagev1	Multistage (3 stages)	VP	DPM-Solver	✓	✗
Multistagev2	Multistage (4 stages)	VP	DPM-Solver	✓	✗
Multistagev3	Multistage (5 stages)	VP	DPM-Solver	✓	✗
EDMv1	EDM	VP	Heun-Solver	✓	✗
EDMv2	EDM	VE	Heun-Solver	✓	✗
UViT	UViT	VP	DPM-Solver	✓	✗
DiT	DiT	VP	DPM-Solver	✓	✗
CD	EDM	VE	1-step	✓	✓
CT	EDM	VE	1-step	✓	✗
Progressivev1	DDPM++	VP	DDIM (1-step)	✓	✓
Progressivev2	DDPM++	VP	DDIM (16-step)	✓	✓
Progressivev3	DDPM++	VP	DDIM (64-step)	✓	✓

surpass a threshold of 0.6, signifying robust model reproducibility, some scores do exhibit variations. As highlighted in Figure 9, we observe that similar architectures yield higher reproducibility scores (e.g., DDPMv1-8), models distilled from analogous teacher models exhibit enhanced reproducibility (e.g., Progressivev1-3), and models differing solely in their ODE samplers also display elevated reproducibility scores (e.g., DDPMv4, DDPMv5).We hypothesize that the disparities in reproducibility scores are primarily attributed to biases in parameter estimation. These biases may arise from factors such as differences in architecture, optimization strategies, and other variables affecting model training.

B. Compare GAN & VAE

To further investigate this observation within the realm of diffusion models, we extend our assessment to model similarity in Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and Variational Autoencoders (VAEs) (Kingma & Welling, 2013). We gauge this similarity through the application of a reproducibility score. In our evaluation of GAN-based methods, we contrast two prominent variants: Wasserstein GAN (wGAN) (Arjovsky et al., 2017) and Spectral Normalization GAN (SNGAN) (Miyato et al., 2018). We conduct this analysis using the CIFAR-10 dataset. Simultaneously, within the realm of VAE-based approaches, we consider both the standard VAE and the Variational Autoencoding Mutual Information Bottleneck (VAMP) model (Tomczak & Welling, 2018). Our evaluation focuses on the MNIST dataset introduced by Deng (LeCun et al., 1998). It's important to note that each model utilized in this analysis was provided by its respective author, and the reproducibility score calculation follows a similar methodology to that applied in the diffusion model experiments. Of particular significance is the fact that the latent space for VAE-based methods is learned through the encoder, and this encoder architecture varies among different models. In this context, our approach involves sampling initial noise from the latent space of one model and employing it for the generation of another. The similarity matrices, presented in Figure 12, collectively indicate a notable absence of reproducibility in both GAN and VAE methods.

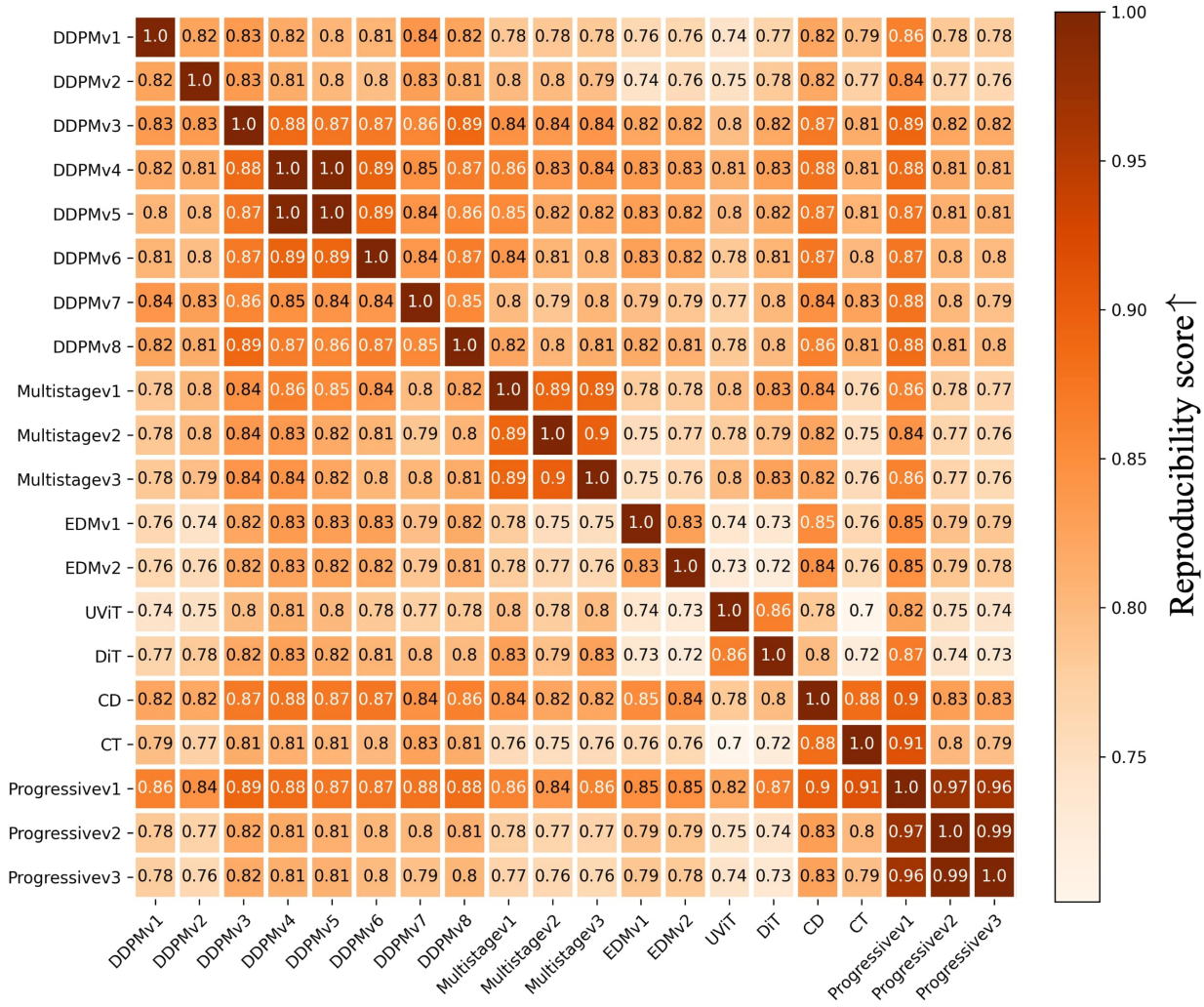


Figure 9. **Comprehensive reproducibility score among different unconditional diffusion model settings.**

C. Theoretical Analysis

This section mainly focuses on the proof of Proposition 3.2 in Section 3.1, the empirical score function would minimize the score matching loss function, Proposition 3.3 in Section 3.2.

As the background, let $p_t(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\mathbf{I})$ be the perturbation kernel of diffusion model, which is a continuous process gradually adding noise from original image $\boldsymbol{x}_0$ to $\boldsymbol{x}_t$ along the timestep $t \in [0, 1]$. Both $s_t = s(t), \sigma_t = \sigma(t)$ here are simplified as scalar functions of t to control the perturbation kernel. It has been shown that this perturbation kernel is equivalent to a stochastic differential equation $\mathrm{d}\boldsymbol{x} = f(t)\boldsymbol{x}\mathrm{d}t + g(t)\mathrm{d}\boldsymbol{\omega}_t$, where $f(t), g(t)$ are a scalar function of t. The relations of $f(t), g(t)$ and s_t, σ_t are:

$$s_t = \exp(\int_0^t f(\xi)\mathrm{d}\xi), \quad \text{and } \sigma_t = \sqrt{\int_0^t \frac{g^2(\xi)}{s^2(\xi)}\mathrm{d}\xi} \tag{2}$$

Proposition 3.2. Given a training dataset $\{\boldsymbol{y}_i\}_{i=1}^N$ of N-samples, consider the same setting of Lemma 3.1 with $p(\boldsymbol{x}_0)$ following the empirical multi-delta distribution $p(\boldsymbol{x}_0) = \frac{1}{N}\sum_{i=1}^N \delta(\boldsymbol{x}_0 - \boldsymbol{y}_i)$. In this setting, we can show that the score

