DIFFUSION MODELS NOTE

A PREPRINT

ChoCho
Mathematics
National Central University
Taoyuan City 320317
kycho@math.ncu.edu.tw

July 3, 2024

ABSTRACT

The purpose of this survey is to introduce the diffusion model. We will first introduce the basic concepts of DDPM, and then introduce some developments based on DDPM, including DDIM and the condition diffusion model. We will be writing using symbols customary to the mathematics department.

***K*eywords**

Diffusion Models

1 Introduction

Diffusion Probabilistic Models (DPM, or Diffusion Models) were first proposed by Sohl-Dickstein et al. (2015). We will focus on the DDPM (Denoising Diffusion Probabilistic Models) (Ho, Jain, and Abbeel (2020)). We will also introduce some developments based on DDPM: including DDIM (Denoising Diffusion Implicit Models) (Section 2.4) and the condition diffusion model (Section 2.5).

The history of generative AI is rich and multifaceted, dating back several decades. Initially, generative models were relatively simplistic, but advancements over time have led to the development of more sophisticated techniques. One of the earliest breakthroughs in this field was the introduction of the Variational Autoencoder (VAE) (Kingma and Welling (2022)). VAEs employ a probabilistic approach to model the distribution of data, allowing for the generation of new, similar data points by sampling from this distribution. Following VAEs, Generative Adversarial Networks (GANs) (Goodfellow et al. (2014)) revolutionized generative AI by using a game-theoretic approach, where two neural networks—the generator and the discriminator—compete in a zero-sum game, resulting in the creation of highly realistic data.

Diffusion models are a newer addition to this landscape and have shown remarkable promise. These models work by simulating the diffusion process, wherein data points are progressively transformed from a simple distribution (like Gaussian noise) to a complex data distribution. Notable types of diffusion models include Denoising Diffusion Probabilistic Models (DDPMs) and Noise-Conditional Score Networks (NCSNs). DDPMs iteratively refine noisy data points until they resemble the target distribution, whereas NCSNs use score matching to model the gradient of the data distribution, which guides the generation process.

Recent developments in diffusion models have focused on enhancing their efficiency and quality. Innovations such as improved noise scheduling, hybrid architectures combining features from VAEs and GANs, and advancements in training techniques have all contributed to the rapid evolution of diffusion models. These advancements have enabled diffusion models to generate data with unprecedented fidelity and have opened new avenues for their application across various domains, including image synthesis, natural language processing, and beyond.

In summary, diffusion models have emerged as a powerful tool within the generative AI toolkit. Their ongoing development promises to further push the boundaries of what is possible in data generation, offering exciting possibilities for both research and practical applications.

Next, we introduce the basic concepts of DDPM.

2 Background

The diffusion model consists of two main parts:

1. **Adding Noise (Forward Process):** We gradually introduce independent noise to the starting image until it becomes pure noise.
2. **Denoising (Backward Process):** Beginning with pure noise, we use the current image to estimate what the previous image looked like. Repeating this process step by step, the final output image is our generated picture.

TODO: 補圖

We use mathematical formulas to describe the above statement. Given $T \in \mathbb{N}$. Fix constants $\alpha_t, \beta_t \in (0.001, 0.999)$ for $t = 1, 2, \cdots, T$ such that $\alpha_t + \beta_t = 1$. We set the following random vectors of $\mathbb{R}^n$ (note that here we only have random vectors and not probability measures):

- X_0: The initial image.
- $\mathcal{E}_t$, $t = 1, 2, \cdots, T$: The noise added in step t.
- $X_t = \sqrt{\alpha_t} X_{t-1} + \sqrt{\beta_t} \mathcal{E}_t$, $t = 1, 2, \cdots, T$: The image in step t.

To have the concepts of **independence** and **noise**, we need to have probability measures. In the following text, we use lowercase $q(x)$ to denote the density of a probability measure $\mathbf{Q}$ corresponding to the random variable X. Others (e.g., $q(x_t), p_\theta(x_t)$) are the same (p_θ corresponds to $\mathbf{P}_\theta$). We also use $q(x_{0:t})$ to denote the density of $(X_0, X_1, \cdots, X_t) := X_{0:t}$ for the probability measure $\mathbf{Q}$. Others are the same.

Suppose $q_{\text{want}}(x_0)$ is the density of X_0 we want to pursue. We do not know what $q_{\text{want}}(x_0)$ is. We only have some eligible images (discrete data) with mass function $q(x_0)$. When this discrete data large, $q(x_0) \approx q_{\text{want}}(x_0)$ in some sense of distribution. **Our goal** is to find a density $p(x_0)$ of X_0 such that $p(x_0) \approx q_{\text{want}}(x_0)$ in some sense of distribution.

2.1 Forward process

TODO: Notation

In the forward process, we add noise independently to the image. Note that adding noise independently is equivalent to the Markov property (see Section A.1). We define the **forward process** $(\{X_0, \cdots, X_T\}, \mathbf{Q})$ as a Markov chain with

- the initial density $q(x_0)$, and
- the transition density
$$q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{\alpha_t} x_{t-1}, \beta_t \mathbf{I}).$$

By the Markov property, the joint density of $(X_T, X_{T-1}, \cdots, X_1, X_0)$ for the forward process (or we say under $\mathbf{Q}$) is
$$q(x_{T:0}) = q(x_T|x_{T-1}) \cdot q(x_{T-1}|x_{T-2}) \cdots q(x_1|x_0) \cdot q(x_0).$$

Recall that $X_t = \sqrt{\alpha_t} X_{t-1} + \sqrt{\beta_t} \mathcal{E}_t$. Then under $\mathbf{Q}$, $\underline{\mathcal{E}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}$ and
$$\underline{X_0, \mathcal{E}_1, \mathcal{E}_2, \cdots, \mathcal{E}_t \text{ are independent}}$$
(see TODO: appendix). Define a random vector $\overline{\mathcal{E}}_t$ by
$$X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1 - \overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t, \tag{1}$$
where $\overline{\alpha}_t = \alpha_t \cdot \alpha_{t-1} \cdots \alpha_1$. Then under $\mathbf{Q}$, $\underline{\overline{\mathcal{E}}_t \perp X_0}$ and $\underline{\overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}$. This implies that X_T converges in distribution to $\mathcal{N}(\mathbf{0}, \mathbf{I})$ under $\mathbf{Q}$ for T large.

