Diffusion Model

Luo Zhenlin

October 10, 2024

Contents

1 Denoising Diffusion Probabilistic Model

Denoising Diffusion Probabilistic Models.

1.1 Overview

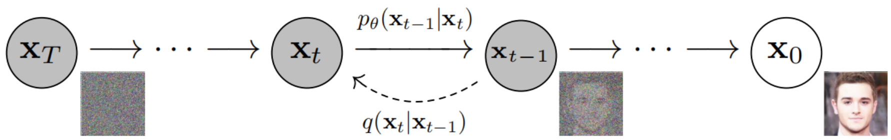

Figure 1: DDPM Framework

Given the input image $\mathbf{x}_0$, in the diffusion model, we want to add random noise continually (as $q()$ shown in Figure 1). After T steps, we obtain $\mathbf{x}_T$ and it follows standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. In this case, we could sample from $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Further, if we could learn the relationship between $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ and implement denoising, then the denoising process can be applied to remove noise from $\mathbf{x}_T$ and finally generate a new sample $\mathbf{x}'_0$, which is a new image. And the two processes are called forward (diffusion) process and reverse process respectively.

More specifically, we need derive the distribution $p(\mathbf{x}_{t-1}|\mathbf{x}_t)$ of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$, not the $\mathbf{x}_{t-1}$, which ensures the diversity. And as a result, derive the distribution $p_\theta(\mathbf{x}_0)$.

Consider the joint distribution of $\mathbf{x}_{0:T}$, we have

$$p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T} \tag{1}$$

where $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ and $\mathbf{x}_1, ..., \mathbf{x}_T$ are latents with the same size of $\mathbf{x}_0$.

If we see the process as a Markov chain, then

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \tag{2}$$

Therefore, our target is to estimate reverse process $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$.

By the Bayesian formula, we have

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \frac{p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1})}{p(\mathbf{x}_t)} \tag{3}$$

1.2 Forward Process

In the right hand of formula (3), the term $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is the adding noise process defined by ourselves. In the later, we use $q()$ to present the process.

The definition of forward (diffusion) process is

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{4}$$

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}) \tag{5}$$

Using the reparameterization skill, we obtain $\mathbf{x}_{t-1}$ by

$$\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon \tag{6}$$

where $\alpha_t = 1 - \beta_t$ and $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This method is equivalent to sample from the distribution (5). And we can control it by setting β_i.

In addition, $p(\mathbf{x}_t)$ and $p(\mathbf{x}_{t-1})$ is hard to derive, but according to our forward process, we can derive $p(\mathbf{x}_t|\mathbf{x}_0)$ and $p(\mathbf{x}_{t-1}|\mathbf{x}_0)$. Hence, it implies us to change (3) given on $\mathbf{x}_0$, that is

$$\begin{aligned} q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|, \mathbf{x}_0)} && (7) \\ &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} && (8) \end{aligned}$$

where Markov chain property is utilized.

Moreover, there is a notable property is that it admits sampling $\mathbf{x}_t$ at an arbitrary time step t in closed form: $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$,

$$\begin{aligned} q(\mathbf{x}_t|\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1 - \bar{\alpha}_t)\mathbf{I}) && (9) \\ \mathbf{x}_t &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{(1 - \bar{\alpha}_t)}\epsilon && (10) \end{aligned}$$

1.3 Reverse Process

Now, the right hand of formula (8) is defined. Thus, we could obtain the distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ is

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}) \tag{11}$$

where

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1 - \bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t}\mathbf{x}_t \qquad \tilde{\beta}_t := \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}\beta_t \tag{12}$$

In this case, we could sample $\mathbf{x}_{t-1}$ by

$$\mathbf{x}_{t-1} = \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) + \sqrt{\tilde{\beta}_t}\epsilon \tag{13}$$

The variance controlled by $\tilde{\beta}_t$ is known, we just need to estimate $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0)$, since $\mathbf{x}_0$ is unknown in practical reverse process.

Further only $\mathbf{x}_0$ is unknown so we want to estimate $\mathbf{x}_0$, based on (10), that is

$$\mathbf{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1 - \bar{\alpha}_t}\epsilon) \tag{14}$$

Put (14) into (12), we have

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{\alpha_t}}(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}}\epsilon) \tag{15}$$

As a result, the final objective is to estimate ϵ, which is the noise between $\mathbf{x}_t$ and $\mathbf{x}_0$.

