MLPhys 学術変革領域研究(A) 学習物理学の創成
Foundation of "Machine Learning Physics"

JST CREST
Core Research for Evolutional Science and Technology
Japan Science and Technology Agency

Generative Models
Potentially Useful for Science

Yoshiyuki Kabashima
Institute for Physics of Intelligence
The University of Tokyo, Japan

東京大学
THE UNIVERSITY OF TOKYO

東京大学 大学院
理学系研究科・理学部
SCHOOL OF SCIENCE, THE UNIVERSITY OF TOKYO

知の物理学研究センター
Institute for Physics of Intelligence

Background and Purpose

- Generative models represent one of the most groundbreaking advancements in machine learning research over the past decade.
- They have the potential to dramatically transform the future of science.

- In this talk, we will introduce representative generative models and their applications in scientific research.

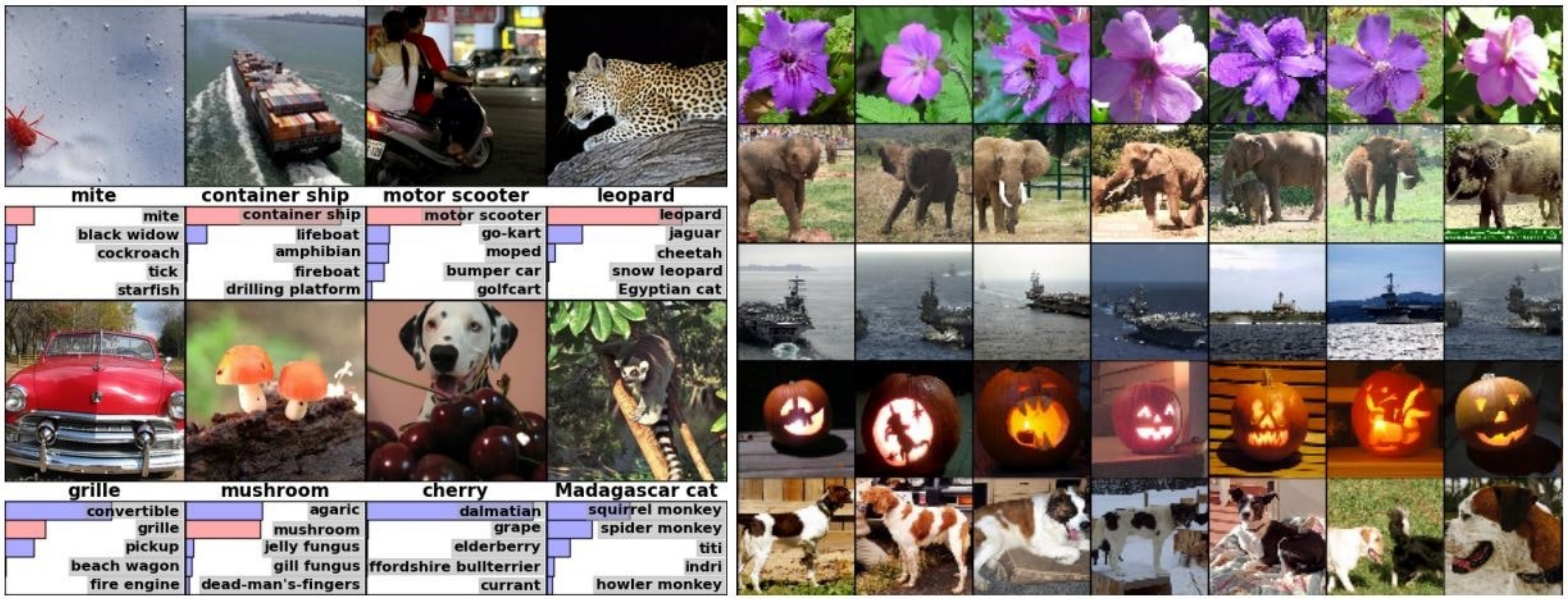
Outline

- What are generative models?
- Three generative models
  - Generative Adversarial Networks (GANs)
  - Variational AutoEncoders (VAEs)
  - Diffusion Models (DMs)
- Summary

WHAT ARE GENERATIVE MODELS?

Deep Neural Networks

- Over the past decade, it was shown that deep neural networks (DNNs) outperform human recognition abilities *by learning only* 10^6 *images*.



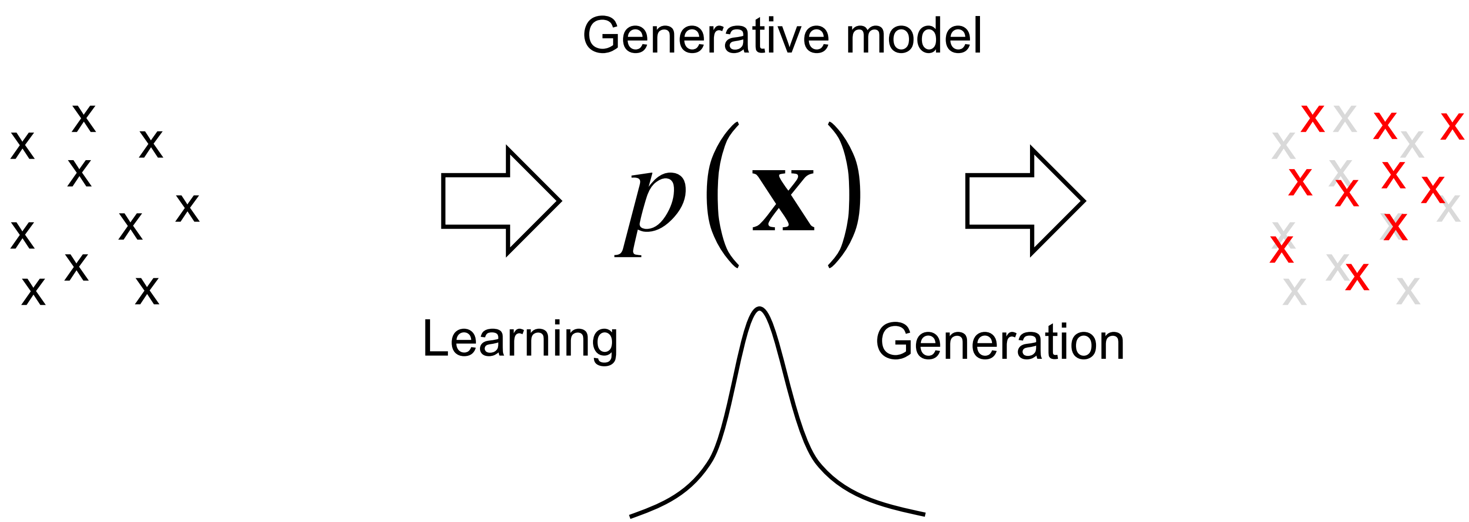
Krizhevsky et al, NIPS2012, pp.1097-1105 (2012)

Training data: 1,000 images per category across 1,000 categories
Test data: Approximately 150,000 images
TOP-5 error: 15.3% (1st place) # 2nd place was 26.2%. DNNs achieved an overwhelming victory.

From Recognition to Generation

- This suggests that data such as images, language, and music, which humans can understand meaningfully, might not actually contain as much information (or be as complex) as we might think.

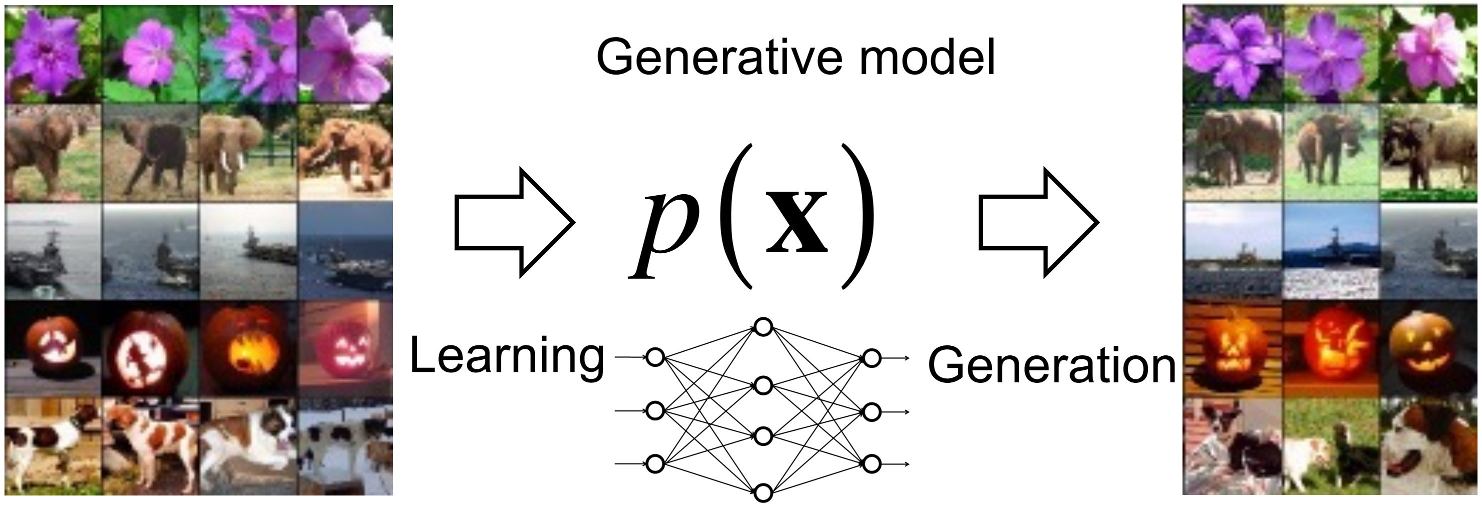
Just as a Gaussian distribution can be estimated from numerical data to generate new data, it may be possible to estimate a "generative model for images" from image data to generate new images.



