Mean Flows for One-step Generative Modeling

Zhengyang Geng[1]* Mingyang Deng[2] Xingjian Bai[2] J. Zico Kolter[1] Kaiming He[2]

[1]CMU [2]MIT

arxiv 250519

2025.6.30 Minghao Liu

目录

CONTENTS

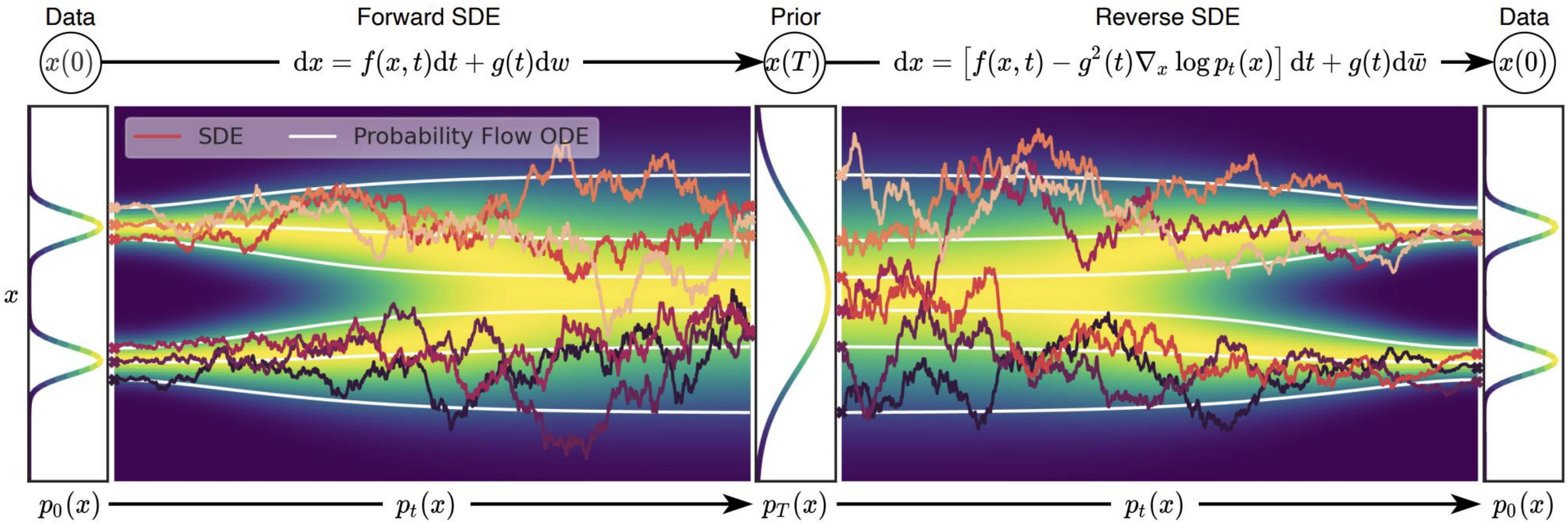
北京大學
PEKING UNIVERSITY

PART 02

Background

Background

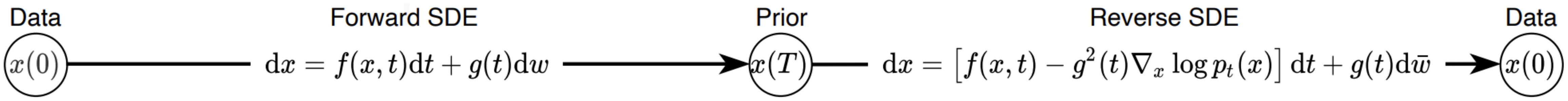
Score Matching



Song Y, Sohl-Dickstein J, Kingma D P, et al. Score-based generative modeling through stochastic differential equations. ICLR, 2021.

Background

Score Matching



Training: $\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\arg\min}\, \mathbb{E}_t\Big\{\lambda(t)\mathbb{E}_{\mathbf{x}(0)}\mathbb{E}_{\mathbf{x}(t)\mid\mathbf{x}(0)}\Big[\big\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}(t),t) - \nabla_{\mathbf{x}(t)}\log p_{0t}(\mathbf{x}(t)\mid\mathbf{x}(0))\big\|_2^2\Big]\Big\}.$

Sampling:

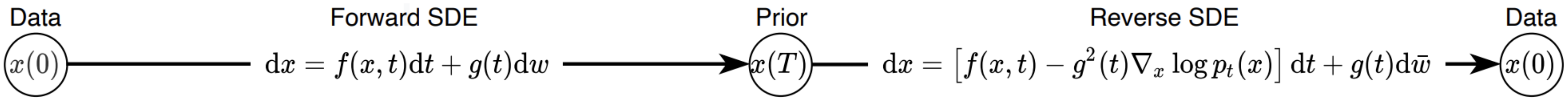
1. Predictor
2. Predictor-Corrector
3. SDE -> ODE $\quad \mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x},t) - \frac{1}{2}g(t)^2\nabla_{\mathbf{x}}\log p_t(\mathbf{x})\right]\mathrm{d}t,$

Song Y, Sohl-Dickstein J, Kingma D P, et al. Score-based generative modeling through stochastic differential equations. ICLR, 2021.

Background

北京大学 PEKING UNIVERSITY

Score Matching



Variance Preserving, VP: $\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\mathbf{w}.$

Sub-VP: $\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\,\mathrm{d}t + \sqrt{\beta(t)(1 - e^{-2\int_0^t \beta(s)\mathrm{d}s})}\mathrm{d}\mathbf{w}.$

Song Y, Sohl-Dickstein J, Kingma D P, et al. Score-based generative modeling through stochastic differential equations. ICLR, 2021.

Background

PEKING UNIVERSITY 北京大学

Flow Matching

Objective: $\min_\theta \mathbb{E}_{t\sim\mathcal{U}[0,1],x_t\sim p_t}\|v_\theta(x_t,t)-u(x_t,t)\|^2$ (hard to compute)

$\Updownarrow$

Objective: $\min_\theta \mathbb{E}_{t\sim\mathcal{U}[0,1],x_0\sim p_0,x_1\sim p_1,\epsilon\sim\mathcal{N}(0,I)}\|v_\theta(x_t,t)-\frac{x_1-x_0}{1}\|^2$ (easy to compute)

where $x_t=(1-t)x_0+tx_1+\sigma(t)\epsilon$.

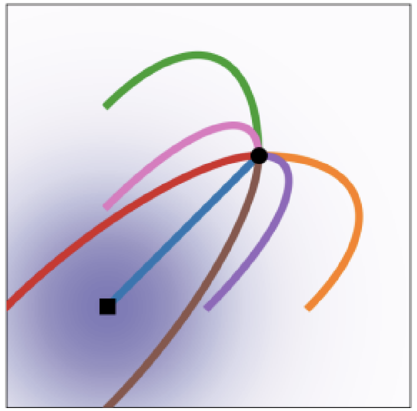
Simulation-Free: No need to solve ODE/SDE.

Lipman Y, Chen R T Q, Ben-Hamu H, et al. Flow matching for generative modeling. ICLR, 2023.

