Paper Presentation: Generative Modeling by Estimating Gradients of the Data Distribution (NeurIPS 2019)

Overall Aim

- Goal of Generative Modeling: Use dataset to learn a model for generating new samples from PDF of data

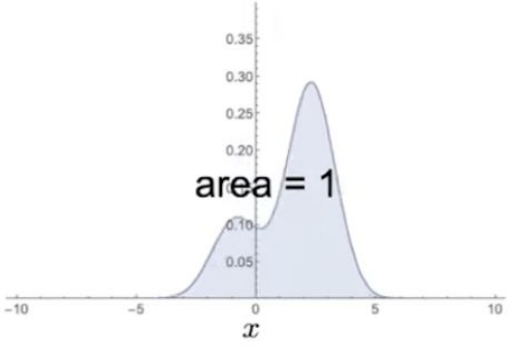
- Score matching: Originally designed to learn non-normalized statistical models from i.i.d. samples drawn from unknown data distribution

Why Score Modeling?

Flexible models

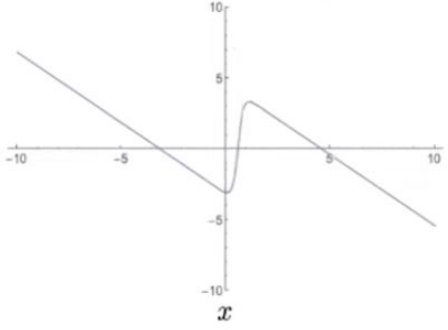
Score functions bypass the normalizing constant

$p(\mathbf{x})$

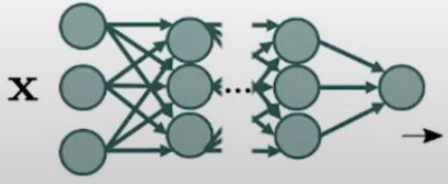


Probability density function

$\nabla_{\mathbf{x}} \log p(\mathbf{x})$



Score function



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

$$\begin{aligned} \nabla_{\mathbf{x}} \log p_{\boldsymbol{\theta}}(\mathbf{x}) &= \nabla_{\mathbf{x}} f_{\boldsymbol{\theta}}(\mathbf{x}) - \boxed{\nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}}} \\ &= \nabla_{\mathbf{x}} f_{\boldsymbol{\theta}}(\mathbf{x}) \qquad \boxed{\nabla_{\mathbf{x}} \log Z_{\boldsymbol{\theta}}} \rightarrow \mathbf{0} \\ &= \boxed{\boldsymbol{s}_{\boldsymbol{\theta}}(\mathbf{x})} \end{aligned}$$

Score model

Energy Based Model Score Training

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

$$\max_\theta \sum_{i=1}^{N} \log p_\theta(\mathbf{x}_i).$$

- F is real-valued function parameterized by learnable parameter theta
- Different from typical usage of score matching, authors do not use gradient of energy-based model as the score network to avoid extra computation from higher-order gradients

- Score network s_θ : R^d → R^d to be used is a neural network parameterized by θ
  - Will be trained to approximate score of p_data(x)

Score models can be estimated from data

Given: $\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p_{\text{data}}(\mathbf{x})$

Goal: $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$

Score Model: $s_\theta(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}^d \approx \nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$

Objective: How to compare two vector fields of scores?

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

(Fisher divergence)

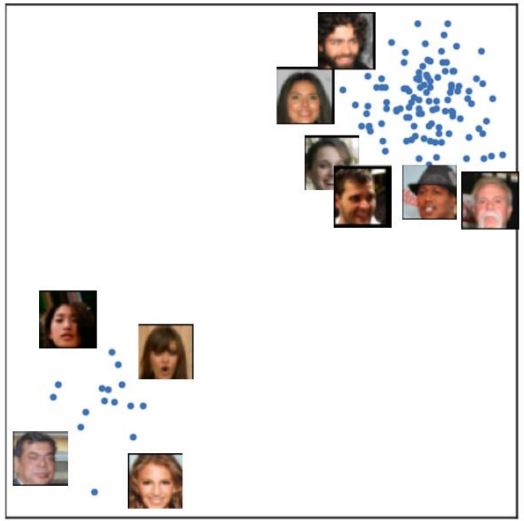
Integration by parts (Gauss's theorem)

