

# Deep Generative Models

## Lecture 14

Roman Isachenko

Moscow Institute of Physics and Technology

2022 – 2023

## Recap of previous lecture

$$\mathbf{a}_z(t) = \frac{\partial L(\mathbf{y})}{\partial \mathbf{z}(t)}; \quad \mathbf{a}_{\theta}(t) = \frac{\partial L(\mathbf{y})}{\partial \theta(t)} - \text{adjoint functions.}$$

### Theorem (Pontryagin)

$$\frac{d\mathbf{a}_z(t)}{dt} = -\mathbf{a}_z(t)^T \cdot \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \mathbf{z}}; \quad \frac{d\mathbf{a}_{\theta}(t)}{dt} = -\mathbf{a}_z(t)^T \cdot \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \theta}.$$

### Forward pass

$$\mathbf{z}(t_1) = \int_{t_0}^{t_1} f(\mathbf{z}(t), t, \theta) dt + \mathbf{z}_0 \quad \Rightarrow \quad \text{ODE Solver}$$

### Backward pass

$$\left. \begin{aligned} \frac{\partial L}{\partial \theta(t_0)} &= \mathbf{a}_{\theta}(t_0) = - \int_{t_1}^{t_0} \mathbf{a}_z(t)^T \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \theta(t)} dt + 0 \\ \frac{\partial L}{\partial \mathbf{z}(t_0)} &= \mathbf{a}_z(t_0) = - \int_{t_1}^{t_0} \mathbf{a}_z(t)^T \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \mathbf{z}(t)} dt + \frac{\partial L}{\partial \mathbf{z}(t_1)} \\ \mathbf{z}(t_0) &= - \int_{t_1}^{t_0} f(\mathbf{z}(t), t, \theta) dt + \mathbf{z}_1. \end{aligned} \right\} \Rightarrow \text{ODE Solver}$$

## Recap of previous lecture

### Continuous-in-time normalizing flows

$$\frac{d\mathbf{z}(t)}{dt} = f(\mathbf{z}(t), t, \theta); \quad \frac{d \log p(\mathbf{z}(t), t)}{dt} = -\text{tr} \left( \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \mathbf{z}(t)} \right).$$

### Theorem (Picard)

If  $f$  is uniformly Lipschitz continuous in  $\mathbf{z}$  and continuous in  $t$ , then the ODE has a **unique** solution.

### Forward transform + log-density

$$\begin{bmatrix} \mathbf{x} \\ \log p(\mathbf{x}|\theta) \end{bmatrix} = \begin{bmatrix} \mathbf{z} \\ \log p(\mathbf{z}) \end{bmatrix} + \int_{t_0}^{t_1} \begin{bmatrix} f(\mathbf{z}(t), t, \theta) \\ -\text{tr} \left( \frac{\partial f(\mathbf{z}(t), t, \theta)}{\partial \mathbf{z}(t)} \right) \end{bmatrix} dt.$$

### Hutchinson's trace estimator

$$\log p(\mathbf{z}(t_1)) = \log p(\mathbf{z}(t_0)) - \mathbb{E}_{p(\epsilon)} \int_{t_0}^{t_1} \left[ \epsilon^T \frac{\partial f}{\partial \mathbf{z}} \epsilon \right] dt.$$

## Recap of previous lecture

### SDE basics

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

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

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

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

### Langevin dynamics

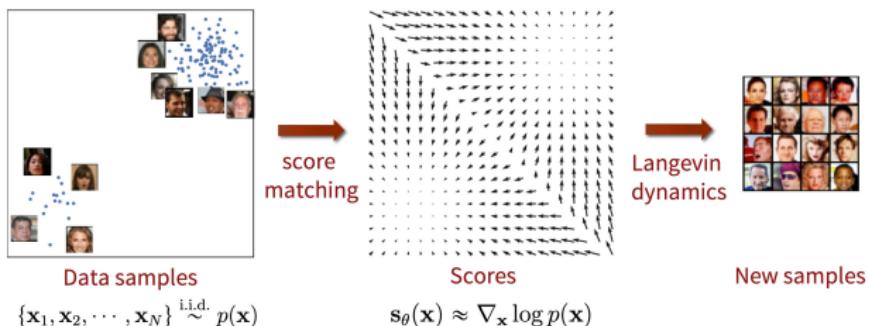
Let  $\mathbf{x}_0$  be a random vector. Then under mild regularity conditions for small enough  $\eta$  samples from the following dynamics