From Recognition to Generation

- This suggests that data such as images, language, and music, which humans can understand meaningfully, might not actually contain as much information (or be as complex) as we might think.

Just as a Gaussian distribution can be estimated from numerical data to generate new data, it may be possible to estimate a "generative model for images" from image data to generate new images.



Challenges in Modeling High-dimensional Distributions

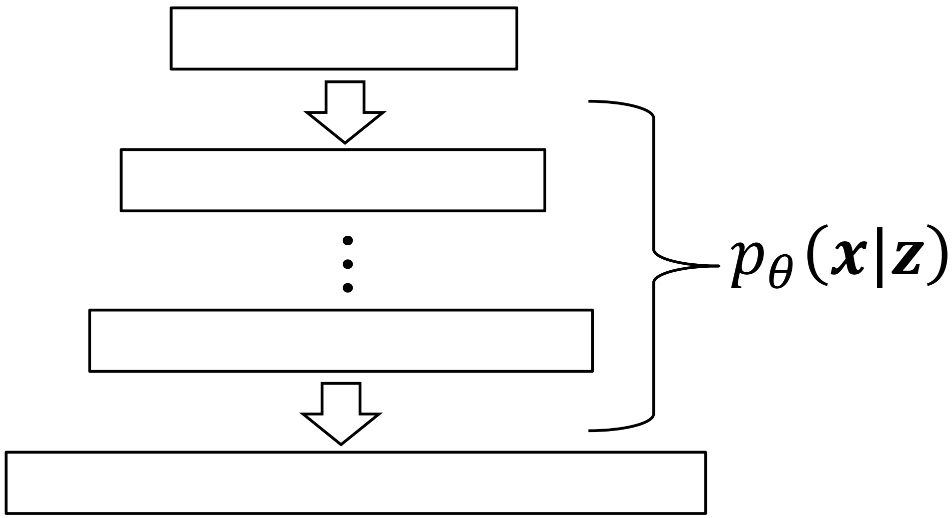
- High-dimensional distributions often require complex models to accurately capture their structure. High-dimensional and complex models are often accompanied by the following challenges:
  - **Curse of dimensionality**: The volume of the space grows exponentially with dimensionality, requiring an exponentially larger amount of data to sufficiently cover the space and accurately model the distribution.
  - **Computational complexity:** Methods such as kernel-based approaches or Bayesian inference can become infeasible due to high computational overhead.

Practically Unique Solution

- **Latent variable models**
  - Express complex distributions in high-dimensional space by mixtures of computationally manageable models *such as Gaussians.*

$\boldsymbol{z}$: Latent variable (Gaussian)

$$p(\boldsymbol{z}) = N(\boldsymbol{z}|0, I)$$



$$p_\theta(\boldsymbol{x}) = \int p_\theta(\boldsymbol{x}|\boldsymbol{z}) p(\boldsymbol{z})\, d\boldsymbol{z}$$

$\boldsymbol{x}$: Visible variable (Image, languages, music, etc)

Advantage:
Can represent high-D. and complex distributions suppressing
- Curse of dimensionality
- Computational complexity

How to Train Models

- Currently, there are three major approaches for training the high-dimensional latent variable models
  - Generative adversarial networks (GANs)
  - Variational auto encoders (VAEs)
  - Diffusion models (DMs)

- In the following, we will sequentially explore these ideas and their applications.

GENERATIVE ADVERSARIAL NETWORKS (GANS)

Core Idea of GANs

- It is difficult for anything other than a human to evaluate whether an image represents a "human face."
- However, having humans intervene to assess the quality of training outcomes is too costly.

⇩

Create the evaluator using machine learning as well

Generative Adversarial Networks: GANs

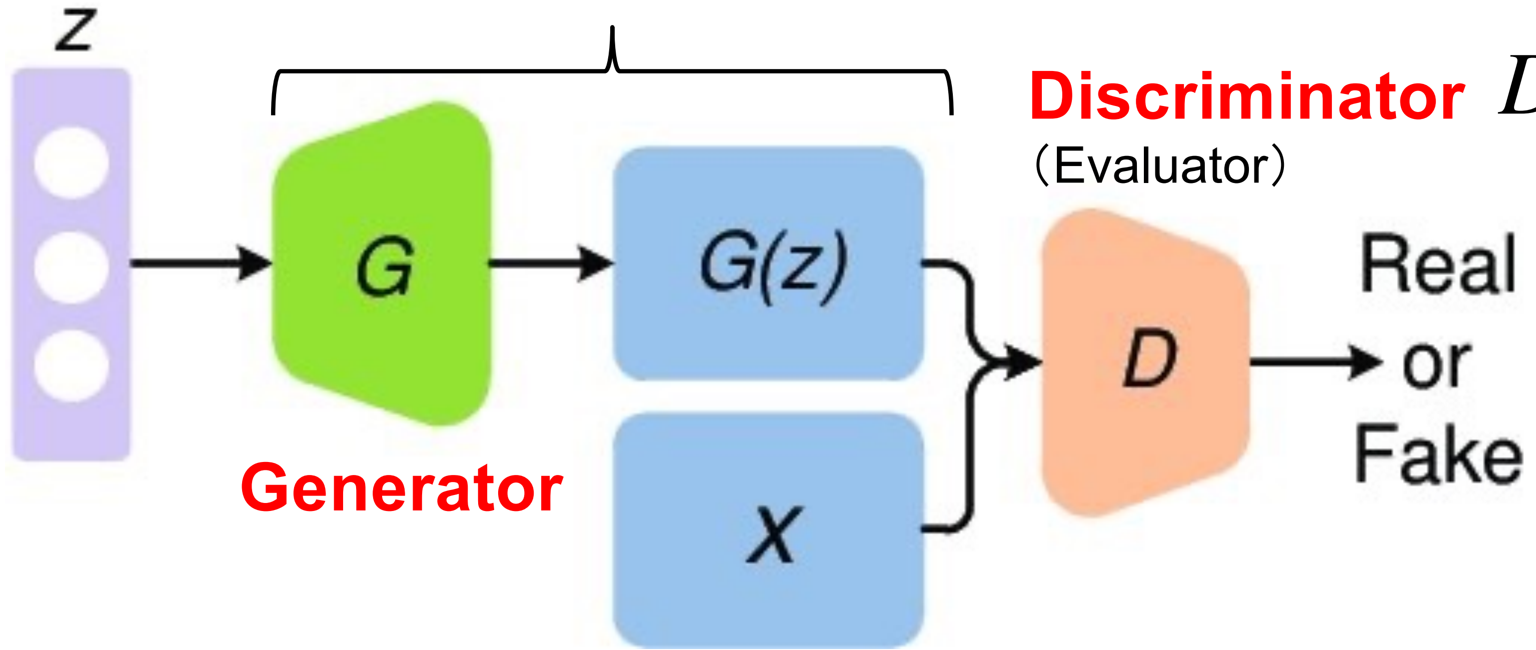
Latent variable

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

Visible variable (Fake image)

$$\mathbf{x} \sim p(\mathbf{x} | \mathbf{z}; \theta_g)$$

$$D(\mathbf{x}; \theta_d)$$



Training data (Real image)

Pan et al, IEEE Access 7, pp. 36322—36333 (2019)

Requirements for Each Module

Generator：
To create fakes that are indistinguishable
from the real ones

Discriminator：
To accurately differentiate between “real” and “fake”

DNNs

Learning is advanced by making them compete
with each other
(a game of cat-and-mouse, like counterfeiting money).

Objective Function of GANs

Goodfellow et al, Advances in Neural Information Processing Systems 27 (2014)
(DOI: 10.1145/3422622)

$D(\mathbf{x}) \triangleq$ The confidence that x is "***Real***"

$1 - D(\mathbf{x}) =$ The confidence that x is "***Fake***"

$$V(D,G) \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data(\mathbf{x})}}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim p_{(\mathbf{z})}}\left[\log\left(1 - D\left(G(\mathbf{z})\right)\right)\right]$$

Learning Rule

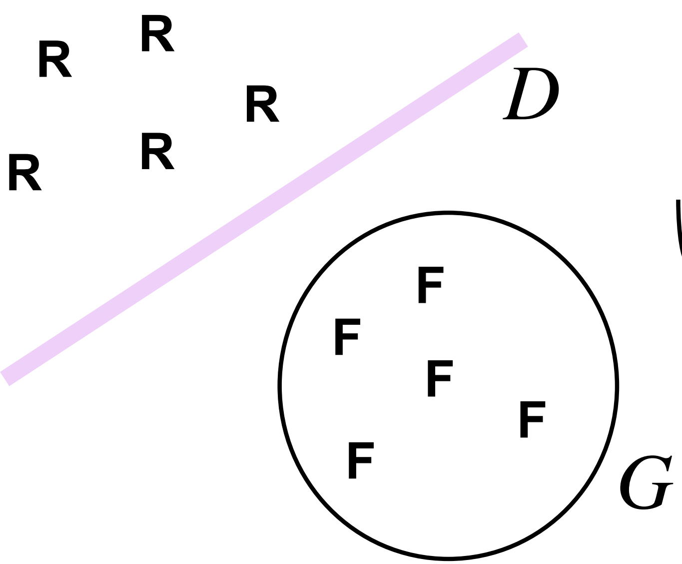
$$\min_{G}\max_{D}\left\{V(D,G)\right\}$$

Implication

Discriminator: Aim to classify real as “real” and fake as “fake”.

