Mathematical Dissection of Diffusion-based Generative Models

chao ji

Abstract

Diffusion-based deep generative models have become the de facto standard option for creating diverse and realistic content across a wide range of data modalities. The progress of research in diffusion models is driven in part by a deeper understanding of connections and equivalences of multiple seemingly distinct formalisms and an effort to unify them into a common framework. Among these, Stochastic Differential Equation (SDE) is quite favorable since it links other work such as denoising diffusion model and score matching w/ Langevin Dynamics to the problem of numerical solving differential equations from which well-developed theory and practice can be borrowed. However, SDE involves advanced concepts and techniques that many deep learning practitioners may not be proficient or familiar with. The goal of this document is to provide a beginner-friendly introduction from the perspective of *SDE* and *stochastic process*. The first two sections are dedicated to explaining why Fokker-Planck Equation describes the time evolution of stochastic process induced by an SDE. In later sections, the main results of several landmark papers will be presented in a unified and streamlined fashion to reveal their interconnections and the intuition behind some design choices. The hope is that this will enable readers to be in a better position to understand other important papers in this field.

Overview

We begin our discussion by briefly introducing Stochastic Differential Equation (SDE) in Section 1. Section 2 provides a detailed derivation of Fokker-Planck Equation (FPE) which is key to understanding the probabilistic behavior of sampling techniques such as Probability Flow (PF) ODE. An alternative derivation of FPE is also presented in Section 4. In Section 3 and 5, we prove the marginal preserving property of PF ODE and reverse-time SDE. In Section 6, 7 and 8 we focus on a subset of SDEs with affine drift and diffusion coefficient which allows the mean and covariance of the induced stochastic process to be expressed analytically. Section 9 discuss how to estimate the score function. In Section 10 we present the framework that unifies PF ODE and SDE. Several advanced techniques are introduced in Section 11 (Flow Matching), Section 12 (Consistency Model), and Section 13 (Neural ODE).

1 A (mini) Primer on SDE

A typical Stochastic Differential Equation (SDE) takes the form

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t + \mathbf{G}(\mathbf{x}, t)\mathrm{d}\mathbf{w} \tag{1}$$

where

- $\mathbf{x} \in \mathbb{R}^n$, $t \in \mathbb{R}_+$ and $\mathbf{w} \in \mathbb{R}^n$ are the variables of this equation.

- $\mathbf{f}(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$ (drift coefficient) and $\mathbf{G}(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^{n \times n}$ (diffusion coefficient) are deterministic functions of $\mathbf{x}$ and t.
- The differential notation "d" in front of $\mathbf{x}$, t and $\mathbf{w}$ denotes the infinitesimally small change of the variable.
- $\mathbf{w}$ is the standard *Wiener Process* (aka Brownian Motion). In short, a Wiener process starts with the value of zero, has independent increments, and the increments are normally distributed: $\mathrm{d}\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathrm{d}t\mathbf{I})$.
- $\mathbf{x}$ and $\mathbf{w}$ are technically functions of t (denoted as $\mathbf{x}_t$ or $\mathbf{x}(t)$). We leave out t in the notation unless we want to explicitly indicate the dependence on it.
- $\mathbf{x}$ is commonly referred to as the *state variable* and t the *time variable*. Examples of $\mathbf{x}$: stock prices, positions of particles in a thermodynamic system, etc.
- t is varies continuously from 0 to T for $T > 0$.

Intuitively the SDE describes how to quantitatively relate the instantaneous increment $\mathrm{d}\mathbf{x}$ of the variable $\mathbf{x}$ with its current value at time t. We can run Monte Carlo simulation[1] to sample a sequence of $\mathbf{x}$: starting at time $t = 0$ with $\mathbf{x}$ drawn from some initial distribution, we substitute them into the RHS of the equation. By choosing a (small) finite value of $\mathrm{d}t$ and then sampling $\mathrm{d}\mathbf{w}$ according to $\mathcal{N}(\mathbf{0}, \mathrm{d}t\mathbf{I})$, we can compute $\mathrm{d}\mathbf{x}$ and use it to get the updated value of the state variable $\mathbf{x} + \mathrm{d}\mathbf{x}$ and time variable $t + \mathrm{d}t$. Further, by repeating this process we end up with a trajectory of $\mathbf{x}$'s for a sequence of time points.

Note that because of the stochasticity of the Gaussian noise $\mathrm{d}\mathbf{w}$ every step of the way and the uncertainty of $\mathbf{x}$'s initial value, we are unable to predict the exact path $\mathbf{x}$ will go through. But if we were to run a large number of such simulations, chances are[2] that the values of $\mathbf{x}$ at a given time t would trace out some probability distribution $p(\mathbf{x}, t)$. If this is true, we say that the stochastic process with probability density $p(\mathbf{x}, t)$ is *induced* by the SDE, or put differently, the stochastic process *solves* the SDE.

2 Fokker-Planck Equation

SDE seems to describe only the local dynamics of a single instance of $\mathbf{x}$ — how it behaves in response to an instantaneous change of time. In contrast, the global view of how an ensemble of them evolve over time can be provided by the path of marginal probability density $p(\mathbf{x}, t)$, which is characterized by an important equation we will derive next.

First we rewrite the SDE (Eq. (1)) as finite difference

$$\Delta\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\Delta t + \mathbf{G}(\mathbf{x}, t)\Delta\mathbf{w}$$

where the differential notation is replaced with Δ, meaning a super small but finite difference of the variable.

Suppose we have an arbitrary function $\varphi(\mathbf{x}, t) : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$ that is twice-differentiable in $\mathbf{x}$ and t. We consider the finite difference of φ by a small displacement around the inputs, that is

$$\Delta\varphi(\mathbf{x}, t) = \varphi(\mathbf{x} + \Delta\mathbf{x}, t + \Delta t) - \varphi(\mathbf{x}, t)$$

[1] It's called Euler method for numerically solving differential equations.

[2] Under the condition that $\mathbf{f}$ and $\mathbf{G}$ are Lipschitz continuous (i.e. the slope w.r.t. any two inputs is bounded)

Expand $\varphi(\mathbf{x}+\Delta\mathbf{x}, t+\Delta t)$ up to second derivative of a Taylor series around $\mathbf{x}$ and t:

$$
\begin{aligned}
&\Delta\varphi(\mathbf{x},t)\\
=&\varphi(\mathbf{x}+\Delta\mathbf{x},t+\Delta t)-\varphi(\mathbf{x},t)\\
=&\frac{\partial\varphi}{\partial t}\Delta t+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}\Delta x_i+\frac{1}{2}\left[\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}\Delta x_i\Delta x_j+\frac{\partial^2\varphi}{\partial t^2}\Delta t^2+2\sum_{i=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial t}\Delta x_i\Delta t\right]\\
=&\frac{\partial\varphi}{\partial t}\Delta t+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}\,\Delta x_i+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}\,\Delta x_i\Delta x_j+o(\Delta t)
\end{aligned}
\tag{2}
$$

Recall that it makes sense to express $\mathrm{d}\mathbf{x}$ as a linear function of $\mathrm{d}t$ (Eq. (1)) only if the change of time is infinitely close to zero (this is what the idea of differential is all about!), so all that matters here is the behavior of $\Delta\varphi(\mathbf{x},t)$ as $\Delta t\to 0$. $o(\Delta t)$ is discarded because it approaches zero *faster than* Δt.

Substituting elementwise finite difference $\Delta x_i = f_i(\mathbf{x},t)\Delta t+\sum_{k=1}^{n}G_{ik}(\mathbf{x},t)\Delta w_k$ into Eq. (2),

$$
\begin{aligned}
&\varphi(\mathbf{x}+\Delta\mathbf{x},t+\Delta t)-\varphi(\mathbf{x},t)\\
=&\frac{\partial\varphi}{\partial t}\Delta t+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}\,f_i(\mathbf{x},t)\Delta t+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}\sum_{k=1}^{n}G_{ik}(\mathbf{x},t)\Delta w_k\\
&+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}f_i(\mathbf{x},t)f_j(\mathbf{x},t)\Delta t^2+\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}f_i(\mathbf{x},t)\sum_{k=1}^{n}G_{jk}(\mathbf{x},t)\Delta w_k\Delta t\\
&+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}\left[\sum_{k=1}^{n}G_{ik}(\mathbf{x},t)\Delta w_k\right]\left[\sum_{k=1}^{n}G_{jk}(\mathbf{x},t)\Delta w_k\right]
\end{aligned}
\tag{3}
$$

Note that in Eq. (3):

- Δt^2 and $\Delta t\Delta w_i$ (for $i=1,\cdots,n$) are $o(\Delta t)$ because

$$
\lim_{\Delta t\to 0}\frac{\Delta t^2}{\Delta t}=\lim_{\Delta t\to 0}\Delta t=0
$$
$$
\lim_{\Delta t\to 0}\frac{\Delta t\Delta w_i}{\Delta t}=\lim_{\Delta t\to 0}\Delta w_i=\mathbb{E}[\Delta w_i]=0
$$

- $\Delta w_i\Delta w_j$ are random variables that vary around their means. As $\Delta t\to 0$, we are essentially squashing the range in which they can vary and eventually they will be forced to their means — if $i\neq j$, $\Delta w_i\Delta w_j\to\mathbb{E}[\Delta w_i\Delta w_j]=0$; if $i=j$, $\Delta w_i\Delta w_j\to\mathbb{E}[\Delta w_i\Delta w_j]=\Delta t$ ($\Delta w_i\Delta w_j$ will first converge to Δt, before it goes on to converge to zero).

Discarding $o(\Delta t)$ and the terms with zero expectation, and replacing the finite difference notation Δ with the differential notation d,

$$
\mathrm{d}\varphi(\mathbf{x},t)=\left[\frac{\partial\varphi}{\partial t}+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}f_i(\mathbf{x},t)+\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2\varphi}{\partial x_i\partial x_j}\sum_{k=1}^{n}G_{ik}(\mathbf{x},t)G_{jk}(\mathbf{x},t)\right]\mathrm{d}t+\sum_{i=1}^{n}\frac{\partial\varphi}{\partial x_i}\sum_{k=1}^{n}G_{ik}(\mathbf{x},t)\mathrm{d}w_k
\tag{4}
$$

This is an SDE (*cf.* Eq. (1)) called **Ito's formula (lemma)**. It provides the first-order approximation of the dynamics of the function $\varphi(\mathbf{x}, t)$.

A time-independent version of Ito's formula can likewise be derived where the function $\varphi(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ is independent of t:

$$\mathrm{d}\varphi(\mathbf{x}) = \underbrace{\left[\sum_{i=1}^{n} \frac{\partial \varphi}{\partial x_i} f_i(\mathbf{x}, t) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 \varphi}{\partial x_i \partial x_j} \sum_{k=1}^{n} G_{ik}(\mathbf{x}, t) G_{jk}(\mathbf{x}, t)\right]}_{\text{drift}} \mathrm{d}t + \underbrace{\sum_{i=1}^{n} \frac{\partial \varphi}{\partial x_i} \sum_{k=1}^{n} G_{ik}(\mathbf{x}, t)}_{\text{diffusion}} \mathrm{d}w_k \tag{5}$$

We can tell from Eq. (5) that the randomness of $\mathrm{d}\varphi(\mathbf{x})$ comes from both $\mathbf{x}$ and $\mathrm{d}\mathbf{w}$ — we first sample $\mathbf{x}$ from its marginal density $p(\mathbf{x}, t)$ to compute the drift and diffusion coefficients, and then combine them with $\mathrm{d}\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathrm{d}t\mathbf{I})$, which is independent of $\mathbf{x}$, to compute $\mathrm{d}\varphi(\mathbf{x})$.

Luckily, the randomness of $\mathrm{d}\varphi(\mathbf{x})$ can be averaged out by considering its *expected* value: we take expectation with respect to $\mathbf{x}$ and $\mathrm{d}\mathbf{w}$ — the source of randomness — on both sides and divide by $\mathrm{d}t$:

$$\begin{aligned}
\text{LHS} &= \frac{1}{\mathrm{d}t} \mathbb{E}_{\mathbf{x}, \mathrm{d}\mathbf{w}} \left[\mathrm{d}\varphi(\mathbf{x})\right] \\
&= \frac{1}{\mathrm{d}t} \int \mathrm{d}\varphi(\mathbf{x}) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} \\
&= \frac{\mathrm{d}}{\mathrm{d}t} \int \varphi(\mathbf{x}) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} \\
&= \int \varphi(\mathbf{x}) \frac{\partial}{\partial t} p(\mathbf{x}, t) \mathrm{d}\mathbf{x} \quad \text{// bring } \frac{\mathrm{d}}{\mathrm{d}t} \text{ inside of } \int \text{ by Leibniz integral rule}
\end{aligned}$$

and

$$\begin{aligned}
\text{RHS} &= \mathbb{E}_{\mathbf{x}, \mathrm{d}\mathbf{w}} \left[\text{drift term}\right] + \mathbb{E}_{\mathbf{x}, \mathrm{d}\mathbf{w}} \left[\text{diffusion term}\right] \\
&= \mathbb{E}_{\mathbf{x}} \left[\text{drift term}\right] + \mathbb{E}_{\mathrm{d}\mathbf{w}} \left[\text{diffusion term}\right] \\
&= \mathbb{E}_{\mathbf{x}} \left[\text{drift term}\right] \quad \text{// diffusion term vanishes because } \mathrm{d}w_k \text{ are *expected* to be zero} \\
&= \int \left[\sum_{i=1}^{n} \frac{\partial \varphi}{\partial x_i} f_i(\mathbf{x}, t) + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 \varphi}{\partial x_i \partial x_j} D_{ij}(\mathbf{x}, t)\right] p(\mathbf{x}, t) \mathrm{d}\mathbf{x}
\end{aligned}$$

where we denote $D_{ij}(\mathbf{x}, t) = \sum_{k=1}^{n} G_{ik}(\mathbf{x}, t) G_{jk}(\mathbf{x}, t)$ or $\mathbf{D}(\mathbf{x}, t) = \mathbf{G}(\mathbf{x}, t)\mathbf{G}^\top(\mathbf{x}, t)$ in matrix form.

Matching LHS with RHS, we are left with

$$\int \varphi(\mathbf{x}) \frac{\partial}{\partial t} p(\mathbf{x}, t) \mathrm{d}\mathbf{x} = \sum_{i=1}^{n} \int \frac{\partial \varphi}{\partial x_i} f_i(\mathbf{x}, t) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \int \frac{\partial^2 \varphi}{\partial x_i \partial x_j} D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} \tag{6}$$

We will shift the partial derivative operator from $\varphi(\mathbf{x})$ to $f_i(\mathbf{x}, t)p(\mathbf{x}, t)$ and $D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)$ using integration by parts:

Recall the rule of integration by parts for multivariate function:

$$\int_\Omega v(\mathbf{x}) \frac{\partial}{\partial x_i} u(\mathbf{x}) \mathrm{d}\mathbf{x} = \int_\Omega \frac{\partial}{\partial x_i} \left[u(\mathbf{x}) v(\mathbf{x})\right] \mathrm{d}\mathbf{x} - \int_\Omega u(\mathbf{x}) \frac{\partial}{\partial x_i} v(\mathbf{x}) \mathrm{d}\mathbf{x}$$

$$= \int_{\partial\Omega} u(\mathbf{x})v(\mathbf{x})n_i \mathrm{d}S - \int_{\Omega} u(\mathbf{x})\frac{\partial}{\partial x_i}v(\mathbf{x})\mathrm{d}\mathbf{x}$$
$$= -\int_{\Omega} u(\mathbf{x})\frac{\partial}{\partial x_i}v(\mathbf{x})\mathrm{d}\mathbf{x} \tag{7}$$

For the 1st integral in RHS of Eq. (6), let $u(\mathbf{x}) = \varphi(\mathbf{x})$ and $v(\mathbf{x}) = f_i(\mathbf{x}, t)p(\mathbf{x}, t)$, then apply Eq. (7),

$$\int \frac{\partial \varphi}{\partial x_i} f_i(\mathbf{x}, t)p(\mathbf{x}, t)\mathrm{d}\mathbf{x} = -\int \varphi(\mathbf{x})\frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} \tag{8}$$

Then apply Eq. (7) to the 2nd term twice,

$$\int \frac{\partial^2 \varphi}{\partial x_i \partial x_j} D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\mathrm{d}\mathbf{x} = \int \varphi(\mathbf{x})\frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} \tag{9}$$

Combining Eq. (8) and (9) with Eq. (6),

$$\int \varphi(\mathbf{x})\frac{\partial}{\partial t}p(\mathbf{x}, t)\mathrm{d}\mathbf{x} = -\int \varphi(\mathbf{x})\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} + \int \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\varphi(\mathbf{x})\frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} \tag{10}$$

The only way for Eq. (10) to hold for any function $\varphi(\mathbf{x})$ is the following:

$$\frac{\partial}{\partial t}p(\mathbf{x}, t) = -\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right] \tag{11}$$

This is a Partial Differential Equation (PDE) commonly referred to as **Fokker-Planck Equation** (FPE) in physics literature (aka **Komolgorov Forward Equation** in stochastic process literature). It states that the rate of change of the marginal density $p(\mathbf{x}, t)$ at a given point $\mathbf{x}$ w.r.t. time t can be explained by the negative divergence of the vector field $\mathbf{f}(\mathbf{x}, t)p(\mathbf{x}, t)$ at that point (i.e. the net *inward* flow of probability mass), plus the difference caused by diffusion (like heat dissipation).

