Deep Generative Models

Lecture 9

Roman Isachenko

AI Masters

2025, Spring

Recap of previous lecture

Let perturb original data by normal 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

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

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 small enough.

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 do not need to compute $\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)$ at the RHS.
- $\mathbf{s}_{\boldsymbol{\theta},\sigma}(\mathbf{x}_\sigma)$ tries to **denoise** a corrupted sample.

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

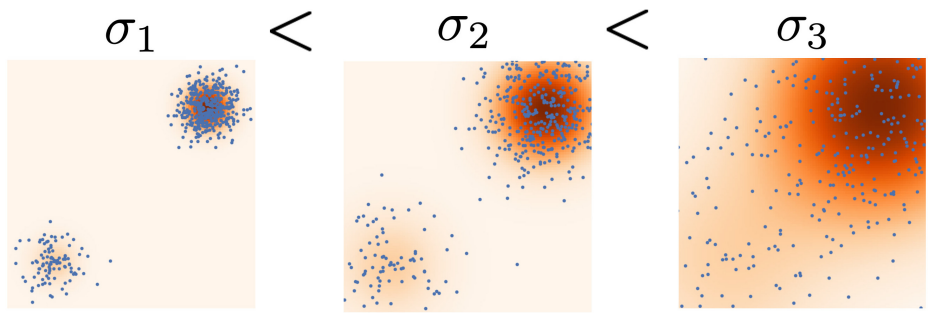
Recap of previous lecture

Noise conditioned score network

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

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

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



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

Recap of previous lecture

NCSN training

1. Get the sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Sample noise level $t \sim U\{1, T\}$ and the noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.
3. Get noisy image $\mathbf{x}_t = \mathbf{x}_0 + \sigma_t \cdot \boldsymbol{\epsilon}$.
4. Compute 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 dynamic

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

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

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

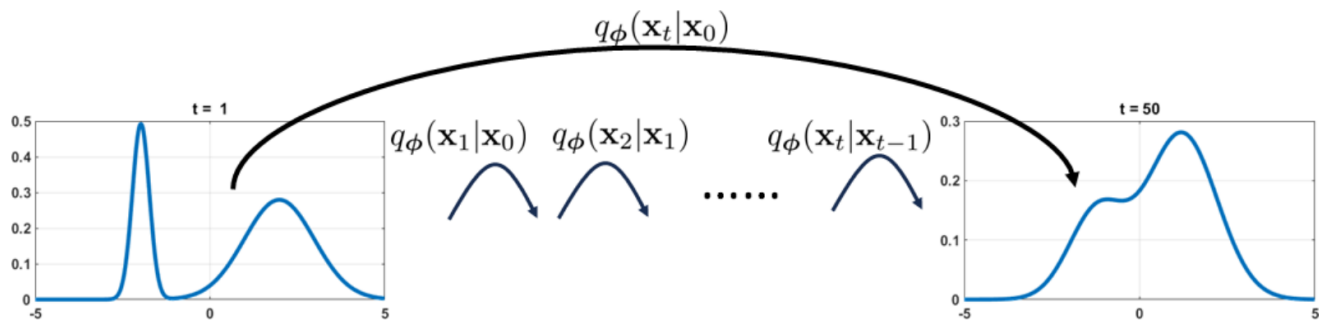
Recap of previous lecture

Forward Gaussian diffusion process

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

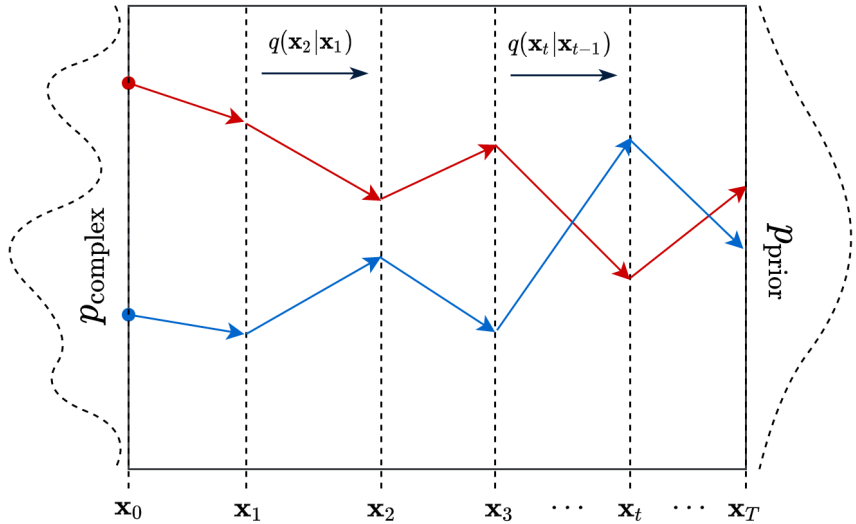
$$\mathbf{x}_t = \sqrt{1-\beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I});$$
$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t} \cdot \boldsymbol{\epsilon}, \quad \text{where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

