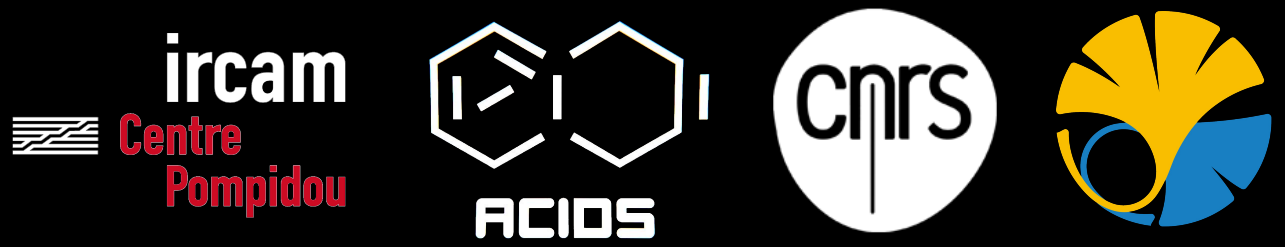


Variational AEs and Normalizing flows

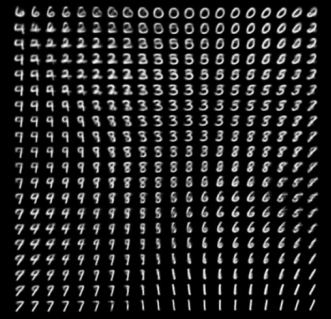
Creative Machine Learning - Course 08

Pr. Philippe Esling
esling@ircam.fr

Brief history of AI

Families of generative models that we will learn

1

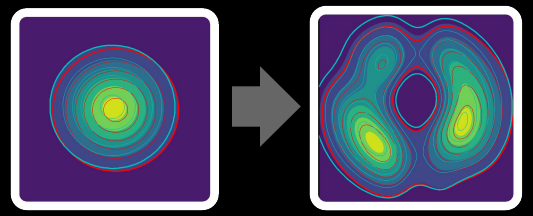


Variational Auto-Encoder (VAE)

Random variables, distributions, independence

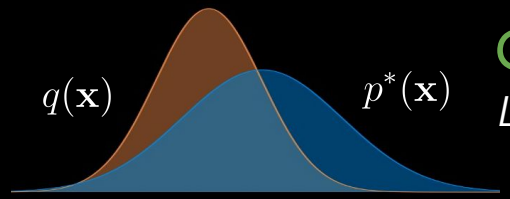
Normalizing flows

Bayes' theorem, likelihood, conjugate priors



2

3



Generative Adversarial Network (GAN)

Latent variables, probability graphs

Diffusion models

Latent variables, probability graphs

4

2015 - *Generative model*

First wave of interest in generating data
Led to current model craze (VAEs, GANs, Diffusion)

2012 onwards Deep learning era

Supervised learning

Typical machine learning tasks

Until now, we mostly dealt with **supervised** problems

- *Classification, regression*
- *Implies that we have labels*

Based on *labeled* data, assign class to *unlabeled data*

- *Multiple approaches to solve this task*
- *We have also seen the probabilistic formulation*

Probabilistic formulation

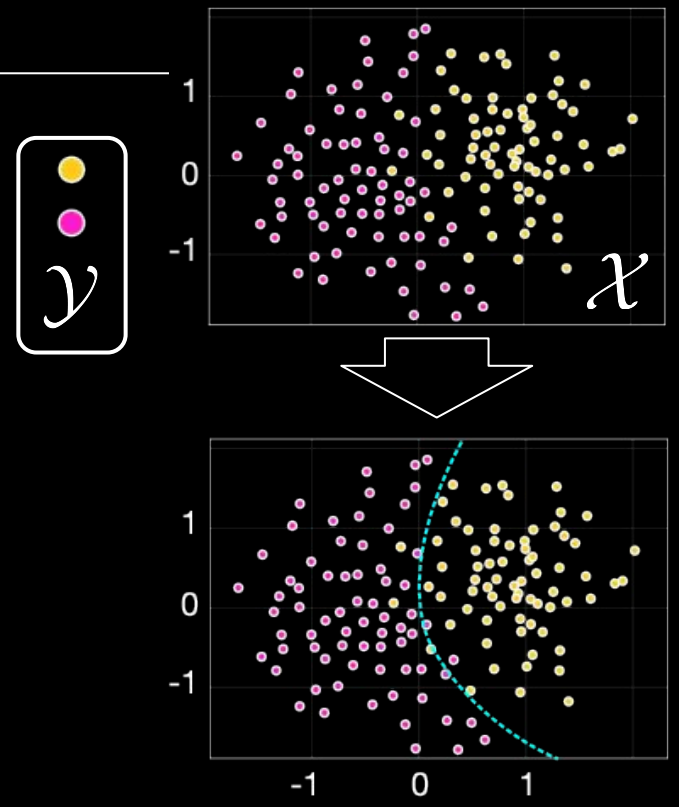
We have data $\mathbf{x} \in \mathbb{R}^n$ and labels $\mathbf{y} \in \mathbb{R}^m$

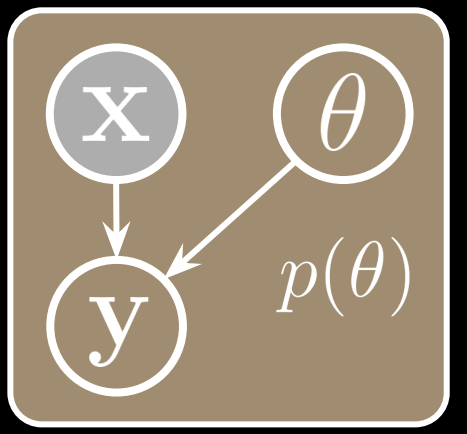
Supervised learning tries to model $p(\mathbf{y}|\mathbf{x})$

We define a parametric model $p_\theta(\mathbf{y}|\mathbf{x})$

Optimization might be seen as solving for

$$p(\mathbf{y}|\mathbf{x}) = \int p(\mathbf{y}|\mathbf{x}, \theta) d\theta$$

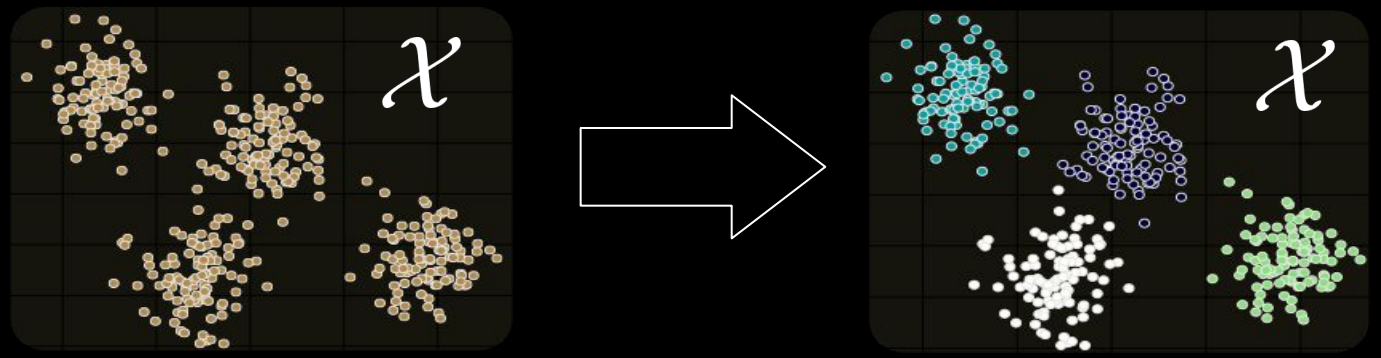




Graphical model

Unsupervised learning

We need to **add information** to our problem, but only have $\mathbf{x} \in \mathbb{R}^n$



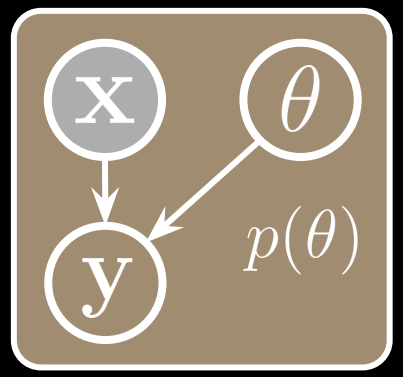
*Structure is clearly defined by the **cluster identity***

