Optimizing Input of Denoising Score Matching is Biased Towards Higher Score Norm

Tongda Xu
AIR & CST, Tsinghua University
x.tongda@nyu.edu

Abstract

Many recent works utilize denoising score matching to optimize the conditional input of diffusion models. In this workshop paper, we demonstrate that such optimization breaks the equivalence between denoising score matching and exact score matching. Furthermore, we show that this bias leads to higher score norm. Additionally, we observe a similar bias when optimizing the data distribution using a pre-trained diffusion model. Finally, we discuss the wide range of works across different domains that are affected by this bias, including MAR for auto-regressive generation [1], PerCo [2] for image compression, and DreamFusion [3] for text to 3D generation.

1 Introduction

The diffusion model is a type of generative model that consists of a Gaussian Markov chain in continuous space [4, 5, 6]. We denote the source image as x_0, and the forward diffusion process involves perturbing the image with Gaussian noise, i.e., $q(x_t) = x + \mathcal{N}(0, \sigma_t^2)$. To sample from the backward diffusion model, it is necessary to know the score function $\nabla_{x_t} \log q(x_t)$ in order to construct the reverse stochastic differential equation [7].

The learning of diffusion models relies on the estimation of the score function $\nabla_{x_t} \log q(x_t)$. The most straightforward approach is explicit score matching (ESM), which employs a neural network $s_\theta(x_t, t)$ parameterized by θ to approximate the score:

$$\mathcal{L}_{ESM}(\theta) = \mathbb{E}_{q(x_t|x)p(x)}[||s_\theta(x_t, t) - \nabla_{x_t} \log q(x_t)||^2]. \tag{1}$$

However, $\mathcal{L}_{ESM}$ is intractable because $\nabla x_t \log q(x_t)$ is itself intractable. [8, 9] proposed using denoising score matching (DSM) instead, as $\nabla_{x_t} \log q(x_t|x)$ is simply the score of a Gaussian distribution. Furthermore, [8] proved the equivalence between the $\mathcal{L}_{ESM}$ objective and the $\mathcal{L}_{DSM}$ objective when optimizing θ:

$$\mathcal{L}_{DSM}(\theta) = \mathbb{E}_{q(x_t|x)p(x)}[||s_\theta(x_t, t) - \nabla_{x_t} \log q(x_t|x)||^2]. \tag{2}$$

Theorem 1. *[8] When optimizing θ, the denoising score matching is equivalent to explicit score matching, i.e.,*

$$\mathcal{L}_{DSM}(\theta) \backsim \mathcal{L}_{ESM}(\theta). \tag{3}$$

Many recent works have begun to use $\mathcal{L}_{DSM}$ to optimize targets other than θ. For example, MAR [1] adopts $\mathcal{L}_{DSM}$ to optimize c, the conditional input of the diffusion model to train an auto-regressive image generation model:

$$\theta^*, c^* \leftarrow \arg\max \mathcal{L}_{DSM}(\theta, c) = \mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||s_\theta(x_t, t, c) - \nabla_{x_t} \log q(x_t|x, c)||^2]. \tag{4}$$

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Frontiers in Probabilistic Inference: Sampling Meets Learning.

Another example is SDS [3] for text-to-3D generation, which utilizes a pre-trained diffusion model $s_\theta(.,.,.)$ and $\mathcal{L}_{DSM}$ to optimize the input distribution $p(x)$:

$$p(x)^* \leftarrow \arg\max \mathcal{L}_{DSM}(p(x)) = \mathbb{E}_{q(x_t|x)p(x)}[\frac{1}{2}||s_\theta(x_t,t) - \nabla_{x_t}\log q(x_t|x)||^2]. \tag{5}$$

In this workshop paper, we show that the above usage of $\mathcal{L}_{DSM}$ is biased, as it breaks the equivalence between $\mathcal{L}_{DSM}$ and $\mathcal{L}_{ESM}$. Moreover, we demonstrate that this bias drives the optimization towards higher score norm. Finally, we discuss the broad range of works that are potentially affected by this bias.