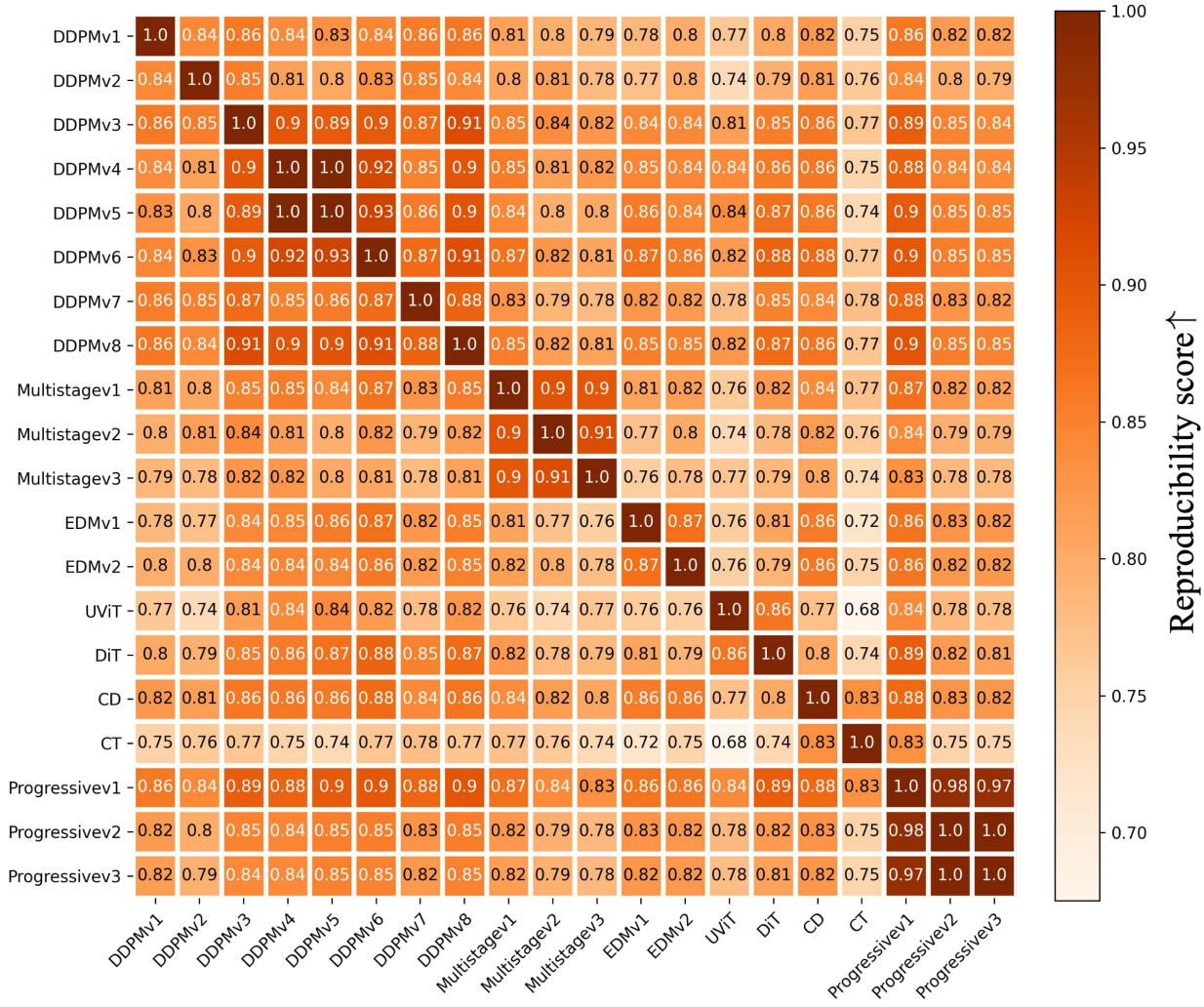


Figure 10. **Comprehensive MAE score among different unconditional diffusion model settings.**

function can be characterized as

$$\boldsymbol{s}_{\text{emp}}(\boldsymbol{x}_t;t) = -\frac{1}{s_t^2\sigma_t^2}\left[\boldsymbol{x}_t - s_t\frac{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})\boldsymbol{y}_i}{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})}\right]$$

Proof. we compute

$$p_t(\boldsymbol{x}) = \int p_t(\boldsymbol{x}|\boldsymbol{x}_0)p(\boldsymbol{x}_0)\mathrm{d}\boldsymbol{x}_0 = \frac{1}{N}\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I}).$$

Therefore, the score function is:

$$\begin{aligned}\boldsymbol{s}_{\text{emp}}(\boldsymbol{x}_t;t) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) = \frac{\nabla_{\boldsymbol{x}_t}p_t(\boldsymbol{x}_t)}{p_t(\boldsymbol{x}_t)} &= -\frac{1}{\beta_t^2}\frac{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})\,(\boldsymbol{x}_t - s_t\boldsymbol{y}_i)}{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})} \\ &= -\frac{1}{s_t^2\sigma_t^2}\left[\boldsymbol{x}_t - s_t\frac{\sum_{i=1}^N N(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})\boldsymbol{y}_i}{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})}\right]\end{aligned}$$

(a) DDPMv1 (b) DDPMv2 (c) DDPMv3

(d) DDPMv5 (e) DDPMv6 (f) DDPMv7

(g) DDPMv8 (h) EDMv1 (i) EDMv2

(j) DiT (k) CD (l) Progressivev1

(m) Progressivev2 (n) Progressivev3

Figure 11. **Comprehensive samples visulization for unconditional diffusion model**

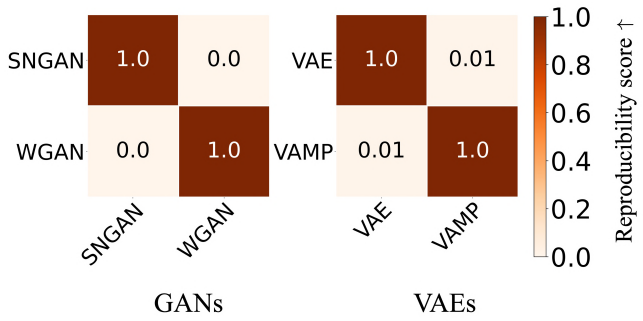


Figure 12. **Quantitative results for GANS and VAEs.**

a) Visualization of pretrained EDM on CIFAR10

b) Visualization of EDM trained on dataset sampled from a pretrained model.

Figure 13. Pretrained model and the model trained on the sampled dataset produce almost identical results.

From the relationship of predict $\boldsymbol{\epsilon}_{\text{emp}}$, predict $\boldsymbol{x}_{\text{emp}}$, and the score function:

$$\boldsymbol{\epsilon}_{\text{emp}}(\boldsymbol{x}_t, t) = -s_t\sigma_t\boldsymbol{s}(\boldsymbol{x}_t, t) = \frac{1}{s_t\sigma_t}\left[\boldsymbol{x}_t - s_t\frac{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})\boldsymbol{y}_i}{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})}\right]$$

$$\boldsymbol{x}_{\text{emp}}(\boldsymbol{x}_t, t) = \frac{\boldsymbol{x}_t - s_t\sigma_t\boldsymbol{\epsilon}_{\text{emp}}(\boldsymbol{x}_t, t)}{s_t} = \frac{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})\boldsymbol{y}_i}{\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})}$$

□

Then given the noise prediction loss $\mathcal{L}(\boldsymbol{\epsilon}_{\boldsymbol{\theta}}; t) = \mathbb{E}_{\boldsymbol{x}_t\sim p_t(\boldsymbol{x}_t)}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)||^2]$ with $p_t(\boldsymbol{x}_t) = \frac{1}{N}\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\boldsymbol{I})$, we will show that $\arg\min_{\boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t;t)}\mathcal{L}(\boldsymbol{\epsilon}_{\boldsymbol{\theta}}; \boldsymbol{x}_t, t) = \boldsymbol{\epsilon}_{\text{emp}}(\boldsymbol{x}_t, t)$.