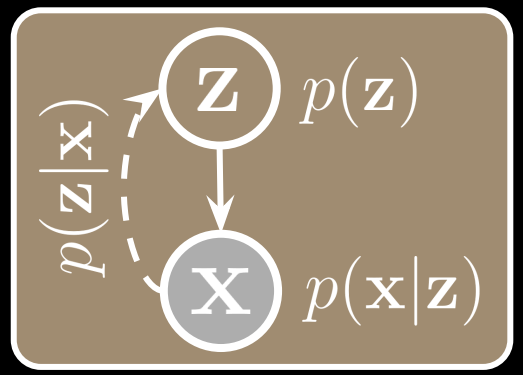
Introducing the need for **latent variables** $\mathbf{z} \in \mathbb{R}^z$

Allows to clearly define the *hidden information*

Can be seen as the **hidden factors that generated the data**

Supervised





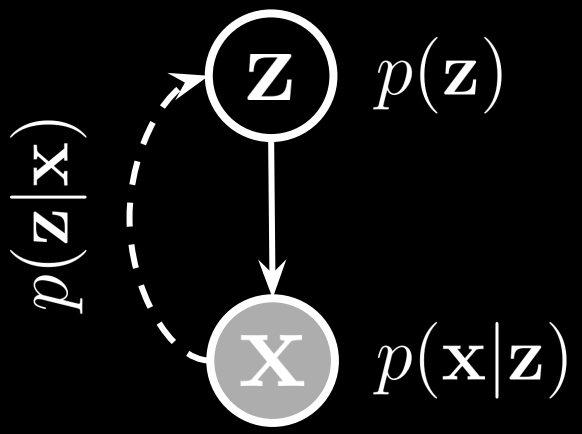
Latent (unsupervised)

Latent variables

We are interested in the **generative problem** $\mathbf{x} \sim p(\mathbf{x})$

This problem is highly complex (distribution)

So we introduce l**atent variables** $\mathbf{z} \in \mathbb{R}^z \lll \mathbb{R}^x$



Happy, female

Neutral, male

$$p(\mathbf{x}) = \int_{\mathbb{R}} p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

Understanding latent variables

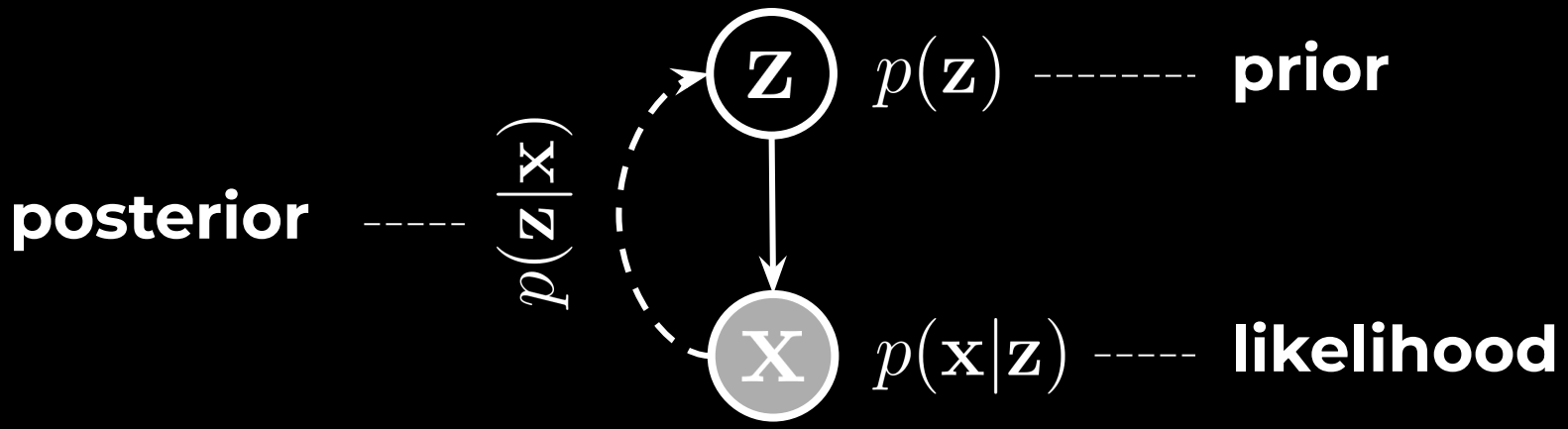
Variables that led to generate the data

Hidden variation factors in the data structure

Variables that simplify our optimization task

Variational inference

We take back our inference problem $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$



$$p(\mathbf{x}) = \int_{\mathbb{R}} p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

How can we transform this intractable integral

Optimize a simpler and tractable distribution $q(\mathbf{z})$?

Variational inference

Starting from intractable integral $p(\mathbf{x}) = \int_{\mathbb{R}} p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$

Goal of Variational Inference

Approximate this by using **optimization** instead of **derivation**.

Select a family of *parametric* distributions $q_\phi \in \mathcal{Q}$

Optimize the parameters ϕ in order to solve

$$\underset{\phi}{\operatorname{argmin}}\ \mathcal{D}_{KL}\left[q_\phi \| p\right]$$

Modeling the joint probability

Deriving the variational objective

$$\log p(\mathbf{x}) = \log \int p(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$$

$$\log p(\mathbf{x}) = \log \int p(\mathbf{x}|\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}q(\mathbf{z}|\mathbf{x})d\mathbf{z}$$

Introducing $q_\phi \in \mathcal{Q}$

$$\geq \int q(\mathbf{z}|\mathbf{x}) \log \left(p(\mathbf{x}|\mathbf{z})\frac{p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})} \right) d\mathbf{z}$$

Jensen's inequality

$f\left(\mathbb{E}_{q_\phi(\mathbf{z})}[\mathbf{x}]\right) \geq \mathbb{E}_{q_\phi(\mathbf{z})}[f(\mathbf{x})]$

$$\geq \int q(\mathbf{z}|\mathbf{x}) \log p(\mathbf{x}|\mathbf{z})d\mathbf{z} - \int q(\mathbf{z}|\mathbf{x}) \log \left(\frac{p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})} \right) d\mathbf{z}$$

$\mathbb{E}_{q_\phi(\mathbf{z})}[\log p_\theta(\mathbf{x}|\mathbf{z})]$ $\quad \mathcal{D}_{KL}[q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})]$

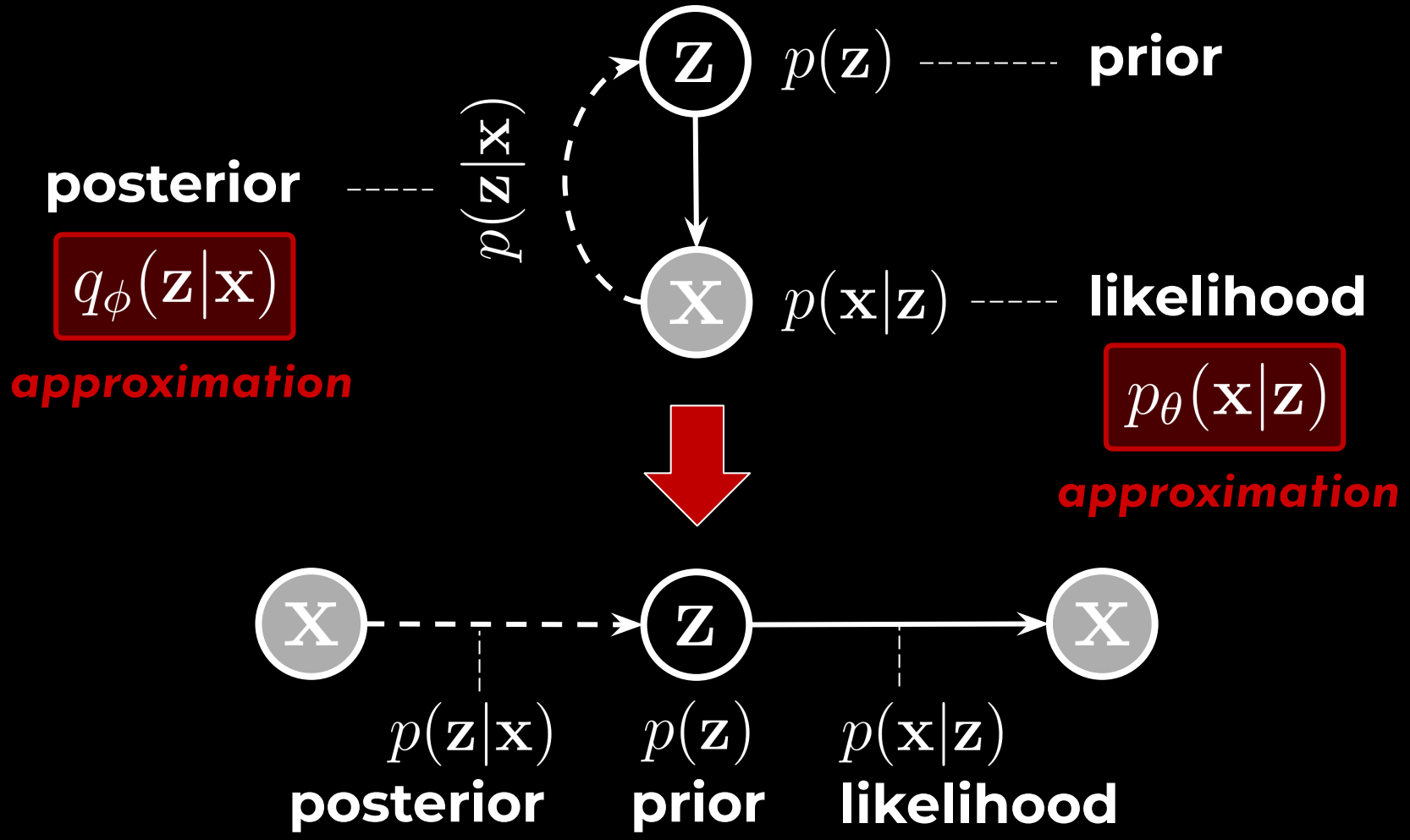
$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_\phi(\mathbf{z})}[\log p_\theta(\mathbf{x}|\mathbf{z})] - \mathcal{D}_{KL}[q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})]$$

