Introduction to DDPM

Liu Chengeng
NTU, SCSE
chengeng001@e.ntu.edu.sg

Abstract

This document contains priori knowledge about probability, diffussion forward process and reverse process, as well as training loss of DDPM.

1. Priori knowledge

$y \propto x$ means y has positive linear correlation with x, x increases leads to y increasing.

$P(A|B)$ means given event B happens, the probability that event A happens.

$P(A|B,C)$ means given event B and event C happen, the probability that event A happens.

$P(A|B) = \frac{P(B|A)*P(A)}{P(B)}$ – Baysian Theorem.

Gaussian Distribution probability density function:

$\mathcal{N}(\mu, \sigma^2)$ and $p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}$

Meaning of :

$q(x_t|x_t - 1) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I})$
Conditional probability distribution, $q(x_t|x_{t-1})$ means given x_{t-1}, the distribution of x_t (the probability that x_t chooses any values). The latter Gaussian Distribution denotes its input is x_t, and the mean and std of the distribution are $\sqrt{1-\beta_t}x_{t-1}$ and $\beta_t\mathbf{I}$) respectively, which closely related with x_{t-1}

Add two Gaussian distribution: $\mathcal{N}(0, \sigma_1^2) + \mathcal{N}(0, \sigma_2^2)$, a new Gaussian distribution: $\mathcal{N}(0, \sigma_1^2 + \sigma_2^2)$

Jensen' inequality: $\log \mathbb{E}[f(x)] \leq \mathbb{E}[\log f(x)]$
KL divergence: $D_{\text{KL}}(q(x)||p(x)) = \mathbb{E}_{q(x)}[\log q(x)/p(x)]$

1.1. Reparametrization trick

Reparametrization trick:
How to train a model when you want to sample from a distribution? Sampling is a stochastic process and therefor we cannot backpropagate the gradient. To make it trainable, we need a trick called reparametrization trick. The randomness is introduced by:
$z = \tau_\phi(x, \epsilon)$, where ϵ is an auxiliary independent random variable and the transformation function τ_ϕ parameterized by ϕ converts ϵ to z.

For example,

$$\begin{aligned}&\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)}\boldsymbol{I})\\ &\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\epsilon}, \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I}) \quad \text{; Reparameterization trick.}\end{aligned} \tag{1}$$

where $\odot$ denots element-wise product.

2. Diffusion forward process

So the forward pass is described as: the forward diffusion process, parameterized as a Markov Chain, corrupts a datapoint (e.g., an image here) by gradually adding tiny amount of noises and ultimately resulting in a pure Gaussian. given every timestep of the forward process, we can thereby learn a model p_θ to approximate the reverse, namely the generative process, to gradually remove this series of noises by imitating the inverse of each pair of consecutive timesteps along the forward chain, recovering the data/image with salient features.

The forward process is defined by a serics of Gaussians: $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I})$ with first order Markov property.

$$\begin{aligned}q(\mathbf{x}_t|\mathbf{x}_{t-1}) &= \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})\\ q(\mathbf{x}_{1:T}|\mathbf{x}_0) &= \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})\end{aligned} \tag{2}$$

Given a data point sampled from a real data distribution $\mathbf{x}_0 \sim q(\mathbf{x})$, add noise in T steps, producing a sequence of noisy samples: $\mathbf{x}_1, \ldots, \mathbf{x}_T$. The step sizes are controlled by a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^T$.

Given a data point sampled from a real data distribution $\mathbf{x}_0 \sim q(\mathbf{x})$, add noise in T steps, producing a sequence of noisy samples: $\mathbf{x}_1, \ldots, \mathbf{x}_T$. The step sizes are controlled by a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^T$.

