Lecture 8: Diffusion Models

Deep Generative Models

Sajjad Amini

Department of Electrical Engineering
Sharif University of Technology

Contents

Section 1

Citation

Citation

The presentation of Variational Diffusion models in this lecture is inspired by the following reference:

- Luo, Calvin. "Understanding diffusion models: A unified perspective." arXiv preprint arXiv:2208.11970 (2022).

Section 2

Intuition

Noise Conditional Score Network (source of images: [1])

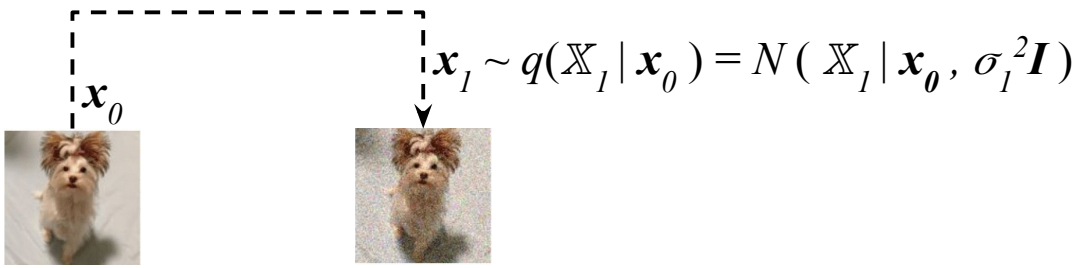


Figure: Samples with noise level σ_1

Noise Conditional Score Network (source of images: [1])

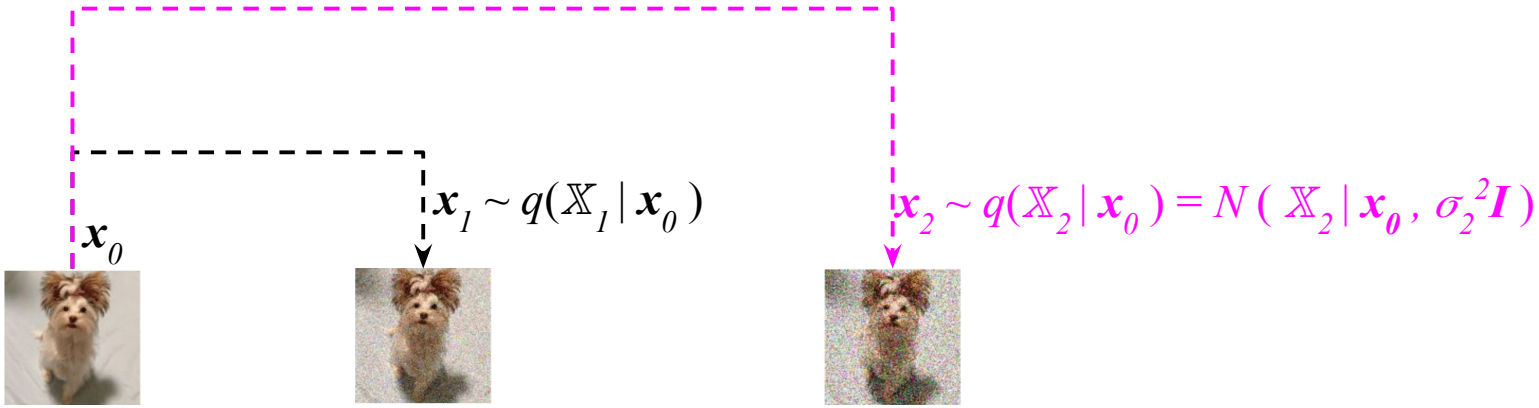


Figure: Samples with noise level σ_2 ($\sigma_2 \geq \sigma_1$)

Noise Conditional Score Network (source of images: [1])

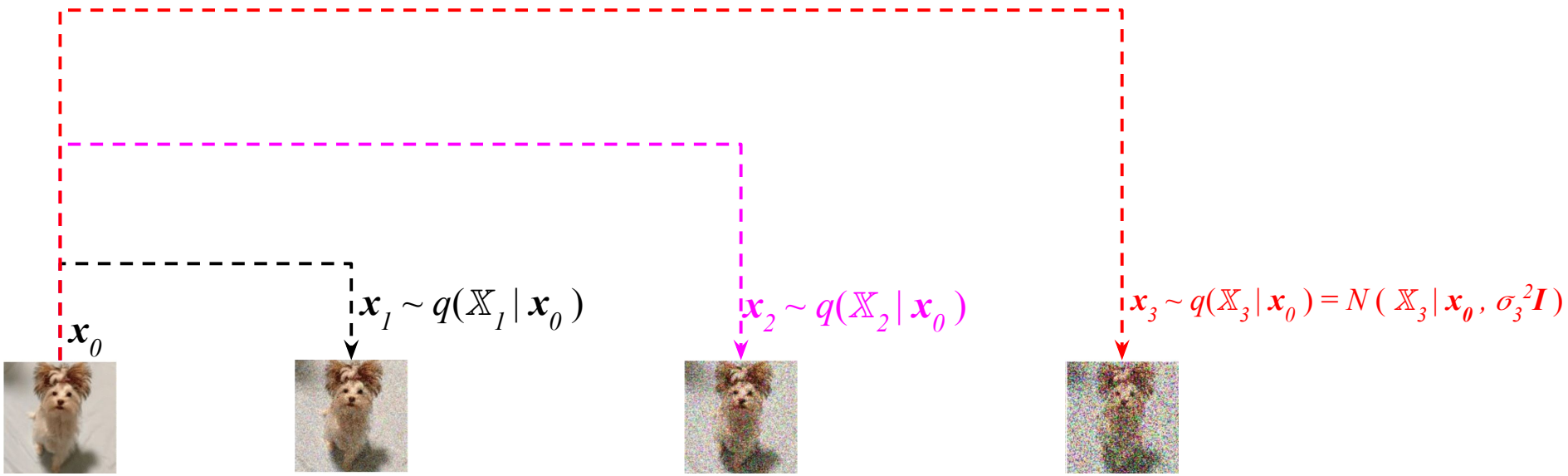


Figure: Samples with noise level σ_3 ($\sigma_3 \geq \sigma_2 \geq \sigma_1$)

Noise Conditional Score Network (source of images: [1])

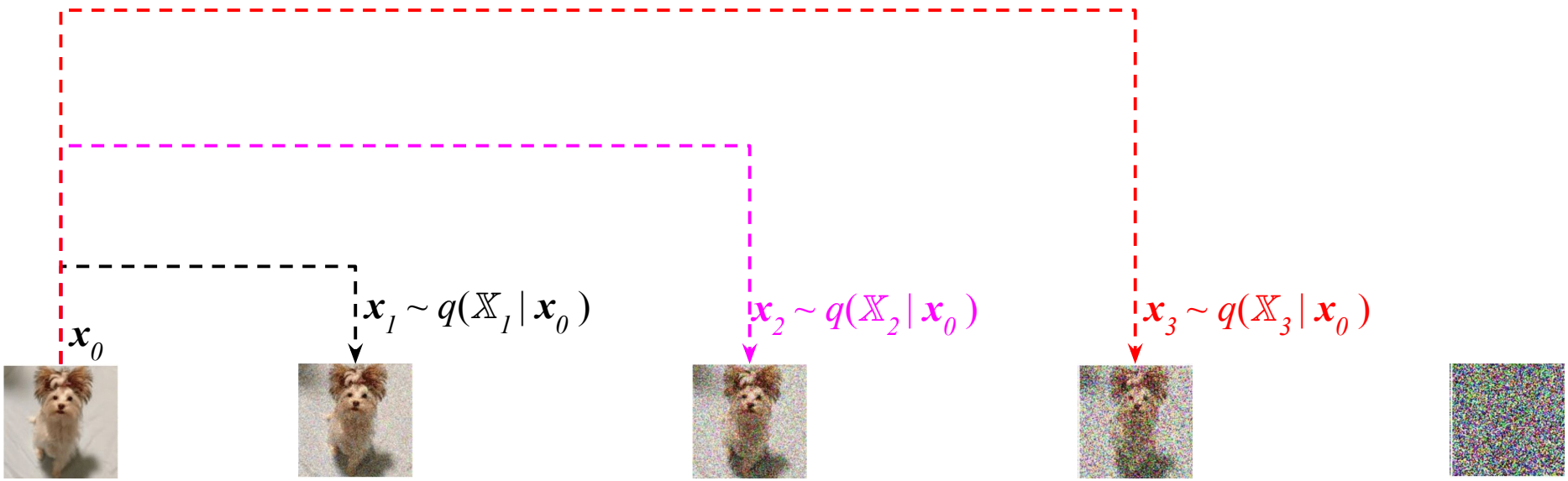


Figure: Sampling Initialization using noise sample

Noise Conditional Score Network (source of images: [1])

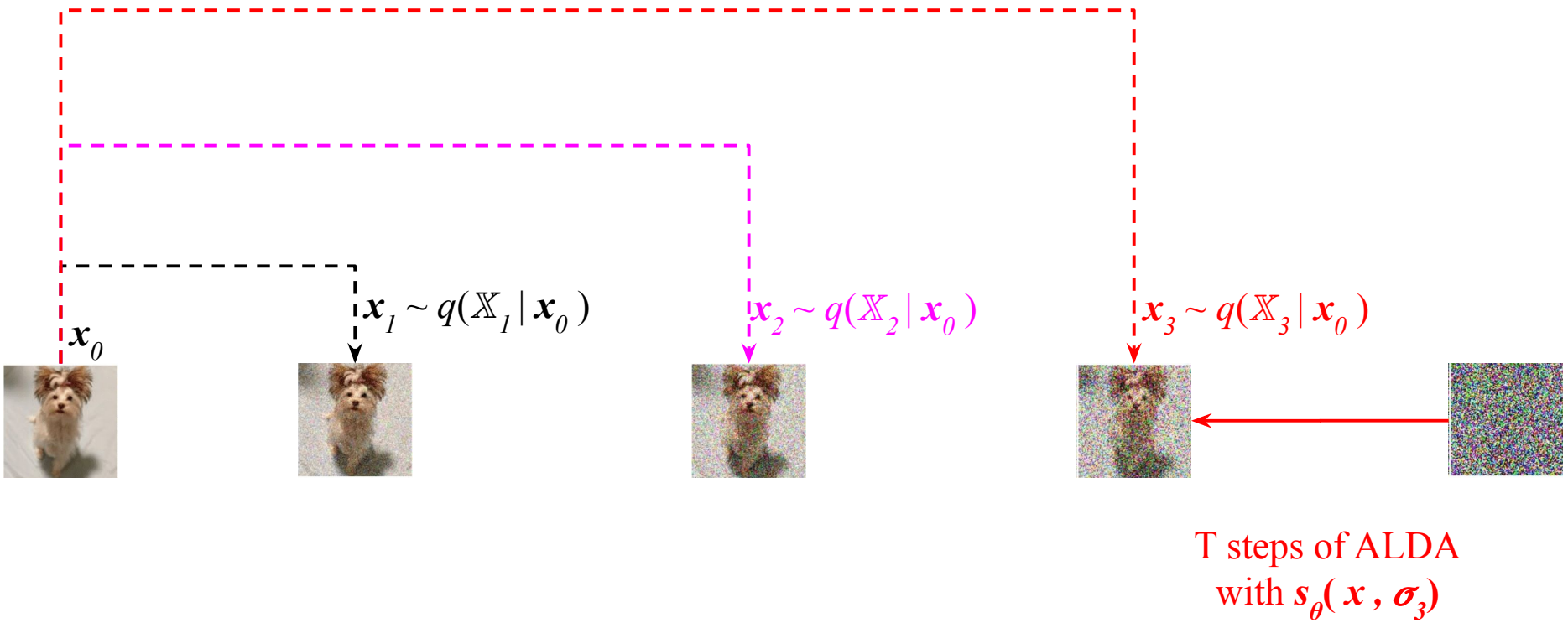


Figure: Approaching $p(\mathbb{X}_3)$ using Annealed Langevin dynamics algorithm (ALDA)

Noise Conditional Score Network (source of images: [1])

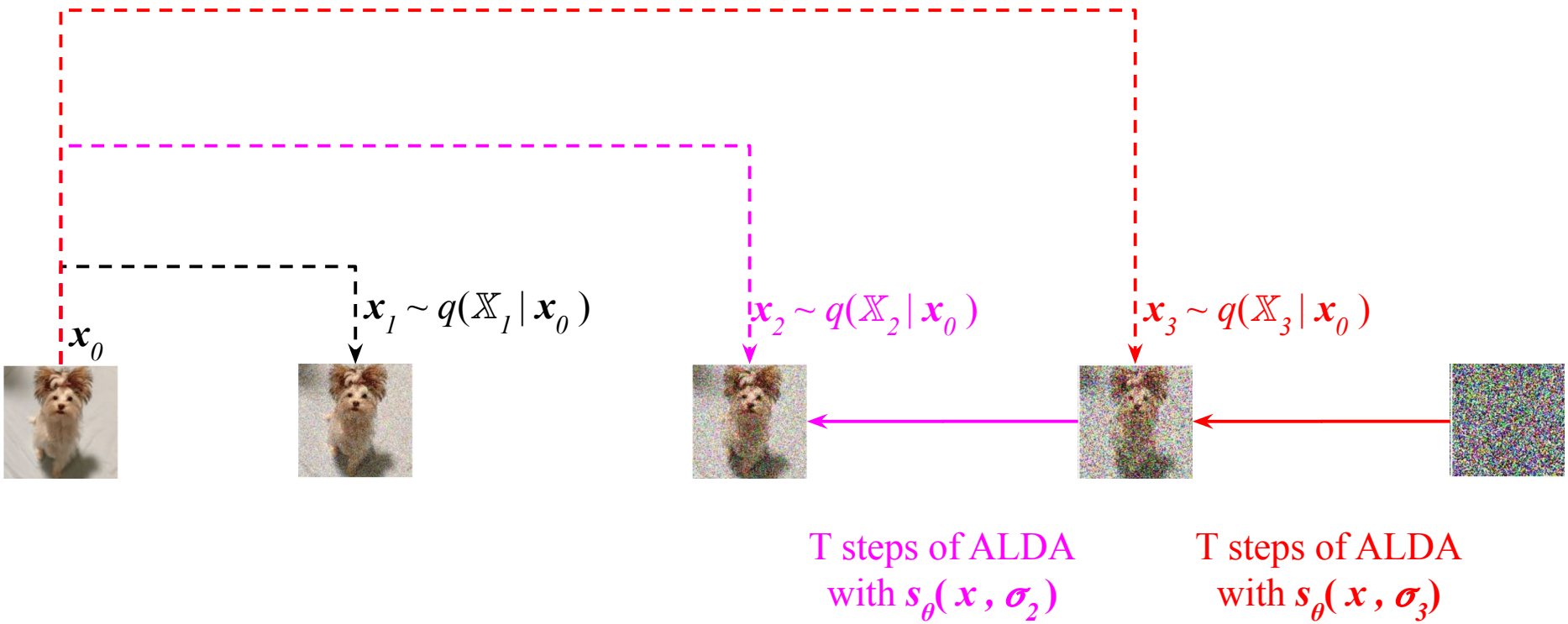


Figure: Approaching $p(\mathbb{X}_2)$ using Annealed Langevin dynamics algorithm (ALDA)

Noise Conditional Score Network (source of images: [1])

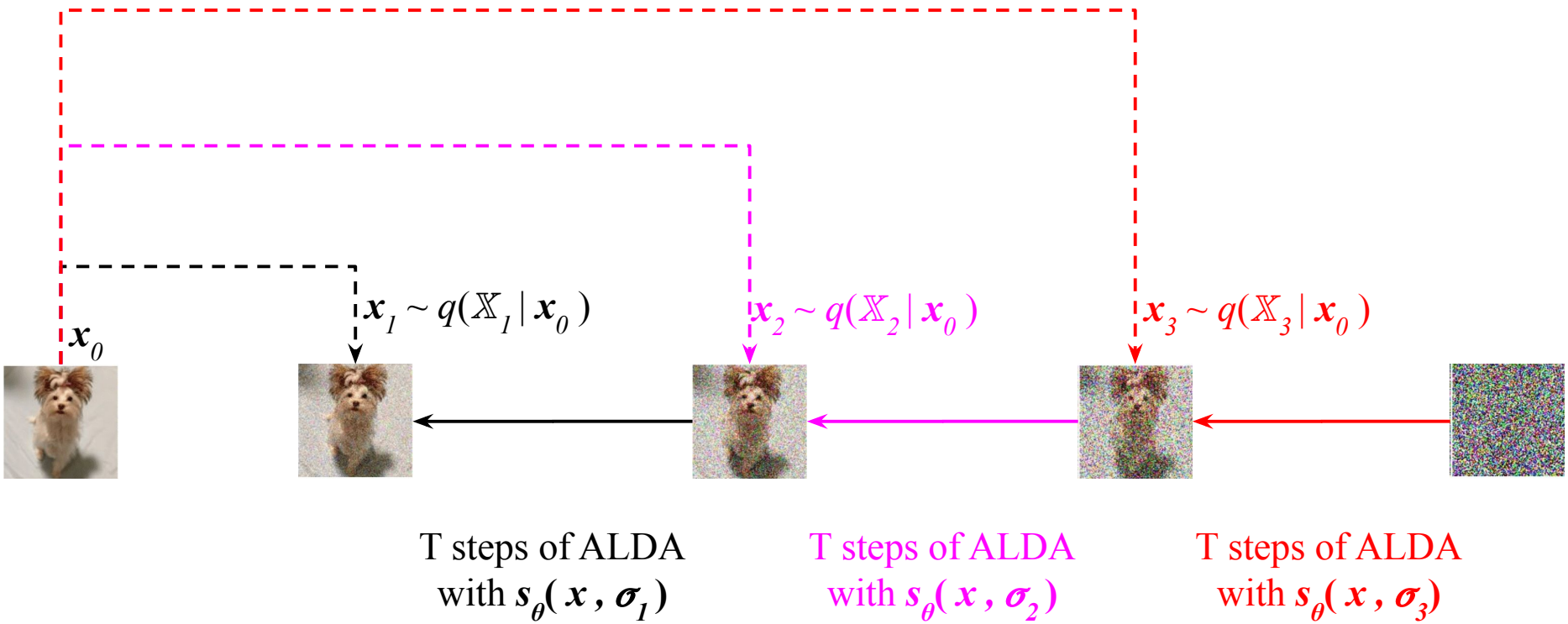


Figure: Approaching $p(\mathbb{X}_1)$ using Annealed Langevin dynamics algorithm (ALDA)

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

- Manifold hypothesis

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

- Manifold hypothesis
- Inaccurate score estimation with score matching

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

- Manifold hypothesis
- Inaccurate score estimation with score matching
- Slow mixing of Langevin dynamics

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

- Manifold hypothesis
- Inaccurate score estimation with score matching
- Slow mixing of Langevin dynamics

Most Noisy Data

In NCSN and for the most noisy case:

- The input $\boldsymbol{x}$ is *almost* buried with noise.

Intuition

NCSN Overview

Train a model to learn the score function for noisy data $\mathbf{s}_\theta(\boldsymbol{x}, \sigma)$ for several levels of noise to solve:

- Manifold hypothesis
- Inaccurate score estimation with score matching
- Slow mixing of Langevin dynamics

Most Noisy Data

In NCSN and for the most noisy case:

- The input $\boldsymbol{x}$ is *almost* buried with noise.
- There is *still* $\boldsymbol{x}$ information in the noisy input $\widetilde{\boldsymbol{x}}$

Intuition

Denoising Diffusion Probabilistic Model

Adding Gaussian noise to input in a Markov chain model until we reach a pure standard Gaussian noise in $\widetilde{\boldsymbol{x}}$.

