

Deep Generative Models

Lecture 11

Roman Isachenko

Moscow Institute of Physics and Technology
Yandex School of Data Analysis

2025, Autumn

Recap of Previous Lecture

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}_\theta(\mathbf{x}(t), t); \quad \text{with initial condition } \mathbf{x}(t_0) = \mathbf{x}_0$$

Theorem (Continuity Equation)

If \mathbf{f} is uniformly Lipschitz continuous in \mathbf{x} and continuous in t , then

$$\frac{d \log p_t(\mathbf{x}(t))}{dt} = -\text{tr} \left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right)$$

Solution of the Continuity Equation

$$\log p_1(\mathbf{x}(1)) = \log p_0(\mathbf{x}(0)) - \int_0^1 \text{tr} \left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \right) dt.$$

- ▶ This solution gives us the density along the trajectory (not the total probability path).
- ▶ However, it's difficult to efficiently estimate the last term.

Recap of Previous Lecture

SDE Basics

Let's define a stochastic process $\mathbf{x}(t)$ with initial condition $\mathbf{x}(0) \sim p_0(\mathbf{x})$:

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w},$$

where $\mathbf{w}(t)$ is the standard Wiener process (Brownian motion):

$$\mathbf{w}(t) - \mathbf{w}(s) \sim \mathcal{N}(0, (t-s)\mathbf{I}), \quad d\mathbf{w} = \boldsymbol{\epsilon} \cdot \sqrt{dt}, \text{ where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

Discretization of SDE (Euler Method) - SDESolve

$$\mathbf{x}(t + dt) = \mathbf{x}(t) + \mathbf{f}(\mathbf{x}(t), t) \cdot dt + g(t) \cdot \boldsymbol{\epsilon} \cdot \sqrt{dt}$$

- ▶ At each time t , we have the density $p_t(\mathbf{x}) = p(\mathbf{x}, t)$.
- ▶ $p : \mathbb{R}^m \times [0, 1] \rightarrow \mathbb{R}_+$ is a **probability path** between $p_0(\mathbf{x})$ and $p_1(\mathbf{x})$.

Recap of Previous Lecture

Theorem (Kolmogorov-Fokker-Planck)

The evolution of the distribution $p_t(\mathbf{x})$ is given by:

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) + \frac{1}{2}g^2(t)\Delta_{\mathbf{x}}p_t(\mathbf{x})$$

Langevin SDE (Special Case)

$$d\mathbf{x} = \frac{1}{2} \frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) dt + \mathbf{1} \cdot d\mathbf{w}$$

The density $p_\theta(\mathbf{x})$ is a **stationary** distribution for the SDE.

Langevin Dynamics

Samples from the following dynamics will come from $p_\theta(\mathbf{x})$ under mild regularity conditions for a small enough η :

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \frac{\eta}{2} \nabla_{\mathbf{x}_t} \log p_\theta(\mathbf{x}_t) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

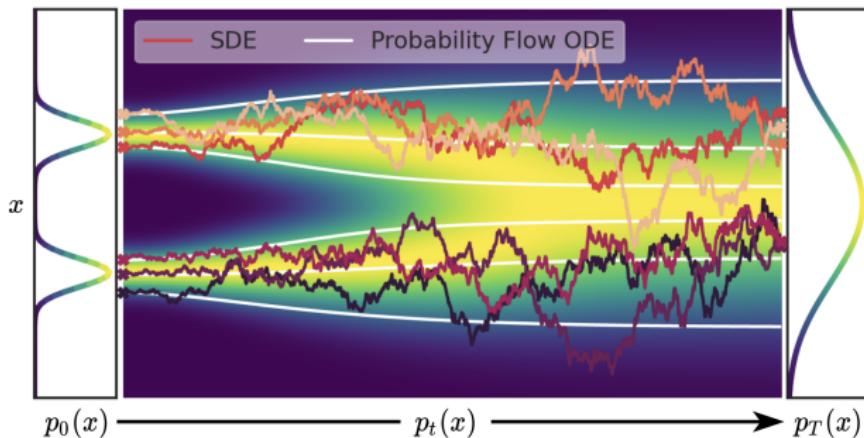
Recap of Previous Lecture

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w} \quad (\text{SDE with the probability path } p_t(\mathbf{x}))$$

Probability Flow ODE

There exists an ODE with the identical probability path $p_t(\mathbf{x})$ of the form:

$$d\mathbf{x} = \left(\mathbf{f}(\mathbf{x}, t) - \frac{1}{2}g^2(t)\frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt$$



Recap of Previous Lecture

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt, \quad \mathbf{x}(t + dt) = \mathbf{x}(t) + \mathbf{f}(\mathbf{x}, t)dt$$

Reverse ODE

Let $\tau = 1 - t$ ($d\tau = -dt$):

$$d\mathbf{x} = -\mathbf{f}(\mathbf{x}, 1 - \tau)d\tau$$

Reverse SDE

There's a reverse SDE for $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ in the following form:

$$d\mathbf{x} = \left(\mathbf{f}(\mathbf{x}, t) - g^2(t) \frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt + g(t)d\mathbf{w}, \quad dt < 0$$

