Deep Generative Models

Lecture 12

Roman Isachenko

Moscow Institute of Physics and Technology

2023, Autumn

Recap of previous lecture

SDE basics

Let define stochastic process $\mathbf{x}(t)$ with initial condition $\mathbf{x}(0) \sim p_0(\mathbf{x})$:

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

where $\mathbf{w}(t)$ is the standard Wiener process (Brownian motion)

$$\mathbf{w}(t) - \mathbf{w}(s) \sim \mathcal{N}(0, (t-s)\mathbf{I}), \quad d\mathbf{w} = \boldsymbol{\epsilon} \cdot \sqrt{dt}, \text{ where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

Langevin dynamics

Let $\mathbf{x}_0$ be a random vector. Then under mild regularity conditions for small enough η samples from the following dynamics

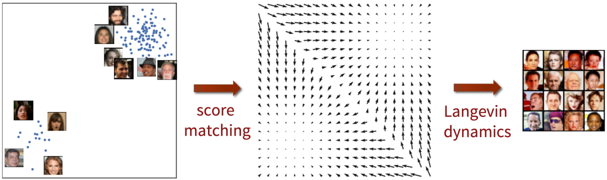
$$\mathbf{x}_{t+1} = \mathbf{x}_t + \eta \frac{1}{2} \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \boldsymbol{\theta}) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

will comes from $p(\mathbf{x}|\boldsymbol{\theta})$.

The density $p(\mathbf{x}|\boldsymbol{\theta})$ is a **stationary** distribution for the Langevin SDE.

Welling M. Bayesian Learning via Stochastic Gradient Langevin Dynamics, 2011

Recap of previous lecture



Data samples $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p(\mathbf{x})$ | Scores $\mathbf{s}_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p(\mathbf{x})$ | New samples

Theorem (implicit score matching)

$$\frac{1}{2}\mathbb{E}_\pi \Big\| \mathbf{s}(\mathbf{x}, \boldsymbol{\theta}) - \nabla_{\mathbf{x}} \log \pi(\mathbf{x}) \Big\|_2^2 = \mathbb{E}_\pi \left[\frac{1}{2} \| \mathbf{s}(\mathbf{x}, \boldsymbol{\theta}) \|_2^2 + \operatorname{tr}\Big(\nabla_{\mathbf{x}} \mathbf{s}(\mathbf{x}, \boldsymbol{\theta})\Big) \right] + \text{const}$$

1. The left hand side is intractable due to unknown $\pi(\mathbf{x})$ – **denoising score matching**.
2. The right hand side is complex due to Hessian matrix – **sliced score matching (Hutchinson's trace estimation)**.

Song Y. Generative Modeling by Estimating Gradients of the Data Distribution, blog post, 2021

Recap of previous lecture

Let perturb original data by normal noise $p(\mathbf{x}'|\mathbf{x}, \sigma) = \mathcal{N}(\mathbf{x}'|\mathbf{x}, \sigma^2\mathbf{I})$

$$\pi(\mathbf{x}'|\sigma) = \int \pi(\mathbf{x}) p(\mathbf{x}'|\mathbf{x}, \sigma) d\mathbf{x}.$$

Then the solution of

$$\frac{1}{2}\mathbb{E}_{\pi(\mathbf{x}'|\sigma)}\big\|\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma) - \nabla_{\mathbf{x}'} \log \pi(\mathbf{x}'|\sigma)\big\|_2^2 \to \min_{\boldsymbol{\theta}}$$

satisfies $\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma) \approx \mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, 0) = \mathbf{s}(\mathbf{x}', \boldsymbol{\theta})$ if σ is small enough.

Theorem (denoising score matching)

$$\mathbb{E}_{\pi(\mathbf{x}'|\sigma)}\big\|\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma) - \nabla_{\mathbf{x}'} \log \pi(\mathbf{x}'|\sigma)\big\|_2^2 =$$

$$= \mathbb{E}_{\pi(\mathbf{x})}\mathbb{E}_{p(\mathbf{x}'|\mathbf{x},\sigma)}\big\|\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma) - \nabla_{\mathbf{x}'} \log p(\mathbf{x}'|\mathbf{x}, \sigma)\big\|_2^2 + \text{const}(\boldsymbol{\theta})$$

Here $\nabla_{\mathbf{x}'} \log p(\mathbf{x}'|\mathbf{x}, \sigma) = -\frac{\mathbf{x}'-\mathbf{x}}{\sigma^2}$.

- The RHS does not need to compute $\nabla_{\mathbf{x}'} \log \pi(\mathbf{x}'|\sigma)$ and even more $\nabla_{\mathbf{x}'} \log \pi(\mathbf{x}')$.
- $\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma)$ tries to **denoise** a corrupted sample.
- Score function $\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma)$ parametrized by σ.