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



Recap of previous lecture

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

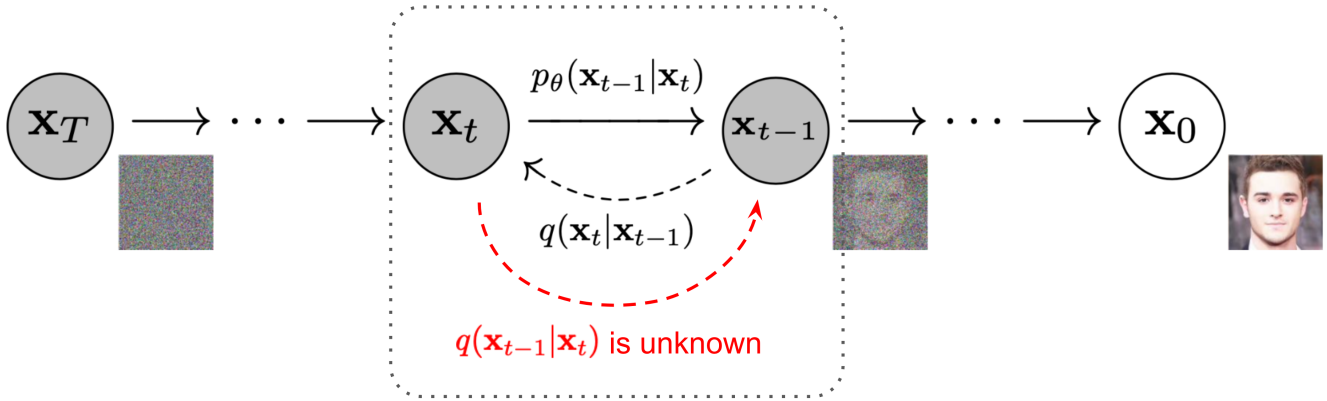


1. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;
2. $\mathbf{x}_t = \sqrt{1-\beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$, $t \geq 1$;
3. $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$,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

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

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

Recap of previous lecture



Reverse process (ancestral sampling)

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

Feller theorem shows that it is a reasonable assumption.

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. $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$;

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

Outline

1. Denoising Diffusion Probabilistic Model (DDPM)
   - Gaussian diffusion model as VAE
   - ELBO derivation
   - Reparametrization
   - Overview

Outline

1. Denoising Diffusion Probabilistic Model (DDPM)
   - Gaussian diffusion model as VAE
   - ELBO derivation
   - Reparametrization
   - Overview

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 goes from any distribution $\pi(\mathbf{x})$ to $\mathcal{N}(0, \mathbf{I})$ via noise injection.

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

Reverse process is Intractable distribution that is able to be approximated by Normal (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 Normal with the known parameters, which defines how to denoise a noisy image $\mathbf{x}_t$ with access to what the final, completely denoised image $\mathbf{x}_0$ should be.

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

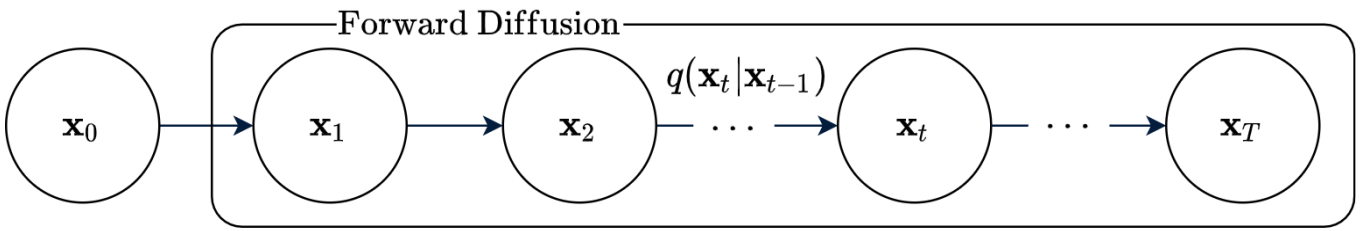
Outline

Gaussian diffusion model as VAE

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

Latent Variable Model

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



Forward diffusion

- Variational posterior distribution (encoder)

