

Diffusion Model Training

Sai Sri Teja Kuppa

Immerso.ai

December 23, 2025

Abstract

Contents

Abstract	i
List of Figures	iii
List of Tables	iv
1 Introduction	1
2 Sampling Strategies	2
2.1 DDPM Sampler	2
2.1.1 The transition distribution is a Gaussian	3
2.1.2 Forward Step in Diffusion Process	4
2.1.3 Reverse Step in Diffusion Process	5
2.1.4 Training and Inference	15
2.1.5 Prediction of Noise	17
2.1.6 Probability Distributions in DDIM	18
3 Methodology	21
4 Experiments	22
5 Conclusion	23
A Appendix	24

List of Figures

2.1 Basic blocks architecture	2
---	---

List of Tables

Chapter 1

Introduction

We are going forward with the diffusion model from scratch. we are using the repo pixart sigma, this is due to release of base trained model without distillation.

Chapter 2

Sampling Strategies

This chapter presents an overview of various sampling strategies used in diffusion models. The following is a comprehensive list of sampling strategies that will be discussed:

1. **DDIM** (Denoising Diffusion Implicit Models)
2. **DDPM** (Denoising Diffusion Probabilistic Models)
3. **iDDPM** (Improved Denoising Diffusion Probabilistic Models)
4. **Euler Sampler** and variants

2.1 DDPM Sampler

the posterior will remain a Gaussian if the likelihood and the prior are both Gaussians. if each transitional distribution above is a Gaussian, the joint distribution is also a Gaussian. transition distributions are only dependent on its immediate previous stage

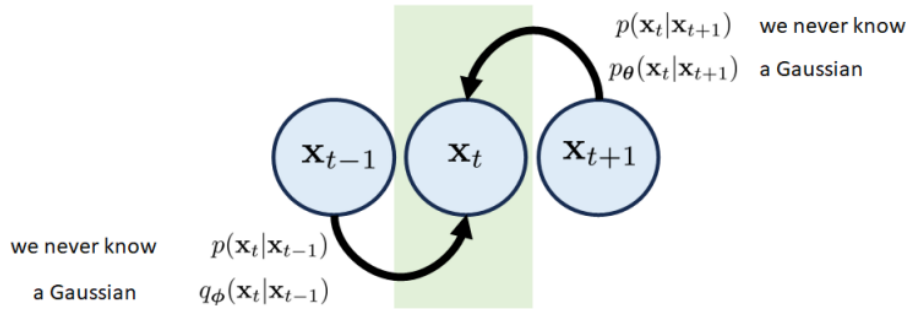


Figure 2.1: Basic blocks architecture

The transition distribution $q_\phi(x_t | x_{t-1})$ is defined as

$$q_\phi(x_t | x_{t-1}) \stackrel{\text{def}}{=} \mathcal{N}(x_t | \sqrt{\alpha_t} x_{t-1}, (1 - \alpha_t) \mathbf{I}). \quad (2.1)$$

In other words, $q_\phi(x_t | x_{t-1})$ is a Gaussian distribution. The mean is $\sqrt{\alpha_t} x_{t-1}$ and the variance is $1 - \alpha_t$. The choice of the scaling factor $\sqrt{\alpha_t}$ ensures that the magnitude of the variance is preserved throughout the process, preventing it from exploding or vanishing after many iterations.

2.1.1 The transition distribution is a Gaussian

Consider the following recursive formulation for the forward process:

$$x_t = ax_{t-1} + b\epsilon_{t-1}, \quad \epsilon_{t-1} \sim \mathcal{N}(0, \mathbf{I}) \quad (2.2)$$

By recursively substituting x_{t-1}, x_{t-2}, \dots , we obtain:

$$\begin{aligned} x_t &= ax_{t-1} + b\epsilon_{t-1} \\ &= a(ax_{t-2} + b\epsilon_{t-2}) + b\epsilon_{t-1} \\ &= a^2x_{t-2} + ab\epsilon_{t-2} + b\epsilon_{t-1} \\ &\vdots \\ &= a^t x_0 + b \sum_{i=0}^{t-1} a^i \epsilon_{t-1-i} \end{aligned}$$

Define:

$$w_t = \sum_{i=0}^{t-1} a^i \epsilon_{t-1-i}$$

Since each ϵ_i is standard Gaussian and independent, w_t is also a zero-mean Gaussian random variable. The covariance of w_t is given by:

$$\begin{aligned} \text{Cov}[w_t] &= \mathbb{E}[w_t w_t^T] \\ &= \sum_{i=0}^{t-1} (a^i)^2 \text{Cov}(\epsilon_{t-1-i}) \\ &= \sum_{i=0}^{t-1} a^{2i} \mathbf{I} \\ &= \frac{1 - a^{2t}}{1 - a^2} \mathbf{I} \end{aligned}$$

Thus,

$$\text{Cov}[bw_t] = b^2 \frac{1 - a^{2t}}{1 - a^2} \mathbf{I}$$

As $t \rightarrow \infty$, and for $0 < a < 1$, we have $a^{2t} \rightarrow 0$, so

$$\lim_{t \rightarrow \infty} \text{Cov}[bw_t] = \frac{b^2}{1 - a^2} \mathbf{I}$$

