Deep Generative Models

14. Denoising Diffusion Probabilistic Models

- 국가수리과학연구소 산업수학혁신센터 김민중

Recap. of score-based model

- **Fisher divergence** between $p(\boldsymbol{x})$ and $q(\boldsymbol{x})$:

$$D_F(p,q) := \frac{1}{2}E_{\boldsymbol{x}\sim p}[\ \|\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}) - \nabla_{\boldsymbol{x}} \log q(\boldsymbol{x})\|_2^2]$$

- Score matching(Hyvärinen, 2005)

$$\frac{1}{2}E_{\boldsymbol{x}\sim p_{data}}[\ \|\boldsymbol{s}_\theta(\boldsymbol{x}) - \nabla_{\boldsymbol{x}} \log p_{data}(\boldsymbol{x})\|_2^2]$$
$$= E_{\boldsymbol{x}\sim p_{data}}\left[\frac{1}{2}\|\boldsymbol{s}_\theta(\boldsymbol{x})\|_2^2 + tr\big(\nabla_{\boldsymbol{x}}\boldsymbol{s}_\theta(\boldsymbol{x})\big)\right] + const.$$

- Not scalable for deep score-based models and high dimensional data

Denoising score matching with Langevin dynamics

$\boldsymbol{x} \sim p_{data}(\boldsymbol{x})$
Data distribution

$q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})$

Perturbation distribution/kernel

$\tilde{\boldsymbol{x}} \sim q_\sigma(\tilde{\boldsymbol{x}})$
Noise-perturbed data distribution

$$E_{\tilde{\boldsymbol{x}} \sim q_\sigma}\left[\, \|\nabla_{\tilde{\boldsymbol{x}}} \log q_\sigma(\tilde{\boldsymbol{x}}) - \boldsymbol{s}_\theta(\tilde{\boldsymbol{x}})\|_2^2\right]$$
$$= E_{\boldsymbol{x} \sim p_{data}(\boldsymbol{x})} E_{\tilde{\boldsymbol{x}} \sim q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x})}\left[\|\nabla_{\tilde{\boldsymbol{x}}} \log q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) - \boldsymbol{s}_\theta(\tilde{\boldsymbol{x}})\|_2^2\right] + \text{const.}$$
$$= E_{\boldsymbol{x} \sim p_{data}(\boldsymbol{x})} E_{\boldsymbol{z} \sim N(\boldsymbol{0}, \boldsymbol{I})}\left[\left\|\frac{1}{\sigma}\boldsymbol{z} + \boldsymbol{s}_\theta(\boldsymbol{x} + \sigma \boldsymbol{z})\right\|_2^2\right] + \text{const.}$$

- **Pros**
  - more scalable than score matching
  - reduces score estimation to a denoising task
- **Con**: cannot estimate the score of clean data (noise-free)

$$\boldsymbol{s}_\theta(\boldsymbol{x}) \approx \nabla_{\boldsymbol{x}} \log q_\sigma(\boldsymbol{x}) \neq \nabla_{\boldsymbol{x}} \log p_{data}(\boldsymbol{x})$$

Denoising score matching with Langevin dynamics

- Let $q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \coloneqq N(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \sigma^2\boldsymbol{I})$, $q_\sigma(\tilde{\boldsymbol{x}}) \coloneqq \int p_{data}(\boldsymbol{x}) q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) d\boldsymbol{x}$
- The objective

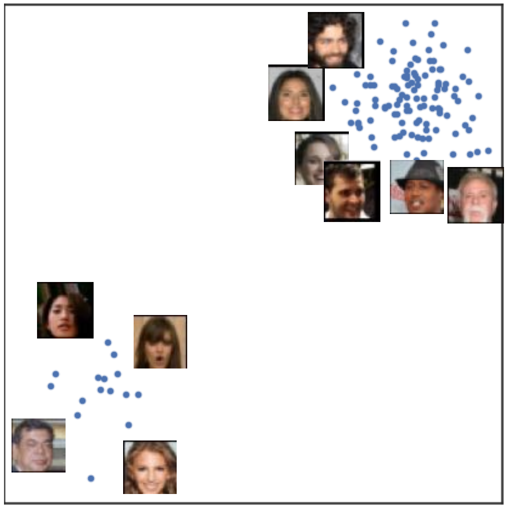
$$E_{\tilde{\boldsymbol{x}} \sim q_\sigma(\tilde{\boldsymbol{x}})}[\,\|\boldsymbol{s}_\theta(\tilde{\boldsymbol{x}}) - \nabla_{\tilde{\boldsymbol{x}}} \log q_\sigma(\tilde{\boldsymbol{x}})\|_2^2]$$
$$= E_{\boldsymbol{x} \sim p_{data}(\boldsymbol{x})} E_{\tilde{\boldsymbol{x}} \sim q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x})}[\,\|\boldsymbol{s}_\theta(\tilde{\boldsymbol{x}}) - \nabla_{\tilde{\boldsymbol{x}}} \log q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x})\|_2^2] + const.$$

- Consider a sequence of positive noise scales

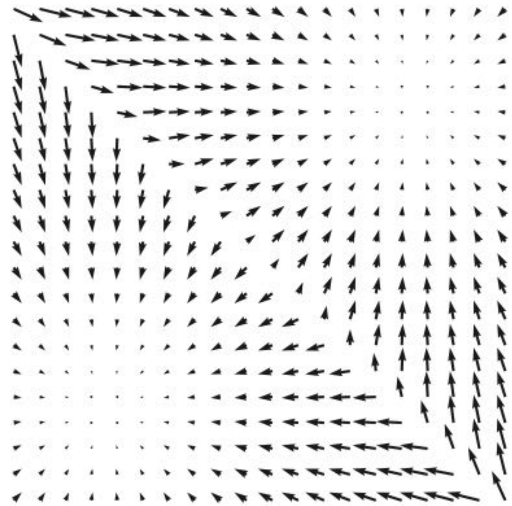
$$\sigma_1 < \sigma_2 < \cdots < \sigma_L$$

- σ_1 is small enough $q_{\sigma_1}(\boldsymbol{x}) \approx p_{data}(\boldsymbol{x})$
- σ_L is large enough $q_{\sigma_L}(\boldsymbol{x}) \approx N(\boldsymbol{x}|\boldsymbol{0}, \sigma_L^2\boldsymbol{I})$

