Deep Unsupervised Learning using Nonequilibrium Thermodynamics

서울시립대학교
UNIVERSITY OF SEOUL

CIDA Lab.

Presentation: Baekryun Seong

2024.10.08

Table of Contents

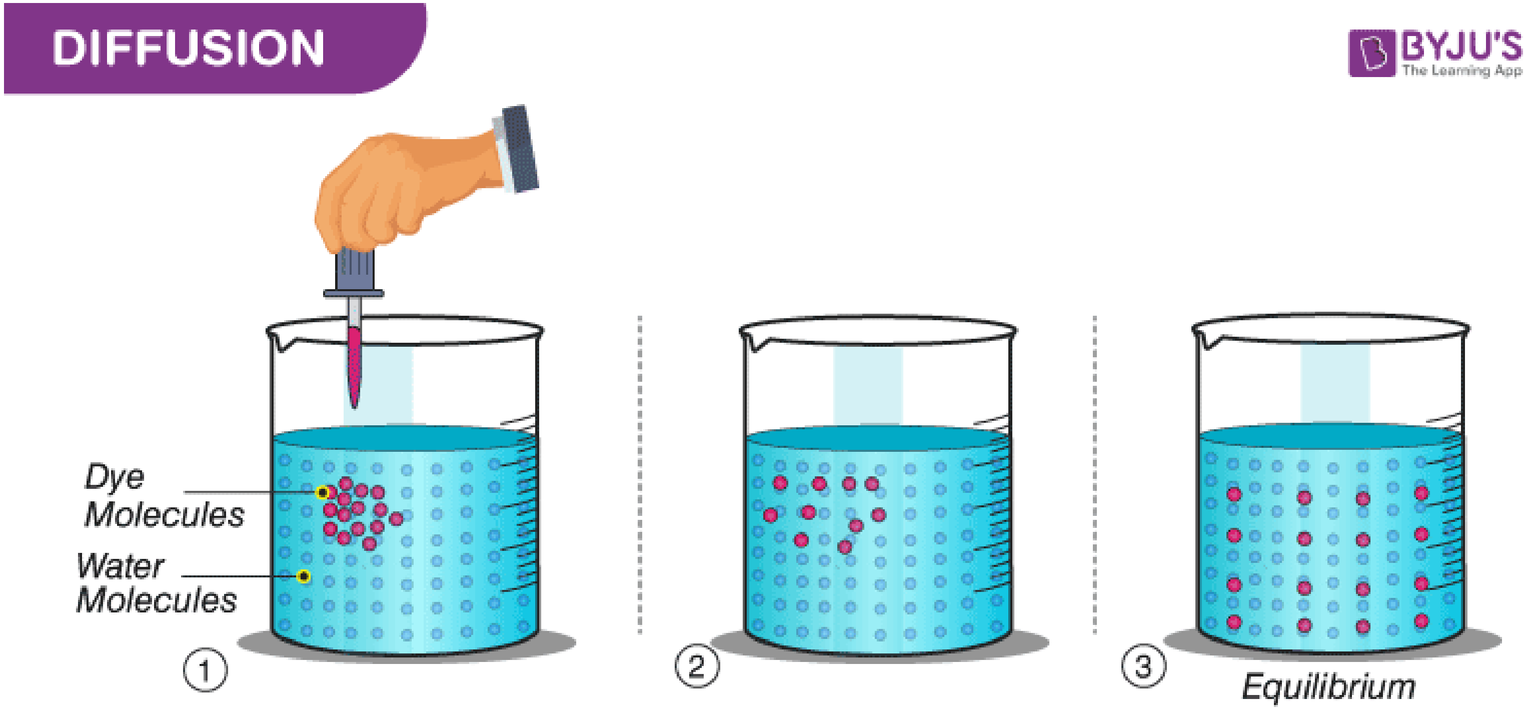
- Preliminary
    - Annealed Importance Sampling
    - Fokker-Planck Equation
    - Kolmogorov Forward and Backward Equation
- Deep Unsupervised Learning using Nonequilibrium Thermodynamics
- Denoising Diffusion Probabilistic Models

Intro: Diffusion Models

"In machine learning, diffusion models, also known as diffusion probabilistic models or score-based generative models, are a class of latent variable generative models.
(…)
Diffusion-based image generators have seen widespread commercial interest, such as Stable Diffusion and DALL-E. These models typically combine diffusion models with other models, such as text-encoders and cross-attention modules to allow text-conditioned generation.
Other than computer vision, diffusion models have also found applications in ***natural language processing*** *such as text generation and summarization,* ***sound generation****, and* ***reinforcement learning****."*

Diffusion model – Wikipedia, obtained 2024.09.27.

Preliminary



[What Is Diffusion? - Definition, Types & Examples Of Diffusion (byjus.com)]

Theoretical Bounds

Diffusion models have three foundation theories:

Annealed importance sampling (AIS) (Neal, 2001)

Fokker-Planck Equation

Kolmogorov forward and backward equation

Theoretical Bounds

Diffusion models have three foundation theories:

Annealed importance sampling (AIS) (Neal, 2001)

Fokker-Planck Equation

Kolmogorov forward and backward equation

Sampling Theory

Sampling theory (or **Monte Carlo methods**) generally discuss about drawing samples from complex distribution. e.g. How can we draw random sample from a Gaussian $\mathcal{N}(0,1)$?

There are many famous sampling methods, such as:

- **Inverse CDF** converts uniform distribution to others.
- **Importance sampling** converts some random distribution to target distribution.
- **Rejection sampling** is much faster in some cases.
- **The Metropolis-Hastings method** (MH) is feasible in high-dimensional distributions.
- **Gibbs sampling** is specialized form of MH, which can be utilized in high-dimensional and complex distribution.
- **Markov chain Monte Carlo** (MCMC) utilizes Markov chain with Gibbs sampling.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.
Bishop, C. M., & Nasrabadi, N. M. (2006). *Pattern recognition and machine learning* (Vol. 4, No. 4, p. 738). New York: springer.