Score Matching [Hyvärinen 2005]:

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\frac{1}{2}\left\|s_\theta(\mathbf{x})\right\|_2^2 + \text{trace}\big(\underbrace{\nabla_{\mathbf{x}} s_\theta(\mathbf{x})}_{\text{Jacobian of } s_\theta(\mathbf{x})}\big)\right]$$

$$\approx \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{2}\left\|s_\theta(\mathbf{x}_i)\right\|_2^2 + \text{trace}(\nabla_{\mathbf{x}} s_\theta(\mathbf{x}_i))\right]$$

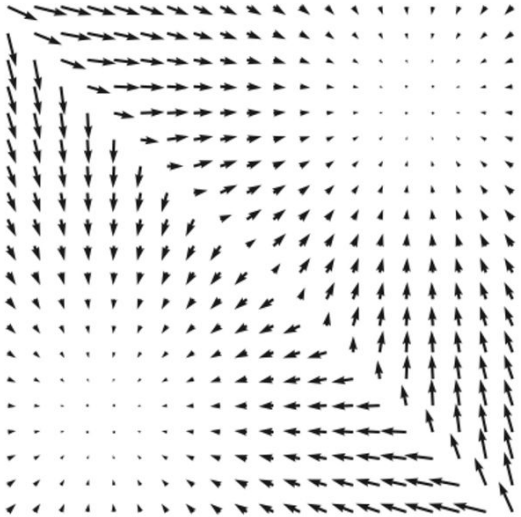
Naive Approach (Little Success)



Data samples

$$\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N\} \overset{\text{i.i.d.}}{\sim} p(\mathbf{x})$$

score matching



Scores

$$\mathbf{s}_\theta(\mathbf{x}) \approx \nabla_{\mathbf{x}} \log p(\mathbf{x})$$

Langevin dynamics

New samples

Score Matching Runs Into Problems

In regions of low data density, naive score matching may not have enough evidence to estimate score functions accurately, due to lack of data samples