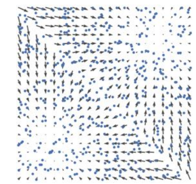
Score-based generative modeling



Score Matching



Langevin dynamics



Data samples

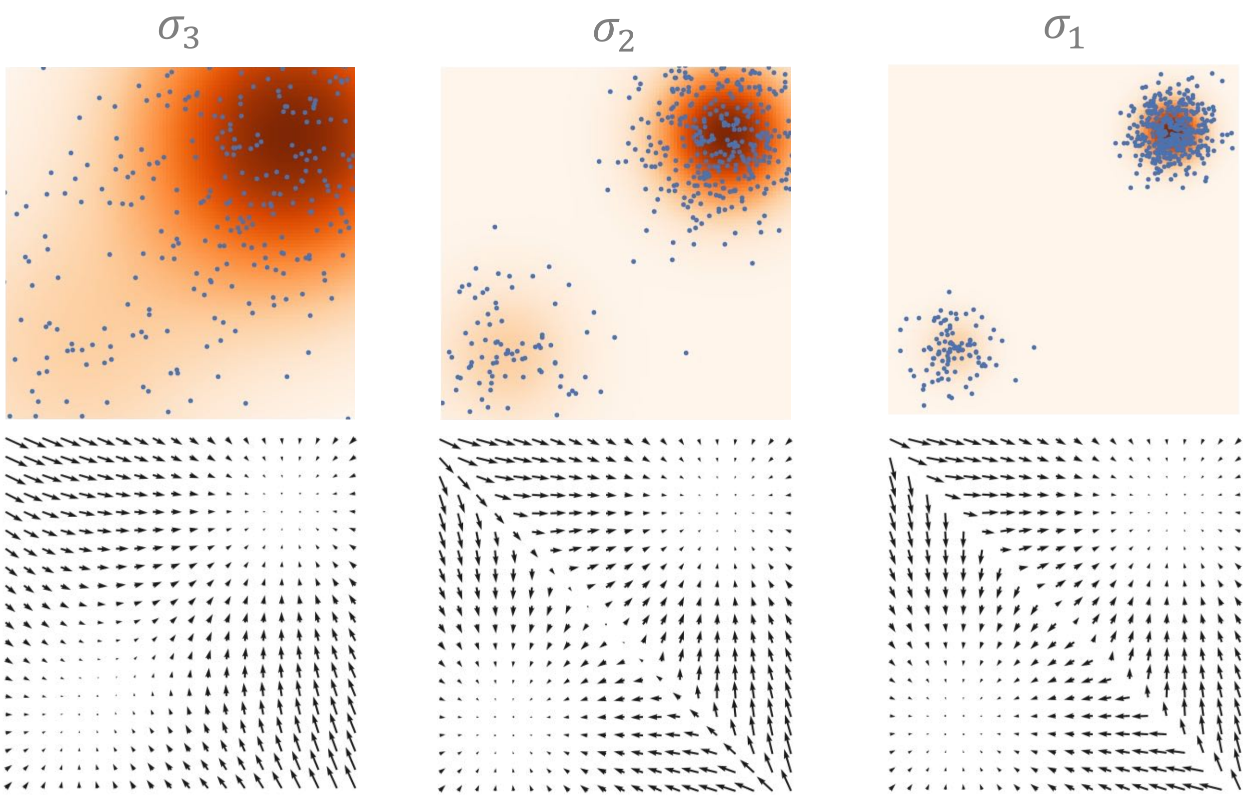
$\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}$

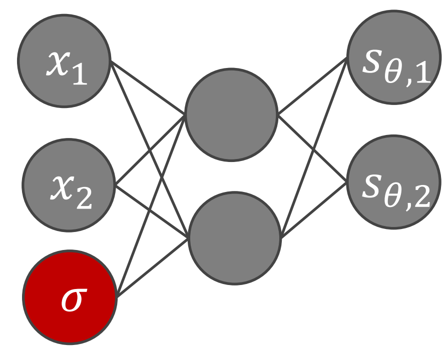
Scores function

$\boldsymbol{s}_\theta(\boldsymbol{x}) \approx \nabla_{\boldsymbol{x}} \log p_{data}(\boldsymbol{x})$

New samples

Joint score estimation via noise conditional score networks





Noise Conditional Score Network (NCSN)

Denoising score matching with Langevin dynamics

- For each $q_{\sigma_i}(\boldsymbol{x})$ with $\sigma_1 < \sigma_2 < \cdots < \sigma_L$, Song & Ermond run T steps of Langevin MCMC to get a sample sequentially

$$\boldsymbol{x}_i^t \coloneqq \boldsymbol{x}_i^{t-1} + \frac{\alpha_i}{2}\boldsymbol{s}_{\theta^*}\left(\boldsymbol{x}_i^{t-1}, \sigma_i\right) + \sqrt{\alpha_i}\boldsymbol{z}, \qquad t = 1,2,\ldots,T$$

- where $\alpha_i > 0$ is the step size and $\boldsymbol{z} \sim N(\boldsymbol{0}, \boldsymbol{I})$

