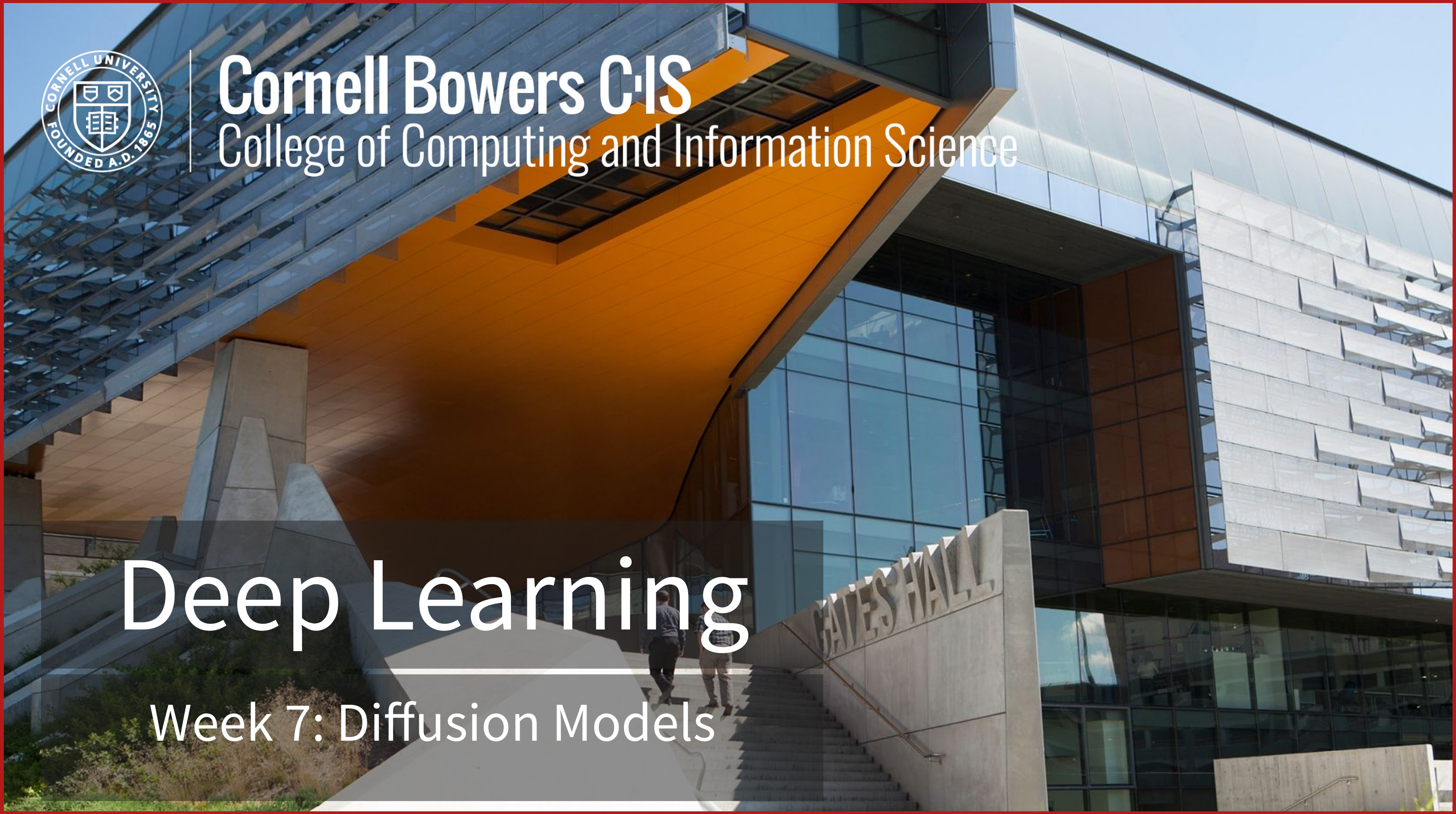


Cornell Bowers C·IS
College of Computing and Information Science

Deep Learning

Week 7: Diffusion Models

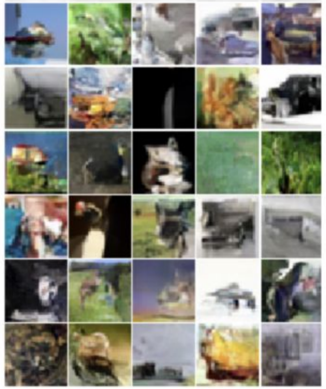
Cornell Bowers C·IS

Progress In Generative Modeling

VAEs, 2013

GANs, 2014

PixelCNN, 2016



BigGAN, 2019

Imagen, 2022

Cornell Bowers C·IS

Text-to-Image Diffusion Models


A bread, an apple, and a knife on a table


a robot cooking dinner in the kitchen


A teddy bear and a stuffed raccoon sitting on a wooden chair side by side


The oil painting shows a cow standing near a tree with red leaves


A traditional tea house in a tranquil garden with blooming cherry blossom trees


a painting of trees near a peaceful lake


A heart made of wood


an old man with green eyes and a long grey beard


A painting of an adorable rabbit sitting on a colorful splash


an afrofuturist lady wearing gold jewelry


a black basketball shoe with a lightning bolt on it


A cool orange cat wearing sunglasses playing a guitar with a group of dancing bananas

Dai, Xiaoliang, et al. "Emu: Enhancing image generation models using photogenic needles in a haystack." arXiv preprint arXiv:2309.15807 (2023).

Cornell Bowers C·IS

Video Generation

Sora

Cornell Bowers C·IS

Video Generation

Sora

Prompt: 3D animation of a small, round, fluffy creature with big, expressive eyes explores a vibrant, enchanted forest.

Cornell Bowers C·IS

Video Generation

Sora

Prompt: 3D animation of a small, round, fluffy creature with big, expressive eyes explores a vibrant, enchanted forest.

Cornell Bowers C·IS

Video Generation

Sora

Prompt: 3D animation of a small, round, fluffy creature with big, expressive eyes explores a vibrant, enchanted forest.

Prompt: A cat waking up its sleeping owner demanding breakfast.

Cornell Bowers C·IS

Video Generation

Sora

Prompt: 3D animation of a small, round, fluffy creature with big, expressive eyes explores a vibrant, enchanted forest.

Prompt: A cat waking up its sleeping owner demanding breakfast.

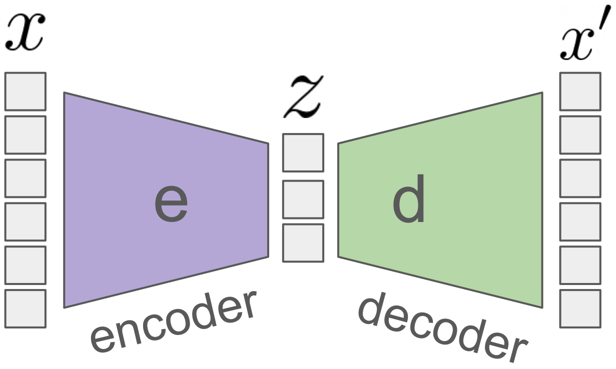
Cornell Bowers C·IS

Autoencoders

- Reconstruction loss: mean squared error

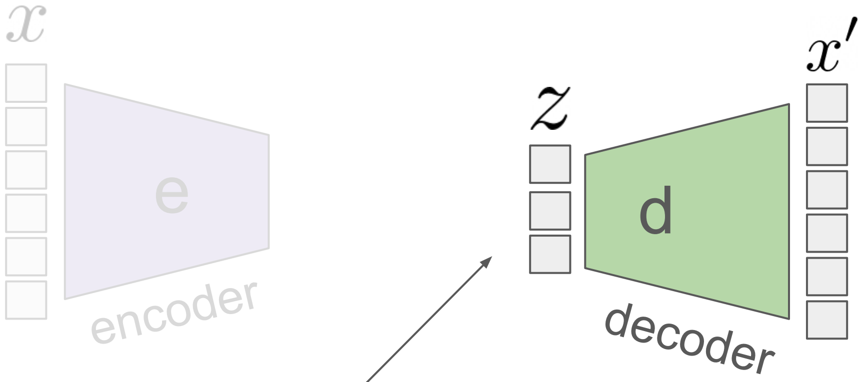
$$\sum_{x \in \mathcal{D}} (x - x')^2$$

where $x' = e(d(x))$



The Result: an Autoencoder.
[Kramer, 1991]

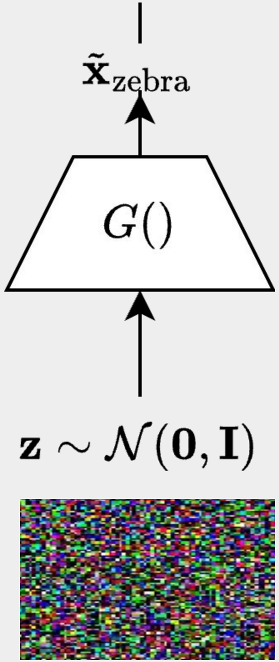
Sampling from an Autoencoder



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

feed decoder
(Gaussian) noise?

Recall how we could sample with GANs…



Cornell Bowers C·IS

Autoencoder trained on MNIST: latent space

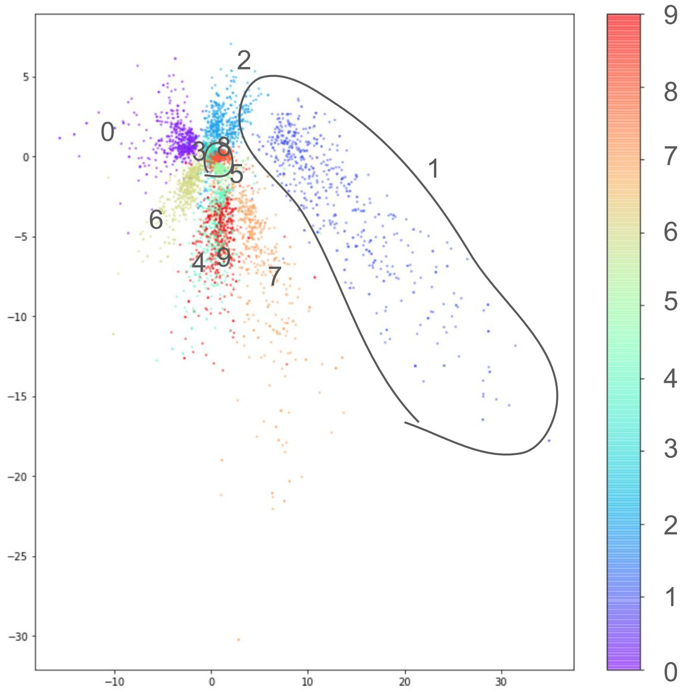


Figure 3-8. Plot of the latent space, colored by digit

Not a very nice representation…

- lots of empty space
- no symmetries between digit representations

Question:

What are the implications for sampling?

Cornell Bowers C·IS

reconstructed sample

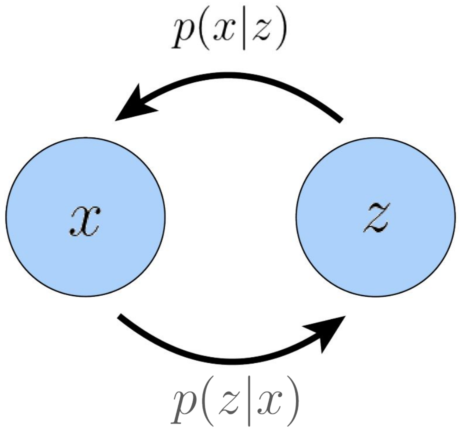
$$x' = d(e(x))$$

new image?

$$x' = d(\text{ noise })$$

Cornell Bowers C·IS

Variational Inference

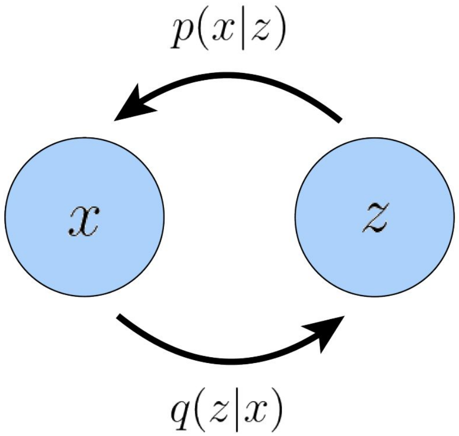


- Have joint model $p(\boldsymbol{x}, \boldsymbol{z})$ $p(\boldsymbol{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$
- observe x (but not z);
- want to calculate posterior $p(\boldsymbol{z}|\boldsymbol{x}) = \dfrac{p(\boldsymbol{x}, \boldsymbol{z})}{p(\boldsymbol{x})}$
- which requires $$p(x) = \int p(x, z)\,\mathrm{d}z$$
  - i.e., the "evidence".

but the integral is often intractable! So, instead …

Cornell Bowers CIS

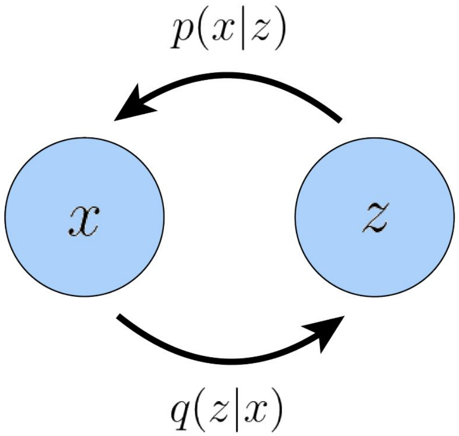
Variational Inference



- Introduce a learnable variational approximation of the posterior

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

Variational Inference



- Introduce a learnable variational approximation of the posterior

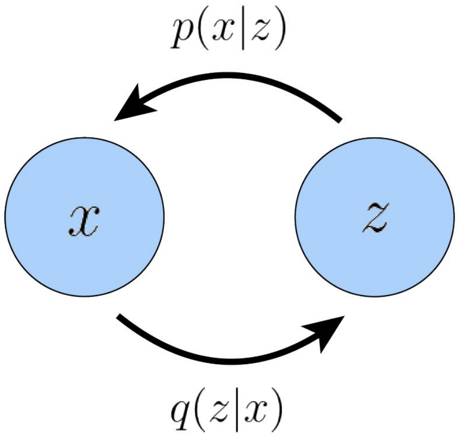
$$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \approx p(\boldsymbol{z}|\boldsymbol{x})$$

