SCORE-BASED GENERATIVE MODELING THROUGH STOCHASTIC DIFFERENTIAL EQUATIONS

김용진

Score based model & DDPM

- Score based model은 여러 강도의 gaussian noise가 더해진 이미지를 이용하여 score matching -> annealed langevin dynamics로 sampling

- DDPM은 하나의 Image에 대해서 여러 강도의 gaussian noise를 연속해서 add -> reverse process를 통해 noise를 걷어내며 sampling

Recap Score based model

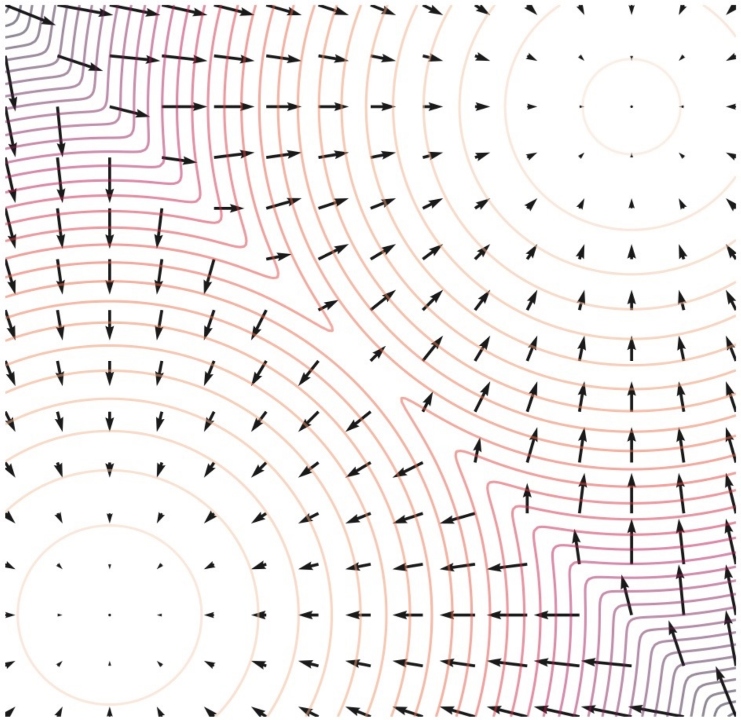
Fisher divergence를 통해 학습시킬 수 있다!

- Fisher divergence란 두 분포의 유사도를 측정하기 위해 derivative를 사용하는 것

$$\mathbb{E}_{p(\mathbf{x})}[\|\nabla_{\mathbf{x}} \log p(\mathbf{x}) - \nabla_{\mathbf{x}} \log q(\mathbf{x})\|_2^2].$$

Score based model에서의 Fisher divergence는

$$\mathbb{E}_{p(\mathbf{x})}[\|\nabla_{\mathbf{x}} \log p(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x})\|_2^2]$$



Recap Score based model

1. Denoising score matching

Data에 perturbation noise를 더해서 score matching을 진행

Data point x -> $q_\sigma(\tilde{x}|x)$
Perturbed data distribution $q_\sigma(\tilde{x}) \triangleq \int q_\sigma(\tilde{x}|x)\, p_{data}(x)dx$

$$\frac{1}{2}\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p_{\mathrm{data}}(\mathbf{x})}[\|\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})\|_2^2].$$

Recap Score based model

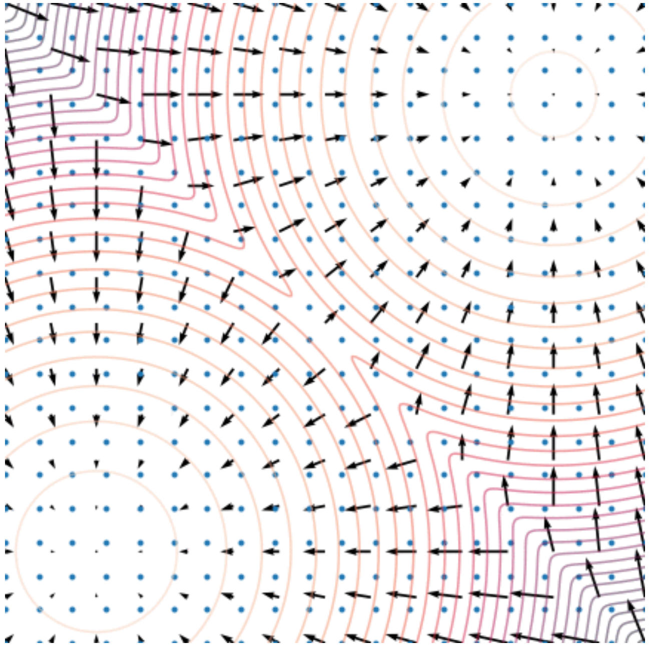
Score function을 구하는건 알겠는데 Sampling은 어떻게?

- Langevin dynamics는 MCMC방법을 통해 Score function만 가지고 p(x)를 sampling할 수 있는 방법론

$$\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i + \epsilon \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \sqrt{2\epsilon}\, \mathbf{z}_i, \quad i = 0, 1, \cdots, K, \quad \mathbf{x}_0 \sim \pi(\mathbf{x}).$$

$$\mathbf{z}_i \sim \mathcal{N}(0, I)$$

일반적인 상황에서 ε -> 0 & K -> ∞ 이면, x_k는 P(x)로 수렴한다.
ε이 충분히 작고 K가 충분히 크다면 error는 무시할 수 있는 수준이 된다.



