A Note on Denoising Diffusion Probabilistic Model

J. S.

2025.02.28

Abstract

This note is a personal summary of the mathematical foundations of *Denoising Diffusion Probabilistic Model* (DDPM) [Ho et al., 2020].

Contents

1 Background: Diffusion

Suppose we have observation $\mathbf{x}_0 \in \mathbb{R}^d$, and latent variables $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T$ that are of the same dimensionality as $\mathbf{x}_0$. We use the notation $\mathbf{x}_{a:b}$ to denote the collection of $\mathbf{x}$ from index a to index b (endpoints included), e.g., $p(\mathbf{x}_{1:T}) = p(\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_T)$.

Diffusion models are latent variable models of the form $p_\theta(\mathbf{x}_0) := \int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T}$, where $\mathbf{x}_1, \ldots, \mathbf{x}_T$ are latent variables. The joint distribution $p_\theta(\mathbf{x}_{0:T})$ is called the **reverse process**, and it is defined as a Markov chain with learned Gaussian transitions:

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I}),$$

$$
\begin{aligned}
p_\theta(\mathbf{x}_{0:T}) &:= p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \\
p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &:= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)).
\end{aligned} \tag{1.1}
$$

In diffusion model, we approximate the posterior distribution of latent variables by $q(\mathbf{x}_{1:T}|\mathbf{x}_0)$, which is called the **forward process** or **diffusion process**. The forward process is fixed to a Markov chain that gradually adds Gaussian noise to the data according to a variance schedule $\beta_1, \ldots, \beta_T$:

$$
\begin{aligned}
q(\mathbf{x}_{1:T}|\mathbf{x}_0) &:= \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \\
q(\mathbf{x}_t|\mathbf{x}_{t-1}) &:= \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}).
\end{aligned} \tag{1.2}
$$

Note. Why choose $\sqrt{1-\beta_t}$ as the scale of mean? By choosing the scale $\sqrt{1-\beta_t}$, we have

$$
\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \beta_t),
$$

$$
\begin{aligned}
\mathrm{Var}(\mathbf{x}_t) &= (1-\beta_t)\mathrm{Var}(\mathbf{x}_{t-1}) + \mathrm{Var}(\boldsymbol{\epsilon}_t) \\
&= (1-\beta_t)\mathrm{Var}(\mathbf{x}_{t-1}) + \beta_t.
\end{aligned}
$$

It's easy to verify that if $\mathrm{Var}(\mathbf{x}_0) = 1$, then $\mathrm{Var}(\mathbf{x}_t) = 1$ for all $t \geq 1$. So the variance is stablized in the diffusion process.

The objective is to minimize the negative log-likelihood $-\log p_\theta(\mathbf{x}_0)$. This is equivalent to minimizing its upper bound L, given by

$$
\begin{aligned}
-\log p_\theta(\mathbf{x}_0) &= -\log \int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T} \\
&= -\log \int q(\mathbf{x}_{1:T}|\mathbf{x}_0) \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} d\mathbf{x}_{1:T} \\
&\leq \mathbb{E}_{\mathbf{x}_{1:T}\sim q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right] \\
&= \mathbb{E}_{\mathbf{x}_{1:T}\sim q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[-\log \left(p_\theta(\mathbf{x}_T) \frac{\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right) \right] \\
&= \mathbb{E}_{\mathbf{x}_{1:T}\sim q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[-\log p_\theta(\mathbf{x}_T) - \sum_{t=1}^{T} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right] =: L.
\end{aligned} \tag{1.3}
$$

A notable property of the forward process is that it admits sampling $\mathbf{x}_t$ at an arbitrary timestep t in closed form. Using the notation $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$, we have

$$
q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}). \tag{1.4}
$$

Note. See the appendix in section 3.2 for the derivation of (1.4).