Markov Chain Monte Carlo

Example 29.6. An example of a Markov chain is given by the Metropolis demonstration of section 29.4 (figure 29.12), for which the transition probability is

$$\mathbf{T} = \left[\begin{array}{*{21}{c}}
1/2 & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & \cdot & 1/2 \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1/2 & 1/2
\end{array}\right]$$

and the initial distribution was

$$p^{(0)}(x) = [\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ 1\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ \cdot\ ]. \tag{29.40}$$

The probability distribution $p^{(t)}(x)$ of the state at the tth iteration is shown for $t = 0, 1, 2, 3, 5, 10, 100, 200, 400$ in figure 29.14; an equivalent sequence of distributions is shown in figure 29.15 for the chain that begins in initial state $x_0 = 17$. Both chains converge to the target density, the uniform density, as $t \to \infty$.

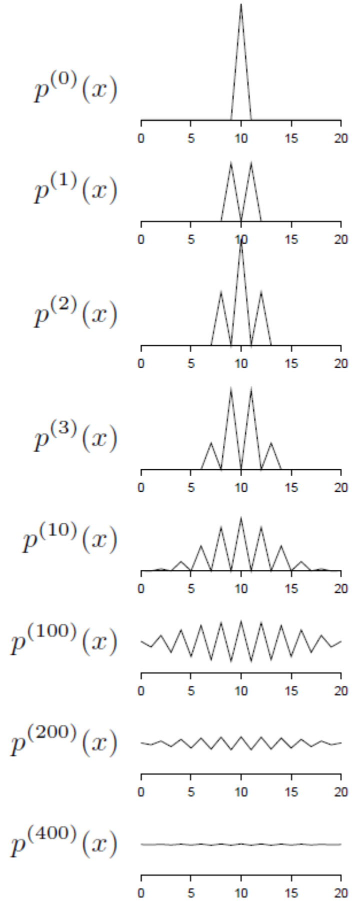


Figure 29.14. The probability distribution of the state of the Markov chain of example 29.6.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

Markov Chain Monte Carlo Cont'd

We want to utilize a simple distribution (e.g. uniform or gaussian) to sample from extremely complex high-dimensional distribution.

Idea: Sometimes, states distribution of Markov chain converges to a unique distribution.

The probability distribution of the state at $(t+1)$th iteration of the Markov chain, $p^{t+1}(\boldsymbol{x})$, is given by:

$$p^{(t+1)}(\boldsymbol{x}') = \int d^N\boldsymbol{x}\, P_{\boldsymbol{x}}^{\boldsymbol{x}'} p^{(t)}(\boldsymbol{x}).$$

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

Markov Chain Monte Carlo Cont’d

State distribution of Markov chain converges to some invariant distribution of the chain $\pi(\boldsymbol{x}')$ under two condition.

1. π must be the eigenvector of stochastic matrix P, and its eigenvalue is 1.
   - “… every right stochastic matrix has an "obvious" column eigenvector associated to the eigenvalue 1.” Stochastic matrix – Wikipedia, obtained 2024.09.28.
2. The chain must be ergodic: $p^{(t)}(\boldsymbol{x}) \to \pi(\boldsymbol{x})$ as $t \to \infty$, for any $p^{(0)}(\boldsymbol{x})$. In other words, there is only one eigenvalue $|\lambda| = 1$.
   - … Its matrix might be reducible, which means that the state space contains two or more subsets of states that can never be reached from each other. (…) The chain might have a periodic set, which means that, for some initial conditions, p(t)(x) doesn't tend to an invariant distribution, but instead tends to a periodic limit-cycle. – MacKay (2003). 373p.

MacKay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.
Stochastic matrix – Wikipedia

Annealed Importance Sampling

But MCMC still has the problem: It takes a lot of time to converge.

Simulated Annealing was introduced by Kirkpatrick, Gelatt, and Vecchi (1983).

Instead of simple iteration of P_π matrix, we progressively transform our distribution in many steps:

$$P_{\pi_0} \rightarrow P_{\pi_1} \rightarrow P_{\pi_2} \rightarrow \cdots \rightarrow P_\pi.$$

e.g. We get $\boldsymbol{s}_1$ by k iteration of P_{π_0} over $\boldsymbol{s}_0$, get $\boldsymbol{s}_2$ by $\left(P_{\pi_1}\right)^k$ over $\boldsymbol{s}_1$,…

Annealed importance sampling applies this technic to MCMC, making fast convergence.

Neal, R. M. (2001). Annealed importance sampling. *Statistics and computing*, 11, 125-139.

Theoretical Bounds

Diffusion models have three foundation theories:

Annealed importance sampling (AIS) (Neal, 2001)

Fokker-Planck Equation

Kolmogorov forward and backward equation

Stochastic Process

Let X_t be t-th trial in an infinite number of trials.