reconstruction

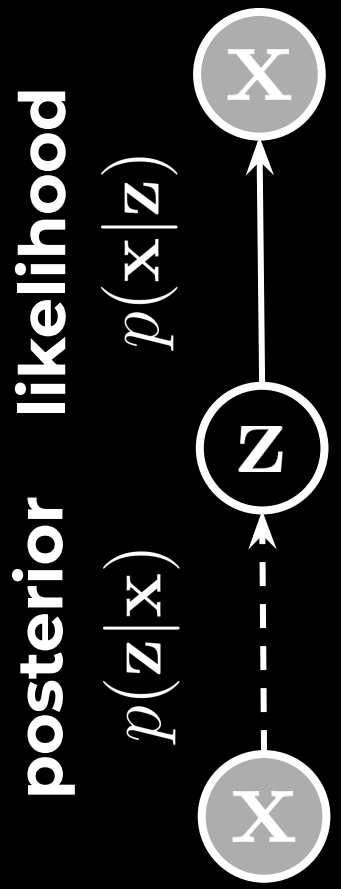
regularization

Variational inference

We take back our inference problem $p(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}|\mathbf{z})p(\mathbf{z})$

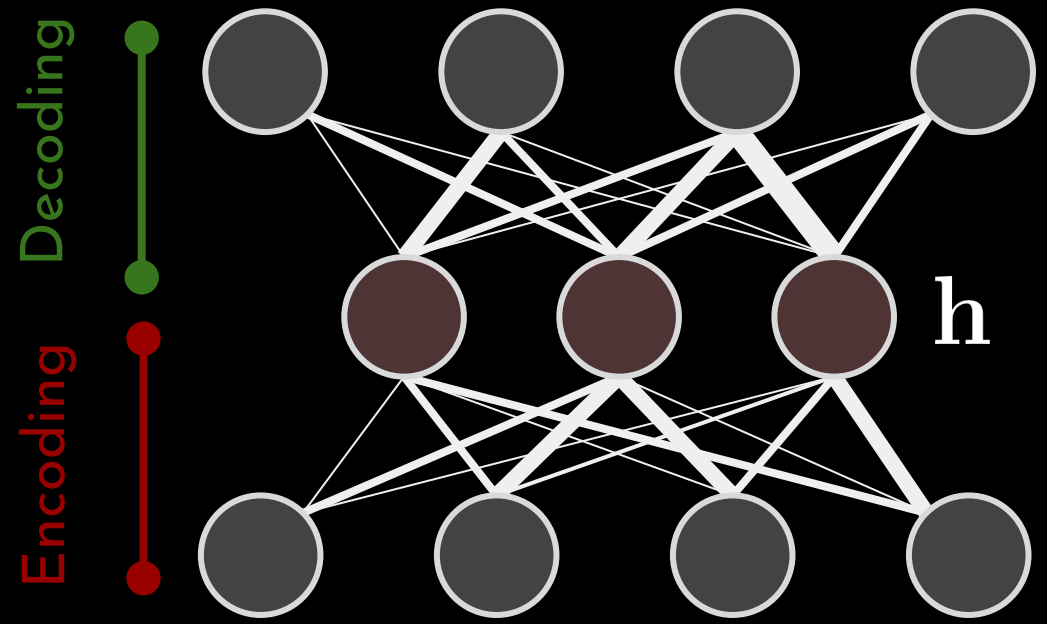


Auto-encoders



$\approx$

Probability structure is akin to auto-encoders



$\mathbf{y}$

$d : \mathbb{R}^{d_h} \rightarrow \mathbb{R}^{d_x}$

$\min \ [\mathcal{L}(\mathbf{x}, \mathbf{y})]$

$e : \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{d_h}$

$\mathbf{x}$

Introducing **variational auto-encoders**

Parametrize the *posterior* and *likelihood* with **deep networks**

Introduce two submodels: an ***encoder*** and a ***decoder***

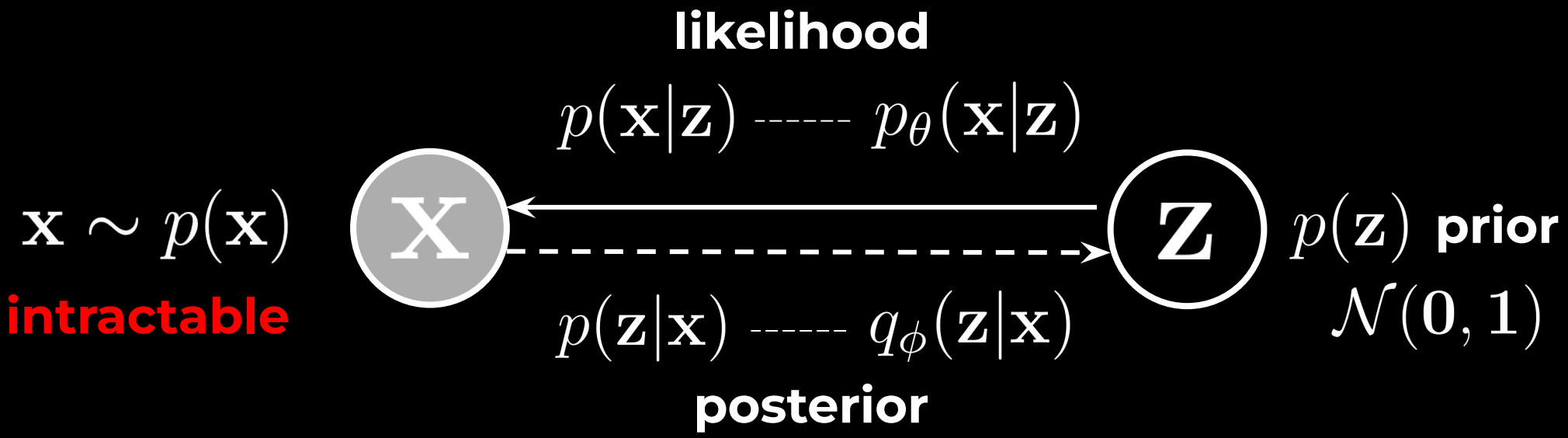
Mixing variational inference and deep auto-encoders

