Deep Generative Models — VAEs and GANs
(part 1)

Puneet K. Dokania
(University of Oxford)

<u>Presented at</u>

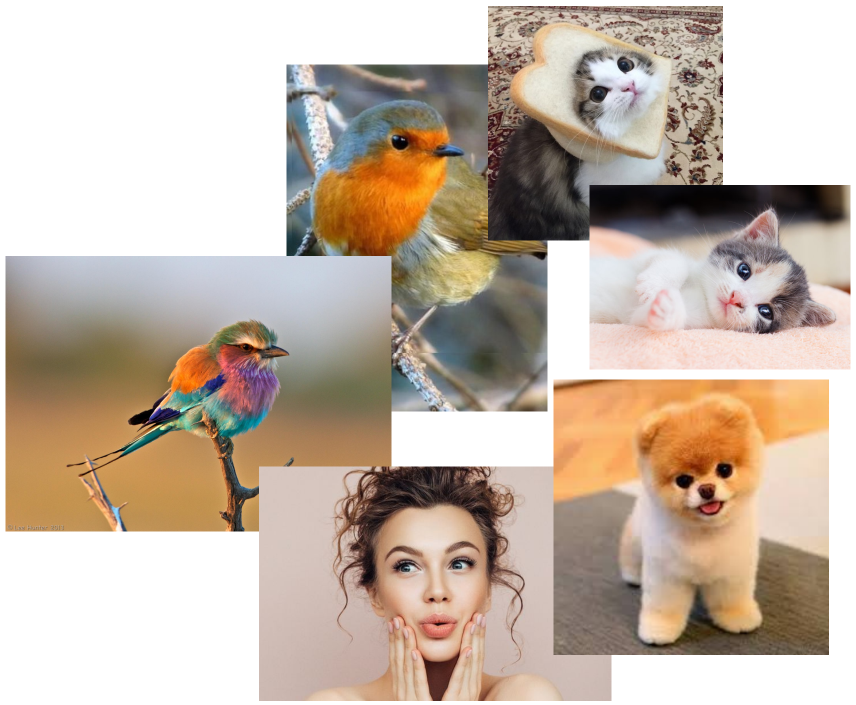
Machine Learning Summer School
IIIT Hyderabad, India
(9th July 2019)

Roadmap

- Objective
- Why?
- How?
- Classical Bayes' to Deep Generative Models
- VAEs — derive Elbo
- GANs
- GAN regularisations
- BigGAN

Generative Models — Objective

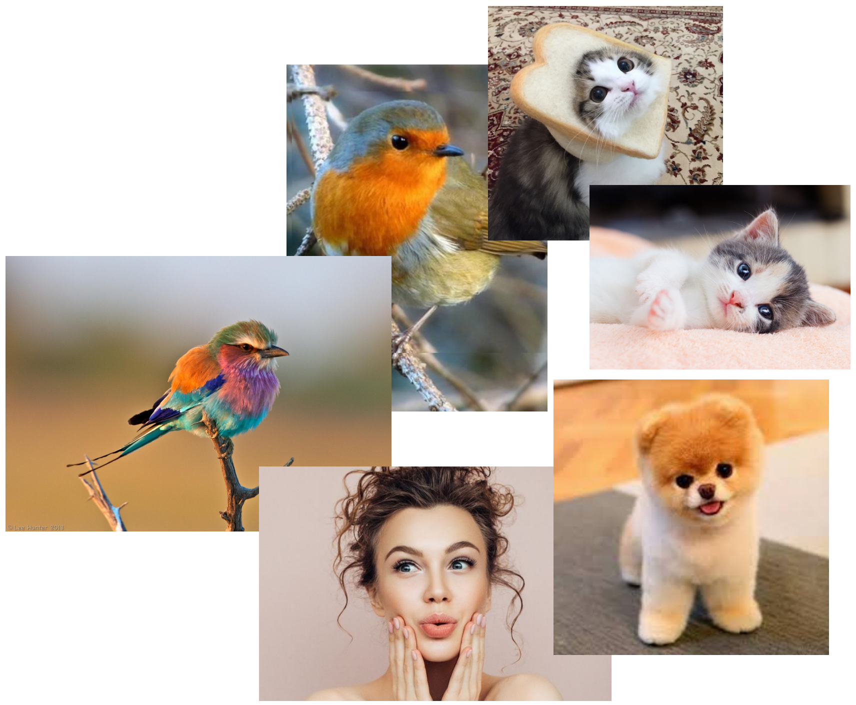
Generative Models — Objective

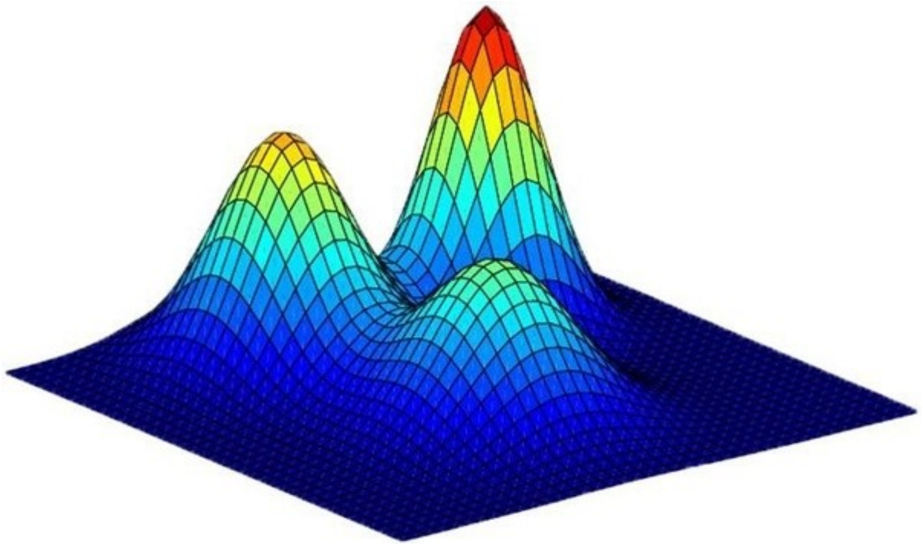


Given Dataset

$$\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{n}$$

Generative Models — Objective





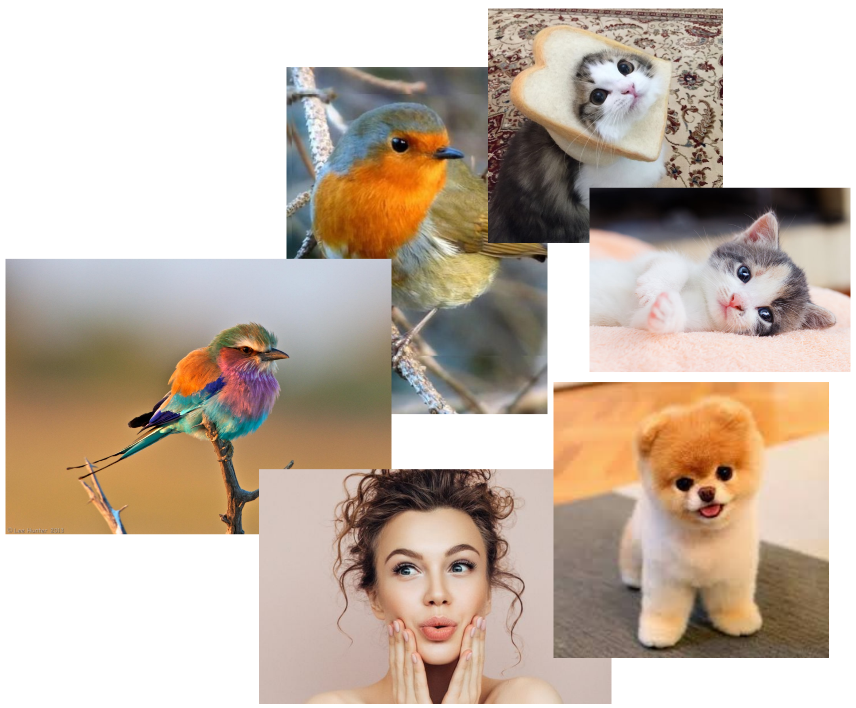
Given Dataset

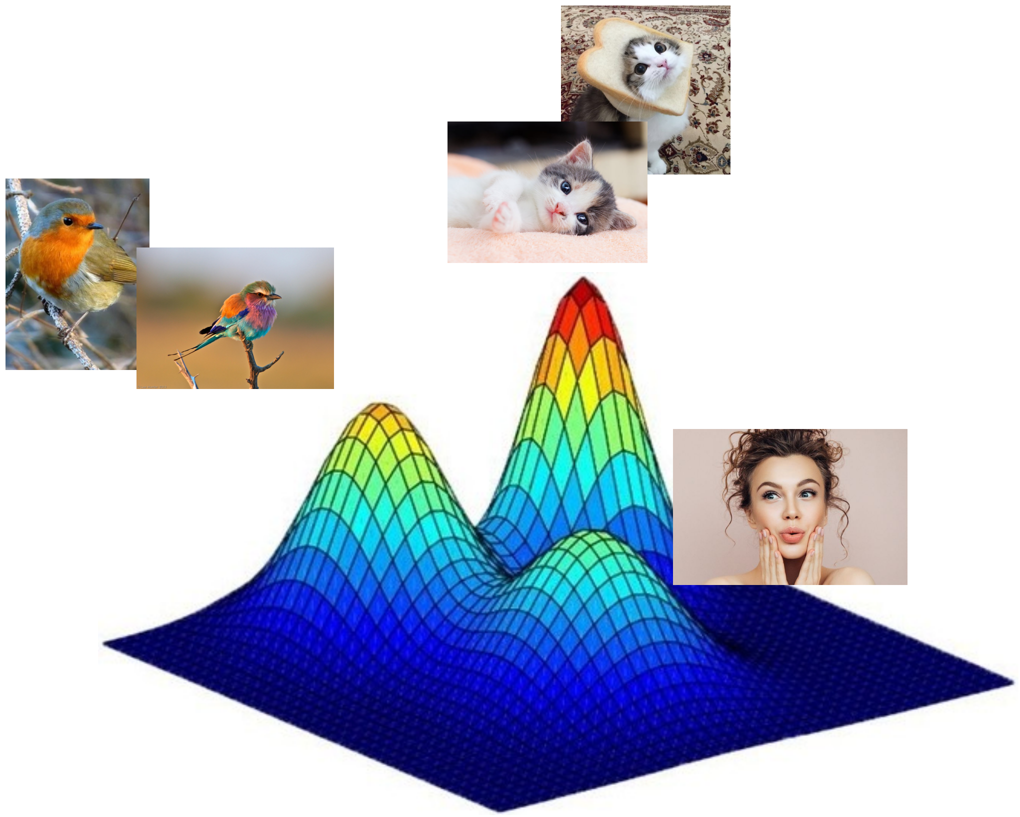
$$\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^n$$

Learn

$$p_\theta(\mathbf{x})$$

Generative Models — Objective





Given Dataset

$$\mathcal{D} = \{\mathbf{x}_i\}_{i=1}^{n}$$

Learn

$$p_\theta(\mathbf{x})$$

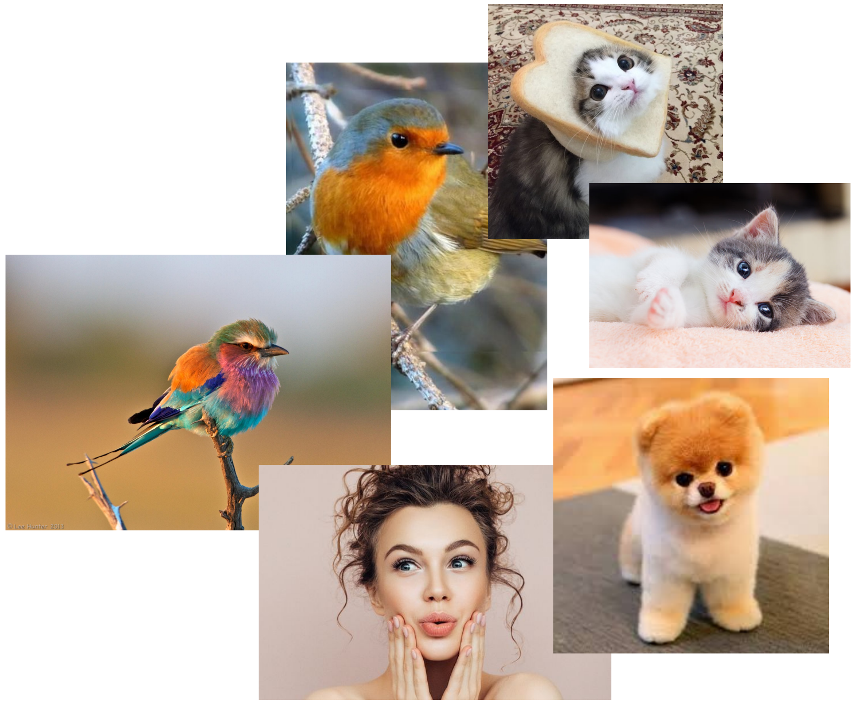
Generative Models — Why?

Generative Models — Why?

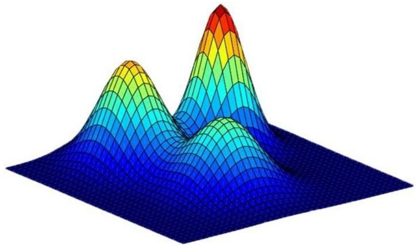
- **Create more data** — Sample from the learned distribution

Generative Models — Why?

- **Create more data** — Sample from the learned distribution



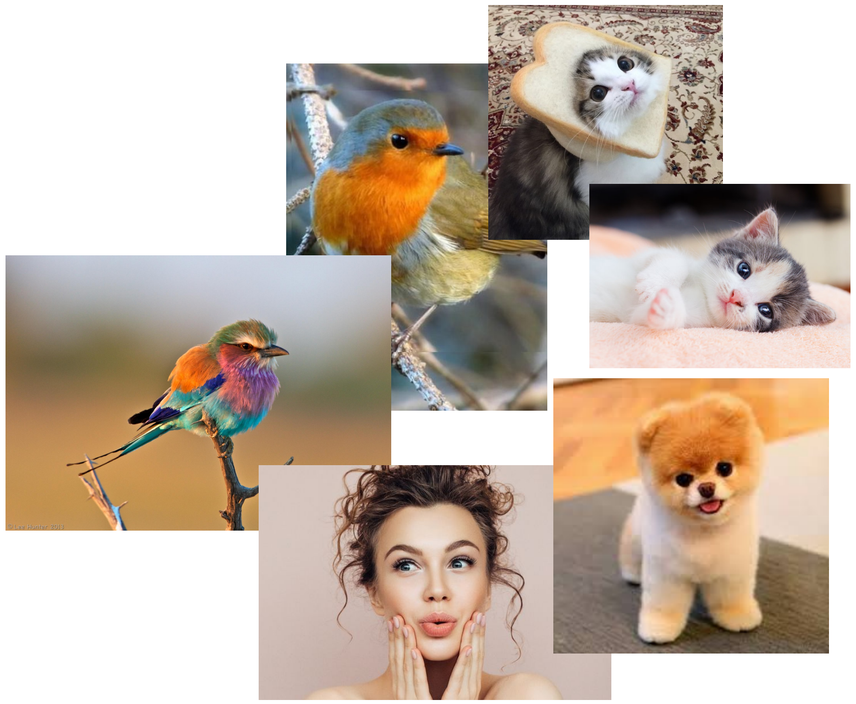
$$\mathbf{x} \sim p_\theta(\mathbf{x})$$



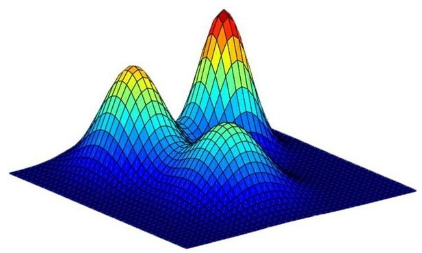
Given Dataset

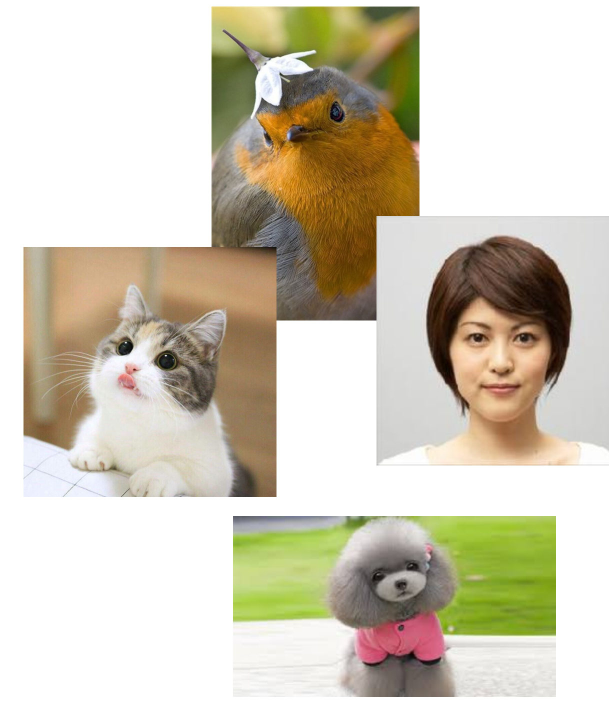
Generative Models — Why?

- **Create more data** — Sample from the learned distribution



$$\mathbf{x} \sim p_\theta(\mathbf{x})$$



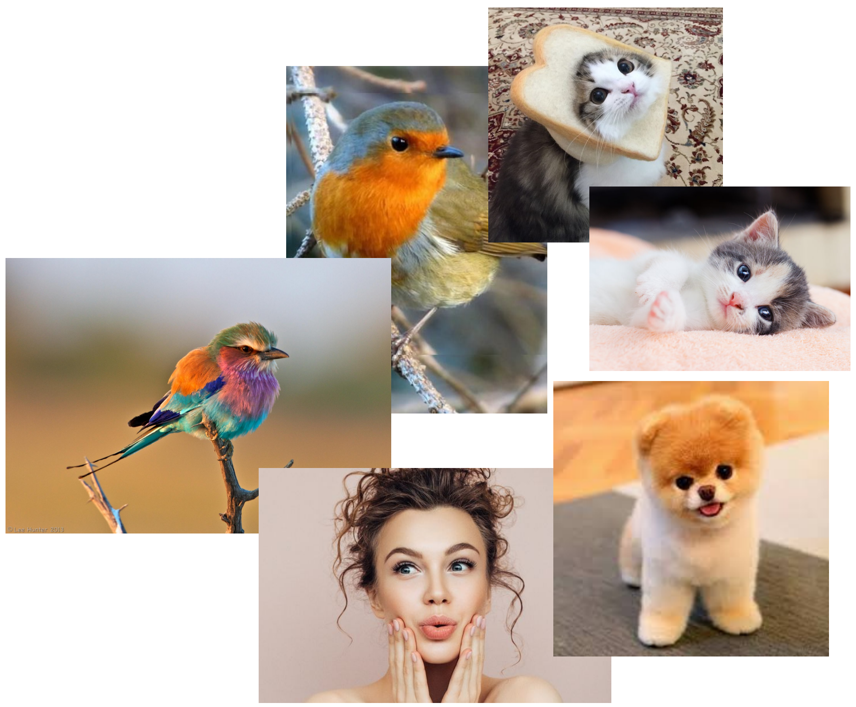


Given Dataset

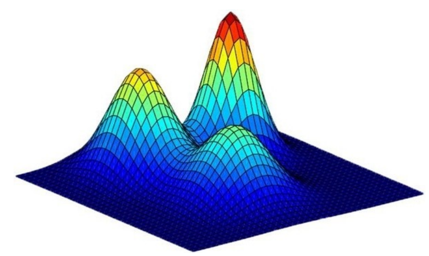
New Data

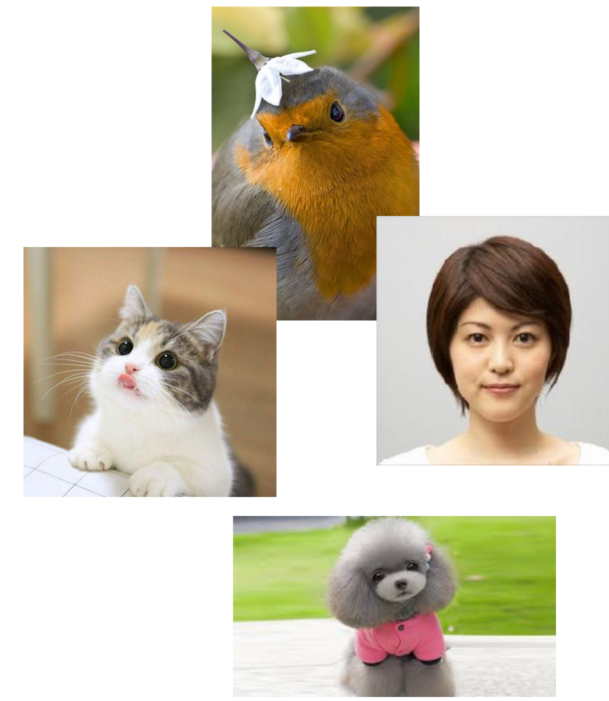
Generative Models — Why?

- **Create more data** — Sample from the learned distribution



$$\mathbf{x} \sim p_\theta(\mathbf{x})$$





Given Dataset **New Data**

- **Density and Uncertainty Estimation**

Generative Models — Why?

Generative Models — Why?

- **Many right answers** — Sample from the conditional distribution

Generative Models — Why?

- **Many right answers** — Sample from the conditional distribution

Sketch to Real Images

$$\mathbf{x} \sim p_\theta(\mathbf{x}|y = sketch)$$

Generative Models — Why?

- **Many right answers** — Sample from the conditional distribution

Sketch to Real Images

$$\mathbf{x} \sim p_\theta(\mathbf{x}|y = sketch)$$

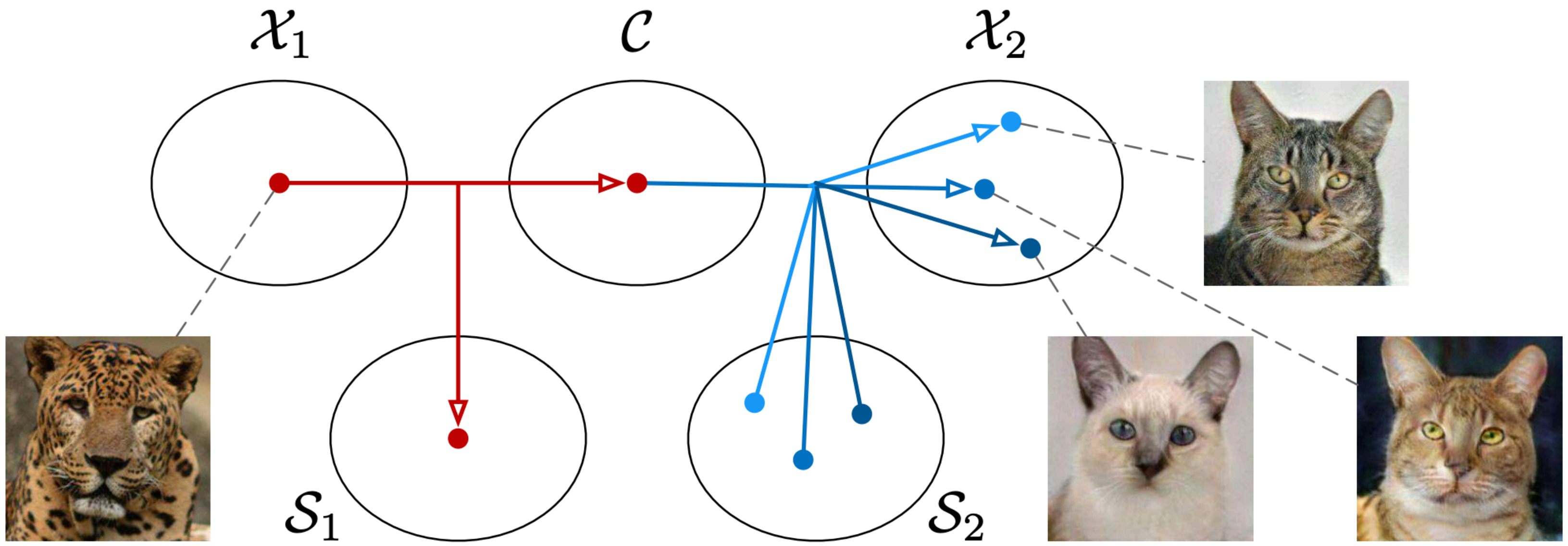
Visual Q&A

Q: How old is the girl
A1: about 7 or 8
A2: looks young

$$\mathbf{x} \sim p_\theta(\mathbf{x}|y = I,\ Q)$$

Generative Models — Why?

- **Learning intrinsic properties of the data**
  - composition, style etc.



Generative Models — Why?