<u>Learning D for fixed G</u>

$$V(D,G) \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data(\mathbf{x})}}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim p_{(\mathbf{z})}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$



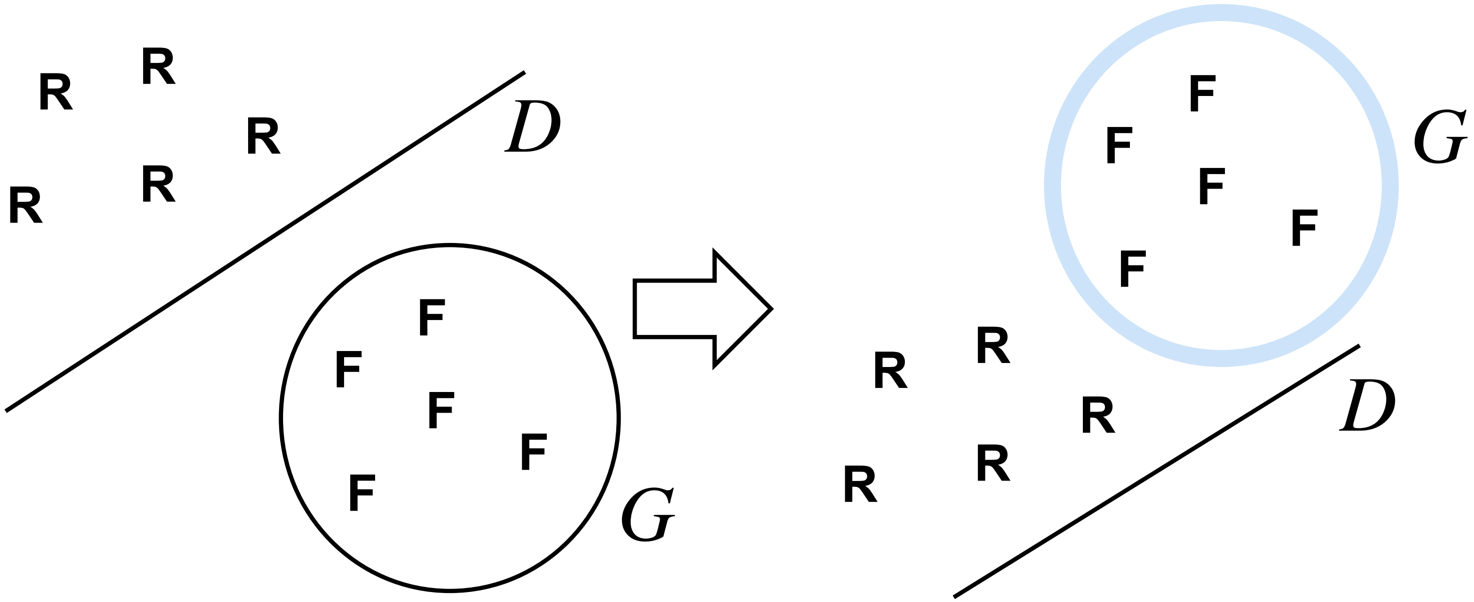
$$\underbrace{\text{"1"} \qquad\qquad\qquad \text{"0"}}_{\displaystyle \max_D V}$$

Implication

Generator : Aim to make Disc. classify fake as "real"

<u>Learning G for fixed D</u>

$$V(D, G) \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data(\mathbf{x})}}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim p_{(\mathbf{z})}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$



"1"

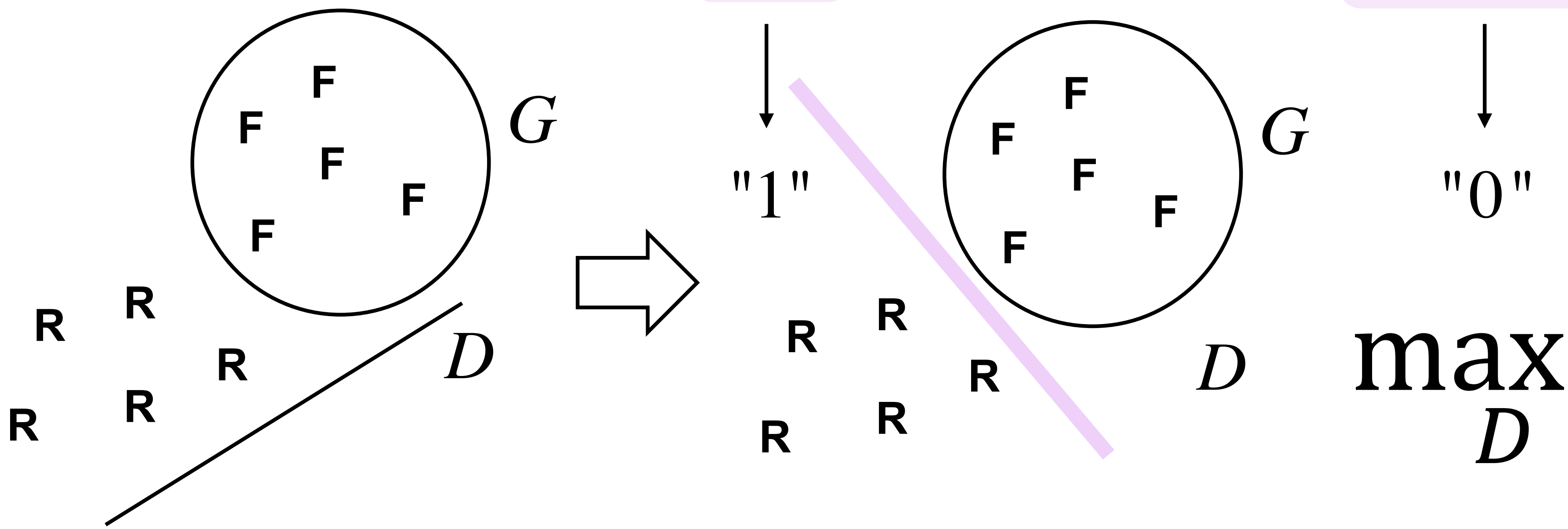
$$\min_G V$$

Implication

Discriminator：Aim to classify real as “real” and fake as “fake”.

<u>Learning D for fixed G</u>

$$V(D,G) \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data(\mathbf{x})}}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim p_{(\mathbf{z})}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$



$$\max_D V$$

Implication

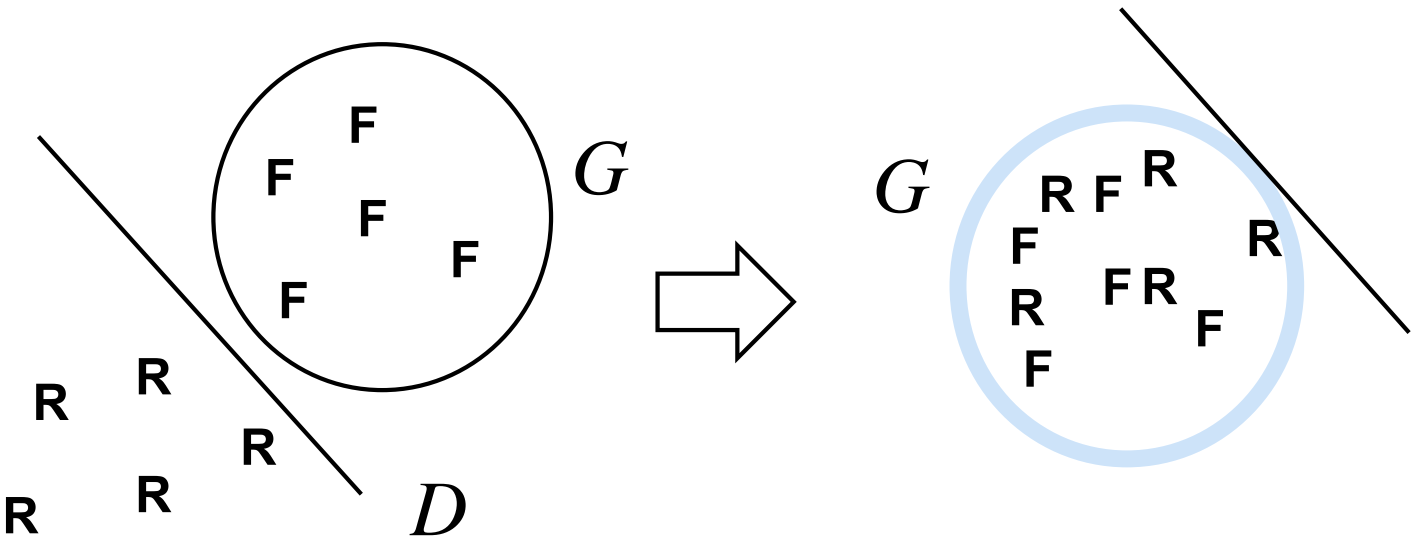
Generator : Aim to make Disc. classify fake as “real”

<u>Learning G for fixed D</u>

$$V(D,G) \triangleq \mathbb{E}_{\mathbf{x}\sim p_{data}(\mathbf{x})}\left[\log D(\mathbf{x})\right] + \mathbb{E}_{\mathbf{z}\sim p_{(\mathbf{z})}}\left[\log\left(1 - D(G(\mathbf{z}))\right)\right]$$

$$\downarrow$$

$$"1"$$



$$\min_G V$$

Can’t distinguish “fake” from “real”!