Variational inference

Summarizing the specificities of the approach

$p(\mathbf{x})$ is usually intractable due to the complexity of real-life data

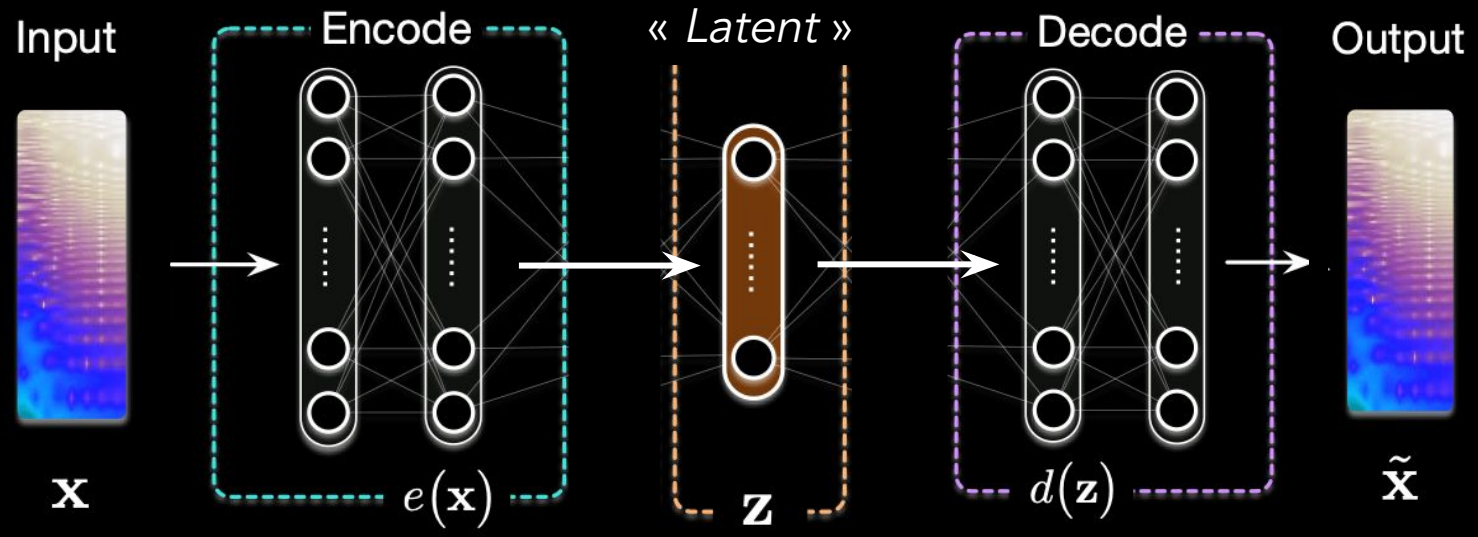
we introduce a high-level *latent variable* $\mathbf{z}$ **(variations in a dataset)**



Parametrize the *posterior* and *likelihood* with **deep networks**

Introduce two submodels: an ***encoder*** and a ***decoder***

Auto-encoding



Issues

Defines a **deterministic** mapping $\tilde{\mathbf{x}} = d\left(e\left(\mathbf{x}\right)\right)$

No theoretical properties on the latent $\mathbf{z}$

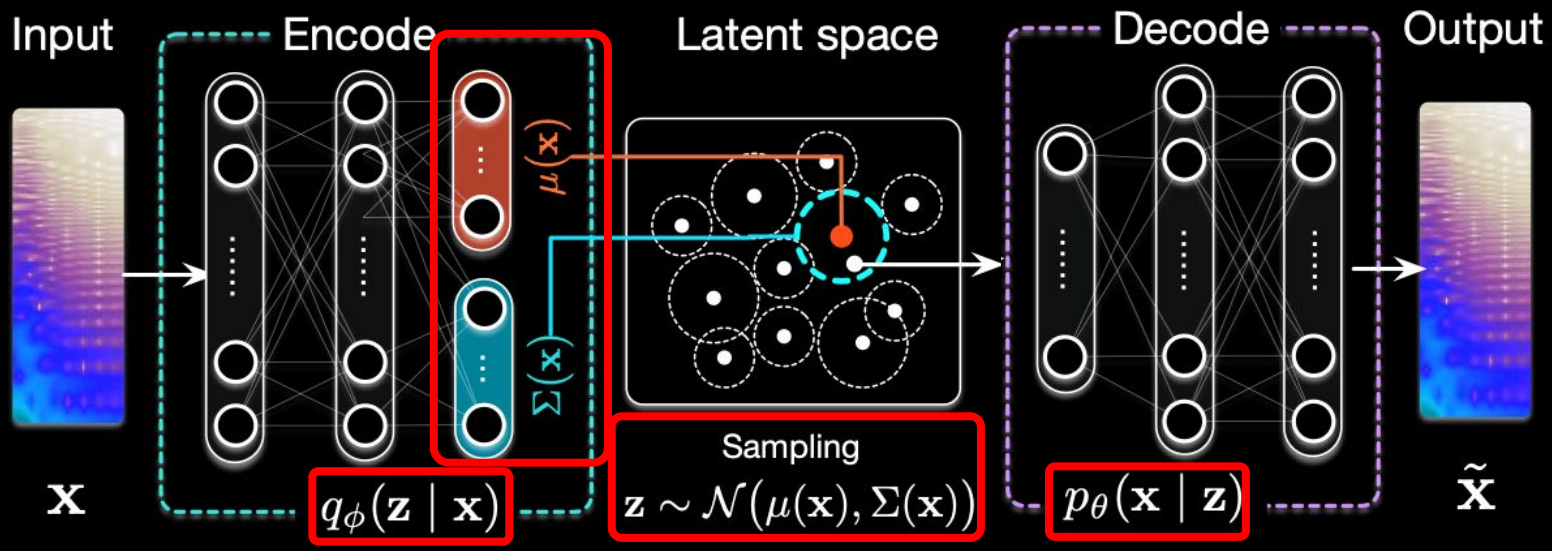
Very **limited generalization** abilities (does not model $p(\mathbf{x}, \mathbf{z})$)

Goal

Modeling the complete distribution $p(\mathbf{x}, \mathbf{z})$

Introducing variational inference inside the structure

Variational Auto-Encoders (VAEs)



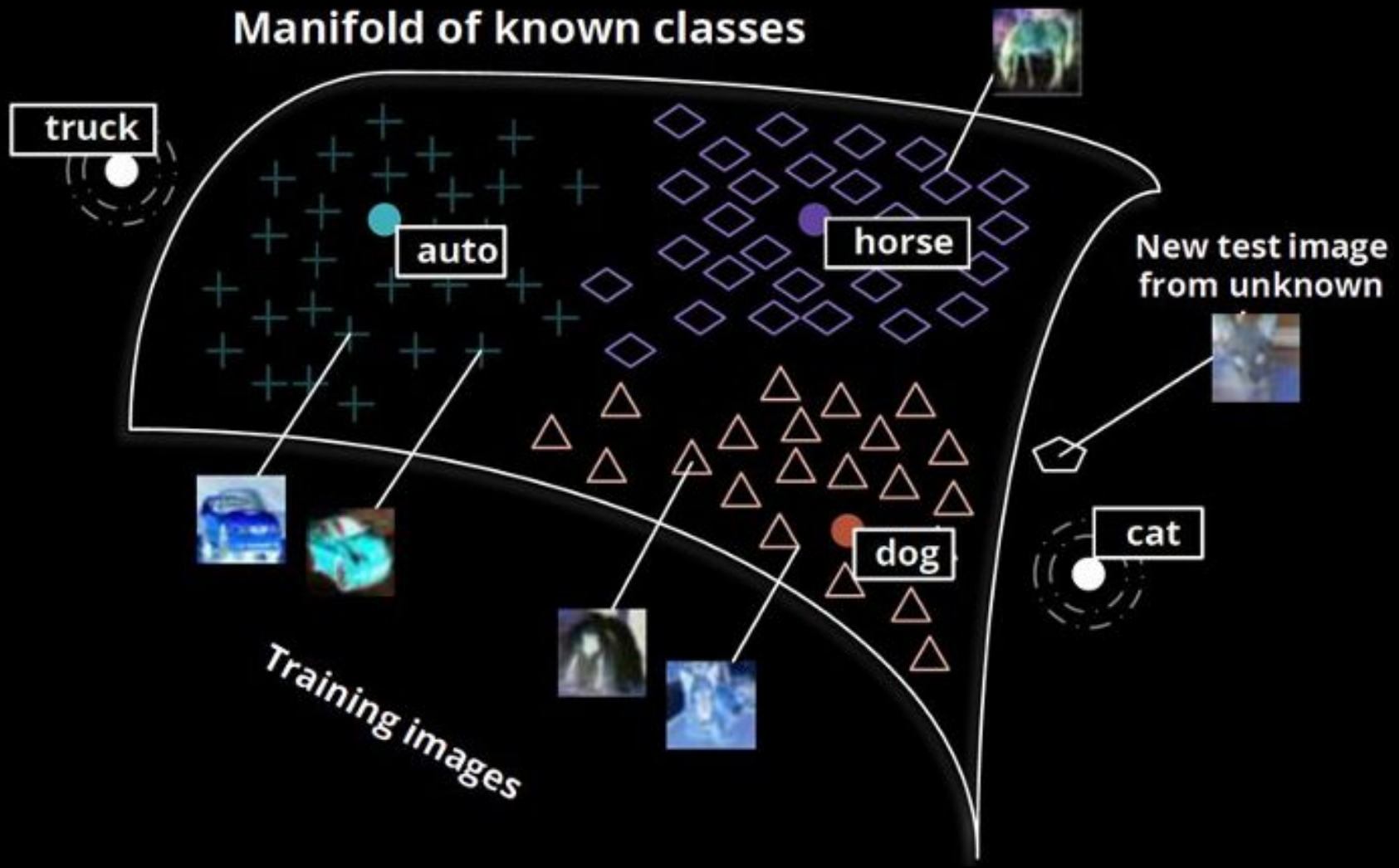
Goal | Minimize approximation error $q^*(\mathbf{z}) = \underset{q_\phi(\mathbf{z}) \in \mathcal{Q}}{\operatorname{argmin}} \mathcal{D}_{KL}\left[q_\phi(\mathbf{z}) \parallel p(\mathbf{z}|\mathbf{x})\right]$

