All About Score-based Models

Seunghan Lee

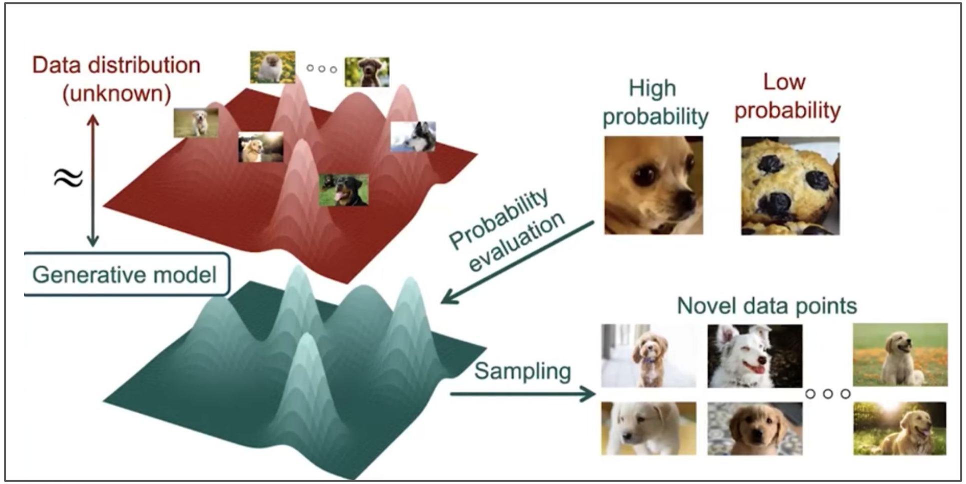
Department of Statistics and Data Science, Yonsei University

Outlines

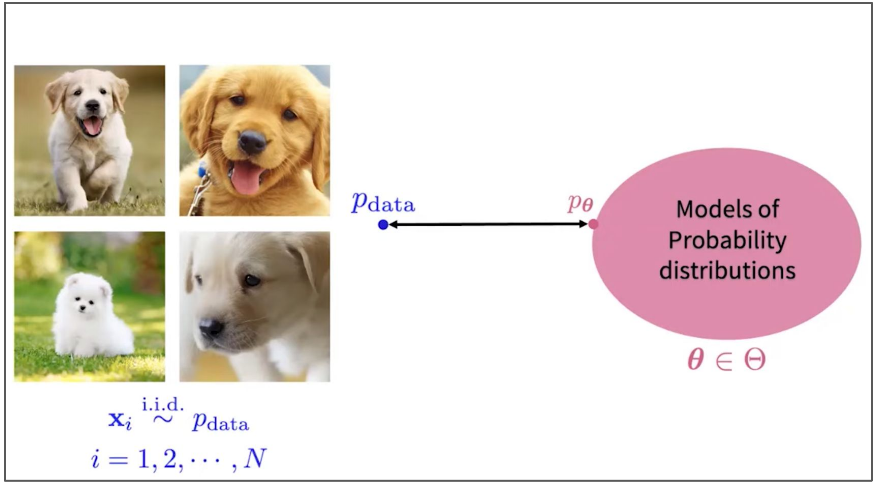
1. Introduction
2. Score-based Models
    a. Flexible Model
    b. Improved Generation
    c. Probability Evaluation

1. Introduction

1. Introduction



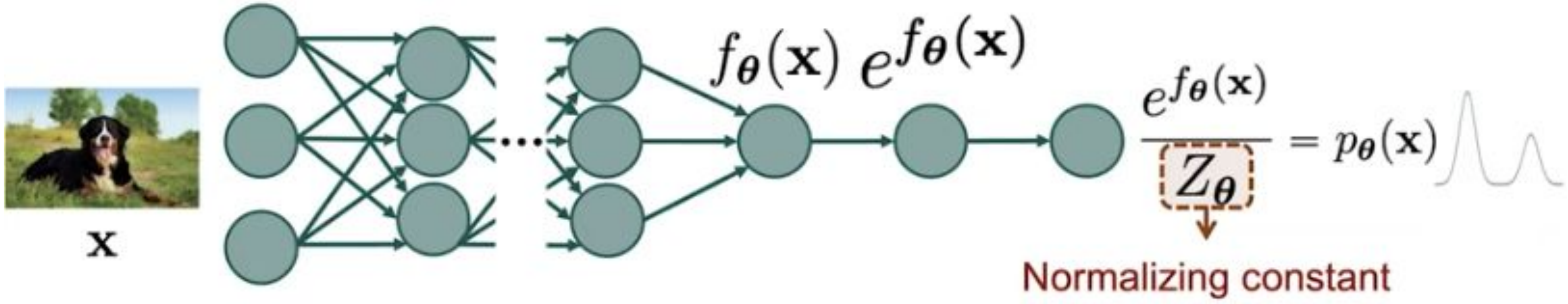
Generative Model



Approximate p_data using the model!

1. Introduction

Key challenge: **HIGH**-dimensional data



Intractable!

(1) Approximate normalizing constant

(2) Restricted NN

(3) Model only the generation process

1. Introduction

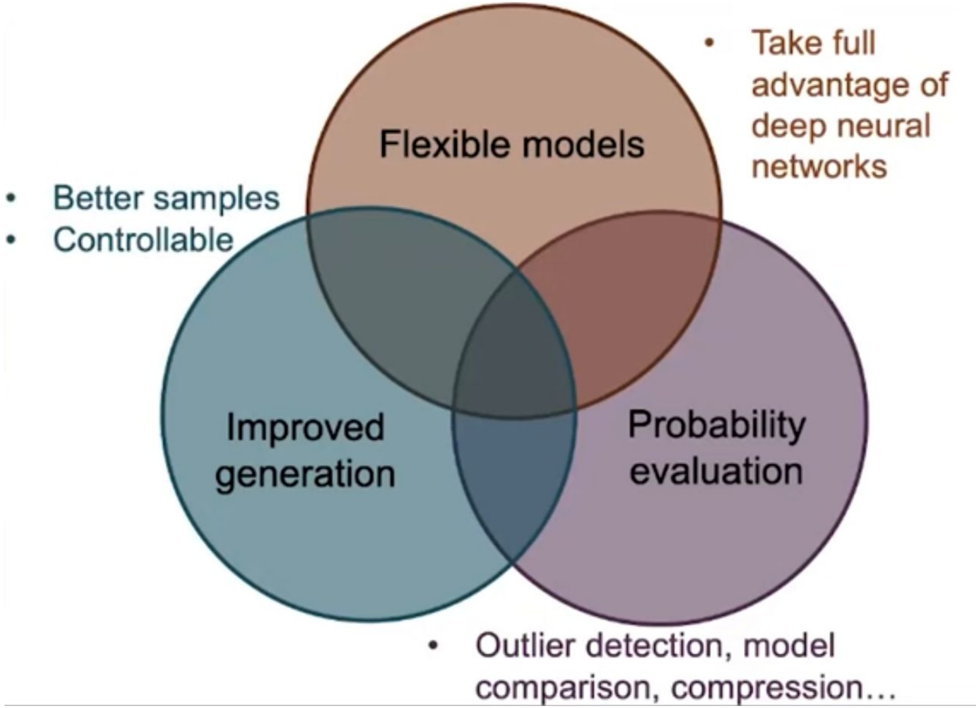
(1) Approximate normalizing constant
- ex) EBMs (Energy-Based Models)
- limitation: inaccurate probability evaluation