- Bound the likelihood using the variational posterior

$$\underbrace{\boxed{\log p(\boldsymbol{x})}}_{\text{Intractable; Evidence/ Log-likelihood}} = \underbrace{\boxed{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right]}}_{\text{Tractable; Evidence Lower Bound (ELBO)}} + \underbrace{\boxed{D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}|\boldsymbol{x}))}}_{\text{Intractable; Divergence from true posterior}}$$

Cornell Bowers C·IS

Variational Inference



- Introduce a learnable variational approximation of the posterior

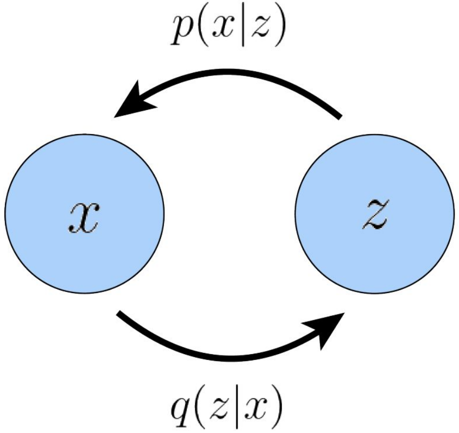
$$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \approx p(\boldsymbol{z}|\boldsymbol{x})$$

- Bound the likelihood with the ELBO

$$\log p(\boldsymbol{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right] + D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}|\boldsymbol{x}))$$

Cornell Bowers C·IS

Variational Inference



- Introduce a learnable variational approximation of the posterior

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

- Bound the likelihood with the ELBO

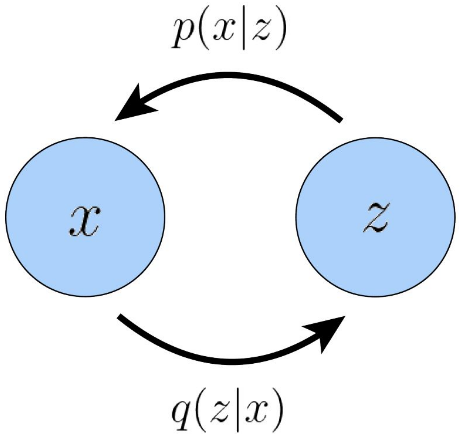
$$\log p(\boldsymbol{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right] + D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}|\boldsymbol{x}))$$

$$\geq \boxed{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right]}$$

Non-negativity of KL

Tractable; ELBO

The Evidence Lower Bound (ELBO)



- Maximize the ELBO
- Either:
  - Maximizes the likelihood of the observed data
  - Improves the approximation of the unknown posterior

$$\boxed{\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_\phi(\boldsymbol{z}|\boldsymbol{x})}\right]} = \boxed{\log p(\boldsymbol{x})} - \boxed{D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}|\boldsymbol{x}))}$$

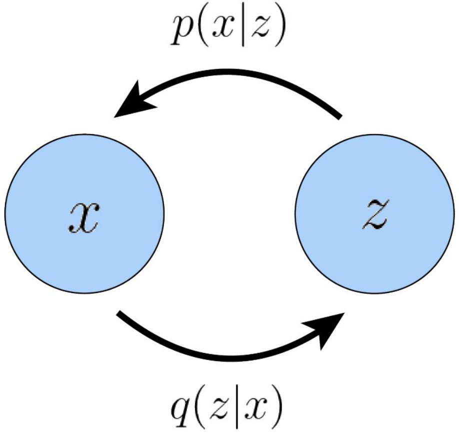
Tractable; ELBO

Intractable; Evidence

Intractable; Divergence between approximate and true posterior

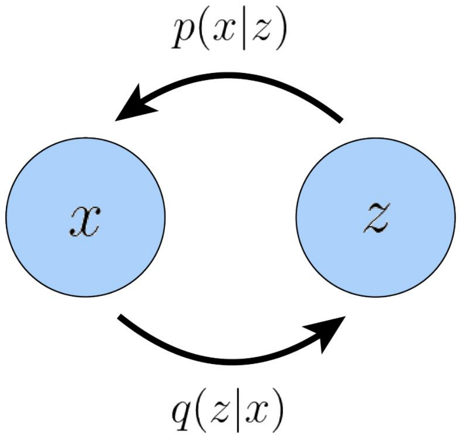
Cornell Bowers C·IS

The Evidence Lower Bound (ELBO)



$$\mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] = \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] \qquad \text{(Chain Rule of Probability)}$$

The Evidence Lower Bound (ELBO)

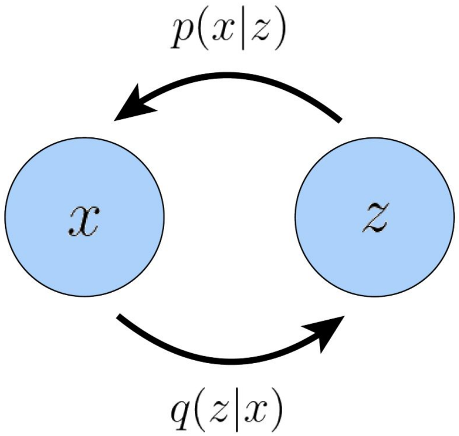


$$\mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] = \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] \qquad \text{(Chain Rule of Probability)}$$

$$= \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right] + \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] \qquad \text{(Split the Expectation)}$$

Cornell Bowers C·IS

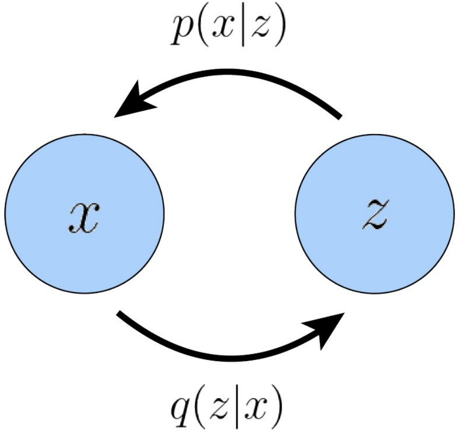
The Evidence Lower Bound (ELBO)



$$
\begin{aligned}
\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right] &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right] && \text{(Chain Rule of Probability)} \\
&= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right] + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{z})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\right] && \text{(Split the Expectation)} \\
&= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right] - D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z})) && \text{(Definition of KL Divergence)}
\end{aligned}
$$

Cornell Bowers CIS

The Evidence Lower Bound (ELBO)



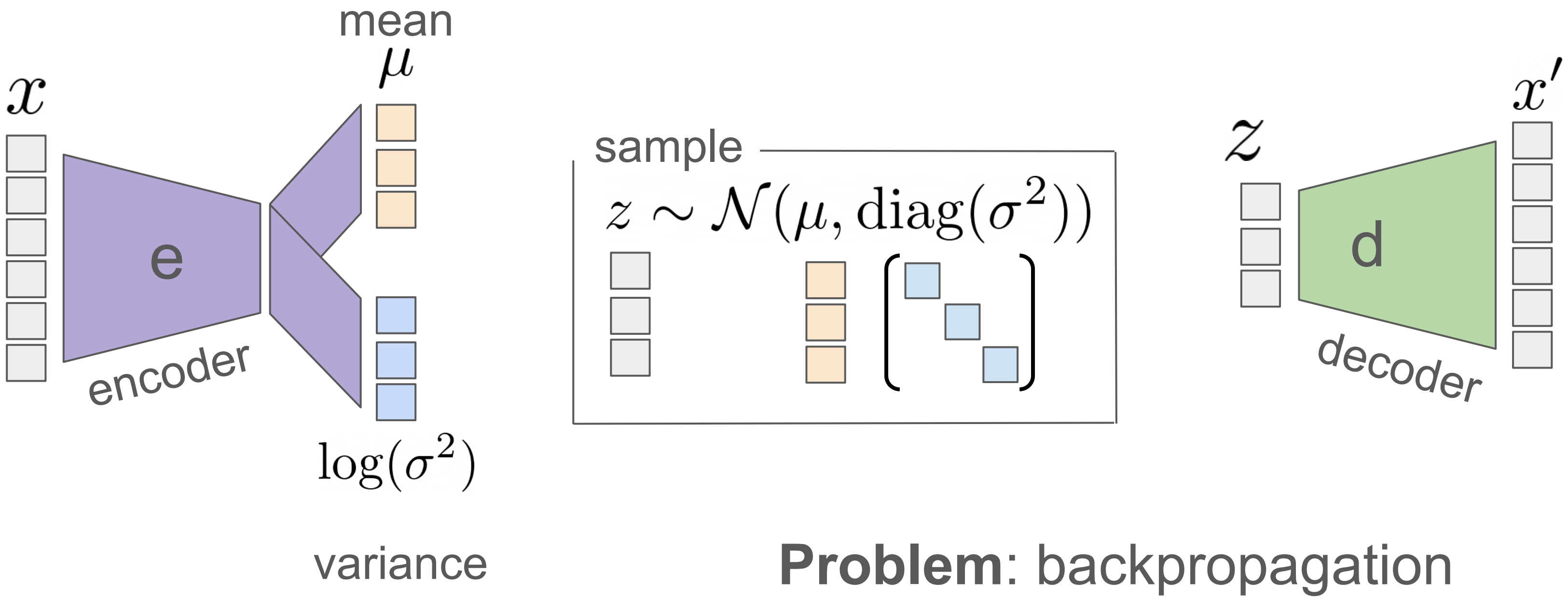
$$
\begin{aligned}
\mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x},\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] &= \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] && \text{(Chain Rule of Probability)}\\
&= \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right] + \mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{z})}{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\right] && \text{(Split the Expectation)}\\
&= \underbrace{\mathbb{E}_{q_{\phi}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q_{\phi}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}))}_{\text{prior matching term}} && \text{(Definition of KL Divergence)}
\end{aligned}
$$

Mean-Squared Error/Cross-Entropy

Regularization

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

An Architecture for Gaussians

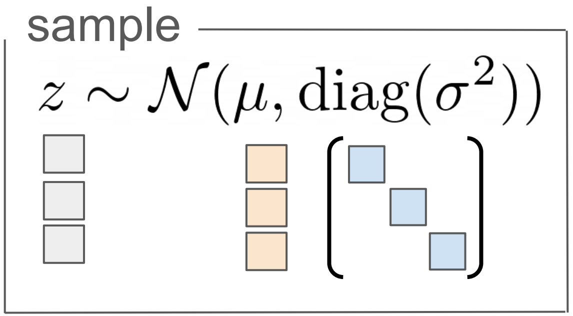


$$q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})$$

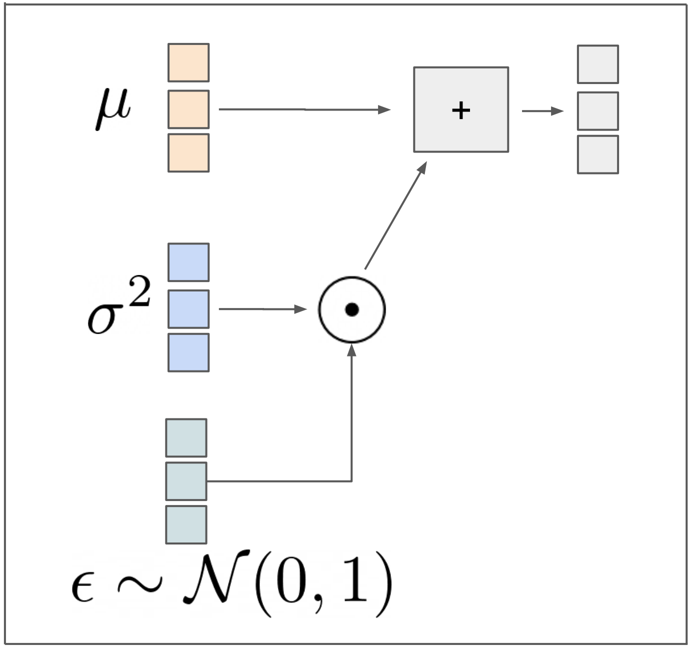
Problem: backpropagation through sampling process?