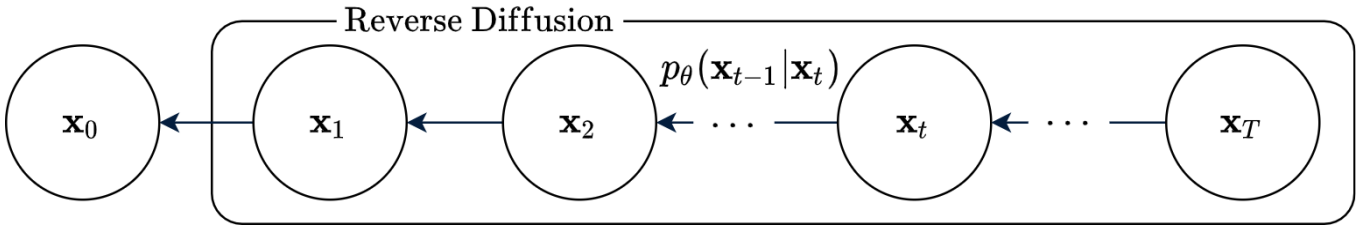
$$q(\mathbf{z}|\mathbf{x}) = q(\mathbf{x}_1, \ldots, \mathbf{x}_T|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}).$$

- **Note:** there is 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

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

Note: this differs from the vanilla VAE with 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 try to decompose the ELBO to separate KL divergences.
- We have to swap the distribution $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ to $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ in the denominator.
- Let add conditioning on $\mathbf{x}_0$ to make reverse distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ 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)

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian diffusion model

Derivation (continued)