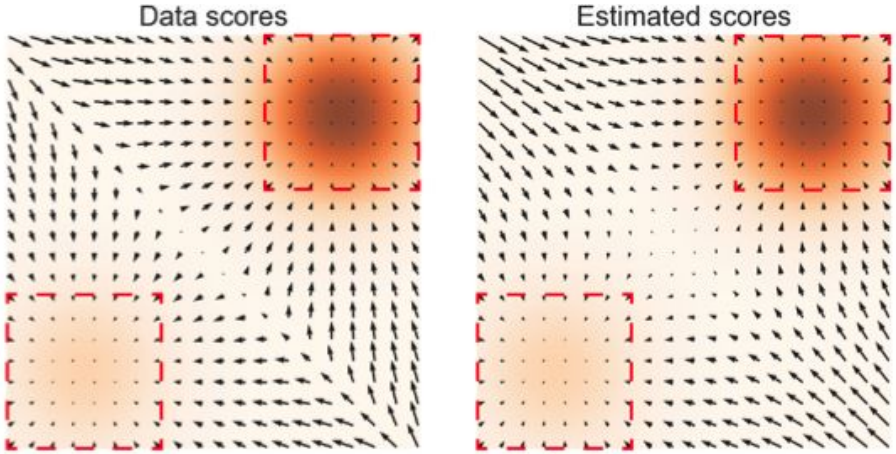
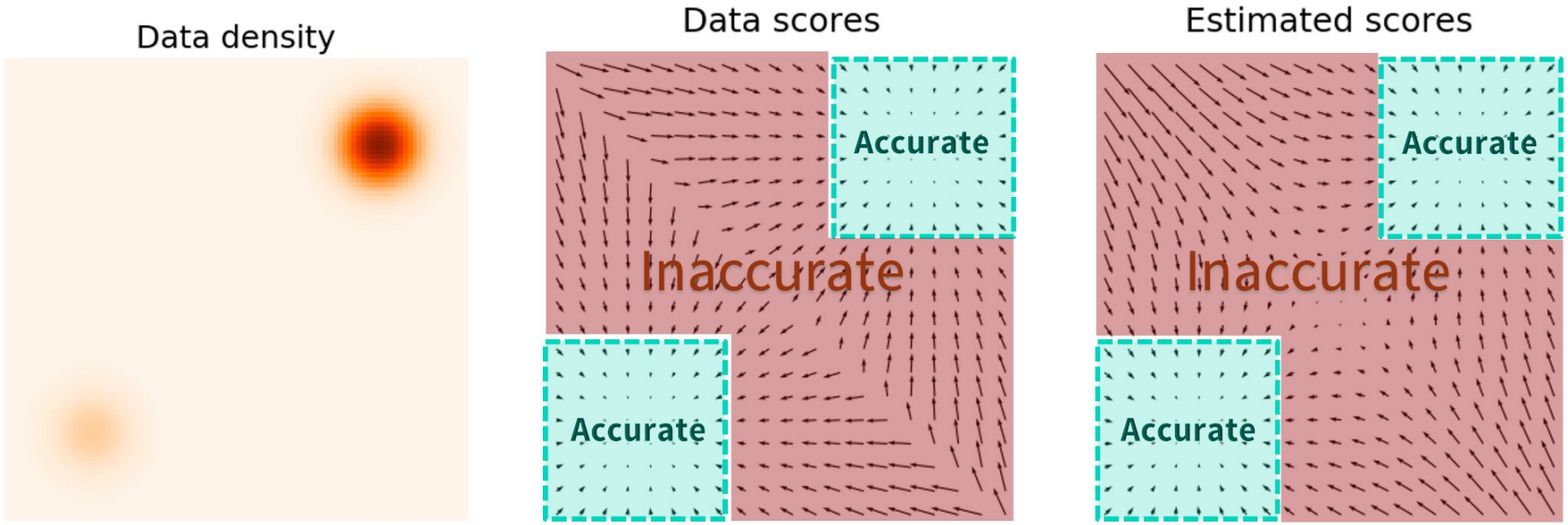


Figure 2: **Left**: $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x})$; **Right**: $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$. The data density $p_{\text{data}}(\mathbf{x})$ is encoded using an orange colormap: darker color implies higher density. Red rectangles highlight regions where $\nabla_{\mathbf{x}} \log p_{\text{data}}(\mathbf{x}) \approx \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x})$.

Low Data Density Causes Problems Estimating Scores



Denoising Score Matching

- Denoising score matching is a variant of score matching that completely circumvents calculating $\operatorname{tr}(\nabla_{\mathbf{x}} \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}))$
- Perturbs data point x with pre-specified noise distribution $q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x})$
- Employs score matching to estimate perturbed data distribution's score

$$q_\sigma(\tilde{\mathbf{x}}) \triangleq \int q_\sigma(\tilde{\mathbf{x}} \mid \mathbf{x}) p_{\text{data}}(\mathbf{x}) \mathrm{d}\mathbf{x}$$

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

Sliced score matching

- **Intuition:** one dimensional problems should be easier
- **Idea:** project onto random directions
- **Randomized objective: Sliced Fisher Divergence**

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

Which Score Matching Algorithm to Use?