Encode input using $q_\phi(\mathbf{z}|\mathbf{x})$ to find **mean** and **variance** in the space
Sample from this gaussian distribution to obtain $\mathbf{z} \sim \mathcal{N}(\mu(\mathbf{x}), \Sigma(\mathbf{x}))$
Decode the corresponding position using $p_\theta(\mathbf{x}|\mathbf{z})$

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_\phi(\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \beta \cdot \mathcal{D}_{KL}\left[q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})\right]$$

Manifold hypothesis

The **manifold hypothesis**

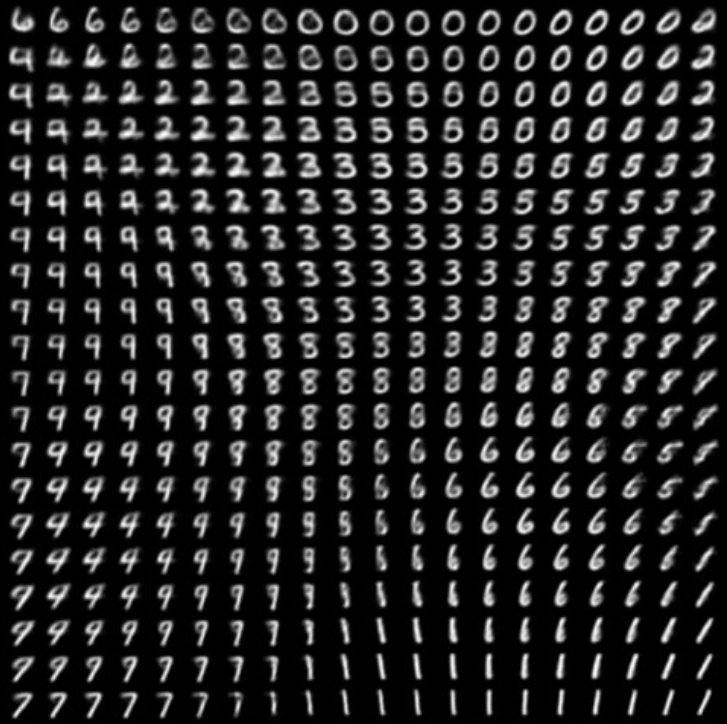


Variational Auto-Encoders

VAEs naturally **uncover the underlying manifold**

Results from the original paper

Faces



Digit

Interpolation

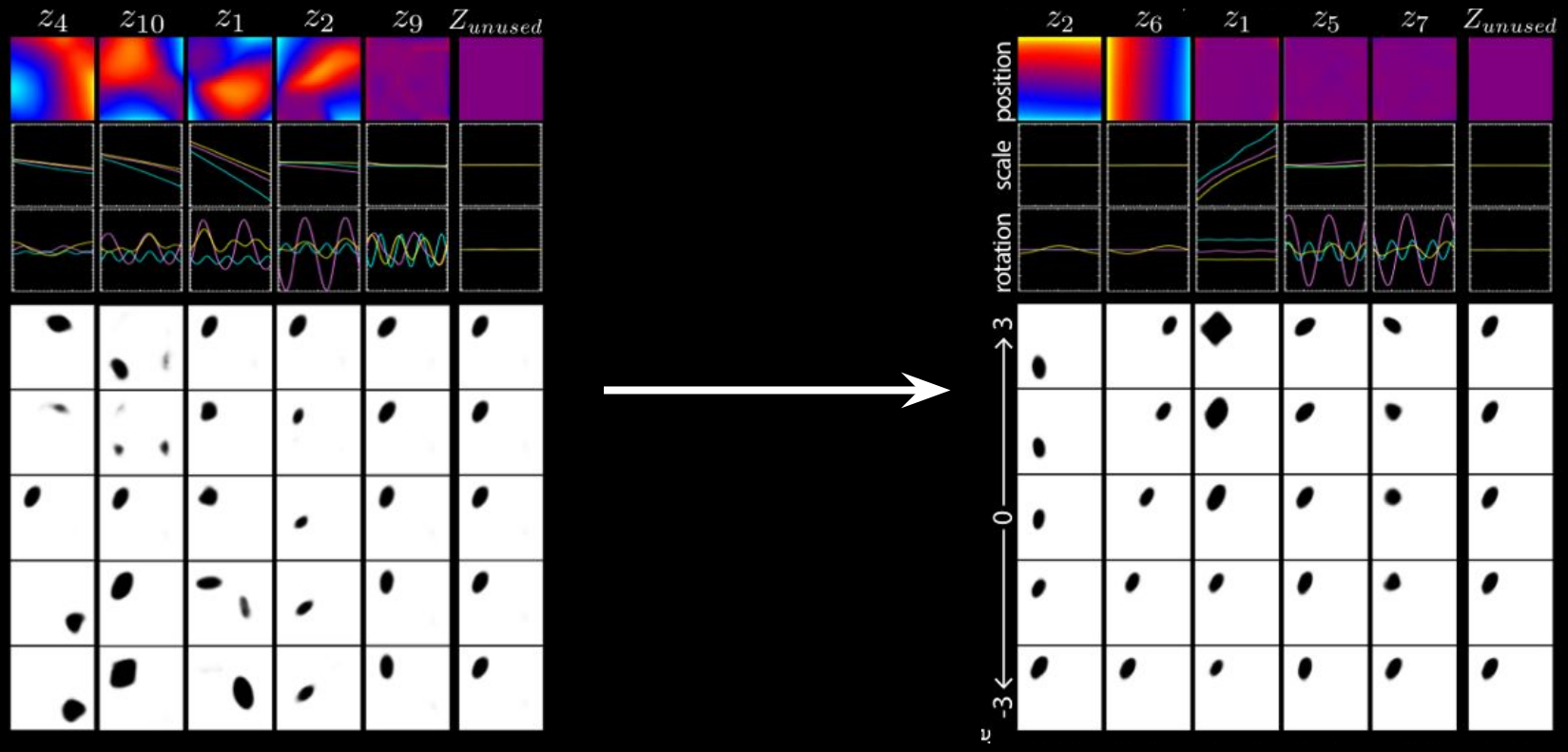
[Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. ICLR 2015, 1050, 1.]

Unsupervised disentanglement with VAE

Find the underlying factors of variation in data

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathbb{E}_{q_\phi(\mathbf{z})}\left[\log p_\theta(\mathbf{x}|\mathbf{z})\right] - \beta \cdot \mathcal{D}_{KL}\left[q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})\right]$$

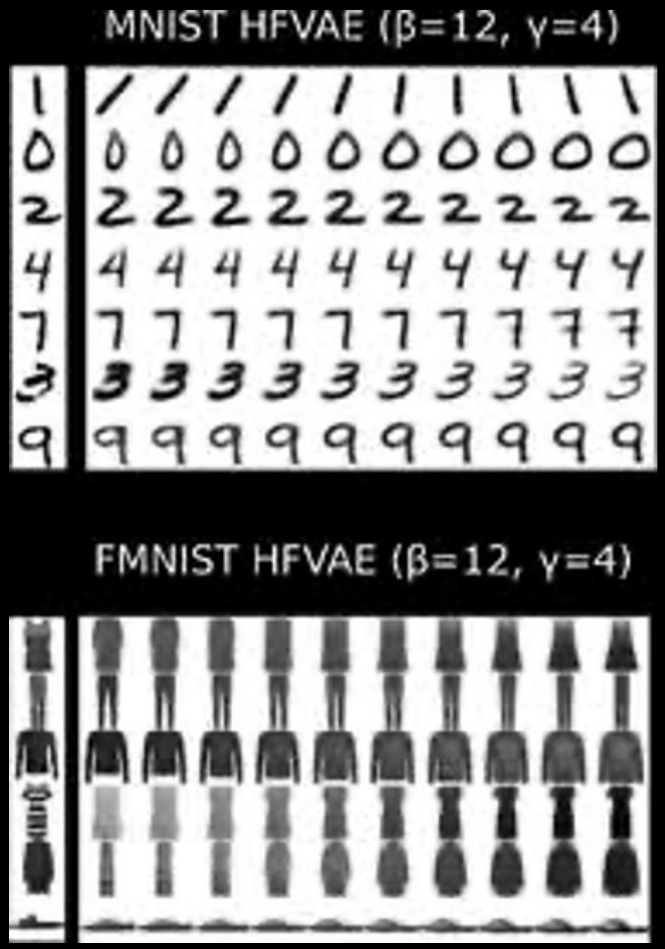
Increase **importance of the regularization** (latent space organization)