Equation 1 is a important relation between X_t and X_0 and the noise $\overline{\mathcal{E}}_t$. For example, if we have an estimator of $\overline{\mathcal{E}}_t$, say $\widehat{\overline{\mathcal{E}}}_t$, then by this relationship, we have an estimator $\widehat{X}_0 = \widehat{X}_0\left(X_t, \widehat{\overline{\mathcal{E}}}_t\right)$ of X_0 satisfies the following:
$$X_t = \sqrt{\overline{\alpha}_t} \widehat{X}_0 + \sqrt{1 - \overline{\alpha}_t} \cdot \widehat{\overline{\mathcal{E}}}_t. \tag{2}$$

We will use this relationship when we reparameterize our model.

2.2 Backward process

In the backward process, we remove the noise according to the current image. This can also be described by the Markov chain. Ideally we define the **backward process** $(\{X_T, X_{T-1}, \cdots, X_1, X_0\}, \mathbf{P})$ as a Markov chain with the initial distribution $p(x_T) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the transition density $p(x_{t-1}|x_t) = q(x_{t-1}|x_t)$. In this assumption, we have $p(x_0) \approx q(x_0)$ in some sense of distribution (see TODO: appendix). We may sample $x_0 \sim p(x_0)$ by the following:

- Sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
- Sample $x_{t-1} \sim q(x_{t-1}|x_t)$ inductively for $t = T, T-1, \cdots, 1$.

However, there is a problem with the sampling above. Although from the properties of conditional density, we have

$$q(x_{t-1}|x_t) = \frac{q(x_{t-1})}{q(x_t)} \cdot q(x_t|x_{t-1}).$$

It's not easy to use this formula to sample $x_{t-1} \sim q(x_{t-1}|x_t)$ through code since the expression of $q(x_{t-1})/q(x_t)$ may be complicated. The way to solve this problem is that we assume there is another probability measure $\mathbf{P}_\theta$ (this is our model, which can be sampled through code). There are several methods (SDE or just Taylor's theory, see TODO: appendix) to show that we can approximate $q(x_{t-1}|x_t)$ with a normal. Hence, we construct $\mathbf{P}_\theta$ such that

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)),$$

where μ_θ, Σ_θ is what we need to give. A way to construct $\mathbf{P}_\theta$ is that we consider $(\{X_T, X_{T-1}, \cdots, X_1, X_0\}, \mathbf{P}_\theta)$ is a Markov chain with

- the initial density $p_\theta(x_T) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ and
- the transition density

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)).$$

The joint density of $X_{0:T}$ (under $\mathbf{P}_\theta$) is, by the Markov property,

$$p_\theta(x_{0:T}) = p_\theta(x_0|x_1) \cdot p_\theta(x_1|x_2) \cdots p_\theta(x_{T-1}|x_T) \cdot p(x_T).$$

We can sample $x_0 \sim p_\theta(x_0)$ by the following:

- Sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
- Sample $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$ inductively for $t = T, T-1, \cdots, 1$.

Now **our goal** becomes to optimize θ such that $p_\theta(x_0) \approx q(x_0)$ in some sense. A common way to measure the difference between $p_\theta(x_0)$ and $q(x_0)$ is the KL-divergence

$$D_{\text{KL}}(q(x_0) \| p_\theta(x_0)) = \int_{x_0 \in \mathbb{R}^n} q(x_0) \log \frac{q(x_0)}{p_\theta(x_0)} \mathrm{d}x_0.$$

By the definition of the KL-divergence,

$$\begin{aligned}
\mu_\theta^*, \Sigma_\theta^* &= \arg\min_{\mu_\theta, \Sigma_\theta} D_{\text{KL}}(q(x_0) \| p_\theta(x_0)) \\
&= \arg\min_{\mu_\theta, \Sigma_\theta} \Big(- \int q(x_0) \log \Big(\frac{p_\theta(x_0)}{q(x_0)} \Big) \mathrm{d}x_0 \Big) \\
&= \arg\min_{\mu_\theta, \Sigma_\theta} \underbrace{\Big(- \int q(x_0) \log p_\theta(x_0) \mathrm{d}x_0 \Big)}_{\mathbb{E}_{X_0 \sim q(x_0)}[-\log p_\theta(X_0)]}.
\end{aligned}$$

Through the evidence lower bound(ELBO),

$$\mathbb{E}_{X_0 \sim q(x_0)}[-\log p_\theta(X_0)] \leq \mathbb{E}_{X_{0:T} \sim q(x_{0:T})} \Big[-\log \frac{p_\theta(X_{0:T})}{q(X_{1:T}|X_0)} \Big] := L.$$

Our goal becomes to minimize L. We separate L into three parts (for details, see TODO: appendix):

$$\begin{aligned} L = & \underbrace{\mathbb{E}_{X_0 \sim q(x_0)}\left[D_{\mathrm{KL}}\left(\underline{q(x_T|x_0)} \| \underline{p(x_T)}\right)\Big|_{x_0 = X_0}\right]}_{L_T} \\ & + \sum_{t=2}^{T} \underbrace{\mathbb{E}_{X_0, X_t \sim q(x_0, x_t)}\left[D_{\mathrm{KL}}\left(\underline{q(x_{t-1}|x_t, x_0)} \| \underline{p_\theta(x_{t-1}|x_t)}\right)\Big|_{x_0, x_t = X_0, X_t}\right]}_{L_{t-1}} \\ & + \underbrace{\mathbb{E}_{X_0, X_1 \sim q(x_0, x_1)}\left[-\log p_\theta(x_0|x_1)\Big|_{x_0, x_1 = X_0, X_1}\right]}_{L_0}. \end{aligned} \tag{3}$$