The following theorem summarizes the results we have so far:

Theorem 1. *The probability density $p(\mathbf{x}, t)$ of the solution (i.e. a stochastic process) to the SDE in Eq. (1) solves the PDE in Eq. (11)*

It's best to view SDE, FPE and their connection through the lens of physics. Consider a large number of particles bouncing back and forth under the influence of random forces. While SDE gives the dynamics of a single particle, FPE depicts the behavior of all of them in an aggregate manner. In particular, SDE's drift coefficient plays the role of the particle's *deterministic* instantaneous velocity. Along with the noise from the diffusion term, they contribute to the particle's instantaneous but *stochastic* displacement.

3 Probability Flow ODE

When simulating SDE, randomness is introduced at both the initial value and the intermediate update steps. In some cases, however, it would be desirable to not inject any noise during the course of simulation so that the trajectory will be deterministic and thus invertible. Paradoxically,

despite each trajectory being deterministic, an ensemble of them as a whole would still behave probabilistically/unpredictably because of their *random* initial conditions. Next we are going to show it is possible to construct such a modified simulation process without altering its probability density $p(\mathbf{x}, t)$.

First we transform the FPE in Eq. (11) into a form where the second derivatives are absorbed into the gradient term:

$$
\begin{aligned}
&\frac{\partial}{\partial t} p(\mathbf{x}, t) \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) \right] + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j} \left[D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \right] \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) \right] + \frac{1}{2} \sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[\sum_{j=1}^{n} \frac{\partial}{\partial x_j} \left[D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \right] \right] \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \frac{\partial}{\partial x_j} \left[p(\mathbf{x}, t) D_{ij}(\mathbf{x}, t) \right] \right] \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \left[D_{ij}(\mathbf{x}, t) \frac{\partial}{\partial x_j} p(\mathbf{x}, t) + p(\mathbf{x}, t) \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t) \right] \right] \quad \text{// product rule} \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \left[D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \frac{\partial}{\partial x_j} \log p(\mathbf{x}, t) + p(\mathbf{x}, t) \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t) \right] \right] \quad \text{// log derivative trick} \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[\left[f_i(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \left[D_{ij}(\mathbf{x}, t) \frac{\partial}{\partial x_j} \log p(\mathbf{x}, t) + \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t) \right] \right] p(\mathbf{x}, t) \right] \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[\left[f_i(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} D_{ij}(\mathbf{x}, t) \frac{\partial}{\partial x_j} \log p(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t) \right] p(\mathbf{x}, t) \right] \\
&= -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[\tilde{f}_i(\mathbf{x}, t) p(\mathbf{x}, t) \right] \qquad (12)
\end{aligned}
$$

the expression in the inner brackets is denoted as a vector field $\tilde{\mathbf{f}}(\mathbf{x}, t) : \mathbb{R}^n \to \mathbb{R}^n$ of $\mathbf{x}$:

$$
\tilde{f}_i(\mathbf{x}, t) = f_i(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} D_{ij}(\mathbf{x}, t) \frac{\partial}{\partial x_j} \log p(\mathbf{x}, t) - \frac{1}{2} \sum_{j=1}^{n} \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t) \qquad (13)
$$

for $i = 1, 2, \cdots, n$. Or expressed in matrix form as

$$
\tilde{\mathbf{f}}(\mathbf{x}, t) = \mathbf{f}(\mathbf{x}, t) - \frac{1}{2} \mathbf{D}(\mathbf{x}, t) \nabla_{\mathbf{x}} \log p(\mathbf{x}, t) - \frac{1}{2} \nabla_{\mathbf{x}} \cdot \mathbf{D}(\mathbf{x}, t)
$$

where the vector field $\nabla_{\mathbf{x}} \log p(\mathbf{x}, t)$ is referred to as the *score function* of $p(\mathbf{x}, t)$, and $\nabla_{\mathbf{x}} \cdot \mathbf{D}(\mathbf{x}, t)$

denotes the *divergence* operator $\nabla\cdot$ applied to each row of $\mathbf{D}(\mathbf{x},t)$:

$$\nabla_{\mathbf{x}}\cdot\mathbf{D}(\mathbf{x},t)=\begin{bmatrix}\frac{\partial}{\partial x_1}D_{11}(\mathbf{x},t)+\frac{\partial}{\partial x_2}D_{12}(\mathbf{x},t)+\cdots+\frac{\partial}{\partial x_n}D_{1n}(\mathbf{x},t)\\ \frac{\partial}{\partial x_1}D_{21}(\mathbf{x},t)+\frac{\partial}{\partial x_2}D_{22}(\mathbf{x},t)+\cdots+\frac{\partial}{\partial x_n}D_{2n}(\mathbf{x},t)\\ \vdots\\ \frac{\partial}{\partial x_1}D_{n1}(\mathbf{x},t)+\frac{\partial}{\partial x_2}D_{n2}(\mathbf{x},t)+\cdots+\frac{\partial}{\partial x_n}D_{nn}(\mathbf{x},t)\end{bmatrix}=\begin{bmatrix}\sum_{j=1}^{n}\frac{\partial}{\partial x_j}D_{1j}(\mathbf{x},t)\\ \sum_{j=1}^{n}\frac{\partial}{\partial x_j}D_{2j}(\mathbf{x},t)\\ \vdots\\ \sum_{j=1}^{n}\frac{\partial}{\partial x_j}D_{nj}(\mathbf{x},t)\end{bmatrix}$$

Now let's consider an Ordinary Differential Equation (ODE) defined in terms of $\tilde{\mathbf{f}}(\mathbf{x},t)$:

$$\mathrm{d}\mathbf{x}=\tilde{\mathbf{f}}(\mathbf{x},t)\mathrm{d}t \tag{14}$$

which can also be interpreted as an SDE where the diffusion coefficient $\tilde{\mathbf{G}}(\mathbf{x},t)$ is set to zero. Let $\tilde{p}(\mathbf{x},t)$ denote the marginal density of the stochastic process induced by Eq. (14) and assume the initial value of Eq. (14) is drawn from the same distribution as Eq. (1), i.e. $\tilde{p}(\mathbf{x},0)=p(\mathbf{x},0)$. According to Theorem (1), $\tilde{p}(\mathbf{x},t)$ must solve the following PDE without the second derivatives[3]

$$\frac{\partial}{\partial t}\tilde{p}(\mathbf{x},t)=-\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\left[\tilde{f}_i(\mathbf{x},t)\tilde{p}(\mathbf{x},t)\right] \tag{15}$$

Obviously Eq. (15) is exactly the same as Eq. (12). Assuming the regularity of $\tilde{\mathbf{f}}$ (i.e. being Lipschitz continuous), Eq. (12 and 15) have unique solution, therefore $\tilde{p}(\mathbf{x},t)=p(\mathbf{x},t)$ for $t>0$. That is, for SDE in Eq. (1) there is a deterministic ODE that induces a stochastic process with the same probability density. We referred to Eq. (14) as the *Probability Flow* (PF) ODE.

4 More on Fokker-Planck Equation

In the derivation of FPE we alluded to the point that the rate of change of probability density depends on the net influx of probability mass. In this section, we show an alternative derivation of FPE from the *master equation* that explicitly states the conservation of probability mass.

Master Equation

The master equation describes the temporal evolution of the probability density function $p(\mathbf{x},t)$ of a stochastic process

$$\frac{\partial}{\partial t}p(\mathbf{x},t)=\int\left[w_t(\mathbf{x}|\mathbf{x}')p(\mathbf{x}',t)-w_t(\mathbf{x}'|\mathbf{x})p(\mathbf{x},t)\right]\mathrm{d}\mathbf{x}' \tag{16}$$

where

- $\mathbf{x}$ and $\mathbf{x}'$ are variables in the state space $\Omega\subseteq\mathbb{R}^n$.
- t is a continuous scalar variable in the time space.
- $p(\mathbf{x},t)$ is marginal probability of $\mathbf{x}$ at time t.

[3]Also known as the *Continuity Equation* or *Liouville Equation*.

- $w_t(\mathbf{x}|\mathbf{x}')$ (a function of $\mathbf{x}$, $\mathbf{x}'$ and t) is the transition probability density, in an infinitesimal time step Δt, from state $\mathbf{x}'$ to $\mathbf{x}$ at time t (likewise $w_t(\mathbf{x}'|\mathbf{x})$ from $\mathbf{x}$ to $\mathbf{x}'$). Formally,

$$w_t(\mathbf{x}|\mathbf{x}') = \lim_{\Delta t \to 0} \frac{1}{\Delta t} p(\mathbf{x}, t + \Delta t|\mathbf{x}', t)$$
$$w_t(\mathbf{x}'|\mathbf{x}) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} p(\mathbf{x}', t + \Delta t|\mathbf{x}, t)$$

  where $p(\mathbf{x}, t+\Delta t|\mathbf{x}', t)$ is the probability density of the state variable being in $\mathbf{x}$ at $t+\Delta t$ given that it was in $\mathbf{x}'$ at t.

- the marginal density $p(\mathbf{x}, t)$ and transition density $p(\mathbf{x}, t + \Delta t|\mathbf{x}', t)$ and $p(\mathbf{x}', t + \Delta t|\mathbf{x}, t)$ are functions of the state and time variable $\mathbf{x}$ and t.

The first and second term on the RHS of master equation can be interpreted as the "inflow" and "outflow" of probability mass at time t. Intuitively, master equation states that the rate of change in probability density is the result of the difference between inflow and outflow.

Intuitive Derivation

Here is an intuitive (from the perspective of physics) derivation of the master equation, assuming the variable x is in 1D for ease of presentation.

First, the probability *mass* at time t of the variable being in x can be expressed as the product of the "density" $p(x, t)$ and a very small "volume" Δx. Likewise, after a small change Δt in time, the same quantity becomes $p(x, t + \Delta t)\Delta x$. By definition, the change in probability mass is equal to $p(x, t + \Delta t)\Delta x - p(x, t)\Delta x$.

As previously pointed out, the change is attributed to the difference between the amount of mass that flows in and that flows out after a small change in time Δt:

$$\text{inflow} = \sum_{x'} p(x', t)\Delta x \cdot p(x, t + \Delta t|x', t)\Delta x$$
$$\text{outflow} = \sum_{x'} p(x, t)\Delta x \cdot p(x', t + \Delta t|x, t)\Delta x$$

Note that for the inflow equation,

- probability mass flows from x' to x.
- $p(x', t)\Delta x$ is the probability mass of the variable being in x' at time t.
- $p(x, t + \Delta t|x', t)\Delta x$, which is again a "density" multiplied by a "volume" Δx, describes the conditional probability (mass) that the variable is in x at $t + \Delta t$ given that it was in x' at t. A simple analogy: for event A and B where $P(A, B) = P(B)P(A|B)$, $p(x', t)\Delta x$ is to $P(B)$ as $p(x, t + \Delta t|x', t)\Delta x$ is to $P(A|B)$
- We sum the contribution over all possible values that x' can take.
- Assume Δx is small enough, but not equal to 0 such that it is possible to sum over a large but finite space of values that x' can take.

The same argument applies also to the outflow equation.

The two ways of expressing the same quantity must be equivalent, therefore:

$$\begin{aligned} p(x,t+\Delta t)\Delta x - p(x,t)\Delta x &= \text{inflow} - \text{outflow} \\ &= \sum_{x'} p(x',t)\Delta x \cdot p(x,t+\Delta t|x',t)\Delta x - \sum_{x'} p(x,t)\Delta x \cdot p(x',t+\Delta t|x,t)\Delta x \\ &= \sum_{x'} \left[p(x',t)\cdot p(x,t+\Delta t|x',t) - p(x,t)\cdot p(x',t+\Delta t|x,t)\right]\Delta x^2 \end{aligned}$$

Dividing both sides by $\Delta x \Delta t$, we have

$$\frac{p(x,t+\Delta t)-p(x,t)}{\Delta t} = \sum_{x'}\left[p(x',t)\cdot\frac{1}{\Delta t}p(x,t+\Delta t|x',t) - p(x,t)\cdot\frac{1}{\Delta t}p(x',t+\Delta t|x,t)\right]\Delta x$$

Now both LHS and RHS represent the approximated rate of change of probability density — probability mass normalized by Δx is density, which then becomes rate of change if normalized by Δt. If we let $\Delta t \to 0$ and $\Delta x \to 0$, the LHS becomes the derivative of density with respect to t, and the RHS becomes an integral over x' (finite sum $\sum$ becomes infinite sum $\int$):

$$\begin{aligned} \text{LHS} &= \lim_{\Delta t\to 0}\frac{p(x,t+\Delta t)-p(x,t)}{\Delta t} = \frac{\partial}{\partial t}p(x,t) \\ \text{RHS} &= \lim_{\Delta x\to 0}\sum_{x'}\left[p(x',t)\lim_{\Delta t\to 0}\frac{1}{\Delta t}p(x,t+\Delta t|x',t) - p(x,t)\lim_{\Delta t\to 0}p(x',t+\Delta t|x,t)\right]\Delta x \\ &= \int \left[p(x',t)w_t(x|x') - p(x,t)w_t(x'|x)\right]\mathrm{d}x' \end{aligned}$$

therefore

$$\frac{\partial}{\partial t}p(x,t) = \int \left[p(x',t)w_t(x|x') - p(x,t)w_t(x'|x)\right]\mathrm{d}x'$$

which is exactly the master equation in 1-D case.