- Authors chose denoising score matching over sliced score matching
- Reason:
  - Slightly faster
  - Naturally fits the task of estimating scores of noise-perturbed data distributions

Noise Conditional Score Networks (NCSN)

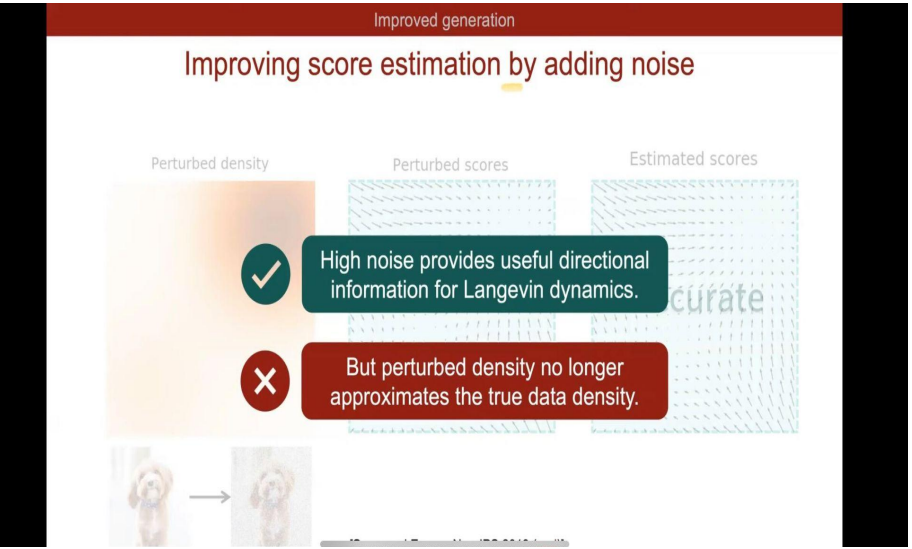
- Approximate score function with a neural network when data is perturbed with L levels of noise (geometric sequence)

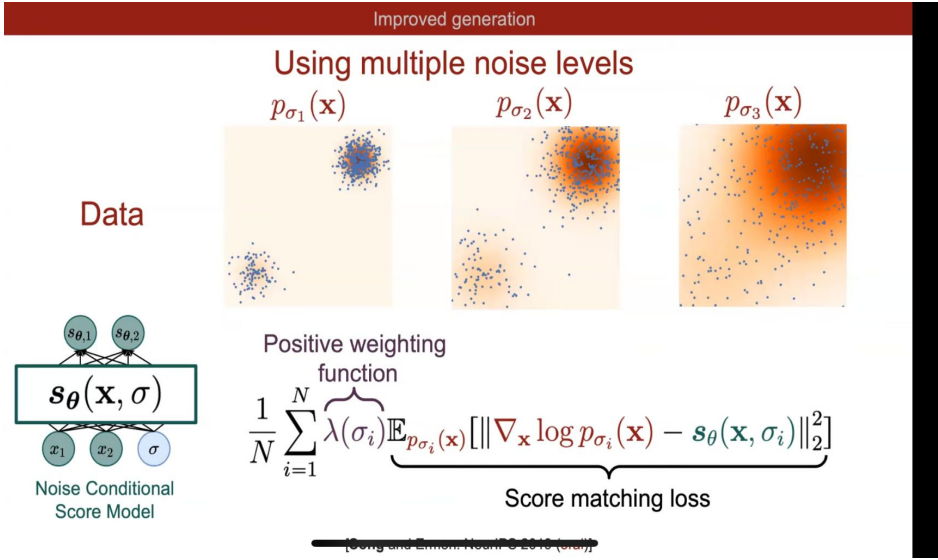
$$\forall \sigma \in \{\sigma_i\}_{i=1}^{L} : \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, \sigma) \approx \nabla_{\mathbf{x}} \log q_{\sigma}(\mathbf{x})$$

- NCSN is conditioned on the noise level and estimates the scores at all noise magnitudes

Why Multiple Noise Scales?

