Studying Classifier(-Free) Guidance from a Classifier-Centric Perspective

Xiaoming Zhao, Alex Schwing

University of Illinois Urbana-Champaign
{xz23, aschwing}@illinois.edu

Abstract

Classifier-free guidance has become a staple for conditional generation with denoising diffusion models. However, a comprehensive understanding of classifier-free guidance is still missing. In this work, we carry out an empirical study to provide a fresh perspective on classifier-free guidance. Concretely, instead of solely focusing on classifier-free guidance, we trace back to the root, *i.e.*, classifier guidance, pinpoint the key assumption for the derivation, and conduct a systematic study to understand the role of the classifier. On 1D data, we find that both classifier guidance and classifier-free guidance achieve conditional generation by pushing the denoising diffusion trajectories away from decision boundaries, *i.e.*, areas where conditional information is usually entangled and is hard to learn. To validate this classifier-centric perspective on high-dimensional data, we assess whether a flow-matching postprocessing step that is designed to narrow the gap between a pre-trained diffusion model's learned distribution and the real data distribution, especially near decision boundaries, can improve the performance. Experiments on various datasets verify our classifier-centric understanding.

1 Introduction

Conditional generation, *e.g.*, class-to-image, text-to-image, or image-to-video, is omnipresent as it provides a compelling way to control the output. Ideally, conditional generation results are both *diverse* and of *high-fidelity*. Namely, the generative models' outputs align with the conditioning information perfectly and diligently follow the training data diversity. However, there is a trade-off between high-fidelity and diversity: without constraining diversity there are always possibilities to sample from areas on the data distribution manifold that are not well-trained. Thus, trading diversity for fidelity is a long-standing problem and the community has developed various approaches, *e.g.*, the truncation trick for generative adversarial nets (GANs) (Brock, Donahue, and Simonyan 2018; Karras, Laine, and Aila 2018), low-temperature sampling for probabilistic models (Ackley, Hinton, and Sejnowski 1985), or temperature control in large language models (Achiam et al. 2023; Dubey et al. 2024).

More recently, to trade diversity and fidelity in denoising diffusion models (Song and Ermon 2019; Vincent 2011; Ho, Jain, and Abbeel 2020; Kingma et al. 2021), several techniques have been developed (Dhariwal and Nichol 2021; Hong et al. 2023; Kim et al. 2022; Dinh, Liu, and Xu 2023b,a), from which classifier-free guidance (Ho and Salimans 2021) emerged as the de-facto standard. *E.g.*, classifier-free guidance, especially at sufficient scale, is critical for high-quality text-to-image (Rombach et al. 2022a) and text-to-3D (Poole et al. 2023) generation.

Despite its popularity, we think a solid understanding of classifier-free guidance is missing. Recently, several efforts provide insights by studying classifier-free guidance from a theoretical perspective (Bradley and Nakkiran 2024; Xia et al. 2024; Chidambaram et al. 2024) showing that sampling from classifier-free guidance is not the same as sampling from a sharpened distribution.

Instead of solely focusing on classifier-free guidance as done in the works mentioned above, we trace back to the root of classifier-free guidance, *i.e.*, classifier guidance (Dhariwal and Nichol 2021). It is classifier guidance that decomposes the *conditional* generation into a combination of an *unconditional* generation and a classifier prediction. Classifier-free guidance adopts this decomposition and replaces the classifier by randomly dropping conditioning during training (Ho and Salimans 2021). This connection motivates us to carefully study classifier guidance's derivation and its behavior.

We first identify a key assumption underlying classifier guidance's decomposition, which is also central to classifier-free guidance due to the connection mentioned above, that often fails to hold. This issue results in different behaviors for 1) a vanilla denoising diffusion conditional generation; and 2) a generation that follows the decomposition of classifier guidance as well as classifier-free guidance. On synthetic 1D data, the vanilla conditional generative model produces straight denoising paths while the decomposed version results in distorted trajectories that are *pushed away from the classifier's decision boundary*. This discrepancy is exacerbated with the commonly used large guidance scale.

The above observation motivates us to further study the sensitivity of classifier guidance to the accuracy of the *classifier*. We find that classifier guidance generations are dominated by the behavior of the classifier that provides guidance. In other words, conditional generation via classifier guidance is achieved via pushing the generation away from the class decision boundaries. A similar observation is ob-

tained for classifier-free guidance as well. To further verify this classifier-centric perspective, we study a postprocessing step to push samples from the trained model, *mainly around the decision boundaries*, to their nearest neighbors in the real data. Experiments on various datasets demonstrate the improvement of generation, verifying our understanding.

In summary, our contribution is a systematic empirical study of both classifier and classifier-free guidance from a classifier-centric perspective for intuitive understandings.

2 Related Works

Trading diversity for fidelity in conditional generation is a long-standing problem that has been actively studied by the community. For probabilistic models trained with the maximum likelihood objective, Ackley, Hinton, and Sejnowski (1985) propose low-temperature sampling to effectively focus on the mode of the learned distribution, borrowing ideas from statistical mechanics (Metropolis et al. 1953). This technique has also been employed beneficially for high-quality image synthesis (Parmar et al. 2018; Kingma and Dhariwal 2018). Recent large language models (LLMs) (Achiam et al. 2023; Dubey et al. 2024) also exploit this idea, balancing creativity and determinism via temperature control during next token prediction via the learned probability model (Brown et al. 2020). For image synthesis with generative adversarial nets (GANs), the truncation trick (Brock, Donahue, and Simonyan 2018; Karras, Laine, and Aila 2018) was developed to enforce sampling from a truncated normal distribution rather than the standard normal prior. This encourages conditional generations to remain close to the mode of the data distribution observed during training, preventing them from diverging too far. More recently, denoising diffusion models have demonstrated impressive generation capabilities in various domains (Kong et al. 2021; Poole et al. 2023; Rombach et al. 2022a; Chen et al. 2021). Classifier-free guidance (Ho and Salimans 2021), built upon classifier guidance (Dhariwal and Nichol 2021), has emerged as a standard for controlling conditional generations of denoising diffusion models. Our work contributes to the understanding of the trade-off between diversity and fidelity in the field of denoising diffusion models via carefully studying classifier and classifier-free guidance from a classifier-centric perspective.