(2) Restricted NN
- ex) AR models, Flow-based models, VAE
- limitation: restricted model family

(3) Model only the generation process
- ex) GAN
- limitation: can not evaluate probabilities

1. Introduction

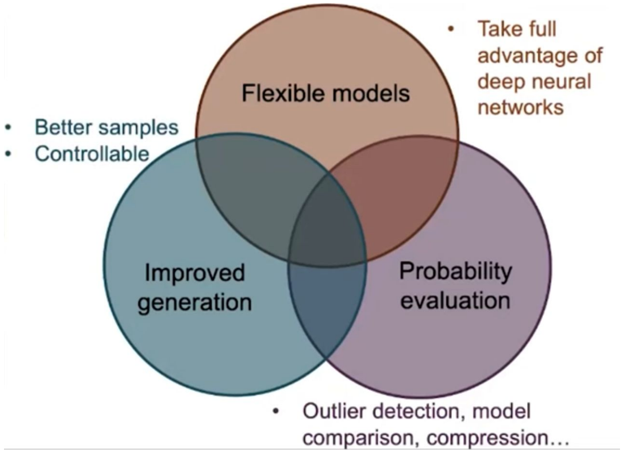
Desiderata of generative model

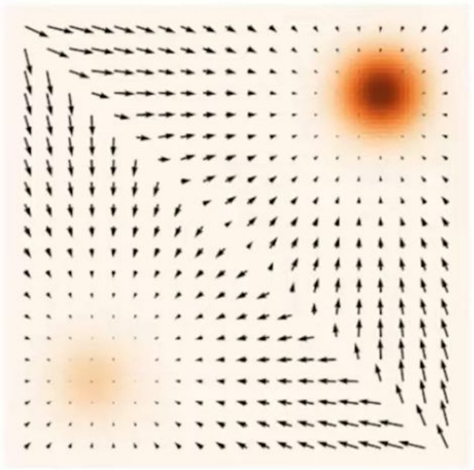


1. Introduction

$p(\mathbf{x})$: pdf

$\nabla_{\mathbf{x}} \log p(\mathbf{x})$: **(Stein) score function**





Score vs. density function

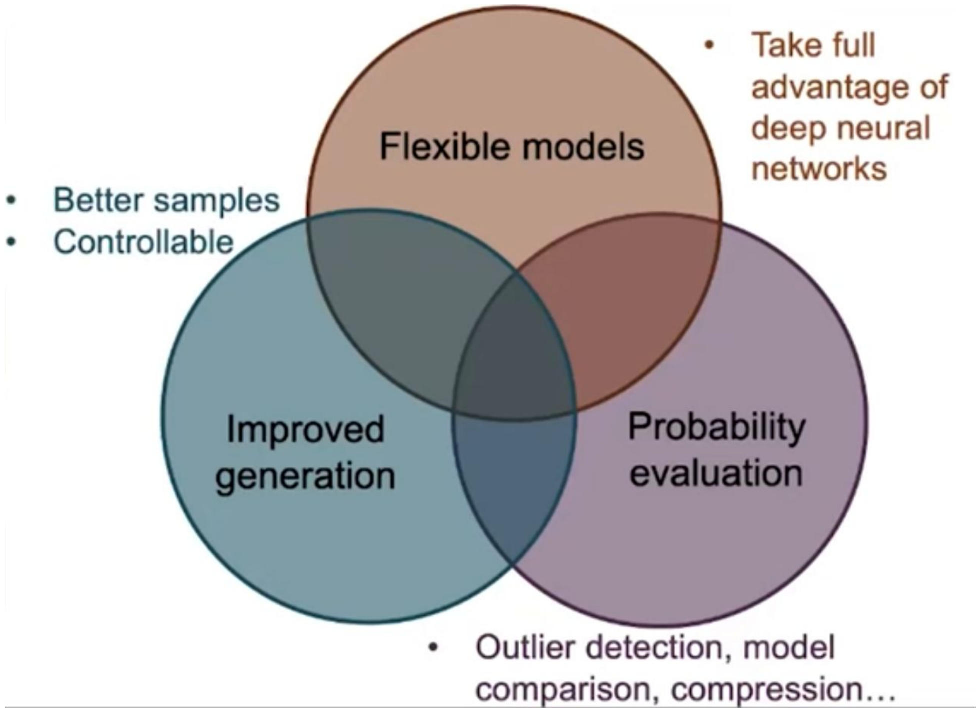
Score-based model satisfies the below!

1. Flexible Model
2. Improved Generation
3. Probability Evaluation

1. Introduction

Score-based model satisfies the below!

