北京大学 PEKING UNIVERSITY

北京中关村学院 Zhongguancun Academy

Diagnosing and Improving Diffusion Models by Estimating the Optimal Loss Value

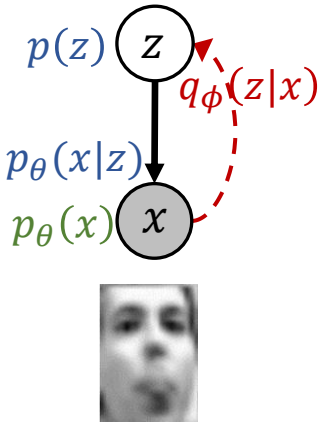
Yixian Xu[=] Shengjie Luo[=] Liwei Wang Di He* Chang Liu*



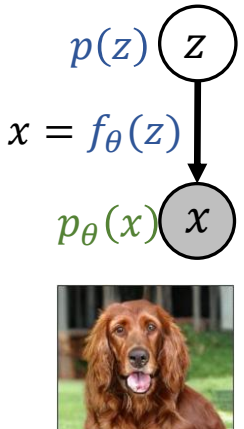


Deep Generative Models

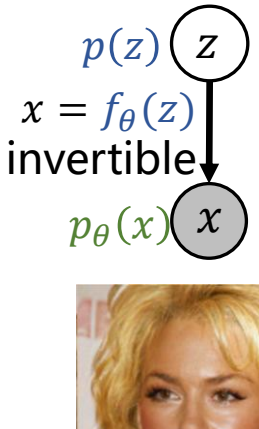
Variational Auto-Encoders (2013)



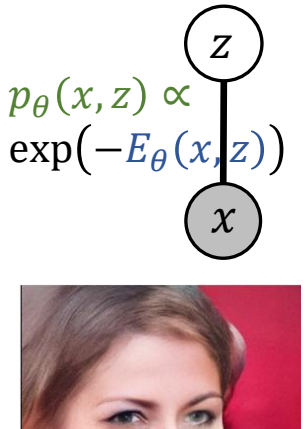
Generative Adversarial Nets (2014)

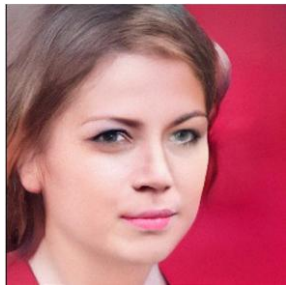


Normalizing Flows (2014)



Energy/Score-based generative models (2019)



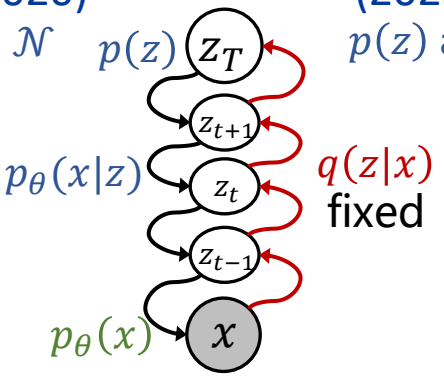


Diffusion Models (2020)

$p(z) = \mathcal{N}$

Flow-Matching (2023)

$p(z)$ arbitrary

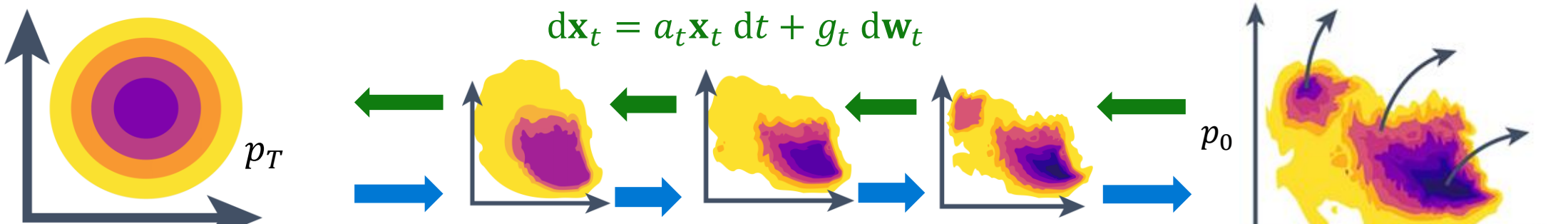


Images from:
[1] Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *International Conference on Learning Representations*.
[2] Brock, A., Donahue, J., & Simonyan, K. (2019). Large Scale GAN Training for High Fidelity Natural Image Synthesis. *International Conference on Learning Representations*.
[3] Kingma, D. P., & Dhariwal, P. (2018). Glow: Generative Flow with Invertible 1x1 Convolutions. *Neural Information Processing Systems*.
[4] Song, Y., & Ermon, S. (2020). Improved Techniques for Training Score-Based Generative Models. *Neural Information Processing Systems*.
[5] https://openai.com/index/sora-2/