- Authors trained a single score network
  - Conditioned on the noise level to estimate the scores at all noise magnitudes
- Multiple noise levels makes it possible to obtain a sequence of noise-perturbed distributions that converge to the true data distribution





Manifold Hypothesis

- Data in the real world tend to concentrate on low dimensional manifolds embedded in a high dimensional space (**“Ambient Space”)**

- Since the distributions $\{q_{\sigma_i}\}_{i=1}^{L}$ are all perturbed by Gaussian noise, their supports span the whole space and their scores are well-defined, avoiding difficulties from the manifold hypothesis

Langevin Dynamics Equation

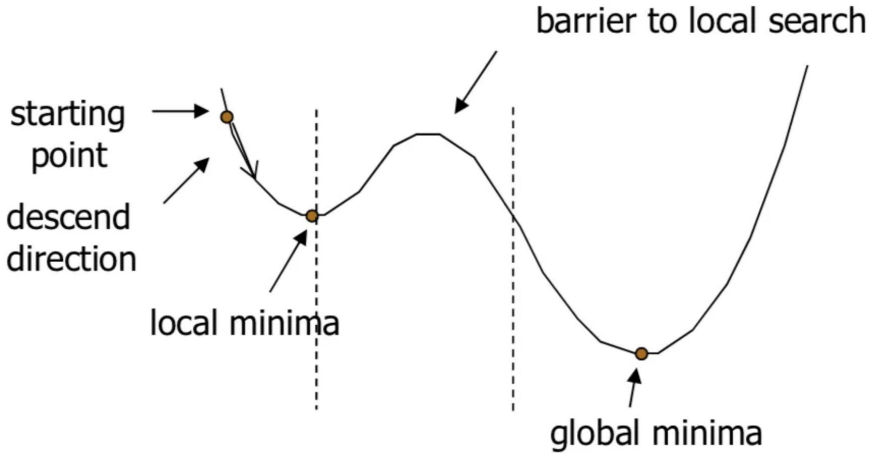
- Describes Brownian Motion

$$\tilde{\mathbf{x}}_t = \tilde{\mathbf{x}}_{t-1} + \frac{\epsilon}{2}\nabla_{\mathbf{x}} \log p(\tilde{\mathbf{x}}_{t-1}) + \sqrt{\epsilon}\,\mathbf{z}_t$$

- When 2 modes of data distribution are separated by low density regions, Langevin dynamics will not be able to correctly recover the relative weights of these two modes in reasonable time
  - May not converge to true distribution

Sampling Process

- After the NCSN $s\theta(x, \sigma)$ is trained, sampling approach is needed
- Authors propose **Annealed Langevin dynamics** for sampling
  - Inspired by **Simulated Annealing (SA)** and **Annealed Importance Sampling**
  - Gradient Descent may not converge to global minima with local minima
- Simulated Annealing mimics the Physical Annealing process but is used for optimizing parameters in a model



Annealed Importance Sampling

The construction of AIS is as follows:

1. Let $p_0(x) = p(x) \propto f_0(x)$ be our target distribution.
2. Let $p_n(x) = q(x) \propto f_n(x)$ be our proposal distribution which only requirement is that we can sample independent point from it. It doesn't have to be close to $p_0(x)$ thus the requirement is more relaxed than importance sampling.
3. Define a sequence of intermediate distributions starting from $p_n(x)$ to $p_0(x)$ call it $p_j(x) \propto f_j(x)$. The requirement is that $p_j(x) \neq 0$ whenever $p_{j-1}(x) \neq 0$. That is, $p_j(x)$ has to cover the support of $p_{j-1}(x)$ so that we can take the ratio.
4. Define local transition probabilities $T_j(x, x')$.

Then to sample from $p_0(x)$, we need to:

- Sample an independent point from $x_{n-1} \sim p_n(x)$.
- Sample x_{n-2} from x_{n-1} by doing MCMC w.r.t. T_{n-1}.
- $\dots$
- Sample x_1 from x_2 by doing MCMC w.r.t. T_2.
- Sample x_0 from x_1 by doing MCMC w.r.t. T_1.