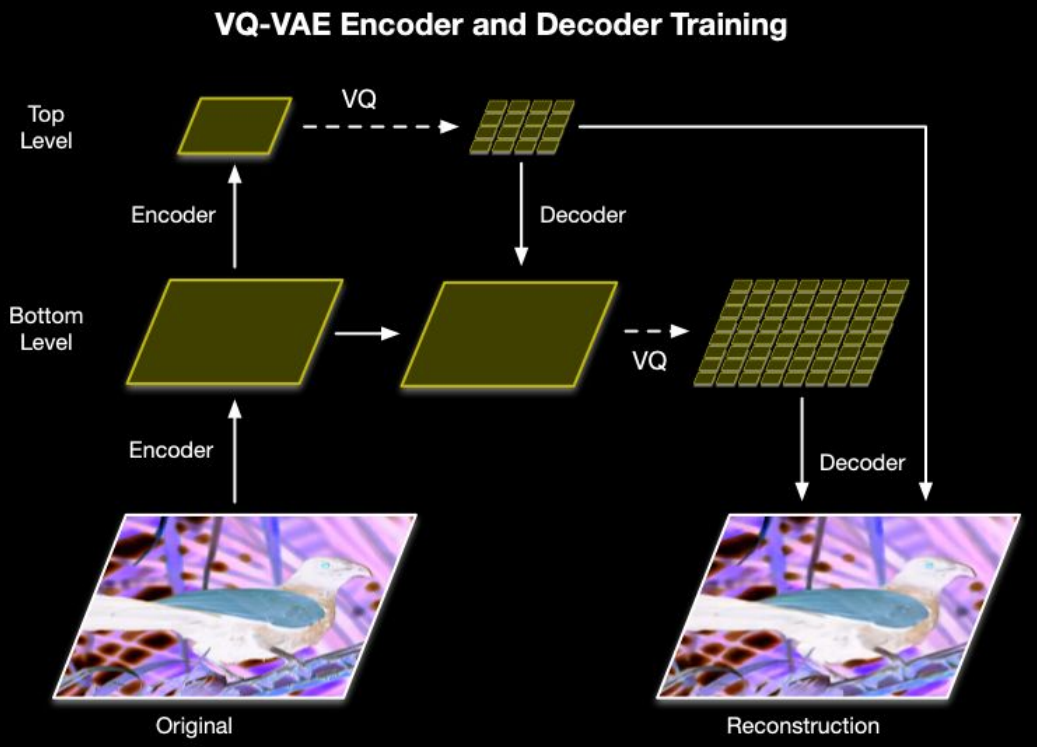
Higgins, I. et al. (2016). beta-vae: Learning basic visual concepts with a constrained variational framework.

Unsupervised disentanglement with VAE

Disentanglement as control

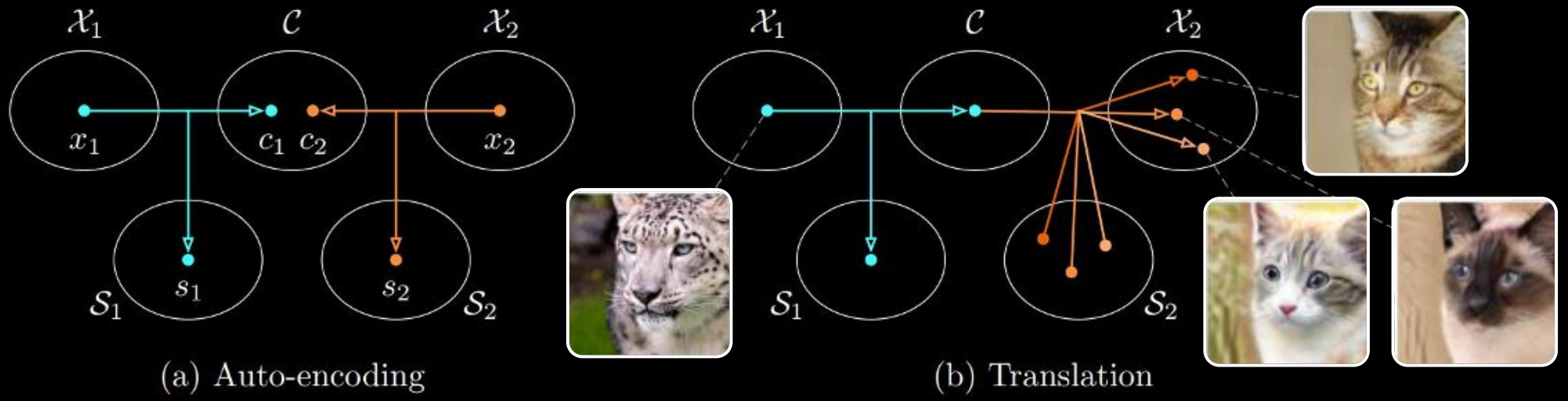


Vector-Quantized VAE



[Razavi, A., van den Oord, A., & Vinyals, O. (2019). Generating diverse high-fidelity images with vq-vae-2. In *Advances in Neural Information Processing Systems* (pp. 14866-14876).]

Unsupervised image transfer with VAE



cats → big cats

dogs → pantheras

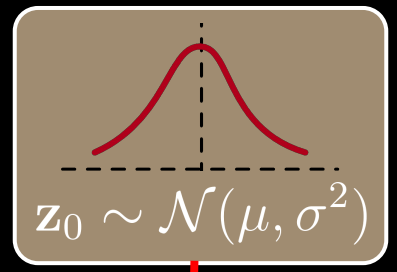
cats → dogs

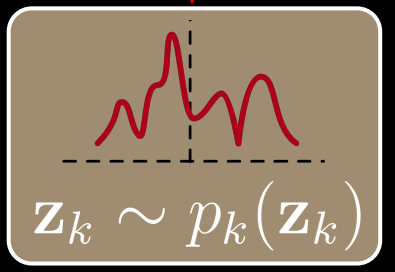
dogs → cats

[Huang, et al. (2018). Multimodal unsupervised image-to-image translation. In *Proceedings of ECCV* (pp. 172-189).]

Normalizing flows

The almighty Gaussian





Real data ?

Probabilistic inference deals with simple distributions

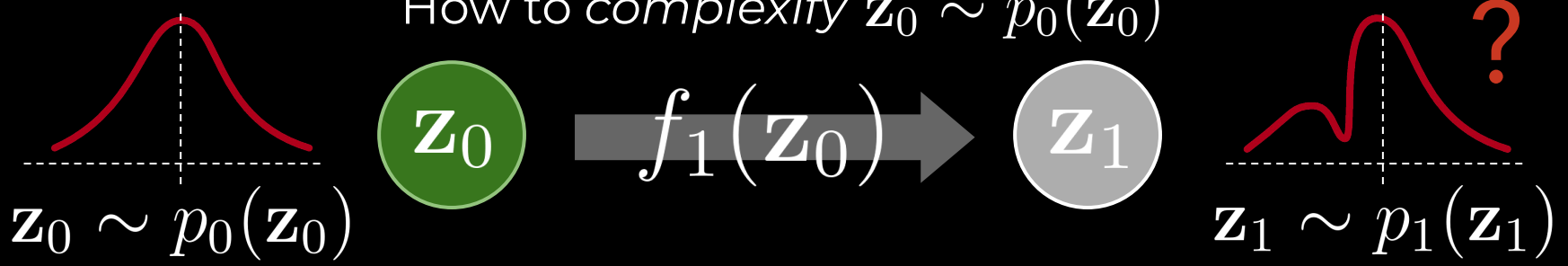
- Provide easier analytical solutions
- Implicit assumption of a simple explanation (capacity)

Issues

- Too simplistic assumption (real data is complex distribution)
- These distributions are usually intractable

How to model complex distributions but keep analytical simplicity ?