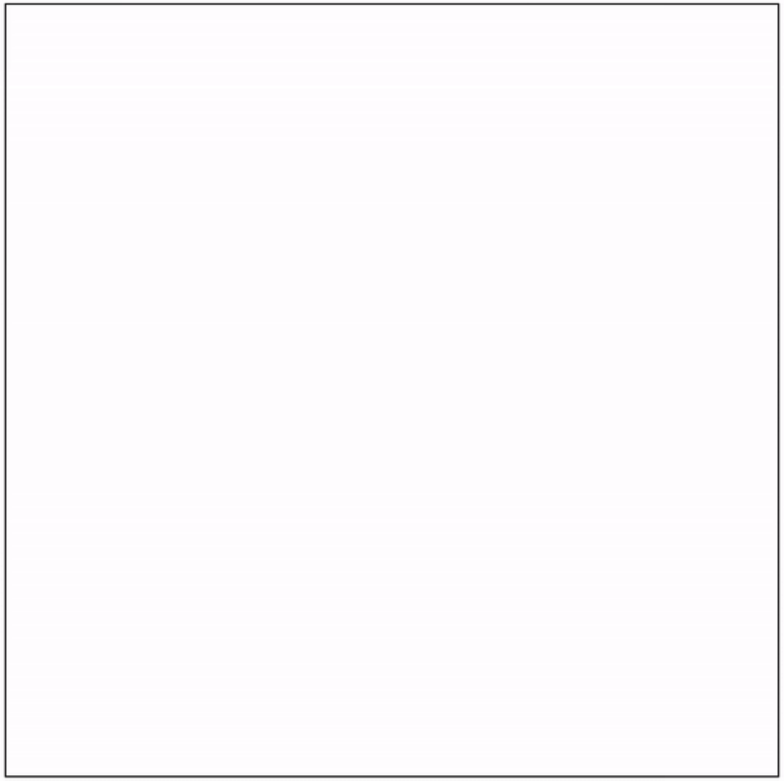
$\{X_t\}_{t=1}^{T}$ is stochastic process from infinite trials.

e.g. Random Walk, Markov process.

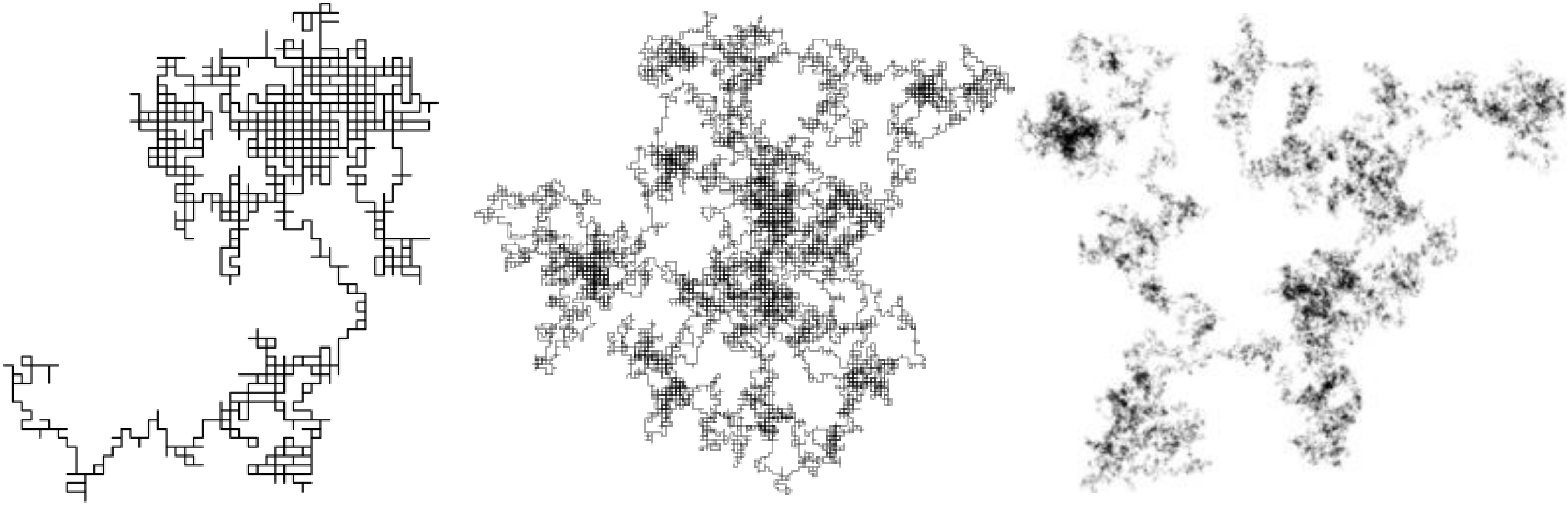
Random Walk

“*In mathematics, a random walk, sometimes known as a drunkard's walk, is a stochastic process that describes a path that consists of a succession of random steps on some mathematical space.*” - Random walk – Wikipedia, obtained 2024.09.28.