Generation with guidance is closely related to our study. Techniques discussed in the preceding paragraph, except classifier guidance, solely require trained generative models, *e.g.*, the generator in GANs, to control the diversity and fidelity trade-off. In contrast, guidance relies on a separate model to influence the conditional generation. Rejection sampling (Casella, Robert, and Wells 2004) is an active area of research in this direction. For GANs, prior works use the discriminator paired with the generator to reject generations for which the discriminator has high confidence (Azadi et al. 2019; Turner et al. 2018). Alternatively, Che et al. (2020) utilize the discriminator to reject samples in the latent space. For variational autoencoders (VAEs) (Kingma and Welling 2014), learnable acceptance functions have been studied for both prior (Bauer and Mnih 2019; Aneja et al. 2021) and posterior (Grover et al. 2018; Jankowiak and Phan 2023) rejection sampling. Other works explore pre-trained classifiers to provide guidance. Razavi, van den Oord, and Vinyals (2019) use a classifier trained on ImageNet (Deng et al. 2009) to reject samples that cannot be well-recognized. Thanks to the recent progress of representation learning, several prior works exploit CLIP (Radford et al. 2021) to provide guidance on conditional generations with GANs (Galatolo, Cimino, and Vaglini 2021; Patashnik et al. 2021). Kim et al. (2022) improves the quality of a pre-trained model via refining denoising trajectories with guidance from a discriminator that distinguishes between real and fake denoising paths. Dinh, Liu, and Xu (2023a,b) mitigates the conflicts between quality and diversity caused by the guidance from a gradient and progressive perspective. Classifier guidance (Dhariwal and Nichol 2021) steers generation using a classifier trained along the denoising path. Inspired by this, we take a classifier-centric view to intuitively understand the behaviors of classifier guidance and classifier-free guidance.

Classifier-free guidance has attracted more and more attention in the community. Theoretically, several recent works clarify that sampling with classifier-free guidance does not correspond to sampling from a tilted distribution (Bradley and Nakkiran 2024; Xia et al. 2024; Chidambaram et al. 2024; Wu et al. 2024), a misconception that is popular in the community. Bradley and Nakkiran (2024) further prove that CFG is equivalent to the predictor-corrector mechanism (Song et al. 2021) in the continuous-time limit. Empirically, prior works improve generation quality by refining classifier-free guidance. Sadat et al. (2024) dynamically adjust the scale of classifier-free guidance to improve the generation diversity. Lin and Yang (2024) argue that classifier-free guidance essentially behaves as a perceptual loss and propose to incorporate a self-perceptual objective during training. Chung et al. (2024) introduce CFG++ to mitigate the issue of an off-manifold denoising path via a refined sampling formulation and a small scale. Different from these works that solely focus on classifier-free guidance, we instead trace back to the origin, *i.e.*, classifier guidance (Dhariwal and Nichol 2021). We systematically study the role the classifier plays in the performance of classifier guidance and found classifier guidance essentially pushes the generation away from the decision boundary. Furthermore, we demonstrate that this is also true for classifier-free guidance.

3 Analysis

3.1 Denoising Diffusion Conditional Generation

The goal of conditional generation is to sample the data of interest $\mathbf{x}_0$, *e.g.*, images, from a conditional distribution, *i.e.*, $\mathbf{x}_0 \sim p_\theta(\mathbf{x}_0|c)$. Here, c is the conditioning information, *e.g.*, class labels. Note, hereafter we use θ to subsume all learnable parameters for simplicity.

In this work, we focus on denoising diffusion models (Song and Ermon 2019; Vincent 2011; Ho, Jain, and Abbeel 2020). A denoising diffusion process generates data from white noise by introducing a sequence of latent variables $\mathbf{x}_{1:T} \triangleq \{\mathbf{x}_1, \ldots, \mathbf{x}_T\}$ that form a Markov chain $p_\theta(\mathbf{x}_0|c) = \int p_\theta(\mathbf{x}_0, \mathbf{x}_{1:T}|c)\, d\mathbf{x}_{1:T} \triangleq$

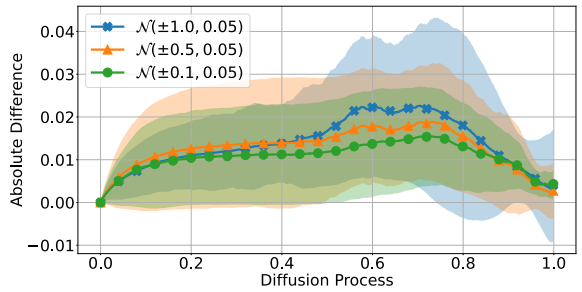


Figure 1: Classifier guidance decomposition (Eq. (4)) does not always hold. We apply classifier guidance on 1D data from $\mathcal{N}(\pm1.0, 0.05)$, $\mathcal{N}(\pm0.5, 0.05)$, and $\mathcal{N}(\pm0.1, 0.05)$ respectively. The denoising diffusion process starts from left to right. For each dataset, we train a vanilla conditional diffusion model and a decomposed version, *i.e.*, an unconditional diffusion model and a classifier. We generate 20k samples (10k for each class) from both sides of Eq. (4) with the same initial noises and compute the absolute differences for each step in the denoising diffusion process. This plot shows the average as well as the standard deviation for the difference. Apparently, the classifier guidance decomposition doesn't hold with equality.

$$\int p(\mathbf{x}_T|c) \prod_{t=0}^{T-1} p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, c)\, d\mathbf{x}_{1:T} \approx \int p(\mathbf{x}_T) \prod_{t=0}^{T-1} p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, c)\, d\mathbf{x}_{1:T}. \tag{1}$$

The last step is due to $p(\mathbf{x}_T|c)$ being almost identical to an isotropic Gaussian, independent of the condition c.

Following DDPM (Ho, Jain, and Abbeel 2020), $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, c)$ is defined as a Gaussian $\mathcal{N}(\mathbf{x}_t; \mu_\theta(\mathbf{x}_{t+1}, t+1, c), (1-\alpha_{t+1})\mathbf{I})$, trying to reverse a forward diffusion process. Here $\{\alpha_t\}_{t=1}^T$ is a pre-defined schedule for the forward diffusion process and $\mu_\theta(\mathbf{x}_{t+1}, t+1, c)$ is tasked to predict the corresponding $\mathbf{x}_t$ in the forward diffusion process. Specifically, the forward diffusion process gradually corrupts the clean data $\mathbf{x}_0$ with Gaussian noise: $\mathbf{x}_{t+1} \sim q(\mathbf{x}_{t+1}|\mathbf{x}_t) \triangleq \mathcal{N}(\mathbf{x}_{t+1}; \sqrt{\alpha_{t+1}}\mathbf{x}_t, (1-\alpha_{t+1})\mathbf{I}), \forall t \in \{0, \dots, T-1\}$. Notably, with $\bar{\alpha}_t \triangleq \prod_{s=1}^t \alpha_s$, we have $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Therefore, Ho, Jain, and Abbeel (2020) propose to reduce the learning of $\mu_\theta(\mathbf{x}_{t+1}, t+1, c)$ to predicting the noise with $\boldsymbol{\epsilon}_\theta(\mathbf{x}_{t+1}, t+1, c)$ as we have $\mu_\theta(\mathbf{x}_{t+1}, t+1, c) =$

$$\frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_{t+1} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_{t+1}, t+1, c) \right). \tag{2}$$