Alternative Derivation of Fokker-Planck Equation From Master Equation

Let's rewrite the master equation:

$$\frac{\partial}{\partial t}p(\mathbf{x},t) = \int_\Omega \left[w_t(\mathbf{x}|\mathbf{x}')p(\mathbf{x}',t) - w_t(\mathbf{x}'|\mathbf{x})p(\mathbf{x},t)\right]\mathrm{d}\mathbf{x}'$$

where $\mathbf{x}'$ is integrated over Ω. Multiply both sides by a test function $\varphi : \Omega \to \mathbb{R}$ that is twice-differentiable in $\mathbf{x}$, and integrate over $\mathbf{x}$, we obtain

$$\text{LHS} = \int_\Omega \varphi(\mathbf{x})\frac{\partial}{\partial t}p(\mathbf{x},t)\mathrm{d}\mathbf{x}$$

and

$$\text{RHS} = \int_\Omega \varphi(\mathbf{x})\int_\Omega \left[w_t(\mathbf{x}|\mathbf{x}')p(\mathbf{x}',t) - w_t(\mathbf{x}'|\mathbf{x})p(\mathbf{x},t)\right]\mathrm{d}\mathbf{x}'\mathrm{d}\mathbf{x}$$

$$= \int_\Omega \int_\Omega \varphi(\mathbf{x}) \left[w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) - w_t(\mathbf{x}'|\mathbf{x}) p(\mathbf{x}, t) \right] \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x}$$

Now Taylor-expand the test function φ around $\mathbf{x}'$:

$$\varphi(\mathbf{x}) = \varphi(\mathbf{x}') + \nabla\varphi(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}')$$

where $\nabla\varphi(\mathbf{x}') \in \mathbb{R}^n$ and $\mathbf{H}_\varphi(\mathbf{x}') \in \mathbb{R}^{n \times n}$ are the gradient and Hessian matrix. Substituting it back into the RHS, we obtain

$$\begin{aligned}
\text{RHS} &= \int_\Omega \int_\Omega \varphi(\mathbf{x}) \left[w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) - w_t(\mathbf{x}'|\mathbf{x}) p(\mathbf{x}, t) \right] \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&= \int_\Omega \int_\Omega \varphi(\mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&\quad + \int_\Omega \int_\Omega \left[\nabla\varphi(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}') \right] w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&\quad - \int_\Omega \int_\Omega \varphi(\mathbf{x}) w_t(\mathbf{x}'|\mathbf{x}) p(\mathbf{x}, t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x}
\end{aligned}$$

The third term is a double integral over $\Omega \times \Omega$, so we can swap the variables $\mathbf{x}$ with $\mathbf{x}'$:

$$\begin{aligned}
\text{RHS} &= \cancel{\int_\Omega \int_\Omega \varphi(\mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x}} \\
&\quad + \int_\Omega \int_\Omega \left[\nabla\varphi(\mathbf{x}')^\top (\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}') \right] w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&\quad - \cancel{\int_\Omega \int_\Omega \varphi(\mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x}} \\
&= \int_\Omega \int_\Omega \left[\nabla\varphi^\top(\mathbf{x}')(\mathbf{x} - \mathbf{x}') + \frac{1}{2} (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}') \right] w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&= \int_\Omega \int_\Omega \nabla\varphi^\top(\mathbf{x}')(\mathbf{x} - \mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} + \frac{1}{2} \int_\Omega \int_\Omega (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x}
\end{aligned}$$

Expanding the matrix multiplication terms,

$$\begin{aligned}
\text{RHS} &= \int_\Omega \int_\Omega \nabla\varphi^\top(\mathbf{x}')(\mathbf{x} - \mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} + \frac{1}{2} \int_\Omega \int_\Omega (\mathbf{x} - \mathbf{x}')^\top \mathbf{H}_\varphi(\mathbf{x}') (\mathbf{x} - \mathbf{x}') w_t(\mathbf{x}|\mathbf{x}') p(\mathbf{x}', t) \mathrm{d}\mathbf{x}' \mathrm{d}\mathbf{x} \\
&= \sum_{i=1}^n \int_\Omega \left[\int_\Omega (x_i - x_i') w_t(\mathbf{x}|\mathbf{x}') \mathrm{d}\mathbf{x} \right] p(\mathbf{x}', t) \frac{\partial\varphi}{\partial x_i'} \mathrm{d}\mathbf{x}' \\
&\quad + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \int_\Omega \left[\int_\Omega (x_i - x_i')(x_j - x_j') w_t(\mathbf{x}|\mathbf{x}') \mathrm{d}\mathbf{x} \right] p(\mathbf{x}', t) \frac{\partial^2\varphi}{\partial x_i' \partial x_j'} \mathrm{d}\mathbf{x}' \\
&= \sum_{i=1}^n \int_\Omega f_i(\mathbf{x}', t) p(\mathbf{x}', t) \frac{\partial\varphi}{\partial x_i'} \mathrm{d}\mathbf{x}' + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \int_\Omega D_{ij}(\mathbf{x}', t) p(\mathbf{x}', t) \frac{\partial^2\varphi}{\partial x_i' \partial x_j'} \mathrm{d}\mathbf{x}'
\end{aligned}$$

where

$$f_i(\mathbf{x}', t) = \int_\Omega (x_i - x_i') w_t(\mathbf{x}|\mathbf{x}') \mathrm{d}\mathbf{x} \qquad \text{and} \qquad D_{ij}(\mathbf{x}', t) = \int_\Omega (x_i - x_i')(x_j - x_j') w_t(\mathbf{x}|\mathbf{x}') \mathrm{d}\mathbf{x}$$

Applying the integration by parts rule (Eq. (7)) again, and renaming $\mathbf{x}'$ to $\mathbf{x}$, we obtain

$$\text{RHS} = -\sum_{i=1}^{n}\int_{\Omega}\varphi(\mathbf{x})\frac{\partial\left[f_i(\mathbf{x},t)p(\mathbf{x},t)\right]}{\partial x_i}\mathrm{d}\mathbf{x} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\int_{\Omega}\varphi(\mathbf{x})\frac{\partial^2\left[D_{ij}(\mathbf{x},t)p(\mathbf{x},t)\right]}{\partial x_i\partial x_j}\mathrm{d}\mathbf{x}$$

Matching LHS with RHS, we are left with

$$\int_{\Omega}\varphi(\mathbf{x})\frac{\partial}{\partial t}p(\mathbf{x},t)\mathrm{d}\mathbf{x} = -\sum_{i=1}^{n}\int_{\Omega}\varphi(\mathbf{x})\frac{\partial\left[f_i(\mathbf{x},t)p(\mathbf{x},t)\right]}{\partial x_i}\mathrm{d}\mathbf{x} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\int_{\Omega}\varphi(\mathbf{x})\frac{\partial^2\left[D_{ij}(\mathbf{x},t)p(\mathbf{x},t)\right]}{\partial x_i\partial x_j}\mathrm{d}\mathbf{x} \tag{17}$$

Again, the only way for Eq. (17) to hold for any $\varphi(\mathbf{x})$ is the following:

$$\frac{\partial}{\partial t}p(\mathbf{x},t) = -\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\left[f_i(\mathbf{x},t)p(\mathbf{x},t)\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2}{\partial x_i\partial x_j}\left[D_{ij}(\mathbf{x},t)p(\mathbf{x},t)\right] \tag{18}$$

which is the same as Eq. (11).

Eq. (18) shows that the drift and diffusion coefficients $\mathbf{f}(\cdot)$ and $\mathbf{G}(\cdot,\cdot)$ in the SDE (Eq. (1)) are in fact the first and second moment of the transition density $w_t(\mathbf{x}|\mathbf{x}')$, and the FPE can be expressed solely in terms of the marginal density $p(\mathbf{x},t)$ and transition density $w_t(\mathbf{x}|\mathbf{x}')$.

5 Reverse-time SDE

Previously we have discussed the (most common) type of SDE where time flows in the forward direction. There are times when we do want to model just the opposite — t flows backwards from T to 0.

Let's consider the following *reverse-time* SDE

$$\mathrm{d}\mathbf{x} = \bar{\mathbf{f}}(\mathbf{x},t)\mathrm{d}t + \mathbf{G}(\mathbf{x},t)\mathrm{d}\bar{\mathbf{w}} \tag{19}$$

where

$$\bar{\mathbf{f}}(\mathbf{x},t) = \mathbf{f}(\mathbf{x},t) - \mathbf{D}(\mathbf{x},t)\nabla_{\mathbf{x}}\log p(\mathbf{x},t) - \nabla_{\mathbf{x}}\cdot\ \mathbf{D}(\mathbf{x},t)$$

and $\mathbf{D}(\mathbf{x},t)$ denotes $\mathbf{G}(\mathbf{x},t)\mathbf{G}^{\top}(\mathbf{x},t)$. In Eq. (19),

- $\mathrm{d}t$ represents a negative infinitesimal time step.
- $\mathrm{d}\bar{\mathbf{w}}$ is a standard Wiener process when time flows backwards from T to 0.
- $p(\mathbf{x},t)$ is the marginal probability density of the stochastic process that solves Eq. (1).
- $\mathbf{f}(\cdot)$ and $\mathbf{G}(\cdot,\cdot)$ are the same functions in Eq. (1)

Simulating the reverse-time SDE is pretty much the same as the forward-time counterpart, albeit with a twist — we start with $\mathbf{x}$ drawn from the "final" distribution at time $t = T$, and we compute the negative increment as the negative step of $-\bar{\mathbf{f}}(\mathbf{x},t)\Delta t$ plus the random noise $\mathbf{G}(\mathbf{x},t)\Delta\mathbf{w}$, that is $-\Delta\mathbf{x} = -\bar{\mathbf{f}}(\mathbf{x},t)\Delta t + \mathbf{G}(\mathbf{x},t)\Delta\mathbf{w}$. So the updated state and time variable will be $\mathbf{x} - \Delta\mathbf{x}$ and $t - \Delta t$. By repeating this step we will be left with a sequence of $\mathbf{x}$ values running backwards. Under similar

conditions, Eq. (19) also induces a stochastic process, and we can derive the "reverse-time version" of the FPE that characterizes its marginal probability density.

Suppose $\varphi(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ is any twice-differentiable function, then $\varphi(\mathbf{x} - \Delta\mathbf{x})$ is the displacement of $\varphi(\mathbf{x})$ in the negative direction. We compute the negative increment $-\Delta\varphi(\mathbf{x})$ by computing the 2nd-order Taylor expansion of $\varphi(\mathbf{x} - \Delta\mathbf{x})$ around $\mathbf{x}$:

$$\begin{aligned} -\Delta\varphi(\mathbf{x}) =& \varphi(\mathbf{x} - \Delta\mathbf{x}) - \varphi(\mathbf{x}) \\ =& \sum_{i=1}^{n} \frac{\partial\varphi}{\partial x_i}(-\Delta x_i) + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\varphi}{\partial x_i \partial x_j}(-\Delta x_i)(-\Delta x_j) \end{aligned} \tag{20}$$

We apply to Eq. (20) the same line of derivations starting from Eq. (2) through (10) (i.e. substituting Δx_i, discarding $o(\Delta t)$, and integrating by parts) that we did for the forward SDE, which gives

$$-\int_\Omega \varphi(\mathbf{x}) \frac{\partial}{\partial t}\bar{p}(\mathbf{x}, t)\mathrm{d}\mathbf{x} = \sum_{i=1}^{n} \int_\Omega \varphi(\mathbf{x}) \frac{\partial}{\partial x_i}\left[\bar{f}_i(\mathbf{x}, t)\bar{p}(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\int_\Omega \varphi(\mathbf{x}) \frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)\bar{p}(\mathbf{x}, t)\right]\mathrm{d}\mathbf{x} \tag{21}$$

where $\bar{p}(\mathbf{x}, t)$ is the probability density of the stochastic process induced by Eq. (19), and $D_{ij}(\mathbf{x}, t) = \sum_{k=1}^{n} G_{ik}(\mathbf{x}, t)G_{jk}(\mathbf{x}, t)$. Again, the only way for Eq. (21) to hold for any function $\varphi(\mathbf{x})$ is

$$-\frac{\partial}{\partial t}\bar{p}(\mathbf{x}, t) = \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[\bar{f}_i(\mathbf{x}, t)\bar{p}(\mathbf{x}, t)\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)\bar{p}(\mathbf{x}, t)\right] \tag{22}$$

which is the PDE describing the backwards evolution of $\bar{p}(\mathbf{x}, t)$. It turns out that the forward time FPE (Eq. (11)) can be transformed into exactly the same form as Eq. (22).

We negate both sides of Eq. (11), and split the coefficient for second derivatives $-\frac{1}{2}$ as $-1 + \frac{1}{2}$, and absorb the first part into the gradient term:

$$\begin{aligned} & -\frac{\partial}{\partial t}p(\mathbf{x}, t) \\ =& \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right] - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2}{\partial x_i \partial x_j}\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right] \\ =& \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right] - \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} \\ =& \sum_{i=1}^{n} \left[\frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right] - \sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j}\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} \\ =& \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t) - \sum_{j=1}^{n}\left[D_{ij}(\mathbf{x}, t)\frac{\partial p(\mathbf{x}, t)}{\partial x_j} + p(\mathbf{x}, t)\frac{\partial D_{ij}(\mathbf{x}, t)}{\partial x_j}\right]\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} \\ =& \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[f_i(\mathbf{x}, t)p(\mathbf{x}, t) - \sum_{j=1}^{n}\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\frac{\partial \log p(\mathbf{x}, t)}{\partial x_j} + p(\mathbf{x}, t)\frac{\partial D_{ij}(\mathbf{x}, t)}{\partial x_j}\right]\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} \\ =& \sum_{i=1}^{n} \frac{\partial}{\partial x_i}\left[\left[f_i(\mathbf{x}, t) - \sum_{j=1}^{n}\left[D_{ij}(\mathbf{x}, t)\frac{\partial \log p(\mathbf{x}, t)}{\partial x_j} + \frac{\partial D_{ij}(\mathbf{x}, t)}{\partial x_j}\right]\right]p(\mathbf{x}, t)\right] + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2\left[D_{ij}(\mathbf{x}, t)p(\mathbf{x}, t)\right]}{\partial x_i \partial x_j} \end{aligned}$$

$$= \sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[\bar{f}_i(\mathbf{x}, t) p(\mathbf{x}, t) \right] + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 \left[D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \right]}{\partial x_i \partial x_j} \tag{23}$$

where

$$\bar{f}_i(\mathbf{x}, t) = f_i(\mathbf{x}, t) - \sum_{j=1}^{n} D_{ij}(\mathbf{x}, t) \frac{\partial}{\partial x_j} \log p(\mathbf{x}, t) - \sum_{j=1}^{n} \frac{\partial}{\partial x_j} D_{ij}(\mathbf{x}, t)$$

for $i = 1, 2, \cdots, n$.

Eq. (23) has the same form as Eq. (22). Assuming again the uniqueness of the solution, it must hold that $p(\mathbf{x}, t) = \bar{p}(\mathbf{x}, t)$ for $t > 0$. So we have shown for the forward-time SDE (Eq. (1)), there exists a reverse-time SDE (Eq. (19)) that induces the same path of marginal probability density (also true for PF ODE), which we refer to as the *marginal preserving* property.

6 Means and Covariances of SDEs

The FPE (Eq. (11) and (18)) contains the complete specifications of the marginal probability density $p(\mathbf{x}, t)$ of a stochastic process induced by SDE. Sometimes we are only interested in the moments (e.g. mean and covariance) of the distribution and would like to model how they evolve without solving the FPE as an intermediate step.

It turns out that the dynamics of the mean and covariance can be also described by ODEs. Consider the time-dependent Ito's formula in Eq. (4). We take expectation w.r.t. $\mathbf{x}$ and $\mathrm{d}\mathbf{w}$ to average out the stochasticity (as we did for the time-independent case in Eq. (5)):

$$\frac{\mathrm{d}}{\mathrm{d}t} \int_{\Omega} \varphi(\mathbf{x}, t) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} = \int_{\Omega} \frac{\partial \varphi}{\partial t} p(\mathbf{x}, t) \mathrm{d}\mathbf{x} + \sum_{i=1}^{n} \int_{\Omega} \frac{\partial \varphi}{\partial x_i} f_i(\mathbf{x}, t) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \int_{\Omega} \frac{\partial^2 \varphi}{\partial x_i \partial x_j} D_{ij}(\mathbf{x}, t) p(\mathbf{x}, t) \mathrm{d}\mathbf{x} \tag{24}$$

where $\varphi(\mathbf{x}, t)$ is any function of our choice.

We set φ to different functions so that the ODE for mean and covariance can be derived from Eq. (24):

1. If $\varphi(\mathbf{x}, t) = x_u$ where x_u is the u-th component of $\mathbf{x}$, then

   - for $i = 1, \cdots, n$

     $$\frac{\partial}{\partial x_i} \varphi(\mathbf{x}, t) = \begin{cases} 0 & i \neq u \\ 1 & i = u \end{cases}$$

     therefore

     $$\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \varphi(\mathbf{x}, t) f_i(\mathbf{x}, t) = f_u(\mathbf{x}, t)$$

   - for $i = 1, \cdots, n$ and $j = 1, \cdots, n$

     $$\frac{\partial^2}{\partial x_i \partial x_j} \varphi(\mathbf{x}, t) = 0$$

- and

$$\frac{\partial}{\partial t}\varphi(\mathbf{x},t)=0$$

So Eq. (24) reduces to

$$\frac{\mathrm{d}}{\mathrm{d}t}\mu_u(t)=\int f_u(\mathbf{x},t)p(\mathbf{x},t)\mathrm{d}\mathbf{x} \tag{25}$$

where $\mu_i(t)$ is the mean of x_i at time t:

$$\mu_i(t)=\int x_i p(\mathbf{x},t)\mathrm{d}\mathbf{x}$$

2. If $\varphi(\mathbf{x},t)=x_u x_v-\mu_u(t)\mu_v(t)$, then

   - for $i=1,\cdots,n$

$$\frac{\partial}{\partial x_i}\varphi(\mathbf{x},t)=\begin{cases}x_u & \text{if } i=v\\ x_v & \text{if } i=u\\ 0 & \text{everywhere else}\end{cases}$$

     therefore

$$\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\varphi(\mathbf{x},t)f_i(\mathbf{x},t)=x_u f_v(\mathbf{x},t)+x_v f_u(\mathbf{x},t)$$

   - for $i=1,\cdots,n$ and $j=1,\cdots,n$

$$\frac{\partial^2}{\partial x_i\partial x_j}\varphi(\mathbf{x},t)=\begin{cases}1 & \text{if } i=u \text{ and } j=v \text{ and } u\neq v\\ 2 & \text{if } i=j=u=v\\ 0 & \text{everywhere else}\end{cases}$$

     therefore (recall that $\mathbf{D}(\mathbf{x},t)$ is symmetric)

$$\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\frac{\partial^2}{\partial x_i\partial x_j}\varphi(\mathbf{x},t)D_{ij}(\mathbf{x},t)=D_{uv}(\mathbf{x},t)$$

   - and

$$\begin{aligned}\frac{\partial\varphi(\mathbf{x},t)}{\partial t}&=-\mu_v(t)\frac{\mathrm{d}\mu_u(t)}{\mathrm{d}t}-\mu_u(t)\frac{\mathrm{d}\mu_v(t)}{\mathrm{d}t}\\&=-\mu_v(t)\int f_u(\mathbf{x},t)p(\mathbf{x},t)\mathrm{d}\mathbf{x}-\mu_u(t)\int f_v(\mathbf{x},t)p(\mathbf{x},t)\mathrm{d}\mathbf{x}\\&=-\int\mu_v(t)f_u(\mathbf{x},t)p(\mathbf{x},t)\mathrm{d}\mathbf{x}-\int\mu_u(t)f_v(\mathbf{x},t)p(\mathbf{x},t)\mathrm{d}\mathbf{x}\end{aligned}$$

     which is a function independent of $\mathbf{x}$.

