KSBi-BIML 2026

Bioinformatics & Machine Learning(BIML) Workshop for Life Scientists

생명정보학 & 머신러닝 워크샵 (온라인)

Diffusion Models – 이해와 응용

노영균 _ 한양대학교

KSBI KOREAN SOCIETY FOR BIOINFORMATICS | 한국생명정보학회

KSBi-BIML 2026

Bioinformatics & Machine Learning (BIML) Workshop for Life Scientists

한국생명정보학회가 주최하는 BIML-2026 동계 Bioinformatics &Machine Learning 교육 워크숍에 여러분을 초대합니다.

BIML 워크숍은 생명정보학 연구자들이 최신 AI바이오 분야의 인공지능 기반 분석 기술과 바이오 데이터 분석 기법을 이론과 실습을 통해 체계적으로 배울 수 있는 전문 교육 프로그램입니다. 2015년에 시작된 BIML 워크숍은 올해로 12년 차를 맞이하며, 국내 생명정보학 분야의 최초이자 최고 수준의 교육 프로그램으로 자리 잡았습니다. 이번 워크숍은 크게 인공지능바이오(AI바이오) 분야와 디지털바이오 분야, 두 분야로 구성됩니다.

AI바이오 분야에서는 생명정보 분석에 폭넓게 응용되고 있는 다양한 인공지능 기반 자료 모델링 기법을 다룰 예정입니다. 특히, 인공지능 심층학습을 활용한 단백질 구조 예측, 유전체 분석, 신약 개발에 대한 이론 및 실습 강의가 진행됩니다.

또한 디지털바이오 분야에서는 단일세포오믹스, 공간오믹스, 멀티오믹스, 메타오믹스에 대한 강의도 마련되어 있어, 연구자들의 분석 역량 강화에 실질적인 도움을 줄 것으로 기대됩니다.

또한 2024년부터 추가된 의료정보 자료 분석을 다루는 강의를 올해도 지속해서 운영하고자 합니다. 이는 최근 의료정보 자료 분석에 관한 연구 수요 증가를 반영한 것으로, 관련 연구를 수행하는 의과학자 및 의료정보 연구자들에게 유용한 지침을 제공할 것입니다.

또한, 올해도 생명정보학 기술의 다양화에 발맞춰 온라인 강좌를 대폭 확대했습니다. 올해는 무료 강좌 10개를 포함한 총 40개 이상의 강좌가 개설되며, 연구 주제에 맞는 강좌 추천과 강연료 할인 혜택도 제공합니다.

BIML-2026는 국내 주요 연구 중심 대학의 전임 교수 및 각 분야 최고 전문가들의 강의로 구성되어 있으며, 기초 이론부터 최신 연구 동향까지 아우르는 심도 있는 교육의 장이 될 것으로 확신합니다.

여러분의 많은 관심과 참여를 기대합니다!

2026년 2월

한국생명정보학회장 류 성 호

강의개요

Diffusion Models - 이해와 응용

최근의 생성모델의 성능 향상은 새로운 형태의 인공지능 응용 가능성을 보여주고 있다. 본 강의에서는 생성모델 가운데 하나인 Diffusion 모델을 설명한다. 가우시안을 통한 추론 방법의 이해에서 시작하여 Diffusion 모델의 수식을 이해하고, diffusion 모델이 다른 생성 모델과 근본적으로 어떻게 다른지에 대한 논의를 제공할 예정이다. 간단한 실습을 통해 diffusion 모델이 어떻게 작동하는지 살펴본다.

- Diffusion model 개요 및 다른 생성모델과의 개념 비교
- Diffusion 작용과 역작용을 위한 노이즈 예측
- 데이터 생성의 간단한 실습

* 참고논문:

1. Jonathan Ho, Ajay Jain, Pieter Abbeel (2020) Denoising diffusion probabilistic models, *Advances in Neural Information Processing Systems 33*
2. Jonathan Ho, Tim Salimans (2022) Classifier-Free Diffusion Guidance, *arXiv:2207.12598*

* 교육생준비물: 노트북 (웹브라우저로 구글 CoLab을 실행시킬 수 있는 노트북)

* 강의 난이도: 중급

* 강의: 노영균 교수 (한양대학교 컴퓨터소프트웨어학부 / 고등과학원 계산과학부)

Curriculum Vitae

Speaker Name: Yung-Kyun Noh, Ph.D.

▶ Personal Info

Name	Yung-Kyun Noh
Title	Department Chair, Associate Professor
Affiliation	1. Department of Artificial Intelligence, Hanyang University 2. Department of Computer Science, Hanyang University 3. School of Computational Sciences, Korea Institute for Advanced Study

▶ Contact Information

Address	605 ITBT, 222 Wangsimni-ro, Seongdong-gu, Seoul 04763, Rep. of Korea
Email	nohyung@hanyang.ac.kr

Research Interest

Machine Learning, Nonparametric methods, Information theory

Educational Experience

1998	B.S. in Physics, POSTECH, Rep. of Korea
2011	Ph.D. in Computer Science (Interdisciplinary Program in Cognitive Science), Seoul National University, Rep. of Korea

Professional Experience

2007-2012	Visiting Scholar, Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, U.S.A.
2019-2021	Assistant Professor, Department of Computer Science, Hanyang University, Seoul, Korea
2019-2021	Associate Member, School of Computational Sciences, Korea Institute for Advanced Study, Seoul, Korea
2020-2021	Visiting Scientist, Gastroenterology and Hepatology, Mayo Clinic, Rochester, MN, USA
2018-	Visiting Scientist, RIKEN Center for Advanced Intelligence Project (AIP), Tokyo, Japan
2021-	Affiliate Professor, School of Computational Sciences, Korea Institute for Advanced Study, Seoul, Korea
2021-	Associate Professor, Department of Computer Science, Hanyang University, Seoul, Korea
2022-	Research Collaborator, Gastroenterology and Hepatology, Mayo Clinic, Rochester, MN, USA
2023-	Chair, Dept. of Artificial Intelligence, Hanyang University, Korea