2 Optimizing Condition with Diffusion Loss is Biased Towards Higher Score Norm

According to [8], $\mathcal{L}_{DSM}$ and $\mathcal{L}_{ESM}$ have the following relationship:

$$\begin{aligned}\mathcal{L}_{DSM}(\theta,c) =& \mathcal{L}_{ESM}(\theta,c) - \underbrace{\mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t}\log q(x_t|c)||^2]}_{C_2} \\ &+ \underbrace{\mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t}\log q(x_t|x,c)||^2]}_{C_3}.\end{aligned} \tag{6}$$

When optimizing θ, the C_2 and C_3 terms cancel out, as they are not related to θ. However, although θ is not related to these terms, the condition c is involved in both the C_2 and C_3 terms. Since x_t-x-c forms a Markov chain, the C_3 term has no gradient with respect to c.

Therefore, when optimizing c using $\mathcal{L}_{DSM}$, there is an additional term compared to $\mathcal{L}_{ESM}$. Since the equivalence between $\mathcal{L}_{DSM}$ and $\mathcal{L}_{ESM}$ is broken, $\mathcal{L}_{DSM}$ becomes biased.

Theorem 2. *When optimizing both θ and c, the denoising score matching is equivalent to explicit score matching with a bias term, i.e.,*

$$\mathcal{L}_{DSM}(\theta,c) \backsim \mathcal{L}_{ESM}(\theta,c) - \underbrace{\mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t}\log q(x_t|c)||^2]}_{C_2}. \tag{7}$$

We have shown that for the condition c, minimizing $\mathcal{L}_{DSM}$ is equivalent to simultaneously minimizing $\mathcal{L}_{ESM}$ and maximizing $C_2 = \mathbb{E}q(x_t|x)p(x|c)\left[\frac{1}{2}||\nabla x_t \log q(x_t|c)||^2\right]$. And therefore, optimizing $\mathcal{L}_{DSM}$ is biased towards higher score norm.

3 Optimizing Data Distribution with Diffusion Loss is Also Biased Towards Higher Score Norm

A similar result can be trivially shown when using the score matching diffusion loss with a pre-trained diffusion model to optimize the input data distribution $p(x)$.

Corollary 1. *When optimizing input distribution $p(x)$ with pre-trained diffusion model, the denosiing score matching is equivalent to explicit score matching with a bias term, i.e.,*

$$\mathcal{L}_{DSM}(p(x)) \backsim \mathcal{L}_{ESM}(p(x)) - \mathbb{E}_{q(x_t|x)p(x)}[\frac{1}{2}||\nabla_{x_t}\log q(x_t)||^2]. \tag{8}$$

4 Discussion: Works That are Effected by This Bias

Optimizing the condition or input distribution using the diffusion loss is a common technique in the application of diffusion models across various domains. Below, we highlight four areas with recent works that are affected by this bias; this list is by no means exhaustive.

Auto-regressive Generation: Various works in tokenization-free auto-regressive generation are affected by this bias. MAR [1] adopts the diffusion loss to optimize c, which is the output of the

auto-regressive model. MetaQuery [10] employs diffusion loss to optimize c, which is the output of the connector in MLLM models. Similarly, [11, 12, 13, 14, 15, 16, 17, 18] are also effected by this bias.

Image Compression: CDC [19] jointly minimizes the bitrate of c alongside the diffusion loss. PerCo [2] optimizes c with the diffusion loss, where the resulting gradient is used to refine a VQ-VAE. FlowMo [20] optimizes c with diffusion loss, whose gradient is used to jointly train a VQ-VAE [21].

Text-to-3D Generation: DreamFusion [3] utilizes a pre-trained 2D diffusion model to optimize the input distribution $p(x)$ using the diffusion loss. The gradient with respect to $p(x)$ is then used to optimize a 3D representation model that is employed to render $p(x)$. Similarly, [22, 23] are also effected by this bias.

Diffusion-based Inverse Problem Solving: RED-Diff [24] adopts the loss function used in DreamFusion to optimize $p(x)$, thereby solving inverse problems for operators such as super-resolution and de-blurring. Similarly, [25, 26] are also effected by this bias.

References

[1] Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024.

[2] Marlene Careil, Matthew J Muckley, Jakob Verbeek, and Stéphane Lathuilière. Towards image compression with perfect realism at ultra-low bitrates. In *The Twelfth International Conference on Learning Representations*, 2023.

[3] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

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

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

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

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

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

[9] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Neural Information Processing Systems*, 2019.

[10] Xichen Pan, Satya Narayan Shukla, Aashu Singh, Zhuokai Zhao, Shlok Kumar Mishra, Jialiang Wang, Zhiyang Xu, Jiuhai Chen, Kunpeng Li, Felix Juefei-Xu, et al. Transfer between modalities with metaqueries. *arXiv preprint arXiv:2504.06256*, 2025.