Leveraging the link between denoising diffusion and score matching (Song et al. 2021; Vincent 2011), we have

$$\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, c) = -\sqrt{1-\bar{\alpha}_t}\, \nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|c). \tag{3}$$

3.2 Classifier Guidance Revisited

Dhariwal and Nichol (2021) propose *classifier guidance* to decompose the conditional denoising diffusion process in Eq. (1) as follows:

$$p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, c) = Z\, p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})\, p_\theta(c|\mathbf{x}_t). \tag{4}$$

Z is a normalizing factor independent of $\mathbf{x}_t$. $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})$ is an unconditional denoising diffusion process and $p_\theta(c|\mathbf{x}_t)$ is a classifier used to predict the probability that $\mathbf{x}_t$ aligns with the conditioning information c. Note, *Eq.* (4) *is not a trivial Bayes expansion*. See the appendix for more.

Revisiting the derivation of Eq. (4), the key step reduces to the following definition (see appendix):

$$\hat{q}(\mathbf{x}_{t+1}|\mathbf{x}_t, c) \triangleq q(\mathbf{x}_{t+1}|\mathbf{x}_t). \tag{5}$$

Note, $\hat{q}(\mathbf{x}_{t+1}|\mathbf{x}_t, c)$ is a newly-defined conditional forward diffusion process. At a high level, Eq. (5) tries to convert any *conditional* forward diffusion process, *i.e.*, $\hat{q}(\mathbf{x}_{t+1}|\mathbf{x}_t, c)$, into an *unconditional* forward diffusion process, *i.e.*, $q(\mathbf{x}_{t+1}|\mathbf{x}_t)$. Based on Eq. (5), Dhariwal and Nichol (2021) derive that the reverse process of $\hat{q}(\mathbf{x}_{t+1}|\mathbf{x}_t, c)$ can be decomposed into a combination of an unconditional denoising diffusion process and a classifier prediction as in Eq. (4).

However, it is questionable whether the assumption of Eq. (5) always holds: why should a conditional denoising process behave identical to an unconditional one? If Eq. (5) does not hold everywhere, the two sides in Eq. (4) may differ too. Indeed, our experiments on synthetic 1D data verify our suspicion as shown in Fig. 1. Furthermore, not only does the vanilla conditional model (left side of Eq. (4)) behave differently from the proposed decomposition (right side of Eq. (4)), but different instantiations of the classifier $p_\theta(c|\mathbf{x}_t)$ will produce significantly divergent behaviors as well. See the "guidance scale = 1" plots in Fig. 2a and 2b.

Large classifier guidance scale w introduced by Dhariwal and Nichol (2021) will amplify the difference demonstrated above. Specifically, Dhariwal and Nichol (2021) suggest increasing the impact of the classifier with $w > 1$ and sampling from a distribution that is skewed towards high classifier confidence:

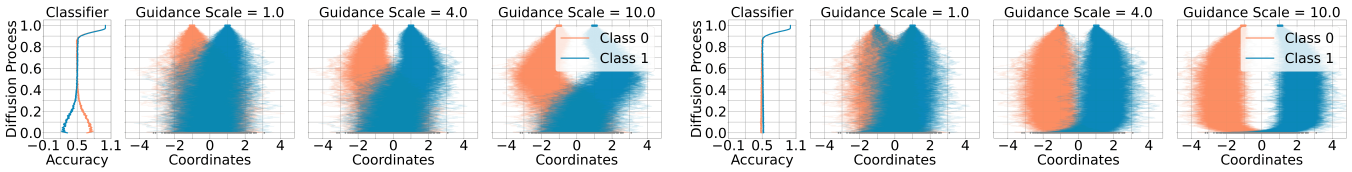
$$\mathbf{x}_t \sim p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})\, p_\theta(c|\mathbf{x}_t)^w. \tag{6}$$

Similar to Eq. (3), Dhariwal and Nichol (2021) show that Eq. (6) can be re-formulated such that $\mathbf{x}_t$ can be sampled via predicting the following noise $\tilde{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t, t, c) \triangleq$

$$\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) - w \cdot \sqrt{1-\bar{\alpha}_t}\, \nabla_{\mathbf{x}_t} \log p_\theta(c|\mathbf{x}_t), \tag{7}$$

where $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ is the corresponding noise estimator for the unconditional denoising diffusion process $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})$. As shown in Fig. 2, classifier guidance's behaviors are dominated by the characteristics of the classifier.

It is worth noticing that classifier guidance with larger and larger guidance scales continuously distorts the straight-like denoising diffusion trajectories to *push them away from the classifier's decision boundary*. In other words, the goal of *conditional* generation via classifier guidance is achieved by avoiding those areas in which the classifier is uncertain. This explains why a large guidance scale can produce high-fidelity images compared to results generated with a low guidance scale. The reason is that a low guidance scale is



(a) Classifier guidance with a nonlinear classifier.

(b) Classifier guidance with a linear classifier.

Figure 2: Classifier guidance behavior is dominated by the classifier. We apply denoising diffusion models with classifier guidance on a 1D dataset with data from $\mathcal{N}(\pm 1.0, 0.05)$. The classifiers in Fig. 2a and 2b differ. The denoising diffusion process for all plots starts from the bottom to the top. In Fig. 2a and 2b, the first plot demonstrates the classifier's accuracy on a validation set for each class through the diffusion process, *i.e.*, $p_\theta(c|\mathbf{x}_t)$ in Eq. (4), while the remaining three plots display the diffusion trajectories with different guidance scales. We observe: 1) classifier guidance essentially pushes the diffusion process away from the classifier's decision boundary that is around the origin; and 2) different classifiers can produce entirely different trajectories (Fig. 2a *vs*. 2b). Since we use the same initial noise and the same unconditional diffusion model, *i.e.*, $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1})$ in Eq. (4), for all plots, differences are solely due to the classifier.