So, in the article, given the input $\mathbf{x}_t$ and t, the model's output is $\epsilon_\theta(\mathbf{x}_t, t)$ to estimate ϵ, which is trained by neural network.

Note that

- Here $\mathbf{x}_0$ is not the original image, but should be seen as the iteration direction.
- We can't directly use (14) to obtain $\mathbf{x}_0$, which doesn't follow the Markov chain.
- t is also the input, because $\mathbf{x}_t$ is obtained by $\mathbf{x}_0$ directly (10). And it reflects the noise intensity.

1.4 Theory

In this paper, it takes likelihood method to estimate $p_\theta(\mathbf{x}_0)$ and proves the process is feasible, we consider the *variational bound* on negative log likelihoood.

$$\mathbb{E}\left[-\log p_\theta(\mathbf{x}_0)\right] \leq \mathbb{E}_q\left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right] = \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\geq 1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right] =: L \tag{16}$$

Further improvements come from variance reduction by rewriting L as

$$\mathbb{E}_q\Bigg[\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p(\mathbf{x}_T))}_{L_T} + \sum_{t>1} \underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}\Bigg] \tag{17}$$

It uses KL divergence to compare $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ and forward process posterior.

1.4.1 Forward process and L_T

In the forward process, the parameter β_t can be learnable by reparameterization, but we just fix them to constants. Thus, there is no learnable parameters, and L_T can be seen as a constant and ignored.

1.4.2 Reverse process and $L_{1:T-1}$

Our estimation model is $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$. Presuppose the $\boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)$ as $\sigma^2\mathbf{I}$. We can just compare the mean, that is

$$L_{t-1} = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2\right] + C \tag{18}$$

In terms of (12), we train

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) \tag{19}$$

In the end, the loss function can be simplified as

$$\mathbb{E}_{\mathbf{x}_0, \boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2\right] \tag{20}$$

Notice that the factorial part can be ignored since they are fixed. That's why in the Algorithm 1, the loss just compare ϵ and ϵ_θ.

1.5 Method

1.5.1 Training

Algorithm 1 Training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
    $\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$
until converged

In fact, the model ϵ_θ is trained by the input $\mathbf{x}_t$ and t to estimate the noise ϵ, which is noise intensity between t and 0. The measurement is MSE and take gradient descent to obtain the solution.

1.5.2 Sampling

Algorithm 2 Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right) + \sigma_t \mathbf{z}$
end for
return $\mathbf{x}_0$

In reverse process, we must generate step by step as mentioned above. Using the estimated ϵ_θ to construct the estimated mean and then sample by the reparameterization (13). In the end, generate the new $\mathbf{x}_0$

1.5.3 Parameters

- T should big enough
- $\alpha_t < 1$ and should be approximate to 1.

According to (6), the reason is that $\mathbf{x}_t$ should retain the information $\mathbf{x}_{t-1}$ but not equal to 1 since it would be a deterministic process.

Besides, based on (10), we want $\sqrt{\bar{\alpha}_t}$ close to 0 and $\sqrt{1-\bar{\alpha}_t}$ close to 1. Thus, we want T big enough.

1.6 Supplement

1.6.1 Proof of (10)

$$
\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
&= \sqrt{\alpha_t}(\sqrt{\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{\sqrt{\alpha_t-\alpha_t\alpha_{t-1}}^2 + \sqrt{1-\alpha_t}^2}\bar{\epsilon}_{t-2} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\epsilon}_{t-2} \\
&= \ldots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon
\end{aligned}
\tag{21}
$$

