Diffusion model

Hwichang Jeong [1]

[1] Department of Statistics, Seoul National University, South Korea

September 6, 2024

Table of Contents

Generative model

- The purpose of generative model is to create synthetic data which is similar to the real data.
  1. VAE
  2. DDPM
  3. GAN

Diffusion model

- The Diffusion model transforms a complex data distribution into a noise distribution by slowly injecting noise into data and transforms the noise distribution into a data distribution by slowly removing the noise.

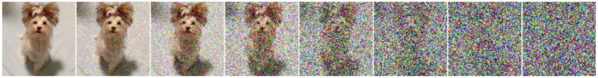

Figure: Process in diffusion model

Notation

- D : Dimension of data.
- $\mathbf{x}_0$: data $\in \mathbb{R}^D$
- $q(\mathbf{x}_0)$: True density function of $\mathbf{x}_0$.

Forward process

- Diffusion model transforms data $\mathbf{x}_0$ sampled from $q(\mathbf{x}_0)$ to noise $\mathbf{x}_T$ through the following markov chain process :

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

  where $t \in \{1, \cdots, T\}$, $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \sigma^2 \mathbf{I}_D)$ is a normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\sigma^2 \mathbf{I}_D$.
- $\beta_1, \cdots, \beta_T$ are small positive hyperparameters.
- Note that when $\int_0^T \beta_t dt \to \infty$ as $T \to \infty$, $q(\mathbf{x}_T)$ converges to standard normal distribution.

Reverse process

- To sample data from noise, we have to calculate the reverse process

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

- However the reverse process is intractable, so we approximate it using Neural Network with parameter θ.

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I}_D)$$

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

Objective function

- Diffusion model is trained by maximizing lower bound of negative log likelihood

$$\mathbb{E}_q\left[-\log p_\theta\left(\mathbf{x}_0\right)\right] \leq \mathbb{E}_q\left[-\log p\left(\mathbf{x}_T\right)-\sum_{t \geq 1} \log \frac{p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)}{q\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}\right)}\right]=: L$$

- For efficient training, we decomposed L to

$$\mathbb{E}_q[\underbrace{D_{\mathrm{KL}}\left(q\left(\mathbf{x}_T \mid \mathbf{x}_0\right) \| p\left(\mathbf{x}_T\right)\right)}_{L_T} + \sum_{t>1} \underbrace{D_{\mathrm{KL}}\left(q\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0\right) \| p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)\right)}_{L_{t-1}} \underbrace{-\log p_\theta\left(\mathbf{x}_0 \mid \mathbf{x}_1\right)}_{L_0}]$$

where D_{KL} is KL-divergence.

Objective function

- When $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=1}^{t} \alpha_s$, we have

$$q(\mathbf{x}_t \mid \mathbf{x}_0) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1 - \bar{\alpha}_t)\,\mathbf{I}_D\right)$$

- Also when conditioned on $\mathbf{x}_0$

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

$$\text{where} \quad \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1 - \bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}\,(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t}\mathbf{x}_t \quad \text{and} \quad \tilde{\beta}_t := \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t}\beta_t$$

Objective function

- L_T : constant w.r.t θ
- $L_{1:T-1}$: When $p_\theta\left(\mathbf{x}_{t-1} \mid \mathbf{x}_t\right)=\mathcal{N}\left(\mathbf{x}_{t-1} ; \boldsymbol{\mu}_\theta\left(\mathbf{x}_t, t\right), \sigma_t^2 \mathbf{I}_D\right)$ for $1<t \leq T$,

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

  where C is constant does not depend on θ.
- L_0 : When $p_\theta\left(\mathbf{x}_0 \mid \mathbf{x}_1\right)=\mathcal{N}\left(\mathbf{x}_0 ; \boldsymbol{\mu}_\theta\left(\mathbf{x}_1, 1\right), \sigma_1^2 I\right)$, just calculate.
- We assume that $\sigma_t^2=\tilde{\beta}_t$ or β_t.

Objective function

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

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

- If we parametrize, $\boldsymbol{\mu}_\theta(\mathbf{x}_t, t)$ as $\frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right)$

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

  where ϵ_θ is a function approximator to predict ϵ form $\mathbf{x}_t$.
- Experimentally, using unweighted loss and expectation of t shows better results

$$\mathbb{E}_{\mathbf{x}_0, \boldsymbol{\epsilon}, t}\left[\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right)\right\|^2\right]$$

  where $t \sim \text{Unif}\{1, \cdots, T\}$, ϵ_θ is a function approximator to predict ϵ form $\mathbf{x}_t$.

