

Deep Generative Models

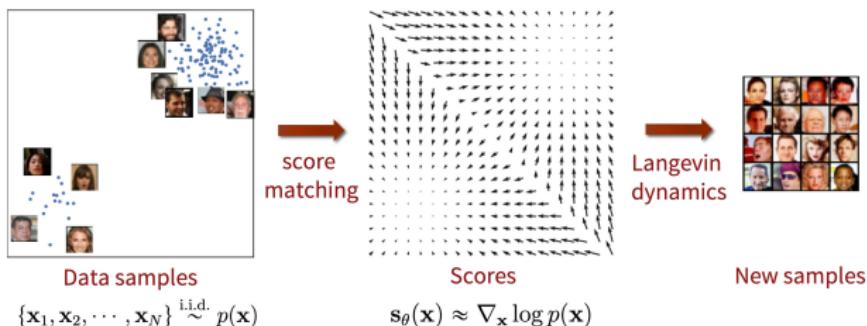
Lecture 13

Roman Isachenko



2024, Spring

Recap of previous lecture



Theorem (implicit score matching)

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

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

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

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

Theorem (denoising score matching)

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

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 $\nabla_{\mathbf{x}'} \log \pi(\mathbf{x}')$.
- ▶ $\mathbf{s}_\theta(\mathbf{x}', \sigma)$ tries to **denoise** a corrupted sample.
- ▶ Score function $\mathbf{s}_\theta(\mathbf{x}', \sigma)$ parametrized by σ .

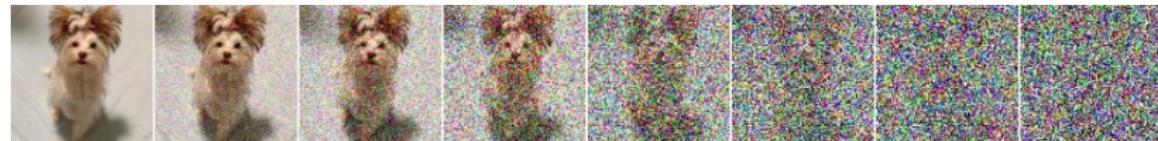
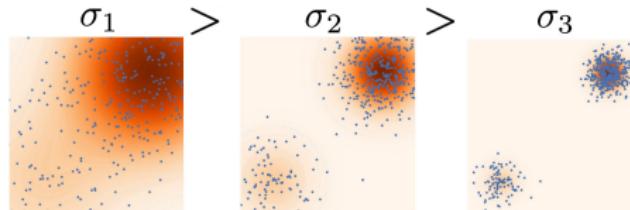
Recap of previous lecture

Noise conditioned score network

- ▶ Define the sequence of noise levels: $\sigma_1 > \sigma_2 > \dots > \sigma_L$.
- ▶ Train denoised score function $s_\theta(\mathbf{x}', \sigma)$ for each noise level:

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

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



Recap of previous lecture

NCSN training

1. Get the sample $\mathbf{x}_0 \sim \pi(\mathbf{x})$.
2. Sample noise level $l \sim U[1, L]$ and the noise $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.
3. Get noisy image $\mathbf{x}' = \mathbf{x}_0 + \sigma_l \cdot \epsilon$.
4. Compute loss $\mathcal{L} = \|\mathbf{s}_\theta(\mathbf{x}', \sigma_l) + \frac{\epsilon}{\sigma_l}\|^2$.

NCSN sampling (annealed Langevin dynamics)

- ▶ Sample $\mathbf{x}_0 \sim \mathcal{N}(0, \sigma_1 \mathbf{I}) \approx \pi(\mathbf{x}|\sigma_1)$.
- ▶ Apply T steps of Langevin dynamic

$$\mathbf{x}_t = \mathbf{x}_{t-1} + \frac{1}{2} \eta_l \mathbf{s}_\theta(\mathbf{x}_{t-1}, \sigma_l) + \sqrt{\eta_l} \epsilon_t.$$

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

Recap of previous lecture

NCSN objective

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

DDPM objective

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

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

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

Let reparametrize our model:

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

Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

3. The worst course overview

Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

3. The worst course overview

Stochastic differential equation (SDE)