The Reparameterization Trick

$$\mathcal{N}(\mu, \mathrm{diag}(\sigma^2)) = \mu + \sigma^2 \odot \mathcal{N}(0, I)$$

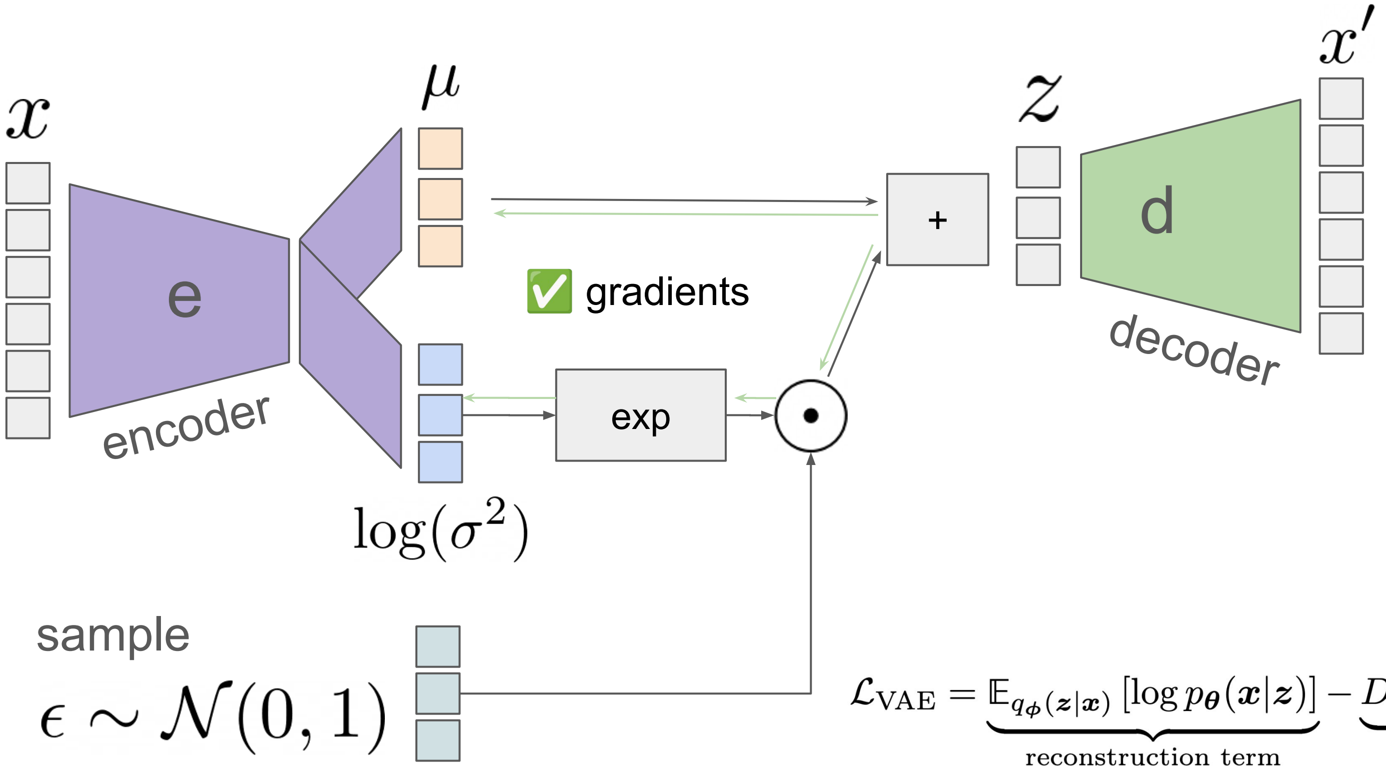


$$=$$



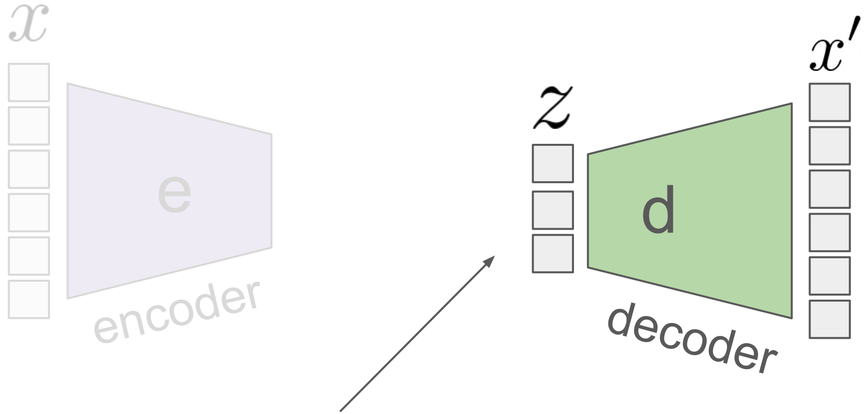
Cornell Bowers C·IS

The Reparameterization Trick



$$\mathcal{L}_{\text{VAE}} = \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z})\right]}_{\text{reconstruction term}} - \underbrace{D_{\text{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x}) \parallel p(\boldsymbol{z}))}_{\text{prior matching term}}$$

Sampling from a VAE

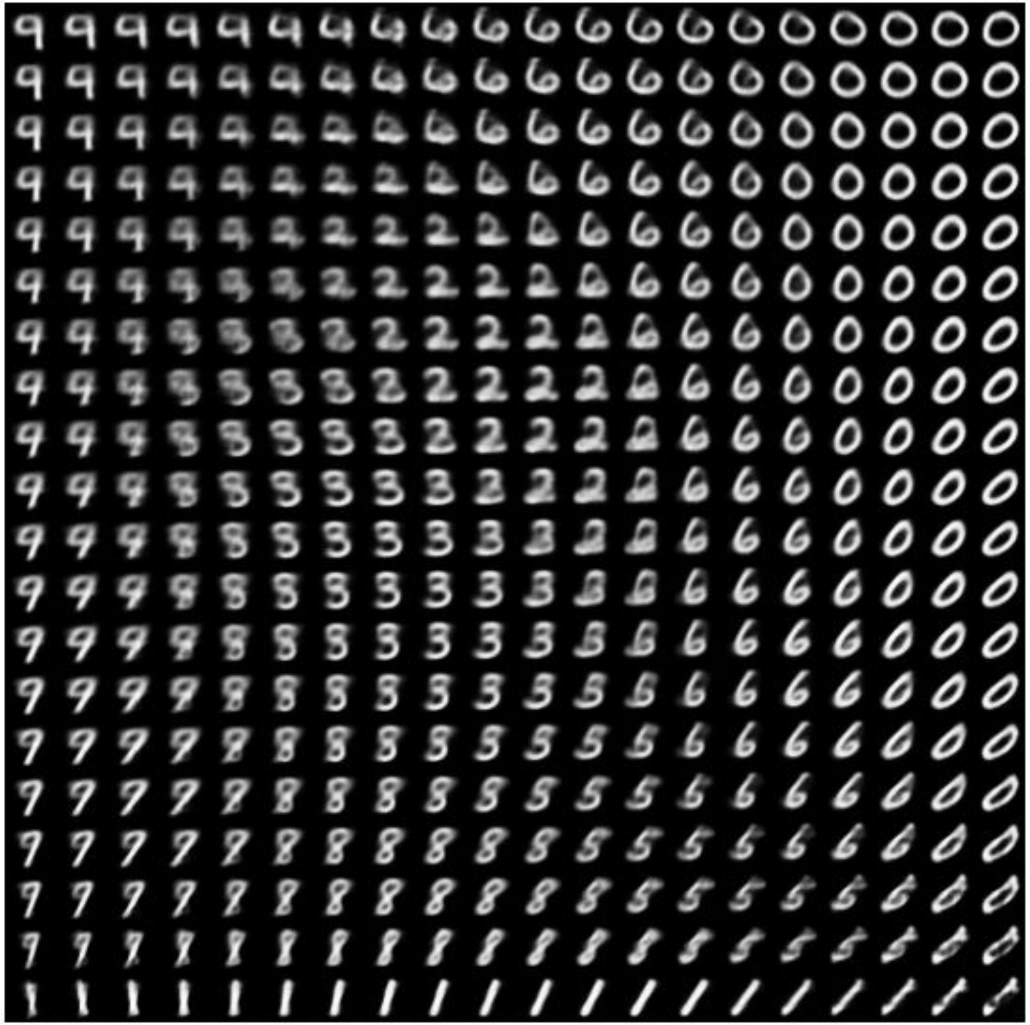


$$z \sim p(\boldsymbol{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$$

Cornell Bowers C·IS

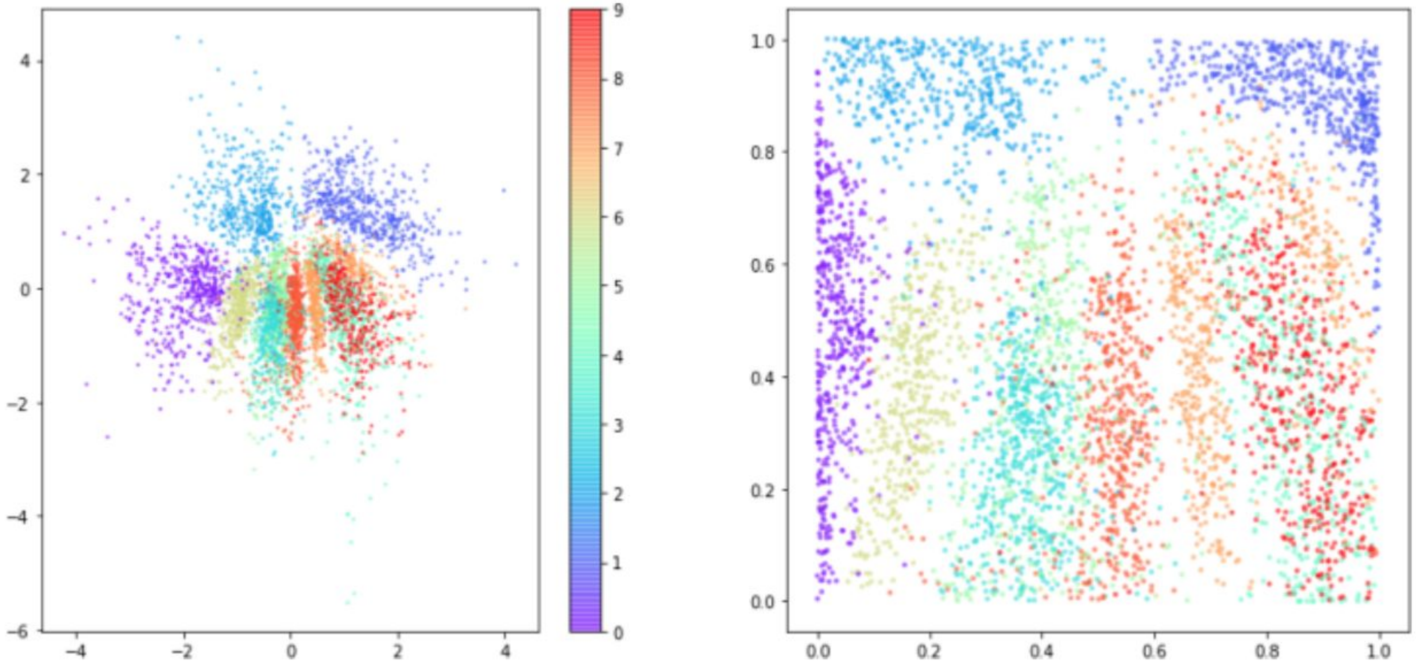
a much nicer space…

can smoothly interpolate digits in a meaningful, digit-y kind of way



Back to MNIST: Visualizing latent space again

VAE Latent space, note the distribution is centered, and each digit has an equal portion

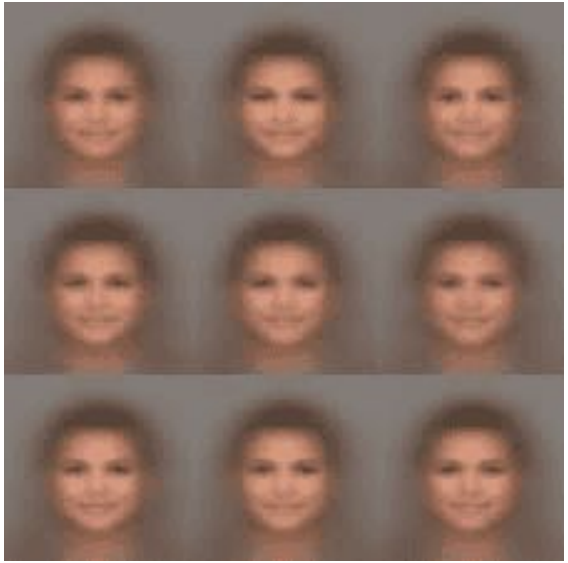


The Biggest Drawback of VAEs

- Out of the box, generated images can be blurry.

Question: Why?

VAE v. GAN



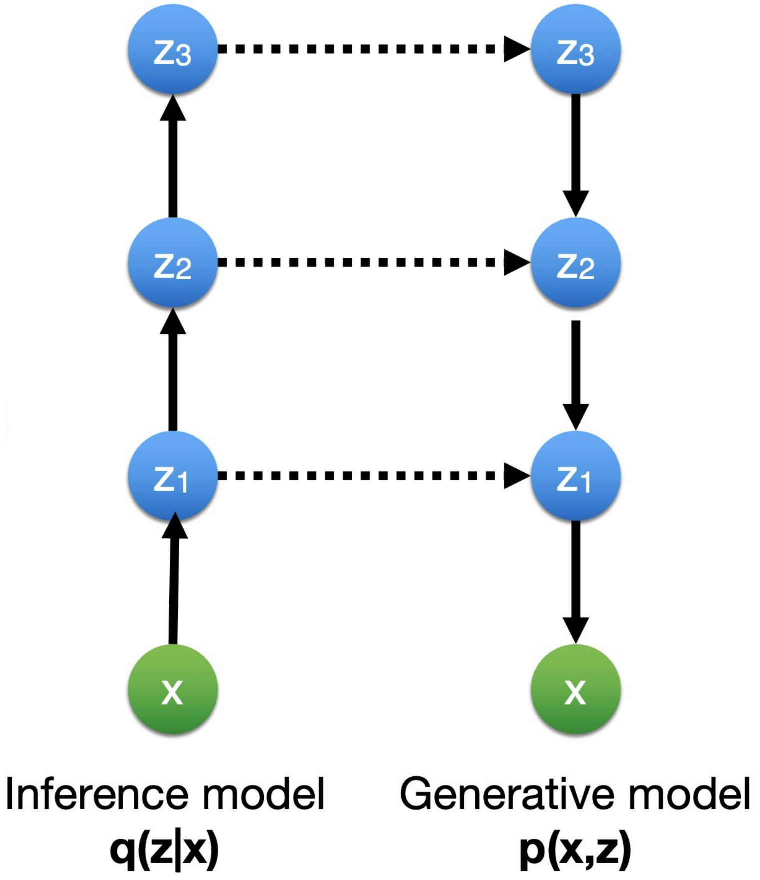
https://borisburkov.net/2022-12-31-1/

Hierarchical VAEs

- "Flat" VAEs suffer from simple priors
- Define a hierarchical generative process

$$q_\phi(\mathbf{z}_{1,2,3}|\mathbf{x}) = q_\phi(\mathbf{z}_1|\mathbf{x})q_\phi(\mathbf{z}_2|\mathbf{z}_1)q_\phi(\mathbf{z}_3|\mathbf{z}_2)$$

$$p_\theta(\mathbf{z}_{1,2,3}) = p_\theta(\mathbf{z}_3)p_\theta(\mathbf{z}_2|\mathbf{z}_3)p_\theta(\mathbf{z}_1|\mathbf{z}_2)p_\theta(\mathbf{x}|\mathbf{z}_1)$$