Strength and Weakness of GANs

- Strength
  - High-Quality: Can produce highly realistic and high-resolution images
  - Fast generation: Can generate data fast
  - Versatile Applications: Have a wide range of applications
- Weakness
  - Training Instability: Can be unstable and challenging
  - Evaluation Difficulty: Lack of standardized metrics
  - Computational Cost: Significant computational resources for training

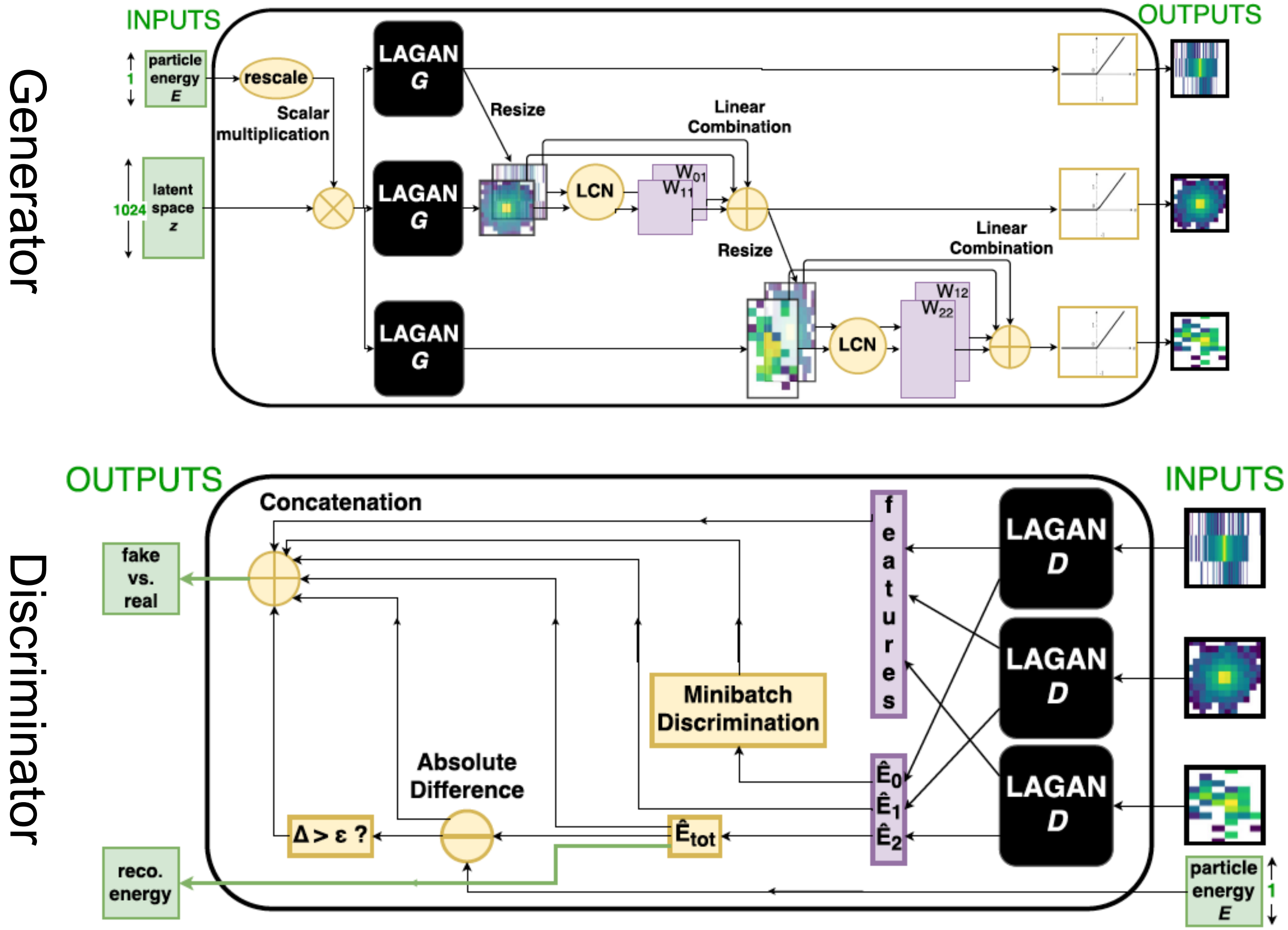
Applications to Science

- Currently, GANs are employed in various fields of science.

Acceleration of Particle Physics Simulation

PHYSICAL REVIEW D 97, 014021 (2018)

- Accelerating 3D particle shower simulations using GANs



Modeling of electromagnetic showers in a longitudinally segmented calorimeter

Performance Gain

CPU: Same or $10^2 \sim 10^3 \times$ faster

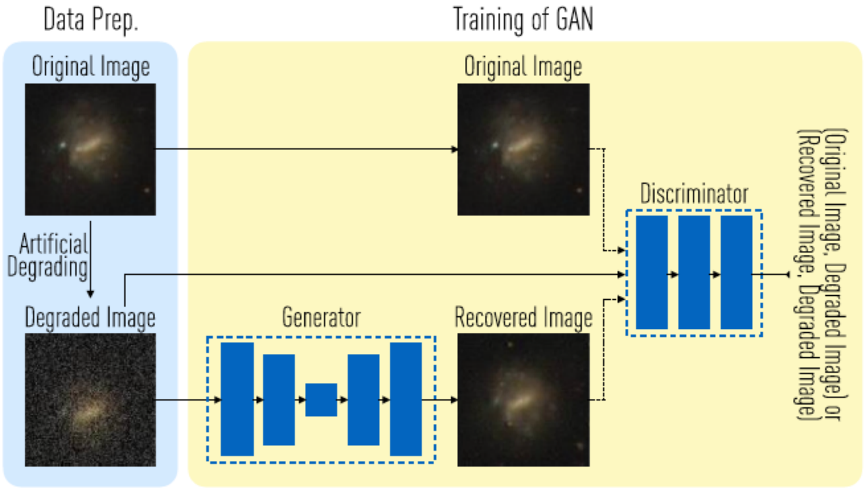
GPU: $10^5 \times$ faster

than conventional method

Restoration of Galaxy Images

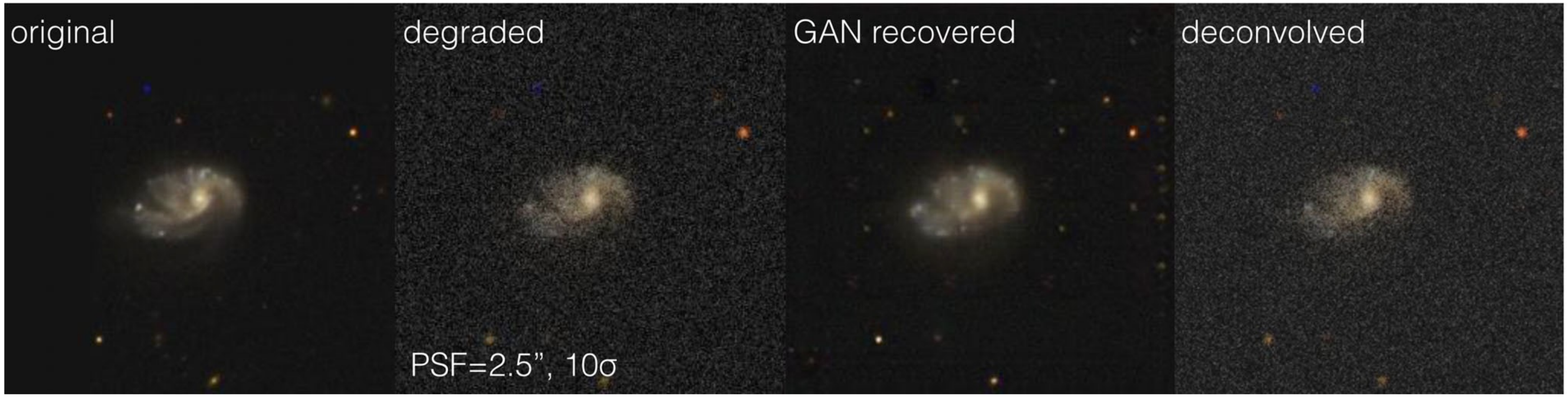
MNRAS 467, L110–L114 (2017)

- Restore features of galaxy images beyond the observational limits using GANs.



