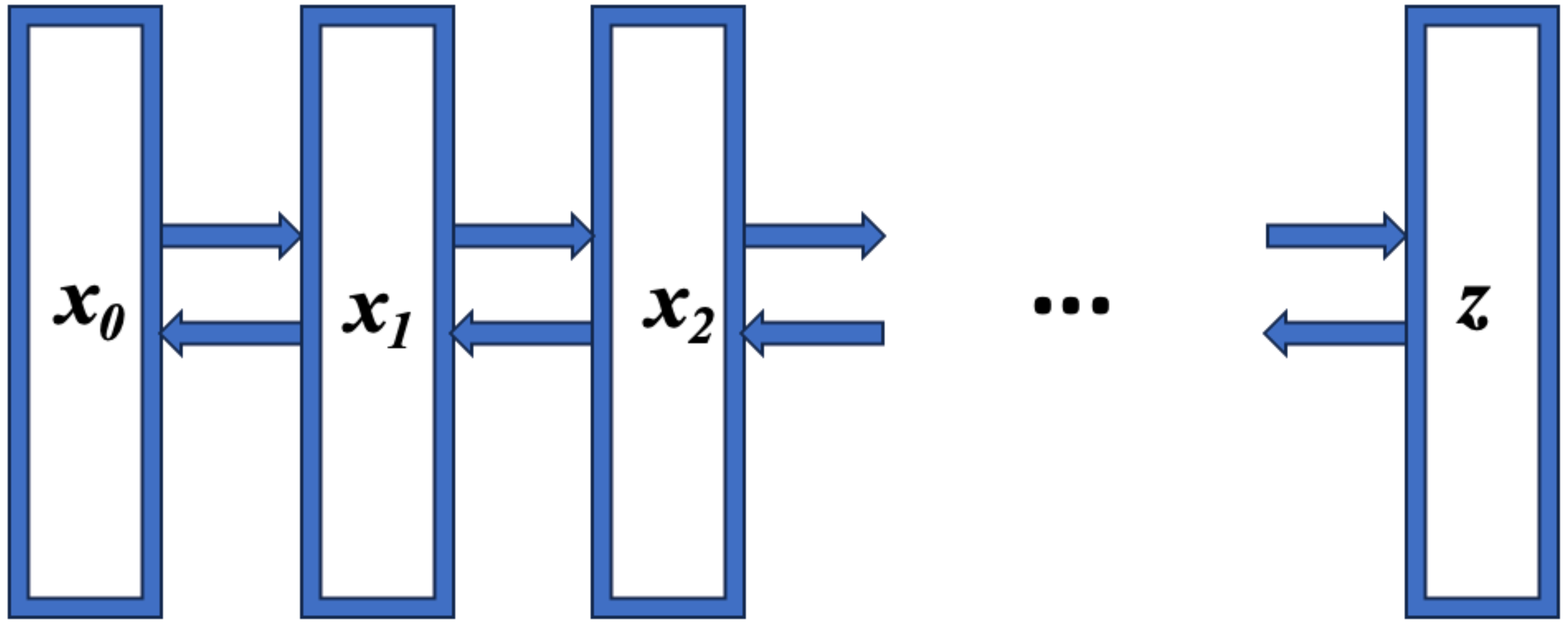
Lecture 2. Diffusion models

Introduction to Bayesian statistical Learning II

04.06.2024 Instructors: Alina Bazarova, Oleg Filatov, Sabrina Benassou

Denoising Diffusion Probabilistic Models (DDPM)

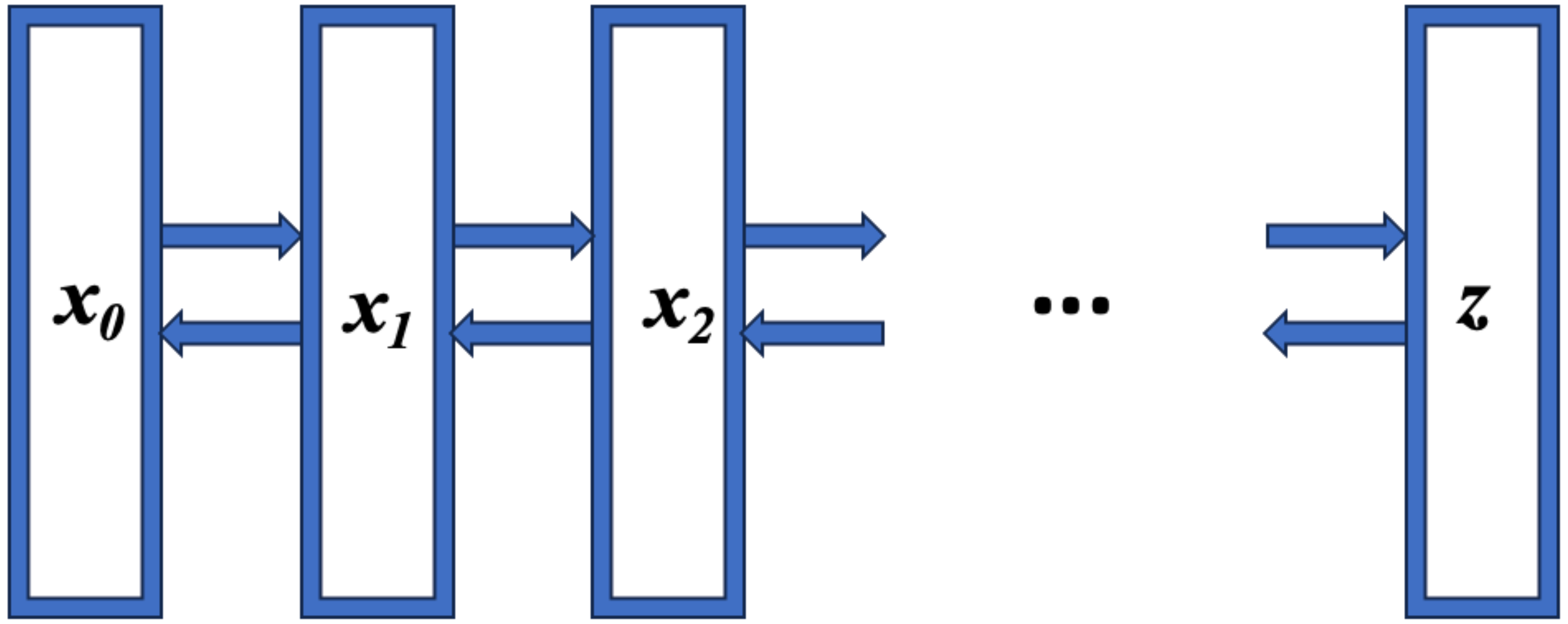
Another type of generative models. What do they have to do with Bayes?