Many more

Generative Model — Basics

- **Use Bayes' theorem with some standard approximations**

Generative Model — Basics

- **Use Bayes' theorem with some standard approximations**

$$p_\theta(\mathbf{x})$$

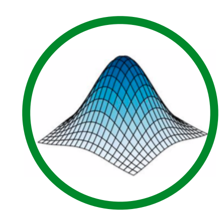
High Dim Complex Space

Generative Model — Basics

- **Use Bayes' theorem with some standard approximations**

$p_\theta(\mathbf{x})$

High Dim
Complex Space



$p(z)$

Low Dim
Interpretable Space

- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

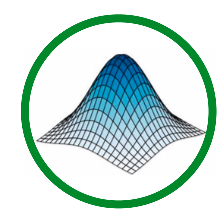
Generative Model — Basics

- **Use Bayes' theorem with some standard approximations**

$p_\theta(\mathbf{x}, z)$

$p_\theta(\mathbf{x})$

High Dim
Complex Space



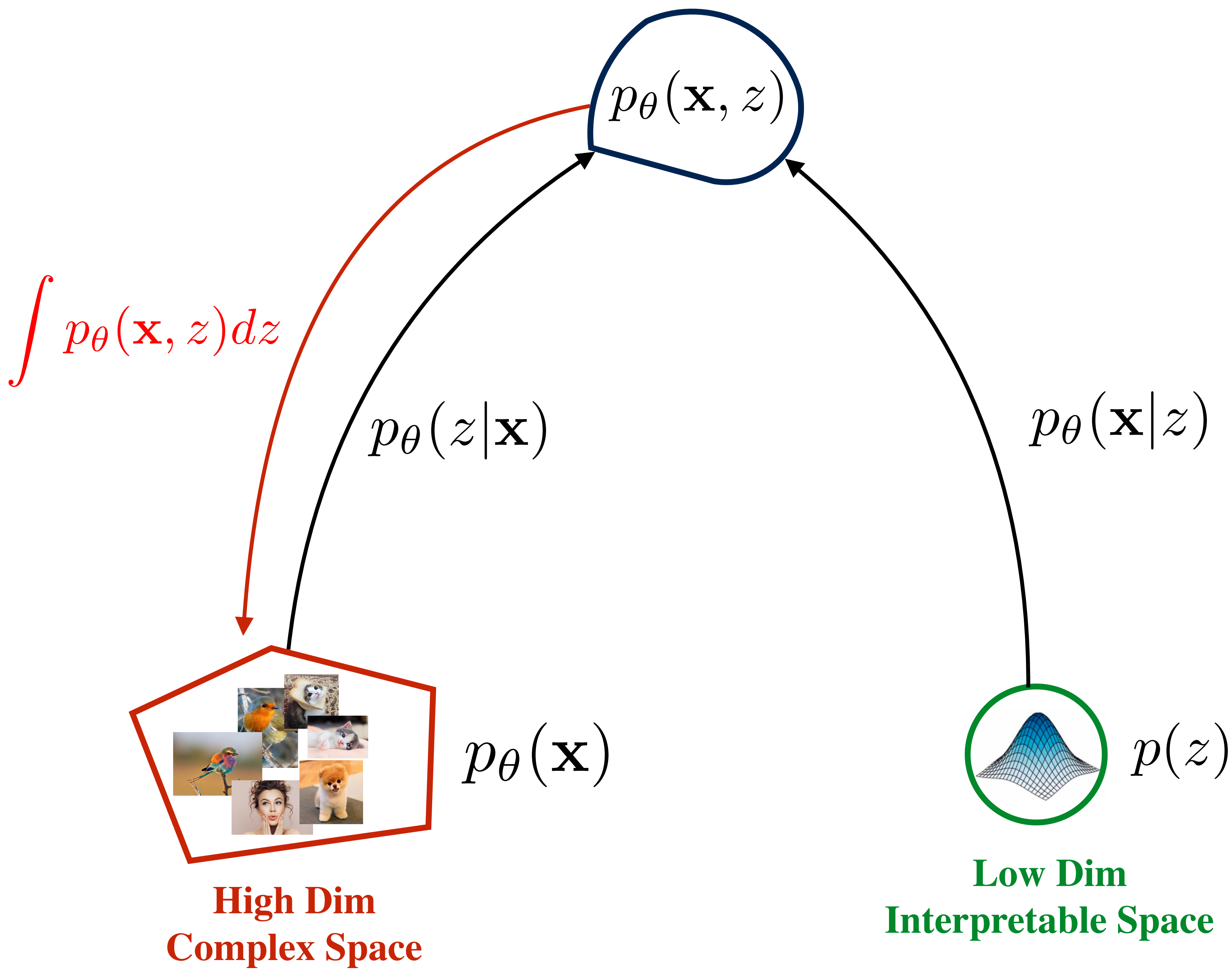
$p(z)$

Low Dim
Interpretable Space

- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Basics

- **Use Bayes' theorem with some standard approximations**

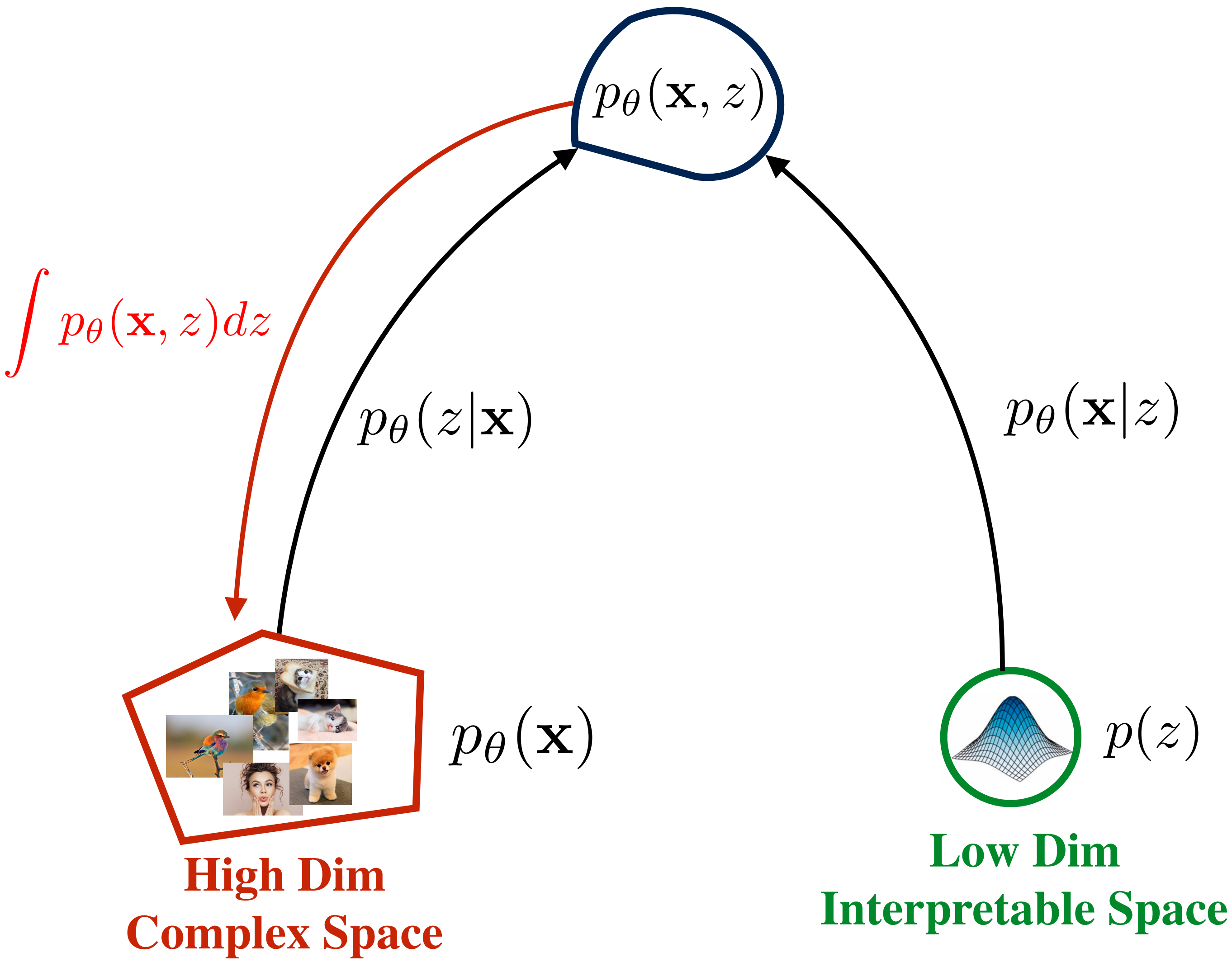


- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



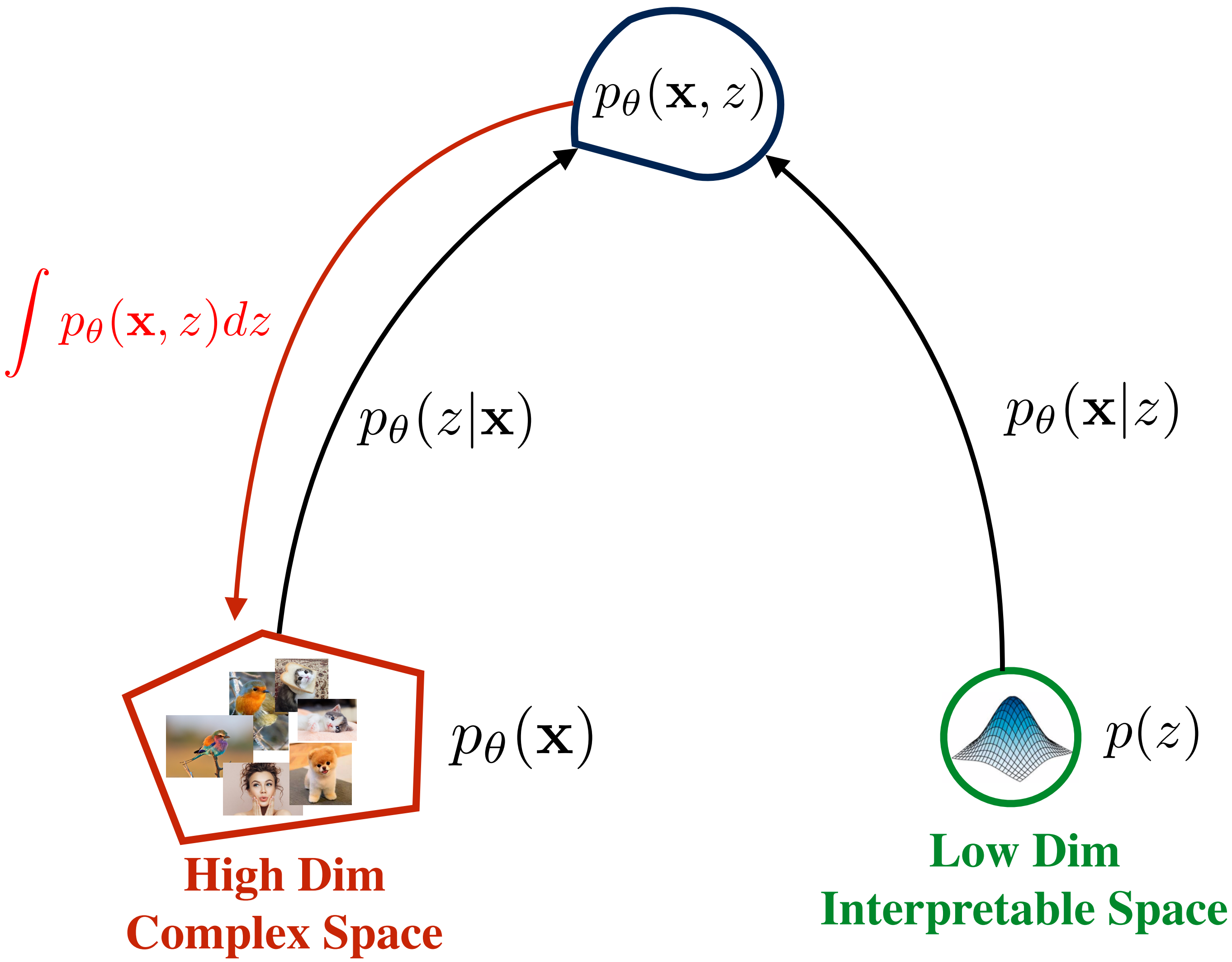
- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

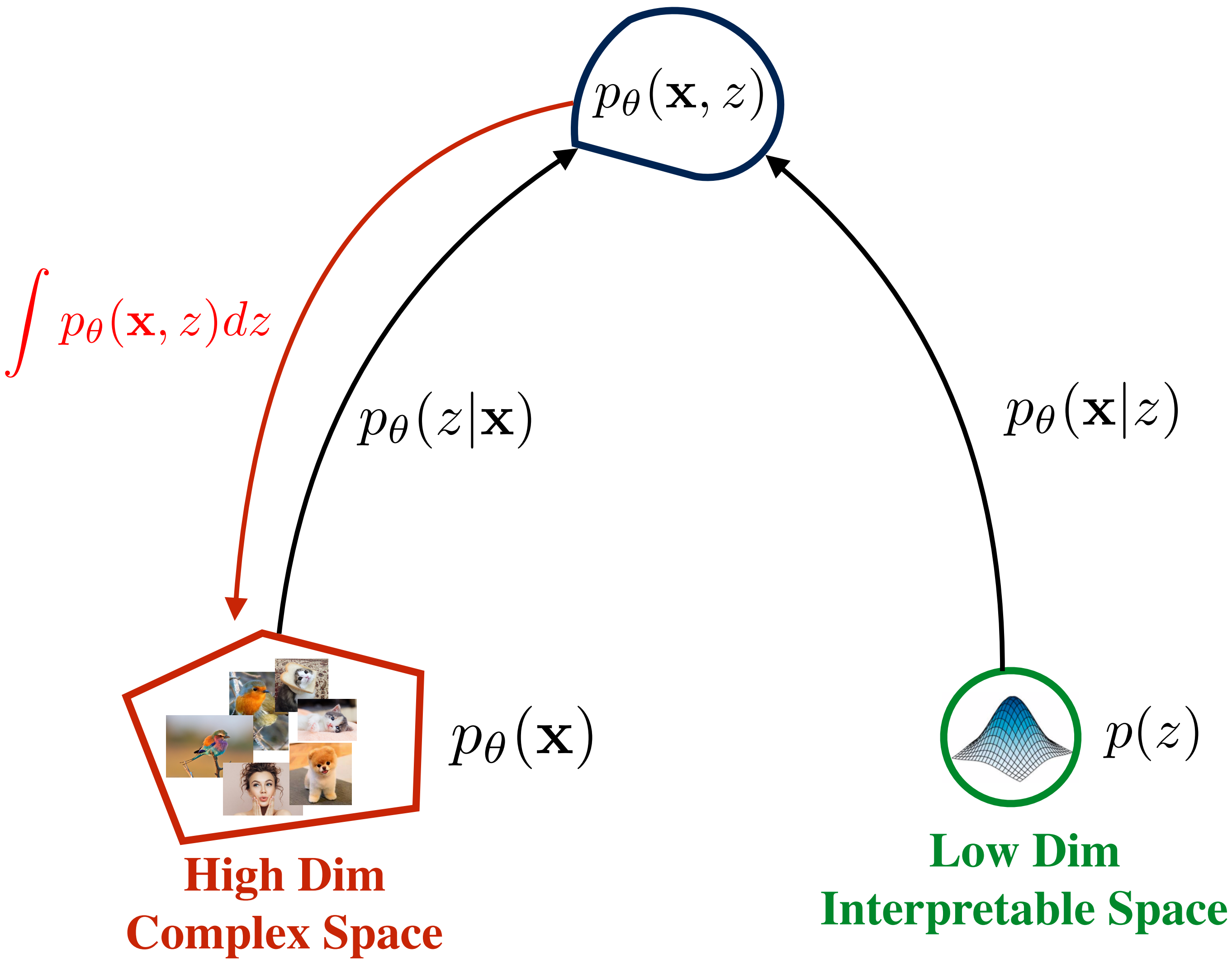
Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**

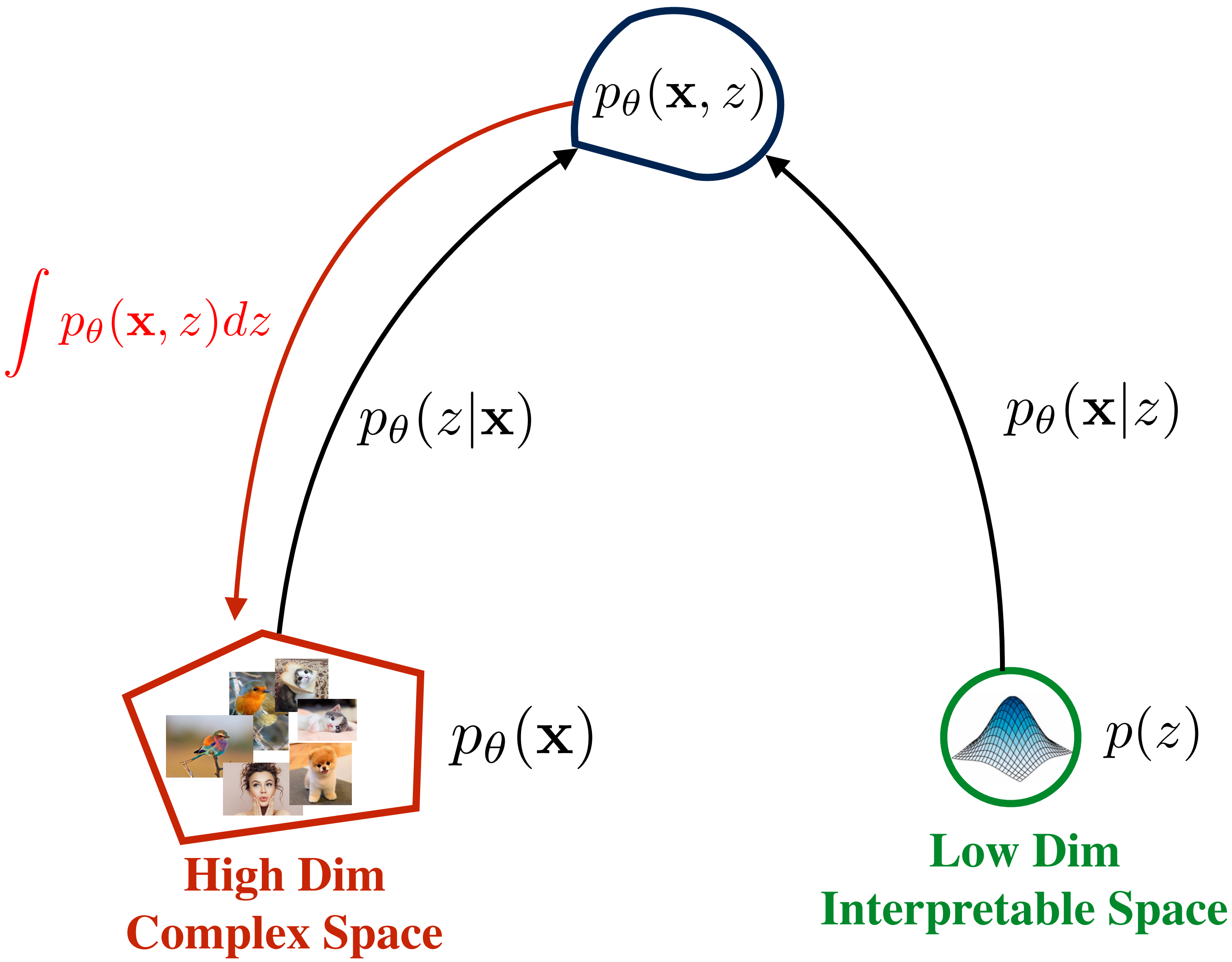
Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**

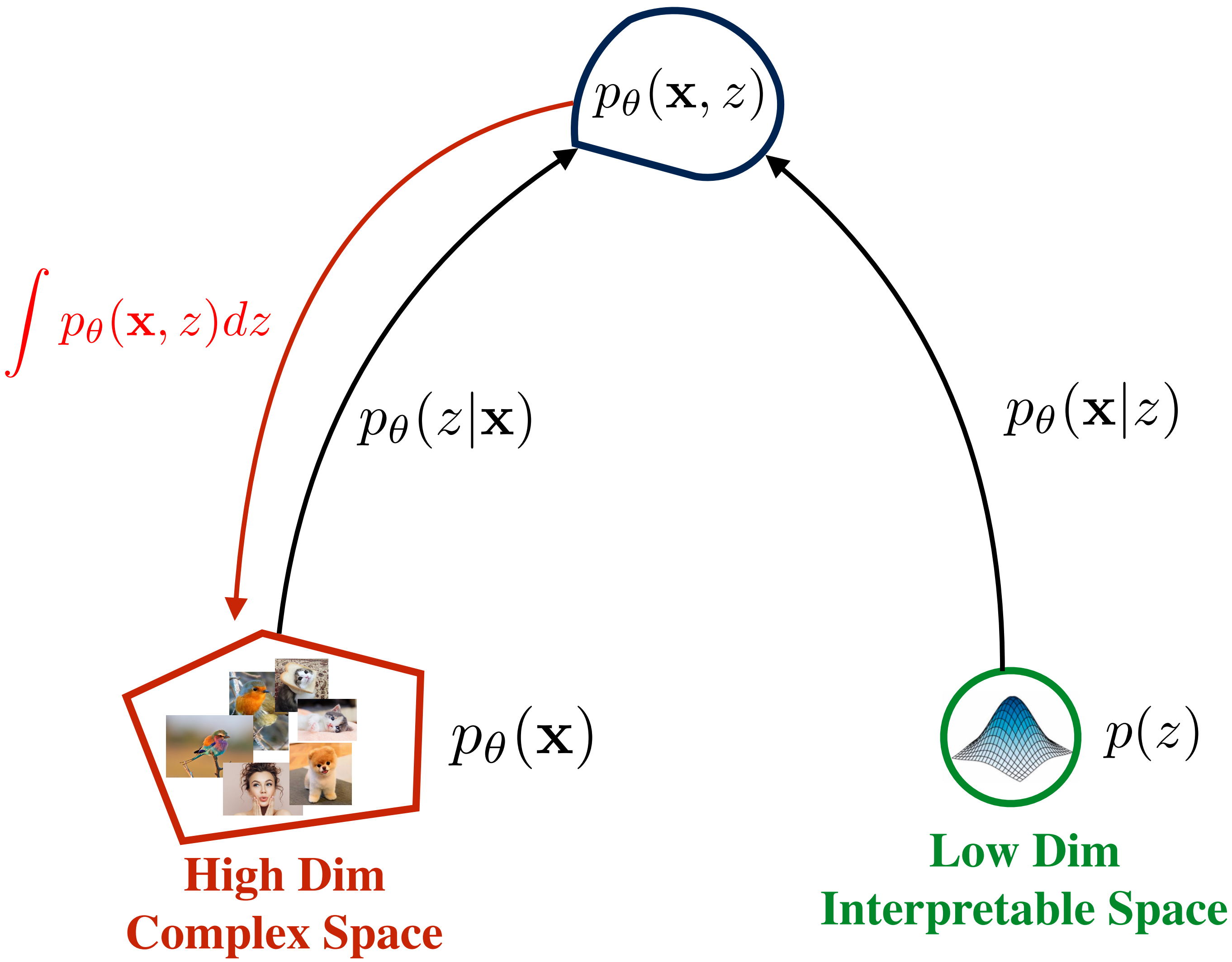
Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

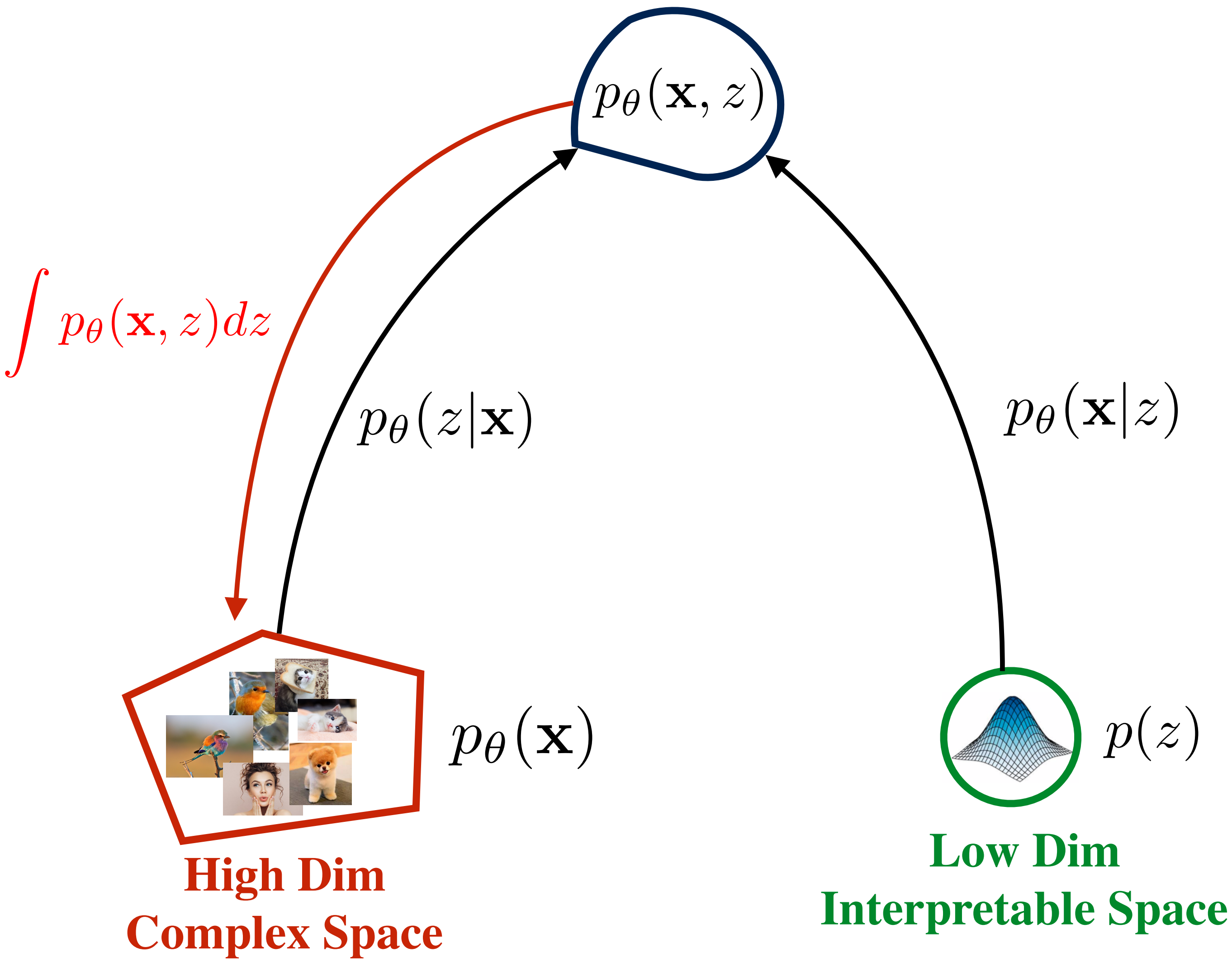
$$p(z|\mathbf{x}) = \frac{p(\mathbf{x}|z)p(z)}{p(\mathbf{x})}$$



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

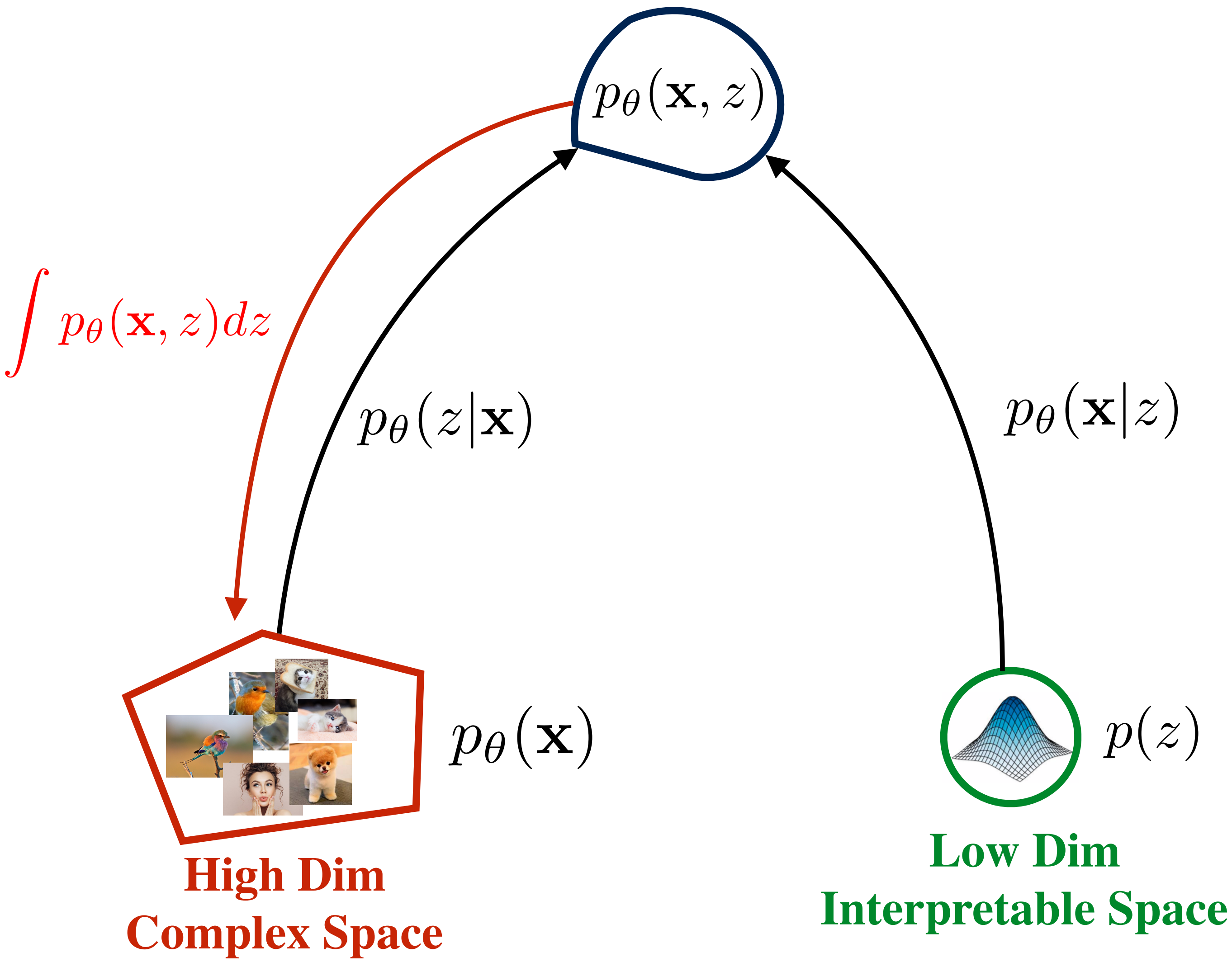
$$p(z|\mathbf{x}) = \frac{p(\mathbf{x}|z)p(z)}{p(\mathbf{x})}$$

Approximations

- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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

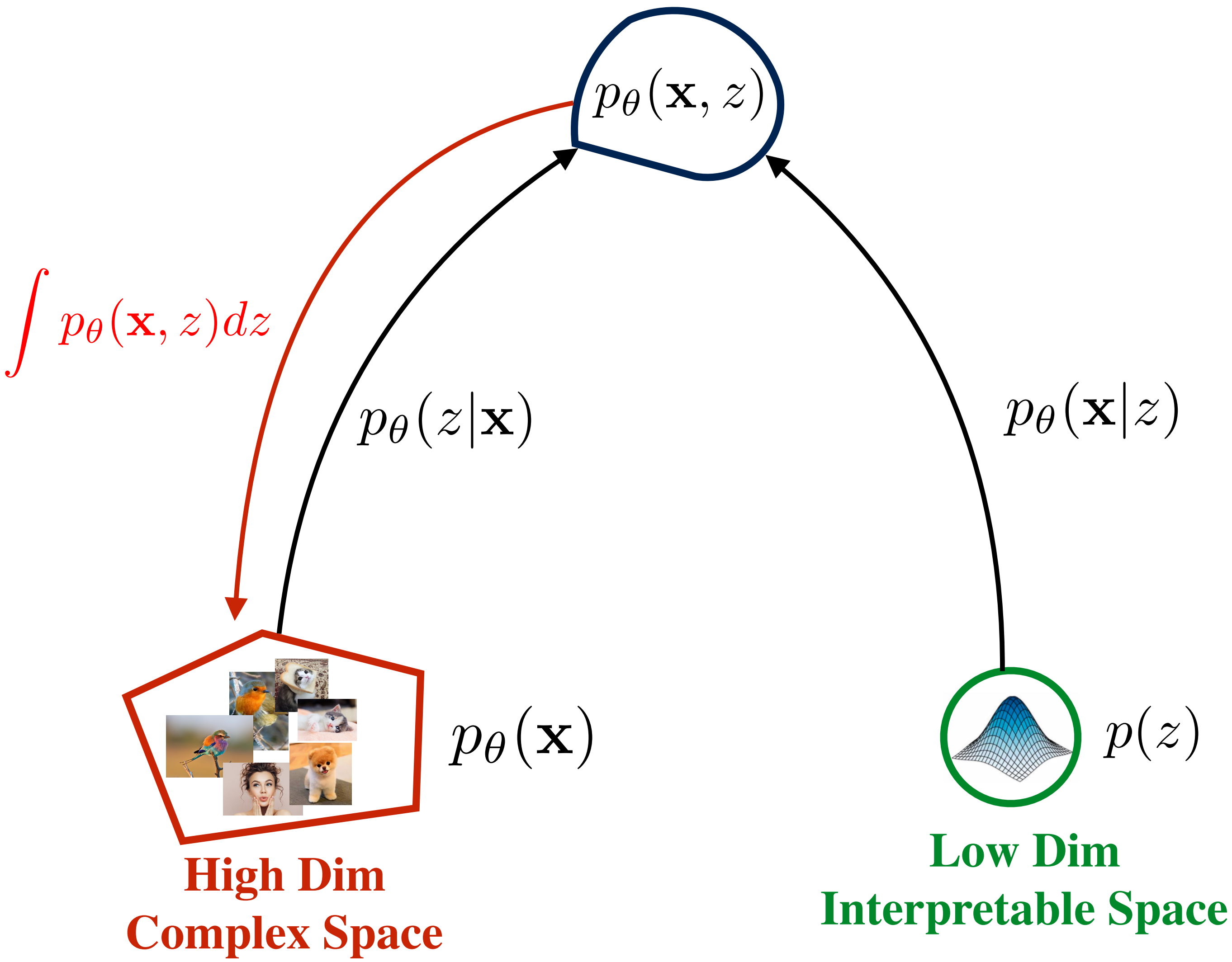
Approximations

MCMC

$$p(\mathbf{x}) \approx \frac{1}{M}\sum_{I=1}^{M} p(\mathbf{x}|z_i), z_i \sim p(z)$$

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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

Approximations

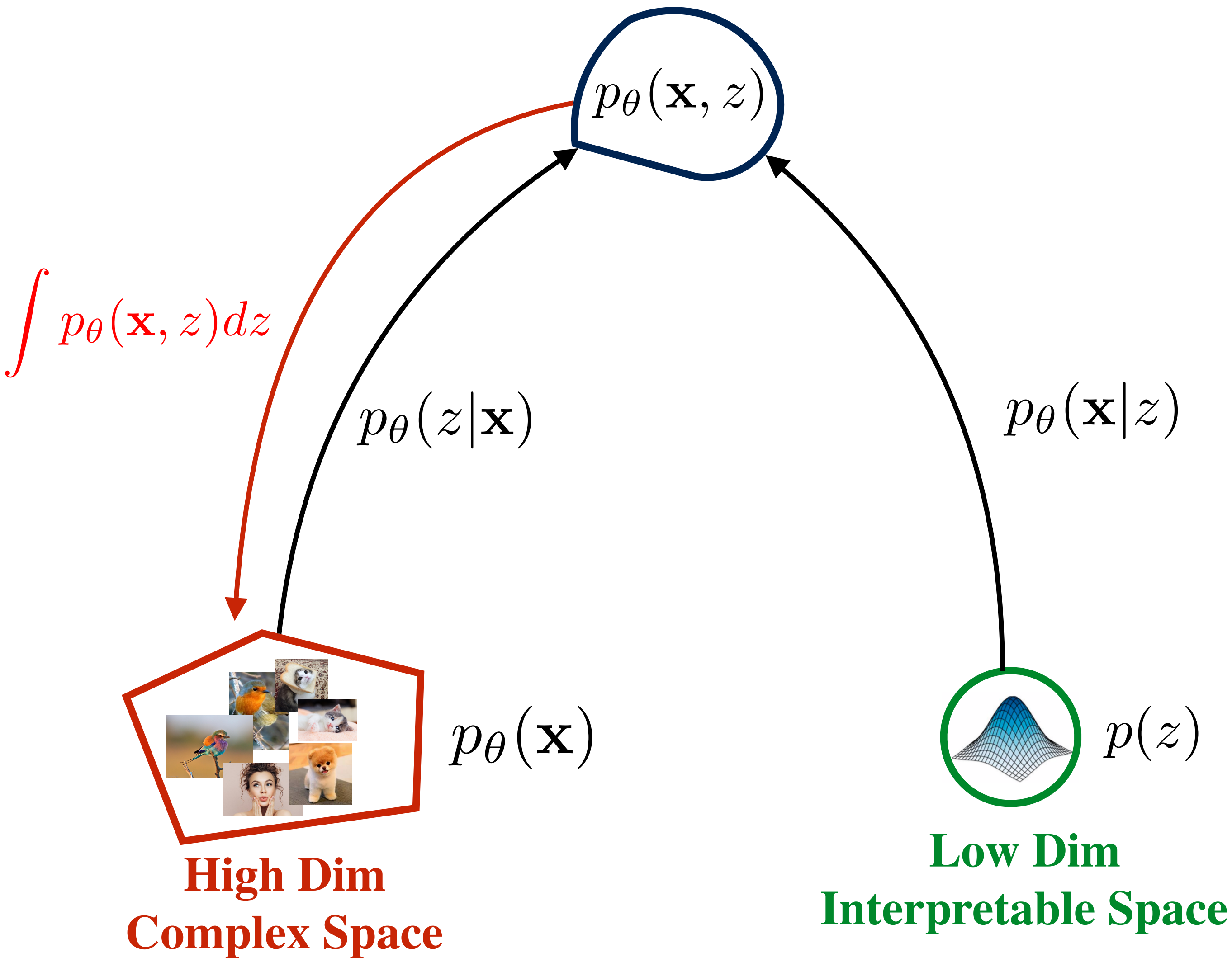
MCMC

$$p(\mathbf{x}) \approx \frac{1}{\boxed{M}} \sum_{I=1}^{M} p(\mathbf{x}|z_i),\, z_i \sim p(z)$$