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

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

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

# Recap of previous lecture



## Theorem (implicit score matching)

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

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

## Recap of previous lecture

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

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

Then the solution of

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

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

## Theorem (denoising score matching)

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

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

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

# Outline

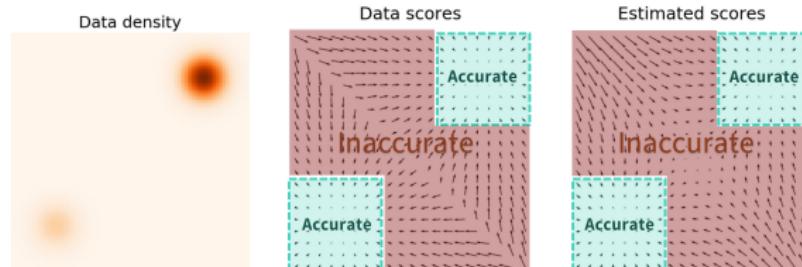
1. Noise conditioned score network
2. Diffusion models
  - Gaussian diffusion process
  - Denoising diffusion probabilistic model (DDPM)

# Outline

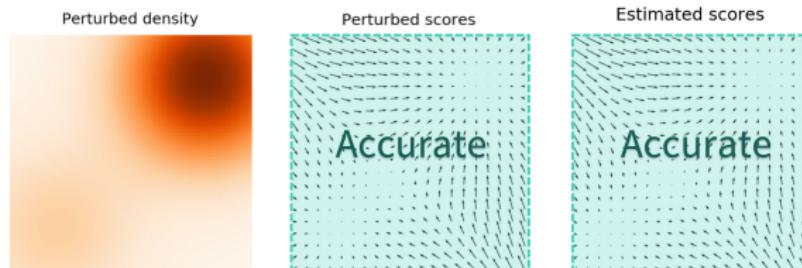
1. Noise conditioned score network
2. Diffusion models
  - Gaussian diffusion process
  - Denoising diffusion probabilistic model (DDPM)

# 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.

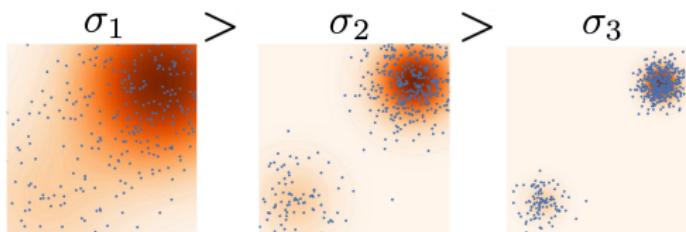


## Noise conditioned score network

- ▶ Define the sequence of noise levels:  $\sigma_1 > \sigma_2 > \dots > \sigma_L$ .
- ▶ Perturb the original data with the different noise level to get  $\pi(\mathbf{x}|\sigma_1), \dots, \pi(\mathbf{x}|\sigma_L)$ .
- ▶ Train denoised score function  $\mathbf{s}(\mathbf{x}, \theta, \sigma)$  for each noise level:

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

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



# Noise conditioned score network

Training: loss function

$$\sum_{I=1}^L \sigma_I^2 \mathbb{E}_{\pi(\mathbf{x}')} \mathbb{E}_\epsilon \left\| \mathbf{s}_I + \frac{\epsilon}{\sigma_I} \right\|_2^2,$$

Here

- ▶  $\mathbf{s}_I = \mathbf{s}(\mathbf{x}' + \sigma_I \cdot \epsilon, \theta, \sigma_I)$ .
- ▶  $\nabla_{\mathbf{x}} \log p(\mathbf{x} | \mathbf{x}', \sigma) = -\frac{\mathbf{x} - \mathbf{x}'}{\sigma^2} = -\frac{\epsilon}{\sigma_I}$ .

