UNIVERSITÄT
HEIDELBERG

Generative Modelling

L7, Structural Bioinformatics

WiSe 2023/24, Heidelberg University

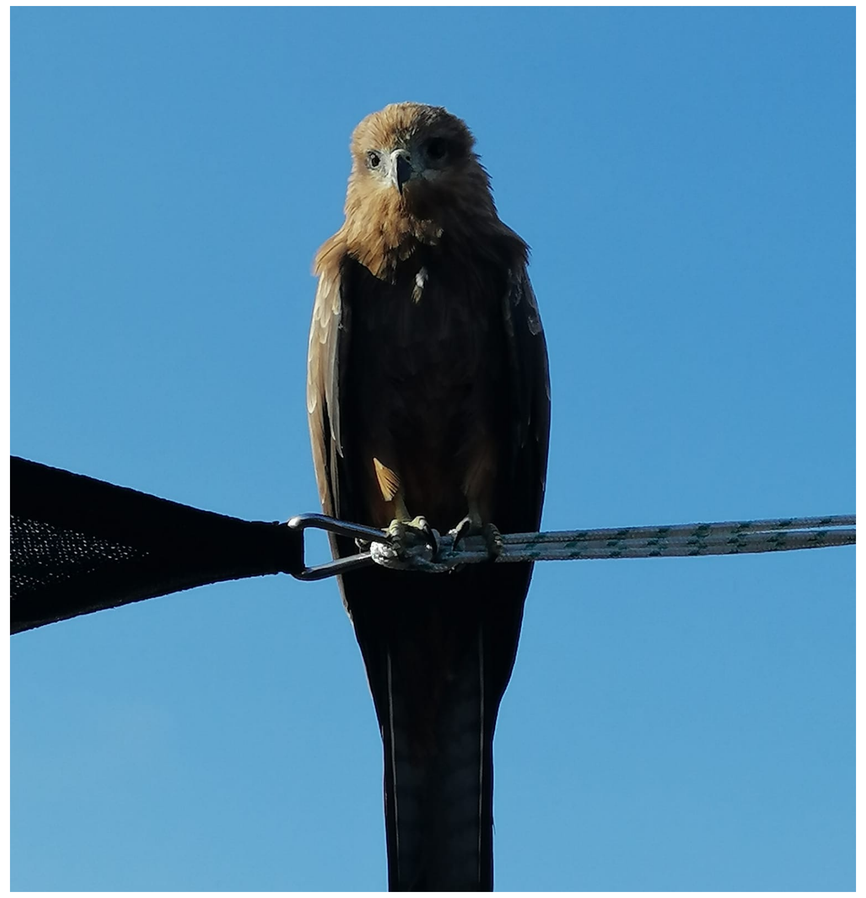
Overview

1. What is generative modelling?
2. Autoencoders in all flavours (Classic/Denoising/Variational)
3. Diffusion Models
4. Applications and Outlook

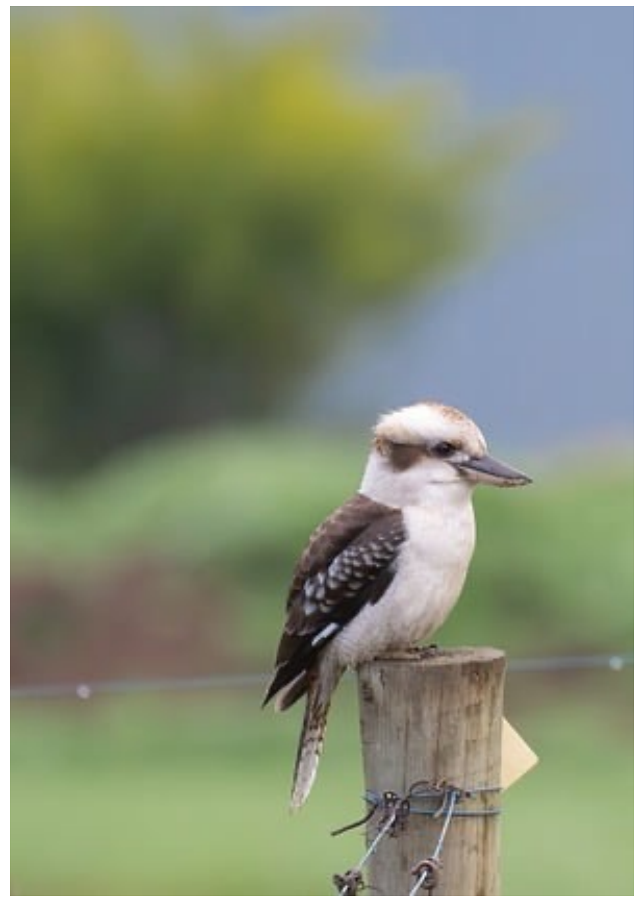
1. What is generative modelling?