Millions

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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

Approximations

MCMC

$$p(\mathbf{x}) \approx \frac{1}{M}\sum_{I=1}^{M} p(\mathbf{x}|z_i), z_i \sim p(z)$$

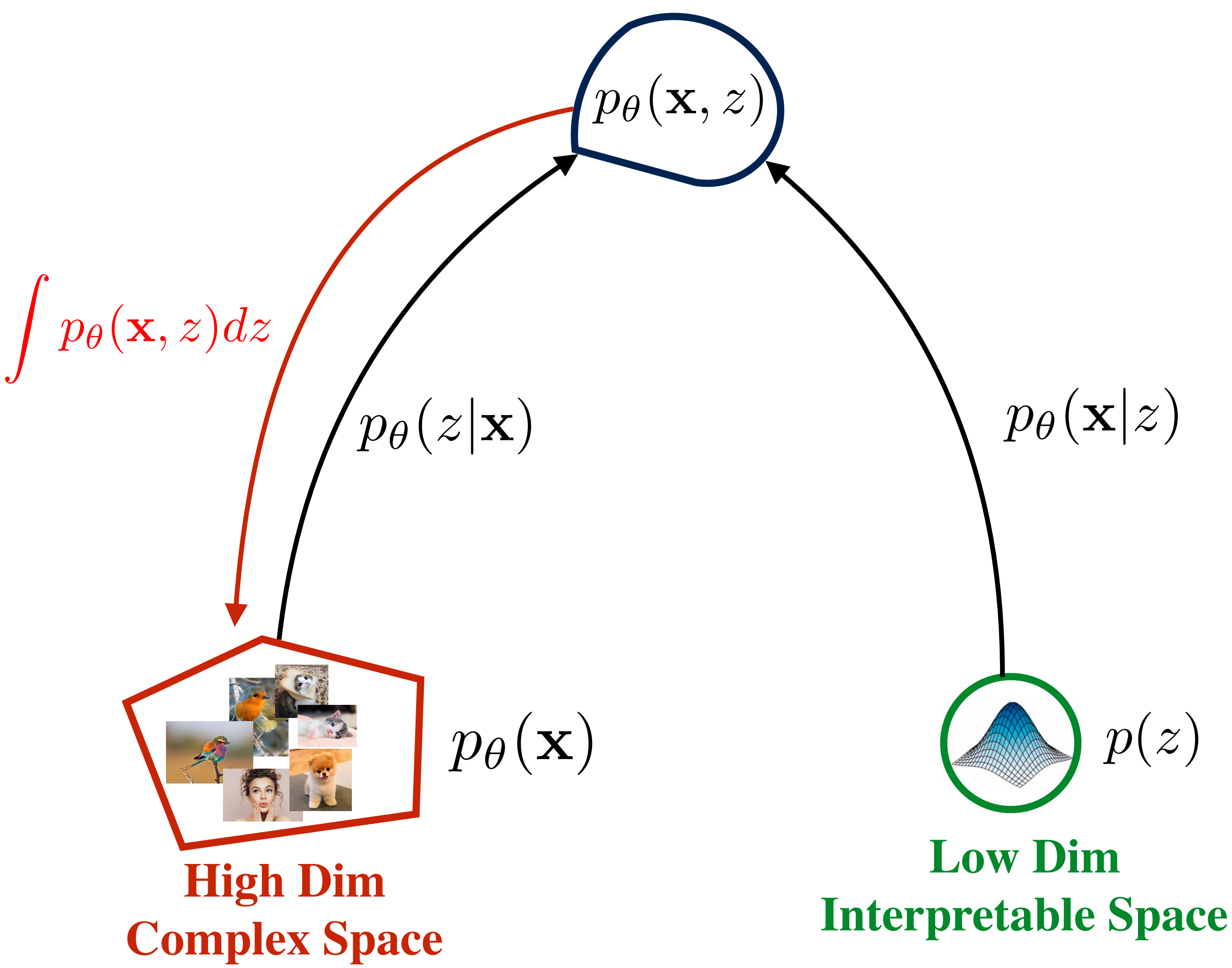
Millions

Importance Sampling

$$p_\theta(\mathbf{x}) = \mathbb{E}_{p(z)} p_\theta(\mathbf{x}|z) = \mathbb{E}_{q_\phi(z|\mathbf{x})} \frac{p_\theta(\mathbf{x}|z)p(z)}{q_\phi(z|\mathbf{x})}$$

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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

Approximations

MCMC

$$p(\mathbf{x}) \approx \frac{1}{\boxed{M}} \sum_{I=1}^{M} p(\mathbf{x}|z_i),\, z_i \sim p(z)$$

Millions

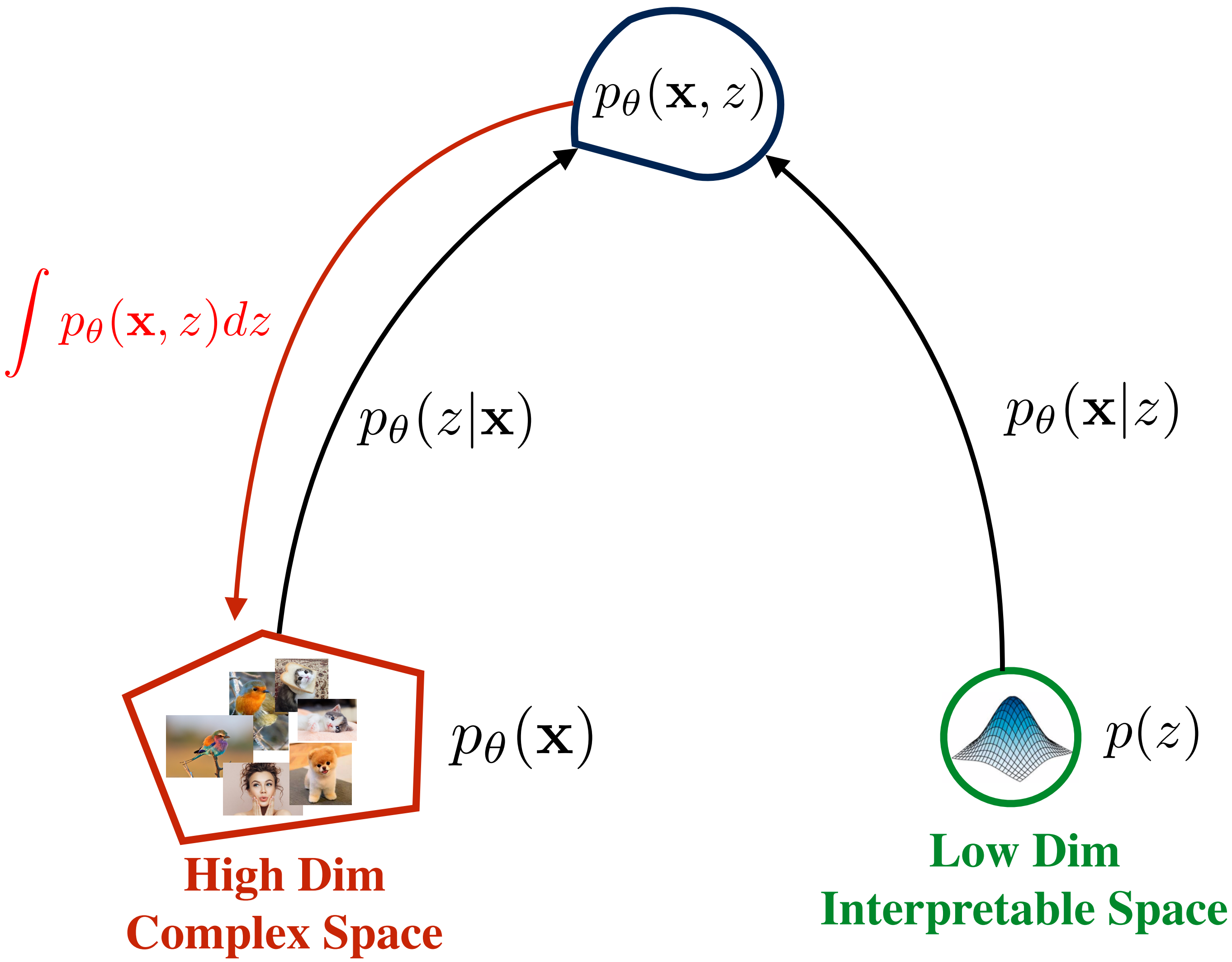
Importance Sampling

$$p_\theta(\mathbf{x}) = \mathbb{E}_{p(z)} p_\theta(\mathbf{x}|z) = \mathbb{E}_{q_\phi(z|\mathbf{x})} \frac{p_\theta(\mathbf{x}|z)p(z)}{q_\phi(z|\mathbf{x})}$$

proposal distribution

Generative Model — Main Problems

- **Use Bayes' theorem with some standard approximations**



- May contain information about
  - Number of objects
  - Type/Appearance of objects
  - Background and lighting condition

Marginal? — Infinite mixture model

$$p(\mathbf{x}) = \int p(\mathbf{x}, z)dz = \int p_\theta(\mathbf{x}|z)p(z)dz$$

Intractable

Posterior?

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

Approximations

MCMC

$$p(\mathbf{x}) \approx \frac{1}{M}\sum_{I=1}^{M} p(\mathbf{x}|z_i),\, z_i \sim p(z)$$

Millions

Importance Sampling

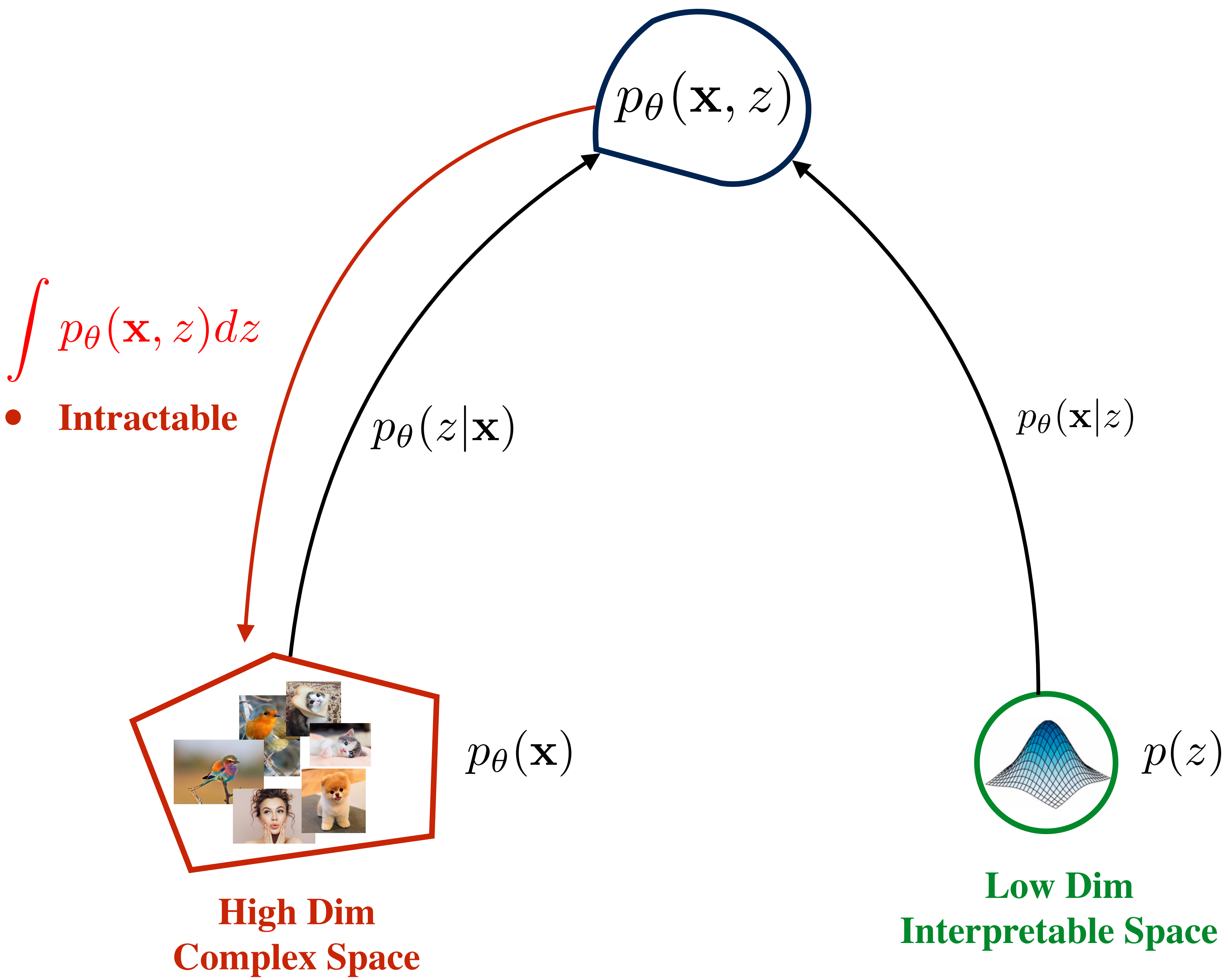
$$p_\theta(\mathbf{x}) = \mathbb{E}_{p(z)} p_\theta(\mathbf{x}|z) = \mathbb{E}_{q_\phi(z|\mathbf{x})} \frac{p_\theta(\mathbf{x}|z)p(z)}{q_\phi(z|\mathbf{x})}$$

proposal distribution

- **Use NN to define relationships**
- **Learn parameters to minimise some notion of distance (KL) — VAE**

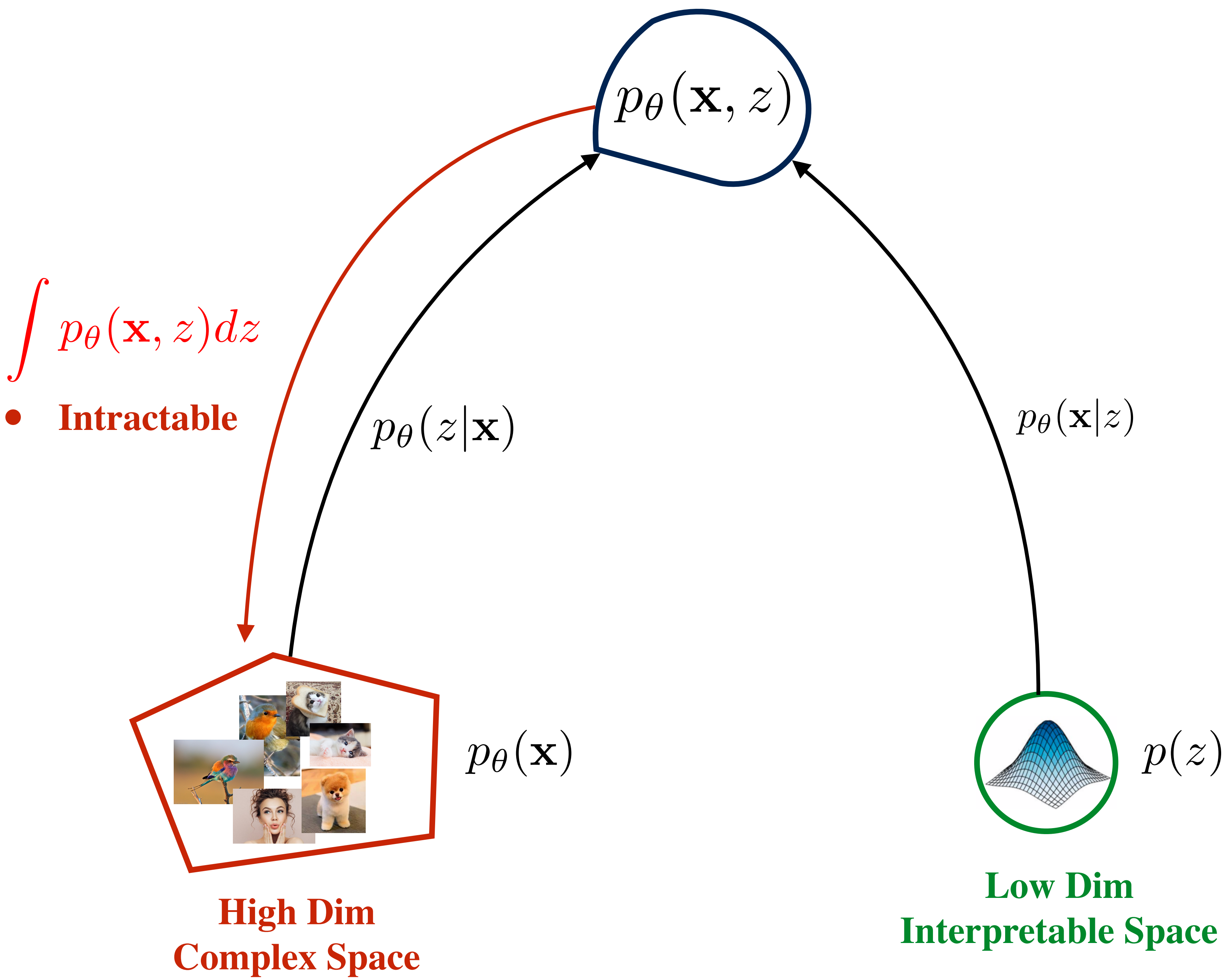
Deep Generative Models — Overview of VAE and GAN

Deep Generative Models — Overview of VAE and GAN



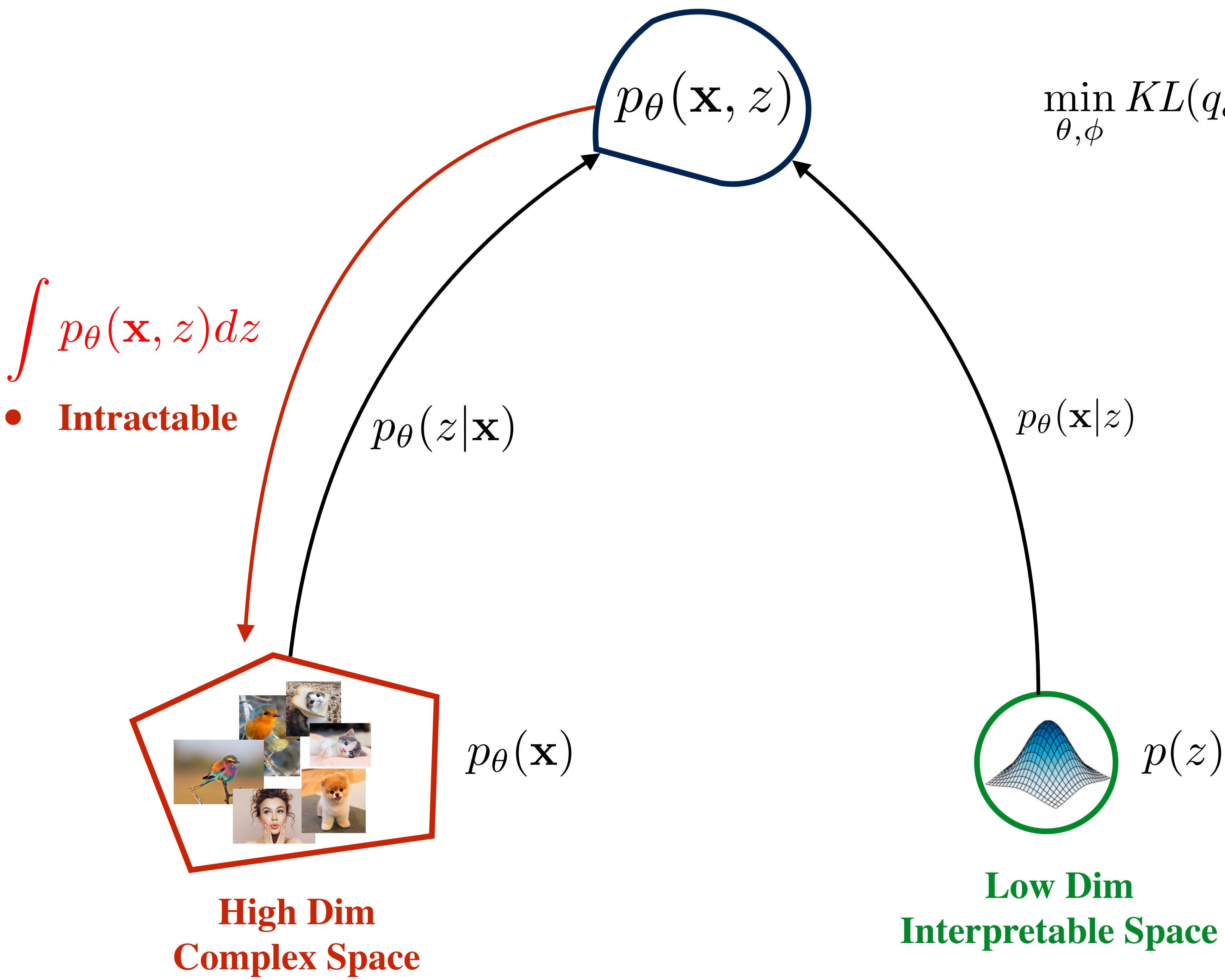
Deep Generative Models — Overview of VAE and GAN

VAEs



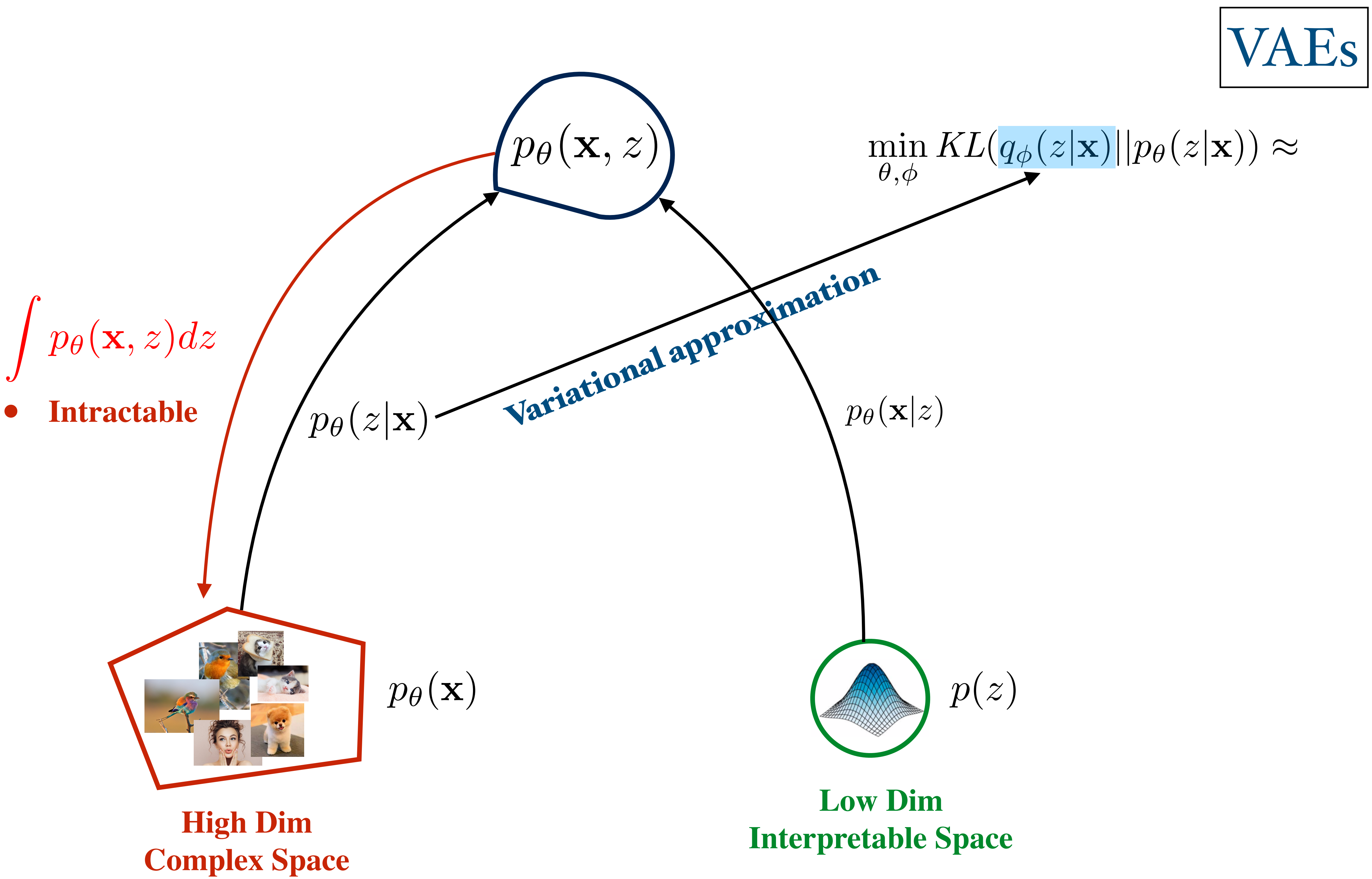
Deep Generative Models — Overview of VAE and GAN

$\boxed{\text{VAEs}}$

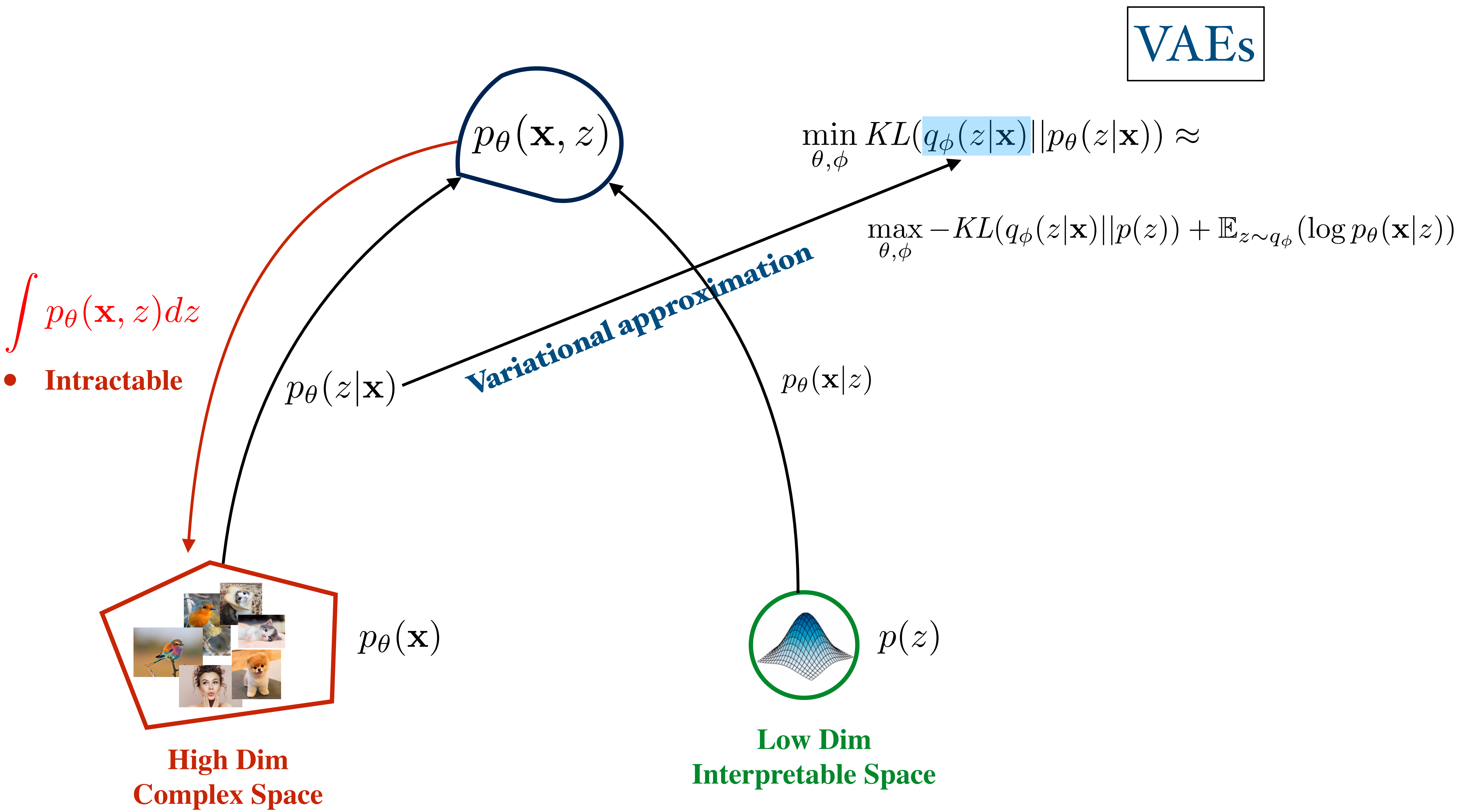


$$\min_{\theta,\phi} KL(q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})) \approx$$