Samples



Inference: annealed Langevin dynamic

---

**Algorithm 1** Annealed Langevin dynamics.

---

**Require:**  $\{\sigma_i\}_{i=1}^L, \epsilon, T$ .

```
1: Initialize  $\tilde{\mathbf{x}}_0$ 
2: for  $i \leftarrow 1$  to  $L$  do
3:    $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$        $\triangleright \alpha_i$  is the step size.
4:   for  $t \leftarrow 1$  to  $T$  do
5:     Draw  $\mathbf{z}_t \sim \mathcal{N}(0, I)$ 
6:      $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2} \mathbf{s}_\theta(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i} \mathbf{z}_t$ 
7:   end for
8:    $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$ 
9: end for
return  $\tilde{\mathbf{x}}_T$ 
```

---

# Outline

1. Noise conditioned score network
2. Diffusion models
  - Gaussian diffusion process
  - Denoising diffusion probabilistic model (DDPM)

# Outline

1. Noise conditioned score network
2. Diffusion models
  - Gaussian diffusion process
  - Denoising diffusion probabilistic model (DDPM)

## Forward gaussian diffusion process

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

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

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

### Statement 1

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

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

### Statement 2

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

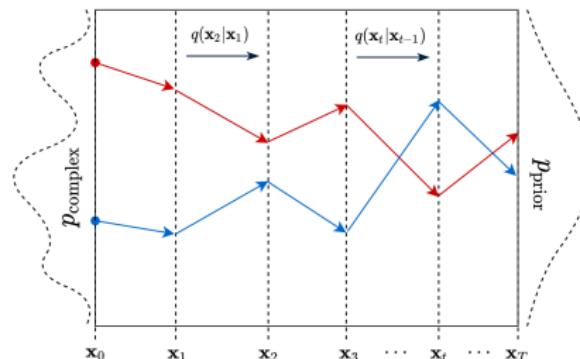
$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \cdot \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \cdot \boldsymbol{\epsilon}, \quad \text{where } \boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$$

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

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

# Forward gaussian diffusion process

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

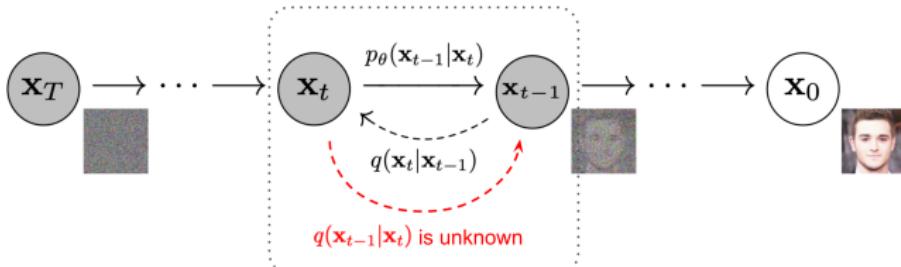


1.  $\mathbf{x}_0 = \mathbf{x} \sim \pi(\mathbf{x});$
2.  $\mathbf{x}_t = \sqrt{1 - \beta} \cdot \mathbf{x}_{t-1} + \sqrt{\beta} \cdot \boldsymbol{\epsilon}$ , where  $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$ ,  $t \geq 1$ ;
3.  $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1)$ , where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

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

Now our goal is to revert this process.