Simulated Annealing

- When electron has enough thermal energy it can jump over barriers and end up in a higher energy state and hence can avoid some local energy minima
- Calculate $-\Delta\mathcal{L}(\cdot)$
  If >0, new state is accepted
- Otherwise a uniform number is generated between [0, 1) and compared with P_b If P_b is greater it is accepted, otherwise it is rejected

$$\mathcal{P}_b = \min\left(1, \exp\left(\frac{-\Delta\mathcal{L}(\cdot)}{k \cdot T}\right)\right)$$

Algorithm 1 Annealed Langevin dynamics.

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

Initialize $\tilde{\mathbf{x}}_0$
for $i \leftarrow 1$ to L **do**
  $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2/\sigma_L^2$ ▷ α_i is the step size.
  for $t \leftarrow 1$ to T **do**
    Draw $\mathbf{z}_t \sim \mathcal{N}(0, I)$
    $\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2}\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \sqrt{\alpha_i}\,\mathbf{z}_t$
  end for
  $\tilde{\mathbf{x}}_0 \leftarrow \tilde{\mathbf{x}}_T$
end for
return $\tilde{\mathbf{x}}_T$

$$p_{\text{data}} = \frac{1}{5}\mathcal{N}((-5,-5), I) + \frac{4}{5}\mathcal{N}((5,5), I)$$