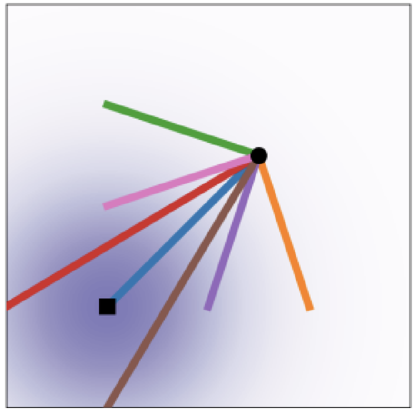
Background

Flow Matching

Ideal:

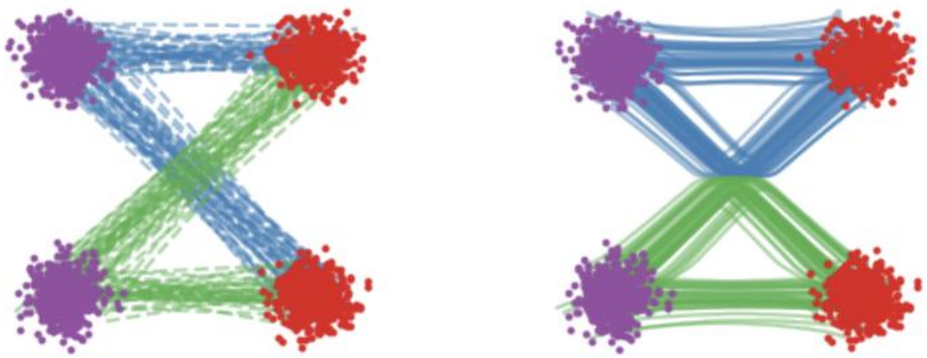


Diffusion



OT

Reality:



(a) Linear interpolation $X_t = tX_1 + (1-t)X_0$

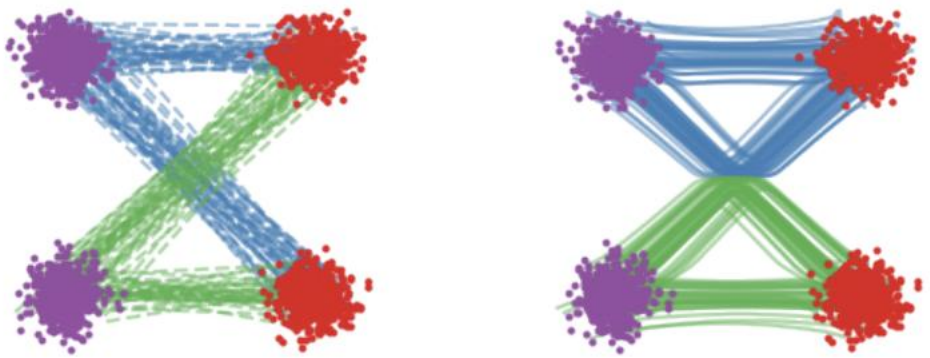
(b) Rectified flow Z_t induced by (X_0, X_1)

Lipman Y, Chen R T Q, Ben-Hamu H, et al. Flow matching for generative modeling. ICLR, 2023.

Background

Rectified flow

$$\min_{v} \int_0^1 \mathbb{E}\left[\left\|(X_1 - X_0) - v(X_t,\, t)\right\|^2\right] \mathrm{d}t, \quad \text{with} \quad X_t = tX_1 + (1-t)X_0,$$



(a) Linear interpolation $X_t = tX_1 + (1-t)X_0$

(b) Rectified flow Z_t induced by (X_0, X_1)

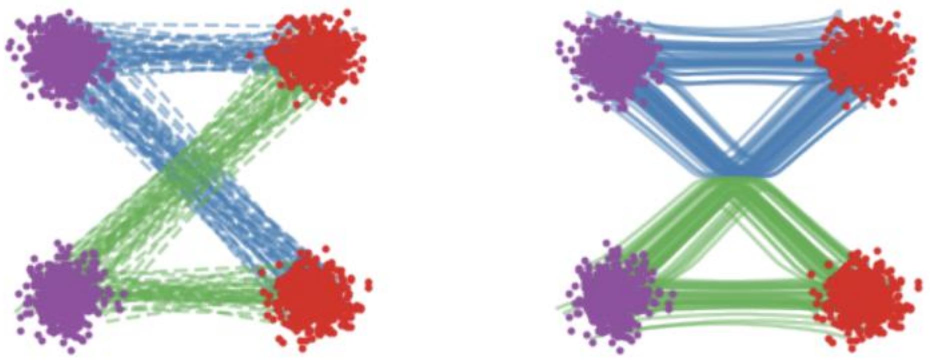
Step 1

Liu X, Gong C, Liu Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR, 2023.

Background

Rectified flow

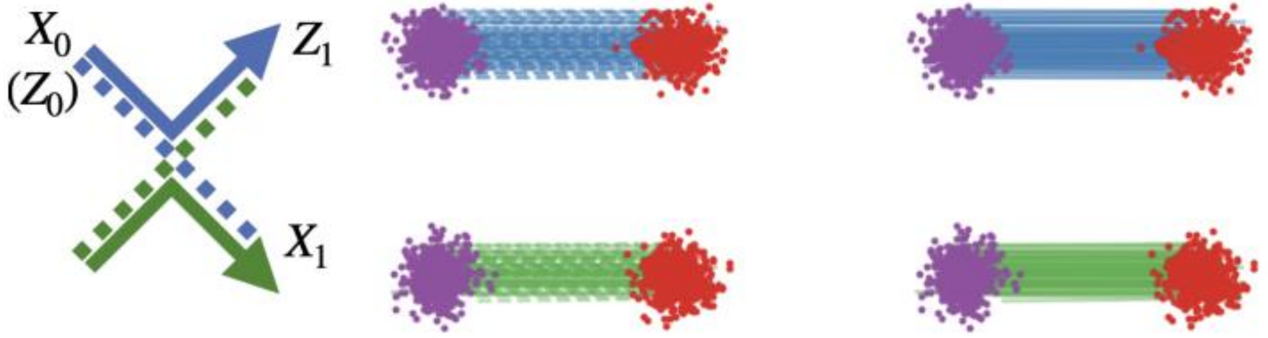
$$\min_v \int_0^1 \mathbb{E}\left[\left\|(X_1 - X_0) - v(X_t,\, t)\right\|^2\right] \mathrm{d}t, \quad \text{with} \quad X_t = tX_1 + (1-t)X_0,$$



(a) Linear interpolation $X_t = tX_1 + (1-t)X_0$

(b) Rectified flow Z_t induced by (X_0, X_1)