우리는 앞서 Score based model을 이용하여 Score function을 근사했기 때문에 Langevin dynamics를 이용할 수 있게 된다.

Recap DDPM

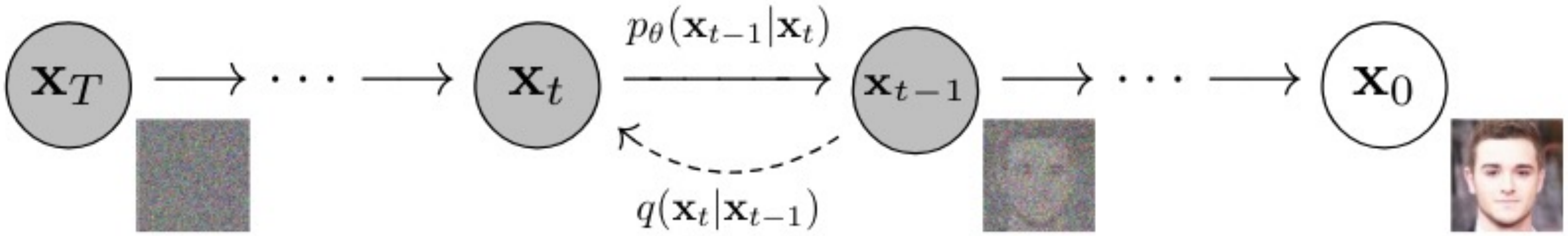


Figure 2: The directed graphical model considered in this work.

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon}-\boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0+\sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon},t)\right\|^2\right]$$

Recap DDPM

DDPM algorithm

Algorithm 1 Training

repeat
  $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
  $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
    $\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$
until converged

Algorithm 2 Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \ldots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
end for
return $\mathbf{x}_0$

Langevin Dynamics sampling

$$\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i + \epsilon \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \sqrt{2\epsilon}\, \mathbf{z}_i, \quad i = 0, 1, \cdots, K,$$

DDPM = Score matching!

Score matching Loss function

$$\frac{1}{2}\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p_{\text{data}}(\mathbf{x})}[\|\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}}\log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})\|_2^2].$$

DDPM

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2\right]$$

DDPM = Score matching!

Score matching Loss function

$$\frac{1}{2}\mathbb{E}_{q_\sigma(\tilde{\mathbf{x}}|\mathbf{x})p_{\text{data}}(\mathbf{x})}[\|\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}) - \nabla_{\tilde{\mathbf{x}}} \log q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})\|_2^2].$$

DDPM

$$\mathbb{E}_{\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\frac{\beta_t^2}{2\sigma_t^2\alpha_t(1-\bar{\alpha}_t)}\left\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t)\right\|^2\right]$$

$\log q_\sigma(\tilde{x}|x)$ → x input에 대한 gaussian noise 추가

$$q_\sigma(\tilde{x}|x) = \tilde{x} \sim N(x, \sigma^2 I)$$

$$q_\sigma(\tilde{x}|x) = \frac{1}{\sqrt{2\pi}\cdot\sigma}\exp\left(-\frac{1}{2}\cdot\frac{(\tilde{x}-x)^2}{\sigma^2}\right)$$

$$\nabla_{\tilde{x}} \log q_\sigma(\tilde{x}|x) = -\frac{\tilde{x}-x}{\sigma^2} = -\frac{\sigma\varepsilon}{\sigma^2} = -\frac{\varepsilon}{\sigma}$$

$$\tilde{x} = x + \sigma\varepsilon \qquad \varepsilon \sim (0, I)$$

1. $s_\theta(\tilde{x},\sigma) = -\dfrac{\tilde{x} - D_\theta(\tilde{x},\sigma)}{\sigma^2} = -\dfrac{\varepsilon_\theta(\tilde{x},\sigma)}{\sigma}$

2. $s_\theta(\tilde{x},\sigma) - \nabla_x \log q_\sigma(\tilde{x}|x)$

$$= \frac{D_\theta(\tilde{x},\sigma) - x}{\sigma^2} = -\frac{\varepsilon_\theta(\tilde{x},\sigma) - \varepsilon}{\sigma}$$

$$= \frac{\varepsilon - \varepsilon_\theta(\tilde{x},\sigma)}{\sigma}$$

DDPM = Score based model

DDPM Loss

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} (1-\alpha_i)\mathbb{E}_{p_{\mathrm{data}}(\mathbf{x})}\mathbb{E}_{p_{\alpha_i}(\tilde{\mathbf{x}}|\mathbf{x})}[\|\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}, i) - \nabla_{\tilde{\mathbf{x}}} \log p_{\alpha_i}(\tilde{\mathbf{x}} \mid \mathbf{x})\|_2^2].$$

DDPM Sampling

$$\mathbf{x}_{i-1} = \frac{1}{\sqrt{1-\beta_i}}(\mathbf{x}_i + \beta_i \mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_i, i)) + \sqrt{\beta_i}\mathbf{z}_i, \quad i = N, N-1, \cdots, 1.$$

