Deep Generative Models

Lecture 12

Roman Isachenko

AI Masters

2025, Spring

Recap of previous lecture

Neural ODE

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

$x(t_0) \sim p_0(x_0)$

Theorem (continuity equation)

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

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

Solution of continuity equation

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

$\partial\theta$

- This solution will give us the density along the trajectory (not the total probability path).
- But it is difficult to estimate the last term efficiently.

Chen R. T. Q. et al. Neural Ordinary Differential Equations, 2018

Recap of previous lecture

$$\frac{x_{t+\Delta t} - x_t}{\Delta t} = f$$

SDE basics

Let define stochastic process $\mathbf{x}(t)$ with initial condition $\mathbf{x}(0) \sim p_0(\mathbf{x})$:

$p_t(x)$

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

Euler

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

Recap of previous lecture

Theorem (Kolmogorov-Fokker-Planck)

Evolution of the distribution $p_t(\mathbf{x})$ is given by the following equation:

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

Langevin SDE (special case)

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

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

Langevin dynamics

Samples from the following dynamics will comes from $p(\mathbf{x}|\boldsymbol{\theta})$ under mild regularity conditions for small enough η.

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

Welling M. Bayesian Learning via Stochastic Gradient Langevin Dynamics, 2011

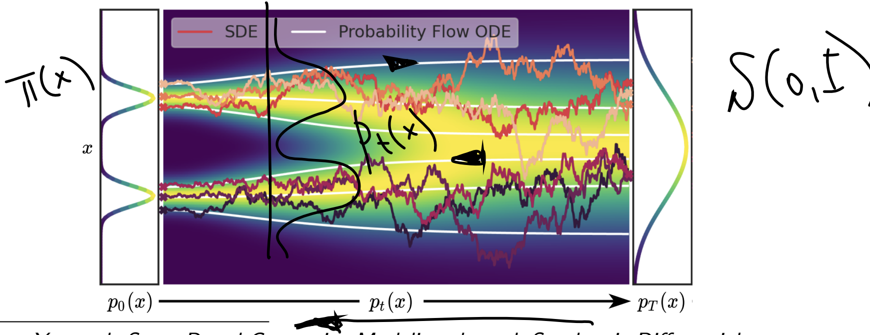
Recap of previous lecture

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

Probability flow ODE

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

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



Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

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

$p_t(x)$

There exists the reverse SDE for the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ that has the following form

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

Sketch of the proof

$$dx = \underbrace{f\, dt} + g(t)\, dw$$

- Convert initial SDE to probability flow ODE.
- Revert probability flow ODE.
- Convert reverse probability flow ODE to reverse SDE.

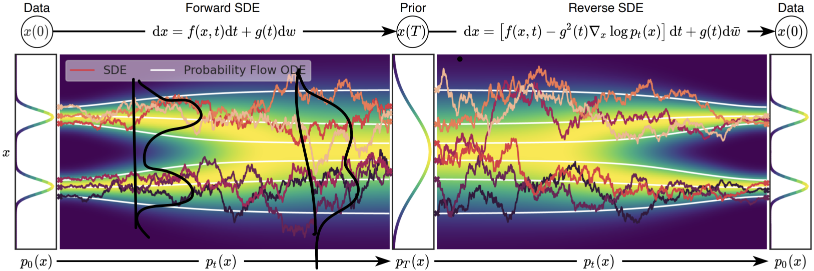
Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Recap of previous lecture

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

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

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



Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Outline

1. Diffusion and Score matching SDEs
2. Score-based generative models through SDEs
3. Flow Matching
4. Conditional Flow Matching

Outline

Score matching SDE

$dx = x(t) - x(t - dt)$

$f(x, t)dt + g(t)d$

Denoising score matching

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

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

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

Let turn this Markov chain to the continuous stochastic process $\mathbf{x}(t)$ taking $T \to \infty$:

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

Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Score matching SDE

$\sigma(t)$

$\mathcal{N}(0, \sigma^2 I)$

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

Denois.
sc. match

$\boxed{\text{Variance Exploding}}$ SDE

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

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

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

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

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

Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Diffusion SDE

$dx = \boxed{f\,dt} + \boxed{g\,dw}$

Denoising Diffusion

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

Let turn this Markov chain to the continuous stochastic process taking $T \to \infty$ and taking $\beta_t = \beta(\frac{t}{T}) \cdot \frac{1}{T}$ (with $dt = \frac{1}{T}$)

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

Variance Preserving SDE

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

Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

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 if $\mathbf{x}(0)$ has a unit variance.

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

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

Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Diffusion SDE

Euler

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

Diff.

Score Matching

VAE-ELBO

SDEs

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

DPM++

- Converting SDEs to PF-ODEs gives us the more efficient inference.