1.6.2 Proof of (12)

$$
\begin{aligned}
q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1},\mathbf{x}_0)\,q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \\
&= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})\,q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \\
&= \frac{\mathcal{N}\left(\mathbf{x}_t;\sqrt{\alpha_t}\mathbf{x}_{t-1},(1-\alpha_t)\mathbf{I}\right)\mathcal{N}\left(\mathbf{x}_{t-1};\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0,(1-\bar{\alpha}_{t-1})\mathbf{I}\right)}{\mathcal{N}\left(\mathbf{x}_t;\sqrt{\bar{\alpha}_t}\mathbf{x}_0,(1-\bar{\alpha}_t)\mathbf{I}\right)} \\
&\propto \exp\left\{-\left[\frac{\left(\mathbf{x}_t-\sqrt{\alpha_t}\mathbf{x}_{t-1}\right)^2}{2(1-\alpha_t)} + \frac{\left(\mathbf{x}_{t-1}-\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0\right)^2}{2(1-\bar{\alpha}_{t-1})} - \frac{\left(\mathbf{x}_t-\sqrt{\bar{\alpha}_t}\mathbf{x}_0\right)^2}{2(1-\bar{\alpha}_t)}\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{\left(\mathbf{x}_t-\sqrt{\alpha_t}\mathbf{x}_{t-1}\right)^2}{1-\alpha_t} + \frac{\left(\mathbf{x}_{t-1}-\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0\right)^2}{1-\bar{\alpha}_{t-1}} - \frac{\left(\mathbf{x}_t-\sqrt{\bar{\alpha}_t}\mathbf{x}_0\right)^2}{1-\bar{\alpha}_t}\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{-2\sqrt{\alpha_t}\mathbf{x}_t\mathbf{x}_{t-1}+\alpha_t\mathbf{x}_{t-1}^2}{1-\alpha_t} + \frac{\mathbf{x}_{t-1}^2-2\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_{t-1}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}} + C(\mathbf{x}_t,\mathbf{x}_0)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[-\frac{2\sqrt{\alpha_t}\mathbf{x}_t\mathbf{x}_{t-1}}{1-\alpha_t} + \frac{\alpha_t\mathbf{x}_{t-1}^2}{1-\alpha_t} + \frac{\mathbf{x}_{t-1}^2}{1-\bar{\alpha}_{t-1}} - \frac{2\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_{t-1}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}} + C(\mathbf{x}_t,\mathbf{x}_0)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\left(\frac{\alpha_t}{1-\alpha_t}+\frac{1}{1-\bar{\alpha}_{t-1}}\right)\mathbf{x}_{t-1}^2 - 2\left(\frac{\sqrt{\alpha_t}\mathbf{x}_t}{1-\alpha_t}+\frac{\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}}\right)\mathbf{x}_{t-1} + C(\mathbf{x}_t,\mathbf{x}_0)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{\alpha_t(1-\bar{\alpha}_{t-1})+1-\alpha_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}\mathbf{x}_{t-1}^2 - 2\left(\frac{\sqrt{\alpha_t}\mathbf{x}_t}{1-\alpha_t}+\frac{\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}}\right)\mathbf{x}_{t-1} + C(\mathbf{x}_t,\mathbf{x}_0)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{\alpha_t-\bar{\alpha}_t+1-\alpha_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}\mathbf{x}_{t-1}^2 - 2\left(\frac{\sqrt{\alpha_t}\mathbf{x}_t}{1-\alpha_t}+\frac{\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}}\right)\mathbf{x}_{t-1} + C(\mathbf{x}_t,\mathbf{x}_0)\right]\right\} \\
&= \exp\left\{-\frac{1}{2}\left[\frac{1-\bar{\alpha}_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}\mathbf{x}_{t-1}^2 - 2\left(\frac{\sqrt{\alpha_t}\mathbf{x}_t}{1-\alpha_t}+\frac{\sqrt{\alpha_{t-1}}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}}\right)\mathbf{x}_{t-1}\right] - \frac{1}{2}C(\mathbf{x}_t,\mathbf{x}_0)\right\} \\
&= \exp\left\{-\frac{1}{2}\left(\frac{1-\bar{\alpha}_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}\right)\left[\mathbf{x}_{t-1}^2 - 2\frac{\left(\frac{\sqrt{\alpha_t}\mathbf{x}_t}{1-\alpha_t}+\frac{\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0}{1-\bar{\alpha}_{t-1}}\right)}{\frac{1-\bar{\alpha}_t}{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}}\mathbf{x}_{t-1}\right] - \frac{1}{2}C(\mathbf{x}_t,\mathbf{x}_0)\right\} \\
&= \exp\left\{-\frac{1}{2}\left(\frac{1}{\frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}}\right)\left[\mathbf{x}_{t-1}^2 - 2\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{x}_t+\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)\mathbf{x}_0}{1-\bar{\alpha}_t}\mathbf{x}_t\right] - \frac{1}{2}C(\mathbf{x}_t,\mathbf{x}_0)\right\}
\end{aligned}
$$

