

# Deep Generative Models

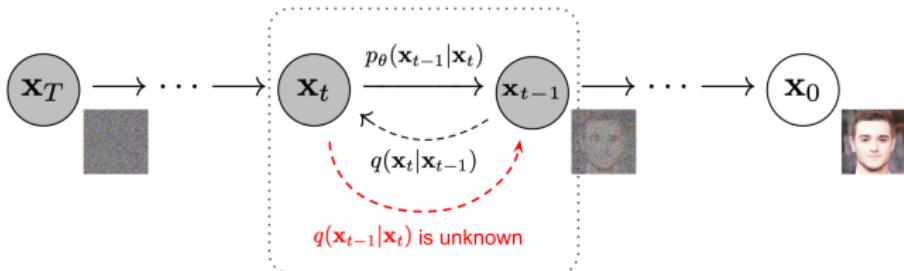
## Lecture 10

Roman Isachenko

Moscow Institute of Physics and Technology  
Yandex School of Data Analysis

2024, Autumn

# Recap of previous lecture



## Reverse process

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

Feller theorem shows that it is a reasonable assumption.

## Forward process

1.  $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x});$
2.  $\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \epsilon;$
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.  $\mathbf{x}_{t-1} = \sigma_{\boldsymbol{\theta},t}(\mathbf{x}_t) \cdot \epsilon + \mu_{\boldsymbol{\theta},t}(\mathbf{x}_t);$
3.  $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x});$

## Recap of previous lecture

**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  $\beta_t$ .

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

**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.

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

## Recap of previous lecture

- ▶  $\mathbf{z} = (\mathbf{x}_1, \dots, \mathbf{x}_T)$  is a latent variable.
- ▶ Variational posterior distribution

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

- ▶ Generative distribution and prior

$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = p(\mathbf{x}_0|\mathbf{x}_1, \boldsymbol{\theta}); \quad p(\mathbf{z}|\boldsymbol{\theta}) = \prod_{t=2}^T p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) \cdot p(\mathbf{x}_T)$$

## ELBO

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

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

## Recap of previous lecture

### ELBO of Gaussian diffusion model

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

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

Our assumption:  $\sigma_{\theta,t}^2(\mathbf{x}_t) = \tilde{\beta}_t \mathbf{I}$ .

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

# Recap of previous lecture

## 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

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

## Sampling

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

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

### Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Reparametrization of DDPM

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

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

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

- ▶ There is linear dependence between  $\epsilon$ ,  $\mathbf{x}_t$ ,  $\mathbf{x}_0$ .
- ▶ Let try to rewrite this mean in terms of  $\mathbf{x}_t$  and  $\epsilon$ .

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

# Reparametrization of DDPM

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

Reparametrization

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

$$\mu_{\theta,t}(\mathbf{x}_t) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \epsilon_{\theta,t}(\mathbf{x}_t)$$

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

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

# Reparametrization of DDPM

$$\begin{aligned}\mathcal{L}_{\phi, \theta}(\mathbf{x}) &= \mathbb{E}_{q(\mathbf{x}_1|\mathbf{x}_0)} \log p(\mathbf{x}_0|\mathbf{x}_1, \theta) - \textcolor{violet}{KL}(q(\mathbf{x}_T|\mathbf{x}_0)||p(\mathbf{x}_T)) - \\ &\quad - \sum_{t=2}^T \underbrace{\mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \textcolor{violet}{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)||p(\mathbf{x}_{t-1}|\mathbf{x}_t, \theta))}_{\mathcal{L}_t} \\ \mathcal{L}_t &= \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[ \frac{(1 - \alpha_t)^2}{2\tilde{\beta}_t \alpha_t (1 - \bar{\alpha}_t)} \left\| \epsilon - \epsilon_{\theta, t}(\sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon) \right\|^2 \right]\end{aligned}$$

Let drop the scaling coefficient.