1. Forward diffusion process: gradually **adding noise** to samples

Denoising Diffusion Probabilistic Models (DDPM)

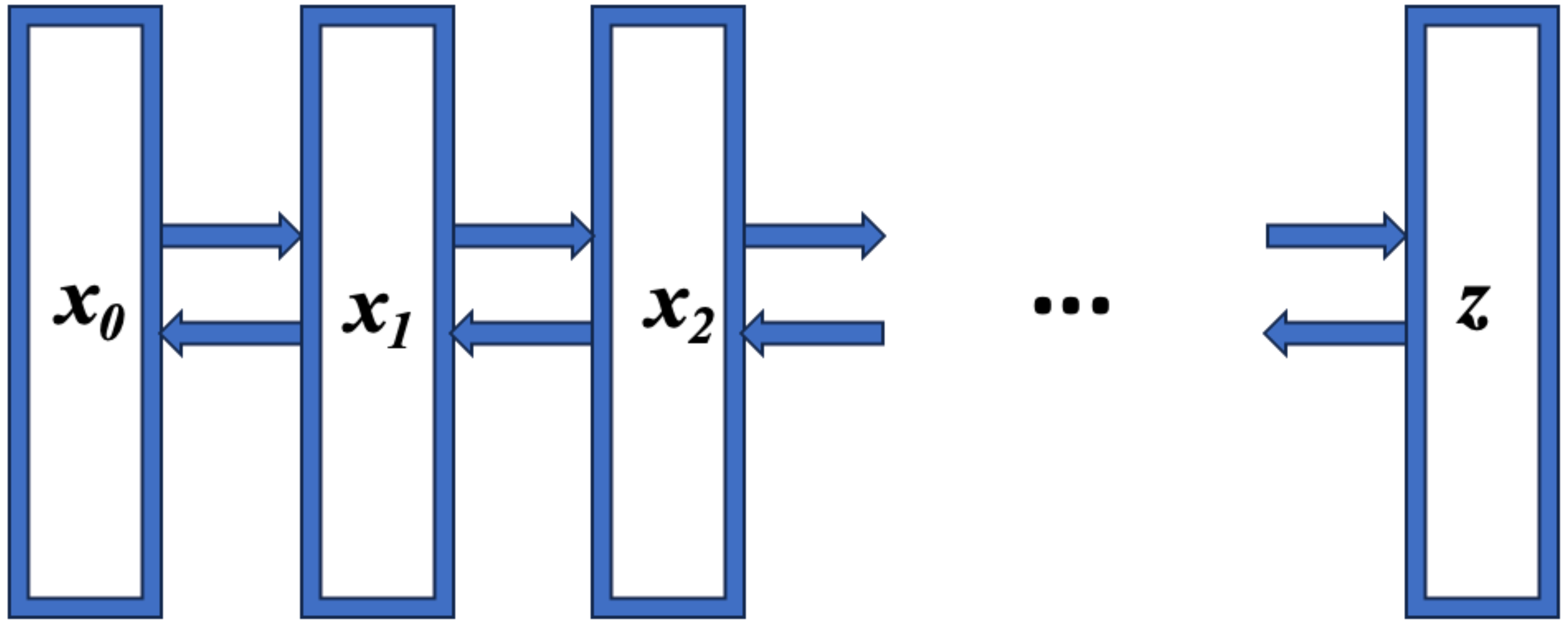
Another type of generative models. What do they have to do with Bayes?



1. Forward diffusion process: gradually **adding noise** to samples
2. Reverse diffusion process: **recreating** the sample **from noise**

Denoising Diffusion Probabilistic Models (DDPM)

Another type of generative models. What do they have to do with Bayes?



1. Forward diffusion process: gradually **adding noise** to samples
2. Reverse diffusion process: **recreating** the sample **from noise**
3. Heavily relying on conditional probability and **Bayes theorem** in particular

Forward diffusion process

Let $x_0 \sim q(x)$ be a real data distribution.

We produce a sequence of **noisy samples** $x_1, \ldots, x_T$

Forward diffusion process

Let $x_0 \sim q(x)$ be a real data distribution.

We produce a sequence of **noisy samples** $x_1, \ldots, x_T$

Noise steps controlled by a **variance schedule** $\{\beta_t \in (0,1)\}_{t=1}^{T}$

Forward diffusion process

Let $x_0 \sim q(x)$ be a real data distribution.

We produce a sequence of **noisy samples** $x_1, \ldots, x_T$

Noise steps controlled by a **variance schedule** $\{\beta_t \in (0,1)\}_{t=1}^{T}$

$$q(x_t \,|\, x_{t-1}) \sim \mathcal{N}(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}), \qquad q(x_{1:T} \,|\, x_0) = \prod_{t=1}^{T} q(x_t \,|\, x_{t-1})$$

Forward diffusion process

Let $x_0 \sim q(x)$ be a real data distribution.

We produce a sequence of **noisy samples** $x_1, \ldots, x_T$

Noise steps controlled by a **variance schedule** $\{\beta_t \in (0,1)\}_{t=1}^{T}$

$$q(x_t \mid x_{t-1}) \sim \mathcal{N}(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t \mathbf{I}), \qquad q(x_{1:T} \mid x_0) = \prod_{t=1}^{T} q(x_t \mid x_{t-1})$$

As $T \rightarrow \infty$, x_T is equivalent to a **Gaussian distribution**

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1 - \alpha_t} \varepsilon_{t-1} = \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \bar{\varepsilon}_{t-2}$$

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1 - \alpha_t} \varepsilon_{t-1} = \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \bar{\varepsilon}_{t-2}$$

NB: $\sqrt{\alpha_t}\sqrt{1 - \alpha_t} \varepsilon_{t-1} + \sqrt{1 - \alpha_{t-1}} \varepsilon_{t-2}, \qquad \varepsilon_i \sim \mathcal{N}(0, \mathbf{I})$, hence

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1 - \alpha_t} \varepsilon_{t-1} = \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \bar{\varepsilon}_{t-2}$$

NB: $\sqrt{\alpha_t}\sqrt{1 - \alpha_t}\varepsilon_{t-1} + \sqrt{1 - \alpha_{t-1}}\varepsilon_{t-2}, \qquad \varepsilon_i \sim \mathcal{N}(0, \mathbf{I})$, hence

$$\sqrt{\alpha_{t-1}}\sqrt{1 - \alpha_t}\varepsilon_{t-1} + \sqrt{1 - \alpha_{t-1}}\varepsilon_{t-2} \sim \mathcal{N}(0, \alpha_{t-1} - \alpha_t \alpha_{t-1} + 1 - \alpha_{t-1})$$