Sketch of the Proof

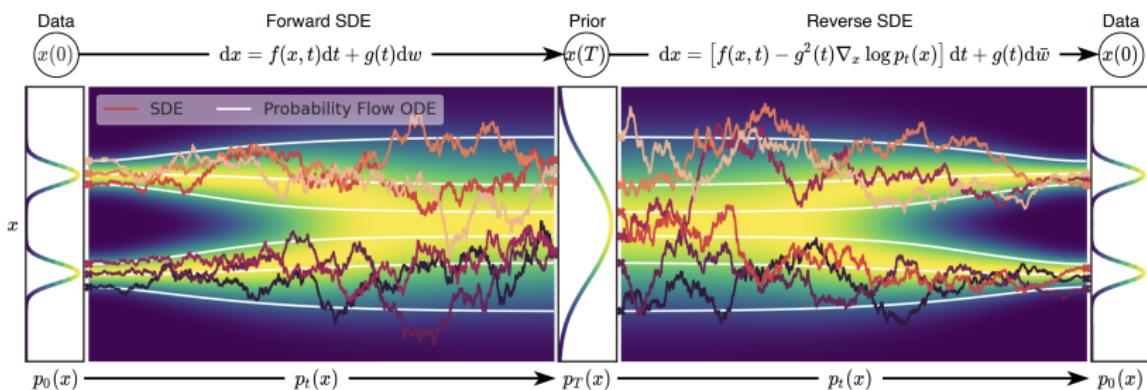
- ▶ Convert the initial SDE to the probability flow ODE.
- ▶ Reverse the probability flow ODE.
- ▶ Convert the reverse probability flow ODE to the reverse SDE.

Recap of Previous Lecture

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w} \quad (\text{SDE})$$

$$d\mathbf{x} = \left(\mathbf{f}(\mathbf{x}, t) - \frac{1}{2}g^2(t)\frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt \quad (\text{probability flow ODE})$$

$$d\mathbf{x} = \left(\mathbf{f}(\mathbf{x}, t) - g^2(t)\frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt + g(t)d\bar{\mathbf{w}} \quad (\text{reverse SDE})$$



Outline

1. Diffusion and Score Matching SDEs
2. Score-Based Generative Models Through SDEs
3. Flow Matching
4. Conditional Flow Matching

Outline

1. Diffusion and Score Matching SDEs
2. Score-Based Generative Models Through SDEs
3. Flow Matching
4. Conditional Flow Matching

Score Matching SDE

Denoising Score Matching

$$\begin{aligned}\mathbf{x}_t &= \mathbf{x} + \sigma_t \cdot \boldsymbol{\epsilon}_t, & q(\mathbf{x}_t | \mathbf{x}) &= \mathcal{N}(\mathbf{x}, \sigma_t^2 \cdot \mathbf{I}) \\ \mathbf{x}_{t-1} &= \mathbf{x} + \sigma_{t-1} \cdot \boldsymbol{\epsilon}_{t-1}, & q(\mathbf{x}_{t-1} | \mathbf{x}) &= \mathcal{N}(\mathbf{x}, \sigma_{t-1}^2 \cdot \mathbf{I})\end{aligned}$$

Score Matching SDE

Denoising Score Matching

$$\mathbf{x}_t = \mathbf{x} + \sigma_t \cdot \boldsymbol{\epsilon}_t, \quad q(\mathbf{x}_t | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_t^2 \cdot \mathbf{I})$$

$$\mathbf{x}_{t-1} = \mathbf{x} + \sigma_{t-1} \cdot \boldsymbol{\epsilon}_{t-1}, \quad q(\mathbf{x}_{t-1} | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_{t-1}^2 \cdot \mathbf{I})$$

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_{t-1}, (\sigma_t^2 - \sigma_{t-1}^2) \cdot \mathbf{I})$$

Score Matching SDE

Denoising Score Matching

$$\mathbf{x}_t = \mathbf{x} + \sigma_t \cdot \boldsymbol{\epsilon}_t, \quad q(\mathbf{x}_t | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_t^2 \cdot \mathbf{I})$$

$$\mathbf{x}_{t-1} = \mathbf{x} + \sigma_{t-1} \cdot \boldsymbol{\epsilon}_{t-1}, \quad q(\mathbf{x}_{t-1} | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_{t-1}^2 \cdot \mathbf{I})$$

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_{t-1}, (\sigma_t^2 - \sigma_{t-1}^2) \cdot \mathbf{I})$$

Let's transform this Markov chain into the continuous stochastic process $\mathbf{x}(t)$ by letting $T \rightarrow \infty$:

$$\mathbf{x}(t) = \mathbf{x}(t - dt) + \sqrt{\sigma^2(t) - \sigma^2(t - dt)} \cdot \boldsymbol{\epsilon}$$

Score Matching SDE

Denoising Score Matching

$$\mathbf{x}_t = \mathbf{x} + \sigma_t \cdot \boldsymbol{\epsilon}_t, \quad q(\mathbf{x}_t | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_t^2 \cdot \mathbf{I})$$

$$\mathbf{x}_{t-1} = \mathbf{x} + \sigma_{t-1} \cdot \boldsymbol{\epsilon}_{t-1}, \quad q(\mathbf{x}_{t-1} | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_{t-1}^2 \cdot \mathbf{I})$$

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_{t-1}, (\sigma_t^2 - \sigma_{t-1}^2) \cdot \mathbf{I})$$

Let's transform this Markov chain into the continuous stochastic process $\mathbf{x}(t)$ by letting $T \rightarrow \infty$:

$$\begin{aligned}\mathbf{x}(t) &= \mathbf{x}(t - dt) + \sqrt{\sigma^2(t) - \sigma^2(t - dt)} \cdot \boldsymbol{\epsilon} \\ &= \mathbf{x}(t - dt) + \sqrt{\frac{\sigma^2(t) - \sigma^2(t - dt)}{dt}} dt \cdot \boldsymbol{\epsilon}\end{aligned}$$