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

$$\begin{aligned}
\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1} \\
&= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \sqrt{1-\alpha_t\alpha_{t-1}}\bar{\boldsymbol{\epsilon}}_{t-2} \\
&= \ldots \\
&= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \\
&\text{where } \boldsymbol{\epsilon}_{t-1}, \boldsymbol{\epsilon}_{t-2}, \cdots \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\
&\text{and } \bar{\boldsymbol{\epsilon}}_{t-2} \text{ merges two Gaussians (*).} \\
q(\mathbf{x}_t|\mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})
\end{aligned} \tag{3}$$

Now expand the equation:

$$\begin{aligned}
&\sqrt{\alpha_t}x_{t-1} + \sqrt{1-\alpha_t}z_{t-1} \\
=&\sqrt{\alpha_t}(\sqrt{\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}}z_{t-2}) + \sqrt{1-\alpha_t}z_{t-1} \\
=&\sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{\alpha_t(1-\alpha_{t-1})}z_{t-2} + \sqrt{1-\alpha_t}z_{t-1} \\
=&\sqrt{\alpha_t\alpha_{t-1}}x_{t-2} + \sqrt{1-\alpha_{t-1}\alpha_t}\bar{z}_{t-2}
\end{aligned} \tag{4}$$

And sum up the last two terms:

$$\begin{aligned}
\sigma_{\text{new}} = \sqrt{\sigma_1^2 + \sigma_2^2} &= \sqrt{\left(\sqrt{\alpha_t(1-\alpha_{t-1})}\right)^2 + \left(\sqrt{1-\alpha_t}\right)^2} \\
&= \sqrt{\alpha_t(1-\alpha_{t-1}) + (1-\alpha_t)} \\
&= \sqrt{1-\alpha_t\alpha_{t-1}}
\end{aligned} \tag{5}$$

Langevin dynamics is a concept from physics, developed for statistically modeling molecular systems. Combined with stochastic gradient descent, stochastic gradient Langevin dynamics (Welling Teh 2011) can produce samples from a probability density $p(x)$ using only the gradients $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ in a Markov chain of updates: test test

Langevin dynamics is a concept from physics, developed for statistically modeling molecular systems. Combined with stochastic gradient descent, stochastic gradient Langevin dynamics (Welling Teh 2011) can produce samples from a probability density $p(x)$ using only the gradients $\nabla_{\mathbf{x}} \log p(\mathbf{x})$ in a Markov chain of updates:

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \frac{\delta}{2}\nabla_{\mathbf{x}} \log q(\mathbf{x}_{t-1}) + \sqrt{\delta}\boldsymbol{\epsilon}_t, \quad \text{where } \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \tag{6}$$

3. Diffusion reverse process

3.1. True conditional distribution

Previously, we know how to get $q(x_t|x_{t-1})$ and $q(x_t|x_0)$ by gradually adding noise. Our goal now becomes: if we can reverse the above forward process and sample from $q(x_{t-1}|x_t)$, we will recreate the true sample from the Gaussian noise. However, it is not easy to estimate $q(x_{t-1}|x_t)$ because the distribution is unknown. You can brute-forcely iterate all possible choices(iterate the entire dataset), from Baysian theorem:

$$q(x_{t-1}|x_t) = \frac{q(x_t|x_{t-1})(x_{t-1})}{q(x_t)} \tag{7}$$

$q(x_t)$ and $q(x_{t-1})$ are unknown. t is unknown, since t could be very large(pure noise) or t could be small(original image with little noise). From equation (3) we known $q(x_t|x_0)$ and $q(x_{t-1}|x_0)$. Now the distribution dependednt on x_t and x_0: $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I})$. Again, using Bayes' rule, the previous equation becomes:

$$\begin{aligned}
&q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \\
&= q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \\
&\propto \exp\Big(-\frac{1}{2}\big(\frac{(\mathbf{x}_t - \sqrt{\alpha_t}\mathbf{x}_{t-1})^2}{\beta_t} + \frac{(\mathbf{x}_{t-1} - \sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0)^2}{1-\bar{\alpha}_{t-1}} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{1-\bar{\alpha}_t}\big)\Big) \\
&= \exp\Big(-\frac{1}{2}\big(\frac{\mathbf{x}_t^2 - 2\sqrt{\alpha_t}\mathbf{x}_t\mathbf{x}_{t-1} + \alpha_t\mathbf{x}_{t-1}^2}{\beta_t} + \frac{\mathbf{x}_{t-1}^2 - 2\sqrt{\bar{\alpha}_{t-1}}\mathbf{x}_0\mathbf{x}_{t-1} + \bar{\alpha}_{t-1}\mathbf{x}_0^2}{2} - \frac{(\mathbf{x}_t - \sqrt{\bar{\alpha}_t}\mathbf{x}_0)^2}{1-\bar{\alpha}_t}\big)\Big) \\
&= \exp\Big(-\frac{1}{2}\big((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})\mathbf{x}_{t-1}^2 - (\frac{2\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{2\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}\mathbf{x}_0)\mathbf{x}_{t-1} + C(\mathbf{x}_t, \mathbf{x}_0)\big)\Big)
\end{aligned} \tag{8}$$