Simplified objective

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

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

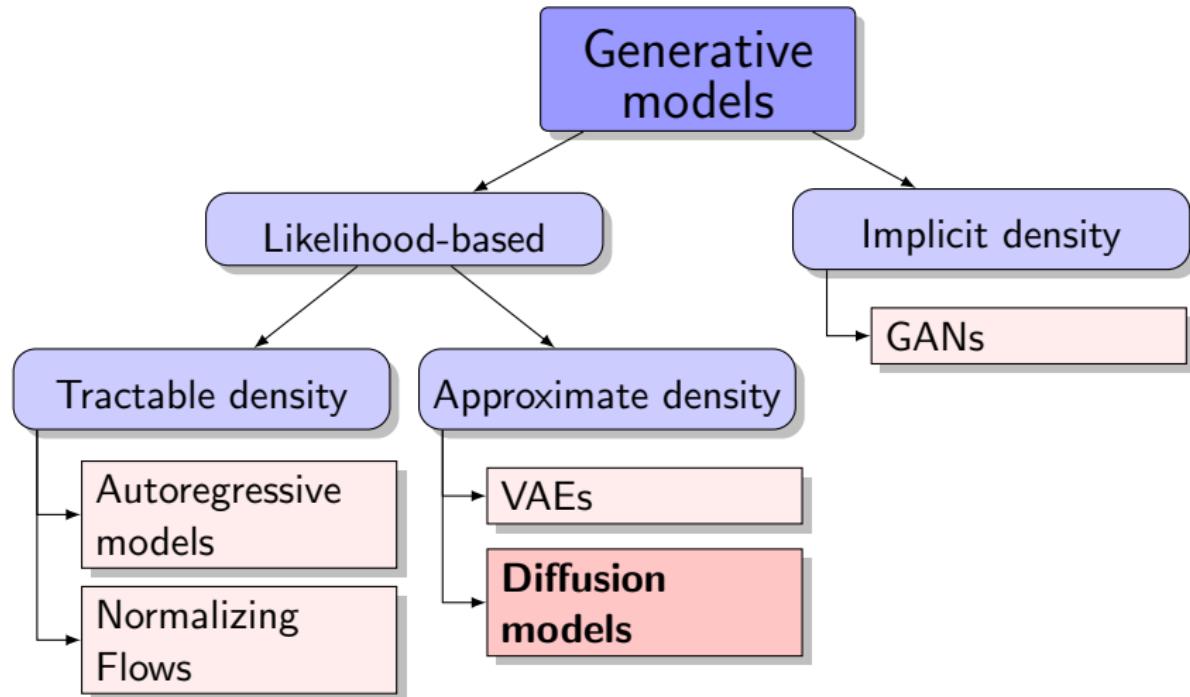
DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Generative models zoo



# Denoising diffusion probabilistic model (DDPM)

DDPM is a VAE model

- ▶ Encoder is a fixed Gaussian Markov chain  $q(\mathbf{x}_1, \dots, \mathbf{x}_T | \mathbf{x}_0)$ .
- ▶ 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  $p(\mathbf{x}_0 | \mathbf{x}_1, \theta)$ .
- ▶ Prior distribution is given by parametric Gaussian Markov chain  $p(\mathbf{x}_{t-1} | \mathbf{x}_t, \theta)$ .

Forward process

1.  $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$ ;
2.  $\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \epsilon$ ;
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.  $\mathbf{x}_{t-1} = \sigma_{\theta, t}(\mathbf{x}_t) \cdot \epsilon + \mu_{\theta, t}(\mathbf{x}_t)$ ;
3.  $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x})$ ;

# 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  $\epsilon \sim \mathcal{N}(0, \mathbf{I})$ .
3. Get noisy image  $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \epsilon$ .
4. Compute loss  $\mathcal{L}_{\text{simple}} = \|\epsilon - \epsilon_{\theta, t}(\mathbf{x}_t)\|^2$ .