If we desire $\lim_{t \rightarrow \infty} \text{Cov}[x_t] = \mathbf{I}$, then we set:

$$\frac{b^2}{1-a^2} = 1 \quad \implies \quad b = \sqrt{1-a^2}$$

Letting $a = \sqrt{\alpha}$ and consequently $b = \sqrt{1-\alpha}$, the update rule becomes:

$$x_t = \sqrt{\alpha} x_{t-1} + \sqrt{1-\alpha} \epsilon_{t-1} \quad (2.3)$$

2.1.2 Forward Step in Diffusion Process

The goal is to somehow get the x_t in a closed form. to do this, we will use the recursion formula and the fact that the noise is Gaussian.

$$\begin{aligned} x_t &= \sqrt{\alpha_t} x_{t-1} + \sqrt{1-\alpha_t} \epsilon_{t-1} \\ &= \sqrt{\alpha_t} \left(\sqrt{\alpha_{t-1}} x_{t-2} + \sqrt{1-\alpha_{t-1}} \epsilon_{t-2} \right) + \sqrt{1-\alpha_t} \epsilon_{t-1} \\ &= \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{\alpha_t (1-\alpha_{t-1})} \epsilon_{t-2} + \sqrt{1-\alpha_t} \epsilon_{t-1}. \end{aligned}$$

We notice that x_t is a sum of independent Gaussian terms (as each ϵ_i is independent), so x_t is also Gaussian. The new covariance for the noise part $w_1 = \sqrt{\alpha_t (1-\alpha_{t-1})} \epsilon_{t-2} + \sqrt{1-\alpha_t} \epsilon_{t-1}$ is:

$$\begin{aligned} \mathbb{E}[w_1 w_1^T] &= [\alpha_t (1-\alpha_{t-1}) + (1-\alpha_t)] \mathbf{I} \\ &= [1-\alpha_t \alpha_{t-1}] \mathbf{I}. \end{aligned}$$

Continuing this recursion for a few more steps, we get:

$$\begin{aligned} x_t &= \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1-\alpha_t \alpha_{t-1}} \epsilon_{t-2} \\ &= \sqrt{\alpha_t \alpha_{t-1} \alpha_{t-2}} x_{t-3} + \sqrt{1-\alpha_t \alpha_{t-1} \alpha_{t-2}} \epsilon_{t-3} \\ &\vdots \\ &= \sqrt{\prod_{i=1}^t \alpha_i} x_0 + \sqrt{1-\prod_{i=1}^t \alpha_i} \epsilon_0. \end{aligned}$$

Define

$$\bar{\alpha}_t = \prod_{i=1}^t \alpha_i$$

so the above can be written as

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t} \epsilon_0, \quad (2.4)$$

where $\epsilon_0 \sim \mathcal{N}(0, \mathbf{I})$.

In other words, the marginal distribution $q_\phi(x_t | x_0)$ is,

$$x_t \sim q_\phi(x_t | x_0) = \mathcal{N}(x_t | \sqrt{\bar{\alpha}_t}x_0, (1 - \bar{\alpha}_t)\mathbf{I}).$$

2.1.3 Reverse Step in Diffusion Process

Reverse process has much more math but lets take it down step by step.

Step 1: Rewrite the Likelihood Using Latent Variables

We want to compute the log-likelihood of data:

$$\log p(x) = \log p(x_0)$$

but $p(x_0)$ is intractable in diffusion models because x_0 is generated through many hidden diffusion steps.

Trick: Introduce latent diffusion states. Define intermediate latent variables:

$$x_{1:T} = \{x_1, x_2, \dots, x_T\}$$

which are noise-corrupted versions of x_0 (the forward diffusion chain). Collectively:

$$x_{0:T} = \{x_0, x_1, \dots, x_T\}.$$

Why this helps. Instead of treating x_0 as appearing “from nowhere”, we view it as part of a joint trajectory:

$$p(x_{0:T})$$

and obtain the marginal:

$$p(x_0) = \int p(x_{0:T}) dx_{1:T}.$$

Taking logs:

$$\log p(x_0) = \log \int p(x_{0:T}) dx_{1:T}.$$

Key idea. Making the hidden steps explicit allows us to write the exact likelihood of x_0 as a marginal over the diffusion trajectory.

Step 2: Insert the Variational Distribution

- Multiply and divide by $q_\phi(x_{1:T} \mid x_0)$.
- This is the standard *variational inference trick*.
- Allows us to express the likelihood as an expectation:

$$\log p(x_0) = \log \mathbb{E}_{q_\phi} \left[\frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} \right]$$

We multiply and divide by the same distribution $q_\phi(x_{1:T} \mid x_0)$:

$$\begin{aligned} \log p(x_0) &= \log \int p(x_{0:T}) dx_{1:T} \\ &= \log \int q_\phi(x_{1:T} \mid x_0) \frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} dx_{1:T} \end{aligned}$$

This operation does not change the value—it is simply multiplying by 1—but it allows us to rewrite the integral as an expectation:

$$\log p(x_0) = \log \mathbb{E}_{q_\phi(x_{1:T} \mid x_0)} \left[\frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} \right]$$

Step 3: Apply Jensen’s Inequality

- Use concavity of the log.
- Move the log *inside* the expectation.
- This creates the *Evidence Lower Bound (ELBO)*:

$$\log p(x_0) \geq \mathbb{E}_{q_\phi} \left[\log \frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} \right]$$

- From here on, everything is about **rewriting and decomposing this ELBO**.