Train GAN to recover original images handling artificially degraded images as “latent variables”

Trained GAN outperforms the standard method in restoration performance



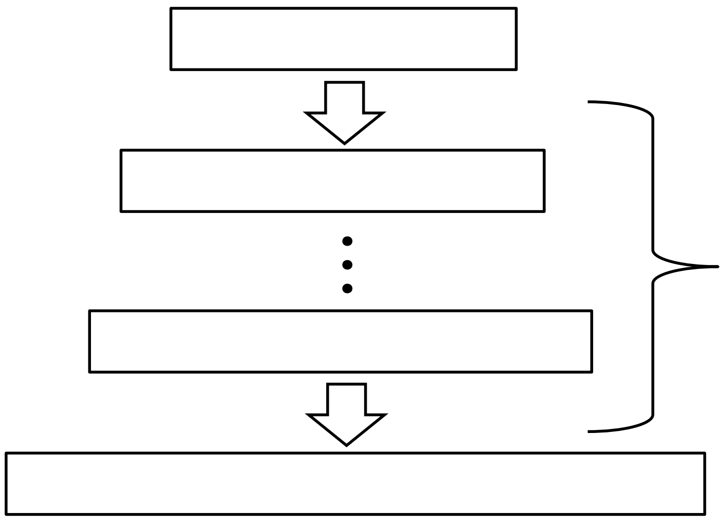
VARIATIONAL AUTOENCODERS (VAES)

Core Idea of VAEs

- Train the generative model and the *recognition model* simultaneously.
  - Efficient training becomes possible by finding appropriate representations in the latent space.

Generative model (Decoder)

$$p(\boldsymbol{z}) = N(\boldsymbol{z}|0, I)$$

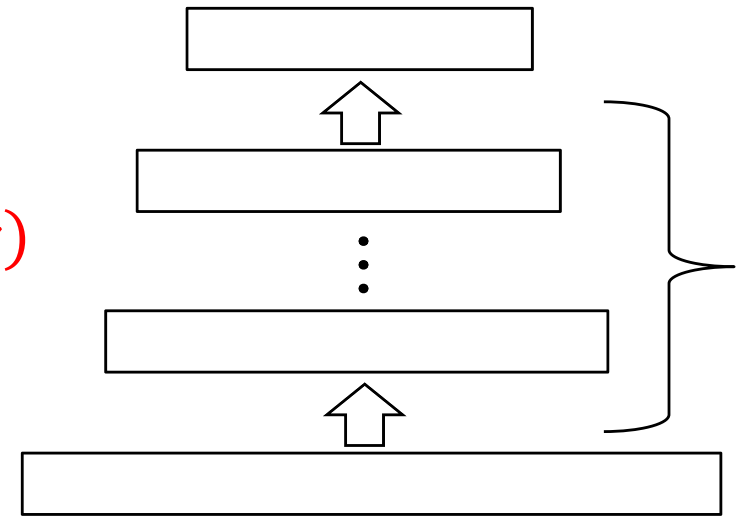


$$p_\theta(\boldsymbol{x}|\boldsymbol{z})$$

$$p_\theta(\boldsymbol{x}) = \int p_\theta(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})\,d\boldsymbol{z}$$

Recognition model (Encoder)

$$q_\phi(\boldsymbol{z}) = \int q_\phi(\boldsymbol{z}|\boldsymbol{x})p_{data}(\boldsymbol{x})d\boldsymbol{x}$$



$$q_\phi(\boldsymbol{z}|\boldsymbol{x})$$

$$p_{data}(\boldsymbol{x}) = \frac{1}{N}\sum_{\mu=1}^{N}\delta(\boldsymbol{x} - \boldsymbol{x}^\mu)$$

Training Principle

- Basically, VAE aims to maximize the marginal likelihood (evidence)

$$\max \sum_{\mu=1}^{N} \log p_\theta(\boldsymbol{x}^\mu) = \max \sum_{\mu=1}^{N} \log \int p_\theta(\boldsymbol{x}^\mu|\boldsymbol{z}^\mu) p(\boldsymbol{z}^\mu)\, d\boldsymbol{z}^\mu \,.$$

- Unfortunately, this is generally difficult to carry out due to the model complexity.

Tractable Modeling and ELBO

- Two techniques for tractability
  - Analytically tractable modeling

$$q_\phi(\boldsymbol{z}|\boldsymbol{x}) = N\left(z|\mu_\phi(\boldsymbol{x}), \Sigma_\phi(\boldsymbol{x})\right)$$: Recognition (Encoder)

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}) = N\left(\boldsymbol{x}|\mu_\theta(\boldsymbol{z}), \Sigma_\theta(\boldsymbol{z})\right)$$: Generative (Decoder)

  - Replacement of the log likelihood with its lower bound termed "Evidence Lower Bound (ELBO)".

$$\sum_{\mu=1}^{N} \log p_\theta(\boldsymbol{x}^\mu)$$
$$\geq \sum_{\mu=1}^{N} \left\{\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu)}[\log p_\theta(\boldsymbol{x}^\mu|\boldsymbol{z})] - D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu)||p(\boldsymbol{z})\right)\right\} =: ELBO(\phi, \theta)$$

Implication of ELBO

- Implication of each term

$$ELBO(\phi, \theta) = \sum_{\mu=1}^{N} \left\{ \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu)}[\log p_\theta(\boldsymbol{x}^\mu|\boldsymbol{z})] - D_{KL}\left(q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu)||p(\boldsymbol{z})\right)\right\}$$

Reconstruction quality for training data
in the *visible space*

Implication of ELBO

- Implication of each term

$$ELBO(\phi,\theta) = \sum_{\mu=1}^{N} \left\{ \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu)}[\log p_\theta(\boldsymbol{x}^\mu|\boldsymbol{z})] - D_{KL}\Big(q_\phi(\boldsymbol{z}|\boldsymbol{x}^\mu) || p(\boldsymbol{z}) \Big) \right\}$$

Distance btw two distributions
$p(\boldsymbol{z}) = N(0, I)$ and $q_\phi(\boldsymbol{z}) = \int q_\phi(\boldsymbol{z}|\boldsymbol{x}) p_{data}(\boldsymbol{x}) d\boldsymbol{x}$
in the *latent space*.

Strength and Weakness of VAEs

- Strength
  - Capable of learning a clear structure of the latent space.
  - Simple design that integrates reconstruction and generation.

- Weakness
  - The quality of generated data is slightly inferior compared to GANs.
  - The analytically tractable modeling may not adapt well to complex data.

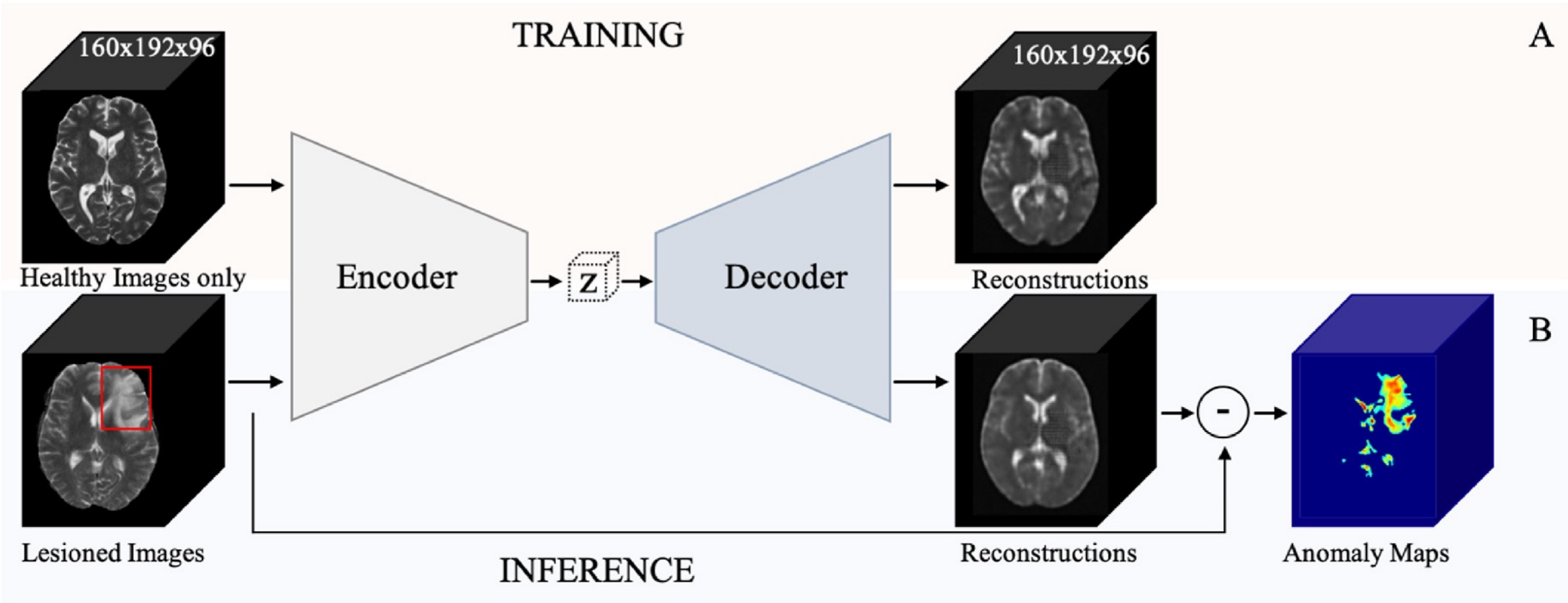
Applications to Science

- VAEs are also employed for various purposes of science.

