Deep Generative Models

Lecture 9

Roman Isachenko

Moscow Institute of Physics and Technology
Yandex School of Data Analysis

2025, Autumn

Recap of Previous Lecture

Let us perturb the original data with Gaussian noise $q(\mathbf{x}_\sigma|\mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$.

$$q(\mathbf{x}_\sigma) = \int q(\mathbf{x}_\sigma|\mathbf{x})\pi(\mathbf{x})d\mathbf{x}.$$

Then the solution of

$$\frac{1}{2}\mathbb{E}_{q(\mathbf{x}_\sigma)}\left\|\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)\right\|_2^2 \rightarrow \min_{\boldsymbol{\theta}}$$

satisfies $\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma) \approx \mathbf{s}_{\boldsymbol{\theta},0}(\mathbf{x}_0) = \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$ if σ is sufficiently small.

Theorem (Denoising Score Matching)

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

Here, $\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) = -\frac{\mathbf{x}_\sigma - \mathbf{x}}{\sigma^2} = -\frac{\boldsymbol{\epsilon}}{\sigma}$.

- We don't need to compute $\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)$ on the RHS.
- $\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma)$ attempts to **denoise** a corrupted sample.

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

Recap of Previous Lecture

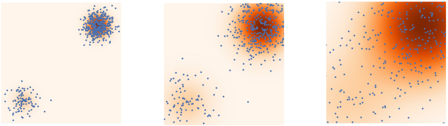
Noise-Conditioned Score Network

- Define a sequence of noise levels: $\sigma_1 < \sigma_2 < \dots < \sigma_T$.
- Train a denoised score function $\mathbf{s}_{\boldsymbol{\theta}, \sigma_t}(\mathbf{x}_t)$ for each noise level:

$$
\sum_{t=1}^{T} \sigma_t^2 \cdot \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x})} \big\| \mathbf{s}_{\boldsymbol{\theta}, \sigma_t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t|\mathbf{x}) \big\|_2^2 \to \min_{\boldsymbol{\theta}}
$$

- Sample using **annealed** Langevin dynamics (for $t = 1, \dots, T$).

$$\sigma_1 < \sigma_2 < \sigma_3$$



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

Recap of Previous Lecture

NCSN Training

1. Obtain a sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Sample noise level $t \sim U\{1, T\}$ and noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.
3. Construct noisy image $\mathbf{x}_t = \mathbf{x}_0 + \sigma_t \cdot \boldsymbol{\epsilon}$.
4. Compute the loss $\mathcal{L} = \sigma_t^2 \cdot \|\mathbf{s}_{\boldsymbol{\theta},\sigma_t}(\mathbf{x}_t) + \frac{\boldsymbol{\epsilon}}{\sigma_t}\|^2$.

NCSN Sampling (Annealed Langevin Dynamics)

- Sample $\mathbf{x}_0 \sim \mathcal{N}(0, \sigma_T^2 \cdot \mathbf{I}) \approx q(\mathbf{x}_T)$.
- Apply L steps of Langevin dynamics:

$$\mathbf{x}_l = \mathbf{x}_{l-1} + \frac{\eta_t}{2} \cdot \mathbf{s}_{\boldsymbol{\theta},\sigma_t}(\mathbf{x}_{l-1}) + \sqrt{\eta_t} \cdot \boldsymbol{\epsilon}_l.$$

- Update $\mathbf{x}_0 := \mathbf{x}_L$ and proceed to the next σ_t.

Song Y. et al. Improved Techniques for Training Score-Based Generative Models, 2020

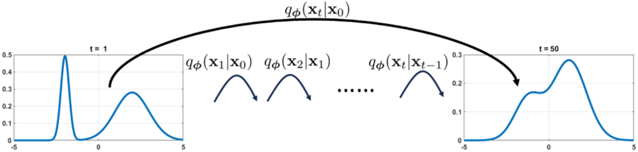
Recap of Previous Lecture

Forward Gaussian Diffusion Process

Let $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$, $\beta_t \ll 1$, $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$.