Selected Publications (5 maximum)

1. Lee, J.-W., Won, J.-H., Jeon, S., Choo, Y., Yeon, Y., Oh, J.-S., Kim, M., Kim, S., Joung, I., Jang, C., Lee, S. J., Kim, T. H., Jin, K. H., Song, G., Kim, E.-S., Yoo, J., Paek, E., Noh, Y.-K., Joo, K. (2023) DeepFold: Enhancing Protein Structure Prediction Through Optimized Loss Functions, Improved Template Features, and Re-optimized Energy Function, *Bioinformatics, 39:12, btad712*
2. Yoon, S., Park, F. C., Yun, G. Kim, S., I., Noh, Y.-K. (2023) Variational Weighting for Kernel Density Ratios, *Advances in Neural Information Processing Systems 36 (NeurIPS)*
3. Yoon, S., Jin, Y.-U., Noh, Y.-K., Park, F. C. (2023) Energy-Based Models for Anomaly Detection: A Manifold Diffusion Recovery Approach, *Advances in Neural Information Processing Systems 36 (NeurIPS)*
4. Jang, C., Lee, S., F. C. Park, Y.-K. Noh (2022) A Reparametrization-Invariant Sharpness Measure Based on Information Geometry, *Advances in Neural Information Processing Systems 35 (NeurIPS)*
5. Lee, H., Lee, J., Choi, Y., Jeon, W., Lee, B.-J., Noh, Y.-K., Kim, K.-E. (2022) Local Metric Learning for Off-Policy Evaluation in Contextual Bandits with Continuous Actions, *Advances in Neural Information Processing Systems 35 (NeurIPS)*

KSBI | 한국생명정보학회

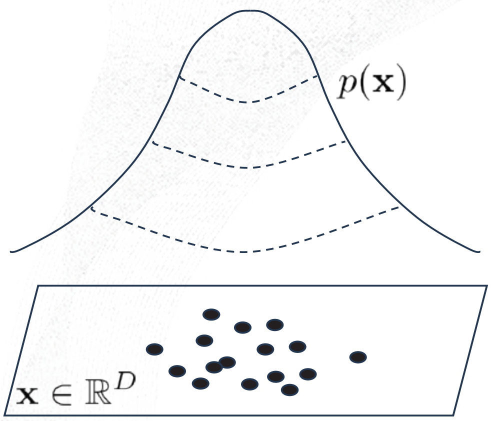
KSBi-BIML 2024

Diffusion Models - 이해와 응용

노영균 (한양대학교 & 고등과학원)
nohyung@hanyang.ac.kr

Data generation

$\mathbf{x} \sim p(\mathbf{x})$



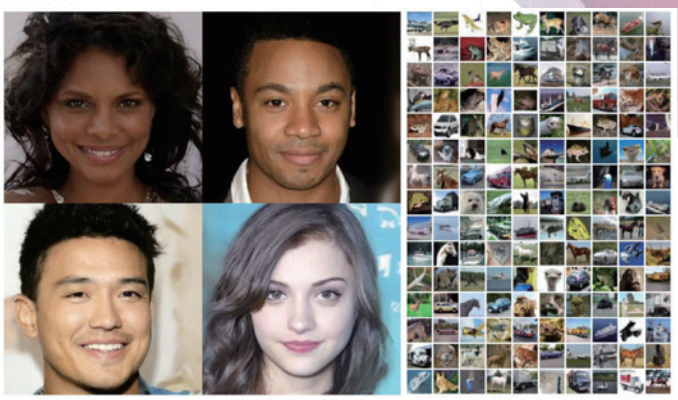


Figure 1: Generated samples on CelebA-HQ 256 × 256 (left) and unconditional CIFAR10 (right)

$\mathbf{x} \in \mathbb{R}^{\text{Pixel}}$

2

Data generation 101

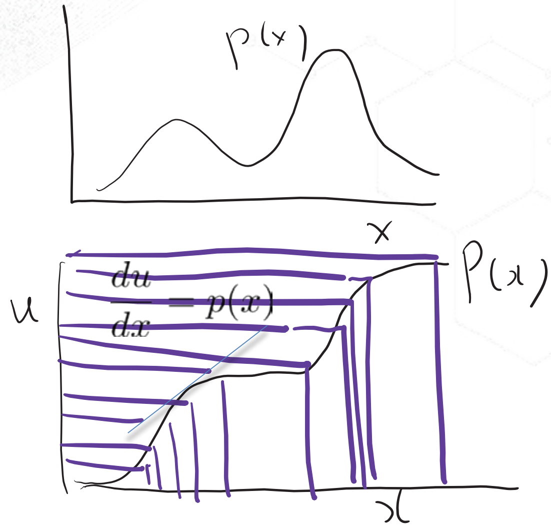
Cumulative distribution

$$P(x) = \int_{-\infty}^{x} p(x)dx \equiv u$$

$$x = P^{-1}(u)$$

$$du = p(x)dx$$

$$u \sim \mathrm{Unif}(0,1)$$



$$\int_{-\infty}^{u} p(u)du = \int_{0}^{u} du = \int_{-\infty}^{P^{-1}(u)} p(x)dx, \quad 0 \le u \le 1$$

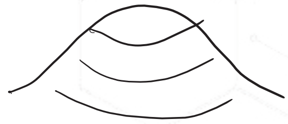
Flow-based model (Normalizing flow)