Intuition

Denoising Diffusion Probabilistic Model

Adding Gaussian noise to input in a Markov chain model until we reach a pure standard Gaussian noise in $\widetilde{\boldsymbol{x}}$.

Now assume that we can train a model that can estimate the score function for different noise levels, then:

Intuition

Denoising Diffusion Probabilistic Model

Adding Gaussian noise to input in a Markov chain model until we reach a pure standard Gaussian noise in $\widetilde{\boldsymbol{x}}$.

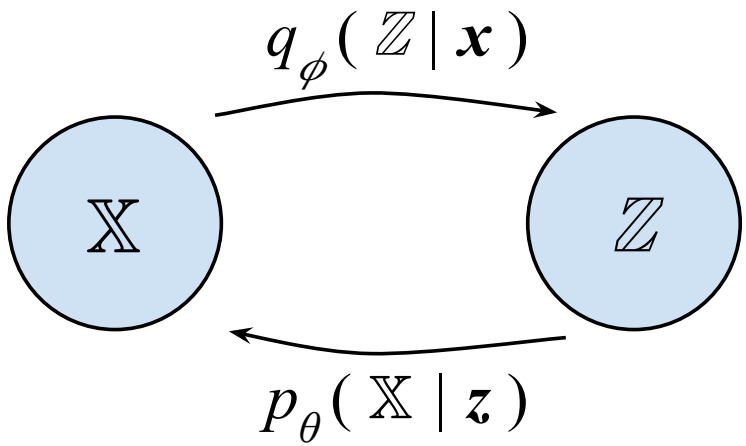
Now assume that we can train a model that can estimate the score function for different noise levels, then:

Starting from pure standard Gaussian noise, we can progressively denoise the input using the trained model and finally generate a sample from the data distribution. Thus we have a generative model

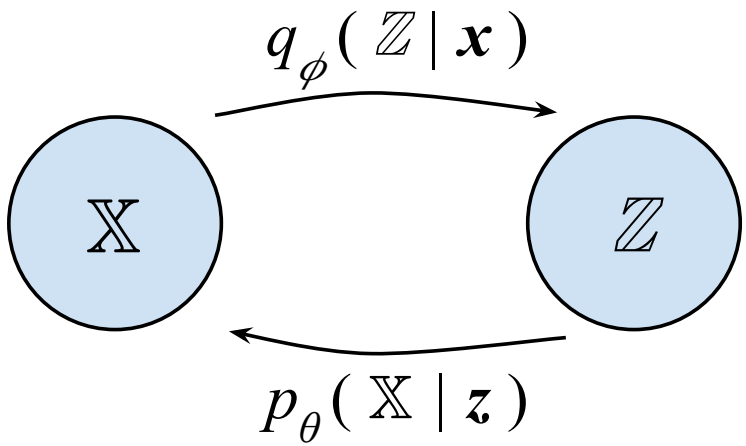
Section 3

Variational Deffusion Models

Variational AutoEncoder



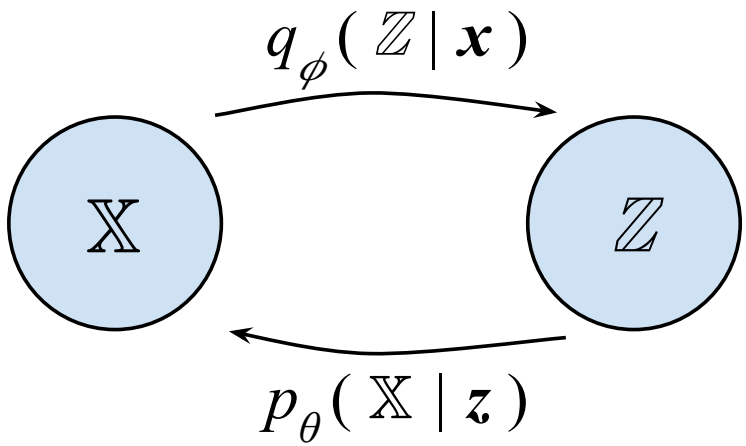
Variational AutoEncoder



VAE

In VAE, the evidence (observation log-likelihood) is lower bounded as:

Variational AutoEncoder

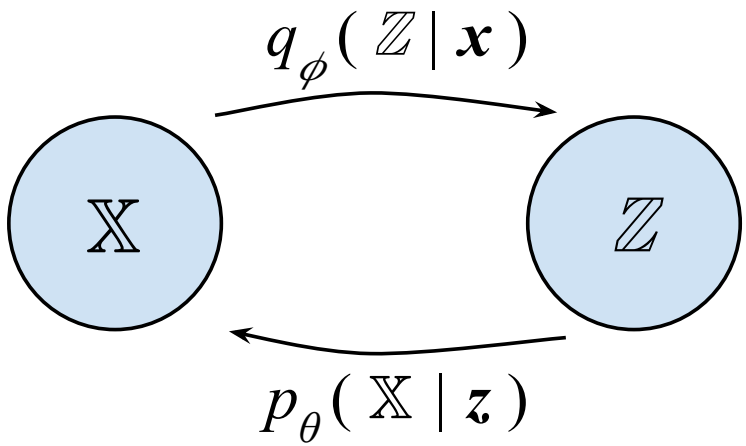


VAE

In VAE, the evidence (observation log-likelihood) is lower bounded as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi})$$

Variational AutoEncoder

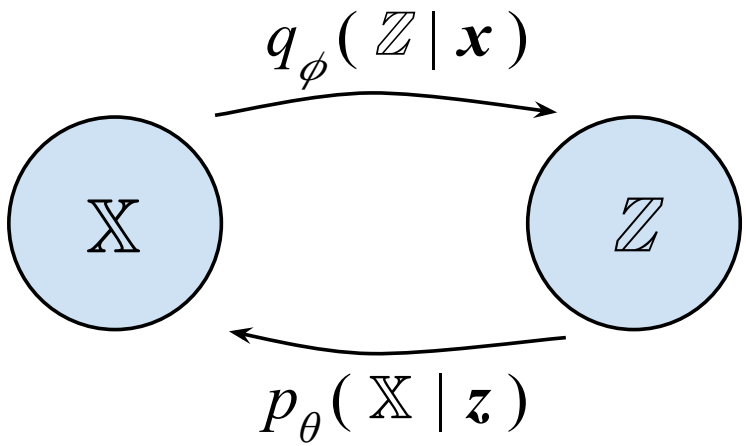


VAE

In VAE, the evidence (observation log-likelihood) is lower bounded as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{z\sim q_\phi(\mathbb{Z}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right)$$

Variational AutoEncoder



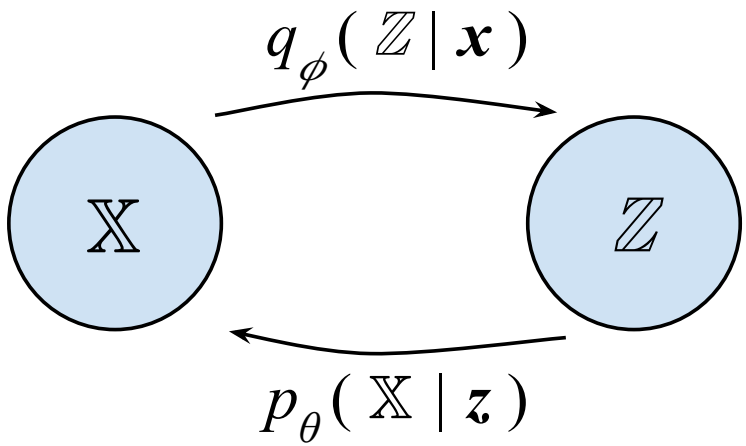
VAE

In VAE, the evidence (observation log-likelihood) is lower bounded as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{z\sim q_\phi(\mathbb{Z}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right)$$

$$= \mathbb{E}_{z\sim q_\phi(\mathbb{Z}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right) + \text{KL}\left(q_\phi(\mathbb{Z}|\boldsymbol{x})\|p(\mathbb{Z})\right)$$

Variational AutoEncoder



VAE

In VAE, the evidence (observation log-likelihood) is lower bounded as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{z\sim q_\phi(\mathbb{Z}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right)$$

$$= \mathbb{E}_{z\sim q_\phi(\mathbb{Z}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x}|\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right) + \text{KL}\left(q_\phi(\mathbb{Z}|\boldsymbol{x})\|p(\mathbb{Z})\right)$$

Assuming the prior distribution has no trainable parameter: $p_\theta(\mathbb{Z}) = p(\mathbb{Z})$