$$
= \exp \left\{ -\frac{\left(\mathbf{x}_{t-1} - \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)\mathbf{x}_0}{1-\bar{\alpha}_t}\right)^2}{2\left(\frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\right)} \right\}
$$

$$
\propto \mathscr{N}\left(\mathbf{x}_{t-1}; \underbrace{\frac{\sqrt{\alpha_t}\,(1-\bar{\alpha}_{t-1})\,\mathbf{x}_t + \sqrt{\bar{\alpha}_{t-1}}\,(1-\alpha_t)\,\mathbf{x}_0}{1-\bar{\alpha}_t}}_{\tilde{\mu}(\mathbf{x}_t,\mathbf{x}_0)}, \underbrace{\frac{(1-\alpha_t)\,(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}}_{\tilde{\beta}_t}\mathbf{I}\right)
$$

1.6.3 Proof of (17)

$$
\begin{aligned}
L &= \mathbb{E}_q\left[-\log\frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right] \\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\geq 1}\log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right] \\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t>1}\log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} - \log\frac{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)}\right] \\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t>1}\log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}\cdot\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} - \log\frac{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)}\right] \\
&= \mathbb{E}_q\left[-\log\frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} - \sum_{t>1}\log\frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)} - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)\right] \\
&= \mathbb{E}_q\left[D_{\mathrm{KL}}(q(\mathbf{x}_T|\mathbf{x}_0)\,\|\,p(\mathbf{x}_T)) + \sum_{t>1}D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)\,\|\,p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)\right]
\end{aligned} \tag{22}
$$

2 Denoising Diffusion Implicit Models

Denoising Diffusion Implicit Models

2.1 Overview

The generation in DDPM is very slow, so DDIM is proposed to accelerate the sampling process. DDIM has the same training process with DDPM, but can accelerate sampling. So DDIM can be seen as a new sampling method, but decrease the diversity. Review that the simplified loss of DDPM is

$$
L = \sum_t \left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2 \tag{23}
$$

which only depends on the $q(\mathbf{x}_t|\mathbf{x}_0)$, but not directly on the joint $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$. Since there are many inference distributions with the same marginals, we explore alternative inference processes that are non-Markovian, which leads to new generative processes.

2.2 Non-Markovian

In the DDPM, we estimate

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|, \mathbf{x}_0)} \tag{24}$$

$$= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \tag{25}$$

If we want to jump over some steps, then we need to estimate

$$q(\mathbf{x}_s|\mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t|\mathbf{x}_s, \mathbf{x}_0)q(\mathbf{x}_s|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \tag{26}$$

where $s < t - 1$ and the $\mathbf{x}_0$ cannot be dropped. Note that $q(\mathbf{x}_s|\mathbf{x}_0)$ and $q(\mathbf{x}_t|\mathbf{x}_0)$ are defined as DDPM, and the other is unknown.

Now we may suppose the form of $q(\mathbf{x}_s|\mathbf{x}_t, \mathbf{x}_0)$ and use method of undefined coefficients to obtain its analytical form. In DDPM, the mean of $\mathbf{x}_{t-1}$ is the weight summation of $\mathbf{x}_t$ and $\mathbf{x}_0$. So it implies that we set the form as

$$q(\mathbf{x}_s|\mathbf{x}_t, \mathbf{x}_0) \sim \mathcal{N}(k\mathbf{x}_0 + m\mathbf{x}_t, \sigma^2\mathbf{I}) \tag{27}$$

$$\mathbf{x}_s = (k\mathbf{x}_0 + m\mathbf{x}_t) + \sigma\epsilon \tag{28}$$

Replace $\mathbf{x}_t$ by (10), we have

$$\begin{aligned}
\mathbf{x}_s &= (k\mathbf{x}_0 + m\mathbf{x}_t) + \sigma\epsilon \\
&= k\mathbf{x}_0 + m(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{(1-\bar{\alpha}_t)}\epsilon') + \sigma\epsilon \\
&= (k + m\sqrt{\bar{\alpha}_t})\mathbf{x}_0 + (m\sqrt{(1-\bar{\alpha}_t)}\epsilon' + \sigma\epsilon) \\
&= (k + m\sqrt{\bar{\alpha}_t})\mathbf{x}_0 + \sqrt{m^2(1-\bar{\alpha}_t) + \sigma^2}\epsilon
\end{aligned} \tag{29}$$