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

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

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

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

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

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian diffusion model

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

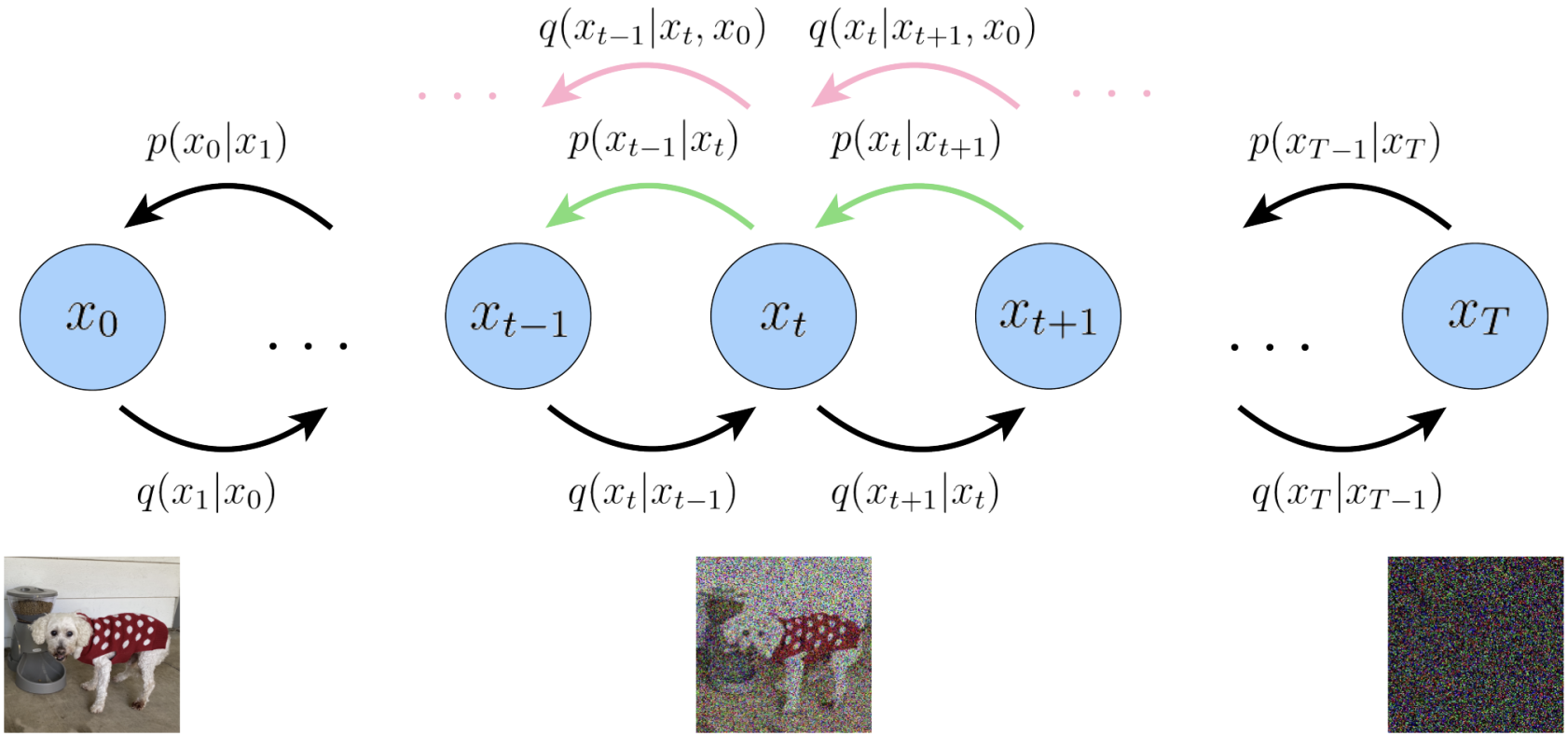
- First term is a decoder distribution

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

  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 makes the main contribution to the ELBO.

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian diffusion model



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

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

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

ELBO for Gaussian diffusion model

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

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

Let assume

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

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

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

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

ELBO for Gaussian diffusion model

Training

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

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

Sampling

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

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 linear dependence between $\boldsymbol{\epsilon}$, $\mathbf{x}_t$, $\mathbf{x}_0$.
- Let try to rewrite this mean in terms of $\mathbf{x}_t$ 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 each step of the reverse diffusion process we try to predict the noise ϵ that we used in the forward diffusion process!

Ho J. Denoising Diffusion Probabilistic Models, 2020

Reparametrization of DDPM

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

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

Let drop the scaling coefficient.