$$P(\Delta W = 1) = P(\Delta W = -1) = 0.5$$



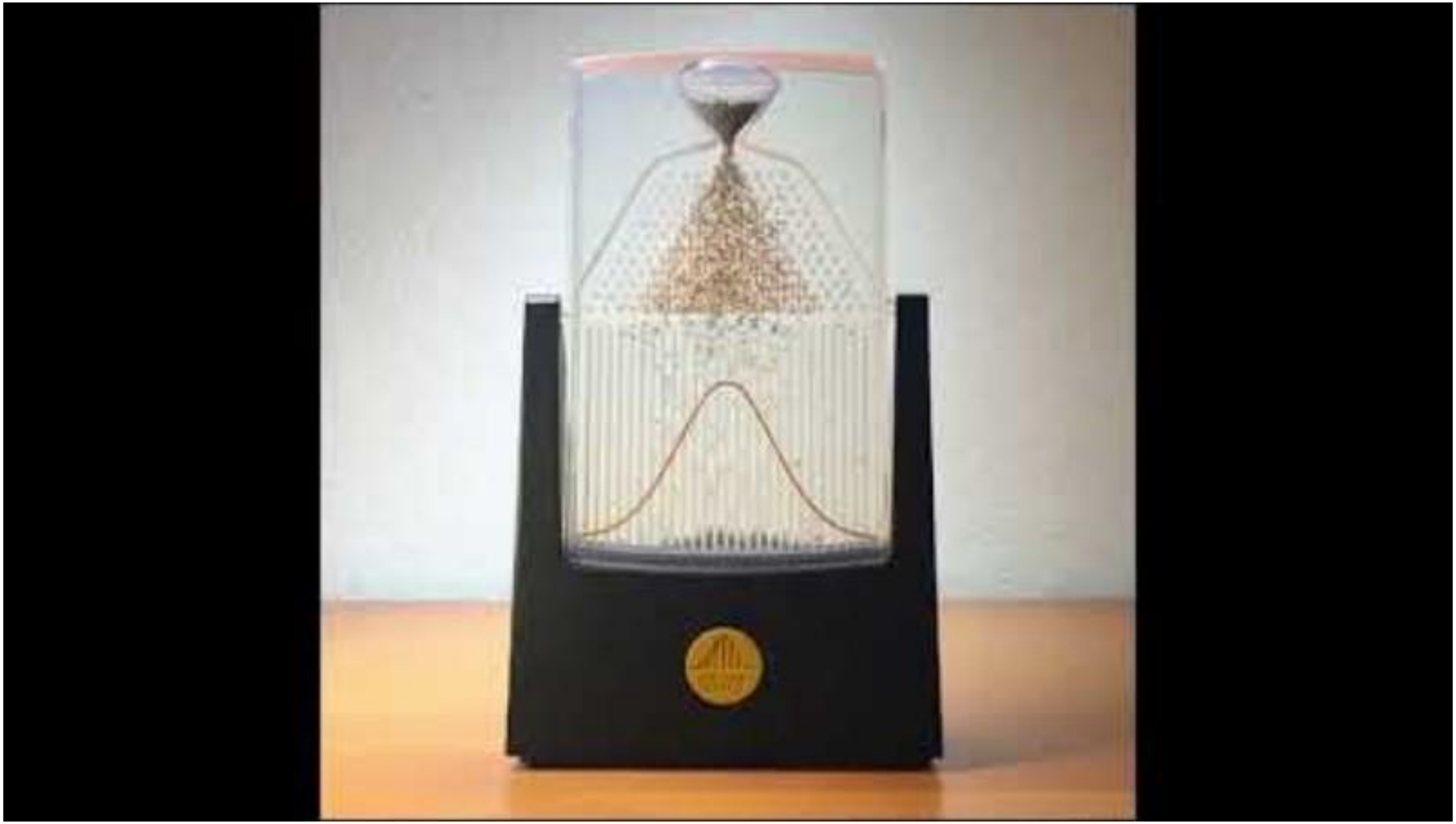
Random Walk Cont’d



But it is in descrete space. Let ΔZ be continuous.

Random walk – Wikipedia

At large N, binomial distributed random walk becomes normal distribution.

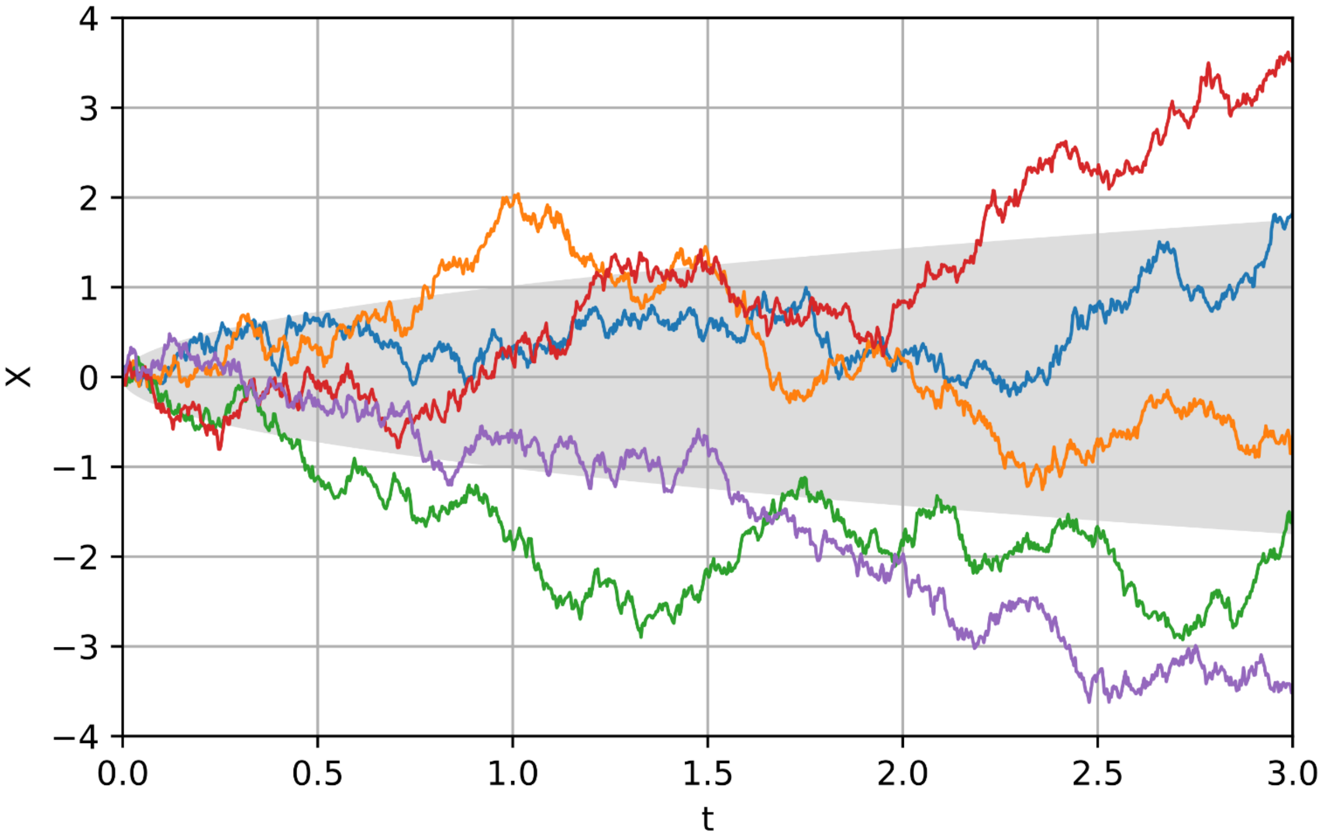


Galton Board - YouTube

Wiener Process Cont’d

Winner process is stochastic process on infinitesimal.

$$dW \sim \mathcal{N}(0, dt)$$



Wiener process - Wikipedia

Stochastic Differential Equation

Now we can define diffusion process with differential equation form.

