

# Deep Generative Models

## Lecture 10

Roman Isachenko

Moscow Institute of Physics and Technology

2024, Autumn

## Recap of previous lecture

How to evaluate likelihood-free models?

$p(y|x)$  – pretrained image classification model (e.g. ImageNet classifier).

What do we want from samples?

- ▶ Sharpness



$p(y|x)$  has low entropy (each image  $x$  should have distinctly recognizable object).

- ▶ Diversity



$p(y) = \int p(y|x)p(x)dx$  has high entropy (there should be as many classes generated as possible).

## Recap of previous lecture

### Frechet Inception Distance (FID)

In case of Normal distributions  $\pi(\mathbf{x}) = \mathcal{N}(\boldsymbol{\mu}_\pi, \boldsymbol{\Sigma}_\pi)$ ,  
 $p(\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}_p, \boldsymbol{\Sigma}_p)$ :

$$\begin{aligned}\text{FID}(\pi, p) &= W_2^2(\pi, p) = \inf_{\gamma \in \Gamma(\pi, p)} \mathbb{E}_{(\mathbf{x}, \mathbf{y}) \sim \gamma} \|\mathbf{x} - \mathbf{y}\|^2 \\ &= \|\boldsymbol{\mu}_\pi - \boldsymbol{\mu}_p\|_2^2 + \text{tr} \left[ \boldsymbol{\Sigma}_\pi + \boldsymbol{\Sigma}_p - 2 \left( \boldsymbol{\Sigma}_\pi^{1/2} \boldsymbol{\Sigma}_p \boldsymbol{\Sigma}_\pi^{1/2} \right)^{1/2} \right]\end{aligned}$$

### Maximum Mean Discrepancy (MMD)

$\pi(\mathbf{x}) = p(\mathbf{y})$  if and only if  $\mathbb{E}_{\pi(\mathbf{x})} f(\mathbf{x}) = \mathbb{E}_{p(\mathbf{y})} f(\mathbf{y})$  for any bounded and continuous  $f$

$$\text{MMD}^2(\pi, p) = \mathbb{E}_{\substack{\mathbf{x} \sim \pi(\mathbf{x}) \\ \mathbf{x}' \sim \pi(\mathbf{x})}} k(\mathbf{x}, \mathbf{x}') + \mathbb{E}_{\substack{\mathbf{y} \sim p(\mathbf{y}) \\ \mathbf{y}' \sim p(\mathbf{y})}} k(\mathbf{y}, \mathbf{y}') - 2 \mathbb{E}_{\substack{\mathbf{x} \sim \pi(\mathbf{x}) \\ \mathbf{y} \sim p(\mathbf{y})}} k(\mathbf{x}, \mathbf{y})$$

---

Heusel M. et al. *GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium*, 2017

Jayasumana S. et al. *Rethinking FID: Towards a Better Evaluation Metric for Image Generation*, 2024

## Recap of previous lecture

- ▶  $\mathcal{S}_\pi = \{\mathbf{x}_i\}_{i=1}^n \sim \pi(\mathbf{x})$  – real samples;
- ▶  $\mathcal{S}_p = \{\mathbf{x}_i\}_{i=1}^n \sim p(\mathbf{x}|\boldsymbol{\theta})$  – generated samples.

Embed samples using pretrained classifier network (as previously):

$$\mathcal{G}_\pi = \{\mathbf{g}_i\}_{i=1}^n, \quad \mathcal{G}_p = \{\mathbf{g}_i\}_{i=1}^n.$$

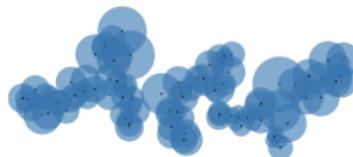
Define binary function:

$$f(\mathbf{g}, \mathcal{G}) = \begin{cases} 1, & \text{if exists } \mathbf{g}' \in \mathcal{G} : \|\mathbf{g} - \mathbf{g}'\|_2 \leq \|\mathbf{g}' - \text{NN}_k(\mathbf{g}', \mathcal{G})\|_2; \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{Precision}(\mathcal{G}_\pi, \mathcal{G}_p) = \frac{1}{n} \sum_{\mathbf{g} \in \mathcal{G}_p} f(\mathbf{g}, \mathcal{G}_\pi); \quad \text{Recall}(\mathcal{G}_\pi, \mathcal{G}_p) = \frac{1}{n} \sum_{\mathbf{g} \in \mathcal{G}_\pi} f(\mathbf{g}, \mathcal{G}_p).$$