$p(\mathbf{x})d\mathbf{x} = p(\mathbf{u})d\mathbf{u}$, $\mathbf{x} = T(\mathbf{u})$ (full-rank transformation)

$$p(\mathbf{x}) = p(\mathbf{u}) \left| \frac{d\mathbf{u}}{d\mathbf{x}} \right|$$

$$= p(\mathbf{u}) \det J_T^{-1}(\mathbf{u})$$

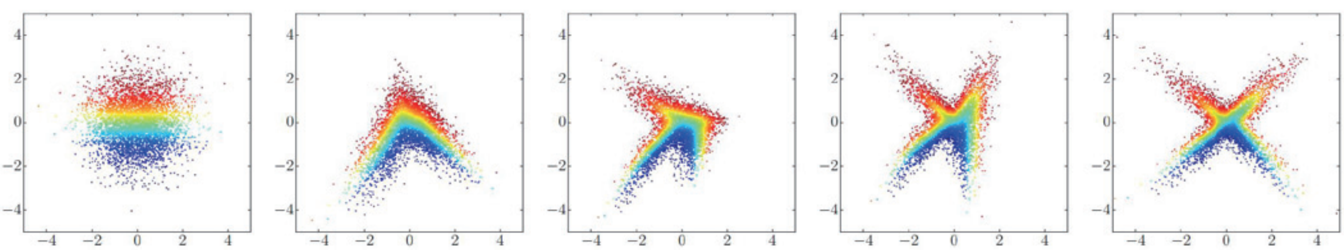
Base distribution $p_{\mathbf{u}}(\mathbf{u})$



Flow – Gradually transformed by the sequence of (consecutive) transformations $T_1, \ldots, T_K$

$$T = T_K \cdot T_{K-1} \cdots T_1$$

Normalizing – the inverse flow $T_K^{-1}, \ldots, T_1^{1}$



G. Papamakarios et al. JMLR 2021

Flow-based model (Normalizing flow)

Learning

Data $\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{N} \sim p(\mathbf{x})$

Prevent compression from x to u

$$\det J_T(\mathbf{u}_i) = \left|\frac{d\mathbf{x}}{d\mathbf{u}}\right|_{\mathbf{u}}$$

Maximum Likelihood

Minimize

Gaussian

$$L = \sum_{i=1}^{N} -\log p(\mathbf{x}_i) = \sum_{i=1}^{N} -\log p_{\mathbf{u}}(\mathbf{u}_i) - \log \det J_{T^{-1}}(\mathbf{u}_i)$$

$$\mathbf{u}_i = T^{-1}(\mathbf{x}_i)$$

Likelihood representation with respect to $p(\mathbf{u})$

The loss is sensitive to the volume change of transformation due to the determinant of Jacobian.

5

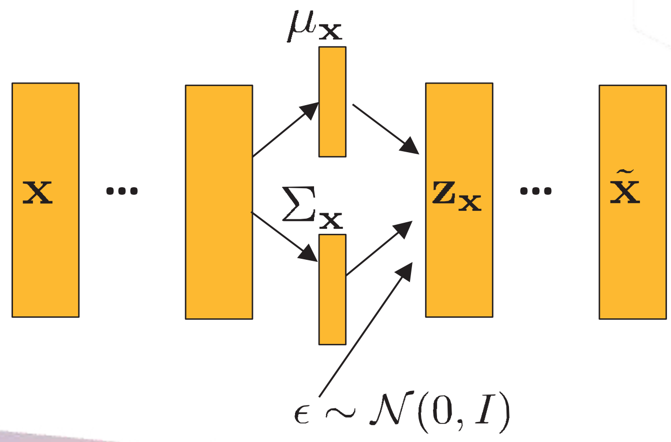
Deterministic flow:

Reversed flow is "exactly" the backward flow

Variational Autoencoder

$$L(\mathbf{x}) = \log p(\mathbf{x}|\mathbf{z_x}) - KL(q_\mathbf{x}(\mathbf{z})||p(\mathbf{z}))$$
$$= -||\mathbf{x} - \tilde{\mathbf{x}}(\mathbf{z_x})||^2 - KL(\mathcal{N}(\mu_\mathbf{x}, \Sigma_\mathbf{x})||\mathcal{N}(0, I))$$

Neural Networks (Decoder) ($\tilde{\mathbf{x}}$) *Neural Networks (Encoder)* ($\mu_\mathbf{x}$, $\Sigma_\mathbf{x}$)



Variational Autoencoder

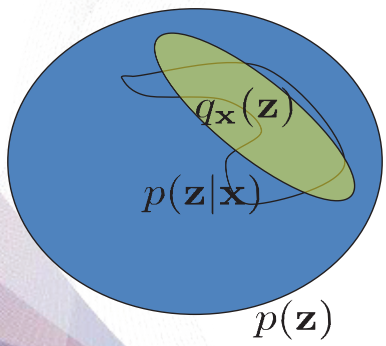
$p(\mathbf{x})$ $p(\mathbf{z}|\mathbf{x})$ ← Some appropriate mapping

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

$p(\mathbf{x}|\mathbf{z})$
$= \dfrac{p(\mathbf{x})p(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})}$

$q_\mathbf{x}(\mathbf{z})$: Gaussian
$= \mathcal{N}(\mu_\mathbf{x}, \Sigma_\mathbf{x})$

$\mathbf{z} \in \mathbb{R}^{D_\mathbf{z}} \rightarrow$
$\mu_\mathbf{x} \in \mathbb{R}^{D_\mathbf{z}}, \Sigma_\mathbf{x} \in \mathbb{R}^{D_\mathbf{z} \times D_\mathbf{z}}$



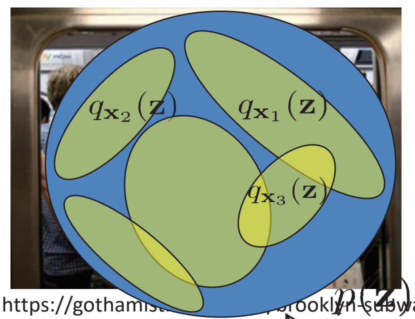
Reparametrization from $q_\mathbf{x}(\mathbf{z})$