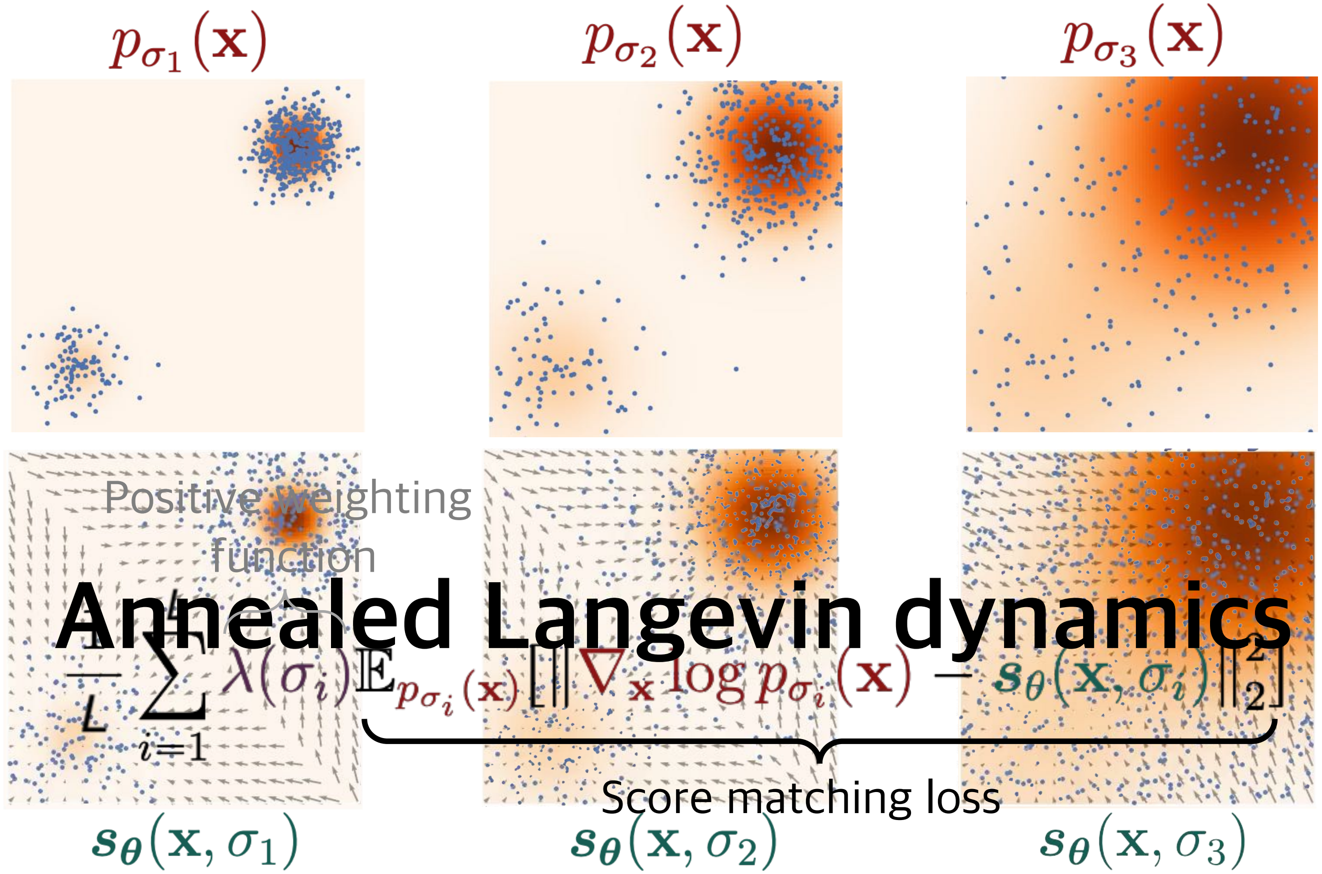
$$\alpha_i \coloneqq \epsilon \frac{\sigma_i^2}{\sigma_1^2}$$

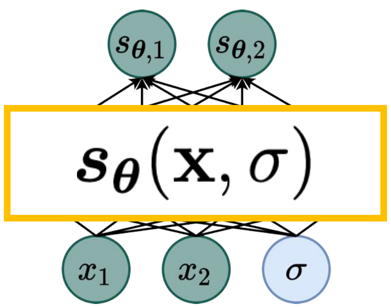
- $\epsilon > 0$

Generative Modeling by Estimating Gradients of the Data Distribution
Song Yang, and Stefano Ermon. NeurIPS 2019

Using multiple noise levels

Data





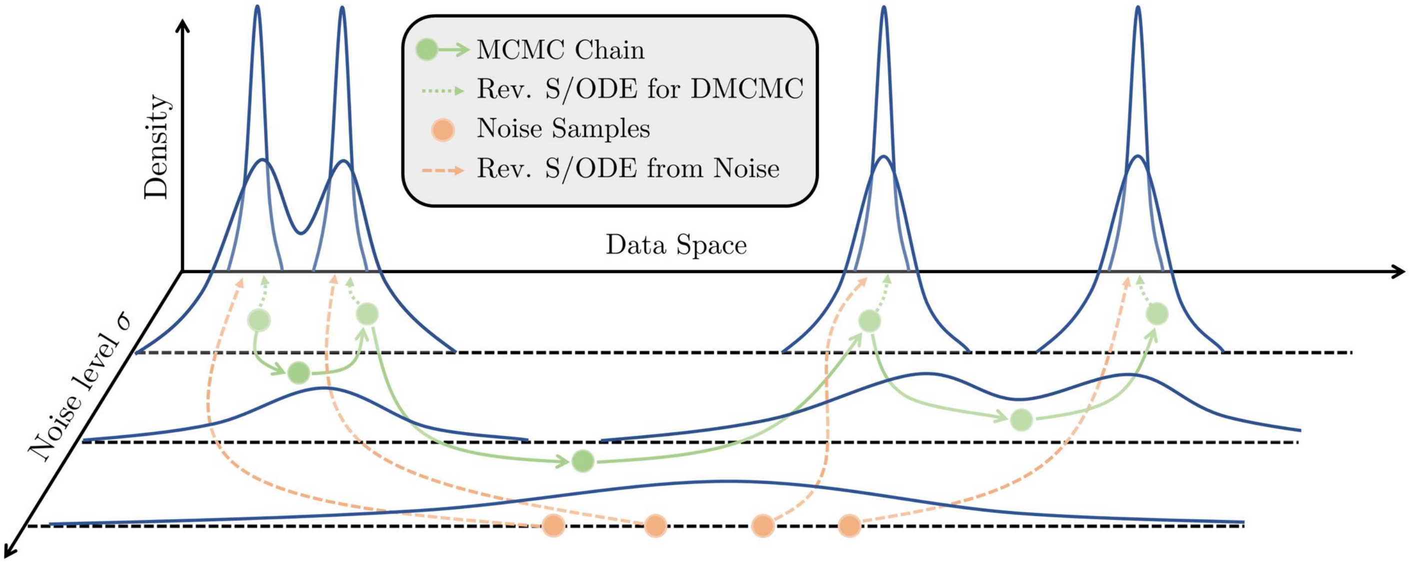
Noise Conditiona
l
Score Model

Positive weighting function

Annealed Langevin dynamics

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

Denoising score matching with Langevin dynamics



Conceptual illustration of a multiple noise score matching with Langevin sampling process