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

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

- ▶ $\mathbf{f}(\mathbf{x}, t)$ is the **drift** function of $\mathbf{x}(t)$.
- ▶ $g(t)$ is the **diffusion** coefficient of $\mathbf{x}(t)$.
- ▶ If $g(t) = 0$ we get standard ODE.
- ▶ $\mathbf{w}(t)$ is the standard Wiener process (Brownian motion):
 1. $\mathbf{w}(0) = 0$ (almost surely);
 2. $\mathbf{w}(t)$ has independent increments;
 3. $\mathbf{w}(t) - \mathbf{w}(s) \sim \mathcal{N}(0, (t-s)\mathbf{I})$.
- ▶ $d\mathbf{w} = \mathbf{w}(t+dt) - \mathbf{w}(t) = \mathcal{N}(0, \mathbf{I} \cdot dt) = \epsilon \cdot \sqrt{dt}$, where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$.

Note: In contrast to ODE, initial condition $\mathbf{x}(0)$ does not uniquely determine the process trajectory.

Stochastic differential equation (SDE)

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + g(t)d\mathbf{w}, \quad d\mathbf{w} = \boldsymbol{\epsilon} \cdot \sqrt{dt}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}).$$

- ▶ At each moment t we have the density $p(\mathbf{x}(t), t)$.
- ▶ How to get distribution $p(\mathbf{x}, t)$ for $\mathbf{x}(t)$?

Theorem (Kolmogorov-Fokker-Planck)

Evolution of the distribution $p(\mathbf{x}, t)$ is given by the following ODE:

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} = \text{tr} \left(-\frac{\partial}{\partial \mathbf{x}} [\mathbf{f}(\mathbf{x}, t)p(\mathbf{x}, t)] + \frac{1}{2}g^2(t)\frac{\partial^2 p(\mathbf{x}, t)}{\partial \mathbf{x}^2} \right)$$

Note: This is the generalization of KFP theorem that we used in continuous-in-time NF.

Langevin SDE (special case)

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

$$d\mathbf{x} = \frac{1}{2}\frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}, t)dt + \mathbf{1} \cdot d\mathbf{w}$$

Langevin SDE (special case)

$$d\mathbf{x} = \frac{1}{2} \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}, t) dt + \mathbf{1} \cdot d\mathbf{w}$$

Let apply KFP theorem.

$$\begin{aligned}\frac{\partial p(\mathbf{x}, t)}{\partial t} &= \text{tr} \left(-\frac{\partial}{\partial \mathbf{x}} \left[p(\mathbf{x}, t) \frac{1}{2} \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}, t) \right] + \frac{1}{2} \frac{\partial^2 p(\mathbf{x}, t)}{\partial \mathbf{x}^2} \right) = \\ &= \text{tr} \left(-\frac{\partial}{\partial \mathbf{x}} \left[\frac{1}{2} \frac{\partial}{\partial \mathbf{x}} p(\mathbf{x}, t) \right] + \frac{1}{2} \frac{\partial^2 p(\mathbf{x}, t)}{\partial \mathbf{x}^2} \right) = 0\end{aligned}$$

The density $p(\mathbf{x}, t) = \text{const}(t)!$

Discretized Langevin SDE

$$\mathbf{x}_{t+1} - \mathbf{x}_t = \eta \frac{1}{2} \frac{\partial}{\partial \mathbf{x}} \log p(\mathbf{x}, t) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \eta \approx dt.$$

Langevin dynamic

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \eta \frac{1}{2} \nabla_{\mathbf{x}} \log p(\mathbf{x}|\theta) + \sqrt{\eta} \cdot \boldsymbol{\epsilon}, \quad \eta \approx dt.$$

Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

3. The worst course overview

Score matching SDE

Denoising score matching

$$\mathbf{x}_I = \mathbf{x} + \sigma_I \cdot \boldsymbol{\epsilon}_I, \quad p(\mathbf{x}_I | \mathbf{x}, \sigma_I) = \mathcal{N}(\mathbf{x}_I | \mathbf{x}, \sigma_I^2 \mathbf{I})$$

$$\mathbf{x}_{I-1} = \mathbf{x} + \sigma_{I-1} \cdot \boldsymbol{\epsilon}_{I-1}, \quad p(\mathbf{x}_{I-1} | \mathbf{x}, \sigma_{I-1}) = \mathcal{N}(\mathbf{x}_{I-1} | \mathbf{x}, \sigma_{I-1}^2 \mathbf{I})$$

$$\mathbf{x}_I = \mathbf{x}_{I-1} + \sqrt{\sigma_I^2 - \sigma_{I-1}^2} \cdot \boldsymbol{\epsilon}, \quad p(\mathbf{x}_I | \mathbf{x}_{I-1}, \sigma_I) = \mathcal{N}(\mathbf{x}_I | \mathbf{x}_{I-1}, (\sigma_I^2 - \sigma_{I-1}^2) \cdot \mathbf{I})$$