Anomaly Detection in Brain

Computers in Biology and Medicine 154 (2023) 106610

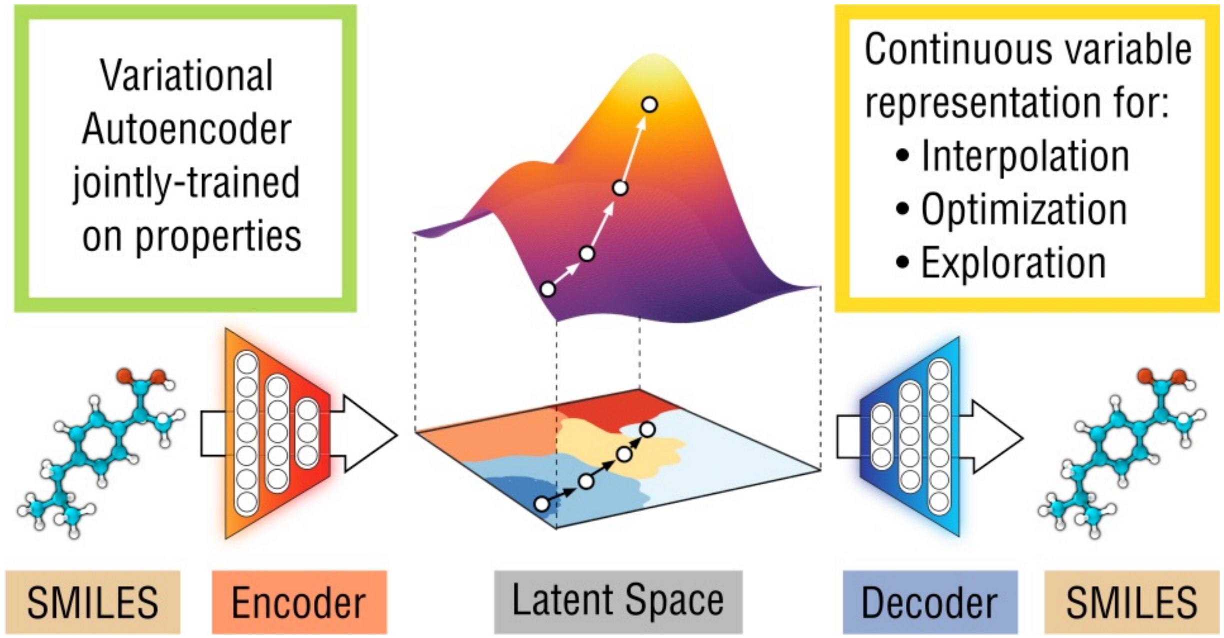
- Create a filter using VAE to extract the essential parts from brain MRI images
- Detect anomaly by examining difference btw “input” and “output”



Design of Molecules

ACS Cent. Sci. 2018, 4, 268−276

- Extract characteristic features of functional molecules in the latent space by VAE.
- Train a regression model to predict molecular properties from the latent space.
- Exploring better molecules that maximize specific properties (e.g., drug-likeness, synthesizability) using the regression model.



DIFFUSION MODELS (DMS)

Core Idea of Diffusion Models

- Sampling from Gaussians is easy even for high-dimensional space.

Gradually transform Gaussian to data distribution using non-equilibrium statistical mechanics

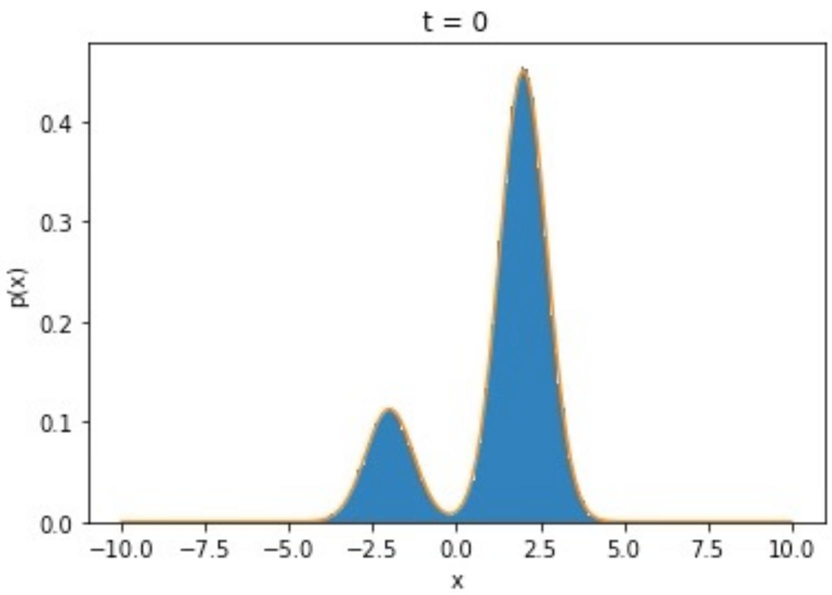
Morphing of Distributions

- Any distribution can be transformed to a Gaussian by Langevin Eq. gradually changing damping/noise parameters.

Set $1 = \alpha_0 > \alpha_1 > \cdots > \alpha_T = 0$

Initial state: $\boldsymbol{x}_0 \sim p_{data}(\boldsymbol{x}_0)$

$$\boldsymbol{x}_t = \sqrt{\alpha_t/\alpha_{t-1}}\,\boldsymbol{x}_{t-1} + \sqrt{1-\alpha_t/\alpha_{t-1}}\,\boldsymbol{z}_{t-1} \ \ (\boldsymbol{z}_{t-1} \sim N(0, I); t \in \{1,2,\ldots,T\})$$



Simulation:
$\alpha_t = 0.99^t$
$T = 1000$ (up to $t = 300$)
#particles = 10^5

Reverse Morphing

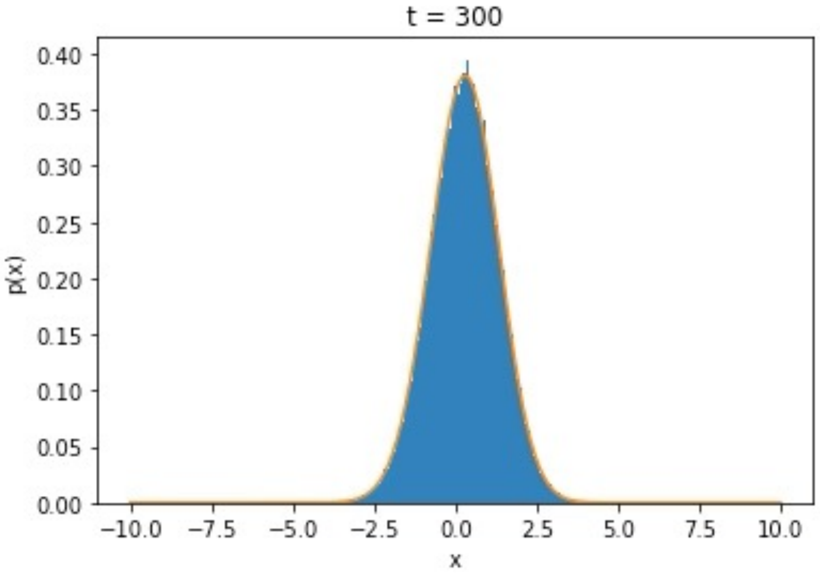
- If vector field $\nabla_{x_t} \log p_{\alpha_t}(\boldsymbol{x}_t)$ (score) is available, we can recover $p_{\alpha_0}(\boldsymbol{x}_0) = p_{data}(\boldsymbol{x}_0)$ from the Gaussian by the *reverse Langevin Eq.*

Set $1 = \alpha_0 > \alpha_1 > \cdots > \alpha_T = 0$

Initial state: $\boldsymbol{x}_T \sim p_{\alpha_T}(\boldsymbol{x}_T) = N(0, I)$

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t/\alpha_{t-1}}}\left(\boldsymbol{x}_t + (1 - \alpha_t/\alpha_{t-1})\nabla_{\boldsymbol{x}_t} \log p_{\alpha_t}(\boldsymbol{x}_t)\right) + \sqrt{1 - \alpha_t/\alpha_{t-1}}\ \boldsymbol{z}_t$$