Efficient training is therefore possible by optimizing random terms of L with stochastic gradient descent. Further improvements come from variance reduction by rewriting L as:

$$\begin{aligned}
L &= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \sum_{t=1}^{T}\log\frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \log\frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \log\frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} + \sum_{t=2}^{T}\log\left(\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\cdot\frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}\right)\right]\\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \log\frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \log\frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} + \log\prod_{t=2}^{T}\frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) + \log\frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} + \log\frac{q(\mathbf{x}_T|\mathbf{x}_0)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[\log\frac{q(\mathbf{x}_T|\mathbf{x}_0)}{p(\mathbf{x}_T)} - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1) + \sum_{t=2}^{T}\log\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}\right]\\
&= \mathbb{E}_q\left[\underbrace{D_{\mathrm{KL}}\Big(q(\mathbf{x}_T|\mathbf{x}_0)\|p(\mathbf{x}_T)\Big)}_{L_T} + \underbrace{\Big(-\log p_\theta(\mathbf{x}_0|\mathbf{x}_1)\Big)}_{L_0} + \sum_{t=2}^{T}\underbrace{D_{\mathrm{KL}}\Big(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)\|p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)\Big)}_{L_{t-1}}\right].
\end{aligned} \tag{1.5}$$

Note. The marginal distribution $p(\mathbf{x}_T)$ is set to $\mathcal{N}(\mathbf{x}_T;\mathbf{0},\mathbf{I})$ as previously mentioned. Since there is no trainable parameter, we use $p(\mathbf{x}_T)$ instead of $p_\theta(\mathbf{x}_T)$.

Previously we assumed

$$\begin{aligned}
p(\mathbf{x}_T) &= \mathcal{N}(\mathbf{x}_T;\mathbf{0},\mathbf{I}),\\
p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &:= \mathcal{N}(\mathbf{x}_{t-1};\boldsymbol{\mu}_\theta(\mathbf{x}_t,t),\boldsymbol{\Sigma}_\theta(\mathbf{x}_t,t)),
\end{aligned} \tag{1.1}$$

and derived

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t;\sqrt{\bar\alpha_t}\mathbf{x}_0,(1-\bar\alpha_t)\mathbf{I}). \tag{1.4}$$

After some algebraic operations, we can also derive that

$$\begin{aligned}
&q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1};\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0),\tilde\beta_t\mathbf{I})\\
&\qquad\text{where}\quad \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0) := \frac{\sqrt{\bar\alpha_{t-1}}\beta_t}{1-\bar\alpha_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar\alpha_{t-1})}{1-\bar\alpha_t}\mathbf{x}_t\\
&\qquad\text{and}\quad \tilde\beta_t := \frac{1-\bar\alpha_{t-1}}{1-\bar\alpha_t}\beta_t.
\end{aligned} \tag{1.6}$$

Note. See the appendix in section 3.1 for derivation details of (1.6).

Note. Since the noise scale $\beta_1,\beta_2,\cdots,\beta_T$ are pre-set constants, the distribution of the forward process is known. Therefore, $q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)$ is a fixed, non-trainable distribution.

- $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0)$ stands for the conditional mean of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ and $\mathbf{x}_0$.
- $\tilde\beta_t$ stands for the conditional variance of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ and $\mathbf{x}_0$.

- Since $\beta_1, \beta_2, \cdots, \beta_T$ are fixed, $\tilde{\boldsymbol{\mu}}_t$ is a fixed deterministic function of $\mathbf{x}_t$ and $\mathbf{x}_0$.

Consequently, all KL divergences in (1.5) are comparisons between Gaussians, so they can be calculated in a Rao-Blackwellized fashion with closed form expressions instead of high variance Monte-Carlo estimates.

2 Diffusion Models and Denoising Autoencoders

2.1 Forward Process

We ignore the fact that the forward process variances β_t are learnable by reparameterization and instead fix them to constants. Thus, in our implementation, the approximate posterior q has no learnable parameters, so L_T is a constant during training and can be ignored.

2.2 Reverse Process

Now we discuss our choices in $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$ for $1 < t \leq T$.

First, we set $\Sigma_\theta(\mathbf{x}_t, t) = \sigma_t^2\mathbf{I}$ to untrained time dependent constants. Experimentally, both $\sigma_t^2 = \beta_t$ and $\sigma_t^2 = \tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$ had similar results. The first choice is optimal for $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the second is optimal for $\mathbf{x}_0$ deterministically set to one point.

Second, to express the mean $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$, we propose a specific parameterization motivated by the following analysis of L_t. With $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$, we can write

$$
\begin{aligned}
L_{t-1} &= D_{\mathrm{KL}}\Big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)\Big) \\
&= D_{\mathrm{KL}}\Big(\mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}) \| \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})\Big) \\
&= D_{\mathrm{KL}}\Big(\mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \sigma_t^2\mathbf{I}) \| \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})\Big) \\
&= \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0)}\left\{\mathbb{E}_{\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)}\left[\frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2\right]\right\}.
\end{aligned}
\tag{2.1}
$$

Note. See the appendix in section 3.3 for derivation of the last step of (2.1).