In addition, we have

$$\mathbf{x}_s = \sqrt{\bar{\alpha}_s}\mathbf{x}_0 + \sqrt{(1-\bar{\alpha}_s)}\epsilon \tag{30}$$

And we can solve k and m,

$$m = \frac{\sqrt{1-\bar{\alpha}_s-\sigma^2}}{\sqrt{1-\bar{\alpha}_t}} \tag{31}$$

$$k = \sqrt{\bar{\alpha}_s} - \frac{\sqrt{1-\bar{\alpha}_s-\sigma^2}}{\sqrt{1-\bar{\alpha}_t}}\sqrt{\bar{\alpha}_t} \tag{32}$$

Finally,

$$q(\mathbf{x}_s|\mathbf{x}_t, \mathbf{x}_0) \sim \mathcal{N}(\sqrt{\bar{\alpha}_s}\mathbf{x}_0 + \frac{\sqrt{1-\bar{\alpha}_s-\sigma^2}}{\sqrt{1-\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0), \sigma^2\mathbf{I}) \tag{33}$$

$$\mathbf{x}_s = \sqrt{\bar{\alpha}_s}\mathbf{x}_0 + \frac{\sqrt{1-\bar{\alpha}_s-\sigma^2}}{\sqrt{1-\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0) + \sigma\epsilon \tag{34}$$

To obtain the form in this article, we just let $s = t-1$, and $\mathbf{x}_0$ is predicted by our train model ϵ_θ. $\sigma = \sigma_t$ represents the random noise.

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\left(\frac{\boldsymbol{x}_t - \sqrt{1-\alpha_t}\epsilon_\theta^{(t)}(\boldsymbol{x}_t)}{\sqrt{\alpha_t}} \right)}_{\text{"predicted } \boldsymbol{x}_0 \text{"}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2} \cdot \epsilon_\theta^{(t)}(\boldsymbol{x}_t)}_{\text{"direction pointing to } \boldsymbol{x}_t \text{"}} + \underbrace{\sigma_t \epsilon_t}_{\text{random noise}} \tag{35}$$

In fact, the first two term in right hand should be seen as the mean of $\mathbf{x}_{t-1}$.
We try to understand the formula by Figure 2.

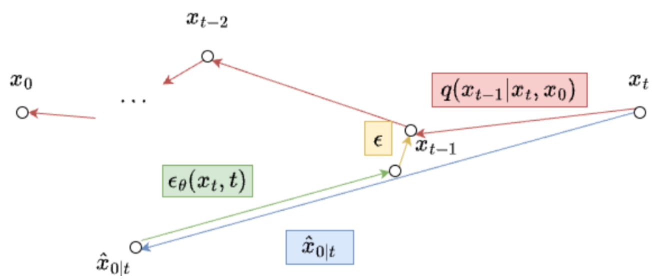

Figure 2: DDIM sampling

Note that $\hat{x}_{0|t}$ is the estimation of x_0.

1. The first term $\sqrt{\alpha_{t-1}}$ is the estimation of x_0 multiplying a coefficient as shown in the blue part in Figure 2.
2. The second term means return some distance to x_t (the green part). If this part is 0, this makes the sampling process more volatile, so it ensure the process smooth.
3. The third term is a little disturbance (The yellow part). If $\sigma_t = 0$, then randomness is lost and x^T can define the x_0, which is similar to a latent variable and

2.3 Choice of variance

- $\sigma = 0$: it becomes a deterministic process, has the best speed but may lose diversity.
- $\sigma = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$: it is equal to DDPM (as paper mentioned). Also require $s = t-1$.

2.4 Experiments

To compare DDIM and DDPM, the paper set $\sigma_t = \eta_t \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$. If $\eta = 1$, it becomes DDPM. If $\eta = 0$, it is DDIM.