where the first term:

$$q(x_t|x_{t-1}, x_0) = q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_{t-1}, \beta_t\mathbf{I}) \propto \exp\left(-\frac{1}{2}\frac{(x_t - \sqrt{\alpha_t}x_{t-1})^2}{\beta_t}\right) \tag{9}$$

and the second term:

$$q(x_{t-1}|x_0) = \mathcal{N}(x_{t-1}; \sqrt{\bar{\alpha}_{t-1}}x_0, (1-\bar{\alpha}_{t-1})\mathbf{I}) \propto \exp(-\frac{1}{2}\frac{(x_{t-1} - \sqrt{\bar{\alpha}_{t-1}}x_0)^2}{1-\bar{\alpha}_{t-1}}) \tag{10}$$

by using the fact that:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{11}$$

The last term is similar to the second term, but substitute $t-1$ with t in the second term. Finally, $q(x_{t-1}|x_t, x_0)$ equals:

$$\begin{aligned} &q(x_{t-1}|x_t, x_0) \\ &= \exp\Big(-\frac{1}{2}\big((\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}})\mathbf{x}_{t-1}^2 - \\ &(\frac{2\sqrt{\alpha_t}}{\beta_t}\mathbf{x}_t + \frac{2\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_{t-1}}\mathbf{x}_0)\mathbf{x}_{t-1} + C(\mathbf{x}_t, \mathbf{x}_0)\big)\Big) \end{aligned} \tag{12}$$

Rearrange the above equation:

$$ax^2 + bx + C = a(x + \frac{b}{2a})^2 \tag{13}$$

such that:

$$\begin{aligned} a &= \frac{\alpha_t}{\beta_t} + \frac{1}{1-\sqrt{\bar{\alpha}_{t-1}}} \\ b &= -(\frac{2\sqrt{\alpha_t}}{\beta_t}x_t + \frac{2\sqrt{\bar{\alpha}_{t-1}}}{1-\sqrt{\bar{\alpha}_{t-1}}}x_0) \end{aligned} \tag{14}$$

The goal of the above process is to calculate the distribution of $q(x_{t-1}|x_t, x_t)$. To do this, we need to know its mean and variance. From the equation(12,13&14), we can calculate the mean and the variance in equation (15,16&17), since $\tilde{\beta}_t = \frac{1}{\alpha}$ and the mean $\tilde{\mu}_t(x_t, x_0) = -\frac{b}{2a}$

$$q(x_{t-1}|x_t, x_0) = \mathcal{N}(x_{t-1}; \tilde{\mu}(x_t, x_0), \tilde{\beta}_t\mathbf{I}) \approx \exp\left(-\frac{(x - \tilde{\mu}(x_t, x_0))^2}{2\tilde{\beta}_t}\right) \tag{15}$$

Now the variance of the distribution is:

$$\begin{aligned} \tilde{\beta}_t &= \frac{1}{\frac{\alpha_t}{\beta_t} + \frac{1}{1-\bar{\alpha}_{t-1}}} \\ &= \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t \end{aligned} \tag{16}$$

The mean is:

$$\begin{aligned} \tilde{\mu}_t(x_t, x_0) &= -\frac{b}{2a} \\ &= \frac{(\frac{\sqrt{\alpha_t}}{\beta_t}x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}}{1-\sqrt{\bar{\alpha}_{t-1}}}x_0)}{\frac{\alpha_t}{\beta_t} + \frac{1}{1-\sqrt{\bar{\alpha}_{t-1}}}} \\ &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}x_t + \frac{\sqrt{\bar{\alpha}_t - 1}\beta_t}{1-\bar{\alpha}_t}x_0 \end{aligned} \tag{17}$$