Diffusion Models

forward diffusion process

$$\mathrm{d}\mathbf{x}_t = a_t \mathbf{x}_t \, \mathrm{d}t + g_t \, \mathrm{d}\mathbf{w}_t$$



A simple distribution

reverse process

$$\mathrm{d}\mathbf{x}_{\bar{t}} = -a_{T-\bar{t}}\mathbf{x}_{\bar{t}} \, \mathrm{d}\bar{t} + g_{T-\bar{t}}^2 \nabla \log p_{T-\bar{t}}(\mathbf{x}_{\bar{t}}) \, \mathrm{d}\bar{t} + g_{T-\bar{t}} \, \mathrm{d}\mathbf{w}_{\bar{t}}$$

Target distribution

$$\log p_T(\mathbf{x}_T) + \int_0^T \nabla \cdot \left(-a_{T-\bar{t}}\mathbf{x}_{\bar{t}} + \frac{g_{T-\bar{t}}^2}{2} \nabla \log p_{T-\bar{t}}(\mathbf{x}_{\bar{t}}) \right) \mathrm{d}\bar{t} = \log p_\theta(\mathbf{x}_0)$$

$$\approx \uparrow \mathbf{s}_{\theta, T-\bar{t}}(\mathbf{x}_{\bar{t}})$$

Diffusion Models

- Training: $\min_\theta \mathbb{E}_{p_t(\mathbf{x}_t)} \left\| \mathbf{s}_{\theta,t}(\mathbf{x}_t) - \nabla \log p_t(\mathbf{x}_t) \right\|^2$

$\nabla \log p_t(\mathbf{x}_t) = \mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)} \left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) \right]$

➔ $\min_\theta \mathbb{E}_{\underbrace{p_t(\mathbf{x}_t)p(\mathbf{x}_0|\mathbf{x}_t)}} \left\| \mathbf{s}_{\theta,t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) \right\|^2$

$= p_0(\mathbf{x}_0) p(\mathbf{x}_t|\mathbf{x}_0)$: samples available!

$p(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\alpha_t \mathbf{x}_0, \sigma_t^2 \mathbf{I})$,

where $\alpha_t := e^{\int_0^t a_s \, \mathrm{d}s}$, $\sigma_t^2 := \int_0^t \frac{g_s^2}{\alpha_s^2} \, \mathrm{d}s$ available.

$\mathbf{x}_t \sim p(\mathbf{x}_t|\mathbf{x}_0) \Leftrightarrow \mathbf{x}_t = \alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}, \ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

➔ $\min_\theta \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \mathbf{s}_{\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) + \frac{\boldsymbol{\epsilon}}{\sigma_t} \right\|^2$

$\mathbf{s}_{\theta,t}(\mathbf{x}_t) =: -\boldsymbol{\epsilon}_{\theta,t}(\mathbf{x}_t)/\sigma_t$

➔ $\min_\theta \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \boldsymbol{\epsilon}_{\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) - \boldsymbol{\epsilon} \right\|^2$: noise prediction

$\boldsymbol{\epsilon}_{\theta,t}(\mathbf{x}_t) =: \left(\mathbf{x}_t - \alpha_t \mathbf{x}_{0\theta,t}(\mathbf{x}_t) \right) / \sigma_t$

➔ $\min_\theta \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \mathbf{x}_{0\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) - \mathbf{x}_0 \right\|^2$: clean-data prediction (denoising)