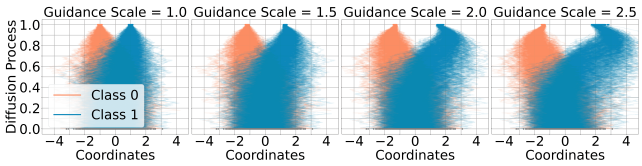


Figure 3: Classifier-free guidance distorts denoising diffusion trajectories. We apply denoising diffusion models with classifier-free guidance on a 1D dataset composed of data from $\mathcal{N}(\pm 1.0, 0.05)$. The denoising diffusion process for all plots starts from the bottom to the top. We use the same trained model as well as the same initial noise for all plots. The trajectory differences are solely caused by different guidance scales. Different scales in Fig. 2 and this figure arise from classifier guidance and classifier-free guidance's differing sensitivities. Here, scale=2.5 distorts trajectories significantly, while Fig. 2's scale=4 causes minor changes. We hypothesize that classifier-free guidance's greater sensitivity stems from its training with conditioning dropout. test

not strong enough to move the diffusion trajectories away from areas on the data distribution manifold where different conditional information intersect. Due to the entanglement, these areas naturally form the decision boundary for a *well-trained* classifier. With a large guidance scale and a well-trained classifier, classifier guidance can completely avoid ambiguous areas on the image manifold and generate results *unambiguously* aligned with the conditional information.

The obvious next question: can this reasoning for classifier guidance be generalized to classifier-free guidance?

3.3 Classifier-Free Guidance Revisited

Classifier-free guidance was introduced to eliminate the reliance on a separate classifier (Ho and Salimans 2021). Intuitively, with Bayes rule, we have $p(c|\mathbf{x}_t) = p(c)\,p(\mathbf{x}_t|c)/p(\mathbf{x}_t)$. Consqeuently, $\nabla_{\mathbf{x}_t} \log p_\theta(c|\mathbf{x}_t)$ can be decomposed as $\nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t|c) - \nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t)$, where $p(c)$ disappears as it is independent of $\mathbf{x}_t$. When substituting this into Eq. (7) and considering Eq. (3), we have

$$\tilde{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t, t, c) = \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) + w \cdot (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, c) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t))\,. \tag{8}$$

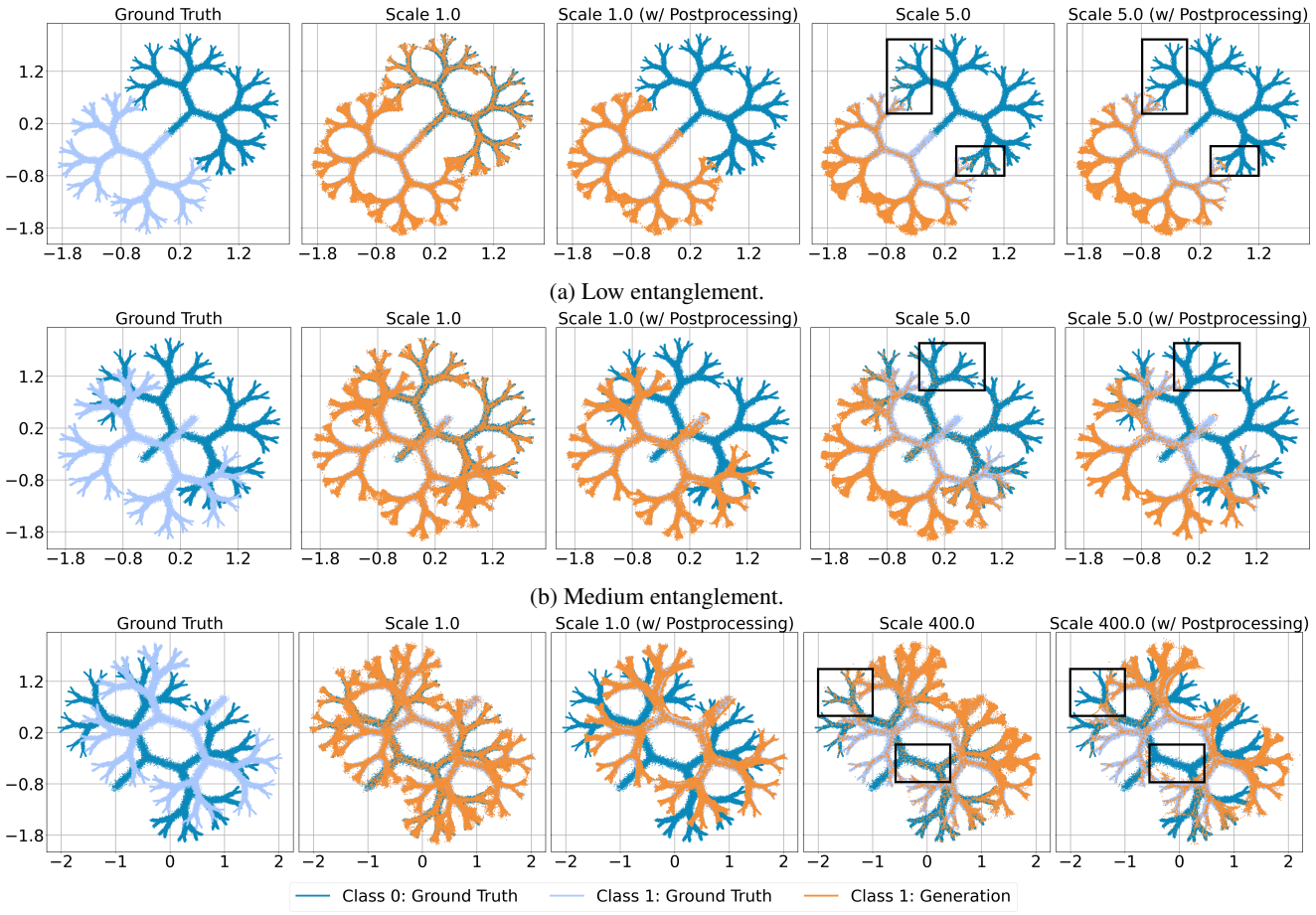
The effect of classifier guidance can be achieved by training two noise estimators for both conditional ($\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, c)$) and unconditional ($\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$) denoising diffusion processes respectively. In practice, Ho and Salimans (2021) propose to only train one conditional denoising diffusion model but randomly drop out the conditioning information c during training to mimic the unconditional process.