Proof. The proof is inspired from (Karras et al., 2022). The loss could be calculated as:

$$\mathcal{L}(\boldsymbol{\epsilon}_{\boldsymbol{\theta}}; t) = \mathbb{E}_{\boldsymbol{x}_t\sim p_t(\boldsymbol{x}_t)}[||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)||^2] \tag{3}$$

$$= \int_{\mathbb{R}_d} \frac{1}{N}\sum_{i=1}^N \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})||\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta}}(\boldsymbol{x}_t, t)||^2 \mathrm{d}\boldsymbol{x}_t \tag{4}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is defined follow the perturbation kernel $p_t(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{x}_0, s_t^2\sigma_t^2\mathbf{I})$:

$$\boldsymbol{x}_t = s_t\boldsymbol{y}_i + s_t\sigma_t\boldsymbol{\epsilon} \Rightarrow \boldsymbol{\epsilon} = \frac{\boldsymbol{x}_t - s_t\boldsymbol{y}_i}{s_t\sigma_t} \tag{5}$$

And $\boldsymbol{\epsilon_\theta}$ is a "denoiser" network for learning the noise $\boldsymbol{\epsilon}$, under the assumption that the $\boldsymbol{\epsilon_\theta}$ has infinite model capacity, and can approximate any continuous function to an arbitrary level of accuracy based on the Universal Approximation Theorem. So plugging Eq. 5 into 4, we could reparameterization the loss as:

$$\mathcal{L}(\boldsymbol{\epsilon_\theta}; t) = \int_{\mathbb{R}_d} \underbrace{\frac{1}{N}\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})||\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t, t) - \frac{\boldsymbol{x}_t - s_t\boldsymbol{y}_i}{s_t\sigma_t}||^2}_{=:\mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}_t, t)} \mathrm{d}\boldsymbol{x}_t \tag{6}$$

Eq. 6 means we could minimize $\mathcal{L}(\boldsymbol{\epsilon_\theta}; t)$ by minimizing $\mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}_t, t)$ for each $\boldsymbol{x}_t$. And to find the "optimal denoiser" $\boldsymbol{\epsilon}^*_{\boldsymbol{\theta}}$ that minimize the $\mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}_t, t)$ for every given $\boldsymbol{x}_t, t$:

$$\boldsymbol{\epsilon}^*_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t) = \arg\min_{\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t;t)} \mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}_t, t) \tag{7}$$

Since $\boldsymbol{\epsilon_\theta}$ can approximate any continuous function to an arbitrary level of accuracy, this is a convex optimization problem; the solution could be solved by setting the gradient of $\mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}, t)$ w.r.t $\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t; t)$ to zero:

$$\nabla_{\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t;t)}[\mathcal{L}(\boldsymbol{\epsilon_\theta}; \boldsymbol{x}_t, t)] = 0 \tag{8}$$

$$\Rightarrow \nabla_{\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t;t)}[\frac{1}{N}\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})||\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t, t) - \frac{\boldsymbol{x}_t - s_t\boldsymbol{y}_i}{s_t\sigma_t}||^2] = 0 \tag{9}$$

$$\Rightarrow \frac{1}{N}\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})[\boldsymbol{\epsilon}^*_{\boldsymbol{\theta}}(\boldsymbol{x}; t) - \frac{\boldsymbol{x}_t - s_t\boldsymbol{y}_i}{s_t\sigma_t}] = 0 \tag{10}$$

$$\Rightarrow \boldsymbol{\epsilon}^*_{\boldsymbol{\theta}}(\boldsymbol{x}_t; t) = \frac{1}{s_t\sigma_t}[\boldsymbol{x}_t - s_t\frac{\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})\boldsymbol{y}_i}{\sum_{i=1}^{N}\mathcal{N}(\boldsymbol{x}_t; s_t\boldsymbol{y}_i, s_t^2\sigma_t^2\mathbf{I})}] \tag{11}$$

□

Proposition 3.3. Under the same setting of Lemma 3.1 with $p(\boldsymbol{x}_0)$ following the MoG distribution introduced in equation 1, we can show that the optimal score function is:

$$\boldsymbol{s}_{\mathrm{MoG}}(\boldsymbol{x}_t, t) = \sum_{i\in[C]} \frac{\pi_i(\boldsymbol{x}_t, t)}{s_t^2\sigma_t^2}\left(-\boldsymbol{x}_t + \frac{1}{1+\sigma_t^2}\boldsymbol{U}_i\boldsymbol{U}_i^\top\boldsymbol{x}_t\right),$$

with $\pi_i(\boldsymbol{x}_t, t) = \frac{\mathcal{N}\left(\boldsymbol{x}_t; \mathbf{0}, s_t^2\boldsymbol{U}_i\boldsymbol{U}_i^\top + s_t^2\sigma_t^2\boldsymbol{I}_d\right)}{\sum_{i\in[C]}\mathcal{N}\left(\boldsymbol{x}_t; \mathbf{0}, s_t^2\boldsymbol{U}_i\boldsymbol{U}_i^\top + s_t^2\sigma_t^2\boldsymbol{I}_d\right)}$.

Proof. First, let's consider the simplified case when $C = 1$:

$$p(\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_0; \mathbf{0}, \boldsymbol{U}^*\boldsymbol{U}^{*^T}\right)$$

Which is equivalent to:

$$\boldsymbol{x} = \boldsymbol{U}^*\boldsymbol{a}, \tag{12}$$

where $\boldsymbol{a} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_d)$. Then, we compute

$$\begin{aligned}
p_t(\boldsymbol{x}_t) &= \int p_t(\boldsymbol{x}_t|\boldsymbol{U}^*\boldsymbol{a})\mathcal{N}(\boldsymbol{a};\mathbf{0},\boldsymbol{I})d\boldsymbol{a} \\
&= \frac{1}{(2\pi)^{n/2}s_t^n\sigma_t^n}\int \frac{1}{(2\pi)^{d/2}}\exp\left(-\frac{1}{2s_t^2\sigma_t^2}\|\boldsymbol{x}_t - s_t\boldsymbol{U}^*\boldsymbol{a}\|^2\right)\exp\left(-\frac{\|\boldsymbol{a}\|^2}{2}\right)d\boldsymbol{a} \\
&= \frac{1}{(2\pi)^{n/2}s_t^n\sigma_t^n}\left(\frac{1+\sigma_t^2}{\sigma_t^2}\right)^{-d/2}\exp\left(-\frac{1}{2s_t^2\sigma_t^2}\boldsymbol{x}_t^T\left(\boldsymbol{I}_n - \frac{1}{1+\sigma_t^2}\boldsymbol{U}^*\boldsymbol{U}^{*^T}\right)\boldsymbol{x}_t\right) \\
&\quad \cdot \int \frac{1}{(2\pi)^{d/2}}\left(\frac{\sigma_t^2}{1+\sigma_t^2}\right)^{-d/2}\exp\left(-\frac{1+\sigma_t^2}{2\sigma_t^2}||\boldsymbol{a} - \frac{1}{s_t+s_t\sigma_t^2}U^{*T}\boldsymbol{x}_t||_2^2\right)d\boldsymbol{a} \\
&= \frac{1}{(2\pi)^{n/2}s_t^n\sigma_t^n}\left(\frac{1+\sigma_t^2}{\sigma_t^2}\right)^{-d/2}\exp\left(-\frac{1}{2s_t^2\sigma_t^2}\boldsymbol{x}_t^T\left(\boldsymbol{I}_n - \frac{1}{1+\sigma_t^2}\boldsymbol{U}^*\boldsymbol{U}^{*^T}\right)\boldsymbol{x}_t\right) \\
&= \frac{1}{(2\pi)^{n/2}\det\left(s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right)^{1/2}}\exp\left(-\frac{1}{2}\boldsymbol{x}_t^T\left(s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right)^{-1}\boldsymbol{x}_t\right) \\
&= \mathcal{N}\left(\boldsymbol{x}_t;\mathbf{0}, s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right).
\end{aligned}$$

Note that the fifth equality follows from

$$\begin{aligned}
\det\left(s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right) &= (s_t^2 + s_t^2\sigma_t^2)^d \cdot (s_t^2\sigma_t^2)^{n-d} \\
\left(s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right)^{-1} &= \frac{1}{s_t^2\sigma_t^2}\left(\boldsymbol{I}_n - \frac{\sigma_t^2}{1+\sigma_t^2}\boldsymbol{U}^*\boldsymbol{U}^{*^T}\right)
\end{aligned}$$

