SCORE-BASED GENERATIVE MODEL

黄政宇
北京大学北京国际数学研究中心
北京大学国际机器学习研究中心

Applications

- Image, video, and sound generation

Generative Model

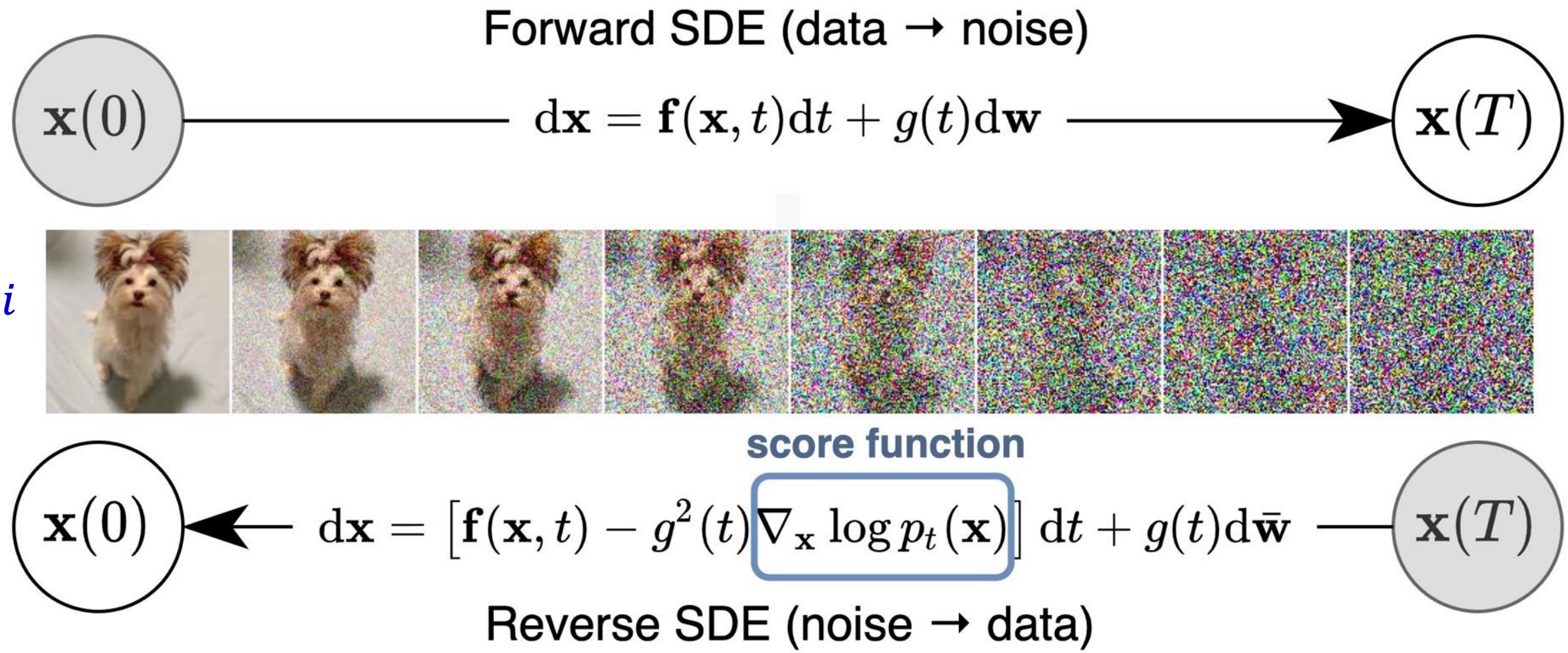
- Target distribution

$$q_0(x) \approx \frac{1}{N}\sum_i \delta(x - x^{*i})$$

Goal: sample $x \sim q_0(x)$

Score-based diffusion model

x^{*i}



Outline

- Diffusion process
  - Ornstein-Uhlenbeck(OU) process
- Score matching
  - U-Net: score representation
- Theoretical analysis
  - denoising diffusion probabilistic model
  - probability flow ODE
- Numerical study
  - Gaussian mixture model
  - MINST

Diffusion Process

- Forward equation ($0 \to T$)：

  stochastic differential equation(SDE):

  $$dX_t = f(t, X_t)dt + g(t)dW_t$$

  density: $X_t \sim q_t(x)$

  Fokker Planck equation:

  $$\partial_t q_t(x) = -\nabla \cdot (f(t,x)q_t) + \frac{1}{2}\nabla \cdot \nabla \cdot (g(t)^2 q_t)$$

Diffusion Process

- Backward equation($T \to 0$):

$$-\partial_t q_{T-t}(x) = -\nabla \cdot (f(T-t,x)q_{T-t}) + \frac{1}{2}\nabla \cdot \nabla \cdot (g(T-t)^2 q_{T-t})$$

Define: $\rho_t(x) = q_{T-t}(x)$ $(0 \to T)$

$$\partial_t \rho_t(x) = \nabla \cdot (f(T-t,x)\rho_t) - \frac{1}{2}\nabla \cdot \nabla \cdot (g(T-t)^2 \rho_t)$$

Generating: $Y_0 \sim \rho_0(x) = q_T(x)$ $(0 \to T)$

$$dY_t = \hat{f}(t, Y_t)dt + \hat{g}(t)dw_t$$

Diffusion Process

- Forward equation $(0 \rightarrow T)$:

  stochastic differential equation(SDE):

  $$dX_t = f(t)X_t dt + g(t)dW_t$$

  distribution: $X_t \sim \lambda_t X_0 + \sigma_t W \quad W \sim \mathcal{N}(0, I)$

  $$f(t) = \frac{d \log \lambda_t}{dt} \qquad g(t)^2 = \frac{d\sigma_t^2}{dt} - 2\frac{d \log \lambda_t}{dt}\sigma_t^2$$

  consider: $L(t, x) = \frac{x}{\lambda_t} \qquad dL(t, x_t)$

Diffusion Process

- Ornstein-Uhlenbeck(OU) process $(0 \rightarrow T)$:

$$dX_t = -\frac{1}{2}\beta(t)X_t dt + \sqrt{\beta(t)}dW_t$$
$$dX_t = -X_t dt + \sqrt{2}dW_t$$

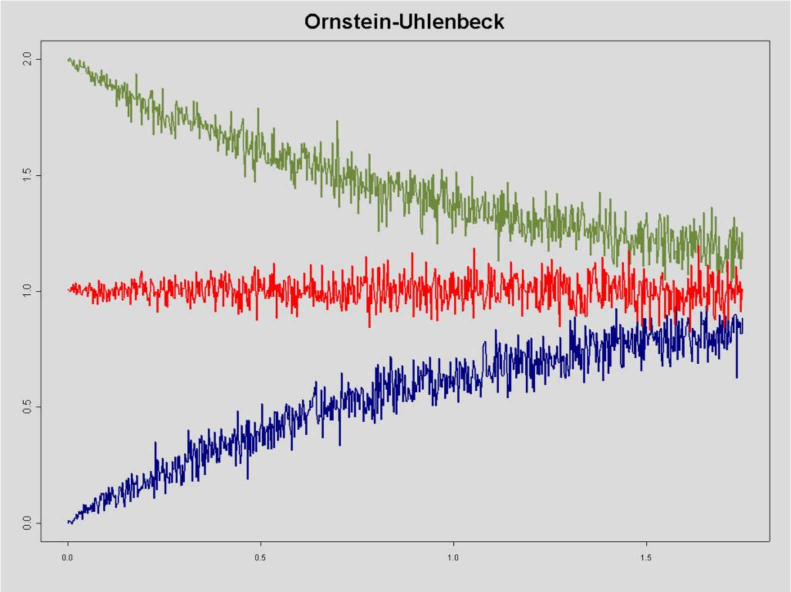
- Variance exploding SDE dynamics $(0 \rightarrow T)$:

$$dX_t = \sigma(t)dW_t$$
$$dX_t = dW_t$$

OU Process

- Forward equation $(0 \to T)$:

SDE: $dX_t = -X_t dt + \sqrt{2} dW_t$



$$X_t = e^{-t} X_0 + \sqrt{1 - e^{-2t}} W$$

$$X_t \to \mathcal{N}(0, I)$$

Density: $X_t \sim q_t(x)$, Fokker Planck equation:

$$\partial_t q_t(x) = \nabla \cdot \big((x + \nabla \log q_t(x)) \, q_t(x) \big)$$

OU Process

- Backward equation $(T \to 0)$:

$$-\partial_t q_{T-t}(x) = \nabla \cdot (x q_{T-t}) + \nabla \cdot \nabla \cdot (q_{T-t})$$

define: $\rho_t(x) = q_{T-t}(x) \qquad (0 \to T)$

$$\partial_t \rho_t(x) = -\nabla \cdot ((x + \nabla \log q_{T-t}(x))\, \rho_t)$$

generating: $Y_0 \sim \mathcal{N}(0, I) \approx q_T(x) \qquad (0 \to T)$

- denoising diffusion probabilistic model :

$$dY_t = (Y_t + 2\nabla \log q_{T-t}(Y_t))dt + \sqrt{2}dW_t$$

- probability flow ODE :

$$\frac{dY_t}{dt} = Y_t + \nabla \log q_{T-t}(Y_t)$$

Score-Based Generative Model

- Score-based generative model

  Step 1: creating noise from data

  $$X_t = e^{-t}X_0 + \sqrt{1-e^{-2t}}W \qquad w \sim \mathcal{N}(0, I)$$

  Step 2: estimate score: $s_\theta(t, x) \approx \nabla \log q_{T-t}(x)$

  Step 3: generating: $\hat{Y}_0 \sim q_T(x) \approx \mathcal{N}(0, I)$

  - denoising diffusion probabilistic model :

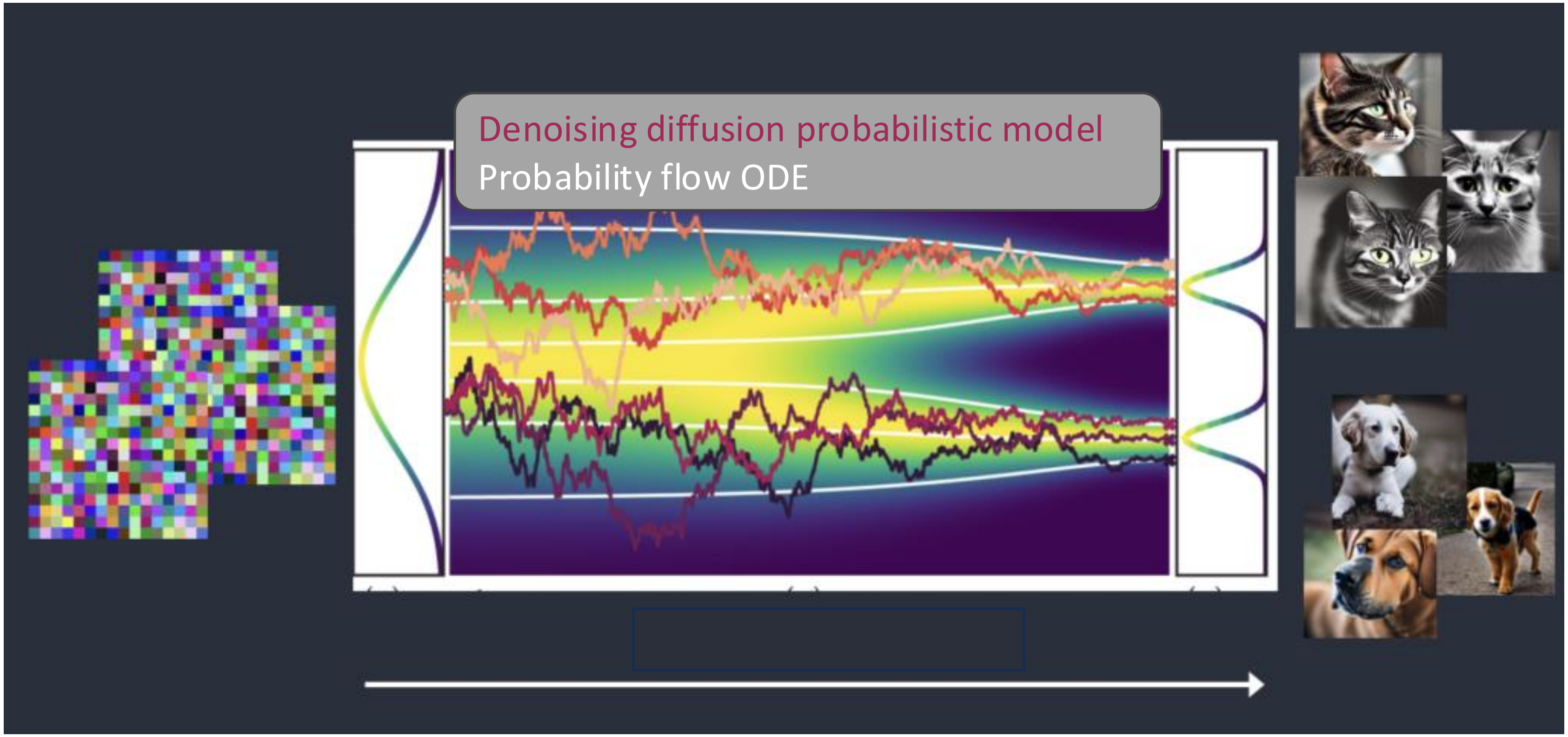
  $$d\hat{Y}_t = (\hat{Y}_t + 2s_\theta(t, \hat{Y}_t))dt + \sqrt{2}dW_t$$

  - probability flow ODE :

  $$\frac{d\hat{Y}_t}{dt} = \hat{Y}_t + s_\theta(t, \hat{Y}_t)$$

Score-Based Generative Model

- Backward equation $(0 \rightarrow T)$ ：



Score Matching

➢ Objective function

$$\min_{\theta} \int_0^T w_{T-t} \mathbb{E}_{q_{T-t}} \parallel s_\theta(t,x) - \nabla \log q_{T-t}(x) \parallel^2 dt$$

Density: $X_t \sim q_t(x)$

$$X_t = e^{-t} X_0 + \sqrt{1 - e^{-2t}} W \qquad W \sim \mathcal{N}(0, I_d)$$

$$q_t(x|x_0) = \mathcal{N}(x; e^{-t} x_0, (1 - e^{-2t}) I)$$

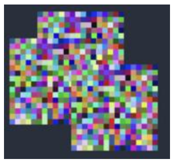
$$q_t(x) = \int \mathcal{N}(x; e^{-t} x_0, (1 - e^{-2t}) I) q_0(x_0) dx_0$$

Implementable:

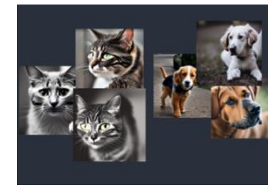
$$\min_{\theta} \int_0^T w_{T-t} \mathbb{E}_{q_{0(x_0)}} \mathbb{E}_{q_{T-t(x|x_0)}} \parallel s_\theta(t,x) - \nabla \log q_{\mathrm{T}-t}(x|x_0) \parallel^2 dt$$

Score Matching

- Time discretization



$$0 = t_0 < t_1 < \cdots t_N = T - \tau$$



τ: avoid singularity.

Exercise : $q_0 = \delta(x)$, what is $\nabla \log q_{\mathrm{T}-t}(x)$?