So Eq. (24) reduces to

$$\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}t}\int \left[x_u x_v - \mu_u(t)\mu_v(t)\right] p(\mathbf{x},t)\mathrm{d}\mathbf{x} \\
=&-\int \mu_v(t) f_u(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} - \int \mu_u(t) f_v(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} + \int \left[x_u f_v(\mathbf{x},t) + x_v f_u(\mathbf{x},t)\right] p(\mathbf{x},t)\mathrm{d}\mathbf{x} \\
&+\int D_{uv}(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} \\
=&\int \left[x_u - \mu_u(t)\right] f_v(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} + \int \left[x_v - \mu_v(t)\right] f_u(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} + \int D_{uv}(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x}
\end{aligned} \tag{26}$$

Let $\boldsymbol{\Sigma}(t)$ be the covariance matrix of $\mathbf{x}$ at time t:

$$\boldsymbol{\Sigma}(t) = \int \left[(\mathbf{x}-\boldsymbol{\mu}(t))(\mathbf{x}-\boldsymbol{\mu}(t))^\top\right] p(\mathbf{x},t)\mathrm{d}\mathbf{x} = \int \mathbf{x}\mathbf{x}^\top p(\mathbf{x},t)\mathrm{d}\mathbf{x} - \boldsymbol{\mu}(t)\boldsymbol{\mu}^\top(t) = \mathbb{E}\left[\mathbf{x}\mathbf{x}^\top\right] - \boldsymbol{\mu}(t)\boldsymbol{\mu}^\top(t)$$

We can "tile" the elementwise Eq. (25) and (26) to obtain the matrix-form ODE for the mean:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\mu}(t) = \int \mathbf{f}(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} = \mathbb{E}\left[\mathbf{f}(\mathbf{x},t)\right] \tag{27}$$

and covariances

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\Sigma}(t) &= \int (\mathbf{x}-\boldsymbol{\mu}(t))\,\mathbf{f}^\top(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} + \int \mathbf{f}(\mathbf{x},t)(\mathbf{x}-\boldsymbol{\mu}(t))^\top p(\mathbf{x},t)\mathrm{d}\mathbf{x} + \int \mathbf{D}(\mathbf{x},t) p(\mathbf{x},t)\mathrm{d}\mathbf{x} \\
&= \mathbb{E}\left[(\mathbf{x}-\boldsymbol{\mu}(t))\,\mathbf{f}^\top(\mathbf{x},t)\right] + \mathbb{E}\left[\mathbf{f}(\mathbf{x},t)(\mathbf{x}-\boldsymbol{\mu}(t))^\top\right] + \mathbb{E}\left[\mathbf{G}(\mathbf{x},t)\mathbf{G}^\top(\mathbf{x},t)\right]
\end{aligned} \tag{28}$$

7 Linear SDE

Note that the expectations in Eq. (27) and (28) are taken w.r.t. $p(\mathbf{x},t)$ which still requires us to solve the FPE first. However, for a special form of SDE

$$\mathrm{d}\mathbf{x} = f(t)\mathbf{x}\mathrm{d}t + g(t)\mathrm{d}\mathbf{w} \tag{29}$$

where $\mathbf{f}$ and $\mathbf{G}$ are affine transformations (i.e. input is scaled and optionally shifted), it is possible to express $\boldsymbol{\mu}(t)$ and $\boldsymbol{\Sigma}(t)$ analytically:

Substituting the drift and diffusion coefficient $\mathbf{f}(\mathbf{x},t) = f(t)\mathbf{x}$ and $\mathbf{G}(\mathbf{x},t) = g(t)\mathbf{I}$ into Eq. (27) and (28), we obtain differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\mu}(t) = f(t)\boldsymbol{\mu}(t) \tag{30}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\Sigma}(t) = 2f(t)\boldsymbol{\Sigma}(t) + g^2(t)\mathbf{I} \tag{31}$$

Solving[4] Eq. (30) and (31), we obtain

$$\boldsymbol{\mu}(t) = s(t)\boldsymbol{\mu}(0) \tag{32}$$

[4]Math refresher/Hint: 1. Homogeneous differential equation (Eq. (30)) can be solved by separation of variables, while inhomogeneous equation (Eq. (31)) can be solved by first finding the general solution to the associated homogeneous equation, and combining it with the particular solution to the inhomogeneous equation; 2. Leibniz integral rule: $\mathrm{d}\int_0^t f(z)\mathrm{d}z = f(t)\mathrm{d}t$; 3. try to find the elementwise solutions first, then combine them in matrix form.

and

$$\boldsymbol{\Sigma}(t) = s^2(t)\boldsymbol{\Sigma}(0) + s^2(t)\sigma^2(t)\mathbf{I} \tag{33}$$

where we denote

$$s(t) = \exp\left[\int_0^t f(z)\mathrm{d}z\right] \tag{34}$$

and

$$\sigma(t) = \sqrt{\int_0^t \frac{g^2(z)}{s^2(z)}\mathrm{d}z} \tag{35}$$

$s(t)$ can be regarded as a time-dependent scaling factor on the mean, and $\sigma(t)$ is meant to adjust the standard deviation.

Interestingly, restricting $\mathbf{f}$ and $\mathbf{G}$ to affine transforms not only renders the mean and covariance computable analytically, but also guarantees the Normality of $\mathbf{x}$:

Dividing Eq. (29) by $s(t)$ on both sides

$$\frac{\mathrm{d}\mathbf{x}}{s(t)} = f(t)\frac{\mathbf{x}}{s(t)}\mathrm{d}t + \frac{g(t)}{s(t)}\mathrm{d}\mathbf{w}$$

Rearranging and applying differential quotient rule,

$$\frac{\mathrm{d}\mathbf{x}}{s(t)} - f(t)\frac{\mathbf{x}}{s(t)}\mathrm{d}t = \mathrm{d}\frac{\mathbf{x}}{s(t)} = \frac{g(t)}{s(t)}\mathrm{d}\mathbf{w}$$

Integrate both sides from 0 to t and rearranging,

$$\mathbf{x} = \mathbf{x}_t = s(t)\left[\mathbf{x}_0 + \int_0^t \frac{g(z)}{s(z)}\mathrm{d}\mathbf{w}\right] \tag{36}$$

The integral can be approximated as the following sum

$$\int_0^t \frac{g(z)}{s(z)}\mathrm{d}\mathbf{w} \approx \sum_{i=0}^{N-1} \frac{g(t_i)}{s(t_i)}(\mathbf{w}_{t_{i+1}} - \mathbf{w}_{t_i}) \tag{37}$$

where $\{t_i\}$ partitions the interval $[0, t]$, and the Wiener increments $\mathbf{w}_{t_{i+1}} - \mathbf{w}_{t_i} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}(t_{i+1} - t_i))$.

Since each term in the sum is an independent Gaussian random variable (which are closed under linear combination), according to Central Limit Theorem the infinite sum remains Gaussian as $N \to \infty$. So $\mathbf{x}$ is Gaussian too (conditioned on the initial value $\mathbf{x}_0$ being constant).

Putting everything together, $\mathbf{x}$ follows diagonal Gaussian distribution given that $\mathbf{x}_0$ is known:

$$\mathbf{x}|\mathbf{x}_0 \sim \mathcal{N}\left(s(t)\mathbf{x}_0, s^2(t)\sigma^2(t)\mathbf{I}\right) \tag{38}$$

and the corresponding PF ODE and reverse-time SDE are (*cf.* Eq. (14 and 19))

$$\mathrm{d}\mathbf{x} = \left[f(t)\mathbf{x} - \frac{1}{2}g^2(t)\nabla \log p(\mathbf{x}, t)\right]\mathrm{d}t \tag{39}$$

and

$$\mathrm{d}\mathbf{x} = \left[f(t)\mathbf{x} - g^2(t)\nabla \log p(\mathbf{x}, t)\right]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}} \tag{40}$$

which have the marginal preserving property as previously illustrated.

8 Case study: VE, VP, and sub-VP SDE

So far we have introduced the general properties of SDEs and their induced stochastic processes. Now let's turn our attention to three concrete instances of SDEs: they all fit into the form of Eq. (29), but differ in the way that $f(t)$ and $g(t)$ are actually implemented. We are going to show how different choices of $f(t)$ and $g(t)$ will impact the dynamics of the mean and covariance.

1. **VE SDE**

$$\mathrm{d}\mathbf{x} = \sqrt{\frac{\mathrm{d}\left[\sigma^2(t)\right]}{\mathrm{d}t}}\mathrm{d}\mathbf{w} \tag{41}$$

where[5] $f(t) = 0$, $g(t) = \sqrt{\frac{\mathrm{d}[\sigma^2(t)]}{\mathrm{d}t}}$.

Since $s(t) = e^{\int_0^t 0\mathrm{d}z} = 1$, we have the covariance

$$\boldsymbol{\Sigma}(t) = s^2(t)\boldsymbol{\Sigma}(0) + s^2(t)\sigma^2(t)\mathbf{I} = \boldsymbol{\Sigma}(0) + \sigma^2(t)\mathbf{I} \tag{42}$$

and mean

$$\boldsymbol{\mu}(t) = \boldsymbol{\mu}(0) \tag{43}$$

2. **VP SDE**

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}\mathbf{w} \tag{44}$$

where $f(t) = -\frac{1}{2}\beta(t)$ and $g(t) = \sqrt{\beta(t)}$.

Since

$$\sigma^2(t) = \int_0^t \frac{g^2(z)}{s^2(z)}\mathrm{d}z = \int_0^t -2f(z)\, e^{-2\int_0^z f(z')\mathrm{d}z'}\,\mathrm{d}z = \int_0^t \mathrm{d}e^{-2\int_0^z f(z')\mathrm{d}z'} = s^{-2}(t) - 1 \tag{45}$$

We have the covariance

$$\boldsymbol{\Sigma}(t) = \mathbf{I} + s^2(t)\left(\boldsymbol{\Sigma}(0) - \mathbf{I}\right) = \mathbf{I} + e^{-\int_0^t \beta(z)\mathrm{d}z}\left(\boldsymbol{\Sigma}(0) - \mathbf{I}\right) \tag{46}$$

and mean

$$\boldsymbol{\mu}(t) = s(t)\boldsymbol{\mu}(0) = \boldsymbol{\mu}(0)e^{\int_0^t f(z)\mathrm{d}z} = \boldsymbol{\mu}(0)e^{-\frac{1}{2}\int_0^t \beta(z)\mathrm{d}z} \tag{47}$$

3. **sub-VP SDE**

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\mathrm{d}t + \sqrt{\beta(t)(1 - e^{-2\int_0^t \beta(z)\mathrm{d}z})}\mathrm{d}\mathbf{w} \tag{48}$$

where $f(t) = -\frac{1}{2}\beta(t)$ and $g(t) = \sqrt{\beta(t)(1 - e^{-2\int_0^t \beta(z)\mathrm{d}z})}$.

Since

$$\sigma^2(t) = \int_0^t \frac{g^2(z)}{s^2(z)}\mathrm{d}z = \int_0^t -2f(z)\left[1 - e^{-2\int_0^z -2f(z')\mathrm{d}z'}\right] e^{-2\int_0^z f(z')\mathrm{d}z'}\,\mathrm{d}z$$

[5] In the definition of VE SDE we are free to set $\sigma(t)$ to any function, e.g. t, $\sqrt{t}$, which is the standard deviation of the marginal density as $t \to \infty$

$$= \int_0^t -2f(z)e^{-2\int_0^z f(z')\mathrm{d}z'}\mathrm{d}z + \int_0^t 2f(z)e^{2\int_0^z f(z')\mathrm{d}z'}\mathrm{d}z = s^{-2}(t) - 2 + s^2(t) \tag{49}$$

We have the covariance

$$\boldsymbol{\Sigma}(t) = \mathbf{I} + s^2(t)\left(\boldsymbol{\Sigma}(0) - 2\mathbf{I}\right) + \mathbf{I}s^4(t) = \mathbf{I} + e^{-\int_0^t \beta(z)\mathrm{d}z}\left(\boldsymbol{\Sigma}(0) - 2\mathbf{I}\right) + \mathbf{I}e^{-2\int_0^t \beta(z)\mathrm{d}z} \tag{50}$$

and mean

$$\boldsymbol{\mu}(t) = s(t)\boldsymbol{\mu}(0) = \boldsymbol{\mu}(0)e^{-\frac{1}{2}\int_0^t \beta(z)\mathrm{d}z} \tag{51}$$

which is the same as the VP SDE since they share the same $f(t)$.

Let's put the means and covariances in the following table

	Mean	**Covariance**
VE	$\boldsymbol{\mu}(0)$	$\boldsymbol{\Sigma}(0) + \sigma^2(t)\mathbf{I}$
VP	$s(t)\boldsymbol{\mu}(0)$	$\mathbf{I} + s^2(t)\left(\boldsymbol{\Sigma}(0) - \mathbf{I}\right)$
sub-VP	$s(t)\boldsymbol{\mu}(0)$	$\mathbf{I} + s^2(t)\left(\boldsymbol{\Sigma}(0) - 2\mathbf{I}\right) + \mathbf{I}s^4(t)$

Suppose VP and sub-VP SDE share the same value of $\beta(t)$ and $\boldsymbol{\Sigma}(0)$, then

- $0 < s(t) \leq 1$ and is monotonically decreasing because: 1. $s(0) = 1$; 2. $s(t) > 0$; 3. $s'(t) = s(t)f(t) = -\frac{1}{2}s(t)\beta(t) < 0$ (Note the nonnegativity of $\beta(t)$).
- $\lim_{t\to\infty} s^2(t) = \lim_{t\to\infty} e^{-\int_0^t \beta(z)\mathrm{d}z} = 0$, because $\lim_{t\to\infty}\int_0^t \beta(z)\mathrm{d}z = \infty$

We can quickly make the following observations:

- VE SDE's variance $\sigma^2(t)$ is unbounded as $t \to \infty$.
- Both VP and sub-VP SDE's covariance are bounded between $\boldsymbol{\Sigma}(0)$ and $\mathbf{I}$.
- VP SDE induces *higher* variance than sub-VP SDE: the covariance matrix of VP subtracted by that of sub-VP is positive definite.
- As t increases, the mean of VP and sub-VP will be nudged towards the origin, and the covariance moves towards $\mathbf{I}$.
- In the limit of $t \to \infty$, $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for VP and sub-VP SDE, and $\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \sigma^2(t)\mathbf{I})$ for VE SDE.

which is why they are named *Variance Exploding* (VE), *Variance Preserving* (VP), and sub-VP SDE. We can likewise define PF ODEs corresponding to VE, VP, and sub-VP SDEs.

This implies that when simulating the forward SDE/ODE for sufficiently long period of time, the final distribution will be close to an isotropic Gaussian which is efficient to sample from. By leveraging the "marginal preserving" property, we can simulate them backwards in time so that the initial distribution is recovered.