Extending the ELBO

- ELBO derivation is unchanged

$$\log p(\boldsymbol{x}) = \log \int p(\boldsymbol{x}, \boldsymbol{z}_{1:T}) d\boldsymbol{z}_{1:T}$$

$$\geq \mathbb{E}_{q_{\phi}(\boldsymbol{z}_{1:T}|\boldsymbol{x})} \left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}_{1:T})}{q_{\phi}(\boldsymbol{z}_{1:T}|\boldsymbol{x})} \right]$$

Extending the ELBO

- Omitting some steps
  - See “Understanding Diffusion Models: A Unified Perspective” by Calvin Luo for a nice walkthrough

$$\log p(\boldsymbol{x}) = \log \int p(\boldsymbol{x}, \boldsymbol{z}_{1:T}) d\boldsymbol{z}_{1:T}$$

$$\geq \mathbb{E}_{q_{\phi}(\boldsymbol{z}_{1:T}|\boldsymbol{x})} \left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}_{1:T})}{q_{\phi}(\boldsymbol{z}_{1:T}|\boldsymbol{x})} \right]$$

$$= \dots$$

Extending the ELBO

KL-Div between Gaussians

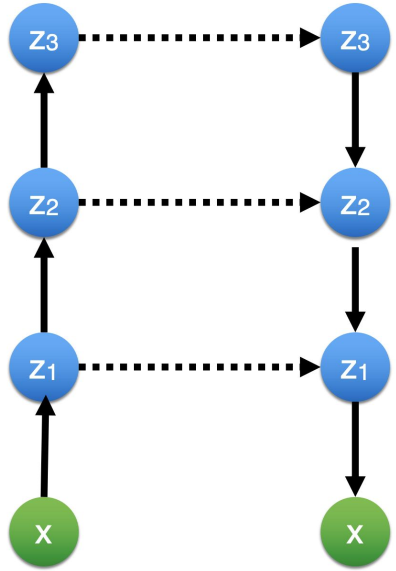
$$\frac{1}{2}\left\{\left(\frac{\sigma_0}{\boxed{\sigma_1}}\right)^2 + \frac{(\mu_1-\mu_0)^2}{\boxed{\sigma_1^2}} - 1 + \ln\frac{\sigma_1^2}{\sigma_0^2}\right\}$$

$$\begin{aligned}
\log p(\boldsymbol{x}) &= \log \int p(\boldsymbol{x}, \boldsymbol{z}_{1:T}) d\boldsymbol{z}_{1:T} \\
&\geq \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_{1:T}|\boldsymbol{x})}\left[\log \frac{p(\boldsymbol{x}, \boldsymbol{z}_{1:T})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}_{1:T}|\boldsymbol{x})}\right] \\
&= \dots \\
&= \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_1|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_1)\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_T|\boldsymbol{x}) \parallel p(\boldsymbol{z}_T))}_{\text{prior matching term}} - \boxed{\sum_{t=2}^{T} \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_t|\boldsymbol{x})}\left[D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t, \boldsymbol{x}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t))\right]}
\end{aligned}$$

Consistency Term- Unstable Optimization!

Extending the ELBO

Hard to jointly learn hierarchical encoders and decoders!



Inference model
q(z|x)

Generative model
p(x,z)

$$= \underbrace{\mathbb{E}_{q_{\phi}(\boldsymbol{z}_1|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{z}_1)\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q_{\phi}(\boldsymbol{z}_T|\boldsymbol{x}) \parallel p(\boldsymbol{z}_T))}_{\text{prior matching term}} - \boxed{\sum_{t=2}^{T} \mathbb{E}_{q_{\phi}(\boldsymbol{z}_t|\boldsymbol{x})}\left[D_{\mathrm{KL}}(q_{\phi}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t, \boldsymbol{x}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z}_{t-1}|\boldsymbol{z}_t))\right]}$$

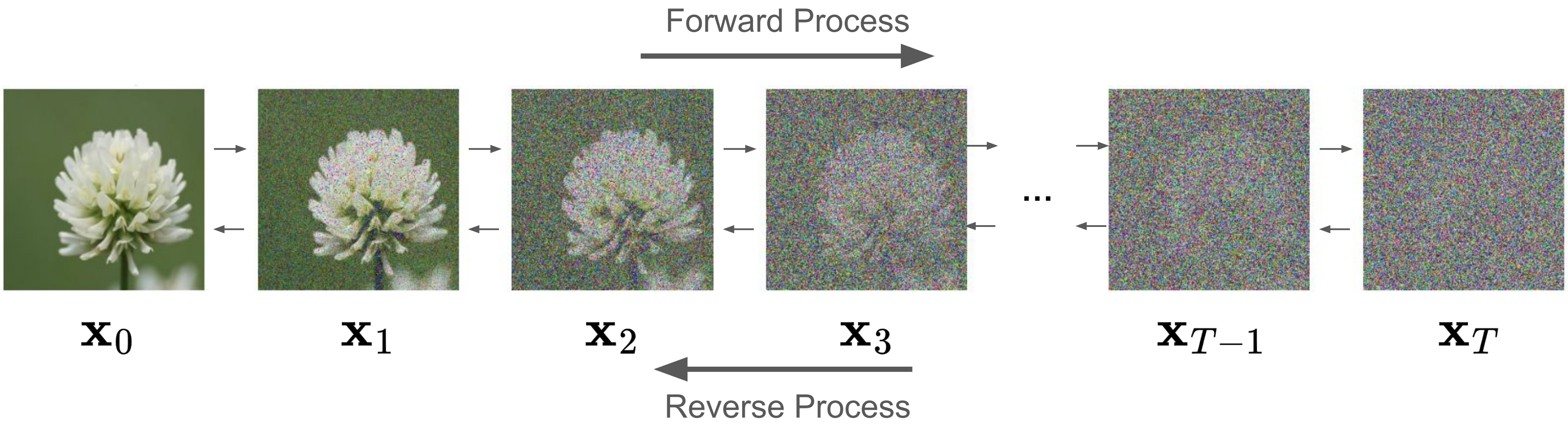
Consistency Term- Unstable Optimization!

Cornell Bowers C·IS

Denoising Diffusion Models

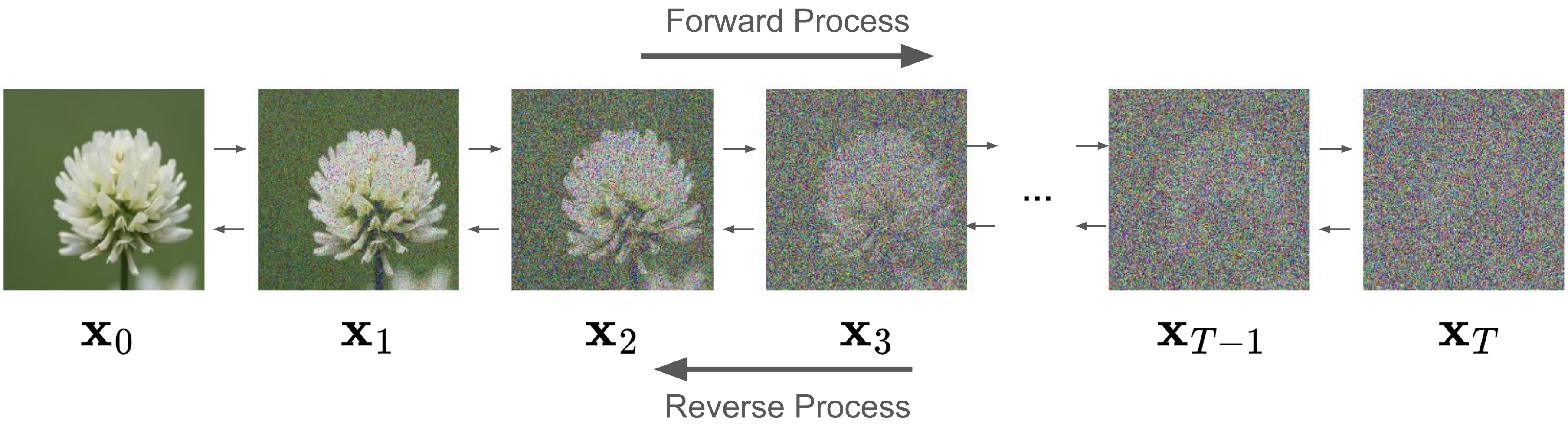
Denoising diffusion models consist of two processes:

- Forward diffusion process that gradually adds noise to input
- Reverse denoising process that learns to generate data by denoising



Discuss:

- How to define the forward and reverse directions?



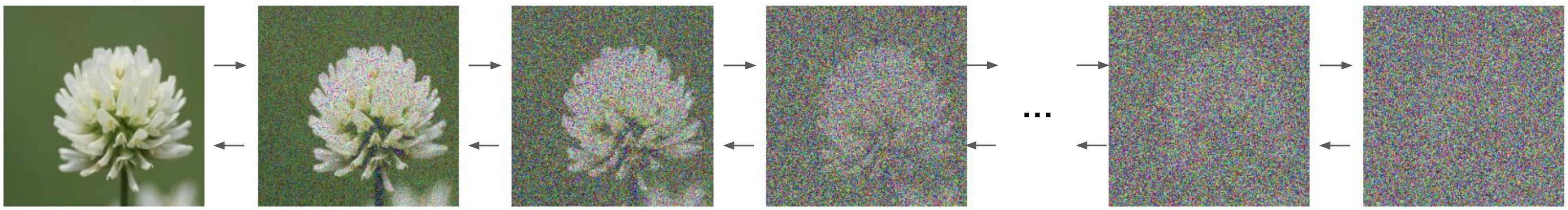
Cornell Bowers C·IS

Forward Process: high level idea

Forward Process (think encoder)

Destroy by successively adding Gaussian noise

(Markov Chain)



$\mathbf{x}_0$ Training Sample

$\mathbf{x}_1$

$\mathbf{x}_2$

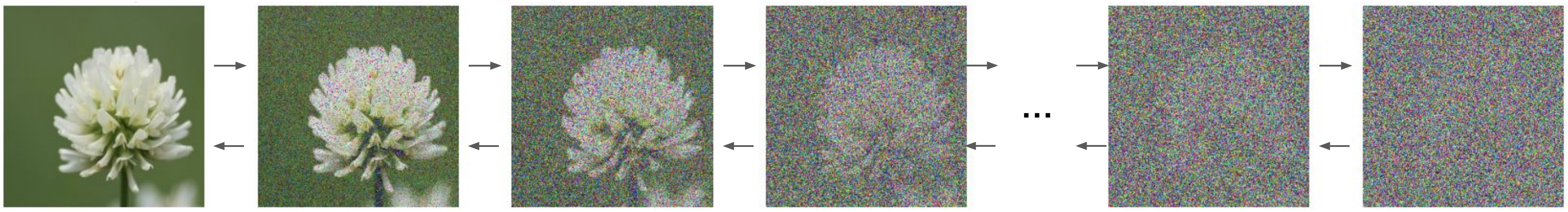
$\mathbf{x}_3$

...

$\mathbf{x}_{T-1}$

$\mathbf{x}_T$ Gaussian Noise (distribution)

Reverse Process: high level idea



$\mathbf{x}_0$ $\mathbf{x}_1$ $\mathbf{x}_2$ $\mathbf{x}_3$ ... $\mathbf{x}_{T-1}$ $\mathbf{x}_T$

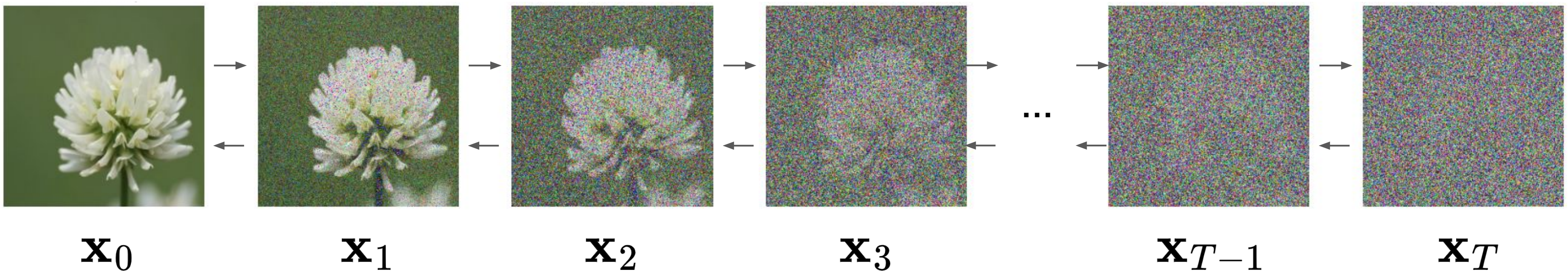
Training Sample

Gaussian Noise (distribution)

Reverse Process (think decoder)
Learn to recover original image
(i.e. learn to reverse the Markov Chain)

Putting it together

Forward Process (think encoder)
Destroy by successively adding Gaussian noise
(Markov Chain)



Training Sample

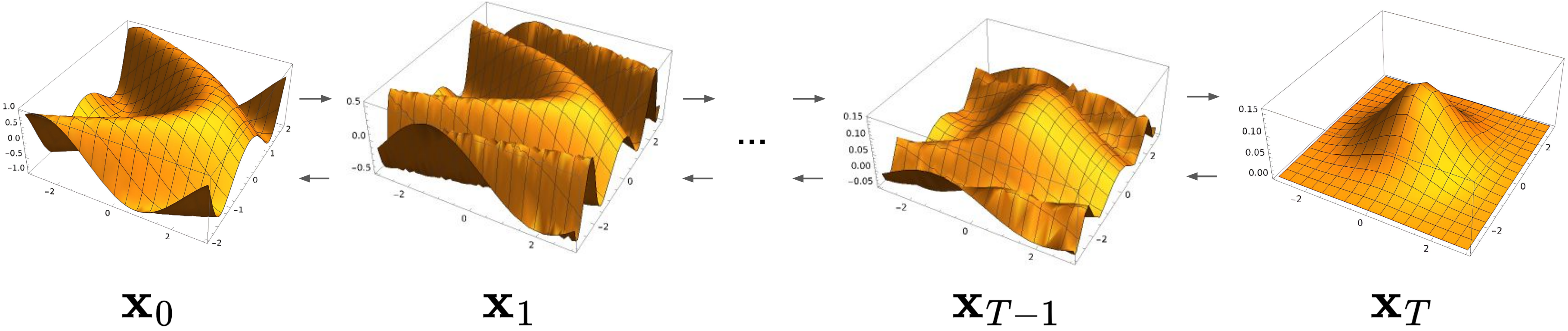
Gaussian Noise (distribution)