Now, we use Jensen’s inequality, which states that for any random variable X and any concave function f , it holds that $f(\mathbb{E}[X]) \geq \mathbb{E}[f(X)]$. By recognizing that $f(\cdot) = \log(\cdot)$, we can show that:

$$\log p(x) = \log \mathbb{E}_{q_\phi(x_{1:T} \mid x_0)} \left[\frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} \right] \geq \mathbb{E}_{q_\phi(x_{1:T} \mid x_0)} \left[\log \frac{p(x_{0:T})}{q_\phi(x_{1:T} \mid x_0)} \right].$$

Step 4: Factorize the True Joint Distribution $p(x_{0:T})$

- Use the *reverse Markov structure*.
- Factor it into:
 - a prior at time T
 - reverse transitions
 - a reconstruction term
- This aligns the math with the *reverse diffusion process*.

We want to decouple $p(x_{0:T})$ by conditioning on x_t for x_{t-1} . This leads to:

$$p(x_{0:T}) = p(x_T) \prod_{t=1}^T p(x_{t-1} \mid x_t) = p(x_T) p(x_0 \mid x_1) \prod_{t=2}^T p(x_{t-1} \mid x_t).$$

Step 5: Factorize the Variational Distribution $q_\phi(x_{1:T} \mid x_0)$

- Use the *forward Markov structure*.
- Factor it into forward transitions.
- This aligns with the *forward diffusion process*.

For $q_\phi(x_{1:T} \mid x_0)$, we condition on x_t given x_{t-1} . Due to the sequential structure, we can write:

$$q_\phi(x_{1:T} \mid x_0) = \prod_{t=1}^T q_\phi(x_t \mid x_{t-1}) = q_\phi(x_T \mid x_{T-1}) \prod_{t=1}^{T-1} q_\phi(x_t \mid x_{t-1}).$$

Step 6: Plug Both Factorizations into the ELBO

- Take the ratio $\frac{p}{q}$.
- The log turns products into sums.
- Expectation distributes over time.
- This is where the ELBO starts breaking into *interpretable blocks*

Use the step 4 and step 5 output in step 3.

$$\begin{aligned}
\log p(x) &\geq \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_{0:T})}{q_\phi(x_{1:T} | x_0)} \right] \\
&= \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1) \prod_{t=2}^T p(x_{t-1} | x_t)}{q_\phi(x_T | x_{T-1}) \prod_{t=1}^{T-1} q_\phi(x_t | x_{t-1})} \right] \\
&= \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1) \prod_{t=1}^{T-1} p(x_t | x_{t+1})}{q_\phi(x_T | x_{T-1}) \prod_{t=1}^{T-1} q_\phi(x_t | x_{t-1})} \right] \quad (\text{shift } t \rightarrow t+1) \\
&= \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1)}{q_\phi(x_T | x_{T-1})} \right] + \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \prod_{t=1}^{T-1} \frac{p(x_t | x_{t+1})}{q_\phi(x_t | x_{t-1})} \right] \quad (\text{split expectation})
\end{aligned}$$

Now, we can decompose the first term as follows:

$$\mathbb{E}_{q_\phi(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T) p(\mathbf{x}_0 | \mathbf{x}_1)}{q_\phi(\mathbf{x}_T | \mathbf{x}_{T-1})} \right] = \underbrace{\mathbb{E}_{q_\phi(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log p(\mathbf{x}_0 | \mathbf{x}_1) \right]}_{\text{Reconstruction}} + \underbrace{\mathbb{E}_{q_\phi(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p(\mathbf{x}_T)}{q_\phi(\mathbf{x}_T | \mathbf{x}_{T-1})} \right]}_{\text{Prior Matching}}. \quad (2.5)$$

From this decomposition, we have reconstruction and prior matching terms.

Let us further analyze how the decomposition of the Evidence Lower Bound (ELBO) naturally splits into reconstruction and prior-matching terms.

The Reconstruction Term. The reconstruction term only depends on x_0 and x_1 . We can write:

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} [\log p(x_0 | x_1)] = \mathbb{E}_{q_\phi(x_1|x_0)} [\log p(x_0 | x_1)]$$

This simplification occurs because of the Markov structure of q_ϕ ; for a function that only involves x_0 and x_1 , conditioning on (x_2, \dots, x_T) is unnecessary. Thus, the reconstruction term is essentially a reconstruction log-likelihood for the first denoising step.

The Prior Matching Term. For the prior matching term, we focus on the last step of the chain:

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T)}{q_\phi(x_T | x_{T-1})} \right]$$

This term only depends on x_{T-1} and x_T . Therefore, we can marginalize everything else and write:

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} [\cdot] = \mathbb{E}_{q_\phi(x_{T-1}, x_T|x_0)} [\cdot]$$

By applying the chain rule, we factor the joint as $q_\phi(x_T, x_{T-1} \mid x_0) = q_\phi(x_T \mid x_{T-1}, x_0) q_\phi(x_{T-1} \mid x_0)$. Due to the Markov nature of q_ϕ , $q_\phi(x_T \mid x_{T-1}, x_0) = q_\phi(x_T \mid x_{T-1})$. This allows us to further factor the expectation as:

$$\mathbb{E}_{q_\phi(x_{T-1}, x_T \mid x_0)}[\cdot] = \mathbb{E}_{q_\phi(x_{T-1} \mid x_0)} \left[\mathbb{E}_{q_\phi(x_T \mid x_{T-1})}[\cdot] \right]$$

Recognizing the KL Divergence. Now, the prior matching part becomes:

$$\begin{aligned} \mathbb{E}_{q_\phi(x_{T-1} \mid x_0)} \left[\mathbb{E}_{q_\phi(x_T \mid x_{T-1})} \left[\log \frac{p(x_T)}{q_\phi(x_T \mid x_{T-1})} \right] \right] \\ = -\mathbb{E}_{q_\phi(x_{T-1} \mid x_0)} [D_{\text{KL}}(q_\phi(x_T \mid x_{T-1}) \parallel p(x_T))] \end{aligned}$$

That is, this term represents a KL divergence between the distribution $q_\phi(x_T \mid x_{T-1})$ produced by the forward process's last step and the target prior $p(x_T)$.