		CIFAR10 (32×32)				
	S	10	20	50	100	1000
η	0.0	**13.36**	**6.84**	**4.67**	**4.16**	4.04
	0.2	14.04	7.11	4.77	4.25	4.09
	0.5	16.66	8.35	5.25	4.46	4.29
	1.0	41.07	18.36	8.01	5.78	4.73

Figure 3: DDIM results

2.4.1 Supplement

When $\sigma = \sqrt{\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t}$, DDIM is equivalent to DDPM.

Obviously, their variance are equal, so we just compare their mean, that is

$$\sqrt{\bar{\alpha}_{t-1}}(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t)}{\sqrt{\bar{\alpha}_t}}) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma^2}\epsilon_\theta(x_t) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t))$$

First, simplify $\sqrt{1-\bar{\alpha}_{t-1}-\sigma^2}$,

$$\begin{aligned}
\sqrt{1-\bar{\alpha}_{t-1}-\sigma^2} &= \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{1-\bar{\alpha}_t}}\sqrt{1-\bar{\alpha}_{t-1}-\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}(1-\alpha_t)} \\
&= \frac{\sqrt{(1-\bar{\alpha}_{t-1})(1-\frac{1}{1-\bar{\alpha}_t}(1-\alpha_t))(1-\bar{\alpha}_t)}}{\sqrt{1-\bar{\alpha}_t}} \\
&= \frac{\sqrt{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_t-1+\alpha_t)}}{\sqrt{1-\bar{\alpha}_t}} \\
&= \frac{\sqrt{(1-\bar{\alpha}_{t-1})(\alpha_t-\bar{\alpha}_t)}}{\sqrt{1-\bar{\alpha}_t}} \\
&= \frac{\sqrt{(1-\bar{\alpha}_{t-1})(1-\bar{\alpha}_{t-1})\alpha_t}}{\sqrt{1-\bar{\alpha}_t}} \\
&= \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{\sqrt{1-\bar{\alpha}_t}}
\end{aligned}$$

Then we have

$$\begin{aligned}
&\sqrt{\bar{\alpha}_{t-1}}(\frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t)}{\sqrt{\bar{\alpha}_t}}) + \sqrt{1-\bar{\alpha}_{t-1}-\sigma^2}\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}x_t - \frac{1}{\sqrt{\alpha_t}}\sqrt{1-\bar{\alpha}_t}\epsilon_\theta(x_t) + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}x_t - (\frac{1}{\sqrt{\alpha_t}}\sqrt{1-\bar{\alpha}_t} - \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}x_t - (\frac{(1-\bar{\alpha}_t)-(1-\bar{\alpha}_{t-1})\alpha_t}{\sqrt{\alpha_t}\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}x_t - (\frac{1-\bar{\alpha}_t-\alpha_t+\bar{\alpha}_t}{\sqrt{\alpha_t}\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}x_t - (\frac{1-\alpha_t}{\sqrt{\alpha_t}\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(x_t) \\
&= \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}})\epsilon_\theta(x_t)
\end{aligned}$$

3 Score-Based Generative Modeling through Stochastic Differential Equations

SCORE-BASED GENERATIVE MODELING THROUGH STOCHASTIC DIFFERENTIAL EQUATIONS

3.1 Concept

SDE firstly is proposed to describe *Brownian Motion* also called *Wiener process*.

$$W(t) \sim N(0, t) \tag{36}$$

According to its property, we have difference equation

$$W(t + \Delta t) - W(t) \sim N(0, \Delta t) \tag{37}$$

Let $\Delta t \to 0$, we can obtain the differential equation

$$\mathrm{d}w = \sqrt{\mathrm{d}t}z \tag{38}$$

where $z \sim N(0, 1)$. Moreover, the generalization of Brownian motion is proposed by Ito, which is called Ito process, that is

$$\mathrm{d}x = f(x, t)\mathrm{d}t + g(t)\mathrm{d}w \tag{39}$$

where $f(x, t)$ is drift factor and $g(t)$ is diffusion factor.