Let turn this Markov chain to the continuous stochastic process $\mathbf{x}(t)$ taking $L \rightarrow \infty$:

$$\mathbf{x}(t + dt) = \mathbf{x}(t) + \sqrt{\frac{\sigma^2(t + dt) - \sigma^2(t)}{dt}} \cdot \boldsymbol{\epsilon} = \mathbf{x}(t) + \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Exploding SDE

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Diffusion SDE

Denoising Diffusion

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

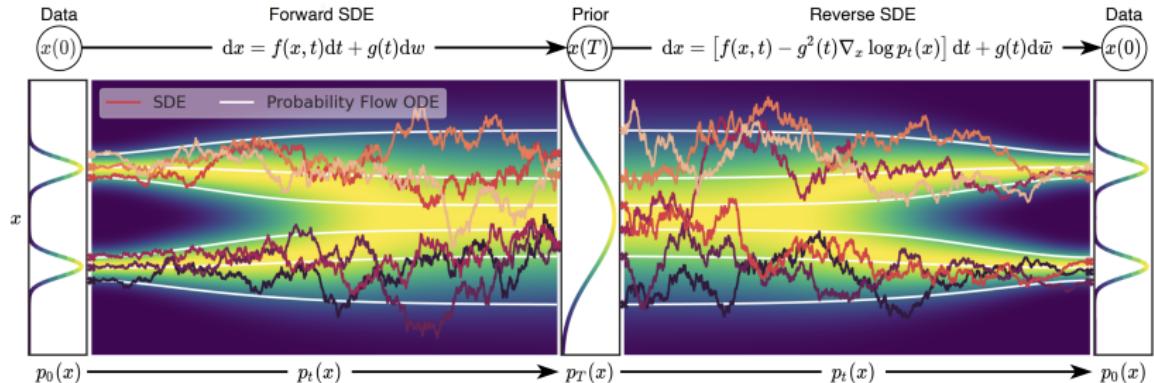
Let turn this Markov chain to the continuous stochastic process taking $T \rightarrow \infty$ and taking $\beta(\frac{t}{T}) = \beta_t \cdot T$

$$\begin{aligned}\mathbf{x}(t) &= \sqrt{1 - \beta(t)dt} \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} \approx \\ &\approx (1 - \frac{1}{2}\beta(t)dt) \cdot \mathbf{x}(t - dt) + \sqrt{\beta(t)dt} \cdot \boldsymbol{\epsilon} = \\ &= \mathbf{x}(t - dt) - \frac{1}{2}\beta(t)\mathbf{x}(t - dt)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}\end{aligned}$$

Variance Preserving SDE

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

Diffusion SDE



Variance Exploding SDE (NCSN)

$$d\mathbf{x} = \sqrt{\frac{d[\sigma^2(t)]}{dt}} \cdot d\mathbf{w}$$

Variance Preserving SDE (DDPM)

$$d\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}(t)dt + \sqrt{\beta(t)} \cdot d\mathbf{w}$$

Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

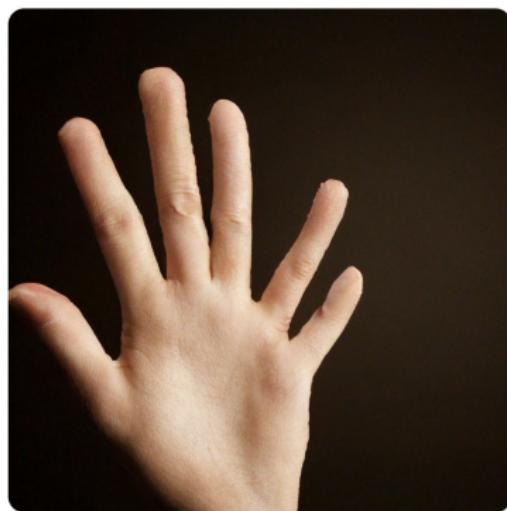
3. The worst course overview

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

Guidance

How to make conditional model $p(\mathbf{x}|\mathbf{y}, \theta)$?