(a) True manifold



(b) Approx. manifold

# Recap of previous lecture

## Langevin dynamic

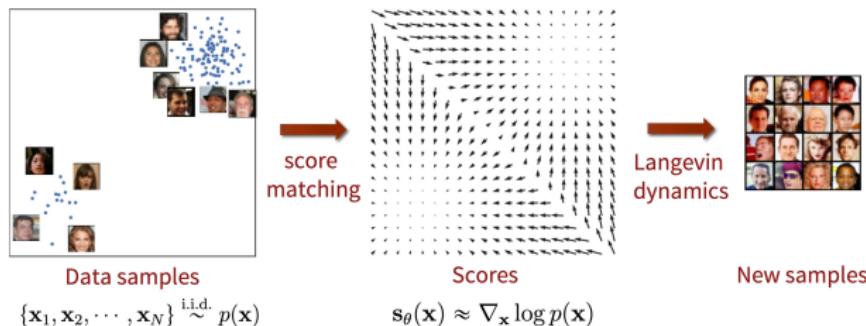
$$\mathbf{x}_{I+1} = \mathbf{x}_I + \frac{\eta}{2} \cdot \nabla_{\mathbf{x}_I} \log p(\mathbf{x}_I | \boldsymbol{\theta}) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

## Fisher divergence

$$D_F(\pi, p) = \frac{1}{2} \mathbb{E}_\pi \| \nabla_{\mathbf{x}} \log p(\mathbf{x} | \boldsymbol{\theta}) - \nabla_{\mathbf{x}} \log \pi(\mathbf{x}) \|_2^2 \rightarrow \min_{\boldsymbol{\theta}}$$

## Score function

$$\mathbf{s}_\theta(\mathbf{x}) = \nabla_{\mathbf{x}} \log p(\mathbf{x} | \boldsymbol{\theta})$$



# Outline

1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

# Outline

1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

## Denoising score matching

Let perturb original data  $\mathbf{x} \sim \pi(\mathbf{x})$  by random normal noise

$$\mathbf{x}_\sigma = \mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}), \quad q(\mathbf{x}_\sigma | \mathbf{x}) = \mathcal{N}(\mathbf{x}, \sigma^2 \cdot \mathbf{I})$$

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

### Assumption

The solution of

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

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

- ▶  $\mathbf{s}_{\theta, \sigma}(\mathbf{x}_\sigma)$  tries to **denoise** a corrupted sample  $\mathbf{x}_\sigma$ .
- ▶ Score function  $\mathbf{s}_{\theta, \sigma}(\mathbf{x}_\sigma)$  parametrized by  $\sigma$ .
- ▶ **Problem:** We don't know  $q(\mathbf{x}_\sigma)$ , just like  $\pi(\mathbf{x})$ .

# Denoising score matching

## Theorem

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

## Proof

$$\begin{aligned} \mathbb{E}_{q(\mathbf{x}_\sigma)} h(\mathbf{x}_\sigma) &= \int q(\mathbf{x}_\sigma) h(\mathbf{x}_\sigma) d\mathbf{x}_\sigma = \\ &= \int \left( \int q(\mathbf{x}_\sigma|\mathbf{x}) \pi(\mathbf{x}) d\mathbf{x} \right) h(\mathbf{x}_\sigma) d\mathbf{x}_\sigma = \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} h(\mathbf{x}_\sigma) d\mathbf{x}_\sigma \end{aligned}$$

$$\begin{aligned} \mathbb{E}_{q(\mathbf{x}_\sigma)} \left\| \mathbf{s}_{\theta,\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2 &= \\ &= \mathbb{E}_{q(\mathbf{x}_\sigma)} \left[ \left\| \mathbf{s}_{\theta,\sigma}(\mathbf{x}_\sigma) \right\|^2 + \underbrace{\left\| \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right\|_2^2}_{\text{const}(\theta)} - 2 \mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma) \right] \end{aligned}$$