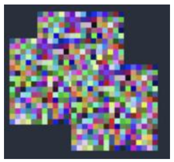
$$q_t(x) = \int \mathcal{N}(x; e^{-t}x_0, (1 - e^{-2t})I)q_0(x_0)dx_0$$

$$= \mathcal{N}(x; 0, (1 - e^{-2t})I)$$

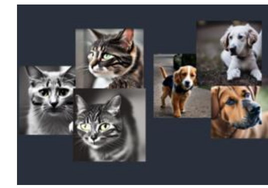
$\mathrm{s}(t, x) = \nabla \log q_{T-t}(x) = -\frac{x}{1-e^{-2(T-t)}} \approx \mathcal{O}\left(\frac{1}{T-t}\right)$, when $t \to T$

Score Matching

- Time discretization



$$0 = t_0 < t_1 < \cdots t_N = T - \tau$$



$$\min_{\theta} \int_0^{T-\tau} w_{T-t} \mathbb{E}_{q_{0(x_0)}} \mathbb{E}_{q_{T-t}(x|x_0)} \parallel s_\theta(t,x) - \nabla \log q_{\mathrm{T}-t}(x|x_0) \parallel^2 dt$$

Randomly sample t for the training

Score Matching

- Estimate $s(x,t) \approx \nabla \log q_{T-t}$ from data

$$\min_{\theta} \int_{\tau}^{T} w_t \mathbb{E}_{q_{0(x_0)}} \mathbb{E}_{q_t(x|x_0)} \parallel s_\theta(x, T-t) - \nabla \log q_t(x|x_0) \parallel^2 dt$$

- weight choice

$$q_t(x|x_0) = \mathcal{N}(x; e^{-t}x_0, (1-e^{-2t})I)$$

$$\nabla \log q_t(x|x_0) = -\frac{x - e^{-t}x_0}{1-e^{-2t}}$$

$$w_t \propto \frac{1}{\mathbb{E}_{q_t(x|x_0)} \|\nabla \log q_t(x|x_0)\|^2} \propto 1 - e^{-2t}$$

- rescaling

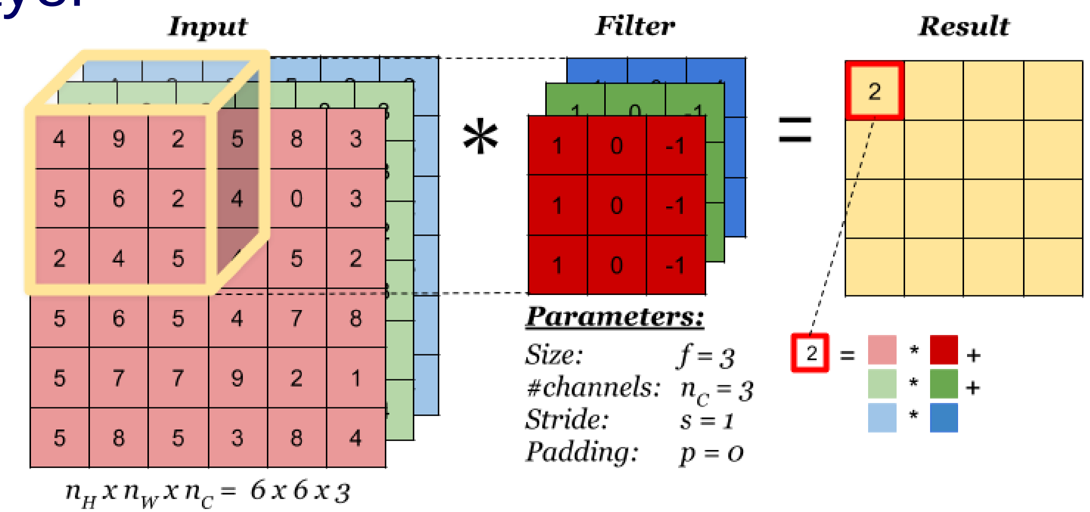
$$s_\theta(x, T-t) := \epsilon_\theta(x, T-t)/\sqrt{1-e^{-2t}}$$

- time embedding

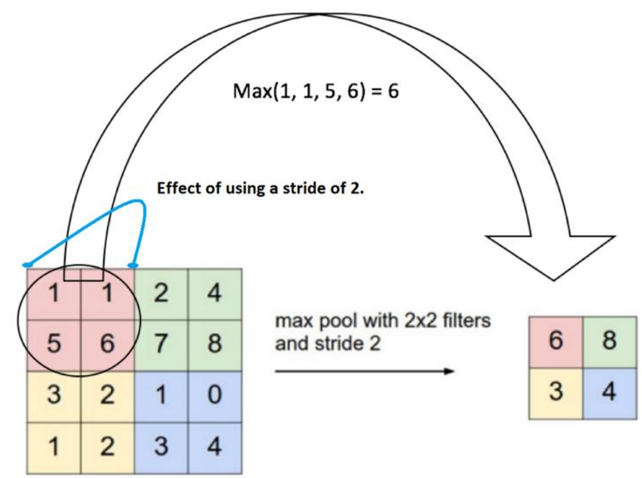
input: x, [sin($2\pi\omega t$); cos($2\pi\omega t$)]