The question becomes: how to remove x_0? From equation (3), we have: $x_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. So $x_0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t}z_t)$, and substitute into equation(17), the final mean $\tilde{\mu}_t(x_t, x_0)$ is:

$$
\begin{aligned}
\tilde{\mu}_t(x_t, x_0) &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} x_0 \\
&= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} \cdot \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t} z_t) \\
&= \frac{\sqrt{\alpha_t} \cdot \sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{\sqrt{\alpha_t} \cdot (1-\bar{\alpha}_t)} x_t + \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t} \cdot \frac{1}{\sqrt{\bar{\alpha}_t}}(x_t - \sqrt{1-\bar{\alpha}_t} z_t) \\
&= \frac{\alpha_t - \bar{\alpha}_t}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} x_t + \frac{\beta_t}{(1-\bar{\alpha}_t)\sqrt{\alpha_t}}(x_t - \sqrt{1-\bar{\alpha}_t} z_t) \\
&= \frac{1-\bar{\alpha}_t}{\sqrt{\alpha_t}(1-\bar{\alpha}_t)} x_t - \frac{\beta_t}{(1-\bar{\alpha}_t)\sqrt{\alpha_t}}(\sqrt{1-\bar{\alpha}_t} z_t) \\
&= \frac{1}{\sqrt{\alpha_t}} x_t - \frac{\beta_t}{\sqrt{(1-\bar{\alpha}_t)}\sqrt{\alpha_t}} z_t \\
&= \frac{1}{\sqrt{\alpha_t}} \left(x_t - \frac{\beta_t}{\sqrt{(1-\bar{\alpha}_t)}} z_t\right)
\end{aligned} \tag{18}
$$

The only unknown variable is z_t, which is the noise. If we know z_t, then we can calculate $q(x_{t-1}|x_t)$.

3.2. Learned model

Now, here comes the power of Deep Neural Net! We need to learn a model p_θ such that it can approximate the distribution $q(x_{t-1}|x_t)$ to complete the reverse diffusion process. Moreover, we need to :

$$
\begin{aligned}
p_\theta(\mathbf{x}_{0:T}) &= p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \\
p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))
\end{aligned} \tag{19}
$$

3.2.1 model and true distribution

In short, we aim at minimizing the difference between q and p_θ:

$$
\begin{aligned}
L_{\text{CE}} &= -\mathbb{E}_{q(\mathbf{x}_0)} \log p_\theta(\mathbf{x}_0) \\
&= -\mathbb{E}_{q(\mathbf{x}_0)} \log \left(\int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T} \right) \\
&= -\mathbb{E}_{q(\mathbf{x}_0)} \log \left(\int q(\mathbf{x}_{1:T}|\mathbf{x}_0) \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} d\mathbf{x}_{1:T} \right) \\
&= -\mathbb{E}_{q(\mathbf{x}_0)} \log \left(\mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right) \\
&\leq -\mathbb{E}_{q(\mathbf{x}_{0:T})} \log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \\
&= \mathbb{E}_{q(\mathbf{x}_{0:T})} \left[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})} \right] = L_{\text{VLB}}
\end{aligned} \tag{20}
$$

, by Jensen' inequality: $\log \mathbb{E}[f(x)] \leq \mathbb{E}[\log f(x)]$ and combine expection over $q(x_0)$ and expection over $q(x_{1:T}|x_0)$