Score based model Loss

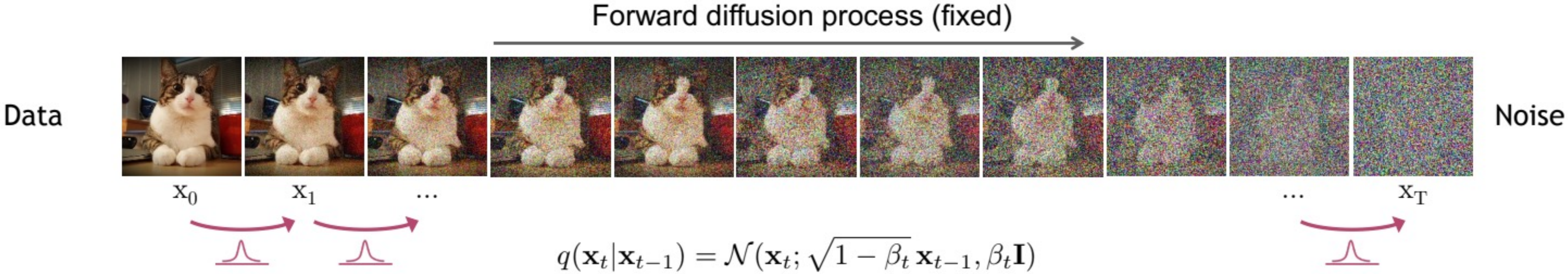
$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{N} \sigma_i^2 \mathbb{E}_{p_{\mathrm{data}}(\mathbf{x})}\mathbb{E}_{p_{\sigma_i}(\tilde{\mathbf{x}}|\mathbf{x})}\big[\|\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}, \sigma_i) - \nabla_{\tilde{\mathbf{x}}} \log p_{\sigma_i}(\tilde{\mathbf{x}} \mid \mathbf{x})\|_2^2\big].$$

Score based model Sampling

$$\mathbf{x}_i^m = \mathbf{x}_i^{m-1} + \epsilon_i \mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_i^{m-1}, \sigma_i) + \sqrt{2\epsilon_i}\mathbf{z}_i^m, \quad m = 1, 2, \cdots, M,$$

Forward Diffusion Process

Consider the limit of **many small steps:**



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

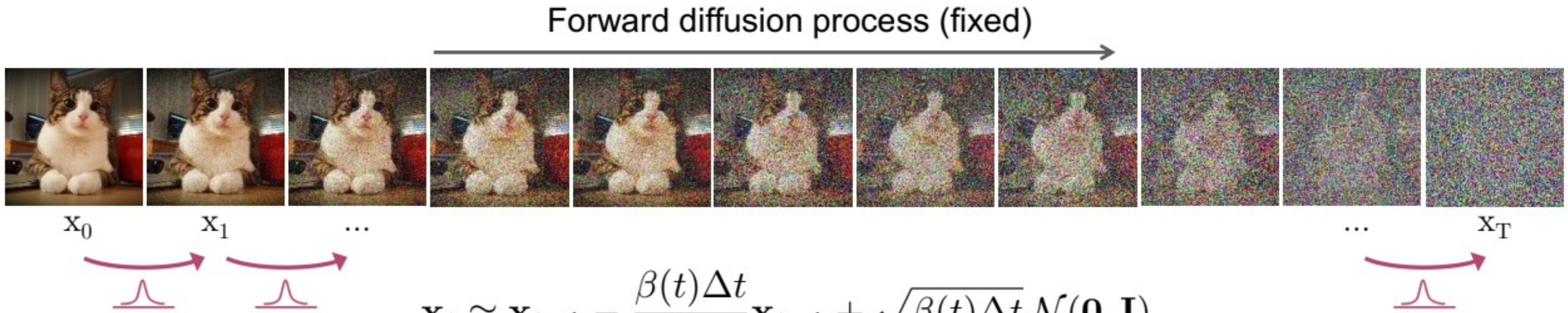
$$\begin{aligned}
\mathbf{x}_t &= \sqrt{1-\beta_t}\,\mathbf{x}_{t-1} + \sqrt{\beta_t}\,\mathcal{N}(\mathbf{0},\mathbf{I}) \\
&= \sqrt{1-\beta(t)\Delta t}\,\mathbf{x}_{t-1} + \sqrt{\beta(t)\Delta t}\,\mathcal{N}(\mathbf{0},\mathbf{I}) && (\beta_t := \beta(t)\Delta t) \\
\Rightarrow \quad &\approx \mathbf{x}_{t-1} - \frac{\beta(t)\Delta t}{2}\mathbf{x}_{t-1} + \sqrt{\beta(t)\Delta t}\,\mathcal{N}(\mathbf{0},\mathbf{I}) && \text{(Taylor expansion)}
\end{aligned}$$

Song et al., "Score-Based Generative Modeling through Stochastic Differential Equations", *ICLR*, 2021