U-Net: Score Representation $\epsilon_\theta(x, t)$

- Convolution layer



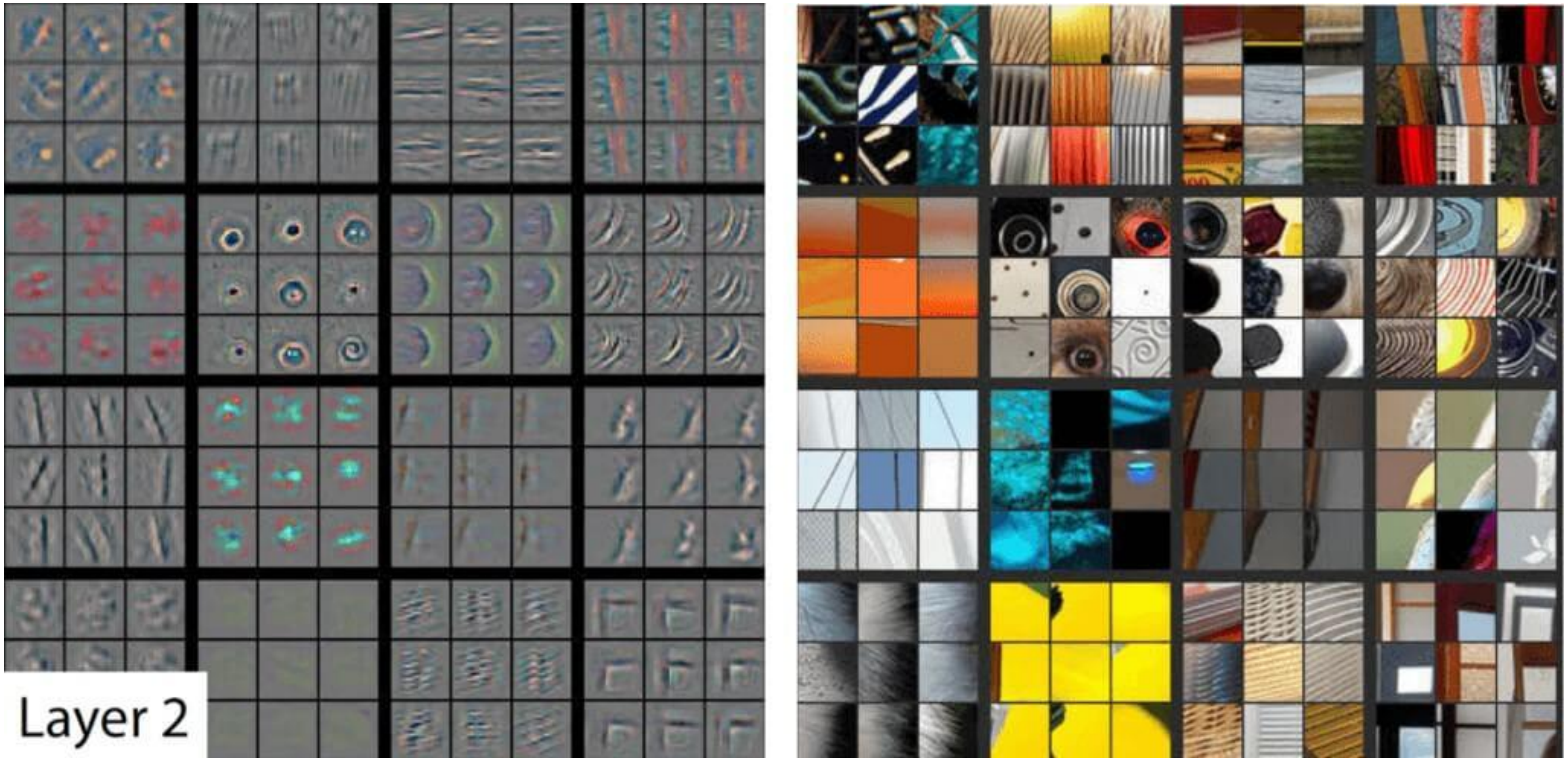
- Pooling layer



We do the pooling with 2x2 filters, so we will devide an input image on 2x2 regions, and we will use a stride of 2.
Because we are using a stride of 2, these regions don't overlap.

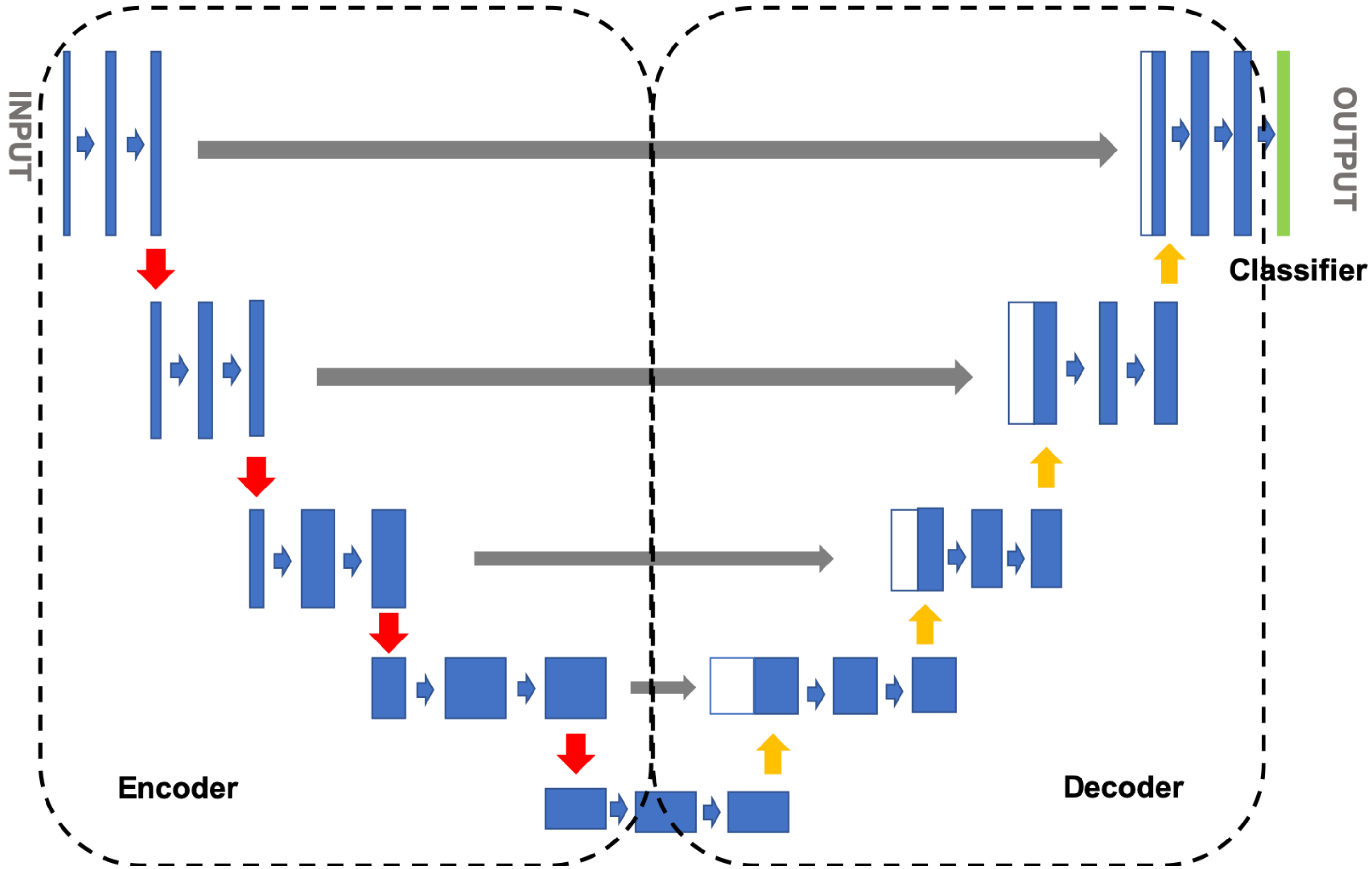
U-Net: Score Representation $\epsilon_\theta(x, t)$

- Convolutional neural network



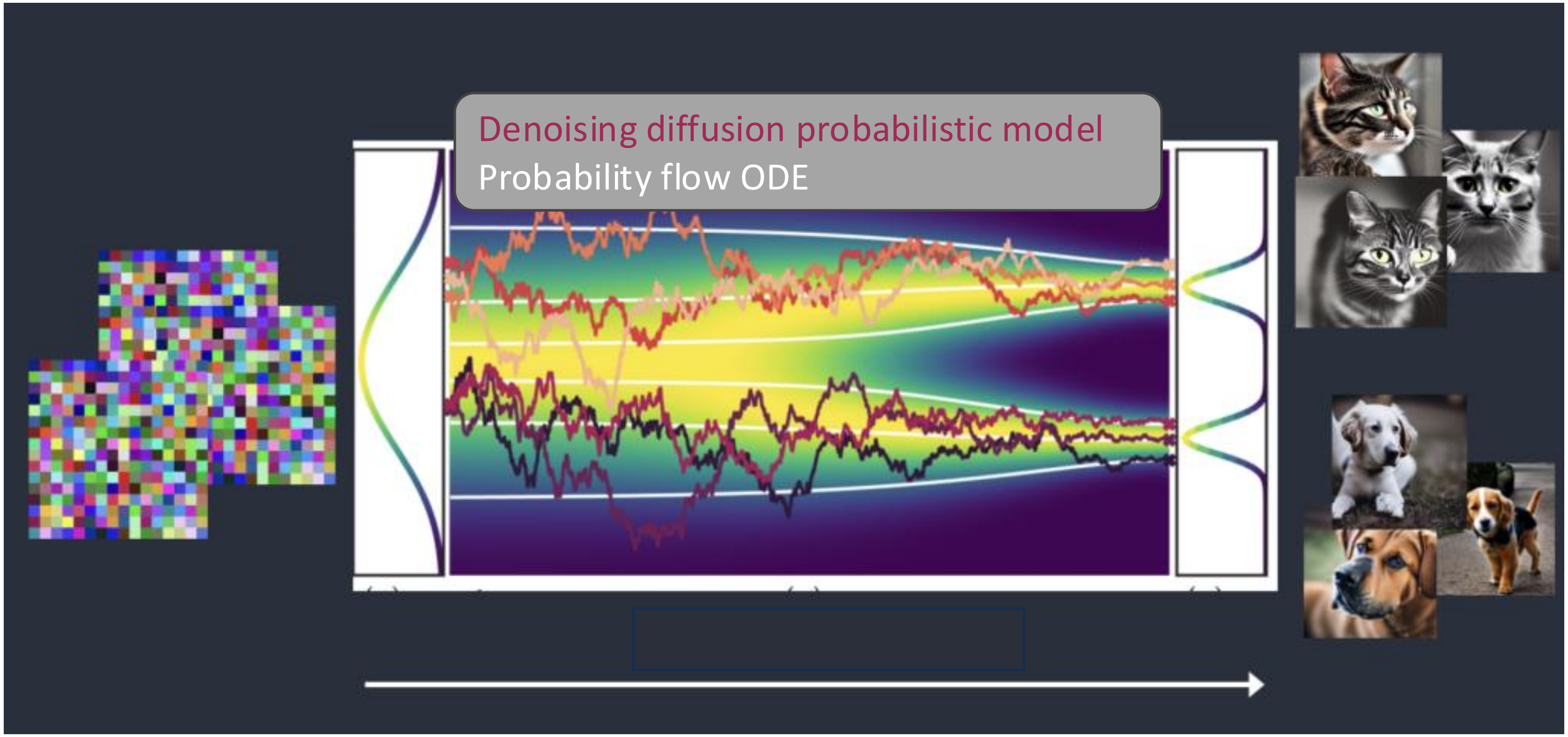
U-Net: Score Representation $\epsilon_\theta(x, t)$