9 Learn the Score-function from Data

In order to simulate PF ODE or reverse-time SDE to sample from the initial data distribution, the only thing yet to be determined is the value of the score function $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t)$ for all $\mathbf{x}_t$ and t. It turns out we can use a data-driven approach to learn a parametric model $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ that approximates the score function by minimizing the following objective

$$L_t = \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t}\left[\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\|^2\right] = \iint p(\mathbf{x}_t|\mathbf{x}_0)p(\mathbf{x}_0)\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\|^2 \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t \tag{52}$$

which is a squared difference between $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ and the *denoising score* $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)$. As we will see shortly, the minimizer of L_t happens to be equal to the true score function $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t)$.

To compute the expectation in Eq. (52), we first sample $\mathbf{x}_0$ from the data distribution $p_{\text{data}}(\mathbf{x}_0)$ used to initialize the SDE, and then obtain a corrupted version of $\mathbf{x}_0$ by sampling from $p(\mathbf{x}_t|\mathbf{x}_0)$ (Eq. (38)), given that a linear SDE is considered (Eq. (29)). Importantly, this allows us to bypass the simulation process which would otherwise require a cost proportional to time t, thus enabling efficient *simulation-free* training.

Now given a specific value of $\mathbf{x}_t$, our goal is to minimize

$$L(\mathbf{x}_t, t) = \int p(\mathbf{x}_0, \mathbf{x}_t)\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\|^2 \mathrm{d}\mathbf{x}_0$$

Let $\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t)$ denote the optimal score model minimizing $L(\mathbf{x}_t, t)$. It is necessary that the gradient of $L(\mathbf{x}_t, t)$ w.r.t. $\mathbf{s}_{\boldsymbol{\theta}}$ is zero:

$$\nabla_{\mathbf{s}_{\boldsymbol{\theta}}} L(\mathbf{x}_t, t) = \int p(\mathbf{x}_0, \mathbf{x}_t) \cdot 2\left[\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\right] \mathrm{d}\mathbf{x}_0 = \mathbf{0}$$

Rearranging, and simplifying by factorizing $p(\mathbf{x}_0, \mathbf{x}_t)$ as $p(\mathbf{x}_t) \cdot p(\mathbf{x}_0|\mathbf{x}_t)$,

$$\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) = \int p(\mathbf{x}_0|\mathbf{x}_t)\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\mathrm{d}\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0}\left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)|\mathbf{x}_t\right]$$

which can be interpreted as *conditional expectation* of the score of the perturbation kernel $p(\mathbf{x}_t|\mathbf{x}_0)$ given $\mathbf{x}_t$. It provides an unbiased estimate for the score function we are seeking in the first place:

$$\begin{aligned}
\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) =& \mathbb{E}_{\mathbf{x}_0}\left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)|\mathbf{x}_t\right] \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)\mathrm{d}\mathbf{x}_0 \\
=& \int \frac{p(\mathbf{x}_t|\mathbf{x}_0)p(\mathbf{x}_0)}{p(\mathbf{x}_t)} \frac{\nabla_{\mathbf{x}_t} p(\mathbf{x}_t|\mathbf{x}_0)}{p(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{d}\mathbf{x}_0 \\
=& \frac{1}{p(\mathbf{x}_t)} \int p(\mathbf{x}_0)\nabla_{\mathbf{x}_t} p(\mathbf{x}_t|\mathbf{x}_0)\mathrm{d}\mathbf{x}_0 \\
=& \frac{1}{p(\mathbf{x}_t)} \nabla_{\mathbf{x}_t} \int p(\mathbf{x}_0)p(\mathbf{x}_t|\mathbf{x}_0)\mathrm{d}\mathbf{x}_0 \\
=& \frac{\nabla_{\mathbf{x}_t} p(\mathbf{x}_t)}{p(\mathbf{x}_t)} \\
=& \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t)
\end{aligned} \tag{53}$$

We can further express $\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t)$ as function of $\mathbf{x}_0$ and $\mathbf{x}_t$. According to Eq. (38),

$$p(\mathbf{x}_t|\mathbf{x}_0) = \left[2\pi s^2(t)\sigma^2(t)\right]^{-\frac{n}{2}} \exp\left[-\frac{\|\mathbf{x}_t - s(t)\mathbf{x}_0\|^2}{2s^2(t)\sigma^2(t)}\right]$$

so the denoising score

$$\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) = \frac{\nabla_{\mathbf{x}_t} p(\mathbf{x}_t|\mathbf{x}_0)}{p(\mathbf{x}_t|\mathbf{x}_0)} = \frac{1}{p(\mathbf{x}_t|\mathbf{x}_0)} \cdot p(\mathbf{x}_t|\mathbf{x}_0) \frac{s(t)\mathbf{x}_0 - \mathbf{x}_t}{s^2(t)\sigma^2(t)} = \frac{s(t)\mathbf{x}_0 - \mathbf{x}_t}{s^2(t)\sigma^2(t)} \tag{54}$$

and the optimal score model

$$\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t) = \mathbb{E}_{\mathbf{x}_0}\left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)|\mathbf{x}_t\right] = \mathbb{E}_{\mathbf{x}_0}\left[\frac{s(t)\mathbf{x}_0 - \mathbf{x}_t}{s^2(t)\sigma^2(t)}\middle|\mathbf{x}_t\right] = \frac{s(t)\mathbb{E}_{\mathbf{x}_0}\left[\mathbf{x}_0|\mathbf{x}_t\right] - \mathbf{x}_t}{s^2(t)\sigma^2(t)} \tag{55}$$

Note that there is some flexibility in how to parameterize the score model — we can make $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ to learn the score function itself as originally formulated, or only part of it (i.e. $\mathbb{E}_{\mathbf{x}_0}\left[\mathbf{x}_0|\mathbf{x}_t\right]$) from which the whole score function can be easily restored. Either way, the approximation of the score function remains the same.

10 SDE and PF ODE under Common Framework

We have proved that both PF ODE and reverse SDE have the marginal preserving property to make sure a sample point doesn't stray from the correct path as it evovles backwards in time. Let's rewrite here the PF ODE (Eq. (14)),

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - \frac{1}{2}\mathbf{D}(\mathbf{x}, t)\nabla \log p(\mathbf{x}, t) - \frac{1}{2}\nabla \cdot \mathbf{D}(\mathbf{x}, t)\right] \mathrm{d}t$$

reverse-time SDE (Eq. (19))

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - \mathbf{D}(\mathbf{x}, t)\nabla \log p(\mathbf{x}, t) - \nabla \cdot \ \mathbf{D}(\mathbf{x}, t)\right] \mathrm{d}t + \mathbf{G}(\mathbf{x}, t)\mathrm{d}\bar{\mathbf{w}}$$

along with the forward SDE (Eq. (1)),

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t + \mathbf{G}(\mathbf{x}, t)\mathrm{d}\mathbf{w}$$

It turns out that they all belong to the following families of generalized SDEs,

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - (1 - \alpha^2)\frac{1}{2}\left[\mathbf{D}(\mathbf{x}, t)\nabla \log p(\mathbf{x}, t) + \nabla \cdot \mathbf{D}(\mathbf{x}, t)\right]\right] \mathrm{d}t + \alpha\mathbf{G}(\mathbf{x}, t)\mathrm{d}\mathbf{w} \tag{56}$$

and

$$\mathrm{d}\mathbf{x} = \left[\mathbf{f}(\mathbf{x}, t) - (1 + \alpha^2)\frac{1}{2}\left[\mathbf{D}(\mathbf{x}, t)\nabla \log p(\mathbf{x}, t) + \nabla \cdot \mathbf{D}(\mathbf{x}, t)\right]\right] \mathrm{d}t + \alpha\mathbf{G}(\mathbf{x}, t)\mathrm{d}\bar{\mathbf{w}} \tag{57}$$

such that they can be recovered by setting the hyperparameter α to 0 or 1. If $\alpha = 0$, both Eq. (56) and (57) reduce to the same deterministic PF ODE. Conversely, if $\alpha = 1$, Eq. (56) reduces to the forward SDE, while Eq. (57) reduces to reverse-time SDE. Inuitively, α moderates the amount of stochasticity.

In addition, for any $\alpha \in [0,1]$ both SDEs always induce stochastic process with the same marginal probability density $p(\mathbf{x}, t)$. To see why this is the case, we can rewrite them as

$$\mathrm{d}\mathbf{x} = \underbrace{\left[\mathbf{f}(\mathbf{x},t) - \frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla \log p(\mathbf{x},t) + \nabla\cdot\mathbf{D}(\mathbf{x},t)\right]\right]\mathrm{d}t}_{\text{Probability Flow ODE}} + \underbrace{\alpha^2\frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla \log p(\mathbf{x},t) + \nabla\cdot\mathbf{D}(\mathbf{x},t)\right]\mathrm{d}t + \alpha\mathbf{G}(\mathbf{x},t)\mathrm{d}\mathbf{w}}_{\text{forward Langevin diffusion SDE}} \tag{58}$$

and

$$\mathrm{d}\mathbf{x} = \underbrace{\left[\mathbf{f}(\mathbf{x},t) - \frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla \log p(\mathbf{x},t) + \nabla\cdot\mathbf{D}(\mathbf{x},t)\right]\right]\mathrm{d}t}_{\text{Probability Flow ODE}} + \underbrace{\left(-\alpha^2\right)\frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla \log p(\mathbf{x},t) + \nabla\cdot\mathbf{D}(\mathbf{x},t)\right]\mathrm{d}t + \alpha\mathbf{G}(\mathbf{x},t)\mathrm{d}\bar{\mathbf{w}}}_{\text{reverse Langevin diffusion SDE}} \tag{59}$$

where they are broken down as the sum of a deterministic component (i.e.PF ODE) and a stochastic component, the latter of which is referred to as "Langevin diffusion SDE".

Interestingly, the Langevin diffusion SDE corresponds to a stochastic process that remove noise and inject the *same* amount of noise simultaneously, resulting net change of zero in the noise level.

We can write the PDE (i.e. the "FPE" for the reverse-time process in Eq. (22)) describing evolution of marginal density for the Langevin diffusion SDE in Eq. (59) as follows

$$\begin{aligned}
-\frac{\partial}{\partial t}p(\mathbf{x},t) =& \nabla\cdot\left[\left(-\alpha^2\right)\frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla \log p(\mathbf{x},t) + \nabla\cdot\mathbf{D}(\mathbf{x},t)\right]p(\mathbf{x},t)\right] + \frac{1}{2}\alpha^2\nabla\cdot\left[\nabla\cdot\left[\mathbf{G}(\mathbf{x},t)\mathbf{G}^\top(\mathbf{x},t)p(\mathbf{x},t)\right]\right] \\
=& \nabla\cdot\left[\left(-\alpha^2\right)\frac{1}{2}\left[\mathbf{D}(\mathbf{x},t)\nabla p(\mathbf{x},t) + p(\mathbf{x},t)\nabla\cdot\mathbf{D}(\mathbf{x},t)\right]\right] + \frac{1}{2}\alpha^2\nabla\cdot\left[\nabla\cdot\left[\mathbf{D}(\mathbf{x},t)p(\mathbf{x},t)\right]\right] \\
=& \nabla\cdot\left[\left(-\alpha^2\right)\frac{1}{2}\nabla\cdot\left[\mathbf{D}(\mathbf{x},t)p(\mathbf{x},t)\right]\right] + \frac{1}{2}\alpha^2\nabla\cdot\left[\nabla\cdot\left[\mathbf{D}(\mathbf{x},t)p(\mathbf{x},t)\right]\right] \quad \text{// product rule of divergence} \\
=& 0
\end{aligned}$$

which simply says the effect of denoising and noise injection cancels out (hence no change to $p(\mathbf{x}, t)$). (We can do this similarly for the forward time version (Eq. (58)) as well)

Given the equivalence of PF ODE and reverse-time SDE in terms of keeping the same path of marginal probabilities, what is the difference? While in theory they are effectively equivalent, it is inevitable that in practice we introduce *additional* noise (e.g. error caused by time discretization, inaccuracy of the score model's prediction) over the course of simulation whose very purpose is to remove noise. Unlike PF ODE that takes deterministic steps, the stochasticity in reverse-time SDE enables it to actively explore and correct for existing and new noise, which increases the chance that $\mathbf{x}$ remains on the correct path.

11 Flow Matching

11.1 Match Flows, Not Scores

We previously mentioned the idea that we can choose different targets for the parametric score model which all lead to the same approximated score function. Recall the PF ODE in Eq. (39)

$$\mathrm{d}\mathbf{x}_t = \left[f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\nabla \log p(\mathbf{x}_t, t)\right]\mathrm{d}t$$

in which the target of the score model, $\nabla \log p(\mathbf{x}, t) = \mathbb{E}_{\mathbf{x}_0}\left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)|\mathbf{x}_t\right]$, can be expressed as a conditional expectation. Coincidentally, the dynamics of PF ODE itself turns out to be a conditional expectation as well:

$$
\begin{aligned}
& f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\nabla \log p(\mathbf{x}_t, t) \\
=& f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\mathbb{E}_{\mathbf{x}_0}\left[\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0)|\mathbf{x}_t\right] \\
=& f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\int p(\mathbf{x}_0|\mathbf{x}_t)\frac{s(t)\mathbf{x}_0 - \mathbf{x}_t}{s^2(t)\sigma^2(t)}\mathrm{d}\mathbf{x}_0 \quad \text{// Eq. (55)} \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\left[f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\frac{s(t)\mathbf{x}_0 - \mathbf{x}_t}{s^2(t)\sigma^2(t)}\right]\mathrm{d}\mathbf{x}_0 \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\left[f(t)\left(s(t)\mathbf{x}_0 + s(t)\sigma(t)\mathbf{z}\right) + \frac{1}{2}g^2(t)\frac{s(t)\sigma(t)\mathbf{z}}{s^2(t)\sigma^2(t)}\right]\mathrm{d}\mathbf{x}_0 \quad \text{// } \mathbf{x}_t = s(t)\mathbf{x}_0 + s(t)\sigma(t)\mathbf{z}\text{, Eq. (38), where } \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\left[\dot{s}(t)\mathbf{x}_0 + \dot{s}(t)\sigma(t)\mathbf{z} + \frac{1}{2}g^2(t)\frac{s(t)\sigma(t)\mathbf{z}}{s^2(t)\sigma^2(t)}\right]\mathrm{d}\mathbf{x}_0 \quad \text{// } \dot{s}(t) = s(t)f(t)\text{, Eq. (34). Dot denotes time-derivative} \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\left[\dot{s}(t)\mathbf{x}_0 + \dot{s}(t)\sigma(t)\mathbf{z} + \dot{\sigma}(t)s(t)\mathbf{z}\right]\mathrm{d}\mathbf{x}_0 \quad \text{// } g^2(t) = 2\sigma(t)\dot{\sigma}(t)s^2(t)\text{, Eq. (35)} \\
=& \int p(\mathbf{x}_0|\mathbf{x}_t)\left[\dot{s}(t)\mathbf{x}_0 + \dot{r}(t)\mathbf{z}\right]\mathrm{d}\mathbf{x}_0 \quad \text{// denote } r(t) := s(t)\sigma(t) \\
=& \mathbb{E}_{\mathbf{x}_0}\left[\dot{s}(t)\mathbf{x}_0 + \dot{r}(t)\mathbf{z}|\mathbf{x}_t\right] \qquad (60)
\end{aligned}
$$

and we can potentially learn a model to match this as opposed to the score function.

The expression inside the expectation is equal to the time-derivative of $\mathbf{x}_t = s(t)\mathbf{x}_0 + r(t)\mathbf{z}$, and $\mathbf{x}_t$ is a weighted interpolation of samples from two independent distributions $\mathbf{x}_0 \sim p_{\text{data}}$ and $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. For some choices of $f(t)$ and $g(t)$ (i.e. VP and sub-VP SDE/ODE), from which the weighting factors $s(t)$ and $r(t) = s(t)\sigma(t)$ are determined, there is an interesting relationship that if $s(t) = 1$, then $r(t) = 0$, or reversely if $s(t) = 0$, then $\sigma(t) = 1$ (i.e. Eq. (45) and (49)).