Forward Diffusion Process as Stochastic Differential Equation

Consider the limit of **many small steps:**

Data



Noise

$$\mathbf{x}_t \approx \mathbf{x}_{t-1} - \frac{\beta(t)\Delta t}{2}\mathbf{x}_{t-1} + \sqrt{\beta(t)\Delta t}\,\mathcal{N}(\mathbf{0},\mathbf{I})$$

$$\mathrm{d}\mathbf{x}_t = -\frac{1}{2}\beta(t)\mathbf{x}_t\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\boldsymbol{\omega}_t$$

Stochastic Differential Equation (SDE)
describing the diffusion in infinitesimal limit

Song et al., "Score-Based Generative Modeling through Stochastic Differential Equations", *ICLR*, 2021

Perturbing data with SDE

$\{\mathbf{x}(t)\}_{t=0}^{T} \quad t \in [0, T] \qquad \mathbf{x}(0) \sim p_0, \ \ \mathbf{x}(T) \sim p_T$

t=0부터 T까지의 continuous time에서 Ito SDE는 다음과 같이 정의된다.

$\mathrm{d}\mathbf{x} = \underbrace{\mathbf{f}(\mathbf{x}, t)\mathrm{d}t}_{\text{Drift term}} + \underbrace{g(t)\mathrm{d}\mathbf{w},}_{\text{Diffusion term}}$ w = Gaussian white Noise

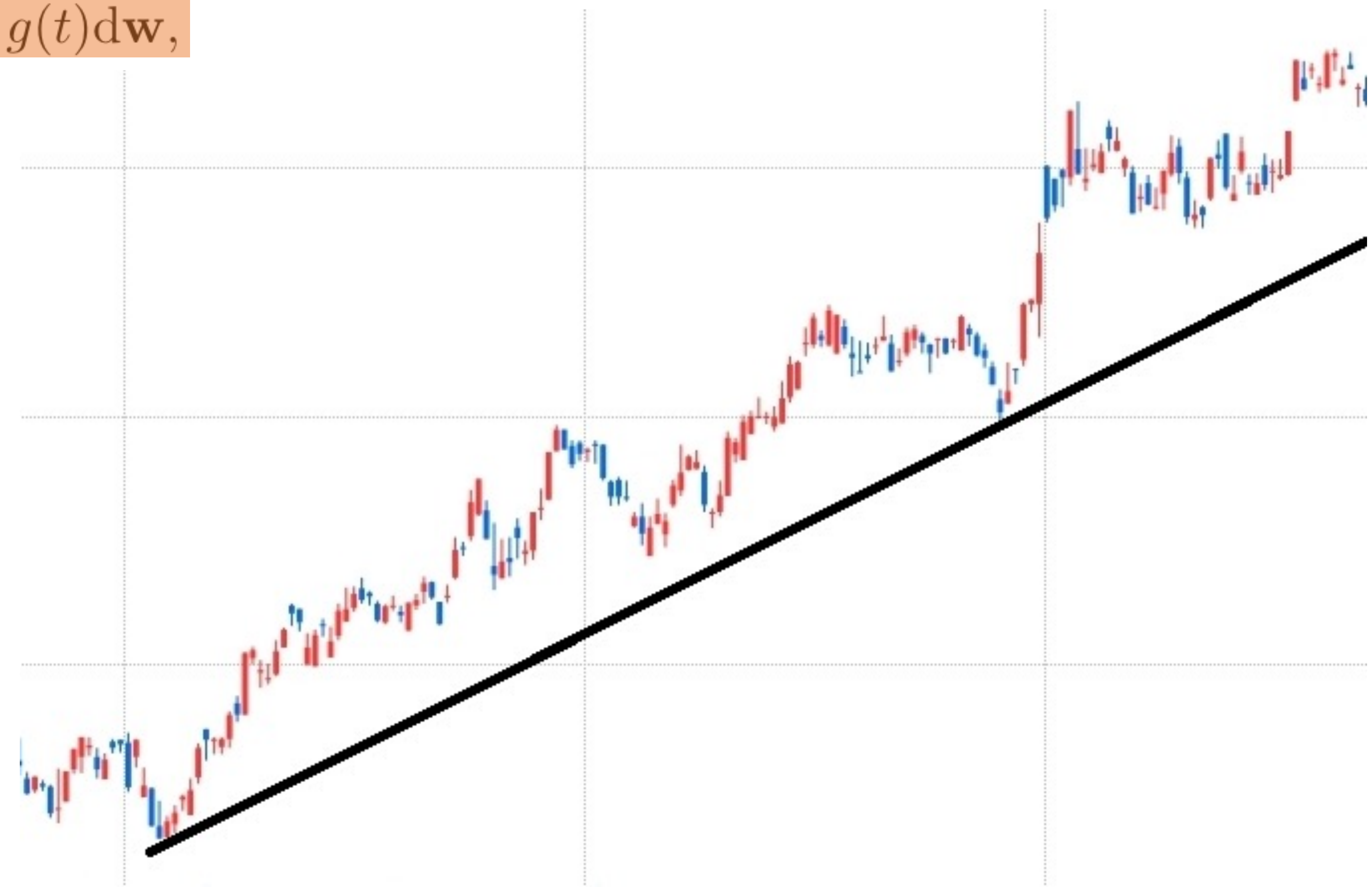
$\mathbf{f}(\cdot, t) : \mathbb{R}^d \rightarrow \mathbb{R}^d$