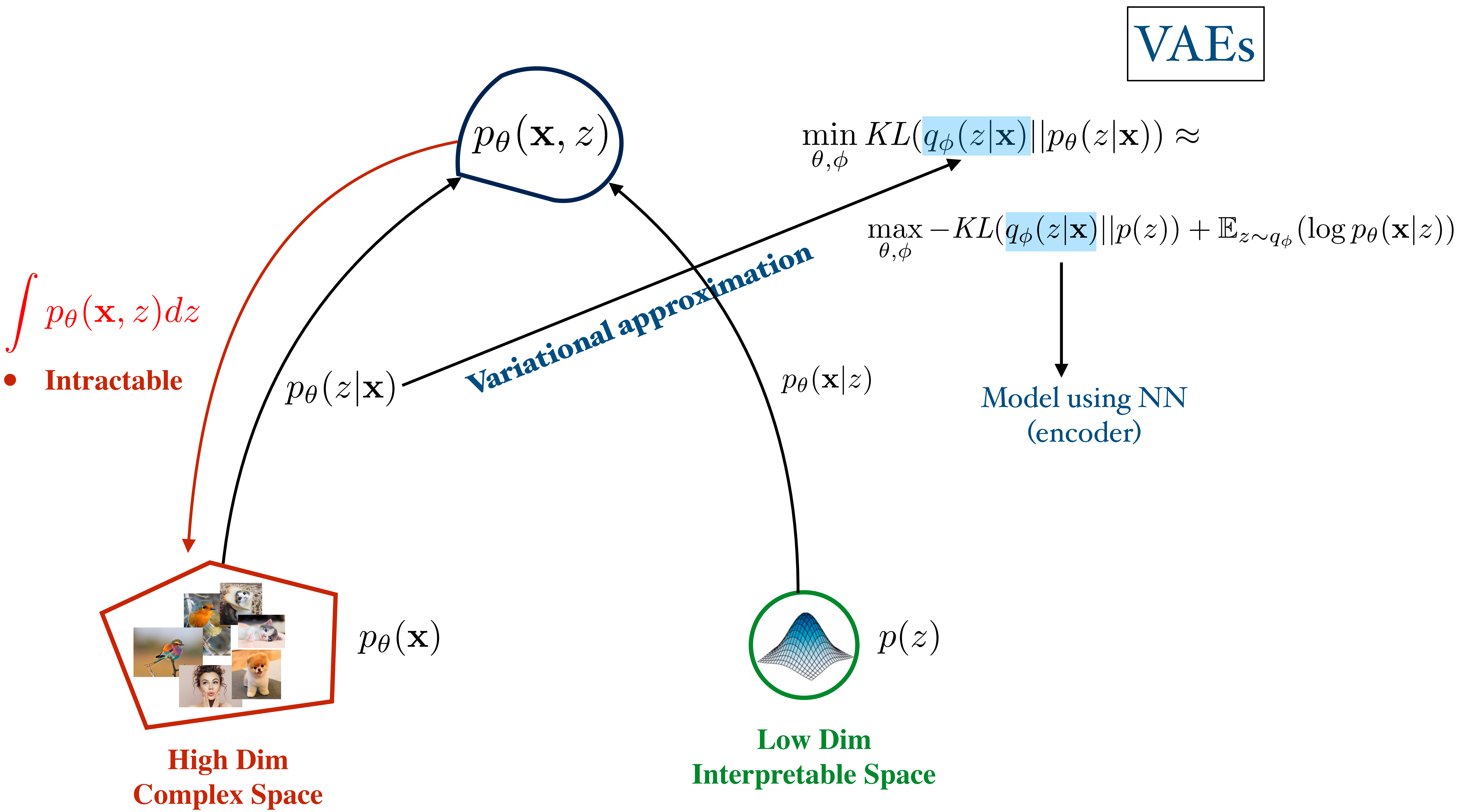
Deep Generative Models — Overview of VAE and GAN



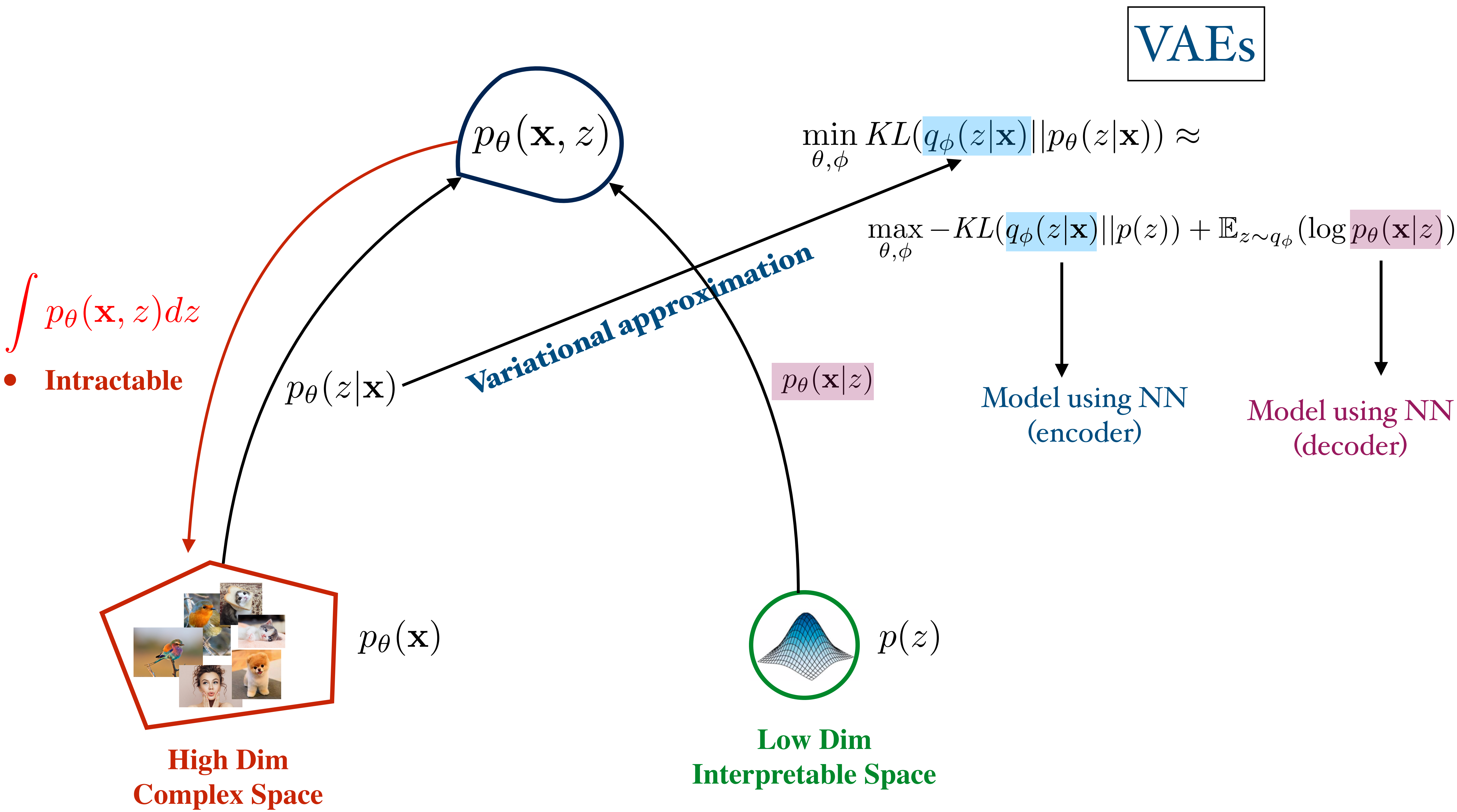
Deep Generative Models — Overview of VAE and GAN



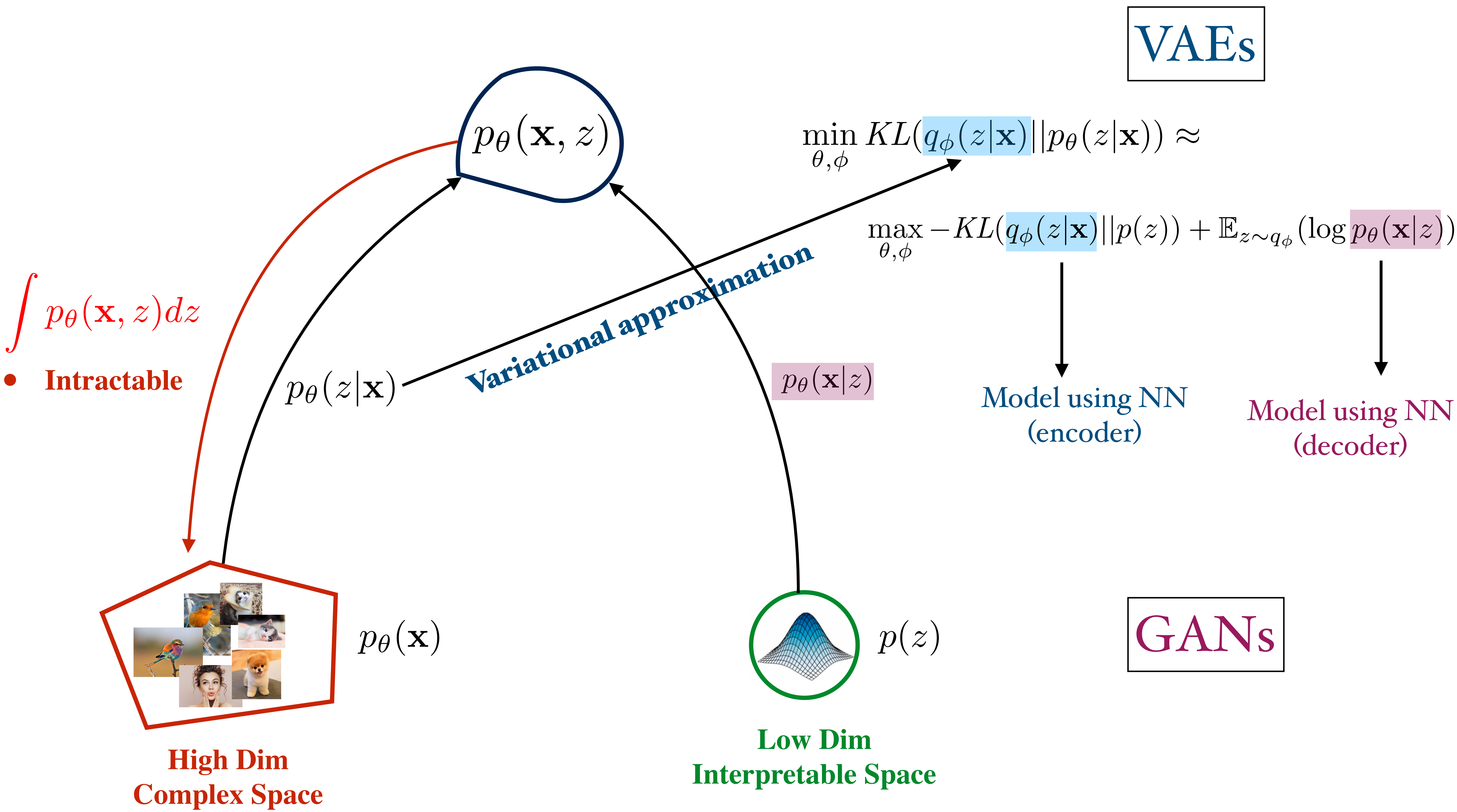
Deep Generative Models — Overview of VAE and GAN



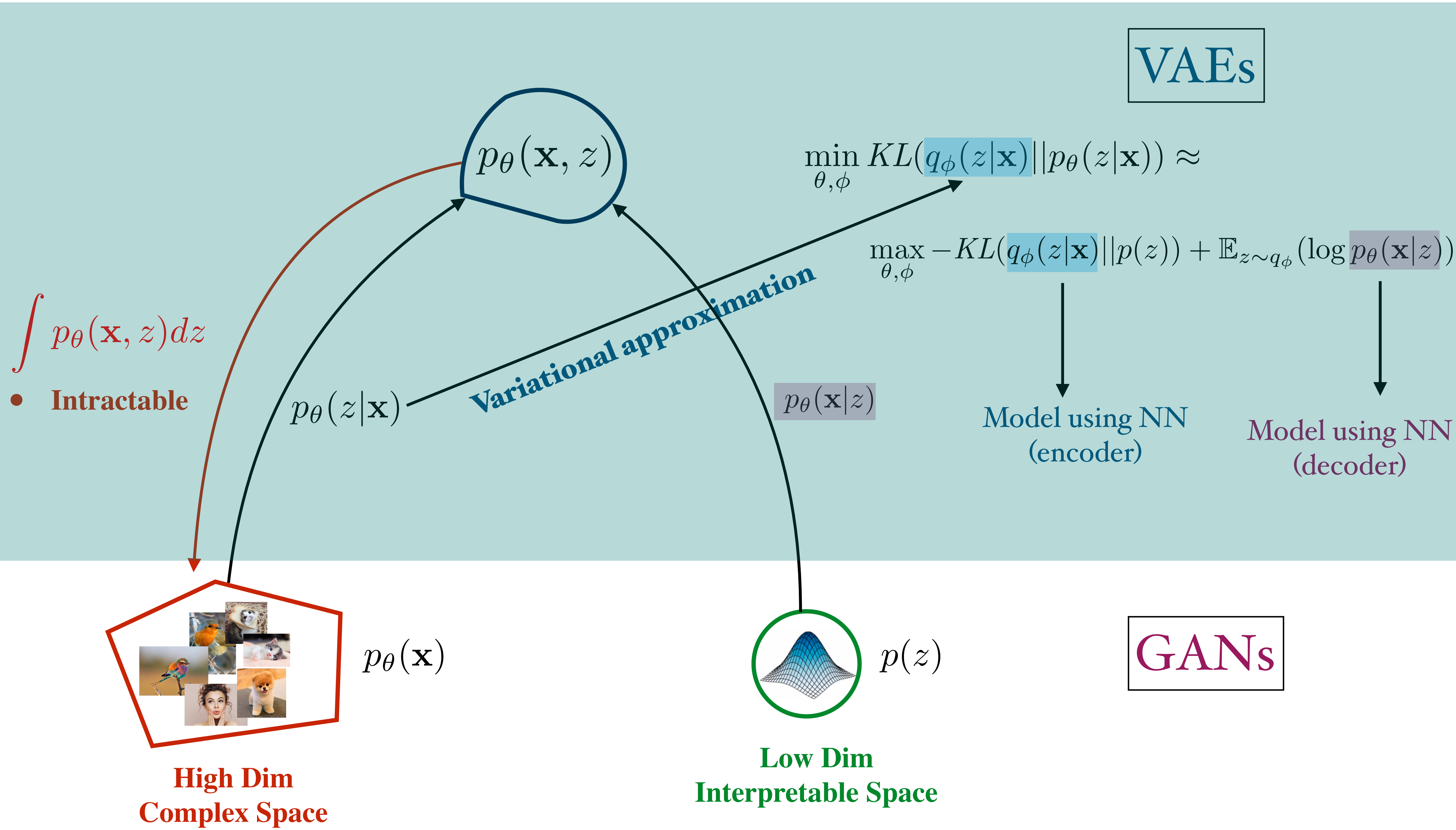
Deep Generative Models — Overview of VAE and GAN



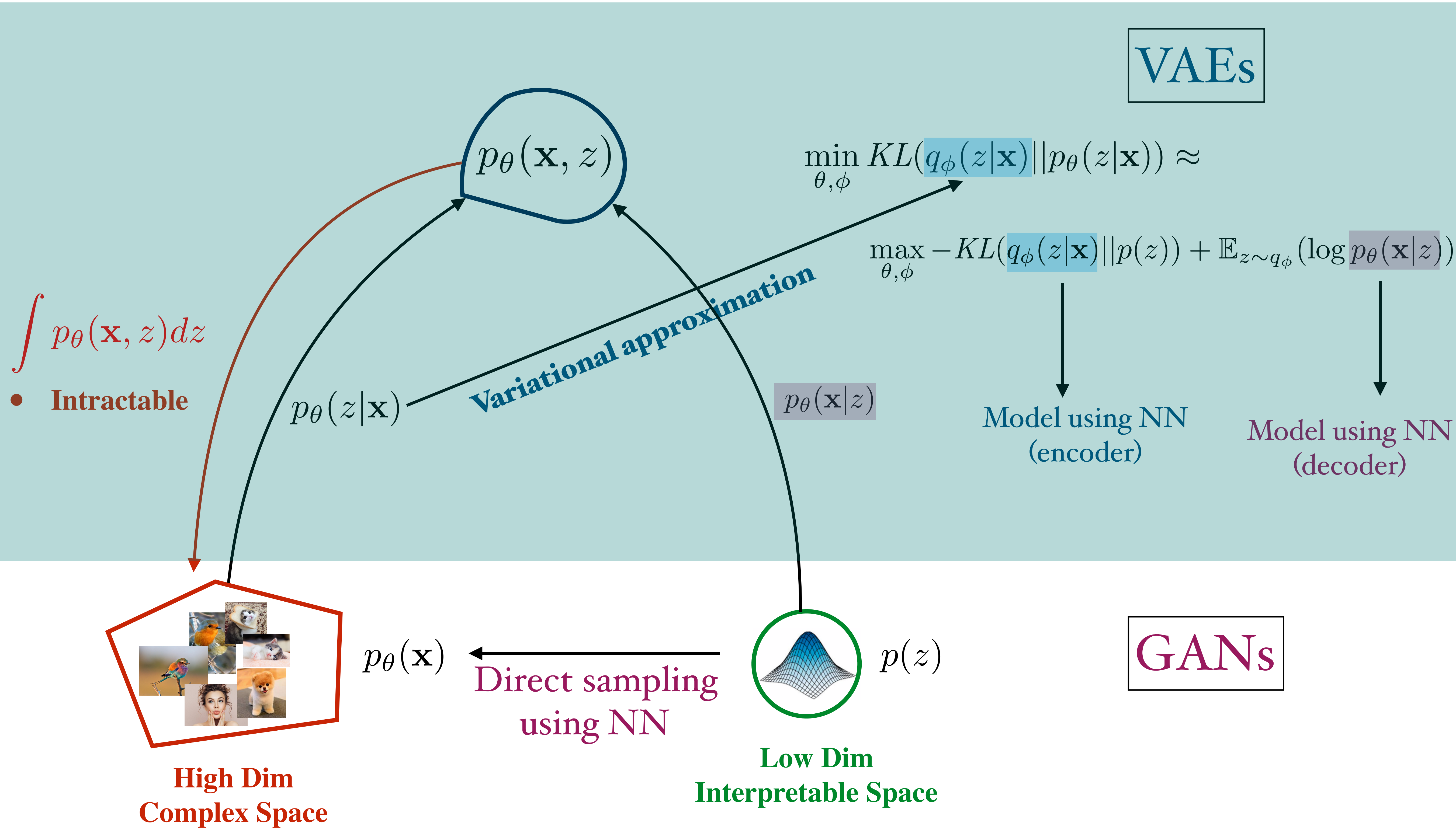
Deep Generative Models — Overview of VAE and GAN



Deep Generative Models — Overview of VAE and GAN



Deep Generative Models — Overview of VAE and GAN

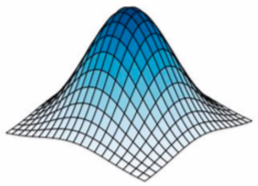


Variational Auto-encoders (VAEs)

(Kingma et al., 2013)

Variational Auto-encoders (VAEs)

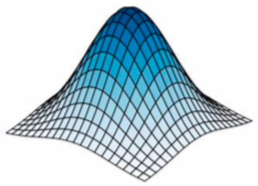
(Kingma et al., 2013)



$p(z) = N(0, 1)$

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



$p(z) = N(0, 1)$

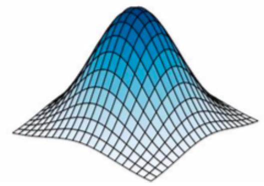
Test

$z \sim p(z)$

(low dim)

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



$p(z) = N(0, 1)$

Test

$z \sim p(z)$

(low dim)

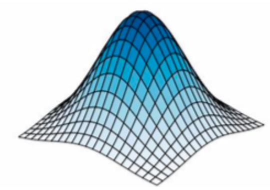
Decoder

NN with parameters

θ

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



$p(z) = N(0, 1)$

Test

$z \sim p(z)$

(low dim)

Decoder

NN with parameters

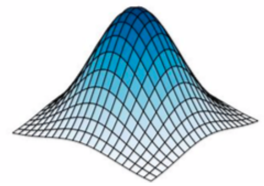
θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



$p(z) = N(0, 1)$

Test

$z \sim p(z)$

(low dim)

Decoder

NN with parameters

θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

(high dim)

$\mathbf{x} \sim \mathcal{D}$

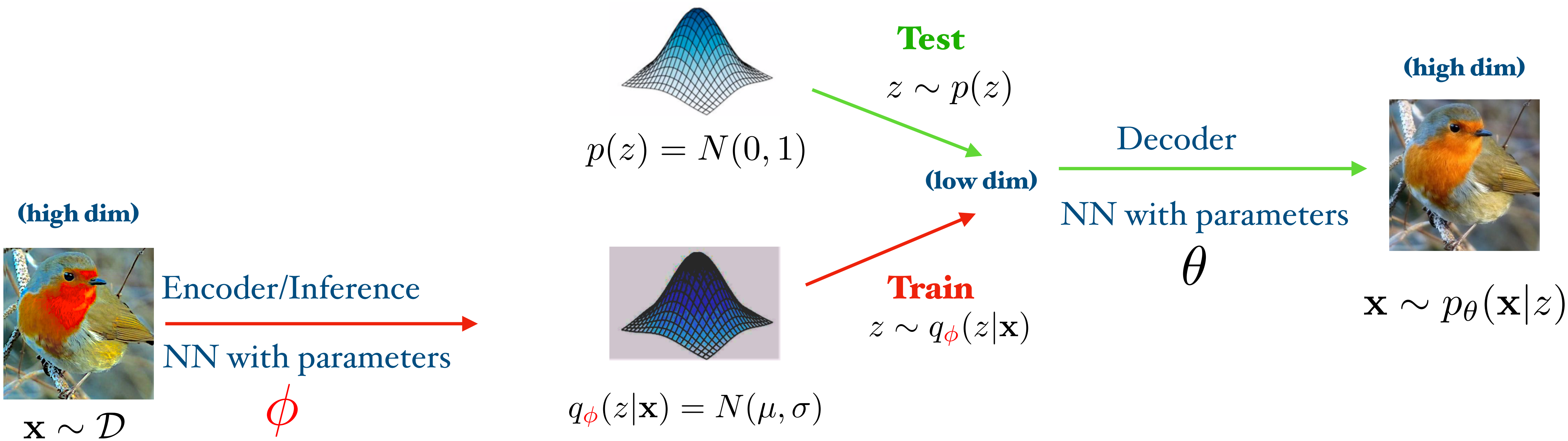
Encoder/Inference

NN with parameters

ϕ

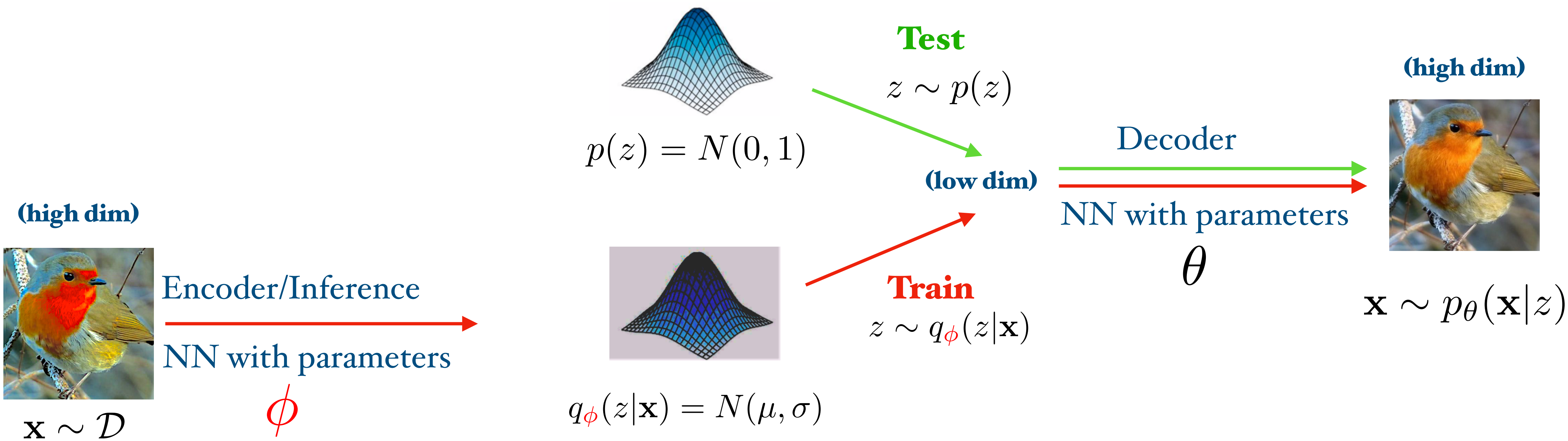
Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



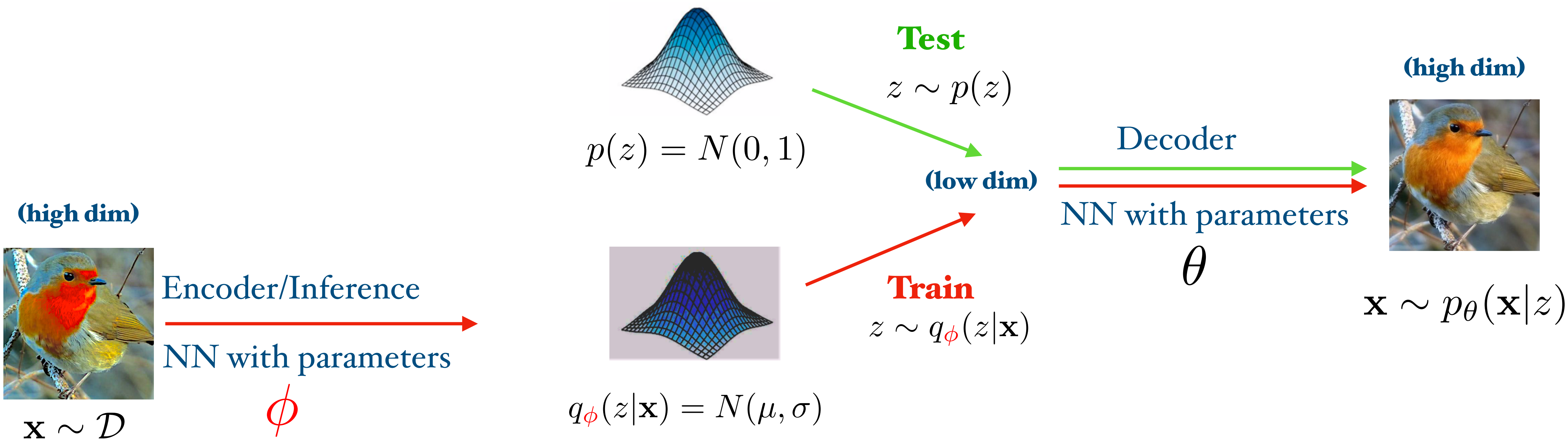
Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



Variational Auto-encoders (VAEs)

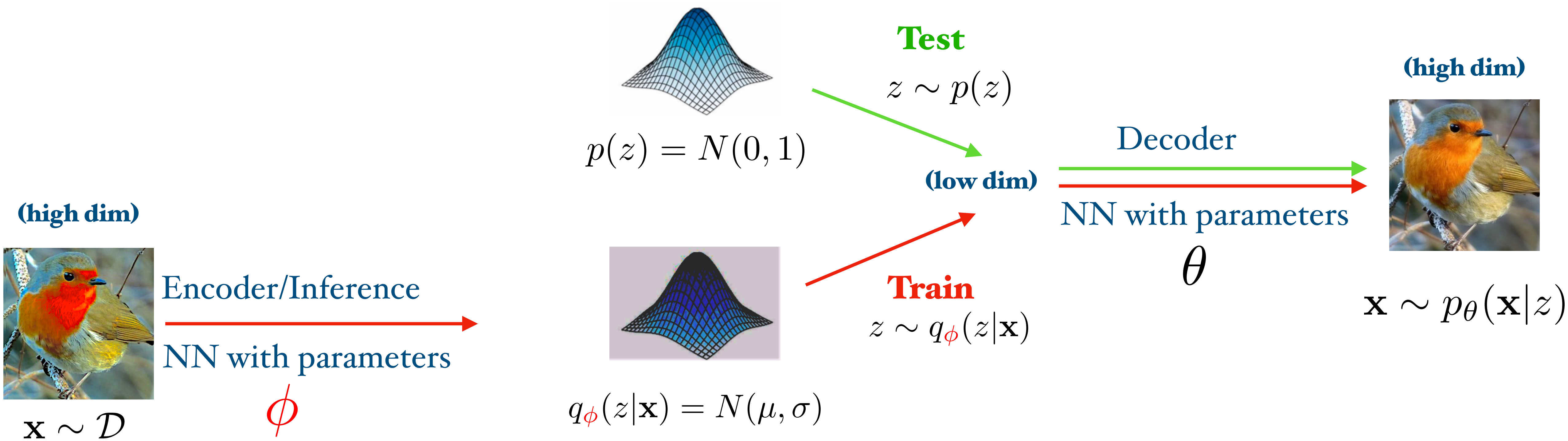
(Kingma et al., 2013)



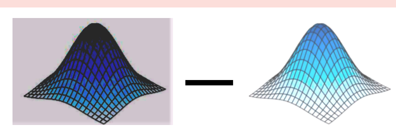
$$\max \quad L_{elbo} := -KL(q_{\phi}(z|\mathbf{x})||p(z)) + \mathbb{E}_{z \sim q_{\phi}(z|\mathbf{x})}(\log p_{\theta}(\mathbf{x}|z))$$

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



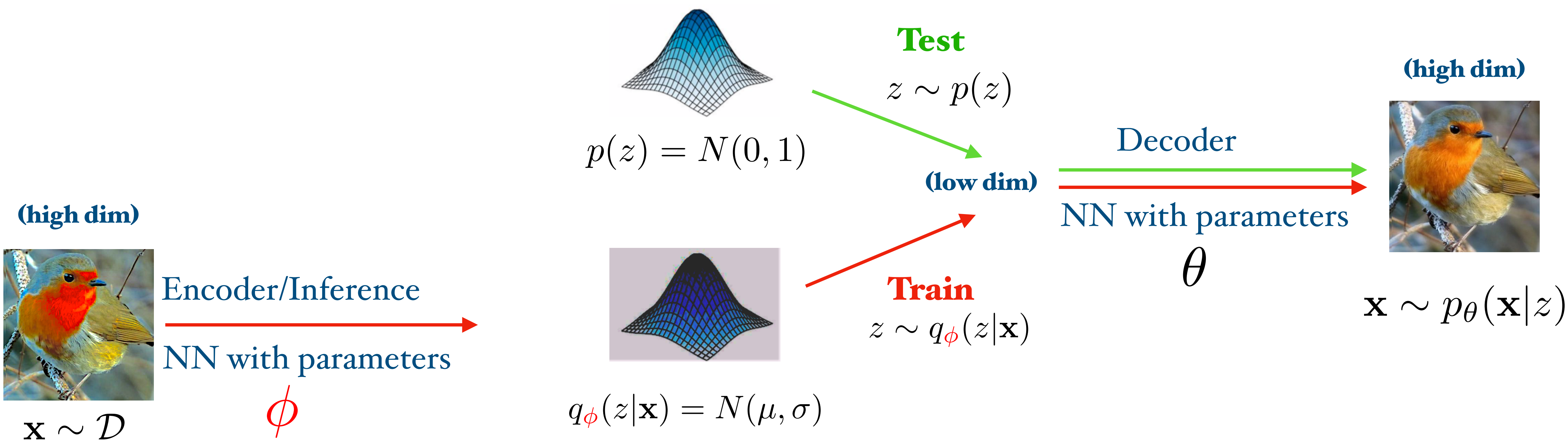
$$\max \quad L_{elbo} := \boxed{-KL(q_\phi(z|\mathbf{x})||p(z))} + \mathbb{E}_{z \sim q_\phi(z|\mathbf{x})}(\log p_\theta(\mathbf{x}|z))$$



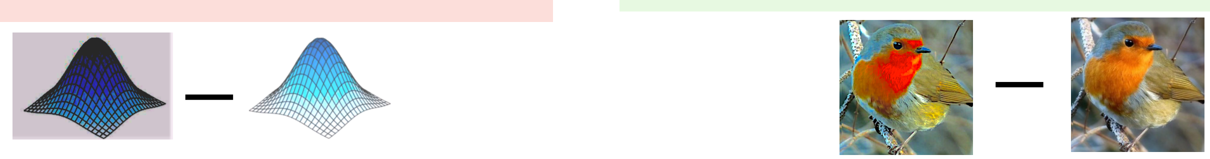
(Regularization)