And the score function is:

$$\begin{aligned}
\boldsymbol{s}_{Gaussian}(\boldsymbol{x}_t, t) = \nabla_{\boldsymbol{x}_t}\log p_t(\boldsymbol{x}_t) = \frac{\nabla_{\boldsymbol{x}_t}p_t(\boldsymbol{x}_t)}{p_t(\boldsymbol{x}_t)} &= -\left(s_t^2\boldsymbol{U}^*\boldsymbol{U}^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}\right)^{-1}\boldsymbol{x}_t \\
&= -\frac{1}{s_t^2\sigma_t^2}\left(\boldsymbol{I}_d - \frac{1}{1+\sigma_t^2}\cdot\boldsymbol{U}^*\boldsymbol{U}^{*^T}\right)\boldsymbol{x}_t = -\frac{1}{s_t^2\sigma_t^2}\boldsymbol{x}_t + \frac{1}{s_t^2\sigma_t^2}\frac{1}{1+\sigma_t^2}\boldsymbol{U}^*\boldsymbol{U}^{*^T}\boldsymbol{x}_t.
\end{aligned}$$

Similarity, when the target distribution is Mixture of low rank gaussian:

$$p(\boldsymbol{x}_0) = \sum_{i\in[C]}\mathcal{N}\left(\boldsymbol{x}_0;\mathbf{0},\boldsymbol{U}_i^*\boldsymbol{U}_i^{*^T}\right)$$

Then:

$$\begin{aligned}
p_t(\boldsymbol{x}) &= \sum_{i\in[C]}\int p_t(\boldsymbol{x}|\boldsymbol{U}_i^*\boldsymbol{a})\mathcal{N}(\boldsymbol{a};\mathbf{0},\boldsymbol{I})d\boldsymbol{a} \\
&= \sum_{i\in[C]}\mathcal{N}\left(\boldsymbol{x};\mathbf{0},s_t^2\boldsymbol{U}_i^*\boldsymbol{U}_i^{*^T} + s_t^2\sigma_t^2\boldsymbol{I}_n\right).
\end{aligned}$$

And the score function is:

$$\begin{aligned}
\boldsymbol{s}(\boldsymbol{x}, t) &= \nabla_{\boldsymbol{x}}\log p_t(\boldsymbol{x}) \\
&= \frac{\nabla_{\boldsymbol{x}}p_t(\boldsymbol{x})}{p_t(\boldsymbol{x})} \\
&= \frac{\sum_i \pi_i\mathcal{N}\left(\boldsymbol{x}_0;\mathbf{0},\boldsymbol{U}_i^*\boldsymbol{U}_i^{*^T}\right)\left(-\frac{1}{s_t^2\sigma_t^2}\boldsymbol{x} + \frac{1}{s_t^2\sigma_t^2}\frac{1}{1+\sigma_t^2}\boldsymbol{U}_i^*\boldsymbol{U}_i^{*^T}\boldsymbol{x}\right)}{\sum_i \pi_i\mathcal{N}\left(\boldsymbol{x}_0;\mathbf{0},\boldsymbol{U}_i^*\boldsymbol{U}_i^{*^T}\right)}
\end{aligned}$$

□

Additional Experiment Setting for Figure 4 For a more comprehensive view of our results, we present additional visualizations in Figure 14 and Figure 15. In these experiments, we train UNet models with varying numbers of channels on subsets of the CIFAR-10 dataset, each comprising different training samples. Our standard batch size for all experiments is set at 128, and we continue training until the generated samples reach visual convergence, characterized by minimal changes in both appearance and semantic information.

D. Experiment setting for Section 3.2.1

D.1. Learning score functions of a mixture of Gaussian

For the mixture of Gaussian distribution, we set $C = 2, d = 48, r = 6$. We utilize the EDM diffusion model with embed dimension 128, training with 3000 iterations for all N. We generate totally 100k $(\boldsymbol{x}_t, t)$ pairs for estimate $\mathcal{L}_{\text{score}}$

D.2. Model Recovery of Diffusion Models

In order to show how diffusion models can be recovered, we train an EDM model on the dataset sampled from a pretrained model with same architecture. We use a well-trained diffusion model in the generalization regime, the mapping of which is denoted as $f_{\boldsymbol{\theta}_1}$, as an implicit representation of the distribution, denoted as $p_{DM}(\boldsymbol{x}_0) = f_{\boldsymbol{\theta}_1}(\boldsymbol{\epsilon}), \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, s_t^2\sigma_t^2\boldsymbol{I}_d)$. We sample N data points $\{\boldsymbol{y}_i\}_{i=1}^N \subseteq \mathbb{R}^n$ from $p_{DM}(\boldsymbol{x}_0)$, following the sampling process of the diffusion model to train another diffusion model, denoted as $f_{\boldsymbol{\theta}_2}$. We then calculate the reproducibility of the two models $f_{\boldsymbol{\theta}_1}, f_{\boldsymbol{\theta}_2}$ following the same practice as in section 2.1.

In detail, $f_{\boldsymbol{\theta}_2}$ is pretrained on CIFAR10 and $N = 50k$ which is the same as the size of CIFAR10 training set. We follow the same practive as in EDM(Karras et al., 2022). We use the DDPM++ model architecure and variance preserving(VP) formulation. We train the model until convergence.

As we can see in Figure 13, $f_{\boldsymbol{\theta}_1}$ and $f_{\boldsymbol{\theta}_2}$ almost generates identical results.

E. Conditional Diffusion Model

Extended Experiment setting To investigate the reproducibility of the conditional diffusion model, we opted for three distinct architectures: the conditional EDM (Karras et al., 2022), conditional multistage EDM (Zhang et al., 2023), and conditional U-ViT (Bao et al., 2023). Our training data consisted of the CIFAR-10 dataset, with the class labels serving as conditions. It's worth noting that the primary distinction between EDM and multistage EDM lies in the architecture of the score function. Conversely, the contrast between EDM and conditional U-ViT extends beyond architectural differences to encompass conditional embeddings. Specifically, EDM transforms class labels into one-hot vectors, subjects them to a single-layer Multilayer Perceptron (MLP), and integrates the output with timestep embeddings. In contrast, U-ViT handles class labels by embedding them through a trainable lookup table, concatenating them with other inputs, including timestep information and noisy image patches represented as tokens. For all three architectures, we pursued training until convergence was achieved, marked by the lowest FID. The DPM-Solver was employed for sampling purposes. To generate samples, we employed the same 10K initial noise distribution as utilized in the unconditional setting (refer to Section 3.2.2). For each such initial noise instance, we generated 10 images, guided by 10 distinct classes, resulting in a total of 100K images.

Discussion The observed reproducibility between the unconditional diffusion model and the conditional diffusion model presents an intriguing phenomenon. It appears that the conditional diffusion model learns a mapping function, denoted as $f_{c\in\mathcal{C}} : \mathcal{E} \mapsto \mathcal{I}_{c\in\mathcal{C}}$, which maps from the same noise space $\mathcal{E}$ to each individual image manifold $\mathcal{I}_{c\in\mathcal{C}}$ corresponding to each class c. In contrast, the mapping of the unconditional diffusion model, denoted as $f : \mathcal{E} \mapsto \mathcal{I}$, maps the noise space to a broader image manifold $\mathcal{I} \subset \bigcup_{c\in\mathcal{C}} \mathcal{I}_c$. A theoretical analysis of this unique reproducibility relationship holds the promise of providing valuable insights.

Currently, our research is exclusively focused on the conditional diffusion model. It raises the question of how the reproducibility phenomenon manifests in the context of the text-to-image diffusion model (Rombach et al., 2022a; Ramesh et al., 2021; Nichol et al., 2021), where the conditioning factor is not confined to finite classes but instead involves complex text embeddings.