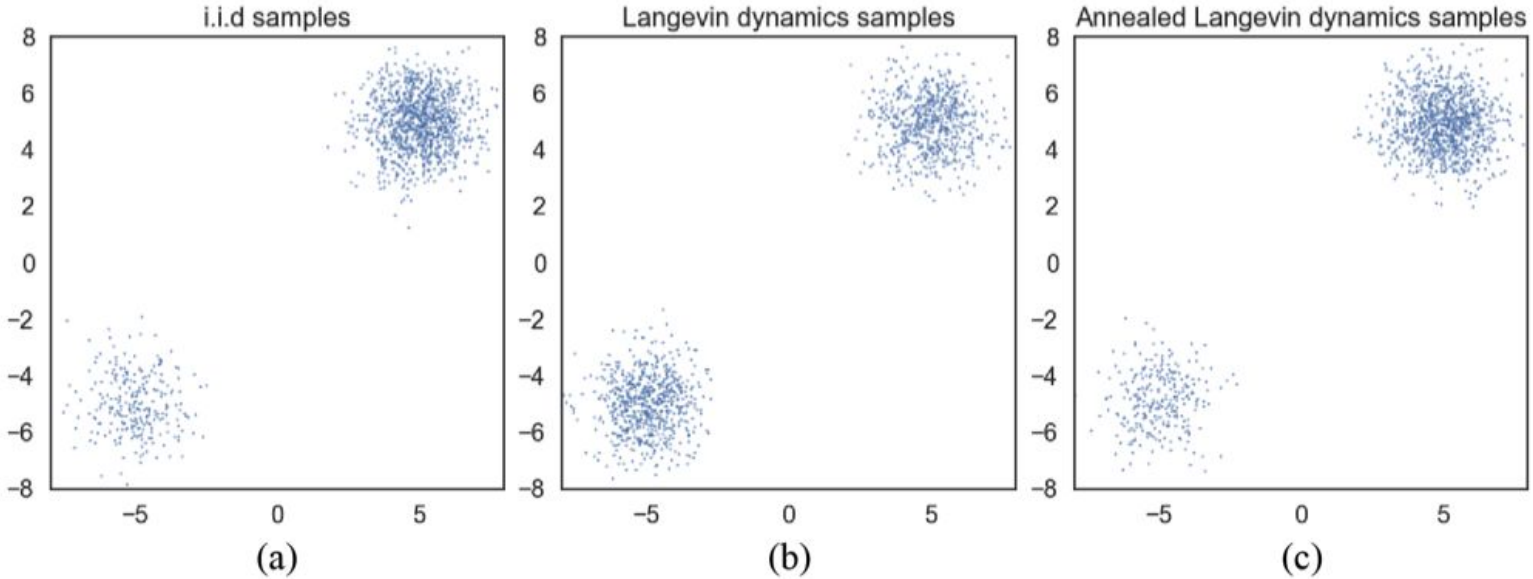


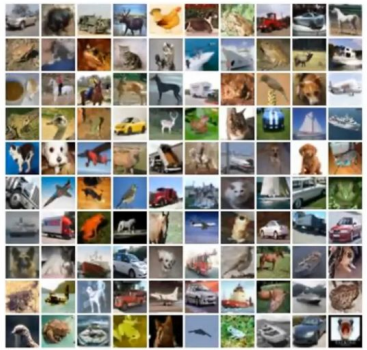
Figure 3: Samples from a mixture of Gaussian with different methods. (a) Exact sampling. (b) Sampling using Langevin dynamics with the exact scores. (c) Sampling using annealed Langevin dynamics with the exact scores. Clearly Langevin dynamics estimate the relative weights between the two modes incorrectly, while annealed Langevin dynamics recover the relative weights faithfully.