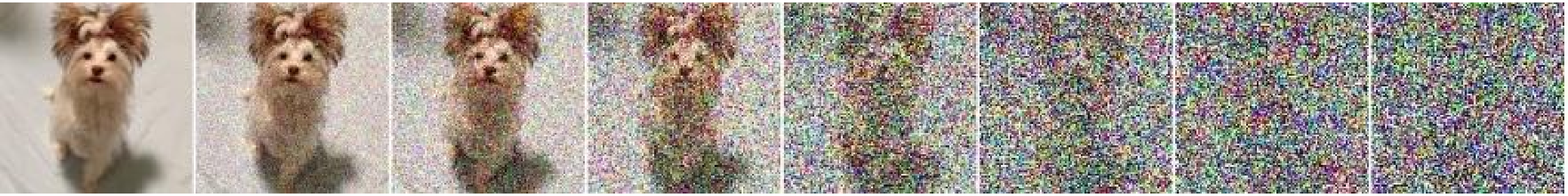
DENOISING MCMC FOR ACCELERATING DIFFUSION-BASED GENERATIVE MODELS
Beomsu Kim, Jong Chul Ye. ICML 2023

Denoising score matching with Langevin dynamics

- Let $q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \coloneqq N(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \sigma^2 I)$, $q_\sigma(\tilde{\boldsymbol{x}}) \coloneqq \int p_{data}(\boldsymbol{x}) q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) d\boldsymbol{x}$
- Consider a sequence of positive noise scales

$$\sigma_1 < \sigma_2 < \cdots < \sigma_L$$

- σ_1 is small enough $q_{\sigma_1}(\boldsymbol{x}) \approx p_{data}(\boldsymbol{x})$
- σ_L is large enough $q_{\sigma_L}(\boldsymbol{x}) \approx N(\boldsymbol{x}|\boldsymbol{0}, \sigma_L^2 \boldsymbol{I})$



Denoising score matching with Langevin dynamics

- Let $q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \coloneqq N(\tilde{\boldsymbol{x}}|\boldsymbol{x}, \sigma^2 I)$, $q_\sigma(\tilde{\boldsymbol{x}}) \coloneqq \int p_{data}(\boldsymbol{x}) q_\sigma(\tilde{\boldsymbol{x}}|\boldsymbol{x}) d\boldsymbol{x}$
- Consider a sequence of positive noise scales

$$\sigma_1 < \sigma_2 < \cdots < \sigma_L$$

- **Noise conditional score network**

$$\sum_{i=1}^{L} \sigma_i^2 \, E_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})} E_{\tilde{\boldsymbol{x}}\sim q_{\sigma_i}(\tilde{\boldsymbol{x}}|\boldsymbol{x})} \left[\left\| \boldsymbol{s}_\theta(\tilde{\boldsymbol{x}}, \sigma_i) - \nabla_{\tilde{\boldsymbol{x}}} \log q_{\sigma_i}(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \right\|_2^2 \right]$$

- Given sufficient data and model capacity, the optimal score-based model

$$\boldsymbol{s}_{\theta^*}(\boldsymbol{x}, \sigma_i) \approx \nabla_{\boldsymbol{x}} \log q_{\sigma_i}(\boldsymbol{x}) \quad \text{for } \sigma \in \{\sigma_1, \dots, \sigma_L\}$$

- The weights σ_i^2 are related to $\sigma_i^2 \propto 1/E\left[\left\| \nabla_{\tilde{\boldsymbol{x}}} \log p_{\sigma_i}(\tilde{\boldsymbol{x}}|\boldsymbol{x}) \right\|_2^2 \right]$

Generation with annealed Langevin dynamics

- For each $q_{\sigma_i}(\boldsymbol{x})$ with $\sigma_1 < \sigma_2 < \cdots < \sigma_L$, Song & Ermond run T steps of Langevin MCMC to get a sample sequentially

$$\boldsymbol{x}_i^t := \boldsymbol{x}_i^{t-1} + \frac{\alpha_i}{2}\boldsymbol{s}_{\theta^*}\left(\boldsymbol{x}_i^{t-1}, \sigma_i\right) + \sqrt{\alpha_i}\boldsymbol{z}, \qquad t = 1,2,\ldots,T$$

- where $\alpha_i > 0$ is the step size and $\boldsymbol{z} \sim N(\boldsymbol{0}, \boldsymbol{I})$

$$\alpha_i := \epsilon \frac{\sigma_i^2}{\sigma_1^2}$$

- $\epsilon > 0$

Generative Modeling by Estimating Gradients of the Data Distribution
Song Yang, and Stefano Ermon. NeurIPS 2019

Denoising diffusion probabilistic models(DDPM)

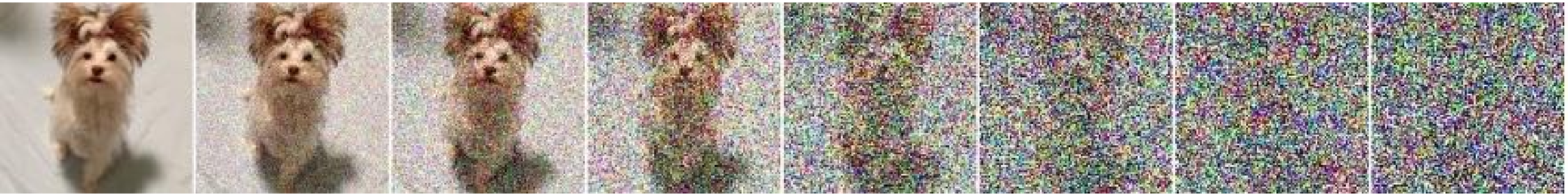
- Consider a seq. of positive noise scales $0 < \beta_1 < \beta_2 \cdots < \beta_T < 1$
- $\boldsymbol{x}_0 \sim p_{data}(\boldsymbol{x})$, construct latent variables $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_T\}$ s.t.