As illustrated in Figure 16 and Figure 17, our previous comparisons were made with the same initial noise and class conditions. However, when comparing the same model with identical initial noise but different class conditions, we

uncovered intriguing findings. For instance, the first row and column images in Figure 16 (i) and (l) exhibited remarkable similarity in low-level structural attributes, such as color, despite differing in semantics. This observation is consistent with findings in Figure 22, where we explored generation using diffusion models trained on mutually exclusive CIFAR-100 and CIFAR-10 datasets. These findings bear a striking resemblance to the conclusions drawn in (Khrulkov et al., 2022), which also demonstrated a similar phenomenon in a simplified scenario, where $\mathcal{I}$ follows a Gaussian distribution. To gain a deeper understanding of reproducibility and the phenomena mentioned in this paragraph, leveraging optimal transport methods (e.g., Schrödinger bridge (Shi et al., 2023; De Bortoli et al., 2021; Luo et al., 2023b; Delbracio & Milanfar, 2023; Liu et al., 2023b)) holds significant potential.

F. Stable Diffusion

Our study also explores the reproducibility of the text-to-image diffusion model, Stable Diffusion (Rombach et al., 2022a), trained on the LAION-5B dataset (Schuhmann et al., 2022). We utilize the series of pre-trained Stable Diffusion models (versions v1-1 to v1-4) released by (Rombach et al., 2022b). These models exhibit key differences:

- Versions v1-1, v1-2, and v1-3 each are trained on different subsets of the LAION-5B dataset.
- Versions v1-3 and v1-4 share the same training subset from LAION-5B.
- Version v1-2 is resumed from v1-1, while v1-3 and v1-4 are resumed from v1-2.

Further details on their training settings are available at (Rombach et al., 2022b).

For reproducibility assessment, we use the prompt "a photograph of an astronaut riding a horse" along with 1,000 randomly generated initial noises. The reproducibility score is determined with SSCD metric larger than 0.4. To isolate the impact of the guiding prompt on reproducibility, we also evaluate the reproducibility score with the same prompt but different initial noises.

The results, shown in Figure 18a, reveal the highest reproducibility score between v1-3 and v1-4 (0.63), likely due to their same training datasets. Lesser but noticeable reproducibility scores (below 0.21) are observed among v1-1, v1-2, and v1-3, which might be attributable to their sequential training and overlapping datasets. This finding aligns with (Kadkhodaie et al., 2023), suggesting that training on exclusive subsets of the same dataset can yield reproducible results in diffusion models. A notable observation in Figure 18c is the presence of flip generations between v1-3 and v1-4, potentially a result of data augmentation introducing randomness. We hypothesize that excluding data augmentation could further increase the reproducibility score between v1-3 and v1-4. Furthermore, when varying the initial noise but with the same prompt, the reproducibility scores approach zero, as evidenced in Figure 18b, indicating only the same prompt but different initial noise will not have reproducibility.

G. Diffusion Model for Solving Inverse Problem

To explore the reproducibility of diffusion models in solving inverse problems, we adopted the Diffusion Posterior Sampling (DPS) strategy proposed by Chung et al. (Chung et al., 2022a). Our adaptation involved a slight modification of their algorithm, specifically by eliminating all sources of stochasticity within it. Additionally, we employed the DPM-Solver for Diffusion Posterior Sampling.

Extended Experiment setting To explore the reproducibility of diffusion models in solving inverse problems, we adopted the Diffusion Posterior Sampling (DPS) strategy proposed by Chung et al. (Chung et al., 2022a). Our adaptation involved a slight modification of their algorithm, specifically by eliminating all sources of stochasticity within it. Additionally, we employed the DPM-Solver for Diffusion Posterior Sampling: Algorithm 1, with $N_{\text{dps}} = 34$ posterior samping steps, 33 iterations for 3rd order DPM-Solver, 1 for 1st order DPM-Solver, thus 100 function evaluations. We also set all $\xi_i = 1$.

For the task involving image inpainting on the CIFAR-10 dataset, we applied two square masks to the center of the images. One mask measured 16 by 16 pixels, covering 25% of the image area, and the other measured 25 by 25 pixels, covering 61% of the image area. We denoted these as "easy inpainting" and "hard inpainting" tasks. In Figure 8 and Figure 19, we utilized the "easy inpainting" scenario with a specific observation $\boldsymbol{z}$ as illustrated in the figure. In Figure 20, we considered both the "easy inpainting" and "hard inpainting" tasks. We also employed 10K distinct initial noise and their corresponding 10K distinct observations $\boldsymbol{z}$ to calculate the reproducibility score, as presented in Figure 20.

Algorithm 1 Determinsitic DPS with DPM-Solver.

Input: N_{dps}, $\boldsymbol{u}$,$f(t)$,$g(t)$, s_t, σ_t, $\{\xi_i\}_{i=1}^{N_{\text{dps}}}$
$\boldsymbol{x}_{N_{\text{dps}}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$
for $i = N_{\text{dps}}$ **to** q **do**
  $\hat{\boldsymbol{x}}_0 = \frac{1}{f(i)}\left(\boldsymbol{x}_i - \frac{g^2(i)}{s_i\sigma_i}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i, i\right)\right)$
  $\boldsymbol{x}'_{i-1} \leftarrow$ Dpm-Solver$(\boldsymbol{x}_i, i)$
  $\boldsymbol{x}_{i-1} \leftarrow \boldsymbol{x}'_{i-1} - \xi_i \nabla_{\boldsymbol{x}_i} ||\boldsymbol{u} - \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)||_2^2$
end for
return $\hat{\mathbf{x}}_0$

Discussion Reproducibility is a highly desirable property when employing diffusion models to address inverse problems, particularly in contexts such as medical imaging where it ensures the reliability of generated results. As observed in Figure 19, the reproducibility scores vary for different observations $\boldsymbol{z}$, and the decrease in reproducibility differs across various architecture categories. For instance, when considering observation $\boldsymbol{z}_1$, the reproducibility scores across different architecture categories remain above 0.5, whereas for $\boldsymbol{z}_3$, they fall below 0.3. Since the choice of observation $\boldsymbol{z}$ also significantly impacts reproducibility, we conducted a complementary experiment presented in Figure 20. In this experiment, for each initial noise instance, we employed a different observation $\boldsymbol{z}$. From the results, it is evident that reproducibility decreases between different categories of diffusion models. Furthermore, reproducibility diminishes as the inpainting task becomes more challenging, with "hard inpainting" being more demanding than "easy inpainting."

Here is an intuitive hypothesis of the decreasing reproducibility:

The update step of Diffusion Posterior Sampling (DPS), is constrained by the data consistency through the following equation:

$$\boldsymbol{x}_{i-1} \leftarrow \boldsymbol{x}'_{i-1} - \xi_i \nabla_{\boldsymbol{x}_i} ||\boldsymbol{u} - \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)||_2^2 \tag{13}$$

Where $\hat{\boldsymbol{x}}_0 = \frac{1}{f(i)}\left(\boldsymbol{x}_i - \frac{g^2(i)}{s_i\sigma_i}\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i, i\right)\right)$, we could show that:

$$\xi_i \nabla_{\boldsymbol{x}_i} ||\boldsymbol{z} - \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)||_2^2 = \frac{\partial \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)}{\partial \boldsymbol{x}_i}\left(\mathcal{A}\left(\hat{\boldsymbol{x}}_0\right) - \boldsymbol{z}\right) \tag{14}$$

$$= \frac{\partial \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)}{\partial \hat{\boldsymbol{x}}_0}\frac{\partial \hat{\boldsymbol{x}}_0}{\partial \boldsymbol{x}_i}\left(\mathcal{A}\left(\hat{\boldsymbol{x}}_0\right) - \boldsymbol{z}\right) \tag{15}$$

$$= \frac{1}{f(i)}\frac{\partial \mathcal{A}\left(\hat{\boldsymbol{x}}_0\right)}{\partial \hat{\boldsymbol{x}}_0}\left(1 - \frac{g^2(i)}{s_i\sigma_i}\frac{\partial \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i, i\right)}{\partial x_i}\right)\left(\mathcal{A}\left(\hat{\boldsymbol{x}}_0\right) - \boldsymbol{z}\right) \tag{16}$$

This analysis highlights that the unconditional diffusion model is reproducible as long as the function $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}$ is reproducible. However, for the diffusion model used in inverse problems to be reproducible, both the function $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t, t\right)$ and its first-order derivative with respect to $\boldsymbol{x}_t$ must be reproducible. In other words, the denoiser should exhibit reproducibility not only in its results but also in its gradients. Combining the findings in Figure 20, we can infer that for similar architectures, reproducibility also extends to the gradient space $\frac{\partial \boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t, t\right)}{\partial \boldsymbol{x}_t}$, which may not hold true for dissimilar architectures. Ensuring reproducibility in the gradient space should thus be a significant focus for achieving reproducibility in diffusion models for solving inverse problems.