Step 1



(c) Linear interpolation $Z_t = tZ_1 + (1-t)Z_0$

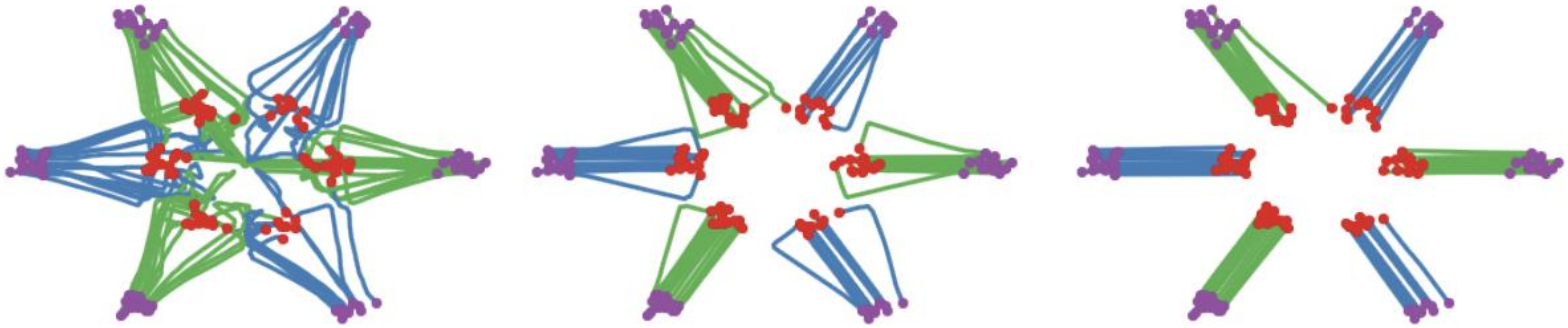
(d) Rectified flow Z'_t induced by (Z_0, Z_1)

Step 2

Liu X, Gong C, Liu Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR, 2023.

Background

Rectified flow



(a)The 1st rectified flow $\boldsymbol{Z}^1$

$\boldsymbol{Z}^1 = \texttt{RectFlow}((X_0, X_1))$

(b) Reflow $\boldsymbol{Z}^2$

$\boldsymbol{Z}^2 = \texttt{RectFlow}((Z_0^1, Z_1^1))$

(c) Reflow $\boldsymbol{Z}^3$

$\boldsymbol{Z}^3 = \texttt{RectFlow}((Z_0^2, Z_1^2))$

Liu X, Gong C, Liu Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR, 2023.

Background

Rectified flow

Distillation: $\mathbb{E}\left[\left\|(Z_1^k - Z_0^k) - v(Z_0^k, 0)\right\|^2\right]$ One-step (CIFAR-10 FID 4.85)

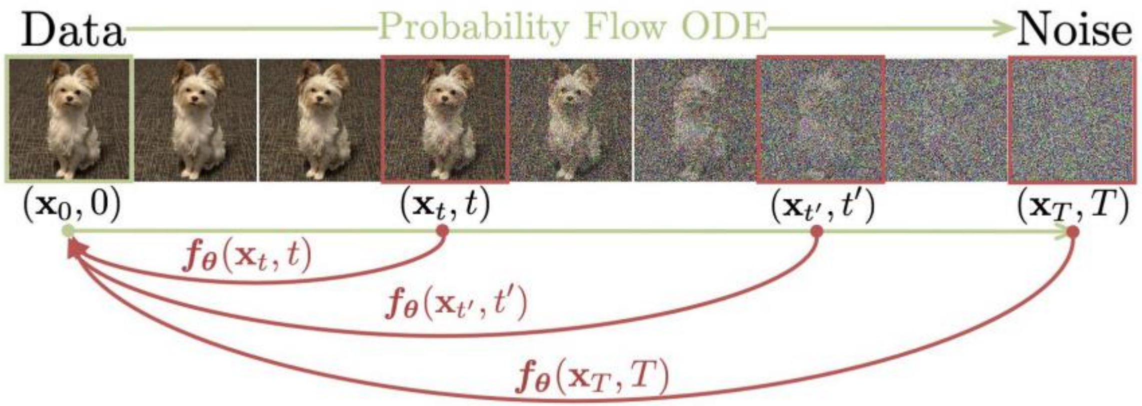
Step 3

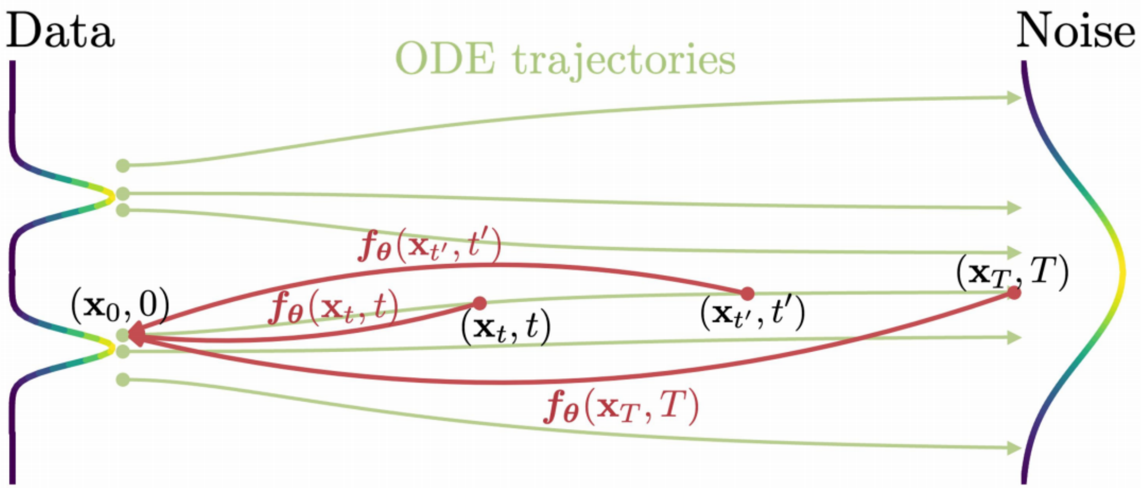
Liu X, Gong C, Liu Q. Flow straight and fast: Learning to generate and transfer data with rectified flow. ICLR, 2023.

Background

Consistency Models

Models of these mappings are called consistency models, as their outputs are trained to be consistent for points on the same trajectory.





Song Y, Dhariwal P, Chen M, et al. Consistency models. ICML, 2023.

Background

Consistency Models

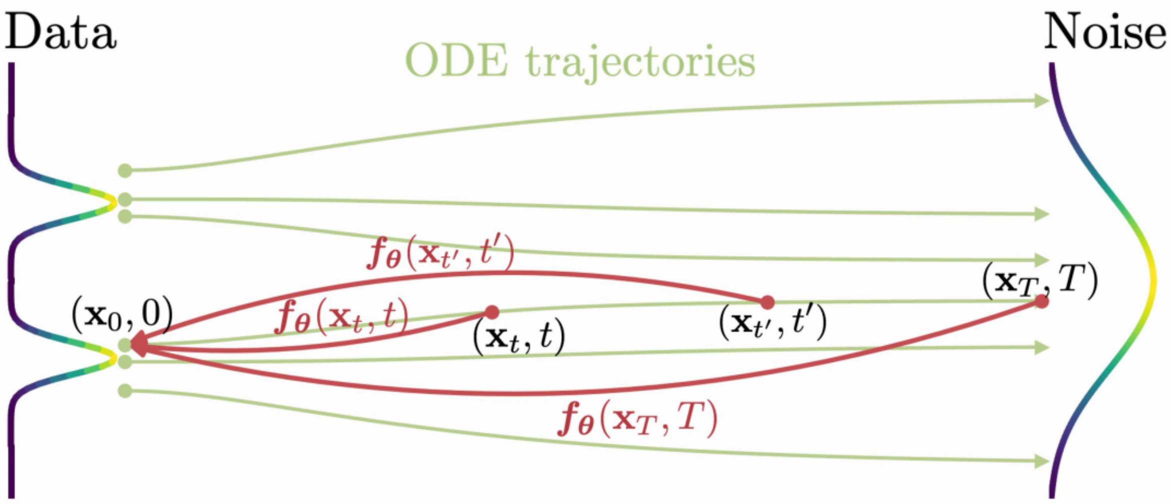
Boundary condition: $\boldsymbol{f}(\mathbf{x}_\epsilon, \epsilon) = \mathbf{x}_\epsilon$, $\epsilon = 0.002$