$g(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$

$$X_m = X_0 + \sum_{0 \le j \le m-1} f(t_j, X_j)\Delta t_j + \sum_{0 \le j \le m-1} G(t_j, X_j)\Delta W_j$$

주식 차트

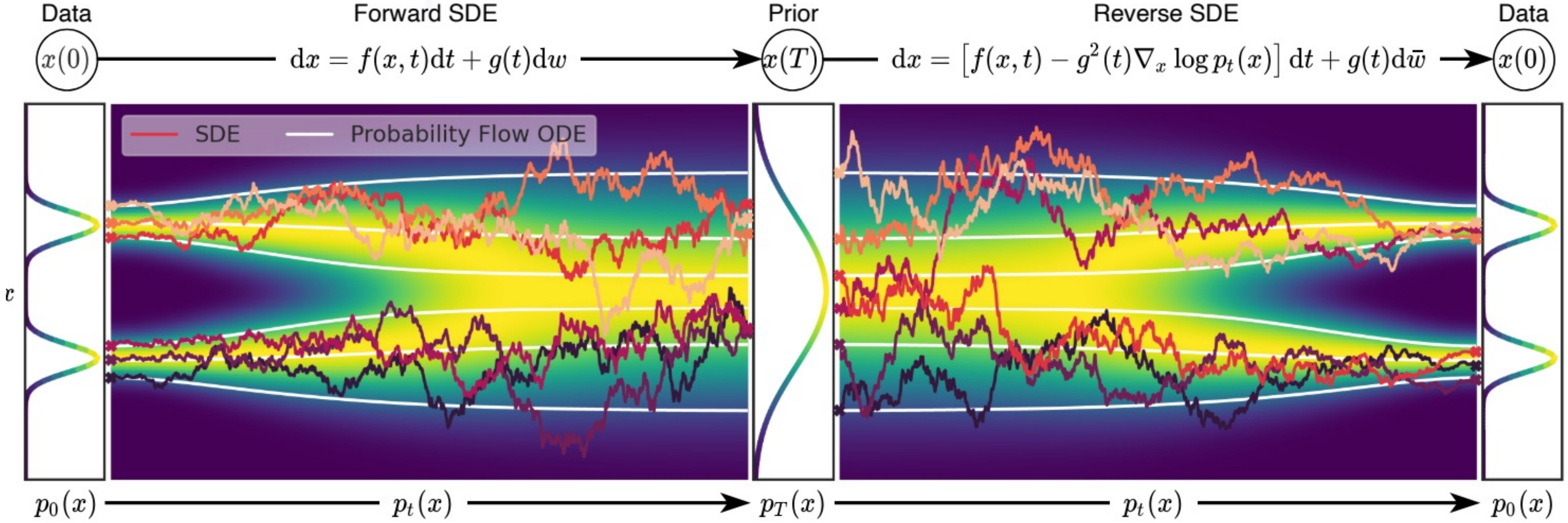
$$d\mathbf{x} = \boxed{\mathbf{f}(\mathbf{x}, t)dt} + \boxed{g(t)d\mathbf{w}},$$



Perturbing data with SDE

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

Drift term Diffusion term



SDE

- Variance Exploding(VE) SDE -> SMLD

$$\mathrm{d}\mathbf{x} = \sqrt{\frac{\mathrm{d}\left[\sigma^2(t)\right]}{\mathrm{d}t}}\mathrm{d}\mathbf{w}.$$

- Variance Preserving(VP) SDE -> DDPM

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\,\mathrm{d}t + \sqrt{\beta(t)}\,\mathrm{d}\mathbf{w}.$$

- Sub-VP SDE (Proposed)

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x}\,\mathrm{d}t + \sqrt{\beta(t)(1 - e^{-2\int_0^t \beta(s)\mathrm{d}s})}\mathrm{d}\mathbf{w}.$$