Score Matching SDE

Denoising Score Matching

$$\mathbf{x}_t = \mathbf{x} + \sigma_t \cdot \boldsymbol{\epsilon}_t, \quad q(\mathbf{x}_t | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_t^2 \cdot \mathbf{I})$$

$$\mathbf{x}_{t-1} = \mathbf{x} + \sigma_{t-1} \cdot \boldsymbol{\epsilon}_{t-1}, \quad q(\mathbf{x}_{t-1} | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma_{t-1}^2 \cdot \mathbf{I})$$

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \sqrt{\sigma_t^2 - \sigma_{t-1}^2} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_{t-1}, (\sigma_t^2 - \sigma_{t-1}^2) \cdot \mathbf{I})$$

Let's transform this Markov chain into the continuous stochastic process $\mathbf{x}(t)$ by letting $T \rightarrow \infty$:

$$\begin{aligned}\mathbf{x}(t) &= \mathbf{x}(t - dt) + \sqrt{\sigma^2(t) - \sigma^2(t - dt)} \cdot \boldsymbol{\epsilon} \\ &= \mathbf{x}(t - dt) + \sqrt{\frac{\sigma^2(t) - \sigma^2(t - dt)}{dt}} dt \cdot \boldsymbol{\epsilon} \\ &= \mathbf{x}(t - dt) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}\end{aligned}$$

Score Matching SDE

$$\mathbf{x}(t) = \mathbf{x}(t - dt) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Score Matching SDE

$$\mathbf{x}(t) = \mathbf{x}(t - dt) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Exploding SDE

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

$\sigma(t)$ is a monotonically increasing function.

Score Matching SDE

$$\mathbf{x}(t) = \mathbf{x}(t - dt) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Exploding SDE

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

$\sigma(t)$ is a monotonically increasing function.

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}, \quad \mathbf{f}(\mathbf{x}, t) = 0, \quad g(t) = \sqrt{\frac{d[\sigma^2(t)]}{dt}}$$

Score Matching SDE

$$\mathbf{x}(t) = \mathbf{x}(t - dt) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Exploding SDE

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

$\sigma(t)$ is a monotonically increasing function.

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}, \quad \mathbf{f}(\mathbf{x}, t) = 0, \quad g(t) = \sqrt{\frac{d[\sigma^2(t)]}{dt}}$$

$$d\mathbf{x} = \left(-\frac{1}{2} \frac{d[\sigma^2(t)]}{dt} \frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt \quad (\text{probability flow ODE})$$

$$d\mathbf{x} = \left(-\frac{d[\sigma^2(t)]}{dt} \frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt + \sqrt{\frac{d[\sigma^2(t)]}{dt}} d\mathbf{w} \quad (\text{reverse SDE})$$

Diffusion SDE

Denoising Diffusion

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I})$$

Diffusion SDE

Denoising Diffusion

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I})$$

Let's turn this Markov chain into a continuous stochastic process by letting $T \rightarrow \infty$ and setting $\beta_t = \beta\left(\frac{t}{T}\right) \cdot \frac{1}{T}$ (where $dt = \frac{1}{T}$):

$$\mathbf{x}(t) = \sqrt{1 - \beta(t)dt} \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon}$$

Diffusion SDE

Denoising Diffusion

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I})$$

Let's turn this Markov chain into a continuous stochastic process by letting $T \rightarrow \infty$ and setting $\beta_t = \beta\left(\frac{t}{T}\right) \cdot \frac{1}{T}$ (where $dt = \frac{1}{T}$):

$$\begin{aligned}\mathbf{x}(t) &= \sqrt{1 - \beta(t)dt} \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} \approx \\ &\approx \left(1 - \frac{1}{2}\beta(t)dt\right) \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon}\end{aligned}$$

Diffusion SDE

Denoising Diffusion

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I})$$

Let's turn this Markov chain into a continuous stochastic process by letting $T \rightarrow \infty$ and setting $\beta_t = \beta\left(\frac{t}{T}\right) \cdot \frac{1}{T}$ (where $dt = \frac{1}{T}$):

$$\begin{aligned}\mathbf{x}(t) &= \sqrt{1 - \beta(t)dt} \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} \approx \\ &\approx \left(1 - \frac{1}{2}\beta(t)dt\right) \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} = \\ &= \mathbf{x}(t - dt) - \frac{1}{2}\beta(t)\mathbf{x}(t - dt)dt + \sqrt{\beta(t)} \cdot \mathbf{d}\mathbf{w}\end{aligned}$$

Diffusion SDE

Denoising Diffusion

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I})$$

Let's turn this Markov chain into a continuous stochastic process by letting $T \rightarrow \infty$ and setting $\beta_t = \beta\left(\frac{t}{T}\right) \cdot \frac{1}{T}$ (where $dt = \frac{1}{T}$):

$$\begin{aligned}\mathbf{x}(t) &= \sqrt{1 - \beta(t)dt} \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} \approx \\ &\approx \left(1 - \frac{1}{2}\beta(t)dt\right) \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} = \\ &= \mathbf{x}(t - dt) - \frac{1}{2}\beta(t)\mathbf{x}(t - dt)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}\end{aligned}$$

Variance Preserving SDE

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

Diffusion SDE

Variance Preserving SDE

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

$$\mathbf{f}(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}(t), \quad g(t) = \sqrt{\beta(t)}$$

Variance is preserved as long as $\mathbf{x}(0)$ has unit variance.