$$
\begin{aligned}
L_{\text{VLB}} &= \mathbb{E}_{q(\mathbf{x}_{0:T})} \left[\log \frac{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}{p_\theta(\mathbf{x}_{0:T})} \right] \\
&= \mathbb{E}_q \left[\log \frac{\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} \right] \\
&= \mathbb{E}_q \left[-\log p_\theta(\mathbf{x}_T) + \sum_{t=1}^{T} \log \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} \right] \\
&= \mathbb{E}_q \left[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} \right] \\
&= \mathbb{E}_q \left[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^{T} \log \left(\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} \cdot \frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} \right) + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} \right] \\
&= \mathbb{E}_q \left[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} \right] \\
&= \mathbb{E}_q \left[-\log p_\theta(\mathbf{x}_T) + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} + \log \frac{q(\mathbf{x}_T|\mathbf{x}_0)}{q(\mathbf{x}_1|\mathbf{x}_0)} + \log \frac{q(\mathbf{x}_1|\mathbf{x}_0)}{p_\theta(\mathbf{x}_0|\mathbf{x}_1)} \right] \\
&= \mathbb{E}_q \left[\log \frac{q(\mathbf{x}_T|\mathbf{x}_0)}{p_\theta(\mathbf{x}_T)} + \sum_{t=2}^{T} \log \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)} - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1) \right] \\
&= \mathbb{E}_q [\underbrace{D_{\text{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \parallel p_\theta(\mathbf{x}_T))}_{L_T} + \sum_{t=2}^{T} \underbrace{D_{\text{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \parallel p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}]
\end{aligned} \tag{21}
$$

Finally, to convert each term to be analytically computable, the above objective is further rewritten to be a combination of several KL-divegence and entropy terms as shwon in equation 21: (from Sohl-Dickstein et al., 2015). From Line4 to Line5:

$\mathrm{q}(\mathrm{x}_t|x_{t-1}) = q(x_t|x_{t-1}, x_0) = \frac{q(x_t, x_{t-1}|x_0)}{q(x_{t-1}|x_0)} = q(x_{t-1}|x_t, x_0) \cdot \frac{q(x_t|x_0)}{q(x_{t-1}|x_0)}$

From Line8 to Line9, recall $D_{\text{KL}}(q(x)||p(x)) = \mathbb{E}_{q(x)}[\log q(x)/p(x)]$

3.3. training

Remember the mean we calculate:

$$\tilde{\mu}_t(x_t, x_0) = \frac{1}{\sqrt{\alpha_t}}\big(x_t - \frac{\beta_t}{\sqrt{(1-\bar{\alpha}_t)}} z_t\big) \tag{22}$$

We want to design a network that can train on $\epsilon(x_t, t)$ to predict noise:

1. the input of the network contains the current image x_t and noising step t. We need t because we only know (and x_t only exists when t exists).

2. after we get $\tilde{z}_t$, we can calculate mean and variance of the distribution $q(x_{t-1}|x_t)$

3. having $q(x_{t-1}|x_t)$, we can sample an image x_{t-1} given current x_t.

Algorithm 1 Training

1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \text{Uniform}(\{1, \dots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

line2: randomly pick an image. Sample x_0 from true distribution $q(x_0)$
line3: randomly select adding noise timestep t. Sample from 1 to T
line4: randomly generate a Gaussian noise ϵ. Sample from normal distribution $N(0, \mathbf{I})$
line5: calculate loss–gradient descent. Mean square loss.Remember $x_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. So the loss becomes: $\| \epsilon - \epsilon_\theta(x_t, t) \|^2$.

To summarize: from step 2, the image x_0 and step 3 the time t, using forward process to calculate x_t. Now we have a network ϵ_θ, we want it output with a proper noise z_t.

Further explanation: x_t can be calculated from x_0 and t. so: $\| z_t - \epsilon_\theta(x_0, t, z_t) \|^2$.

3.4. Inference

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \dots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

line1: sample a noise from a normal distribution.
line2: do a reverse process repeatedly, and for each step:
line3: randomly sample a normal distribution noise z.
line4: $\frac{1}{\sqrt{\alpha_t}}\big(x_t - \frac{\beta_t}{\sqrt{(1-\bar{\alpha}_t)}} z_t\big)$. From reparametrization trick: $\epsilon = \mu + z \cdot \sigma$, where z is normal noise. From equation 22 we know that μ and σ is calculable from α, so

$$q(x_{t-1}|x_t) = N(\frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\tilde{z}_t, \sigma_t^2)) \tag{23}$$

where $\tilde{z}_t$ is the output of the network, and it can be replaced by: $\epsilon_\theta(x_t, t)$. So the final inference step is:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)) + \sigma_t z \tag{24}$$

To summarize inference process: the network predict and generate $\tilde{z}_t$ so we can sample x_{t-1} from distribution $q(x_{t-1}|x_t)$ given x_t.

References