Note that forward process can be expressed by SDE. Now, We introduce an important theorem that if the process of diffusion is SDE, then the reverse is also SDE, that is

$$\mathrm{d}x = [f(x, t) - g^2(t)\nabla_x \log p_t(x)]\mathrm{d}t + g(t)\mathrm{d}w \tag{40}$$

where only $\nabla_x \log p_t(x)$, the *score*, is unknown. If it is known, we can sample in terms of the reverse SDE. Usually the score is obtained by a neural network $s_\theta(\cdot)$.

3.2 DDPM and SDE

3.2.1 SDE and reverse SDE

Review DDPM,

$$\mathbf{x}_i = \sqrt{1 - \beta_i}\mathbf{x}_{i-1} + \sqrt{\beta_i}\epsilon \quad i = 1, ..., N \tag{41}$$

In order to distinguish continuous t here and discrete t in (6), the latter t is denoted as i.

To apply differential equation, we transfer it into continuous form. Suppose

$$\mathbf{x}(t = \frac{1}{N}) = \mathbf{x}_i \quad t \in [0, 1] \tag{42}$$

$$\beta(t = \frac{i}{N}) = N\beta_i \tag{43}$$

$$\Delta t = \frac{1}{N} \tag{44}$$

Notice the setting of $\beta(t)$. In this case,

$$\begin{aligned}\mathbf{x}(t + \Delta t) &= \sqrt{1 - \frac{\beta(t + \Delta t)}{N}} \cdot \mathbf{x}(t) + \sqrt{\frac{\beta(t + \Delta t)}{N}} \cdot \epsilon \\ &= \sqrt{1 - \beta(t + \Delta t)\Delta t} \cdot \mathbf{x}(t) + \sqrt{\beta(t + \Delta t)\Delta t} \cdot \epsilon\end{aligned}$$

$$
\begin{aligned}
&\approx (1 - \frac{1}{2}\beta(t+\Delta t)\Delta t) \cdot \mathbf{x}(t) + \sqrt{\beta(t+\Delta t)\Delta t} \cdot \epsilon \quad \text{(Taylor expansion)} \\
&\approx \mathbf{x}_t - \frac{1}{2}\beta(t)\Delta t \cdot \mathbf{x}(t) + \sqrt{\beta(t)\Delta t} \cdot \epsilon \qquad (45)
\end{aligned}
$$

$$
\Rightarrow \quad \mathbf{x}(t+\Delta t) - \mathbf{x}(t) = -\frac{1}{2}\beta(t)\Delta t \cdot \mathbf{x}(t) + \sqrt{\beta(t)\Delta t} \cdot \epsilon \tag{46}
$$

Therefore, the final differential equation of DDPM is

$$
\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}w \tag{47}
$$

where $f(x,t) = -\frac{1}{2}\beta(t)\mathbf{x}(t)$ and $g(t) = \sqrt{\beta(t)}$. Further, we could derive the reverse SDE, that is

$$
\mathrm{d}\mathbf{x} = [-\frac{1}{2}\beta(t)\mathbf{x}(t) - \beta(t)s_\theta(t)]\mathrm{d}t + \sqrt{\beta(t)}\mathrm{d}w \tag{48}
$$

3.2.2 s_θ and ϵ_θ

Notice that in DDPM, we only trained the ϵ_θ, not the s_θ, we would explain the connection between them. In the process, the score is

$$
\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) = -\nabla_{\mathbf{x}_t} \frac{(\mathbf{x}_t - \mu_t)^2}{2\sigma_t^2} = -\frac{\mathbf{x}_t - \mu_t}{\sigma_t^2} \tag{49}
$$

$$
\mathbf{x}_t = \mu_t - \sigma_t^2 \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) \tag{50}
$$

Notice that (50) has the same form of *Tweedie's Formula*. Review the $\mathbf{x}_t$ obtained by $\mathbf{x}_0$ in DDPM, we have

$$
\mu_t = \sqrt{\bar{\alpha}_t} \tag{51}
$$

$$
\sigma_t = \sqrt{1 - \bar{\alpha}_t} \tag{52}
$$

Replace them in 49, then

$$
\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) = -\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \tag{53}
$$

and

$$
\epsilon = \frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t}}{\sqrt{1 - \bar{\alpha}_t}} \tag{54}
$$