# Denoising score matching

## Theorem

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

## Proof (continued)

$$\begin{aligned}\mathbb{E}_{q(\mathbf{x}_\sigma)} [\mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)] &= \int q(\mathbf{x}_\sigma) \left[ \mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \frac{\nabla_{\mathbf{x}_\sigma} q(\mathbf{x}_\sigma)}{q(\mathbf{x}_\sigma)} \right] d\mathbf{x}_\sigma = \\ &= \int \left[ \mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \left( \int q(\mathbf{x}_\sigma|\mathbf{x}) \pi(\mathbf{x}) d\mathbf{x} \right) \right] d\mathbf{x}_\sigma = \\ &= \int \int \pi(\mathbf{x}) [\mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} q(\mathbf{x}_\sigma|\mathbf{x})] d\mathbf{x}_\sigma d\mathbf{x} = \\ &= \int \int \pi(\mathbf{x}) q(\mathbf{x}_\sigma|\mathbf{x}) [\mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x})] d\mathbf{x}_\sigma d\mathbf{x} = \\ &= \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} [\mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x})]\end{aligned}$$

# Denoising score matching

## Theorem

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

## Proof (continued)

$$\begin{aligned}\mathbb{E}_{q(\mathbf{x}_\sigma)} \|\mathbf{s}_{\theta,\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)\|_2^2 &= \\ &= \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \left[ \|\mathbf{s}_{\theta,\sigma}(\mathbf{x}_\sigma)\|^2 - 2\mathbf{s}_{\theta,\sigma}^T(\mathbf{x}_\sigma) \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x}) \right] + \text{const}(\theta) \\ &= \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_\sigma|\mathbf{x})} \|\mathbf{s}_{\theta,\sigma}(\mathbf{x}_\sigma) - \nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma|\mathbf{x})\|_2^2 + \text{const}(\theta)\end{aligned}$$

## Gradient of the noise kernel

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

The RHS does not need to compute  $\nabla_{\mathbf{x}_\sigma} \log q(\mathbf{x}_\sigma)$  and even  $\nabla_{\mathbf{x}_\sigma} \log \pi(\mathbf{x}_\sigma)$ .

# Denoising score matching

Initial objective:

$$\mathbb{E}_\pi \|\mathbf{s}_\theta(\mathbf{x}) - \nabla_{\mathbf{x}} \log \pi(\mathbf{x})\|_2^2 \rightarrow \min_\theta$$

Noised objective:

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

This is equivalent to denoising task

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

$$\mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{\mathcal{N}(0, \mathbf{I})} \left\| \mathbf{s}_{\theta,\sigma}(\mathbf{x} + \sigma \cdot \boldsymbol{\epsilon}) + \frac{\boldsymbol{\epsilon}}{\sigma} \right\|_2^2 \rightarrow \min_\theta$$

## Langevin dynamic

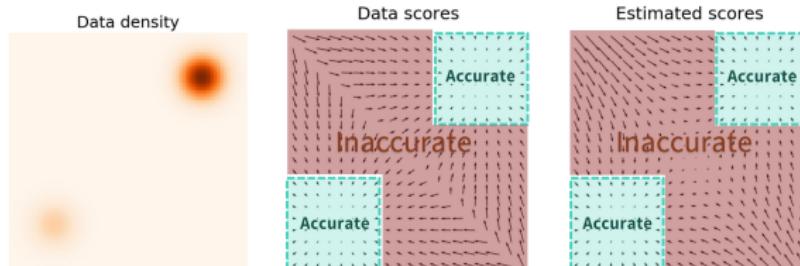
$$\mathbf{x}_{I+1} = \mathbf{x}_I + \frac{\eta}{2} \cdot \mathbf{s}_{\theta,\sigma}(\mathbf{x}_I) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