Reverse Process (think decoder)
Learn to recover original image
(i.e. learn to reverse the Markov Chain)

Cornell Bowers C·IS

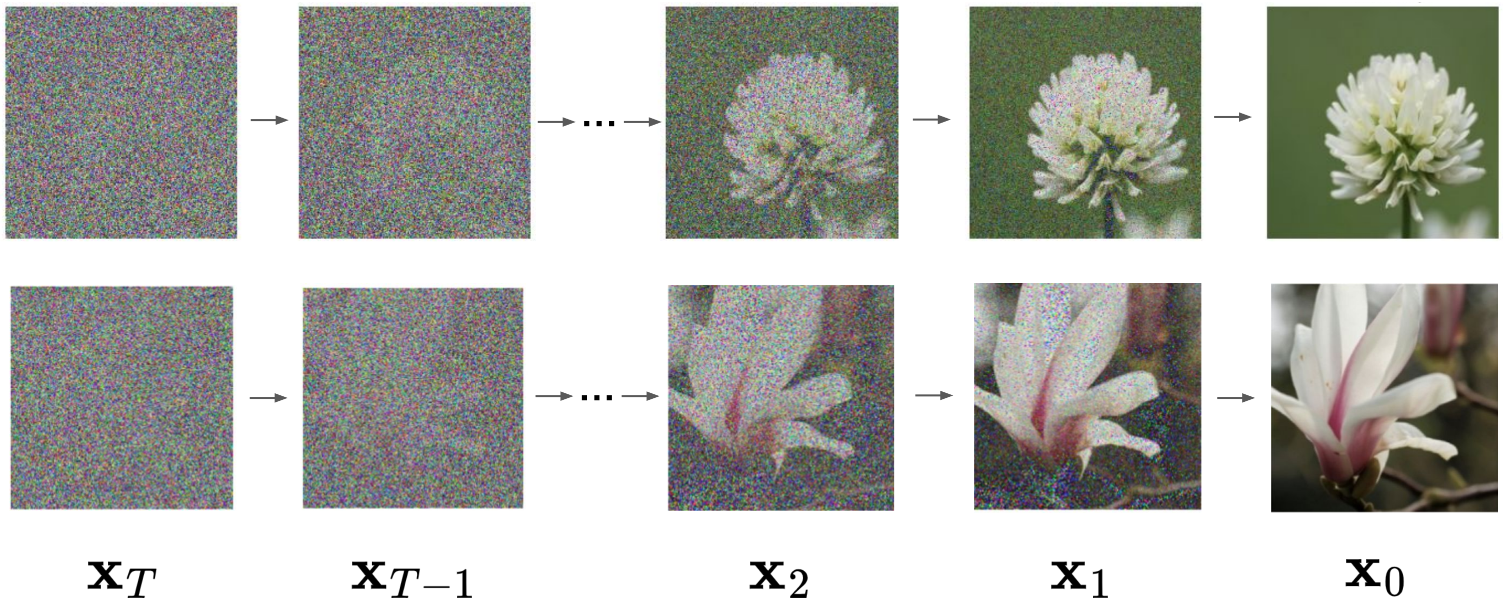
Diffusion Models

We define a mapping to Gaussian noise (forward process)
Want to **learn the reverse mapping to generate data** (reverse process)



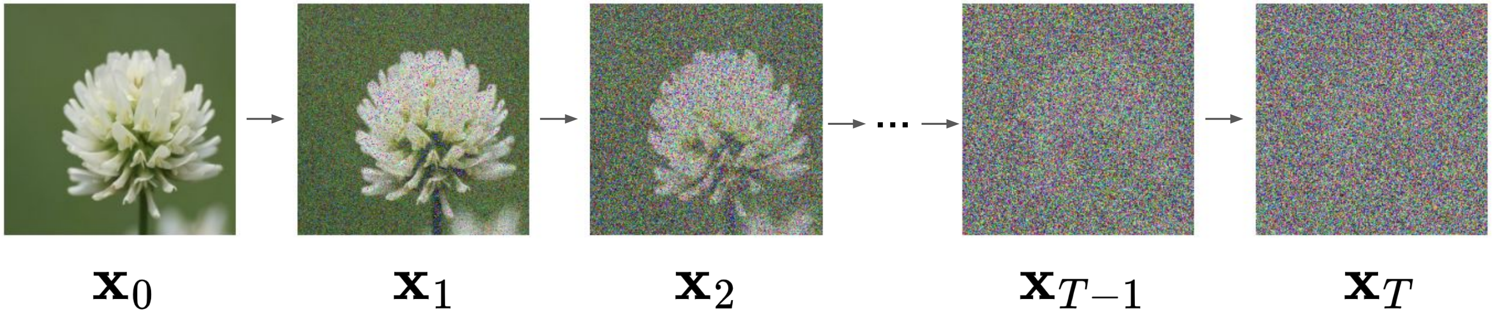
Diffusion Sampling

Different draws of initial noise lead to diverse of outputs



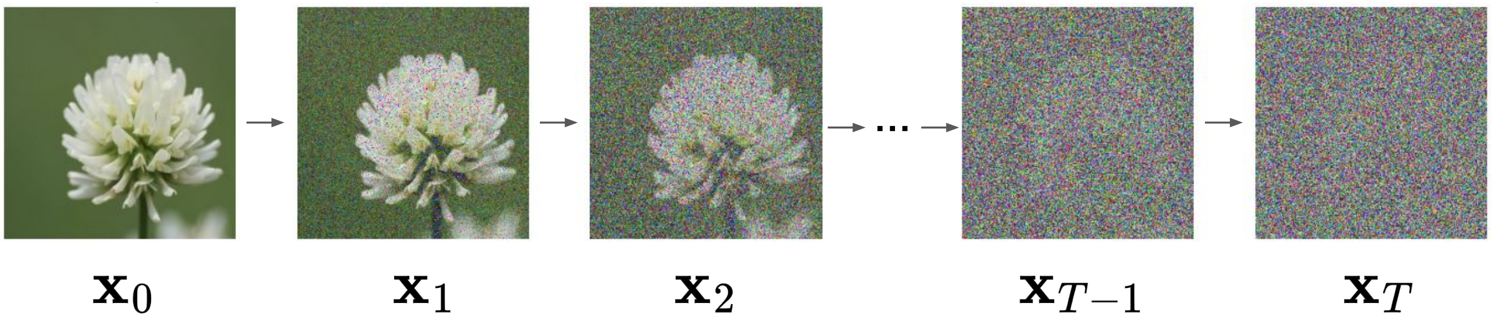
Forward Process Overview

- Destroys original image $\mathbf{x}_0$ by **successively adding Gaussian noise**
- Desired outcome: At step T , $\mathbf{x}_T$ is a **pure Gaussian noise**
  - i.e. the distribution we map the data manifold to



Details: Forward Process

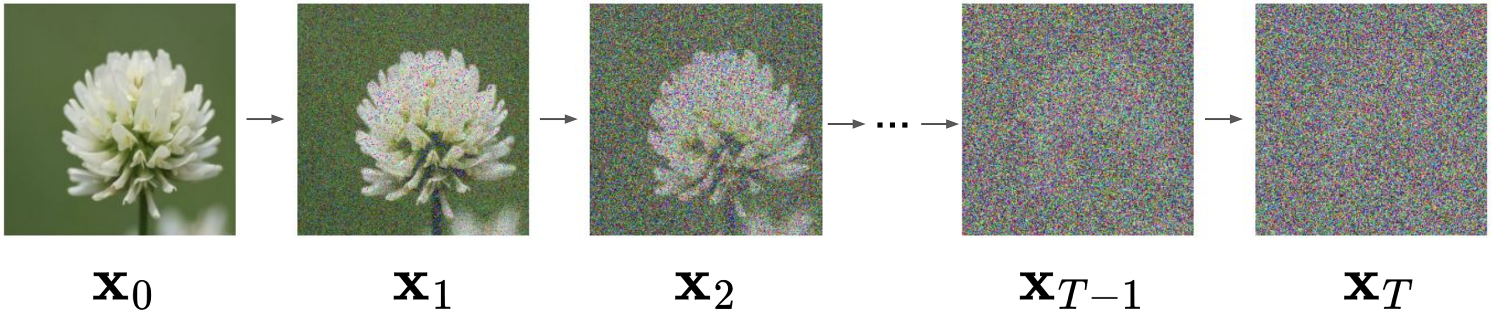
1. $\mathbf{x}_0$ sampled from some distribution



Details: Forward Process

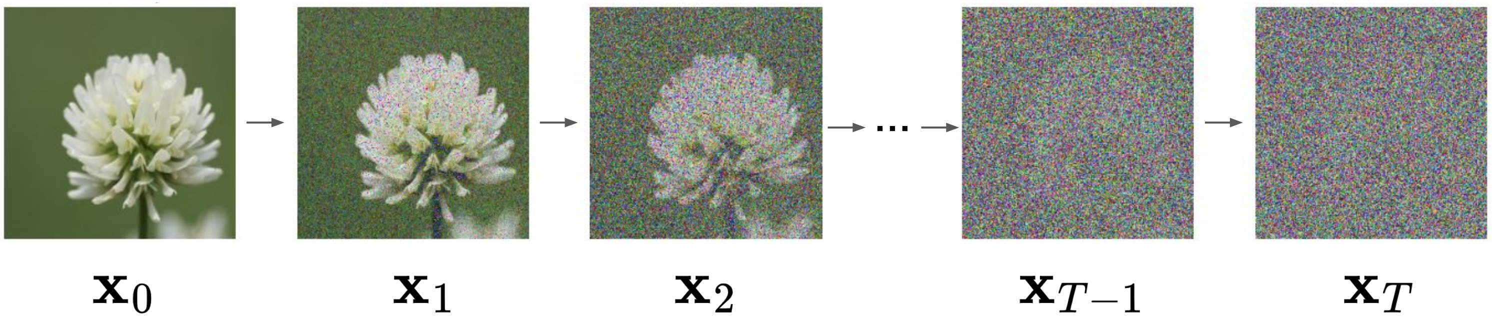
2. $\mathbf{x}_t$ sampled from normal distribution conditioned on $\mathbf{x}_{t-1}$ given by:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}) \qquad \{\beta_t \in (0,1)\}_{t=1}^{T}$$



Details: Forward Process

$\{\beta_t \in (0,1)\}_{t=1}^{T}$ is variance schedule (controlling **how** we move toward Gaussian noise)



Details: Forward Process

$\mathbf{x}_t$ sampled from normal distribution conditioned on $\mathbf{x}_{t-1}$ given by:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$$

Can we extend this to sampling $\mathbf{x}_t$ in a closed form? We use the re-parametrization trick:

Let $\alpha_t \coloneqq 1 - \beta_t$, and let $\boldsymbol{\epsilon}_{t-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

$$\mathbf{x}_t = \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1}$$

Cornell Bowers C·IS

Details: Forward Process

Inductively, we can say

$$\begin{aligned}\mathbf{x}_t &= \sqrt{\alpha_t}\mathbf{x}_{t-1} + \sqrt{1-\alpha_t}\boldsymbol{\epsilon}_{t-1} \\ &= \sqrt{\alpha_t\alpha_{t-1}}\mathbf{x}_{t-2} + \underline{\sqrt{1-\alpha_t\alpha_{t-1}}\bar{\boldsymbol{\epsilon}}_{t-2}} \\ &= \dots \\ &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}\end{aligned}$$

Merged noise.
epsilon is still $\sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

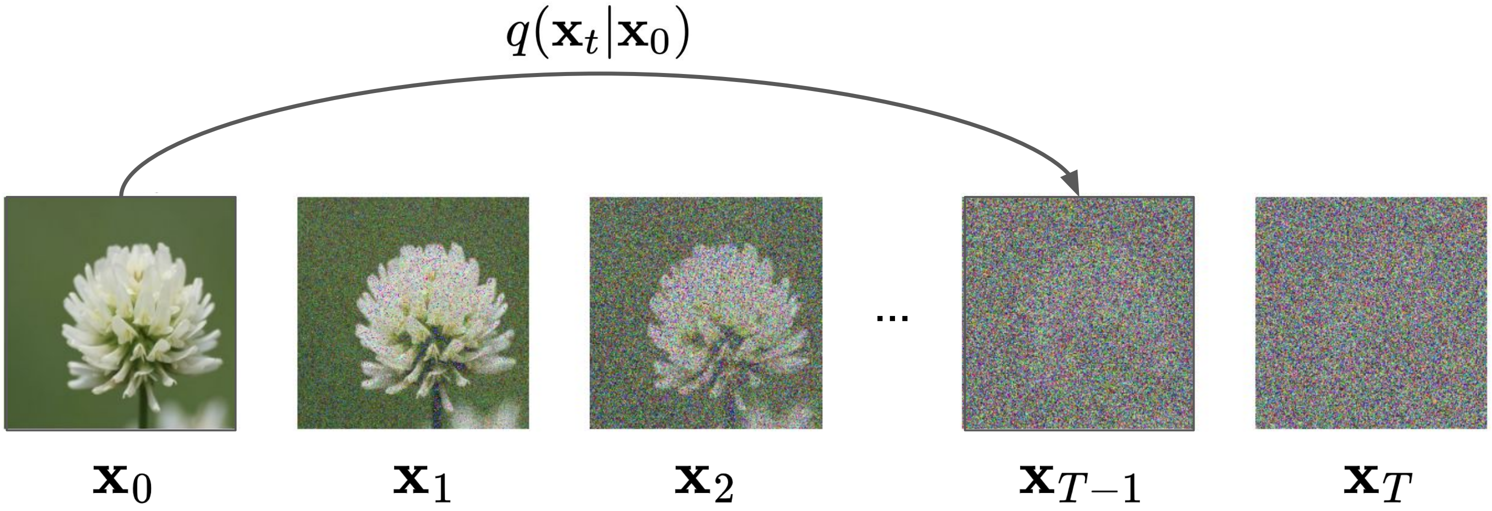
$$\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$$