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

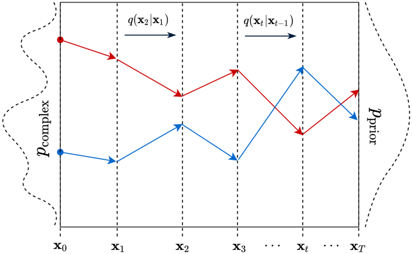
$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1-\beta_t} \cdot \mathbf{x}_{t-1}, \beta_t \cdot \mathbf{I});$$
$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0, (1-\bar{\alpha}_t) \cdot \mathbf{I}).$$



Chan S. Tutorial on Diffusion Models for Imaging and Vision, 2024

Recap of Previous Lecture

Diffusion describes the process where particles migrate from regions of high density to regions of low density.

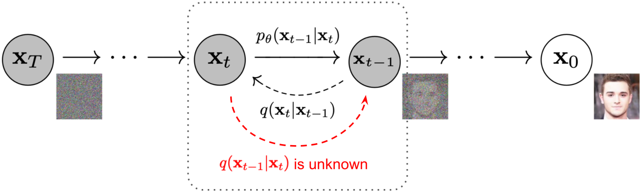


1. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;
2. $\mathbf{x}_t = \sqrt{1-\beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}$, with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$, $t \geq 1$;
3. $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$,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

If we can invert this process, we would have a way to sample $\mathbf{x} \sim \pi(\mathbf{x})$ using noise samples, i.e. $p_\infty(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$.
Hence, our objective becomes to reverse this process.

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

Recap of Previous Lecture



Reverse Process (Ancestral Sampling)

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

The Feller theorem guarantees this approximation is valid.

Forward Process

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

Reverse Process

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

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

Outline

Outline

Conditioned Reverse Distribution

Reverse Kernel (**Intractable**)

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

Conditioned Reverse Kernel (**Tractable**)

$$\begin{aligned}
q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} \\
&= \frac{\mathcal{N}(\sqrt{1-\beta_t}\cdot \mathbf{x}_{t-1}, \beta_t \mathbf{I}) \cdot \mathcal{N}(\sqrt{\bar{\alpha}_{t-1}} \cdot \mathbf{x}_0, (1-\bar{\alpha}_{t-1}) \cdot \mathbf{I})}{\mathcal{N}(\sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0, (1-\bar{\alpha}_t) \cdot \mathbf{I})} \\
&= \mathcal{N}(\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \cdot \mathbf{I})
\end{aligned}$$

Here,

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} \cdot \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} \cdot \mathbf{x}_0;$$

$$\tilde{\beta}_t = \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} = \text{const.}$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Distribution Summary

Forward process maps any distribution $\pi(\mathbf{x})$ to $\mathcal{N}(0, \mathbf{I})$ by injection of noise:

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

Reverse process refers to an intractable distribution that can be approximated by a normal distribution (with unknown parameters) for small β_t:

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

Conditioned reverse process is a normal distribution with known parameters, describing how to denoise a noisy image $\mathbf{x}_t$ when we know the clean image $\mathbf{x}_0$.

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

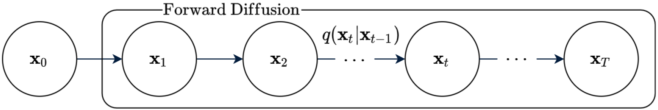
Outline

Gaussian Diffusion Model as VAE

Let's treat $\mathbf{z} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$ as a latent variable (**note:** each $\mathbf{x}_t$ has the same dimension), and $\mathbf{x} = \mathbf{x}_0$ as the observed variable.

Latent Variable Model

$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}|\boldsymbol{\theta})$$



Forward Diffusion

- Variational posterior distribution (encoder)

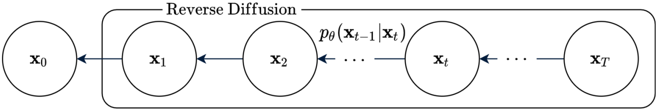
$$q(\mathbf{z}|\mathbf{x}) = q(\mathbf{x}_1, \ldots, \mathbf{x}_T|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}).$$

- **Note:** there are no learnable parameters.

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

Gaussian Diffusion Model as VAE

$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})p(\mathbf{z}|\boldsymbol{\theta})$$



Reverse Diffusion

- Generative distribution (decoder)

$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}).$$

- Prior distribution

$$p(\mathbf{z}|\boldsymbol{\theta}) = p(\mathbf{x}_1, \dots, \mathbf{x}_T|\boldsymbol{\theta}) = \prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) \cdot p(\mathbf{x}_T).$$

Note: This differs from the vanilla VAE due to the complex decoder $p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta})$ and the standard normal prior $p(\mathbf{z})$.