One may notice that Eq. (4) *lays the foundation for classifier-free guidance (Ho and Salimans 2021)*. If this is not clear, please refer to the appendix for more details. We want to know whether the connection between classifier guidance and classifier-free guidance can be used to show that classifier-free guidance inherits the characteristics of classifier guidance. Namely, does classifier-free guidance also try to push the diffusion trajectories away from the data's decision boundary? Note, classifier-free guidance does not directly involve any explicit classifier. However, based on our discussion in Sec. 3.2, a well-trained classifier's decision boundary naturally aligns with the data's decision boundary. Experiments on 1D synthetic datasets provide an affirmative answer as shown in Fig. 3.

3.4 Verification on High-Dimensional Data

So far, we have developed and verified our understanding on synthetic 1D data thanks to clear visualizations. A natural question arises: how can we assess our understanding on high-dimensional data, where visualizing the decision boundary, let alone the denoising trajectory, is non-trivial? To address this, we propose an alternative approach that perturbs samples away from the decision boundary. This allows us to *indirectly* evaluate our classifier-centric understanding.

If our classifier-centric understanding is correct, *i.e.*, if classifier guidance and classifier-free guidance perform conditional generation by moving samples away from the decision boundary, then low-quality generations should occur more frequently near the decision boundary, due to its complex structure in high-dimensional spaces. For a pre-trained generative model, if we can construct a postprocess-



(a) Low entanglement.

(b) Medium entanglement.

(c) High entanglement.

Figure 4: Classifier guidance with flow-matching based postprocessing (Sec. 3.4) on 2D fractal data. After training, all three classifiers' decision boundaries roughly align with the diagonal from top-left to bottom-right. See appendix for the experiment setup. In Fig. 4a to 4c, the 3^{rd} (and 5^{th}) plot show generated samples after applying postprocessing on generations from the 2^{nd} (and 4^{th}) plot. For a clear visualization, we only display generations for one class (see appendix for the other class).

ing step that moves samples away from the decision boundary, the overall quality of the generation should improve.

For this, we develop a generic postprocessing step to help verify our understanding on high-dimensional data. Let $\mathcal{X}_{\text{real}}$ refer to a set of samples from the real data distribution while $\mathcal{X}$ denotes a set of generated samples from a generative model. Note, $\mathcal{X}$ is agnostic to the specific choice of generation strategy, *e.g.*, it does not matter whether $\mathcal{X}$ was produced with classifier guidance or classifier-free guidance. Our goal is to move the distribution underlying $\mathcal{X}$ closer to the distribution represented by $\mathcal{X}_{\text{real}}$ mainly around the decision boundary. We train a rectified flow (Liu, Gong, and Liu 2023; Lipman et al. 2023) v_θ via

$$\min_{v_\theta} \int_0^1 \mathbb{E}_{\mathcal{X}} \left[\| (\hat{\mathbf{x}} - \texttt{NN}(\hat{\mathbf{x}}, \mathcal{X}_{\text{real}})) - v_\theta(\hat{\mathbf{x}}_t, c, t) \|^2 \right] dt, \quad (9)$$

where $\hat{\mathbf{x}} \sim \mathcal{X}$, $\hat{\mathbf{x}}_t = (1-t) \cdot \hat{\mathbf{x}} + t \cdot \texttt{NN}(\hat{\mathbf{x}}, \mathcal{X}_{\text{real}})$. Here $\texttt{NN}(\hat{\mathbf{x}}, \mathcal{X}_{\text{real}})$ represent the nearest neighbor sample for $\hat{\mathbf{x}}$ in the real data set $\mathcal{X}_{\text{real}}$.

Guidance Scale	$w = 1$	$w = 2$	$w = 4$
Class 0	5.95 / 3.11 / 1.57	5.31 / 1.70 / 1.01	11.7 / 2.05 / 1.12
Class 1	12.4 / 3.25 / 1.70	6.09 / 1.88 / 1.00	6.51 / 1.41 / 0.79

Table 1: Nearest neighbor distance between generations and ground truth for Fig. 5. We report average nearest neighbor (NN) distance ($\times 10^{-5}$) for 20k generations from the same noise, formatted as $A/B/C$: before postprocessing (A) / postprocessed with nearest (B) / postprocessed with random sampling from 20 candidates (C). B *vs*. C shows random sampling outperforms picking the nearest.

We emphasize the use of NN in Eq. (9), which differs from classic rectified flow formulations. Importantly, the use of NN automatically balances between 1) already-high-quality generations; and 2) low-quality generations. Based on our study in Sec. 3.2 and Sec. 3.3, the training will focus on generations around decision boundaries where low-quality generations usually occur. Concretely, if $\hat{\mathbf{x}}$ is already a high-

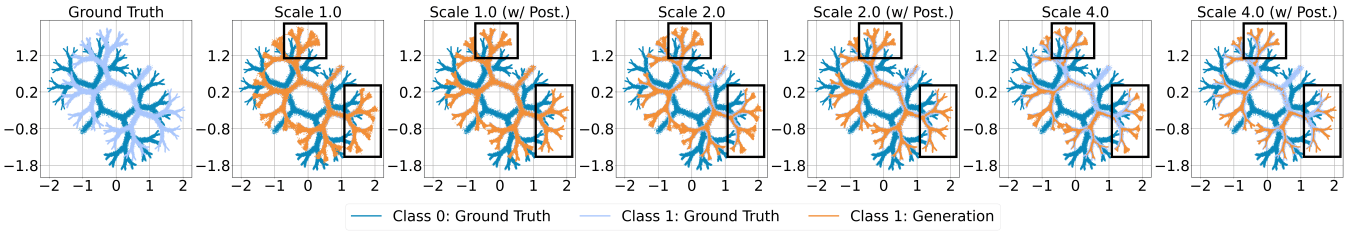


Figure 5: Classifier-free guidance with flow-matching based postprocessing (Sec. 3.4) on 2D fractal data. The level of entanglement is the same as that in Fig. 4c. Each plot with 'Post.' in the title displays generations after applying postprocessing on samples from the corresponding previous plot. We observe that the postprocessing step continuously improves the fidelity of the generations via moving samples around the decision boundary back to the real data distribution as the leaf branches become much sharper, regardless of the guidance scale we use.

fidelity generation, *i.e.*, close to $\mathcal{X}_{\text{real}}$, $\hat{\mathbf{x}} - \texttt{NN}(\hat{\mathbf{x}}, \mathcal{X}_{\text{real}})$ will be extremely small, providing a negligible learning signal. In practice, inspired by (Tong et al. 2024), we do not always use the nearest neighbor $\texttt{NN}(\hat{\mathbf{x}}, \mathcal{X}_{\text{real}})$. Instead, we first find top-k nearest neighbors and randomly select one from the top-k as the target during each training iteration. This randomness provides more opportunities to avoid local optima.