Vincent P. A Connection Between Score Matching and Denoising Autoencoders, 2010

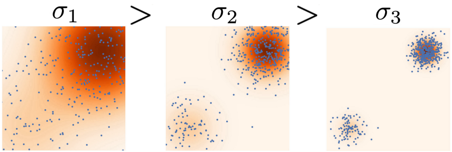
Recap of previous lecture

Noise conditioned score network

- Define the sequence of noise levels: $\sigma_1 > \sigma_2 > \dots > \sigma_L$.
- Train denoised score function $\mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma)$ for each noise level:

$$\sum_{l=1}^{L} \sigma_l^2 \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{p(\mathbf{x}'|\mathbf{x}, \sigma_l)} \left\| \mathbf{s}(\mathbf{x}', \boldsymbol{\theta}, \sigma_l) - \nabla'_{\mathbf{x}} \log p(\mathbf{x}'|\mathbf{x}, \sigma_l) \right\|_2^2 \to \min_{\boldsymbol{\theta}}$$

- Sample from **annealed** Langevin dynamics (for $l = 1, \dots, L$).



Song Y. et al. Generative Modeling by Estimating Gradients of the Data Distribution, 2019

Outline

1. Gaussian diffusion process

2. Denoising diffusion probabilistic model (DDPM)
   - Objective of DDPM
   - Reparametrization of DDPM

Outline

1. Gaussian diffusion process

2. Denoising diffusion probabilistic model (DDPM)
   - Objective of DDPM
   - Reparametrization of DDPM

Forward gaussian diffusion process

Let $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$, $\beta \in (0, 1)$. Define the Markov chain

$$\mathbf{x}_t = \sqrt{1-\beta} \cdot \mathbf{x}_{t-1} + \sqrt{\beta} \cdot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, 1);$$

$$q(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t | \sqrt{1-\beta} \cdot \mathbf{x}_{t-1}, \beta \cdot \mathbf{I}).$$

Statement 1

Applying the Markov chain to samples from any $\pi(\mathbf{x})$ we will get $\mathbf{x}_\infty \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$. Here $p_\infty(\mathbf{x})$ is a **stationary** distribution:

$$p_\infty(\mathbf{x}) = \int q(\mathbf{x}|\mathbf{x}') p_\infty(\mathbf{x}') d\mathbf{x}'.$$

Statement 2

Denote $\bar{\alpha}_t = \prod_{s=1}^{t} (1 - \beta_s)$. Then

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$$

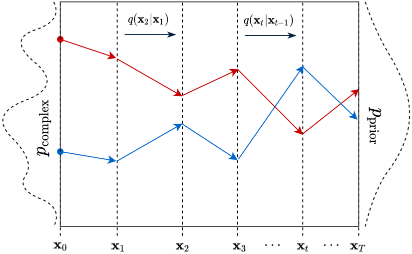
$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t | \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0, (1 - \bar{\alpha}_t) \cdot \mathbf{I}).$$

We could sample from any timestamp using only $\mathbf{x}_0$!

Sohl-Dickstein J. Deep Unsupervised Learning using Nonequilibrium Thermodynamics, 2015

Forward gaussian diffusion process

Diffusion refers to the flow of particles from high-density regions towards low-density regions.

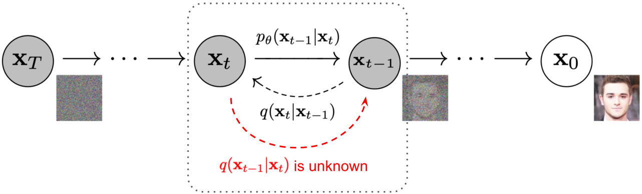


1. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;
2. $\mathbf{x}_t = \sqrt{1-\beta} \cdot \mathbf{x}_{t-1} + \sqrt{\beta} \cdot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1),\ t \geq 1$;
3. $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$, where $T >> 1$.

If we are able to invert this process, we will get the way to sample $\mathbf{x} \sim \pi(\mathbf{x})$ using noise samples $p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$.
Now our goal is to revert this process.

Das A. An introduction to Diffusion Probabilistic Models, blog post, 2021

Reverse gaussian diffusion process



Let define the reverse process

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{x}_{t-1}|\boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t), \boldsymbol{\sigma}^2(\mathbf{x}_t, \boldsymbol{\theta}, t)\right)$$

Forward process

1. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;
2. $\mathbf{x}_t = \sqrt{1-\beta} \cdot \mathbf{x}_{t-1} + \sqrt{\beta} \cdot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$, $t \geq 1$;
3. $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$.