Base on this observation we are motivated to propose a more *general* target

$$\dot{\mathbf{x}}_t = \dot{a}(t)\mathbf{x}_0 + \dot{b}(t)\mathbf{x}_1 \qquad (61)$$

where

$$\mathbf{x}_t = a(t)\mathbf{x}_0 + b(t)\mathbf{x}_1 \qquad (62)$$

$\mathbf{x}_0 \in \mathbb{R}^n$ and $\mathbf{x}_1 \in \mathbb{R}^n$ are random variables sampled from two independent probability distributions π_0 and π_1. $a(t)$ and $b(t)$ are scalar-valued functions that are differentiable w.r.t. time t and have the property that $a(t)$ monotonically varies from 0 to 1 while $b(t)$ from 1 to 0. Intuitively, this defines a continuous flow that transforms $\mathbf{x}_t$ from π_0 to π_1 or symmetrically from from π_1 to π_0. Since the dot-notation denotes time-derivative, $\dot{\mathbf{x}}_t$ can be interpreted as the *velocity* of the flow.

We can learn a parametric *flow model* $\mathbf{v}_{\boldsymbol{\theta}}(\cdot, t) : \mathbb{R}^n \to \mathbb{R}^n$ to learn this target by minimizing the *Flow Matching* (FM) objective

$$L_t = \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_1}\left[\left\|\mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \left(\dot{a}(t)\mathbf{x}_0 + \dot{b}(t)\mathbf{x}_1\right)\right\|^2\right] \qquad (63)$$

Because the mapping from $(\mathbf{x}_0, \mathbf{x}_1)$ to $(\mathbf{x}_0, \mathbf{x}_t)$ is invertible [6], we can equivalently take the expectation in L_t w.r.t. $\mathbf{x}_0$ and $\mathbf{x}_t$ using the change of variable formula:

$$\begin{aligned}
L_t &= \iint p(\mathbf{x}_0, \mathbf{x}_1) \left\| \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \left(\dot{a}(t)\mathbf{x}_0 + \dot{b}(t)\mathbf{x}_1\right) \right\|^2 \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_1 \\
&= \iint p(\mathbf{x}_0, \mathbf{x}_t) \left| \det \frac{\partial(\mathbf{x}_0, \mathbf{x}_t)}{\partial(\mathbf{x}_0, \mathbf{x}_1)} \right| \left\| \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \left(\dot{a}(t)\mathbf{x}_0 + \dot{b}(t)\mathbf{x}_1\right) \right\|^2 \left| \det \frac{\partial(\mathbf{x}_0, \mathbf{x}_t)}{\partial(\mathbf{x}_0, \mathbf{x}_1)} \right|^{-1} \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t \\
&= \iint p(\mathbf{x}_0, \mathbf{x}_t) \left\| \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \dot{\mathbf{x}}_t \right\|^2 \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t
\end{aligned}$$

Since the loss function

$$L(\mathbf{x}_t, t) = \int p(\mathbf{x}_0, \mathbf{x}_t) \| \mathbf{v}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \dot{\mathbf{x}}_t \|^2 \mathrm{d}\mathbf{x}_0$$

for the given value of $\mathbf{x}_t$ is convex, it can be proved (i.e. by setting the gradient w.r.t. $\mathbf{v}_{\boldsymbol{\theta}}$ to zero) that the minimizer

$$\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) = \int p(\mathbf{x}_0 | \mathbf{x}_t) \dot{\mathbf{x}}_t \mathrm{d}\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0} \left[\dot{\mathbf{x}}_t | \mathbf{x}_t \right] \tag{64}$$

Notably, the way $\mathbf{x}_t$ is defined in Eq. (62) implicitly induces a stochastic process $\{\mathbf{x}_t\}$: For each time point t, $\mathbf{x}_t$ follows a distribution $p(\mathbf{x}_t, t)$ determined jointly by the distributions π_0, π_1 and the weighting factors $a(t)$, and $b(t)$. Although it's difficult to obtain the analytical form of $p(\mathbf{x}_t, t)$, we can derive a PDE characterizing the evolution of $p(\mathbf{x}_t, t)$ using the optimal flow model $\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t)$. The idea is to prove the Continuity Equation (i.e. Fokker-Planck Equation with zero diffusion) for $p(\mathbf{x}_t, t)$. We use the same trick that was used to prove FPE by using an arbitrary auxiliary function $\varphi(\mathbf{x}_t)$ that is differentiable w.r.t. $\mathbf{x}_t$.

Let's denote $\mathbf{v} = \mathbf{v}_{\boldsymbol{\theta}^*}$. Starting from the integral of the product between φ and the divergence $\nabla \cdot [\mathbf{v}(\mathbf{x}_t, t) p(\mathbf{x}_t, t)]$, we have

$$\begin{aligned}
&\int \varphi(\mathbf{x}_t) \left[\sum_{i=1}^n \frac{\partial \left[v_i(\mathbf{x}_t, t) p(\mathbf{x}_t, t) \right]}{\partial x_i} \right] \mathrm{d}\mathbf{x}_t \\
=& \sum_{i=1}^n \int \varphi(\mathbf{x}_t) \frac{\partial \left[v_i(\mathbf{x}_t, t) p(\mathbf{x}_t, t) \right]}{\partial x_i} \mathrm{d}\mathbf{x}_t \quad \text{// } x_i \text{ denotes the } i\text{-th component of } \mathbf{x}_t \\
=& -\sum_{i=1}^n \int v_i(\mathbf{x}_t, t) p(\mathbf{x}_t, t) \frac{\partial \varphi}{\partial x_i} \mathrm{d}\mathbf{x}_t \quad \text{// integration by parts, Eq. (7)} \\
=& -\sum_{i=1}^n \int \left[\int p(\mathbf{x}_0 | \mathbf{x}_t) \frac{\mathrm{d}x_i}{\mathrm{d}t} \mathrm{d}\mathbf{x}_0 \right] p(\mathbf{x}_t, t) \frac{\partial \varphi}{\partial x_i} \mathrm{d}\mathbf{x}_t \quad \text{// Eq. (64)} \\
=& -\iint p(\mathbf{x}_0, \mathbf{x}_t) \left[\frac{1}{\mathrm{d}t} \sum_{i=1}^n \frac{\partial \varphi}{\partial x_i} \mathrm{d}x_i \right] \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t \quad \text{// } p(\mathbf{x}_0, \mathbf{x}_t) = p(\mathbf{x}_0 | \mathbf{x}_t) p(\mathbf{x}_t, t) \\
=& -\iint p(\mathbf{x}_0, \mathbf{x}_t) \frac{1}{\mathrm{d}t} \mathrm{d}\varphi(\mathbf{x}_t) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t \quad \text{// } \mathrm{d}\varphi(\mathbf{x}_t) = \langle \nabla \varphi(\mathbf{x}_t), \mathrm{d}\mathbf{x}_t \rangle \\
=& -\frac{\mathrm{d}}{\mathrm{d}t} \iint p(\mathbf{x}_0, \mathbf{x}_t) \varphi(\mathbf{x}_t) \mathrm{d}\mathbf{x}_0 \mathrm{d}\mathbf{x}_t
\end{aligned}$$

[6] Because $\mathbf{x}_1 = \left[\mathbf{x}_t - a(t)\mathbf{x}_0 \right] / b(t)$, assuming $b(t) \neq 0$. Likewise the mapping from $(\mathbf{x}_0, \mathbf{x}_1)$ to $(\mathbf{x}_1, \mathbf{x}_t)$ is invertible too.

$$= -\frac{\mathrm{d}}{\mathrm{d}t}\int p(\mathbf{x}_t, t)\varphi(\mathbf{x}_t)\mathrm{d}\mathbf{x}_t \;\; // \; p(\mathbf{x}_t, t) := p(\mathbf{x}_t) = \int p(\mathbf{x}_0, \mathbf{x}_t)\mathrm{d}\mathbf{x}_0$$
$$= -\int \varphi(\mathbf{x}_t)\frac{\partial}{\partial t}p(\mathbf{x}_t, t)\mathrm{d}\mathbf{x}_t \tag{65}$$

The only way for Eq. (65) to hold for any function φ is the following PDE

$$\frac{\partial}{\partial t}p(\mathbf{x}_t, t) = -\sum_{i=1}^{n}\frac{\partial}{\partial x_i}\left[v_i(\mathbf{x}_t, t)p(\mathbf{x}_t, t)\right] \tag{66}$$

which is solved by $p(\mathbf{x}_t, t)$, the marginal distribution of $\mathbf{x}_t$.

Note that Eq. (66) is also the Continuity Equation for the stochastic process $\{\mathbf{z}_t\}$ (with marginal probability $p(\mathbf{z}_t, t)$) induced by the following ODE

$$\mathrm{d}\mathbf{z}_t = \mathbf{v}(\mathbf{z}_t, t)\mathrm{d}t \tag{67}$$

whose marginal density is $p(\mathbf{z}_t, t)$. Assuming Eq. (67) and the stochastic process $\{\mathbf{x}_t\}$ (defined in Eq. (62)) are initialized with the same distribution (e.g. π_0), it must hold that $p(\mathbf{x}_t, t)$ is equal to $p(\mathbf{z}_t, t)$ for all t.

Given such "marginal preserving" property of the flow model $\mathbf{v}$, we can recover the distribution π_0 in a way similar to PF ODE simulation by initializing Eq. (67) with a sample from π_1 and simulating it backwards in time, or similarly recover π_1 by starting from π_0.

The original formulation of matching the dynamics of ODE in Eq. (60) is simply a special case of Flow Matching. By setting $a(t) = s(t)$ and $b(t) = r(t)$, and $\pi_0 = p_{\text{data}}$, $\pi_1 = \mathcal{N}(\mathbf{0}, \mathbf{I})$, we have

$$\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) = \mathbb{E}_{\mathbf{x}_0}\left[\dot{\mathbf{x}}_t|\mathbf{x}_t\right] = \mathbb{E}_{\mathbf{x}_0}\left[\dot{s}(t)\mathbf{x}_0 + \dot{r}(t)\mathbf{z}|\mathbf{x}_t\right] = f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\nabla\log p(\mathbf{x}_t, t)$$

Although stochastic processes $\{\mathbf{x}_t\}$ and $\{\mathbf{z}_t\}$ share the same path of marginal densities, it's important to note that the joint probability density over multiple points on the trajectory are generally different. Specifically, $p(\mathbf{x}_0, \mathbf{x}_1)$ can be factorized as $p(\mathbf{x}_0)p(\mathbf{x}_1)$ since π_0 and π_1 are independent. In contrast, any two random variables $\mathbf{z}_u$ and $\mathbf{z}_v$ in $\{\mathbf{z}_t\}$ are completely dependent on each other since they are related according to

$$\mathbf{z}_u = \mathbf{z}_v + \int_v^u \mathbf{v}(\mathbf{z}_t, t)\mathrm{d}t \tag{68}$$

In principle, Flow Matching is a more generalized formalism than "score matching + SDE/ODE simulation" since it provides a way to transform back and forth between *arbitrary* probability distributions, without restricting one of them to be a standard Normal distribution. In addition, we get to choose the form of $a(t)$ and $b(t)$ which shape the curvature of the trajectory that $\mathbf{x}_t$ will travel through. As we will see shortly, this "straightness" will impact how fast and robustly we can sample from the data distribution.

11.2 Rectified Flow

Rectified Flow (RF) is a special case of Flow Matching where we simply set

$$a(t) = 1 - t \text{ and } b(t) = t$$

so Eq. (62) and (61) reduces to

$$\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$$

and

$$\dot{\mathbf{x}}_t = \mathbf{x}_1 - \mathbf{x}_0 = \frac{\mathbf{x}_t - \mathbf{x}_0}{t} = \frac{\mathbf{x}_1 - \mathbf{x}_t}{1-t}$$

where the time variable $t \in [0,1]$ varies continuously from 0 to 1 to make sure $\mathbf{x}_t$ coincides with $\mathbf{x}_0$ and $\mathbf{x}_1$ at two ends of the trajectory. The flow model becomes

$$\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t) = \int p(\mathbf{x}_0|\mathbf{x}_t)(\mathbf{x}_1 - \mathbf{x}_0)\mathrm{d}\mathbf{x}_0 = \mathbb{E}_{\mathbf{x}_0}\left[\mathbf{x}_1 - \mathbf{x}_0|\mathbf{x}_t\right] \tag{69}$$

In Rectified Flow, $\mathbf{x}_t$ is a linear interpolation between two random samples from π_0 and π_1, and the velocity $\dot{\mathbf{x}}_t$ no longer depends on t, meaning $\mathbf{x}_t$ travels in constant speed. The flow model $\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t)$ is supposed to learn the averaged direction of the straight line at time t that passes through the input $\mathbf{x}_t$ and connects $\mathbf{x}_0$ and $\mathbf{x}_1$.

The fact that $\mathbf{v}_{\boldsymbol{\theta}^*}(\mathbf{x}_t, t)$ is trained to match $\mathbf{x}_1 - \mathbf{x}_0$ enables $\mathbf{z}_t$ to flow as straight as possible when evolving according to ODE in Eq. (67). Theoretically, this has the effect of lowering the cost of transporting the probability mass from π_0 to π_1 (or vice versa), and in practice the numerical simulation is less susceptible to error caused by coarse discretization of time for nonlinear paths with large curvatures (e.g. VP and VE ODEs), and ideally we can use it to sample from the data distribution in as few as a single step.

To illustrate what lowering the cost of transportation means, consider a scalar-valued function $c(\cdot)$ that takes as input the distance between two points (e.g. $\mathbf{x}_1 - \mathbf{x}_0$) and outputs the cost of moving one point to the position of another.

Assume $c(\cdot)$ is convex in the interval $[0,1]$, e.g. $c(\cdot) = \|\cdot\|^p$ for $p \geq 1$. We are going to prove the expected value of $c(\mathbf{x}_1 - \mathbf{x}_0)$ is no smaller than that of $c(\mathbf{z}_1 - \mathbf{z}_0)$:

$$\begin{aligned}
\mathbb{E}_{\mathbf{z}_1}\left[c(\mathbf{z}_1 - \mathbf{z}_0)\right] &= \mathbb{E}_{\mathbf{z}_1}\left[c\left(\int_0^1 \mathbf{v}(\mathbf{z}_t, t)\mathrm{d}t\right)\right] \quad \text{//Eq. (68): } \mathbf{z}_1 - \mathbf{z}_0 = \int_0^1 \mathbf{v}(\mathbf{z}_t, t)\mathrm{d}t \\
&\leq \mathbb{E}_{\mathbf{z}_1}\left[\int_0^1 c\left(\mathbf{v}(\mathbf{z}_t, t)\right)\mathrm{d}t\right] \quad \text{// Jensen's inequality: } c(\cdot) \text{ is convex} \\
&= \mathbb{E}_{\mathbf{x}_1}\left[\int_0^1 c\left(\mathbf{v}(\mathbf{x}_t, t)\right)\mathrm{d}t\right] \quad \text{// interchangeablility of } \mathbf{x}_t \text{ and } \mathbf{z}_t\text{: } p(\mathbf{x}_t, t) = p(\mathbf{z}_t, t) \\
&= \mathbb{E}_{\mathbf{x}_1}\left[\int_0^1 c\left(\mathbb{E}_{\mathbf{x}_0}\left[\mathbf{x}_1 - \mathbf{x}_0|\mathbf{x}_t\right]\right)\mathrm{d}t\right] \quad \text{substitute in Eq. (69)} \\
&\leq \mathbb{E}_{\mathbf{x}_1}\left[\int_0^1 \mathbb{E}_{\mathbf{x}_0}\left[c(\mathbf{x}_1 - \mathbf{x}_0)|\mathbf{x}_t\right]\mathrm{d}t\right] \quad \text{// Jensen's inequality: } c(\cdot) \text{ is convex} \\
&= \int_0^1 \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t}\left[c(\mathbf{x}_1 - \mathbf{x}_0)\right]\mathrm{d}t \\
&= \int_0^1 \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_1}\left[c(\mathbf{x}_1 - \mathbf{x}_0)\right]\mathrm{d}t \quad \text{// change of variables: } (\mathbf{x}_0, \mathbf{x}_t) \text{ to } (\mathbf{x}_0, \mathbf{x}_1) \\
&= \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_1}\left[c(\mathbf{x}_1 - \mathbf{x}_0)\right]
\end{aligned}$$

11.3 VP, sub-VP ODE as Special Case

VP, sub-VP ODE, and Rectified Flow can be viewed as special cases of Flow Matching:

- **VP ODE**: $a(t) = s(t)$, $b(t) = r(t) = s(t)\sigma(t)$, $a^2(t) + b^2(t) = s^2(t) + 1 - s^2(t) = 1$
- **sub-VP ODE**: $a(t) = s(t)$, $b(t) = \sqrt{r(t)} = \sqrt{s(t)\sigma(t)}$, $a^2(t) + b^2(t) = s^2(t) + \sqrt{s^2(t)\sigma^2(t)} = s^2(t) + \sqrt{1 - 2s^2(t) + s^4(t)} = s^2(t) + \sqrt{[1 - s^2(t)]^2} = s^2(t) + 1 - s^2(t) = 1$
- **Rectified Flow**: $a(t) = t$, $b(t) = 1 - t$, $a(t) + b(t) = 1$

12 Consistency Model

As we have pointed out, PF ODE (and reverse-time SDE) can be used to sample from an arbitrary data distribution. Consider the PF ODE (with $g(t) = \sqrt{2t}$ and $f(t) = 0$, see Eq. (39))

$$\mathrm{d}\mathbf{x}_t = -t\nabla \log p(\mathbf{x}_t, t)\mathrm{d}t \tag{70}$$

initialized with $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$. The marginal density of the induced stochastic process is given by

$$p(\mathbf{x}_t, t) = \int p_{\text{data}}(\mathbf{x}) p(\mathbf{x}_t|\mathbf{x})\mathrm{d}\mathbf{x}$$

where $p(\mathbf{x}_t|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \mathbf{I}t^2)$[7], and the final distribution $p(\mathbf{x}_T, T)$ is approximately $\mathcal{N}(\mathbf{0}, \mathbf{I}T^2)$ when T is large enough.