1. **Flexible Model**
   - Bypass the normalizing constant
   - Principled statistical methods
2. **Improved Generation**
   - Higher sample quality (vs. GANs)
   - Controllable generation
3. **Probability Evaluation**
   - Accurate probability evaluation
   - Estimate data probability well

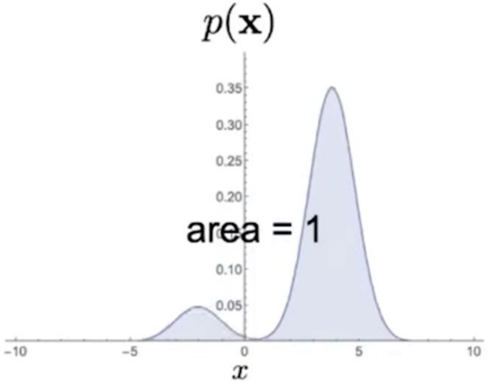


2. Score-based Models

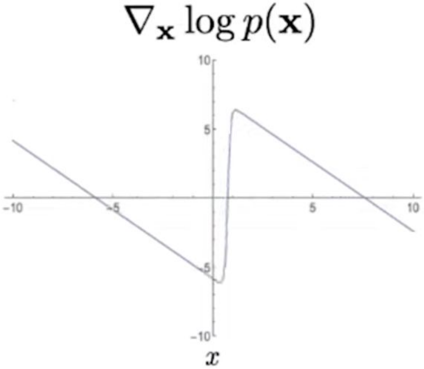
2-(a). Flexible Model

2-(a). Flexible Model

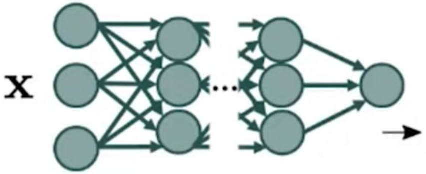
- Bypass the normalizing constant
- Principled statistical methods



Probability density function



Score function



$f_{\boldsymbol{\theta}}(\mathbf{x})$

$\rightarrow p_{\boldsymbol{\theta}}(\mathbf{x}) = \frac{e^{f_{\boldsymbol{\theta}}(\mathbf{x})}}{Z_{\boldsymbol{\theta}}}$

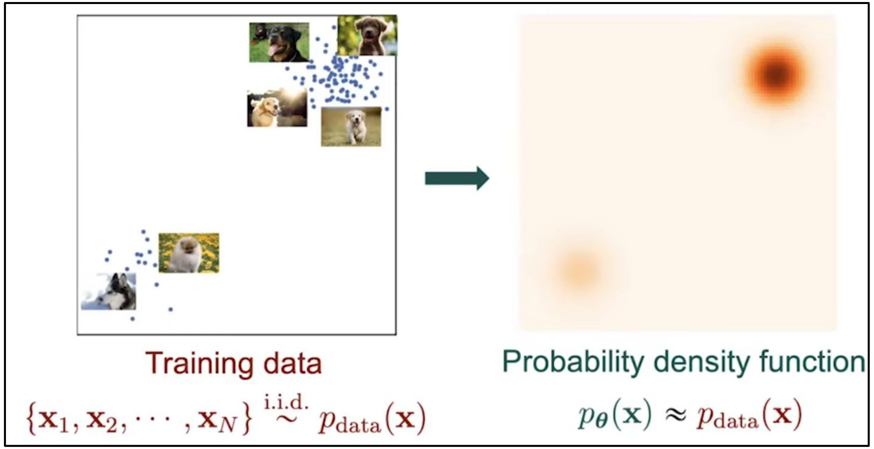
$$\nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x}) = \nabla_{\mathbf{x}} f_{\boldsymbol{\theta}}(\mathbf{x}) - \nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}}$$

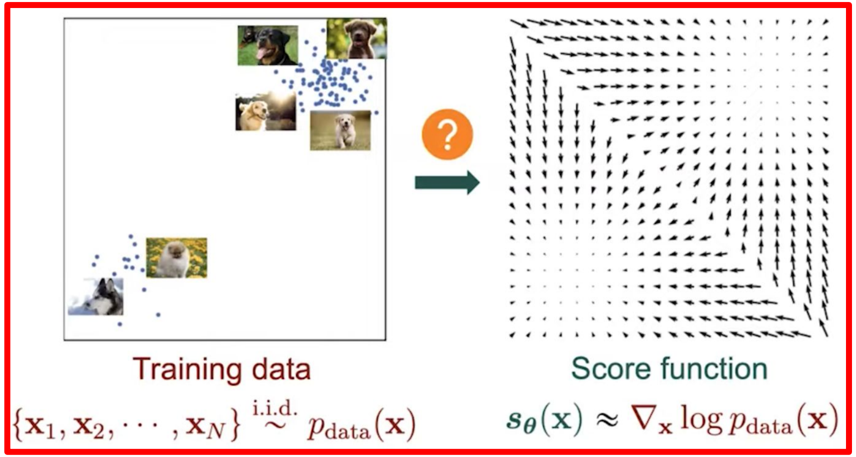
$$= \nabla_{\mathbf{x}} f_{\boldsymbol{\theta}}(\mathbf{x})$$

$\nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}} \rightarrow 0$

2-(a). Flexible Model

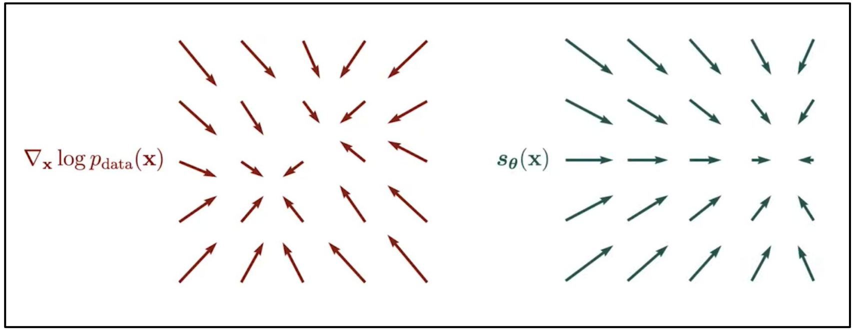
- Bypass the normalizing constant
- Principled statistical methods

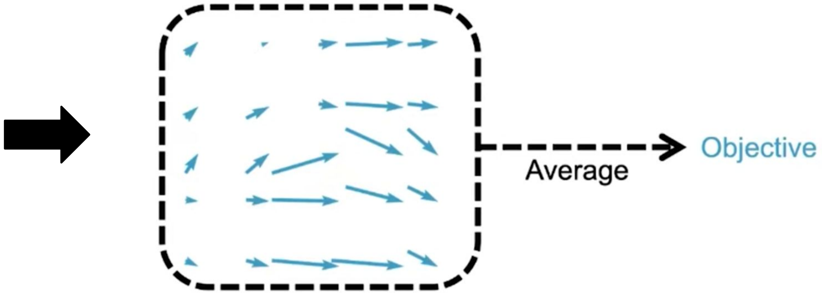




2-(a). Flexible Model

- Input: $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p_{\text{data}}(\mathbf{x})$
- Output: $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$
- Model: $s_\theta(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}^d \approx \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$





Loss Function

2-(a). Flexible Model

Loss Function

$$\frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) - s_{\boldsymbol{\theta}}(\mathbf{x})\|_2^2\right]$$ **Fisher Divergence**

But we don't know the target! How to solve?