We can see that the most straightforward parameterization of $\boldsymbol{\mu}_\theta$ is a model that predicts $\tilde{\boldsymbol{\mu}}_t$, the forward process posterior mean of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ and $\mathbf{x}_0$. But in the following part, other parameterization of $\boldsymbol{\mu}_\theta$ will be discussed.

Based on (2.1), The process of computing L_{t-1} can be written as:

Algorithm 1 Reverse Process

1: **for** m in $1, 2, \cdots, M$ **do**
2: Sample $\mathbf{x}_0 \sim q(\mathbf{x}_0)$. This corresponds to drawing a sample from the dataset.
3: Sample $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$. This corresponds to generating a noised sample.
4: Compute $L_{t-1}^{(m)} = \frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2$.
5: **end for**
6: Compute the expectation: $L_{t-1} = \frac{1}{M}\sum_{m=1}^{M} L_{t-1}^{(m)}$.

The above algorithm contains two sampling procedure. To sample $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ is easy, since we can simply use the dataset. However, to sample $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$ is a bit more challenging, since it involves the distribution of the forward process. Thankfully, we can explicitly give the distribution of $q(\mathbf{x}_t|\mathbf{x}_0)$ by (1.4):

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

We can reparameterize (1.4) as

$$\mathbf{x}_t = \mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{2.2}$$

So now by sampling $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and computing $\mathbf{x}_t$ as a weighted sum of $\boldsymbol{\epsilon}$ and $\mathbf{x}_0$, we can efficiently sample $\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)$.

Our goal is to write the optimization problem (2.1) as an expression with respect to $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$ only, since they are the only variables that are directly sampled. But $\mathbf{x}_t$ is still present, so we can eliminate the existence of $\mathbf{x}_t$ in (2.1) by plugging (2.2) back. We get

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0)}\left\{\mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}\left[\frac{1}{2\sigma_t^2}\left\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}), \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\right\|^2\right]\right\}. \tag{2.3}$$

Note. Remember that $\tilde{\boldsymbol{\mu}}_t$ stands for the conditional mean of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ and $\mathbf{x}_0$ in the forward process. So $\mathbf{x}_t$ and $\mathbf{x}_0$ are available as inputs. What we do is to replace $\mathbf{x}_t$ with $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$ by (2.2). So now $\tilde{\boldsymbol{\mu}}_t = \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}), \mathbf{x}_0)$. The inputs become $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$.

Note. Remember that $\boldsymbol{\mu}_\theta$ stands for the conditional mean of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ in the distribution of **reverse process** (by definition in Eq.(1)). So only $\mathbf{x}_t$ is available as input, and $\mathbf{x}_0$ cannot be used. This appears natural, since we don't know $\mathbf{x}_0$ in the reverse process. Therefore, in Eq.(8b), we can only write $\boldsymbol{\mu}_\theta$ as $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$ rather than $\boldsymbol{\mu}_\theta(\mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}), t)$.

So there are two functions that need breaking down: $\tilde{\boldsymbol{\mu}}_t$ and $\boldsymbol{\mu}_\theta$. $\boldsymbol{\mu}_\theta$ is a neural network that can be customized, and we want it to match the form of $\tilde{\boldsymbol{\mu}}_t$ in order that the similar terms can eliminated. So the concern becomes how to choose the expression of $\tilde{\boldsymbol{\mu}}_t$.

To match the form of $\boldsymbol{\mu}_\theta$, which takes $\mathbf{x}_t$ as input but doesn't take $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$ as input, we can write $\tilde{\boldsymbol{\mu}}_t$ as a combination of $\mathbf{x}_t$ and some additional term, namely $\mathbf{x}_0$ or $\boldsymbol{\epsilon}$[1]. In the expression of $\boldsymbol{\mu}_\theta$, the additional term will be modeled as a neural network[2]. Here we choose $\boldsymbol{\epsilon}$ as the additional term, so we want to get the exact form of $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \boldsymbol{\epsilon})$.

[1]Note that in (1.6) $\tilde{\boldsymbol{\mu}}_t$ is expressed with $\mathbf{x}_t$ and $\mathbf{x_0}$, and (2.2) gives the relationship of $\mathbf{x}_t$, $\mathbf{x_0}$ and ϵ. So $\tilde{\boldsymbol{\mu}}_t$ can be expressed with any two of $\mathbf{x}_t$, $\mathbf{x_0}$ and ϵ.