How to *complexify* $\mathbf{z}_0 \sim p_0(\mathbf{z}_0)$

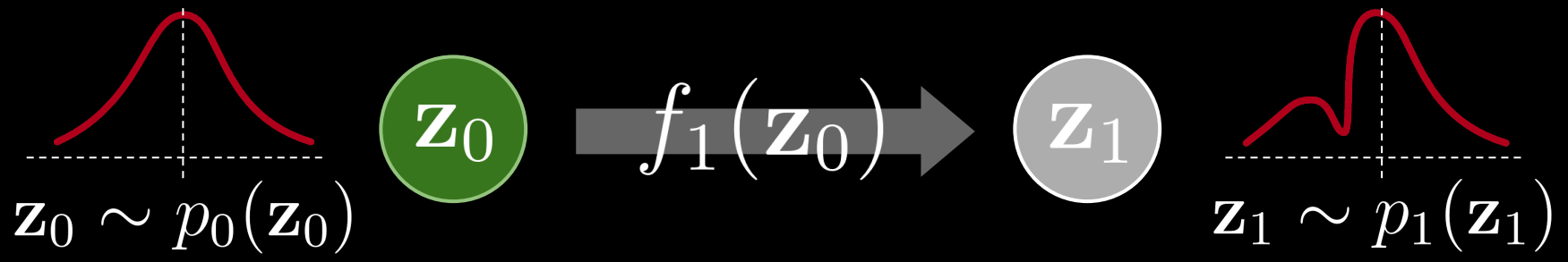


Condition for $\mathbf{z}_1 \sim p_1(\mathbf{z}_1)$ to be a valid distribution ? —— **integrates to 1**

Applying $f : \mathbb{R}^d \to \mathbb{R}^d$ the change of volume is the determinant of the Jacobian

Normalizing flows

Given a random variable $\mathbf{z}_0 \sim p_0(\mathbf{z}_0)$ we can transform it with $f : \mathbb{R}^d \rightarrow \mathbb{R}^d$



In order for $p_1(\mathbf{z}_1)$ to be a distribution, we only need it to sum (integrate) to 1

Since we have $\mathbf{z}_1 = f_1(\mathbf{z}_0)$ we need to know how it changes the volume

Change of volume is given by the **determinant of the Jacobian**

$$p(\mathbf{z}_1) = p(\mathbf{z}_0) \left| \det \frac{\delta f^{-1}}{\delta \mathbf{z}_1} \right| = p(\mathbf{z}_0) \left| \det \frac{\delta f}{\delta \mathbf{z}_0} \right|^{-1}$$

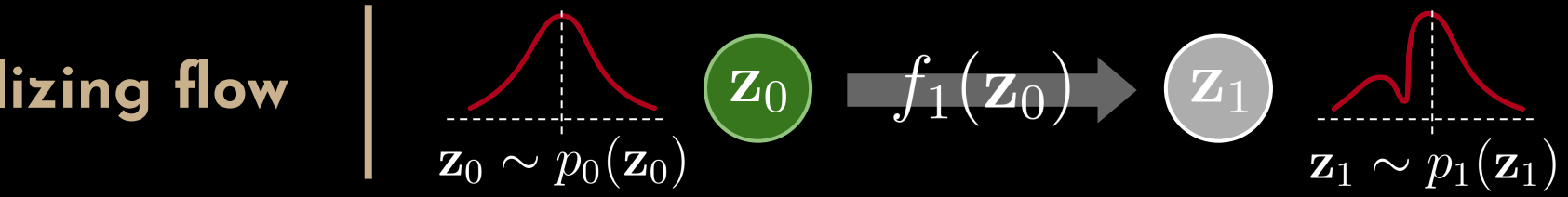
Planar flow proposed in the original paper $f(\mathbf{z}) = \mathbf{z} + \mathbf{u}h(\mathbf{w}^T\mathbf{z} + b)$

$$\left| \det \frac{\delta f}{\delta \mathbf{z}} \right| = \left| \det \left(\mathbf{I} + \mathbf{u}\psi(\mathbf{z})^T \right) \right| = \left| 1 + \mathbf{u}^T \psi(\mathbf{z}) \right| \quad \psi(\mathbf{z}) = h'(\mathbf{w}^T\mathbf{z} + b)\mathbf{w}$$

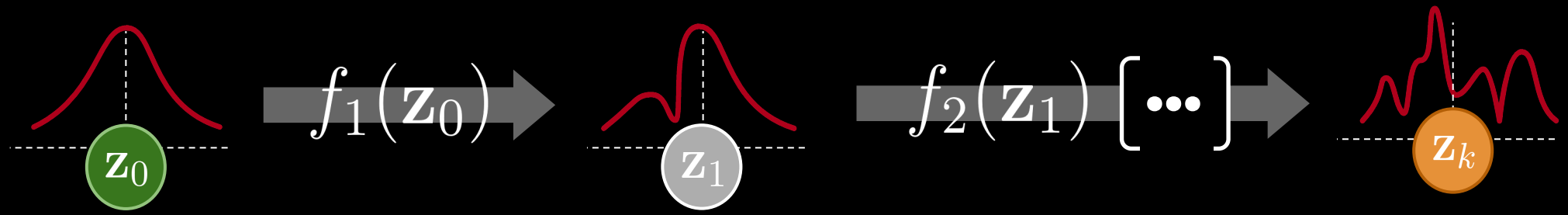
[Danilo Rezende and Shakir Mohamed, "Variational inference with normalizing flows," ICML Conference, 2015]

Normalizing flows

Normalizing flow



We chain these transforms, such that $\mathbf{z}_k = f_k \circ f_{k-1} \circ \dots \circ f_1(\mathbf{z}_0)$

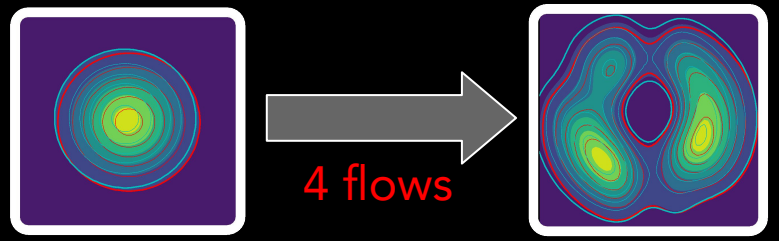


We obtain the final distribution $\mathbf{z}_k \sim p_k(\mathbf{z}_k)$ by reapplying the same reasoning

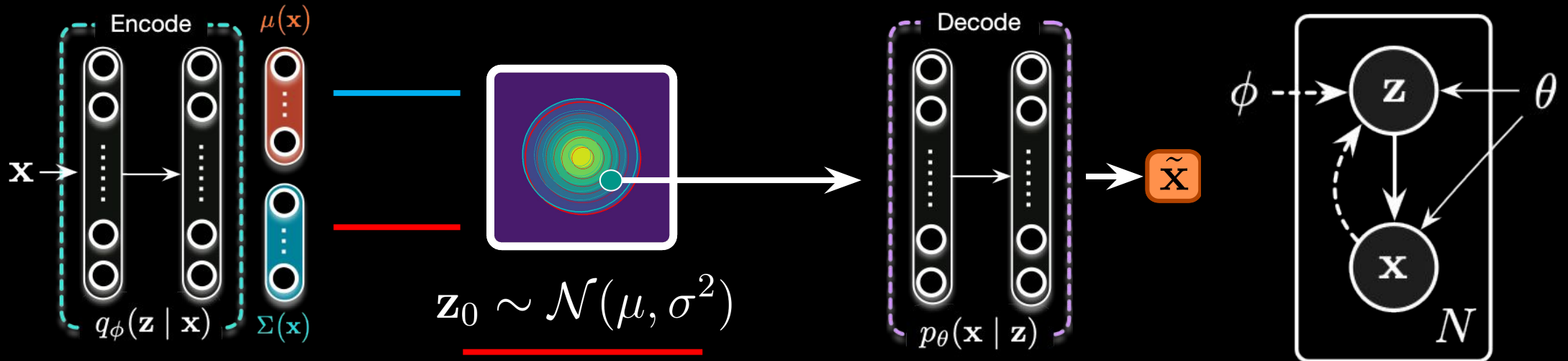
$$p_k(\mathbf{z}_k) = p_0(f_1^{-1} \circ \dots \circ f_k^{-1}(\mathbf{z}_k)) \prod_{i=1}^{k} \left| \det \frac{\partial f_i^{-1}}{\partial \mathbf{z}_i} \right|$$

$$\log p_K(\mathbf{z}_k) = \log p_0(\mathbf{z}_0) - \sum_{i=1}^{k} \log \left| \det \frac{\delta f_i}{\delta \mathbf{z}_{i-1}} \right|$$