## Sampling (ancestral sampling)

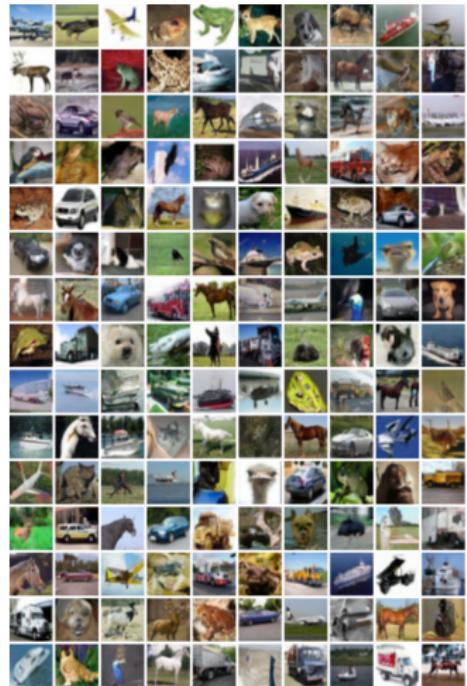
1. Sample  $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$ .
2. Compute mean of  $p(\mathbf{x}_{t-1} | \mathbf{x}_t, \theta) = \mathcal{N}(\mu_{\theta, t}(\mathbf{x}_t), \tilde{\beta}_t \cdot \mathbf{I})$ :

$$\mu_{\theta, t}(\mathbf{x}_t) = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \epsilon_{\theta, t}(\mathbf{x}_t)$$

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

# Denoising diffusion probabilistic model (DDPM)

## Samples



# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Denoising diffusion as score-based generative model

## DDPM objective

$$\begin{aligned}\mathcal{L}_t &= \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[ \frac{(1 - \alpha_t)^2}{2\tilde{\beta}_t \alpha_t (1 - \bar{\alpha}_t)} \|\epsilon_{\theta, t}(\mathbf{x}_t) - \epsilon\|_2^2 \right] \\ &= \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[ \frac{(1 - \alpha_t)^2}{2\tilde{\beta}_t \alpha_t} \left\| \frac{\epsilon_{\theta, t}(\mathbf{x}_t)}{\sqrt{1 - \bar{\alpha}_t}} - \frac{\epsilon}{\sqrt{1 - \bar{\alpha}_t}} \right\|_2^2 \right]\end{aligned}$$

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

$$\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) = -\frac{\mathbf{x}_t - \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0}{1 - \bar{\alpha}_t} = -\frac{\epsilon}{\sqrt{1 - \bar{\alpha}_t}}.$$

Let reparametrize our model:

$$\mathbf{s}_{\theta, t}(\mathbf{x}_t) = \frac{\epsilon_{\theta, t}(\mathbf{x}_t)}{\sqrt{1 - \bar{\alpha}_t}} = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \theta).$$

$$\mathcal{L}_t = \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \left[ \frac{(1 - \alpha_t)^2}{2\tilde{\beta}_t \alpha_t} \left\| \mathbf{s}_{\theta, t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \right\|_2^2 \right]$$

## DDPM vs NCSN: objectives

### DDPM objective

$$\mathbb{E}_{\pi(\mathbf{x}_0)} \mathbb{E}_{t \sim U\{1, T\}} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \left[ \frac{(1 - \alpha_t)^2}{2\tilde{\beta}_t \alpha_t} \left\| \mathbf{s}_{\theta, t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \right\|_2^2 \right]$$

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

In practice the coefficient is omitted.

### NCSN objective

$$\mathbb{E}_{\pi(\mathbf{x}_0)} \mathbb{E}_{t \sim U\{1, T\}} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x}_0)} \left\| \mathbf{s}_{\theta, \sigma_t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) \right\|_2^2$$

$$\mathbf{x}_t = \mathbf{x}_0 + \sigma_t \cdot \boldsymbol{\epsilon}$$

**ELBO maximization approach gives the same objective as denoising score-matching approach!**

## DDPM vs NCSN: sampling

### DDPM sampling (ancestral sampling)

$$\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$$

$$\begin{aligned}\mathbf{x}_{t-1} &= \mu_{\theta,t}(\mathbf{x}_t) + \sigma_t \cdot \epsilon \\&= \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \epsilon_{\theta,t}(\mathbf{x}_t) + \sigma_t \cdot \epsilon \\&= \frac{1}{\sqrt{1 - \beta_t}} \cdot \mathbf{x}_t + \frac{\beta_t}{\sqrt{1 - \beta_t}} \cdot \mathbf{s}_{\theta,t}(\mathbf{x}_t) + \sigma_t \cdot \epsilon\end{aligned}$$