- ▶ If we have **supervised** data $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$ 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 $\{\mathbf{x}_i\}_{i=1}^m$ we need to create the way to convert unconditional model $p(\mathbf{x}|\theta)$ to the conditional.

DDPM **unsupervised** guidance

- ▶ Let imagine we are given the distribution $q(\mathbf{y}|\mathbf{x}_0)$.
- ▶ Since we have already defined Markov chain, we have $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y}) = q(\mathbf{x}_t|\mathbf{x}_{t-1})$.
- ▶ Let try to find reverse $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{y})$.

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: $\gamma = 3$.



Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

3. The worst course overview

Classifier guidance

$$\begin{aligned} q(\mathbf{y}|\mathbf{x}_{t-1}, \mathbf{x}_t) &= \frac{q(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{y})}{q(\mathbf{x}_{t-1}, \mathbf{x}_t)} = \\ &= \frac{q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{y})q(\mathbf{y}|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1})}{q(\mathbf{x}_t|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1})} = q(\mathbf{y}|\mathbf{x}_{t-1}). \end{aligned}$$

Conditional distribution

$$\begin{aligned} q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{y}) &= \frac{q(\mathbf{x}_{t-1}, \mathbf{x}_t, \mathbf{y})}{q(\mathbf{x}_t, \mathbf{y})} = \\ &= \frac{q(\mathbf{y}|\mathbf{x}_{t-1}, \mathbf{x}_t)q(\mathbf{x}_{t-1}|\mathbf{x}_t)q(\mathbf{x}_t)}{q(\mathbf{y}|\mathbf{x}_t)q(\mathbf{x}_t)} = \\ &= q(\mathbf{y}|\mathbf{x}_{t-1})q(\mathbf{x}_{t-1}|\mathbf{x}_t) \cdot \text{const}(\mathbf{x}_{t-1}). \end{aligned}$$

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{y}, \theta, \phi) = p(\mathbf{y}|\mathbf{x}_{t-1}, \phi)p(\mathbf{x}_{t-1}|\mathbf{x}_t, \theta) \cdot \text{const}(\mathbf{x}_{t-1}).$$

- ▶ $p(\mathbf{x}_{t-1}|\mathbf{x}_t, \theta)$ - our unsupervised diffusion model.
- ▶ $p(\mathbf{y}|\mathbf{x}_{t-1}, \phi)$ - classifier for noised samples \mathbf{x}_{t-1}

Classifier guidance

Conditional distribution

$$p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\phi}) = p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi}) \cdot p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) \cdot \text{const}(\mathbf{x}_{t-1})$$

$$\log p(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{y}, \boldsymbol{\theta}, \boldsymbol{\phi}) = \log p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi}) + \log p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) + \text{const}$$

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

$$\log p(\mathbf{x}_{t-1}|\mathbf{x}_t, \boldsymbol{\theta}) = -\frac{\|\mathbf{x}_{t-1} - \boldsymbol{\mu}\|^2}{2\boldsymbol{\sigma}^2} + \text{const}(\mathbf{x}_{t-1})$$

Taylor expansion

$$\begin{aligned} \log p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi}) &\approx \log p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi})|_{\mathbf{x}_{t-1}=\boldsymbol{\mu}} + \\ &+ (\mathbf{x}_{t-1} - \boldsymbol{\mu}) \cdot \nabla_{\mathbf{x}_{t-1}} \log p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi})|_{\mathbf{x}_{t-1}=\boldsymbol{\mu}} = \\ &= (\mathbf{x}_{t-1} - \boldsymbol{\mu}) \cdot \mathbf{g} + \text{const}(\mathbf{x}_{t-1}), \end{aligned}$$

where $\mathbf{g} = \nabla_{\mathbf{x}_{t-1}} \log p(\mathbf{y}|\mathbf{x}_{t-1}, \boldsymbol{\phi})|_{\mathbf{x}_{t-1}=\boldsymbol{\mu}}$.