Thus

$$
s_\theta(t) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{x}_0) = -\frac{\epsilon_\theta}{\sqrt{1 - \bar{\alpha}_t}} \tag{55}
$$

Replace ϵ_θ with s_θ in sampling of DDPM, that is

$$
\mathbf{x}(t) = \frac{1}{\sqrt{1 - \beta(t+1)}}[\mathbf{x}(t+1) + \beta(t+1)s_\theta(t+1)] + \sqrt{\beta(t+1)}\epsilon \tag{56}
$$

3.2.3 Euler-Maruyama

Euler-Maruyama method is a numerical method for solving SDE. It is an extension of Euler method for solving ordinary differential equations (ODE).

Review of the Euler method in ODE,

$$\frac{\mathrm{d}x}{\mathrm{d}t} = a(X(t)) \tag{57}$$

$$X(t+\Delta t) = X(t) + a(X(t))\Delta t \tag{58}$$

In SDE, the form is (39), and the Euler-Maruyama method is

$$x(t+\Delta t) = x(t) + f(x,t)\Delta t + g(t)\Delta w \tag{59}$$

Consider the discrete situation, we have

$$\mathbf{x}_{t+1} = \mathbf{x}_t + f(x,t)\Delta t + g(t)\sqrt{\Delta t} z \tag{60}$$

$$\mathbf{x}_{t+1} = \mathbf{x}_t + f(x,t) + g(t)z \tag{61}$$

If we presume Δt fixed, then it can be ignored. This is the Euler-Maruyama method in discrete situation.

Then the reverse of Euler-Maruyama method is

$$\begin{aligned}
\mathbf{x}_t &= \mathbf{x}_{t+1} - f(x,t) + g^2(t)s_\theta(t) + g(t)z && (62)\\
&= \mathbf{x}_{t+1} + \frac{1}{2}\beta_{t+1}\mathbf{x}_{t+1} + \beta_{t+1}s_\theta + \sqrt{\beta_{t+1}}z && (63)\\
&\approx \mathbf{x}_{t+1} + \frac{1}{2}\beta_{t+1}\mathbf{x}_{t+1} + \beta_{t+1}s_\theta + \frac{1}{2}\beta_{t+1}^2 s_\theta + \sqrt{\beta_{t+1}}z && (64)\\
&\approx (1+\frac{1}{2}\beta_{t+1})(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (65)\\
&\approx (1+\frac{1}{2}\beta_{t+1} + o(\beta_{t+1}))(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (66)\\
&\approx \frac{1}{\sqrt{1-\beta_{t+1}}}(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (67)
\end{aligned}$$

where $\frac{1}{2}\beta_{t+1}^2 s_\theta$ is infinitesimal about β. Adding this term is to identify common factors. Then continually adding infinitesimal is to reversely use Taylor expansion.

To derive the VP-SDE sampling formula,

$$\begin{aligned}
(1+\frac{1}{2}\beta_{t+1})(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z &= (2-(1-\frac{1}{2}\beta_{t+1}))(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (68)\\
&\approx (2-(1-\frac{1}{2}\beta_{t+1}) + o(\beta_{t+1}))(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (69)\\
&== (2-\sqrt{1-\beta_{t+1}})(\mathbf{x}_{t+1} + \beta_{t+1}s_\theta) + \sqrt{\beta_{t+1}}z && (70)
\end{aligned}$$

3.2.4 Predictor Corrector

Algorithm 3 PC sampling (VP SDE)

$\mathbf{x}_N \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $i = N - 1$ **to** 0 **do**
  $\mathbf{x}'_i \leftarrow (2 - \sqrt{1 - \beta_{i+1}})\mathbf{x}_{i+1} + \beta_{i+1}\mathbf{s}_{\theta*}(\mathbf{x}_{i+1}, i + 1)$
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\mathbf{x}_i \leftarrow \mathbf{x}'_i + \sqrt{\beta_{i+1}}\mathbf{z}$ Predictor
  for $j = 1$ **to** M **do**   Corrector
    $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \epsilon_i \mathbf{s}_{\theta*}(\mathbf{x}_i, i) + \sqrt{2\epsilon_i}\mathbf{z}$
return $\mathbf{x}_0$

In the corrector part, we take **Langevin Dynamics** method.