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



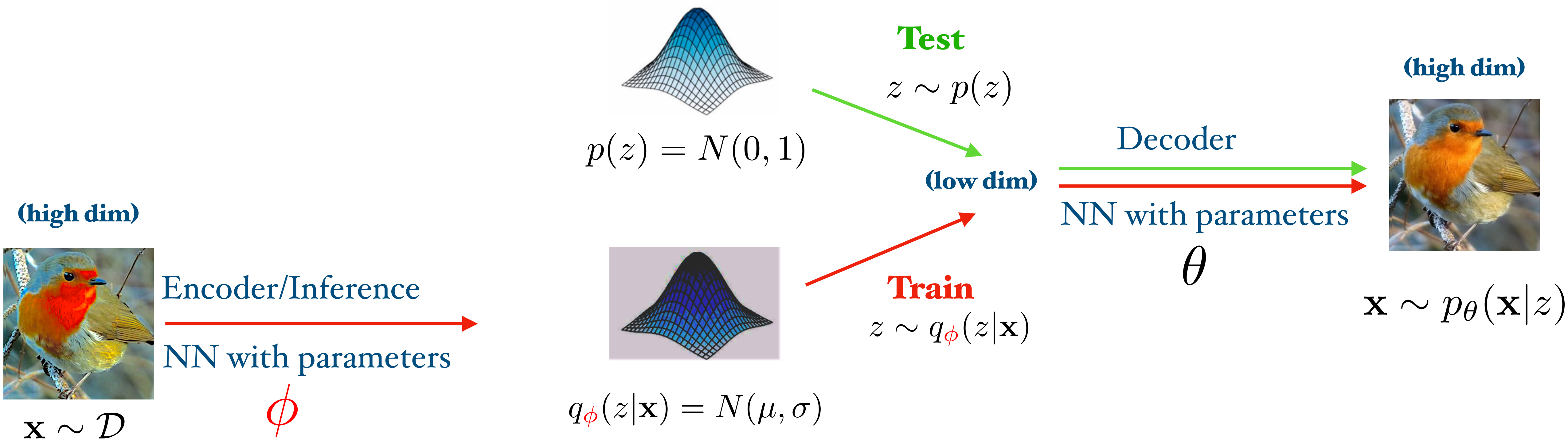
$$\max \quad L_{elbo} := -KL(q_{\phi}(z|\mathbf{x})||p(z)) + \mathbb{E}_{z \sim q_{\phi}(z|\mathbf{x})}(\log p_\theta(\mathbf{x}|z))$$



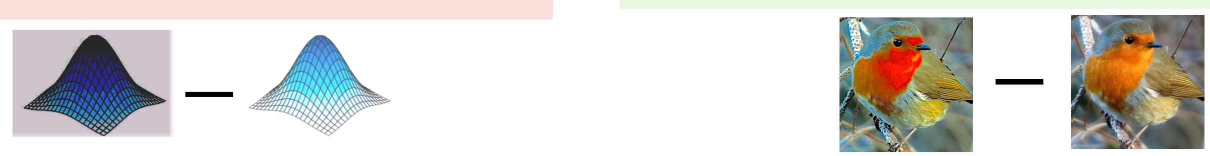
(Regularization) (Reconstruction)

Variational Auto-encoders (VAEs)

(Kingma et al., 2013)



$$\max \quad L_{elbo} := \boxed{-KL(q_\phi(z|\mathbf{x})||p(z))} + \boxed{\mathbb{E}_{z\sim q_\phi(z|\mathbf{x})}(\log p_\theta(\mathbf{x}|z))}$$



(Regularization) (Reconstruction)

- Why does the above objective make sense?
  - It maximizes the lower-bound on the data-loglikelihood $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

VAE — Elbo (quick derivation) $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

VAE — Elbo (quick derivation) $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

Using $KL(p||q) = \mathbb{E}_p(\log p - \log q)$

VAE — Elbo (quick derivation) $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

Using $KL(p||q) = \mathbb{E}_p(\log p - \log q)$

$$KL(q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})) = \mathbb{E}_{q_\phi(z|\mathbf{x})}(\log q_\phi(z|\mathbf{x}) - \log p_\theta(z|\mathbf{x}))$$

VAE — Elbo (quick derivation) $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

Using $KL(p||q) = \mathbb{E}_p(\log p - \log q)$

$$KL(q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})) = \mathbb{E}_{q_\phi(z|\mathbf{x})}(\log q_\phi(z|\mathbf{x}) - \log p_\theta(z|\mathbf{x}))$$

Using $p_\theta(z|\mathbf{x}) = \dfrac{p_\theta(\mathbf{x}, z)}{p_\theta(x)}$ and rearranging terms

VAE — Elbo (quick derivation) $\log p_\theta(\mathbf{x}) \geq L_{elbo}$

Using $KL(p||q) = \mathbb{E}_p(\log p - \log q)$

$$KL(q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x})) = \mathbb{E}_{q_\phi(z|\mathbf{x})}(\log q_\phi(z|\mathbf{x}) - \log p_\theta(z|\mathbf{x}))$$

Using $p_\theta(z|\mathbf{x}) = \dfrac{p_\theta(\mathbf{x}, z)}{p_\theta(x)}$ and rearranging terms

$$\log p_\theta(\mathbf{x}) = \underbrace{\mathbb{E}_{q_\phi(z|\mathbf{x})} \log \frac{p_\theta(\mathbf{x}, z)}{q_\phi(z|\mathbf{x})}}_{L_{elbo}} + \underbrace{KL(q_\phi(z|\mathbf{x})||p_\theta(z|\mathbf{x}))}_{\text{Non-negative}}$$

L_{elbo}

$(-KL(q_\phi(z|\mathbf{x})||p(z)) + \mathbb{E}_{z\sim q_\phi}(\log p_\theta(\mathbf{x}|z)))$

Non-negative

Variational Auto-encoders (VAEs) — Training

Variational Auto-encoders (VAEs) — Training

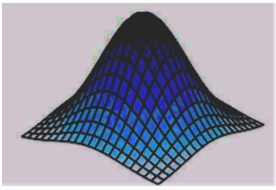
(high dim)

$\mathbf{x} \sim \mathcal{D}$

Encoder/Inference

NN with parameters

ϕ



$q_\phi(z|\mathbf{x}) = N(\mu, \sigma)$

Train

$z \sim q_\phi(z|\mathbf{x})$

(low dim)

Decoder

NN with parameters

θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

Variational Auto-encoders (VAEs) — Training

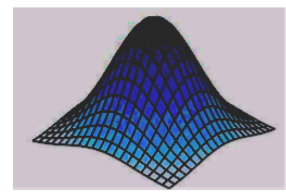
(high dim)

$\mathbf{x} \sim \mathcal{D}$

Encoder/Inference

NN with parameters

ϕ



$q_\phi(z|\mathbf{x}) = N(\mu, \sigma)$

(low dim)

Train

$z \sim q_\phi(z|\mathbf{x})$

Backpropagate?

Decoder

NN with parameters

θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

Variational Auto-encoders (VAEs) — Training

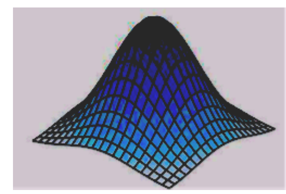
(high dim)

$\mathbf{x} \sim \mathcal{D}$

Encoder/Inference

NN with parameters

ϕ



$q_\phi(z|\mathbf{x}) = N(\mu, \sigma)$

(low dim)

Train

$z \sim q_\phi(z|\mathbf{x})$

Backpropagate?

Decoder

NN with parameters

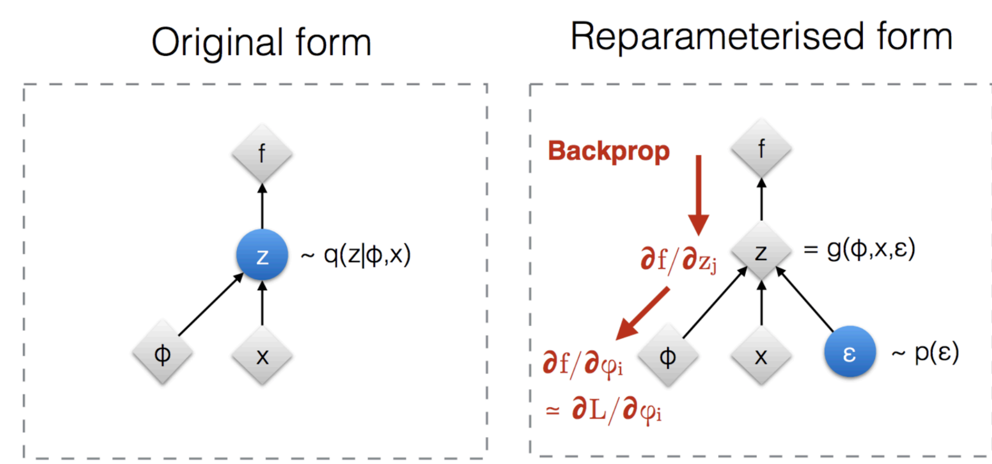
θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

Reparametrization Trick

$z \sim N(\mu, \sigma)$ equivalent to $\mu + \epsilon\, \sigma, \epsilon \sim N(0, 1)$



Puneet Dokania

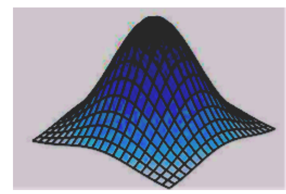
Variational Auto-encoders (VAEs) — Training

(high dim)

$\mathbf{x} \sim \mathcal{D}$

Encoder/Inference

NN with parameters ϕ



$q_\phi(z|\mathbf{x}) = N(\mu, \sigma)$

(low dim)

Train

$z \sim q_\phi(z|\mathbf{x})$

Backpropagate?

Decoder

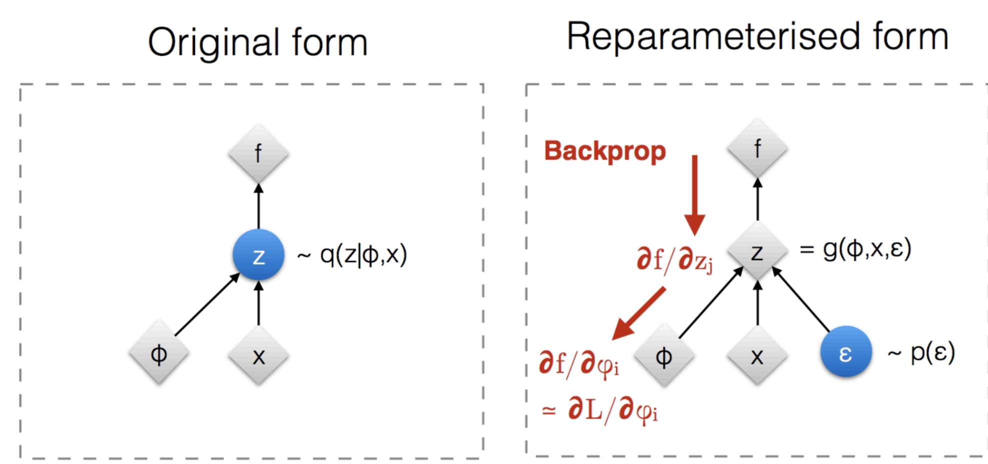
NN with parameters θ

(high dim)

$\mathbf{x} \sim p_\theta(\mathbf{x}|z)$

Reparametrization Trick

$$z \sim N(\mu, \sigma) \quad \text{equivalent to} \quad \mu + \epsilon\,\sigma, \epsilon \sim N(0,1)$$

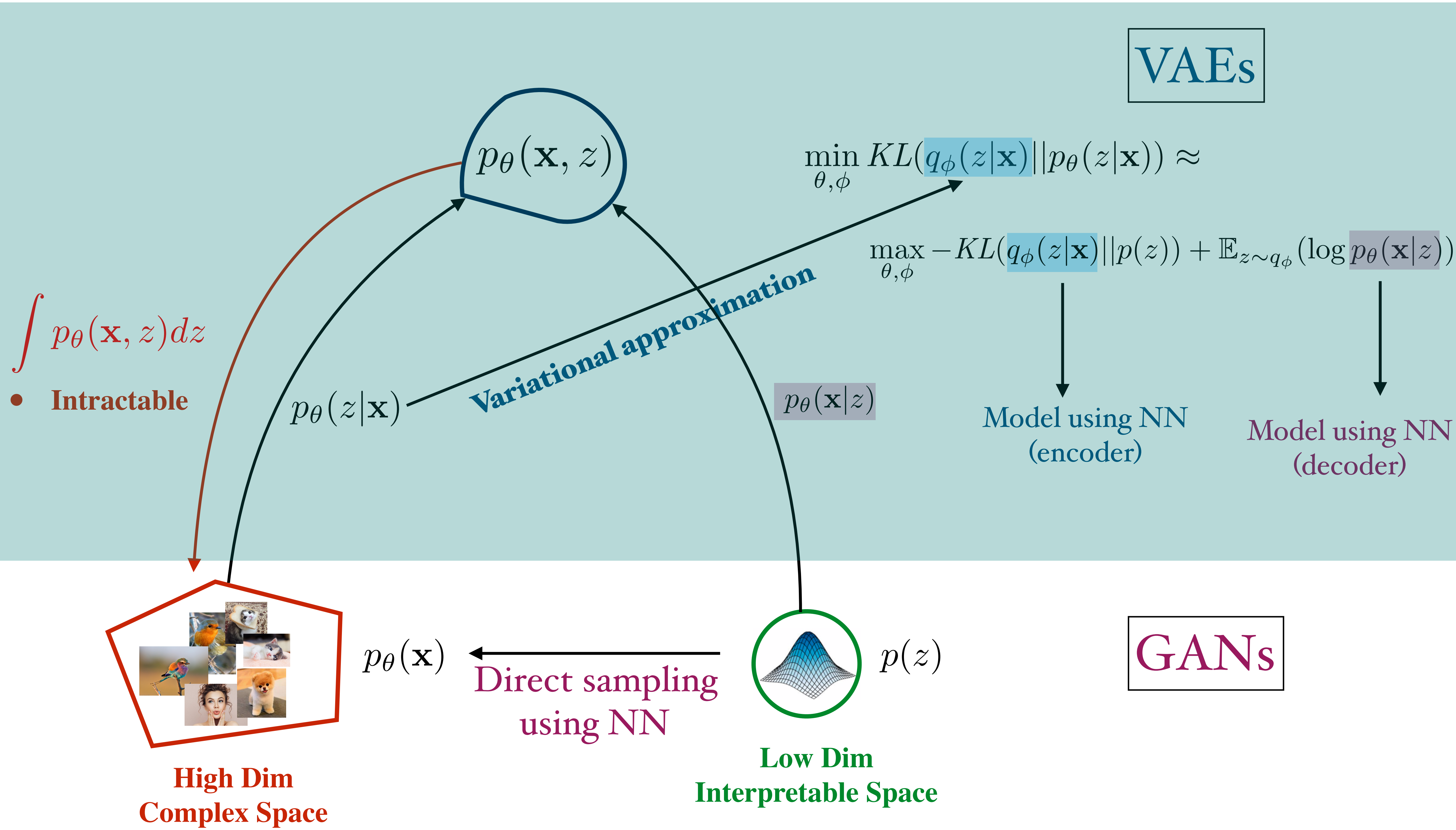


Training

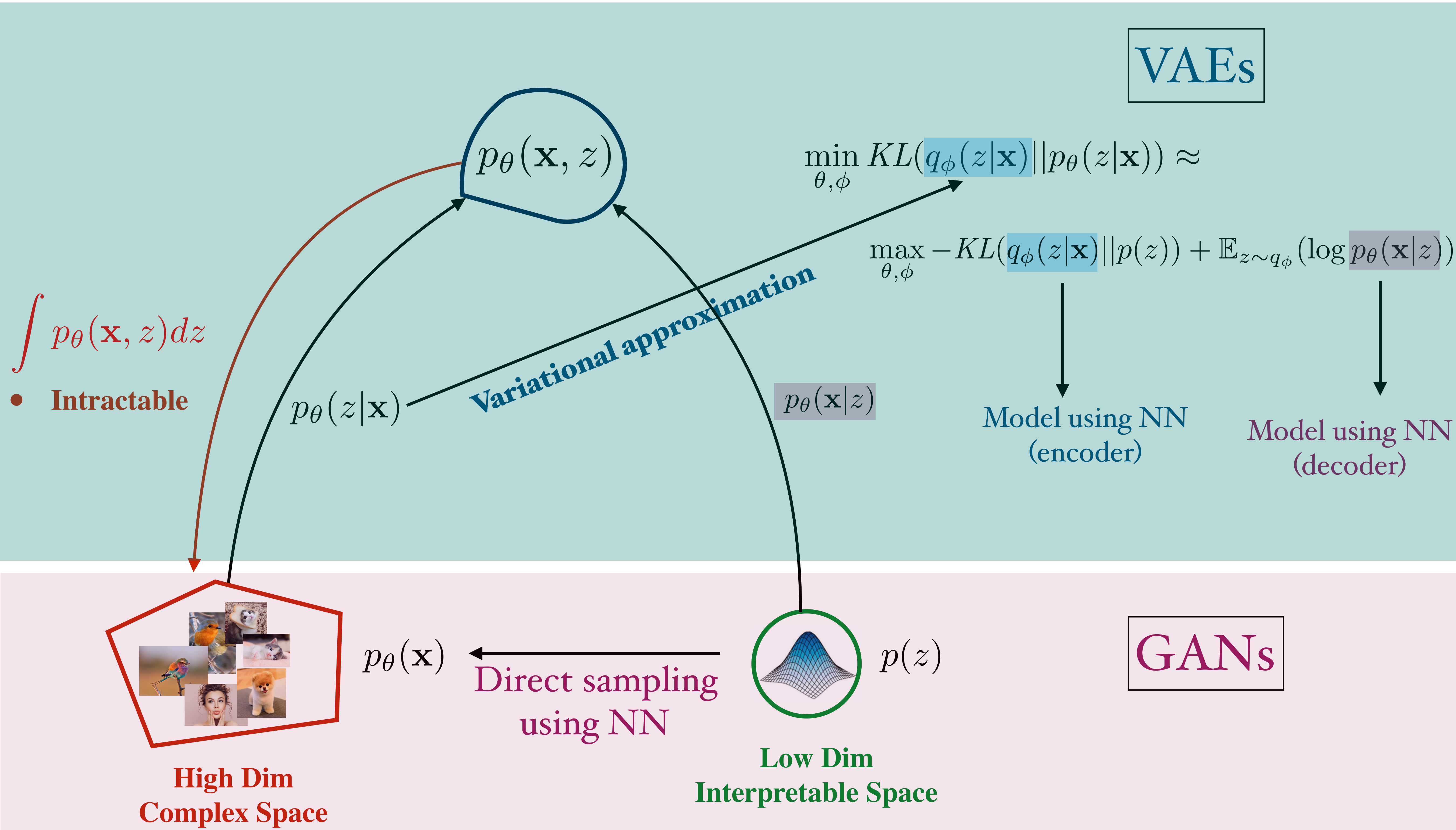
- Random mini batch of M training samples
- Sample M noise $\epsilon \sim N(0,1)$
- Perform forward and backward pass

Generative Adversarial Networks

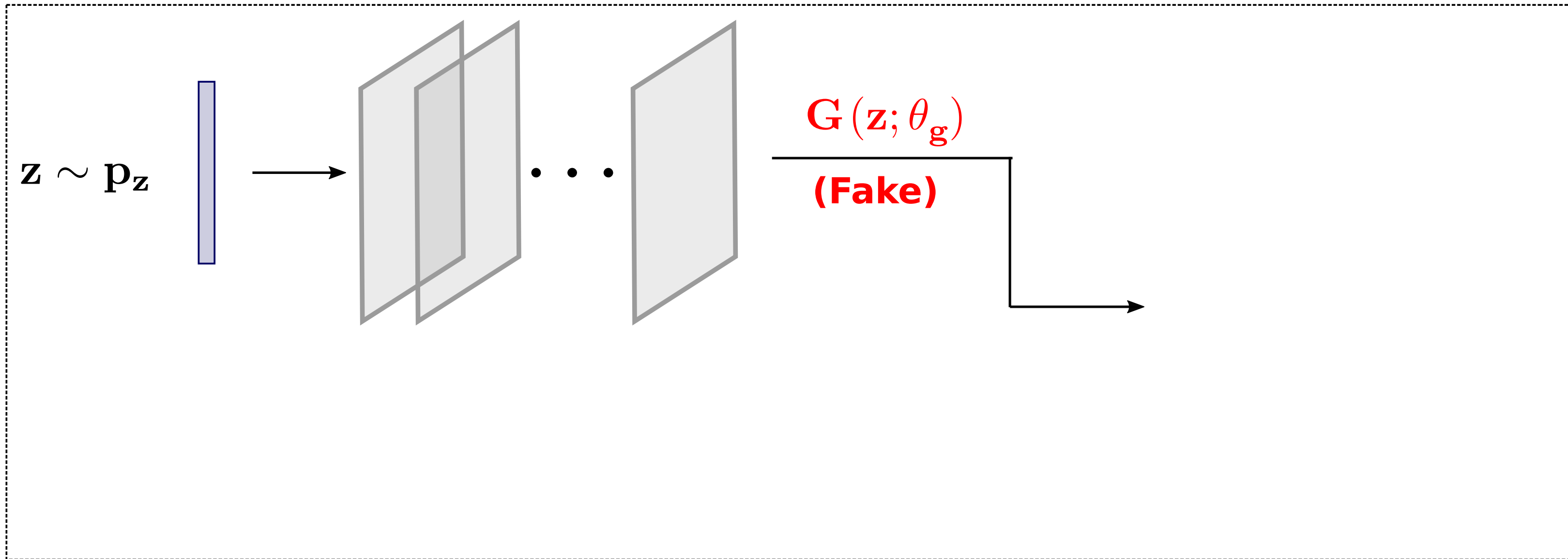
Generative Adversarial Networks



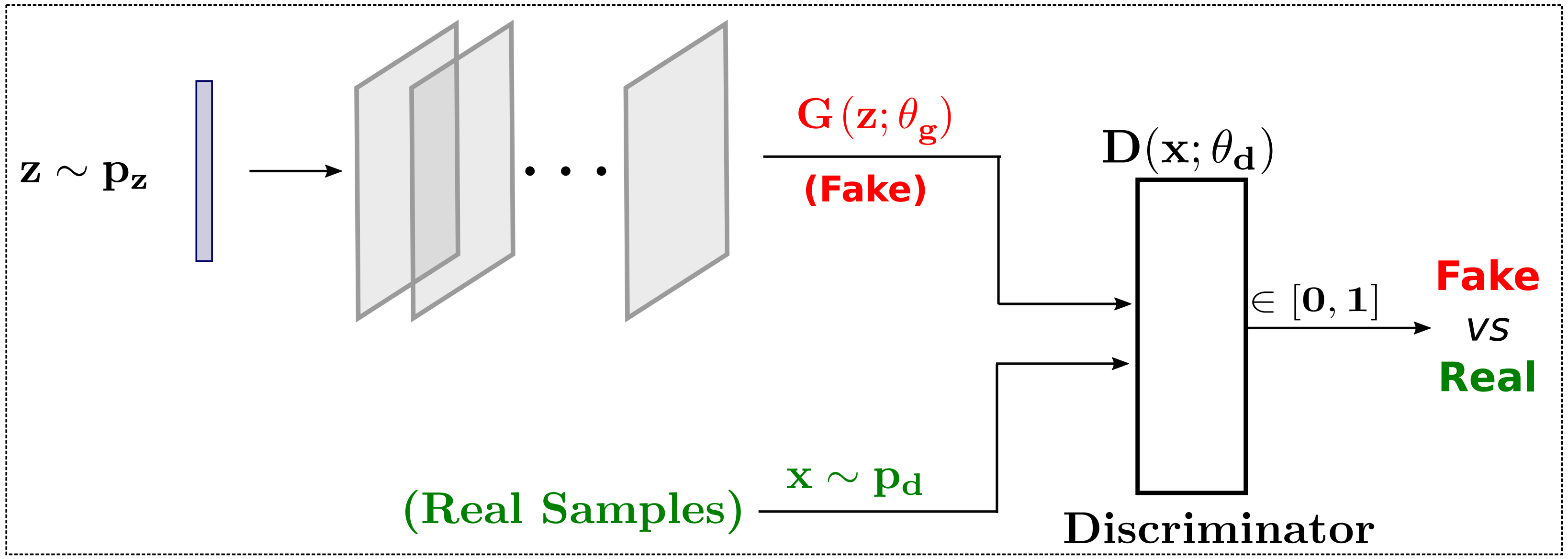
Generative Adversarial Networks



GAN — Objective function (Goodfellow et al., 2014)



GAN — Objective function (Goodfellow et al., 2014)



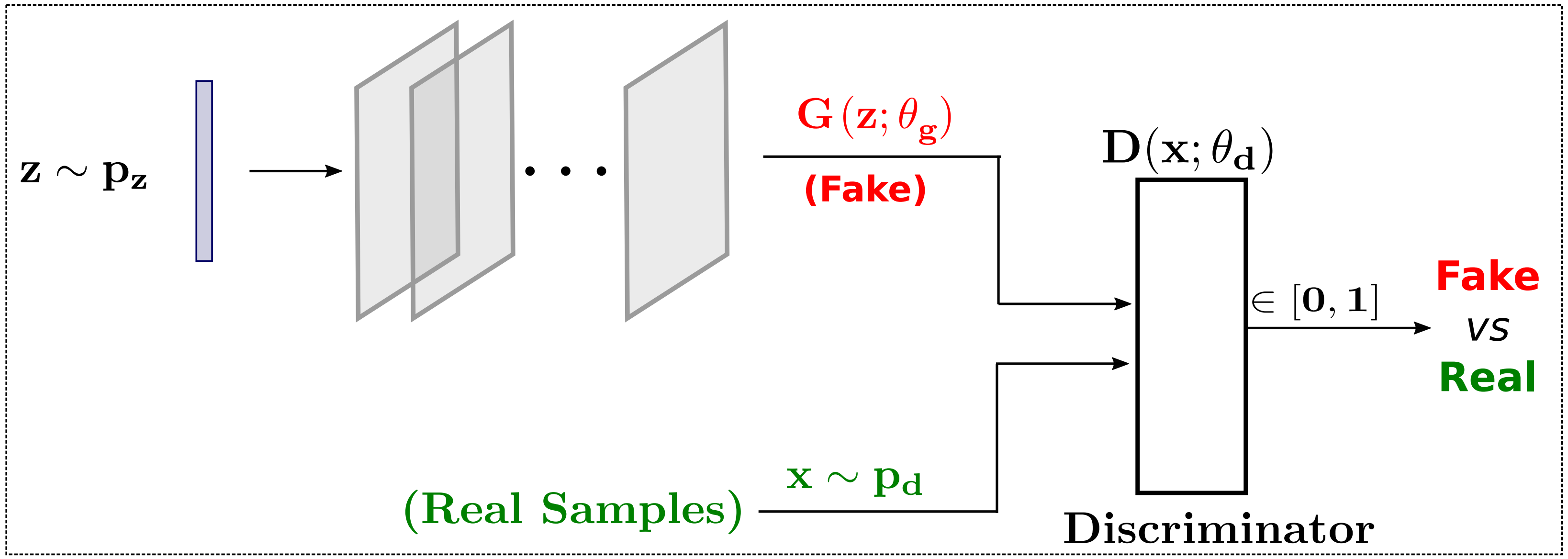
The minimax game

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x \sim p_d} \log D(x; \theta_d) + \mathbb{E}_{z \sim p_z} \log \left(1 - D(G(z; \theta_g); \theta_d)\right)$$

Real

Fake

GAN — Objective function (Goodfellow et al., 2014)



The minimax game

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x \sim p_d} \log D(x; \theta_d) + \mathbb{E}_{z \sim p_z} \log \left(1 - D(G(z; \theta_g); \theta_d)\right)$$

Real **Fake**

at equilibrium

$$p_g = p_d$$

GAN — A closer look at the generator

GAN — A closer look at the generator

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x\sim p_d} \log D(x;\theta_d) + \mathbb{E}_{z\sim p_z} \log\big(1 - D(G(z;\theta_g);\theta_d)\big)$$

GAN — A closer look at the generator

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x\sim p_d} \log D(x;\theta_d) + \mathbb{E}_{z\sim p_z} \log\left(1 - D(\underbrace{G(z;\theta_g)}_{x_g};\theta_d)\right)$$

GAN — A closer look at the generator

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x \sim p_d} \log D(x; \theta_d) + \mathbb{E}_{z \sim p_z} \log \big(1 - D(\underbrace{G(z; \theta_g)}_{x_g}; \theta_d)\big)$$

$$\nabla_{x_g} \log \big(1 - D(x_g)\big) = -D(x_g)$$

GAN — A closer look at the generator

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x\sim p_d} \log D(x;\theta_d) + \mathbb{E}_{z\sim p_z} \log\left(1 - D(\underbrace{G(z;\theta_g)}_{x_g};\theta_d)\right)$$

$$\nabla_{x_g} \log\left(1 - D(x_g)\right) = -D(x_g)$$

Vanishing gradient

- if the discriminator is confident $D(x_g) \approx 0$

GAN — A closer look at the generator

$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x \sim p_d} \log D(x; \theta_d) + \mathbb{E}_{z \sim p_z} \log \big(1 - D(\underbrace{G(z; \theta_g)}_{x_g}; \theta_d)\big)$$

$$\nabla_{x_g} \log \big(1 - D(x_g)\big) = -D(x_g)$$

Vanishing gradient

- if the discriminator is confident $D(x_g) \approx 0$

Modify generator objective, keep discriminator intact

$$\min_{\theta_g} \mathbb{E}_{z \sim p_z} - \log D(G(z; \theta_g); \theta_d)$$

Problems with GANs

Problems with GANs

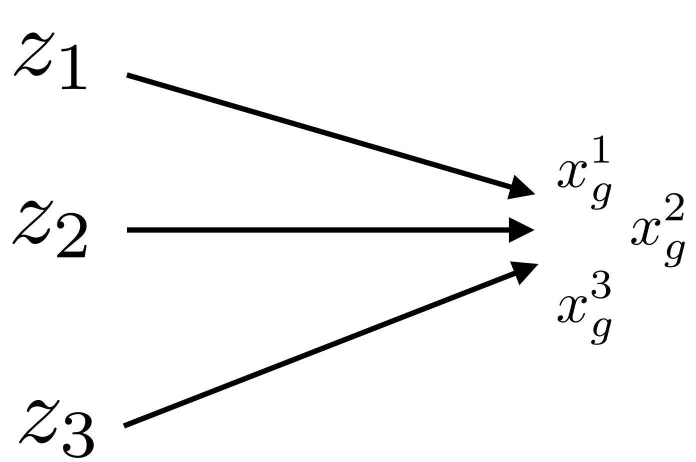
- SGD was not designed to achieve Nash equilibrium
    - Training of GANs is highly unstable

Problems with GANs

- SGD was not designed to achieve Nash equilibrium
  - Training of GANs is highly unstable
- Mode Collapse
  - Many to one type mapping