By initializing Eq. (70) with the final value $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}T^2)$, we can simulate it backwards in time using a pre-trained score model $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ as a proxy for $\nabla \log p(\mathbf{x}_t, t)$, which leaves us with a *solution path* $\{\mathbf{x}_t\}_{t\in[\epsilon,T]}$ where the end point is roughly a sample from $p_{\text{data}}(\mathbf{x})$. [8]

Despite being a viable approach, the downside is that we have to obediently follow the simulation process one step at a time before arriving at the data distribution. Note that all the values along a solution path are projected to the same end point $\mathbf{x}_\epsilon$. This suggests a predictable relationship between each value $\mathbf{x}_t \in \{\mathbf{x}_t\}$ and $\mathbf{x}_\epsilon$, which can be formalized as the *consistency function* $\mathbf{c}(\mathbf{x}_t, t) : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}^n$ such that for any $t, t' \in [\epsilon, T]$, $\mathbf{c}(\mathbf{x}_t, t) = \mathbf{c}(\mathbf{x}_{t'}, t') = \mathbf{x}_\epsilon$, in other words the outputs are *self-consistent* for inputs from the same solution path[9]. By learning a consistency model $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ that approximates $\mathbf{c}(\mathbf{x}_t, t)$, we'll be able to jump directly to $p_{\text{data}}(\mathbf{x})$ without resorting to ODE simulation any more.

Formally, a *consistency model* is defined as

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

where $\tilde{\mathbf{c}}_{\boldsymbol{\theta}}(\mathbf{x}, t)$ is typically a neural network with the same input and output dimensionality. Crutially, this definition enforces $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_\epsilon, \epsilon) = \mathbf{x}_\epsilon$, which prevent the consistency model from collapsing to a trivial solution that maps all inputs to a fixed point (e.g. zero).

[7] Given $f(t) = 0$, $g(t) = \sqrt{2t}$, we have $s(t) = 1$ and $\sigma(t) = t$ (Eq. (34) and (35)), so $p(\mathbf{x}_t|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \mathbf{I}t^2)$ (Eq. (38))

[8] We stop the simulation at a small positive time before reaching 0 (e.g. $\epsilon = 10^{-3}$) to avoid numerical instability (i.e. it can be verified that denoising score function (Eq. (55)) is undefined at $t = 0$).

[9] Consistency function is dependent on a PF ODE.

12.1 Training

To train consistency model, we consider a discretization scheme $0 < t_1 = \epsilon < t_2 < \cdots < t_N = T$, and minimize the inconsistency between model outputs from two randomly sampled adjacent points $\mathbf{x}_{t_{n+1}}$ and $\mathbf{x}_{t_n}$ along a trajectory [10]. The self-consistency property can be enforced in two ways:

- In *Consistency Distillation*, we assume the availability of a pre-trained score model $\mathbf{s}_{\boldsymbol{\phi}}$, and compute the loss as

$$\mathcal{L}_{\mathrm{CD}}^N(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi}) = \mathbb{E}\left[\lambda(t_n) d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n))\right] \tag{71}$$

  where the expectation is taken w.r.t. $\mathbf{x}_{t_{n+1}} \sim \mathcal{N}(\mathbf{x}, t_{t_{n+1}}^2 \mathbf{I})$ in which $\mathbf{x}$ is sampled from p_{data}, and n is sampled from $\{1, \cdots, N-1\}$ with equal probability. $\boldsymbol{\theta}^-$ denotes a frozen version of the previous value of $\boldsymbol{\theta}$, $\lambda(\cdot) \in \mathbb{R}_+$ is a positive weighting function, and $d(\cdot, \cdot)$ is a metric function[11] such that $\forall \mathbf{x}, \mathbf{y}$, $d(\mathbf{x}, \mathbf{y}) \geq 0$ and $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\mathbf{x} = \mathbf{y}$.

  $\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}$ denotes the estimate of $\mathbf{x}_{t_n}$ by computing one step of backward simulation of Eq. (70) from $\mathbf{x}_{t_{n+1}}$ using $\mathbf{s}_{\boldsymbol{\phi}}$:

$$\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}} := \mathbf{x}_{t_{n+1}} + t_{n+1} \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1})(t_{n+1} - t_n) \tag{72}$$

- In *Consistency Training*, we no longer need the pre-trained score model, and the loss is defined as

$$\mathcal{L}_{\mathrm{CT}}^N(\boldsymbol{\theta}, \boldsymbol{\theta}^-) = \mathbb{E}\left[\lambda(t_n) d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x} + t_{n+1}\mathbf{z}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x} + t_n \mathbf{z}, t_n))\right] \tag{73}$$

  where the expectation is taken w.r.t. $\mathbf{x} \sim p_{\mathrm{data}}$, $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and n is sampled from $\{1, \cdots, N-1\}$ with equal probability. $\lambda(\cdot)$, $d(\cdot, \cdot)$, and $\boldsymbol{\theta}^-$ are defined similarly as Consistency Distillation.

We will prove that *by minimizing either $\mathcal{L}_{CD}^N$ or $\mathcal{L}_{CT}^N$, the learned model $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ will be asymptotically equal to the groundtruth consistency model.* Below are the notations used in the proof:

- $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}, t)$: consistency model parameterized by $\boldsymbol{\theta}$.
- $\mathbf{c}(\mathbf{x}, t; \boldsymbol{\phi})$: consistency function of the PF ODE in Eq. (54), where the groundtruth score function is replaced by the pre-trained score model $\mathbf{s}_{\boldsymbol{\theta}}$ parameterized by$\boldsymbol{\phi}$.
- Given multivariate function $\mathbf{h}(\mathbf{x}, \mathbf{y})$, $\partial_1 \mathbf{h}(\mathbf{x}, \mathbf{y})$ and $\partial_2 \mathbf{h}(\mathbf{x}, \mathbf{y})$ denote the Jacobian of $\mathbf{h}$ w.r.t. $\mathbf{x}$ and $\mathbf{y}$ respectively.

12.1.1 Justifying Consistency Distillation Loss

Suppose a consistency model $\mathbf{c}_{\boldsymbol{\theta}}$ achieves zero loss:

$$\begin{aligned}
\mathcal{L}_{\mathrm{CD}}^N(\boldsymbol{\theta}, \boldsymbol{\theta}; \boldsymbol{\phi}) =& \mathbb{E}\left[\lambda(t_n) d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1})), \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n)\right] \\
=& \int p_{\mathrm{data}}(\mathbf{x}) \mathcal{N}(\mathbf{x}_{t_{n+1}}; \mathbf{x}, t_{t_{n_1}}^2 \mathbf{I}) \lambda(t_n) d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n)) \mathrm{d}\mathbf{x}_{t_{n+1}} = 0
\end{aligned}$$

[10] Note that the naive alternative — directly optimizing the regression loss between the estimate $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$, and the groundtruth $\mathbf{c}(\mathbf{x}_t, t)$ — is undesirable, since knowing the value of the latter itself requires simulation of the ODE.

[11] Typically we choose $\lambda(t_n) = 1$ and $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^2$

For this to be true, it's necessary that $d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n)) = 0$, since $\lambda(t_n)$, $p_{\text{data}}(\mathbf{x})$, $\mathcal{N}(\mathbf{x}_{t_{n+1}}; \mathbf{x}, t^2_{t_{n_1}}\mathbf{I})$ are all positive. Because $d(\mathbf{x}, \mathbf{y}) = 0 \iff \mathbf{x} = \mathbf{y}$, it must hold that

$$\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}) = \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n)$$

Let $\mathbf{e}_n$ denote the discrepancy between the output of the learned model and groundtruth consistency function at time t_n

$$\mathbf{e}_n = \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n) - \mathbf{c}(\mathbf{x}_{t_n}, t_n; \boldsymbol{\phi})$$

Notice $\mathbf{e}_{n+1}$ can be written in terms of $\mathbf{e}_n$:

$$\begin{aligned}
\mathbf{e}_{n+1} &= \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}) - \mathbf{c}(\mathbf{x}_{t_{n+1}}, t_{n+1}; \boldsymbol{\phi}) \\
&= \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n) - \mathbf{c}(\mathbf{x}_{t_n}, t_n; \boldsymbol{\phi}) \text{ //self-consistency property of } \mathbf{c}(\mathbf{x}_t, t) \\
&= \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n) - \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n) + \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n) - \mathbf{c}(\mathbf{x}_{t_n}, t_n; \boldsymbol{\phi}) \\
&= \mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n) - \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n) + \mathbf{e}_n
\end{aligned}$$

We further make the following assumptions:

- $\mathbf{c}_{\boldsymbol{\theta}}$ satisfies Lipschitz condition: there exists $L > 0$ such that $\forall t \in [\epsilon, T]$, $\mathbf{x}$ and $\mathbf{y}$, we have $\|\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}, t) - \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{y}, t)\| \leq L\|\mathbf{x} - \mathbf{y}\|$
- The discretization error of simulating one step of ODE is bounded by $O((t_{n+1} - t_n)^{p+1})$ for $n \in \{1, \cdots, N-1\}$, namely $\|\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n} - \mathbf{x}_{t_n}\| \leq O((t_{n+1} - t_n)^{p+1})$ for $p \geq 1$.

Therefore

$$\begin{aligned}
\|\mathbf{e}_{n+1}\| - \|\mathbf{e}_n\| \leq& \|\mathbf{e}_{n+1} - \mathbf{e}_n\| \\
=& \|\mathbf{c}_{\boldsymbol{\theta}}(\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n}, t_n) - \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n)\| \\
\leq& L\|\hat{\mathbf{x}}^{\boldsymbol{\phi}}_{t_n} - \mathbf{x}_{t_n}\| \\
=& L \cdot O((t_{n+1} - t_n)^{p+1}) \\
=& O((t_{n+1} - t_n)^{p+1})
\end{aligned} \tag{74}$$

By definition of consistency model and consistency function, $\mathbf{e}_1$ must be zero at $t_1 = \epsilon$:

$$\mathbf{e}_1 = \mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_1}, t_1) - \mathbf{c}(\mathbf{x}_{t_1}, t_1; \boldsymbol{\phi}) = \mathbf{x}_{t_1} - \mathbf{x}_{t_1} = \mathbf{0}$$

Let Δt be the biggest step size of the discretization scheme, namely, $\Delta t = \max_n\{t_{n+1} - t_n\}$ for $n \in \{1, \cdots, N-1\}$. We can apply Eq. (58) recursively to obtain

$$\begin{aligned}
\|\mathbf{e}_n\| \leq& \|\mathbf{e}_1\| + \sum_{k=1}^{n-1} O((t_{k+1} - t_k)^{p+1}) \\
=& \sum_{k=1}^{n-1} O((t_{k+1} - t_k)^{p+1}) \\
=& \sum_{k=1}^{n-1} (t_{k+1} - t_k) O((t_{k+1} - t_k)^{p})
\end{aligned}$$

$$
\begin{aligned}
&\leq \sum_{k=1}^{n-1}(t_{k+1}-t_k)O((\Delta t)^p) \\
&= O((\Delta t)^p)\sum_{k=1}^{n-1}(t_{k+1}-t_k) \\
&= O((\Delta t)^p)(t_n - t_1) \\
&\leq O((\Delta t)^p)(T-\epsilon) \\
&= O((\Delta t)^p)
\end{aligned}
$$

This implies the necessary condition that Consistency Distillation loss equals zero is that the difference between the learned model $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_n}, t_n)$ and the target $\mathbf{c}(\mathbf{x}_{t_n}, t_n; \boldsymbol{\phi})$ can be made arbitrarily close to zero as long as the step size is sufficiently small.

12.1.2 Justifying Consistency Training Loss

To justify Consistency Training loss, we are going to show that it is asymptotically the same as the Consistency Distillation loss as step size Δt approaches zero. The idea is to apply Taylor expansion on the model $\mathbf{c}_{\boldsymbol{\theta}}$ and the metric function $d(\cdot,\cdot)$ (which are assumed to be twice-differentiable), and use the following empirical score as a proxy to the score function since it is an unbiased estimator (using Eq. (53)):

$$
\nabla \log p(\mathbf{x}_{t_{n+1}}, t_{n+1}) = \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}) = \mathbb{E}_{\mathbf{x}}\left[\frac{\mathbf{x}-\mathbf{x}_{t_{n+1}}}{t_{n+1}^2}\middle|\mathbf{x}_{t_{n+1}}\right]
$$

Recall that the Consistency Distillation loss

$$
\mathcal{L}_{\mathrm{CD}}^N(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi}) = \mathbb{E}\left[\lambda(t_n) d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n))\right]
$$

We first expand $\mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n)$ in Taylor series up to first derivative at input $\mathbf{x}_{t_{n+1}}, t_{n+1}$ where we make the substitution $\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}} = \mathbf{x}_{t_{n+1}} + (t_{n+1}-t_n)t_{n+1}\nabla \log p(\mathbf{x}_{t_{n+1}}, t_{n+1})$,

$$
\begin{aligned}
&\mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n) \\
=&\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}} + (t_{n+1}-t_n)t_{n+1}\nabla \log p(\mathbf{x}_{t_{n+1}}, t_{n+1}), t_n) \\
=&\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1}) + \partial_1 \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})(t_{n+1}-t_n)t_{n+1}\nabla \log p(\mathbf{x}_{t_{n+1}}, t_{n+1}) \\
&+ \partial_2 \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})(t_n - t_{n+1}) + o(|t_{n+1}-t_n|)
\end{aligned}
$$

Then we expand $d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n)\right)$ at input $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})$,

$$
\begin{aligned}
&d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}_{t_n}^{\boldsymbol{\phi}}, t_n)\right) \\
=&d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})\right) + \partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}}, t_{n+1}), \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})\right) \\
&\left[\partial_1 \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})(t_{n+1}-t_n)t_{n+1}\nabla \log p(\mathbf{x}_{t_{n+1}}, t_{n+1}) + \partial_2 \mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}, t_{n+1})(t_n - t_{n+1}) + o(|t_{n+1}-t_n|)\right]
\end{aligned}
$$

The Consistency Distillation loss can be derived as

$$
\mathcal{L}_{\mathrm{CD}}^N(\boldsymbol{\theta}, \boldsymbol{\theta}^-; \boldsymbol{\phi})
$$

$$
\begin{aligned}
=&\mathbb{E}\left[\lambda(t_n)d(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\hat{\mathbf{x}}^{\phi}_{t_n},t_n))\right]\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\right]\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_1\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_{n+1}-t_n)t_{n+1}\nabla\log p(\mathbf{x}_{t_{n+1}},t_{n+1})\right]\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_2\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_n-t_{n+1})\right]\\
&+\mathbb{E}\left[o(|t_{n+1}-t_n|)\right]\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\right]\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_1\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_{n+1}-t_n)t_{n+1}\mathbb{E}\left[\frac{\mathbf{x}-\mathbf{x}_{t_{n+1}}}{t_{n+1}^2}\middle|\mathbf{x}_{t_{n+1}}\right]\right]\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_2\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_n-t_{n+1})\right]\\
&+\mathbb{E}\left[o(|t_{n+1}-t_n|)\right]\ \text{// plug in empirical score}\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\right]\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_1\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_{n+1}-t_n)\left(\frac{\mathbf{x}-\mathbf{x}_{t_{n+1}}}{t_{n+1}}\right)\right]\ \text{// total expectation}\\
&+\mathbb{E}\left[\lambda(t_n)\partial_2 d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})\right)\partial_2\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}},t_{n+1})(t_n-t_{n+1})\right]\\
&+\mathbb{E}\left[o(|t_{n+1}-t_n|)\right]\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_{t_{n+1}},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}_{t_{n+1}}+(t_{n+1}-t_n)(\mathbf{x}-\mathbf{x}_{t_{n+1}})/t_{n+1},t_n)\right)\right]+\mathbb{E}\left[o(|t_{n+1}-t_n|)\right]\ \text{// reverse Taylor expn.}\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}+t_{n+1}\mathbf{z},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}+t_{n+1}\mathbf{z}+(t_n-t_{n+1})\mathbf{z},t_n)\right)\right]+\mathbb{E}\left[o(|t_{n+1}-t_n|)\right]\ \text{// } \mathbf{x}_{t_{n+1}}=\mathbf{x}+t_{n+1}\mathbf{z}\\
=&\mathbb{E}\left[\lambda(t_n)d\left(\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}+t_{n+1}\mathbf{z},t_{n+1}),\mathbf{c}_{\boldsymbol{\theta}^-}(\mathbf{x}+t_n\mathbf{z},t_n)\right)\right]+\mathbb{E}\left[o(\Delta t)\right]\\
=&\mathcal{L}^N_{\mathrm{CT}}(\boldsymbol{\theta},\boldsymbol{\theta}^-)+o(\Delta t)
\end{aligned}
$$

13 Neural ODE

Neural Ordinary Differential Equation (Neural ODE)[12] is a first-of-its-kind idea that generalizes the conventional feed-forward neural networks from having discretely finite number of hidden layers to ones with continuously infinite number of layers, where parameters are shared across all layers.