Classifier guidance

$$\begin{aligned}\log p(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{y}, \boldsymbol{\theta}, \phi) &= (\mathbf{x}_{t-1} - \boldsymbol{\mu}) \cdot \mathbf{g} - \frac{\|\mathbf{x}_{t-1} - \boldsymbol{\mu}\|^2}{2\sigma^2} + \text{const}(\mathbf{x}_{t-1}) \\ &= -\frac{\|\mathbf{x}_{t-1} - \boldsymbol{\mu} - \boldsymbol{\sigma} \odot \mathbf{g}\|^2}{2\sigma^2} + \text{const}(\mathbf{x}_{t-1}) \\ &= \log \mathcal{N}(\boldsymbol{\mu} + \boldsymbol{\sigma} \odot \mathbf{g}, \sigma^2) + \text{const}(\mathbf{x}_{t-1})\end{aligned}$$

Guided sampling

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

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

3. Compute $\mathbf{g} = \nabla_{\mathbf{x}_{t-1}} \log p(\mathbf{y} | \mathbf{x}_{t-1}, \phi)|_{\mathbf{x}_{t-1}=\boldsymbol{\mu}}$.
4. Get denoised image $\mathbf{x}_{t-1} = (\boldsymbol{\mu}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) + \sqrt{\tilde{\beta}_t} \cdot \mathbf{g}) + \sqrt{\tilde{\beta}_t} \cdot \boldsymbol{\epsilon}$, where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$.

Classifier guidance

Theorem (denoising score matching)

$$\begin{aligned}\mathbb{E}_{q(\mathbf{x}_t)} \|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t)\|_2^2 &= \\ = \mathbb{E}_{\pi(\mathbf{x}_0)} \mathbb{E}_{q(\mathbf{x}_t|\mathbf{x}_0)} \|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t|\mathbf{x}_0)\|_2^2 + \text{const}(\theta) \\ \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\theta) &\approx -\frac{\epsilon_\theta(\mathbf{x}_t, t)}{\sqrt{1 - \bar{\alpha}_t}} = \mathbf{s}_\theta(\mathbf{x}_t, t).\end{aligned}$$

Conditional distribution

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

Classifier-corrected noise prediction

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

Classifier guidance

Classifier-corrected noise prediction

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

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

Here we introduce **guidance scale γ** that controls the magnitude of the classifier guidance.

Conditional distribution

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

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

$$p(\mathbf{x}_t|\mathbf{y}, \theta, \phi) = \frac{p(\mathbf{y}|\mathbf{x}_t, \phi)^{\gamma} p(\mathbf{x}_t|\theta)}{Z}$$

Check that $\nabla_{\mathbf{x}_t} \log Z \neq 0$.

Classifier guidance

Guided sampling

1. Sample $\mathbf{x}_T \sim \mathcal{N}(0, \mathbf{I})$.
2. Compute "corrected" $\hat{\epsilon}_{\theta}(\mathbf{x}_t, t)$:

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

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

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

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

Outline

1. Diffusion and Score matching SDEs

2. Guidance

Classifier guidance

Classifier-free guidance

3. The worst course overview

Classifier-free guidance

Previous methods require training the additional classifier model $p(\mathbf{y}|\mathbf{x}_t, \theta)$ on the noisy data. Let's try to avoid this requirement.

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

What will we get if $\gamma = 1$?

Classifier-free-corrected noise prediction

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

In practice we could train the single model $\epsilon_\theta(\mathbf{x}_t, \mathbf{y}, t)$ on **supervised** data alternating with real conditioning \mathbf{y} and empty conditioning $\mathbf{y} = \emptyset$.