Summary. To summarize, we have:

1. Reconstruction:

$$\mathbb{E}_{q_\phi(x_1 \mid x_0)} [\log p(x_0 \mid x_1)]$$

which depends only on the first reverse denoising step ($x_1 \rightarrow x_0$).

2. Prior Matching:

$$-\mathbb{E}_{q_\phi(x_{T-1} \mid x_0)} [D_{\text{KL}}(q_\phi(x_T \mid x_{T-1}) \parallel p(x_T))]$$

which depends only on the last forward step ($x_{T-1} \rightarrow x_T$) and encourages the final latent to match the prior.

This decomposition is standard in DDPMs: the ELBO is written as a sum over reconstruction terms (for all timesteps) and KL divergence terms for prior matching. This structure also makes the training procedure straightforward to implement.

The Final Step of ELBO Decomposition

The final remaining term in the ELBO for diffusion models is:

$$\mathbb{E}_{q_\phi(x_{1:T} \mid x_0)} \left[\log \prod_{t=1}^{T-1} \frac{p(x_t \mid x_{t+1})}{q_\phi(x_t \mid x_{t-1})} \right]$$

By the property that $\log \prod = \sum \log$, this can be rewritten as a sum over timesteps:

$$\sum_{t=1}^{T-1} \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_t|x_{t+1})}{q_\phi(x_t|x_{t-1})} \right]$$

More explicitly,

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \prod_{t=1}^{T-1} \frac{p(x_t|x_{t+1})}{q_\phi(x_t|x_{t-1})} \right] = \sum_{t=1}^{T-1} \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_t|x_{t+1})}{q_\phi(x_t|x_{t-1})} \right]$$

Reducing the Expectation Note that for a fixed t , the term

$$\log \frac{p(x_t|x_{t+1})}{q_\phi(x_t|x_{t-1})}$$

depends only on x_{t-1} , x_t , and x_{t+1} . Thus, the expectation over all $x_{1:T}$ can be reduced:

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)}[\cdot] = \mathbb{E}_{q_\phi(x_{t-1}, x_t, x_{t+1}|x_0)}[\cdot]$$

This reduction uses the Markov property of the forward process.

Conditional Independence Factorization By conditional independence, we can factor:

$$q_\phi(x_{t-1}, x_t, x_{t+1}|x_0) = q_\phi(x_{t-1}, x_{t+1}|x_0) q_\phi(x_t|x_{t-1}, x_{t+1}, x_0)$$

But the Markov property gives us:

$$q_\phi(x_t|x_{t-1}, x_{t+1}, x_0) = q_\phi(x_t|x_{t-1})$$

Therefore, the expectation factorizes as:

$$\mathbb{E}_{q_\phi(x_{t-1}, x_t, x_{t+1}|x_0)}[\cdot] = \mathbb{E}_{q_\phi(x_{t-1}, x_{t+1}|x_0)} \mathbb{E}_{q_\phi(x_t|x_{t-1})}[\cdot]$$

Recognizing KL Divergence The inner expectation is precisely a Kullback–Leibler divergence between the forward and reverse transitions:

$$\mathbb{E}_{q_\phi(x_t|x_{t-1})} \left[\log \frac{p(x_t|x_{t+1})}{q_\phi(x_t|x_{t-1})} \right] = -D_{\text{KL}}(q_\phi(x_t|x_{t-1}) \parallel p(x_t|x_{t+1}))$$

Hence, the total sum over timesteps becomes:

$$- \sum_{t=1}^{T-1} \mathbb{E}_{q_{\phi}(x_{t-1}, x_{t+1} | x_0)} [D_{\text{KL}}(q_{\phi}(x_t | x_{t-1}) \parallel p(x_t | x_{t+1}))]$$

Parameterization in Practice In practice, the reverse conditionals are learned and parameterized by a neural network with parameters θ :

$$\begin{aligned} p(x_0 | x_1) &\rightarrow p_{\theta}(x_0 | x_1) \\ p(x_t | x_{t+1}) &\rightarrow p_{\theta}(x_t | x_{t+1}) \end{aligned}$$

where θ are the weights of the learned reverse diffusion model.

Step 7: Identify the Three Conceptual Terms

From the algebra, three kinds of terms emerge:

1. **Reconstruction term**

Measures how well x_1 reconstructs x_0 .

2. **Prior matching term**

Ensures the final latent x_T matches the Gaussian prior.

3. **Consistency (transition) terms**

Enforces agreement between forward diffusion transitions and learned reverse transitions.

Step 8: Identify a Practical Problem

- The consistency term requires sampling *both past and future*, specifically (x_{t-1}, x_{t+1}) .
- This is awkward and inefficient.
- This is the *core motivation* for rewriting the ELBO.

The challenge in the above variational diffusion model is that we need to draw samples (x_{t-1}, x_{t+1}) from a joint distribution $q_{\phi}(x_{t-1}, x_{t+1} \mid x_0)$. While we can assume it is Gaussian, we still need to use future samples x_{t+1} to draw the current sample x_t , which is not straightforward.

Inspecting the consistency term, we notice that $q_{\phi}(x_t \mid x_{t-1})$ and $p_{\theta}(x_t \mid x_{t+1})$ move along two opposite directions. Thus, it appears unavoidable to use both x_{t-1} and

x_{t+1} . The question is: can we design a method to avoid handling two opposite directions while still checking consistency?