- The first term L_T controls how similar is the last image of the forward process to the pure noise. L_T can be calculated directly since both $q(x_T|x_0)$, $p(x_T)$ are normal. The value is
$$L_T = \frac{1}{2}\left(\log(1 - \overline{\alpha}_T) + n\left(\frac{1}{1 - \overline{\alpha}_T} - 1\right) + \frac{\overline{\alpha}_T}{1 - \overline{\alpha}_T}\mathbb{E}_{X_0 \sim q(x_0)}\left[\|X_0\|^2\right]\right).$$
It is clear that $\lim_{T \to \infty} L_t = 0$. From the above formula, depending only on the L^2-norm of X_0, L_T can be smaller if we shift X_0 by its mean. We may see the case $n = 1$. TODO: 圖 For the question of how to choose the size of T, see TODO: ref.
- The second term L_{t-1}, $t = 2, \cdots, T$, is the most important since it determines how to choose $\mu_\theta, \Sigma_\theta$. This term controls the similarity of X_{t-1} in the forward and backward process. By Bayes' rule and after a long calculation (see TODO: appendix),
$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \mu_t(x_t, x_0), \Sigma_t), \quad t = 2, \cdots, T,$$
where
$$\mu_t(x_t, x_0) = \frac{\sqrt{\alpha_t}(1 - \overline{\alpha}_{t-1})}{1 - \overline{\alpha}_t}x_t + \frac{\sqrt{\overline{\alpha}_{t-1}}\beta_t}{1 - \overline{\alpha}_t}x_0, \quad \Sigma_t = \frac{1 - \overline{\alpha}_{t-1}}{1 - \overline{\alpha}_t}\beta_t\mathbf{I}. \tag{4}$$

2.2.1 To determine Σ_θ for $t \geq 2$

Since both $q(x_{t-1}|x_t, x_0)$, $p_\theta(x_{t-1}|x_t)$ are normal, it is natural to choose

$$\Sigma_\theta(x, t) = \Sigma_t = \frac{1 - \overline{\alpha}_{t-1}}{1 - \overline{\alpha}_t}\beta_t\mathbf{I} := \sigma_t^2\mathbf{I}. \tag{5}$$

2.2.2 To determine μ_θ for $t \geq 2$

By the choice of Σ_θ, we have

$$\begin{aligned} L_{t-1} &= \mathbb{E}_{X_0, X_t \sim q(x_0, x_t)}\left[\frac{1}{2\sigma_t^2}\left\|\mu_t(X_t, X_0) - \mu_\theta(X_t, t)\right\|^2\right] \\ &= \mathbb{E}_{\substack{X_0 \sim q(x_0), \overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ X_0, \overline{\mathcal{E}}_t \text{ are independent} \\ X_t = \sqrt{\overline{\alpha}_t}X_0 + \sqrt{1 - \overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t}}\left[\frac{1}{2\sigma_t^2}\left\|\mu_t(X_t, X_0) - \mu_\theta(X_t, t)\right\|^2\right]. \end{aligned}$$

Then we reparametrize μ_θ by

$$\mu_\theta(X_t, t) = \mu_t(X_t, \widehat{X}_0), \tag{6}$$

where $\widehat{X}_0 = \widehat{X}_0(X_t)$ is the estimate of X_0 via our model by giving X_t (we will give the details of $\widehat{X}_0$ later in Equation 8). With this parametrization and by the expression of μ_t in Equation 4, we have

$$\left\|\mu_t(X_t, X_0) - \mu_\theta(X_t, t)\right\| = \frac{\sqrt{\overline{\alpha}_{t-1}}\beta_t}{1 - \overline{\alpha}_t}\left\|X_0 - \widehat{X}_0(X_t)\right\|. \tag{7}$$

Let $\mathtt{Net}_\theta : \mathbb{R}^n \times \{1, 2, \cdots, T\} \longrightarrow \mathbb{R}^n$ be our neural network (with parameters θ) we need to train. We can choose $\mathtt{Net}_\theta$ to predict X_0, or $\overline{\mathcal{E}}_t$ or the velocity V_t (see Hang et al. (2023)). DDPM chooses to predict the noise $\overline{\mathcal{E}}_t$. Then by Equation 2, we have the following relation

$$X_t = \sqrt{\overline{\alpha}_t} \cdot \widehat{X}_0(X_t) + \sqrt{1 - \overline{\alpha}_t} \cdot \mathtt{Net}_\theta(X_t, t). \tag{8}$$

Note that $\widehat{X}_0 = \widehat{X}_0(X_t) = \widehat{X}_0(X_t, \theta)$. Hence, we have

$$L_{t-1} = \mathbb{E}_{\substack{X_0 \sim q(x_0), \overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ X_0, \overline{\mathcal{E}}_t \text{ are independent} \\ X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1-\overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t}} \left(\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\overline{\alpha}_t)} \left\| \overline{\mathcal{E}}_t - \mathtt{Net}_\theta(X_t, t) \right\|^2 \right).$$

- For the third term L_0. Recall that we assume $p_\theta(x_0|x_1) = \mathcal{N}(x_0; \mu_\theta(x_1, 1), \Sigma_\theta(x_1, 1))$. For convience (see Equation 5), we choose $\Sigma_\theta(x_1, 1)$ to be a constant matrix indepdent of θ and x_1, e.g.,

  $$\Sigma_\theta(x_1, 1) = \beta_1 \mathbf{I} := \sigma_1^2 \mathbf{I}.$$

  Note that

  $$-\log p_\theta(x_0|x_1) = \frac{1}{2\beta_1} \left\| x_0 - \mu_\theta(x_1, 1) \right\|^2 + \mathtt{const},$$

  where `const` is some constant independent of (x_0, x_1, θ). Here we also reparametrize μ_θ by Equation 6 for $t = 1$ with $\overline{\alpha}_0 := 1$. In this setting,

  $$\mu_\theta(X_1, 1) = \mu_1(X_1, \widehat{X}_0(X_1)) = \widehat{X}_0(X_1).$$

  To maximize

  $$L_0 = \mathbb{E}_{X_0, X_1 \sim q(x_0, x_1)} \left[-\log p_\theta(x_0|x_1) \Big|_{x_0, x_1 = X_0, X_1} \right]$$

  is equivalent to maximize

  $$L_0' = \mathbb{E}_{X_0, X_1 \sim q(x_0, x_1)} \left[\frac{1}{2\beta_1} \left\| X_0 - \widehat{X}_0(X_1) \right\|^2 \right].$$

  Hence, if we use the same assumption from Equation 8, our goal is to minimize

  $$\begin{aligned} L_0' &= \mathbb{E}_{X_0, X_1 \sim q(x_0, x_1)} \left[\frac{1-\alpha_1}{2\beta_1 \alpha_1} \left\| X_0 - \widehat{X}_0(X_1) \right\|^2 \right] \\ &= \mathbb{E}_{\substack{X_0 \sim q(x_0), \overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ X_0, \overline{\mathcal{E}}_t \text{ are independent} \\ X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1-\overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t}} \left(\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\overline{\alpha}_t)} \left\| \overline{\mathcal{E}}_t - \mathtt{Net}_\theta(X_t, t) \right\|^2 \right) \end{aligned}$$

  with $t = 1$.