[2]Since only $\mathbf{x}_t$ and t are taken as input, and the additional term (either $\mathbf{x_0}$ or ϵ) is not directly taken as input, we can model the additional term as a neural network with $\mathbf{x}_t$ and t as input, which is an estimator of the intended additional term. In this way, $\boldsymbol{\mu}_\theta$ can still have a similar form with $\tilde{\boldsymbol{\mu}}_t$.

Previously, we know

$$\tilde{\boldsymbol{\mu}}_t\Big(\mathbf{x}_t, \mathbf{x}_0\Big) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0. \tag{1.6}$$

So to express $\tilde{\boldsymbol{\mu}}_t$ with $\mathbf{x}_t$ and $\boldsymbol{\epsilon}$, we need to eliminate $\mathbf{x}_0$ in the expression. We can reformulate (2.2) as

$$\hat{\mathbf{x}}_0(\mathbf{x}_t, \boldsymbol{\epsilon}) = \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}), \tag{2.4}$$

where $\hat{\mathbf{x}}_0(\mathbf{x}_t, \boldsymbol{\epsilon})$ is the predicted value of $\mathbf{x}_0$ given $\mathbf{x}_t$ and $\boldsymbol{\epsilon}$. Substituting $\mathbf{x}_0$ in (1.6) by $\hat{\mathbf{x}}_0(\mathbf{x}_t, \boldsymbol{\epsilon})$, we obtain:

$$\begin{aligned}
\tilde{\boldsymbol{\mu}}_t &= \tilde{\boldsymbol{\mu}}_t\Big(\mathbf{x}_t, \hat{\mathbf{x}}_0(\mathbf{x}_t, \boldsymbol{\epsilon})\Big) \\
&= \tilde{\boldsymbol{\mu}}_t\left(\mathbf{x}_t, \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon})\right) && \text{(By 2.4)} \\
&= \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\left[\frac{1}{\sqrt{\bar{\alpha}_t}}\Big(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}\Big)\right] + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t && \text{(By 1.6)} \\
&= \left[\frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{(1-\bar{\alpha}_t)\sqrt{\bar{\alpha}_t}} + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\right]\mathbf{x}_t - \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{\sqrt{\bar{\alpha}_t}(1-\bar{\alpha}_t)}\sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \\
&= \left[\frac{\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{\alpha_t}}(1-\alpha_t) + \sqrt{\alpha_t}\sqrt{\bar{\alpha}_t}(1-\frac{\bar{\alpha}_t}{\alpha_t})}{(1-\bar{\alpha}_t)\sqrt{\bar{\alpha}_t}}\right]\mathbf{x}_t - \frac{\beta_t}{\sqrt{\alpha_t}\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon} \\
&= \left[\frac{\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{\alpha_t}} - \sqrt{\alpha_t}\sqrt{\bar{\alpha}_t} + \sqrt{\alpha_t}\sqrt{\bar{\alpha}_t} - \bar{\alpha}_t\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{\alpha_t}}}{(1-\bar{\alpha}_t)\sqrt{\bar{\alpha}_t}}\right]\mathbf{x}_t - \frac{1}{\sqrt{\alpha_t}}\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon} \\
&= \left[\frac{(1-\bar{\alpha}_t)\frac{\sqrt{\bar{\alpha}_t}}{\sqrt{\alpha_t}}}{(1-\bar{\alpha}_t)\sqrt{\bar{\alpha}_t}}\right]\mathbf{x}_t - \frac{1}{\sqrt{\alpha_t}}\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon} \\
&= \frac{1}{\sqrt{\alpha_t}}\mathbf{x}_t - \frac{1}{\sqrt{\alpha_t}}\frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon} \\
&= \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\Big).
\end{aligned} \tag{2.5}$$

So we get the form of $\tilde{\boldsymbol{\mu}}_t$ expressed with $\mathbf{x}_t$ and $\boldsymbol{\epsilon}$.