Balancing Time Steps

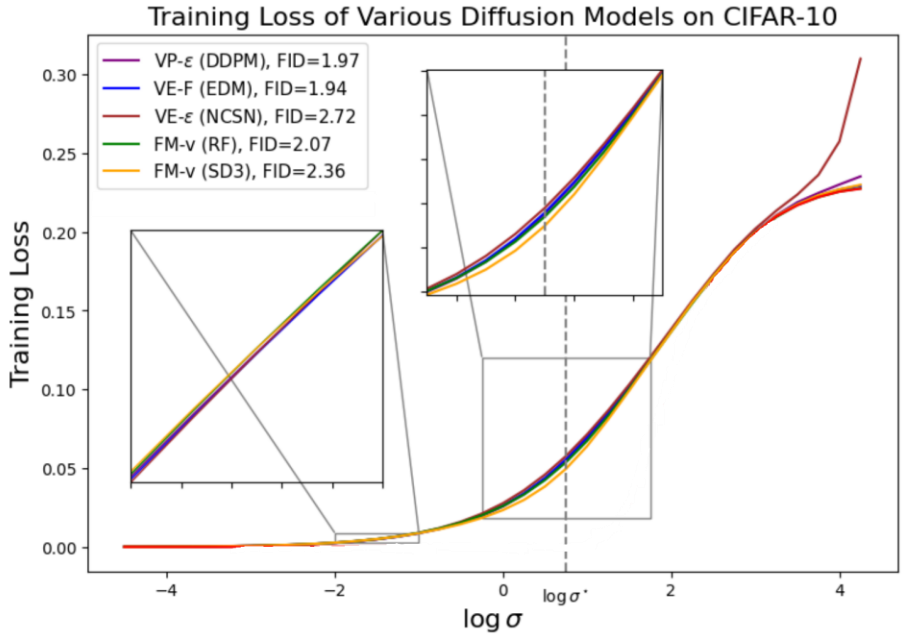
- Training: $\min_\theta \mathbb{E}_{p(t)} w_t \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \mathbf{x}_{0\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) - \mathbf{x}_0 \right\|^2$
  - The loss value does not reflect absolute fit to data!

    $\Leftrightarrow \mathbb{E}_{p_0(\mathbf{x}_0)p(\mathbf{x}_t|\mathbf{x}_0)} \left\| \mathbf{x}_{0\theta,t}(\mathbf{x}_t) - \mathbf{x}_0 \right\|^2$:

    for a given $\mathbf{x}_t$, $\mathbf{x}_{0\theta,t}(\mathbf{x}_t)$ minimizes $\sum_i \left\| \mathbf{x}_{0\theta,t}(\mathbf{x}_t) - \mathbf{x}_0^{(i)} \right\|^2$ where $\mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0|\mathbf{x}_t)$.

    ➔ Optimal $\mathbf{x}^\star_{0\theta,t}(\mathbf{x}_t) = \mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0]$.

    ➔ Optimal loss: $\operatorname{tr} \operatorname{Cov}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0] > 0$ and unknown!
  - Cannot compare data fitness at different time steps.
  - Cannot diagnose training status from training loss.



Balancing Time Steps

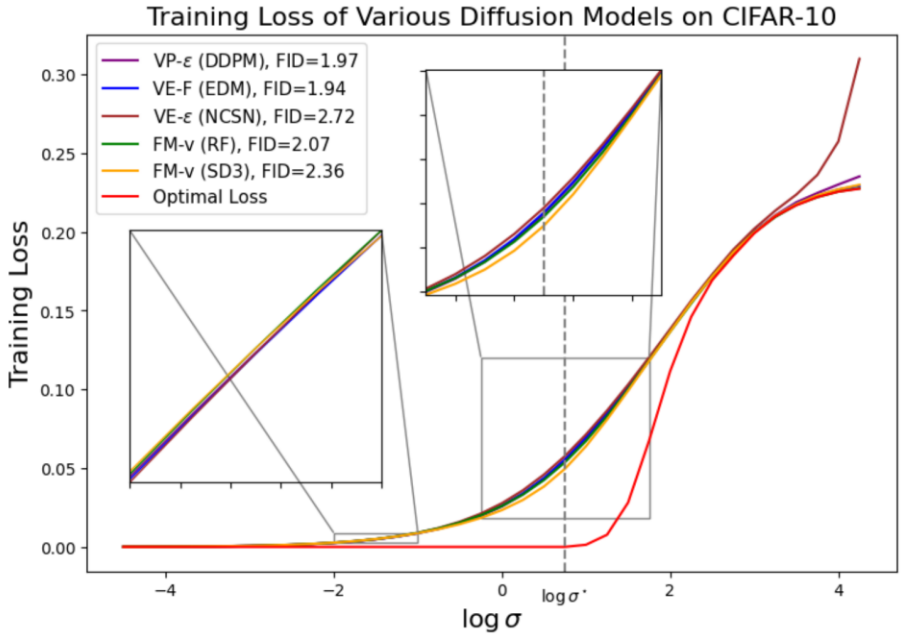
- Training: $\min_\theta \mathbb{E}_{p(t)} w_t \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \mathbf{x}_{0\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) - \mathbf{x}_0 \right\|^2$
  - The loss value does not reflect absolute fit to data!