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

Details: Forward Process

Can sample $\mathbf{x}_t$ in closed-form as $q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1-\bar{\alpha}_t)\mathbf{I})$

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



Cornell Bowers C·IS

Aside: Noise Schedules

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

- Define the noise schedule in terms of $\bar{\alpha}_t \in (0,1)$
  - Some monotonically decreasing function from 1 to 0

- Cosine Noise schedule:

$$\bar{\alpha}_t = \cos(.5\pi t/T)^2$$

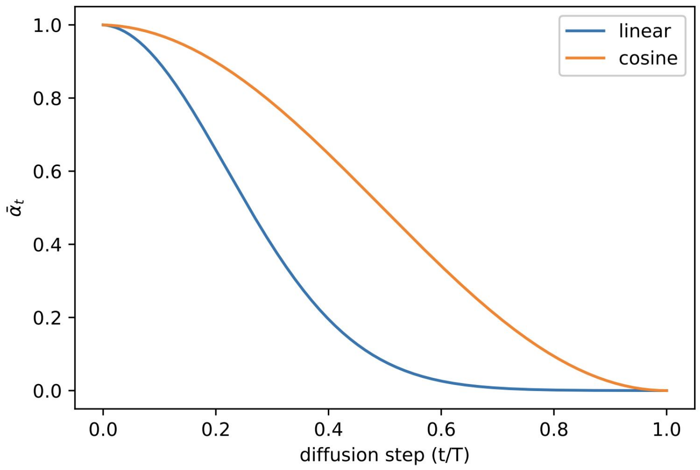
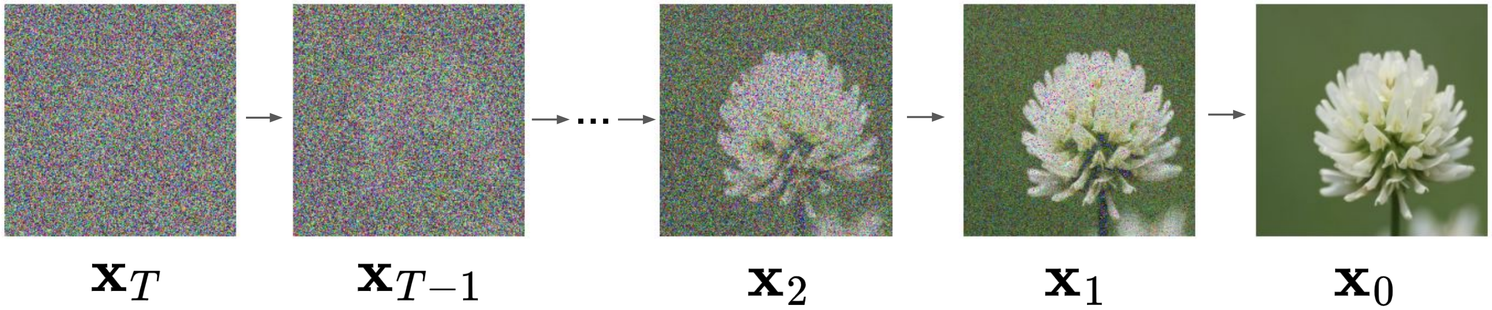


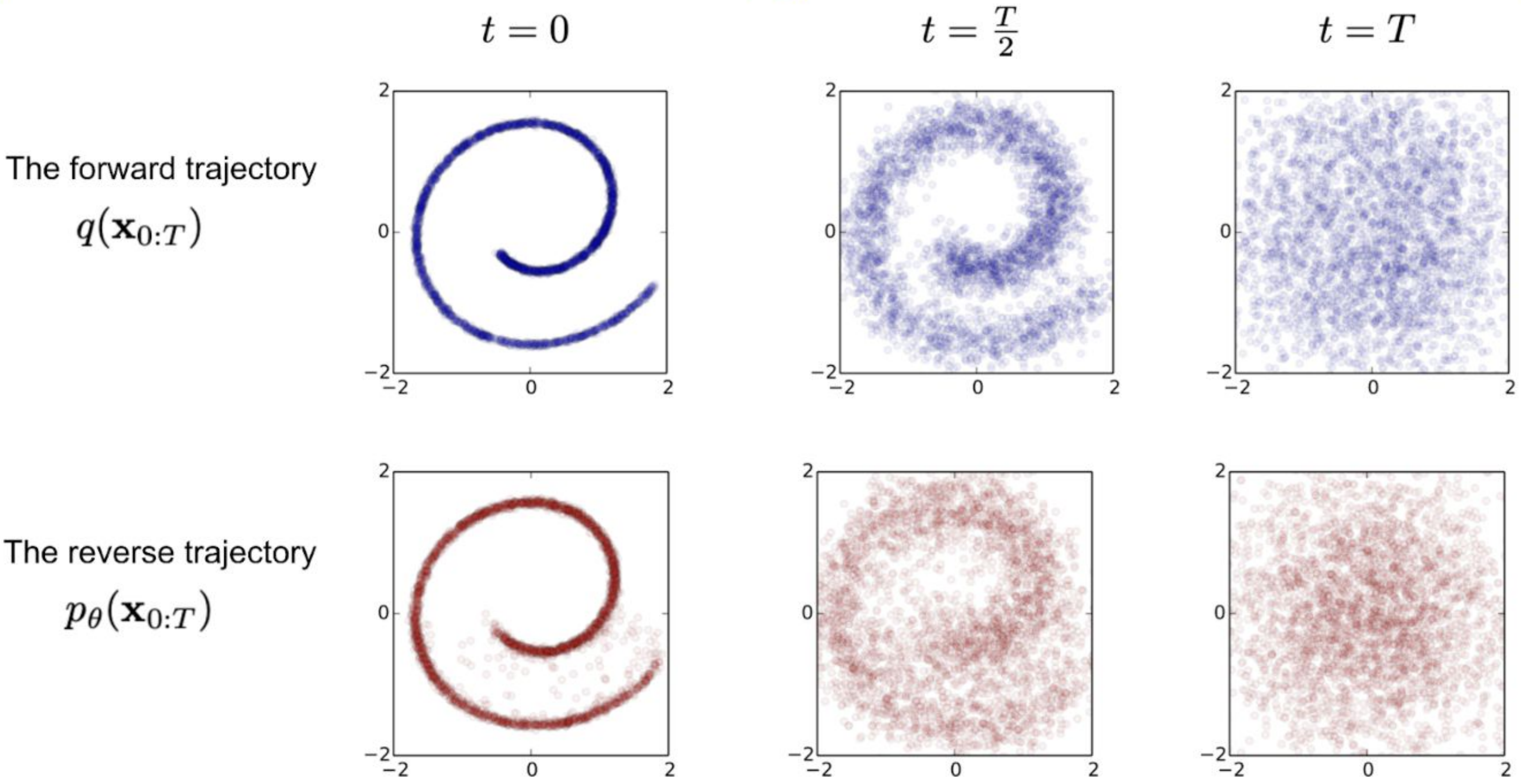
Figure 5. $\bar{\alpha}_t$ throughout diffusion in the linear schedule and our proposed cosine schedule.

Nichol, Alexander Quinn, and Prafulla Dhariwal. "Improved denoising diffusion probabilistic models." International conference on machine learning. PMLR, 2021.

Reverse Process Overview

- "Learn to reverse what we just destroyed"
  - Learn time reversal of Markov Chain; we train a model for this
- Desired outcome: some $\mathbf{x}_0$ close to the original data distribution





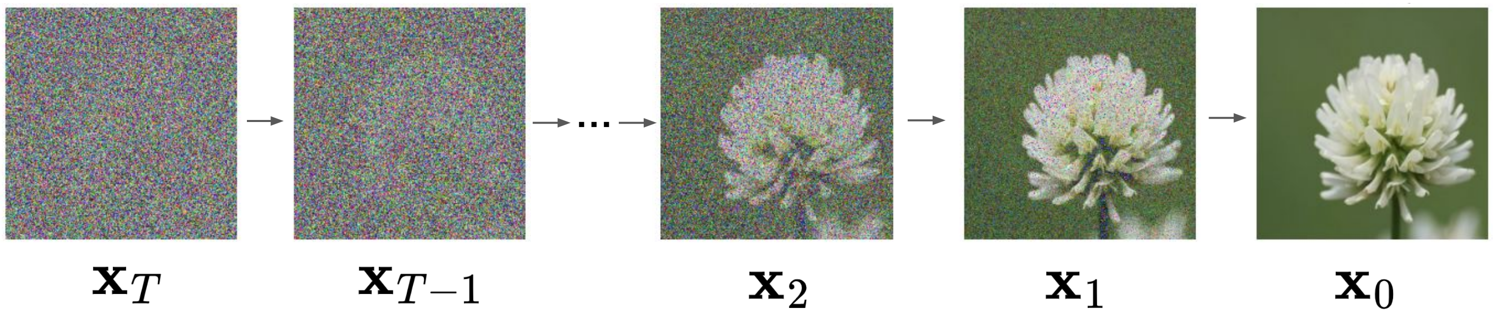
Sohl-Dickstein et al., 2015

Reverse process:

Cornell Bowers C·IS

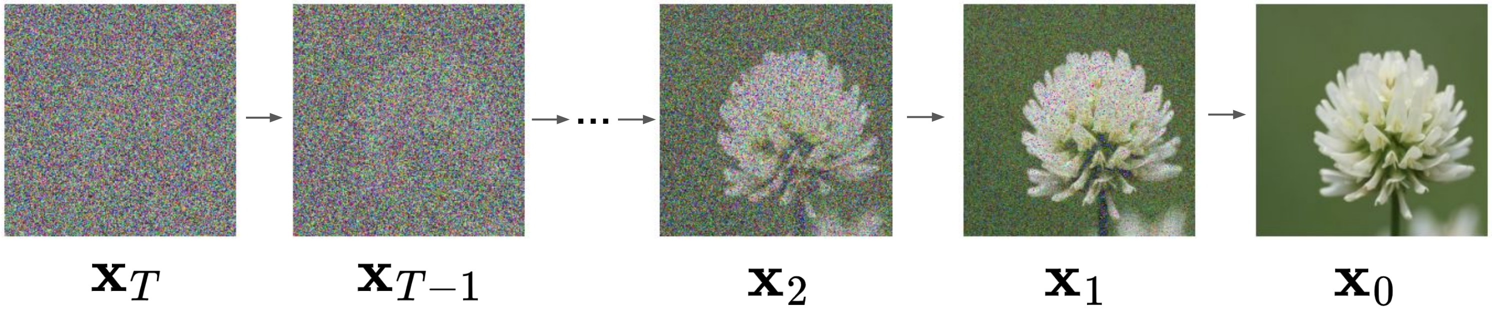
Details: Reverse Process

1. Ideally, sample from reversed conditional distribution $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$



Details: Reverse Process

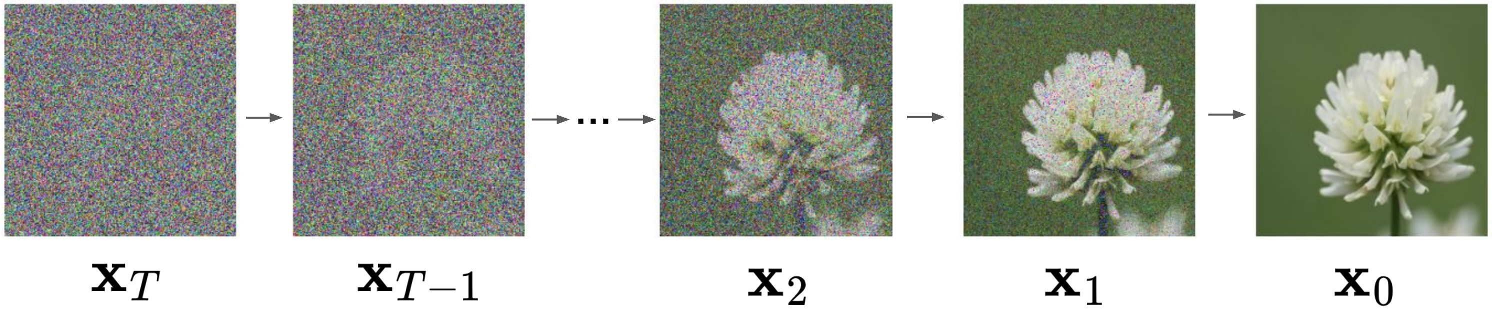
Problem: $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is **intractable** (can't compute easily)



Details: Reverse Process

Problem: $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$ is **intractable** (can't compute easily)

You need to use the entire dataset!

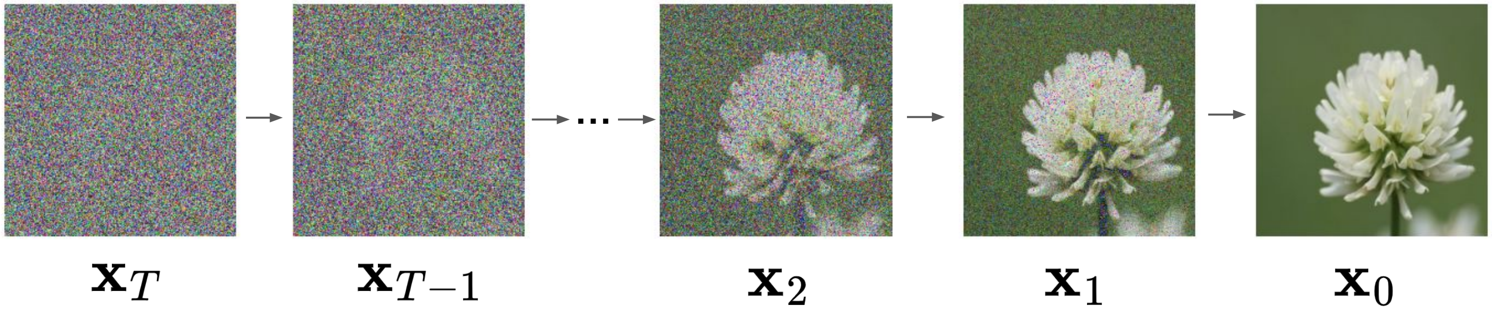


Details: Reverse Process

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

However: $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$ is **tractable**

Can reverse the forward process given the original data!