참고: https://github.com/yang-song/score_sde/blob/main/sde_lib.py

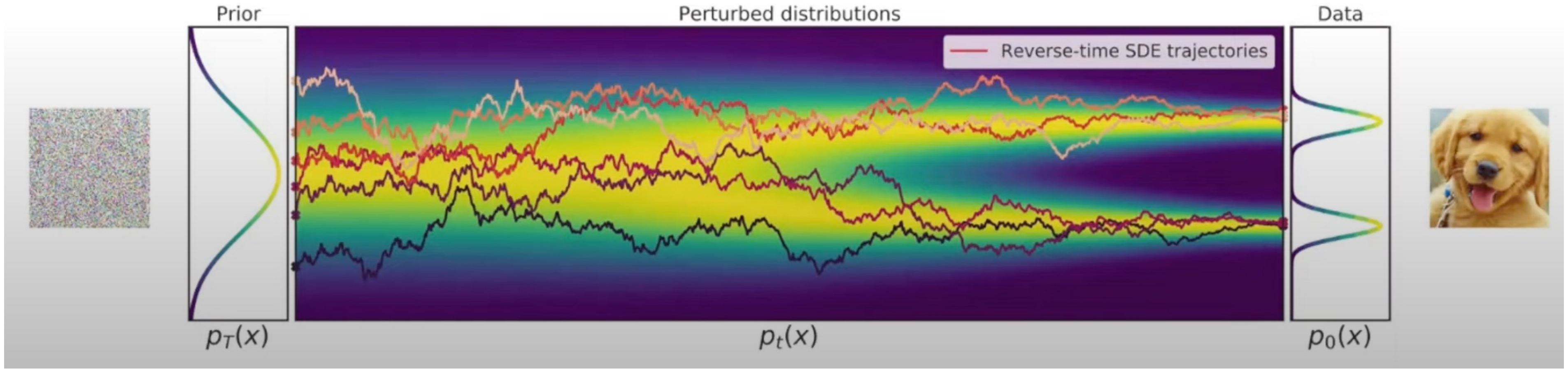
Reverse SDE

Forward SDE

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

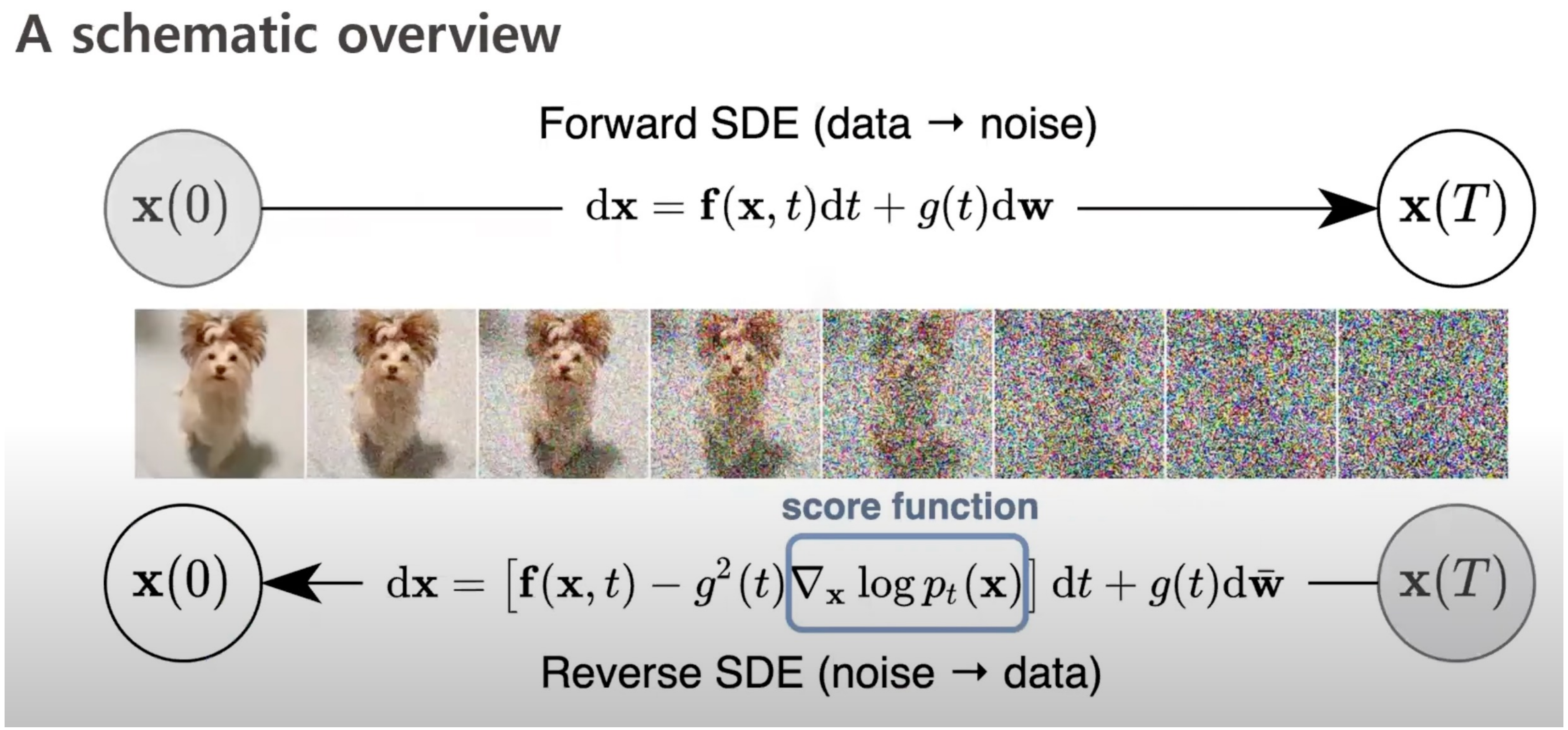
Reverse SDE (Anderson 1982)

$$d\mathbf{x} = [\boldsymbol{f}(\mathbf{x}, t) - g^2(t)\nabla_{\mathbf{x}} \log p_t(\mathbf{x})]\,dt + g(t)\,d\mathbf{w}$$



출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

SDE Solver



출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

SDE solver

- Approximate the reverse SDE with our score-based model.

$$\mathrm{d}\mathbf{x} = [\boldsymbol{f}(\mathbf{x},t) - g^2(t)\boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x},t)]\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}$$