    $\Leftrightarrow \mathbb{E}_{p_0(\mathbf{x}_0)p(\mathbf{x}_t|\mathbf{x}_0)} \left\| \mathbf{x}_{0\theta,t}(\mathbf{x}_t) - \mathbf{x}_0 \right\|^2$:

    for a given $\mathbf{x}_t$, $\mathbf{x}_{0\theta,t}(\mathbf{x}_t)$ minimizes $\sum_i \left\| \mathbf{x}_{0\theta,t}(\mathbf{x}_t) - \mathbf{x}_0^{(i)} \right\|^2$ where $\mathbf{x}_0^{(i)} \sim p(\mathbf{x}_0|\mathbf{x}_t)$.

    ➔ Optimal $\mathbf{x}^\star_{0\theta,t}(\mathbf{x}_t) = \mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0]$.

    ➔ Optimal loss: $\operatorname{tr} \operatorname{Cov}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0] > 0$ and unknown!
  - Cannot compare data fitness at different time steps.
  - Cannot diagnose training status from training loss.



Estimating the Optimal Loss

$$J_t^{(\mathbf{x}_0)^\star} = \underbrace{\mathbb{E}_{p_0(\mathbf{x}_0)}\|\mathbf{x}_0\|^2}_{=:A} - \underbrace{\mathbb{E}_{p(\mathbf{x}_t)}\left\|\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0]\right\|^2}_{=:B_t} \qquad \hat{A} = \frac{1}{N}\sum\nolimits_{n\in[N]}\|\mathbf{x}_0^{(n)}\|^2$$

- Estimation by importance sampling: $\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0] = \mathbb{E}_{p(\mathbf{x}_0)}\left[\frac{p(\mathbf{x}_0|\mathbf{x}_t)}{p(\mathbf{x}_0)}\mathbf{x}_0\right] \propto \mathbb{E}_{p(\mathbf{x}_0)}[p(\mathbf{x}_t|\mathbf{x}_0)\,\mathbf{x}_0]$

$$\hat{B}_t = \frac{1}{M}\sum_{m\in[M]}\left\|\frac{\sum_{n\in[N]}\mathbf{x}_0^{(n)}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n)})}{\sum_{n'\in[N]}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n')})}\right\|^2 \quad \text{where } K_t(\mathbf{x}_t,\mathbf{x}_0) := \exp\left\{-\frac{\|\mathbf{x}_t-\alpha_t\mathbf{x}_0\|^2}{2\sigma_t^2}\right\}$$

Estimating the Optimal Loss

$$J_t^{(\mathbf{x}_0)^\star} = \underbrace{\mathbb{E}_{p_0(\mathbf{x}_0)}\|\mathbf{x}_0\|^2}_{=:A} - \underbrace{\mathbb{E}_{p(\mathbf{x}_t)}\left\|\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0]\right\|^2}_{=:B_t} \qquad \hat{A} = \frac{1}{N}\sum_{n\in[N]}\|\mathbf{x}_0^{(n)}\|^2$$

- Estimation by importance sampling: $\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0] = \mathbb{E}_{p(\mathbf{x}_0)}\left[\frac{p(\mathbf{x}_0|\mathbf{x}_t)}{p(\mathbf{x}_0)}\mathbf{x}_0\right] \propto \mathbb{E}_{p(\mathbf{x}_0)}[p(\mathbf{x}_t|\mathbf{x}_0)\,\mathbf{x}_0]$

$$\hat{B}_t = \frac{1}{M}\sum_{m\in[M]}\left\|\frac{\sum_{n\in[N]}\mathbf{x}_0^{(n)}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n)})}{\sum_{n'\in[N]}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n')})}\right\|^2 \quad \text{where } K_t(\mathbf{x}_t,\mathbf{x}_0) := \exp\left\{-\frac{\|\mathbf{x}_t-\alpha_t\mathbf{x}_0\|^2}{2\sigma_t^2}\right\}$$