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1 - \alpha_t} \varepsilon_{t-1} = \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \bar{\varepsilon}_{t-2}$$

NB: $\sqrt{\alpha_t}\sqrt{1 - \alpha_t}\varepsilon_{t-1} + \sqrt{1 - \alpha_{t-1}}\varepsilon_{t-2}, \qquad \varepsilon_i \sim \mathcal{N}(0, \mathbf{I})$, hence

$$\sqrt{\alpha_{t-1}}\sqrt{1 - \alpha_t}\varepsilon_{t-1} + \sqrt{1 - \alpha_{t-1}}\varepsilon_{t-2} \sim \mathcal{N}(0, \cancel{\alpha}_{t-1} - \alpha_t \alpha_{t-1} + 1 - \cancel{\alpha}_{t-1})$$

Forward diffusion process

Can we sample x_t at any arbitrary time step t ? Yes!!!

The key: good old **reparametrisation trick**!

Let $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \sqrt{\alpha_t \alpha_{t-1}} x_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\varepsilon}_{t-2}$$

NB: $\sqrt{\alpha_t}\sqrt{1-\alpha_t}\varepsilon_{t-1} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2}, \qquad \varepsilon_i \sim \mathcal{N}(0,\mathbf{I})$, hence

$$\sqrt{\alpha_{t-1}}\sqrt{1-\alpha_t}\varepsilon_{t-1} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2} \sim \mathcal{N}(0, \alpha_{t-1} - \alpha_t\alpha_{t-1} + 1 - \alpha_{t-1})$$

$$\sqrt{\alpha_{t-1}}\sqrt{1-\alpha_t}\varepsilon_{t-1} + \sqrt{1-\alpha_{t-1}}\varepsilon_{t-2} \sim \sqrt{1-\alpha_t\alpha_{t-1}}\mathcal{N}(0,\mathbf{I})$$

Forward diffusion process

Following the same argument

$$x_t = \sqrt{\alpha_t} x_{t-1} + \sqrt{1 - \alpha_t} \varepsilon_{t-1} = \ldots = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \mathbf{I}), \quad \bar{\alpha}_t = \prod_{t=1}^{t} \alpha_i$$

Forward diffusion process

Following the same argument

$$x_t \sqrt{\alpha_t} x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \ldots = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,\mathbf{I}), \quad \bar{\alpha}_t = \prod_{t=1}^{t} \alpha_i$$

Hence $q(x_t \mid x_0) \sim \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1-\bar{\alpha}_t \mathbf{I}))$

Forward diffusion process

Following the same argument

$$x_t\sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}\varepsilon_{t-1} = \ldots = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon, \quad \varepsilon \sim \mathcal{N}(0,\mathbf{I}), \quad \bar{\alpha}_t = \prod_{t=1}^{t}\alpha_i$$

Hence $q(x_t \,|\, x_0) \sim \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t\mathbf{I}))$

Larger update step when the sample gets noisier: $\beta_1 < \beta_2 < \ldots < \beta_T$,

and therefore $\bar{\alpha}_1 > \ldots > \bar{\alpha}_T$

Reverse diffusion process

If we reverse the above process and sample from $q(x_{t-1} | x_t)$,

we can recreate a sample from $x_T \sim \mathcal{N}(0, \mathbf{I})$

We need to estimate $q(x_{t-1} | x_t)$. We do it with another probability density function p_θ

Reverse diffusion process

If we reverse the above process and sample from $q(x_{t-1} | x_t)$,

we can recreate a sample from $x_T \sim \mathcal{N}(0,\mathbf{I})$

We need to estimate $q(x_{t-1} | x_t)$. We do it with another probability density function p_θ

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1} | x_t),$$

Reverse diffusion process

If we reverse the above process and sample from $q(x_{t-1}|x_t)$,

we can recreate a sample from $x_T \sim \mathcal{N}(0,\mathbf{I})$

We need to estimate $q(x_{t-1}|x_t)$. We do it with another probability density function p_θ

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \quad p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

Which is learned via a neural network (θ are the parameters of NN)!

Reverse diffusion process

If we reverse the above process and sample from $q(x_{t-1} | x_t)$,

we can recreate a sample from $x_T \sim \mathcal{N}(0,\mathbf{I})$

We need to estimate $q(x_{t-1} | x_t)$. We do it with another probability density function p_θ

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1} | x_t), \quad p_\theta(x_{t-1} | x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

Which is learned via a neural network (θ are the parameters of NN)!

Important: $q(x_{t-1} | x_t, x_0) \sim \mathcal{N}(x_{t-1}; \tilde{\mu}(x_t, x_0), \tilde{\beta}\mathbf{I})$ **tractable!**

Conditioning on x_0. Bayes rule to the rescue! $P(A \mid B) = \dfrac{P(B)P(B \mid A)}{P(A)}$

Conditioning on x_0. Bayes rule to the rescue! $P(A \mid B) = \dfrac{P(B)P(B \mid A)}{P(A)}$

$$P(A \mid B, C) = = \frac{P(AB \mid C)}{P(B \mid C)} = \frac{P(B \mid A, C)P(A \mid C)}{P(B \mid C)}$$

Conditioning on x_0. Bayes rule to the rescue! $P(A|B) = \dfrac{P(B)P(B|A)}{P(A)}$

$$P(A|B,C) == \frac{P(AB|C)}{P(B|C)} = \frac{P(B|A,C)P(A|C)}{P(B|C)}$$

$$q(x_{t-1}|x_t,x_0) = q(x_t|x_{t-1},x_0)\frac{q(x_{t-1}|x_0)}{q(x_t|x_0)} \propto \exp(-\frac{1}{2}(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t-\sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}))$$

Conditioning on x_0. Bayes rule to the rescue! $P(A|B) = \dfrac{P(B)P(B|A)}{P(A)}$

$$P(A|B,C) = = \frac{P(AB|C)}{P(B|C)} = \frac{P(B|A,C)P(A|C)}{P(B|C)}$$

$$q(x_{t-1}|x_t,x_0) = q(x_t|x_{t-1},x_0)\frac{q(x_{t-1}|x_0)}{q(x_t|x_0)} \propto \exp(-\frac{1}{2}(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t-\sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}))$$

$$= \exp(-\frac{1}{2}((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})x_{t-1}^2 - 2(\frac{\sqrt{\alpha_t}}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)x_{t-1} + C(x_0,x_1)))$$