# Outline

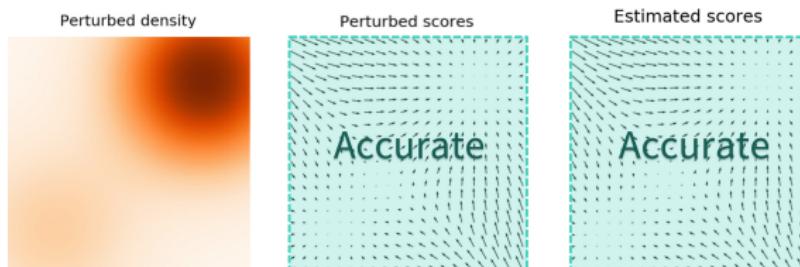
1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

# Denoising score matching

- ▶ If  $\sigma$  is **small**, the score function is not accurate and Langevin dynamics will probably fail to jump between modes.



- ▶ If  $\sigma$  is **large**, it is good for low-density regions and multimodal distributions, but we will learn too corrupted distribution.

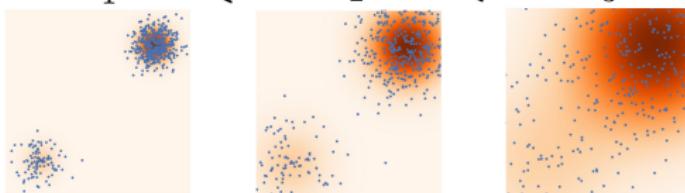


# Noise Conditioned Score Network (NCSN)

- ▶ Define the sequence of the noise levels:  $\sigma_1 < \sigma_2 < \dots < \sigma_T$ .
- ▶ Perturb the original data with the different noise levels to obtain  $\mathbf{x}_t = \mathbf{x} + \sigma_t \cdot \epsilon$ ,  $\mathbf{x}_t \sim q(\mathbf{x}_t)$ .
- ▶ Choose  $\sigma_1, \sigma_T$  such that:

$$q(\mathbf{x}_1) \approx \pi(\mathbf{x}), \quad q(\mathbf{x}_T) \approx \mathcal{N}(0, \sigma_T^2 \cdot \mathbf{I}).$$

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



# Noise Conditioned Score Network (NCSN)

Train the denoising score function  $s_{\theta, \sigma_t}(\mathbf{x}_t)$  for each noise level using unified weighted objective:

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

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

## 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  $\mathcal{L} = \sigma_t^2 \cdot \|s_{\theta, \sigma_t}(\mathbf{x}_t) + \frac{\boldsymbol{\epsilon}}{\sigma_t}\|^2$ .

How to sample from this model?

# Noise Conditioned Score Network (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

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

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



# Outline

1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

# Outline

1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

## Forward gaussian diffusion process

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

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

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

### Statement 1

Let denote  $\alpha_t = 1 - \beta_t$  and  $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s = \prod_{s=1}^t (1 - \beta_s)$ . Then

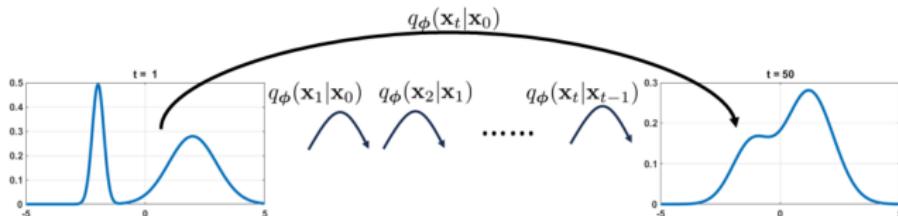
$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0, (1 - \bar{\alpha}_t) \cdot \mathbf{I})$$

We are able to sample from any timestamp using only  $\mathbf{x}_0$ !