As demonstrated before, the input to $\boldsymbol{\mu}_\theta$ is only $\mathbf{x}_t$. To match the form of $\tilde{\boldsymbol{\mu}}_t$, we can customize the form of $\boldsymbol{\mu}_\theta$ as a combination of $\mathbf{x}_t$ and $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$:

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\underbrace{\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}_{\text{a network}}\right). \tag{2.6}$$

Remember that $\boldsymbol{\mu}_\theta$ stands for the conditional mean of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ in the **reverse process**. To sample $\mathbf{x}_{t-1} \sim p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is to compute $\mathbf{x}_{t-1} = \boldsymbol{\mu}_\theta(\mathbf{x}_t, t) + \sigma_t\mathbf{z} = \frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\Big) + \sigma_t\mathbf{z}$, where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

With the expression of $\tilde{\boldsymbol{\mu}}_t$ and $\boldsymbol{\mu}_\theta$ in (2.5) and (2.6), (2.3) becomes

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0)}\left\{\mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})}\left[\frac{1}{2\sigma_t^2}\left\|\frac{1}{\sqrt{\alpha_t}}\Big(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\Big) - \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right)\right\|^2\right]\right\},$$

$$\implies L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0), \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta \left(\mathbf{x}_t, t \right) \|^2 \right]. \tag{2.7}$$

Lastly, we should substitute $\mathbf{x}_t$ by $\mathbf{x}_0$ and $\boldsymbol{\epsilon}$. By (2.5), $\mathbf{x}_t = \mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. Applying this to (2.7), we get:

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0), \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\frac{\beta_t^2}{2\sigma_t^2 \alpha_t (1-\bar{\alpha}_t)} \| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta \left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t \right) \|^2 \right]. \tag{2.8}$$

which resembles denoising score matching over multiple noise scales indexed by t.

As (2.8) is equal to (one term of) the variational bound for the Langevin-like reverse process (2.6), we see that optimizing an objective resembling denoising score matching is equivalent to using variational inference to fit the finite-time marginal of a sampling chain resembling Langevin dynamics.

To summarize, we can train the reverse process mean function approximator $\boldsymbol{\mu}_\theta$ to predict $\tilde{\boldsymbol{\mu}}_t$, or by modifying its parameterization, we can train it to predict $\boldsymbol{\epsilon}$. We have shown that the $\boldsymbol{\epsilon}$-prediction parameterization both resembles Langevin dynamics and simplifies the diffusion model's variational bound to an objective that resembles denoising score matching.

2.3 Reverse Process Decoder

Remember that the objective is to minimize L:

$$L = \mathbb{E}_q \Bigg[\underbrace{D_{\mathrm{KL}}\Big(q(\mathbf{x}_T|\mathbf{x}_0) \| p(\mathbf{x}_T) \Big)}_{L_T} + \underbrace{\Big(-\log p_\theta(\mathbf{x}_0|\mathbf{x}_1) \Big)}_{L_0} + \sum_{t=2}^{T} \underbrace{D_{\mathrm{KL}}\Big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \Big)}_{L_{t-1}} \Bigg].$$

In Forward Process section, we demonstrated that L_T is a constant under our assumption, so we can ignore it. In Reverse Process section, we give the expression of L_{t-1}. The remaining part is L_0.

We parameterize last decoding step as a Gaussian:

$$\begin{aligned} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I}), \\ \implies p_\theta(\mathbf{x}_0|\mathbf{x}_1) &= \mathcal{N}(\mathbf{x}_0; \boldsymbol{\mu}_\theta(\mathbf{x}_1, 1), \sigma_0^2 \mathbf{I}). \end{aligned} \tag{2.9}$$

So L_0 can be calculated.

2.4 Simplified Training Objective

With the reverse process and decoder defined above, the variational bound, consisting of terms derived from (2.8) and (2.9), is clearly differentiable with respect to θ and is ready to be employed for training.

However, we found it beneficial to sample quality (and simpler to implement) to train on the following variant of the variational bound

$$L_{\text{simple}}(\theta) := \mathbb{E}_{t, \mathbf{x}_0 \sim q(\mathbf{x}_0), \boldsymbol{\epsilon} \in \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta \left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t \right) \|^2 \right], \tag{2.10}$$

where t is uniform between 1 and T.

Since our simplified objective (2.10) discards the weighting in (2.8), it is a weighted variational bound that emphasizes different aspects of reconstruction compared to the standard variational bound.

References

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/4c5bcfec8584af0d967f1ab10179ca4b-Paper.pdf.

Stanley H. Chan. Tutorial on diffusion models for imaging and vision, 2025. URL https://arxiv.org/abs/2403.18103.