Conditioning on x_0. Bayes rule to the rescue! $P(A|B) = \dfrac{P(B)P(B|A)}{P(A)}$

$$P(A|B,C) = = \frac{P(AB|C)}{P(B|C)} = \frac{P(B|A,C)P(A|C)}{P(B|C)}$$

$$q(x_{t-1}|x_t,x_0) = q(x_t|x_{t-1},x_0)\frac{q(x_{t-1}|x_0)}{q(x_t|x_0)} \propto \exp(-\frac{1}{2}(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t-\sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}))$$

$$= \exp(-\frac{1}{2}((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})x_{t-1}^2 - 2(\frac{\sqrt{\alpha_t}}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)x_{t-1} + C(x_0,x_1)))$$

Hence $\tilde{\beta}_t = 1/(\dfrac{\alpha_t}{\beta_t} + \dfrac{1}{1-\bar{\alpha}_{t-1}}) = \dfrac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t,$

Conditioning on x_0. Bayes rule to the rescue! $P(A \mid B) = \dfrac{P(B)P(B \mid A)}{P(A)}$

$$P(A \mid B, C) == \frac{P(AB \mid C)}{P(B \mid C)} = \frac{P(B \mid A, C)P(A \mid C)}{P(B \mid C)}$$

$$q(x_{t-1} \mid x_t, x_0) = q(x_t \mid x_{t-1}, x_0)\frac{q(x_{t-1} \mid x_0)}{q(x_t \mid x_0)} \propto \exp(-\frac{1}{2}(\frac{(x_t - \sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1} - \sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t - \sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}))$$

$$= \exp(-\frac{1}{2}((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})x_{t-1}^2 - 2(\frac{\sqrt{\alpha_t}}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)x_{t-1} + C(x_0, x_1)))$$

Hence $\tilde{\beta}_t = 1/(\dfrac{\alpha_t}{\beta_t} + \dfrac{1}{1-\bar{\alpha}_{t-1}}) = \dfrac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t,$

$$\tilde{\mu}_t = (\frac{\sqrt{\alpha}_t}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)/\tilde{\beta}_t = (\frac{\sqrt{\alpha}_t}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

Conditioning on x_0. Bayes rule to the rescue! $P(A|B) = \dfrac{P(B)P(B|A)}{P(A)}$

$$P(A|B,C) == \frac{P(AB|C)}{P(B|C)} = \frac{P(B|A,C)P(A|C)}{P(B|C)}$$

$$q(x_{t-1}|x_t,x_0) = q(x_t|x_{t-1},x_0)\frac{q(x_{t-1}|x_0)}{q(x_t|x_0)} \propto \exp(-\frac{1}{2}(\frac{(x_t-\sqrt{\alpha_t}x_{t-1})^2}{1-\alpha_t} + \frac{(x_{t-1}-\sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(x_t-\sqrt{\bar{\alpha}_t}x_0)^2}{1-\bar{\alpha}_t}))$$

$$= \exp(-\frac{1}{2}((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})x_{t-1}^2 - 2(\frac{\sqrt{\alpha_t}}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)x_{t-1} + C(x_0,x_1)))$$

Hence $\tilde{\beta}_t = 1/(\dfrac{\alpha_t}{\beta_t} + \dfrac{1}{1-\bar{\alpha}_{t-1}}) = \dfrac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t,$

$$\tilde{\mu}_t = (\frac{\sqrt{\alpha}_t}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)/\tilde{\beta}_t = (\frac{\sqrt{\alpha}_t}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}x_0)\frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

We can reparametrise it further! —> $= \dfrac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \dfrac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}x_0$

Remember! $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon,\qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon,\qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon,$ $\qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} \,|\, x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that?

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} \,|\, x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that? Minimise negative log-likelihood $-\log p_\theta(x_0)$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t} \varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} | x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that? Minimise negative log-likelihood $-\log p_\theta(x_0)$

$$-\log p_\theta(x_0) \leq -\log p_\theta(x_0) + D_{KL}(q(x_{1:T} | x_0) || p_\theta(x_{1:T} | x_0))$$

Remember! $x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1}\,|\,x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that? Minimise negative log-likelihood $-\log p_\theta(x_0)$

$$-\log p_\theta(x_0) \leq -\log p_\theta(x_0) + \underbrace{D_{KL}(q(x_{1:T}|x_0)\,||\,p_\theta(x_{1:T}|x_0))}_{\textbf{Kullback-Leibler divergence}\ \geq 0}$$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1 - \bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} \,|\, x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that? Minimise negative log-likelihood $-\log p_\theta(x_0)$

Kullback-Leibler divergence ≥ 0

$$-\log p_\theta(x_0) \leq -\log p_\theta(x_0) + D_{KL}(q(x_{1:T} \,|\, x_0) \,||\, p_\theta(x_{1:T} \,|\, x_0)) =$$

$$-\log p_\theta(x_0) + E_{x_{1:T} \sim q(x_{1:T}|x_0)}[\log \frac{q(x_{1:T} \,|\, x_0)}{p_\theta(x_{0:T})/p_\theta(x_0)}] = -\log p_\theta(x_0) + E_q[\log \frac{q(x_{1:T} \,|\, x_0)}{p_\theta(x_{0:T})}] + \log p_\theta(x_0)$$

Remember! $x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon, \qquad x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}\varepsilon_t)$

Substitute this into expression of $\tilde{\mu}_t$ and get $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

Recap! We need to learn the distributions $p_\theta(x_{t-1} | x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

So we will train μ_θ to predict $\tilde{\mu}_t$

How are we going to do that? Minimise negative log-likelihood $-\log p_\theta(x_0)$

$$-\log p_\theta(x_0) \leq -\log p_\theta(x_0) + \underbrace{D_{KL}(q(x_{1:T} | x_0) || p_\theta(x_{1:T} | x_0))}_{\textbf{Kullback-Leibler divergence } \geq 0} =$$

$$-\log p_\theta(x_0) + E_{x_{1:T} \sim q(x_{1:T}|x_0)}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})/p_\theta(x_0)}] = -\log \cancel{p_\theta(x_0)} + E_q[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] + \log \cancel{p_\theta(x_0)} = E_q[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}]$$

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] \geq -E_{q(x_0)} \log p_\theta(x_0)$

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] \geq - E_{q(x_0)} \log p_\theta(x_0)$

We used that $E_{q(x_0)} E_{q(x_{1:T}|x_0)} f(x_{0:T}) = E_{q(x_{0:T})} f(x_{0:T})$ for any f of our interest