$$\begin{aligned}\mathbf{x}_t &= \sqrt{\alpha_t} \cdot \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} \cdot \boldsymbol{\epsilon}_t = \\ &= \sqrt{\alpha_t} (\sqrt{\alpha_{t-1}} \cdot \mathbf{x}_{t-2} + \sqrt{1 - \alpha_{t-1}} \cdot \boldsymbol{\epsilon}_{t-1}) + \sqrt{1 - \alpha_t} \cdot \boldsymbol{\epsilon}_t = \\ &= \sqrt{\alpha_t \alpha_{t-1}} \cdot \mathbf{x}_{t-2} + (\sqrt{\alpha_t (1 - \alpha_{t-1})} \cdot \boldsymbol{\epsilon}_{t-1} + \sqrt{1 - \alpha_t} \cdot \boldsymbol{\epsilon}_t) = \\ &= \sqrt{\alpha_t \alpha_{t-1}} \cdot \mathbf{x}_{t-2} + \sqrt{1 - \alpha_{t-1} \alpha_t} \cdot \boldsymbol{\epsilon}'_t = \\ &= \dots = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).\end{aligned}$$

# Forward gaussian diffusion process

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



## Statement 2

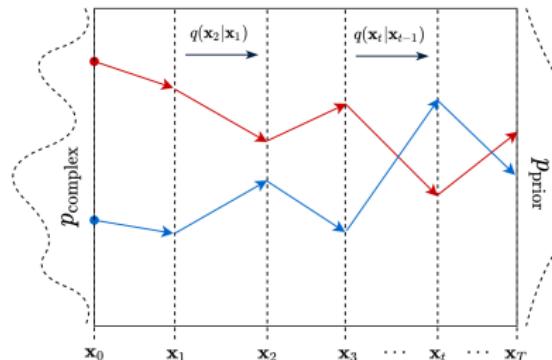
Applying the Markov chain to samples from any  $\pi(\mathbf{x})$  we will get  $\mathbf{x}_\infty \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, \mathbf{I})$ . Here  $p_\infty(\mathbf{x})$  is a **stationary** and **limiting** distribution:

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

$$p_\infty(\mathbf{x}) = \int q(\mathbf{x}_\infty | \mathbf{x}_0) \pi(\mathbf{x}_0) d\mathbf{x}_0 \approx \mathcal{N}(0, \mathbf{I}) \int \pi(\mathbf{x}_0) d\mathbf{x}_0 = \mathcal{N}(0, \mathbf{I})$$

# 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.  $\mathbf{x}_t = \sqrt{1 - \beta_t} \cdot \mathbf{x}_{t-1} + \sqrt{\beta_t} \cdot \boldsymbol{\epsilon}$ , 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.

# Outline

1. Denoising score matching
2. Noise Conditioned Score Network (NCSN)
3. Gaussian diffusion process
  - Forward gaussian diffusion process
  - Denoising score matching

## Denoising score matching

### NCSN

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0, \sigma_t^2 \cdot \mathbf{I}), \quad q(\mathbf{x}_1) \approx \pi(\mathbf{x}), \quad q(\mathbf{x}_T) \approx \mathcal{N}(0, \sigma_T^2 \cdot \mathbf{I}).$$

$$\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}) = -\frac{\mathbf{x}_t - \mathbf{x}}{\sigma_t^2}$$

### Gaussian diffusion

$$q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0, (1 - \bar{\alpha}_t) \cdot \mathbf{I}), \quad q(\mathbf{x}_1) \approx \pi(\mathbf{x}), \quad q(\mathbf{x}_T) \approx \mathcal{N}(0, \mathbf{I})$$

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

### Theorem (denoising score matching)

$$\begin{aligned} \mathbb{E}_{q(\mathbf{x}_t)} \|\mathbf{s}_{\theta,t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t)\|_2^2 &= \\ &= \mathbb{E}_{\pi(\mathbf{x})} \mathbb{E}_{q(\mathbf{x}_t | \mathbf{x})} \|\mathbf{s}_{\theta,t}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x})\|_2^2 + \text{const}(\theta) \end{aligned}$$

**Note:** We are able to apply NCSN approach with annealed Langevin dynamics to get diffusion denoising model.

## Summary

- ▶ Denoising score matching minimizes Fisher divergence on noisy samples. It allows to estimate Fisher divergence using samples.
- ▶ Noise conditioned score network uses multiple noise levels and annealed Langevin dynamics to fit score function and sample from the model.
- ▶ Gaussian diffusion process is a Markov chain that injects special form of Gaussian noise to the samples.
- ▶ Denoising score matching is applicable to gaussian diffusion process.