Two ways to implement boundary condition:

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}, t) = \begin{cases} \mathbf{x} & t = \epsilon \\ F_{\boldsymbol{\theta}}(\mathbf{x}, t) & t \in (\epsilon, T] \end{cases}$$

$$\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}, t) = c_{\text{skip}}(t)\mathbf{x} + c_{\text{out}}(t) F_{\boldsymbol{\theta}}(\mathbf{x}, t)$$

$$c_{\text{skip}}(\epsilon) = 1, \text{ and } c_{\text{out}}(\epsilon) = 0.$$



Song Y, Dhariwal P, Chen M, et al. Consistency models. ICML, 2023.

Background

Consistency Models

Training Consistency Models via Distillation

Algorithm 2 Consistency Distillation (CD)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, ODE solver $\Phi(\cdot, \cdot; \boldsymbol{\phi})$, $d(\cdot, \cdot)$, $\lambda(\cdot)$, and μ
$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$
repeat
  Sample $\mathbf{x} \sim \mathcal{D}$ and $n \sim \mathcal{U}[\![1, N-1]\!]$
  Sample $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 \boldsymbol{I})$
  $\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}} \leftarrow \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{\phi})$
  $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi}) \leftarrow$
    $\lambda(t_n) d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n))$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi})$
  $\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu \boldsymbol{\theta}^- + (1-\mu)\boldsymbol{\theta})$
until convergence

Distilling a pre-trained score model $\boldsymbol{s}_{\boldsymbol{\phi}}(\mathbf{x}, t)$

Song Y, Dhariwal P, Chen M, et al. Consistency models. ICML, 2023.

Background

PEKING UNIVERSITY 北京大学

Consistency Models

Training Consistency Models via Distillation

Algorithm 2 Consistency Distillation (CD)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, ODE solver $\Phi(\cdot, \cdot; \boldsymbol{\phi})$, $d(\cdot, \cdot)$, $\lambda(\cdot)$, and μ

$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$

repeat

Sample $\mathbf{x} \sim \mathcal{D}$ and $n \sim \mathcal{U}[\![1, N-1]\!]$

Sample $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}; t_{n+1}^2 \boldsymbol{I})$

$\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}} \leftarrow \mathbf{x}_{t_{n+1}} + (t_n - t_{n+1})\Phi(\mathbf{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{\phi})$

$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi}) \leftarrow$

$\quad \lambda(t_n) d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n))$

$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi})$

$\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu \boldsymbol{\theta}^- + (1-\mu)\boldsymbol{\theta})$

until convergence

One-step (CIFAR-10 FID 3.55)

Two-step (CIFAR-10 FID 2.93)

Limited by the quality of pre-trained diffusion models.

Song Y, Dhariwal P, Chen M, et al. Consistency models. ICML, 2023.

Consistency Models

Training Consistency Models in Isolation

Algorithm 3 Consistency Training (CT)

Input: dataset $\mathcal{D}$, initial model parameter $\boldsymbol{\theta}$, learning rate η, step schedule $N(\cdot)$, EMA decay rate schedule $\mu(\cdot)$, $d(\cdot,\cdot)$, and $\lambda(\cdot)$

$\boldsymbol{\theta}^- \leftarrow \boldsymbol{\theta}$ and $k \leftarrow 0$

repeat

  Sample $\mathbf{x} \sim \mathcal{D}$, and $n \sim \mathcal{U}[\![1, N(k)-1]\!]$

  Sample $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$

  $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-) \leftarrow$

    $\lambda(t_n) d(\boldsymbol{f}_{\boldsymbol{\theta}}(\mathbf{x} + t_{n+1}\mathbf{z}, t_{n+1}), \boldsymbol{f}_{\boldsymbol{\theta}^-}(\mathbf{x} + t_n\mathbf{z}, t_n))$

  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\theta}^-)$

  $\boldsymbol{\theta}^- \leftarrow \text{stopgrad}(\mu(k)\boldsymbol{\theta}^- + (1-\mu(k))\boldsymbol{\theta})$

  $k \leftarrow k + 1$

until convergence

Isolation.

One-step (CIFAR-10 FID 8.70)

Two-step (CIFAR-10 FID 5.83)

Rely on LPIPS

Song Y, Dhariwal P, Chen M, et al. Consistency models. ICML, 2023.