理工科的MBA. 论文denoising diffusion probabilistic models笔记, 2022. URL https://zhuanlan.zhihu.com/p/583032549.

苏剑林. 生成扩散模型漫谈（一）：ddpm = 拆楼 + 建楼, 2022a. URL https://spaces.ac.cn/archives/9119.

苏剑林. 生成扩散模型漫谈（三）：ddpm = 贝叶斯 + 去噪, 2022b. URL https://spaces.ac.cn/archives/9164.

Jia-Bin Huang. How i understand diffusion models, 2024. URL https://www.youtube.com/watch?v=i2qSxMVeVLI.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 2256–2265. PMLR, 2015. URL https://proceedings.mlr.press/v37/sohl-dickstein15.html.

3 Appendix

3.1 Derivation of Equation 1.6

We model the forward process as a Gaussian distribution, so the conditional distribution of $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$ and $\mathbf{x}_0$ is also a Gaussian:

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

Since the forward process is determined by the noise scales, $\beta_1, \beta_2, \cdots, \beta_T$, and that the β's are fixed, the explicit expression of $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ should be able to be derived. The solution is shown as follows.

By Bayes' rule, we have

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

where

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

$$q(\mathbf{x}_{t-1}|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0, (1-\bar{\alpha}_{t-1})\mathbf{I}).$$

Note that the distributions are all dimension-wise independent, so we can break the PDF of $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)$ into the product of independent Gaussians.

In each dimension, using x_0, x_{t-1}, x_t to denote the corresponding component of $\mathbf{x}_0, \mathbf{x}_{t-1}, \mathbf{x}_t$, we have

$$\begin{aligned}
q(x_t|x_{t-1}, x_0) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t^2) &= \frac{1}{\sqrt{2\pi\beta_t}} \exp\Big[-\frac{(x_t - \sqrt{1-\beta_t}x_{t-1})^2}{2\beta_t}\Big]), \\
q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)^2) &= \frac{1}{\sqrt{2\pi(1-\bar{\alpha}_t)}} \exp\Big[-\frac{(x_t - \sqrt{\bar{\alpha}_t}x_0)^2}{2(1-\bar{\alpha}_t)}\Big], \\
q(x_{t-1}|x_0) = \mathcal{N}(x_{t-1}; \sqrt{\bar{\alpha}_{t-1}}x_0, (1-\bar{\alpha}_{t-1})^2) &= \frac{1}{\sqrt{2\pi(1-\bar{\alpha}_{t-1})}} \exp\Big[-\frac{(x_{t-1} - \sqrt{\bar{\alpha}_{t-1}}x_0)^2}{2(1-\bar{\alpha}_{t-1})}\Big].
\end{aligned}$$

Applying Bayes' rule, we have

$$\begin{aligned}
&q(x_{t-1}|x_t, x_0) = \frac{q(x_t|x_{t-1}, x_0)q(x_{t-1}|x_0)}{q(x_t|x_0)} \\
&= \frac{\sqrt{2\pi(1-\bar{\alpha}_t)}}{\sqrt{2\pi\beta_t}\sqrt{2\pi(1-\bar{\alpha}_{t-1})}} \exp\Bigg[-\frac{1}{2}\Bigg(\frac{(x_t - \sqrt{\alpha_t}x_{t-1})^2}{\beta_t} + \frac{(x_t - \sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t - \sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}\Bigg)\Bigg] \\
&= \frac{1}{\sqrt{2\pi\beta_t\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}}} \exp\Bigg[-\frac{1}{2}\Bigg(\frac{x_t^2 - 2\sqrt{\alpha_t}x_t x_{t-1} + \alpha_t x_{t-1}^2}{\beta_t} \\
&\qquad\qquad + \frac{x_t^2 - \sqrt{\bar{\alpha}_{t-1}}x_0 x_t + \bar{\alpha}_{t-1}x_0^2}{1-\bar{\alpha}_{t-1}} - \frac{x_t^2 - \sqrt{\bar{\alpha}_t}x_0 x_t + \bar{\alpha}_t x_0^2}{1-\bar{\alpha}_t}\Bigg)\Bigg] \\
&= \frac{1}{\sqrt{2\pi\beta_t\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}}} \exp\Bigg\{-\frac{1}{2}\Bigg[\Big(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}\Big)x_{t-1}^2 - 2\Big(\frac{\sqrt{\alpha_t}x_t}{\beta_t} + \frac{\sqrt{\bar{\alpha}_{t-1}}x_0}{1-\bar{\alpha}_{t-1}}\Big)x_{t-1} + C\Bigg]\Bigg\}.
\end{aligned}$$