Diffusion SDE

Variance Preserving SDE

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

$$\mathbf{f}(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}(t), \quad g(t) = \sqrt{\beta(t)}$$

Variance is preserved as long as $\mathbf{x}(0)$ has unit variance.

$$d\mathbf{x} = \left(-\frac{1}{2}\beta(t)\mathbf{x}(t) - \frac{1}{2}\beta(t)\frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt \quad (\text{probability flow ODE})$$

$$d\mathbf{x} = \left(-\frac{1}{2}\beta(t)\mathbf{x}(t) - \beta(t)\frac{\partial}{\partial \mathbf{x}} \log p_t(\mathbf{x}) \right) dt + \sqrt{\beta(t)}d\mathbf{w} \quad (\text{reverse SDE})$$

Diffusion SDE

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + \mathbf{g}(t)d\mathbf{w}$$

Diffusion SDE

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + \mathbf{g}(t)d\mathbf{w}$$

Variance Exploding SDE (NCSN)

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Preserving SDE (DDPM)

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

Diffusion SDE

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + \mathbf{g}(t)d\mathbf{w}$$

Variance Exploding SDE (NCSN)

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Preserving SDE (DDPM)

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

Efficient Solvers

- ▶ Converting SDEs to PF-ODEs yields more efficient inference.
- ▶ We can apply any ODESolve procedure to reduce the number of inference steps.
- ▶ In practice, this reduces the number of steps from 100–1000 to 20–50.

Outline

1. Diffusion and Score Matching SDEs
2. Score-Based Generative Models Through SDEs
3. Flow Matching
4. Conditional Flow Matching

Score-Based Generative Models Through SDEs

Discrete-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \mathbb{E}_{t \sim U\{1, T\}} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \left\| \mathbf{s}_{\theta, t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \right\|_2^2$$

Is it possible to train score-based diffusion models in continuous time?

Score-Based Generative Models Through SDEs

Discrete-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \mathbb{E}_{t \sim U\{1, T\}} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \| \mathbf{s}_{\theta, t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \|_2^2$$

Is it possible to train score-based diffusion models in continuous time?

Continuous-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0, 1]} \mathbb{E}_{q(\mathbf{x}(t) | \mathbf{x}(0))} \| \mathbf{s}_{\theta}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t) | \mathbf{x}(0)) \|_2^2$$

Score-Based Generative Models Through SDEs

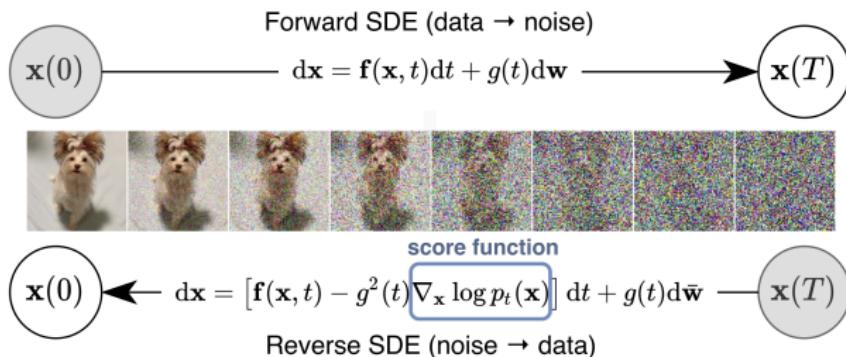
Discrete-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \mathbb{E}_{t \sim U\{1, T\}} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \| \mathbf{s}_{\theta, t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \|_2^2$$

Is it possible to train score-based diffusion models in continuous time?

Continuous-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0, 1]} \mathbb{E}_{q(\mathbf{x}(t) | \mathbf{x}(0))} \| \mathbf{s}_{\theta}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t) | \mathbf{x}(0)) \|_2^2$$



Score-Based Generative Models Through SDEs

Continuous-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_{\theta}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

Score-Based Generative Models Through SDEs

Continuous-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_{\theta}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\sigma}^2(t, \mathbf{x}(0)) \cdot \mathbf{I}\right)$$

$$\nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) = -\frac{1}{\sigma} \odot (\mathbf{x}(t) - \boldsymbol{\mu})$$

Note: Normality holds for $\mathbf{f}(\mathbf{x}, t)$ affine in \mathbf{x} .

Score-Based Generative Models Through SDEs

Continuous-Time Objective

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_{\theta}(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\sigma}^2(t, \mathbf{x}(0)) \cdot \mathbf{I}\right)$$

$$\nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) = -\frac{1}{\sigma} \odot (\mathbf{x}(t) - \boldsymbol{\mu})$$

Note: Normality holds for $\mathbf{f}(\mathbf{x}, t)$ affine in \mathbf{x} .

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w} \quad (\text{Variance Exploding SDE})$$

$$d\mathbf{x} = -\frac{1}{2} \beta(t) \mathbf{x}(t) dt + \sqrt{\beta(t)} \cdot d\mathbf{w} \quad (\text{Variance Preserving SDE})$$

Is it possible to explicitly derive $\boldsymbol{\mu}(t, \mathbf{x}(0))$ and $\boldsymbol{\Sigma}(t, \mathbf{x}(0))$ for VE-SDE and VP-SDE?