This is called stochastic differential equation.

$$dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dW_t$$

Ito’s lemma gives us the way to differentiate stochastic process.

For random process $Y_t = f(X_t, t)$,

$$dY_t = \left(\frac{\partial f}{\partial t} + \frac{\partial f}{\partial X_t}\mu + \frac{\partial^2 f}{\partial X_t^2}\mu^2\right)dt + \left(\frac{\partial f}{\partial X_t}\sigma\right)dW_t.$$

Fokker-Planck equation uses Ito’s lemma to get $P(t, x_t)$.

$$\frac{\partial p(t, x_t)}{\partial t} = -\frac{\partial[p\mu]}{\partial x_t} + \frac{1}{2}\frac{\partial^2[p\sigma^2]}{\partial x_t^2}$$

Ito's lemma : 네이버 블로그 (naver.com)
무작위 걸음(랜덤 워크 random walk) 104 … : (tistory.com)
확률미분방정식 (freshrimpsushi.github.io)

Theoretical Bounds

Diffusion models have three foundation theories:

Annealed importance sampling (AIS) (Neal, 2001)

Fokker-Planck Equation

Kolmogorov forward and backward equation

Kolmogorov Forward and Backward Equation

For stochastic matrix $P(t)$ and infinitesimal generator matrix $Q \coloneqq P'(0)$,

$$\frac{dP(t)}{dt} = QP(t) = P(t)Q.$$

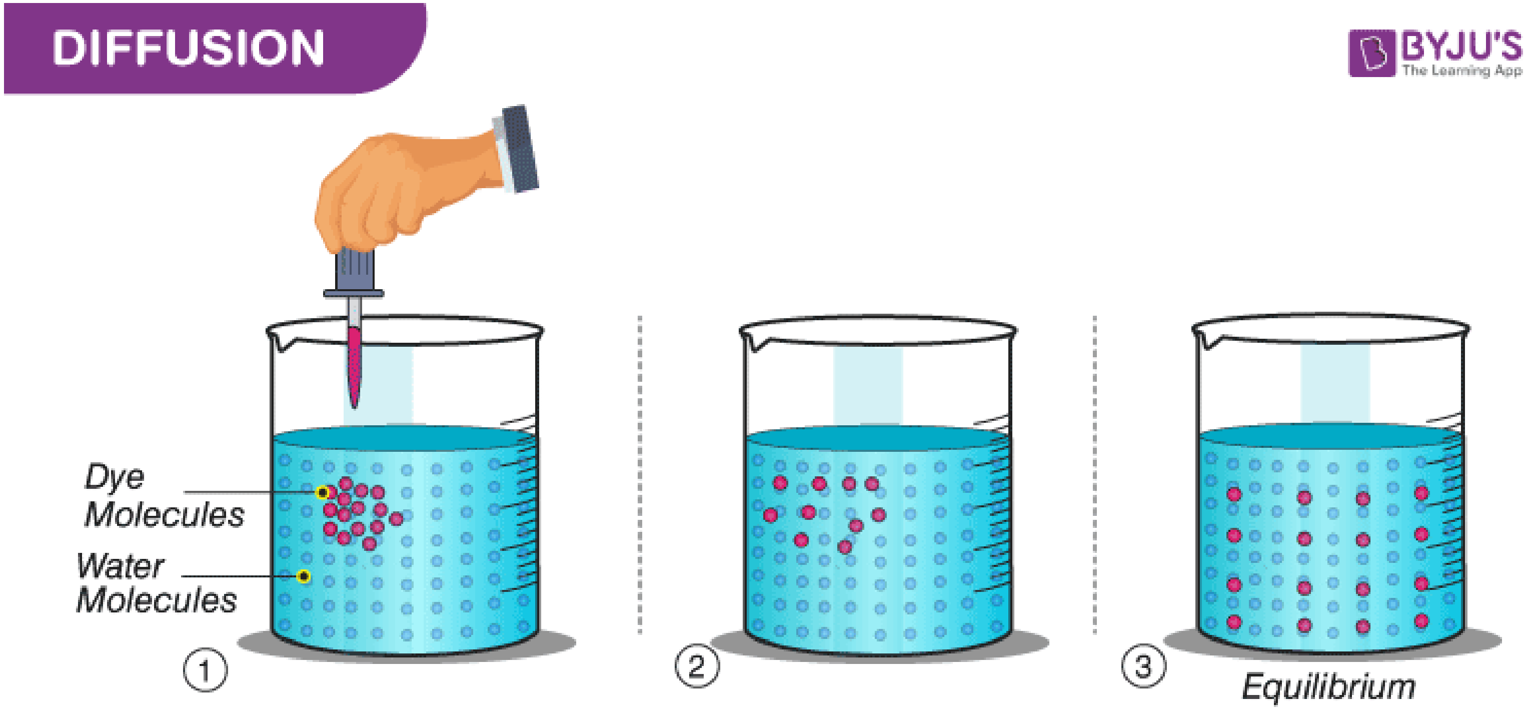
$\frac{dP(t)}{dt} = P(t)Q$ is called backward Kolmogorov equation.

$\frac{dP(t)}{dt} = QP(t)$ is called forward Kolmogorov equation.

This means that for infinitesimal t, reverse Markov process can be described in same functional form.