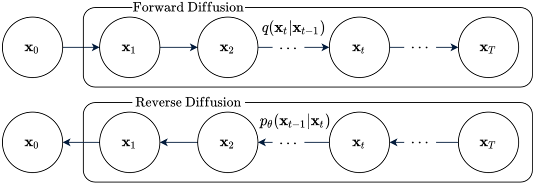
Reverse process

1. $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$;
2. $\mathbf{x}_{t-1} = \boldsymbol{\sigma}(\mathbf{x}_t, \boldsymbol{\theta}, t) \cdot \boldsymbol{\epsilon} + \boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t)$;
3. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;

Note: The forward process does not have any learnable parameters!

Weng L. What are Diffusion Models?, blog post, 2021

Gaussian diffusion model as VAE



- Let treat $\mathbf{z} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$ as a latent variable (**note:** each $\mathbf{x}_t$ has the same size).
- Variational posterior distribution (**note:** there is no learnable parameters)
$$q(\mathbf{z}|\mathbf{x}) = q(\mathbf{x}_1, \dots, \mathbf{x}_T|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}).$$
- Probabilistic model
$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}|\boldsymbol{\theta})$$
- Generative distribution and prior
$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}); \quad p(\mathbf{z}|\boldsymbol{\theta}) = \prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) \cdot p(\mathbf{x}_T)$$

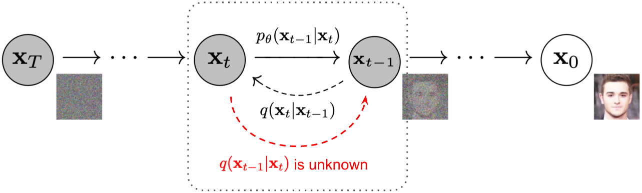
Das A. An introduction to Diffusion Probabilistic Models, blog post, 2021

Outline

1. Gaussian diffusion process

2. Denoising diffusion probabilistic model (DDPM)
   - Objective of DDPM
   - Reparametrization of DDPM

Reverse gaussian diffusion process



Forward process

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

Reverse process

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t) = \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1})}{q(\mathbf{x}_t)} \approx p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})$$

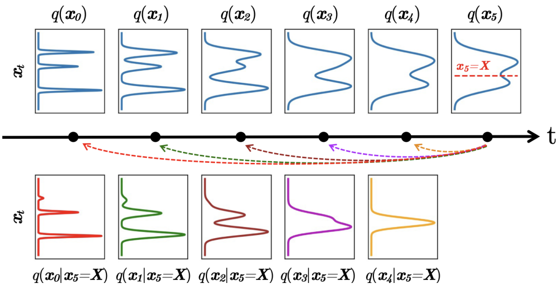
- $q(\mathbf{x}_{t-1})$, $q(\mathbf{x}_t)$ are intractable.
- If β_t is small enough, $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ will be Gaussian (Feller, 1949).

Feller W. On the theory of stochastic processes, with particular reference to applications, 1949

Reverse gaussian diffusion process

Reverse process

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t) \approx p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{x}_{t-1}|\boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t), \boldsymbol{\sigma}^2(\mathbf{x}_t, \boldsymbol{\theta}, t)\right)$$



Important distribution

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} = \mathcal{N}(\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I})$$

Xiao Z., Kreis K., Vahdat A. Tackling the generative learning trilemma with denoising diffusion GANs, 2021

Outline

Objective of DDPM

ELBO

$$\log p(\mathbf{x}|\boldsymbol{\theta}) \geq \mathbb{E}_{q(\mathbf{z}|\mathbf{x})} \log \frac{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z}|\mathbf{x})} = \mathcal{L}(q, \boldsymbol{\theta}) \to \max_{q, \boldsymbol{\theta}}$$

Derivation

$$\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_0, \mathbf{x}_{1:T}|\boldsymbol{\theta})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} = \mathbb{E}_q \log \frac{\prod_{t=1}^T p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) p(\mathbf{x}_T)}{\prod_{t=1}^T q(\mathbf{x}_t|\mathbf{x}_{t-1})}$$

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

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

$$\log \left(\frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} \right) = \log \left(\frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) q(\mathbf{x}_t|\mathbf{x}_0)} \right)$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Objective of DDPM