- U-Net architecture



Score-Based Generative Model

- Backward equation $(0 \to T)$：



$$0 = t_0 < t_1 < \cdots t_N = T - \tau$$

Denoising Diffusion Probabilistic Model

- Denoising diffusion probabilistic model

$$d\hat{Y}_t = (\hat{Y}_t + 2s_\theta(t, \hat{Y}_t))\, dt + \sqrt{2} dW_t$$

- Euler-Maruyama:

$$d\hat{Y}_t = \left(\hat{Y}_{t_k} + 2s_\theta(t_k, \hat{Y}_{t_k})\right) dt + \sqrt{2} dW_t \qquad t \in [t_k, t_{k+1}]$$

- exponential integrator:

$$d\hat{Y}_t = \left(\hat{Y}_t + 2s_\theta(t_k, \hat{Y}_{t_k})\right) dt + \sqrt{2} dW_t \qquad t \in [t_k, t_{k+1}]$$

$\hat{Y}_t \sim \hat{\rho}_t$,what is the difference between $\hat{\rho}_{t_N}$ and ρ_T?

$$\partial_t \rho_t(x) = -\nabla \cdot ((x + \nabla \log q_{T-t}(x))\, \rho_t)$$

$$dY_t = \left(Y_t + 2\nabla \log q_{T-t}(Y_t)\right) dt + \sqrt{2} dw_t$$

Denoising Diffusion Probabilistic Model

Girsanov's theorem

Assume $y_0 \sim q_T$, consider the path measures Q_T and P_T on $C\big([0,T]; R^d\big)$ corresponding the following two diffusion processes:

$$Q_T: dY_t = \big(Y_t + 2\nabla \log q_{T-t}(Y_t)\big)dt + \sqrt{2}dW_t$$

$$P_T: d\hat{Y}_t = \left(\hat{Y}_t + 2s_\theta\big(t_k, \hat{Y}_{t_k}\big)\right)dt + \sqrt{2}dW_t \quad t \in [t_k, t_{k+1}]$$

$$\mathrm{KL}\big[\rho_{t_N} \parallel \hat{\rho}_{t_N}\big] \le \mathrm{KL}\big[Q_{t_N} \parallel P_{t_N}\big]$$

$$\le \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \mathbb{E}_{Q_{t_N}} \left\| s_\theta\big(t_k, y_{t_k}\big) - \nabla \log q_{T-t}(y_t) \right\|^2 \, dt$$

Denoising Diffusion Probabilistic Model

Convergence Analysis (Chen el al.2023; Benton el al. 2024)

A1: score approximation:

$$\sum_{k=0}^{N-1}(t_{k+1}-t_k)\mathbb{E}_{q_{t_k}}\left\|s_\theta(t_k,x)-\nabla\log q_{T-t_k}(x)\right\|\leq\delta^2$$

A2: data has finite second moments, and $\mathrm{Cov}_{q_0}(x)=I_d$,

We have $0=t_0<t_1\cdots<t_N=T-\tau$, and $t_{k+1}-t_k\leq\kappa\min\{1,T-t_{k+1}\}$, then

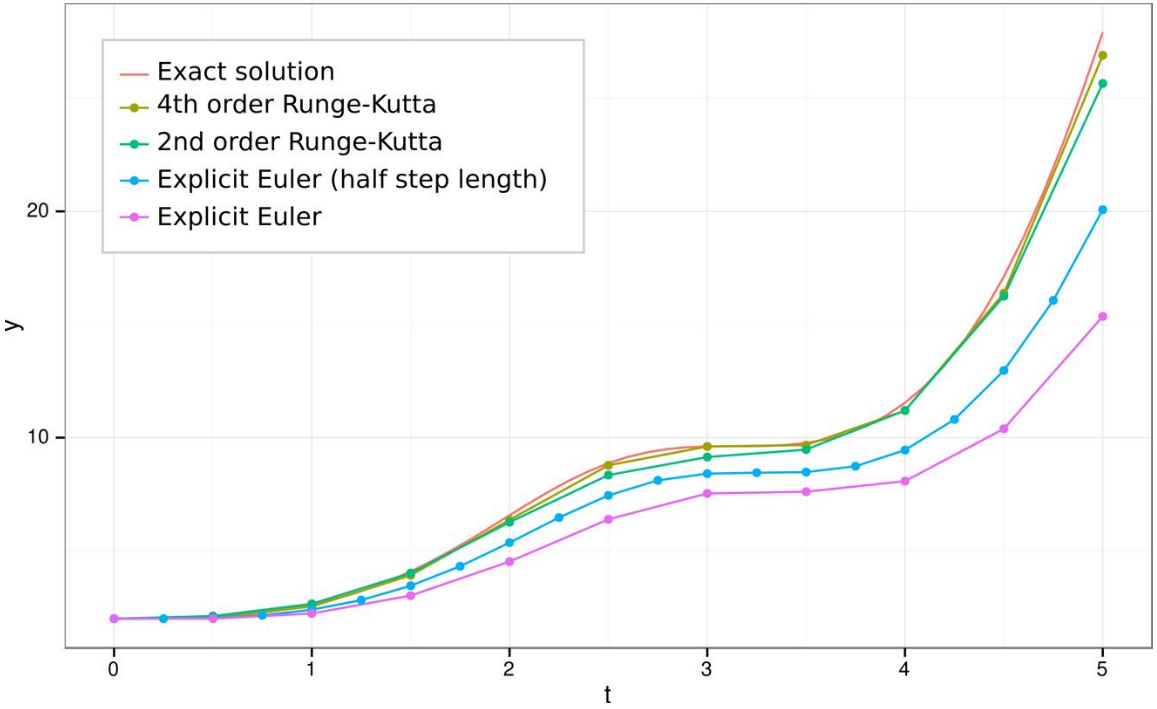
$$KL\left[\rho_T \parallel \hat{\rho}_{t_N}\right]\lesssim\delta^2+d\kappa^2N+\kappa dT+de^{-2T}$$

Probability Flow ODE

- Probability flow ODE

$$\frac{d\hat{Y}_t}{dt} = \hat{Y}_t + s_\theta(t, \hat{Y}_t)$$

High order schemes!
O(10) time steps, better efficiency!