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

Outline

ELBO for Gaussian Diffusion Model

Standard ELBO

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

Derivation

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

- Let's try to decompose the ELBO into individual KL divergence terms.
- We need to replace $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ with $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ in the denominator.
- Let's condition on $\mathbf{x}_0$ to make the reverse $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ tractable.

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian Diffusion Model

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

Derivation (continued)

$$\begin{aligned}
\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}}(\mathbf{x}) &= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)\prod_{t=1}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})} \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)\prod_{t=1}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)} \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta})\prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{q(\mathbf{x}_1|\mathbf{x}_0)\prod_{t=2}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{x}_0)} \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta})\prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{q(\mathbf{x}_1|\mathbf{x}_0)\prod_{t=2}^{T} \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)q(\mathbf{x}_t|\mathbf{x}_0)}{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}} \\
&= \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta})\prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})}{q(\mathbf{x}_T|\mathbf{x}_0)\prod_{t=2}^{T} q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)}
\end{aligned}$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian Diffusion Model

Derivation (continued)

$$\mathcal{L}_{\phi,\theta}(\mathbf{x}) = \mathbb{E}_{q(\mathbf{x}_{1:T}|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)p(\mathbf{x}_0|\mathbf{x}_1,\theta)\prod_{t=2}^{T} p(\mathbf{x}_{t-1}|\mathbf{x}_t,\theta)}{q(\mathbf{x}_T|\mathbf{x}_0)\prod_{t=2}^{T} q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)} =$$

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

$$= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p(\mathbf{x}_0|\mathbf{x}_1,\theta) + \mathbb{E}_{q(\mathbf{x}_T|\mathbf{x}_0)} \log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} +$$

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

$$= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p(\mathbf{x}_0|\mathbf{x}_1,\theta) - \mathrm{KL}\big(q(\mathbf{x}_T|\mathbf{x}_0)\|p(\mathbf{x}_T)\big) -$$

$$- \sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)}\mathrm{KL}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)\|p(\mathbf{x}_{t-1}|\mathbf{x}_t,\theta)\big)}_{\mathcal{L}_t}$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian Diffusion Model

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

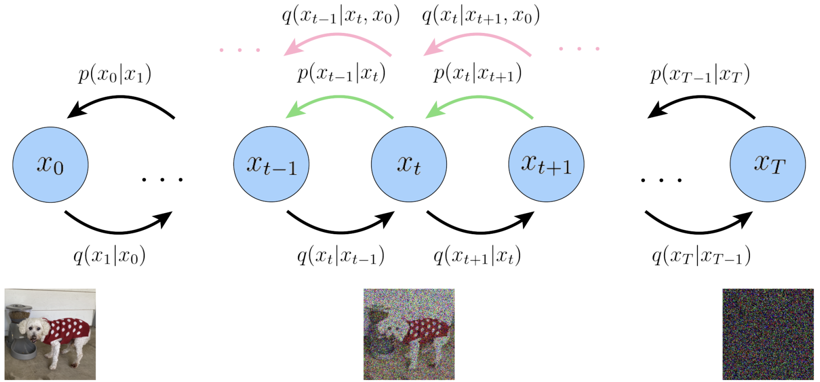
- First term is the decoder distribution

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

  with $\mathbf{x}_1 \sim q(\mathbf{x}_1|\mathbf{x}_0)$.
- Second term is constant:
  - $p(\mathbf{x}_T) = \mathcal{N}(0, \mathbf{I})$;
  - $q(\mathbf{x}_T|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_T} \cdot \mathbf{x}_0, (1 - \bar{\alpha}_T) \cdot \mathbf{I})$.
- Third term is the main contributor to the ELBO.

