

---

# Notes for Diffusion Model

---

Jingxuan Zhang

Department of Computer Science and Engineering

East China University of Science and Technology

Shanghai, China

y21220033@mail.ecust.edu.cn

## 1 DDPM

We first introduce the basic theory of Denoising Diffusion Probabilistic Models (DDPM) [1]. Overall, the DDPM consists of two processes: a forward diffusion process that gradually adds noise to the data, and a reverse denoising process that learns to remove the noise and recover the original data.

### 1.1 Forward Diffusion Process

The forward diffusion process is defined as a Markov chain that progressively adds Gaussian noise to the data over  $T$  time steps. Given a data point  $\mathbf{x}_0$  sampled from the data distribution  $q(\mathbf{x}_0)$ , the forward process produces a sequence of noisy samples  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_T$  according to the following transition probabilities:

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

where  $\beta_t$  is a variance schedule that controls the amount of noise added at each time step. The cumulative effect of this process can be expressed as:

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}), \quad (2)$$

where  $\alpha_t = 1 - \beta_t$ ,  $\bar{\alpha}_t = \prod_{i=1}^T \alpha_i$ .

### 1.2 Reverse Diffusion Process

The reverse process can be shown as:

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t) = \frac{q(\mathbf{x}_t, \mathbf{x}_{t-1})}{q(\mathbf{x}_t)} = \frac{q(\mathbf{x}_t | \mathbf{x}_{t-1}) q(\mathbf{x}_{t-1})}{q(\mathbf{x}_t)} \quad (3)$$

According to Eq. (1) and the definition of the Gaussian distribution, we have:

$$\begin{aligned} q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) &= q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0) \frac{q(\mathbf{x}_{t-1} | \mathbf{x}_0)}{q(\mathbf{x}_t | \mathbf{x}_0)} \\ &\propto \exp \left( -\frac{1}{2} \left( \frac{(\mathbf{x}_t - \sqrt{\alpha_t} \mathbf{x}_{t-1})^2}{\beta_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_{t-1}} \mathbf{x}_0)^2}{1 - \bar{\alpha}_{t-1}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \mathbf{x}_0)^2}{1 - \bar{\alpha}_t} \right) \right) \\ &= \exp \left( -\frac{1}{2} \left( \left( \frac{\alpha_t}{\beta_t} + \frac{1}{1 - \bar{\alpha}_{t-1}} \right) \mathbf{x}_{t-1}^2 - \left( \frac{2\sqrt{\alpha_t}}{\beta_t} \mathbf{x}_t + \frac{2\sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \mathbf{x}_0 \right) \mathbf{x}_{t-1} + C(\mathbf{x}_t, \mathbf{x}_0) \right) \right) \end{aligned} \quad (4)$$

Hence, we can get the expressions of the Gaussian parameters of  $q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ :

$$q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_{t-1}; \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}\right) \approx \exp \left( -\frac{(x - \tilde{\mu}(x_t, x_0))^2}{2\tilde{\beta}_t} \right), \quad (5)$$

where the expression of  $\tilde{\beta}_t$  and  $\tilde{\mu}(x_t, x_0)$  are:

$$\tilde{\beta}_t = 1 / \left( \frac{\alpha_t}{\beta_t} + \frac{1}{1 - \bar{\alpha}_{t-1}} \right) = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \cdot \beta_t, \quad (6)$$

$$\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) = \left( \frac{\sqrt{\alpha_t}}{\beta_t} \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_t}}{1 - \bar{\alpha}_t} \mathbf{x}_0 \right) / \left( \frac{\alpha_t}{\beta_t} + \frac{1}{1 - \bar{\alpha}_{t-1}} \right) = \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0. \quad (7)$$

According to Eq. (2), we can re-express  $\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$  as:

$$\begin{aligned} \tilde{\mu}_t(x_t, x_0) &= \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} x_0 \\ &= \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \cdot \frac{1}{\sqrt{\alpha_t}} (x_t - \sqrt{1 - \bar{\alpha}_t} z_t) \\ &= \frac{\sqrt{\alpha_t} \cdot \sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{\sqrt{\alpha_t} \cdot (1 - \bar{\alpha}_t)} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \cdot \frac{1}{\sqrt{\alpha_t}} (x_t - \sqrt{1 - \bar{\alpha}_t} z_t) \\ &= \frac{\alpha_t - \bar{\alpha}_t}{\sqrt{\alpha_t} (1 - \bar{\alpha}_t)} x_t + \frac{\beta_t}{(1 - \bar{\alpha}_t) \sqrt{\alpha_t}} (x_t - \sqrt{1 - \bar{\alpha}_t} z_t) \\ &= \frac{1 - \bar{\alpha}_t}{\sqrt{\alpha_t} (1 - \bar{\alpha}_t)} x_t - \frac{\beta_t}{(1 - \bar{\alpha}_t) \sqrt{\alpha_t}} (\sqrt{1 - \bar{\alpha}_t} z_t) \\ &= \frac{1}{\sqrt{\alpha_t}} x_t - \frac{\beta_t}{\sqrt{(1 - \bar{\alpha}_t) \sqrt{\alpha_t}}} z_t \\ &= \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{(1 - \bar{\alpha}_t)}} z_t \right) \end{aligned} \quad (8)$$

Therefore, after we sample a  $z \sim \mathcal{N}(0, \mathbf{I})$  and train a UNet model  $z_t = \text{UNet}(x_t, t)$ , we get  $x_{t-1}$ :

$$x_{t-1} = \tilde{\mu}_t(x_t, x_0) + \sqrt{\tilde{\beta}_t} z = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{(1 - \bar{\alpha}_t)}} z_t \right) + \sqrt{\tilde{\beta}_t} z. \quad (9)$$

The reverse diffusion process aims to learn a parameterized model  $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$  that can reverse the forward diffusion process. The reverse process is also defined as a Markov chain, but it starts from pure noise  $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$  and iteratively denoises the samples to recover the original data:

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)), \quad (10)$$

where  $\mu_\theta$  and  $\Sigma_\theta$  are neural networks that predict the mean and covariance of the reverse transition.

### 1.3 Training Objective

The training objective of DDPM is to minimize the variational bound on the negative log-likelihood of the data. This can be simplified to a mean squared error loss between the predicted noise and the true noise added during the forward process:

$$L(\theta) = \mathbb{E}_{\mathbf{x}_0, \epsilon, t} [\|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2], \quad (11)$$

where  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$  is the noise added to the data, and  $\epsilon_\theta$  is the neural network that predicts the noise given the noisy sample  $\mathbf{x}_t$  and time step  $t$ .

## 2 DDIM

### References

- [1] J. Ho, A. Jain, and P. Abbeel, “Denoising diffusion probabilistic models,” in *NeurIPS*, 2020, pp. 6840–6851.