2.3 Training and Sampling

Note that we will minimize $\mathbb{E}_{X \sim q(x)}[f_\theta(X)]$ by **repeating** the following:

- Sampling $x \sim q(x)$ and then
- minimizing $f_\theta(x)$ by taking gradient descent on θ.

Recall that for $t = 2, 3, \cdots, T$,

$$L_{t-1} = \mathbb{E}_{\substack{X_0 \sim q(x_0), \overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ X_0, \overline{\mathcal{E}}_t \text{ are independent} \\ X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1-\overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t}} \left(\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\overline{\alpha}_t)} \left\| \overline{\mathcal{E}}_t - \mathtt{Net}_\theta(X_t, t) \right\|^2 \right).$$

DDPM chooses a simple version that minimizes L_{t-1}^{simple}, ignoring the weights in the expectation:

$$L_{t-1}^{\text{simple}} = \mathbb{E}_{\substack{X_0 \sim q(x_0), \overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ X_0, \overline{\mathcal{E}}_t \text{ are independent} \\ X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1-\overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t}} \left(\left\| \overline{\mathcal{E}}_t - \mathtt{Net}_\theta(X_t, t) \right\|^2 \right).$$

L_0 is the same. Therefore, our training algorithm is as follows:

Algorithm 1 Training (DDPM)

```
repeat
    $t \sim \text{Uniform}(\{1, \cdots, T\})$                          ▷ Sample random step
    $x_0 \sim q(x_0)$                                                 ▷ Sample random initial image
    $\bar{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$    ▷ Sample random noise
    $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \bar{\varepsilon}_t$
    Take gradient descent step on $\left\| \bar{\varepsilon}_t - \mathtt{Net}_\theta(x_t, t) \right\|^2$    ▷ Optimization
until converged
```

For the sampling, we may sample $x_0 \sim p_\theta(x_0)$ by the following:

- Sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
- Sample $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$ inductively for $t = T, T-1, \cdots, 1$.

Recall that $p_\theta(x_{t-1}|x_t) \sim \mathcal{N}(\mu_\theta(x_t, t), \sigma_t \mathbf{I})$, where

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}} \Big(x_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \mathtt{Net}_\theta(x_t, t) \Big).$$

Therefore, our sampling algorithm is as follows:

Algorithm 2 Sampling (DDPM)

```
$x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \cdots, 1$ do
    if $t > 1$ then
        $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    else
        $z = \mathbf{0}$
    end if
    $x_{t-1} = \frac{1}{\sqrt{\alpha_t}} \Big( x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \mathtt{Net}_\theta(x_t, t) \Big) + \sigma_t z$
end for
return $x_0$
```

2.4 DDIM

One of the major drawbacks of DDPM is the lengthy time required for data generation, especially when compared to other generative AI methods. In response to this issue, an improved version of DDPM, known as Denoising Diffusion Implicit Models (DDIM), was introduced by Song et al. (Song, Meng, and Ermon (2022)). The primary innovation of DDIM is its ability to significantly accelerate the data generation process. By refining the underlying diffusion mechanism, DDIM reduces the number of required diffusion steps without sacrificing the quality of the generated data. This breakthrough makes DDIM a more practical and efficient alternative for generative AI tasks, offering faster performance while maintaining high-quality outputs.

Now we introduce the DDIM. The main reason why we can decompose L in Equation 3 in DDPM is that we have the following production of two densities:

$$\begin{aligned} p_\theta(x_{0:T}) &= p_\theta(x_T) \cdot \prod_{t=2}^{T} p_\theta(x_{t-1}|x_t) \cdot p_\theta(x_0|x_1), \\ q(x_{1:T}|x_0) &= q(x_T|x_0) \cdot \prod_{t=2}^{T} q(x_{t-1}|x_t, x_0). \end{aligned} \tag{9}$$

DDIM consider a new forward process $(\{X_0, X_1, \cdots, X_T\}, \mathbf{Q}_\sigma)$, where $\mathbf{Q}_\sigma$ is some probability measure indexed by $\sigma \in [0, \infty)^T$. The forward process is not a Markov chain but has the same conditional density of X_t given $X_0 = x_0$ for each t as DDPM. Inspired by Equation 9, DDIM directly defines the joint density

$$q_\sigma(x_{0:T}) := q_\sigma(x_T|x_0) \cdot \prod_{t=2}^{T} q_\sigma(x_{t-1}|x_t, x_0) \cdot q(x_0),$$

where $q_\sigma(x_T|x_0) := \mathcal{N}(\sqrt{\overline{\alpha}_T}x_0, (1-\overline{\alpha}_T)\mathbf{I})$ and

$$q_\sigma(x_{t-1}|x_t, x_0) := \mathcal{N}\left(\sqrt{\overline{\alpha}_{t-1}}x_0 + \sqrt{1-\overline{\alpha}_{t-1}-\sigma_t^2}\cdot\frac{x_t-\sqrt{\overline{\alpha}_t}x_0}{\sqrt{1-\overline{\alpha}_t}}, \sigma_t^2\mathbf{I}\right), \quad t = 2, \cdots, T.$$

Note that $q_\sigma(x_{0:T})$ is a density since it is a product of densities. This seems a little weird that the joint density of $q_\sigma(x_{0:T})$ is determined by some conditional density. In fact, $(\{X_0, X_1, \cdots, X_T\}, \mathbf{Q}_\sigma)$ is a process satisfying the following conditions:

1. Under $\mathbf{Q}_\sigma$, X_0 has the density $q(x_0)$.
2. Conditioned on $X_0 = x_0$, the process $\left(\{X_T, X_{T-1}, \cdots, X_2, X_1\}\Big|_{X_0=x_0}, \mathbf{Q}_\sigma\right)$ is a Markov chain with
   - the initial density $q_\sigma(x_T|x_0) = \mathcal{N}(\sqrt{\overline{\alpha}_T}x_0, (1-\overline{\alpha}_T)\mathbf{I})$ and
   - the transition density

   $$q_\sigma(x_{t-1}|x_t, x_0) = \mathcal{N}\left(\sqrt{\overline{\alpha}_{t-1}}x_0 + \sqrt{1-\overline{\alpha}_{t-1}-\sigma_t^2}\cdot\frac{x_t-\sqrt{\overline{\alpha}_t}x_0}{\sqrt{1-\overline{\alpha}_t}}, \sigma_t^2\mathbf{I}\right), \quad t = 2, \cdots, T.$$

   Note that if we write $q_\sigma(x_{t-1}|x_t, x_0) = \mathcal{N}(f(x_t, x_0, t), \sigma_t^2\mathbf{I})$, then the process $\left(\{X_T, X_{T-1}, \cdots, X_2, X_1\}\Big|_{X_0=x_0}, \mathbf{Q}_\sigma\right)$ can be write as, conditioned on $X_0 = x_0$,

   $$X_{t-1} = f(X_t, x_0, t) + \sigma_t\xi_t, \quad t = T, \cdots, 2,$$

   where $X_T, \xi_{T-1}, \xi_{T-2}, \cdots, \xi_1$ are independent under $\mathbf{Q}_\sigma$.

For each $\sigma \in [0, \infty)^T$, we can show that for this joint density $q_\sigma(x_{0:T})$,

$$\begin{aligned} q_\sigma(x_0) &= q(x_0), \\ q_\sigma(x_t|x_0) &= \mathcal{N}(\sqrt{\overline{\alpha}_t}x_0, (1-\overline{\alpha}_t)\mathbf{I}) = q(x_t|x_0), \quad t = 1, \cdots, T. \end{aligned}$$

DDIM consider the backward process $(\{X_T, X_{T-1}, \cdots, X_1, X_0\}, \mathbf{P}_\theta)$ as a Markov chain with the initial distribution $p_\theta(x_T) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ and the transition density

$$\begin{aligned} p_\theta(x_0|x_1) &= \mathcal{N}(\hat{x}_0(x_1, 1), \sigma_1^2\mathbf{I}), \\ p_\theta(x_{t-1}|x_t) &= q_\sigma(x_{t-1}|x_t, \hat{x}_0) \\ &= \mathcal{N}\left(\sqrt{\overline{\alpha}_{t-1}}\hat{x}_0 + \sqrt{1-\overline{\alpha}_{t-1}-\sigma_t^2}\cdot\frac{x_t-\sqrt{\overline{\alpha}_t}\hat{x}_0}{\sqrt{1-\overline{\alpha}_t}}, \sigma_t^2\mathbf{I}\right), \quad t = 2, \cdots, T, \end{aligned}$$

where $\hat{x}_0 = \hat{x}_0(x_t, t)$ satisfies

$$x_t = \sqrt{\overline{\alpha}_t}\cdot\hat{x}_0(x_t, t) + \sqrt{1-\overline{\alpha}_t}\cdot\texttt{Net}_\theta(x_t, t), \quad x \in \mathbb{R}^n, t \in \mathbb{N}.$$

By the constructions of q_σ, p_θ, we still have the decomposition

$$\begin{aligned} &\mathbb{E}_{X_{0:T}\sim q_\sigma(x_{0:T})}\left[-\log\frac{p_\theta(X_{0:T})}{q_\sigma(X_{1:T}|X_0)}\right] \\ &= \underbrace{\mathbb{E}_{X_0\sim q_\sigma(x_0)}\left[D_{\mathtt{KL}}\left(\underline{q_\sigma(x_T|x_0)}\|\underline{p(x_T)}\right)\Big|_{x_0=X_0}\right]}_{L_T} \\ &\quad + \sum_{t=2}^{T}\underbrace{\mathbb{E}_{X_0,X_t\sim q_\sigma(x_0,x_t)}\left[D_{\mathtt{KL}}\left(\underline{q_\sigma(x_{t-1}|x_t, x_0)}\|\underline{p_\theta(x_{t-1}|x_t)}\right)\Big|_{x_0,x_t=X_0,X_t}\right]}_{L_{t-1}} \\ &\quad + \underbrace{\mathbb{E}_{X_0,X_1\sim q_\sigma(x_0,x_1)}\left[-\log p_\theta(x_0|x_1)\Big|_{x_0,x_1=X_0,X_1}\right]}_{L_0}. \end{aligned}$$

There are two special values for σ.

- The first one is
$$\sigma_t = \sqrt{(1-\overline{\alpha}_{t-1})/(1-\overline{\alpha}_t)}\sqrt{1-\alpha_t}, \quad t = 1, \cdots, T.$$
Under this σ, the process $\left(\{X_0, X_1, \cdots, X_T\}, \mathbf{Q}_\sigma\right)$ becomes a Markov chain, hence the DDIM becomes the original DDPM.
- The second one is $\sigma_t = 0$ for $t = 1, 2, \cdots, T$. In this case, the backward process $\left(\{X_T, X_{T-1}, \cdots, X_0\}, \mathbf{P}_\theta\right)$ becomes deterministic when we condition on $X_T = x_T$. This greatly speeds up the sampling of diffusion models. In this case, we may write
$$X_{t-1} = \sqrt{\overline{\alpha}_{t-1}}\widehat{X}_0(X_t, t) + \sqrt{1-\overline{\alpha}_{t-1}} \cdot \mathtt{Net}_\theta(X_t, t), \quad t = T, T-1, \cdots, 1.$$

2.5 Conditional Diffusion Model

Dhariwal and Nichol (2021)