For example, given input vector $\mathbf{x}_0$ and the dynamics of hidden layer $\mathrm{d}\mathbf{x}_t = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\mathrm{d}t$, the forward pass can be implemented as the numerical integration (aka solving or simulating):

$$\mathbf{x}(T) = \mathbf{x}(0) + \int_0^T \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}(t), t)\mathrm{d}t$$

where each hidden layer updates $\mathbf{x}_t$ by $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\Delta t$ until the sum of Δt reaches T.

Given the scalar loss L defined as

$$L(\mathbf{x}(T)) = L\left(\mathbf{x}(0) + \int_0^T \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}(t), t)\mathrm{d}t\right)$$

the backward pass needs to compute the gradient of L w.r.t. each intermediate hidden state $\mathbf{x}(t)$, which is called the *adjoint* state $\mathbf{a}(t)$[13]:

$$\mathbf{a}(t) = \frac{\partial L}{\partial \mathbf{x}(t)}$$

[12] We briefly discuss Neural ODE here since PF ODE and Flow Matching are closely related to it.

[13] $\frac{\partial \cdot}{\partial \cdot}$ denote Jacobian matrix. For example, $\mathbf{a}(t) = \frac{\partial L}{\partial \mathbf{x}(t)}$ is a $1 \times n$ row matrix.

According to chain rule of derivative:

$$\mathbf{a}(t) = \frac{\partial L}{\partial \mathbf{x}(t)} = \frac{\partial L}{\partial \mathbf{x}(t+\epsilon)} \frac{\partial \mathbf{x}(t+\epsilon)}{\partial \mathbf{x}(t)} = \mathbf{a}(t+\epsilon) \frac{\partial \mathbf{x}(t+\epsilon)}{\partial \mathbf{x}(t)}$$

It can be shown that $\mathbf{a}(t)$ evolves according to an ODE

$$\begin{aligned}
\frac{\partial}{\partial t}\mathbf{a}(t) &= \lim_{\epsilon \to 0^+} \frac{1}{\epsilon} \left[\mathbf{a}(t+\epsilon) - \mathbf{a}(t)\right] \quad \text{// definition of derivative} \\
&= \lim_{\epsilon \to 0^+} \frac{1}{\epsilon} \left[\mathbf{a}(t+\epsilon) - \mathbf{a}(t+\epsilon) \frac{\partial \mathbf{x}(t+\epsilon)}{\partial \mathbf{x}(t)}\right] \\
&= \lim_{\epsilon \to 0^+} \frac{1}{\epsilon} \left[\mathbf{a}(t+\epsilon) - \mathbf{a}(t+\epsilon) \frac{\partial}{\partial \mathbf{x}(t)} \left[\mathbf{x}(t) + \epsilon \mathbf{f}(\mathbf{x}(t), t)\right]\right] \quad \text{// Taylor expand } \mathbf{x}(t+\epsilon) \text{ around } t \\
&= \lim_{\epsilon \to 0^+} \frac{1}{\epsilon} \left[\mathbf{a}(t+\epsilon) - \mathbf{a}(t+\epsilon) \left[\mathbf{I} + \epsilon \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}\right]\right] \\
&= \lim_{\epsilon \to 0^+} -\mathbf{a}(t+\epsilon) \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)} \\
&= -\mathbf{a}(t) \frac{\partial \mathbf{f}(\mathbf{x}(t), t)}{\partial \mathbf{x}(t)}
\end{aligned}$$

This ODE can be simulated backwards in time to compute the gradient w.r.t. the parameter $\boldsymbol{\theta}$ for training the Neural ODE.

13.1 Instantaneous Change of Variable Formula

We know that given a stochastic process driven by ODE

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\mathrm{d}t$$

the Continuity Equation

$$\frac{\partial}{\partial t} p(\mathbf{x}, t) = -\sum_{i=1}^{n} \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t)\right]$$

specifies the time evolution of a probability density $p(\mathbf{x}, t)$. However, this divergence term involves $p(\mathbf{x}, t)$ which makes it intractable to compute and requires approximation using finite difference. It turns out we can derive another PDE that describes the evolution of the the *logarithm* of $p(\mathbf{x}, t)$:

Let $\mathbf{h} = \mathbf{x}(t+\epsilon)$ denote the value of $\mathbf{x}$ at time $t+\epsilon$ by simulating the ODE from t to $t+\epsilon$. By definition,

$$\mathbf{x}(t+\epsilon) = \mathbf{x}(t) + \int_t^{t+\epsilon} \mathbf{f}(\mathbf{x}(z), z)\mathrm{d}z$$

Since $\mathbf{h}$ is automatically invertible (i.e. simulating the ODE backwards from $t+\epsilon$ to t gives back $\mathbf{x}$), the probability density of $\mathbf{h}$ can be expressed as the following

$$p(\mathbf{h}) = p(\mathbf{x}) \left|\det \frac{\partial \mathbf{x}}{\partial \mathbf{h}}\right| = p(\mathbf{x}) \left|\det \frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|^{-1}$$

using change of variables formula.

The derivation is as follows:

$$
\begin{aligned}
&\frac{\partial}{\partial t}\log p(\mathbf{x},t)\\
&=\lim_{\epsilon\to 0^+}\frac{\log p(\mathbf{h})-\log p(\mathbf{x})}{\epsilon} \quad \text{// definition of derivative: } \mathbf{h}=\mathbf{x}(t+\epsilon)\\
&=\lim_{\epsilon\to 0^+}\frac{\log p(\mathbf{x})-\log\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|-\log p(\mathbf{x})}{\epsilon} \quad \text{// change of variables}\\
&=-\lim_{\epsilon\to 0^+}\frac{\frac{\partial}{\partial\epsilon}\log\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|}{\frac{\partial}{\partial\epsilon}\epsilon} \quad \text{// L'Hopital rule}\\
&=-\lim_{\epsilon\to 0^+}\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|^{-1}\frac{\partial}{\partial\epsilon}\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right| \quad \text{// derivative chain rule}\\
&=-\cancel{\lim_{\epsilon\to 0^+}\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right|^{-1}}\cdot\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\left|\det\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right| \quad \text{// distribute lim over product factors; the first term = 1}\\
&=-\lim_{\epsilon\to 0^+}\operatorname{tr}\left(\operatorname{adj}\left(\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)\frac{\partial}{\partial\epsilon}\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right) \quad \text{// Jacobi's formula: backprop derivative through matrix determinant}\\
&=-\operatorname{tr}\left(\cancel{\lim_{\epsilon\to 0^+}\operatorname{adj}\left(\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right)}\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\frac{\partial \mathbf{h}}{\partial \mathbf{x}}\right) \quad \text{// bring lim inside trace and distribute over matrix product; the first term = } \mathbf{I}\\
&=-\operatorname{tr}\left(\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\frac{\partial \mathbf{x}(t+\epsilon)}{\partial \mathbf{x}}\right) \quad \text{// } \mathbf{h}=\mathbf{x}(t+\epsilon)\\
&=-\operatorname{tr}\left(\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\frac{\partial}{\partial \mathbf{x}}\left(\mathbf{x}(t)+\epsilon\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t)\right)\right) \quad \text{// Taylor expand } \mathbf{z}(t+\epsilon) \text{ at } t \text{ up to 1st derivative}\\
&=-\operatorname{tr}\left(\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\frac{\partial}{\partial \mathbf{x}}\left(\mathbf{x}(t)+\epsilon\cdot\mathbf{f}(\mathbf{x}(t),t)\right)\right) \quad \text{// plug in the ODE}\\
&=-\operatorname{tr}\left(\lim_{\epsilon\to 0^+}\frac{\partial}{\partial\epsilon}\left(\mathbf{I}+\epsilon\frac{\partial \mathbf{f}(\mathbf{x},t)}{\partial \mathbf{x}}\right)\right)\\
&=-\operatorname{tr}\left(\frac{\partial \mathbf{f}(\mathbf{x},t)}{\partial \mathbf{x}}\right)\\
&=-\sum_{i=1}^{n}\frac{\partial}{\partial x_i}f_i(\mathbf{x},t) \quad \text{// trace of Jacobian matrix = divergence of vector field } \mathbf{f}(\mathbf{x},t)
\end{aligned}
$$

We refer to

$$\frac{\partial}{\partial t}\log p(\mathbf{x},t)=-\sum_{i=1}^{n}\frac{\partial}{\partial x_i}f_i(\mathbf{x},t) \tag{75}$$

as the *instantaneous change of variable formula.*

13.2 Computing Log-likelihood

The instantaneous change of variable formula is much more tractable to compute, and as an application we use it to estimate the log-likelihood of samples obtained from simulating PF ODE.

The log-likelihood of the sample $\mathbf{x}_0$ from PF ODE simulation can be computed as follows

$$\log p(\mathbf{x}_0, 0) = \log p(\mathbf{x}_T, T) + \int_0^T \nabla \cdot \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) \mathrm{d}t$$

which is obtained by integrating both sides of Eq. (75) from 0 to T. Since $p(\mathbf{x}_T, T)$ is a tractable distribution, we only need to numerically estimate the integral. The divergence $\nabla \cdot \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ can be approximated by the following

$$\nabla \cdot \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) = \mathbb{E}_{p(\boldsymbol{\epsilon})} \left[\boldsymbol{\epsilon}^\top \frac{\partial \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, t)}{\partial \mathbf{x}} \boldsymbol{\epsilon} \right]$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. The vector-Jacobian product $\boldsymbol{\epsilon}^\top \frac{\partial \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, t)}{\partial \mathbf{x}} = \left[\sum_{i=1}^n \epsilon_i \frac{\partial f_i}{\partial x_1}, \cdots, \sum_{i=1}^n \epsilon_i \frac{\partial f_i}{\partial x_n} \right]$ can be efficiently computed using reverse-mode auto-differentiation: backpropagate $\boldsymbol{\epsilon}$ through $\mathbf{f}$, and the gradient $\boldsymbol{\epsilon}$ will be accumulated at each component x_j of the input, which is $\sum_{i=1}^n \epsilon_i \frac{\partial f_i}{\partial x_j}$ for $j = 1 \cdots n$. We can then average $\boldsymbol{\epsilon}^\top \frac{\partial \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}, t)}{\partial \mathbf{x}} \boldsymbol{\epsilon}$ over multiple samples of $\boldsymbol{\epsilon}$ to obtain an unbiased estimator of $\nabla \cdot \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$.

13.3 Alternative Derivation

Interestingly, the instantaneous change of variable formula can also be derived from the Continuity Equation itself. We reinterprete the partial derivative on the LHS of the Continuity Equation

$$\frac{\partial}{\partial t} p(\mathbf{x}, t) = -\sum_{i=1}^n \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t) p(\mathbf{x}, t) \right]$$

as derivative of a univariate function of t, therefore

$$\begin{aligned}
\frac{\partial}{\partial t} p(\mathbf{x}, t) =& \frac{\mathrm{d}}{\mathrm{d}t} p(\mathbf{x}(t), t) \text{ // interprete partial derivative as derivative of } p(\mathbf{x}(t), t) \text{ w.r.t. } t \\
=& \frac{\partial p(\mathbf{x}, t)}{\partial \mathbf{x}} \frac{\partial \mathbf{x}(t)}{\partial t} + \frac{\partial p(\mathbf{x}(t), t)}{\partial t} \text{ // total derivative} \\
=& \sum_{i=1}^n \frac{\partial p(\mathbf{x}, t)}{\partial x_i} \frac{\partial x_i(t)}{\partial t} - \sum_{i=1}^n p(\mathbf{x}, t) \frac{\partial f_i(\mathbf{x}, t)}{\partial x_i} - \sum_{i=1}^n f_i(\mathbf{x}, t) \frac{\partial p(\mathbf{x}, t)}{\partial x_i} \text{ //product rule} \\
=& \cancel{\sum_{i=1}^n \frac{\partial p(\mathbf{x}, t)}{\partial x_i} f_i(\mathbf{x}, t)} - \sum_{i=1}^n p(\mathbf{x}, t) \frac{\partial f_i(\mathbf{x}, t)}{\partial x_i(t)} - \cancel{\sum_{i=1}^n f_i(\mathbf{x}, t) \frac{\partial p(\mathbf{x}, t)}{\partial x_i}} \\
=& -\sum_{i=1}^n p(\mathbf{x}, t) \frac{\partial}{\partial x_i} f_i(\mathbf{x}, t)
\end{aligned}$$

Thus

$$\frac{\partial}{\partial t} \log p(\mathbf{x}, t) = \frac{1}{p(\mathbf{x}, t)} \frac{\partial}{\partial t} p(\mathbf{x}, t) = -\sum_{i=1}^n \frac{\partial}{\partial x_i} f_i(\mathbf{x}, t)$$

14 Key Takeaways

- SDE expresses the instantaneous change of a variable as the sum of deterministic component (drift) and stochastic component (diffusion). We are free to choose the form of the drift and diffusion coefficient, giving rise to SDEs with different behaviors.

- SDE induces a stochastic process characterized by (a path of) marginal probability density $p(\mathbf{x}, t)$.
- FPE can be derived given SDE. FPE describes the time evolution of $p(\mathbf{x}, t)$.
- The trick in deriving FPE is to use an arbitrary test function $\varphi(\cdot)$.
- FPE with zero diffusion = Continuity Equation, FPE with zero drift = Heat Equation.
- For a forward SDE there exist a PF ODE and reverse-time SDE that induce stochastic process with the same marginal probability density $p(\mathbf{x}, t)$ as the forward SDE.
- We can sample from an arbitrary data distribution by simulating PF ODE or reverse-time SDE backwards in time from a tractable noise distribution (i.e. isotropic Gaussian).
- The score function points in the direction of steepest increase in log-probability of sample, and can be efficiently estimated using a data-driven approach.
- If the drift and diffusion coefficient are affine transformations, the perturbation kernel follows a Normal distribution whose mean and covariance can be computed analytically, thus enabling simulation-free training of the score model.
- SDE and PF ODE can be regarded as special cases of a family of SDEs controlled by a hyperparameter α. The level of stochasticity ranges from $\alpha = 0$ (PF ODE, no stochasticiy) to $\alpha = 1$ (SDE, fully stochastic).
- The idea of Flow Matching is to discard SDE altogether and instead 1) define a procedure that smoothly transform a random variable $\mathbf{x}_t$ between two arbitrary probability distributions, and 2) learn a parametric model to match the entire dynamics that drives the evolution of $\mathbf{x}_t$.
- Rectified Flow models are a special case of Flow Matching in which the trajectory of the transformation happens to be a straight line.
- Consistency models predict the end point of the trajectory of PF ODE simulation without actually running the simulation.
- Log-likelihood of the samples obtained from PF ODE simulation can be computed using instantaneous change of variables formula.