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian Diffusion Model



$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{KL}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})\big)$$

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

Luo C. Understanding Diffusion Models: A Unified Perspective, 2022

ELBO for Gaussian Diffusion Model

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \mathrm{KL}\big(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})\big)$$

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

Let's assume that

$$\boldsymbol{\sigma}^2_{\boldsymbol{\theta},t}(\mathbf{x}_t) = \tilde{\beta}_t \mathbf{I} \quad \Rightarrow \quad p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}\big(\mathbf{x}_{t-1}|\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t), \tilde{\beta}_t \mathbf{I}\big).$$

Theoretically, the optimal $\boldsymbol{\sigma}^2_{\boldsymbol{\theta},t}(\mathbf{x}_t)$ lies in $[\tilde{\beta}_t, \beta_t]$:

- β_t is optimal for $\mathbf{x}_0 \sim \mathcal{N}(0, \mathbf{I})$;
- $\tilde{\beta}_t$ is optimal for $\mathbf{x}_0 \sim \delta(\mathbf{x}_0 - \mathbf{x}^*)$.

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian Diffusion Model

Training

1. Obtain a sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Generate a noisy image $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}$, with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.
3. Compute the ELBO

$$\mathcal{L}_{\boldsymbol{\phi},\boldsymbol{\theta}}(\mathbf{x}) = \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}) - \mathrm{KL}\big(q(\mathbf{x}_T|\mathbf{x}_0) \| p(\mathbf{x}_T)\big) - \\ - \sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \left[\frac{1}{2\tilde{\beta}_t} \left\| \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) \right\|^2 \right]}_{\mathcal{L}_t}$$

Sampling

1. Sample $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$.
2. Denoise: $\mathbf{x}_{t-1} = \boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) + \sqrt{\tilde{\beta}_t} \cdot \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.

Ho J. Denoising Diffusion Probabilistic Models, 2020

Outline

Reparametrization of DDPM

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)}\left[\frac{1}{2\tilde{\beta}_t}\left\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t)\right\|^2\right]$$

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} \cdot \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} \cdot \mathbf{x}_0$$

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon} \quad \Rightarrow \quad \mathbf{x}_0 = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}}{\sqrt{\bar{\alpha}_t}}$$

- There is a linear relationship between $\boldsymbol{\epsilon}$, $\mathbf{x}_t$, and $\mathbf{x}_0$.
- Let's try to rewrite this mean using only $\mathbf{x}_t$ and $\boldsymbol{\epsilon}$.

$$\begin{aligned}\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \boldsymbol{\epsilon}) &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} \cdot \mathbf{x}_t + \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t} \cdot \left(\frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}}{\sqrt{\bar{\alpha}_t}}\right) \\ &= \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{\alpha_t(1-\bar{\alpha}_t)}} \cdot \boldsymbol{\epsilon}\end{aligned}$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Reparametrization of DDPM

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)}\left[\frac{1}{2\tilde{\beta}_t}\left\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) - \boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t)\right\|^2\right]$$

Reparametrization

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{\alpha_t(1-\bar{\alpha}_t)}} \cdot \boldsymbol{\epsilon}$$

$$\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{\alpha_t(1-\bar{\alpha}_t)}} \cdot \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}(\mathbf{x}_t)$$

$$\mathcal{L}_t = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I})}\left[\frac{(1-\alpha_t)^2}{2\tilde{\beta}_t\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}(\mathbf{x}_t)\right\|^2\right]$$

$$= \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I})}\left[\frac{(1-\alpha_t)^2}{2\tilde{\beta}_t\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}\left(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}\right)\right\|^2\right]$$

At every step of the reverse process, we attempt to predict the noise $\boldsymbol{\epsilon}$ that was used in the forward diffusion process!

Ho J. Denoising Diffusion Probabilistic Models, 2020

Reparametrization of DDPM

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

$$\mathcal{L}_t = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I})}\left[\frac{(1-\alpha_t)^2}{2\tilde{\beta}_t\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}\big(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}\big)\right\|^2\right]$$

Let's drop the scaling coefficient.