After training the postprocessing flow, conditional generation involves two steps: 1) sampling from the original denoising diffusion model $p_\theta(\mathbf{x}_0|c)$ in Eq. (1) to obtain a sample $\hat{\mathbf{x}}_0$; and 2) running an ODE solver over the time interval $[0, 1]$ to solve $d\mathbf{z}_t/dt = v_\theta(\mathbf{z}_t, c, t)$ numerically, starting from $\mathbf{z}_0 = \hat{\mathbf{x}}_0$. The ODE solver's output $\mathbf{z}_1$ is the final generation. We use $\hat{\mathbf{x}}_0$ for the base model output and $\mathbf{z}_t$ to emphasize that postprocessing is based on a separate flow matching procedure.

The proposed postprocessing is related to autoguidance (Karras et al. 2024a), which guides the model training with a bad version of itself. Autoguidance moves samples in the direction given by the difference between an inferior version and the current model. In contrast, our postprocessing flow model is based on a pre-trained model and real data. More importantly, we propose the postprocessing step primarily to verify our classifier-centric understanding.

4 Experiments

4.1 2D Fractal

The 2D Fractal dataset is represented by a mixture of Gaussians, similar to the dataset used by Karras et al. (2024a). Please refer to the appendix for details. As displayed in "Ground Truth" plots in Fig. 4, *this synthetic dataset provides an easy way to control the level of entanglement among data from different classes*. We use this dataset to verify 1) our analysis in Sec. 3 on 1D data can be generalized; and 2) the postprocessing step in Sec. 3.4 is an effective proxy to assess our classifier-centric understanding.

We qualitatively illustrate the results for classifier guidance and classifier-free guidance in Fig. 4 and Fig. 5 respectively. We choose top-20 nearest neighbors when training the rectified flow in the experiments. As can be seen clearly, the larger the guidance scale, the further are the samples from the decision boundaries, corroborating our classifier-centric understanding. Further, the proposed postprocessing step mainly moves samples around the decision boundaries while keeping generations that are already close to real data untouched. Quantitatively, Tab. 1 verifies the effectiveness of our approach for Fig. 5.

Additionally, according to Eq. (8), when using a scale of 1 for classifier-free guidance, we essentially sample from a pure conditional model, *i.e.*, $p_\theta(\mathbf{x}_t|\mathbf{x}_{t+1}, c)$ on Eq. (4) left side. The comparison between two plots of "Scale 1.0" in Fig. 4c and Fig. 5 corroborates our analysis in Sec. 3.2 that the two sides of Eq. (4) are generally not equal.

CFG Scale Before Post.	Post. ✗	Post. ✓
2.25	8.016	5.821
2.50	9.402	5.936
2.75	10.75	6.176

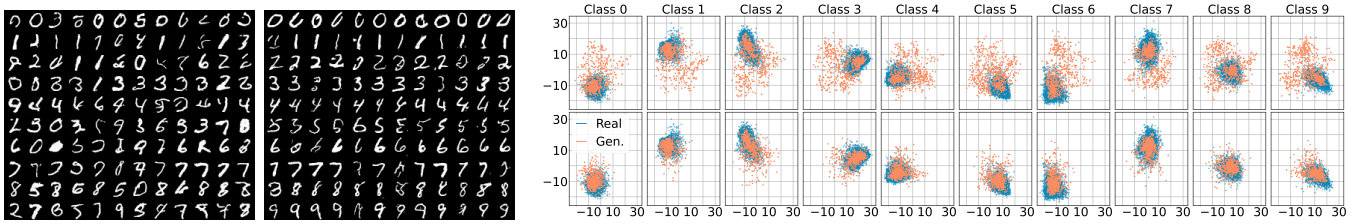
Table 2: Postprocessing for classifier-free guidance on CIFAR-10. We report conditional FID on 50k generations with seeds from 0 to 49999. Lower FID is better, and the best in each row is highlighted. Postprocessing is abbreviated as "Post.". See qualitative results in Fig. 7.

4.2 MNIST

Our proposed postprocessing step improves the fidelity of generations for both classifier guidance and classifier-free guidance on real-world MNIST (LeCun et al. 1998) data as shown in Fig. 6 and appendix. Based on our analysis in Sec. 3.4, this validates our classifier-centric understanding on MNIST. Our denoising diffusion and rectified flow models are based on a UNet (Ronneberger, Fischer, and Brox 2015) similar to the one used by Dhariwal and Nichol (2021). See appendix for experimental details.

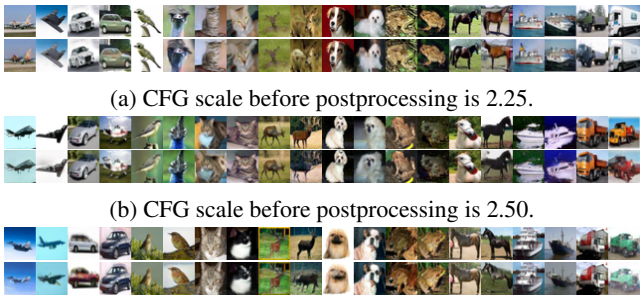
4.3 CIFAR-10

We further verify our classifier-centric understanding on image synthesis via Fréchet Inception Distance (FID) (Heusel et al. 2017), a commonly-used metric for generation quality, on CIFAR-10 (Krizhevsky 2009). See appendix for details. We choose EDM (Karras et al. 2022), one of the state-of-the-art diffusion-based generative models on CIFAR-10, as our pre-trained model. However, the pre-trained EDM model



(a) Generations before postprocessing step. (b) Generations after postprocessing step. (c) Postprocessing moves generations around the decision boundaries closer to the real data distribution.

Figure 6: Classifier guidance (scale 1.0) with flow-matching based postprocessing (Sec. 3.4) on MNIST. Fig. 6a and 6b share the same initial noises for corresponding cells. Conditioning information from top to bottom row is the digit 0 to 9. Fig. 6c shows that the flow matching based postprocessing clearly improves the alignment between generations and conditioning: The top and bottom rows in Fig. 6c correspond to Fig. 6a and 6b respectively. The postprocessing moves the learned distribution (orange clusters) closer to the real one (blue clusters). See appendix for the experiment details.