# Reverse gaussian diffusion process



Let define the reverse process

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

## Forward process

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

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

3.  $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1).$

## Reverse process

1.  $\mathbf{x}_T \sim p_\infty(\mathbf{x}) = \mathcal{N}(0, 1);$

2.  $\mathbf{x}_{t-1} = \sigma(\mathbf{x}_t, \theta, t) \cdot \boldsymbol{\epsilon} + \mu(\mathbf{x}_t, \theta, t);$

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

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

# Outline

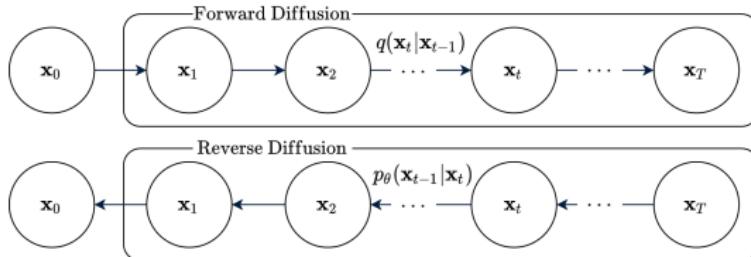
1. Noise conditioned score network

2. Diffusion models

Gaussian diffusion process

Denoising diffusion probabilistic model (DDPM)

# Gaussian diffusion model as VAE



- ▶ Let treat  $\mathbf{z} = (x_1, \dots, x_T)$  as a latent variable (**note**: each  $x_t$  has the same size).
- ▶ Variational posterior distribution (**note**: there is no learnable parameters)

$$q(\mathbf{z}|\mathbf{x}) = q(x_1, \dots, x_T | x_0) = \prod_{t=1}^T q(x_t | x_{t-1}).$$

- ▶ Probabilistic model

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

- ▶ Generative distribution and prior

$$p(\mathbf{x} | \mathbf{z}, \boldsymbol{\theta}) = p(x_0 | x_1, \boldsymbol{\theta}); \quad p(\mathbf{z} | \boldsymbol{\theta}) = \prod_{t=2}^T p(x_{t-1} | x_t, \boldsymbol{\theta})$$

# Gaussian diffusion model as VAE

## ELBO

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

## Statement

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

- ▶ **First term** is a decoder distribution (could be AR model or discretized distribution (like mixture of logistics)).
- ▶ **Third term** is constant (KL between two standard normals).

## Gaussian diffusion model as VAE

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

Here

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

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

- ▶ **Note:** We do not have an analytical expression for  $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ .
- ▶ Assume  $\sigma^2(\mathbf{x}_t, \boldsymbol{\theta}, t) = \tilde{\beta}_t \mathbf{I}$  (reminder:  
 $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}_{t-1} | \mu(\mathbf{x}_t, \boldsymbol{\theta}, t), \sigma^2(\mathbf{x}_t, \boldsymbol{\theta}, t))$ ).
- ▶ Use KL formula between two normal distributions:

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

## Gaussian diffusion model as VAE

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

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

### Reparametrization

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

### KL term

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

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

## Gaussian diffusion model vs Score matching

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

- ▶ Result from Statement 2

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

- ▶ Score of noised distribution

$$\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t | \mathbf{x}_0) = -\frac{\epsilon}{\sqrt{1-\bar{\alpha}_t}}, \quad \text{where } \epsilon \sim \mathcal{N}(0, 1).$$

- ▶ Let reparametrize our model:

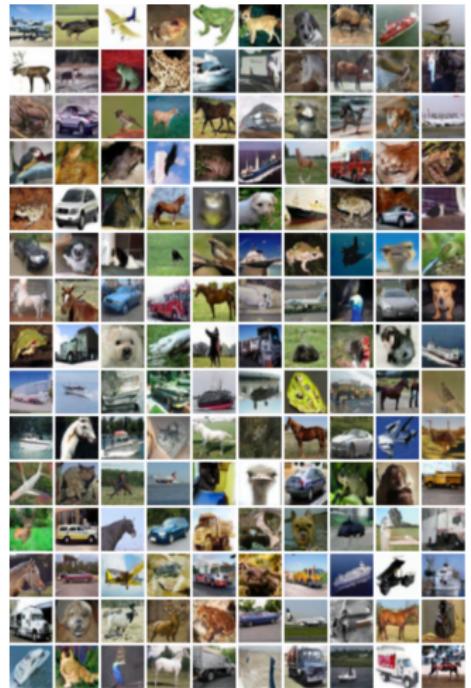
$$\mathbf{s}(\mathbf{x}_t, \theta, t) = \frac{\epsilon(\mathbf{x}_t, \theta, t)}{\sqrt{1-\bar{\alpha}_t}}.$$