A simple solution is to use Bayes' theorem:

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

Conditioning on x_0 gives:

$$q(x_t | x_{t-1}, x_0) = \frac{q(x_{t-1} | x_t, x_0) q(x_t | x_0)}{q(x_{t-1} | x_0)}.$$

With this change in conditioning, we can switch $q(x_t | x_{t-1}, x_0)$ to $q(x_{t-1} | x_t, x_0)$ by including the additional condition variable x_0 . The direction $q(x_{t-1} | x_t, x_0)$ is now aligned with $p_\theta(x_{t-1} | x_t)$. Therefore, a natural option for rewriting the consistency term is to calculate the KL divergence between $q_\phi(x_{t-1} | x_t, x_0)$ and $p_\theta(x_{t-1} | x_t)$.

Step 9: Rewrite the ELBO into a Cleaner Form

- Prior matching becomes simpler.
- Consistency becomes the KL divergence between two reverse-time distributions.
- Reconstruction stays the same.

We start from the marginal likelihood of the data x_0 :

$$\log p(x_0) = \log \int p(x_{0:T}) dx_{1:T}. \quad (2.6)$$

Introducing a variational posterior $q_\phi(x_{1:T} | x_0)$ and applying Jensen's inequality gives the ELBO:

$$\log p(x_0) \geq \mathbb{E}_{q_\phi(x_{1:T} | x_0)} \left[\log \frac{p(x_{0:T})}{q_\phi(x_{1:T} | x_0)} \right]. \quad (2.7)$$

Factorization of joint distributions. The reverse diffusion model factorizes as

$$p(x_{0:T}) = p(x_T) p(x_0 | x_1) \prod_{t=2}^T p(x_{t-1} | x_t), \quad (2.8)$$

and the forward process factorizes as

$$q_\phi(x_{1:T} | x_0) = q_\phi(x_1 | x_0) \prod_{t=2}^T q_\phi(x_t | x_{t-1}, x_0). \quad (2.9)$$

Substituting these into Eq. (2.7) yields

$$\log p(x_0) \geq \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1) \prod_{t=2}^T p(x_{t-1} | x_t)}{q_\phi(x_1 | x_0) \prod_{t=2}^T q_\phi(x_t | x_{t-1}, x_0)} \right]. \quad (2.10)$$

Splitting the chain. We split the logarithm into two terms:

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1)}{q_\phi(x_1 | x_0)} \right] + \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \prod_{t=2}^T \frac{p(x_{t-1} | x_t)}{q_\phi(x_t | x_{t-1}, x_0)} \right]. \quad (2.11)$$

Manipulating the product term. Using Bayes' rule,

$$q_\phi(x_t | x_{t-1}, x_0) = \frac{q_\phi(x_{t-1} | x_t, x_0) q_\phi(x_t | x_0)}{q_\phi(x_{t-1} | x_0)}, \quad (2.12)$$

the product becomes

$$\prod_{t=2}^T \frac{p(x_{t-1} | x_t)}{q_\phi(x_t | x_{t-1}, x_0)} = \prod_{t=2}^T \frac{p(x_{t-1} | x_t)}{q_\phi(x_{t-1} | x_t, x_0)} \cdot \prod_{t=2}^T \frac{q_\phi(x_{t-1} | x_0)}{q_\phi(x_t | x_0)}. \quad (2.13)$$

Using the telescoping identity

$$\prod_{t=2}^T \frac{a_{t-1}}{a_t} = \frac{a_1}{a_T},$$

we obtain

$$\prod_{t=2}^T \frac{q_\phi(x_{t-1} | x_0)}{q_\phi(x_t | x_0)} = \frac{q_\phi(x_1 | x_0)}{q_\phi(x_T | x_0)}. \quad (2.14)$$

Combining terms. Substituting back into Eq. (2.11) and canceling terms yields

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T) p(x_0 | x_1)}{q_\phi(x_T | x_0)} \right] + \mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \prod_{t=2}^T \frac{p(x_{t-1} | x_t)}{q_\phi(x_{t-1} | x_t, x_0)} \right]. \quad (2.15)$$

Reconstruction and prior matching. The first term decomposes as

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} [\log p(x_0 | x_1)] = \mathbb{E}_{q_\phi(x_1|x_0)} [\log p(x_0 | x_1)], \quad (2.16)$$

$$\mathbb{E}_{q_\phi(x_{1:T}|x_0)} \left[\log \frac{p(x_T)}{q_\phi(x_T | x_0)} \right] = -D_{\text{KL}}(q_\phi(x_T | x_0) \| p(x_T)). \quad (2.17)$$

Consistency term. The remaining sum becomes

$$\sum_{t=2}^T \mathbb{E}_{q_\phi(x_t, x_{t-1} | x_0)} \left[\log \frac{p(x_{t-1} | x_t)}{q_\phi(x_{t-1} | x_t, x_0)} \right] = - \sum_{t=2}^T \mathbb{E}_{q_\phi(x_t | x_0)} \left[D_{\text{KL}}(q_\phi(x_{t-1} | x_t, x_0) \| p(x_{t-1} | x_t)) \right]. \quad (2.18)$$