Imporved Consistency Models

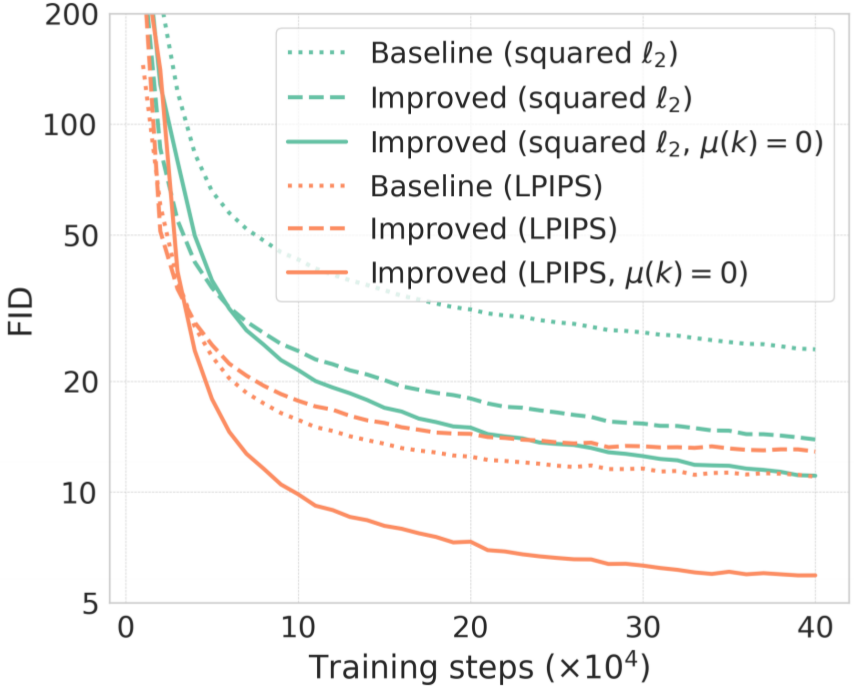
	Design choice in Song et al. (2023)	Our modifications
EMA decay rate for the teacher network	$\mu(k) = \exp(\frac{s_0 \log \mu_0}{N(k)})$	$\mu(k) = 0$
Metric in consistency loss	$d(\boldsymbol{x}, \boldsymbol{y}) = \text{LPIPS}(\boldsymbol{x}, \boldsymbol{y})$	$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\lVert \boldsymbol{x} - \boldsymbol{y} \rVert_2^2 + c^2} - c$
Discretization curriculum	$N(k) = \left\lceil \sqrt{\frac{k}{K}((s_1+1)^2 - s_0^2) + s_0^2} - 1 \right\rceil + 1$	$N(k) = \min(s_0 2^{\lfloor \frac{k}{K'} \rfloor}, s_1) + 1$, where $K' = \left\lfloor \frac{K}{\log_2 \lfloor s_1/s_0 \rfloor + 1} \right\rfloor$
Noise schedule	σ_i, where $i \sim \mathcal{U}[\![1, N(k)-1]\!]$	σ_i, where $i \sim p(i)$, and $p(i) \propto \text{erf}\left(\frac{\log(\sigma_{i+1}) - P_{\text{mean}}}{\sqrt{2} P_{\text{std}}}\right) - \text{erf}\left(\frac{\log(\sigma_i) - P_{\text{mean}}}{\sqrt{2} P_{\text{std}}}\right)$
Weighting function	$\lambda(\sigma_i) = 1$	$\lambda(\sigma_i) = \frac{1}{\sigma_{i+1} - \sigma_i}$
Parameters	$s_0 = 2, s_1 = 150, \mu_0 = 0.9$ on CIFAR-10 $s_0 = 2, s_1 = 200, \mu_0 = 0.95$ on ImageNet 64×64	$s_0 = 10, s_1 = 1280$ $c = 0.00054\sqrt{d}$, d is data dimensionality $P_{\text{mean}} = -1.1, P_{\text{std}} = 2.0$
	$k \in [\![0, K]\!]$, where K is the total training iterations $\sigma_i = (\sigma_{\min}^{1/\rho} + \frac{i-1}{N(k)-1}(\sigma_{\max}^{1/\rho} - \sigma_{\min}^{1/\rho}))^\rho$, where $i \in [\![1, N(k)]\!], \rho = 7, \sigma_{\min} = 0.002, \sigma_{\max} = 80$	

Song Y, Dhariwal P. Improved techniques for training consistency models. ICLR, 2024.

北京大学 PEKING UNIVERSITY

Imporved Consistency Models

	Design choice in Song et al. (2023)	Our modifications
EMA decay rate for the teacher network	$\mu(k) = \exp(\frac{s_0 \log \mu_0}{N(k)})$	$\mu(k) = 0$
Metric in consistency loss	$d(\boldsymbol{x}, \boldsymbol{y})$	$x - \boldsymbol{y}\|_2^2 + c^2 - c$
Discretization curriculum	$N(k) =$ $\lceil\sqrt{}$	$_0 2^{\lfloor \frac{k}{K'} \rfloor}, s_1) + 1$, where $K' = \left\lfloor \frac{K}{\log_2\lfloor s_1/s_0 \rfloor + 1} \right\rfloor$
Noise schedule	σ_i, wher	(i), and $p(i) \propto$ $-P_{\text{mean}}$ $) - \text{erf}\left(\frac{\log(\sigma_i) - P_{\text{mean}}}{\sqrt{2}P_{\text{std}}}\right)$
Weighting function	$\lambda(\sigma_i) =$	σ_i
Parameters	$s_0 = 2,$	1280
	$s_0 = 2,$	l, d is data dimensionality
		$P_{\text{std}} = 2.0$
		ons
	$\sigma_i = ($	$\sigma_{\min} = 0.002, \sigma_{\max} = 80$



Song Y, Dhariwal P. Improved techniques for training consistency models. ICLR, 2024.

北京大学 PEKING UNIVERSITY

Imporved Consistency Models

	Design choice in Song et al. (2023)	Our modifications
EMA decay rate for the teacher network	$\mu(k) = \exp(\frac{s_0 \log \mu_0}{N(k)})$	$\mu(k) = 0$
Metric in consistency loss	$d(\boldsymbol{x}, \boldsymbol{y}) = \text{LPIPS}(\boldsymbol{x}, \boldsymbol{y})$	$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\lVert \boldsymbol{x} - \boldsymbol{y} \rVert_2^2 + c^2} - c$
Discretization curriculum	$N(k) = \left\lceil \sqrt{\frac{k}{K}((s_1+1)^2 - s_0^2) + s_0^2} - 1 \right\rceil + 1$	$N(k) = \min(s_0 2^{\lfloor \frac{k}{K'} \rfloor}, s_1) + 1$, where $K' = \left\lfloor \frac{K}{\log_2 \lfloor s_1/s_0 \rfloor + 1} \right\rfloor$
Noise schedule	σ_i, where $i \sim \mathcal{U}[\![1, N(k)-1]\!]$	σ_i, where $i \sim p(i)$, and $p(i) \propto \text{erf}\left(\frac{\log(\sigma_{i+1}) - P_{\text{mean}}}{\sqrt{2} P_{\text{std}}}\right) - \text{erf}\left(\frac{\log(\sigma_i) - P_{\text{mean}}}{\sqrt{2} P_{\text{std}}}\right)$
Weighting function	$\lambda(\sigma_i) = 1$	$\lambda(\sigma_i) = \frac{1}{\sigma_{i+1} - \sigma_i}$
Parameters	$s_0 = 2, s_1 = 150, \mu_0 = 0.9$ on CIFAR-10 $s_0 = 2, s_1 = 200, \mu_0 = 0.95$ on ImageNet 64×64	$s_0 = 10, s_1 = 1280$ $c = 0.00054\sqrt{d}$, d is data dimensionality $P_{\text{mean}} = -1.1, P_{\text{std}} = 2.0$
	$k \in [\![0, K]\!]$, where K is the total training iterations $\sigma_i = (\sigma_{\min}^{1/\rho} + \frac{i-1}{N(k)-1}(\sigma_{\max}^{1/\rho} - \sigma_{\min}^{1/\rho}))^\rho$, where $i \in [\![1, N(k)]\!]$, $\rho = 7, \sigma_{\min} = 0.002, \sigma_{\max} = 80$	

Song Y, Dhariwal P. Improved techniques for training consistency models. ICLR, 2024.