Markovian Hierarchical Variational Autoencoders

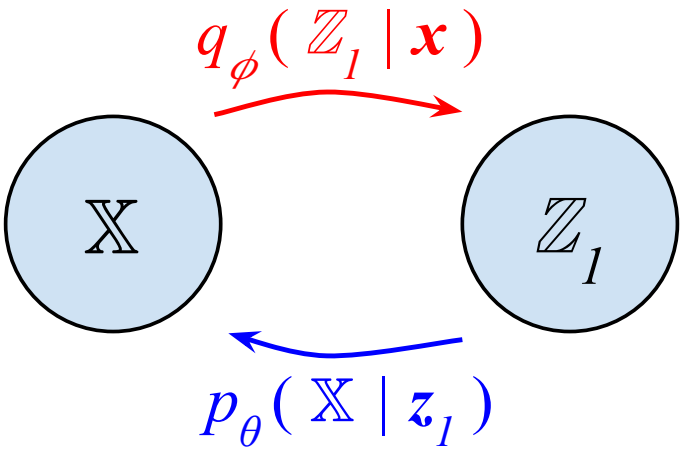


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational Autoencoders

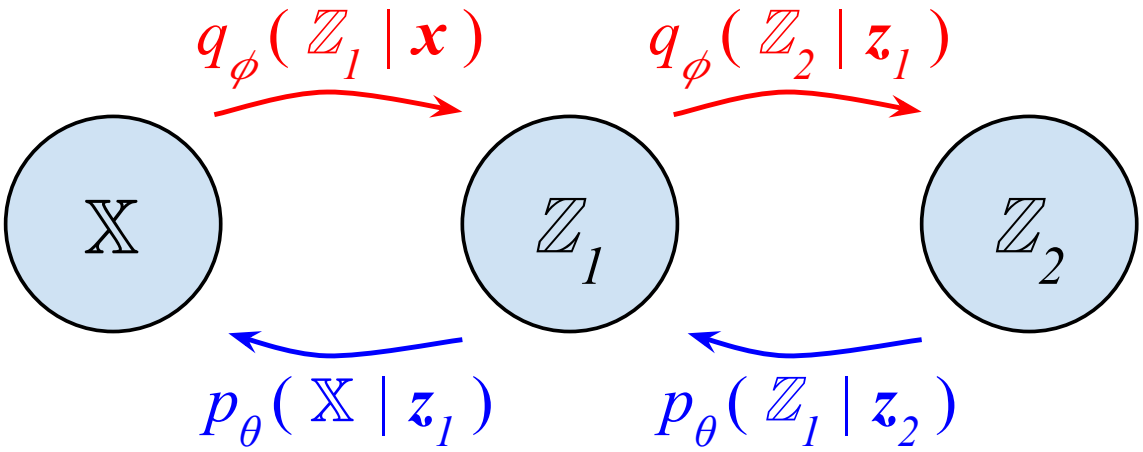


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational Autoencoders

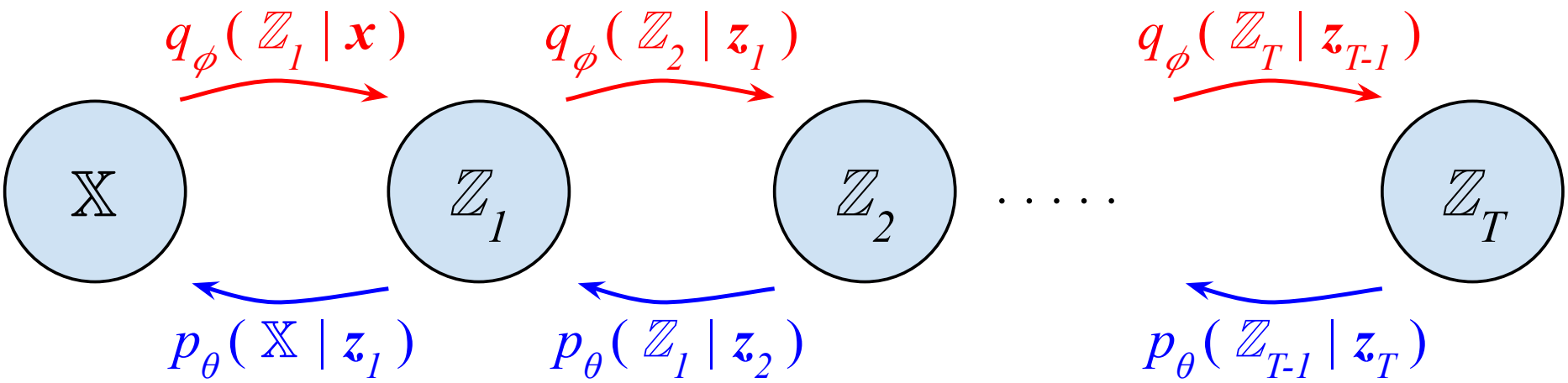


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational Autoencoders

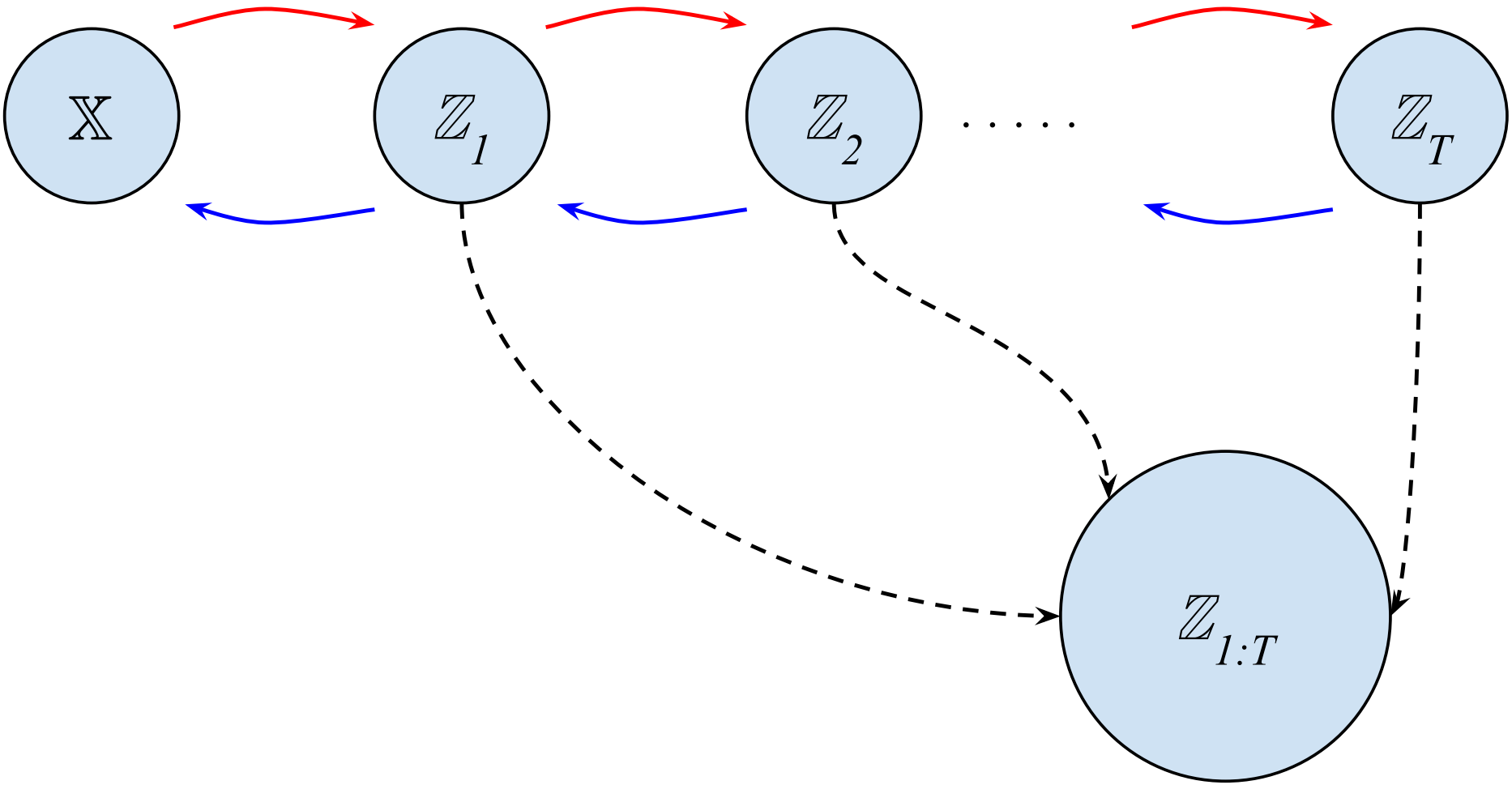


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational Autoencoders

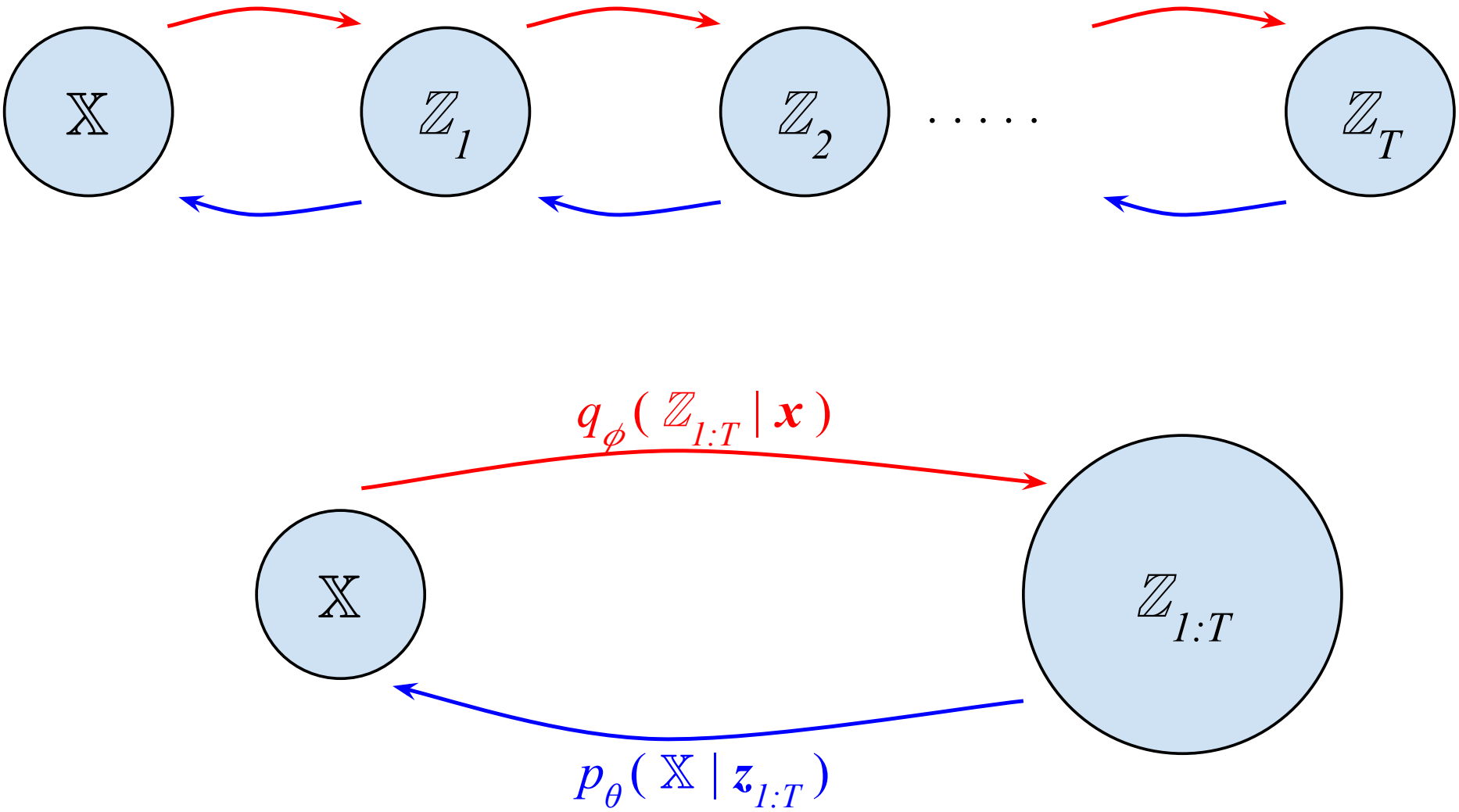


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational Autoencoders

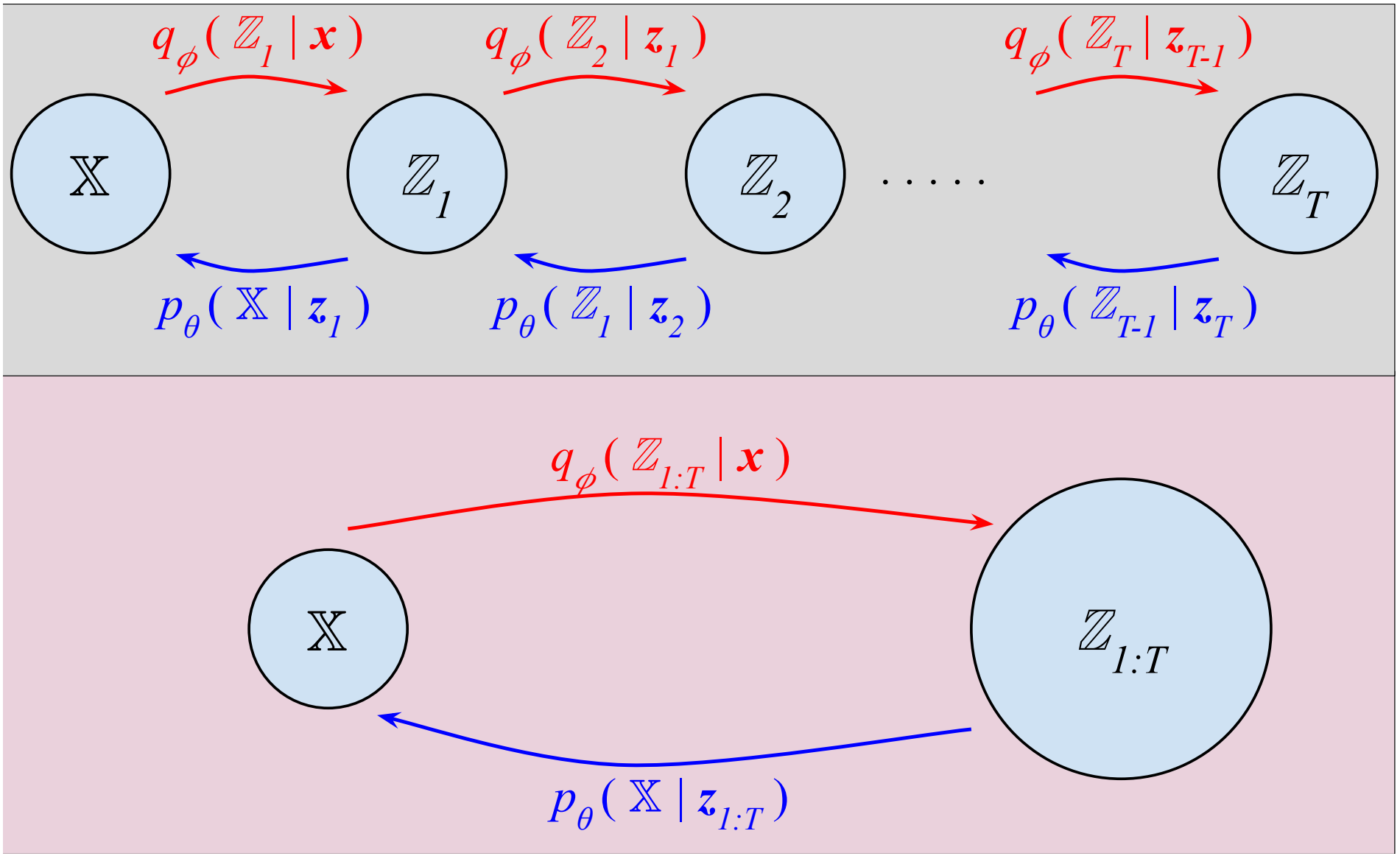
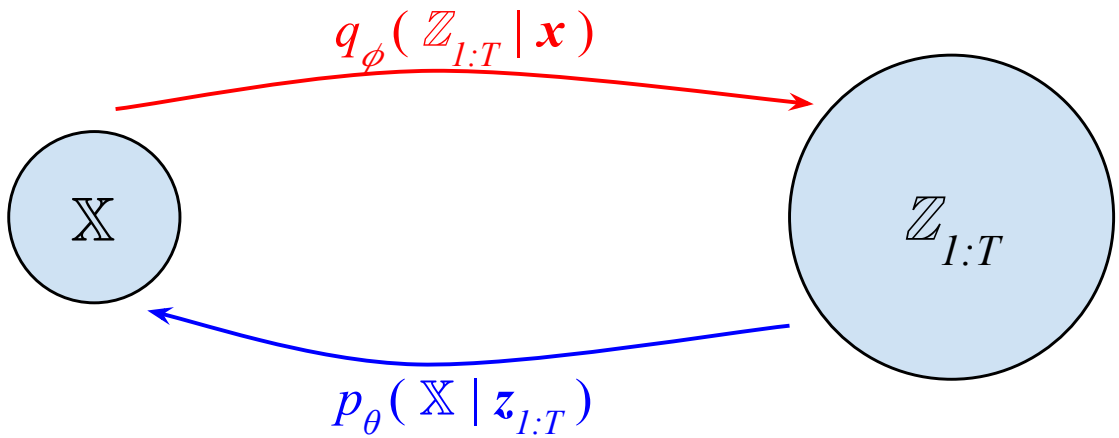


Figure: Hierarchical Variational Autoencoder

Markovian Hierarchical Variational AutoEncoder

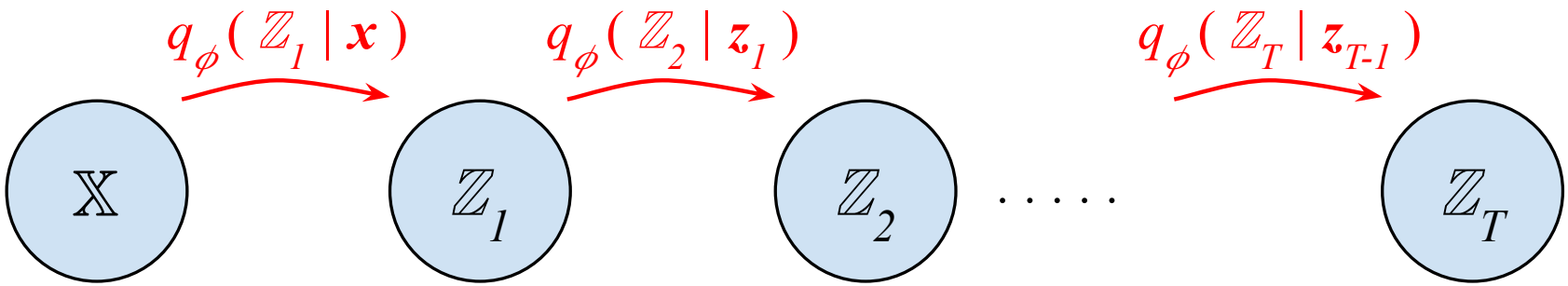


HVAE

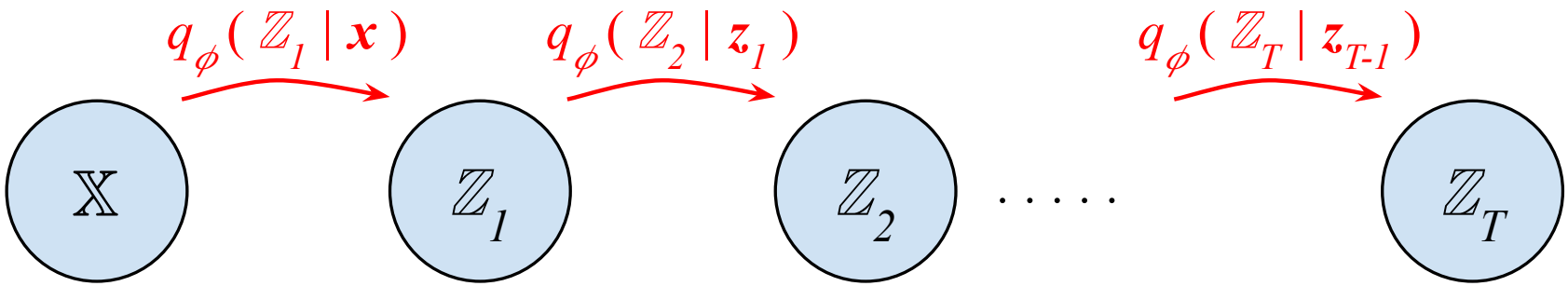
In VAE, the evidence (observation log-likelihood) is lower bounded as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x};\boldsymbol{\theta},\boldsymbol{\phi}) = \mathbb{E}_{q_\phi(\mathbb{Z}_{1:T}|\boldsymbol{x})}\left(\log \frac{p_\theta(\boldsymbol{x},\boldsymbol{z}_{1:T})}{q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x})}\right)$$

Encoder Part



Encoder Part

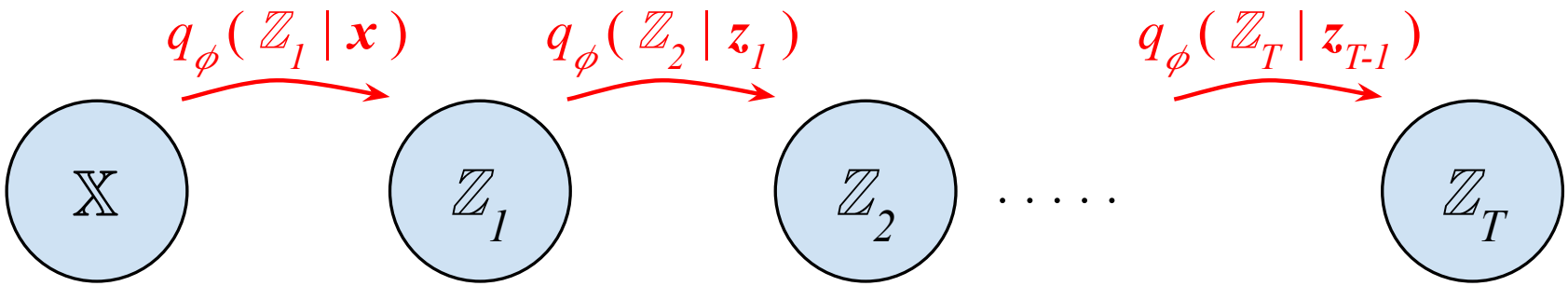


Independencies

The above connections impose the following independencies:

$$\mathbb{Z}_t|\mathbb{Z}_{t-1} \perp \mathbb{Z}_{t-2:T}, \mathbb{X} \Rightarrow p(\mathbb{Z}_t|\boldsymbol{x}, \boldsymbol{z}_{1:t-1}) = p(\mathbb{Z}_t|\boldsymbol{z}_{t-1})$$

Encoder Part



Independencies

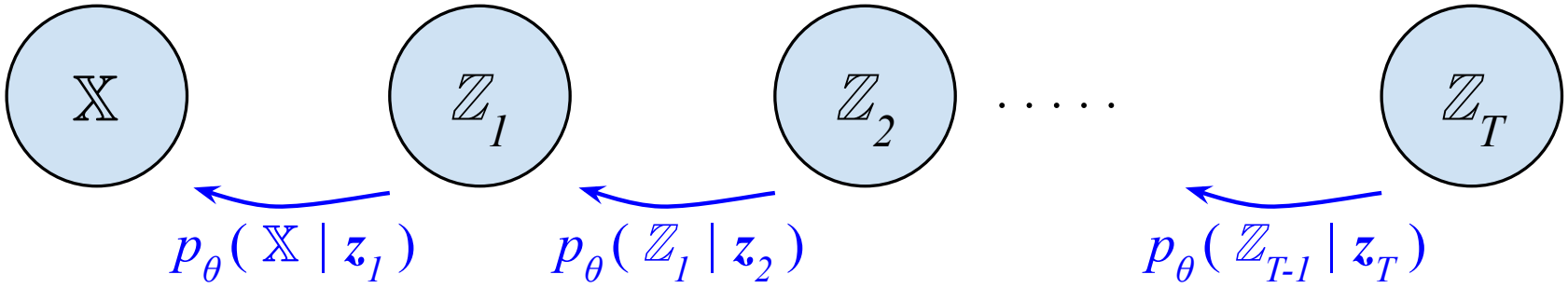
The above connections impose the following independencies:

$$\mathbb{Z}_t|\mathbb{Z}_{t-1} \perp \mathbb{Z}_{t-2:T}, \mathbb{X} \Rightarrow p(\mathbb{Z}_t|\boldsymbol{x}, \boldsymbol{z}_{1:t-1}) = p(\mathbb{Z}_t|\boldsymbol{z}_{t-1})$$

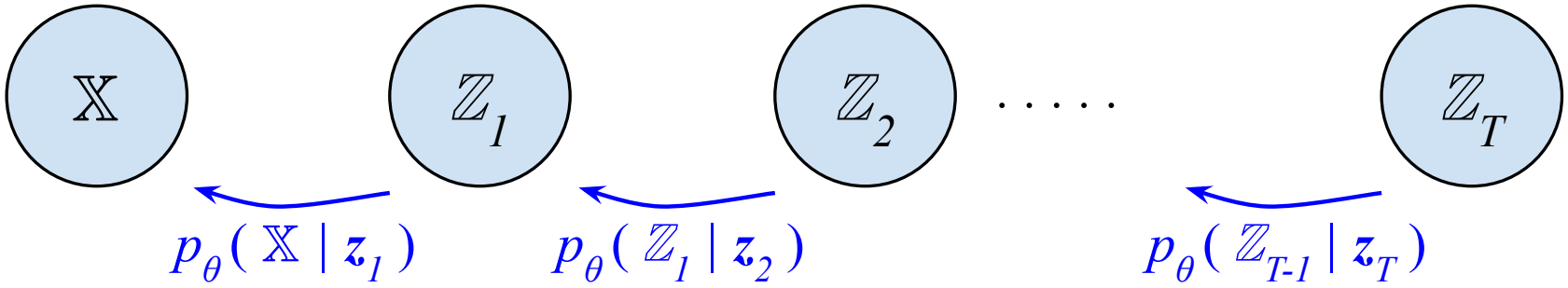
So using the chain rule, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \underbrace{q_\phi(\boldsymbol{z}_2|\boldsymbol{x}, \boldsymbol{z}_1)}_{q_\phi(\boldsymbol{z}_2|\boldsymbol{z}_1)} \dots \underbrace{q_\phi(\boldsymbol{z}_t|\boldsymbol{x}, \boldsymbol{z}_{1:t-1})}_{q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})} \dots \underbrace{q_\phi(\boldsymbol{z}_T|\boldsymbol{x}, \boldsymbol{z}_{q:T-1})}_{q_\phi(\boldsymbol{z}_T|\boldsymbol{z}_{T-1})}$$

Decoder Part



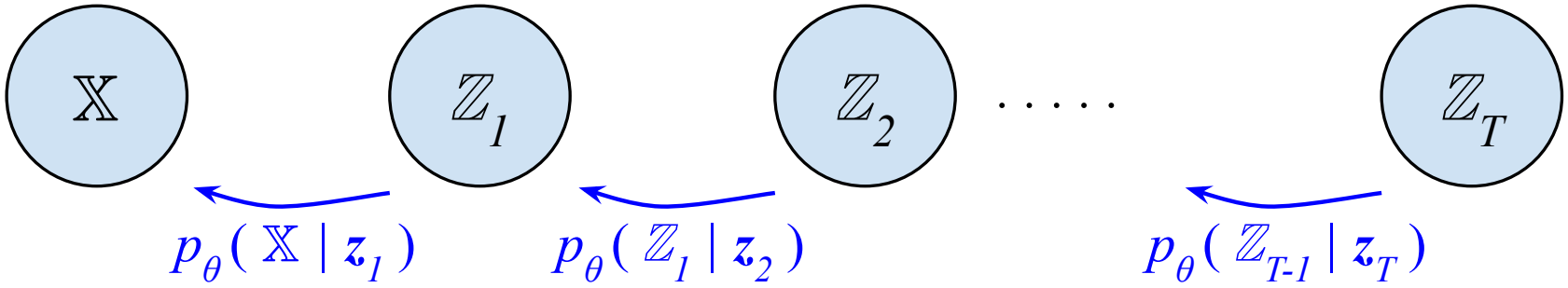
Decoder Part



Independencies

The above connections impose the following independencies:

Decoder Part

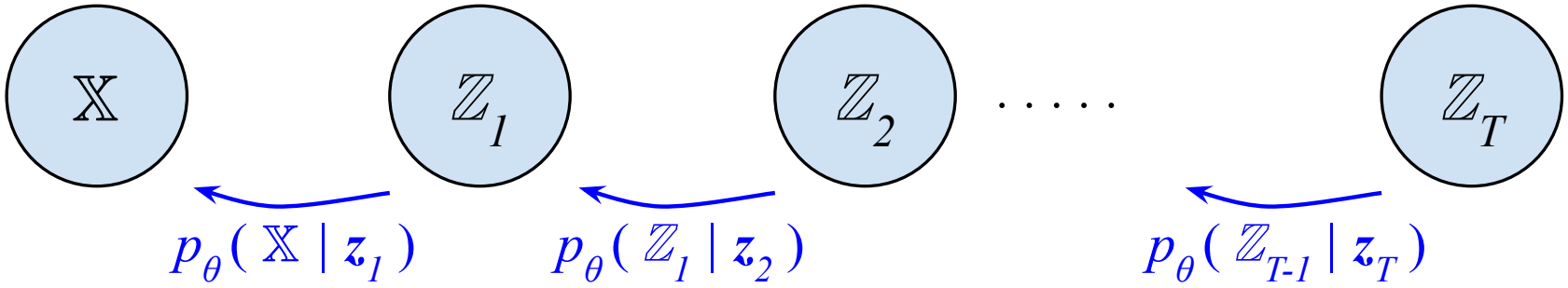


Independencies

The above connections impose the following independencies:

$$\mathbb{Z}_t|\mathbb{Z}_{t+1} \perp \mathbb{Z}_{t+2:T} \Rightarrow p(\mathbb{Z}_t|\boldsymbol{z}_{t+1:T}) = p(\mathbb{Z}_t|\boldsymbol{z}_{t+1})$$