So $q(x_{t-1}|x_t, x_0)$ is a Gaussian with variance

$$\begin{aligned}
\sigma &= 1/\Big(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}\Big) \\
&= 1/\Bigg[\frac{\alpha_t(1-\bar{\alpha}_{t-1}) + \beta_t}{\beta_t(1-\bar{\alpha}_{t-1})}\Bigg] \\
&= 1/\Bigg[\frac{\alpha_t - \bar{\alpha}_t + 1 - \alpha_t}{\beta_t(1-\bar{\alpha}_{t-1})}\Bigg] \\
&= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t
\end{aligned}$$

and mean

$$\begin{aligned}
\mu &= \Big(\frac{\sqrt{\alpha_t}x_t}{\beta_t} + \frac{\sqrt{\bar{\alpha}_{t-1}}x_0}{1-\bar{\alpha}_{t-1}}\Big)/\Big(\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}\Big) \\
&= \Big(\frac{\sqrt{\alpha_t}x_t}{\beta_t} + \frac{\sqrt{\bar{\alpha}_{t-1}}x_0}{1-\bar{\alpha}_{t-1}}\Big)\sigma \\
&= \Big(\frac{\sqrt{\alpha_t}x_t}{\beta_t} + \frac{\sqrt{\bar{\alpha}_{t-1}}x_0}{1-\bar{\alpha}_{t-1}}\Big)\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t \\
&= \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t.
\end{aligned}$$

Based on the component form, we can write the full vector form of $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ as

$$\begin{aligned} q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}) \\ \text{where} \quad \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) &:= \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t \\ \text{and} \quad \tilde{\beta}_t &:= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t. \end{aligned} \tag{1.6}$$

3.2 Derivation of Equation 1.4

In the diffusion process, we have defined the conditional distribution of $\mathbf{x}_t$ given $\mathbf{x}_{t-1}$:

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

We can reparameterize this as

$$\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I}).$$

Using the notation $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{i=1}^{t} \alpha_i$, we have

$$\begin{aligned} \mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\boldsymbol{\epsilon}_t \\ &= \sqrt{\alpha_t}\Big(\sqrt{\alpha_{t-1}}\mathbf{x}_{t-2} + \beta_{t-1}\boldsymbol{\epsilon}_{t-1}\Big) + \sqrt{\beta_t}\boldsymbol{\epsilon}_t \\ &= \Big(\prod_{i=1}^{t}\sqrt{\alpha_i}\Big)\mathbf{x}_0 + \underbrace{\sqrt{\beta_t}\boldsymbol{\epsilon}_t + \sqrt{\alpha_t}\sqrt{\beta_{t-1}}\boldsymbol{\epsilon}_{t-1} + \cdots + \Big(\prod_{i=3}^{t}\sqrt{\alpha_i}\Big)\sqrt{\beta_2}\boldsymbol{\epsilon}_2 + \Big(\prod_{i=2}^{t}\sqrt{\alpha_i}\Big)\sqrt{\beta_1}\boldsymbol{\epsilon}_1}_{\text{sum of multiple independent Gaussians}} \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \mathcal{N}\Bigg(\mathbf{0}, \Big[\beta_t + \beta_{t-1}\alpha_t + \beta_{t-2}\alpha_{t-1}\alpha_t \cdots + \beta_1(\alpha_2\alpha_3\alpha_4\cdots\alpha_t)\Big]\mathbf{I}\Bigg) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \mathcal{N}\Bigg(\mathbf{0}, \Big[1-\alpha_t + (1-\alpha_{t-1})\alpha_t + (1-\alpha_{t-2})\alpha_{t-1}\alpha_t \cdots + (1-\alpha_1)(\alpha_2\alpha_3\alpha_4\cdots\alpha_t)\Big]\mathbf{I}\Bigg) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \mathcal{N}\Bigg(\mathbf{0}, \Big[1-\alpha_t + \alpha_t - \alpha_{t-1}\alpha_t + \alpha_{t-1}\alpha_t - \alpha_{t-2}\alpha_{t-1}\alpha_t \cdots - (\alpha_1\alpha_2\alpha_3\alpha_4\cdots\alpha_t)\Big]\mathbf{I}\Bigg) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \mathcal{N}\Big(\mathbf{0}, \Big[1-(\alpha_1\alpha_2\alpha_3\alpha_4\cdots\alpha_t)\Big]\mathbf{I}\Big) \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \mathcal{N}\Big(\mathbf{0}, (1-\bar{\alpha}_t)\mathbf{I}\Big). \end{aligned}$$

So we have derived (1.4):

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}). \tag{1.4}$$

By (1.4), we can directly sample $\mathbf{x}_t$ given $\mathbf{x}_0$ in a single step.