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] \geq -E_{q(x_0)}p_\theta(x_0)$

We used that $E_{q(x_0)}E_{q(x_{1:T}|x_0)}f(x_{0:T}) = E_{q(x_{0:T})}f(x_{0:T})$ for any f of our interest

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}]$$

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] \geq - E_{q(x_0)} p_\theta(x_0)$

We used that $E_{q(x_0)} E_{q(x_{1:T}|x_0)} f(x_{0:T}) = E_{q(x_{0:T})} f(x_{0:T})$ for any f of our interest

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}]$$

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] \geq -E_{q(x_0)}p_\theta(x_0)$

We used that $E_{q(x_0)}E_{q(x_{1:T}|x_0)}f(x_{0:T}) = E_{q(x_{0:T})}f(x_{0:T})$ for any f of our interest

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

Let $L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] \geq -E_{q(x_0)} p_\theta(x_0)$

We used that $E_{q(x_0)} E_{q(x_{1:T}|x_0)} f(x_{0:T}) = E_{q(x_{0:T})} f(x_{0:T})$ for any f of our interest

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t | x_{t-1})}{p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t | x_{t-1})}{p_\theta(x_{t-1} | x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t | x_{t-1})}{p_\theta(x_{t-1} | x_t)} + \log \frac{q(x_1 | x_0)}{p_\theta(x_0 | x_1)}]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t | x_{t-1})}{p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t | x_{t-1})}{p_\theta(x_{t-1} | x_t)}]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^T q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^T p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^T \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^T \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^T \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)}\frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \ldots$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \dots$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \sum_{t=2}^{T} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T} | x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t | x_{t-1})}{p_\theta(x_T) \prod_{t=1}^{T} p_\theta(x_{t-1} | x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t | x_{t-1})}{p_\theta(x_{t-1} | x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t | x_{t-1})}{p_\theta(x_{t-1} | x_t)} + \log \frac{q(x_1 | x_0)}{p_\theta(x_0 | x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1} | x_t, x_0)}{p_\theta(x_{t-1} | x_t)} \frac{q(x_t | x_0)}{q(x_{t-1} | x_0)} + \dots$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1} | x_t, x_0)}{p_\theta(x_{t-1} | x_t)} + \sum_{t=2}^{T} \log \frac{q(x_t | x_0)}{q(x_{t-1} | x_0)} + \log \frac{q(x_1 | x_0)}{p_\theta(x_0 | x_1)}$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1} | x_t, x_0)}{p_\theta(x_{t-1} | x_t)} + \log \frac{q(x_T | x_0)}{q(x_1 | x_0)} + \log \frac{q(x_1 | x_0)}{p_\theta(x_0 | x_1)}]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \ldots$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \sum_{t=2}^{T} \log \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_T|x_0)}{\cancel{q(x_1|x_0)}} + \log \frac{\cancel{q(x_1|x_0)}}{p_\theta(x_0|x_1)}]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \ldots$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \sum_{t=2}^{T} \log \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_T|x_0)}{q\cancel{(x_1|x_0)}} + \log \frac{\cancel{q(x_1|x_0)}}{p_\theta(x_0|x_1)}]$$

$$= E_q[\log \frac{q(x_T|x_0)}{p_\theta(x_T)} + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} - \log p_\theta(x_0|x_1)]$$

$$L_{VLB} = E_{q(x_{0:T})}[\log \frac{q(x_{1:T}|x_0)}{p_\theta(x_{0:T})}] = E_q[\log \frac{\prod_{t=1}^{T} q(x_t|x_{t-1})}{p_\theta(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t)}] = E_q[-\log p_\theta(x_T) + \sum_{t=1}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)}]$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_t|x_{t-1})}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}] = E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \dots$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \sum_{t=2}^{T} \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)} + \log \frac{q(x_1|x_0)}{p_\theta(x_0|x_1)}$$

$$= E_q[-\log p_\theta(x_T) + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} + \log \frac{q(x_T|x_0)}{\cancel{q(x_1|x_0)}} + \log \frac{\cancel{q(x_1|x_0)}}{p_\theta(x_0|x_1)}]$$

$$= E_q[\log \frac{q(x_T|x_0)}{p_\theta(x_T)} + \sum_{t=2}^{T} \log \frac{q(x_{t-1}|x_t, x_0)}{p_\theta(x_{t-1}|x_t)} - \log p_\theta(x_0|x_1)]$$

$$= E_q[\underbrace{D_{KL}(q(x_T|x_0)||p_\theta(x_T))}_{L_T} + \sum_{t=2}^{T} \underbrace{D_{KL}(q(x_{t-1}|x_t, x_0)||p_\theta(x_{t-1}|x_t))]}_{L_{t-1}} - \underbrace{\log p_\theta(x_0|x_1)}_{L_0}$$

Simplifying the loss further

$$L_{VLB} = L_T + L_{T-1} + \ldots + L_0$$

Simplifying the loss further

$$L_{VLB} = L_T + L_{T-1} + \ldots + L_0$$

$L_T = D_{KL}(q(x_T|x_0)||p_\theta(x_T))$ Constant, since no trainable parameters and x_T is Gaussian noise

Simplifying the loss further

$$L_{VLB} = L_T + L_{T-1} + \ldots + L_0$$

$L_T = D_{KL}(q(x_T | x_0) || p_\theta(x_T))$ Constant, since no trainable parameters and x_T is Gaussian noise

$L_t = D_{KL}(q(x_{t-1} | x_t, x_0) || p_\theta(x_{t-1} | x_t))$ The main thing!

Simplifying the loss further

$$L_{VLB} = L_T + L_{T-1} + \ldots + L_0$$

$L_T = D_{KL}(q(x_T | x_0) || p_\theta(x_T))$ Constant, since no trainable parameters and x_T is Gaussian noise

$L_t = D_{KL}(q(x_{t-1} | x_t, x_0) || p_\theta(x_{t-1} | x_t))$ The main thing!

$L_0 = -\log p_\theta(x_0 | x_1)$ Can be omitted or modelled via a separate decoder, derived from

$$\mathcal{N}(x_0; \mu_\theta(x_1, 1), \Sigma_\theta(x_1, 1))$$

Simplifying the loss further

$$L_{VLB} = L_T + L_{T-1} + \ldots + L_0$$

$L_T = D_{KL}(q(x_T | x_0) || p_\theta(x_T))$ Constant, since no trainable parameters and x_T is Gaussian noise

$L_t = D_{KL}(q(x_{t-1} | x_t, x_0) || p_\theta(x_{t-1} | x_t))$ The main thing!