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \coloneqq \boldsymbol{N}(\boldsymbol{x}_t|\sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I})$$

- I.e., $q(\boldsymbol{x}_t|\boldsymbol{x}_0) = \boldsymbol{N}(\boldsymbol{x}_0|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\boldsymbol{I})$ where $\alpha_t \coloneqq 1-\beta_t$, $\bar{\alpha}_t \coloneqq \prod_{s=1}^{t} \alpha_s$
- Similar to SMLD, we can denote the perturbed data distribution

$$q(\boldsymbol{x}_t) \coloneqq \int q(\boldsymbol{x}_t|\boldsymbol{x}) p_{data}(\boldsymbol{x}) d\boldsymbol{x}$$

- The noise scales are prescribed s.t. $\boldsymbol{x}_T \sim q(\boldsymbol{x}_T) \approx N(\boldsymbol{0}, \boldsymbol{I})$



p_{data} $q(\boldsymbol{x}_1)$ $q(\boldsymbol{x}_2)$ $\cdots$ $q(\boldsymbol{x}_T)$

Denoising diffusion probabilistic models(DDPM)

- A variational Markov chain in the reverse direction is parametrized with

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t \boldsymbol{I})$$

- where $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + \beta_t \boldsymbol{s}_\theta(\boldsymbol{x}_t, t)\right)$
- Re-weighted variant of the evidence lower bound

$$\sum_{t=1}^{T}(1-\bar{\alpha}_t)\, E_{\boldsymbol{x}\sim p_{data}(\boldsymbol{x})} E_{\boldsymbol{x}_t\sim q(\boldsymbol{x}_t|\boldsymbol{x})}\left[\left\|\boldsymbol{s}_\theta(\boldsymbol{x}_t, t) - \nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t|\boldsymbol{x})\right\|_2^2\right]$$

- which is a weighted sum of denoising score matching

$$\boldsymbol{s}_{\theta^*}(\boldsymbol{x}_t, t) \approx \nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t)$$

- The weights $(1-\bar{\alpha}_t)$ are related to

$$(1-\bar{\alpha}_t) \propto 1/E\left[\left\|\nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t|\boldsymbol{x})\right\|_2^2\right]$$

Denoising diffusion probabilistic models(DDPM)

- Generate samples by starting from $\boldsymbol{x}_T \sim N(\mathbf{0}, \boldsymbol{I})$
- $\boldsymbol{x}_{t-1} \coloneqq \underbrace{\frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + \beta_t \boldsymbol{s}_{\theta^*}(\boldsymbol{x}_t, t)\right)}_{=\boldsymbol{\mu}_{\theta^*}(\boldsymbol{x}_t, t)} + \sqrt{\beta_t}\boldsymbol{z},\ t = T, T-1, \dots, 2$
- We call this method **ancestral sampling** ($\prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$)

Denoising Diffusion Probabilistic Models
Jonathan Ho, Ajay Jain, Pieter Abbeel. NeurIPS 2020

Summary of score-based models

- **SMLD** and **DDPM** involve sequentially corrupting training data with slowly increasing noise, and then learning to reverse this corruption to form a generative model of the data
- **SMLD** estimates the score at each noise scale and then use Langevin dynamics to sample from a sequence of decreasing noise scales during generation
- **DDPM** trains a sequence of probabilistic models to reverse each step of the noise corruption, using knowledge of the functional form of the reverse distributions to make training tractable

Recap: Latent Variable Models

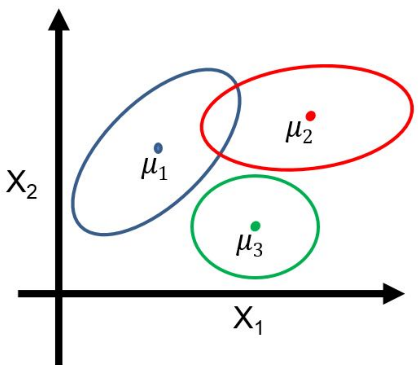
- Observable variables $\boldsymbol{x} \in \mathbb{R}^d$
- Latent variables $\boldsymbol{z} \in \mathbb{R}^h$ (unobservable)

$$p_{data}(\boldsymbol{x}) = \sum_{\boldsymbol{z}} p(\boldsymbol{x}, \boldsymbol{z})$$

$$or \qquad = \int p(\boldsymbol{x}, \boldsymbol{z}) d\boldsymbol{z}$$

Recap: Mixture of Gaussians

- Mixture of Gaussians. Bayes net: $z \to \boldsymbol{x}$
  - $z = Categorical(z|\gamma_1, \cdots, \gamma_K)$
  - $p(\boldsymbol{x}|z = k) = N(\boldsymbol{x}|\mu_k, \Sigma_k)$



- Generative Process
  - Pick a mixture component k by sampling z
  - Generate a data point by sampling from that Gaussian

Denoising diffusion probabilistic models(DDPM)

- DDPM is a latent variable model

$$p_\theta(\boldsymbol{x}_0) := \int p_\theta(\boldsymbol{x}_0, \boldsymbol{x}_1 \dots, \boldsymbol{x}_T) d\boldsymbol{x}_{1:T}$$

- $\boldsymbol{x}_0 = q(\boldsymbol{x}_0) = p_{data}$
- The joint distribution $p_\theta(\boldsymbol{x}_{0:T})$ is called the **reverse process** starting at $p_\theta(\boldsymbol{x}_T) = N(\boldsymbol{x}_T | \boldsymbol{0}, \boldsymbol{I})$

$$p_\theta(\boldsymbol{x}_{0:T}) = p_\theta(\boldsymbol{x}_T) \prod_{t=1}^{T} p_\theta(\boldsymbol{x}_{t-1} | \boldsymbol{x}_t),$$

$$p_\theta(\boldsymbol{x}_{t-1} | \boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1} | \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t))$$

Denoising diffusion probabilistic models(DDPM)

- **Forward process** or diffusion process $q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)$ is fixed to a Markov chain that gradually adds Gaussian noise to the data according to a variance schedule $0 < \beta_1 < \cdots < \beta_T < 1$

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) = \prod_{t=1}^{T} q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}),$$

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) \coloneqq N\left(\boldsymbol{x}_t \middle| \sqrt{1-\beta_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right)$$