1. Score Matching
2. Sliced Score Matching
3. Denoising Score Matching

2-(a). Flexible Model

Loss Function

(1) Score Matching

$$\frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}}\log p_{\text{data}}(\mathbf{x}) - s_{\boldsymbol{\theta}}(\mathbf{x})\|_2^2\right]$$ **Fisher Divergence**

$$\mathbb{E}_{p_{\text{data}}(\mathrm{x})}\left[\frac{1}{2}\|s_\theta(\mathrm{x})\|_2^2 + \text{trace}(\underbrace{\nabla_{\mathrm{x}} s_\theta(\mathrm{x})}_{\text{Jacobian of } s_\theta(\mathrm{x})})\right] \approx \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{2}\|s_\theta(\mathrm{x}_i)\|_2^2 + \text{trace}(\nabla_{\mathrm{x}} s_\theta(\mathrm{x}_i))\right]$$

Score Matching!

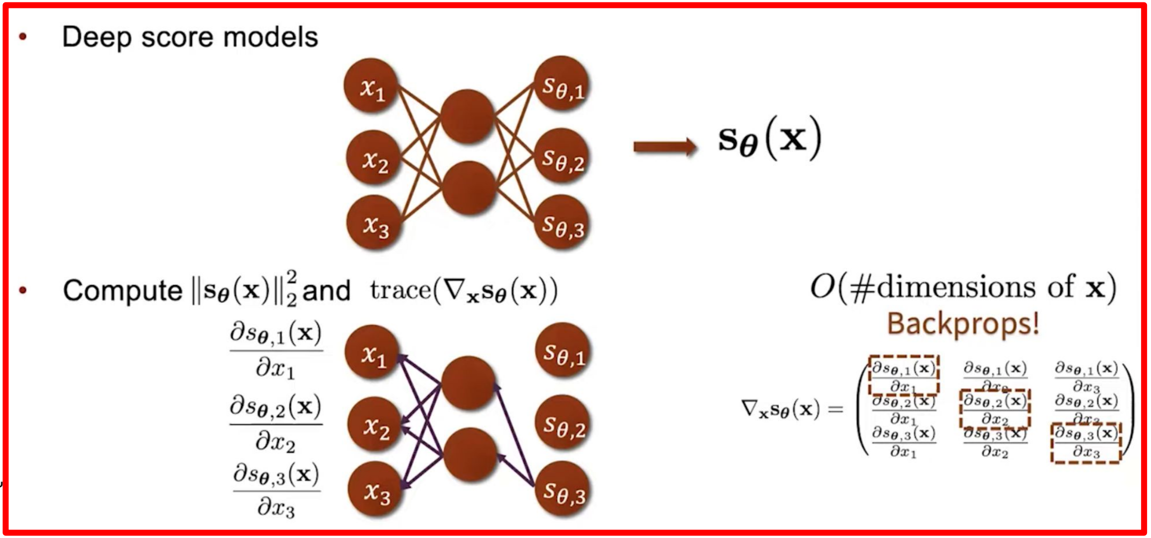
- Do not need the ground truth!
- Limitation: "**not scalable**"

2-(a). Flexible Model

Loss Function

(1) Score Matching

$$\frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}} \log p_{\text{data}}\right.$$



1

$$\mathbb{E}_{p_{\text{data}}(\mathrm{x})}\left[\frac{1}{2}\|s_\theta(\mathrm{x})\|_2^2 + \text{trace}(\underbrace{\nabla_{\mathrm{x}} s_\theta(\mathrm{x})}_{\text{Jacobian of } s_\theta(\mathrm{x})})\right] \approx \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{2}\|s_\theta(\mathrm{x}_i)\|_2^2 + \text{trace}(\nabla_{\mathrm{x}} s_\theta(\mathrm{x}_i))\right]$$

Score Matching!

- Do not need the ground truth!
- Limitation: "**not scalable**"

2-(a). Flexible Model

Loss Function

(2) Sliced Score Matching

$$\frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}}\log p_{\text{data}}(\mathbf{x}) - s_{\boldsymbol{\theta}}(\mathbf{x})\|_2^2\right]$$

Fisher Divergence

$$\frac{1}{2}\mathbb{E}_{p_{\mathbf{v}}}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\left(\mathbf{v}^\top\nabla_{\mathbf{x}}\log p_{\text{data}}(\mathbf{x}) - \mathbf{v}^\top\mathbf{S}_\theta(\mathbf{x})\right)^2\right]$$

Sliced Fisher Divergence

$$= \mathbb{E}_{p_{\mathbf{v}}}\mathbb{E}_{p_{\text{data}}(\mathrm{x})}\left[\mathbf{v}^\top\nabla_{\mathrm{x}}\mathrm{S}_\theta(\mathrm{x})\mathbf{v} + \frac{1}{2}\left(\mathbf{v}^\top\mathrm{S}_\theta(\mathrm{x})\right)^2\right]$$

(Integration by parts)

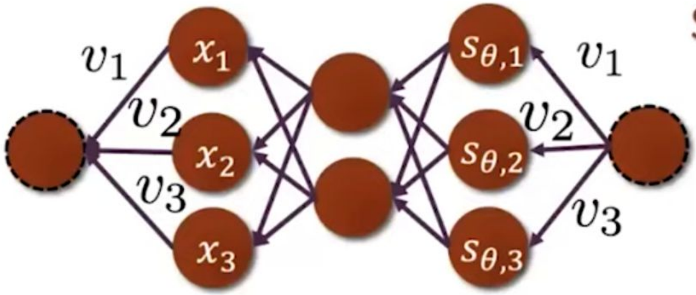
$$\mathbf{v}^\top\nabla_{\mathbf{x}}\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})\mathbf{v} = \mathbf{v}^\top\nabla_{\mathbf{x}}\left(\mathbf{v}^\top\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})\right)$$

Project onto random directions for scalability!

Computing Jacobian-vector products is scalable

$$\mathbf{v}^\top \nabla_{\mathbf{x}} \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})\mathbf{v} = \mathbf{v}^\top \nabla_{\mathbf{x}} (\mathbf{v}^\top \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}))$$

$$\mathbf{v}^\top \nabla_{\mathbf{x}} (\mathbf{v}^\top \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}))$$



One Backprop!