- Numerical SDE solvers.
  - E.g., Euler-Maruyama solver

Initialize $\quad t \leftarrow T, \quad \mathbf{x} \sim p_T(\mathbf{x})$

$$\Delta\mathbf{x} \leftarrow [\boldsymbol{f}(\mathbf{x},t) - g^2(t)\boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x},t)]\Delta t + g(t)\,\mathbf{z}$$

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, |\Delta t|\boldsymbol{I})$$

출처: https://www.youtube.com/watch?v=L9ZegT87QK8
참고: https://github.com/yang-song/score_sde/blob/main/sampling.py#L184

SDE solver

- Approximate the reverse SDE with our score-based model.

$$\mathrm{d}\mathbf{x} = [\boldsymbol{f}(\mathbf{x}, t) - g^2(t)\boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x}, t)]\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}$$

- Numerical SDE solvers.
  - E.g., Euler-Maruyama solver

Initialize $\quad t \leftarrow T, \quad \mathbf{x} \sim p_T(\mathbf{x})$

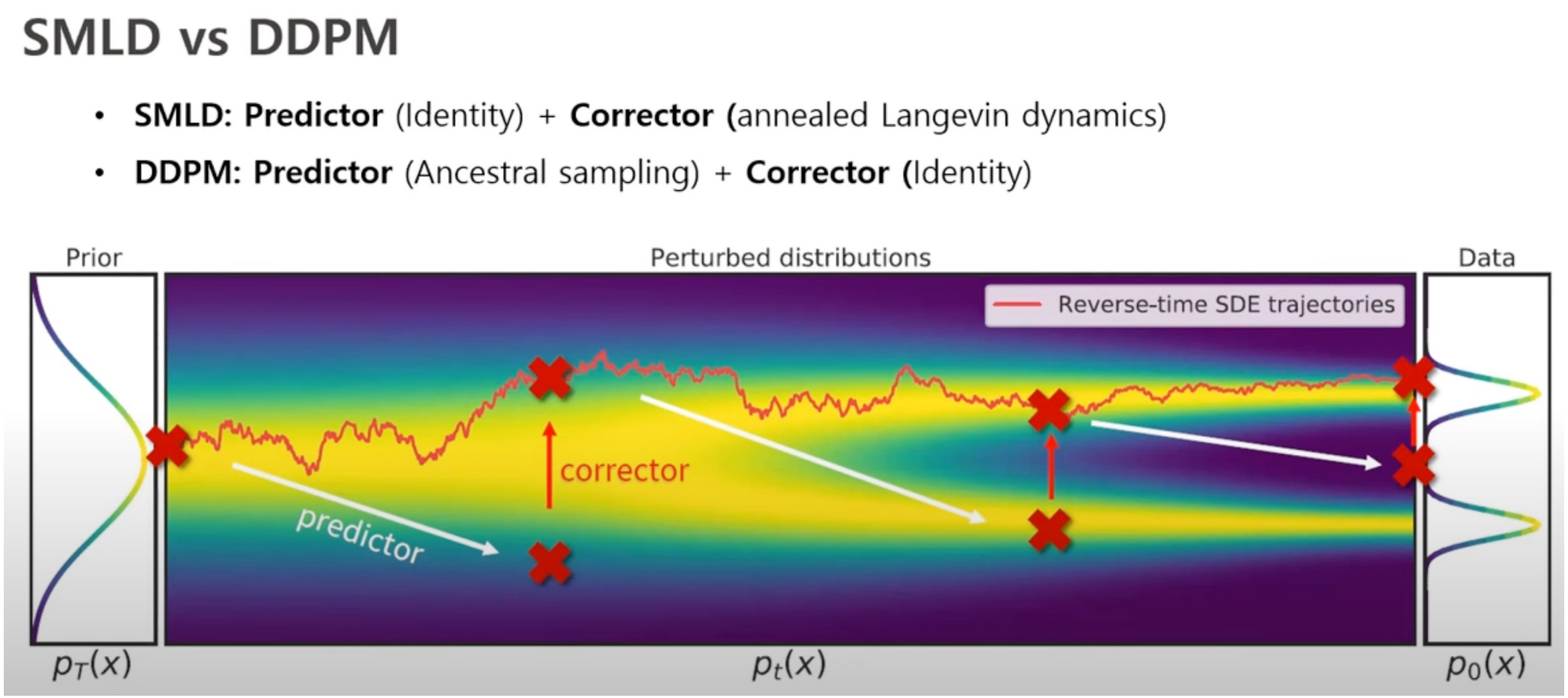
$$\Delta\mathbf{x} \leftarrow [\boldsymbol{f}(\mathbf{x}, t) - g^2(t)\boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x}, t)]\Delta t + g(t)\mathbf{z}$$

$$\mathbf{x} \leftarrow \mathbf{x} + \Delta\mathbf{x} \qquad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, |\Delta t|\boldsymbol{I})$$

$$t \leftarrow t + \Delta t$$

출처: https://www.youtube.com/watch?v=L9ZegT87QK8

Predictor-Corrector Sampling Method



출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

SDE to ODE

Probability flow ODEs as continuous normalizing flows

Probability flow ODE is an instance of Neural ODE (Chen et al. 2018)

$$\mathrm{d}\mathbf{x} = \left[\boldsymbol{f}(\mathbf{x}, t) - \frac{1}{2} g^2(t) \boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x}, t) \right] \mathrm{d}t$$