콜모고로프 미분방정식 유도 (freshrimpsushi.github.io)

Deep Unsupervised Learning using Nonequilibrium Thermodynamics

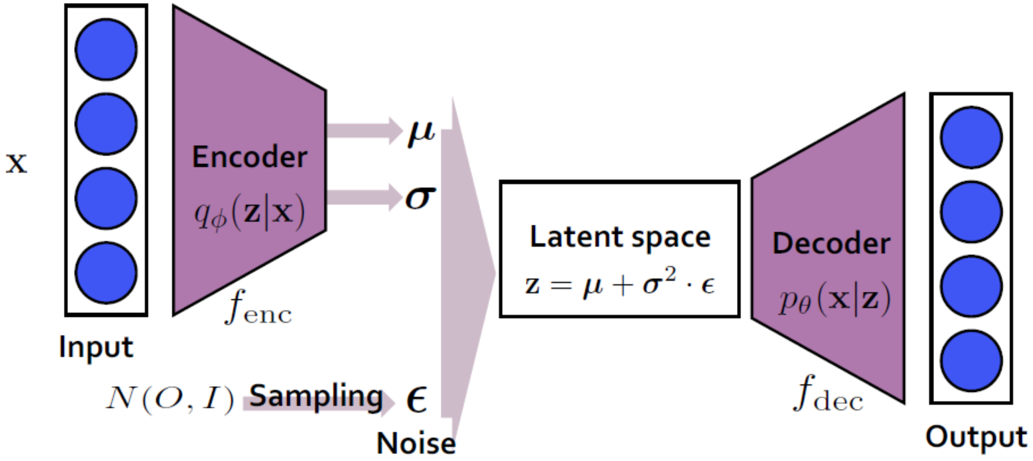


What Is Diffusion? - Definition, Types & Examples Of Diffusion (byjus.com)

GAN is unstable, highly dependent on hyperparameters, and not probability distribution.

VAE is stable, but generated sample is too blurry, because of ELBO maximization, and tends to utilize only a piece of latent space. (from Goodfellow, (2016))

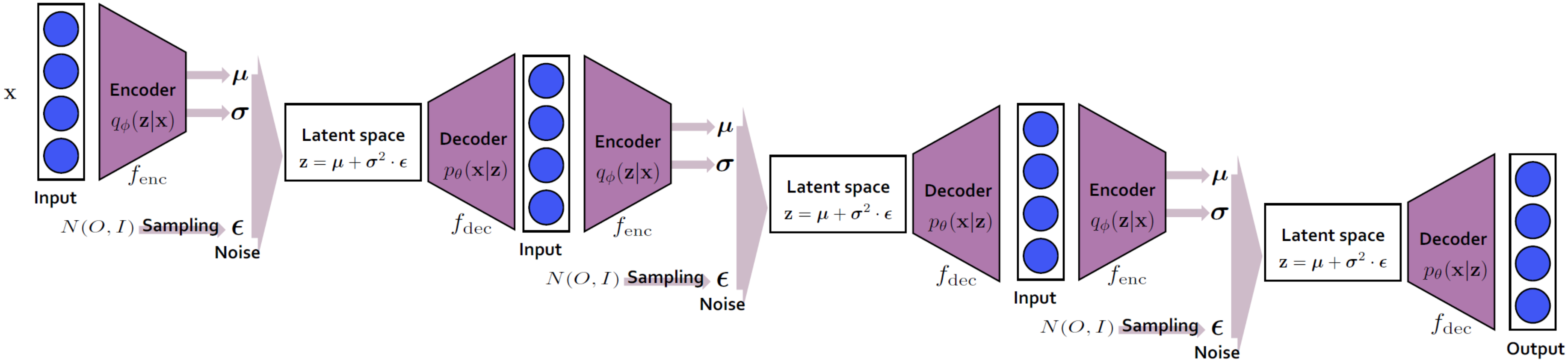
Idea*: Multi-step VAE looks like Monte Carlo Markov chain!



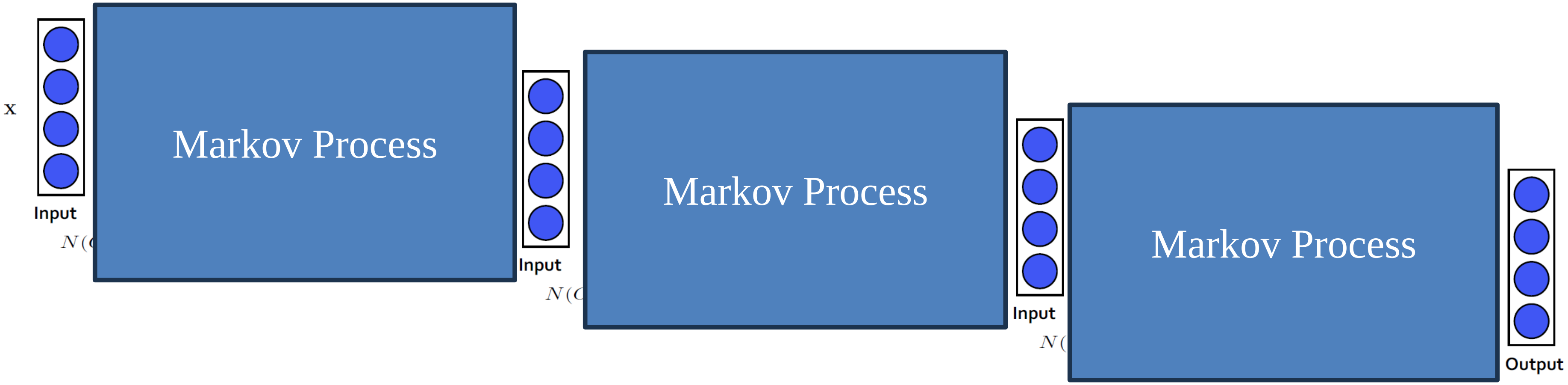
Goodfellow, I. (2016). Deep learning.