Additionally, it's worth noting that the data $\boldsymbol{x}_t$ passed into the denoiser $\boldsymbol{\epsilon}_{\boldsymbol{\theta}}\left(\boldsymbol{x}_t, t\right)$ is always out-of-distribution (OOD) data, especially in tasks like image inpainting. Consequently, the reproducibility of OOD data $\boldsymbol{x}_t$ is also crucial for achieving reproducibility in diffusion models for solving inverse problems.

H. Fine-tuning Diffusion Model

Few-shot image fine-tuning for diffusion models, as discussed in (Ruiz et al., 2023; Gal et al., 2022; Moon et al., 2022; Han et al., 2023), showcases remarkable generalizability. This is often achieved by fine-tuning a small portion of the parameters of a large-scale pre-trained (text-to-image) diffusion model. In this study, we delve into the impacts of partial model fine-tuning on both model reproducibility and generalizability, by extending our analysis in Section 2. We show that *Partial fine-tuning reduces reproducibility but improves generalizability in "memorization regime"*.

Experiment setting In our investigation of reproducibility during fine-tuning, we first trained an unconditional diffusion model using EDM (Karras et al., 2022) on the CIFAR-100 dataset (Krizhevsky et al., 2009). All the fine-tuned models discussed in this section were pre-trained on this model. Subsequently, we examined the impact of dataset size by conducting fine-tuning on the EDM using varying numbers of CIFAR-10 images: 64, 1024, 4096, 16384, and 50000, respectively. Building upon the findings in (Moon et al., 2022), which indicate that fine-tuning the attention blocks is less susceptible to overfitting, we opted to target all attention layers for fine-tuning in our experiments. For comparison purposes, we also trained a diffusion model from scratch on the CIFAR-10 dataset, using the same subset of images. All models were trained for the same number of training iterations and were ensured to reach convergence, as evidenced by achieving a low Fréchet Inception Distance (FID) and maintaining consistent mappings from generated samples. The training utilized a batch size of 128 and did not involve any data augmentation.

Our claim is supported by our results in Figure 21, comparing model fine-tuning and training from scratch of with varying size of the training data, where both models have the same number of parameters. In comparison to training from scratch that we studied in Figure 2b, fine-tuning specific components of pre-trained diffusion models, particularly the attention layer in the U-Net architecture, yields lower model reproducibility score but higher generalization score in the memorization regime. However, in the generalization regime, partial model fine-tuning has a minor impact on both reproducibility and generalization in the diffusion model. Our result reconfirms the improved generalizability of fine-tuning diffusion models on limited data, but shows a surprising tradeoff in terms of model reproducibility that is worth further investigation.

Additional generations produced by both the "from scratch" diffusion models and the fine-tuned diffusion models are presented in Figure 22, encompassing various training dataset sizes. A notable observation arises when comparing the fine-tuned diffusion model's generation using 4096 and 50000 data samples. Even with this limited dataset, the fine-tuned diffusion model demonstrates a remarkable ability to approximate the target distribution. This suggests that the fixed portion of the diffusion model, containing information from the pre-trained CIFAR-100 dataset, aids the model in converging to the target distribution with less training data. In contrast, when attempting to train the diffusion model from scratch on CIFAR-10, even with 16384 data samples, it fails to converge to the target distribution. Additionally, despite the distinct nature of CIFAR-100 and CIFAR-10, their generations from the same initial noise exhibit striking similarities (Figure 22). This similarity might be a contributing factor explaining how the pre-trained CIFAR-100 diffusion model assists in fine-tuning the diffusion model to converge onto the CIFAR-10 manifold with reduced training data.

Figure 14. **Visualization between theoretical and experimental results.**

Figure 15. **Visualization between theoretical and experimental results.**

(a) EDM Class0 (b) Multistage EDM Class0 (c) U-ViT Class0

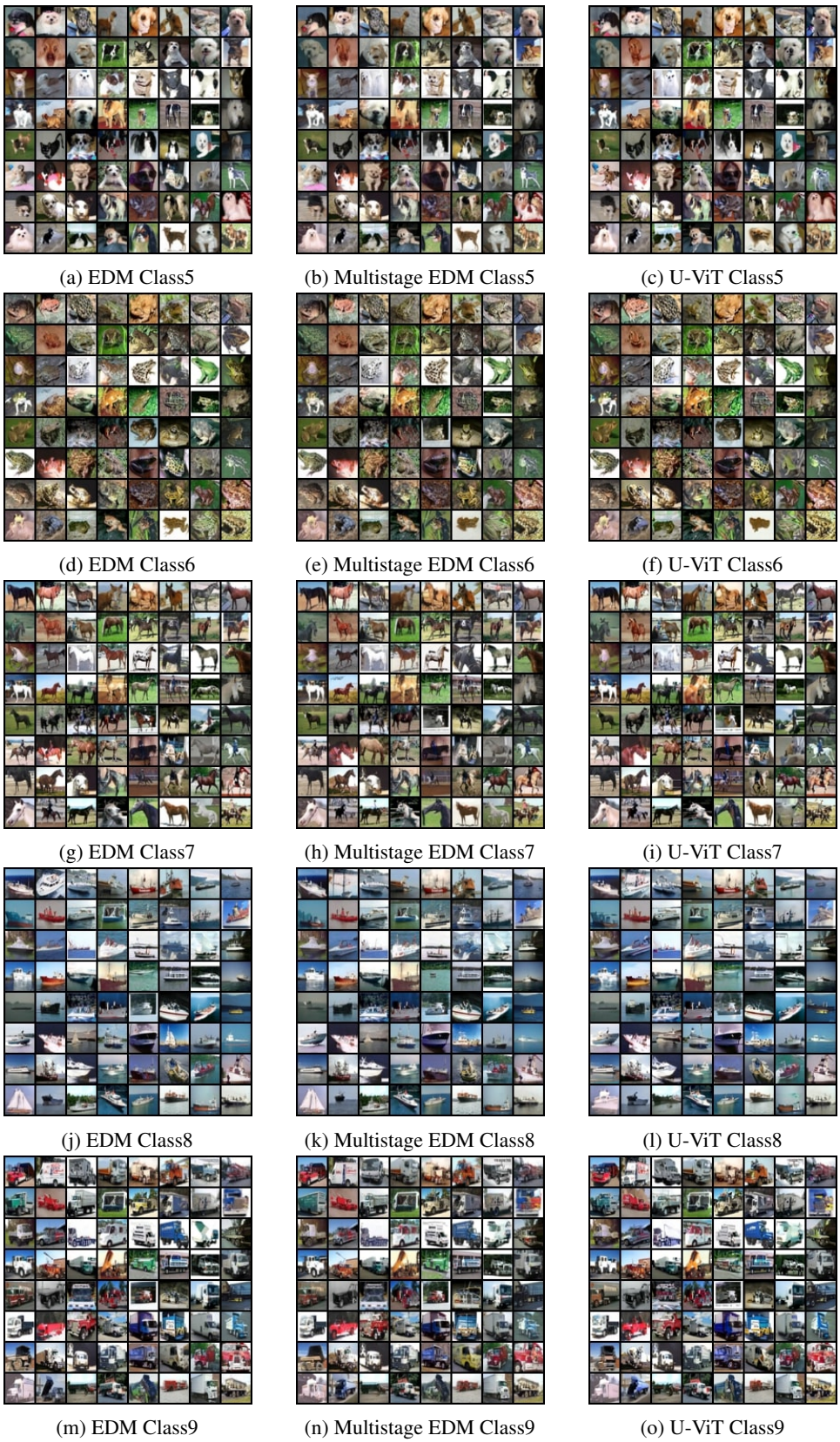
(d) EDM Class1 (e) Multistage EDM Class1 (f) U-ViT Class1

(g) EDM Class2 (h) Multistage EDM Class2 (i) U-ViT Class2

(j) EDM Class3 (k) Multistage EDM Class3 (l) U-ViT Class3

(m) EDM Class4 (n) Multistage EDM Class4 (o) U-ViT Class4

Figure 16. **Visualization of conditional diffusion model generations (class 0 - 4).**