- 一般沒有限定條件的 diffusion model，我們無法去控制想生成的東西。這明顯無法滿足我們的需求。比如說在 mnist 之中，我們想要去控制生成 0~9 的是哪個數字。又比如說 celebA 這資料集中，我們想要去生成的大頭像有什麼特徵（比如說是男是女，有無戴眼鏡）。所以自然而然會有所謂的 Conditional diffusion model。
- 我們先從簡單類別的說起，用 mnist 的數字來解釋。我們現在有資料集 $X \times Y$ 的分佈
$$\widehat{q}(x_0, y), \quad x_0 \in \mathbb{R}^{w \times h}, \quad y \in \mathbb{R}^n,$$
where
  - X_0 是數字圖片;
  - Y 是數字 label 在 $\mathbb{R}^n$ 的 embed
    - That is, $\mathbb{R}^n$ is the embed space of labels.
    - For this example, $0, 1, \cdots, 9$ are `nn.Embedding(10,n)(torch.arange(10))`.（所以這裡 embed 也是要可學習的）.
- Given the label $Y = y$. We want to generate an image x_0 which has the label y.
- Assume that we already have $\widehat{q}(y|x_0)$. That is, when we have x_0, we know the distribution of labels of x_0.
- 如果忽略掉 Y, 只看 X_0, 可視為之前的 unconditional diffusion model
- We define q as before:
  - $q(x_0)$: the distribution of X_0 (無表達式);
  - $q(x_t|x_{t-1}) = \mathcal{N}(\sqrt{\alpha_t}x_{t-1}, (1-\alpha_t)\mathbf{I})$.

2.5.0.1 Important

- 同樣地我們令 $\{X_t\}_{t=0}^T$ 為時間 t 時的加噪圖片, 只是加噪方式是如下:
Define the forward process of $(X_{0:T}, Y)$ by the following:
  - $\widehat{q}(x_0) := q(x_0)$ (無表達式) (eq 28).
    - So that we have $\widehat{q}(x_0, y) = \underbrace{q(x_0)}_{\text{無表達式}} \cdot \underbrace{\widehat{q}(y|x_0)}_{\text{有表達式}}$.
  - $\widehat{q}(x_t|x_{t-1}, y) := q(x_t|x_{t-1})$ (有表達式) (eq 30);
  - $\widehat{q}(x_{1:T}|x_0, y) := \prod_{t=1}^T \widehat{q}(x_t|x_{t-1}, y)$ (eq 31).
    - Conditioned on $Y = y$, the forward process $X_0, X_1, \cdots, X_T$ is a Markov chain with the transition density $q(x_t|x_{t-1})$.

  Note that
$$\begin{aligned}\widehat{q}(x_{0:T}, y) &= \widehat{q}(x_0, y) \cdot \widehat{q}(x_{1:T}|x_0, y), \\ &= \widehat{q}(x_0, y) \cdot \prod_{t=1}^T \widehat{q}(x_t|x_{t-1}, y).\end{aligned}$$
- For this $\widehat{q}$, we have
  - $\widehat{q}(x_t|x_{t-1}) = \widehat{q}(x_t|x_{t-1}, y)$ (eq 32~37) $= q(x_t|x_{t-1})$ (eq 30);
  - $\widehat{q}(x_{1:T}|x_0) = q(x_{1:T}|x_0)$ (eq 38~44);

  - $\hat{q}(x_t) = q(x_t)$ (eq 45~50);
  - $\hat{q}(x_{t-1}|x_t) = q(x_{t-1}|x_t)$;
  - (上面四點說明 $\hat{q}$ 在不考慮 label 時, 跟之前的 diffusion model q 分佈完全一樣);
  - $\hat{q}(y|x_{t-1}, x_t) = \hat{q}(y|x_{t-1})$ (eq 51~54);
  - $\hat{q}(x_{t-1}|x_t, y) = \underbrace{q(x_{t-1}|x_t)}_{\approx p_\theta(x_{t-1}|x_t)} \cdot \underbrace{\hat{q}(y|x_{t-1})}_{\approx p_\phi(y|x_{t-1})} \Big/ \underbrace{\hat{q}(y|x_t)}_{\text{constant}}$ (eq 55~61).
    - Note that $p_\phi(y|x_t)$ 是 $p_\phi(y|x_t, t)$ 的縮寫.
    - Note that $p_\theta(x_{t-1}|x_t), p_\phi(y|x_{t-1})$ is our model.
      - 這裡可以使用已經訓練好的 p_θ (純粹 DDPM 的) 和分類器.
  - Define $p_{\theta,\phi}(x_{t-1}|x_t, y) = \text{constant} \cdot p_\theta(x_{t-1}|x_t) \cdot p_\phi(y|x_{t-1})$. So when given the label y, we sample x_0 (with label y) by the following:
    - **For** $t = T, T-1, \cdots, 1$,
      - Sample $x_t \sim p_{\theta,\phi}(x_{t-1}|x_t, y)$
    - **EndFor**

    We organize the formula $p_{\theta,\phi}(x_{t-1}|x_t, y)$. Consider x_t, y as two given constants. Using a Taylor expansion at $x_{t-1} = \mu$ (some constant), we have

$$\begin{aligned}\log p_\phi(y|x_{t-1}) &\approx \log p_\phi(y|x_{t-1})\Big|_{x_{t-1}=\mu} + (x_{t-1} - \mu)\nabla_{x_{t-1}} \log p_\phi(y|x_{t-1})\Big|_{x_{t-1}=\mu} \\ &= (x_{t-1} - \mu) \cdot \end{aligned}$$

2.5.0.2 Sampling (DDPM with classifier)

- **Given:** 訓練好的 $p_\theta(x_{t-1}|x_t)$ (DDPM) 和分類器 $p_\phi(y|x_{t-1})$.
- **Input:** A label y and a gradient scale $s \in (1, \infty)$
- Sample $x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
- **For** $t = T, T-1, \cdots, 1$
  - $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  - Sample $x_{t-1} \sim \mathcal{N}(\mu, \Sigma)$
    - **Comment** Sample from unconditional diffusion model
  - $x_{t-1} \leftarrow x_{t-1} + s\Sigma\nabla_{x_t} \log p_\phi(y|x_t)$
    - **Comment** 有點像是對 $p_{\theta,\phi}(x_{t-1}|x_t, y)$ 做 gradient ascent, 增加 y 的 log-likelihood. 引導 x_{t-1} 向 label y 的方向前進.
- **EndFor**
- **Return** x_0

2.6 Predict Velocity

We have two predictions in the following.