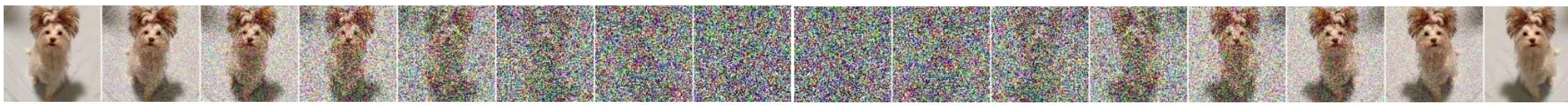
- β_t can be learned by reparameterization or held constants as hyperparameters
- Let $\alpha_t \coloneqq 1 - \beta_t$ and $\bar{\alpha}_t \coloneqq \prod_{s=1}^{t} \alpha_s$, then

$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t \middle| \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Denoising diffusion probabilistic models(DDPM)

Forward Markov chain $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ | Reverse Markov chain $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$

p_{data} | p_{data}



$\boldsymbol{x}_0$ $\boldsymbol{x}_1$ $\cdots$ $\boldsymbol{x}_T \sim N(\boldsymbol{0}, \boldsymbol{I})$ $\cdots$ $\boldsymbol{x}_1$ $\boldsymbol{x}_0$

Image to noise(prescribed)

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = N\left(\boldsymbol{x}_t \middle| \sqrt{\alpha_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right)$$
$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t \middle| \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Noise to image(learnable)

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t))$$
where $\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2 \boldsymbol{I} = \beta_t \boldsymbol{I}$

- What is **target** of $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t \boldsymbol{I})$?
  - $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \approx q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$?

Denoising diffusion probabilistic models(DDPM)

Forward Markov chain $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$

Reverse Markov chain $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$

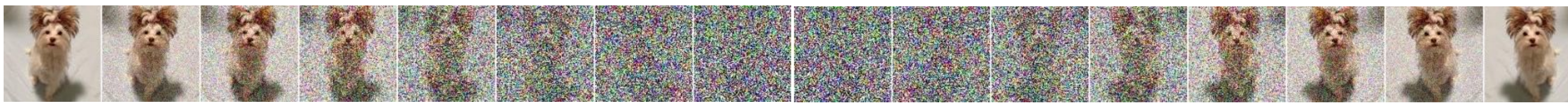


Image to noise(prescribed)

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = N\left(\boldsymbol{x}_t \middle| \sqrt{\alpha_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right)$$
$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t \middle| \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Noise to image(learnable)

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t))$$
where $\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2 \boldsymbol{I} = \beta_t \boldsymbol{I}$

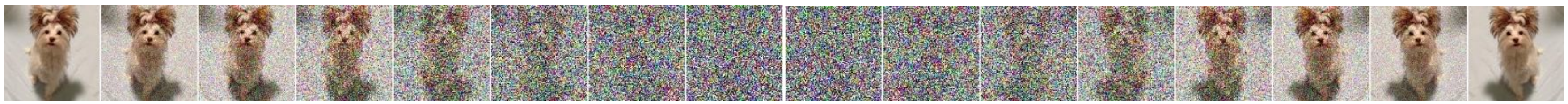
- What is **target** of $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t \boldsymbol{I})$?
  - $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \approx q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$?
  - $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ is not tractable

$$q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = \frac{q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})q(\boldsymbol{x}_{t-1})}{q(\boldsymbol{x}_t)}, q(\boldsymbol{x}_t) = \int q(\boldsymbol{x}_t|\boldsymbol{x}_0)q(\boldsymbol{x}_0)d\boldsymbol{x}_0$$

Denoising diffusion probabilistic models(DDPM)

Forward Markov chain $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ Reverse Markov chain $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$

p_{data} p_{data}



$\boldsymbol{x}_0$ $\boldsymbol{x}_1$ $\cdots$ $\boldsymbol{x}_T \sim N(\boldsymbol{0}, \boldsymbol{I})$ $\cdots$ $\boldsymbol{x}_1$ $\boldsymbol{x}_0$

Image to noise(prescribed)

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = N\left(\boldsymbol{x}_t\middle|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right)$$
$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t\middle|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Noise to image(learnable)

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t))$$
where $\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2 \boldsymbol{I} = \beta_t \boldsymbol{I}$

- What is **target** of $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t \boldsymbol{I})$?
  - $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$ is tractable! Why?

Denoising diffusion probabilistic models(DDPM)

Image to noise(prescribed)

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = N\left(\boldsymbol{x}_t\middle|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, \beta_t\boldsymbol{I}\right)$$
$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t\middle|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Noise to image(learnable)

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \Sigma_\theta(\boldsymbol{x}_t, t))$$
where $\Sigma_\theta(\boldsymbol{x}_t, t) = \sigma_t^2\boldsymbol{I} = \beta_t\boldsymbol{I}$

- What is **target** of $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t\boldsymbol{I})$?
  - $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$ is tractable. Why?

$$q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = \frac{q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{x}_0)q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_0)}{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}$$
$$= N\left(\boldsymbol{x}_{t-1}|\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, t), \tilde{\beta}_t\boldsymbol{I}\right)$$
$$\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, t) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t,$$
$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

Denoising diffusion probabilistic models(DDPM)

Image to noise(prescribed)

$$q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = N\left(\boldsymbol{x}_t\middle|\sqrt{\alpha_t}\boldsymbol{x}_{t-1}, \beta_t \boldsymbol{I}\right)$$
$$q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t\middle|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$$

Noise to image(learnable)

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,t), \Sigma_\theta(\boldsymbol{x}_t,t))$$
where $\Sigma_\theta(\boldsymbol{x}_t,t) = \sigma_t^2\boldsymbol{I} = \beta_t\boldsymbol{I}$

$$p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,t), \beta_t\boldsymbol{I}) \approx q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)$$
$$= N\left(\boldsymbol{x}_{t-1}|\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t,t), \tilde{\beta}_t\boldsymbol{I}\right)$$

- I.e.,

$$\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,t) \approx \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t,t) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t$$

- If $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t,t) := \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + \beta_t\boldsymbol{s}_\theta(\boldsymbol{x}_t,t)\right)$, then

$$\boldsymbol{s}_\theta(\boldsymbol{x}_t,t) \approx \nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t|\boldsymbol{x}_0) = -\frac{\boldsymbol{x}_t - \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0}{1-\bar{\alpha}_t}$$