Sliced Score Matching is scalable

$$= \mathbb{E}_{p_{\mathbf{v}}} \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \left[\mathbf{v}^\top \nabla_{\mathbf{x}} \mathbf{s}_\theta(\mathbf{x}) \mathbf{v} + \frac{1}{2} \left(\mathbf{v}^\top \mathbf{s}_\theta(\mathbf{x}) \right)^2 \right]$$ (Integration by parts)

$$\mathbf{v}^\top \nabla_{\mathbf{x}} \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})\mathbf{v} = \mathbf{v}^\top \nabla_{\mathbf{x}} \left(\mathbf{v}^\top \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})\right)$$

Project onto random directions for scalability!

2-(a). Flexible Model

Denoising Score Matching!

- Match the score of a **noise-perturbed distribution**

Loss Function

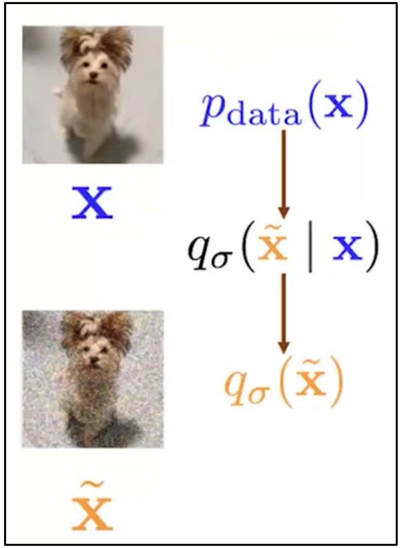
(3) Denoising Score Matching

$$\frac{1}{2}\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) - s_{\boldsymbol{\theta}}(\mathbf{x})\|_2^2\right]$$ **Fisher Divergence**

3

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

if Gaussian noise …
$$\frac{1}{2n}\sum_{i=1}^{n}\left[\left\| s_\theta(\tilde{\mathbf{x}}_i) + \frac{\tilde{\mathbf{x}}_i - \mathbf{x}_i}{\sigma^2}\right\|_2^2\right]$$

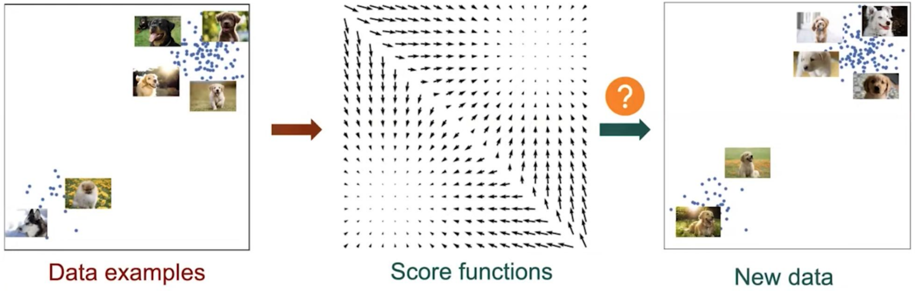


2-(b). Improved Generation

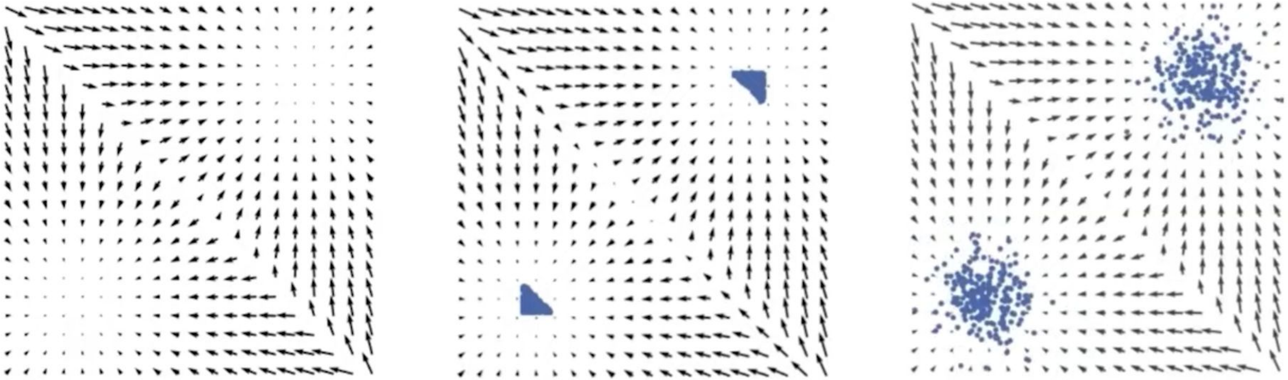
2-(b). Improved Generation

How to sample from score function?

- **Langevin Dynamics**



3



Score function

$\mathbf{s}_\theta(\mathbf{x})$

Follow the scores

$$\tilde{\mathbf{x}}_{t+1} \leftarrow \tilde{\mathbf{x}}_t + \frac{\epsilon}{2}\mathbf{s}_\theta(\tilde{\mathbf{x}}_t)$$

Follow the noisy scores

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

$$\tilde{\mathbf{x}}_{t+1} \leftarrow \tilde{\mathbf{x}}_t + \frac{\epsilon}{2}\mathbf{s}_\theta(\tilde{\mathbf{x}}_t) + \sqrt{\epsilon}\,\mathbf{z}_t$$

Add randomness

2-(b). Improved Generation

How to sample from score function?