VAE 설명 (Variational autoencoder란? VAE ELBO 증명) - 유니의 공부 (tistory.com)

Introduction Cont'd



Introduction Cont’d



Multi-step VAE looks like Monte Carlo Markov chain!

Introduction Cont’d

We can interpret diffusion models as AIS over VAE generation: generating complex input distribution by shifting prior distribution, with restricted complexity. This method constraints the function family of neural network.

Model and Forward Trajectory

Data distribution: $q(\boldsymbol{x}^{(0)})$

Prior distribution: $\pi(\boldsymbol{y})$

Markov diffusion kernel

$$\pi(\boldsymbol{y}) = \int d\boldsymbol{y}' T_\pi(\boldsymbol{y}|\boldsymbol{y}'; \beta)\pi(\boldsymbol{y}') \quad \text{where } \beta \text{ is a diffusion rate.}$$

$$q\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)}\right) = T_\pi\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)}; \beta_t\right)$$

Joint distribution:

$$q(\{\boldsymbol{x}\}_{t=0}^{T}) = q(\boldsymbol{x}_0)\prod_{t=1}^{T} q\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)}\right)$$

Reverse Trajectory

Prior distribution: $p\left(\boldsymbol{x}^{(T)}\right) = \pi\left(\boldsymbol{x}^{(T)}\right)$

Joint distribution

$$p\left(\left\{\boldsymbol{x}^{(t)}\right\}_{t=1}^{T}\right) = p\left(\boldsymbol{x}^{(T)}\right)\prod_{t=1}^{T} p(\boldsymbol{x}^{t-1}|\boldsymbol{x}^{t})$$

In the real learning, we only need to estimate the mean and the variance of the distribution: (Feller, 1949) guarantees that the forms of the forward and backward process are equal in infinitesimal. If the forward process is gaussian, the reverse also is.

Generative Distribution

The probability of a data is

$$p\left(\boldsymbol{x}^{(0)}\right) = \int d\boldsymbol{x}^{(1..T)} p\left(\boldsymbol{x}^{(0..T)}\right).$$

But can we calculate it?

1. Integrate it exactly. (every $p(x^t)$ is separable, so we can disassemble $p\left(x^{(0..T)}\right)$.)
2. Approximate it by Riemann sum.

But both ways are not feasible in this case. We need another option:

AIS gives us the way to approximate the marginal probability!

Generative Distribution Cont’d

$$p\left(\boldsymbol{x}^{(0)}\right) = \int d\boldsymbol{x}^{(1..T)} p\left(\boldsymbol{x}^{(0..T)}\right)$$

$$= \int d\boldsymbol{x}^{(1..T)} p\left(\boldsymbol{x}^{(0..T)}\right) \frac{q\left(\boldsymbol{x}^{(1..T)} \middle| \boldsymbol{x}^{(0)}\right)}{q\left(\boldsymbol{x}^{(1..T)} \middle| \boldsymbol{x}^{(0)}\right)}$$

$$= \int d\boldsymbol{x}^{(1..T)} q\left(\boldsymbol{x}^{(1..T)} \middle| \boldsymbol{x}^{(0)}\right) \frac{p\left(\boldsymbol{x}^{(0..T)}\right)}{q\left(\boldsymbol{x}^{(1..T)} \middle| \boldsymbol{x}^{(0)}\right)}$$

$$= \int d\boldsymbol{x}^{(1..T)} q\left(\boldsymbol{x}^{(1..T)} \middle| \boldsymbol{x}^{(0)}\right) p\left(\boldsymbol{x}^{(T)}\right) \prod_{t=1}^{T} \frac{p\left(\boldsymbol{x}^{(t-1)} \middle| \boldsymbol{x}^{(t)}\right)}{q\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}\right)}$$

$$= \mathbb{E}_{q\left(\boldsymbol{x}^{(1..T)}\right)} \left[p\left(\boldsymbol{x}^{(T)}\right) \prod_{t=1}^{T} \frac{p\left(\boldsymbol{x}^{(t-1)} \middle| \boldsymbol{x}^{(t)}\right)}{q\left(\boldsymbol{x}^{(t)} \middle| \boldsymbol{x}^{(t-1)}\right)} \right]$$

Training

$$\mathrm{argmin}_p\ D_{KL}(q||p)?$$

$$D_{KL}(q||p) = \mathbb{E}\left[q\log\frac{1}{p}\right] - \mathbb{E}\left[q\log\frac{1}{q}\right]$$

$$= \mathrm{CE}(q,p) - \mathrm{H}(q)$$

$$\mathrm{argmin}_p D_{KL}(q||p) = \mathrm{argmax}_p[-\mathrm{CE}(q,p)]$$

We will maximize negative cross entropy of q about p.

$$\mathrm{CE}(q,p) = \int dx^{(0)} q\left(x^{(0)}\right)\log p\left(x^{(0)}\right)$$

$$= \int dx^{(0)} q\left(x^{(0)}\right)\log\int d\boldsymbol{x}^{(1..T)} q\left(\boldsymbol{x}^{(1..T)}\middle|\boldsymbol{x}^{(0)}\right) p\left(\boldsymbol{x}^{(T)}\right)\prod_{t=1}^{T}\frac{p\left(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)}\right)}{q\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)}\right)}$$

$$\geq \int d\boldsymbol{x}^{(0..T)} q\left(\boldsymbol{x}^{(0..T)}\right)\log\left[p\left(\boldsymbol{x}^{(T)}\right)\prod_{t=1}^{T}\frac{p\left(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)}\right)}{q\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)}\right)}\right]$$