### NCSN sampling (annealed Langevin dynamics)

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

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

- ▶ Update  $\mathbf{x}_{t-1}^0 = \mathbf{x}_t^L$  and choose the next  $\sigma_t$ .

# DDPM vs NCSN

## Summary

- ▶ Different Markov chains:
  - ▶ DDPM:  $\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon};$
  - ▶ NCSN:  $\mathbf{x}_t = \mathbf{x}_0 + \sigma_t \cdot \boldsymbol{\epsilon}.$
  - ▶ It is possible to consider the more general framework  
 $q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\alpha_t \cdot \mathbf{x}_0, \sigma_t^2 \cdot \mathbf{I})$
- ▶ Identical objectives: ELBO  $\equiv$  score-matching.
- ▶ Different sampling schemes:
  - ▶ ancestral sampling for DDPM;
  - ▶ annealed Langevin dynamics for NCSN;
  - ▶ there is a combined approach with alternating updates of DDPM and NCSN.

---

Kingma D. et al. *Variational Diffusion Models*, 2021

Song Y. et al. *Score-Based Generative Modeling through Stochastic Differential Equations*, 2020

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

## Guidance

- ▶ Throughout the whole course we have discussed unconditional generative models  $p(\mathbf{x}|\theta)$ .
- ▶ In practice the majority of the generative models are **conditional**:  $p(\mathbf{x}|\mathbf{y}, \theta)$ .
- ▶ Here  $\mathbf{y}$  could be the class label or **text** (for text-to-image models).



Кот ныряет в бассейн, как ребенок на обложке альбома Nevermind, реалистично



рука человека с пятью пальцами, ни четырьмя, ни шестью, а с 5 (пять) пальцами

## Taxonomy of conditional tasks

In practice the popular task is to create a conditional model  $\pi(x|y)$ .

- ▶  $y$  – class label,  $x$  – image  $\Rightarrow$  image conditional model.
- ▶  $y$  – text prompt,  $x$  – image  $\Rightarrow$  text-to-image model.
- ▶  $y$  – image,  $x$  – image  $\Rightarrow$  image-to-image model.
- ▶  $y$  – image,  $x$  – text  $\Rightarrow$  image-to-text model (image captioning).
- ▶  $y$  – sound,  $x$  – text  $\Rightarrow$  speech-to-text model (automatic speech recognition).
- ▶  $y$  – English text,  $x$  – Russian text  $\Rightarrow$  sequence-to-sequence model (machine translation).
- ▶  $y = \emptyset$ ,  $x$  – image  $\Rightarrow$  image unconditional model.

# Label guidance

**Label:** Ostrich (10th ImageNet class)



VQ-VAE (Proposed)

BigGAN deep

# Text guidance

**Prompt:** a stained glass window of a panda eating bamboo

Left:  $\gamma = 1$ , Right:  $\gamma = 3$ .



## Guidance

- ▶ If we have **supervised** data  $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$  we could treat  $\mathbf{y}$  as additional model input:
  - ▶  $p(x_j | \mathbf{x}_{1:j-1}, \mathbf{y}, \theta)$  for AR;
  - ▶ Encoder  $q(\mathbf{z} | \mathbf{x}, \mathbf{y}, \phi)$  and decoder  $p(\mathbf{x} | \mathbf{z}, \mathbf{y}, \theta)$  for VAE;
  - ▶  $G_\theta(\mathbf{z}, \mathbf{y})$  for NF and GAN;
  - ▶  $p(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{y}, \theta)$  for DDPM.
- ▶ If we have **unsupervised** data  $\{\mathbf{x}_i\}_{i=1}^n$  we need to create the way to convert unconditional model  $p(\mathbf{x} | \theta)$  to the conditional.
- ▶ It is really helpful to have the way to control the power of guidance.