- **Langevin Dynamics**

$$\text{Sample from } p(\mathbf{x}) \text{ using only the score } \nabla_{\mathbf{x}} \log p(\mathbf{x})$$

Step 1) Initialize $x^0 \sim \pi(x)$

Step 2) Repeat $t \leftarrow 1, 2, \cdots, T$

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

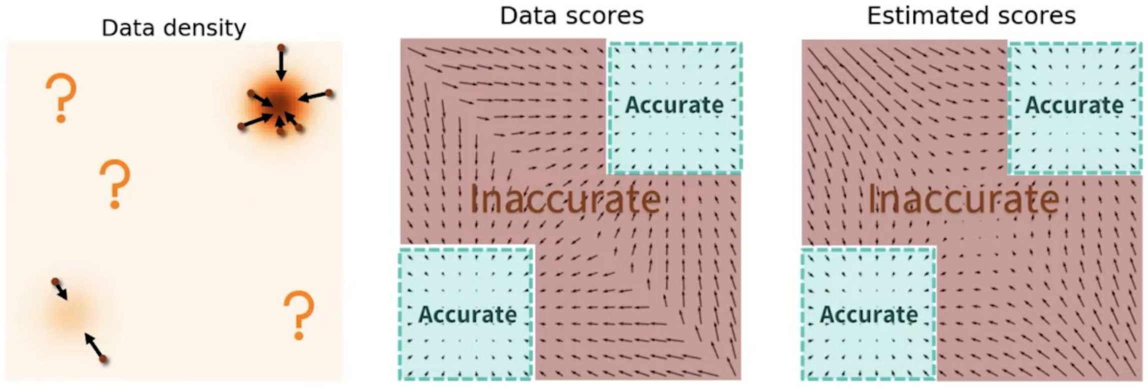
$$\mathbf{x}^t \leftarrow \mathbf{x}^{t-1} + \frac{\epsilon}{2} \nabla_{\mathbf{x}} \log p\left(\mathbf{x}^{t-1}\right) + \sqrt{\epsilon}\mathbf{z}^t$$

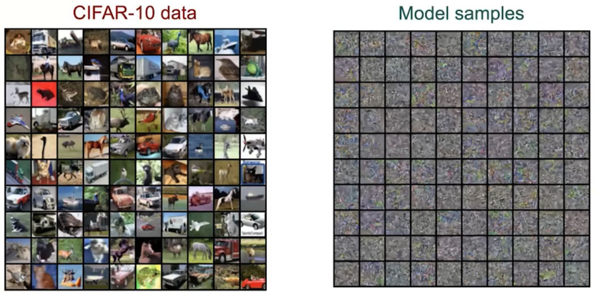
2-(b). Improved Generation

How to sample from score function?

- **Langevin Dynamics**

Limitations: Bad quality in **LOW score region**!

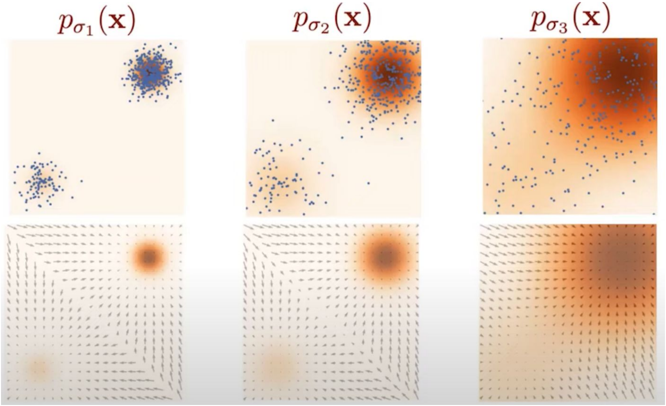




2-(b). Improved Generation

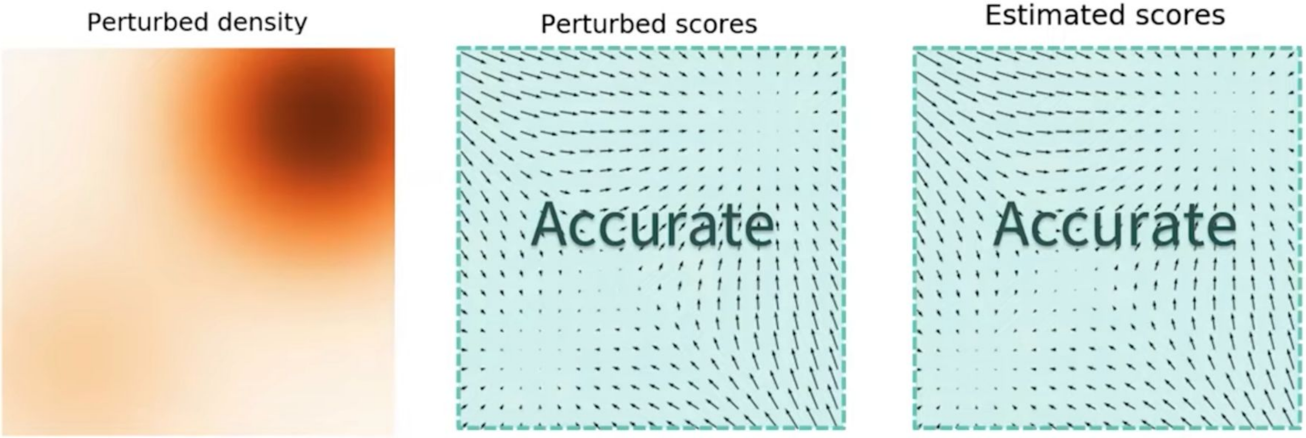
How to sample from score function?

- **Langevin Dynamics**



Solution: **ADD noise** to improve score estimation!

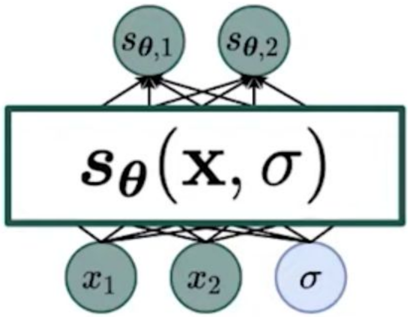
Multiple noise level



2-(b). Improved Generation

How to sample from score function?

- **Annealed Langevin Dynamics**



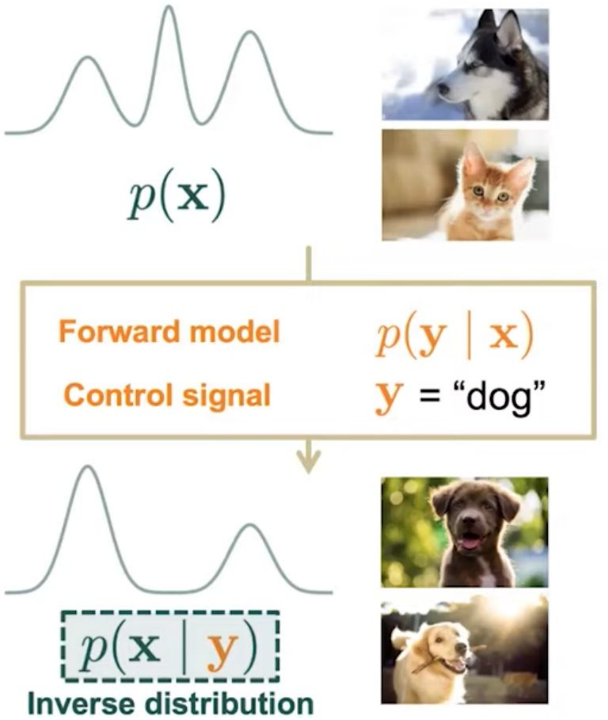
Noise Conditional Score Model

$$\frac{1}{N}\sum_{i=1}^{N}\overbrace{\lambda(\sigma_i)}^{\text{positive weighting function}}\underbrace{\mathbb{E}_{p_{\sigma_i}(\mathbf{x})}\left[\|\nabla_{\mathbf{x}}\log p_{\sigma_i}(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x},\sigma_i)\|_2^2\right]}_{\text{Score matching loss}}$$