Foundation of DDPM

$$\begin{aligned}
&\operatorname*{argmin}_{\theta} D\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) \parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\\
&= \operatorname*{argmin}_{\theta} E_{\boldsymbol{x}_0\sim p_{data}}\left[D\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]
\end{aligned}$$

- Proof: By Fubini theorem,

$$\begin{aligned}
&-\int q(\boldsymbol{z})\log p_\theta(\boldsymbol{z})\, d\boldsymbol{z} = -\int\left[\int q(\boldsymbol{z}|\boldsymbol{x}_0)p_\theta(\boldsymbol{z})\, d\boldsymbol{z}\right]p_{data}(\boldsymbol{x}_0)d\boldsymbol{x}_0\\
&= D\left(q(\boldsymbol{z}) \parallel p_\theta(\boldsymbol{z})\right) + \text{const. w.r.t. } \theta\\
&= E_{\boldsymbol{x}_0\sim p_{data}}\left[D\left(q(\boldsymbol{z}|\boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{z})\right)\right] + \text{const. w.r.t. } \theta
\end{aligned}$$

Learning objective of DDPM

$$\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) \approx \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, t) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \frac{\sqrt{\bar{\alpha}_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t$$

Learning objective of DDPM

- Optimizing the variational bound on log-likelihood

$$\log p_\theta(\boldsymbol{x}_0) = \log \int p_\theta(\boldsymbol{x}_{0:T}) d\boldsymbol{x}_{1:T} = \log \int p_\theta(\boldsymbol{x}_{0:T}) \frac{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} d\boldsymbol{x}_{1:T}$$

$$\geq \int q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) \log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} d\boldsymbol{x}_{1:T}$$

Learning objective of DDPM

- Optimizing the variational bound on log-likelihood

$$\log p_\theta(\boldsymbol{x}_0) = \log \int p_\theta(\boldsymbol{x}_{0:T}) d\boldsymbol{x}_{1:T} = \log \int p_\theta(\boldsymbol{x}_{0:T}) \frac{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} d\boldsymbol{x}_{1:T}$$

$$\geq \int q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) \log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)} d\boldsymbol{x}_{1:T}$$

- The Markov property of $q(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ and $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)$ implies

$$p_\theta(\boldsymbol{x}_{0:T}) = p_\theta(\boldsymbol{x}_T) \prod_{t=2}^{T} p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t),$$

$$q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) = q(\boldsymbol{x}_T|\boldsymbol{x}_0) \prod_{t=2}^{T} q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$$

Learning objective of DDPM

$$\log \frac{p_\theta(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}$$

$$= \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1) + \sum_{t=2}^{T} \log \frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)} + \log \frac{p_\theta(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}$$

$$\int q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\, d\boldsymbol{x}_{1:T} = \int q(\boldsymbol{x}_1|\boldsymbol{x}_0) \log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)\, d\boldsymbol{x}_1$$

$$= E_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)]$$

$$\int q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0) \log \frac{p_\theta(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\, d\boldsymbol{x}_{1:T} = \int q(\boldsymbol{x}_T|\boldsymbol{x}_0) \log \frac{p_\theta(\boldsymbol{x}_T)}{q(\boldsymbol{x}_T|\boldsymbol{x}_0)}\, d\boldsymbol{x}_T$$

$$= -D[q(\boldsymbol{x}_T|\boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_T)]$$

Learning objective of DDPM

$$\int q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)\log\frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}d\boldsymbol{x}_{1:T}$$

$$=\int\int q(\boldsymbol{x}_{t-1},\boldsymbol{x}_t|\boldsymbol{x}_0)\log\frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}d\boldsymbol{x}_{t-1}\,d\boldsymbol{x}_t$$

$$=\int\int \frac{q(\boldsymbol{x}_{t-1},\boldsymbol{x}_t,\boldsymbol{x}_0)}{q(\boldsymbol{x}_0)}\frac{q(\boldsymbol{x}_t,\boldsymbol{x}_0)}{q(\boldsymbol{x}_t,\boldsymbol{x}_0)}\log\frac{p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)}{q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)}d\boldsymbol{x}_{t-1}\,d\boldsymbol{x}_t$$

$$=-\int q(\boldsymbol{x}_t|\boldsymbol{x}_0)\,D[q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)]d\boldsymbol{x}_t$$

$$=-E_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\big[D[q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t,\boldsymbol{x}_0)\parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)]\big]$$

Learning objective of DDPM

$$
\begin{aligned}
&E_{\boldsymbol{x}_0 \sim q(\boldsymbol{x}_0)}[\log p_\theta(\boldsymbol{x}_0)] \\
&\geq E_{q(\boldsymbol{x}_0)}\left[E_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}[\log p_\theta(\boldsymbol{x}_0|\boldsymbol{x}_1)]\right] - E_{q(\boldsymbol{x}_0)}\big[D[q(\boldsymbol{x}_T|\boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_T)]\big] \\
&- \sum_{t=2}^{T} E_{q(\boldsymbol{x}_0)}\left[E_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\big[D[q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)]\big]\right]
\end{aligned}
$$

Learning objective of DDPM

- $p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = N(\boldsymbol{x}_{t-1}|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t), \beta_t \boldsymbol{I})$
- $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) = N\left(\boldsymbol{x}_{t-1}|\tilde{\boldsymbol{\mu}}_t, \tilde{\beta}_t \boldsymbol{I}\right)$
  - $\tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, t) = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t$
  - $\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$
- $q(\boldsymbol{x}_t|\boldsymbol{x}_0) = N\left(\boldsymbol{x}_t\middle|\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0, (1-\bar{\alpha}_t)\,\boldsymbol{I}\right)$, $\quad \boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \sim \boldsymbol{N}(\boldsymbol{0}, \boldsymbol{I})$