- The first is to predict the initial image X_0 by giving X_t. We set

$$\begin{aligned}\mu_\theta(x_t, t) &= \mu_t\Big(x_t, \mathtt{Net}_\theta(x_t, t)\Big) \\ &= \frac{\sqrt{\alpha_t}(1 - \overline{\alpha}_{t-1})}{1 - \overline{\alpha}_t} x_t + \frac{\sqrt{\overline{\alpha}_{t-1}}\beta_t}{1 - \overline{\alpha}_t} \cdot \mathtt{Net}_\theta(x_t, t), \quad x_t \in \mathbb{R}^n,\ t = 2, \cdots, T,\end{aligned}$$

  and then

$$\Big\|\mu_t(x_t, x_0) - \mu_\theta(x_t, t)\Big\| = \frac{\sqrt{\overline{\alpha}_{t-1}}\beta_t}{1 - \overline{\alpha}_t}\Big\|x_0 - \mathtt{Net}_\theta(x_t, t)\Big\|.$$

- The second is to predict the noise $\overline{\varepsilon}_t$ by giving x_t, t. We set

$$\begin{aligned}\mu_\theta(x_t, t) &= \tilde{\mu}_t\Big(x_t, \mathtt{Net}_\theta(x_t, t)\Big) \\ &= \frac{1}{\sqrt{\alpha_t}}\Big(x_t - \frac{\beta_t}{\sqrt{1 - \overline{\alpha}_t}} \cdot \mathtt{Net}_\theta(x_t, t)\Big), \quad x_t \in \mathbb{R}^n,\ t = 2, \cdots, T,\end{aligned}$$

and then

$$\left\|\mu_t(x_t, x_0) - \mu_\theta(x_t, t)\right\| = \frac{\beta_t}{\sqrt{\alpha_t} \cdot \sqrt{1-\overline{\alpha}_t}}\left\|\overline{\varepsilon}_t - \texttt{Net}_\theta(x_t, t)\right\|.$$

In the backward process, we predict the noise $\overline{\mathcal{E}}_t$ or the initial image X_0. There is another prediction (prediction for the velocity, see TODO: pred_v). For simplicity, we set

$$a_t := \sqrt{\overline{\alpha}_t}, \quad b_t := \sqrt{1-\overline{\alpha}_t}.$$

Then we may rewrite

$$X_t = a_t X_0 + b_t \overline{\mathcal{E}}_t, \quad a_t^2 + b_t^2 = 1.$$

Define the velocity, a random vector we want to predict,

$$V_t := -b_t X_0 + a_t \overline{\mathcal{E}}_t.$$

Then we have the following relations:

$$\begin{aligned} X_0 &= a_t X_t - b_t V_t, \\ \overline{\mathcal{E}}_t &= b_t X_t + a_t V_t. \end{aligned}$$

Then our algorithms become

1. Training
   - $x_0 \sim q(x_0)$
   - $\overline{\varepsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
   - $x_t = a_t x_0 + b_t \overline{\varepsilon}_t$
   - $v_t = -b_t x_0 + a_t \overline{\varepsilon}_t$
   - Loss is $\left\|\texttt{Net}_\theta(x_t, t) - v_t\right\|^2$
2. Sampling
   - $\hat{v} = \texttt{Net}_\theta(x_t, t)$
   - $\hat{\varepsilon} = b_t x_t + a_t \hat{v}, \quad \hat{x}_0 = a_t x_t - b_t \hat{v}$
   - $\hat{\mu} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{b_t}\right)\hat{\varepsilon}$

3 Experiments

4 Conclusion

References

Anderson, Brian D. O. 1982. "Reverse-Time Diffusion Equation Models." *Stochastic Processes and Their Applications* 12 (3): 313–26. https://doi.org/10.1016/0304-4149(82)90051-5.

Dhariwal, Prafulla, and Alex Nichol. 2021. "Diffusion Models Beat GANs on Image Synthesis." https://arxiv.org/abs/2105.05233.

Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. "Generative Adversarial Networks." https://arxiv.org/abs/1406.2661.

Hang, Tiankai, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. 2023. "Efficient Diffusion Training via Min-SNR Weighting Strategy." In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 7441–51.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. 2020. "Denoising Diffusion Probabilistic Models." *Advances in Neural Information Processing Systems* 33: 6840–51.

Kingma, Diederik P, and Max Welling. 2022. "Auto-Encoding Variational Bayes." https://arxiv.org/abs/1312.6114.

Sohl-Dickstein, Jascha, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. "Deep Unsupervised Learning Using Nonequilibrium Thermodynamics." In *Proceedings of the 32nd International Conference on Machine Learning*, edited by Francis Bach and David Blei, 37:2256–65. Proceedings of Machine Learning Research. Lille, France: PMLR. https://proceedings.mlr.press/v37/sohl-dickstein15.html.

Song, Jiaming, Chenlin Meng, and Stefano Ermon. 2022. "Denoising Diffusion Implicit Models." https://arxiv.org/abs/2010.02502.

A Appendix

A.1 Markov property is equivalent to adding noise independently

Given the probability measure $\mathbf{Q}$ such that

- $q(x_0)$ is the mass (respect to $\mathbf{Q}$) of our image data, and
- $X_0, \mathcal{E}_1, \cdots, \mathcal{E}_T$ are independent under $\mathbf{Q}$, and
- $\mathcal{E}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ under $\mathbf{Q}$ for $t = 1, \cdots, T$.

Under the assumptions above, we have the following properties under $\mathbf{Q}$:

- Under $\mathbf{Q}$, if we set
$$X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1 - \overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t,$$
then X_0, $\overline{\mathcal{E}}_t$ are independent, and $\overline{\mathcal{E}}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Note that this property says that $q(x_T) \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$ as T large.
- Under $\mathbf{Q}$, $\{X_0, X_1, \cdots, X_T\}$ is a Markov chain with the transition density
$$q(x_t | x_{t-1}) = \mathcal{N}(\sqrt{\alpha_t} x_{t-1}, \beta_t \mathbf{I}).$$

Remark. Note that the Markov property is equivalent to adding noise independently. That is, if $(\{X_t\}_{t=0}^T, \mathbf{Q})$ is a Markov chain with the transition density
$$q(x_t | x_{t-1}) = \mathcal{N}(\sqrt{\alpha_t} x_{t-1}, \beta_t \mathbf{I}).$$
and we set
$$X_t = \sqrt{\overline{\alpha}_t} X_0 + \sqrt{1 - \overline{\alpha}_t} \cdot \overline{\mathcal{E}}_t,$$
then

- $X_0, \mathcal{E}_1, \cdots, \mathcal{E}_T$ are independent under $\mathbf{Q}$, and
- $\mathcal{E}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ under $\mathbf{Q}$ for $t = 1, \cdots, T$.