(a) EDM Class5 (b) Multistage EDM Class5 (c) U-ViT Class5

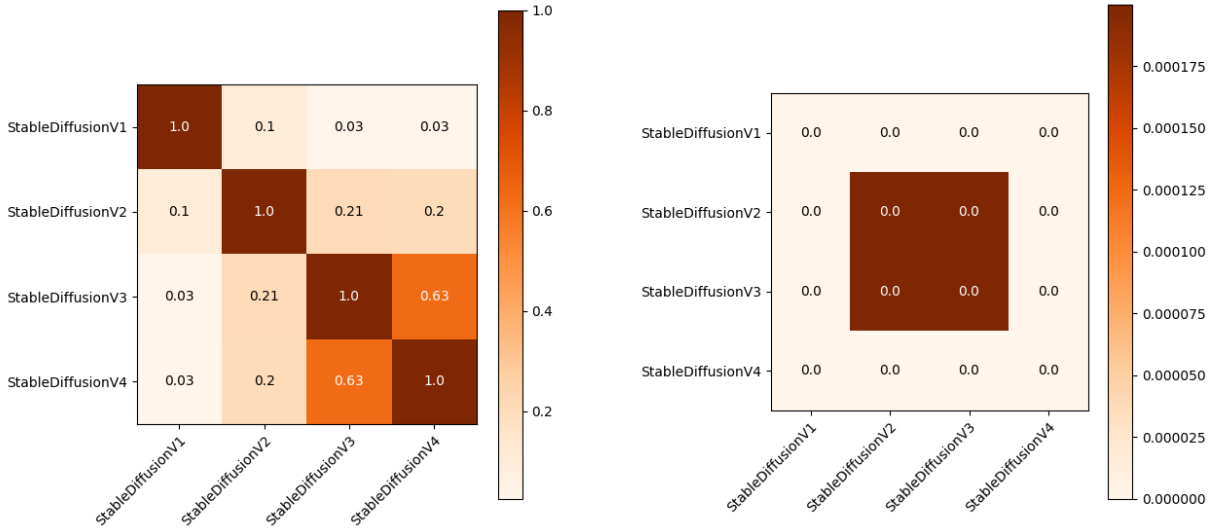
(d) EDM Class6 (e) Multistage EDM Class6 (f) U-ViT Class6

(g) EDM Class7 (h) Multistage EDM Class7 (i) U-ViT Class7

(j) EDM Class8 (k) Multistage EDM Class8 (l) U-ViT Class8

(m) EDM Class9 (n) Multistage EDM Class9 (o) U-ViT Class9

Figure 17. **Visualization of conditional diffusion model generations (class 5 - 9).**

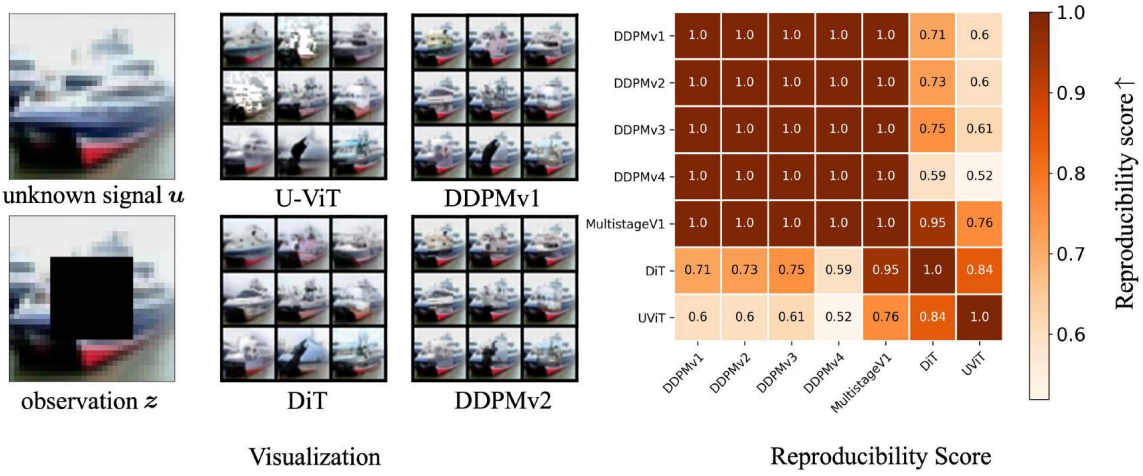


(a) Reproducibility score for same initial noise

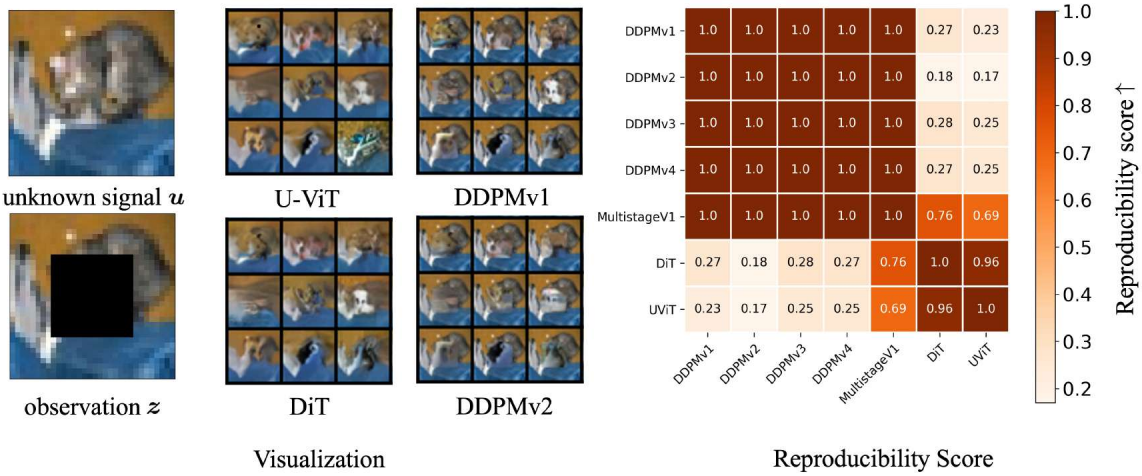
(b) Reproducibility score for different initial noise

(c) Visualization of stable diffusion.

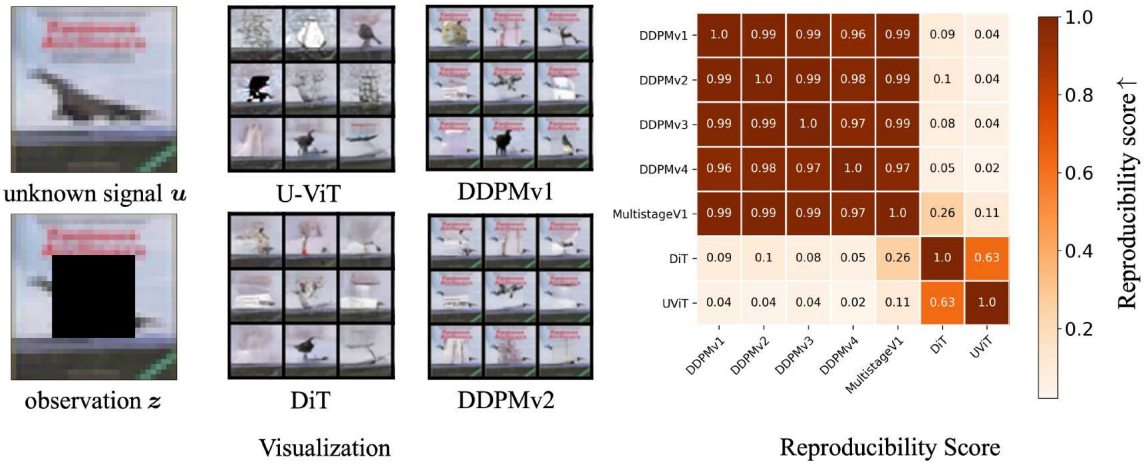
Figure 18. **Reproducibility of Stable Diffusion.**



(a) observation z_1



(b) observation z_2



(c) observation z_3

Figure 19. **Visualization of inverse problem solving with different observations**