Decoder Part



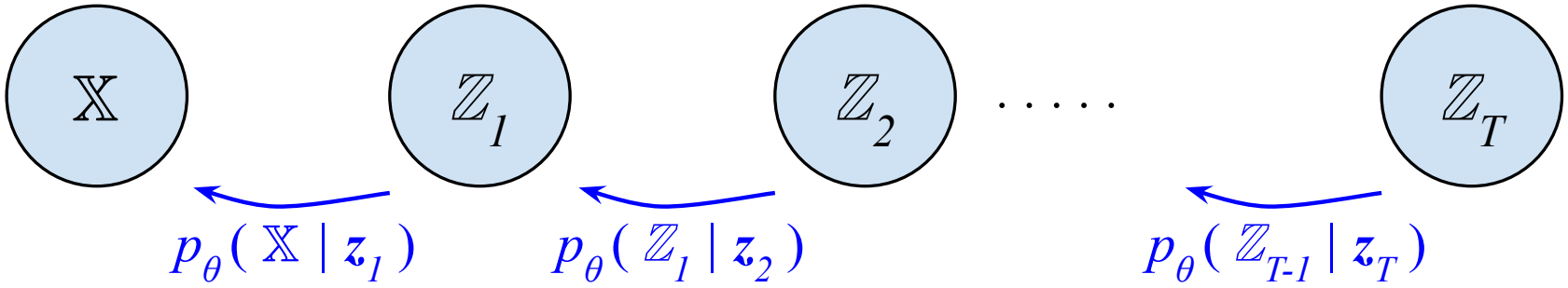
Independencies

The above connections impose the following independencies:

$$\mathbb{Z}_t|\mathbb{Z}_{t+1} \perp \mathbb{Z}_{t+2:T} \Rightarrow p(\mathbb{Z}_t|\boldsymbol{z}_{t+1:T}) = p(\mathbb{Z}_t|\boldsymbol{z}_{t+1})$$

$$\mathbb{X}|\mathbb{Z}_1 \perp \mathbb{Z}_{2:T} \Rightarrow p(\mathbb{X}|\boldsymbol{z}_{1:T}) = p(\mathbb{X}|\boldsymbol{z}_1)$$

Decoder Part



Independencies

The above connections impose the following independencies:

$$\mathbb{Z}_t|\mathbb{Z}_{t+1} \perp \mathbb{Z}_{t+2:T} \Rightarrow p(\mathbb{Z}_t|\boldsymbol{z}_{t+1:T}) = p(\mathbb{Z}_t|\boldsymbol{z}_{t+1})$$

$$\mathbb{X}|\mathbb{Z}_1 \perp \mathbb{Z}_{2:T} \Rightarrow p(\mathbb{X}|\boldsymbol{z}_{1:T}) = p(\mathbb{X}|\boldsymbol{z}_1)$$

So using the chain rule, we have:

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:T}) = p_\theta(\boldsymbol{z}_T)p_\theta(\boldsymbol{z}_{T-1}|\boldsymbol{z}_T)\ldots\underbrace{p_\theta(\boldsymbol{z}_t|\boldsymbol{z}_{t+1:T})}_{p_\theta(\boldsymbol{z}_t|\boldsymbol{z}_{t+1})}\ldots\underbrace{p_\theta(\boldsymbol{x}|\boldsymbol{z}_{1:T})}_{p_\theta(\boldsymbol{x}|\boldsymbol{z}_1)}$$

ELBO

From the encoder conditional independencies, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$$

ELBO

From the encoder conditional independencies, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$$

And from decoder conditional independencies, we have:

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:T}) = p_\theta(\boldsymbol{z}_T) p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)$$

ELBO

From the encoder conditional independencies, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$$

And from decoder conditional independencies, we have:

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:T}) = p_\theta(\boldsymbol{z}_T) p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)$$

So we can rewrite ELBO as:

ELBO

From the encoder conditional independencies, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$$

And from decoder conditional independencies, we have:

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:T}) = p_\theta(\boldsymbol{z}_T) p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)$$

So we can rewrite ELBO as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi})$$

ELBO

From the encoder conditional independencies, we have:

$$q_\phi(\boldsymbol{z}_{1:T}|\boldsymbol{x}) = q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})$$

And from decoder conditional independencies, we have:

$$p_\theta(\boldsymbol{x}, \boldsymbol{z}_{1:T}) = p_\theta(\boldsymbol{z}_T) p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)$$

So we can rewrite ELBO as:

$$\log p_\theta(\boldsymbol{x}) \geq \text{ELBO}(\boldsymbol{x}; \boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_\phi(\mathbb{Z}_{1:T}|\boldsymbol{x})} \left(\log \frac{p_\theta(\boldsymbol{z}_T) p_\theta(\boldsymbol{x}|\boldsymbol{z}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t)}{q_\phi(\boldsymbol{z}_1|\boldsymbol{x}) \prod_{t=2}^{T} q_\phi(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})} \right)$$

Section 4

Variational Diffusion Models

Variational Diffusion Models

From MHVAE to VDM

Assume a MHVAE with the following restrictions:

- Each latent vector $\mathbb{Z}_t, 1 \leq t \leq T$ dimension equals the visible random vector $\mathbb{X}$ dimension.

From MHVAE to VDM

Assume a MHVAE with the following restrictions:

- Each latent vector $\mathbb{Z}_t, 1 \leq t \leq T$ dimension equals the visible random vector $\mathbb{X}$ dimension.
- The encoder is simply a predefined linear Gaussian model (it is not trainable anymore in contrast to vanilla VAE).

From MHVAE to VDM

Assume a MHVAE with the following restrictions:

- Each latent vector $\mathbb{Z}_t, 1 \leq t \leq T$ dimension equals the visible random vector $\mathbb{X}$ dimension.
- The encoder is simply a predefined linear Gaussian model (it is not trainable anymore in contrast to vanilla VAE).
- The encoder is designed such that the distribution over the last latent vector $\mathbb{Z}_T$ is standard Gaussian.

Variational Diffusion Model - Restriction 1

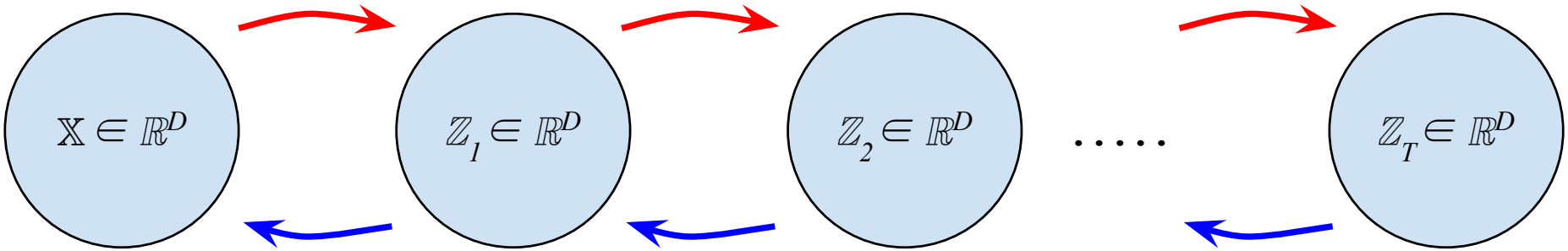


Figure: MHVAE with the same dimension for visible and latent random vectors

Variational Diffusion Model - Restriction 1

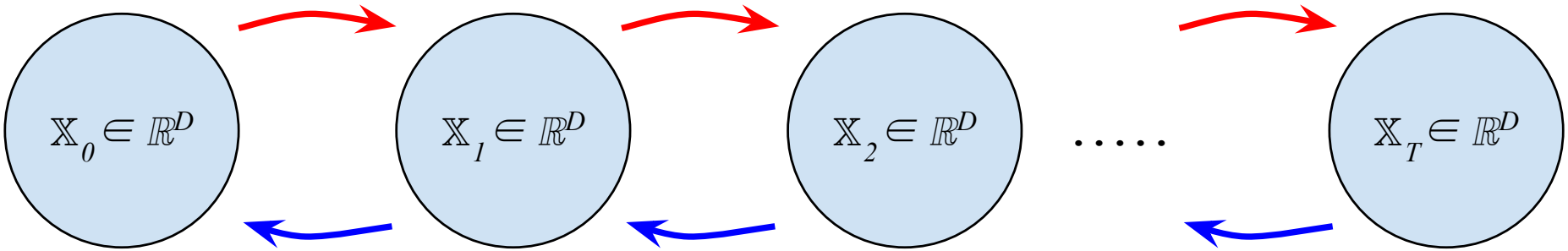


Figure: Notation update for VDM

Variational Diffusion Model - Restriction 1

Notation update

- For a simplified notation:
$$\begin{cases} \text{Observed vector:} & \boldsymbol{x}_0 \\ \text{Latent vector:} & \boldsymbol{x}_t,\ 1 \leq t \leq T \end{cases}$$

Variational Diffusion Model - Restriction 1

Notation update

- For a simplified notation: $\begin{cases} \text{Observed vector:} & \boldsymbol{x}_0 \\ \text{Latent vector:} & \boldsymbol{x}_t,\ 1 \leq t \leq T \end{cases}$
- The nominator and denominator in ELBO can be written as:

$$q_\phi(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) = \prod_{t=1}^{T} q_\phi(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}), \quad p(\boldsymbol{x}_{0:T}) = p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$$

Variational Diffusion Model - Restriction 2

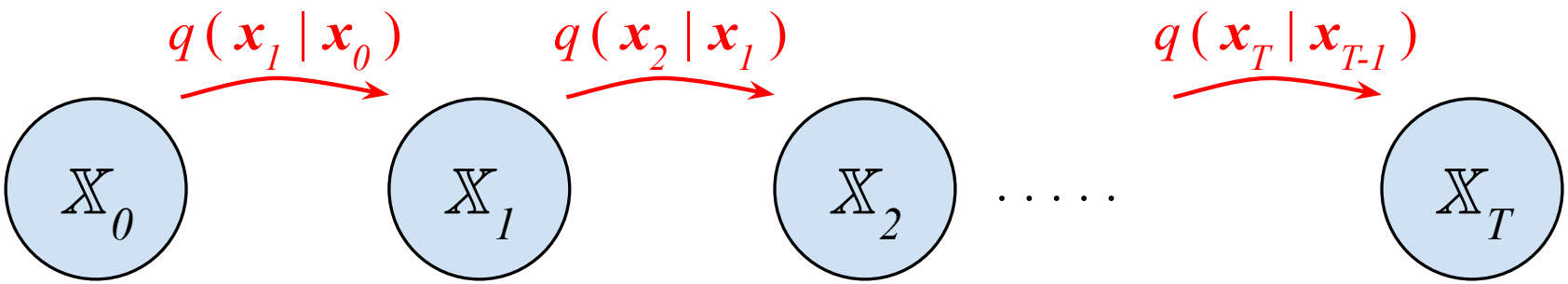


Figure: Encoder for VDM without any trainable parameter

Variational Diffusion Model - Restriction 2

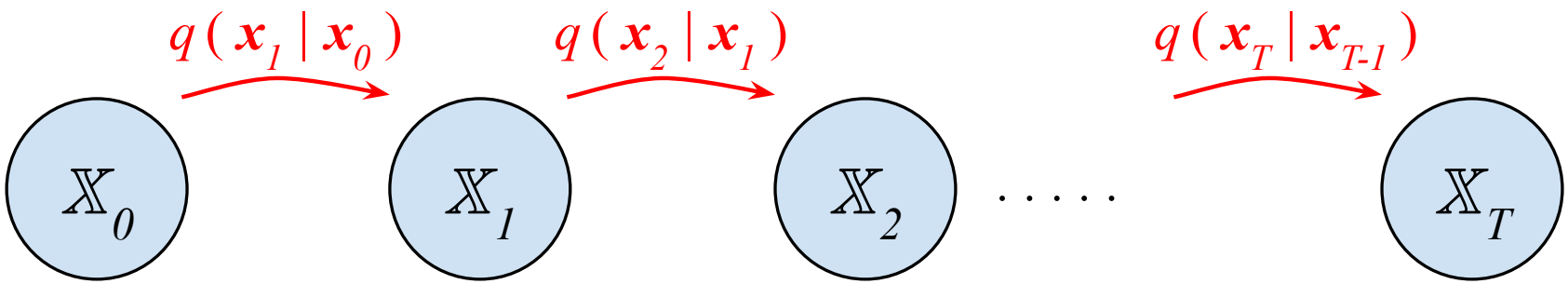


Figure: Encoder for VDM without any trainable parameter

Encoder Transition

The encoder transition is a linear Gaussian model as:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\Big(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\Big)$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

We know:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\Big(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\Big)$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

We know:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\Big(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\Big)$$

Using the Reparameterization trick, we have:

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1},\ \epsilon_{t-1} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

We know:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\Big(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\Big)$$

Using the Reparameterization trick, we have:

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\boldsymbol{x}_{t-1} + \sqrt{1-\alpha_t}\epsilon_{t-1},\ \epsilon_{t-1} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$$

Using the same procedure for $\boldsymbol{x}_{t-1}$, we have:

$$\boldsymbol{x}_{t-1} = \sqrt{\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2},\ \epsilon_{t-2} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

Using two equations for $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ in Slide 21, we have:

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

Using two equations for $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ in Slide 21, we have:

$$\boldsymbol{x}_t = \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}\right) + \sqrt{1-\alpha_t}\epsilon_{t-1}$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

Using two equations for $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ in Slide 21, we have:

$$\begin{aligned}\boldsymbol{x}_t =& \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}\right) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\ =& \sqrt{\alpha_t\alpha_{t-1}}\boldsymbol{x}_{t-2} + \underbrace{\left(\sqrt{\alpha_t - \alpha_t\alpha_{t-1}}\epsilon_{t-2} + \sqrt{1-\alpha_t}\epsilon_{t-1}\right)}_{\sim\mathcal{N}(\mathbf{0},(1-\alpha_t\alpha_{t-1})\boldsymbol{I})}\end{aligned}$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

Using two equations for $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ in Slide 21, we have:

$$\begin{aligned}
\boldsymbol{x}_t =& \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}\right) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
=& \sqrt{\alpha_t\alpha_{t-1}}\boldsymbol{x}_{t-2} + \underbrace{\left(\sqrt{\alpha_t - \alpha_t\alpha_{t-1}}\epsilon_{t-2} + \sqrt{1-\alpha_t}\epsilon_{t-1}\right)}_{\sim\mathcal{N}(\mathbf{0},(1-\alpha_t\alpha_{t-1})\boldsymbol{I})} \\
=& \sqrt{\alpha_t\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\epsilon^\star_{t-2},\ \epsilon^\star_{t-2} \sim \mathcal{N}(\mathbf{0},\boldsymbol{I})
\end{aligned}$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

Using two equations for $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ in Slide 21, we have:

$$\begin{aligned}
\boldsymbol{x}_t =& \sqrt{\alpha_t}\left(\sqrt{\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_{t-1}}\epsilon_{t-2}\right) + \sqrt{1-\alpha_t}\epsilon_{t-1} \\
=& \sqrt{\alpha_t\alpha_{t-1}}\boldsymbol{x}_{t-2} + \underbrace{\left(\sqrt{\alpha_t - \alpha_t\alpha_{t-1}}\epsilon_{t-2} + \sqrt{1-\alpha_t}\epsilon_{t-1}\right)}_{\sim\mathcal{N}(\mathbf{0},(1-\alpha_t\alpha_{t-1})\boldsymbol{I})} \\
=& \sqrt{\alpha_t\alpha_{t-1}}\boldsymbol{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\epsilon^\star_{t-2},\ \epsilon^\star_{t-2} \sim \mathcal{N}(\mathbf{0},\boldsymbol{I})
\end{aligned}$$

If we follow the above procesure (replace $\boldsymbol{x}_{t-2}$), finally we can show:

$$\boldsymbol{x}_t = \sqrt{\prod_{i=1}^{t}\alpha_i}\ \boldsymbol{x}_0 + \sqrt{1-\prod_{i=1}^{t}\alpha_i}\ \epsilon^\star_0$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

If we define $\bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$, then we have (for simplicity we change $\epsilon_0^\star$ to ϵ_0):

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon_0,\ \ \epsilon_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

If we define $\bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$, then we have (for simplicity we change $\epsilon_0^\star$ to ϵ_0):

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon_0,\ \ \epsilon_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Thus we can write:

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

Variational Diffusion Model - Restriction 2 (cont.)

Working on Conditional Posterior

If we define $\bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$, then we have (for simplicity we change $\epsilon_0^\star$ to ϵ_0):

$$\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon_0,\ \ \epsilon_0 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$$

Thus we can write:

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

Notation Update

As the forward diffusion process has no trainable parameter (there is no $\boldsymbol{\phi}$), we can write ELBO as:

$$\text{ELBO}(\boldsymbol{x}_0; \boldsymbol{\theta}) = \mathbb{E}_{q(\mathbb{X}_{1:T}|\boldsymbol{x}_0)} \left(\log \frac{p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})} \right)$$

Variational Diffusion Model - Restriction 3

Last Latent Distribution

During the encoder design, assume we select $\alpha_t, 1 \leq t \leq T$ such that:

$$\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$$

Variational Diffusion Model - Restriction 3

Last Latent Distribution

During the encoder design, assume we select $\alpha_t, 1 \leq t \leq T$ such that:

$$\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$$

Then we conclude:

$$p(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_T|\mathbf{0}, \boldsymbol{I})$$

Variational Diffusion Model - Restriction 3

Last Latent Distribution

During the encoder design, assume we select $\alpha_t, 1 \leq t \leq T$ such that:

$$\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$$

Then we conclude:

$$p(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_T|\mathbf{0}, \boldsymbol{I})$$

So, independent of what you observe, the unconditional distribution over the last latent vector is standard Gaussian.