Background

Imporved Consistency Models

METHOD	NFE (↓)	FID (↓)	IS (↑)
Fast samplers & distillation for diffusion models			
DDIM (Song et al., 2020)	10	13.36	
DPM-solver-fast (Lu et al., 2022)	10	4.70	
3-DEIS (Zhang & Chen, 2022)	10	4.17	
UniPC (Zhao et al., 2023)	10	3.87	
Knowledge Distillation (Luhman & Luhman, 2021)	1	9.36	
DFNO (LPIPS) (Zheng et al., 2022)	1	3.78	
2-Rectified Flow (+distill) (Liu et al., 2022)	1	4.85	9.01
TRACT (Berthelot et al., 2023)	1	3.78	
	2	3.32	
Diff-Instruct (Luo et al., 2023)	1	4.53	9.89
PD* (Salimans & Ho, 2022)	1	8.34	8.69
	2	5.58	9.05
CD (LPIPS) (Song et al., 2023)	1	3.55	9.48
	2	2.93	9.75
Direct Generation			
Score SDE (Song et al., 2021)	2000	2.38	9.83
Score SDE (deep) (Song et al., 2021)	2000	2.20	9.89
DDPM (Ho et al., 2020)	1000	3.17	9.46
LSGM (Vahdat et al., 2021)	147	2.10	
PFGM (Xu et al., 2022)	110	2.35	9.68
EDM* (Karras et al., 2022)	35	2.04	9.84
EDM-G++ (Kim et al., 2023)	35	1.77	
IGEBM (Du & Mordatch, 2019)	60	40.6	6.02
NVAE (Vahdat & Kautz, 2020)	1	23.5	7.18
Glow (Kingma & Dhariwal, 2018)	1	48.9	3.92
Residual Flow (Chen et al., 2019)	1	46.4	
BigGAN (Brock et al., 2019)	1	14.7	9.22
StyleGAN2 (Karras et al., 2020b)	1	8.32	9.21
StyleGAN2-ADA (Karras et al., 2020a)	1	2.92	9.83
CT (LPIPS) (Song et al., 2023)	1	8.70	8.49
	2	5.83	8.85
iCT (ours)	1	2.83	9.54
	2	2.46	9.80
iCT-deep (ours)	1	2.51	9.76
	2	2.24	9.89

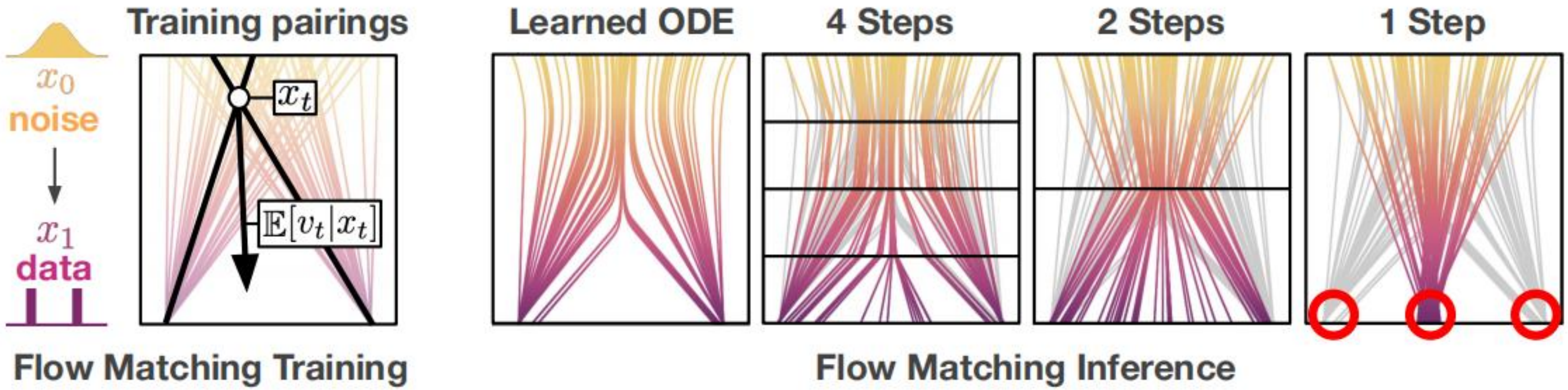
Song Y, Dhariwal P. Improved techniques for training consistency models. ICLR, 2024.

Background

Shortcut Models

Reflow: Multi-stage, Few-step ambiguity problem

Consistency Model: Too many tricks, Hard to train



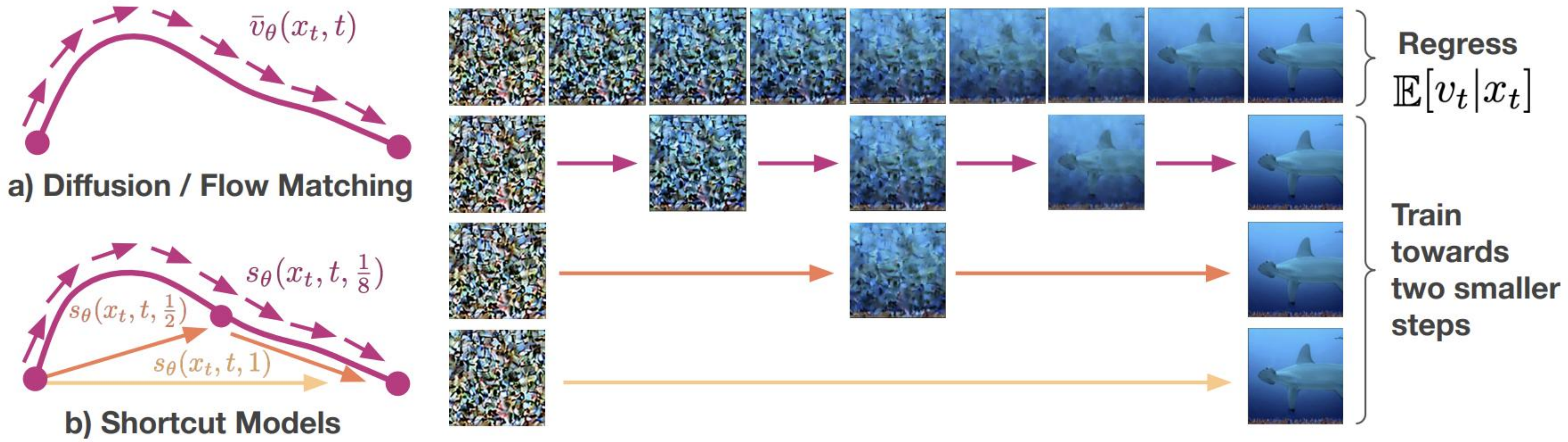
Frans K, Hafner D, Levine S, et al. One step diffusion via shortcut models[J]. ICLR, 2025.

Background

Shortcut Models

Conditioning the model not only on the timestep *t* but also on a desired step size *d*.

$$x'_{t+d} = x_t + s(x_t, t, d)\, d.$$



Frans K, Hafner D, Levine S, et al. One step diffusion via shortcut models. ICLR, 2025.

Background

Shortcut Models

Conditioning the model not only on the timestep *t* but also on a desired step size *d*.

Split the batch into two fraction

d=0: the shortcut is equivalent to the flow.

d>0: one shortcut step equals two consecutive shortcut steps of half the size

$$\mathcal{L}^{\mathrm{S}}(\theta) = E_{x_0 \sim \mathcal{N},\, x_1 \sim D,\, (t,d) \sim p(t,d)} \Big[\underbrace{\| s_\theta(x_t, t, 0) - (x_1 - x_0) \|^2}_{\text{Flow-Matching}} + \underbrace{\| s_\theta(x_t, t, 2d) - s_{\text{target}} \|^2}_{\text{Self-Consistency}} \Big],$$

$$\text{where} \quad s_{\text{target}} = s_\theta(x_t, t, d)/2 + s_\theta(x'_{t+d}, t, d)/2 \quad \text{and} \quad x'_{t+d} = x_t + s_\theta(x_t, t, d)d.$$

Frans K, Hafner D, Levine S, et al. One step diffusion via shortcut models. ICLR, 2025.