## Guidance types

- ▶ **Classifier guidance:**
  - ▶ suitable for unsupervised data;
  - ▶ uses the additional classifier model (we need supervised data to train the classifier).
- ▶ **Classifier-free guidance:**
  - ▶ suitable for supervised data;
  - ▶ get rid of the additional classifier model.

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

# Classifier guidance

## DDPM sampling

1. Sample  $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$ .
2. Get denoised image (unconditional generation)

$$\begin{aligned}\mathbf{x}_{t-1} &= \frac{1}{\sqrt{1 - \beta_t}} \cdot \mathbf{x}_t + \frac{\beta_t}{\sqrt{1 - \beta_t}} \cdot \mathbf{s}_{\theta, t}(\mathbf{x}_t) + \sigma_t \cdot \epsilon \\ &= \frac{1}{\sqrt{1 - \beta_t}} \cdot \mathbf{x}_t + \frac{\beta_t}{\sqrt{1 - \beta_t}} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \theta) + \sigma_t \cdot \epsilon\end{aligned}$$

## Conditional generation

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{1 - \beta_t}} \cdot \mathbf{x}_t + \frac{\beta_t}{\sqrt{1 - \beta_t}} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{y}, \theta) + \sigma_t \cdot \epsilon$$

- ▶ Assume for simplicity  $\mathbf{y}$  is a class labels.
- ▶ Suppose that we have the distribution  $p(\mathbf{y}|\mathbf{x}_t)$  – classifier on noisy samples.

# Classifier guidance

## Conditional generation

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{1 - \beta_t}} \cdot \mathbf{x}_t + \frac{\beta_t}{\sqrt{1 - \beta_t}} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{y}, \theta) + \sigma_t \cdot \epsilon$$

## Conditional distribution

$$\begin{aligned}\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{y}, \theta) &= \nabla_{\mathbf{x}_t} \log \left( \frac{p(\mathbf{y} | \mathbf{x}_t) p(\mathbf{x}_t | \theta)}{p(\mathbf{y})} \right) \\ &= \nabla_{\mathbf{x}_t} \log p(\mathbf{y} | \mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \theta) \\ &= \nabla_{\mathbf{x}_t} \log p(\mathbf{y} | \mathbf{x}_t) - \frac{\epsilon_{\theta, t}(\mathbf{x}_t)}{\sqrt{1 - \bar{\alpha}_t}}\end{aligned}$$

Let parametrize  $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{y}, \theta) = -\frac{\epsilon_{\theta, t}(\mathbf{x}_t, \mathbf{y})}{\sqrt{1 - \bar{\alpha}_t}}$ .

## Classifier-corrected noise prediction

$$\epsilon_{\theta, t}(\mathbf{x}_t, \mathbf{y}) = \epsilon_{\theta, t}(\mathbf{x}_t) - \sqrt{1 - \bar{\alpha}_t} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y} | \mathbf{x}_t)$$

# Classifier guidance

## Classifier-corrected noise prediction

$$\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y}) = \epsilon_{\theta,t}(\mathbf{x}_t) - \sqrt{1 - \bar{\alpha}_t} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)$$

## Guidance scale

$$\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y}) = \epsilon_{\theta,t}(\mathbf{x}_t) - \gamma \cdot \sqrt{1 - \bar{\alpha}_t} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)$$

Here we introduce **guidance scale  $\gamma$**  that controls the magnitude of the classifier guidance.

## Training

- ▶ Train DDPM as usual.
- ▶ Train the additional classifier  $p(\mathbf{y}|\mathbf{x}_t)$  on the noisy samples  $\mathbf{x}_t$ .