Variational Diffusion Model - Restriction 3

Last Latent Distribution

During the encoder design, assume we select $\alpha_t, 1 \leq t \leq T$ such that:

$$\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$$

Then we conclude:

$$p(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}(\boldsymbol{x}_T|\boldsymbol{0}, \boldsymbol{I})$$

So, independent of what you observe, the unconditional distribution over the last latent vector is standard Gaussian. Thus nominator in ELBO is:

$$p_\theta(\boldsymbol{x}_{0:T}) = p(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t), \; p(\boldsymbol{x}_T) = \mathcal{N}(\boldsymbol{x}_T|\boldsymbol{0}, \boldsymbol{I})$$

Forward and Reverse Diffusion

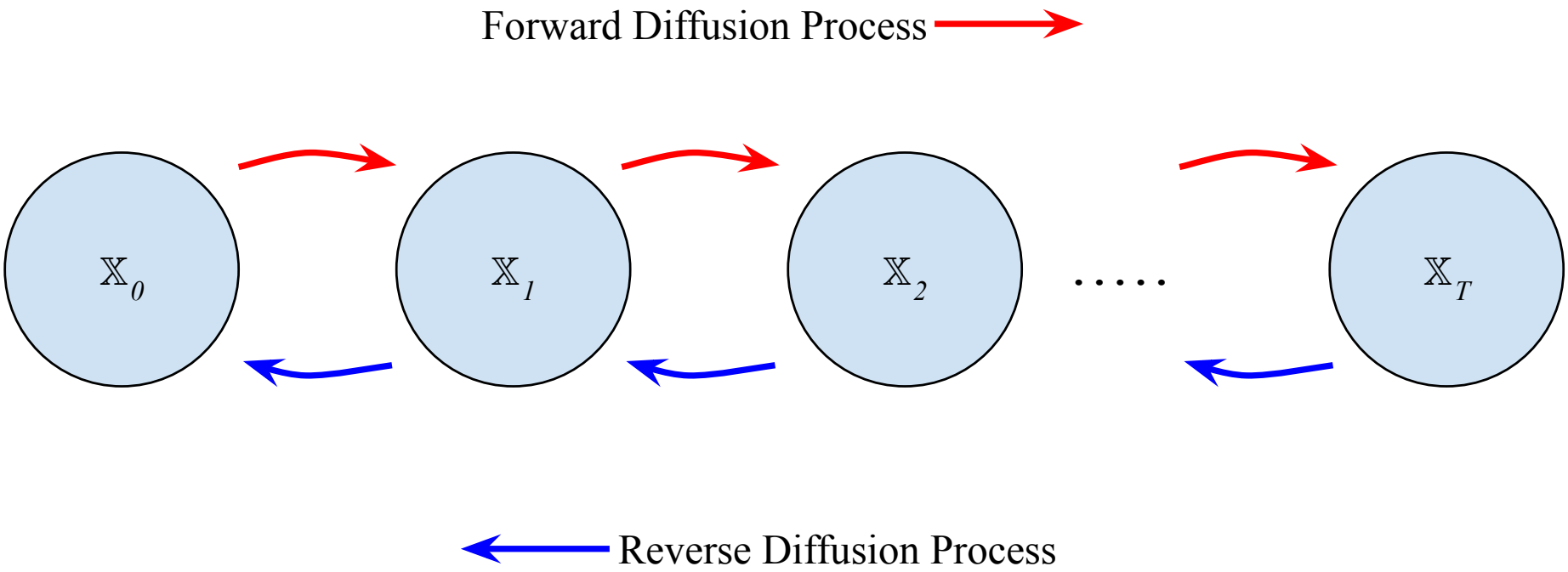


Figure: *Forward diffusion process* and *reverse diffusion process* as an alternative name for encoder and decoder, respectively.

Consequence of Restriction 2

Reverse Process Distribution

For Gaussian forward diffusion and $\alpha_t \lesssim 1$, the reverse diffusion process is also Gaussian [2].

Consequence of Restriction 2

Reverse Process Distribution

For Gaussian forward diffusion and $\alpha_t \lesssim 1$, the reverse diffusion process is also Gaussian [2].So we conclude:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}\Big(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t).\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t)\Big)$$

Consequence of Restriction 2

Reverse Process Distribution

For Gaussian forward diffusion and $\alpha_t \lesssim 1$, the reverse diffusion process is also Gaussian [2].So we conclude:

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}\Big(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t).\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t)\Big)$$

So the problem of training a VDM is equivalent to designing the model based on deep neural networks to produce $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t)$ and $\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t)$.

Pay Attention

Selecting T

- Restriction 3 necessitates:

$$\bar{\alpha}_T = \prod_{t=1}^{T} \alpha_t \simeq 0$$

Pay Attention

Selecting T

- Restriction 3 necessitates:

$$\bar{\alpha}_T = \prod_{t=1}^{T} \alpha_t \simeq 0$$

- To have a Gaussian reverse process, we need:

$$\alpha_t \lesssim 1,\ 1 \leq t \leq T$$

Pay Attention

Selecting T

- Restriction 3 necessitates:

$$\bar{\alpha}_T = \prod_{t=1}^{T} \alpha_t \simeq 0$$

- To have a Gaussian reverse process, we need:

$$\alpha_t \lesssim 1,\ 1 \leq t \leq T$$

☞ Altogether we need a large T to satisfy both conditions.

Forward Diffusion Process (source of images: [1])

$\boldsymbol{x}_0$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

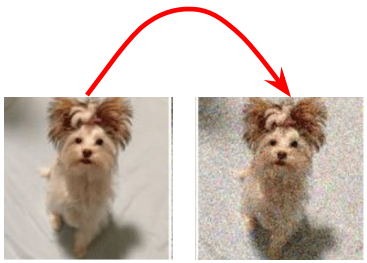
$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\mathbb{X}_1 \middle| \sqrt{\alpha_1}\, \boldsymbol{x}_0, (1-\alpha_1)\, \boldsymbol{I}\right)$$

$\boldsymbol{x}_0$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\mathbb{X}_1 \middle| \sqrt{\alpha_1}\, \boldsymbol{x}_0, (1-\alpha_1)\, \boldsymbol{I}\right)$$



$\boldsymbol{x}_0$ $\boldsymbol{x}_1$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_2 \sim \mathcal{N}\left(\mathbb{X}_2 \middle| \sqrt{\alpha_2}\,\boldsymbol{x}_1, (1-\alpha_2)\,\boldsymbol{I}\right)$$

$\boldsymbol{x}_0$ $\boldsymbol{x}_1$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_2 \sim \mathcal{N}\left(\mathbb{X}_2 \middle| \sqrt{\alpha_2}\,\boldsymbol{x}_1, (1-\alpha_2)\,\boldsymbol{I}\right)$$

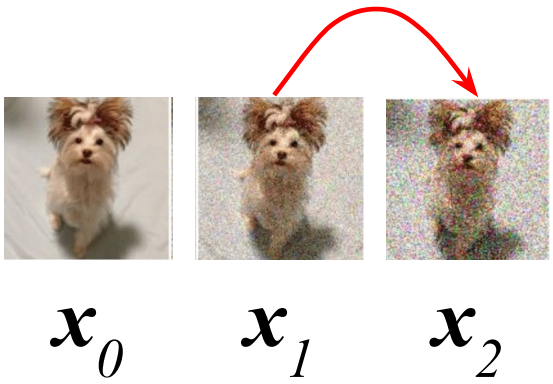


Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_3 \sim \mathcal{N}\left(\mathbb{X}_3 \middle| \sqrt{\alpha_3}\,\boldsymbol{x}_2, (1-\alpha_3)\,\boldsymbol{I}\right)$$

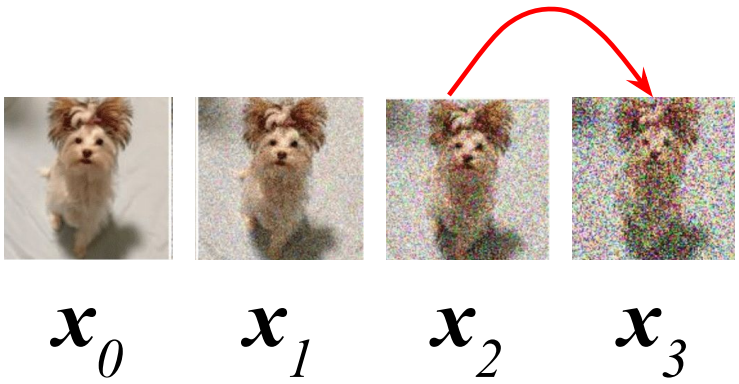


Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_4 \sim \mathcal{N}\left(\mathbb{X}_4 \middle| \sqrt{\alpha_4}\,\boldsymbol{x}_3, (1-\alpha_4)\,\boldsymbol{I}\right)$$

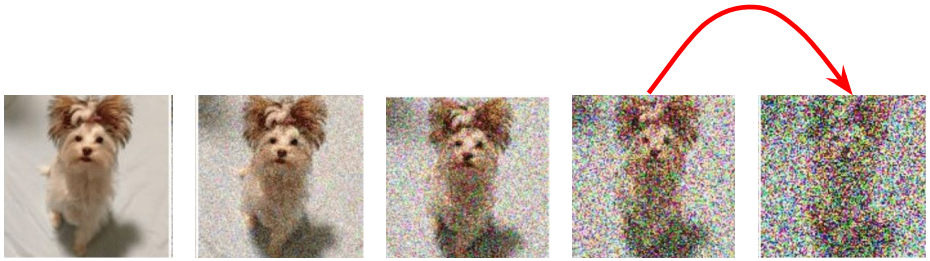


Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

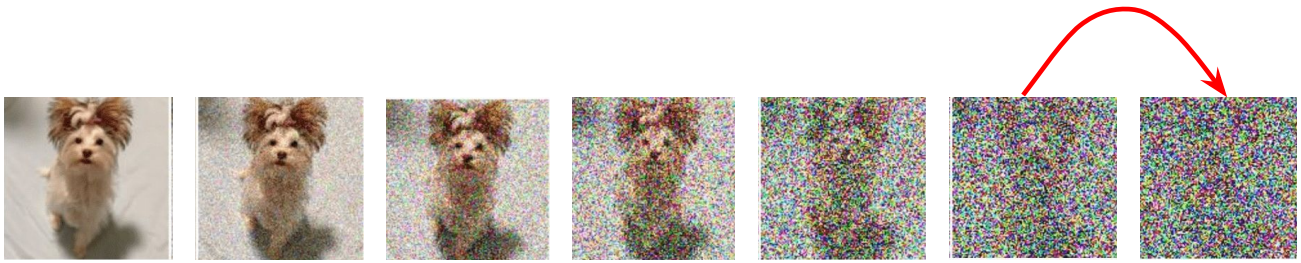
$$\boldsymbol{x}_5 \sim \mathcal{N}\left(\mathbb{X}_5 \middle| \sqrt{\alpha_5}\,\boldsymbol{x}_4, (1-\alpha_5)\,\boldsymbol{I}\right)$$

$\boldsymbol{x}_0$ $\boldsymbol{x}_1$ $\boldsymbol{x}_2$ $\boldsymbol{x}_3$ $\boldsymbol{x}_4$ $\boldsymbol{x}_5$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_6 \sim \mathcal{N}\left(\mathbb{X}_6 \middle| \sqrt{\alpha_6}\,\boldsymbol{x}_5, (1-\alpha_6)\,\boldsymbol{I}\right)$$

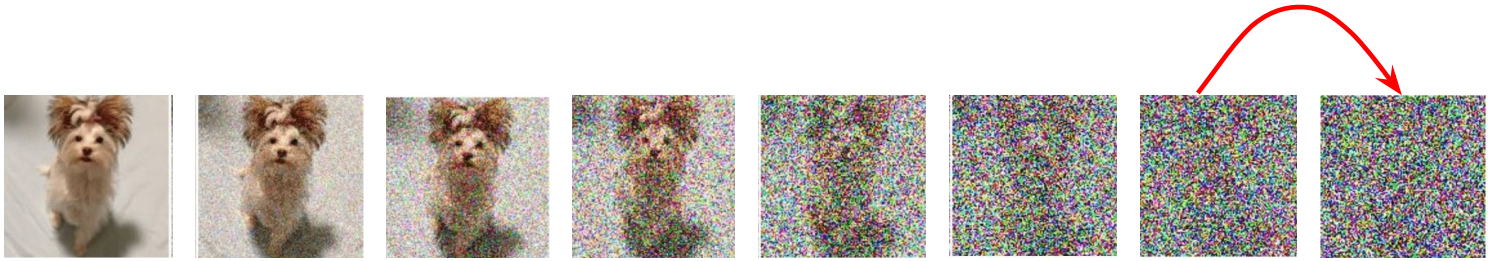


$\boldsymbol{x}_0$ $\boldsymbol{x}_1$ $\boldsymbol{x}_2$ $\boldsymbol{x}_3$ $\boldsymbol{x}_4$ $\boldsymbol{x}_5$ $\boldsymbol{x}_6$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Forward Diffusion Process (source of images: [1])

$$\boldsymbol{x}_7 \sim \mathcal{N}\left(\mathbb{X}_7 \middle| \sqrt{\alpha_7}\,\boldsymbol{x}_6, (1-\alpha_7)\,\boldsymbol{I}\right)$$



$\boldsymbol{x}_0$ $\boldsymbol{x}_1$ $\boldsymbol{x}_2$ $\boldsymbol{x}_3$ $\boldsymbol{x}_4$ $\boldsymbol{x}_5$ $\boldsymbol{x}_6$ $\boldsymbol{x}_7$

Figure: Forward Diffusion Process ($\alpha_i \lesssim 1, 1 \leq i \leq T$)

Reverse Diffusion Process (source of images: [1])

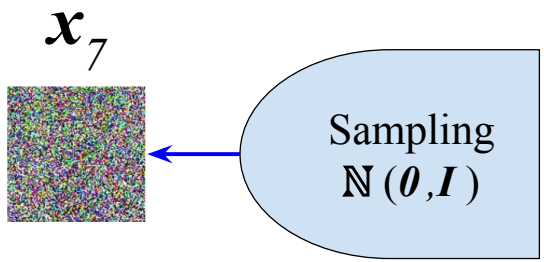
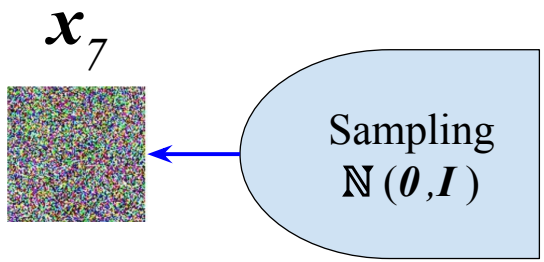


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])



$$x_6 \sim \mathcal{N}\left(\mathbb{X}_6 \middle| \mu_\theta(x_7, 7), \Sigma_\theta(x_7, 7)\right)$$

Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

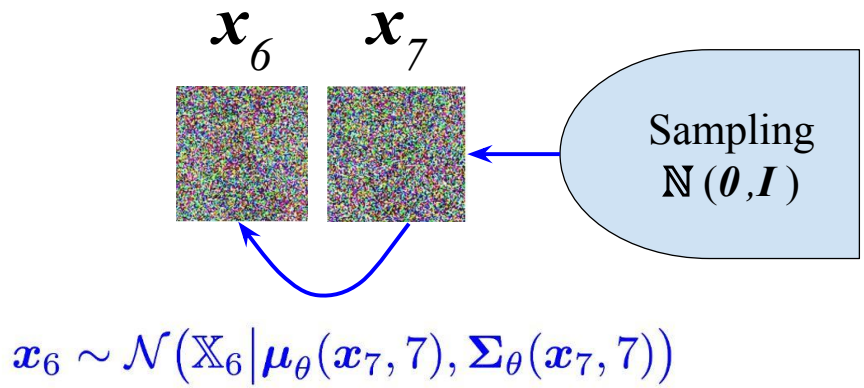


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

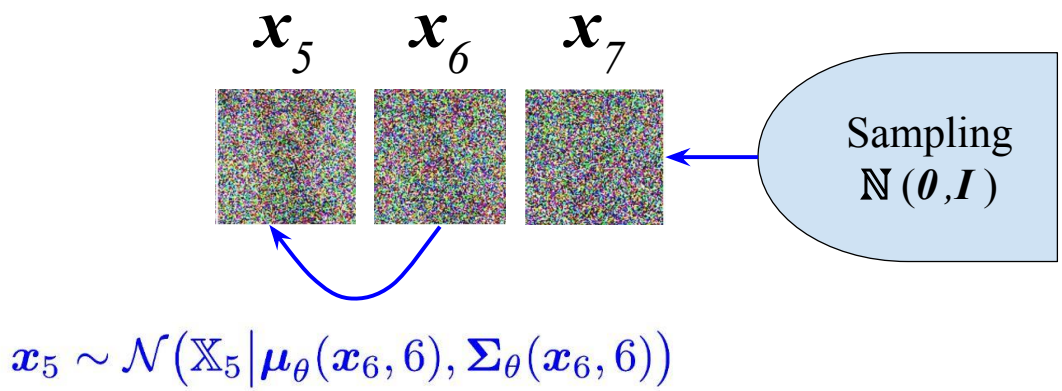


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

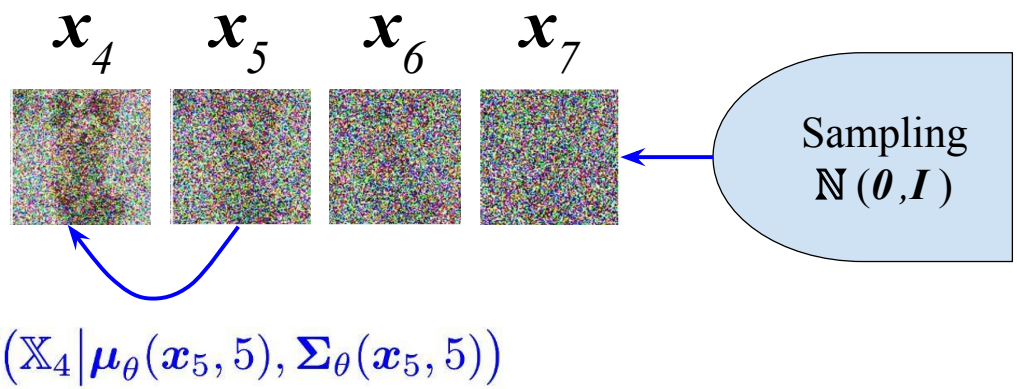


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

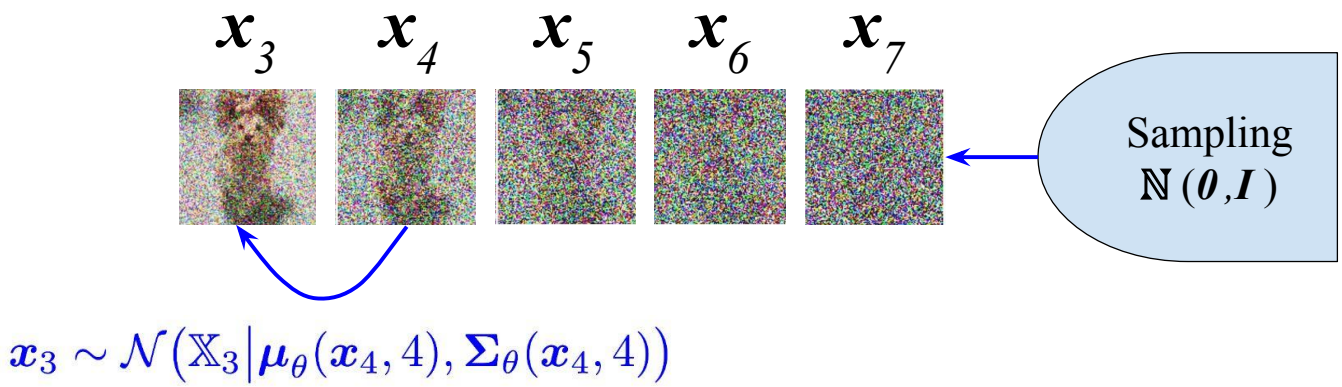


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

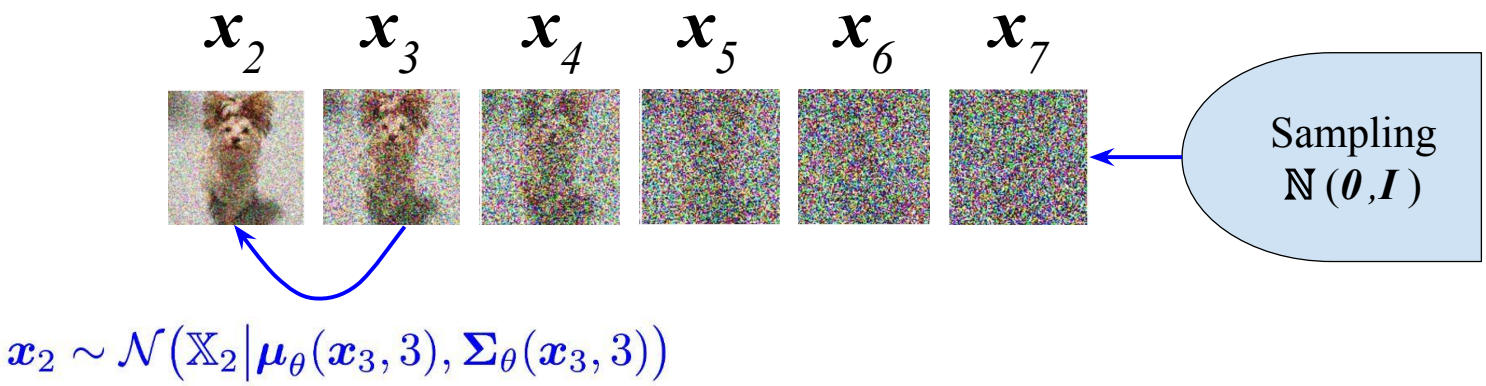


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

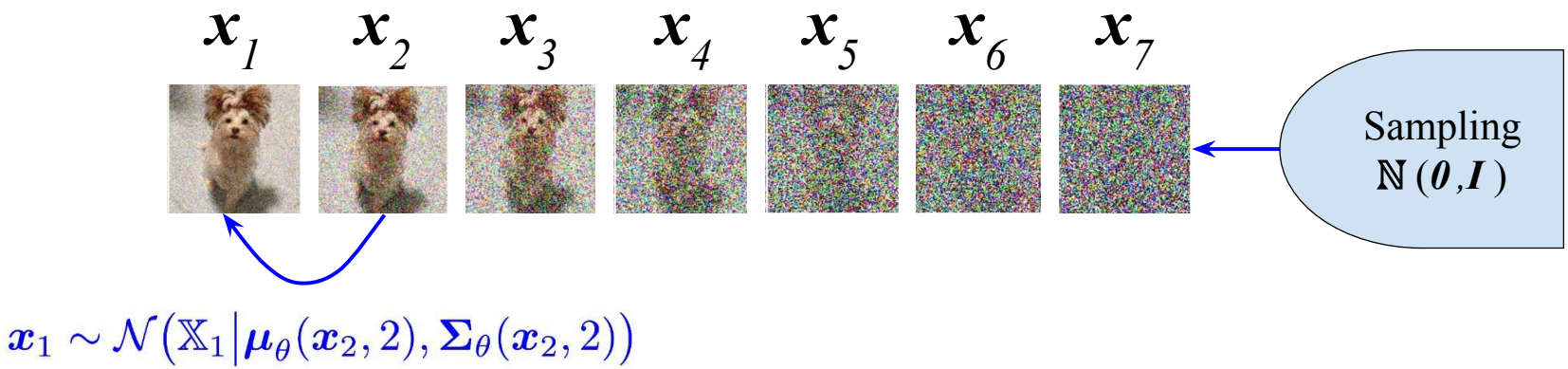


Figure: Reverse Diffusion Process

Reverse Diffusion Process (source of images: [1])

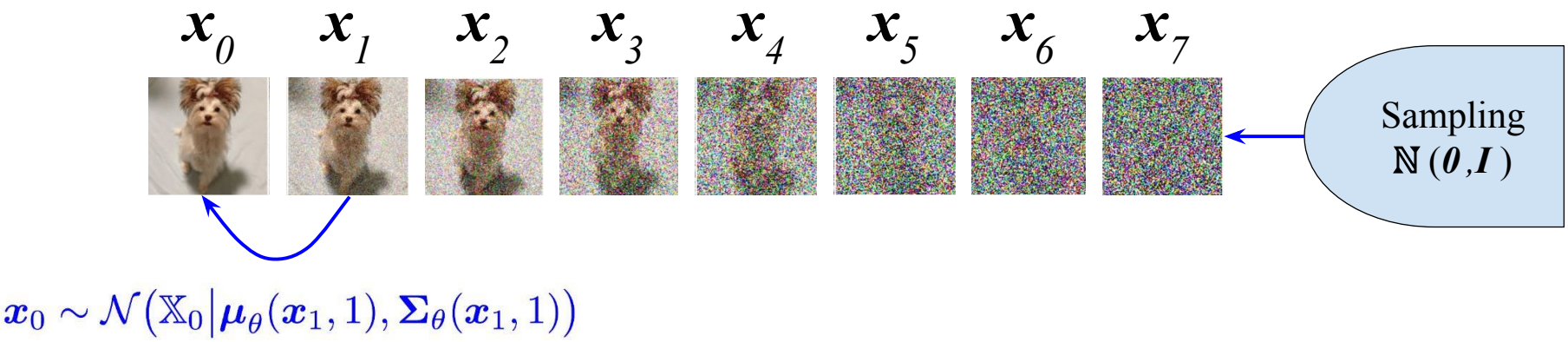


Figure: Reverse Diffusion Process

Section 5

Learning

Conditioning Trick

$$\boldsymbol{x}_t \sim \mathcal{N}\left(\mathbb{X}_t \middle| \sqrt{\alpha_t}\, \boldsymbol{x}_{t-1}, (1-\alpha_t)\, \boldsymbol{I}\right)$$

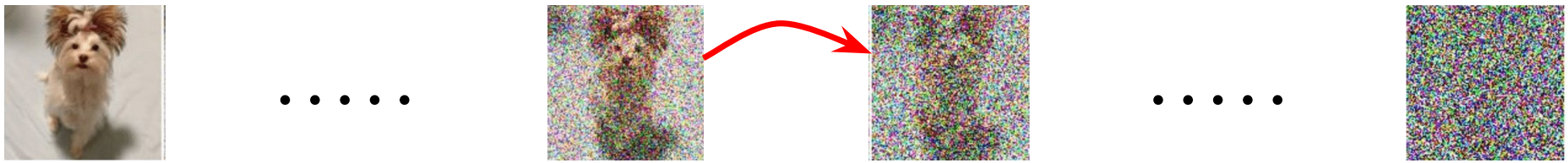
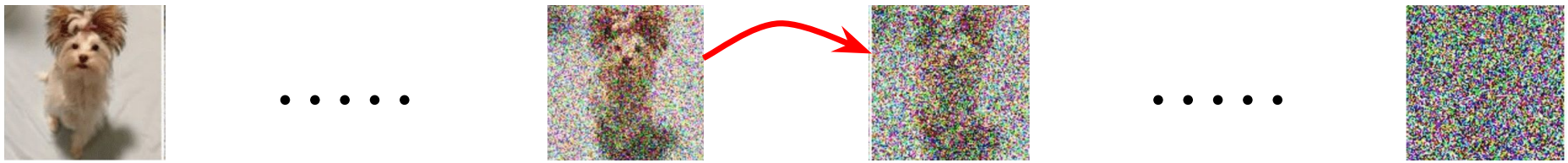


Figure: Typical transition in VDM encoder

Conditioning Trick

$$\boldsymbol{x}_t \sim \mathcal{N}\left(\mathbb{X}_t \middle| \sqrt{\alpha_t}\, \boldsymbol{x}_{t-1}, (1-\alpha_t)\, \boldsymbol{I}\right)$$

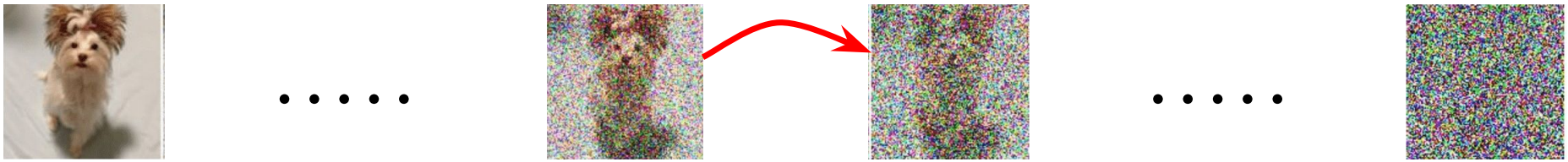


$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$$

Figure: Conditional distribution over $\boldsymbol{x}_t$ given $\boldsymbol{x}_{t-1}$

Conditioning Trick

$$\boldsymbol{x}_t \sim \mathcal{N}\left(\mathbb{X}_t \middle| \sqrt{\alpha_t}\, \boldsymbol{x}_{t-1}, (1-\alpha_t)\, \boldsymbol{I}\right)$$



$\boldsymbol{x}_0$ $\boldsymbol{x}_{t-1}$ $\boldsymbol{x}_t$ $\boldsymbol{x}_T$

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

Figure: Condition trick we use during the training phase

Conditioning Trick

Further Exploration

Based on the conditioning trick, we have:

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

Conditioning Trick

Further Exploration

Based on the conditioning trick, we have:

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

Also:

Conditioning Trick

Further Exploration

Based on the conditioning trick, we have:

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

Also:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0) = \frac{q(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}, \boldsymbol{x}_0)} \quad \#\text{Bayes Rule}$$

Conditioning Trick

Further Exploration

Based on the conditioning trick, we have:

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

Also:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0) = \frac{q(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}, \boldsymbol{x}_0)} \quad \text{\#Bayes Rule}$$

$$= \frac{q(\boldsymbol{x}_0)q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)}{q(\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)} \quad \text{\#Chain Rule}$$

Conditioning Trick

Further Exploration

Based on the conditioning trick, we have:

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

Also:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0) = \frac{q(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}, \boldsymbol{x}_0)} \quad \#\text{Bayes Rule}$$

$$= \frac{q(\boldsymbol{x}_0)q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)}{q(\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)} \quad \#\text{Chain Rule}$$

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

ELBO Maximization

Similar to VAE, we can train VDM using ELBO as:

ELBO Maximization

Similar to VAE, we can train VDM using ELBO as:

$$\log p(\boldsymbol{x}_0) \geq \text{ELBO}(\boldsymbol{x}_0; \boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \right] \quad \#\text{ELBO}$$

ELBO Maximization

Similar to VAE, we can train VDM using ELBO as:

$$\log p(\boldsymbol{x}_0) \geq \text{ELBO}(\boldsymbol{x}_0; \boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \right] \quad \#\text{ELBO}$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})} \right] \quad \#\text{Restriction 3}$$

ELBO Maximization

Similar to VAE, we can train VDM using ELBO as:

$$\log p(\boldsymbol{x}_0) \geq \text{ELBO}(\boldsymbol{x}_0; \boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \right] \quad \#\text{ELBO}$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})} \right] \quad \#\text{Restriction 3}$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} \left[\log \frac{p(\boldsymbol{x}_T) p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) \prod_{t=2}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0) \prod_{t=2}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})} \right]$$

ELBO Maximization

Similar to VAE, we can train VDM using ELBO as:

$$\log p(\boldsymbol{x}_0) \geq \text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\right] \quad \#\text{ELBO}$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)\prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})}\right] \quad \#\text{Restriction 3}$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\prod_{t=2}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)\prod_{t=2}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})}\right]$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1},\boldsymbol{x}_0)}\right] \quad \#\text{Trick}$$

ELBO Maximization (cont.)

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\frac{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)q(\boldsymbol{x}_t|\boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}}\right] \quad \#32$$

ELBO Maximization (cont.)

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\frac{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)q(\boldsymbol{x}_t|\boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}}\right] \quad \#32$$

$$= \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \frac{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right.$$

$$\left. + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]$$

ELBO Maximization (cont.)

$$\begin{aligned}
\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) =& \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{\frac{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)q(\boldsymbol{x}_t|\boldsymbol{x}_0)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}}\right] \quad \#32 \\
=& \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_1|\boldsymbol{x}_0)} + \log \frac{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right. \\
&\left. + \log \prod_{t=2}^{T} \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right] \\
=& \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)} + \sum_{t=2}^{T} \log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]
\end{aligned}$$

ELBO Maximization (cont.)

$$\begin{aligned}\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) =& \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] + \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] \\ &+ \sum_{t=2}^{T} \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]\end{aligned}$$

ELBO Maximization (cont.)

$$\begin{aligned}
\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) =& \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] + \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] \\
&+ \sum_{t=2}^{T} \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right] \\
&= \underbrace{\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right]}_{\text{Term 1}} + \underbrace{\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right]}_{\text{Term 2}} \\
&+ \underbrace{\sum_{t=2}^{T} \mathbb{E}_{q(\boldsymbol{x}_t,\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]}_{\text{Term 3}}
\end{aligned}$$

Term 1: Reconstruction

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\mathbb{X}_1 \middle| \sqrt{\alpha_1}\,\boldsymbol{x}_0, (1-\alpha_1)\,\boldsymbol{I}\right)$$

Figure: Conditional distribution over $\boldsymbol{x}_1$ given $\boldsymbol{x}_0$

Term 1: Reconstruction

$$\boldsymbol{x}_1 \sim \mathcal{N}\left(\mathbb{X}_1 \middle| \sqrt{\alpha_1}\,\boldsymbol{x}_0, (1-\alpha_1)\,\boldsymbol{I}\right)$$

Figure: Conditional distribution over $\boldsymbol{x}_1$ given $\boldsymbol{x}_0$

Term 1: Reconstruction

Justification

Based on MCE, we know:

$$\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] \simeq \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1), \boldsymbol{x}_1 \sim q(\mathbb{X}_1|\boldsymbol{x}_0)$$

Term 1: Reconstruction

Justification

Based on MCE, we know:

$$\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] \simeq \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1), \boldsymbol{x}_1 \sim q(\mathbb{X}_1|\boldsymbol{x}_0)$$

Assume that $\boldsymbol{x}_0 =$  and you need one sample from $q(\boldsymbol{x}_1|\boldsymbol{x}_0)$ for MCE

and that sample is $\boldsymbol{x}_1 =$  .

Term 1: Reconstruction

Justification

Based on MCE, we know:

$$\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] \simeq \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1), \boldsymbol{x}_1 \sim q(\mathbb{X}_1|\boldsymbol{x}_0)$$

Assume that $\boldsymbol{x}_0 =$  and you need one sample from $q(\boldsymbol{x}_1|\boldsymbol{x}_0)$ for MCE and that sample is $\boldsymbol{x}_1 =$  .Then based on the first term, a good model is such that:

 $= \underset{\boldsymbol{v}}{\operatorname{argmax}}\, p_\theta(\boldsymbol{v}|\boldsymbol{x}_1 =$  $)$

Term 1: Reconstruction

Justification

Based on MCE, we know:

$$\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] \simeq \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1), \boldsymbol{x}_1 \sim q(\mathbb{X}_1|\boldsymbol{x}_0)$$

Assume that $\boldsymbol{x}_0 =$  and you need one sample from $q(\boldsymbol{x}_1|\boldsymbol{x}_0)$ for MCE and that sample is $\boldsymbol{x}_1 =$  .Then based on the first term, a good model is such that:

 $= \underset{\boldsymbol{v}}{\operatorname{argmax}}\, p_\theta(\boldsymbol{v}|\boldsymbol{x}_1 =$  $)$

So the first term is known as *Reconstruction*.

Term 2: Prior Matching

Justification

The second term can be simplified as:

Term 2: Prior Matching

Justification

The second term can be simplified as:

$$\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] = -\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}{p(\boldsymbol{x}_T)}\right]$$

Term 2: Prior Matching

Justification

The second term can be simplified as:

$$\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] = -\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}{p(\boldsymbol{x}_T)}\right]$$

$$= -\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

Term 2: Prior Matching

Justification

The second term can be simplified as:

$$\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] = -\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}{p(\boldsymbol{x}_T)}\right]$$

$$= -\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

Thus this term checks the matching of:

- $q(\boldsymbol{x}_T|\boldsymbol{x}_0)$: Conditional distribution of last latent $\boldsymbol{x}_T$ given the starting point $\boldsymbol{x}_0$ in forward diffusion process

Term 2: Prior Matching

Justification

The second term can be simplified as:

$$\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] = -\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}{p(\boldsymbol{x}_T)}\right]$$

$$= -\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

Thus this term checks the matching of:

- $q(\boldsymbol{x}_T|\boldsymbol{x}_0)$: Conditional distribution of last latent $\boldsymbol{x}_T$ given the starting point $\boldsymbol{x}_0$ in forward diffusion process
- $p(\boldsymbol{x}_T)$: The prior over last latent vector $\boldsymbol{x}_0$

Term 2: Prior Matching

Justification

The second term can be simplified as:

$$\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\right] = -\mathbb{E}_{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\left[\log \frac{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}{p(\boldsymbol{x}_T)}\right]$$

$$= -\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

Thus this term checks the matching of:

- $q(\boldsymbol{x}_T|\boldsymbol{x}_0)$: Conditional distribution of last latent $\boldsymbol{x}_T$ given the starting point $\boldsymbol{x}_0$ in forward diffusion process
- $p(\boldsymbol{x}_T)$: The prior over last latent vector $\boldsymbol{x}_0$

Thus justifying the *Prior Matching* name.

Term 2: Prior Matching

$$q(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_T \middle| \sqrt{\overline{\alpha}_T}\, \boldsymbol{x}_0, (1-\overline{\alpha}_T)\, \boldsymbol{I}\right)$$

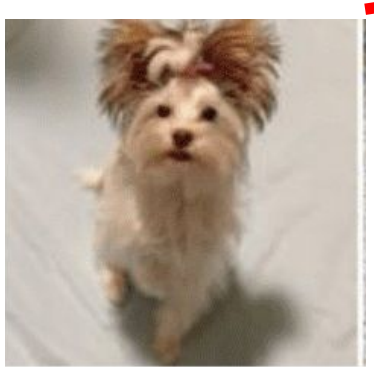


Figure: Matching the assumed prior over $p(\mathbb{X}_T)$ and what you have at the end of forward process $q(\mathbb{X}_T|\boldsymbol{x}_0)$

Term 2: Prior Matching

$$q(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_T \middle| \sqrt{\overline{\alpha}_T}\, \boldsymbol{x}_0, (1-\overline{\alpha}_T)\, \boldsymbol{I}\right)$$

Figure: Matching the assumed prior over $p(\mathbb{X}_T)$ and what you have at the end of forward process $q(\mathbb{X}_T|\boldsymbol{x}_0)$

Term 2: Prior Matching

Parameter-free Property

The second term in the optimization objective is:

$$-\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

Term 2: Prior Matching

Parameter-free Property

The second term in the optimization objective is:

$$-\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

But this term is not dependent on the model parameters θ. Why?