$L_0 = -\log p_\theta(x_0 | x_1)$ Can be omitted or modelled via a separate decoder, derived from

$$\mathcal{N}(x_0; \mu_\theta(x_1, 1), \Sigma_\theta(x_1, 1))$$

Remember again! We need to learn the distributions $p_\theta(x_{t-1} | x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$

That is for L_t !

Parametrisation of L_t for Training Loss

$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

We would like to train μ_θ to predict $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

Parametrisation of L_t for Training Loss

$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

We would like to train μ_θ to predict $\tilde{\mu}_t = \dfrac{1}{\sqrt{\alpha_t}}(x_t - \dfrac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

But x_t is available during training time!

Parametrisation of L_t for Training Loss

$$p_\theta(x_{t-1} \,|\, x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

We would like to train μ_θ to predict $\tilde{\mu}_t = \dfrac{1}{\sqrt{\alpha_t}}(x_t - \dfrac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

But x_t is available during training time!

Hence, let us **predict the noise term instead**!

Parametrisation of L_t for Training Loss

$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

We would like to train μ_θ to predict $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_t)$

But x_t is available during training time!

Hence, let us **predict the noise term instead**!

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}}\varepsilon_\theta(x_t, t))$$

Parametrisation of L_t for Training Loss

$$p_\theta(x_{t-1} \mid x_t) \sim \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t))$$

We would like to train μ_θ to predict $\tilde{\mu}_t = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t)$

But x_t is available during training time!

Hence, let us **predict the noise term instead**!

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t, t)) \qquad x_{t-1} \sim \mathcal{N}(x_{t-1}; \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t, t)), \Sigma_\theta(x_t, t))$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}\left[\frac{1}{2||\Sigma_\theta(x_t, t)||_2^2}||\tilde{\mu}_t(x_t, x_0) - \mu_\theta(x_t, t)||^2\right]$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}\left[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2\right]$$

$$= E_{x_0,\varepsilon}\left[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2\right]$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(\cancel{x_t} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(\cancel{x_t} - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2]$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2] = E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2]$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2] = E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2]$$

In practice can simplify even further!

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2] = E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2]$$

In practice can simplify even further!

$$L_t^{simple} = E_{t,x_0,\varepsilon}[||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2$$

Parametrisation of L_t for Training Loss

$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2] = E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2]$$

In practice can simplify even further!

$$L_t^{simple} = E_{t,x_0,\varepsilon}[||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2$$

Bottom line: what we are doing is predicting the noise!

Parametrisation of L_t for Training Loss

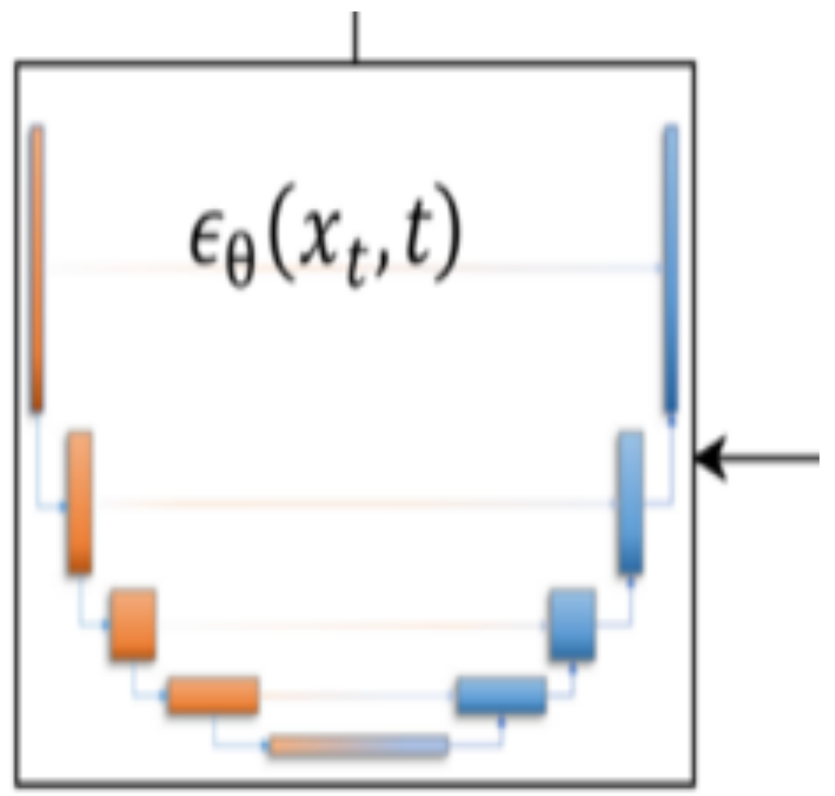
$$L_t = E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta(x_t,t)||_2^2}||\tilde{\mu}_t(x_t,x_0) - \mu_\theta(x_t,t)||^2]$$

$$= E_{x_0,\varepsilon}[\frac{1}{2||\Sigma_\theta||_2^2}||\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_t) - \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(x_t,t))||^2]$$

$$= E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(x_t,t)||^2] = E_{x_0,\varepsilon}[\frac{(1-\alpha_t)^2}{2\alpha_t(1-\bar{\alpha}_t)||\Sigma_\theta||_2^2}||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2]$$

In practice can simplify even further!

$$L_t^{simple} = E_{t,x_0,\varepsilon}[||\varepsilon_t - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\varepsilon_t, t)||^2$$



Bottom line: what we are doing is predicting the noise!

Unet architecture is used for that

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Can be fixed (not learnable)

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Can be fixed (not learnable)

Parametrization of reverse process variance Σ_θ

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Can be fixed (not learnable)

Parametrization of reverse process variance Σ_θ

Can also be fixed $\Sigma_\theta(x_t, t) = \sigma_t^2 \mathbf{I}$, where σ_t is set to be a function of β_t

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Can be fixed (not learnable)

Parametrization of reverse process variance Σ_θ

Can also be fixed $\Sigma_\theta(x_t, t) = \sigma_t^2 \mathbf{I}$, where σ_t is set to be a function of β_t

Alternative $\Sigma_\theta(x_t, t) = \exp(v \log \beta_t + (1 - v) \log \tilde{\beta}_t)$, $\quad \tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}$, v is learnable

Parametrization of β_t

Typically a sequence of linearly increasing constants: e.g. from $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$

Can be fixed (not learnable)

Parametrization of reverse process variance Σ_θ

Can also be fixed $\Sigma_\theta(x_t, t) = \sigma_t^2 \mathbf{I}$, where σ_t is set to be a function of β_t

Alternative $\Sigma_\theta(x_t, t) = \exp(v \log \beta_t + (1 - v) \log \tilde{\beta}_t)$, $\quad \tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}$, v is learnable

Hence, the loss $L = L_{simple} + \lambda L_{VLB}$, λ is small ~ 0.001 and L_{VLB} only guides the training of Σ_θ

(In L_{VLB} stop gradient with respect to μ_θ)

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Langevin dynamics can sample data points from a probability density distribution using only the score $\nabla_x \log q(x)$ in an iterative process.