Optimize

$||\mathbf{x} - f(\mathbf{z})||^2 \downarrow$

$KL(q_\mathbf{x}, p(\mathbf{z})) \downarrow$

Then

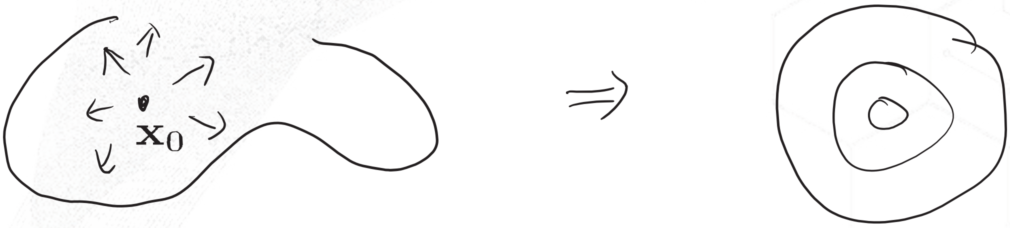


https://gothamist... brooklyn-subway-trains-actually-less-crowded-than-they-appear

Diffusion models

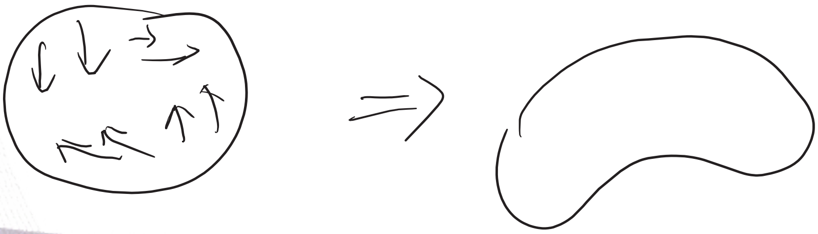
- Forward process

  Pick up $\mathbf{x}_0 \sim p(\mathbf{x})$, then randomly move



- Reverse process

  How does backward process make flow?

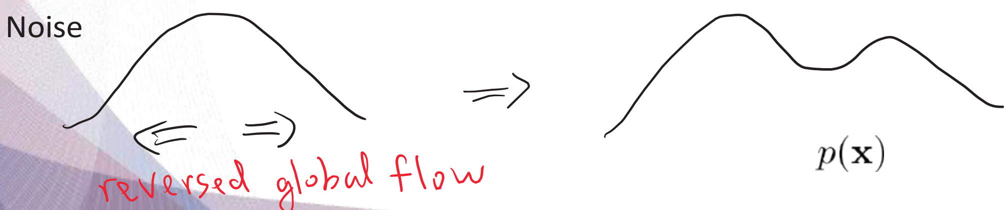


9

Diffusion models

Diffusion of Non-uniform density makes a global flow

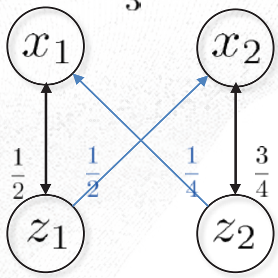
Reverse process in diffusion model reconstructs the backward global flow.



10

Forward-Reverse process

$p(x = x_1) = \frac{1}{3}$ $p(x = x_2) = \frac{2}{3}$



$p(z = z_1) = \frac{1}{3}$ $p(z = z_2) = \frac{2}{3}$

Forward process

$p(z_1|x_1) = 1$

$p(z_2|x_2) = 1$

Reverse process

$p(x_1|z_1) = 1$

$p(x_2|z_2) = 1$

Any process that preserves marginal will work. Do not consider joint density over x and z.

Reverse process

$p(x_1|z_1) = \frac{1}{2}$ $p(x_2|z_1) = \frac{1}{2}$

$p(x_2|z_2) = \frac{3}{4}$ $p(x_2|z_2) = \frac{1}{4}$

⋮

11

Underlying diffusion procedure

$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$

Data

$\mathbf{x}_0 \sim q(\mathbf{x}_0)$

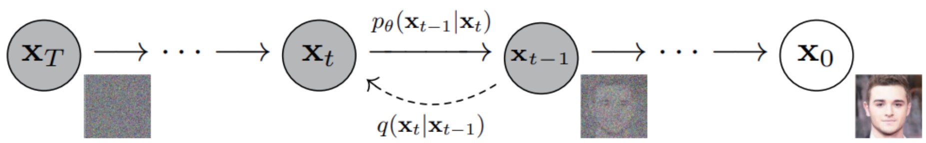


Figure 2: The directed graphical model considered in this work.

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) := \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \qquad q(\mathbf{x}_t|\mathbf{x}_{t-1}) := \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

$\beta_t > 0$

$q(\mathbf{x}_{1:T}|\mathbf{x}_0),\ q(\mathbf{x}_t|\mathbf{x}_{t-1})$: Gaussians

Caution) $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$: Not Gaussian

If $p(\mathbf{x}_0)$ is Gaussian, $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is Gaussian.

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

: Gaussian mixture

12

Model for reverse process

$$p_\theta(\mathbf{x}_{0:T}) := p(\mathbf{x}_T)\prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \qquad p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$$

Objective function:

$$\mathbb{E}\left[-\log p_\theta(\mathbf{x}_0)\right] \le \mathbb{E}_q\left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right]$$

$$= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\ge 1}\log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right] =: L$$

Look at the derivations in the next two pages…

13

Objective function - 1

$$L = \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\ge 1}\log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right]$$

$$= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\ge 1}\log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t)}\cdot\frac{q(\mathbf{x}_{t-1})}{q(\mathbf{x}_t)}\right]$$

$$= \mathbb{E}_q\left[-\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T)} - \sum_{t\ge 1}\log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t)} - \log q(\mathbf{x}_0)\right]$$

$$= D_{\mathrm{KL}}(q(\mathbf{x}_T)\,\|\,p(\mathbf{x}_T)) + \mathbb{E}_q\left[\sum_{t\ge 1} D_{\mathrm{KL}}(\underline{q(\mathbf{x}_{t-1}|\mathbf{x}_t)}\,\|\,p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))\right] + H(\mathbf{x}_0)$$

Can we have this density function?