- Scalable estimator: by dataset subsampling
  - $\hat{B}_t^{\mathrm{SNIS}} := \frac{1}{M}\sum_{m\in[M]}\left\|\frac{\sum_{l\in[L]}\mathbf{x}_0^{(l)}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(l)})}{\sum_{l'\in[L]}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(l')})}\right\|^2$ → Large variance from rarely hitting $\mathbf{x}_t^{(m)}$ that makes a large $K_t\left(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(l)}\right)$.
  - $\hat{B}_t^{\mathrm{DOL}} := \frac{1}{M}\sum_{\tilde{m}\in[M]}\left\|\frac{\sum_{l\in[L]}\mathbf{x}_0^{(l)}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l)})}{\sum_{l'\in[L]}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l')})}\right\|^2$ → Use the same set of $\mathbf{x}_0$ to make $\mathbf{x}_t$: Bias from upweighting matching $(\mathbf{x}_0,\mathbf{x}_t)$ pairs.
  - $\hat{B}_t^{\mathrm{cDOL}} := \frac{1}{M}\sum_{\tilde{m}\in[M]}\left\|\frac{\sum_{l\in[L],l\neq l_{\tilde{m}}}\mathbf{x}_0^{(l)}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l)}) + \frac{1}{C}\mathbf{x}_0^{(l_{\tilde{m}})}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l_{\tilde{m}})})}{\sum_{l'\in[L],l'\neq l_{\tilde{m}}}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l')}) + \frac{1}{C}K_t(\mathbf{x}_t^{(\tilde{m})},\mathbf{x}_0^{(l_{\tilde{m}})})}\right\|^2$ Properly balancing variance an bias! ($\mathbf{x}_0^{(l_{\tilde{m}})}$ is the $\mathbf{x}_0$ that makes $\mathbf{x}_t^{(\tilde{m})}$)