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Langevin dynamics can sample data points from a probability density distribution using only the score $\nabla_x \log q(x)$ in an iterative process.

Score network $s_\theta(x_t, t) \approx \nabla_x \log q(x)$

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Langevin dynamics can sample data points from a probability density distribution using only the score $\nabla_x \log q(x)$ in an iterative process.

Score network $s_\theta(x_t, t) \approx \nabla_x \log q(x)$

If $x \sim \mathcal{N}(\mu, \sigma^2 \mathbf{I}), \quad \nabla_x \log p(x) = \nabla_x (-\frac{1}{2\sigma^2}(x-\mu)^2) = -\frac{x-\mu}{\sigma^2} = -\frac{\varepsilon}{\sigma}, \quad \varepsilon \sim \mathcal{N}(0,\mathbf{I})$

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Langevin dynamics can sample data points from a probability density distribution using only the score $\nabla_x \log q(x)$ in an iterative process.

Score network $s_\theta(x_t, t) \approx \nabla_x \log q(x)$

If $x \sim \mathcal{N}(\mu, \sigma^2 \mathbf{I}), \quad \nabla_x \log p(x) = \nabla_x(-\frac{1}{2\sigma^2}(x-\mu)^2) = -\frac{x-\mu}{\sigma^2} = -\frac{\varepsilon}{\sigma}, \quad \varepsilon \sim \mathcal{N}(0,\mathbf{I})$

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

Bonus! Score networks and guided diffusion

The score of each sample's x probability density function is defined as $\nabla_x \log q(x)$

Langevin dynamics can sample data points from a probability density distribution using only the score $\nabla_x \log q(x)$ in an iterative process.

Score network $s_\theta(x_t, t) \approx \nabla_x \log q(x)$

If $x \sim \mathcal{N}(\mu, \sigma^2 \mathbf{I})$, $\quad \nabla_x \log p(x) = \nabla_x \left(-\frac{1}{2\sigma^2}(x-\mu)^2\right) = -\frac{x-\mu}{\sigma^2} = -\frac{\varepsilon}{\sigma}, \quad \varepsilon \sim \mathcal{N}(0,\mathbf{I})$

$$q(x_t \mid x_0) \sim \mathcal{N}(\sqrt{\bar{\alpha}_t} x_0, (1-\bar{\alpha}_t)\mathbf{I})$$

And therefore $s_\theta(x_t, t) \approx \nabla_{x_t} \log q(x_t) = E_{q(x_0)} \nabla_{x_t} q(x_t \mid x_0) = E_{q(x_0)}\left[-\frac{\varepsilon_\theta(x_t, t)}{\sqrt{1-\bar{\alpha}_t}}\right] = -\frac{\varepsilon_\theta(x_t, t)}{\sqrt{1-\bar{\alpha}_t}}$

Guided diffusion

- We have additional input y (a class label in classifier guided diffusion)
- We want to model a conditional distribution $p(x \mid y)$ instead

Conditioned on dogs

Classifier guided diffusion

We separately train a classifier $f_\phi(y \mid x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y \mid x_t)$

to guide the diffusion. Let us have a joint distribution $q(x_t, y)$, y is e.g. the image label

Classifier guided diffusion

We separately train a classifier $f_\phi(y|x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y|x_t)$

to guide the diffusion. Let us have a joint distribution $q(x_t, y)$, y is e.g. the image label

$$\nabla_{x_t} \log q(x_t|y) = \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y|x_t) - \nabla_{x_t} \log q(y)$$

Classifier guided diffusion

We separately train a classifier $f_\phi(y \,|\, x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y \,|\, x_t)$

to guide the diffusion. Let us have a joint distribution $q(x_t, y),\ y$ is e.g. the image label

$$\nabla_{x_t} \log q(x_t \,|\, y) = \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y \,|\, x_t) - \nabla_{x_t} \log q(y) \approx -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \varepsilon_\theta(x_t, t) + \nabla_{x_t} \log f_\phi(y \,|\, x_t)$$

Classifier guided diffusion

We separately train a classifier $f_\phi(y \mid x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y \mid x_t)$ to guide the diffusion. Let us have a joint distribution $q(x_t, y)$, y is e.g. the image label

$$\nabla_{x_t} \log q(x_t \mid y) = \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y \mid x_t) - \nabla_{x_t} \log q(y) \approx -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \varepsilon_\theta(x_t, t) + \nabla_{x_t} \log f_\phi(y \mid x_t)$$

$$= -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \left(\varepsilon_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, \nabla_{x_t} \log f_\phi(y \mid x_t)\right)$$

Classifier guided diffusion

We separately train a classifier $f_\phi(y \mid x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y \mid x_t)$

to guide the diffusion. Let us have a joint distribution $q(x_t, y)$, y is e.g. the image label

$$\nabla_{x_t} \log q(x_t \mid y) = \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y \mid x_t) - \nabla_{x_t} \log q(y) \approx -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \varepsilon_\theta(x_t, t) + \nabla_{x_t} \log f_\phi(y \mid x_t)$$

$$= -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \left(\varepsilon_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, \nabla_{x_t} \log f_\phi(y \mid x_t)\right)$$

Classifier guided predictor $\bar{\varepsilon}_\theta(x_t, t) = \varepsilon_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, w\, \nabla_{x_t} \log f_\phi(y \mid x_t)$

Classifier guided diffusion

We separately train a classifier $f_\phi(y \mid x_t, t)$ on a noisy image x_t, and use gradients $\nabla_x \log f_\phi(y \mid x_t)$

to guide the diffusion. Let us have a joint distribution $q(x_t, y)$, y is e.g. the image label

$$\nabla_{x_t} \log q(x_t \mid y) = \nabla_{x_t} \log q(x_t) + \nabla_{x_t} \log q(y \mid x_t) - \nabla_{x_t} \log q(y) \approx -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \varepsilon_\theta(x_t, t) + \nabla_{x_t} \log f_\phi(y \mid x_t)$$

$$= -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \left(\varepsilon_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, \nabla_{x_t} \log f_\phi(y \mid x_t)\right)$$

Classifier guided predictor $\bar{\varepsilon}_\theta(x_t, t) = \varepsilon_\theta(x_t, t) - \sqrt{1-\bar{\alpha}_t}\, w \nabla_{x_t} \log f_\phi(y \mid x_t)$

Where w is the strength of the guidance

Classifier guided diffusion

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $f_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + s\Sigma\,\nabla_{x_t} \log f_\phi(y|x_t), \Sigma)$
end for
return x_0

Downsides:

Classifier guided diffusion

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $f_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + s\Sigma\, \nabla_{x_t} \log f_\phi(y|x_t), \Sigma)$
end for
return x_0

Downsides:

- The classifier has to cope with noise, might need to be trained separately

Classifier guided diffusion

Algorithm 1 Classifier guided diffusion sampling, given a diffusion model $(\mu_\theta(x_t), \Sigma_\theta(x_t))$, classifier $f_\phi(y|x_t)$, and gradient scale s.