[11] Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024.

[12] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

[13] Hu Yu, Hao Luo, Hangjie Yuan, Yu Rong, and Feng Zhao. Frequency autoregressive image generation with continuous tokens. *ArXiv*, abs/2503.05305, 2025.

[14] NextStep Team Chunrui Han, Guopeng Li, Jingwei Wu, Quan Sun, Yan Cai, Yuang Peng, Zheng Ge, Deyu Zhou, Haomiao Tang, Hongyu Zhou, Kenkun Liu, Ailin Huang, Bin Wang, Changxing Miao, Deshan Sun, En Yu, Fukun Yin, Gang Yu, Hao Nie, Haoran Lv, Hanpeng Hu, Jia Wang, Jian Zhou, Jian-Yuan Sun, Kaijun Tan, Kang An, Kangheng Lin, Liang Zhao, Mei Chen, Peng Xing, Rui Wang, Shiyu Liu, Shutao Xia, Tianhao You, Wei Ji, Xianfang Zeng, Xin pu Han, Xuelin Zhang, Yana Wei, Yanming Xu, Yimin Jiang, Yingming Wang, Yu Zhou, Yu Han, Zi Meng, Binxing Jiao, Daxin Jiang, Xiangyu Tony Zhang, and Yibo Zhu. Nextstep-1: Toward autoregressive image generation with continuous tokens at scale. 2025.

[15] Qinyu Zhao, Jaskirat Singh, Ming Xu, Akshay Asthana, Stephen Gould, and Liang Zheng. Disa: Diffusion step annealing in autoregressive image generation. *ArXiv*, abs/2505.20297, 2025.

[16] Lijie Fan, Tianhong Li, Siyang Qin, Yuanzhen Li, Chen Sun, Michael Rubinstein, Deqing Sun, Kaiming He, and Yonglong Tian. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens. *ArXiv*, abs/2410.13863, 2024.

[17] Haotian Tang, Yecheng Wu, Shang Yang, Enze Xie, Junsong Chen, Junyu Chen, Zhuoyang Zhang, Han Cai, Yao Lu, and Song Han. Hart: Efficient visual generation with hybrid autoregressive transformer. *ArXiv*, abs/2410.10812, 2024.

[18] Lingwei Meng, Long Zhou, Shujie Liu, Sanyuan Chen, Bing Han, Shujie Hu, Yanqing Liu, Jinyu Li, Sheng Zhao, Xixin Wu, Helen Meng, and Furu Wei. Autoregressive speech synthesis without vector quantization. In *Annual Meeting of the Association for Computational Linguistics*, 2024.

[19] Ruihan Yang and Stephan Mandt. Lossy image compression with conditional diffusion models. *Advances in Neural Information Processing Systems*, 36:64971–64995, 2023.

[20] Kyle Sargent, Kyle Hsu, Justin Johnson, Fei-Fei Li, and Jiajun Wu. Flow to the mode: Mode-seeking diffusion autoencoders for state-of-the-art image tokenization. *ArXiv*, abs/2503.11056, 2025.

[21] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[22] Peihao Wang, Dejia Xu, Zhiwen Fan, Dilin Wang, Sreyas Mohan, Forrest Iandola, Rakesh Ranjan, Yilei Li, Qiang Liu, Zhangyang Wang, et al. Taming mode collapse in score distillation for text-to-3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9037–9047, 2024.

[23] Hyungjin Chung, Jeongsol Kim, Geon Yeong Park, Hyelin Nam, and Jong Chul Ye. Cfg++: Manifold-constrained classifier free guidance for diffusion models. *ArXiv*, abs/2406.08070, 2024.

[24] Morteza Mardani, Jiaming Song, Jan Kautz, and Arash Vahdat. A variational perspective on solving inverse problems with diffusion models. *arXiv preprint arXiv:2305.04391*, 2023.

[25] Cagan Alkan, Julio Oscanoa, Daniel Abraham, Mengze Gao, Aizada Nurdinova, Kawin Setsompop, John M Pauly, Morteza Mardani, and Shreyas Vasanawala. Variational diffusion models for blind mri inverse problems. In *NeurIPS 2023 workshop on deep learning and inverse problems*, 2023.