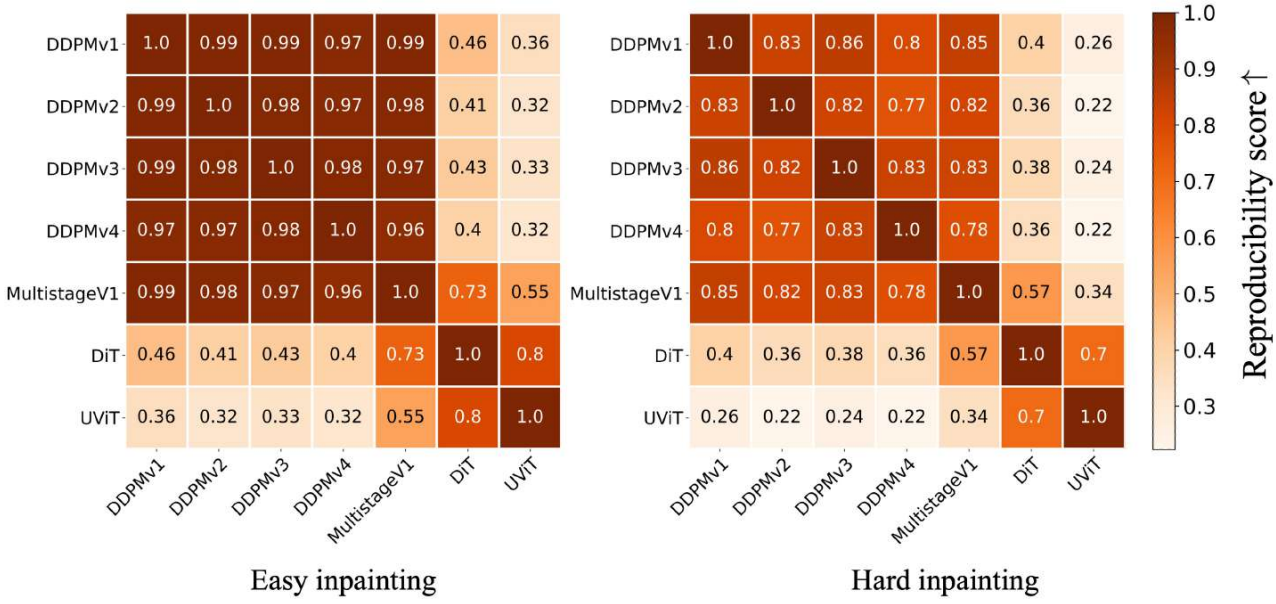
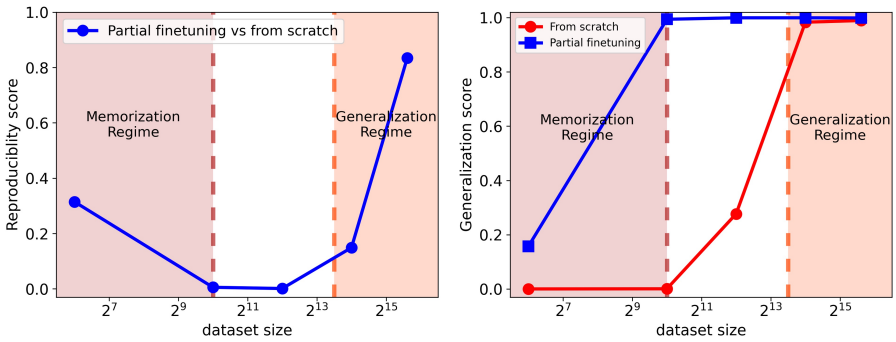


Figure 20. **Extended experiments on image impainting for reproducibility score.**



(a) Reproducibilty (b) Generalizability

Figure 21. **Model reproducibility for diffusion model finetuing.** In this experiment, we employ DDPMv4. Two distinct training strategies are investigated: "from scratch," denoting direct training on a subset of the CIFAR-10 dataset, and "partial fine-tuning," which involves pretraining on the entire CIFAR-100 dataset (Krizhevsky et al., 2009) followed by fine-tuning only the attention layers of the model on a subset of the CIFAR-10 dataset. The dataset sizes for CIFAR-10 range from 2^6 to 2^{15}. Importantly, both "from scratch" and "partial fine-tuning" are trained using the same subset of images for each dataset size. Under different dataset sieze, Figure (a) illustrates the reproducibility score between these two strategies and (b) presents the generalization score for them.

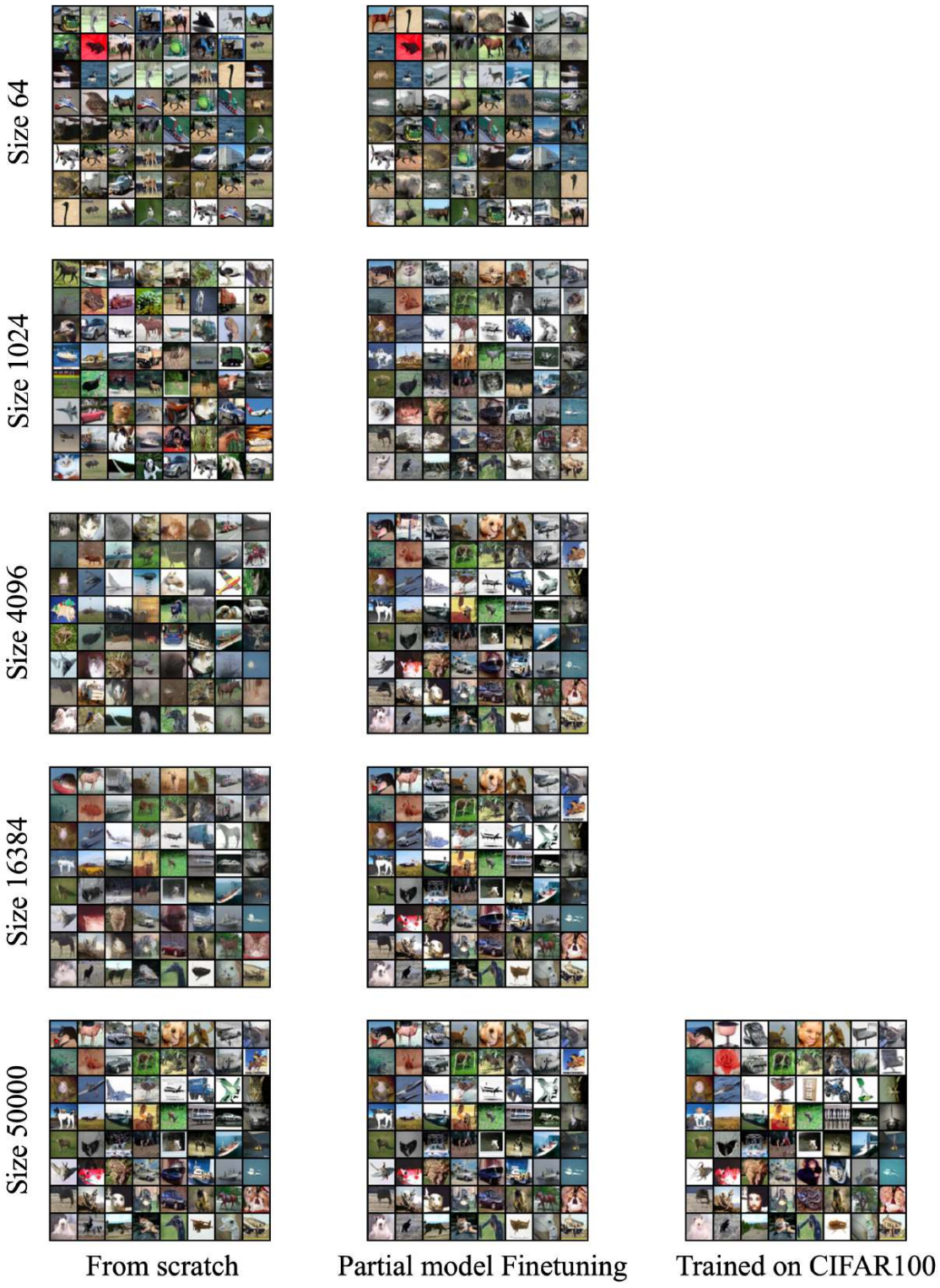


Figure 22. **More visualization of finetuning diffusion models**