## Noise conditioned score network

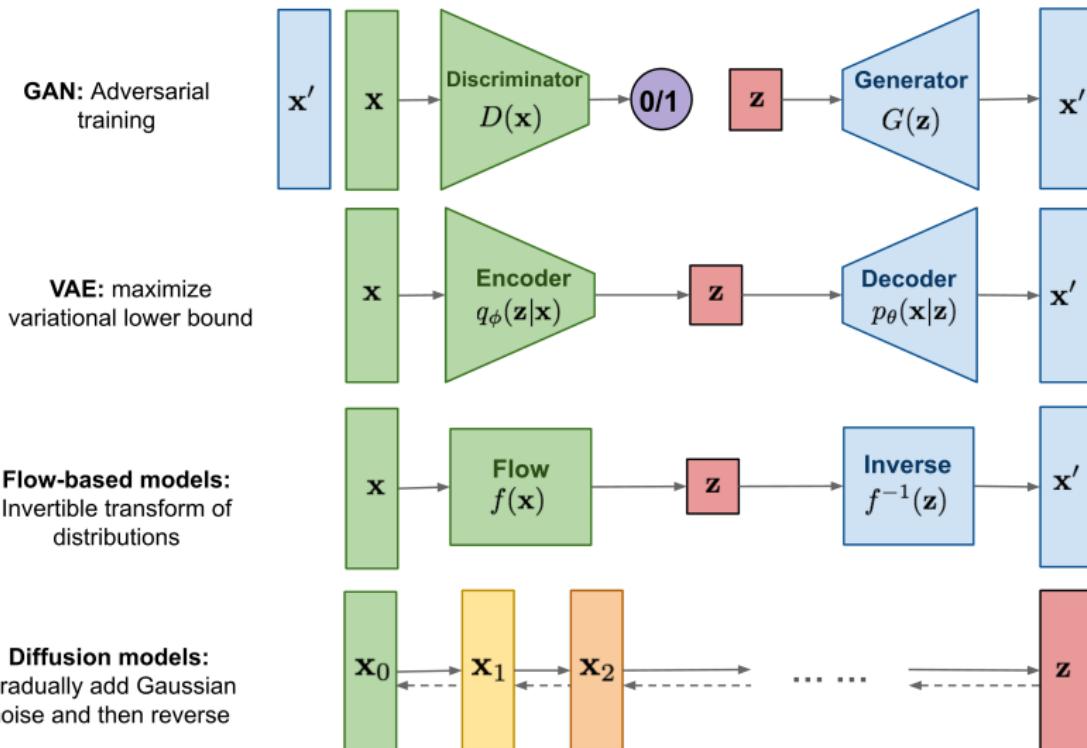
$$\mathbb{E}_{p(\mathbf{x}|\mathbf{x}', \sigma_I)} \left\| \mathbf{s}(\mathbf{x}, \theta, \sigma_I) - \nabla_{\mathbf{x}} \log p(\mathbf{x}|\mathbf{x}', \sigma_I) \right\|_2^2 \rightarrow \min_{\theta}$$

# Denoising diffusion probabilistic model (DDPM)

## Samples



# The poorest course overview :)



## Summary

- ▶ Noise conditioned score network uses multiple noise levels and annealed Langevin dynamics to fit score function.
- ▶ Gaussian diffusion process is a Markov chain that injects special form of Gaussian noise to the samples.
- ▶ Reverse process allows to sample from the real distribution  $\pi(\mathbf{x})$  using samples from noise.
- ▶ Diffusion model is a VAE model which reverts gaussian diffusion process using variational inference.
- ▶ Objective of diffusion model is closely related to the noise conditioned score network and score matching.