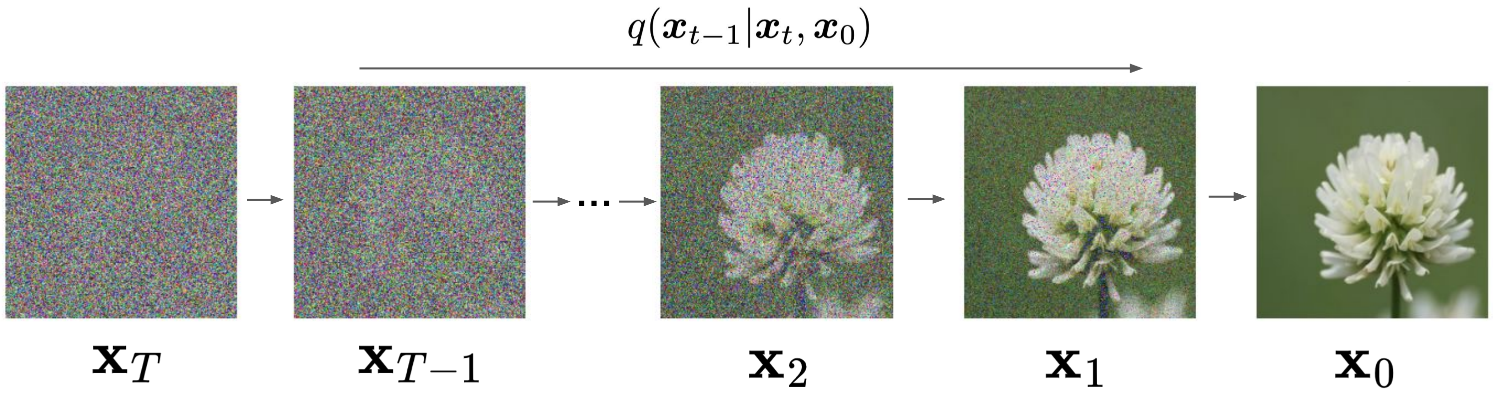
Details: Reverse Process

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

However: $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$ is **tractable**

Can reverse the forward process given the original data!

Problem: Don't have any "original data" for inference

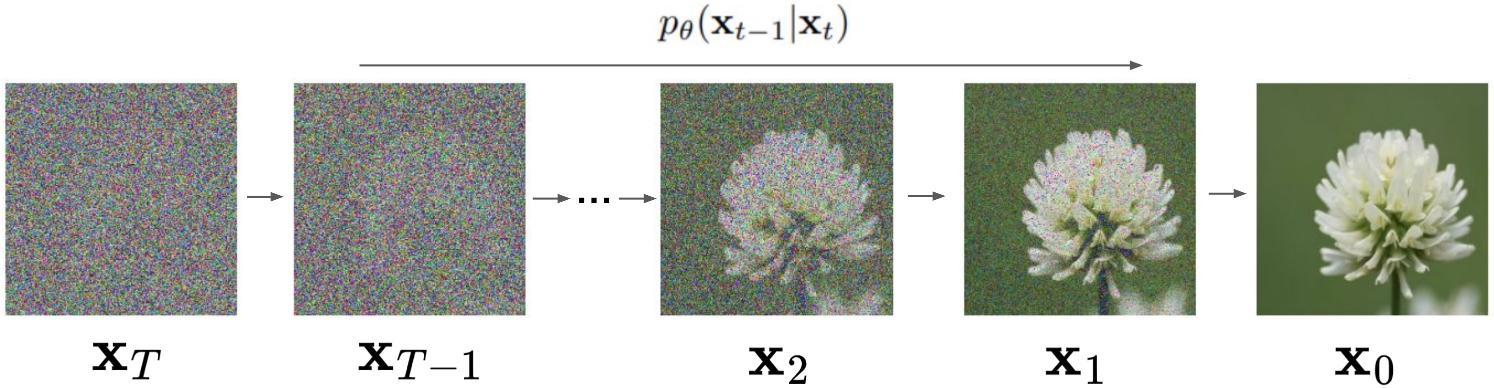


Key Idea

We introduce a generative model to approximate the reverse process:

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$$
$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2 \mathbf{I})$$

$$\Rightarrow \quad p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$



Key Idea

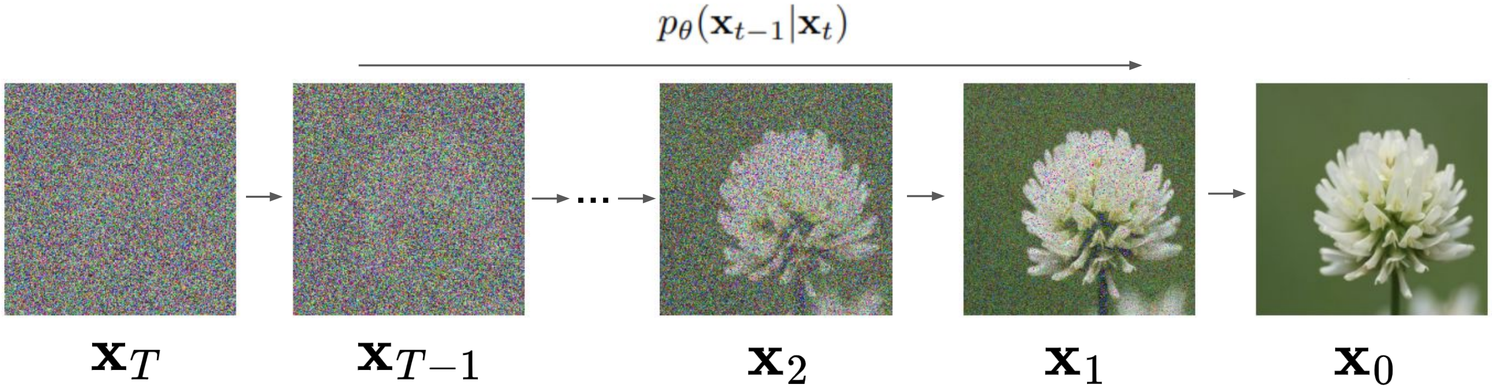
We introduce a generative model to approximate the reverse process:

$$p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$$
$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_t^2\mathbf{I})$$

$$\Rightarrow \quad p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$$

Learning Objective!

$$\boxed{\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]}$$



Training: Principled Derivation

Find the model that maximizes the likelihood of the training data

i.e. same as VAEs, variational inference; approximate the true posterior

Cornell Bowers C·IS

Training Objective

- Bound the likelihood with the ELBO
  - Exactly like hierarchical VAEs

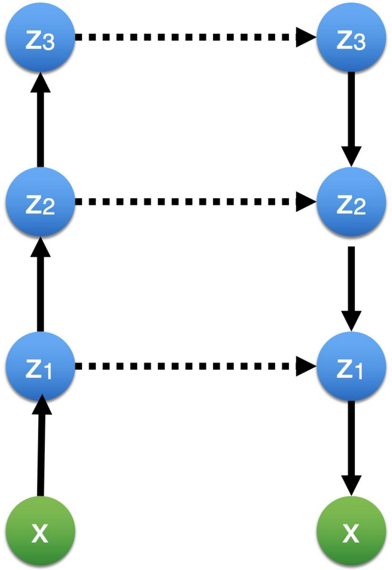
Learning Objective!

$$
\begin{aligned}
\log p(\boldsymbol{x}) &\geq \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\right] \\
&= \underbrace{\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_1)\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q(\boldsymbol{x}_T|\boldsymbol{x}_0) \parallel p(\boldsymbol{x}_T))}_{\text{prior matching term}} - \boxed{\sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]}_{\text{denoising matching term}}}
\end{aligned}
$$

Cornell Bowers C·IS

Discuss:

Differences between diffusion models and hierarchical VAEs?



Inference model
q(z|x)

Generative model
p(x,z)

$$\log p(\boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\right]$$

Learning Objective!

$$= \underbrace{\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_1)\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q(\boldsymbol{x}_T|\boldsymbol{x}_0) \parallel p(\boldsymbol{x}_T))}_{\text{prior matching term}} - \boxed{\sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]}_{\text{denoising matching term}}}$$

http://cs231n.stanford.edu/slides/2023/lecture_15.pdf

Cornell Bowers C·IS

Training Objective

- Bound the likelihood with the ELBO
  - Exactly like VAEs

Learning Objective!

$$\log p(\boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\left[\log \frac{p(\boldsymbol{x}_{0:T})}{q(\boldsymbol{x}_{1:T}|\boldsymbol{x}_0)}\right]$$

$$= \underbrace{\mathbb{E}_{q(\boldsymbol{x}_1|\boldsymbol{x}_0)}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_0|\boldsymbol{x}_1)\right]}_{\text{reconstruction term}} - \underbrace{D_{\mathrm{KL}}(q(\boldsymbol{x}_T|\boldsymbol{x}_0) \parallel p(\boldsymbol{x}_T))}_{\text{prior matching term}} - \boxed{\sum_{t=2}^{T} \underbrace{\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]}_{\text{denoising matching term}}}$$

where $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ is the tractable posterior distribution:

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

$$\text{where } \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon\right) \text{ and } \tilde{\beta}_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$$

Parameterizing the Denoising Model

Since both $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ and $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ are Normal distributions, the KL divergence has a simple form:

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

Recall that $\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\ \mathbf{x}_0 + \sqrt{(1-\bar{\alpha}_t)}\ \epsilon$. Ho et al. NeurIPS 2020 observe that:

$$\tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon\right)$$

They propose to represent the mean of the denoising model using a *noise-prediction* network:

$$\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{1-\beta_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boxed{\epsilon_\theta(\mathbf{x}_t, t)}\right)$$

With this parameterization

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0\sim q(\mathbf{x}_0), \epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\frac{\beta_t^2}{2\sigma_t^2(1-\beta_t)(1-\bar{\alpha}_t)}||\epsilon - \boxed{\epsilon_\theta(\underbrace{\sqrt{\bar{\alpha}_t}\ \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon}_{\mathbf{x}_t}, t)}||^2\right] + C$$

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

Cornell Bowers C·IS

Training Objective Weighting

ELBO objective leads to a specific regression weight at each time step:

$$L_{t-1} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{I})} \left[\underbrace{\frac{\beta_t^2}{2\sigma_t^2(1-\beta_t)\boxed{(1-\bar{\alpha}_t)}}}_{\lambda_t} ||\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t}\ \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon, t)||^2 \right]$$

Approaches zero!

However, this weight is often very large for small t's

Ho et al., 2020 proposed the following objective to improve perceptual quality:

$$L_{\text{simple}} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(\mathbf{0},\mathbf{I}), t \sim \mathcal{U}(1,T)} \left[||\epsilon - \epsilon_\theta(\underbrace{\sqrt{\bar{\alpha}_t}\ \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\ \epsilon}_{\mathbf{x}_t}, t)||^2 \right]$$

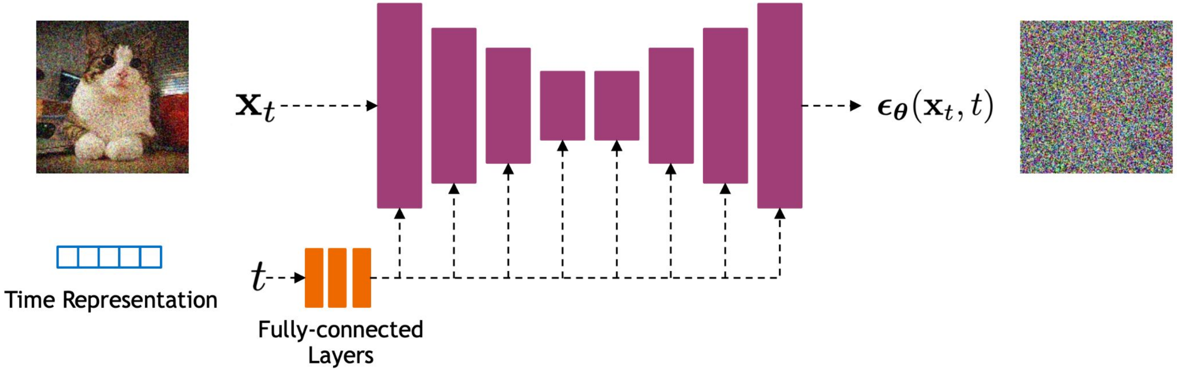
http://cs231n.stanford.edu/slides/2023/lecture_15.pdf

Cornell Bowers C·IS

What Network Architecture to Use For ϵ_θ ?

People often use U-Nets with residual blocks and self-attention layers at low resolutions

Has same input and output image dimensions



Time representation: sinusoidal positional embeddings

Inject time embedding throughout the network (e.g. additive positional embedding)

http://cs231n.stanford.edu/slides/2023/lecture_15.pdf

Diffusion Results

Outperforms prior generative models when using the **simplified** training objective

ELBO objective performs worse!