14

Objective function - 2

$$L = \mathbb{E}_q\left[-\log \frac{p_\theta(\mathbf{x}_{0:T})}{q(\mathbf{x}_{1:T}|\mathbf{x}_0)}\right]$$

$$= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t\geq 1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})}\right]$$

$$= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t>1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_t|\mathbf{x}_{t-1})} - \log \frac{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)}\right]$$

Gaussians
∵ x₀ given

$$= \mathbb{E}_q\left[-\log p(\mathbf{x}_T) - \sum_{t>1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)} \cdot \frac{q(\mathbf{x}_{t-1}|\mathbf{x}_0)}{q(\mathbf{x}_t|\mathbf{x}_0)} - \log \frac{p_\theta(\mathbf{x}_0|\mathbf{x}_1)}{q(\mathbf{x}_1|\mathbf{x}_0)}\right]$$

$$= \mathbb{E}_q\left[-\log \frac{p(\mathbf{x}_T)}{q(\mathbf{x}_T|\mathbf{x}_0)} - \sum_{t>1} \log \frac{p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)}{q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)} - \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)\right]$$

$$= \mathbb{E}_q\Big[\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_T|\mathbf{x}_0) \,\|\, p(\mathbf{x}_T))}_{L_T} + \underbrace{\sum_{t>1} D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) \,\|\, p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}\Big] \quad (22)$$

15

Tractable functions

$$\left.\begin{array}{c} q(\mathbf{x}_{t-1}|\mathbf{x}_0) \\ q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) \\ q(\mathbf{x}_t|\mathbf{x}_0) \end{array}\right\} \text{Gaussians}$$

16

Decomposition for Gaussian inference

$$p(\mathbf{x}_a, \mathbf{x}_b)$$
$$= \frac{1}{\sqrt{2\pi}^D \left| \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix} \right|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \begin{pmatrix} \mathbf{x}_a - \mu_a \\ \mathbf{x}_b - \mu_b \end{pmatrix}^\top \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{x}_a - \mu_a \\ \mathbf{x}_b - \mu_b \end{pmatrix} \right)$$

$$\mu_{a|b} = \Sigma_{ab}\Sigma_b^{-1}(\mathbf{x}_b - \mu_b)$$

$$= C \exp\left(-\frac{1}{2} \left(\mathbf{x}_a - \Sigma_{ab}\Sigma_b^{-1}(\mathbf{x}_b - \mu_b)\right)^\top \left(\Sigma_a - \Sigma_{ab}\Sigma_b^{-1}\Sigma_{ba}\right)^{-1} \left(\mathbf{x}_a - \Sigma_{ab}\Sigma_b^{-1}(\mathbf{x}_b - \mu_b)\right) -\frac{1}{2} (\mathbf{x}_b - \mu_b)^\top \Sigma_b^{-1} (\mathbf{x}_b - \mu_b) \right)$$

$$\Sigma_{a|b} = \Sigma_a - \Sigma_{ab}\Sigma_b^{-1}\Sigma_{ba}$$

$$= C \exp\left(-\frac{1}{2}(\mathbf{x}_a - \mu_{a|b})^\top \Sigma_{a|b}^{-1} (\mathbf{x}_a - \mu_{a|b}) -\frac{1}{2}(\mathbf{x}_b - \mu_b)^\top \Sigma_b^{-1} (\mathbf{x}_b - \mu_b) \right) = p(\mathbf{x}_a|\mathbf{x}_b)\ p(\mathbf{x}_b)$$

17

Decomposition for Gaussian inference

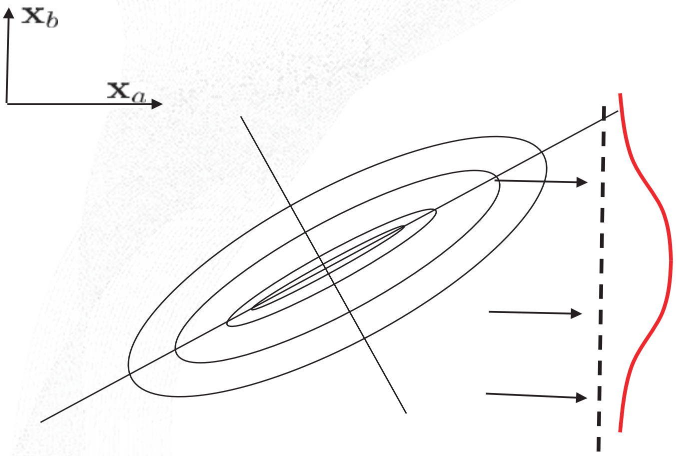
$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix} \quad \begin{matrix} \mathbf{x}_a \in \mathbb{R}^{D_a} \\ \mathbf{x}_b \in \mathbb{R}^{D_b} \end{matrix} \qquad \mu = \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix} \quad \Sigma = \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix}$$

$$p(\mathbf{x}) = \frac{1}{\sqrt{2\pi}^D |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^\top \Sigma^{-1} (\mathbf{x} - \mu) \right)$$
$$= C_1 \exp\left(-\frac{1}{2}(\mathbf{x}_a - \mu_{a|b}(\mathbf{x}_b))^\top \Sigma_{a|b}^{-1} (\mathbf{x}_a - \mu_{a|b}(\mathbf{x}_b)) \right) \cdot C_2 \exp\left(-\frac{1}{2}(\mathbf{x}_b - \mu_b)^\top \Sigma_b^{-1} (\mathbf{x}_b - \mu_b) \right)$$

$$p(\mathbf{x}) = p(\mathbf{x}_a, \mathbf{x}_b) = p(\mathbf{x}_a|\mathbf{x}_b)p(\mathbf{x}_b)$$

18

Gaussian random variable - marginalization



$$p(\mathbf{x}_b) = \int p(\mathbf{x}_a, \mathbf{x}_b) d\mathbf{x}_a$$
$$= \int p(\mathbf{x}_a | \mathbf{x}_b) p(\mathbf{x}_b) d\mathbf{x}_a$$
$$= \mathcal{N}(\mu_b, \Sigma_b)$$