Term 2: Prior Matching

Parameter-free Property

The second term in the optimization objective is:

$$-\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

But this term is not dependent on the model parameters θ. Why?
We have previously assumed $\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$, thus:

$$q(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}\Big(\mathbb{X}_T|\sqrt{\bar{\alpha}_T}\ \boldsymbol{x}_0, (1-\bar{\alpha}_T)\boldsymbol{I}\Big) \simeq \mathcal{N}(\mathbb{X}_T|\mathbf{0}, \boldsymbol{I})$$

Term 2: Prior Matching

Parameter-free Property

The second term in the optimization objective is:

$$-\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

But this term is not dependent on the model parameters θ. Why?
We have previously assumed $\bar{\alpha}_T = \prod_{i=1}^{T} \alpha_t \simeq 0$, thus:

$$q(\boldsymbol{x}_T|\boldsymbol{x}_0) = \mathcal{N}\Big(\mathbb{X}_T|\sqrt{\bar{\alpha}_T}\ \boldsymbol{x}_0, (1-\bar{\alpha}_T)\boldsymbol{I}\Big) \simeq \mathcal{N}(\mathbb{X}_T|\boldsymbol{0}, \boldsymbol{I})$$

Also we assume $p(\boldsymbol{x}_T) = \mathcal{N}(\mathbb{X}_T|\boldsymbol{0}, \boldsymbol{I})$, thus we conclude:

$$\mathrm{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right) = 0$$

Term 3: Denoising Matching

Converting into KLD

Each term in the summation can be converted into an expectation over KLD as:

Term 3: Denoising Matching

Converting into KLD

Each term in the summation can be converted into an expectation over KLD as:

$$\mathbb{E}_{q(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}|\boldsymbol{x}_0)} \left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)} \right]$$

Term 3: Denoising Matching

Converting into KLD

Each term in the summation can be converted into an expectation over KLD as:

$$\mathbb{E}_{q(\boldsymbol{x}_t,\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right] \qquad \#\text{Chain rule}$$

Term 3: Denoising Matching

Converting into KLD

Each term in the summation can be converted into an expectation over KLD as:

$$\mathbb{E}_{q(\boldsymbol{x}_t,\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right] \quad \#\text{Chain rule}$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\mathbb{E}_{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]\right] \quad \#\text{Rule of total expectation}$$

Term 3: Denoising Matching

Converting into KLD

Each term in the summation can be converted into an expectation over KLD as:

$$\mathbb{E}_{q(\boldsymbol{x}_t,\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right] \quad \#\text{Chain rule}$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\mathbb{E}_{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\left[\log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}\right]\right] \quad \#\text{Rule of total expectation}$$

$$=\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[-\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

Term 3: Denoising Matching

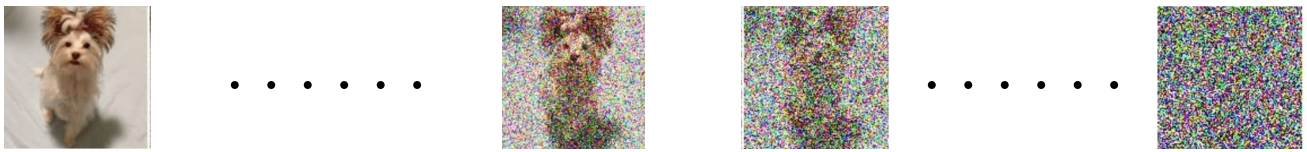


Figure: Encoder latent sequence

Term 3: Denoising Matching

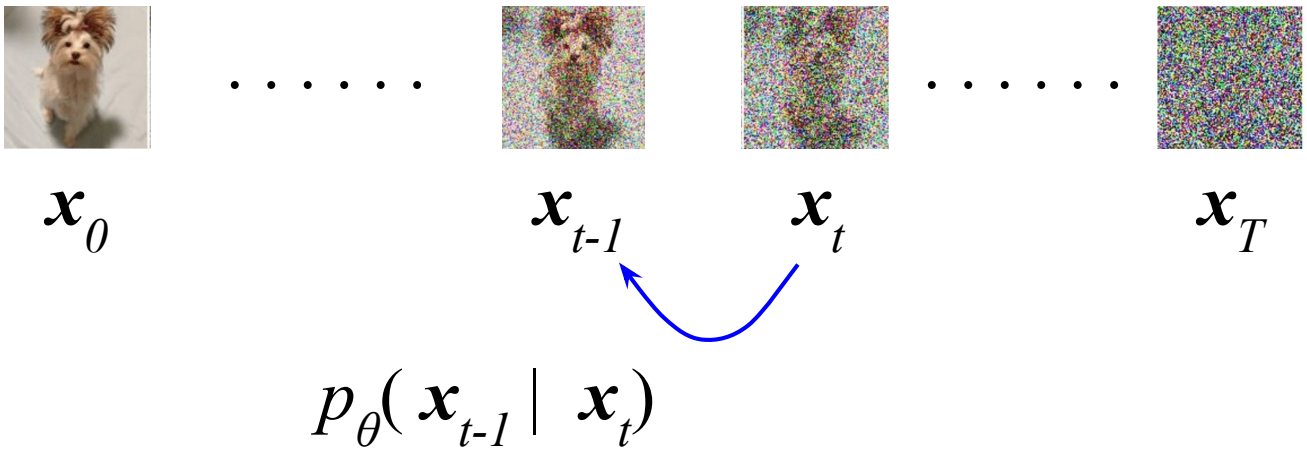


Figure: Inverse diffusion using model conditional probability

Term 3: Denoising Matching

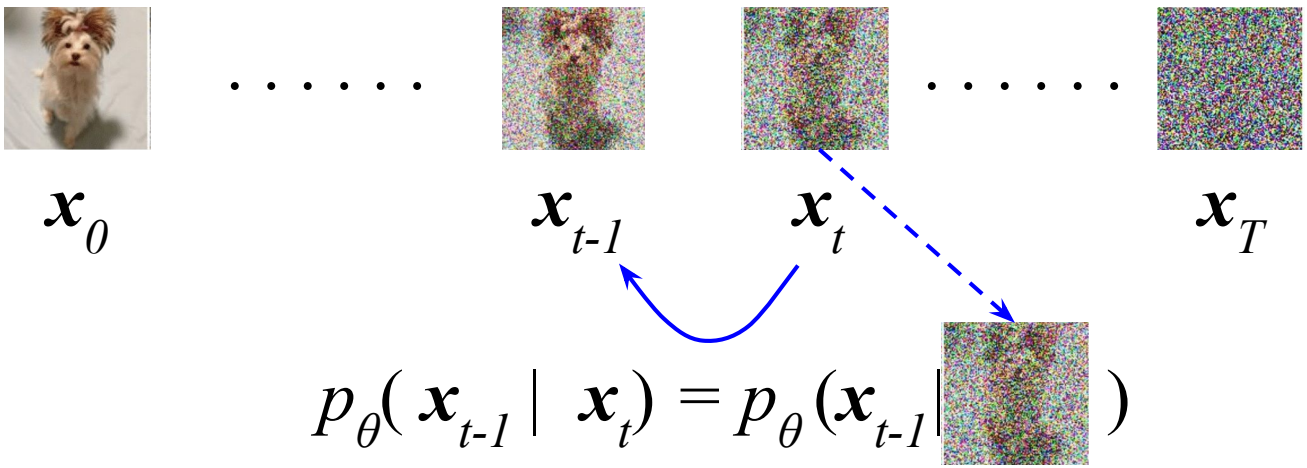


Figure: Inverse diffusion using model conditional probability

Term 3: Denoising Matching

$$q(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_0, \boldsymbol{x}_t) =$$

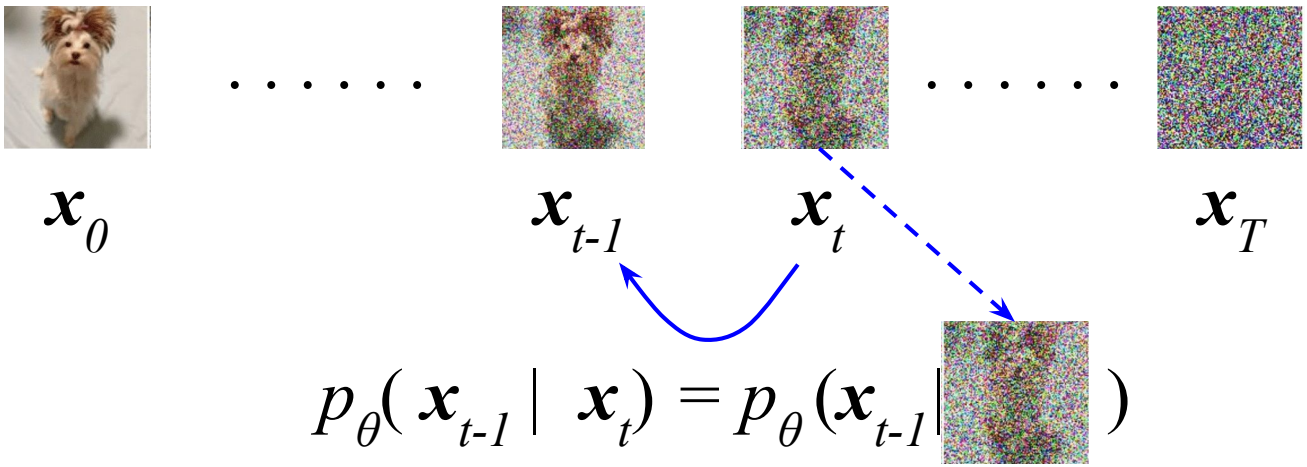


Figure: Conditional distribution appear on the RHS of conditioning trick

Term 3: Denoising Matching

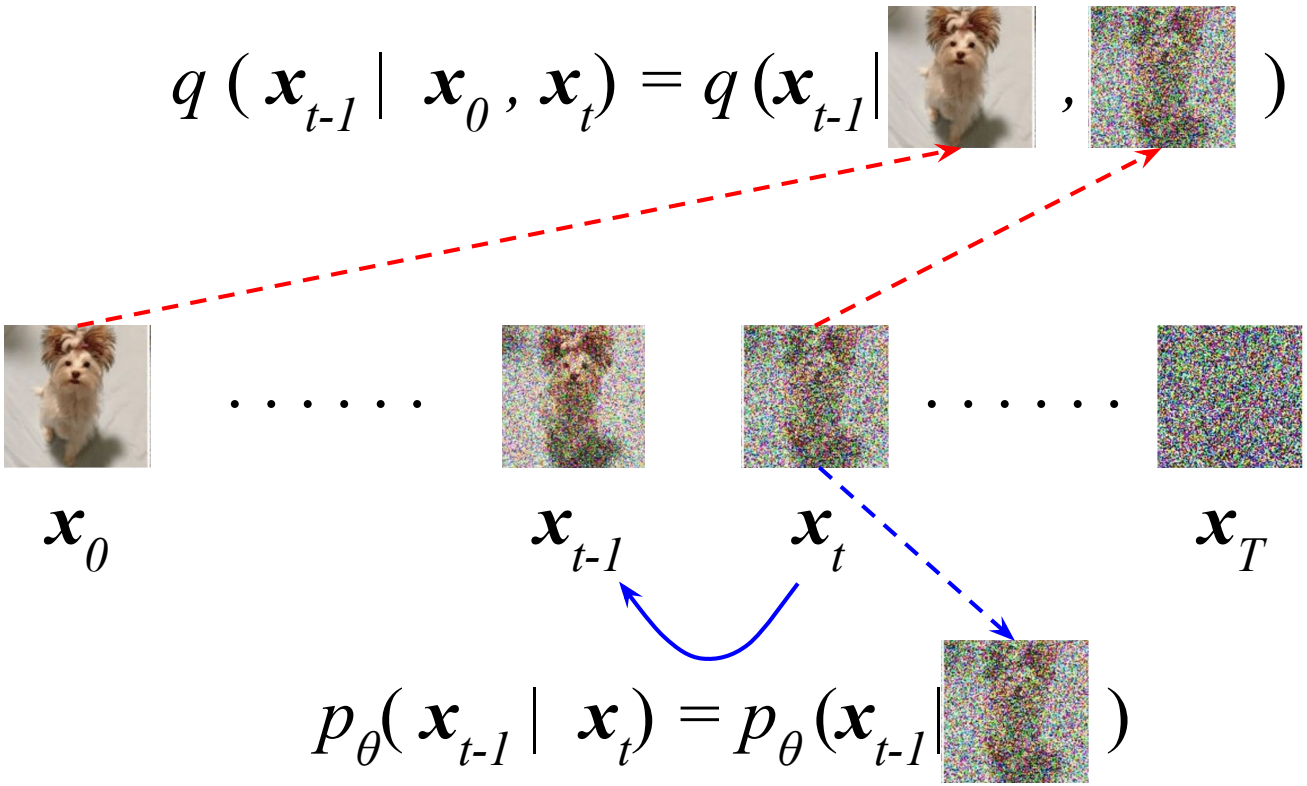


Figure: Conditional distribution appear on the RHS of conditioning trick

Term 3: Denoising Matching

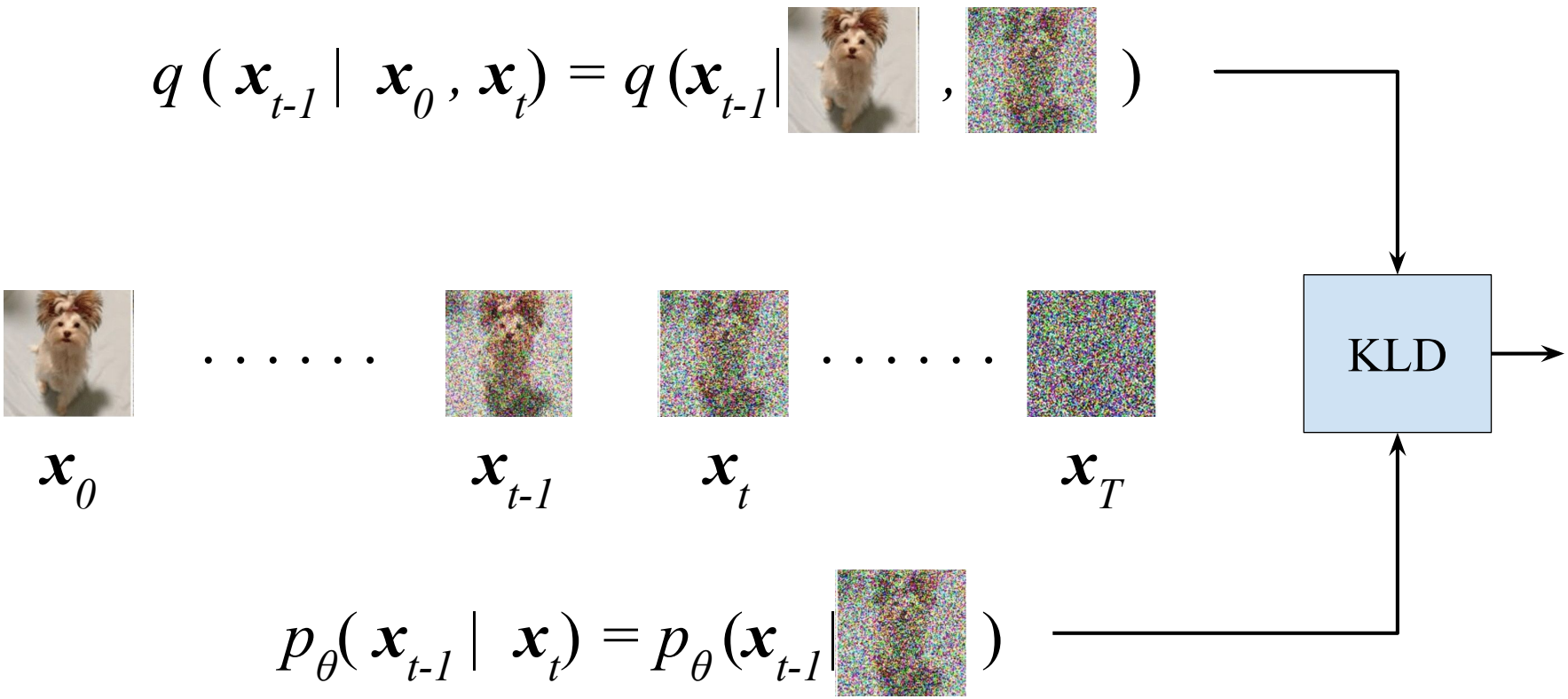


Figure: KLD betweeb two distributions

Term 3: Denoising Matching

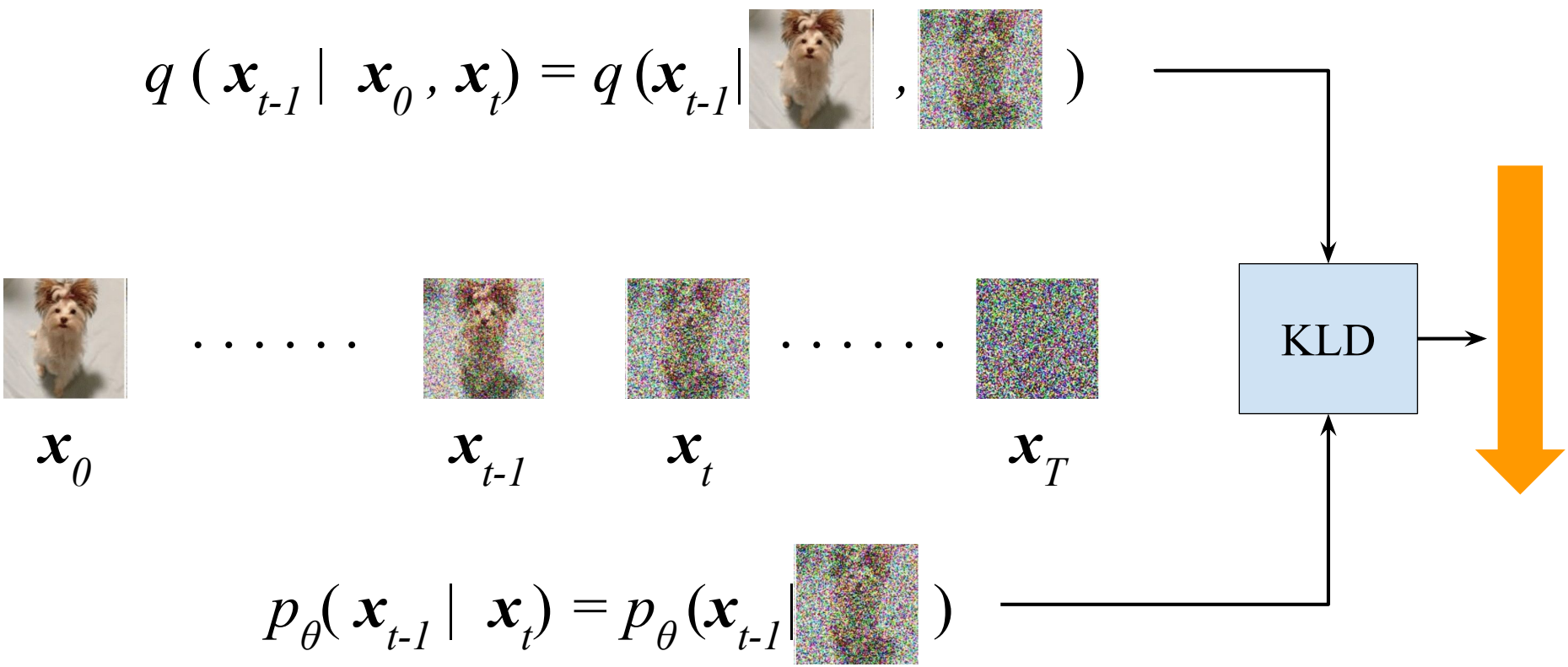


Figure: KLD minimization lead to ELBO maximization

Merging Term 1 and Term 3

Working on Term 3

We know that the value of t in term 3 starts from 2. Let's consider this term for $t = 1$:

Merging Term 1 and Term 3

Working on Term 3

We know that the value of t in term 3 starts from 2. Let's consider this term for $t = 1$:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \overset{t=1}{=} \mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right)$$

Merging Term 1 and Term 3

Working on Term 3

We know that the value of t in term 3 starts from 2. Let's consider this term for $t = 1$:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \overset{t=1}{=} \mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right)$$

But pay attention to two points:

- $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$: For this distribution, $\boldsymbol{x}_0$ is not probabilistic and you are certain about its value.