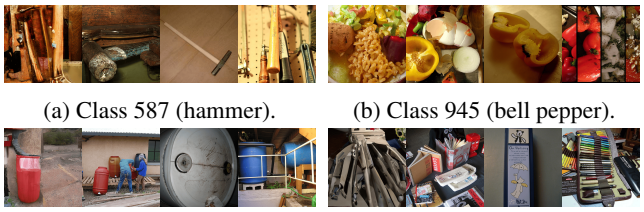


(a) CFG scale before postprocessing is 2.25.

(b) CFG scale before postprocessing is 2.50.

(c) CFG scale before postprocessing is 2.75.

Figure 7: Tab. 2's visualizations. In Fig. 7a to 7c, the top/bottom display generations before/after postprocessing. Most already high-quality samples remain unchanged.



(a) Class 587 (hammer). (b) Class 945 (bell pepper). (c) Class 756 (rain barrel). (d) Class 709 (pencil box).

Figure 8: Nearest neighbor based on different distance metrics for ImageNet 512^2. For each plot, from left to right, we display the generation from pre-traiend EDM2 (Karras et al. 2024b) and its nearest neighbor from the real images based on Euclidean distance in the feature space of DINOv2 (Oquab et al. 2023), Stable Diffusion's VAE (Rombach et al. 2022b), and state-of-the-art classifier from `timm` (Wightman 2019) respectively. Different distance metrics can produce significantly different results.

lacks conditioning dropout, making it incompatible with CFG. We re-train a CFG-compatible EDM, verifying correctness with FID 1.850 (ours with CFG scale 1.0) *vs*. 1.849 (pre-trained). As shown in Tab. 2, the postprocessing step improves the generation quality across various guidance scales. Based on our analysis in Sec. 3.4, our classifier-centric understanding holds on this high-dimensional data.

5 Discussion

`NN` in Eq. (9). Unlike the 2D Fractal case in Sec. 4.1, where Euclidean distance is well-defined, there's no clear distance definition for high-dimensional data. In an ablation on CIFAR-10 in the appendix, we find that while our postprocessing consistently improves overall generation quality, *i.e*., it can be a proxy for assessing our classifier-centric understanding (Sec. 3.4), specific performance can vary.

On ImageNet (Deng et al. 2009). We experimented with our postprocessing on ImageNet 512^2 with various distance metrics, but the performance is not satisfactory. Due to the intricate structure of this high-dimensional space, determining a reasonable distance metric to apply Eq. (9) is difficult, and this is an active research area (Aggarwal, Hinneburg, and Keim 2001; Qian et al. 2015; Zhang et al. 2018; Stein et al. 2023). In Fig. 8, we list some qualitative examples on nearest neighbors with distance space we tried, spanning features from self-supervised model (Oquab et al. 2023), VAE (Rombach et al. 2022a), and classifier (Wightman 2019). The behavior differs significantly. Thus, we leave the experiment as a future work.

Limitations. Since our postprocessing step runs another round of diffusion, to verify our classifier-centric understanding, inference time will be doubled when compared to the generation process without postprocessing. However, with more prevailing distillation techniques and faster samplers, we think the overhead can be largely mitigated.

6 Conclusion

We carry out an empirical study aiming to understand classifier-free guidance from a classifier-centric perspective. Our analysis reveals that both classifier-free guidance and classifier guidance push the denoising diffusion process away from the data's decision boundaries on 1D data. For high-dimensional data, we propose a flow matching based postprocessing step to indirectly assess our classifier-centric understanding and verify its effectiveness across datasets.

Acknowledgments

Work supported in part by NSF grants 2008387, 2045586, 2106825, MRI 1725729, and NIFA award 2020-67021-32799.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. GPT-4 Technical Report. *arXiv*.

Ackley, D. H.; Hinton, G. E.; and Sejnowski, T. J. 1985. A Learning Algorithm for Boltzmann Machines. *Cognitive Science*.

Aggarwal, C. C.; Hinneburg, A.; and Keim, D. A. 2001. On the Surprising Behavior of Distance Metrics in High Dimensional Spaces. In *International Conference on Database Theory*.

Aneja, J.; Schwing, A. G.; Kautz, J.; and Vahdat, A. 2021. A Contrastive Learning Approach for Training Variational Autoencoder Priors. In *NeurIPS*.

Azadi, S.; Olsson, C.; Darrell, T.; Goodfellow, I. J.; and Odena, A. 2019. Discriminator Rejection Sampling. In *ICLR*.

Bauer, M.; and Mnih, A. 2019. Resampled Priors for Variational Autoencoders. In *AISTATS*.

Bradley, A.; and Nakkiran, P. 2024. Classifier-Free Guidance is a Predictor-Corrector. *arXiv*.

Brock, A.; Donahue, J.; and Simonyan, K. 2018. Large Scale GAN Training for High Fidelity Natural Image Synthesis. *arXiv*.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; teusz Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language Models are Few-Shot Learners. In *NeurIPS*.

Casella, G.; Robert, C. P.; and Wells, M. T. 2004. Generalized Accept-Reject Sampling Schemes. *Lecture notes-monograph series*.

Che, T.; Zhang, R.; Sohl-Dickstein, J. N.; Larochelle, H.; Paull, L.; Cao, Y.; and Bengio, Y. 2020. Your GAN is Secretly an Energy-based Model and You Should use Discriminator Driven Latent Sampling. In *NeurIPS*.

Chen, N.; Zhang, Y.; Zen, H.; Weiss, R. J.; Norouzi, M.; and Chan, W. 2021. WaveGrad: Estimating Gradients for Waveform Generation. In *ICLR*.

Chidambaram, M.; Gatmiry, K.; Chen, S.; Lee, H.; and Lu, J. 2024. What Does Guidance Do? A Fine-Grained Analysis in a Simple Setting. In *NeurIPS*.

Chung, H.; Kim, J.; Park, G. Y.; Nam, H.; and Ye, J. C. 2024. CFG++: Manifold-Constrained Classifier Free Guidance for Diffusion Models. *arXiv*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion Models Beat GANs on Image Synthesis. In *NeurIPS*.

Dinh, A.-D.; Liu, D.; and Xu, C. 2023a. PixelAsParam: A Gradient View on Diffusion Sampling with Guidance. In *ICML*.

Dinh, A.-D.; Liu, D.; and Xu, C. 2023b. Rethinking Conditional Diffusion Sampling with Progressive Guidance. In *NeurIPS*.

Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. 2024. The Llama 3 Herd of Models. *arXiv*.