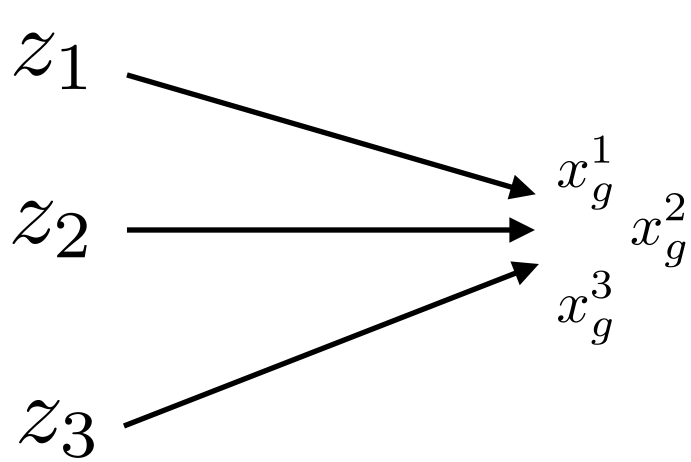
Problems with GANs

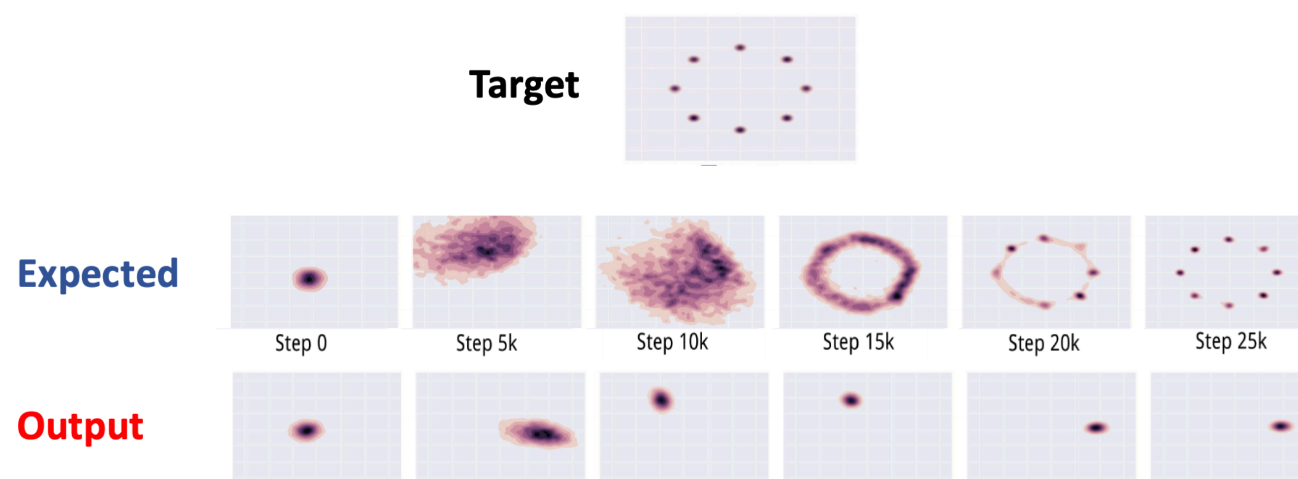
- SGD was not designed to achieve Nash equilibrium
  - Training of GANs is highly unstable
- Mode Collapse
  - Many to one type mapping $G(z_i;\theta_g) \approx G(z_j;\theta_g), z_i \neq z_j$



Problems with GANs

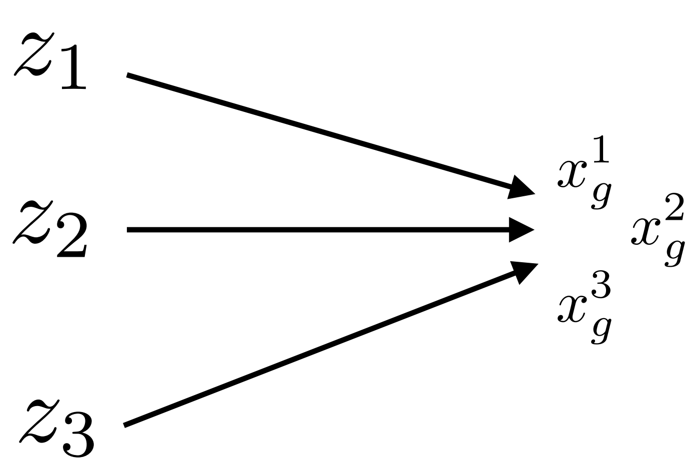
- SGD was not designed to achieve Nash equilibrium
  - Training of GANs is highly unstable
- Mode Collapse
  - Many to one type mapping $G(z_i; \theta_g) \approx G(z_j; \theta_g), z_i \neq z_j$

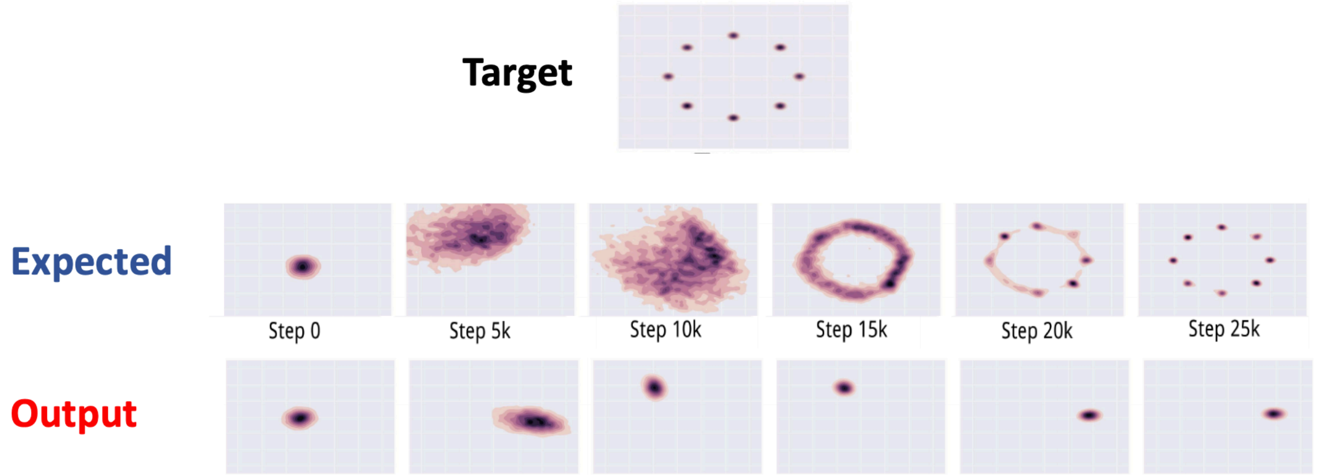




Problems with GANs

- SGD was not designed to achieve Nash equilibrium
  - Training of GANs is highly unstable
- Mode Collapse
  - Many to one type mapping $G(z_i;\theta_g) \approx G(z_j;\theta_g), z_i \neq z_j$





- Tons of papers in 2-3 years trying to resolve these issues

GAN Variants — Discriminator focused

GAN Variants — Discriminator focused

Tension between Capacity and Generalizability

GAN Variants — Discriminator focused

Tension between Capacity and Generalizability

- Most of the GAN variants focus on the
  - objective/generalizability of the discriminator
    - low Lipschitz constant — objective modification
    - 1-Lipschitz discriminator
      - WGAN — objective modification + weight clipping
      - WGAN (Gradient penalty) — objective modification
  - training stability of the discriminator
    - Spectral norm GAN — acts on the linear mappings
    - Orthonormal GAN — acts on the linear weights
    - Stable rank GAN — acts on the linear weights

GAN Variants — Discriminator focused

GAN Variants — Discriminator focused

$$\mathbf{x} \in \mathbb{R}^m \xrightarrow{f_W} f_W(\mathbf{x}) \in \mathbb{R}^n$$

GAN Variants — Discriminator focused

$$\mathbf{x} \in \mathbb{R}^m \xrightarrow{f_W} f_W(\mathbf{x}) \in \mathbb{R}^n$$

Global Lipschitz Constant $$L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$$ **Sensitivity**

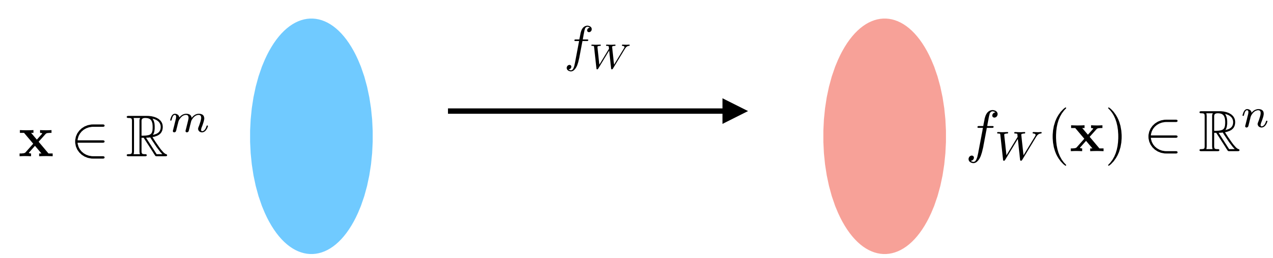
GAN Variants — Discriminator focused

$$\mathbf{x} \in \mathbb{R}^m \xrightarrow{f_W} f_W(\mathbf{x}) \in \mathbb{R}^n$$

Global Lipschitz Constant $$L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$$ **Sensitivity**

Local Lipschitz Constant $$L_l(\mathbf{x}) = ||J_f(x)||_2$$ where $$||W||_2 = \sigma_{max}(W)$$

GAN Variants — Discriminator focused

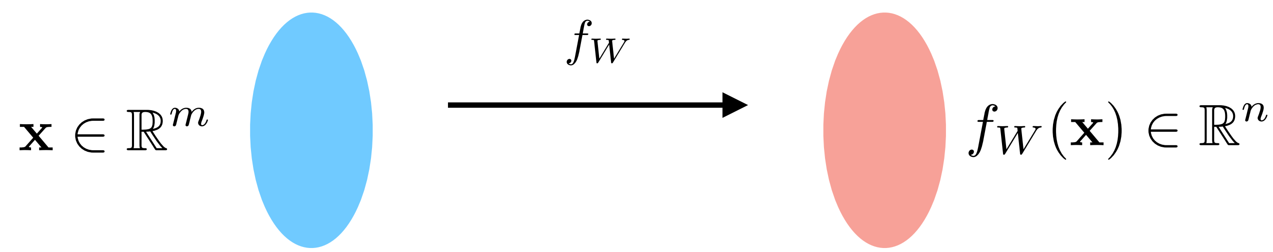


Global Lipschitz Constant $L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$ **Sensitivity**

Local Lipschitz Constant $L_l(\mathbf{x}) = ||J_f(x)||_2$ where $||W||_2 = \sigma_{max}(W)$

Lipschitz Constant Upperbound NN $L_l(f) \leq ||W_1||_2 \cdots ||W_l||_2$

GAN Variants — Discriminator focused



Global Lipschitz Constant $L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$ **Sensitivity**

Local Lipschitz Constant $L_l(\mathbf{x}) = ||J_f(x)||_2$ where $||W||_2 = \sigma_{max}(W)$

Lipschitz Constant Upperbound NN $L_l(f) \leq ||W_1||_2 \cdots ||W_l||_2$

GAN Variants — Discriminator focused

$$\mathbf{x} \in \mathbb{R}^m \xrightarrow{f_W} f_W(\mathbf{x}) \in \mathbb{R}^n$$

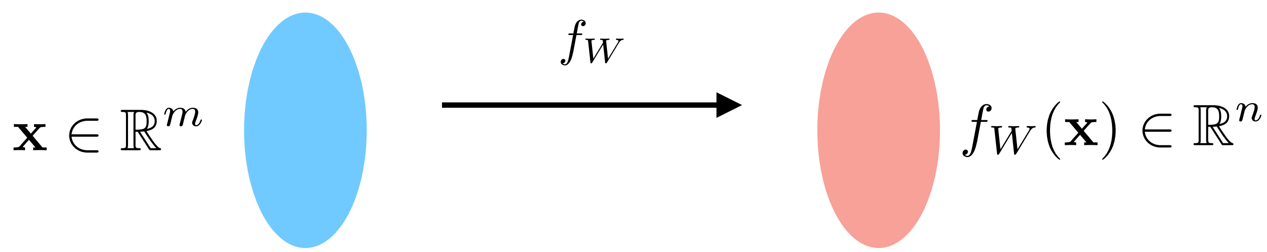
Global Lipschitz Constant $\quad L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$ **Sensitivity**

Local Lipschitz Constant $\quad L_l(\mathbf{x}) = ||J_f(x)||_2$ where $||W||_2 = \sigma_{max}(W)$

Lipschitz Constant Upperbound NN $\quad L_l(f) \leq ||W_1||_2 \cdots ||W_l||_2$

Spectral Norm GAN (Miyato et al., ICLR18) $\quad \bar{W}_{SN} = \frac{W}{\sigma_{max}(W)}$ Optimal?

GAN Variants — Discriminator focused



Global Lipschitz Constant $$L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$$ **Sensitivity**

Local Lipschitz Constant $L_l(\mathbf{x}) = ||J_f(x)||_2$ where $||W||_2 = \sigma_{max}(W)$

Lipschitz Constant Upperbound NN $L_l(f) \leq ||W_1||_2 \cdots ||W_l||_2$

Spectral Norm GAN (Miyato et al., ICLR18) $$\bar{W}_{SN} = \frac{W}{\sigma_{max}(W)}$$ Optimal?

Orthonormal GAN $||WW^\top - I||_F$ $\sigma_i(W) \approx 1, \forall i$ Dynamical Isometry

GAN Variants — Discriminator focused

$$\mathbf{x} \in \mathbb{R}^m \xrightarrow{f_W} f_W(\mathbf{x}) \in \mathbb{R}^n$$

Global Lipschitz Constant $L_g = \max_{\mathbf{x}_i, \mathbf{x}_j} \frac{||f(\mathbf{x}_i) - f(\mathbf{x}_j)||_2}{||\mathbf{x}_i - \mathbf{x}_j||_2}$ **Sensitivity**

Local Lipschitz Constant $L_l(\mathbf{x}) = ||J_f(x)||_2$ where $||W||_2 = \sigma_{max}(W)$

Lipschitz Constant Upperbound NN $L_l(f) \leq ||W_1||_2 \cdots ||W_l||_2$

Spectral Norm GAN (Miyato et al., ICLR18) $\bar{W}_{SN} = \frac{W}{\sigma_{max}(W)}$ Optimal?

Orthonormal GAN $||WW^\top - I||_F$ $\sigma_i(W) \approx 1, \forall i$ Dynamical Isometry

WGAN-GP (Guljarani et al., NeurIPS17) $(||J_f(\mathbf{x})||_2 - 1)^2$ Additional penalty — interpolated samples

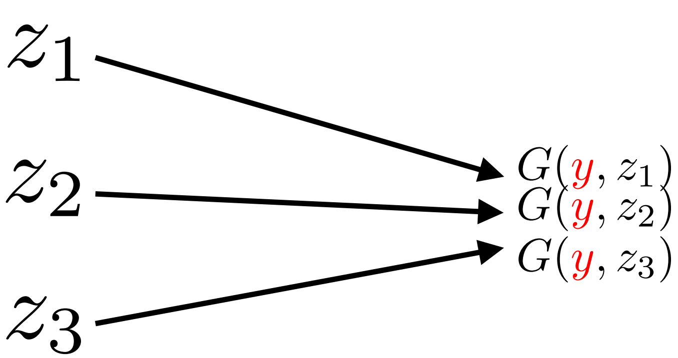
GAN Variants — Generator focused

GAN Variants — Generator focused

- Mode collapse GANs — once again

GAN Variants — Generator focused

- Mode collapse GANs — once again



condition y

GAN Variants — Generator focused

- Mode collapse GANs — once again

$$z_1 \rightarrow G(y, z_1)$$
$$z_2 \rightarrow G(y, z_2)$$
$$z_3 \rightarrow G(y, z_3)$$

$$\max_{\theta_g} \mathbb{E}_{z_i, z_j} \frac{||G(y, z_i) - G(y, z_j)||}{||z_i - z_j||}$$

(Yang et al., ICLR19)

(hinge type version)

condition y

GAN Variants — Generator focused

- Mode collapse GANs — once again

$z_1 \rightarrow G(y, z_1)$

$z_2 \rightarrow G(y, z_2)$

$z_3 \rightarrow G(y, z_3)$

condition y

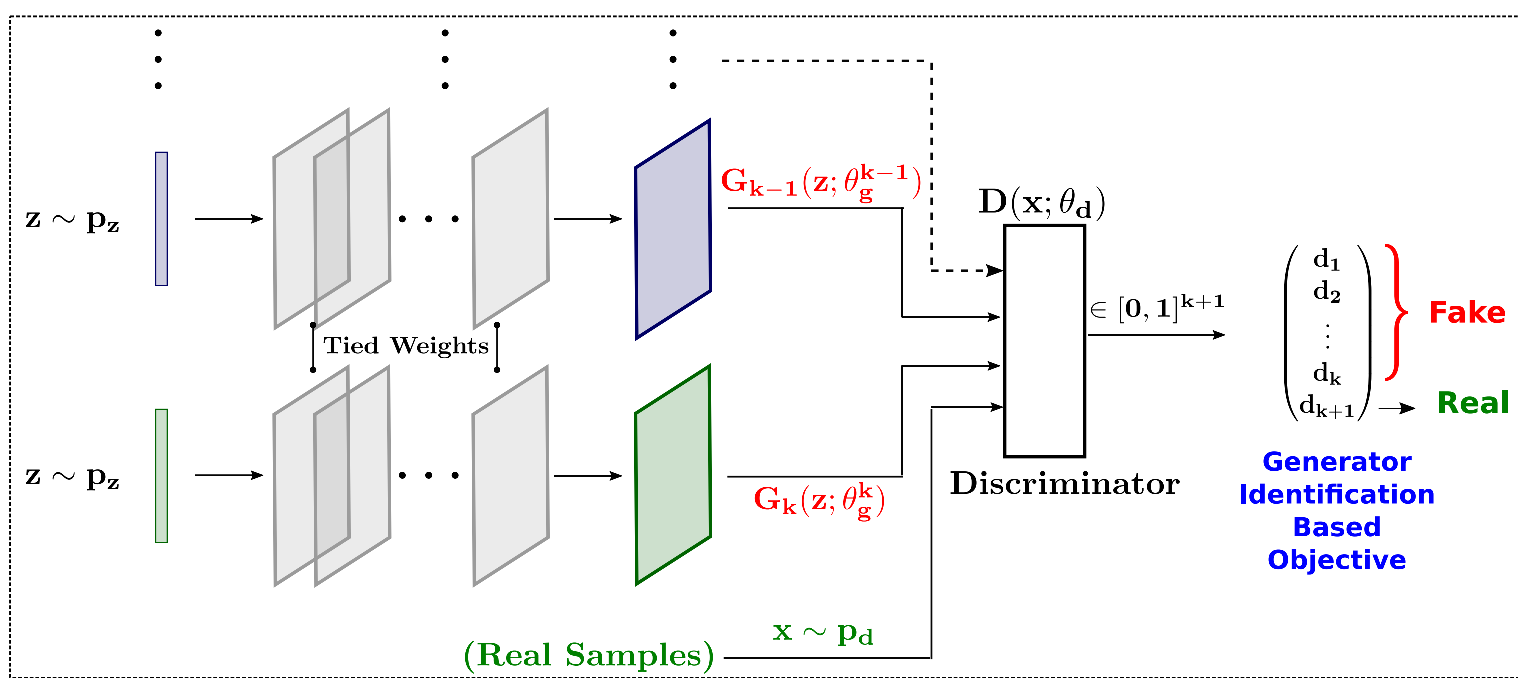
$$\max_{\theta_g} \mathbb{E}_{z_i, z_j} \frac{||G(y, z_i) - G(y, z_j)||}{||z_i - z_j||}$$

(Yang et al., ICLR19)

(hinge type version)

MAD-GAN — Use multiple generators

(Ghosh et al., CVPR18)



Big-GAN (Brock et al., ICLR19)

Big-GAN (Brock et al., ICLR19)

Focus on both — Discriminator and Generator

Big-GAN (Brock et al., ICLR19)

Focus on both — Discriminator and Generator

Among other tricks —

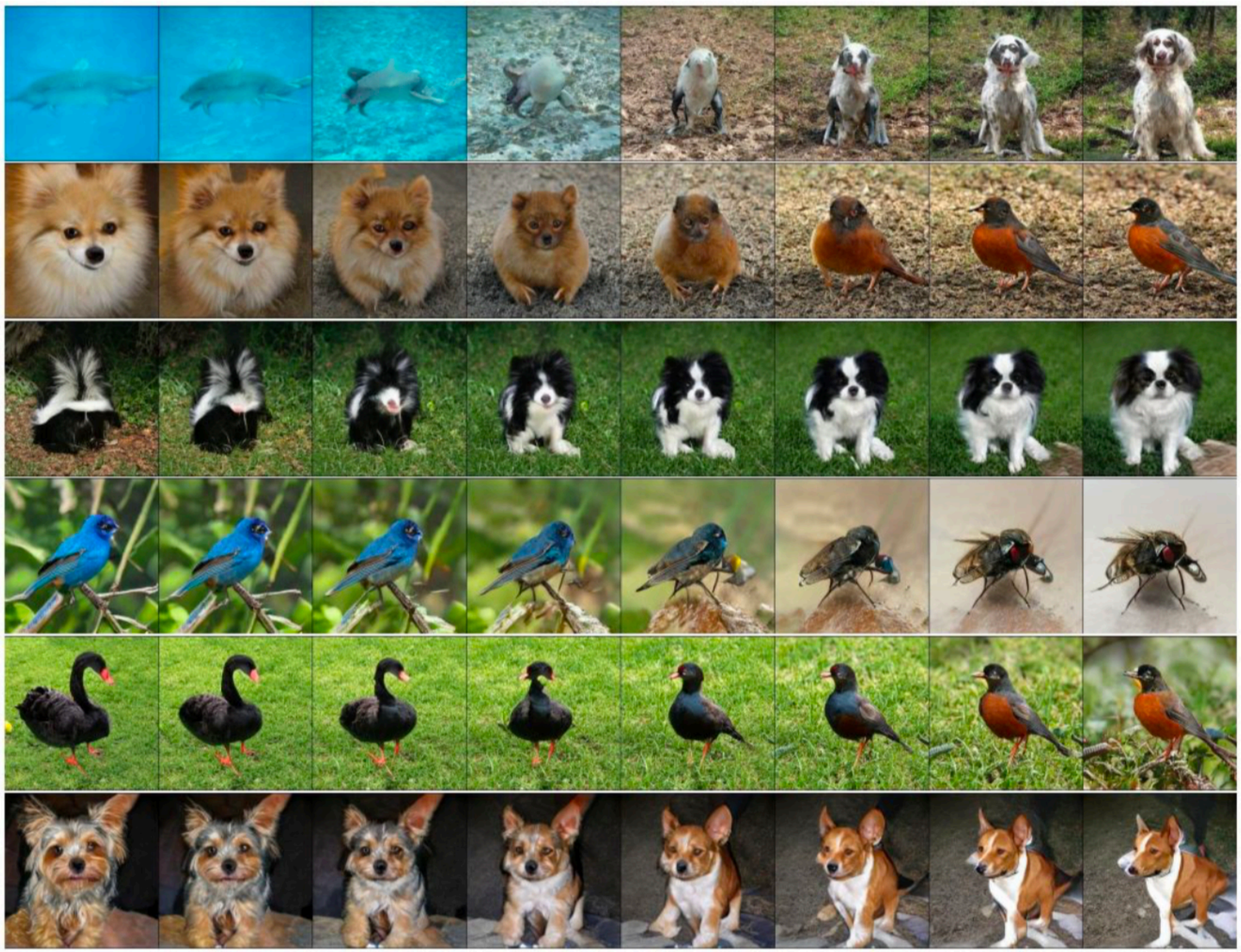
- Spectral norm on both Discriminator and Generator
- Orthogonal initalization of parameters — training speed improves
- A new variant of orthonormal regularisation on the Generator
  - all singular values nearly one
- Increased batch size (8x), increase width (50%), etc.

Big-GAN (Brock et al., ICLR19)

512x512 generations

Big-GAN (Brock et al., ICLR19)

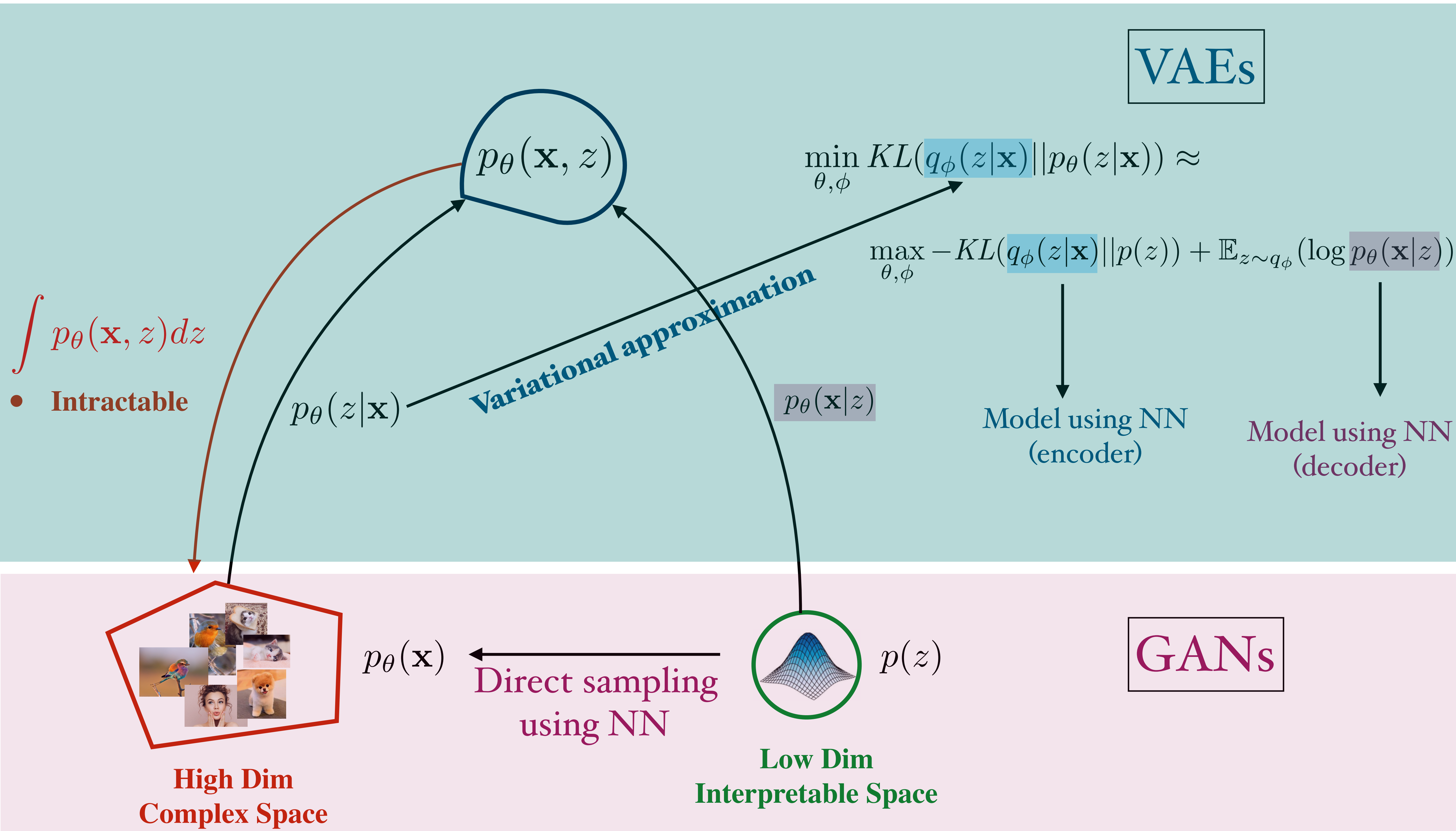
Interpolation



GAN — few insights

- Focus on both — generator and discriminator
- Trade-off between the capacity and the generalizability of discriminator
- Think about the invertibility of the generator
- Invertibility might not be enough, think about improving the condition number

VAEs and GANs



Thank You

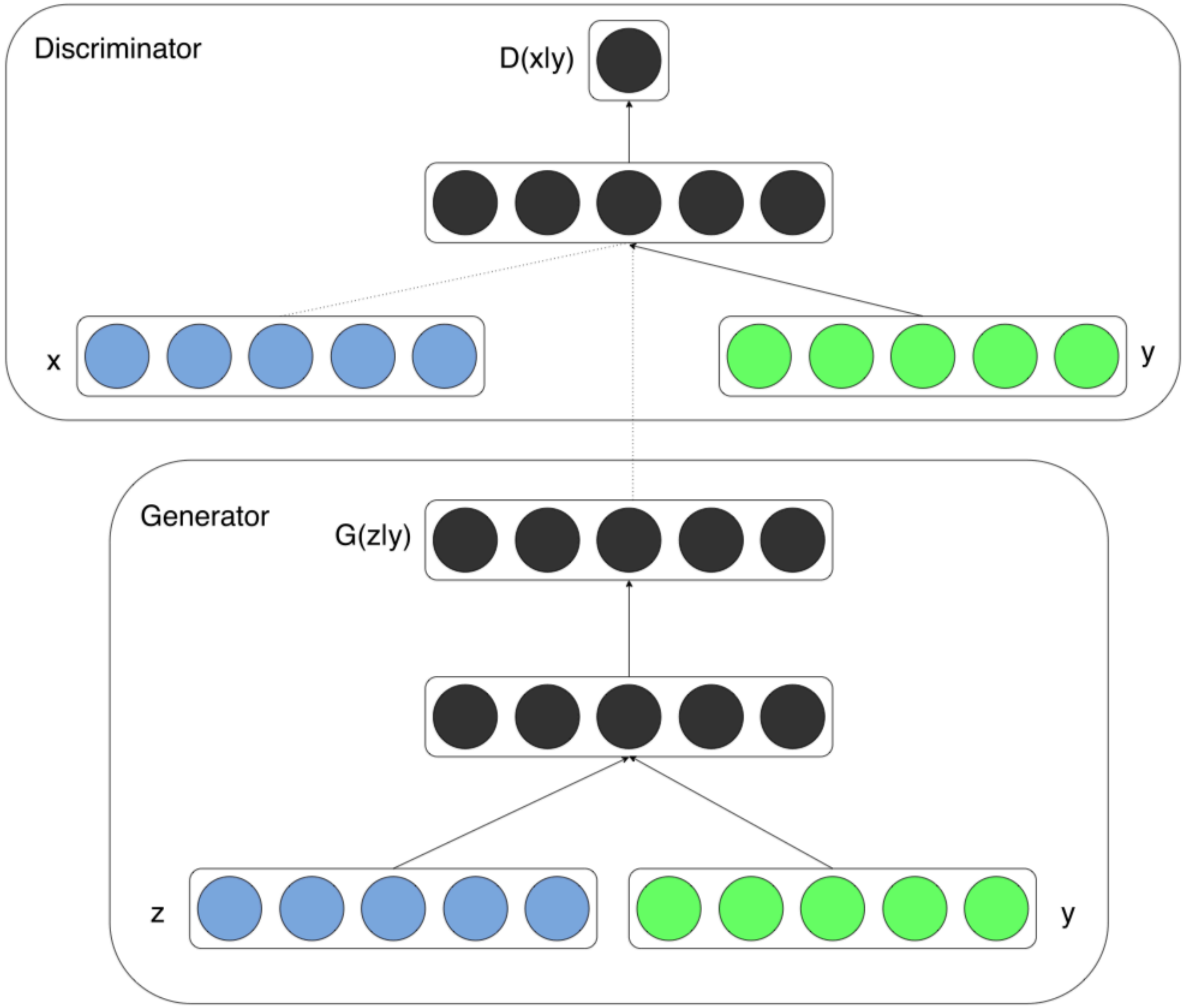
Deep Generative Models — Applications of GANs
(part 2)