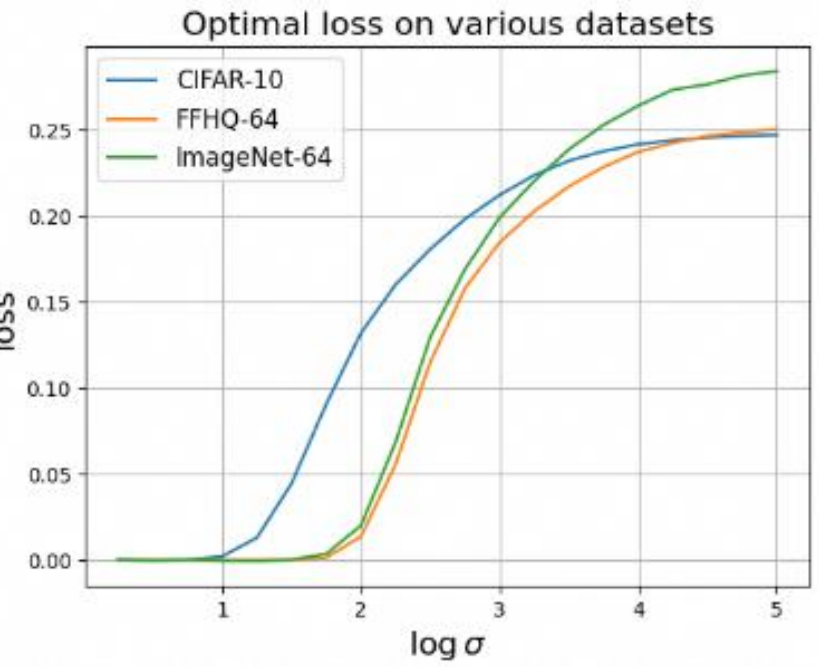
Estimating the Optimal Loss

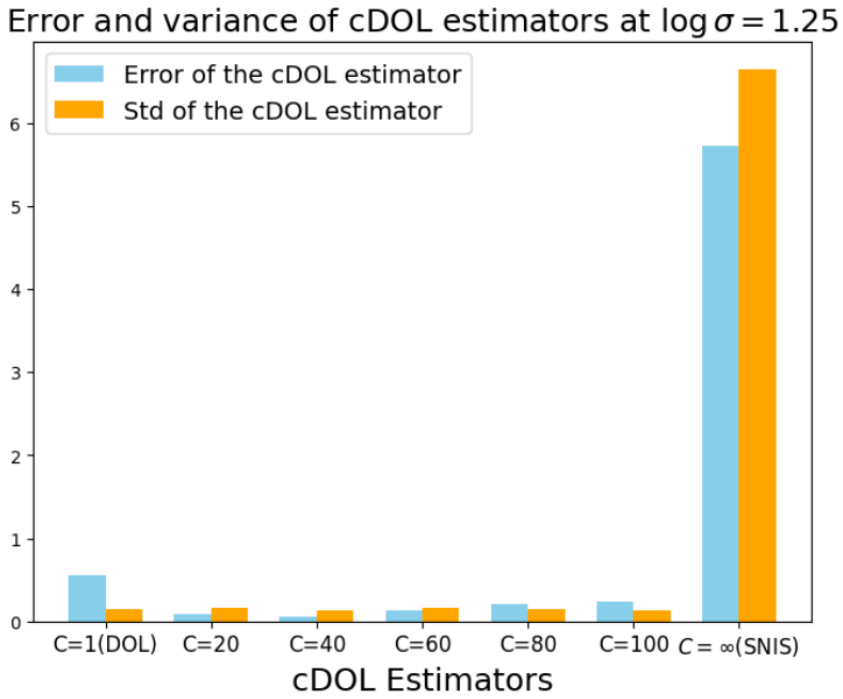
$$J_t^{(\mathbf{x}_0)^\star} = \underbrace{\mathbb{E}_{p_0(\mathbf{x}_0)}\|\mathbf{x}_0\|^2}_{=:A} - \underbrace{\mathbb{E}_{p(\mathbf{x}_t)}\left\|\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0]\right\|^2}_{=:B_t} \qquad \hat{A} = \frac{1}{N}\sum_{n\in[N]}\|\mathbf{x}_0^{(n)}\|^2$$

- Estimation by importance sampling: $\mathbb{E}_{p(\mathbf{x}_0|\mathbf{x}_t)}[\mathbf{x}_0] = \mathbb{E}_{p(\mathbf{x}_0)}\left[\frac{p(\mathbf{x}_0|\mathbf{x}_t)}{p(\mathbf{x}_0)}\mathbf{x}_0\right] \propto \mathbb{E}_{p(\mathbf{x}_0)}[p(\mathbf{x}_t|\mathbf{x}_0)\,\mathbf{x}_0]$

$$\hat{B}_t = \frac{1}{M}\sum_{m\in[M]}\left\|\frac{\sum_{n\in[N]}\mathbf{x}_0^{(n)}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n)})}{\sum_{n'\in[N]}K_t(\mathbf{x}_t^{(m)},\mathbf{x}_0^{(n')})}\right\|^2 \quad \text{where } K_t(\mathbf{x}_t,\mathbf{x}_0) := \exp\left\{-\frac{\|\mathbf{x}_t-\alpha_t\mathbf{x}_0\|^2}{2\sigma_t^2}\right\}$$