Final ELBO. Replacing $p(\cdot)$ with the learnable reverse model $p_\theta(\cdot)$, the final variational lower bound is

$$\log p(x_0) \geq \underbrace{\mathbb{E}_{q_\phi(x_1 | x_0)} [\log p_\theta(x_0 | x_1)]}_{\text{Reconstruction}} - \underbrace{D_{\text{KL}}(q_\phi(x_T | x_0) \| p(x_T))}_{\text{Prior Matching}} \quad (2.19)$$

$$- \underbrace{\sum_{t=2}^T \mathbb{E}_{q_\phi(x_t | x_0)} \left[D_{\text{KL}}(q_\phi(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t)) \right]}_{\text{Consistency}}. \quad (2.20)$$

step 10: L1 Loss function formulation

Theorem 2.5. The distribution $q_\phi(x_{t-1} | x_t, x_0)$ takes the form

$$q_\phi(x_{t-1} | x_t, x_0) = \mathcal{N}(x_{t-1} | \mu_q(x_t, x_0), \Sigma_q(t)),$$

where

$$\mu_q(x_t, x_0) = \frac{(1 - \alpha_{t-1})\sqrt{\alpha_t}}{1 - \alpha_t} x_t + \frac{(1 - \alpha_t)\sqrt{\alpha_{t-1}}}{1 - \alpha_t} x_0,$$

and

$$\Sigma_q(t) = \frac{(1 - \alpha_t)(1 - \sqrt{\alpha_{t-1}})}{1 - \alpha_t} I \triangleq \sigma_q^2(t)I,$$

with

$$\alpha_t = \prod_{i=1}^t \alpha_i.$$

Constructing $p_\theta(x_{t-1} | x_t)$. An important observation is that the distribution $q_\phi(x_{t-1} | x_t, x_0)$ is completely characterized by x_t and x_0 . No neural network is required to estimate its mean or variance. Once the hyperparameters $\{\alpha_t\}$ are fixed, the distribution $q_\phi(x_{t-1} | x_t, x_0)$ is fully determined.

Consider the evidence lower bound

$$\begin{aligned} \text{ELBO}_{\phi, \theta}(x) &= \mathbb{E}_{q_\phi(x_1 | x_0)} [\log p_\theta(x_0 | x_1)] - D_{\text{KL}}(q_\phi(x_T | x_0) \| p(x_T)) \\ &\quad - \sum_{t=2}^T \mathbb{E}_{q_\phi(x_t | x_0)} [D_{\text{KL}}(q_\phi(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t))]. \end{aligned}$$

Since $q_\phi(x_{t-1} | x_t, x_0)$ is fixed once the noise schedule $\{\alpha_t\}$ is defined, there is no

learning involved for this term. Therefore, the only learnable component inside the consistency term is $p_\theta(x_{t-1} | x_t)$.

To simplify the KL divergence, $p_\theta(x_{t-1} | x_t)$ is chosen to be Gaussian:

$$p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1} | \mu_\theta(x_t), \sigma_q^2(t)I),$$

where the mean $\mu_\theta(x_t)$ is parameterized by a neural network, and the variance is fixed to be identical to that of q_ϕ .

Placing the two distributions side by side,

$$q_\phi(x_{t-1} | x_t, x_0) = \mathcal{N}(x_{t-1} | \mu_q(x_t, x_0), \sigma_q^2(t)I),$$

$$p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1} | \mu_\theta(x_t), \sigma_q^2(t)I).$$

Since both Gaussians share the same covariance, the KL divergence reduces to

$$D_{\text{KL}}(q_\phi(x_{t-1} | x_t, x_0) \| p_\theta(x_{t-1} | x_t)) = \frac{1}{2\sigma_q^2(t)} \|\mu_q(x_t, x_0) - \mu_\theta(x_t)\|_2^2.$$

2.1.4 Training and Inference

In this section, we discuss how to train a variational diffusion model and transform it into a denoising diffusion probabilistic model.

We begin with the evidence lower bound defined previously. The ELBO suggests that training requires finding a network μ_θ that minimizes the loss

$$\frac{1}{2\sigma_q^2(t)} \|\mu_q(x_t, x_0) - \mu_\theta(x_t)\|^2,$$

where $\mu_q(x_t, x_0)$ is known and $\mu_\theta(x_t)$ is parameterized by a neural network.

Recall that

$$\mu_q(x_t, x_0) = \frac{(1 - \alpha_{t-1})\sqrt{\alpha_t}}{1 - \alpha_t} x_t + \frac{(1 - \alpha_t)\sqrt{\alpha_{t-1}}}{1 - \alpha_t} x_0.$$

This expression depends only on x_t and x_0 , and is therefore completely determined once these variables are known.

Since μ_θ is a design choice, it can be defined in a more convenient form. We choose

$$\mu_\theta(x_t) = \frac{(1 - \alpha_{t-1})\sqrt{\alpha_t}}{1 - \alpha_t} x_t + \frac{(1 - \alpha_t)\sqrt{\alpha_{t-1}}}{1 - \alpha_t} \hat{x}_\theta(x_t),$$

where $\hat{x}_\theta(x_t)$ is a neural network that predicts the clean image from x_t .

Substituting these expressions into the loss gives

$$\frac{1}{2\sigma_q^2(t)} \|\mu_q(x_t, x_0) - \mu_\theta(x_t)\|^2 = \frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2 \alpha_{t-1}}{(1 - \alpha_t)^2} \|\hat{x}_\theta(x_t) - x_0\|^2.$$

As a result, the ELBO can be written as

$$\text{ELBO}_\theta(x) = \mathbb{E}_{q(x_1|x_0)} [\log p_\theta(x_0 | x_1)] - \sum_{t=2}^T \mathbb{E}_{q(x_t|x_0)} \left[\frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2 \alpha_{t-1}}{(1 - \alpha_t)^2} \|\hat{x}_\theta(x_t) - x_0\|^2 \right],$$

where the prior matching term has been omitted.