Merging Term 1 and Term 3

Working on Term 3

We know that the value of t in term 3 starts from 2. Let's consider this term for $t = 1$:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \overset{t=1}{=} \mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right)$$

But pay attention to two points:

- $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$: For this distribution, $\boldsymbol{x}_0$ is not probabilistic and you are certain about its value.
- Using the above point, we can simplify the KLD as:

$$\mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right) = -\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)$$

Merging Term 1 and Term 3

Working on Term 3

We know that the value of t in term 3 starts from 2. Let's consider this term for $t = 1$:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \overset{t=1}{=} \mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right)$$

But pay attention to two points:

- $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$: For this distribution, $\boldsymbol{x}_0$ is not probabilistic and you are certain about its value.
- Using the above point, we can simplify the KLD as:

$$\mathrm{KL}\left(q(\boldsymbol{x}_0|\boldsymbol{x}_1, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right) = -\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)$$

☞ We can consider term 1 as the term 3 for $t = 1$.

Altogether

Reframing ELBO

- We have previously seen:

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] - \text{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

$$+ \sum_{t=2}^{T} \mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[- \text{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

Altogether

Reframing ELBO

- We have previously seen:

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] - \text{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

$$+\sum_{t=2}^{T}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[-\text{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

- If we define:

$$\widehat{\mathcal{R}}(\boldsymbol{x};\theta) = \sum_{t=1}^{T}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[-\text{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

Altogether

Reframing ELBO

- We have previously seen:

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\right] - \text{KL}\left(q(\boldsymbol{x}_T|\boldsymbol{x}_0)\|p(\boldsymbol{x}_T)\right)$$

$$+\sum_{t=2}^{T}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[-\text{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

- If we define:

$$\widehat{\mathcal{R}}(\boldsymbol{x};\theta) = \sum_{t=1}^{T}\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[-\text{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$

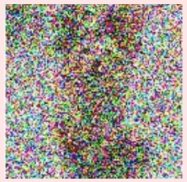
☞ Then:

$$\text{ELBO}(\boldsymbol{x}_0;\boldsymbol{\theta}) \overset{\boldsymbol{\theta}}{\equiv} \widehat{\mathcal{R}}(\boldsymbol{x}_0;\theta)$$

Denoising Matching

Matching Different Denoisers

Again pay attention to the distributions in the KLD of $\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})$:

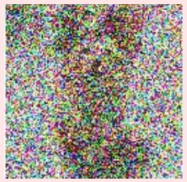
- $q\left(\boldsymbol{x}_{t-1}\middle|\ \boldsymbol{x}_0 = \right.$  $, \boldsymbol{x}_t =$  $\left.\right)$

  The above is a denoising model that can be simplified based on the forward diffusion process.

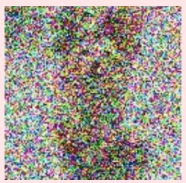
Denoising Matching

Matching Different Denoisers

Again pay attention to the distributions in the KLD of $\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})$:

- $q\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_0=\right.$  $,\boldsymbol{x}_t=$  $\left.\right)$

  The above is a denoising model that can be simplified based on the forward diffusion process.

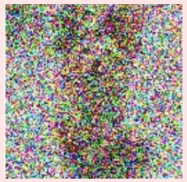
- $p_\theta\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_t=\right.$  $\left.\right)$

  The above is also a denoising where the parameterized model is used.

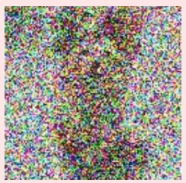
Denoising Matching

Matching Different Denoisers

Again pay attention to the distributions in the KLD of $\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})$:

- $q\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_0=\right.$  $,\boldsymbol{x}_t=$  $\left.\right)$

  The above is a denoising model that can be simplified based on the forward diffusion process.

- $p_\theta\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_t=\right.$  $\left.\right)$

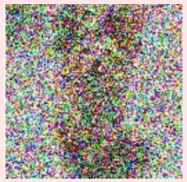
  The above is also a denoising where the parameterized model is used.

☞ We are matching two denoisers and thus the name for this term is justified.

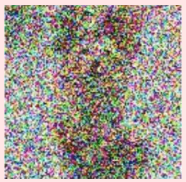
Denoising Matching

Matching Different Denoisers

Again pay attention to the distributions in the KLD of $\mathcal{R}(\boldsymbol{x};\boldsymbol{\theta})$:

- $q\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_0=\right.$  $,\boldsymbol{x}_t=$  $\left.\right)$

  The above is a denoising model that can be simplified based on the forward diffusion process.

- $p_\theta\left(\boldsymbol{x}_{t-1}\middle|\boldsymbol{x}_t=\right.$  $\left.\right)$

  The above is also a denoising where the parameterized model is used.

☞ We are matching two denoisers and thus the name for this term is justified.

☞ If you change your model parameters so that the above denoisers match, then you increase the ELBO and you can train the model!

Denoising Matching

Forward Process Denoiser

Based on the forward diffusion process, we have:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right)$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right) \qquad \bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$$

Denoising Matching

Forward Process Denoiser

Based on the forward diffusion process, we have:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right)$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right) \qquad \bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$$

Also based on the Bayes rule and conditioning trick, we have:

Denoising Matching

Forward Process Denoiser

Based on the forward diffusion process, we have:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right)$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right) \qquad \bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$$

Also based on the Bayes rule and conditioning trick, we have:

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

Denoising Matching

Forward Process Denoiser

Based on the forward diffusion process, we have:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right)$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right) \qquad \bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$$

Also based on the Bayes rule and conditioning trick, we have:

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

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

Denoising Matching

Forward Process Denoiser

Based on the forward diffusion process, we have:

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, (1-\alpha_t)\boldsymbol{I}\right)$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \mathcal{N}\left(\boldsymbol{x}_t|\sqrt{\bar{\alpha}_t}\ \boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I}\right) \qquad \bar{\alpha}_t \triangleq \prod_{i=1}^{t} \alpha_i$$

Also based on the Bayes rule and conditioning trick, we have:

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

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

All the conditional in the last ratio in RHS can be calculated using the forward diffusion process.

Denoising Matching

Forward Process Denoiser

Using the forward diffusion conditionals, we can show that:

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Denoising Matching

Forward Process Denoiser

Using the forward diffusion conditionals, we can show that:

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

where:

Denoising Matching

Forward Process Denoiser

Using the forward diffusion conditionals, we can show that:

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

where:

$$\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0, t) = \frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t + \sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\boldsymbol{x}_0}{1-\overline{\alpha}_t}$$

Denoising Matching

Forward Process Denoiser

Using the forward diffusion conditionals, we can show that:

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

where:

$$\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0, t) = \frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t + \sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\boldsymbol{x}_0}{1-\overline{\alpha}_t}$$

$$\boldsymbol{\Sigma}_q(t) = \sigma_q^2(t)\boldsymbol{I} \text{ where } \sigma_q^2(t) = \frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}$$

Denoising Matching

Forward Process Denoiser

Using the forward diffusion conditionals, we can show that:

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

where:

$$\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0, t) = \frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t + \sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\boldsymbol{x}_0}{1-\overline{\alpha}_t}$$

$$\boldsymbol{\Sigma}_q(t) = \sigma_q^2(t)\boldsymbol{I} \text{ where } \sigma_q^2(t) = \frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}$$

Modeling Inverse Diffusion

We know that when $\alpha_t \lesssim 1$, then we can model the inverse diffusion as:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t))$$

Denoising Matching

KLD Between Gaussians

Assume two D-dimensional Gaussian random vector $p(\mathbb{X}) = \mathcal{N}(\mathbb{X}|\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x)$ and $q(\mathbb{Y}) = \mathcal{N}(\mathbb{Y}|\boldsymbol{\mu}_y, \boldsymbol{\Sigma}_y)$. Then the KLD between these two distributions is:

$$\mathrm{KL}(p\|q) = \frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_y|}{|\boldsymbol{\Sigma}_x|}\right) - D + \mathrm{tr}(\boldsymbol{\Sigma}_y^{-1}\boldsymbol{\Sigma}_x) + (\boldsymbol{\mu}_y - \boldsymbol{\mu}_x)^T\boldsymbol{\Sigma}_y^{-1}(\boldsymbol{\mu}_y - \boldsymbol{\mu}_x)\right)$$

Denoising Matching

Forward Denoiser Variance

By fixing the value of α_t for $1 \leq t \leq T$, then the covariance matrix for forward denoiser can be calculated as:

$$\boldsymbol{\Sigma}_q(t) = \sigma_q^2(t)\boldsymbol{I} = \frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}\boldsymbol{I}$$

Denoising Matching

Forward Denoiser Variance

By fixing the value of α_t for $1 \leq t \leq T$, then the covariance matrix for forward denoiser can be calculated as:

$$\boldsymbol{\Sigma}_q(t) = \sigma_q^2(t)\boldsymbol{I} = \frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}\boldsymbol{I}$$

Simplifying the KLD

To simplify the KLD in the denoising matching term, we can fix covariance in the model conditional probability as:

$$\boldsymbol{\Sigma}_\theta(\boldsymbol{x}_t, t) = \boldsymbol{\Sigma}_q(t)$$

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Using Slide 48, we have (the input argument for mean and covariance are deleted for simplicity):

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Using Slide 48, we have (the input argument for mean and covariance are deleted for simplicity):

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)$$

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Using Slide 48, we have (the input argument for mean and covariance are deleted for simplicity):

$$\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_p|}{|\boldsymbol{\Sigma}_q|}\right) - D + \mathrm{tr}(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T\boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)\right)
\end{aligned}$$

Denoising Matching

Simplified KLD

In Term 3, the KLD between the following two distributions is calculated:

$$p_\theta(\mathbb{X}_{t-1}|\boldsymbol{x}_t) = \mathcal{N}(\mathbb{X}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \boldsymbol{\Sigma}_q(t))$$

$$q(\mathbb{X}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \mathcal{N}\big(\mathbb{X}_{t-1}|\boldsymbol{\mu}_q(\boldsymbol{x}_t, \boldsymbol{x}_0), \boldsymbol{\Sigma}_q(t)\big)$$

Using Slide 48, we have (the input argument for mean and covariance are deleted for simplicity):

$$\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) \\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_p|}{|\boldsymbol{\Sigma}_q|}\right) - D + \mathrm{tr}(\boldsymbol{\Sigma}_p^{-1}\boldsymbol{\Sigma}_q) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T\boldsymbol{\Sigma}_p^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)\right) \\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right) - D + \mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q) + (\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p - \boldsymbol{\mu}_q)\right)
\end{aligned}$$

Denoising Matching

Simplified KLD (Cont.)

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)$$

Simplified KLD (Cont.)

$$\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right)-D+\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q)+(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)
\end{aligned}$$

Denoising Matching

Simplified KLD (Cont.)

$$
\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right)-D+\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q)+(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2}\left(\log(1)-D+\mathrm{tr}(\boldsymbol{I})+\frac{1}{\sigma_q^2}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)
\end{aligned}
$$

Denoising Matching

Simplified KLD (Cont.)

$$
\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right)-D+\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q)+(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2}\left(\log(1)-D+\mathrm{tr}(\boldsymbol{I})+\frac{1}{\sigma_q^2}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2\sigma_q^2(t)}\|\boldsymbol{\mu}_p-\boldsymbol{\mu}_q\|_2^2,\quad \sigma_q^2(t)=\frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}
\end{aligned}
$$

Denoising Matching

Simplified KLD (Cont.)

$$
\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right)-D+\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q)+(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2}\left(\log(1)-D+\mathrm{tr}(\boldsymbol{I})+\frac{1}{\sigma_q^2}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2\sigma_q^2(t)}\|\boldsymbol{\mu}_p-\boldsymbol{\mu}_q\|_2^2,\quad \sigma_q^2(t)=\frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}
\end{aligned}
$$

So minimizing KLD reduces to minimizing the scaled ℓ_2 distance of denoisers mean.

Simplified KLD (Cont.)

$$
\begin{aligned}
&\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
=&\frac{1}{2}\left(\log\left(\frac{|\boldsymbol{\Sigma}_q|}{|\boldsymbol{\Sigma}_q|}\right)-D+\mathrm{tr}(\boldsymbol{\Sigma}_q^{-1}\boldsymbol{\Sigma}_q)+(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\sigma_q^2\boldsymbol{I})^{-1}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2}\left(\log(1)-D+\mathrm{tr}(\boldsymbol{I})+\frac{1}{\sigma_q^2}(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)^T(\boldsymbol{\mu}_p-\boldsymbol{\mu}_q)\right)\\
=&\frac{1}{2\sigma_q^2(t)}\|\boldsymbol{\mu}_p-\boldsymbol{\mu}_q\|_2^2,\quad \sigma_q^2(t)=\frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}
\end{aligned}
$$

So minimizing KLD reduces to minimizing the scaled ℓ_2 distance of denoisers mean.

- ☞ The scale is a function of your selected α_t values.

Simplified KLD (Cont.)

We reach the following point:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\Big\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)=\frac{1}{2\sigma_q^2}\|\boldsymbol{\mu}_p(\boldsymbol{x}_t,t)-\boldsymbol{\mu}_q(\boldsymbol{x}_t,\boldsymbol{x}_0,t)\|_2^2$$

Simplified KLD (Cont.)

We reach the following point:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\Big\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)=\frac{1}{2\sigma_q^2}\|\boldsymbol{\mu}_p(\boldsymbol{x}_t,t)-\boldsymbol{\mu}_q(\boldsymbol{x}_t,\boldsymbol{x}_0,t)\|_2^2$$

where:

$$\boldsymbol{\mu}_q(\boldsymbol{x}_t,\boldsymbol{x}_0,t)=\frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t+\sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\boldsymbol{x}_0}{1-\overline{\alpha}_t}$$

Simplified KLD (Cont.)

We reach the following point:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\Big\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)=\frac{1}{2\sigma_q^2}\|\boldsymbol{\mu}_p(\boldsymbol{x}_t,t)-\boldsymbol{\mu}_q(\boldsymbol{x}_t,\boldsymbol{x}_0,t)\|_2^2$$

where:

$$\boldsymbol{\mu}_q(\boldsymbol{x}_t,\boldsymbol{x}_0,t)=\frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t+\sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\boldsymbol{x}_0}{1-\overline{\alpha}_t}$$

Because we have access to $\boldsymbol{x}_t$ in $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$, one suitable option to formulate $\boldsymbol{\mu}_p(\boldsymbol{x}_t,t)$ is:

$$\boldsymbol{\mu}_p(\boldsymbol{x}_t,t)=\frac{\sqrt{\alpha_t}(1-\overline{\alpha}_{t-1})\boldsymbol{x}_t+\sqrt{\overline{\alpha}_{t-1}}(1-\alpha_t)\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)}{1-\overline{\alpha}_t}$$

Simplified KLD (Cont.)

So using the suggested form for $\boldsymbol{\mu}_p(\boldsymbol{x}_t, t)$ in Slide 52, we have:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) = \underbrace{\frac{1}{2\sigma_q^2}\frac{\overline{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\overline{\alpha}_t)^2}}_{\beta_t}\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|_2^2$$

Simplified KLD (Cont.)

So using the suggested form for $\boldsymbol{\mu}_p(\boldsymbol{x}_t, t)$ in Slide 52, we have:

$$\mathrm{KL}\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)\middle\|p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right) = \underbrace{\frac{1}{2\sigma_q^2}\frac{\overline{\alpha}_{t-1}(1-\alpha_t)^2}{(1-\overline{\alpha}_t)^2}}_{\beta_t}\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|_2^2$$

and the final optimization objective is:

$$\widehat{\mathcal{R}}(\boldsymbol{x}_0; \boldsymbol{\theta}) = \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\big[-\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|_2^2\big]$$

Working on Objective Function

Reframing Objective Function

We can write:

Working on Objective Function

Reframing Objective Function

We can write:

$$\widehat{\mathcal{L}}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{t\sim U\{1,\dots,T\}}\left[\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]\right]$$

Working on Objective Function

Reframing Objective Function

We can write:

$$\widehat{\mathcal{L}}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{t\sim U\{1,\dots,T\}}\left[\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]\right]$$

$$= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]$$

Working on Objective Function

Reframing Objective Function

We can write:

$$\widehat{\mathcal{L}}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{t\sim U\{1,\dots,T\}}\left[\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]\right]$$

$$= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]$$

$$= \frac{-\widehat{\mathcal{R}}(\boldsymbol{x}_0;\boldsymbol{\theta})}{T}$$

Working on Objective Function

Reframing Objective Function

We can write:

$$\widehat{\mathcal{L}}(\boldsymbol{x}_0;\boldsymbol{\theta}) = \mathbb{E}_{t\sim U\{1,\dots,T\}}\left[\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]\right]$$

$$= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[\beta_t\|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t,t)-\boldsymbol{x}_0\|_2^2\right]$$

$$= \frac{-\widehat{\mathcal{R}}(\boldsymbol{x}_0;\boldsymbol{\theta})}{T}$$

☞ Thus we can minimize $\widehat{\mathcal{L}}(\boldsymbol{x}_0;\boldsymbol{\theta})$ instead of maximizing $\widehat{\mathcal{R}}(\boldsymbol{x}_0;\boldsymbol{\theta})$.

ELBO Over Dataset

Complete Objective

The final objective function for training VDM result by minimizing $\widehat{\mathcal{L}}(\boldsymbol{x}_0; \boldsymbol{\theta})$ over the complete dataset which can be written as an expectation:

ELBO Over Dataset

Complete Objective

The final objective function for training VDM result by minimizing $\widehat{\mathcal{L}}(\boldsymbol{x}_0; \boldsymbol{\theta})$ over the complete dataset which can be written as an expectation:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}_0 \sim p_{\text{data}}(\mathbb{X})} \left[\widehat{\mathcal{L}}(\boldsymbol{x}_0; \boldsymbol{\theta}) \right]$$

ELBO Over Dataset

Complete Objective

The final objective function for training VDM result by minimizing $\widehat{\mathcal{L}}(\boldsymbol{x}_0; \boldsymbol{\theta})$ over the complete dataset which can be written as an expectation:

$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x}_0 \sim p_{\text{data}}(\mathbb{X})} \left[\widehat{\mathcal{L}}(\boldsymbol{x}_0; \boldsymbol{\theta}) \right]$$

$$= \mathbb{E}_{\boldsymbol{x}_0 \sim p_{\text{data}}(\mathbb{X})} \mathbb{E}_{t \sim U\{1,\dots,T\}} \mathbb{E}_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)} \beta_t \|\widehat{\boldsymbol{x}}_\theta(\boldsymbol{x}_t, t) - \boldsymbol{x}_0\|_2^2$$

References I

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole,
"Score-based generative modeling through stochastic differential equations,"
arXiv preprint arXiv:2011.13456, 2020.

W. Feller,
"On the theory of stochastic processes, with particularreference to applications,"
in *Berkeley Symposium on Mathematical Statisticsand Probability*. The Regents of the University of California, 1949.