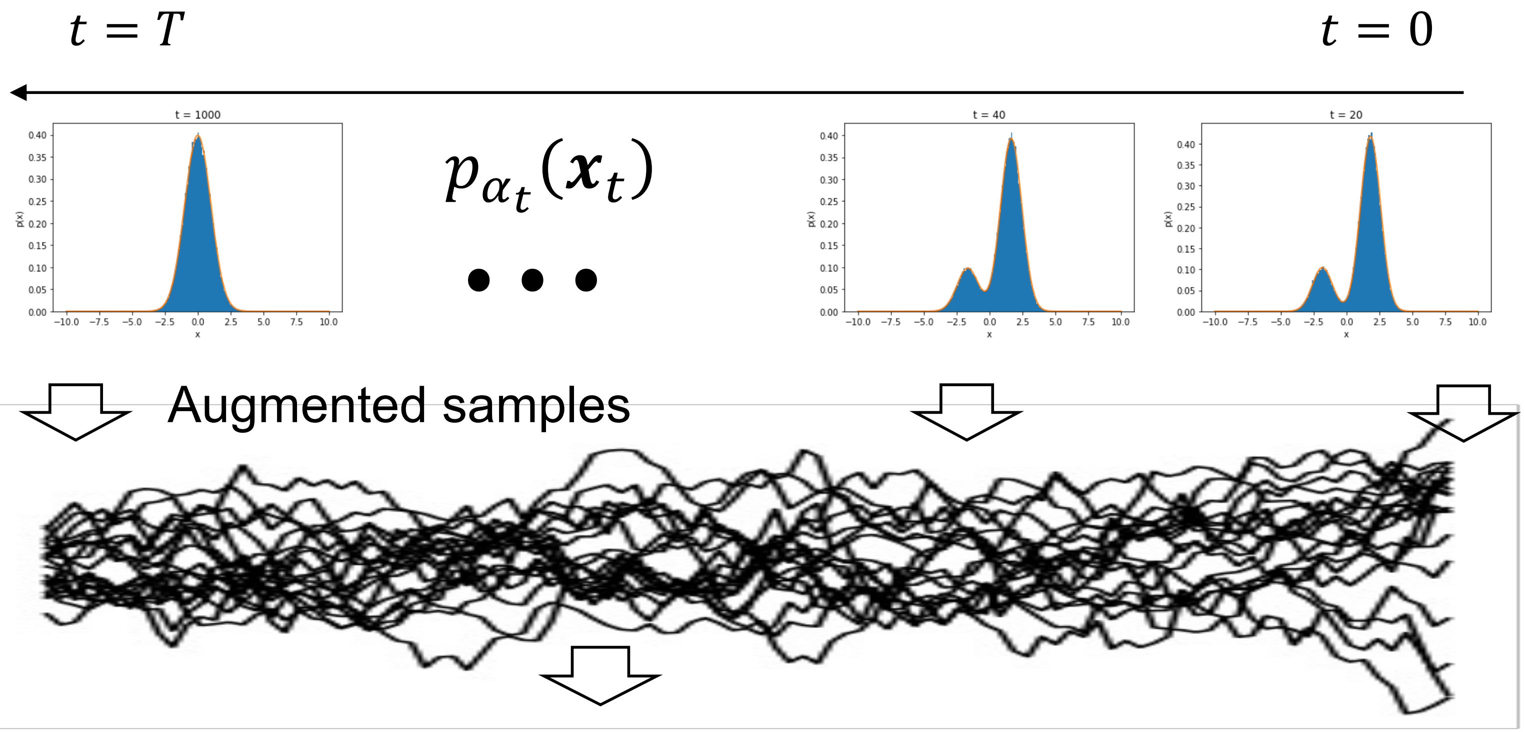
$$(\boldsymbol{z}_t \sim N(0, I); t \in \{T, T-1, \ldots, 1\})$$



Simulation:
$\alpha_t = 0.99^t$
$T = 1000$ (from $t = 300$)
#particles = 10^5

Learning Score by DNNs

- The reverse dynamics can be used for sampling data from high-dimensional complex distributions.
- Unfortunately, $\nabla_{\boldsymbol{x}_t} \log p_{\alpha_t}(\boldsymbol{x}_t)$ cannot be available in practice.
- However, it can be learned by DNNs from augmented samples generated by the *forward* Langevin equation.



Learn $s_\theta(\boldsymbol{x}_t, t) \simeq \nabla_{\boldsymbol{x}_t} \log p_{\alpha_t}(\boldsymbol{x}_t)$ by DNNs

Training in Practice

- Generate augmented samples using the forward Langevin equation.
- Train the score using the entire set of augmented samples.

$$\boldsymbol{x}_t = \sqrt{\alpha_t/\alpha_{t-1}}\, \boldsymbol{x}_{t-1} + \sqrt{1-\alpha_t/\alpha_{t-1}}\, \boldsymbol{z}_{t-1} \quad (\boldsymbol{z}_{t-1} \sim N(0, I); t \in \{1,2,\dots,T\})$$

$\boldsymbol{\alpha}_T = 0$ $\boldsymbol{\alpha}_{T-1}$ $\cdots$ $\boldsymbol{\alpha}_0 = 1$

$\boldsymbol{z}_T$: Gaussian noise

$\boldsymbol{x}_0$: training data

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \left\{ \sum_{t=1}^{T} \mathbb{E}_{p_{data}(x_0)} \mathbb{E}_{p_{\bar{\alpha}_t}(\boldsymbol{x}_t|x_0)} \left\| \underline{s_\theta(\boldsymbol{x}_t, t\,)} + \frac{\boldsymbol{x}_t - \sqrt{\alpha_t}\boldsymbol{x}_0}{1-\alpha_t} \right\|_2^2 \right\}$$

Modeled by DNN (U-net)

Song and Ermon, Generative Modeling by Estimating Gradients of the Data Distribution, NeurIPS2019
Ho et al, Denoising Diffusion Probabilistic Models, NeurIPS2020

Sample Generation in Practice

- Generate samples from the trained model by successively running the reverse Langevin dynamics from Gaussian initial states.

$\boldsymbol{\alpha}_T = 0$ $\boldsymbol{\alpha}_{T-1}$ $\cdots$ $\boldsymbol{\alpha}_0 = 1$

$\boldsymbol{z}_T \sim \mathcal{N}(0, I)$:
latent variable

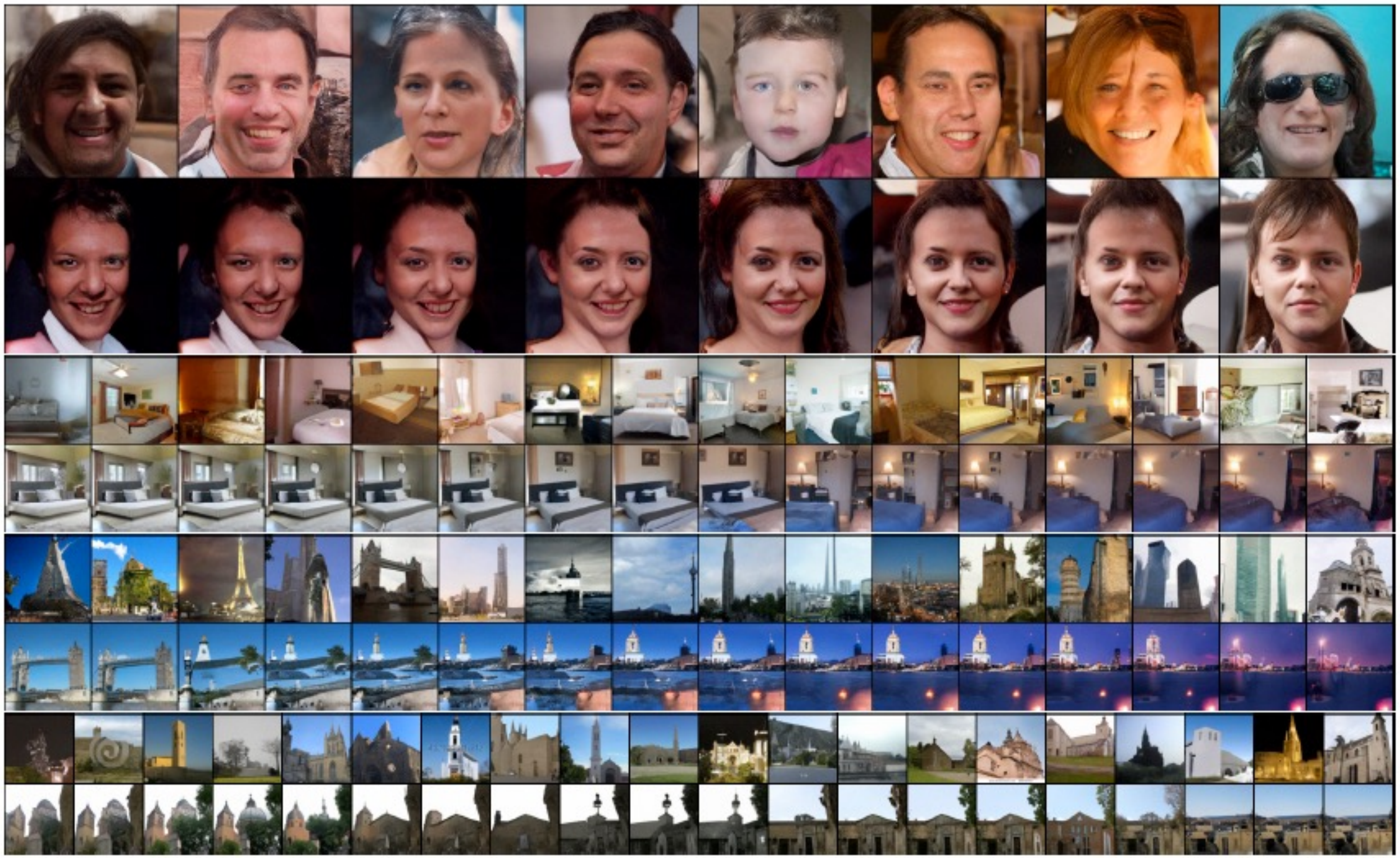
$\boldsymbol{x}$: new data

$$\boldsymbol{x}_{t-1} = \frac{1}{\sqrt{\alpha_t/\alpha_{t-1}}}\Big(\boldsymbol{x}_t + (1 - \alpha_t/\alpha_{t-1})\underline{s_\theta(\boldsymbol{x}_t, t)}\Big) + \sqrt{1 - \alpha_t/\alpha_{t-1}}\ \boldsymbol{z}_t$$