Galatolo, F. A.; Cimino, M. G. C. A.; and Vaglini, G. 2021. Generating Images from Caption and Vice Versa via CLIP-Guided Generative Latent Space Search. In *International Conference on Image Processing and Vision Engineering*.

Grover, A.; Gummadi, R.; Lázaro-Gredilla, M.; Schuurmans, D.; and Ermon, S. 2018. Variational Rejection Sampling. In *AISTATS*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *NeurIPS*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In *NeurIPS*.

Ho, J.; and Salimans, T. 2021. Classifier-Free Diffusion Guidance. In *NeurIPS Workshop*.

Hong, S.; Lee, G.; Jang, W.; and Kim, S. W. 2023. Improving Sample Quality of Diffusion Models Using Self-Attention Guidance. In *ICCV*.

Jankowiak, M.; and Phan, D. 2023. Reparameterized Variational Rejection Sampling. In *AISTATS*.

Karras, T.; Aittala, M.; Aila, T.; and Laine, S. 2022. Elucidating the Design Space of Diffusion-Based Generative Models. In *NeurIPS*.

Karras, T.; Aittala, M.; Kynkäänniemi, T.; Lehtinen, J.; Aila, T.; and Laine, S. 2024a. Guiding a Diffusion Model with a Bad Version of Itself. In *NeurIPS*.

Karras, T.; Aittala, M.; Lehtinen, J.; Hellsten, J.; Aila, T.; and Laine, S. 2024b. Analyzing and Improving the Training Dynamics of Diffusion Models. In *CVPR*.

Karras, T.; Laine, S.; and Aila, T. 2018. A Style-Based Generator Architecture for Generative Adversarial Networks. In *CVPR*.

Kim, D.; Kim, Y.; Kang, W.; and Moon, I.-C. 2022. Refining Generative Process with Discriminator Guidance in Score-based Diffusion Models. In *ICML*.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative Flow with Invertible 1x1 Convolutions. In *NeurIPS*.

Kingma, D. P.; Salimans, T.; Poole, B.; and Ho, J. 2021. Variational Diffusion Models. In *NeurIPS*.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *ICLR*.

Kong, Z.; Ping, W.; Huang, J.; Zhao, K.; and Catanzaro, B. 2021. DiffWave: A Versatile Diffusion Model for Audio Synthesis. In *ICLR*.

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. *arXiv*.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*.

Lin, S.; and Yang, X. 2024. Diffusion Model with Perceptual Loss. *arXiv*.

Lipman, Y.; Chen, R. T. Q.; Ben-Hamu, H.; Nickel, M.; and Le, M. 2023. Flow Matching for Generative Modeling. In *ICLR*.

Liu, X.; Gong, C.; and Liu, Q. 2023. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *ICLR*.

Metropolis, N.; Rosenbluth, A. W.; Rosenbluth, M. N.; Teller, A. H.; and Teller, E. 1953. Equation of State Calculations by Fast Computing Machines. *The journal of Chemical Physics*.

Oquab, M.; Darcet, T.; Moutakanni, T.; Vo, H. V.; Szafraniec, M.; Khalidov, V.; Fernandez, P.; Haziza, D.; Massa, F.; El-Nouby, A.; Howes, R.; Huang, P.-Y.; Xu, H.; Sharma, V.; Li, S.-W.; Galuba, W.; Rabbat, M.; Assran, M.; Ballas, N.; Synnaeve, G.; Misra, I.; Jegou, H.; Mairal, J.; Labatut, P.; Joulin, A.; and Bojanowski, P. 2023. DINOv2: Learning Robust Visual Features without Supervision. *arXiv*.

Parmar, N.; Vaswani, A.; Uszkoreit, J.; Kaiser, L.; Shazeer, N. M.; Ku, A.; and Tran, D. 2018. Image Transformer. In *ICML*.

Patashnik, O.; Wu, Z.; Shechtman, E.; Cohen-Or, D.; and Lischinski, D. 2021. StyleCLIP: Text-Driven Manipulation of StyleGAN Imagery. In *ICCV*.

Poole, B.; Jain, A.; Barron, J. T.; and Mildenhall, B. 2023. DreamFusion: Text-to-3D using 2D Diffusion. In *ICLR*.

Qian, Q.; Jin, R.; Zhu, S.; and Lin, Y. 2015. Fine-grained visual categorization via multi-stage metric learning. In *CVPR*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; Krueger, G.; and Sutskever, I. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*.

Razavi, A.; van den Oord, A.; and Vinyals, O. 2019. Generating Diverse High-Fidelity Images with VQ-VAE-2. In *NeurIPS*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022a. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022b. High-Resolution Image Synthesis with Latent Diffusion Models. In *CVPR*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*.

Sadat, S.; Buhmann, J.; Bradley, D.; Hilliges, O.; and Weber, R. M. 2024. CADS: Unleashing the Diversity of Diffusion Models through Condition-Annealed Sampling. In *ICLR*.

Song, Y.; and Ermon, S. 2019. Generative Modeling by Estimating Gradients of the Data Distribution. In *NeurIPS*.

Song, Y.; Sohl-Dickstein, J. N.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *ICLR*.

Stein, G.; Cresswell, J. C.; Hosseinzadeh, R.; Sui, Y.; Ross, B. L.; Villecroze, V.; Liu, Z.; Caterini, A. L.; Taylor, J. E. T.; and Loaiza-Ganem, G. 2023. Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. In *NeurIPS*.

Tong, A.; Malkin, N.; Huguet, G.; Zhang, Y.; Rector-Brooks, J.; Fatras, K.; Wolf, G.; and Bengio, Y. 2024. Improving and Generalizing Flow-Based Generative Models With Minibatch Optimal Transport. *TMLR*.

Turner, R. D.; Hung, J.; Saatci, Y.; and Yosinski, J. 2018. Metropolis-Hastings Generative Adversarial Networks. In *ICML*.

Vincent, P. 2011. A Connection Between Score Matching and Denoising Autoencoders. *Neural Computation*.

Wightman, R. 2019. PyTorch Image Models. https://huggingface.co/timm/vit_so150m2_patch16_reg1_gap_448.sbb_e200_in12k_ft_in1k.

Wu, Y.; Chen, M.; Li, Z.; Wang, M.; and Wei, Y. 2024. Theoretical Insights for Diffusion Guidance: A Case Study for Gaussian Mixture Models. *arXiv*.

Xia, M.; Xue, N.; Shen, Y.; Yi, R.; Gong, T.; and Liu, Y.-J. 2024. Rectified Diffusion Guidance for Conditional Generation. *arXiv*.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *CVPR*.