- Learn increasingly complex distributions
- Just applying simple invertible transforms
- Of course it holds for multiple dimensions



Variational Auto-Encoders



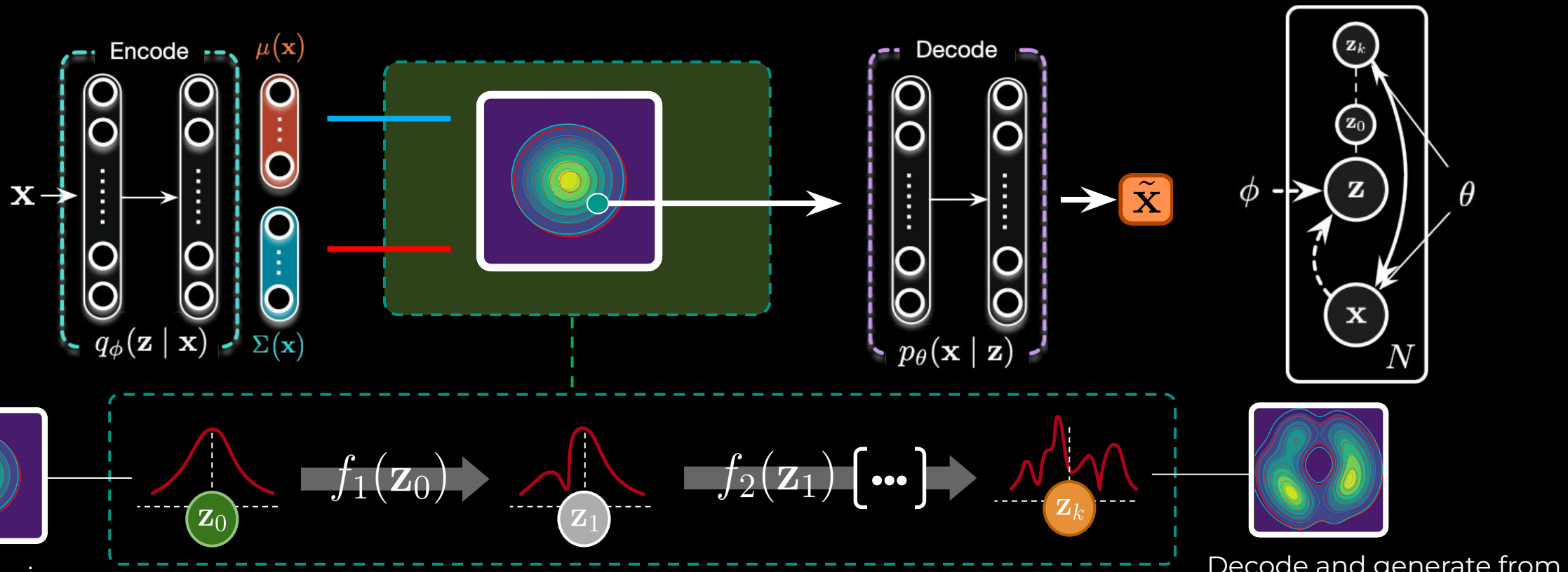
Gaussian prior (analytical simplicity)

Corresponding graphical model

- Gaussian assumption is too simple for real data
- We need to complexify this assumption

Introducing normalizing flows inside variational auto-encoders

Normalizing flows in variational inference



Still regularize with Gaussian

Complexify with normalizing flows

Decode and generate from more complex distribution

$$\mathcal{L} = \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\ln\, q_0(\mathbf{z}_0)\right] - \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\log\, p(\mathbf{x}, \mathbf{z}_K)\right] - \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\sum_{i=1}^{k} \log \left| \det \frac{\partial f_i}{\partial \mathbf{z}_{i-1}} \right|\right]$$

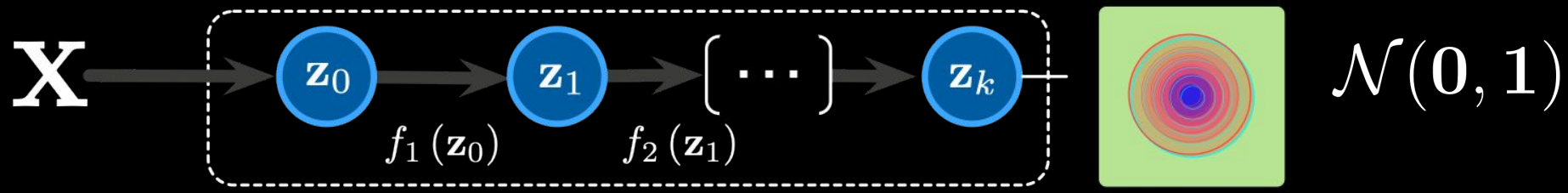
Still regularize on a simple prior (Analytical Gaussian simplicity)

Decode from the complex distribution

Ensure that all latent are distributions (normalizing flow)

Generative flows (GLOW)

Using normalizing flows directly to model the data distribution

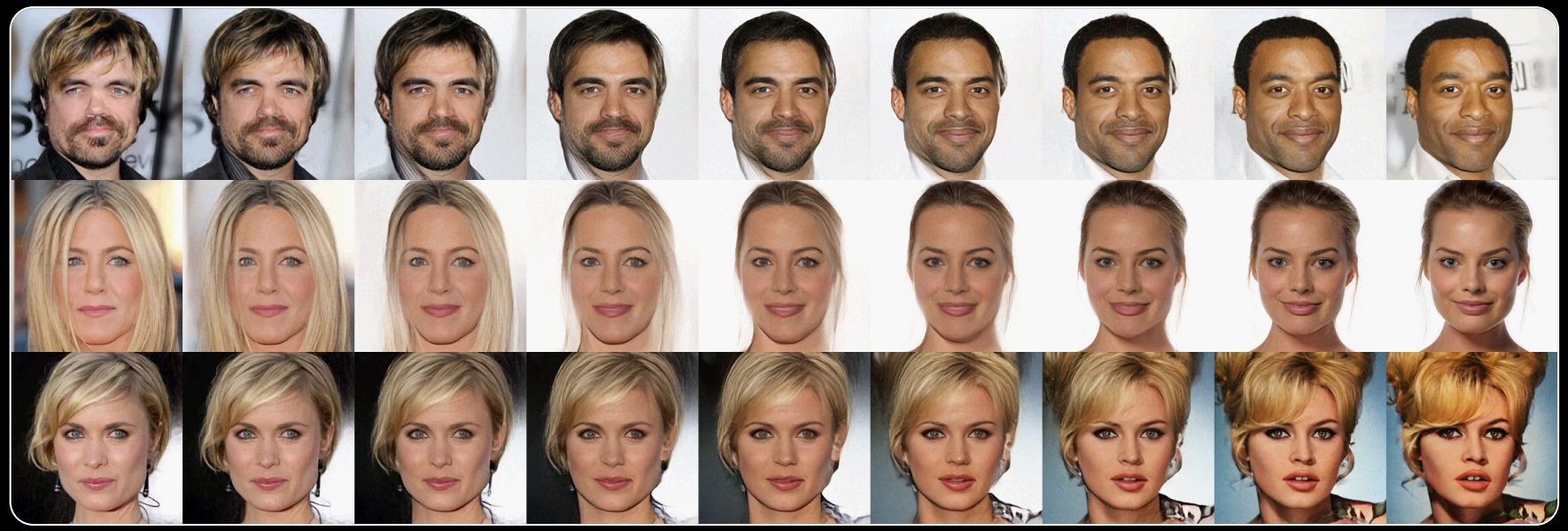
$\mathbf{x} \sim p(\mathbf{x})$ 

$\mathcal{N}(0,1)$

Simple objective

$$\mathbb{E}_{q_0}\left[\sum_{i=1}^{k} \log \left| \det \frac{\delta f_i}{\delta \mathbf{z}_{i-1}} \right|\right]$$

$$\mathbb{E}\left[\log p(\mathbf{z}_k)\right]$$



[Kingma, D. P., & Dhariwal, P. (2018). Glow: Generative flow with invertible 1x1 convolutions. In *NeurIPS* (pp. 10215-10224).]