Input: class label y, gradient scale s
$x_T \leftarrow$ sample from $\mathcal{N}(0, \mathbf{I})$
for all t from T to 1 **do**
  $\mu, \Sigma \leftarrow \mu_\theta(x_t), \Sigma_\theta(x_t)$
  $x_{t-1} \leftarrow$ sample from $\mathcal{N}(\mu + s\Sigma\, \nabla_{x_t} \log f_\phi(y|x_t), \Sigma)$
end for
return x_0

Downsides:

- The classifier has to cope with noise, might need to be trained separately
- If noise-robust might be inefficient: most of the information on x is irrelevant to y

Classifier free guidance

- Train a diffusion model $p(x \mid y)$ with conditioning dropout: conditional information is removed some percentage of the time

Classifier free guidance

- Train a diffusion model $p(x \mid y)$ with conditioning dropout: conditional information is removed some percentage of the time

- Resulting model can be conditional $p(x \mid y)$ and unconditional $p(x)$

Classifier free guidance

- Train a diffusion model $p(x|y)$ with conditioning dropout: conditional information is removed some percentage of the time
- Resulting model can be conditional $p(x|y)$ and unconditional $p(x)$
- No need for a separate classifier!

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

- Conditional information y gets discarded periodically: $\varepsilon_\theta(x_t, t) = \varepsilon_\theta(x_t, t, y = \varnothing)$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

- Conditional information y gets discarded periodically: $\varepsilon_\theta(x_t, t) = \varepsilon_\theta(x_t, t, y = \varnothing)$

$$\nabla_{x_t} \log p(y \mid x_t) = \nabla_{x_t} \log p(x_t \mid y) - \nabla_{x_t} \log p(x_t)$$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

- Conditional information y gets discarded periodically: $\varepsilon_\theta(x_t, t) = \varepsilon_\theta(x_t, t, y = \emptyset)$

$$\nabla_{x_t} \log p(y \mid x_t) = \nabla_{x_t} \log p(x_t \mid y) - \nabla_{x_t} \log p(x_t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}} (\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x \mid y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

- Conditional information y gets discarded periodically: $\varepsilon_\theta(x_t, t) = \varepsilon_\theta(x_t, t, y = \varnothing)$

$$\nabla_{x_t} \log p(y \mid x_t) = \nabla_{x_t} \log p(x_t \mid y) - \nabla_{x_t} \log p(x_t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}}(\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Now $\nabla_{x_t} \log p(x_t, y) = \nabla_{x_t} \log p(x_t) + \nabla_{x_t} \log p(y \mid x_t) = \nabla_{x_t} \log p(x_t \mid y) + \nabla_{x_t} \log y + \nabla_{x_t} p(y \mid x_t)$

Classifier free guidance

Guide the diffusion without an independent classifier f_ϕ

- Unconditional denoising diffusion model $p_\theta(x)$, parametrised through a score estimator $\varepsilon_\theta(x_t, t)$

- Conditional model $p_\theta(x_t | y)$ parametrised through $\varepsilon_\theta(x_t, t, y)$

- Conditional model is trained on paired data (x, y)

- Conditional information y gets discarded periodically: $\varepsilon_\theta(x_t, t) = \varepsilon_\theta(x_t, t, y = \varnothing)$

$$\nabla_{x_t} \log p(y | x_t) = \nabla_{x_t} \log p(x_t | y) - \nabla_{x_t} \log p(x_t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}}(\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Now $\nabla_{x_t} \log p(x_t, y) = \nabla_{x_t} \log p(x_t) + \nabla_{x_t} \log p(y | x_t) = \nabla_{x_t} \log p(x_t | y) + \cancel{\nabla_{x_t} \log y} + \nabla_{x_t} p(y | x_t)$

Classifier free guidance

$$\nabla_{x_t} \log p(x_t, y) = \nabla_{x_t} \log p(x_t \mid y) + \nabla_{x_t} \log p(y \mid x_t)$$

$$\nabla_{x_t} \log p(y \mid x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}}(\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Hence, analogously to the case of classifier guided diffusion

Classifier free guidance

$$\nabla_{x_t} \log p(x_t, y) = \nabla_{x_t} \log p(x_t \,|\, y) + \nabla_{x_t} \log p(y \,|\, x_t)$$

$$\nabla_{x_t} \log p(y \,|\, x_t) = -\frac{1}{\sqrt{1-\bar{\alpha}_t}}(\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Hence, analogously to the case of classifier guided diffusion

$\bar{\varepsilon}_\theta(x_t, t, y) = \varepsilon_\theta(x_t, t, y) - \sqrt{1-\bar{\alpha}_t}\, w \nabla_{x_t} \log p(y \,|\, x_t)$, where $\varepsilon_\theta(x_t, t, y)$ parametrises $p(x_t \,|\, y)$

Classifier free guidance

$$\nabla_{x_t} \log p(x_t, y) = \nabla_{x_t} \log p(x_t | y) + \nabla_{x_t} \log p(y | x_t)$$

$$\nabla_{x_t} \log p(y | x_t) = -\frac{1}{\sqrt{1 - \bar{\alpha}_t}} (\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$

Hence, analogously to the case of classifier guided diffusion

$\bar{\varepsilon}_\theta(x_t, t, y) = \varepsilon_\theta(x_t, t, y) - \sqrt{1 - \bar{\alpha}_t} w \nabla_{x_t} \log p(y | x_t)$, where $\varepsilon_\theta(x_t, t, y)$ parametrises $p(x_t | y)$

$$= \varepsilon_\theta(x_t, t, y) + w(\varepsilon_\theta(x_t, t, y) - \varepsilon_\theta(x_t, t))$$