Learning objective of DDPM

$$L_{t-1}(\theta) = E_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t)}\left[D\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$
$$= E_{x_0 \sim p_{data}} E_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D\left(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_\theta(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t)\right)\right]$$
$$= E_{x_0 \sim p_{data}} E_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)} \|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) - \tilde{\boldsymbol{\mu}}_t(\boldsymbol{x}_t, t)\|_2^2$$
$$= E_{x_0 \sim p_{data}} E_{\boldsymbol{x}_t \sim q(\boldsymbol{x}_t|\boldsymbol{x}_0)} \left\|\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) - \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\boldsymbol{x}_0 - \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\boldsymbol{x}_t\right\|_2^2$$
$$= \frac{\beta_t^2}{\alpha_t(1-\bar{\alpha}_t)} E_{x_0 \sim p_{data}} E_{\boldsymbol{\epsilon} \sim N(\boldsymbol{0}, \boldsymbol{I})} \left\|\boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t\right) - \boldsymbol{\epsilon}\right\|_2^2$$

- if $\boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right)$

Training and noise predictor

$$(\text{mean predictor}) \quad \boldsymbol{\mu}_\theta(\boldsymbol{x}_t, t)$$

$$= \frac{1}{\sqrt{\alpha_t}}\left(\boldsymbol{x}_t + \frac{\beta_t}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t)\right) \quad (\text{noise predictor})$$

- I.e., $\frac{1}{\sqrt{1-\alpha_t}}\boldsymbol{\epsilon}_\theta(\boldsymbol{x}_t, t) \approx \nabla_{\boldsymbol{x}_t} \log q(\boldsymbol{x}_t | \boldsymbol{x}_0)$

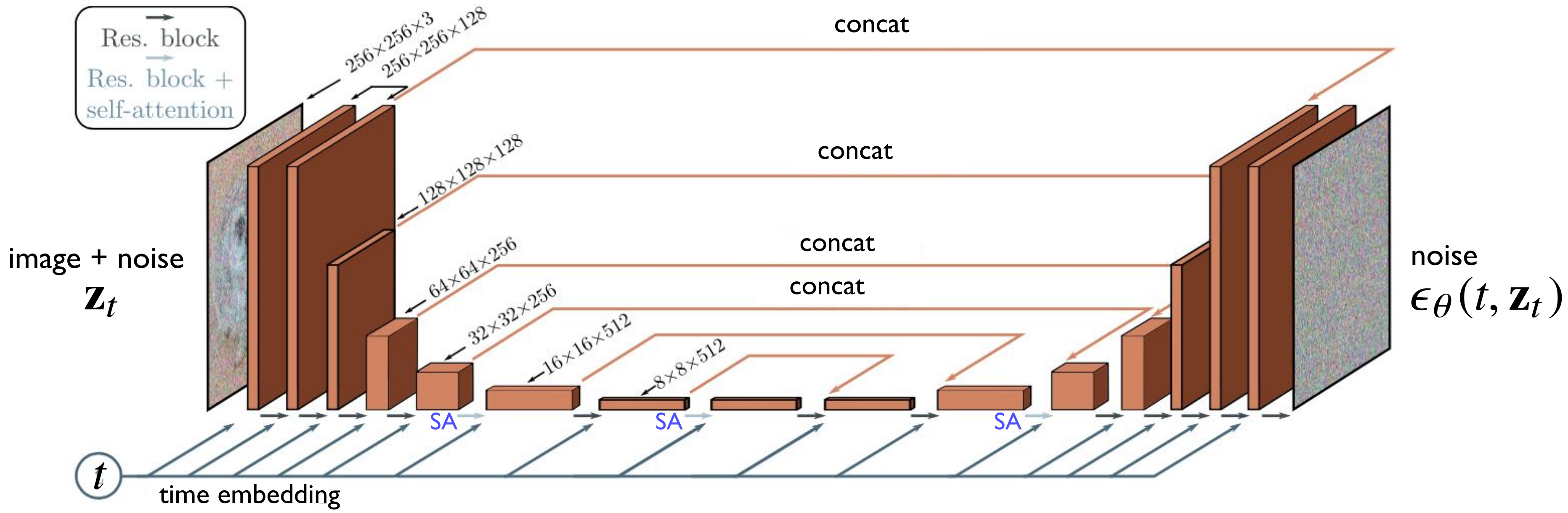
Algorithm 1 Training

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

Algorithm 2 Sampling

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

Architecture: U-net + self-attention + time Embedding



Experiments

- $T = 1000$, linear variance schedule $\beta_1 = 10^{-4}$ to $\beta_T = 0.02$
- U-Net backbone similar to an unmasked PixelCNN++ with group normalization

Figure 3: LSUN Church samples. FID=7.89

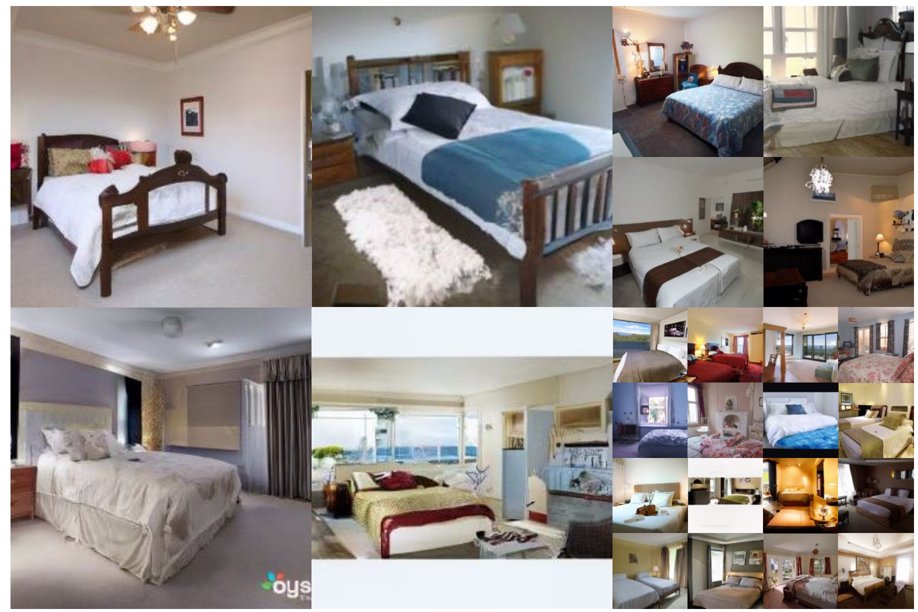


Figure 4: LSUN Bedroom samples. FID=4.90

Denoising Diffusion Probabilistic Models

Jonathan Ho, Ajay Jain, Pieter Abbeel. NeurIPS 2020

Thanks