Background

Shortcut Models

Batch Mixing: 75% d=0 & 25% d>0

Weight Decay: 0.1

EMA version of s_t and s_{t+d}

No results on CIFAR-10 and ImageNet-64

Frans K, Hafner D, Levine S, et al. One step diffusion via shortcut models. ICLR, 2025.

北京大學
PEKING UNIVERSITY

PART 03

Method

Method

Mean Flows

Flow Matching & Rectified Flow: Real Flow trajectories are always curved.

Integral difficulty: Neural networks are difficult to accurately learn a complex integral operator, for one-step generation.

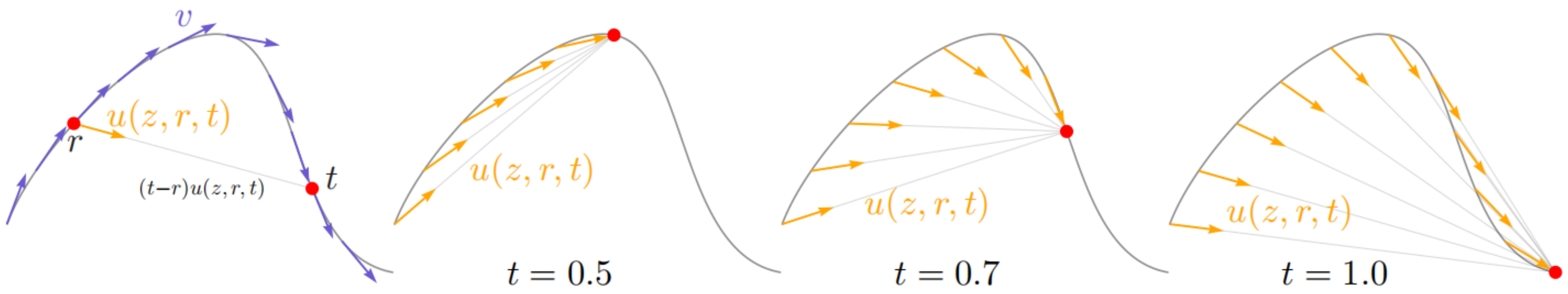
Consistency Model: No solid theoretical foundation and training may be unstable.

Geng Z, Deng M, Bai X, et al. Mean Flows for One-step Generative Modeling. Submit to NeurIPS, 2025.

PEKING UNIVERSITY 北京大学

Mean Flows

Average Velocity Field: $$u(z_t, r, t) \triangleq \frac{1}{t-r}\int_r^t v(z_\tau, \tau)d\tau.$$



Relation between u and v: $$\underbrace{u(z_t, r, t)}_{\text{average vel.}} = \underbrace{v(z_t, t)}_{\text{instant. vel.}} - (t-r)\underbrace{\frac{d}{dt}u(z_t, r, t)}_{\text{time derivative}}$$ (MeanFlow Identity)

$$\frac{d}{dt}u(z_t, r, t) = \frac{dz_t}{dt}\partial_z u + \frac{dr}{dt}\partial_r u + \frac{dt}{dt}\partial_t u. \implies \frac{d}{dt}u(z_t, r, t) = v(z_t, t)\partial_z u + \partial_t u,$$

Geng Z, Deng M, Bai X, et al. Mean Flows for One-step Generative Modeling. Submit to NeurIPS, 2025.

Method

Mean Flows

Training with Average Velocity:

$$\mathcal{L}(\theta) = \mathbb{E}\left\| u_\theta(z_t, r, t) - \mathrm{sg}(u_{\text{tgt}}) \right\|_2^2,$$

where $$u_{\text{tgt}} = v(z_t, t) - (t - r)\left(v(z_t, t)\partial_z u_\theta + \partial_t u_\theta\right),$$

Algorithm 1 MeanFlow: Training.

Note: in PyTorch and JAX, jvp returns the function output and JVP.

```
# fn(z, r, t): function to predict u
# x: training batch

t, r = sample_t_r()
e = randn_like(x)

z = (1 - t) * x + t * e
v = e - x

u, dudt = jvp(fn, (z, r, t), (v, 0, 1))

u_tgt = v - (t - r) * dudt
error = u - stopgrad(u_tgt)

loss = metric(error)
```

Algorithm 2 MeanFlow: 1-step Sampling

```
e = randn(x_shape)
x = e - fn(e, r=0, t=1)
```

Geng Z, Deng M, Bai X, et al. Mean Flows for One-step Generative Modeling. Submit to NeurIPS, 2025.

Method

Mean Flows

Mean Flows with Guidance:

New Ground-truth Field: $v^{\text{cfg}}(z_t, t \mid \mathbf{c}) \triangleq \omega\, v(z_t, t \mid \mathbf{c}) + (1-\omega)\, v(z_t, t),$

New Average Velocity: $u^{\text{cfg}}(z_t, r, t \mid \mathbf{c}) = v^{\text{cfg}}(z_t, t \mid \mathbf{c}) - (t-r)\frac{d}{dt}u^{\text{cfg}}(z_t, r, t \mid \mathbf{c}).$

Training with Guidance:

$$\mathcal{L}(\theta) = \mathbb{E}\left\| u_\theta^{\text{cfg}}(z_t, r, t \mid \mathbf{c}) - \text{sg}(u_{\text{tgt}}) \right\|_2^2,$$

$$\text{where} \quad u_{\text{tgt}} = \tilde{v}_t - (t-r)\left(\tilde{v}_t \partial_z u_\theta^{\text{cfg}} + \partial_t u_\theta^{\text{cfg}}\right).$$

Geng Z, Deng M, Bai X, et al. Mean Flows for One-step Generative Modeling. Submit to NeurIPS, 2025.

北京大學
PEKING UNIVERSITY

PART 04

Experiments

Experiments

北京大学 PEKING UNIVERSITY

- ImageNet generation at 256×256 resolution (A latent space of 32×32×4)，1-NFE, On the latent space of a pre-trained VAE tokenizer.
- Ablation:

% of $r\neq t$	FID, 1-NFE
0% (= FM)	328.91
25%	**61.06**
50%	63.14
100%	67.32

(a) **Ratio of sampling** $r\neq t$. The 0% entry reduces to the standard Flow Matching baseline.

jvp tangent	FID, 1-NFE
$(v, 0, 1)$	**61.06**
$(v, 0, 0)$	268.06
$(v, 1, 0)$	329.22
$(v, 1, 1)$	137.96

(b) **JVP computation.** The correct jvp tangent is $(v, 0, 1)$ for Jacobian $(\partial_z u, \partial_r u, \partial_t u)$.

pos. embed	FID, 1-NFE
(t, r)	61.75
$(t, t-r)$	**61.06**
$(t, r, t-r)$	63.98
$t-r$ only	63.13

(c) **Positional embedding.** The network is conditioned on the embeddings applied to the specified variables.

t, r sampler	FID, 1-NFE
uniform(0, 1)	65.90
lognorm(–0.2, 1.0)	63.83
lognorm(–0.2, 1.2)	64.72
lognorm(–0.4, 1.0)	**61.06**
lognorm(–0.4, 1.2)	61.79

(d) **Time samplers.** t and r are sampled from the specific sampler.

p	FID, 1-NFE
0.0	79.75
0.5	63.98
1.0	**61.06**
1.5	66.57
2.0	69.19

(e) **Loss metrics.** p=0 is squared L2 loss. p=0.5 is Pseudo-Huber loss.