Puneet K. Dokania
(University of Oxford)

<u>Presented at</u>

Machine Learning Summer School
IIIT Hyderabad, India
(9th July 2019)

Conditional GAN (Mirza et al., 2014)



$$\min_{\theta_g} \max_{\theta_d} \mathbb{E}_{x \sim p_d} \log D(x|y; \theta_d) + \mathbb{E}_{z \sim p_z} \log \big(1 - D(G(z|y; \theta_g); \theta_d)\big)$$

Image to Image Translation — Paired
(Isola et al., CVPR17)

Image to Image Translation — Paired
(Isola et al., CVPR17)

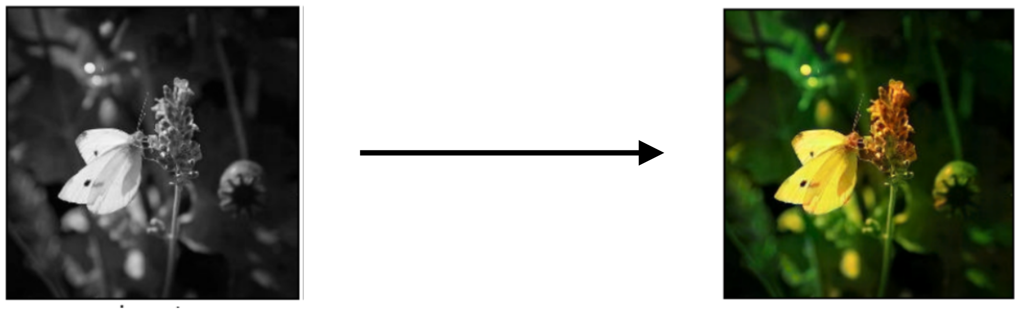
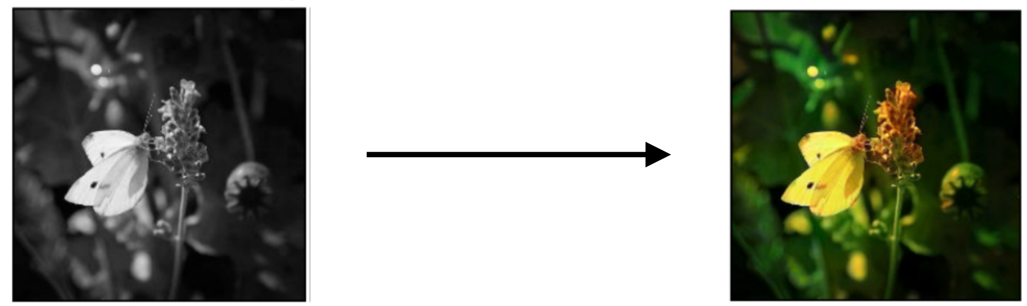


Image to Image Translation — Paired
(Isola et al., CVPR17)



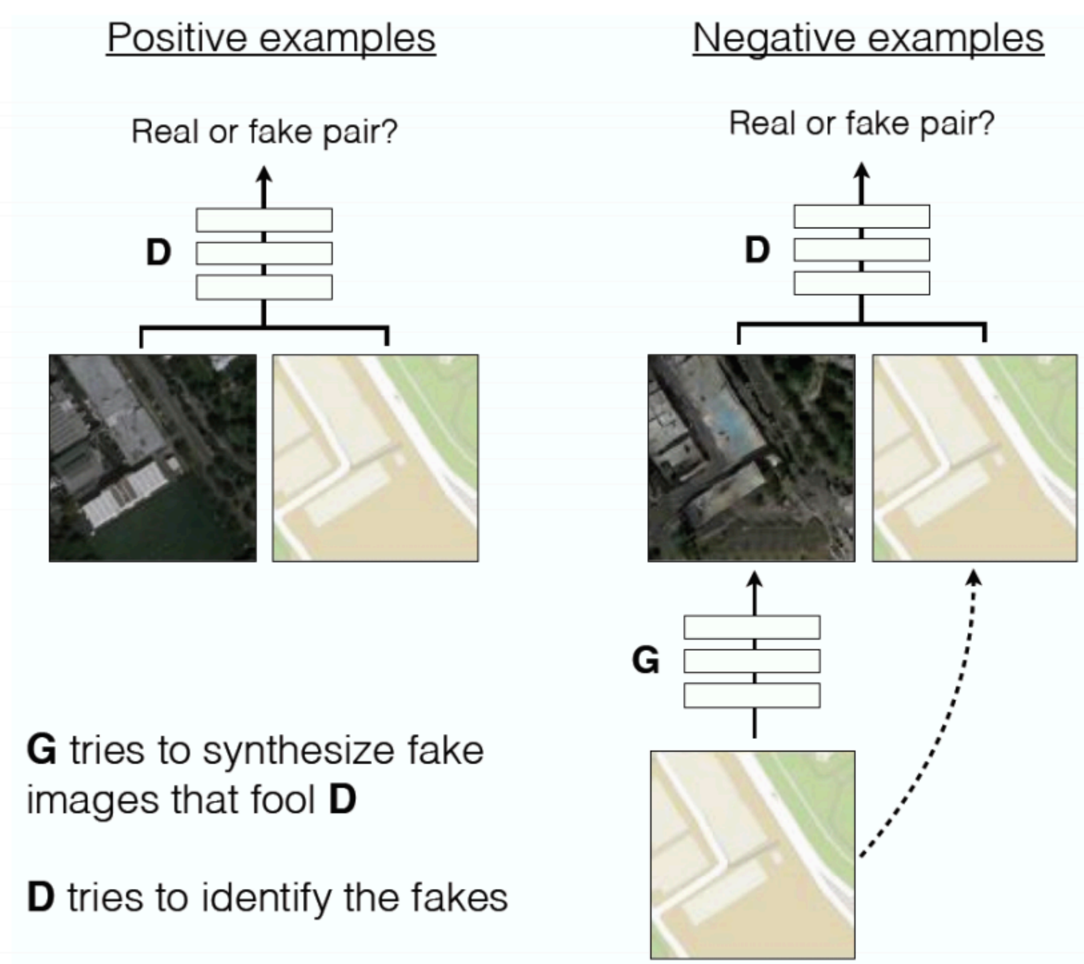


Figure 2 in the original paper.

$$\mathcal{L}_{L1}(G) = \mathbb{E}_{x,y,z}[\|y - G(x,z)\|_1].$$

$$G^* = \arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda\mathcal{L}_{L1}(G).$$

Image to Image Translation — Paired
(Isola et al., CVPR17)

Image to Image Translation — Paired
(Isola et al., CVPR17)

Image to Image Translation — Paired

(Isola et al., CVPR17)

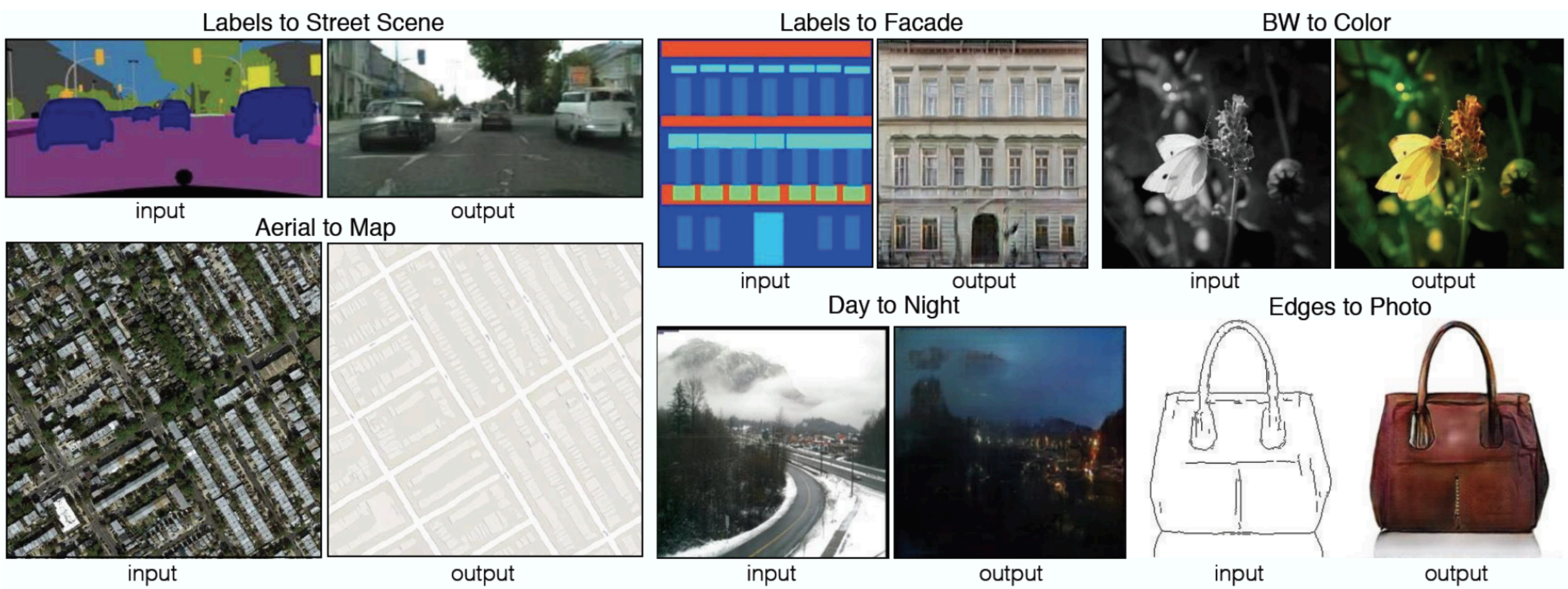


Figure 1 in the original paper.

Image to Image Translation — Paired
(Isola et al., CVPR17)

- **No diversity — Why?**
  - **Noise gets absorbed**

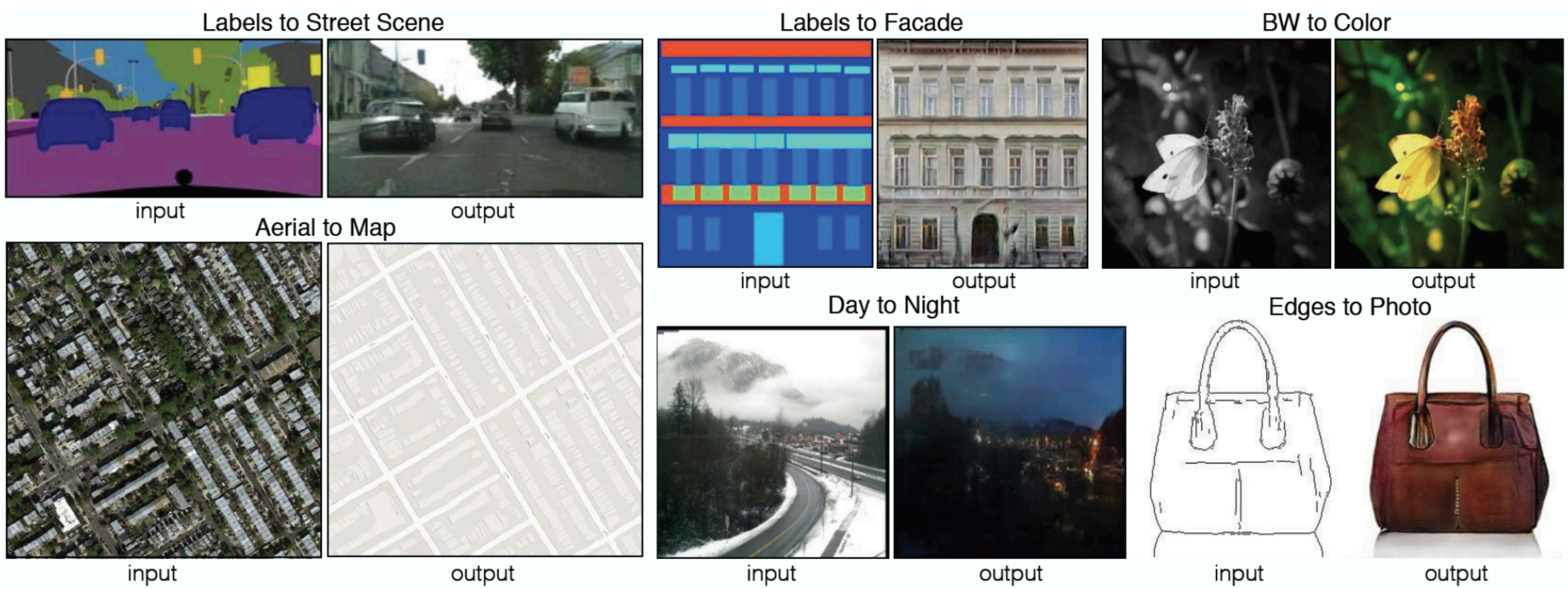
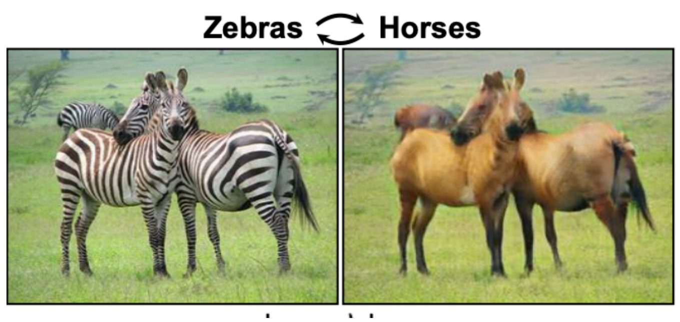


Figure 1 in the original paper.

Image to Image Translation — Unpaired (CycleGAN) Zhu et al., ICCV17

Image to Image Translation — Unpaired (CycleGAN) Zhu et al., ICCV17



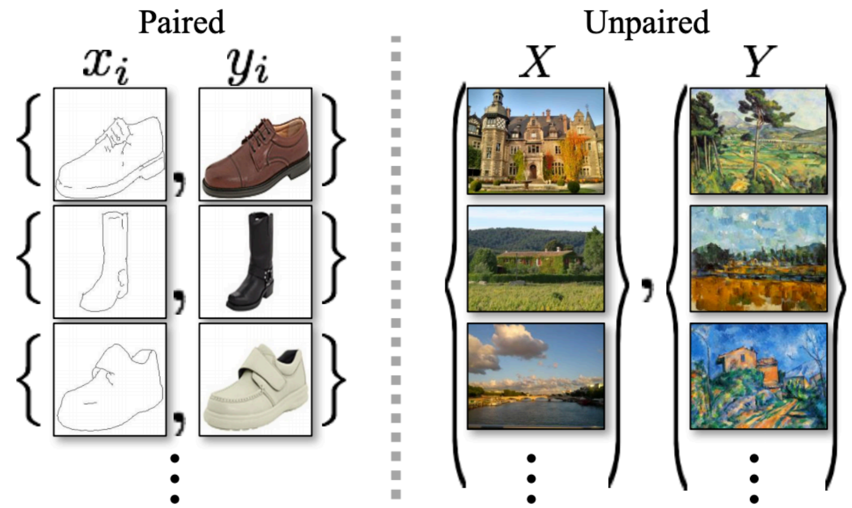
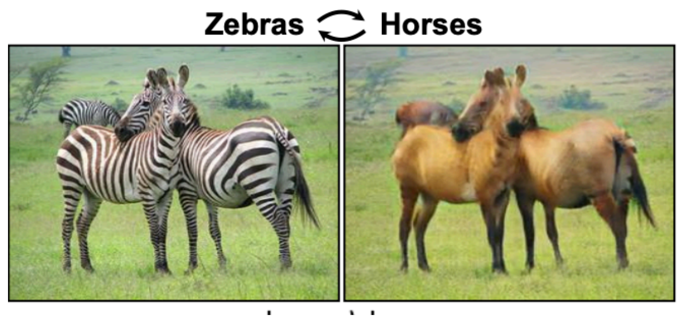
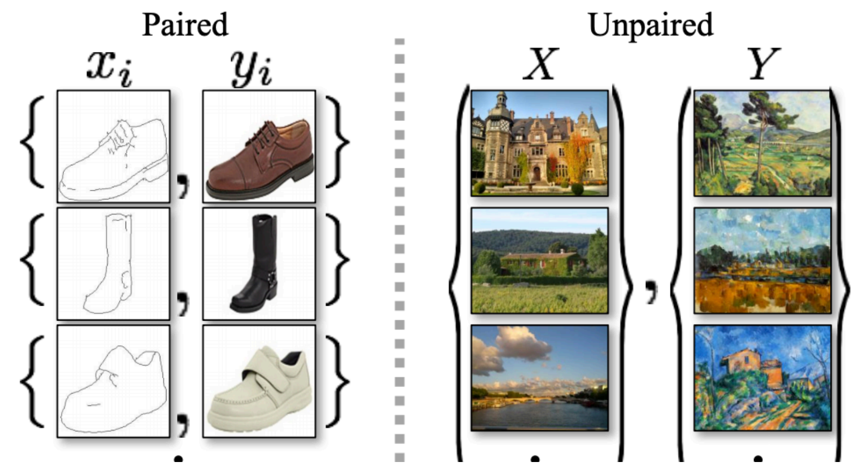
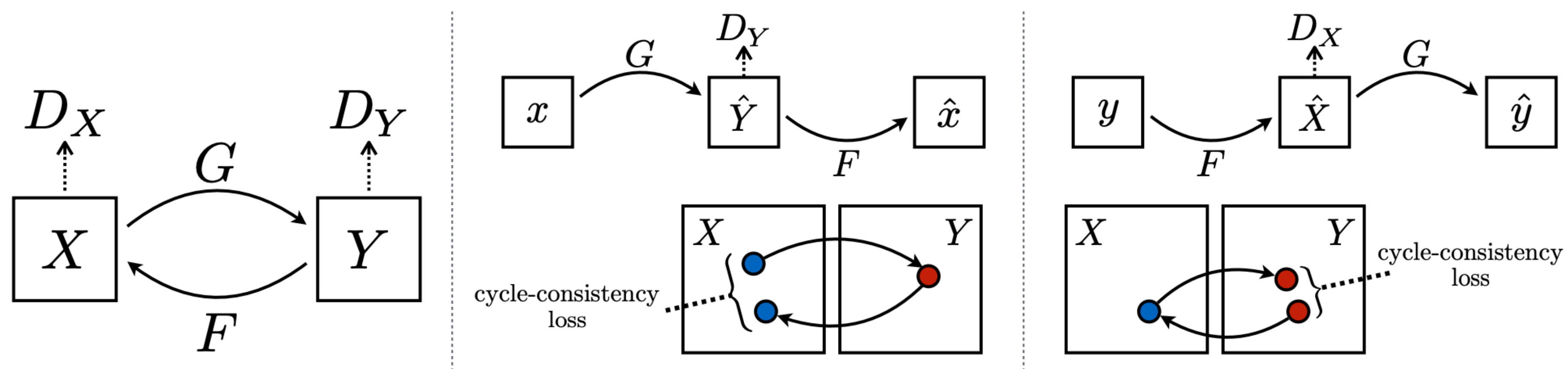


Image to Image Translation — Unpaired (CycleGAN) Zhu et al., ICCV17







(a) (b) (c)

Forward cycle consistency

$$x \rightarrow G(x) \rightarrow F(G(x)) \approx x$$

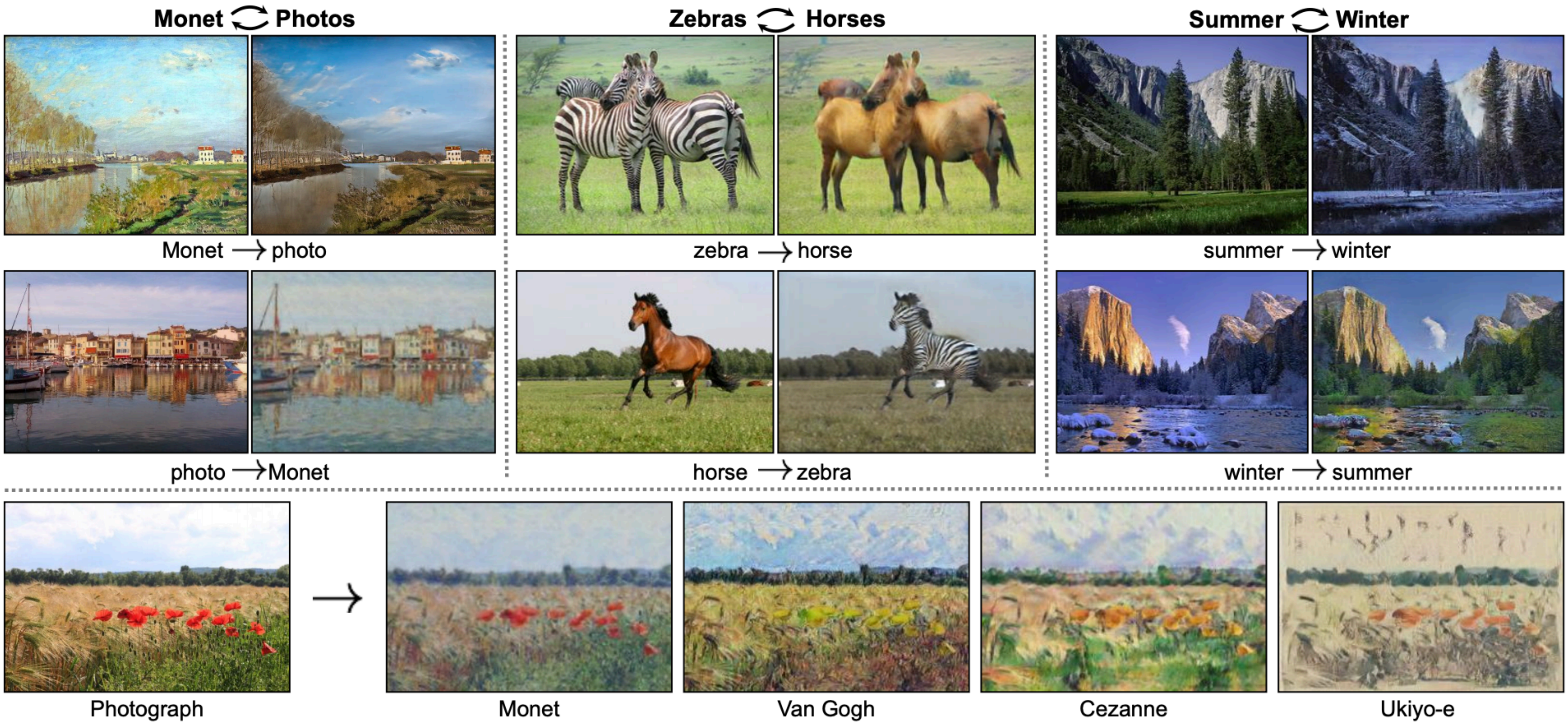
$$||F(G(x)) - x||_1$$

Backward cycle consistency

$$y \rightarrow F(y) \rightarrow G(F(y)) \approx y$$

$$||G(F(y)) - y||_1$$

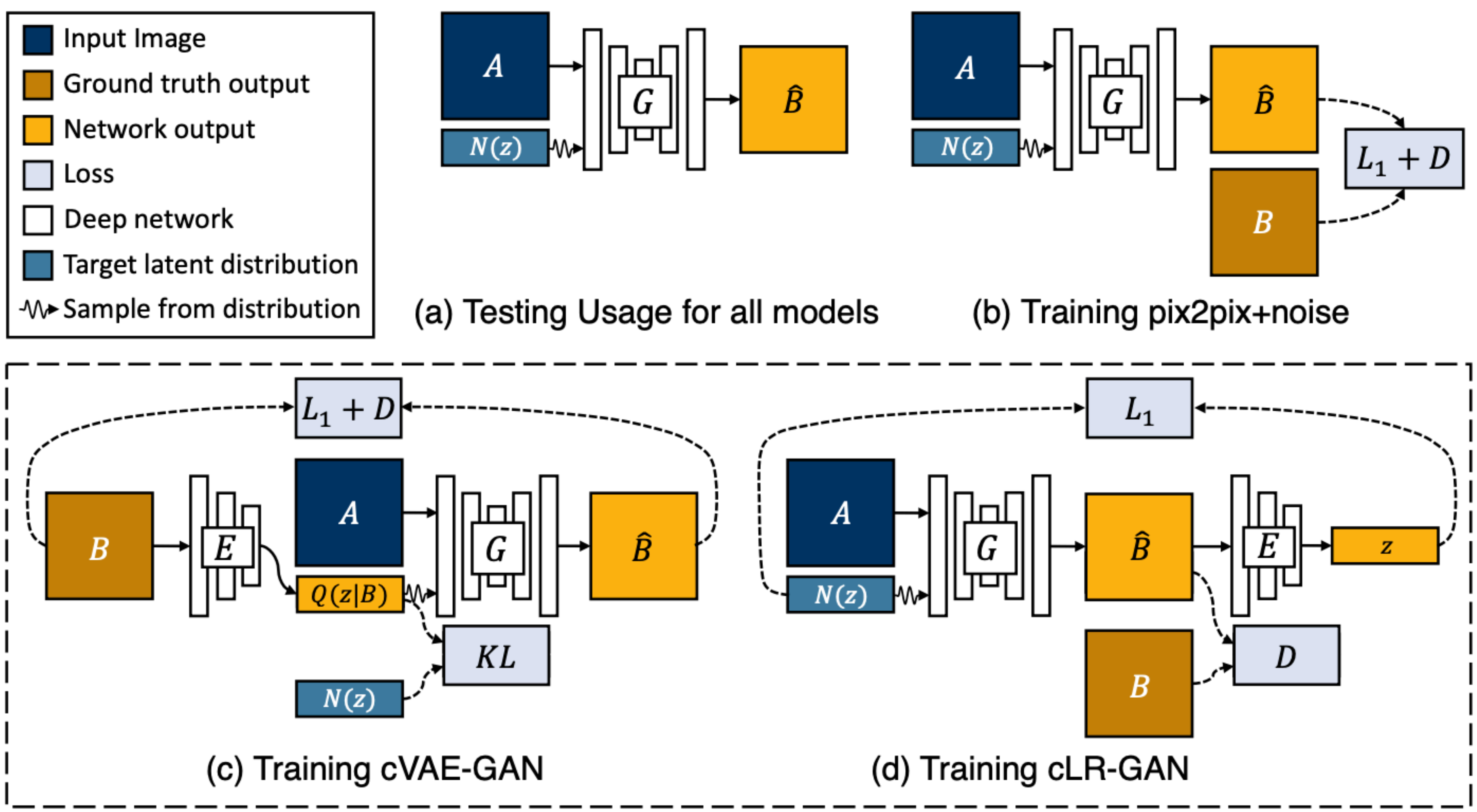
Image to Image Translation — Unpaired (CycleGAN) Zhu et al., ICCV17



Bicycle GAN — Hybrid GAN + VAE

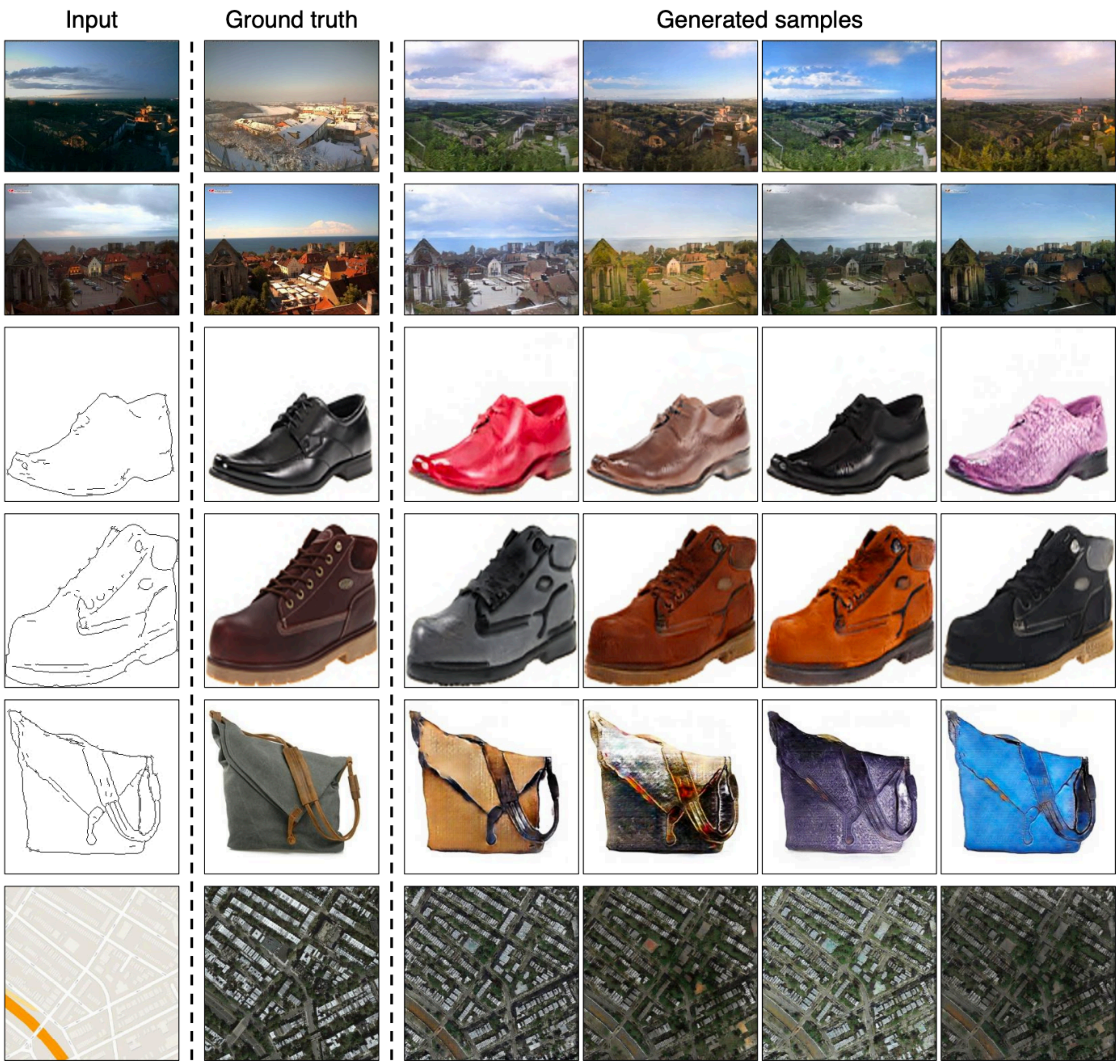
Zhu et al., NeurIPS17

- Handles diversity issue with image to image translation
- Reconstruct the desired output — cVAE type — Use encoder and generator
- Reconstruct the noise