- Generalization of DDPM loss

2-(b). Improved Generation

Controlling the generation process



Class-conditional generation

$$\nabla_{\mathbf{x}} \log p(\mathbf{x} \mid \mathbf{y}) = \nabla_{\mathbf{x}} \log p(\mathbf{x}) + \nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x}) - \nabla_{\mathbf{x}} \log p(\mathbf{y})$$
$$= \underbrace{\nabla_{\mathbf{x}} \log p(\mathbf{x})}_{\text{(Unconditional) Score}} + \underbrace{\nabla_{\mathbf{x}} \log p(\mathbf{y} \mid \mathbf{x})}_{\text{Forward Model (=classifier)}} - 0$$

(Unconditional) Score $\approx s_\theta(\mathbf{x})$

Forward Model (=classifier)

Applications

- Image inpainting
- Image colorization
- Stroke painting to image
- Language guided image generation

2-(c). Probability Evaluation

2-(c). Probability Evaluation

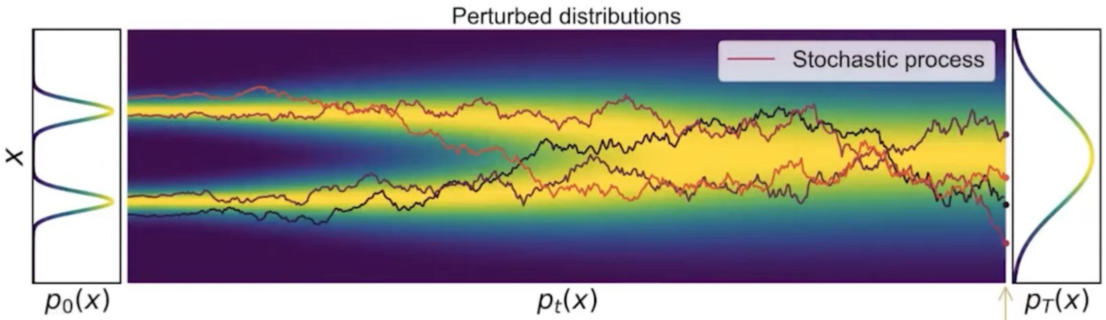
Perturbing Data (Forward)

(1) Stochastic Process

$$\{\mathbf{x}_t\}_{t\in[0,T]}$$

(2) Probability Densities

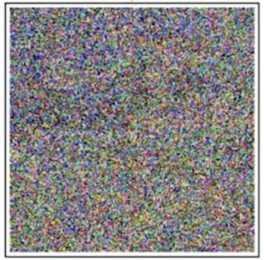
$$\{p_t(\mathbf{x})\}_{t\in[0,T]}$$



(3) **Stochastic Differential Equation (SDE)**

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

Deterministic Stochastic (small noise)



$p_T(\mathbf{x}) \approx \pi(\mathbf{x})$

2-(c). Probability Evaluation

Perturbi

(1) Stoch

(2) Proba

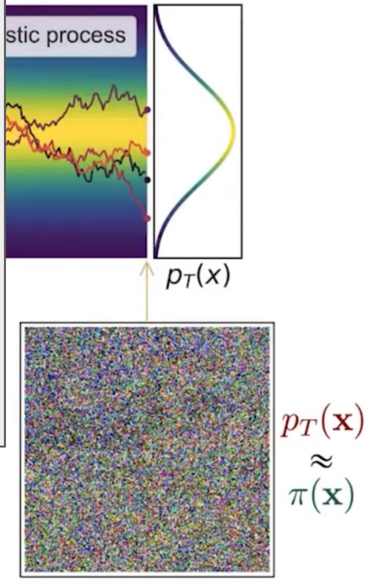
$\{p$

(3) **Stoch**

Generate via **REVERSE** process !!

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

Deterministic | Stochastic (small noise)

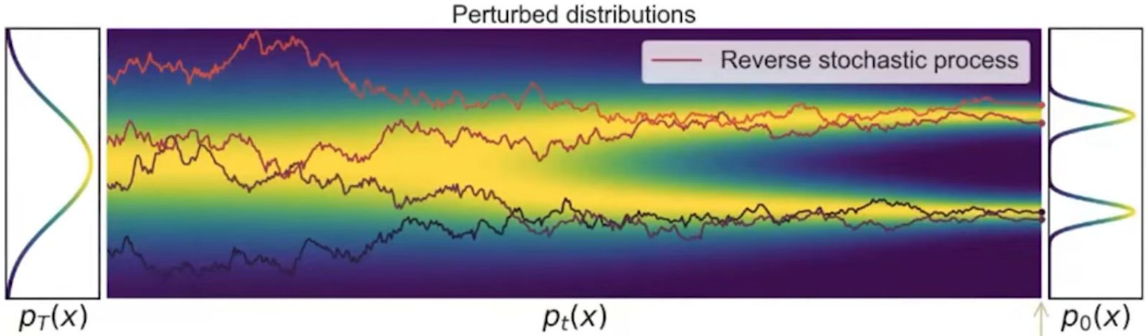


2-(c). Probability Evaluation

Generating Data (Reverse)