Derivation

$$\mathcal{L}(q,\boldsymbol{\theta}) = \mathbb{E}_q\left[\log p(\mathbf{x}_T) + \log\frac{p(\mathbf{x}_0|\mathbf{x}_1,\boldsymbol{\theta})}{q(\mathbf{x}_1|\mathbf{x}_0)} + \sum_{t=2}^{T}\log\left(\frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t,\boldsymbol{\theta})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right)\right]$$

$$= \mathbb{E}_q\left[\log p(\mathbf{x}_T) + \log\frac{p(\mathbf{x}_0|\mathbf{x}_1,\boldsymbol{\theta})}{q(\mathbf{x}_1|\mathbf{x}_0)} + \sum_{t=2}^{T}\log\left(\frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t,\boldsymbol{\theta})}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}\right) + \right.$$

$$\left. + \sum_{t=2}^{T}\log\left(\frac{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)}\right)\right] = \mathbb{E}_q\left[\log\frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} + \log p(\mathbf{x}_0|\mathbf{x}_1,\boldsymbol{\theta}) + \right.$$

$$\left. + \sum_{t=2}^{T}\log\left(\frac{p(\mathbf{x}_{t-1}|\mathbf{x}_t,\boldsymbol{\theta})}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)}\right)\right] = \mathbb{E}_q\Big[-KL(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T)) + $$

$$+ \log p(\mathbf{x}_0|\mathbf{x}_1,\boldsymbol{\theta}) - \sum_{t=2}^{T} KL\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)||p(\mathbf{x}_{t-1}|\mathbf{x}_t,\boldsymbol{\theta})\big)\Big]$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Objective of DDPM

$$\mathcal{L}(q, \boldsymbol{\theta}) = \mathbb{E}_q \Big[\log p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}) - KL\big(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T)\big) - \\ - \sum_{t=2}^{T} \underbrace{KL\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)||p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})\big)}_{\mathcal{L}_t} \Big]$$

- First term is a decoder distribution

$$\log p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}) = \log \mathcal{N}\big(\mathbf{x}_0|\boldsymbol{\mu}(\mathbf{x}_1, \boldsymbol{\theta}, t), \boldsymbol{\sigma}^2(\mathbf{x}_1, \boldsymbol{\theta}, t)\big)$$

- Second term is constant ($p(\mathbf{x}_T)$ is a standard Normal, $q(\mathbf{x}_T|\mathbf{x}_0)$ is a non-parametrical Normal).
- $\mathcal{L}_t$ is a KL between two normal distributions:

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

$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0)$ and $\tilde{\beta}_t$ have analytical formulas (we omit it) and they are both dependent on β_t.

Ho J. Denoising Diffusion Probabilistic Models, 2020

Outline

1. Gaussian diffusion process

2. Denoising diffusion probabilistic model (DDPM)
   - Objective of DDPM
   - Reparametrization of DDPM

Gaussian diffusion model as VAE

$$
\begin{aligned}
q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= \mathcal{N}(\mathbf{x}_{t-1}|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}), \\
p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) &= \mathcal{N}\big(\mathbf{x}_{t-1}|\boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t), \boldsymbol{\sigma}^2(\mathbf{x}_t, \boldsymbol{\theta}, t)\big)
\end{aligned}
$$

- Assume $\boldsymbol{\sigma}^2(\mathbf{x}_t, \boldsymbol{\theta}, t) = \tilde{\beta}_t \mathbf{I}$.
- Use KL formula between two normal distributions:

$$
\begin{aligned}
\mathcal{L}_t &= KL\Big(\mathcal{N}\big(\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}\big) || \mathcal{N}\big(\boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t), \tilde{\beta}_t \mathbf{I}\big)\Big) \\
&= \mathbb{E}_{\boldsymbol{\epsilon}} \left[\frac{1}{2\tilde{\beta}_t} \left\| \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t) \right\|^2 \right] \\
&= \mathbb{E}_{\boldsymbol{\epsilon}} \left[\frac{1}{2\tilde{\beta}_t} \left\| \frac{1}{\sqrt{1-\beta_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon} \right) - \boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t) \right\|^2 \right]
\end{aligned}
$$

Here we used the analytic expression for $\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0)$.

Reparametrization

$$
\boldsymbol{\mu}(\mathbf{x}_t, \boldsymbol{\theta}, t) = \frac{1}{\sqrt{1-\beta_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}(\mathbf{x}_t, \boldsymbol{\theta}, t) \right)
$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Reparametrization of DDPM

KL term

$$\mathcal{L}_t = \mathbb{E}_{\boldsymbol{\epsilon}}\left[\frac{1}{2\tilde{\beta}_t}\left\|\frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\right) - \right.\right.$$

$$\left.\left. - \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}(\mathbf{x}_t, \boldsymbol{\theta}, t)\right)\right\|^2\right] =$$

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

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\cdot\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\cdot\boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$$

At each step of reverse diffusion process we try to predict the noise $\boldsymbol{\epsilon}$ that we used in forward process!

Ho J. Denoising Diffusion Probabilistic Models, 2020

Summary

- Gaussian diffusion process is a Markov chain that injects special form of Gaussian noise to the samples.
- Reverse process allows to sample from the real distribution $\pi(\mathbf{x})$ using samples from noise.
- Diffusion model is a VAE model which reverts gaussian diffusion process using variational inference.
- ELBO of DDPM is a sum of KL terms.
- At each step DDPM predicts the noise used in forward process.