Sampling algorithm

Algorithm 2 Sampling

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

Figure: Sampling algorithm

DDPM as score-matching

- Note that for constants $c_2, \cdots, c_T$,

$$L_{t-1} - C = \mathbb{E}_{\mathbf{x}_0, \mathbf{x}_t}\left[c_t \left\| \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t \mid \mathbf{x}_0) + \frac{\boldsymbol{\epsilon}_\theta\left(\mathbf{x}_t, t\right)}{\sqrt{1-\bar{\alpha}_t}} \right\|^2\right]$$

- DDPM estimates negative gradients of logarithm of conditional densities.

Table of Contents

Score based model

- We can generally express continuous diffusion process as SDE

$$d\mathbf{x}_t = f(\mathbf{x}_t, t)dt + g(t)dw$$

where $f(\cdot, t) : \mathbb{R}^d \to \mathbb{R}^d$ is vector valued function called drift coefficient of $\mathbf{x}_t$ and $g : \mathbb{R} \to \mathbb{R}$ is a scalar function called diffusion coefficient of $\mathbf{x}_t$ and w is a Wiener process.

- It is known that reverse of diffusion process is also a diffusion process

$$d\mathbf{x}_t = [f(\mathbf{x}_t, t) - g(t)^2 \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t)]dt + g(t)d\bar{w}$$

where $\bar{w}$ is Wiener process when time flows backwards.

Score based model

- The aim of score based generative model is to estimate $\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t)$.

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_t \left\{ \lambda(t) \mathbb{E}_{\mathbf{x}_t} \left[\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t) \|_2^2 \right] \right\}$$

where $\lambda : [0, T] \to \mathbb{R}^+$ is a positive weight function.

Objective function

- Except constant, it is equivalent to

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathbb{E}_t \left\{ \lambda(t) \mathbb{E}_{\mathbf{x}_0} \mathbb{E}_{\mathbf{x}_t | \mathbf{x}_0} \left[\| \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t \mid \mathbf{x}_0) \|_2^2 \right] \right\}$$

- After estimate θ^* as $\hat{\theta}$, we sample using discretize reverse-time *SDE*:

$$\mathbf{x}_{t-1} = \mathbf{x}_t - \mathbf{f}(\mathbf{x}_t, t) + g(t)^2 \mathbf{s}_{\hat{\theta}}(\mathbf{x}_t, t) + g(t)\epsilon_t,$$

where $\epsilon_t \sim N(\mathbf{0}, \mathbf{I}_D)$.

DDPM as continuous SDE

- In DDPM, $\mathbf{x}_t$ can be represented as

$$\mathbf{x}_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\epsilon_{t-1}$$

  where $\epsilon_i \sim N(\mathbf{0}, \mathbf{I}_D)$ for $i = 1, \cdots, t$.
- And $\mathbf{x}_{t-1}$ is

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{1-\beta_t}}(\mathbf{x}_t + \beta_t s_{\theta^*}(\mathbf{x}_t, t)) + \sqrt{\beta_t}\epsilon_t$$

  when $s_\theta(\mathbf{x}_t, t) = -\frac{\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{1-\bar{\alpha}_t}}$ and θ^* is a true parameter.

DDPM as continuous SDE

- As t goes to ∞, it converges to SDE

$$d\mathbf{x}_t = -\frac{1}{2}\beta_t \mathbf{x}_t dt + \sqrt{\beta_t} dw$$

where w is Wiener process.

Example

- VE SDE : $d\mathbf{x}_t = \sqrt{\frac{d\beta_t^2}{dt}}dw$ (NSCN)
- VP SDE : $d\mathbf{x}_t = -\frac{1}{2}\beta_t\mathbf{x}_t dt + \sqrt{\beta_t}dw$
- sub VP SDE : $d\mathbf{x}_t = -\frac{1}{2}\beta_t\mathbf{x}_t dt + \sqrt{\beta_t(1 - e^{-2\int_0^t \beta_s ds})}dw$

Table of Contents

Theoretical result

- Ornstein-Ulhenback (OU) process:

$$dx = -\frac{1}{2}\beta_t x dt + \sqrt{\beta_t} dw$$

- Let q be the true density of $\mathbf{x}_0$ which is in Besov space $B^s_{a,b}$ and $\hat{q}$ be estimated density of $\hat{x}_0$ which obtained through the process below:

$$\hat{\mathbf{x}}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\hat{\mathbf{x}}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_{\hat{\theta}}(\hat{\mathbf{x}}_t, t)\right) + \sigma_t \mathbf{z}$$

where $z \sim \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}_D)$ then,

$$TV(\hat{q}, q) \asymp n^{-s/(2s+D)}$$

where n is size of data and TV is total variance distance.