Results of NCSN for Image Sample Generation on CIFAR-10

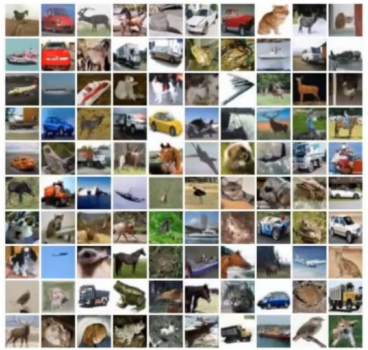
Improved generation

State-of-the-art sample quality on CIFAR-10

Training data



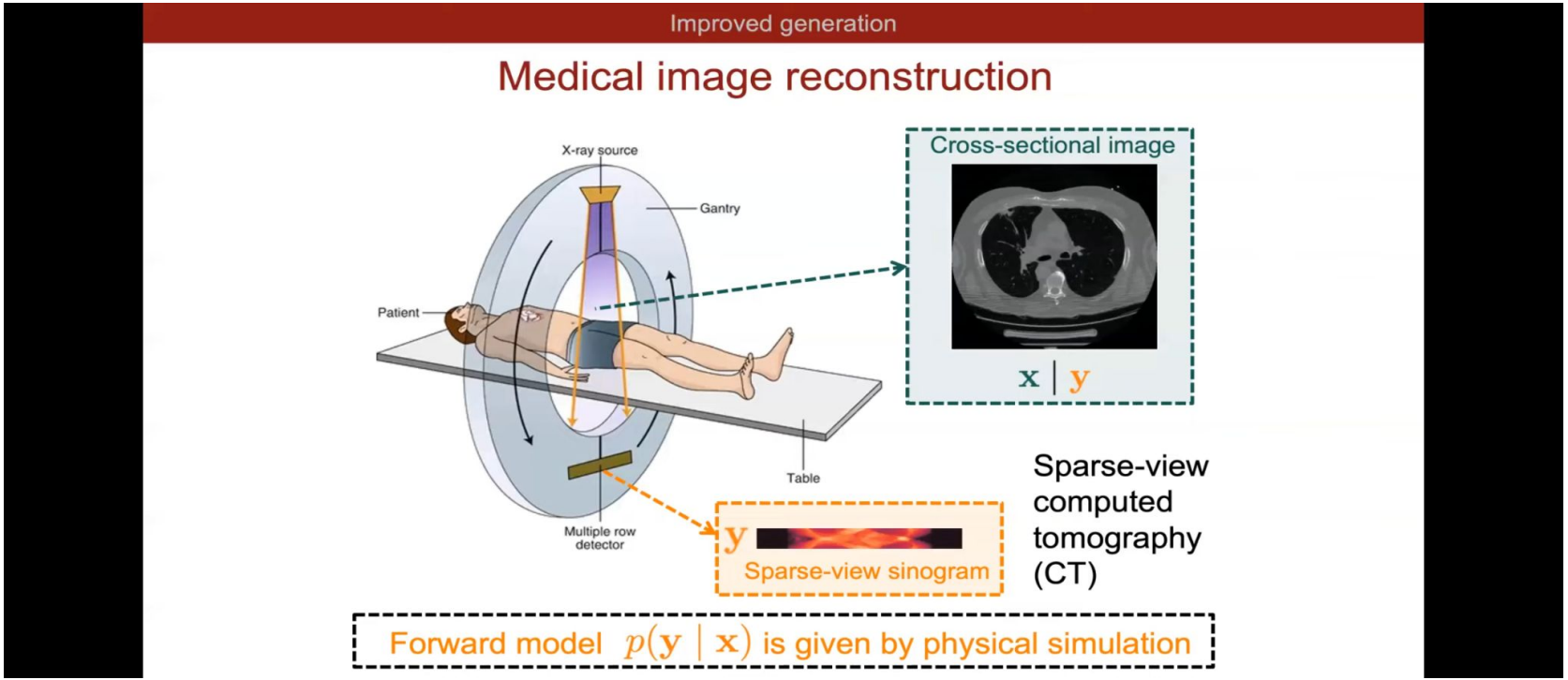
Samples



Method	FID score↓	Inception Score↑
StyleGAN2-ADA (Karras et al. 2020)	2.92	9.83
Ours	**2.20**	**9.89**

[**Song** et al. ICLR 2021 (Outstanding Paper Award)]

Results of NCSN+Uses for Medical Image Processing



Score-based generative modeling: summary

Flexible models

- Bypass the normalizing constant
- Principled statistical methods

[**Song** et al. UAI 2019 oral]

Improved generation

- Higher sample quality than GANs
- Controllable generation

[**Song** & Ermon. NeurIPS 2019 oral]
[**Song** & Ermon. NeurIPS 2020]
[**Song** et al. ICLR 2021 oral]
(Outstanding Paper Award)
[**Song** et al. ICLR 2022]

Probability evaluation

- Accurate probability evaluation
- Better estimation of data probabilities

[**Song** et al. ICLR 2021 oral]
[**Song** et al. NeurIPS 2021 spotlight]

NCSN’s Connection to Diffusion Models

- High Level: Both add noise to the data first before generating samples
  - Diffusion Models: Hierarchical latent variable models that generate samples by learning variational decoder to reverse a discrete diffusion process which perturbs data to noise
  - NCSN learns a score network
    - Jointly learns score functions of PDFs perturbed with various noise levels in generative way

NCSN’s Connection To Diffusion Models

- Without awareness of DDPMs, score-based generative modeling was proposed independently
- Both perturb data with multiple noise scales, but the connection between score-based generative modeling and diffusion probabilistic modeling seemed superficial, since the former is trained by score matching and sampled by Langevin dynamics, while the latter is trained by the evidence lower bound (ELBO) and sampled with a learned decoder

NCSN’s Connection To Diffusion Models

- In 2020, Jonathan Ho and colleagues significantly improved the empirical performance of diffusion probabilistic models and first unveiled a deeper connection to score-based generative modeling

    - Showed that the ELBO used for training diffusion probabilistic models is essentially equivalent to the weighted combination of score matching objectives used in score-based generative modeling

    - Moreover, by parameterizing the decoder as a sequence of score-based models with a U-Net architecture, they demonstrated for the first time that diffusion probabilistic models can also generate high quality image samples comparable or superior to GANs

NCSN’s Connection To Diffusion Models

- Sampling method of diffusion probabilistic models can be integrated with annealed Langevin dynamics of score-based models to create a unified and more powerful sampler (Predictor-Corrector sampler)
- By generalizing the number of noise scales to infinity, it was proved that score-based generative models and diffusion probabilistic models can both be seen as discretizations to SDEs determined by score functions
    - This is the key point: bridges score-based generative modeling and diffusion probabilistic modeling into unified framework

Future Directions

- In NCSN, authors focus mostly on architectures useful for image generation
- Left architecture design in other domains as future work