The likelihood term can be expressed as

$$\log p_\theta(x_0 | x_1) = \log \mathcal{N}(x_0 | \mu_\theta(x_1), \sigma_q^2(1)I) \propto -\frac{1}{2\sigma_q^2(1)} \|\hat{x}_\theta(x_1) - x_0\|^2,$$

using the fact that $\alpha_0 = 1$.

Substituting this into the ELBO yields

$$\text{ELBO}_\theta(x) = - \sum_{t=1}^T \mathbb{E}_{q(x_t|x_0)} \left[\frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2 \alpha_{t-1}}{(1 - \alpha_t)^2} \|\hat{x}_\theta(x_t) - x_0\|^2 \right].$$

Ignoring constants and expectations, the optimization problem reduces to

$$\arg \min_{\theta} \|\hat{x}_\theta(x_t) - x_0\|^2,$$

which corresponds to a denoising task.

However, this is not a conventional denoising problem. The noisy input x_t is not arbitrary, but is sampled according to

$$x_t \sim q(x_t | x_0) = \mathcal{N}(\sqrt{\alpha_t}x_0, (1 - \alpha_t)I),$$

or equivalently,

$$x_t = \sqrt{\alpha_t}x_0 + \sqrt{1 - \alpha_t} \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, I).$$

Moreover, the denoising loss is weighted differently across timesteps through the factor

$$\frac{1}{2\sigma_q^2(t)} \frac{(1 - \alpha_t)^2 \alpha_{t-1}}{(1 - \alpha_t)^2}.$$

Using Monte Carlo sampling to approximate the expectation, the final training objective

becomes

$$\arg \min_{\theta} \sum_{x_0 \in \mathcal{X}} \sum_{t=1}^T \frac{1}{M} \sum_{m=1}^M \frac{1}{2\sigma_q^2(t)} \frac{(1-\alpha_t)^2 \alpha_{t-1}}{(1-\alpha_t)^2} \left\| \hat{x}_{\theta} \left(\sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t} \epsilon_t^{(m)} \right) - x_0 \right\|^2,$$

where $\epsilon_t^{(m)} \sim \mathcal{N}(0, I)$ and \mathcal{X} denotes the training dataset.

Thus, training a diffusion model reduces to training a denoiser $\hat{x}_{\theta}(\cdot)$, which is why the resulting model is referred to as a denoising diffusion probabilistic model.

2.1.5 Prediction of Noise

We consider the forward diffusion process defined as

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t} \epsilon_0, \quad \epsilon_0 \sim \mathcal{N}(0, I).$$

Rearranging this expression gives

$$x_0 = \frac{x_t - \sqrt{1-\alpha_t} \epsilon_0}{\sqrt{\alpha_t}}.$$

The posterior mean of the reverse process can originally be written as a function of x_t and x_0 . By substituting the above expression for x_0 , the posterior mean can instead be expressed in terms of x_t and the noise variable ϵ_0 :

$$\mu_q(x_t, x_0) = \frac{1}{\sqrt{\alpha_t}} x_t - \frac{1-\alpha_t}{\sqrt{\alpha_t(1-\alpha_t)}} \epsilon_0.$$

This reformulation shows that the reverse-process mean depends on the noise added during the forward diffusion step rather than directly on the clean image. Motivated by this observation, we parameterize the model mean using a neural network that predicts the noise:

$$\mu_{\theta}(x_t) = \frac{1}{\sqrt{\alpha_t}} x_t - \frac{1-\alpha_t}{\sqrt{\alpha_t(1-\alpha_t)}} \hat{\epsilon}_{\theta}(x_t),$$

where $\hat{\epsilon}_{\theta}(x_t)$ denotes the network output.

Substituting these expressions into the variational objective yields an evidence lower bound that depends on the discrepancy between the predicted noise and the true noise:

$$\mathcal{L}_{\text{ELBO}}(\theta) = - \sum_{t=1}^T \mathbb{E}_{x_0, \epsilon_0} \left[\frac{1}{2\sigma_q^2(t)} \frac{(1-\alpha_t)^2}{\alpha_t(1-\alpha_t)} \left\| \hat{\epsilon}_{\theta}(\sqrt{\alpha_t} x_0 + \sqrt{1-\alpha_t} \epsilon_0) - \epsilon_0 \right\|^2 \right].$$

As a result, training the diffusion model reduces to a denoising problem in which the network learns to predict the noise injected at each diffusion step. This formulation is commonly referred to as *noise prediction* in denoising diffusion probabilistic models.