ELBO

$$L_{\text{simple}} = \mathbb{E}_{\mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t \sim \mathcal{U}(1,T)} \left[||\epsilon - \epsilon_\theta(\underbrace{\sqrt{\bar{\alpha}_t}\, \mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\, \epsilon}_{\mathbf{x}_t}, t)||^2 \right]$$

Model	IS	FID
Gated PixelCNN [59]	4.60	65.93
Sparse Transformer [7]		
PixelIQN [43]	5.29	49.46
EBM [11]	6.78	38.2
NCSNv2 [56]		31.75
NCSN [55]	8.87±0.12	25.32
SNGAN [39]	8.22±0.05	21.7
SNGAN-DDLS [4]	9.09±0.10	15.42
StyleGAN2 + ADA (v1) [29]	**9.74** ± 0.05	3.26
Ours (L, fixed isotropic $\mathbf{\Sigma}$)	7.67±0.13	13.51
Ours (L_{simple})	9.46±0.11	**3.17**

Ho et al. 2020

Cornell Bowers C·IS

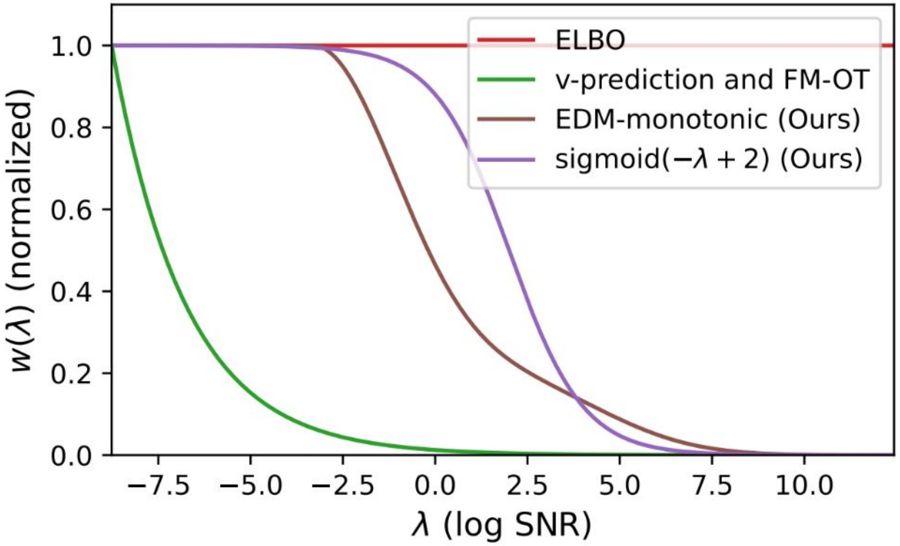
Training Objective Weighting

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

$$\mathrm{SNR}(t) = \bar{\alpha_t}/(1-\bar{\alpha_t})$$

$$\log(\mathrm{SNR}(t)) = \log(\bar{\alpha_t}/(1-\bar{\alpha_t}))$$

- ELBO forces the network to model imperceptible details
  - Less modeling capacity dedicated to perceptible details (global image structure, etc.)
- If you care about perceptual quality:
  - Decrease the loss weighting for low noise levels



Kingma, Diederik, and Ruiqi Gao. "Understanding diffusion objectives as the ELBO with simple data augmentation." Advances in Neural Information Processing Systems 36 (2023).

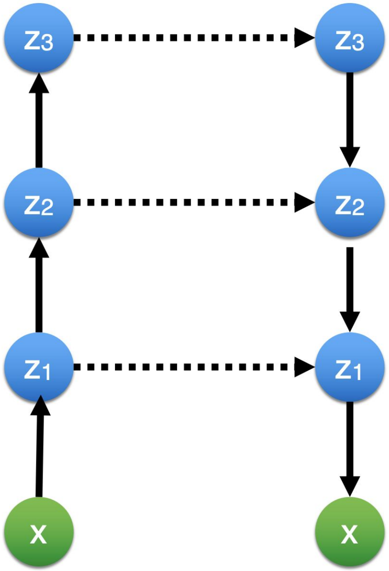
Cornell Bowers CIS

Connection to VAEs

Diffusion models can be considered as a special form of hierarchical VAEs.

However, in diffusion models:

- The inference model is fixed: easier to optimize
- The latent variables have the same dimension as the data.
- The ELBO is decomposed to each time step: fast to train
- Can be made extremely deep (even infinitely deep)
- The model is trained with some reweighting of the ELBO
    - Can trade off likelihood for improved perceptual quality



Inference model
q(z|x)

Generative model
p(x,z)

Cornell Bowers C·IS

Alternative Diffusion Parameterization: Data Prediction

Can also view the diffusion network as learning to predict the **original data**

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\epsilon$$

$$\implies \mathbf{x}_0 = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\epsilon}{\sqrt{\bar{\alpha}_t}}$$

$$\implies \mathbf{x}_\theta(\mathbf{x}_t, t) = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}$$

Cornell Bowers C·IS

Alternative Diffusion Parameterization: Data Prediction

Can also view the diffusion network as learning to predict the **original data**

$$\mathbf{x}_\theta(\mathbf{x}_t, t) = \frac{\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)}{\sqrt{\bar{\alpha}_t}}$$

Diffusion training objective: $\mathbb{E}_{q(\boldsymbol{x}_t|\boldsymbol{x}_0)}\left[D_{\mathrm{KL}}(q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t))\right]$

For sampling, want $q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \boldsymbol{x}_0)$, but don't have access to the original data

Use our estimate of the original data, $\mathbf{x}_\theta(\mathbf{x}_t, t)$, to sample:

$$p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \mathbf{x}_\theta(\mathbf{x}_t, t)) \approx q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \mathbf{x}_0)$$

Cornell Bowers C·IS

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

Training Algorithm

Repeat until convergence

1. $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ ← Sample original image from image distribution
2. $t \sim U\{1, 2, \ldots, T\}$ ← Sample random time step uniformly
3. $\epsilon \sim \mathcal{N}(0, 1)$ ← Sample Gaussian noise
4. Optimizer step on $L(\theta) = \mathbb{E}_{t,\mathbf{x}_0,\epsilon}[||\epsilon - \epsilon_\theta(\mathbf{x}_t, t)||^2]$

← Model predicts noise applied at time step t and calculate loss

Cornell Bowers C·IS

Sampling Algorithm

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ← Sample pure Gaussian noise

For $t = T, T-1 \ldots, 1$

$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$ ← Sample Gaussian noise to apply to image

$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$ ← Predict noise applied to image and remove that noise

Return $\mathbf{x}_0$

$$\boxed{p_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t) = q(\boldsymbol{x}_{t-1}|\boldsymbol{x}_t, \mathbf{x}_\theta(\mathbf{x}_t, t))}$$

Generative Modeling

VAE: maximize variational lower bound

Diffusion models: Gradually add Gaussian noise and then reverse

GAN: Adversarial training

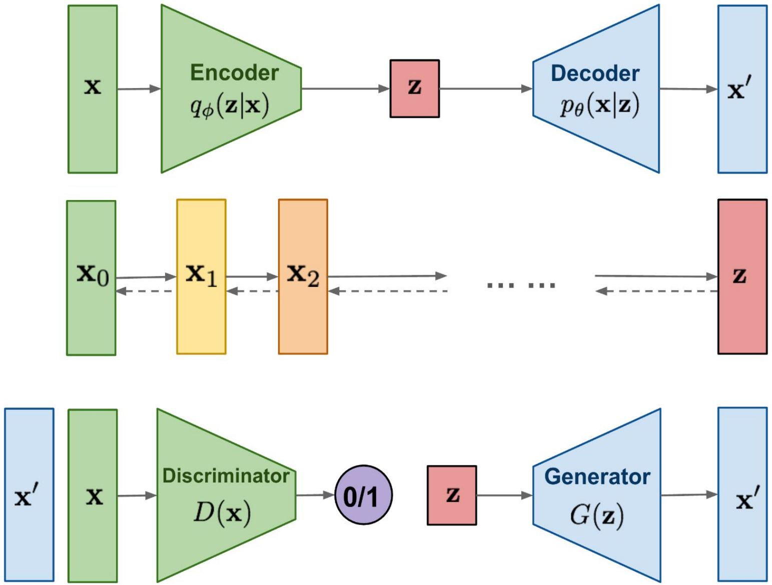
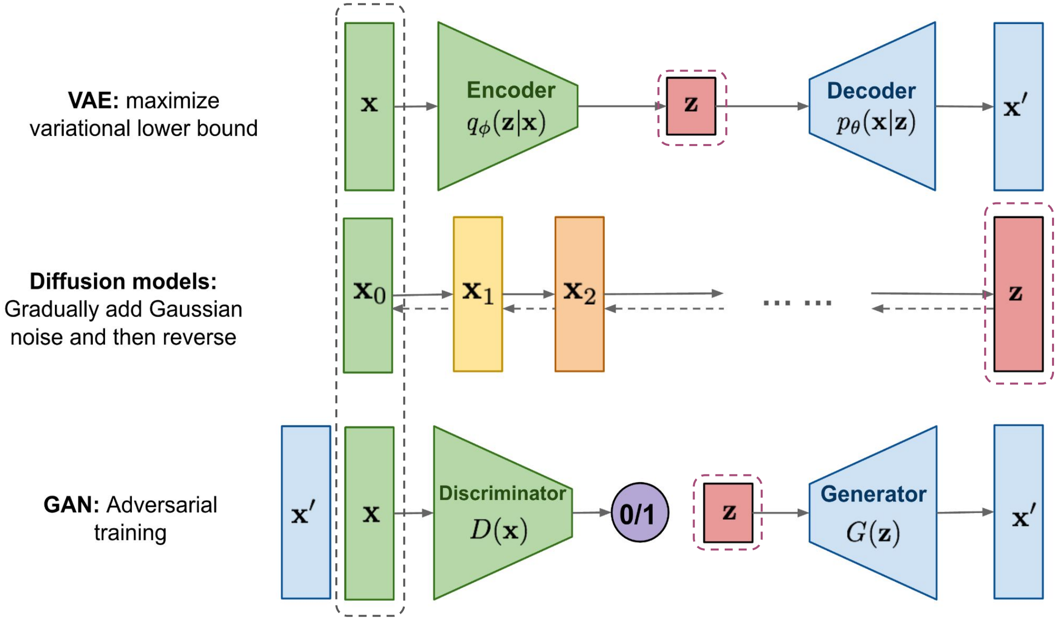


Image Source

Generative Modeling



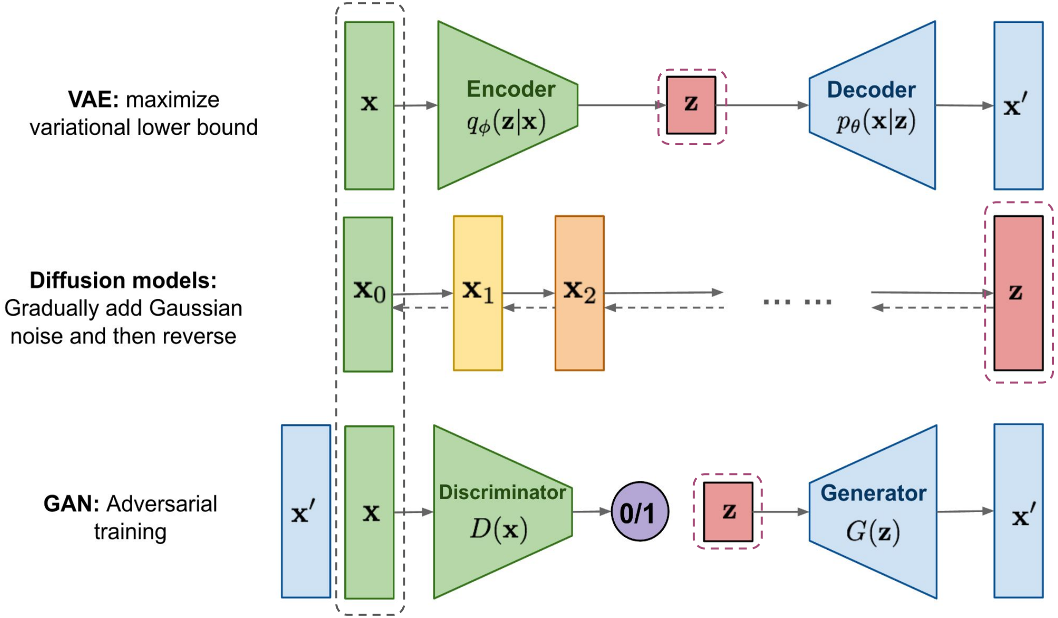
Latent Distribution

Target Distribution

Image Source

Cornell Bowers C·IS

Generative Modeling



VAEs typically have a smaller latent dimension, while diffusion models do not

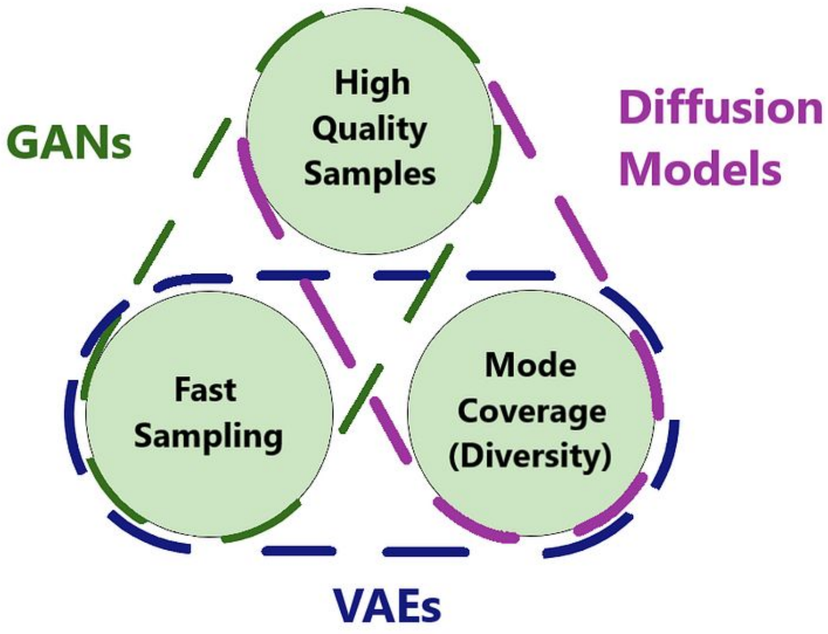
Latent Distribution

Target Distribution

Image Source

Diffusion Models vs. VAEs vs. GAN

Source



Cornell Bowers C·IS

Stable Diffusion Demo!

https://huggingface.co/spaces/stabilityai/stable-diffusion

Sample input: "messi as a real madrid player"

Recap

- Can bound the likelihood of observed data (i.e. the evidence) with the Evidence Lower Bound (i.e. the ELBO)
- Can learn generative models by maximizing the ELBO
  - VAEs, hierarchical VAEs, Diffusion models
- Diffusion models are a special case of hierarchical VAEs
  - The encoder is fixed to a linear Gaussian model
  - Only learn the decoder
  - Easy to train!
- Learning objective decomposed to each timestep
  - Can be made extremely deep!
  - Can focus on higher noise levels to improve perceptual quality!
- Limitation:
  - Can require many sampling steps for good quality