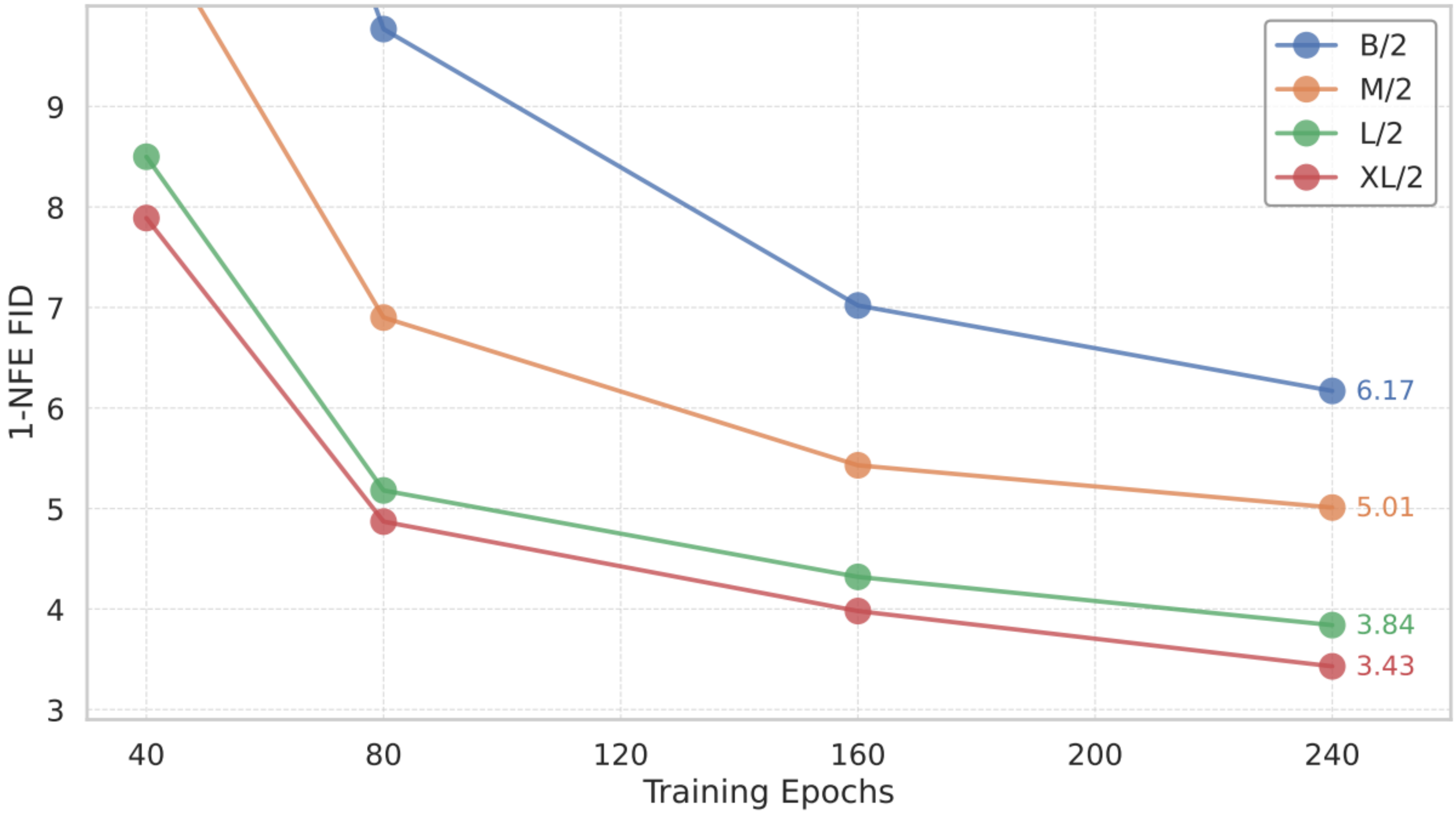
$$w = 1/(\|\Delta\|_2^2 + c)^p,$$

ω	FID, 1-NFE
1.0 (w/o cfg)	61.06
1.5	33.33
2.0	20.15
3.0	**15.53**
5.0	20.75

(f) **CFG scale:**. Our method supports 1-NFE CFG sampling.

Experiments

- **Scalability:**



Experiments

- **Comparisons with Prior Work:**

method	params	NFE	FID
1-NFE diffusion/flow from scratch			
iCT-XL/2 [43]†	675M	1	34.24
Shortcut-XL/2 [13]	675M	1	10.60
MeanFlow-B/2	131M	1	6.17
MeanFlow-M/2	308M	1	5.01
MeanFlow-L/2	459M	1	3.84
MeanFlow-XL/2	676M	1	**3.43**
2-NFE diffusion/flow from scratch			
iCT-XL/2 [43]†	675M	2	20.30
iMM-XL/2 [52]	675M	1×2	7.77
MeanFlow-XL/2	676M	2	2.93
MeanFlow-XL/2+	676M	2	**2.20**

method	params	NFE	FID
GANs			
BigGAN [5]	112M	1	6.95
GigaGAN [21]	569M	1	3.45
StyleGAN-XL [40]	166M	1	2.30
autoregressive/masking			
AR w/ VQGAN [10]	227M	1024	26.52
MaskGIT [6]	227M	8	6.18
VAR-*d*30 [47]	2B	10×2	1.92
MAR-H [27]	943M	256×2	1.55
diffusion/flow			
ADM [8]	554M	250×2	10.94
LDM-4-G [37]	400M	250×2	3.60
SimDiff [20]	2B	512×2	2.77
DiT-XL/2 [34]	675M	250×2	2.27
SiT-XL/2 [33]	675M	250×2	2.06
SiT-XL/2+REPA [51]	675M	250×2	**1.42**

Experiments

- **Comparisons with Prior Work:**

method	precond	NFE	FID
iCT [43]	EDM	1	**2.83**
ECT [15]	EDM	1	3.60
sCT [31]	EDM	1	2.97
IMM [52]	EDM	1	3.20
MeanFlow	none	1	2.92

Table 3: **Unconditional CIFAR-10.**

Experiments

- 1-NFE Generation Results

Conclusions

- **Proposes modeling average velocity instead of instantaneous velocity for generative flows, enabling direct prediction of endpoint displacement in one step.**

- **Establishes a mathematical link between average and instantaneous velocities via a differential relation, eliminating need for integral computations during training.**

- **Demonstrates 3.43 FID on ImageNet 256×256 with true 1-NFE sampling, narrowing the gap between one-step and multi-step diffusion models.**

Thanks for Listening！