Score-Based Generative Models Through SDEs

$$q(\mathbf{x}(t) | \mathbf{x}(0)) = \mathcal{N}(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0)))$$

Theorem

The moments of the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ satisfy:

$$\frac{d\boldsymbol{\mu}(t, \mathbf{x}(0))}{dt} = \mathbb{E}[\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

$$\frac{d\boldsymbol{\Sigma}(t, \mathbf{x}(0))}{dt} = \mathbb{E} \left[\mathbf{f} \cdot (\mathbf{x}(t) - \boldsymbol{\mu})^\top + (\mathbf{x}(t) - \boldsymbol{\mu}) \cdot \mathbf{f}^\top | \mathbf{x}(0) \right] + g^2(t) \cdot \mathbf{I}$$

Score-Based Generative Models Through SDEs

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

Theorem

The moments of the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ satisfy:

$$\frac{d\boldsymbol{\mu}(t, \mathbf{x}(0))}{dt} = \mathbb{E}[\mathbf{f}(\mathbf{x}(t), t)|\mathbf{x}(0)]$$

$$\frac{d\boldsymbol{\Sigma}(t, \mathbf{x}(0))}{dt} = \mathbb{E}\left[\mathbf{f} \cdot (\mathbf{x}(t) - \boldsymbol{\mu})^\top + (\mathbf{x}(t) - \boldsymbol{\mu}) \cdot \mathbf{f}^\top | \mathbf{x}(0)\right] + g^2(t) \cdot \mathbf{I}$$

Proof

$$\mathbb{E}[d\mathbf{x}|\mathbf{x}(0)] = \mathbb{E}[\mathbf{f}(\mathbf{x}, t)dt|\mathbf{x}(0)] + \mathbb{E}[g(t)d\mathbf{w}|\mathbf{x}(0)]$$

Score-Based Generative Models Through SDEs

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

Theorem

The moments of the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ satisfy:

$$\frac{d\boldsymbol{\mu}(t, \mathbf{x}(0))}{dt} = \mathbb{E}[\mathbf{f}(\mathbf{x}(t), t)|\mathbf{x}(0)]$$

$$\frac{d\boldsymbol{\Sigma}(t, \mathbf{x}(0))}{dt} = \mathbb{E}\left[\mathbf{f} \cdot (\mathbf{x}(t) - \boldsymbol{\mu})^\top + (\mathbf{x}(t) - \boldsymbol{\mu}) \cdot \mathbf{f}^\top | \mathbf{x}(0)\right] + g^2(t) \cdot \mathbf{I}$$

Proof

$$\begin{aligned}\mathbb{E}[d\mathbf{x}|\mathbf{x}(0)] &= \mathbb{E}[\mathbf{f}(\mathbf{x}, t)dt|\mathbf{x}(0)] + \mathbb{E}[g(t)d\mathbf{w}|\mathbf{x}(0)] \\ &= \mathbb{E}[\mathbf{f}(\mathbf{x}, t)|\mathbf{x}(0)] dt + g(t)\mathbb{E}[d\mathbf{w}|\mathbf{x}(0)]\end{aligned}$$

Score-Based Generative Models Through SDEs

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

Theorem

The moments of the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ satisfy:

$$\frac{d\boldsymbol{\mu}(t, \mathbf{x}(0))}{dt} = \mathbb{E}[\mathbf{f}(\mathbf{x}(t), t)|\mathbf{x}(0)]$$

$$\frac{d\boldsymbol{\Sigma}(t, \mathbf{x}(0))}{dt} = \mathbb{E}\left[\mathbf{f} \cdot (\mathbf{x}(t) - \boldsymbol{\mu})^\top + (\mathbf{x}(t) - \boldsymbol{\mu}) \cdot \mathbf{f}^\top | \mathbf{x}(0)\right] + g^2(t) \cdot \mathbf{I}$$

Proof

$$\begin{aligned}\mathbb{E}[d\mathbf{x}|\mathbf{x}(0)] &= \mathbb{E}[\mathbf{f}(\mathbf{x}, t)dt|\mathbf{x}(0)] + \mathbb{E}[g(t)d\mathbf{w}|\mathbf{x}(0)] \\ &= \mathbb{E}[\mathbf{f}(\mathbf{x}, t)|\mathbf{x}(0)] dt + g(t)\mathbb{E}[d\mathbf{w}|\mathbf{x}(0)] \\ &= \mathbb{E}[\mathbf{f}(\mathbf{x}, t)|\mathbf{x}(0)] dt\end{aligned}$$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Proof (Continued)

$$\mathbb{E} [d\mathbf{x} | \mathbf{x}(0)] = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)] dt$$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Proof (Continued)

$$\mathbb{E} [d\mathbf{x} | \mathbf{x}(0)] = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)] dt$$

$$\frac{d\mathbb{E} [\mathbf{x}(t) | \mathbf{x}(0)]}{dt} = \frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)]$$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Proof (Continued)

$$\mathbb{E} [d\mathbf{x} | \mathbf{x}(0)] = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)] dt$$

$$\frac{d\mathbb{E} [\mathbf{x}(t) | \mathbf{x}(0)]}{dt} = \frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)]$$

Examples

NCSN: $\mathbf{f}(\mathbf{x}, t) = 0 \quad \Rightarrow \quad \mu = \mathbf{x}(0)$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Proof (Continued)

$$\mathbb{E} [d\mathbf{x} | \mathbf{x}(0)] = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)] dt$$

$$\frac{d\mathbb{E} [\mathbf{x}(t) | \mathbf{x}(0)]}{dt} = \frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)]$$

Examples

NCSN: $\mathbf{f}(\mathbf{x}, t) = 0 \Rightarrow \mu = \mathbf{x}(0)$

DDPM: $\mathbf{f}(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}(t) \Rightarrow \frac{d\mu}{dt} = -\frac{1}{2}\beta(t)\mu$

Score-Based Generative Models Through SDEs

Theorem

$$\frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}(t), t) | \mathbf{x}(0)]$$

Proof (Continued)

$$\mathbb{E} [d\mathbf{x} | \mathbf{x}(0)] = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)] dt$$

$$\frac{d\mathbb{E} [\mathbf{x}(t) | \mathbf{x}(0)]}{dt} = \frac{d\mu(t, \mathbf{x}(0))}{dt} = \mathbb{E} [\mathbf{f}(\mathbf{x}, t) | \mathbf{x}(0)]$$