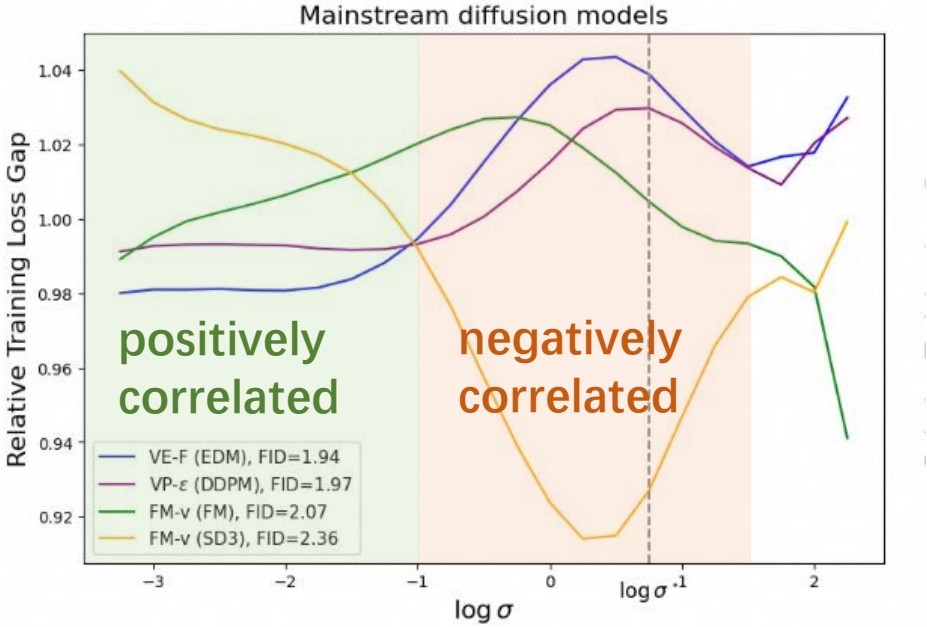
- Scalable es
  - $\hat{B}_t^{\mathrm{SNIS}} :=$ …)).
  - $\hat{B}_t^{\mathrm{DOL}} :=$ … $\mathbf{x}_0, \mathbf{x}_t)$ pairs.
  - $\hat{B}_t^{\mathrm{cDOL}} := \frac{1}{M}$ … variance an bias! … t makes $\mathbf{x}_t^{(\tilde{m})}$)





Principled Diffusion Training and Analysis

- Training: $\min_\theta \mathbb{E}_{p(t)} w_t \mathbb{E}_{p_0(\mathbf{x}_0)p(\boldsymbol{\epsilon})} \left\| \mathbf{x}_{0\theta,t}(\alpha_t \mathbf{x}_0 + \sigma_t \boldsymbol{\epsilon}) - \mathbf{x}_0 \right\|^2$
  - Principled training schedule:



$$w_\sigma = \underbrace{a \ \min\{\tfrac{1}{J^\star_\sigma}, w^\star\}}_{\text{normalize the scale}} \underbrace{+\mathcal{N}(\log\sigma; \mu, \varsigma^2)\mathbb{I}_{\sigma<\sigma^\star}}_{\text{upweight the pos. correl. region}}$$

$$p(\sigma) \propto w_\sigma (J_\sigma(\theta) - J^\star_\sigma)$$

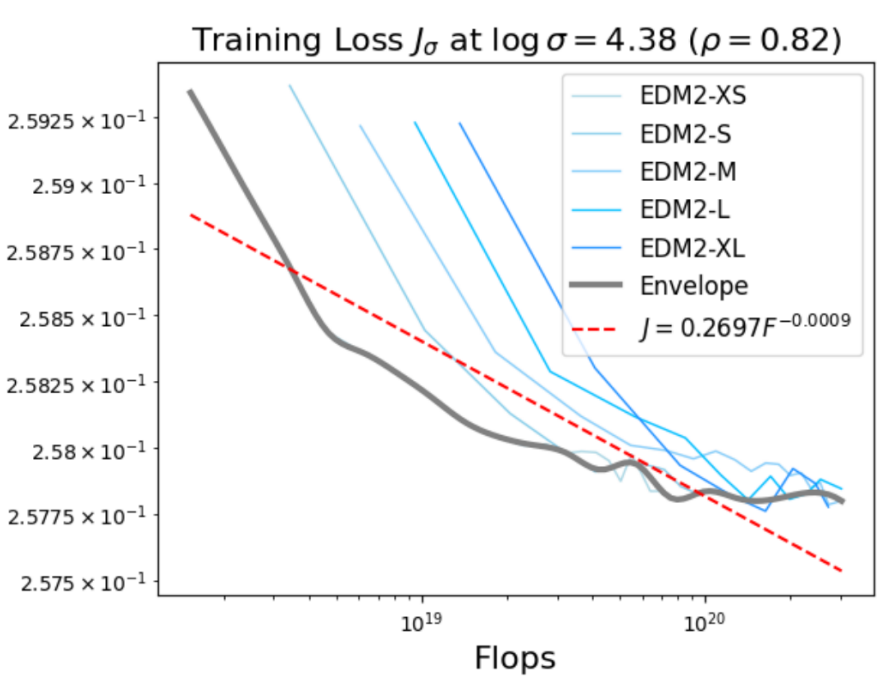
Table 3: Comparison between existing training schedules and ours on ImageNet-256 dataset.