A.2 $q(x_0) \approx p(x_0)$

Note that
$$\begin{aligned} q(x_{0:3}) &= q(x_3|x_2) \cdot q(x_2|x_1) \cdot q(x_1|x_0) \cdot q(x_0) \\ &= \frac{q(x_3)}{q(x_2)} q(x_2|x_3) \cdot \frac{q(x_2)}{q(x_1)} q(x_1|x_2) \cdot \frac{q(x_1)}{q(x_0)} q(x_0|x_1) \cdot q(x_0) \\ &= q(x_0|x_1) \cdot q(x_1|x_2) \cdot q(x_2|x_3) \cdot \underbrace{q(x_3)}_{\approx \mathcal{N}(\mathbf{0}, \mathbf{I})} \end{aligned}$$
and
$$p(x_{0:3}) = q(x_0|x_1) \cdot q(x_1|x_2) \cdot q(x_2|x_3) \cdot \underbrace{p(x_3)}_{\mathcal{N}(\mathbf{0}, \mathbf{I})}.$$
Then
$$q(x_0) = \int_{x_{1:3}} q(x_{0:3}) \, \mathrm{d}x_{0:3} \approx \int_{x_{1:3}} p(x_{1:3}) \, \mathrm{d}x_{0:3} = p(x_0).$$

A.3 SDE

If $(X_t)_{t \in [0,1]}$ satisfies the SDE
$$\mathrm{d}X_t = \mu(X_t, t)\mathrm{d}t + \sigma(X_t, t)\mathrm{d}B_t,$$
where $\mu(\cdot, t) : \mathbb{R}^n \longrightarrow \mathbb{R}^n$ and $\sigma(\cdot, t) : \mathbb{R}^n \longrightarrow \mathbb{R}^{n \times n}$ and $(B_t)_{t \in [0,1]}$ is a standard n-dimensional Brownian Motion. Let $q(\cdot, t)$ be the density of X_t for each $t \in [0, 1]$. We have the following results:

- For $t \in [0, 1]$, we define
$$\begin{aligned} \overline{X}_t &:= X_{1-t}, & \overline{q}(\cdot, t) &:= q(\cdot, 1-t), \\ \overline{\mu}(\cdot, t) &:= \mu(\cdot, 1-t), & \overline{\sigma}(\cdot, t) &:= \sigma(\cdot, 1-t). \end{aligned}$$

Then by Anderson (1982), the **reverse process** $(\overline{X}_t)_{t\in[0,1]}$ satisfies

$$\begin{aligned} \mathrm{d}\overline{X}_t = &\Big(\underbrace{-\overline{\mu}(X_t,t) + \overline{\sigma}(\overline{X}_t,t)\overline{\sigma}(\overline{X}_t,t)^{\mathtt{T}}\nabla_x \log \overline{q}(\overline{X}_t,t) + \nabla_x\overline{\sigma}(\overline{X}_t,t)\overline{\sigma}(\overline{X}_t,t)^{\mathtt{T}}}_{\text{drift coefficient}}\Big)\mathrm{d}t \\ &+ \underbrace{\overline{\sigma}(\overline{X}_t,t)}_{\text{diffusion coefficient}} \mathrm{d}\overline{B}_t, \end{aligned} \tag{10}$$

where $(\overline{B}_t)_{t\in[0,1]}$ is a standard n-dimensional Brownian Motion.

- Note that the diffusion coefficient of the reverse process $(\overline{X}_t)_{t\in[0,1]}$ has the same form as $(X_t)_{t\in[0,1]}$. This explains why it is reasonable to assume that $\Sigma_\theta(x,t)$ is independent of x in Equation 5.
- If $\sigma(\cdot,t)=\sigma(t)$ is independent of x, then $\nabla_x\overline{\sigma}(\overline{X}_t,t)\overline{\sigma}(\overline{X}_t,t)^{\mathtt{T}}=0$ and the drift coefficient of Equation 10 is the original average $-\overline{\mu}(X_t,t)$ guided by the score function $\nabla_x \log \overline{q}(\overline{X}_t,t)$.

- Consider a process $(\widetilde{X}_t)_{t\in[0,1]}$ satisfies the ODE

$$\mathrm{d}\widetilde{X}_t = \Big(\mu(\widetilde{X}_t,t) - \frac{1}{2}\sigma(\widetilde{X}_t,t)\sigma(\widetilde{X}_t,t)^{\mathtt{T}}\nabla_x \log q(\widetilde{X}_t,t) - \frac{1}{2}\nabla_x\sigma(\widetilde{X}_t,t)\sigma(\widetilde{X}_t,t)^{\mathtt{T}}\Big)\mathrm{d}t.$$

Then for each $t\in[0,1]$, X_t and $\widetilde{X}_t$ have the same distribution.

A.4 Seperate L

$$L := \mathbb{E}_{X_{0:T}\sim q(x_{0:T})}\Big[-\log\frac{p_\theta(X_{0:T})}{q(X_{1:T}|X_0)}\Big].$$

$$p_\theta(x_{0:T}) = p_\theta(x_T)\cdot\prod_{t=2}^{T} p_\theta(x_{t-1}|x_t)\cdot p_\theta(x_0|x_1),$$

$$\begin{aligned} q(x_{1:T}|x_0) &= \prod_{t=2}^{T} q(x_t|x_{t-1})\cdot q(x_1|x_0) \\ &= \prod_{t=2}^{T} q(x_t|x_{t-1},x_0)\cdot q(x_1|x_0) \\ &= \prod_{t=2}^{T} \frac{q(x_{t-1}|x_t,x_0)q(x_t|x_0)}{q(x_{t-1}|x_0)}\cdot q(x_1|x_0) \\ &= q(x_T|x_0)\cdot\prod_{t=2}^{T} q(x_{t-1}|x_t,x_0) \end{aligned}$$