Examples

NCSN: $\mathbf{f}(\mathbf{x}, t) = 0 \Rightarrow \mu = \mathbf{x}(0)$

DDPM: $\mathbf{f}(\mathbf{x}, t) = -\frac{1}{2}\beta(t)\mathbf{x}(t) \Rightarrow \frac{d\mu}{dt} = -\frac{1}{2}\beta(t)\mu$

$$\mu = \mathbf{x}(0) \exp \left(-\frac{1}{2} \int_0^t \beta(s) ds \right)$$

Score-Based Generative Models Through SDEs

Training

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_\theta(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

Score-Based Generative Models Through SDEs

Training

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_\theta(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

NCSN

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\mathbf{x}(0), [\sigma^2(t) - \sigma^2(0)] \cdot \mathbf{I}\right)$$

Score-Based Generative Models Through SDEs

Training

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}(0))} \mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{q(\mathbf{x}(t)|\mathbf{x}(0))} \| \mathbf{s}_\theta(\mathbf{x}(t), t) - \nabla_{\mathbf{x}(t)} \log q(\mathbf{x}(t)|\mathbf{x}(0)) \|_2^2$$

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\boldsymbol{\mu}(t, \mathbf{x}(0)), \boldsymbol{\Sigma}(t, \mathbf{x}(0))\right)$$

NCSN

$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\mathbf{x}(0), [\sigma^2(t) - \sigma^2(0)] \cdot \mathbf{I}\right)$$

DDPM

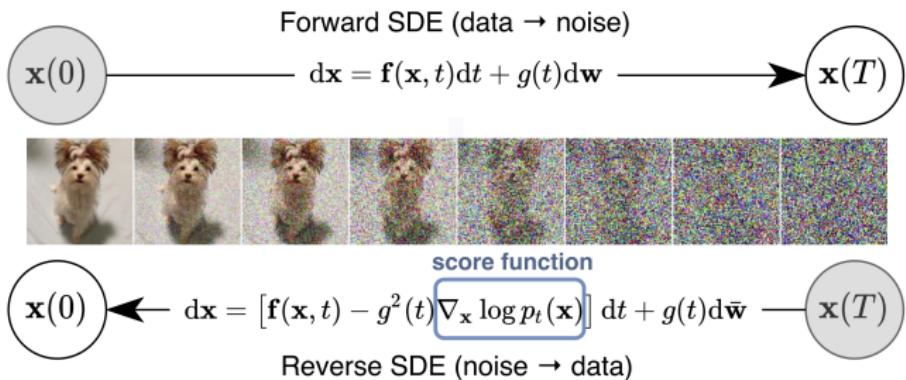
$$q(\mathbf{x}(t)|\mathbf{x}(0)) = \mathcal{N}\left(\mathbf{x}(0)e^{-\frac{1}{2} \int_0^t \beta(s) ds}, \left(1 - e^{-\int_0^t \beta(s) ds}\right) \cdot \mathbf{I}\right)$$

Here we omit the derivations of the variance.

Score-Based Generative Models Through SDEs

Sampling

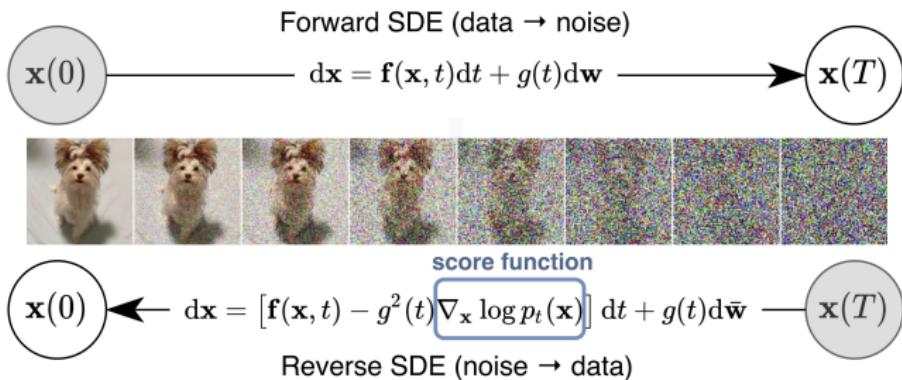
Solve the reverse SDE using numerical solvers (`SDESolve`).



Score-Based Generative Models Through SDEs

Sampling

Solve the reverse SDE using numerical solvers (`SDESolve`).



- ▶ Discretizing the reverse SDE provides ancestral sampling.
- ▶ Discretizing the probability flow ODE yields deterministic sampling.

Outline

1. Diffusion and Score Matching SDEs
2. Score-Based Generative Models Through SDEs
3. Flow Matching
4. Conditional Flow Matching

Continuous-Time Normalizing Flows

Let's return to ODE dynamics $\mathbf{x}(t)$ in the interval $t \in [0, 1]$:

- ▶ $\mathbf{x}_0 \sim p_0(\mathbf{x}) = p(\mathbf{x})$, $\mathbf{x}_1 \sim p_1(\mathbf{x}) = p_{\text{data}}(\mathbf{x})$;
- ▶ $p(\mathbf{x})$ is a base distribution (e.g., $\mathcal{N}(0, \mathbf{I})$), and $p_{\text{data}}(\mathbf{x})$ is the true data distribution.