(1) Forward SDE (0->T)

$$\mathrm{d}\mathbf{x}_t = \sigma(t)\mathrm{d}\mathbf{w}_t$$



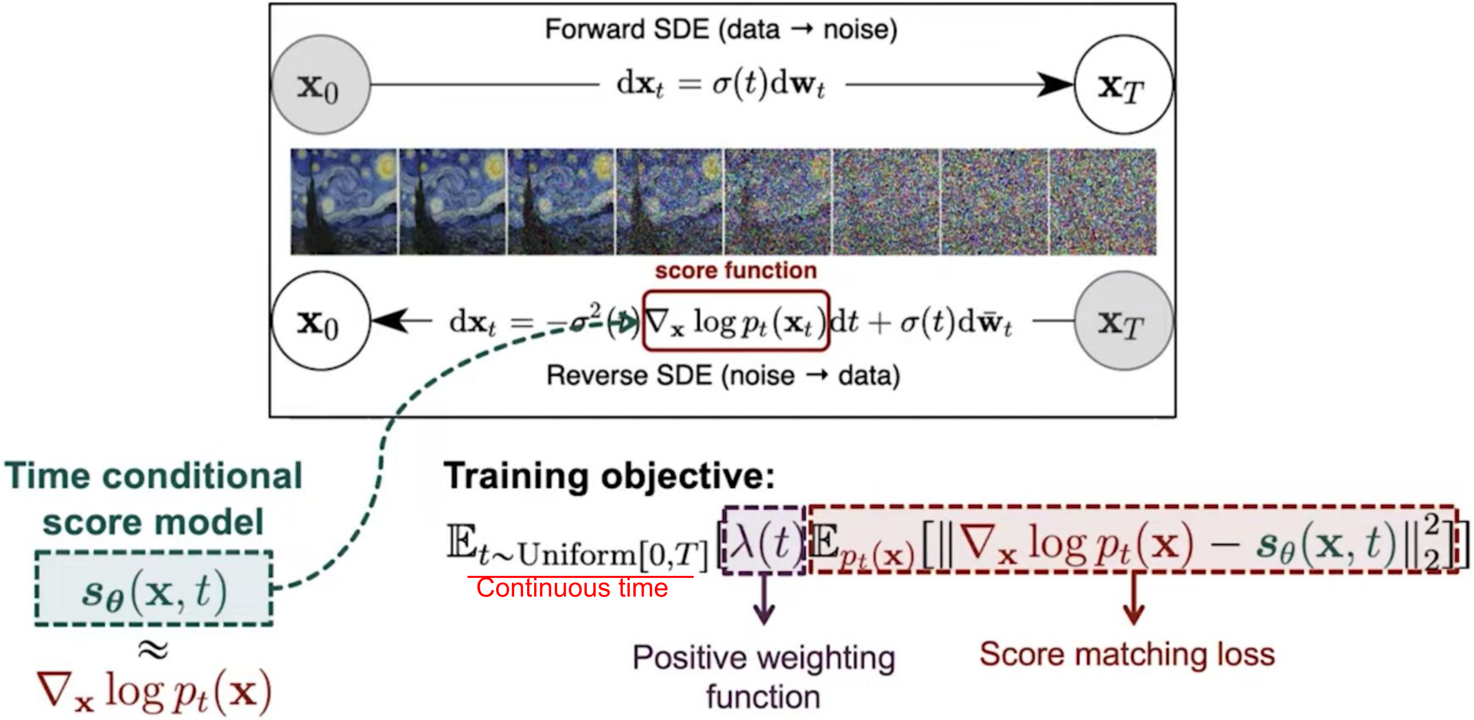
(2) Backward SDE (0->T)

$$\mathrm{d}\mathbf{x}_t = -\sigma(t)^2 \nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t)\mathrm{d}t + \sigma(t)\mathrm{d}\overline{\mathbf{w}}_t$$

small noise during backward

2-(c). Probability Evaluation

Score-based Generative Modeling via SDEs



2-(c). Probability Evaluation

Score-based Generative Modeling via SDEs

(1) Model (= Time-dependent score model)

$$\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, t) \approx \nabla_{\mathbf{x}} \log p_t(\mathbf{x})$$

(2) Training: Loss Function

$$\mathbb{E}_{t \in \mathcal{U}(0,T)} \left[\lambda(t) \mathbb{E}_{p_t(\mathbf{x})} \left[\| \nabla_{\mathbf{x}} \log p_t(\mathbf{x}) - \mathbf{s}_\theta(\mathbf{x}, t) \|_2^2 \right] \right]$$

(3) Sampling: Reverse-time SDE

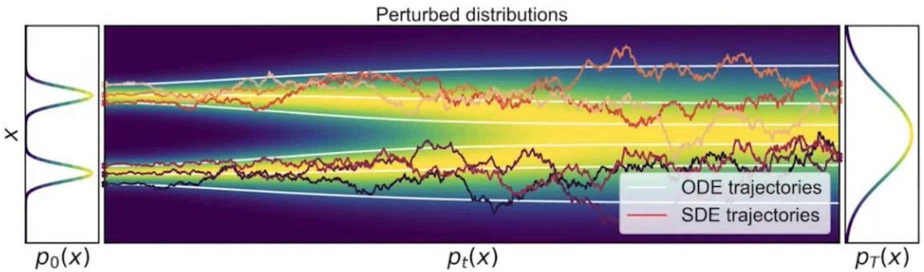
$$\mathrm{d}\mathbf{x} = -\sigma^2(t) \mathrm{s}_\theta(\mathbf{x}, t) \mathrm{d}t + \sigma(t) \mathrm{d}\bar{\mathbf{w}}$$

(3) Sampling: Euler-Maruyama

$$\mathbf{x} \leftarrow \mathbf{x} - \sigma(t)^2 \, \mathrm{s}_\theta(\mathbf{x}, t) \Delta t + \sigma(t) \mathbf{z} \quad (\mathbf{z} \sim \mathcal{N}(\mathbf{0}, |\Delta t| \mathbf{I}))$$

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

2-(c). Probability Evaluation



SDE -> ODE

- To evaluate the probability!

(1) SDE

$$\mathrm{d}\mathbf{x}_t = \sigma(t)\mathrm{d}\mathbf{w}_t$$

(2) ODE

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = -\frac{1}{2}\sigma(t)^2 \;\nabla_{\mathbf{x}} \log p_t(\mathbf{x}_t)$$

Compute the exact likelihood with ODEs

$$\log p_{\boldsymbol{\theta}}(\mathbf{x}_0) = \log \pi(\mathbf{x}_T) - \frac{1}{2}\int_0^T \sigma(t)^2 \operatorname{trace}(\nabla_{\mathbf{x}} s_{\boldsymbol{\theta}}(\mathbf{x}, t))\mathrm{d}t$$

References

- https://www.youtube.com/watch?v=wMmqCMwuM2Q