Gaussian random variable - marginalization

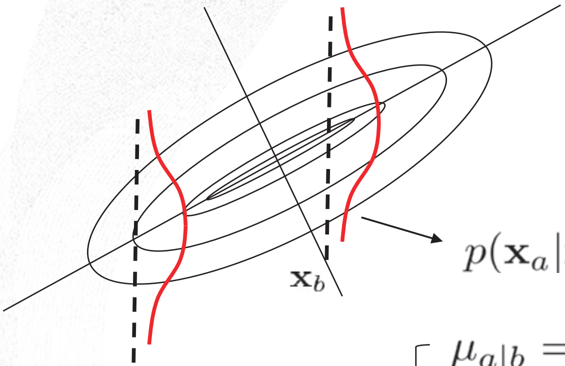
$$p(\mathbf{x}) = \frac{1}{\sqrt{2\pi}^D |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^\top \Sigma^{-1} (\mathbf{x}-\mu)\right)$$

$$p(\mathbf{x}_a, \mathbf{x}_b)$$
$$= \frac{1}{\sqrt{2\pi}^D \left| \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix} \right|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} \begin{pmatrix} \mathbf{x}_a - \mu_a \\ \mathbf{x}_b - \mu_b \end{pmatrix}^\top \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ba} & \Sigma_b \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{x}_a - \mu_a \\ \mathbf{x}_b - \mu_b \end{pmatrix}\right)$$

$$\int p(\mathbf{x}_a, \mathbf{x}_b) d\mathbf{x}_b = \frac{1}{\sqrt{2\pi}^{D_a} |\Sigma_a|^{\frac{1}{2}}} \exp\left(-\frac{1}{2} (\mathbf{x}_a - \mu_a)^\top \Sigma_a^{-1} (\mathbf{x}_a - \mu_a)\right)$$
$$= \mathcal{N}(\mu_a, \Sigma_a)$$

Gaussian random variable - conditioning

$$p(\mathbf{x}) = \frac{1}{\sqrt{2\pi}^D |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^\top \Sigma^{-1} (\mathbf{x}-\mu)\right)$$

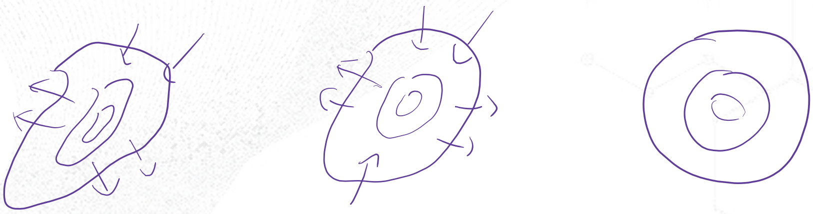


$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{pmatrix} \quad \begin{matrix} \mathbf{x}_a \in \mathbb{R}^{D_a} \\ \mathbf{x}_b \in \mathbb{R}^{D_b} \end{matrix}$$

$$p(\mathbf{x}_a | \mathbf{x}_b) = \mathcal{N}(\mu_{a|b}, \Sigma_{a|b})$$

$$\begin{cases} \mu_{a|b} = \mu_a + \Sigma_{ab}\Sigma_b^{-1}(\mathbf{x}_b - \mu_b) \\ \Sigma_{a|b} = \Sigma_a - \Sigma_{ab}\Sigma_b^{-1}\Sigma_{ba} \end{cases}$$

Diffusion and reverse process



$$x_t = \sqrt{1-\beta_t}\, x_{t-1} + \sqrt{\beta_t}\, \varepsilon_t \qquad \varepsilon_t \sim N(0, I)$$

$L =$

$$\mathbb{E}_q\left[\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_T|\mathbf{x}_0)\,\|\,p(\mathbf{x}_T))}_{L_T} + \sum_{t>1}\underbrace{D_{\mathrm{KL}}(q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0)\,\|\,p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t))}_{L_{t-1}} \underbrace{- \log p_\theta(\mathbf{x}_0|\mathbf{x}_1)}_{L_0}\right]$$

Given $\mathbf{x}_0$, everything is Gaussian. (Joint is not.)

$$\alpha_t := 1 - \beta_t \qquad \bar{\alpha}_t := \prod_{s=1}^{t}\alpha_s$$

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$$

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t,\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0), \tilde{\beta}_t\mathbf{I})$$

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t \quad \text{and} \quad \tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