- Efficient adaptive ODE solvers for sampling

NFE = Number of score Function Evaluations

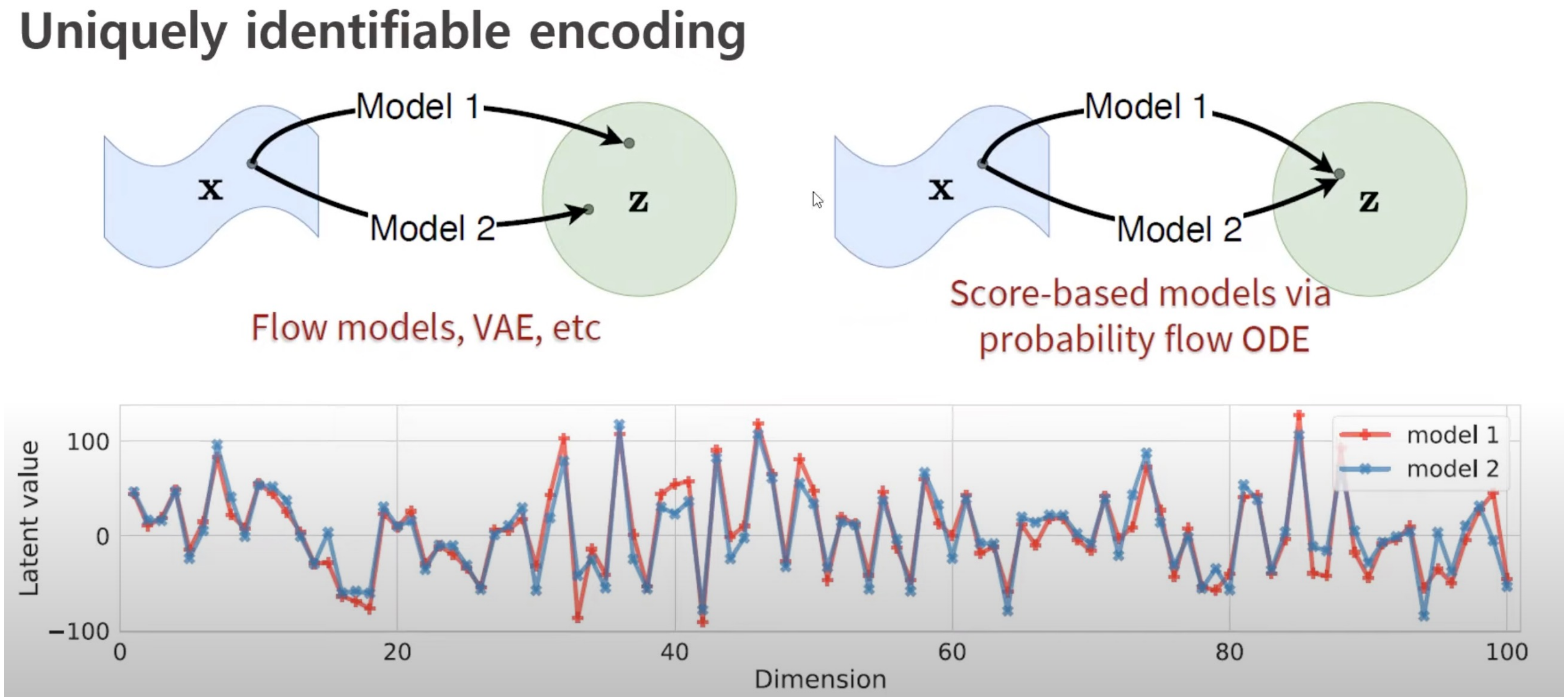
- SDE solvers:
  - $\approx$ 1000 NFE
- Adaptive ODE solver
  - $\approx$ 100 NFE

- Exact likelihood computation via instantaneous change-of-variable

$$\log p_0(\mathbf{x}) \xleftarrow{\int} \log p_T(\mathbf{z})$$

출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

SDE to ODE



출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

Conditional Generation

Conditional reverse-time SDE via unconditional scores

$$\mathrm{d}\mathbf{x} = [\boldsymbol{f}(\mathbf{x}, t) - g^2(t) \boxed{\nabla_{\mathbf{x}} \log p_t(\mathbf{x} \mid \mathbf{y})}]\, \mathrm{d}t + g(t)\, \mathrm{d}\mathbf{w}$$

$$\mathrm{d}\mathbf{x} = [\boldsymbol{f}(\mathbf{x}, t) - g^2(t) \boxed{\nabla_{\mathbf{x}} \log p_t(\mathbf{x})} - g^2(t) \boxed{\nabla_{\mathbf{x}} \log p_t(\mathbf{y} \mid \mathbf{x})}]\, \mathrm{d}t + g(t)\, \mathrm{d}\mathbf{w}$$

unconditional score,
Trained w/o y

Trained separately or
specified with domain knowledge

출처: https://www.youtube.com/watch?v=uG2ceFnUeQU

Q&A