Runge Heun

- We can apply any ODESolve procedure to reduce the number of inference steps.

5-10

- In practice it reduces from 100-1000 steps to 20-50 steps.

Lu C. et al. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps, 2022

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

$\mathcal{N}(\alpha_t x_t, \sqrt{1-\alpha_t}^2)$

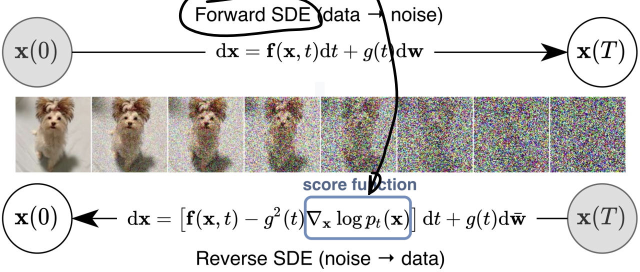
Discrete-in-time objective

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

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

Continuous-in-time objective

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



Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Score-based generative models through SDEs

$dx = f\,dt + g\,dw$

Continuous-in-time objective

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

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

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

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

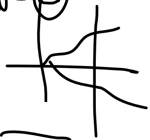
$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)}\cdot d\mathbf{w} \text{ — Variance Preserving SDE}$$

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

Särkkä S., Solin A. Applied stochastic differential equations, 2019

Score-based generative models through SDEs

$dw = \varepsilon \sqrt{dt}$



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

Theorem

Moments of the SDE $d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}$ satisfies the equations

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

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

Proof

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

Särkkä S., Solin A. Applied stochastic differential equations, 2019

Score-based generative models through SDEs

$$\frac{d\mu}{\mu} = -\frac{1}{2}\beta dt$$

$$\frac{d \ln \mu}{dt} = -\frac{1}{2}\beta$$

Theorem

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

Proof (continued)

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

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

Examples

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

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

$$\alpha_t = \prod \beta_s$$

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

Särkkä S., Solin A. Applied stochastic differential equations, 2019

Score-based generative models through SDEs

$$-\frac{x-\mu}{\sigma}$$

Training

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

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

NCSN

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

DDPM

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

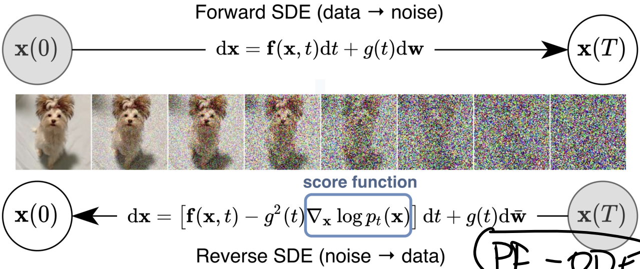
Here we omit the derivations of the variance.

Song Y., et al. Score-Based Generative Modeling through Stochastic Differential Equations, 2020

Score-based generative models through SDEs

Sampling

Solve reverse SDE using numerical solvers (SDESolve).



PF - ODE

- Discretization of the reverse SDE gives us the ancestral sampling.
- Discretization of the probability flow ODE gives us deterministic sampling. DDIM

Song Y., et al. *Score-Based Generative Modeling through Stochastic Differential Equations*, 2020

Outline

1. Diffusion and Score matching SDEs
2. Score-based generative models through SDEs
3. Flow Matching
4. Conditional Flow Matching

Continuous-in-time NF

Let return to the ODE dynamic $\mathbf{x}(t)$ in time interval $t \in [0, 1]$

- $\mathbf{x}_0 \sim p_0(\mathbf{x}) = p(\mathbf{x})$, $\mathbf{x}_1 \sim p_1(\mathbf{x}) = \pi(\mathbf{x})$;
- $p(\mathbf{x})$ is a base distribution ($\mathcal{N}(0, \mathbf{I})$) and $\pi(\mathbf{x})$ is a 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)

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

- It is hard to solve continuity equation directly because of the trace part.
- There is a method (called adjoint method) that solves this equation directly, but it is unstable and not scalable.

Chen R. T. Q. et al. Neural Ordinary Differential Equations, 2018

Continuous-in-time NF

KFP theorem (continuity equation)

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

- If we know the vector field $\boxed{\mathbf{f}(\mathbf{x}, t)}$ then KFP (or continuity) equation gives us the way to compute the density $p_t(\mathbf{x})$.
- Flow matching gives the alternative way to solve the NeuralODE.

Flow Matching

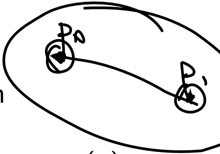
y true

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

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

Lipman Y., et al. Flow Matching for Generative Modeling, 2022

Flow Matching $\rightarrow$ Bridge Match.