3.3 Derivation of Equation 2.1

Proposition 3.1 (Kullback-Leibler Divergence of Two Multidimensional Gaussian Distributions). Given two normal distributions, $p = \mathcal{N}(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)$, $q = \mathcal{N}(\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)$, their KL divergence is

$$D_{\mathrm{KL}}(p\|q) = \frac{1}{2}\Big[(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \boldsymbol{\Sigma}_2^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) - \log\det(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1) + \mathrm{Tr}\left(\boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1\right) - n\Big].$$

In (2.1)'s case, since $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$, we have:

$$\begin{aligned} L_{t-1} &= D_{\mathrm{KL}}\Big(\mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \sigma_t^2\mathbf{I}) \| \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})\Big) \\ &= \frac{1}{2}\Big[\frac{\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2}{\sigma_t^2} - 0 + d - d\Big] = \frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2. \end{aligned}$$

3.4 Supplementary Method of Reverse Process

In the derivation of reverse process, we chose to model $\boldsymbol{\mu}_t$ as $\boldsymbol{\mu}_t(\mathbf{x}_t, \boldsymbol{\epsilon})$, and eliminate $\mathbf{x}_0$. We can choose another way: we can model $\boldsymbol{\mu}_t$ as $\boldsymbol{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$, and eliminate $\boldsymbol{\epsilon}$. This is equivalent to the previous derivation, but with a different notation.

Recall that the optimization goal is to minimize

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0)}\left\{\mathbb{E}_{\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)}\left[\frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t, t)\|^2\right]\right\}. \tag{2.1}$$

By (1.6), we know

$$\tilde{\boldsymbol{\mu}}_t\Big(\mathbf{x}_t, \mathbf{x}_0\Big) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0. \tag{1.6}$$

Our goal is to train a model $\boldsymbol{\mu}_\theta$ to minimize L_{t-1}. The form of $\boldsymbol{\mu}_\theta$ can be customized, so we choose one that is close to (1.6):

$$\underbrace{\tilde{\boldsymbol{\mu}}_\theta\Big(\mathbf{x}_t, t\Big)}_{\text{a network}} = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\underbrace{\bar{\mathbf{x}}_\theta(\mathbf{x}_t)}_{\text{another network}}.$$

$\bar{\mathbf{x}}_\theta(\mathbf{x}_t)$ is a neural network that predicts $\mathbf{x}_0$ given $\mathbf{x}_t$. Applying the above formula and (1.6) to (2.1), we obtain:

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0)}\left\{\mathbb{E}_{\mathbf{x}_t \sim q(\mathbf{x}_t|\mathbf{x}_0)}\left[\frac{1}{2\sigma_t^2}\frac{\bar{\alpha}_{t-1}\beta_t^2}{(1-\bar{\alpha}_t)^2}\Big\|\mathbf{x}_0 - \bar{\mathbf{x}}_\theta(\mathbf{x}_t)\Big\|^2\right]\right\}.$$

We can minimize the objective by training $\bar{\mathbf{x}}_\theta(\mathbf{x}_t)$.

Summary. When we model $\boldsymbol{\mu}_t$ as $\boldsymbol{\mu}_t(\mathbf{x}_t, \boldsymbol{\epsilon})$, $\tilde{\boldsymbol{\mu}}_\theta$ is modeled as

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

Note. We can also model $\boldsymbol{\mu}_t$ as $\boldsymbol{\mu}_t(\mathbf{x}_t, \mathbf{x}_0)$, and we will obtain another form:

$$\tilde{\boldsymbol{\mu}}_\theta\Big(\mathbf{x}_t, t\Big) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\bar{\mathbf{x}}_\theta(\mathbf{x}_t).$$

Created on 20250228;
Finished on 20250307;
Proofread to LaTeX on 20250916;
Proofread on 20250926;
Proofread on 20250928.