Probability Flow ODE

- High order ODE scheme

$$\frac{d\hat{Y}_t}{dt} = \hat{Y}_t + s_\theta\left(t, \hat{Y}\right)$$

- (high order) Runge-Kutta scheme:

$$d\hat{Y}_t = \left(\hat{Y}_{t_k} + s_\theta\left(t_k, \hat{Y}_{t_k}\right)\right) dt \qquad t \in [t_k, t_{k+1}]$$

- (high order) exponential integrator:

$$d\hat{Y}_t = \left(\hat{Y}_t + s_\theta\left(t_k, \hat{Y}_{t_k}\right)\right) dt \qquad t \in [t_k, t_{k+1}]$$

Probability Flow ODE

- Previous theoretical study

$$\frac{d\hat{Y}_t}{dt} = \hat{Y}_t + s_\theta(t, \hat{Y}_t)$$

What is the difference between $\hat{\rho}_{t_N}$ and ρ_T

- exponentially large bound depending on the time T.
- strong assumption on the data distribution, i.e. log concave.
- bad dependence on the dimension d.
- require control of the difference between the derivatives of the true and approximate scores.

...

Probability Flow ODE

Lemma (Huang at el. 2024)

Assume $\rho_t, \hat{\rho}_t \in C^1\big([0,T]; R^d\big) \cap L^1\big([0,T]; R^d\big)$ solve the following two continuity equations

$$\partial_t \rho_t(x) = \nabla \cdot (U_t(x)\, \rho_t(x))$$
$$\partial_t \hat{\rho}_t(x) = \nabla \cdot \big(\hat{U}_t(x) \hat{\rho}_t(x)\big)$$

Then, we have

$$|\mathrm{TV}(\rho_T, \hat{\rho}_T) - \mathrm{TV}(\rho_0, \hat{\rho}_0)| \le \frac{1}{2} \int_0^T \int \left| \nabla \cdot \Big((U_t(x) - \hat{U}_t(x)) \rho_t(x) \Big) \right| dx\, dt$$

Probability Flow ODE

Assumptions

- (Compact support) The data distribution is compactly supported on a compact set $\{x \in R^d : \|x\|_\infty \leq D\}$

- (L^2 accurate score estimation)

$$\int_0^{T-\tau} \mathbb{E}_{q_{T-t}} \|s_\theta(t,x) - \nabla \log q_{T-t}(x)\|^2 dt \leq \delta^2$$

- (Regularity of approximate score) $s_\theta(t,x)$ is L-Lipschitz. To use p-th order Runge-Kunta method, we assume the first $(p+1)$-th derivatives are bounded by L.

Probability Flow ODE

Convergence Analysis

If $\hat{\rho}_{t_N}$ is the output of the probability flow ODE initialized from $\mathcal{N}(0, I_d)$. Using p-th order Runge-Kunta method, with uniform step size $h = \frac{T-\tau}{N}$, leads to :

$$\mathrm{TV}(\rho_{t_N}, \hat{\rho}_{t_N}) \leq e^{-T} dD + dT^{\frac{3}{4}}(L + \tau^{-2} D^3)^{\frac{1}{2}} \delta^{\frac{1}{2}} + d(dh)^p (LD)^{p+1} \log \frac{T}{\tau}$$

initialization error   score matching error   discretization error

- we only estimate the error up to the time $T - \tau$ instead of q_0. The Wasserstein 2-distance between q_0 and q_τ = ρ_{t_N} is bounded by $\mathcal{O}(\tau + d(\tau D)^2)$.
- this gives an error bounds of $\mathcal{O}(d\sqrt{\delta} + d(dh)^p)$.

Numerical Verification

- Data distribution q_0

  5-mode Gaussian mixture in R^d.

- Artificial score matching errors ($\delta(t,x) = s(t,x) - \nabla \log q_{T-t}(x)$)

  1. constant error: $\delta(t,x) = \delta \frac{1}{d}$
  2. linear error: $\delta(t,x) = \delta \frac{x-m}{d}$
  3. sinusoidal error: $\delta(t,x) = \delta \sin x \frac{x-m}{d}$

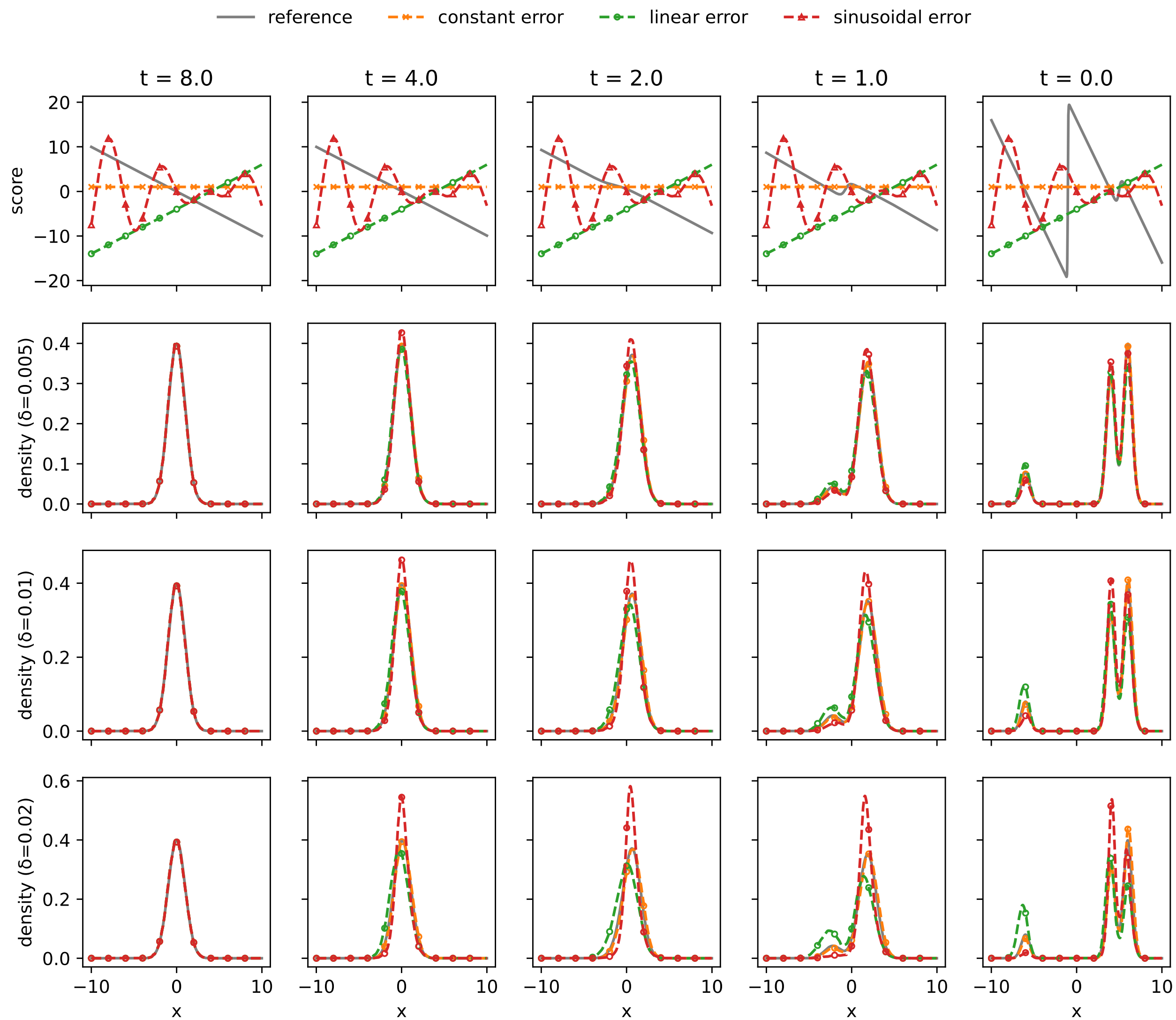
  where m is the mean of the Gaussian mixture; and $\sin x$ applies to each entry.