$$\because \log\ \text{function is concave, jensen inequality.}$$

Training Cont’d

$$\mathrm{CE}(q,p) \geq \int d\boldsymbol{x}^{(0..T)} q\left(\boldsymbol{x}^{(0..T)}\right) \log\left[p\left(\boldsymbol{x}^{(T)}\right) \prod_{t=1}^{T} \frac{p\left(\boldsymbol{x}^{(t-1)}|\boldsymbol{x}^{(t)}\right)}{q\left(\boldsymbol{x}^{(t)}|\boldsymbol{x}^{(t-1)}\right)}\right]$$

$$= -\sum_{t=2}^{T} \int dx^{(0)} dx^{(t)} q\left(x^{(0)}, x^{(t)}\right) \cdot D_{KL}\left(q\left(x^{(t-1)}\middle|x^{(t)}, x^{(0)}\right)\middle|\middle|p\left(x^{(t-1)}\middle|x^{(t)}\right)\right)$$

$$+H_q\left(X^{(T)}\middle|X^{(0)}\right) - H_q\left(X^{(1)}\middle|X^{(0)}\right) - H_p\left(X^{(T)}\right)$$

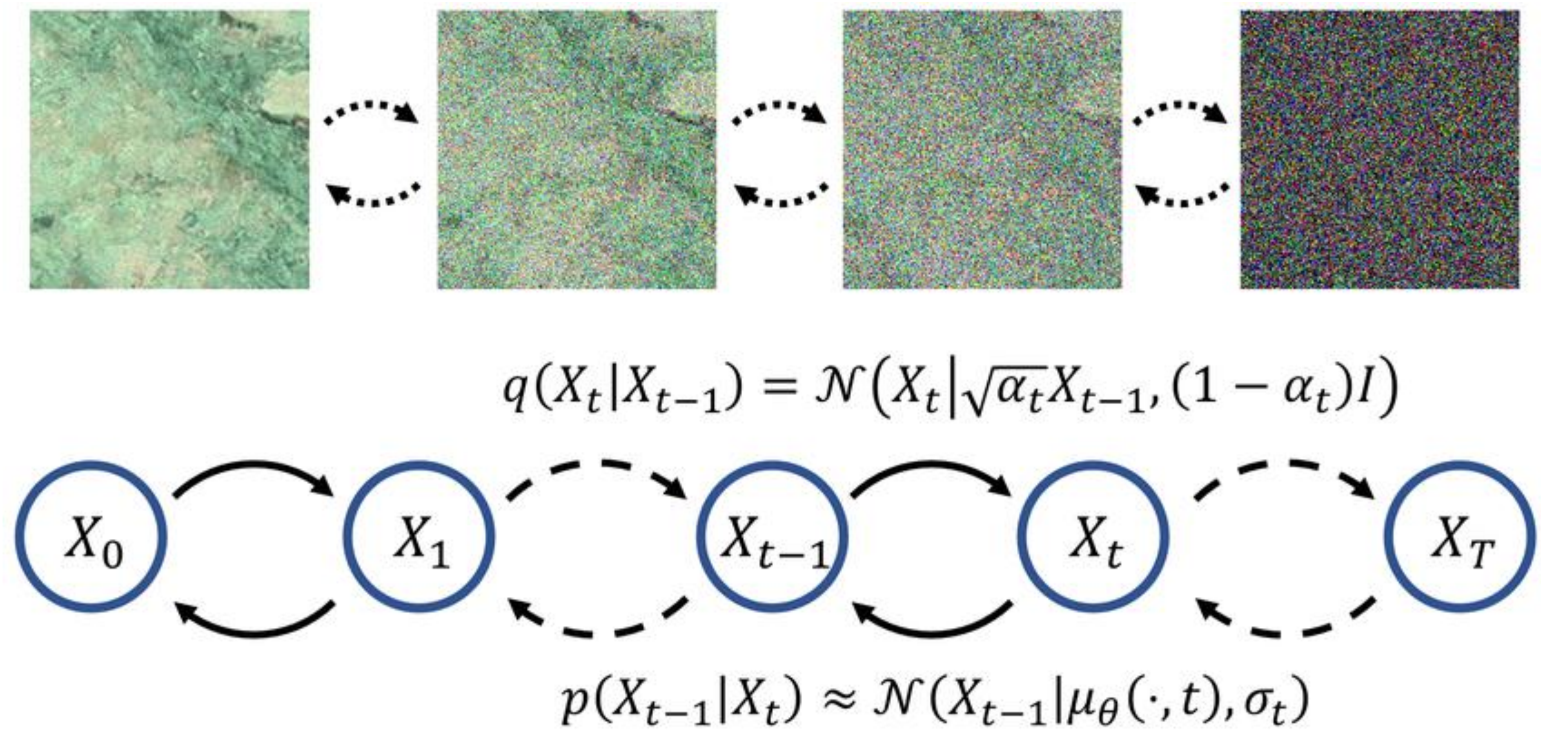
Training Cont’d

The choice of β_t is very important for the performance of AIS. In diffusion model, β_t is learned by gradient ascent.

Implementation

[Papers-in-100-Lines-of-Code/Deep_Unsupervised_Learning_using_Nonequilibrium_Thermodynamics/diffusion_models.py at main · MaximeVandegar/Papers-in-100-Lines-of-Code (github.com)]

Denoising Diffusion Probabilistic Model


$q(X_t|X_{t-1}) = \mathcal{N}\left(X_t\middle|\sqrt{\alpha_t}X_{t-1}, (1-\alpha_t)I\right)$
X_0
X_1
X_{t-1}
X_t
X_T
$p(X_{t-1}|X_t) \approx \mathcal{N}(X_{t-1}|\mu_\theta(\cdot, t), \sigma_t)$

Denoising diffusion probabilistic model. From left to right corresponds... | Download Scientific Diagram (researchgate.net)

Implementation

Papers-in-100-Lines-of-Code/Denoising_Diffusion_Probabilistic_Models/diffusion_models.py at main · MaximeVandegar/Papers-in-100-Lines-of-Code (github.com)

Discussion