[26] Kushagra Pandey, Farrin Marouf Sofian, Felix Draxler, Theofanis Karaletsos, and Stephan Mandt. Variational control for guidance in diffusion models. *ArXiv*, abs/2502.03686, 2025.

A Proof of Main Results

Theorem 2. *When optimizing both θ and c, the denoising score matching is equivalent to explicit score matching with a bias term, i.e.,*

$$\mathcal{L}_{DSM}(\theta, c) \backsim \mathcal{L}_{ESM}(\theta, c) - \underbrace{\mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|c)||^2]}_{C_2}. \tag{9}$$

Proof.

$$\begin{aligned}\mathcal{L}_{ESM}(\theta, c) =& \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||s_\theta(x_t, t, c) - \nabla_{x_t} \log q(x_t|c)||^2] \\ =& \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||s_\theta(x_t, t, c)||^2] - \underbrace{\mathbb{E}_{q(x_t|c)}[s_\theta(x_t, t, c)^T \nabla_{x_t} \log q(x_t|c)]}_{S_1} \\ &+ \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|c)||^2].\end{aligned} \tag{10}$$

According to [8], the S_1 term can be transformed by applying log derivative twice:

$$
\begin{aligned}
S_1 =& \mathbb{E}_{q(x_t|c)}[s_\theta(x_t,t,c)^T \nabla_{x_t} \log q(x_t|c)] \\
=& \int_{x_t} q(x_t|c) s_\theta(x_t,t,c)^T \nabla_{x_t} \log q(x_t|c) dx_t \\
=& \int_{x_t} q(x_t|c) s_\theta(x_t,t,c)^T \frac{\nabla_{x_t} q(x_t|c)}{q(x_t|c)} dx_t \\
=& \int_{x_t} s_\theta(x_t,t,c)^T \nabla_{x_t} q(x_t|c) dx_t \\
=& \int_{x_t} s_\theta(x_t,t,c)^T (\nabla_{x_t} \int_x p(x|c) q(x_t|x,c) dx) dx_t \\
=& \int_{x_t} s_\theta(x_t,t,c)^T (\int_x p(x|c) \nabla_{x_t} q(x_t|x,c) dx) dx_t \\
=& \int_{x_t} s_\theta(x_t,t,c)^T (\int_x p(x|c) q(x_t|x,c) \nabla_{x_t} \log q(x_t|x,c) dx) dx_t \\
=& \int_{x_t} \int_x p(x|c) q(x_t|x,c) s_\theta(x_t,t,c)^T (\nabla_{x_t} \log q(x_t|x,c) dx) dx_t \\
=& \mathbb{E}_{q(x_t|x)p(x|c)}[s_\theta(x_t,t,c)^T \nabla_{x_t} \log q(x_t|x,c)].
\end{aligned} \tag{11}
$$

Substituting Eq. 11 into Eq. 10, we have

$$
\begin{aligned}
\mathcal{L}_{ESM}(\theta,c) =& \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||s_\theta(x_t,t,c)||^2] - \mathbb{E}_{q(xt,x|c)}[s_\theta(x_t,t,c)^T \nabla_{x_t} \log q(x_t|x,c)] \\
&+ \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|c)||^2].
\end{aligned} \tag{12}
$$

On the other hand, we have

$$
\begin{aligned}
\mathcal{L}_{DSM}(\theta,c) =& \mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||s_\theta(x_t,t,c) - \nabla_{x_t} \log q(x_t|x,c)||^2] \\
=& \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||s_\theta(x_t,t,c)||^2] - \mathbb{E}_{q(xt,x|c)}[s_\theta(x_t,t,c)^T \nabla_{x_t} \log q(x_t|x,c)] \\
&+ \underbrace{\mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|x,c)||^2]}_{C_3} \\
=& \mathcal{L}_{ESM}(\theta,c) - \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|c)||^2] + C_3.
\end{aligned} \tag{13}
$$

For the C_3 term, we notice that $x_t - x - c$ forms a Markovian chain, and therefore

$$
C_3 = \mathbb{E}_{q(x_t|x)p(x|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|x)||^2]. \tag{14}
$$

As C_3 is not related to θ and c, we have

$$
\mathcal{L}_{ESM}(\theta,c) - \mathbb{E}_{q(x_t|c)}[\frac{1}{2}||\nabla_{x_t} \log q(x_t|c)||^2] \backsim \mathcal{L}_{DSM}(\theta,c). \tag{15}
$$

□