- Configurations

  second-order Heun's method.
  time $T = 8$, and initialize 4×10^4 particles from $\mathcal{N}(0, I_d)$.

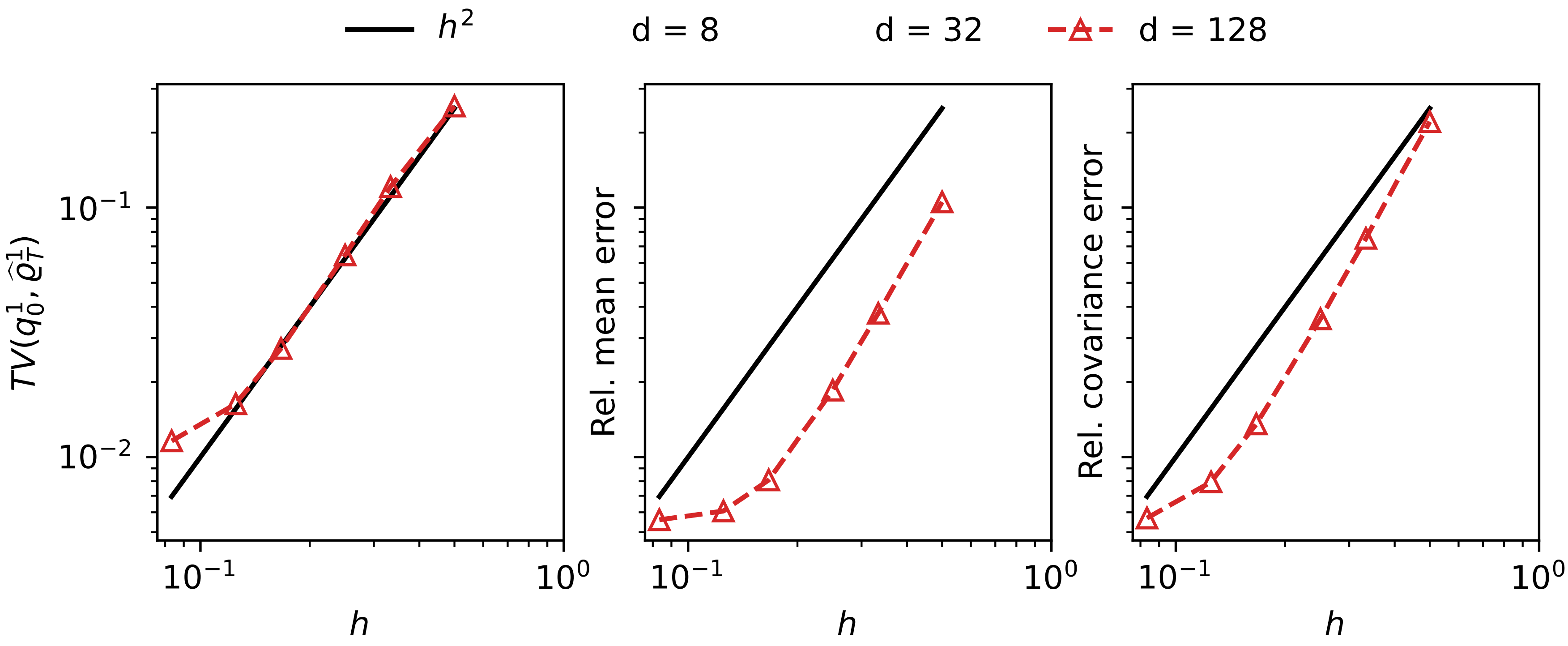
Numerical Verification

- Density evolution (kernel reconstruction)



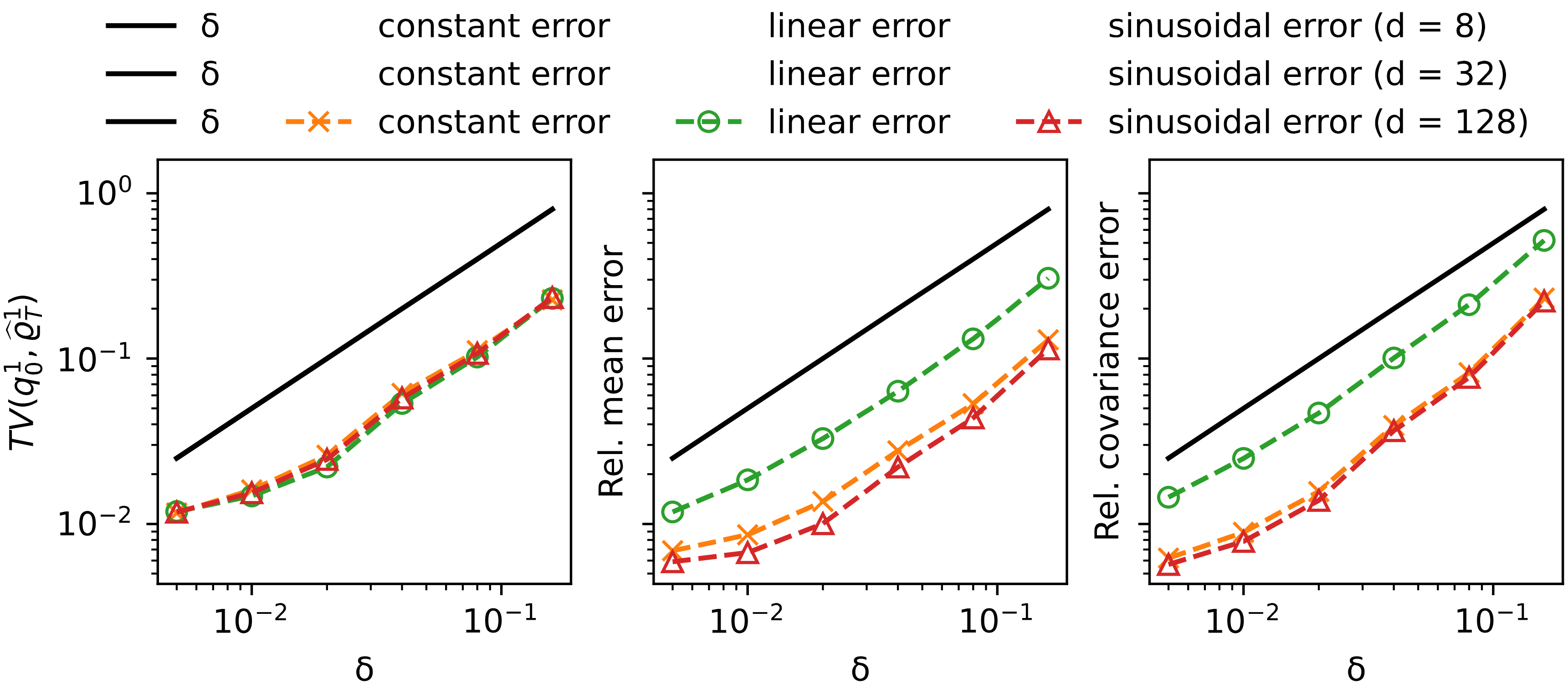
Numerical Verification

- Discretization error (δ=0)