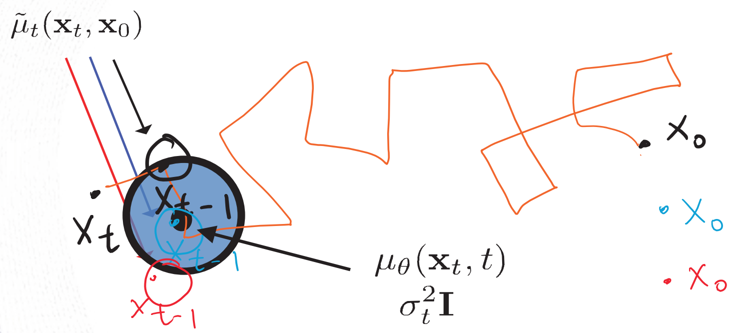
Learning

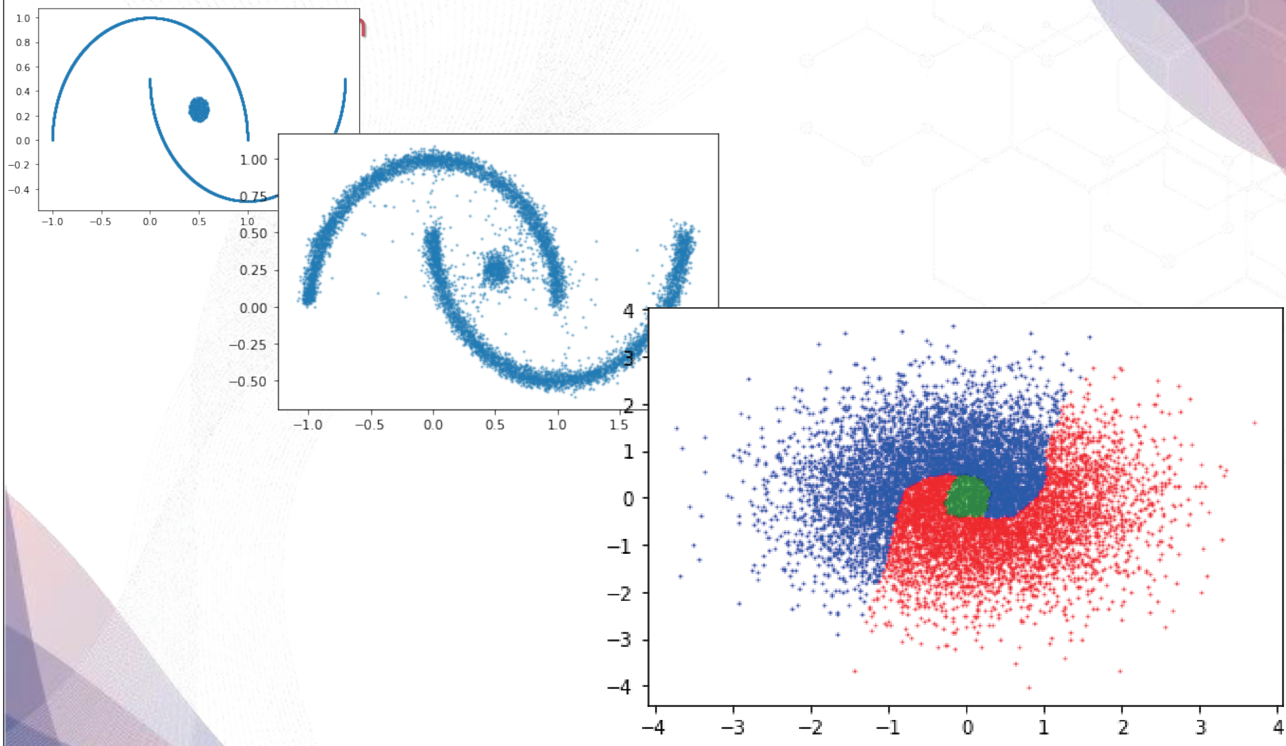
- Model

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t,t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t,t)) \qquad \boldsymbol{\Sigma}_\theta(\mathbf{x}_t,t) = \sigma_t^2\mathbf{I}$$

- K-L divergence:

$$L_{t-1} = \mathbb{E}_q\left[\frac{1}{2\sigma_t^2}\|\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t,\mathbf{x}_0) - \boldsymbol{\mu}_\theta(\mathbf{x}_t,t)\|^2\right] + C$$





Conditional mean

$$\mathbf{x}_t(\mathbf{x}_0, \boldsymbol{\epsilon}) = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon} \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$$

$$L_{t-1} - C$$

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

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

Recall

$$\tilde{\boldsymbol{\mu}}_t(\mathbf{x}_t, \mathbf{x}_0) := \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t \quad \text{and} \quad \tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

μ_θ must predict $\frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}\right)$ given $\mathbf{x}_t$

- Reparameterization

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

$\boldsymbol{\epsilon}_\theta$ is a function approximator intended to predict $\boldsymbol{\epsilon}$ from $\mathbf{x}_t$

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

27

Procedure

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

- From $\mathbf{x}_0$, generate $\mathbf{x}_t$, then predict ϵ.
- The distribution of $\epsilon_\theta(\mathbf{x}_t, \mathbf{x}_0)$ is determined by the distribution of $\mathbf{x}_0$. "Distribution of ϵ is isotropic Gaussian (non-informative)."
- Given $\mathbf{x}_0$, the distribution of ε should be non-informative. After marginalization, the expectation is the global flow of data due to diffusion.

28

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
  $$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\underbrace{\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}}_{= \mathbf{x}_t}, t) \right\|^2$$
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 $\mathbf{x}_0$

Adding noise

Figure 6: Unconditional CIFAR10 progressive generation ($\hat{\mathbf{x}}_0$ over time, from left to right). Extended samples and sample quality metrics over time in the appendix (Figs. 10 and 14).

Results

Figure 7: When conditioned on the same latent, CelebA-HQ 256 $\times$ 256 samples share high-level attributes. Bottom-right quadrants are $\mathbf{x}_t$, and other quadrants are samples from $p_\theta(\mathbf{x}_0|\mathbf{x}_t)$.

Figure 8: Interpolations of CelebA-HQ 256x256 images with 500 timesteps of diffusion.

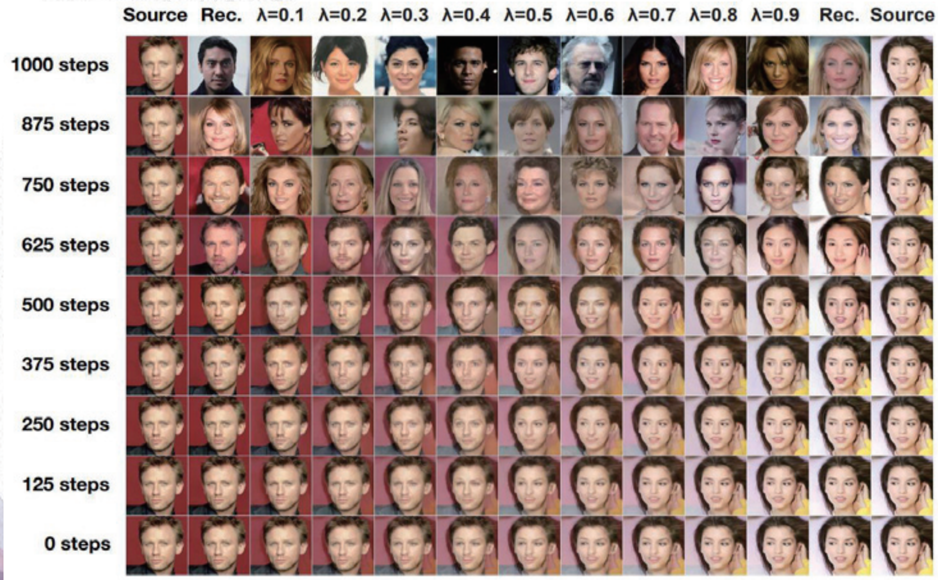


Figure 9: Coarse-to-fine interpolations that vary the number of diffusion steps prior to latent mixing.