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

- There exists infinite number of possible $\mathbf{f}(\mathbf{x},t)$ between $\pi(\mathbf{x})$ 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 makes the objective tractable.

$KL(p \| W$

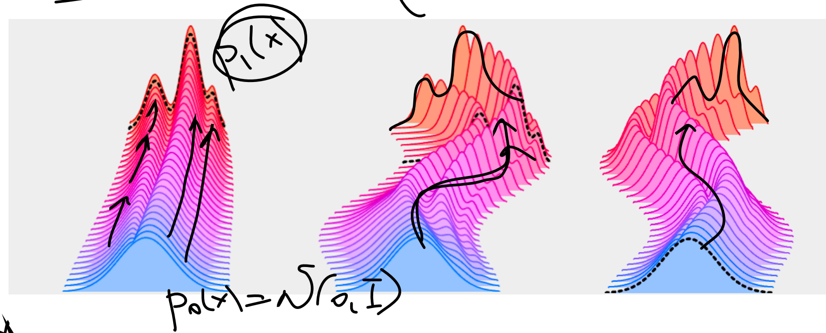


image credit: A Visual Dive into Conditional Flow Matching

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 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 KFP theorem

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

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 is the relationship between $\mathbf{f}(\mathbf{x}, t)$ and $\mathbf{f}(\mathbf{x}, \mathbf{z}, t)$?

Tong A., et al. Improving and Generalizing Flow-Based Generative Models with Minibatch Optimal Transport, 2023

Flow Matching

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

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

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

$$= \int \left(-\mathrm{div}\left(\mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z})\right)\right) p(\mathbf{z}) d\mathbf{z} =$$

$$= -\mathrm{div}\left(\int \mathbf{f}(\mathbf{x}, \mathbf{z}, t) p_t(\mathbf{x}|\mathbf{z}) p(\mathbf{z}) d\mathbf{z} \right) = -\mathrm{div}\left(\mathbf{f}(\mathbf{x}, t) p_t(\mathbf{x})\right)$$

Tong A., et al. Improving and Generalizing Flow-Based Generative Models with Minibatch Optimal Transport, 2023

Flow Matching

$$\frac{dx}{dt} = f(x, t) \longleftrightarrow f(x, z, t)$$

$$S(x, t) = \nabla \log p(x, t)$$

Flow Matching (FM)

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

θ^*

Conditional Flow Matching (CFM)

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

θ^*

$KL(p_\theta \| \pi)$

Theorem

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

Proof

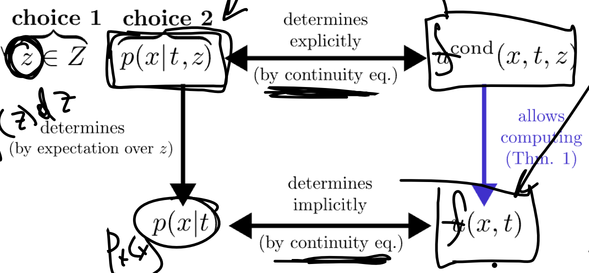
It is proved similarly with the denoising score matching theorem.

Tong A., et al. Improving and Generalizing Flow-Based Generative Models with Minibatch Optimal Transport, 2023

Conditional Flow Matching

$\|f - f_\theta\|^2$

$\int p(z)dz$



$\int p_t(z|x)$

$p_t(x)$

$\|f - f_\theta\|^2$

- We do not want to model $p_t(\mathbf{x})$ because it is complex.
- We showed that it is possible to solve CFM task instead of FM task.
- Let choose the convenient conditioning latent variable $\mathbf{z}$.
- Let parametrize $\int p_t(\mathbf{x}|\mathbf{z}) p(z)dz$ instead of $p_t(\mathbf{x})$. It must satisfy the following constraints

$$p(\mathbf{x}) = \mathcal{N}(0, \mathbf{I}) = \mathbb{E}_{p(\mathbf{z})} p_0(\mathbf{x}|\mathbf{z}); \quad \pi(\mathbf{x}) = \mathbb{E}_{p(\mathbf{z})} p_1(\mathbf{x}|\mathbf{z}).$$

image credit: A Visual Dive into Conditional Flow Matching

Summary

- Score matching (NCSN) and diffusion models (DDPM) are the discretizations of the SDEs (variance exploding and variance preserving).
- It is possible to train the continuous-in-time score-based generative models through forward and reverse SDEs.
- Discretization of the reverse SDE gives the ancestral sampling of the DDPM.
- Flow matching suggests to fit the vector field directly.
- Conditional flow matching introduces the latent variable $\mathbf{z}$ to reformulate the initial task in terms of conditional dynamics.