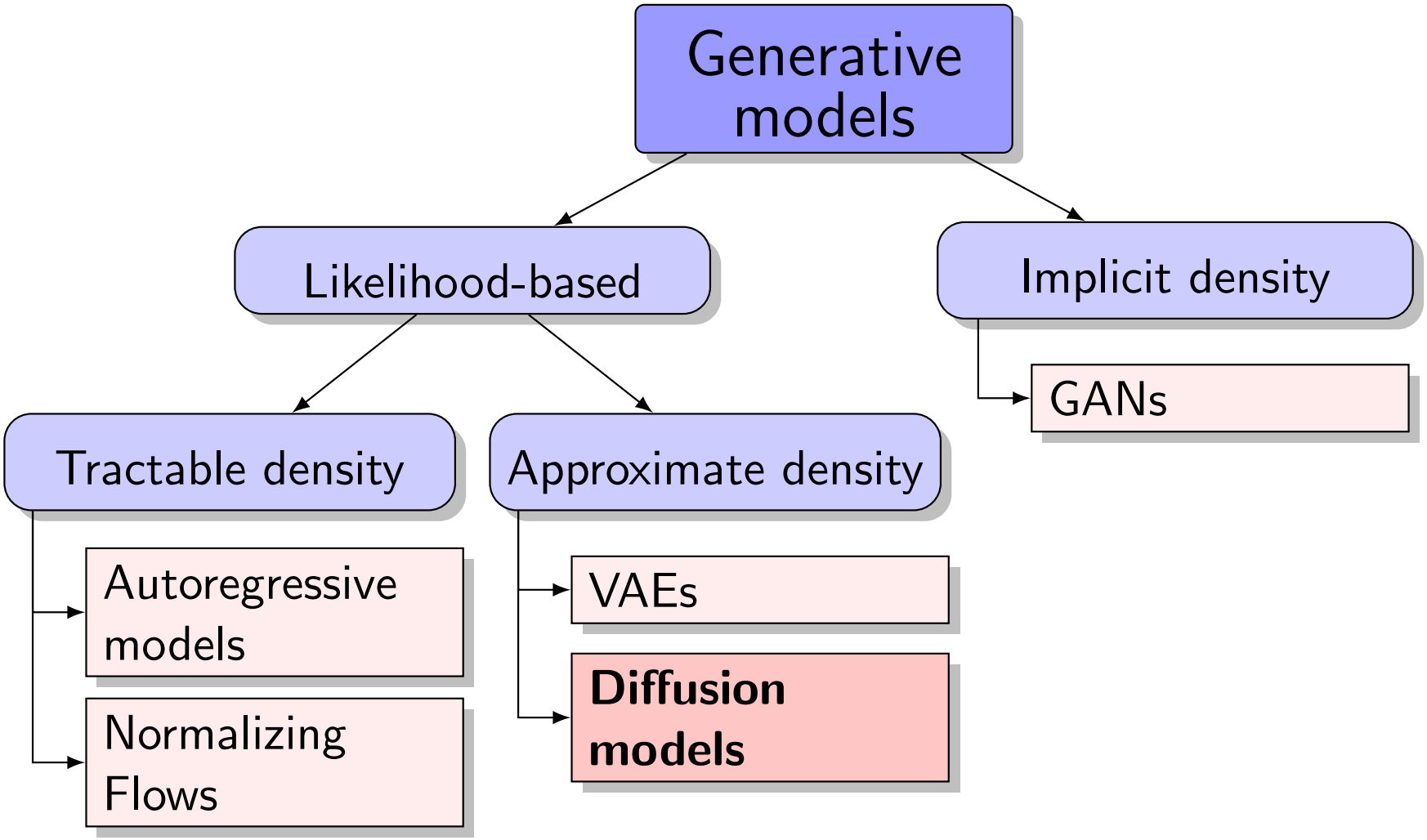
Simplified objective

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

Ho J. Denoising Diffusion Probabilistic Models, 2020

Outline

Generative models zoo



Denoising diffusion probabilistic model (DDPM)

DDPM is a VAE model

- Encoder is a fixed Gaussian Markov chain $q(\mathbf{x}_1, \ldots, \mathbf{x}_T|\mathbf{x}_0)$.
- Latent variable is a hierarchical (in each step the dim. of the latent equals to the dim of the input).
- Decoder is a simple Gaussian model $p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta})$.
- Prior distribution is given by 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. Get the sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Sample timestamp $t \sim U\{1, T\}$ and the noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.
3. Get noisy image $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}$.
4. Compute 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 mean of $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t), \tilde{\beta}_t \cdot \mathbf{I})$:

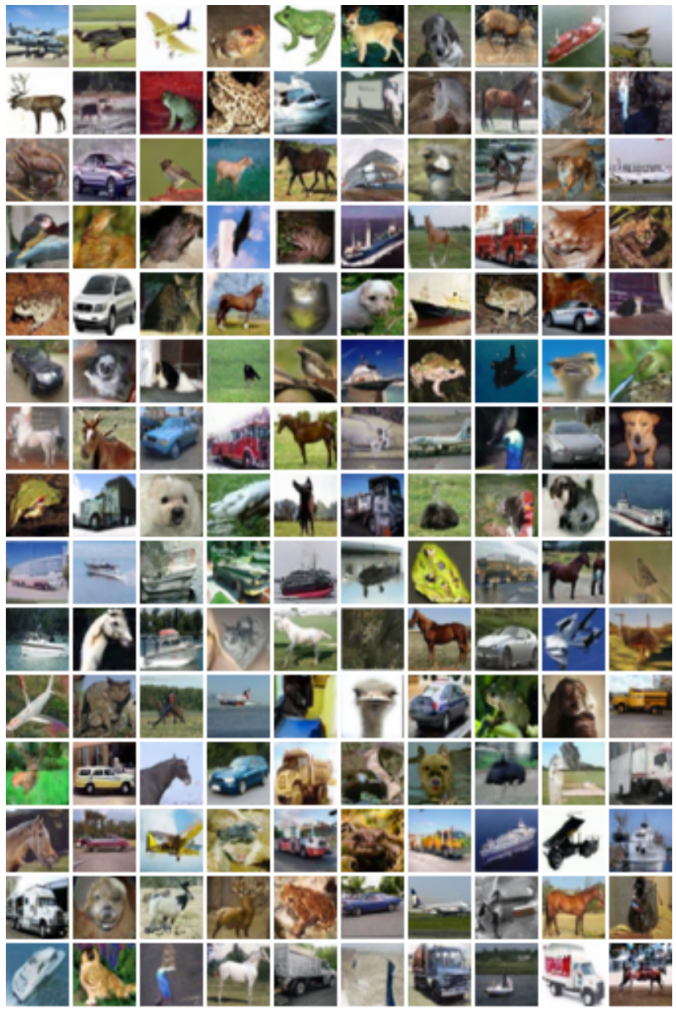
$$\boldsymbol{\mu}_{\boldsymbol{\theta},t}(\mathbf{x}_t) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \boldsymbol{\epsilon}_{\boldsymbol{\theta},t}(\mathbf{x}_t)$$

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

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 Normal assumption.
- One could treat DDPM as VAE model with hierarchical latent variables.
- ELBO of DDPM could be represented as a sum of large number of the KL terms.
- At each step DDPM predicts the noise that was used in the forward diffusion process.
- DDPM is a VAE model that tries to invert forward diffusion process using variational inference.
- DDPM is really slow, because we have to apply the model T times.