Numerical Verification

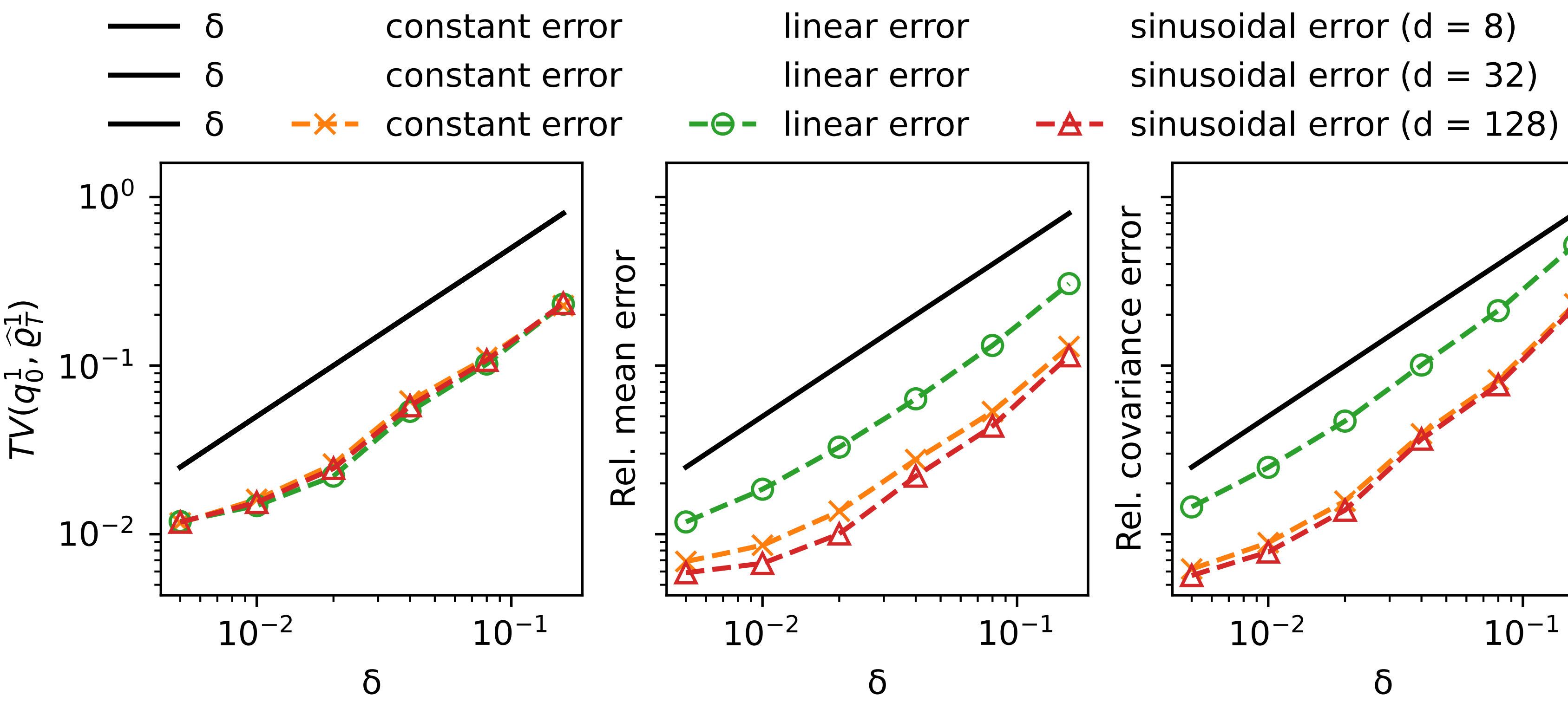
➢ Total error ($h^2 \approx \delta$)



- theoretic error bound of $\mathcal{O}(d\sqrt{\delta} + d(dh)^p)$.
- observed error bound : $\mathcal{O}(\delta + h^p)$.

Numerical Verification

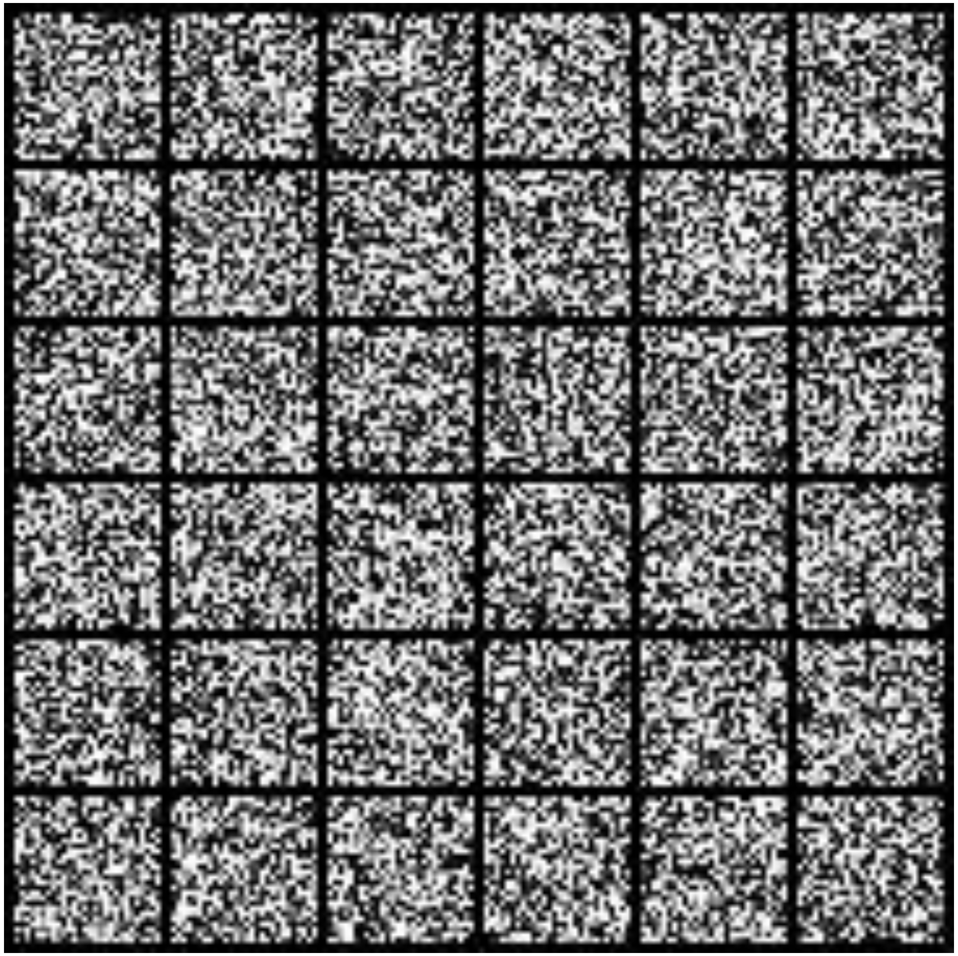
➢ Total error ($h^2 \approx \delta$)



- theoretic error bound of $\mathcal{O}(d\sqrt{\delta} + d(dh)^p)$.
- observed error bound : $\mathcal{O}(\delta + h^p)$.

Numerical Verification

- MINST (https://github.com/bot66/MNISTDiffusion)



References

➢ Methodology

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." Advances in neural information processing systems 33 (2020): 6840-6851.
Song, Yang, et al. “Score-based generative modeling through stochastic differential equations.” International Conference on Learning Representations, 2021.

Lu, Cheng, et al. "Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps." *Advances in Neural Information Processing Systems* 2022.

➢ Theoretical

Chen, Sitan, et al. "Sampling is as easy as learning the score: theory for diffusion models with minimal data assumptions." arXiv preprint arXiv:2209.11215 (2022).

Benton, Joe, et al. "Linear convergence bounds for diffusion models via stochastic localization." arXiv preprint arXiv:2308.03686 (2023).

Huang, Daniel Zhengyu, Jiaoyang Huang, and Zhengjiang Lin. "Convergence Analysis of Probability Flow ODE for Score-based Generative Models." arXiv preprint arXiv:2404.09730 (2024).