2.1.6 Probability Distributions in DDIM

To motivate DDIM, consider a special choice of parameters where the noise schedule is reparameterized by replacing α_t with the ratio α_t/α_{t-1} . With this choice, the forward transition distribution is defined as

$$q(x_t | x_{t-1}) := \mathcal{N}\left(x_t \mid \sqrt{\frac{\alpha_t}{\alpha_{t-1}}} x_{t-1}, \left(1 - \frac{\alpha_t}{\alpha_{t-1}}\right) I\right).$$

This parameterization does not have a strong physical interpretation, but it significantly simplifies notation. In particular, the product of the ratios satisfies

$$\alpha_t = \prod_{i=1}^t \frac{\alpha_i}{\alpha_{i-1}} = \alpha_t,$$

assuming $\alpha_0 = 1$. As a result, the marginal distribution takes the familiar form

$$q(x_t | x_0) = \mathcal{N}(x_t | \sqrt{\alpha_t} x_0, (1 - \alpha_t)I).$$

Using reparameterization, the noisy sample x_t can be written as

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

By the same argument, one can express

$$x_{t-1} = \sqrt{\alpha_{t-1}} x_0 + \sqrt{1 - \alpha_{t-1}} \epsilon, \quad \epsilon \sim \mathcal{N}(0, I).$$

The key observation in DDIM is that instead of treating ϵ as an independent Gaussian noise variable, it can be expressed in terms of x_t and x_0 . From the expression for x_t , we have

$$\sqrt{1 - \alpha_t} \epsilon = x_t - \sqrt{\alpha_t} x_0,$$

which implies

$$\epsilon = \frac{x_t - \sqrt{\alpha_t} x_0}{\sqrt{1 - \alpha_t}}.$$

Substituting this expression into the formula for x_{t-1} yields

$$x_{t-1} = \sqrt{\alpha_{t-1}} x_0 + \sqrt{1 - \alpha_{t-1}} \left(\frac{x_t - \sqrt{\alpha_t} x_0}{\sqrt{1 - \alpha_t}} \right).$$

This formulation differs fundamentally from the standard DDPM construction. In DDPM, the stochasticity is injected by an independent Gaussian noise term, which simplifies theoretical analysis but leads to slow ancestral sampling. In contrast, the above expression replaces the random noise by a deterministic estimate constructed from

x_t and x_0 .

This modification enables a crucial design choice in DDIM: the ability to define transition distributions such that the marginal distribution remains invariant in form. Specifically, the goal is to ensure that

$$q(x_{t-1} \mid x_0) = \mathcal{N}(\sqrt{\alpha_{t-1}} x_0, (1 - \alpha_{t-1})I),$$

which mirrors the structure of $q(x_t \mid x_0)$. Maintaining this marginal form is essential, since the forward process must interpolate smoothly between the data distribution at $t = 0$ and pure Gaussian noise at $t = T$.

While many choices of transition distributions $q(x_{t-1} \mid x_t, x_0)$ are possible, only a restricted class preserves the desired marginal behavior. Accordingly, DDIM defines the conditional distribution as

$$q(x_{t-1} \mid x_t, x_0) = \mathcal{N}\left(\sqrt{\alpha_{t-1}} x_0 + \sqrt{1 - \alpha_{t-1}} \left(\frac{x_t - \sqrt{\alpha_t} x_0}{\sqrt{1 - \alpha_t}}\right), \sigma_t^2 I\right),$$

where σ_t is a hyperparameter controlling the amount of stochasticity. Setting $\sigma_t = 0$ yields a fully deterministic reverse process, which is the defining characteristic of DDIM.

Inference for DDIM

Inference in DDIM is derived from the transition distribution of the forward process. Given a noisy sample x_t , the forward process can be written as

$$x_t = \sqrt{\alpha_t} x_0 + \sqrt{1 - \alpha_t} \epsilon,$$

where ϵ is Gaussian noise.

During inference, x_t is given and the goal is to recover an estimate of x_0 . Rearranging the above expression yields

$$x_0 = \frac{1}{\sqrt{\alpha_t}} (x_t - \sqrt{1 - \alpha_t} \epsilon).$$

Since the true noise ϵ is unknown, it is replaced by a neural network estimate $\epsilon_\theta^{(t)}(x_t)$. This leads to the definition

$$f_\theta^{(t)}(x_t) := \frac{1}{\sqrt{\alpha_t}} \left(x_t - \sqrt{1 - \alpha_t} \epsilon_\theta^{(t)}(x_t) \right),$$

which serves as an estimate of the original clean signal x_0 .

The DDIM transition distribution is defined as $q(x_{t-1} \mid x_t, x_0)$. Since x_0 is unknown

during inference, it is replaced by $f_\theta^{(t)}(x_t)$, giving

$$p_\theta(x_{t-1} \mid x_t) := q(x_{t-1} \mid x_t, f_\theta^{(t)}(x_t)).$$

Substituting this estimate into the transition distribution yields

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\left(\sqrt{\alpha_{t-1}} f_\theta^{(t)}(x_t) + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \frac{x_t - \sqrt{\alpha_t} f_\theta^{(t)}(x_t)}{\sqrt{1 - \alpha_t}}, \sigma_t^2 I\right).$$

Replacing $f_\theta^{(t)}(x_t)$ by its explicit expression leads to

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\left(\sqrt{\alpha_{t-1}} \frac{x_t - \sqrt{1 - \alpha_t} \epsilon_\theta^{(t)}(x_t)}{\sqrt{\alpha_t}} + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \epsilon_\theta^{(t)}(x_t), \sigma_t^2 I\right).$$

Using reparameterization, the DDIM sampling update becomes

$$x_{t-1} = \sqrt{\alpha_{t-1}} \frac{x_t - \sqrt{1 - \alpha_t} \epsilon_\theta^{(t)}(x_t)}{\sqrt{\alpha_t}} + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \epsilon_\theta^{(t)}(x_t) + \sigma_t \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, I).$$

For the special case $t = 1$, the reverse process is defined as

$$p_\theta(x_0 \mid x_1) = \mathcal{N}\left(f_\theta^{(1)}(x_1), \sigma_1^2 I\right),$$

ensuring that the final reconstruction is supported everywhere.

Chapter 3

Methodology

Chapter 4

Experiments

Chapter 5

Conclusion

Appendix A

Appendix