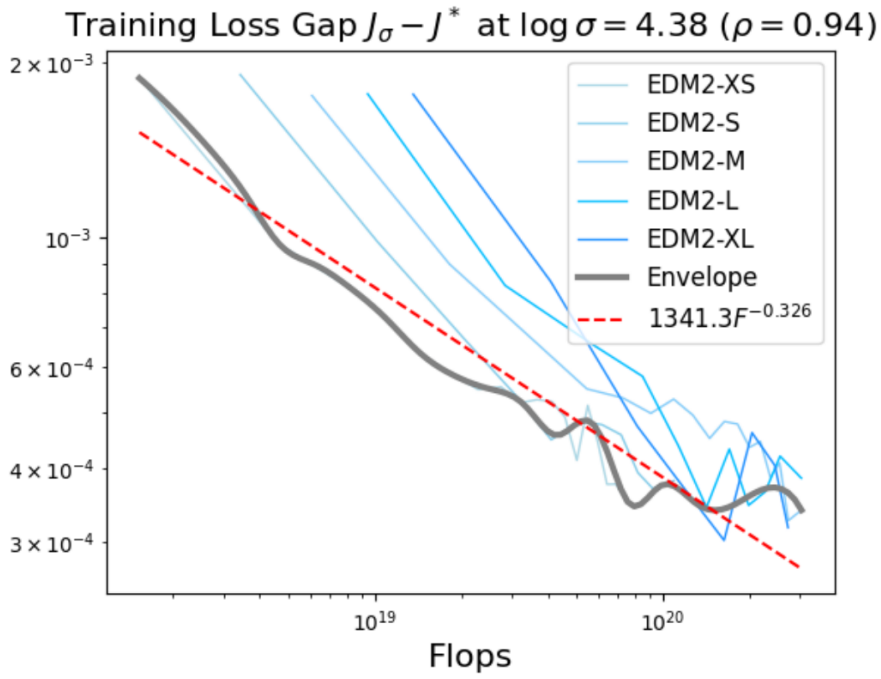
Method	Generation w/o CFG				Generation w/ CFG			
	FID(↓)	IS(↑)	Pre.(↑)	Rec.(↑)	FID(↓)	IS(↑)	Pre.(↑)	Rec.(↑)
Pixel-space Diffusion Models								
ADM (Dhariwal and Nichol, 2021)	10.94	–	0.69	0.63	3.94	215.9	0.83	0.53
RIN (Jabri et al., 2022)	3.42	182.0	–	–	–	–	–	
Simple Diffusion (Hoogeboom et al., 2023)	2.77	211.8	–	–	2.12	256.3	–	–
VDM++ (Kingma and Gao, 2023)	2.40	**225.3**	–	–	2.12	267.7	–	–
SiD2 (Hoogeboom et al., 2024)	–	–	–	–	1.38	–	–	–
Latent Diffusion Models								
MaskDiT (Zheng et al., 2023)	5.69	177.9	0.74	0.60	2.28	276.6	0.80	0.61
DiT (Peebles and Xie, 2023)	9.62	121.5	0.67	0.67	2.27	278.2	**0.83**	0.57
SiT (Ma et al., 2024)	8.61	131.7	0.68	0.67	2.06	270.3	0.82	0.59
FasterDiT (Yao et al., 2024)	7.91	131.3	0.67	**0.69**	2.03	264.0	0.81	0.60
MDT (Gao et al., 2023a)	6.23	143.0	0.71	0.65	1.79	283.0	0.81	0.61
MDTv2 (Gao et al., 2023b)	–	–	–	–	1.58	**314.7**	0.79	0.65
REPA (Yu et al., 2024)	5.90	–	–	–	1.42	305.7	0.80	0.65
LightningDiT (Yao et al., 2025)	2.17	205.6	**0.77**	0.65	1.35	295.3	0.79	0.65
+ reproduction	2.29	206.2	0.76	0.66	1.42	292.9	0.79	0.65
+ our schedule	**2.08**	220.8	**0.77**	0.66	**1.30**	301.3	0.79	**0.66**

Principled Diffusion Training and Analysis

- Scaling-law analysis:

$$J(F) = \beta F^{\alpha} \quad \Rightarrow \quad J(F) - J^{\star} = \beta F^{\alpha}$$





Thanks!

Check out our paper