Outline

1. Diffusion and Score matching SDEs

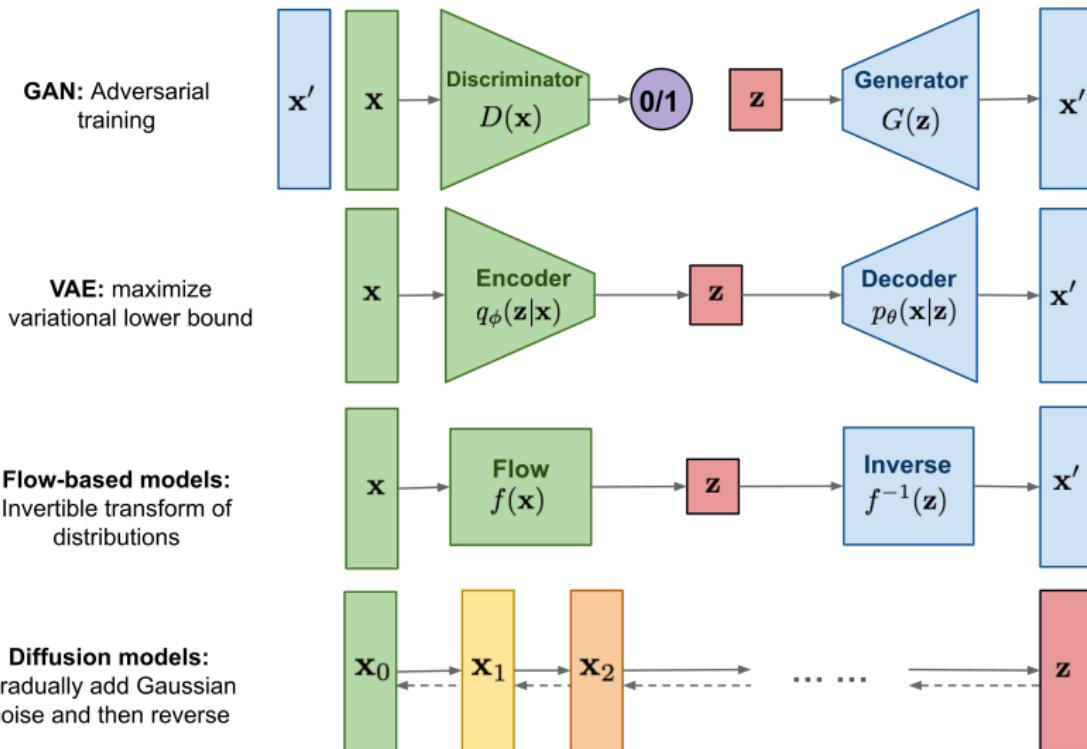
2. Guidance

Classifier guidance

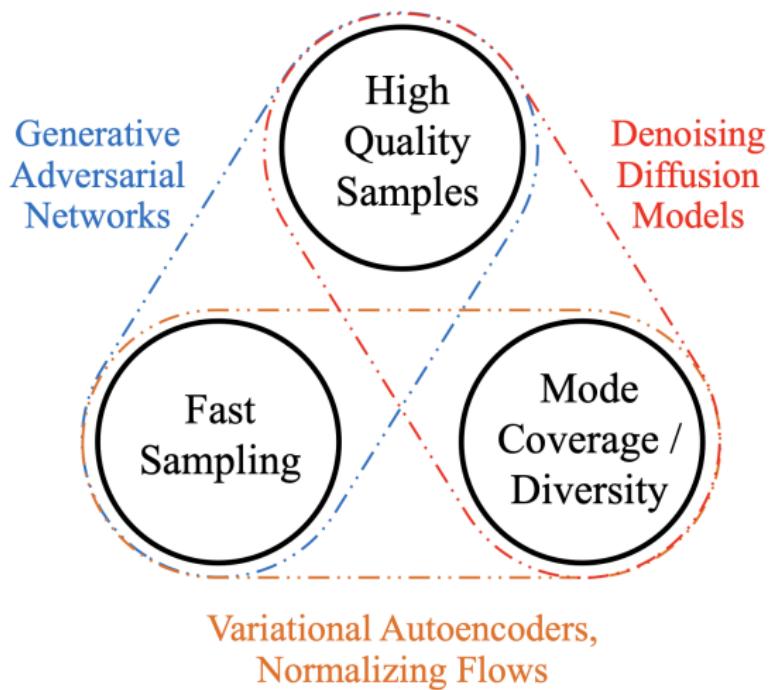
Classifier-free guidance

3. The worst course overview

The worst course overview :)



The worst course overview :)



The worst course overview :)

Model	Efficient	Sample quality	Coverage	Well-behaved latent space	Disentangled latent space	Efficient likelihood
GANs	✓	✓	✗	✓	?	n/a
VAEs	✓	✗	?	✓	?	✗
Flows	✓	✗	?	✓	?	✓
Diffusion	✗	✓	?	✗	✗	✗

Summary

- ▶ Score matching (NCSN) and diffusion models (DDPM) are the discretizations of the SDEs (variance exploding and variance preserving).
- ▶ Conditional models use labels y as the additional input. Majority of the modern generative models are conditional.
- ▶ 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.