## Guided sampling

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \cdot \mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{\alpha_t(1 - \bar{\alpha}_t)}} \cdot \epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y}) + \sigma_t \cdot \epsilon$$

# Classifier guidance

Classifier-corrected noise prediction

$$\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y}) = \epsilon_{\theta,t}(\mathbf{x}_t) - \gamma \cdot \sqrt{1 - \bar{\alpha}_t} \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)$$

Guidance-scaled conditional distribution

$$\frac{\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y})}{\sqrt{1 - \bar{\alpha}_t}} = \frac{\epsilon_{\theta,t}(\mathbf{x}_t)}{\sqrt{1 - \bar{\alpha}_t}} - \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)$$

$$\begin{aligned}\nabla_{\mathbf{x}_t}^{\gamma} \log p(\mathbf{x}_t|\mathbf{y}, \theta) &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t) \\ &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)^{\gamma} \\ &= \nabla_{\mathbf{x}_t} \log \left( \frac{p(\mathbf{y}|\mathbf{x}_t)^{\gamma} p(\mathbf{x}_t|\theta)}{Z} \right)\end{aligned}$$

**Note:** Guidance scale  $\gamma$  tries to sharpen the distribution  $p(\mathbf{y}|\mathbf{x}_t)$  (in this case  $Z$  should not depend on  $\mathbf{x}_t$ ).

# Outline

## 1. Denoising Diffusion Probabilistic Model (DDPM)

Reparametrization

Overview

DDPM as score-based generative model

## 2. Guidance

Classifier guidance

Classifier-free guidance

## Classifier-free guidance

- ▶ Previous method requires training the additional classifier model  $p(\mathbf{y}|\mathbf{x}_t)$  on the noisy data.
- ▶ Let try to avoid this requirement.

$$\nabla_{\mathbf{x}_t}^{\gamma} \log p(\mathbf{x}_t|\mathbf{y}, \theta) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)$$

$$\begin{aligned}\nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t) &= \nabla_{\mathbf{x}_t} \log \left( \frac{p(\mathbf{x}_t|\mathbf{y}, \theta)p(\mathbf{y})}{p(\mathbf{x}_t|\theta)} \right) \\ &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}, \theta) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta)\end{aligned}$$

$$\begin{aligned}\nabla_{\mathbf{x}_t}^{\gamma} \log p(\mathbf{x}_t|\mathbf{y}, \theta) &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t) = \\ &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \gamma \cdot (\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}, \theta) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta)) = \\ &= (1 - \gamma) \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) + \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}, \theta)\end{aligned}$$

**Note:** In the case of  $\gamma = 1$  we will get the identity statement.

## Classifier-free guidance

$$\nabla_{\mathbf{x}_t}^{\gamma} \log p(\mathbf{x}_t | \mathbf{y}, \theta) = (1 - \gamma) \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \theta) + \gamma \cdot \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t | \mathbf{y}, \theta)$$

$$\frac{\hat{\epsilon}_{\theta,t}(\mathbf{x}_t, \mathbf{y})}{\sqrt{1 - \bar{\alpha}_t}} = (1 - \gamma) \cdot \frac{\epsilon_{\theta,t}(\mathbf{x}_t)}{\sqrt{1 - \bar{\alpha}_t}} + \gamma \cdot \frac{\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y})}{\sqrt{1 - \bar{\alpha}_t}}$$

### Classifier-free-corrected noise prediction

$$\hat{\epsilon}_{\theta,t}(\mathbf{x}_t, \mathbf{y}) = \gamma \cdot \epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y}) + (1 - \gamma) \cdot \epsilon_{\theta,t}(\mathbf{x}_t)$$

- ▶ Train the single model  $\epsilon_{\theta,t}(\mathbf{x}_t, \mathbf{y})$  on **supervised** data alternating with real conditioning  $\mathbf{y}$  and empty conditioning  $\mathbf{y} = \emptyset$ .
- ▶ Apply the model twice during inference.

## Summary

- ▶ 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.
- ▶ DDPM and NCSN are closely related in terms of objectives.
- ▶ Classifier guidance is the way to turn the unconditional model to the conditional one via the training additional classifier on the noisy data.
- ▶ Classifier-free guidance allows to avoid the training additional classifier to get the conditional model. It is widely used in practice.