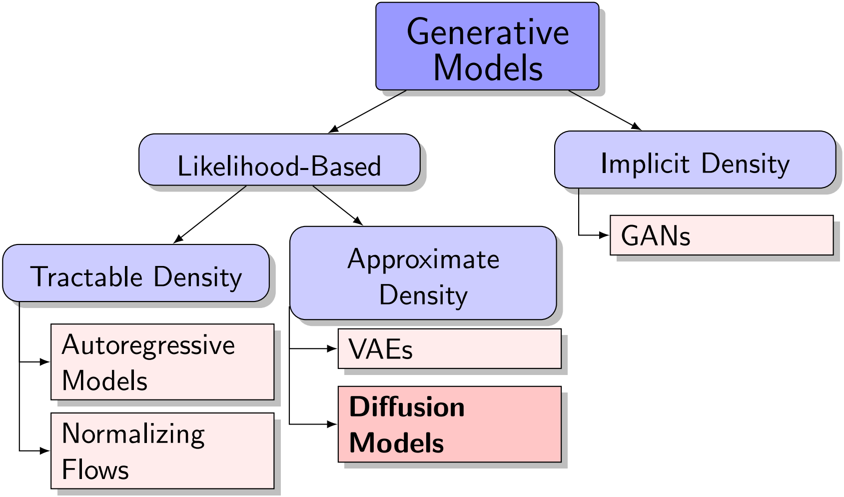
Simplified Objective

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t\sim U\{2,T\}}\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I})}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}\big(\sqrt{\bar{\alpha}_t}\cdot\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\cdot\boldsymbol{\epsilon}\big)\right\|^2$$

Ho J. Denoising Diffusion Probabilistic Models, 2020

Outline

Generative Models Zoo



Denoising Diffusion Probabilistic Model (DDPM)

DDPM is a VAE Model

- The encoder is a fixed Gaussian Markov chain $q(\mathbf{x}_1, \dots, \mathbf{x}_T|\mathbf{x}_0)$.
- The latent variable is hierarchical (at each step, its dimension equals the input's).
- The decoder is a simple Gaussian model $p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta})$.
- The prior distribution is given by a parametric Gaussian Markov chain $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta})$.

Forward Process

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

Reverse Process

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

Denoising Diffusion Probabilistic Model (DDPM)

Training

1. Obtain a sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Sample time index $t \sim U\{1, T\}$ and noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.
3. Generate noisy image $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}$.
4. Compute the loss $\mathcal{L}_{\text{simple}} = \|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}(\mathbf{x}_t)\|^2$.

Sampling (Ancestral Sampling)

1. Sample $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$.
2. Compute the mean of $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t), \tilde{\beta}_t \cdot \mathbf{I})$:

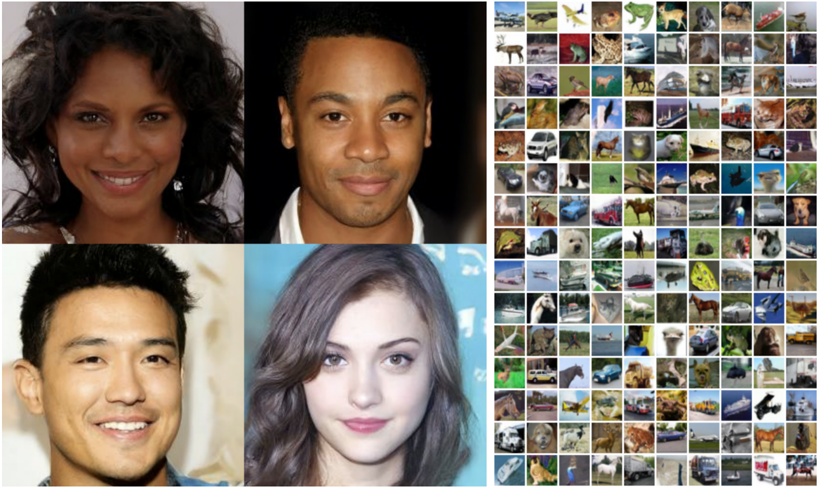
$$\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}(\mathbf{x}_t)$$

3. Denoise: $\mathbf{x}_{t-1} = \boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) + \sqrt{\tilde{\beta}_t} \cdot \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.

Ho J. Denoising Diffusion Probabilistic Models, 2020

Denoising Diffusion Probabilistic Model (DDPM)

Samples



Ho J. Denoising Diffusion Probabilistic Models, 2020

Summary

- DDPM approximates the reverse process using normality assumptions.
- DDPM can be interpreted as a VAE with a hierarchy of latent variables.
- The ELBO for DDPM may be formulated as a sum over many KL divergence terms.
- At each step, DDPM predicts the noise that was injected in the forward process.
- DDPM is a VAE model that tries to invert the forward diffusion process via variational inference.
- DDPMs are quite slow, since the model must be applied T times for sampling.