Learned score

Song and Ermon, Generative Modeling by Estimating Gradients of the Data Distribution, NeurIPS2019
Ho et al, Denoising Diffusion Probabilistic Models, NeurIPS2020

Generated Samples



Song and Ermon, Generative Modeling by Estimating Gradients of the Data Distribution, NeurIPS2019

Strength and Weakness of DMs

- Strength
  - High-Quality: Can produce highly realistic and high-resolution images
  - Stability: Training is stable
  - Intimacy with probability: Easy to use for various image processing tasks
- Weakness
  - Computational cost: Significant computational resources for both training and data generation

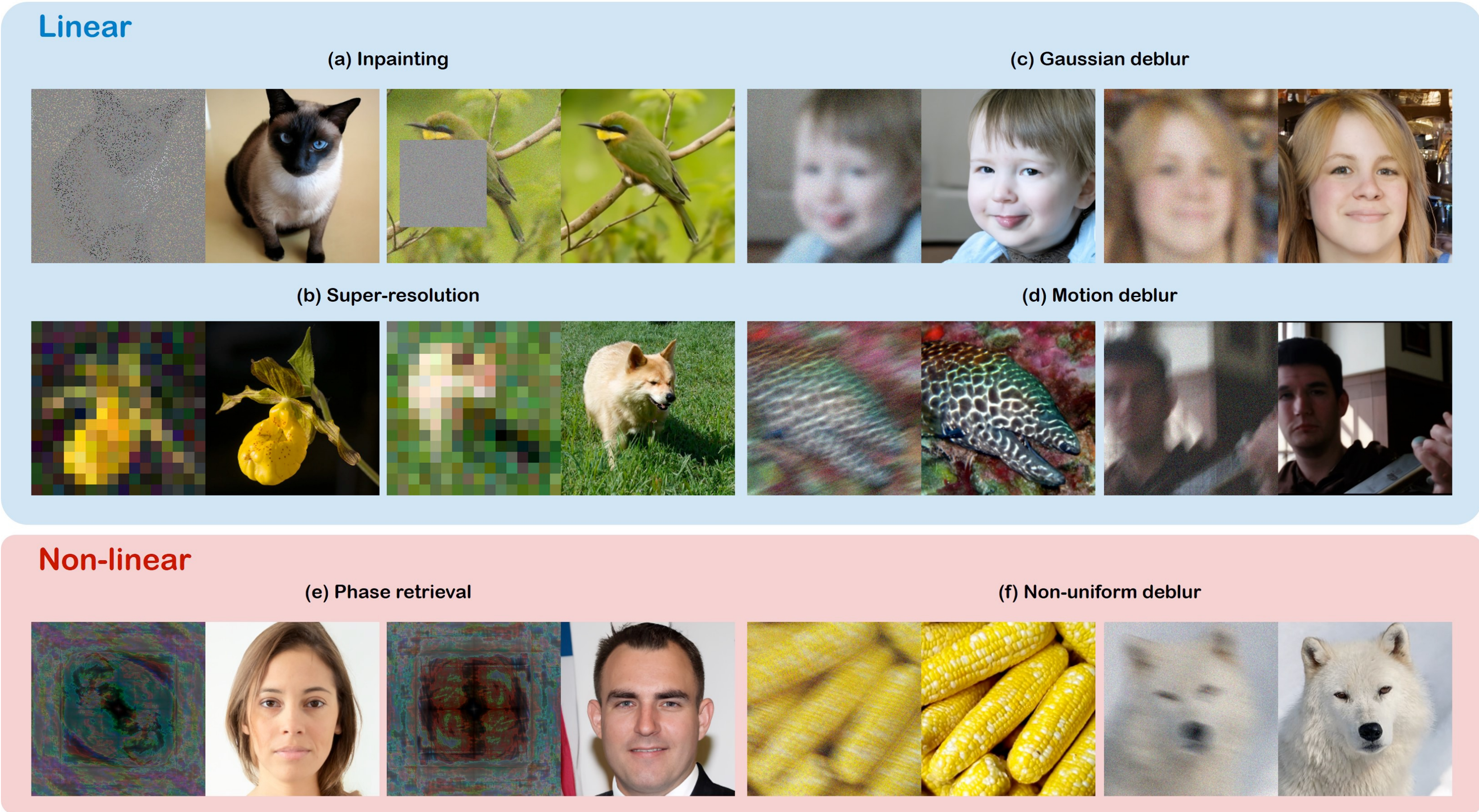
Applications to Science

- DMs are the current de fact standard for image generation.
- They are also used for various tasks of image processing using DMs as prior distribution of target images.

Image Completion

Presented at ICLR (2023); arXiv:2209.14687

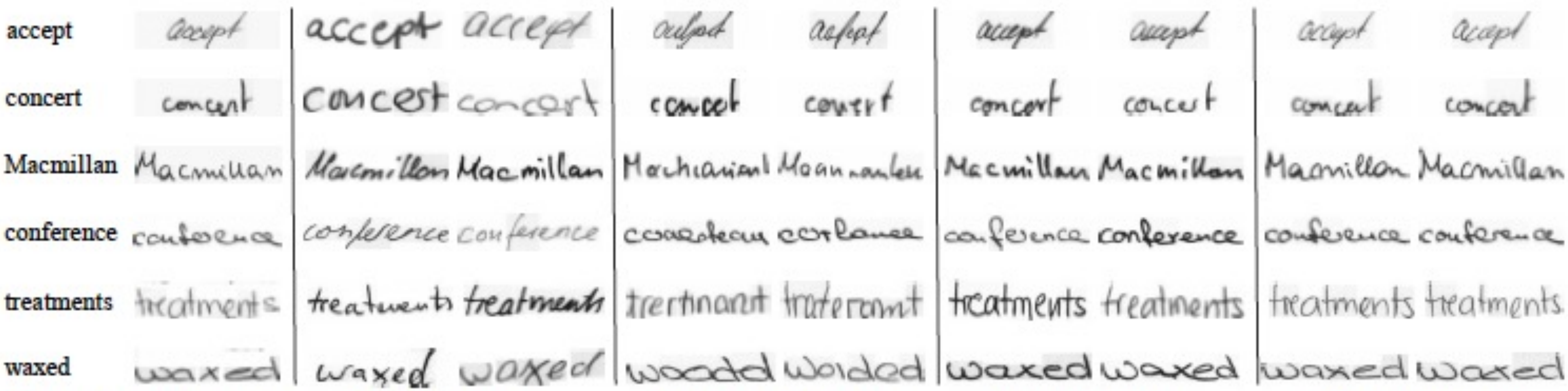
- Image reconstruction from incomplete measurements using DMs as priors.
  - Can be used in a plug-and-play manner.



Conditional Text Image Generation

Presented at CVPR (2023); arXiv:2306.10804

- Converts input handwritten character images into handwritten character images that meet given conditions.



(a) text and original image (b) synthesis mode (c) augmentation mode (d) recovery mode (e) imitation mode

Summary

- We introduced three generative models that are actively studied these days.
- Their comparison is summarized below.

Model	Features	Strength	Weakness	Key Applications
GANs	- G. and D. compete during training - Produces highly realistic outputs	- H.Q. and fast data generation - Tunable for specific tasks	- Unstable training - Limited diversity in some cases	- Super resolution - Surrogate simulator - Other various purposes
VAEs	- Encoder-decoder structure - Optimizes ELBO	- Effective in understanding data - Simple design	- Lower quality than GANs and DMs - May not adapt to complex data	- Data generation - Latent variable analysis - Anomaly detection
DMs	- Trained by adding noise to data - Generation by reverse process	- H.Q. and diverse data generation - Stable training	- Computationally expensive - Complex training setup	- H.Q. image generation - Content completion - Conditional generation

Summary

- More recent techniques involve introducing “attention mechanisms” into generative models, but we did not cover this here.
  - Simply because I don't know much about them.
- New machine learning methods are proposed every day, but they don‘t tell us which purposes we should use them for.
- How to effectively use them for science depends on the ingenuity of “domain researchers”.