(e) Training BicycleGAN

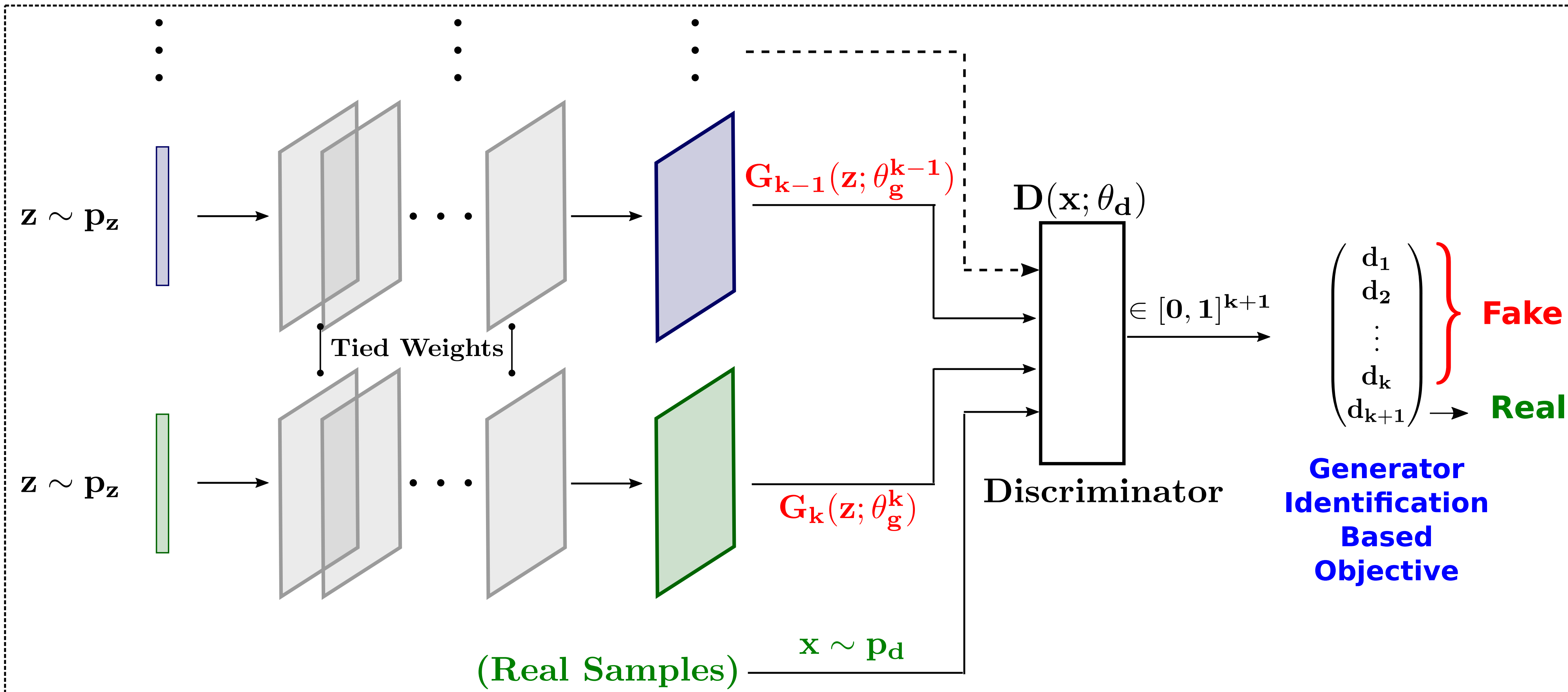
Bicycle GAN
Zhu et al., NeurIPS17



MAD-GAN

Ghosh et al., CVPR18

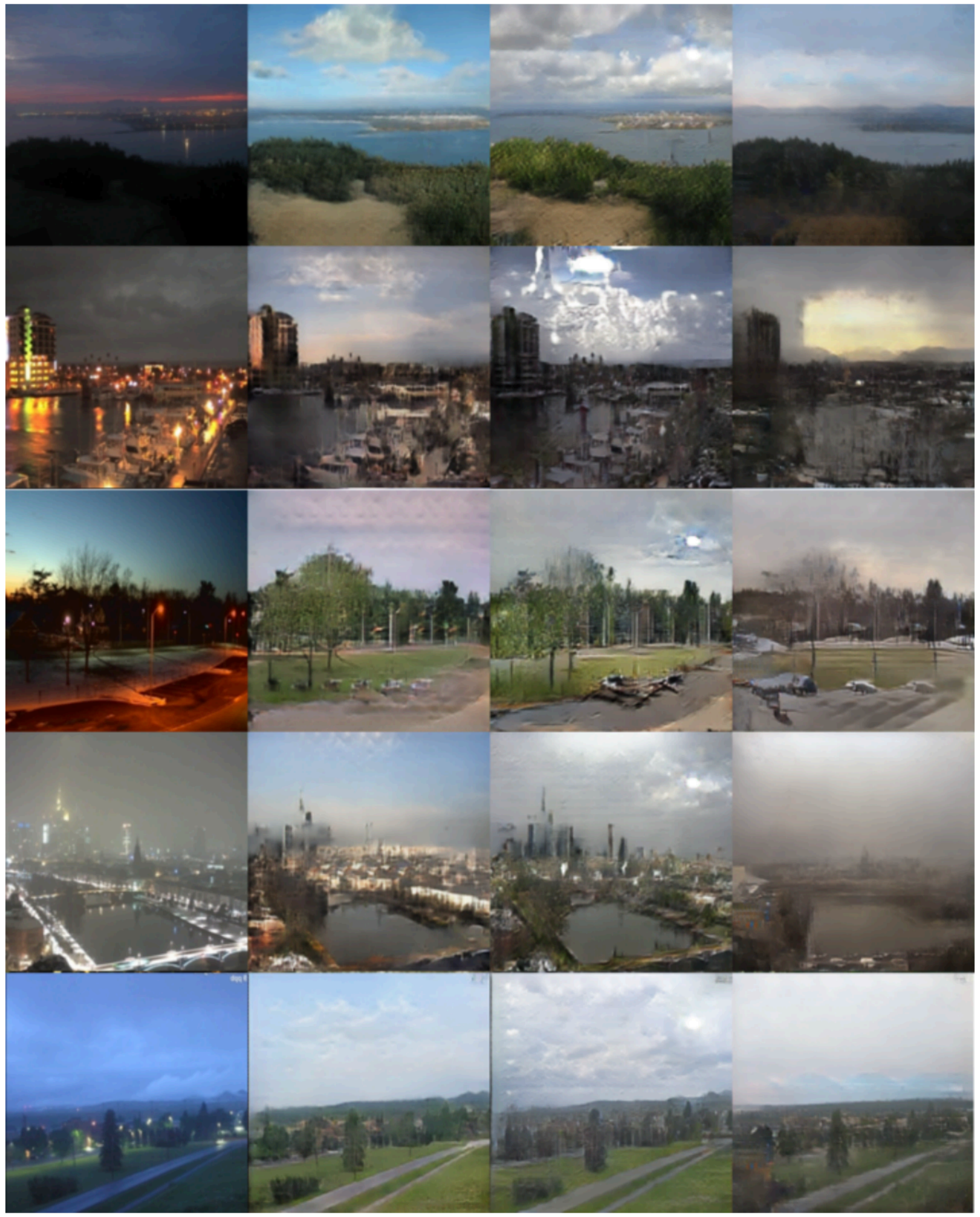
- Handles diversity issue with image to image translation



- Find which generator generated the given `fake' sample
- Disentangles to achieve this task

MAD-GAN

Ghosh et al., CVPR18

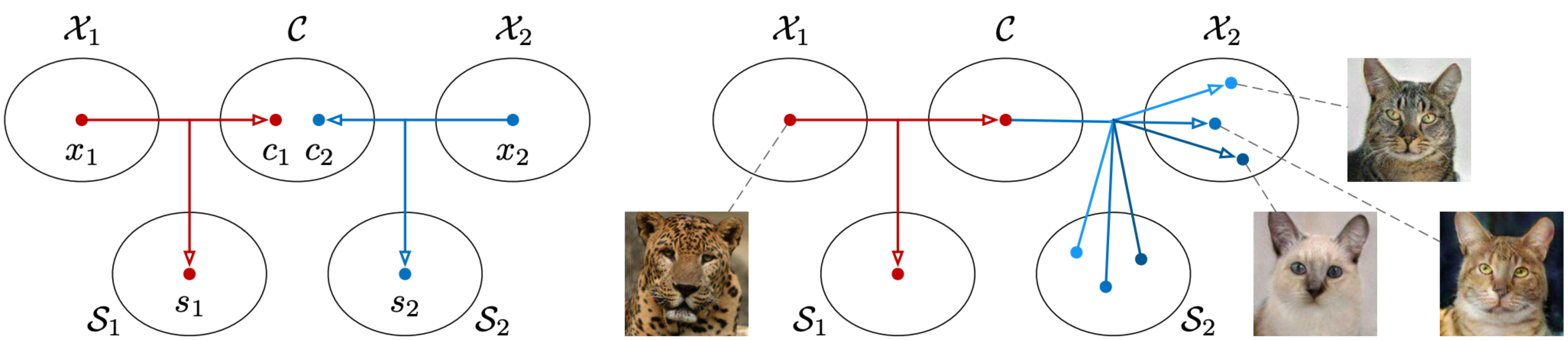


Multimodal Unsupervised Im2Im (MUNIT)

Huang et al., ECCV18

- Image representation is decomposed into two factors
  - Content code — domain invariant
  - Style code — domain specific

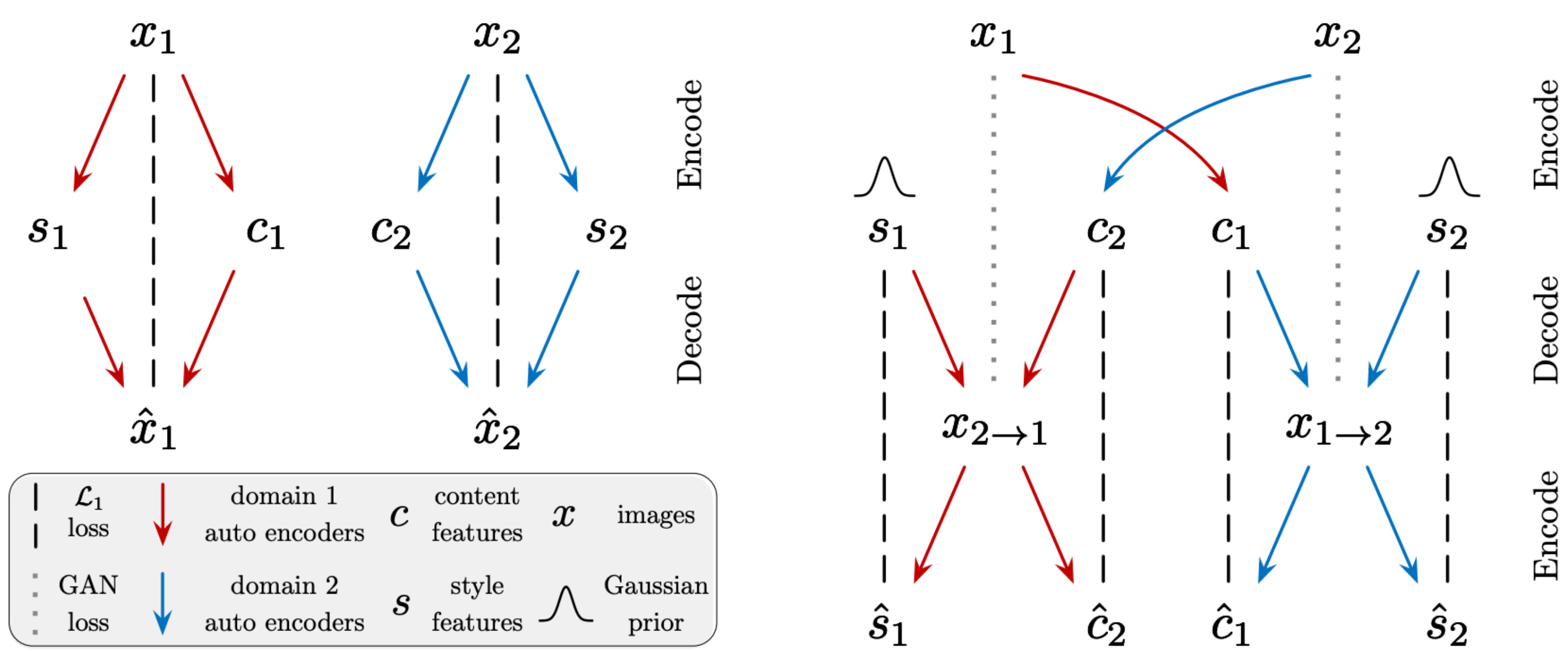
- Domain X1 (Leopard) to Domain X2 (Cat)
  - Combine content code of X1 with randomly sampled style code of X2



(a) Auto-encoding (b) Translation

Multimodal Unsupervised Im2Im (MUNIT)

Huang et al., ECCV18



(a) Within-domain reconstruction (b) Cross-domain translation

- **Image reconstruction.** Given an image sampled from the data distribution, we should be able to reconstruct it after encoding and decoding.

$$\mathcal{L}_{\text{recon}}^{x_1} = \mathbb{E}_{x_1 \sim p(x_1)}[||G_1(E_1^c(x_1), E_1^s(x_1)) - x_1||_1] \tag{1}$$

- **Latent reconstruction.** Given a latent code (style and content) sampled from the latent distribution at translation time, we should be able to reconstruct it after decoding and encoding.

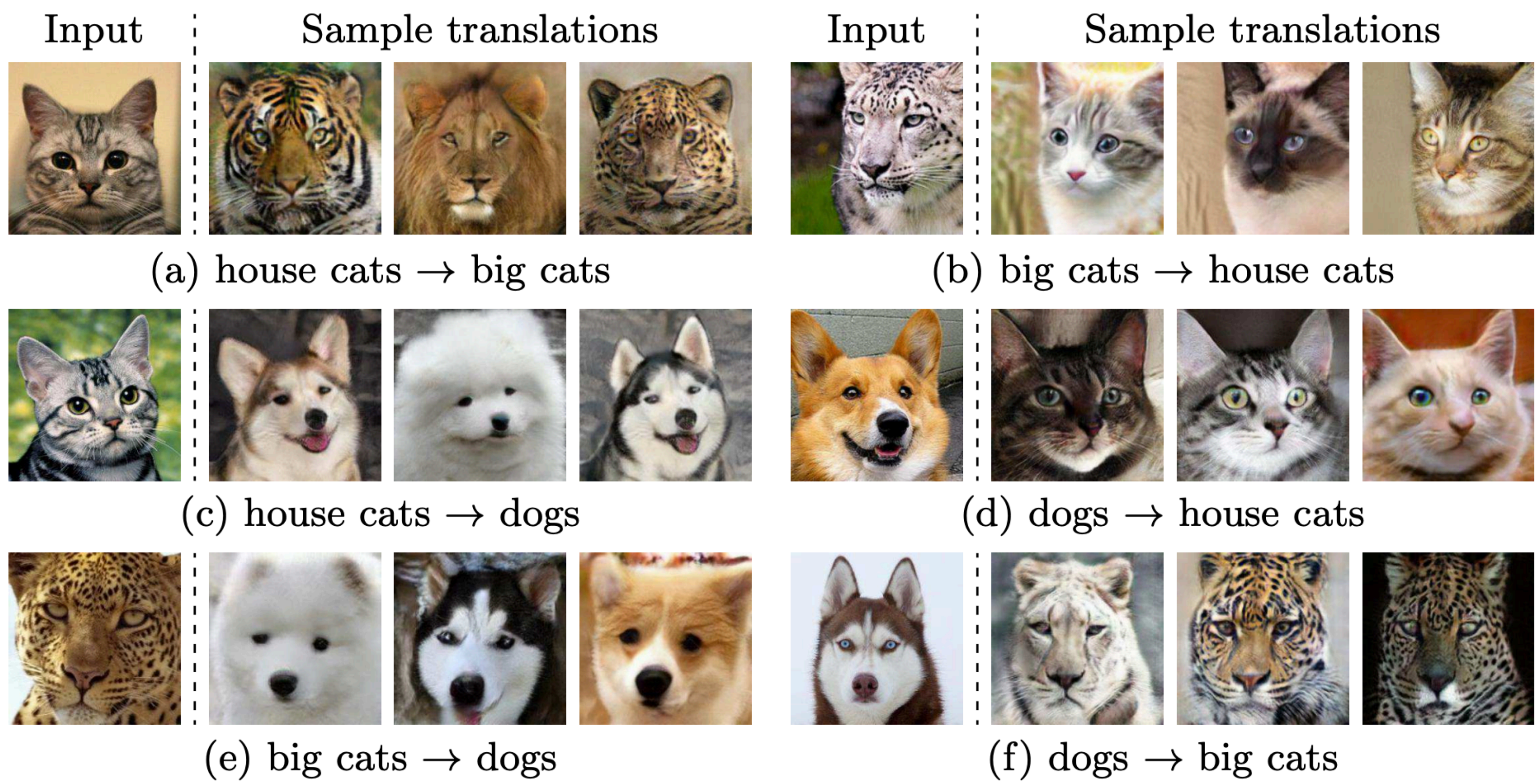
$$\mathcal{L}_{\text{recon}}^{c_1} = \mathbb{E}_{c_1 \sim p(c_1), s_2 \sim q(s_2)}[||E_2^c(G_2(c_1, s_2)) - c_1||_1] \tag{2}$$

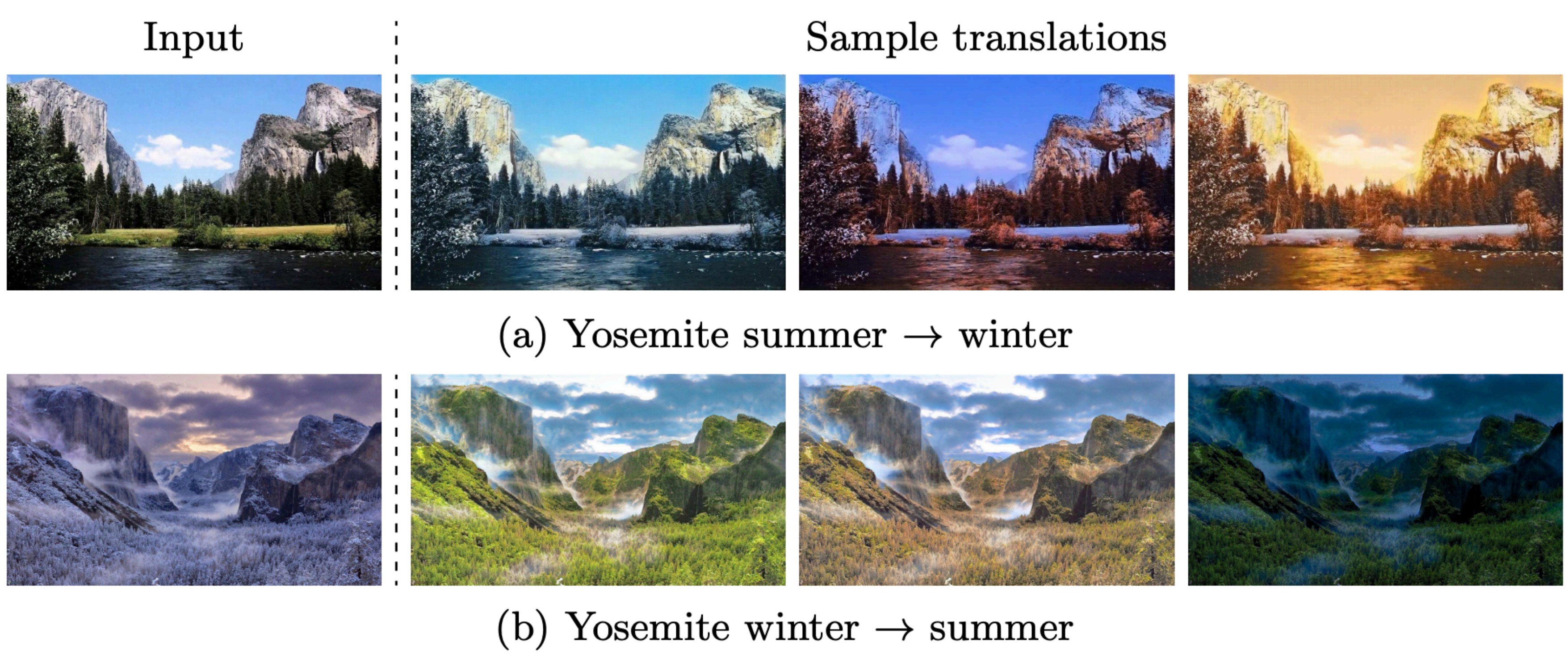
$$\mathcal{L}_{\text{recon}}^{s_2} = \mathbb{E}_{c_1 \sim p(c_1), s_2 \sim q(s_2)}[||E_2^s(G_2(c_1, s_2)) - s_2||_1] \tag{3}$$

where $q(s_2)$ is the prior $\mathcal{N}(0, \mathbf{I})$, $p(c_1)$ is given by $c_1 = E_1^c(x_1)$ and $x_1 \sim p(x_1)$.

Multimodal Unsupervised Im2Im (MUNIT)

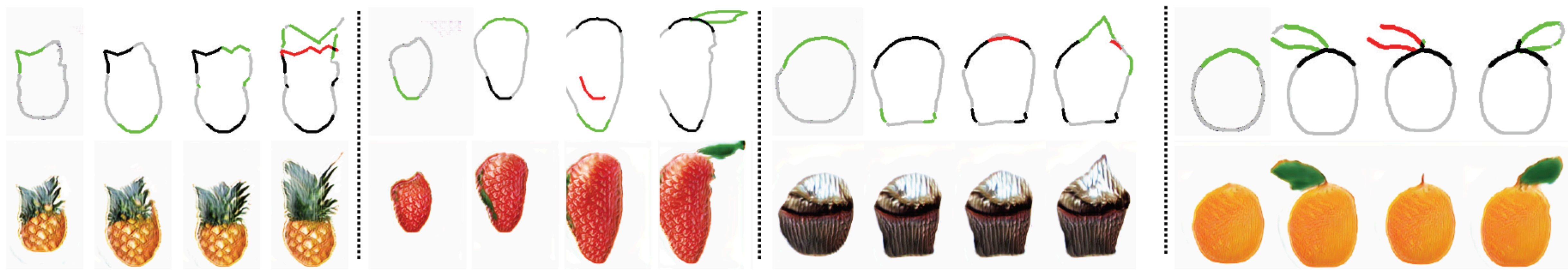
Huang et al., ECCV18





Interactive Sketch & Fill — Work under progress

(A. Ghosh, R. Zhang, P. K. Dokania, A. Efrso, P. H. S. Torr, O. Wang, E. Shechtman)

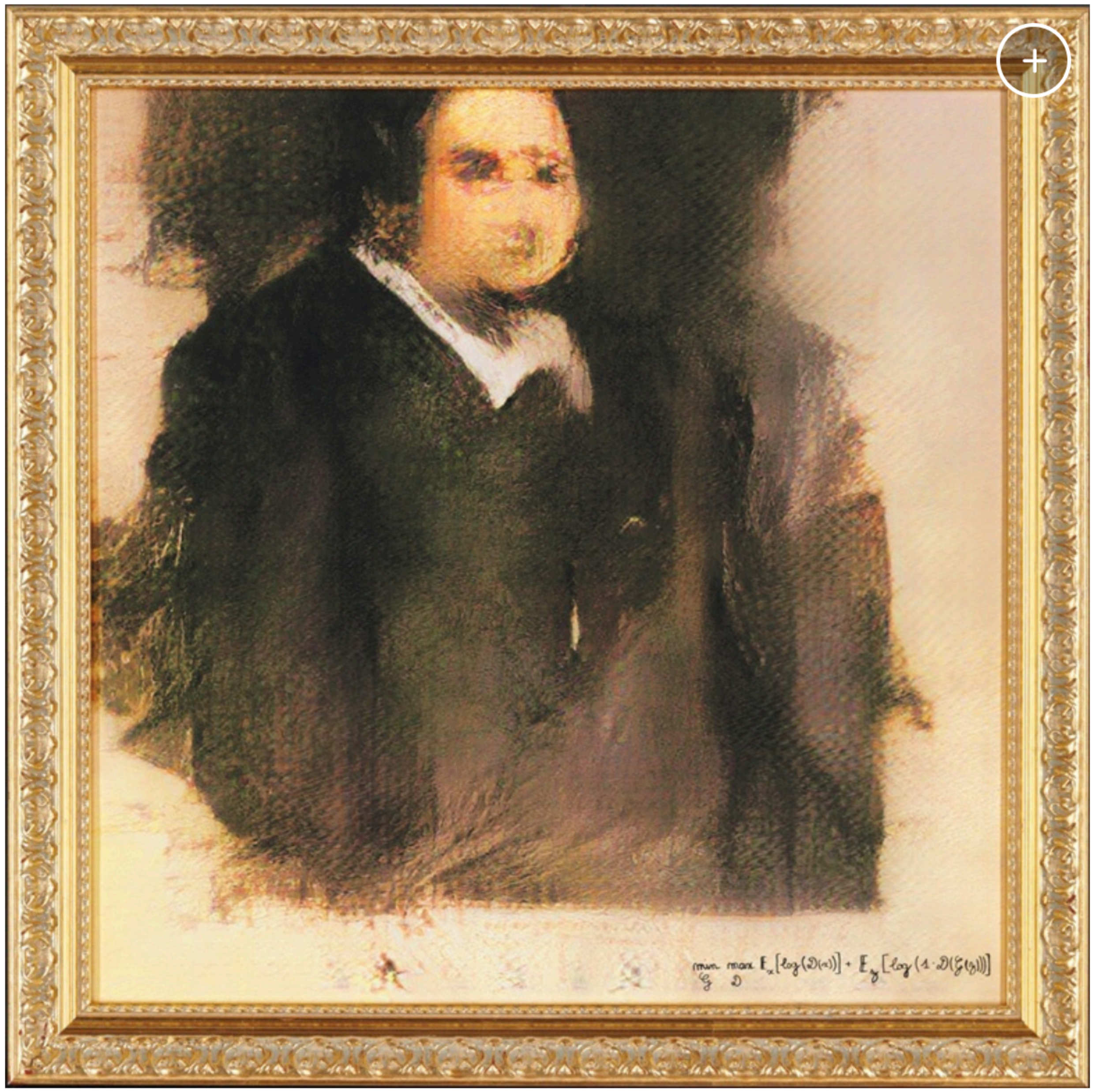


Interactive Sketch & Fill

Class-conditioned Outline-to-Image Translation

GAN Art — Portrait of Edmond de Belamy

GAN Art — Portrait of Edmond de Belamy



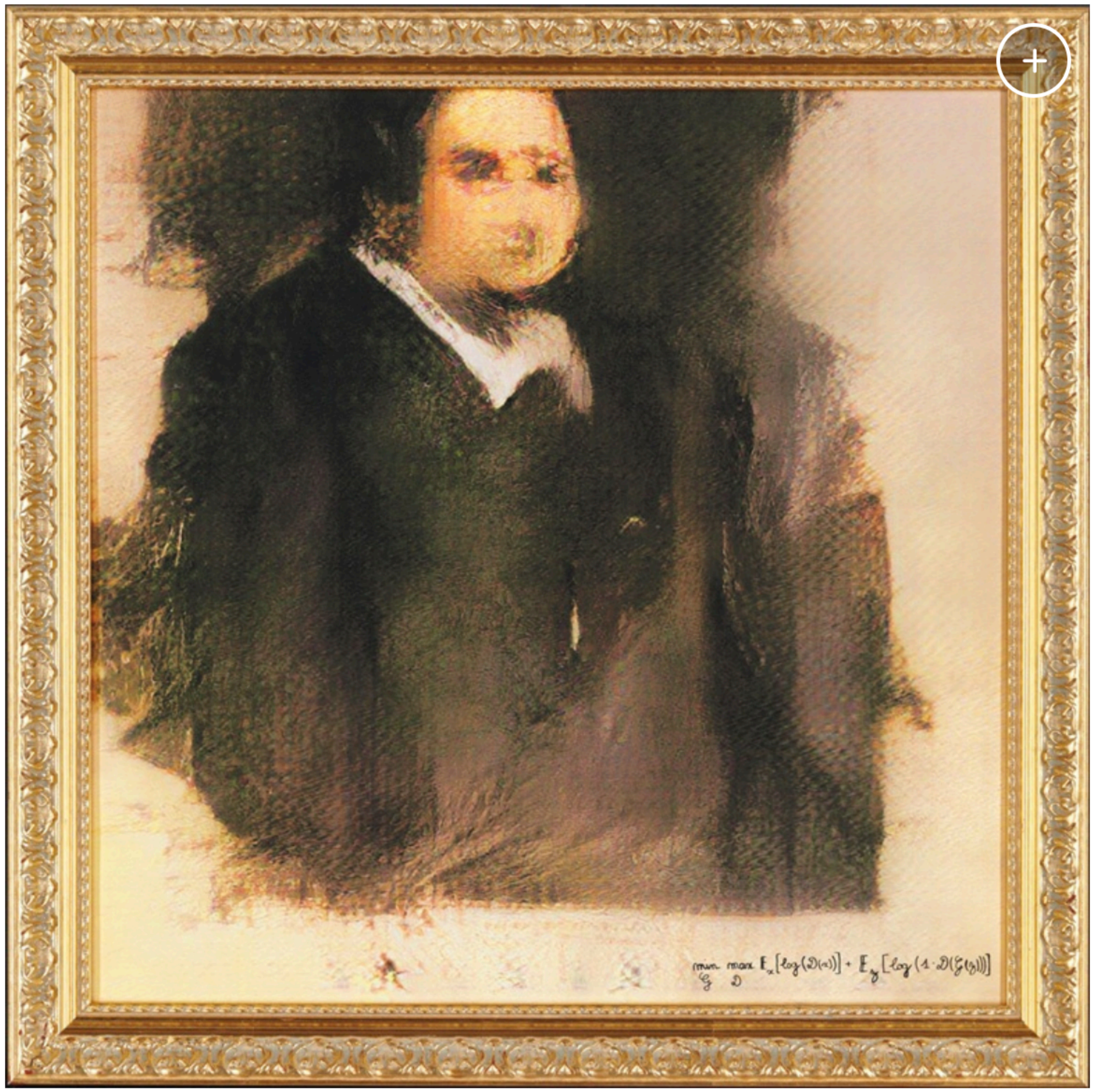
- First AI (GAN) generated art
- Sold (Oct 2018, not sure) for almost 432,500 USD

- Christie's New York

The portrait in its gilt frame depicts a portly gentleman, possibly French and — to judge by his dark frockcoat and plain white collar — a man of the church. The work appears unfinished: the facial features are somewhat indistinct and there are blank areas of canvas. Oddly, the whole composition is displaced slightly to the north-west. A label on the wall states that the sitter is a man named Edmond Belamy, but the giveaway clue as to the origins of the work is the artist's signature at the bottom right. In cursive Gallic script it reads:

$$\min_G \max_D \mathbb{E}_x[\log(D(x))] + \mathbb{E}_z[\log(1 - D(G(z)))]$$

GAN Art — Portrait of Edmond de Belamy



- First AI (GAN) generated art
- Sold (Oct 2018, not sure) for almost 432,500 USD

rejection?

- Christie's New York

The portrait in its gilt frame depicts a portly gentleman, possibly French and — to judge by his dark frockcoat and plain white collar — a man of the church. The work appears unfinished: the facial features are somewhat indistinct and there are blank areas of canvas. Oddly, the whole composition is displaced slightly to the north-west. A label on the wall states that the sitter is a man named Edmond Belamy, but the giveaway clue as to the origins of the work is the artist's signature at the bottom right. In cursive Gallic script it reads:

$$\min_G \max_D \mathbb{E}_x[\log(D(x))] + \mathbb{E}_z[\log(1 - D(G(z)))]$$

Thank You