Classifier-free guidance

$$\begin{aligned}\nabla_{\mathbf{x}_t} \log p(y|\mathbf{x}_t) &= \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|y) - \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t) \\ &= -\frac{1}{\sqrt{1-\bar{\alpha}_t}} \Big(\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \Big)\end{aligned}$$

$$\begin{aligned}\bar{\boldsymbol{\epsilon}}_\theta(\mathbf{x}_t, t, y) &= \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y) - \sqrt{1-\bar{\alpha}_t}\; w \nabla_{\mathbf{x}_t} \log p(y|\mathbf{x}_t) \\ &= \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y) + w\big(\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\big) \\ &= (w+1)\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, y) - w\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\end{aligned}$$

Summary

Diffusion and Construction of Global Flow

Inference with Gaussians

Learning in DDPM

35

36

KSBI | 한국생명정보학회

KSBi-BIML 2024

Diffusion Models - 이해와 응용

노영균 (한양대학교 & 고등과학원)
nohyung@hanyang.ac.kr

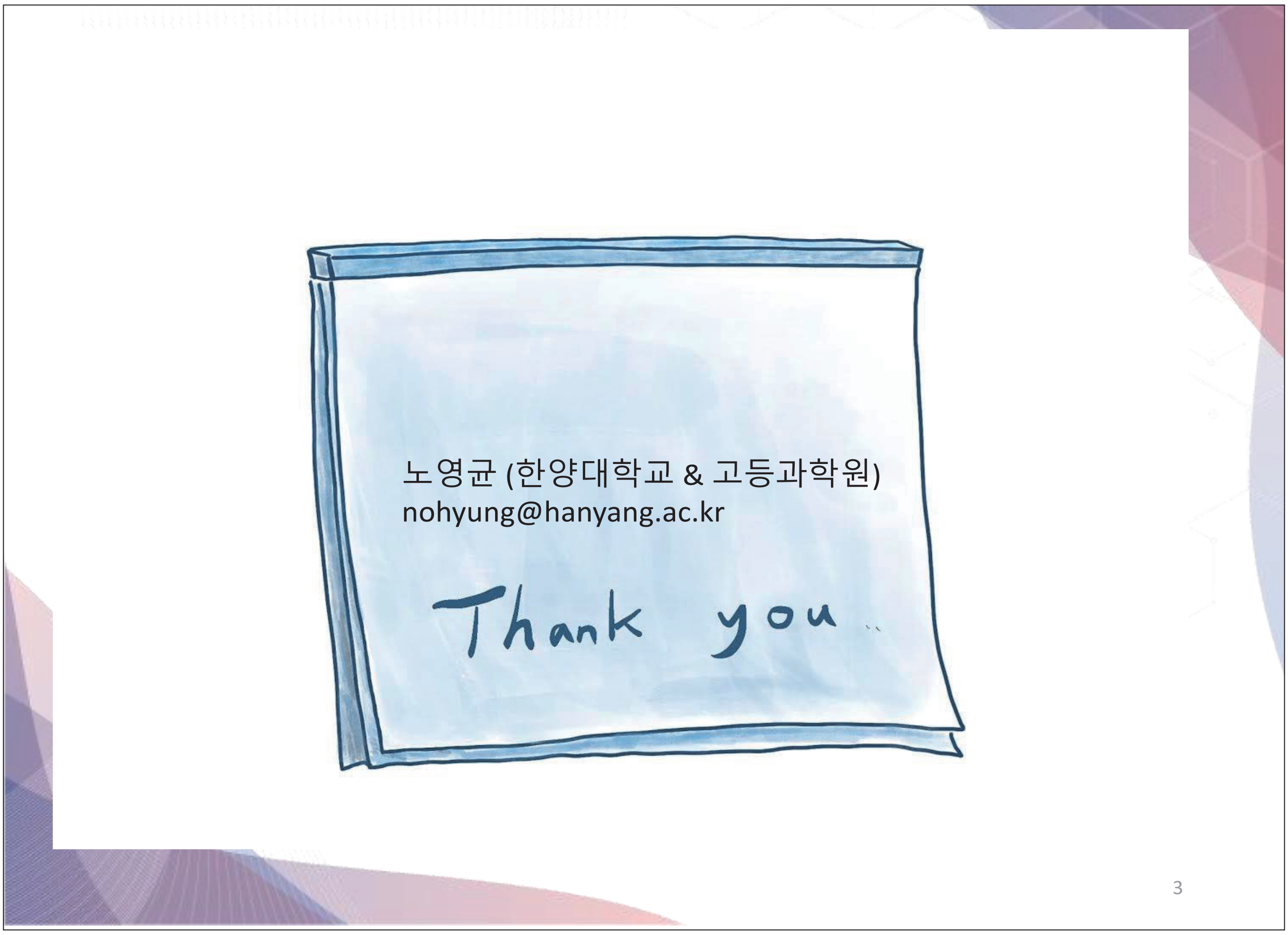
Link

CoLab -> Open notebook -> Github ->

Enter a GitHub URL or search by organization or user
nohyung

Repository
nohyung/2024_BIML

2


노영균 (한양대학교 & 고등과학원)
nohyung@hanyang.ac.kr
Thank you
3