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t), \quad \text{with initial condition } \mathbf{x}(0) = \mathbf{x}_0.$$

Continuous-Time Normalizing Flows

Let's return to ODE dynamics $\mathbf{x}(t)$ in the interval $t \in [0, 1]$:

- ▶ $\mathbf{x}_0 \sim p_0(\mathbf{x}) = p(\mathbf{x})$, $\mathbf{x}_1 \sim p_1(\mathbf{x}) = p_{\text{data}}(\mathbf{x})$;
- ▶ $p(\mathbf{x})$ is a base distribution (e.g., $\mathcal{N}(0, \mathbf{I})$), and $p_{\text{data}}(\mathbf{x})$ is the true data distribution.

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t), \quad \text{with initial condition } \mathbf{x}(0) = \mathbf{x}_0.$$

KFP Theorem (Continuity Equation)

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) \Leftrightarrow \frac{d \log p_t(\mathbf{x}(t))}{dt} = -\text{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right)$$

Continuous-Time Normalizing Flows

Let's return to ODE dynamics $\mathbf{x}(t)$ in the interval $t \in [0, 1]$:

- ▶ $\mathbf{x}_0 \sim p_0(\mathbf{x}) = p(\mathbf{x})$, $\mathbf{x}_1 \sim p_1(\mathbf{x}) = p_{\text{data}}(\mathbf{x})$;
- ▶ $p(\mathbf{x})$ is a base distribution (e.g., $\mathcal{N}(0, \mathbf{I})$), and $p_{\text{data}}(\mathbf{x})$ is the true data distribution.

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t), \quad \text{with initial condition } \mathbf{x}(0) = \mathbf{x}_0.$$

KFP Theorem (Continuity Equation)

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) \Leftrightarrow \frac{d \log p_t(\mathbf{x}(t))}{dt} = -\text{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right)$$

- ▶ It's hard to solve the continuity equation directly due to the trace term.
- ▶ There's a method (the adjoint method) that solves this equation directly, but it's unstable and unscalable.

Continuous-Time Normalizing Flows

KFP Theorem (Continuity Equation)

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\operatorname{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) \Leftrightarrow \frac{d \log p_t(\mathbf{x}(t))}{dt} = -\operatorname{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right)$$

- ▶ Knowing the vector field $\mathbf{f}(\mathbf{x}, t)$, the KFP (or continuity) equation allows us to compute the density $p_t(\mathbf{x})$.
- ▶ Flow matching provides an alternative approach to Neural ODEs.

Continuous-Time Normalizing Flows

KFP Theorem (Continuity Equation)

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) \Leftrightarrow \frac{d \log p_t(\mathbf{x}(t))}{dt} = -\text{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right)$$

- ▶ Knowing the vector field $\mathbf{f}(\mathbf{x}, t)$, the KFP (or continuity) equation allows us to compute the density $p_t(\mathbf{x})$.
- ▶ Flow matching provides an alternative approach to Neural ODEs.

Flow Matching

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Continuous-Time Normalizing Flows

KFP Theorem (Continuity Equation)

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, t)p_t(\mathbf{x})) \Leftrightarrow \frac{d \log p_t(\mathbf{x}(t))}{dt} = -\text{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right)$$

- ▶ Knowing the vector field $\mathbf{f}(\mathbf{x}, t)$, the KFP (or continuity) equation allows us to compute the density $p_t(\mathbf{x})$.
- ▶ Flow matching provides an alternative approach to Neural ODEs.

Flow Matching

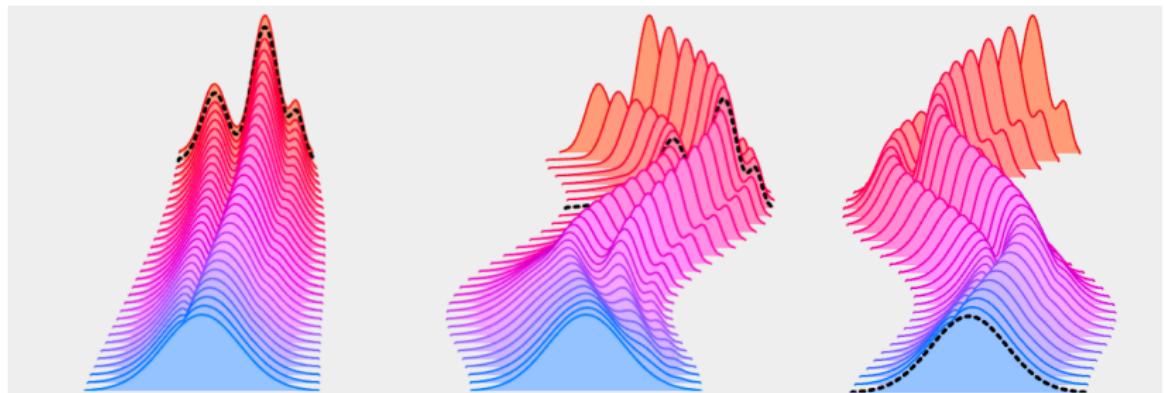
$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

- ▶ Approximate the true vector field $\mathbf{f}(\mathbf{x}, t)$ using $\mathbf{f}_\theta(\mathbf{x}, t)$.
- ▶ Use $\mathbf{f}_\theta(\mathbf{x}, t)$ for deterministic sampling from the ODE.

Flow Matching

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

- ▶ There are infinitely many possible $\mathbf{f}(\mathbf{x}, t)$ between $p_{\text{data}}(\mathbf{x})$ and $p(\mathbf{x})$.
- ▶ The true vector field $\mathbf{f}(\mathbf{x}, t)$ is **unknown**.
- ▶ We need to select the "best" $\mathbf{f}(\mathbf{x}, t)$ and make the objective tractable.



Outline

1. Diffusion and Score Matching SDEs
2. Score-Based Generative Models Through SDEs
3. Flow Matching
4. Conditional Flow Matching

Flow Matching

Latent Variable Model

Let's introduce the latent variable \mathbf{z} :

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Here, $p_t(\mathbf{x}|\mathbf{z})$ is a **conditional probability path**.

Flow Matching

Latent Variable Model

Let's introduce the latent variable \mathbf{z} :

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Here, $p_t(\mathbf{x}|\mathbf{z})$ is a **conditional probability path**. The conditional probability path $p_t(\mathbf{x}|\mathbf{z})$ satisfies the KFP theorem:

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\operatorname{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t)p_t(\mathbf{x}|\mathbf{z})),$$

where $\mathbf{f}(\mathbf{x}, \mathbf{z}, t)$ is a **conditional vector field**:

Flow Matching

Latent Variable Model

Let's introduce the latent variable \mathbf{z} :

$$p_t(\mathbf{x}) = \int p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Here, $p_t(\mathbf{x}|\mathbf{z})$ is a **conditional probability path**. The conditional probability path $p_t(\mathbf{x}|\mathbf{z})$ satisfies the KFP theorem:

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t)p_t(\mathbf{x}|\mathbf{z})),$$

where $\mathbf{f}(\mathbf{x}, \mathbf{z}, t)$ is a **conditional vector field**:

$$\frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, t) \quad \Rightarrow \quad \frac{d\mathbf{x}}{dt} = \mathbf{f}(\mathbf{x}, \mathbf{z}, t)$$

What's the relationship between $\mathbf{f}(\mathbf{x}, t)$ and $\mathbf{f}(\mathbf{x}, \mathbf{z}, t)$?

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\operatorname{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\operatorname{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Proof

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = \frac{\partial}{\partial t} \int p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\operatorname{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Proof

$$\frac{\partial p_t(\mathbf{x})}{\partial t} = \frac{\partial}{\partial t} \int p_t(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z} = \int \left(\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} \right) p(\mathbf{z})d\mathbf{z}$$

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Proof

$$\begin{aligned} \frac{\partial p_t(\mathbf{x})}{\partial t} &= \frac{\partial}{\partial t} \int p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} = \int \left(\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} \right) p(\mathbf{z}) d\mathbf{z} = \\ &= \int (-\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}))) p(\mathbf{z}) d\mathbf{z} \end{aligned}$$

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Proof

$$\begin{aligned}\frac{\partial p_t(\mathbf{x})}{\partial t} &= \frac{\partial}{\partial t} \int p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} = \int \left(\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} \right) p(\mathbf{z}) d\mathbf{z} = \\ &= \int (-\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}))) p(\mathbf{z}) d\mathbf{z} = \\ &= -\text{div} \left(\int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} \right)\end{aligned}$$

Flow Matching

$$\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} = -\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})),$$

Theorem

The following vector field generates the probability path $p_t(\mathbf{x})$:

$$\mathbf{f}(\mathbf{x}, t) = \mathbb{E}_{p_t(\mathbf{z}|\mathbf{x})} \mathbf{f}(\mathbf{x}, \mathbf{z}, t) = \int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) \frac{p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z})}{p_t(\mathbf{x})} d\mathbf{z}$$

Proof

$$\begin{aligned}\frac{\partial p_t(\mathbf{x})}{\partial t} &= \frac{\partial}{\partial t} \int p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} = \int \left(\frac{\partial p_t(\mathbf{x}|\mathbf{z})}{\partial t} \right) p(\mathbf{z}) d\mathbf{z} = \\ &= \int (-\text{div}(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}))) p(\mathbf{z}) d\mathbf{z} = \\ &= -\text{div} \left(\int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} \right) = -\text{div}(\mathbf{f}(\mathbf{x}, t) p_t(\mathbf{x}))\end{aligned}$$

Flow Matching

Flow Matching (FM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Conditional Flow Matching (CFM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x}|\mathbf{z})} \|\mathbf{f}(\mathbf{x}, \mathbf{z}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Flow Matching

Flow Matching (FM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Conditional Flow Matching (CFM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x}|\mathbf{z})} \|\mathbf{f}(\mathbf{x}, \mathbf{z}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Theorem

If $\text{supp}(p_t(\mathbf{x})) = \mathbb{R}^m$, then the optimal value of the FM objective equals the optimal value of the CFM objective.

Flow Matching

Flow Matching (FM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x})} \|\mathbf{f}(\mathbf{x}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Conditional Flow Matching (CFM)

$$\mathbb{E}_{t \sim U[0,1]} \mathbb{E}_{\mathbf{z} \sim p(\mathbf{z})} \mathbb{E}_{\mathbf{x} \sim p_t(\mathbf{x}|\mathbf{z})} \|\mathbf{f}(\mathbf{x}, \mathbf{z}, t) - \mathbf{f}_\theta(\mathbf{x}, t)\|^2 \rightarrow \min_{\theta}$$

Theorem

If $\text{supp}(p_t(\mathbf{x})) = \mathbb{R}^m$, then the optimal value of the FM objective equals the optimal value of the CFM objective.

Proof

This can be proved in a similar way as in the denoising score matching theorem.

Summary

- ▶ Score matching (NCSN) and diffusion models (DDPM) are discretizations of SDEs (variance exploding and variance preserving).
- ▶ It's possible to train continuous-in-time score-based generative models using forward and reverse SDEs.
- ▶ Discretizing the reverse SDE yields ancestral sampling of the DDPM.
- ▶ Flow matching suggests fitting the vector field directly.
- ▶ Conditional flow matching introduces the latent variable z , reformulating the initial task in terms of conditional dynamics.