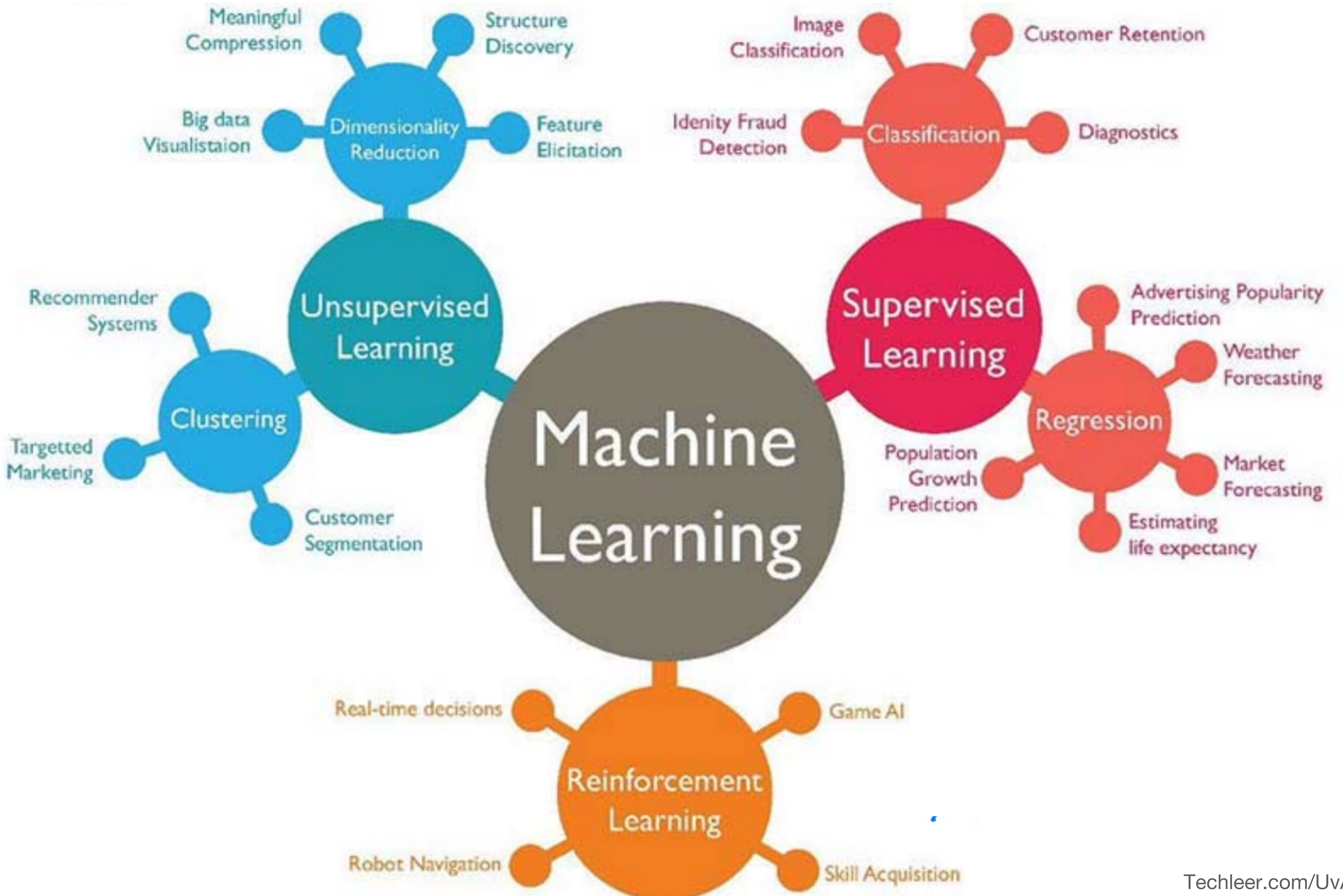
Basic Idea of Generative Modelling

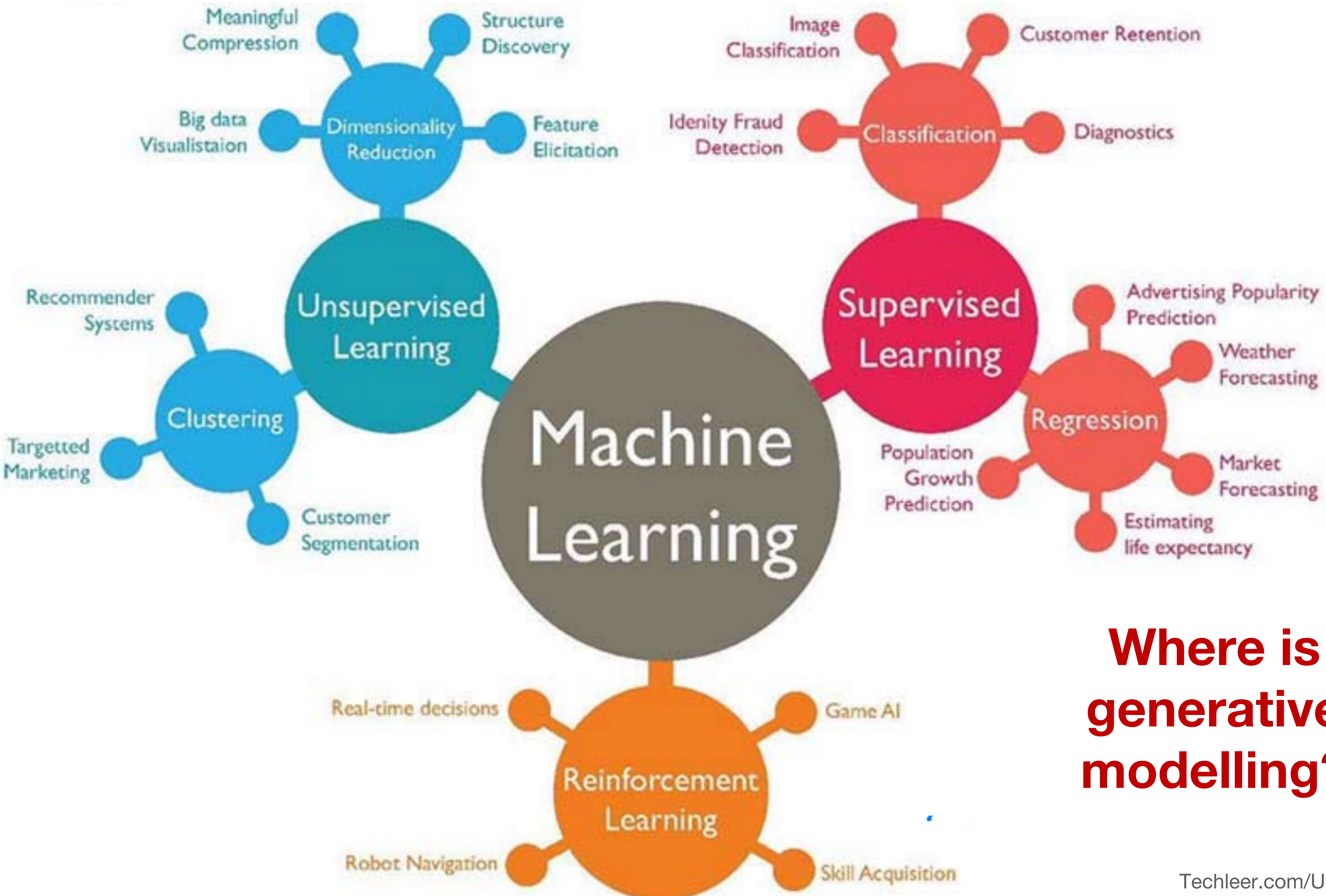
Given data, produce new data that looks similar



Generative Model




Machine Learning
Unsupervised Learning
Dimensionality Reduction
Meaningful Compression
Structure Discovery
Big data Visualistaion
Feature Elicitation
Clustering
Recommender Systems
Targetted Marketing
Customer Segmentation
Supervised Learning
Classification
Image Classification
Customer Retention
Idenity Fraud Detection
Diagnostics
Regression
Advertising Popularity Prediction
Weather Forecasting
Population Growth Prediction
Market Forecasting
Estimating life expectancy
Reinforcement Learning
Real-time decisions
Game AI
Robot Navigation
Skill Acquisition



Where is generative modelling?

Techleer.com/UvA ML1 Course

We can do classification in several ways

Hard decisions (Decision rule)

Input Data

x

Classifier

Class Label

y

Dog

Bird

We can do classification in several ways

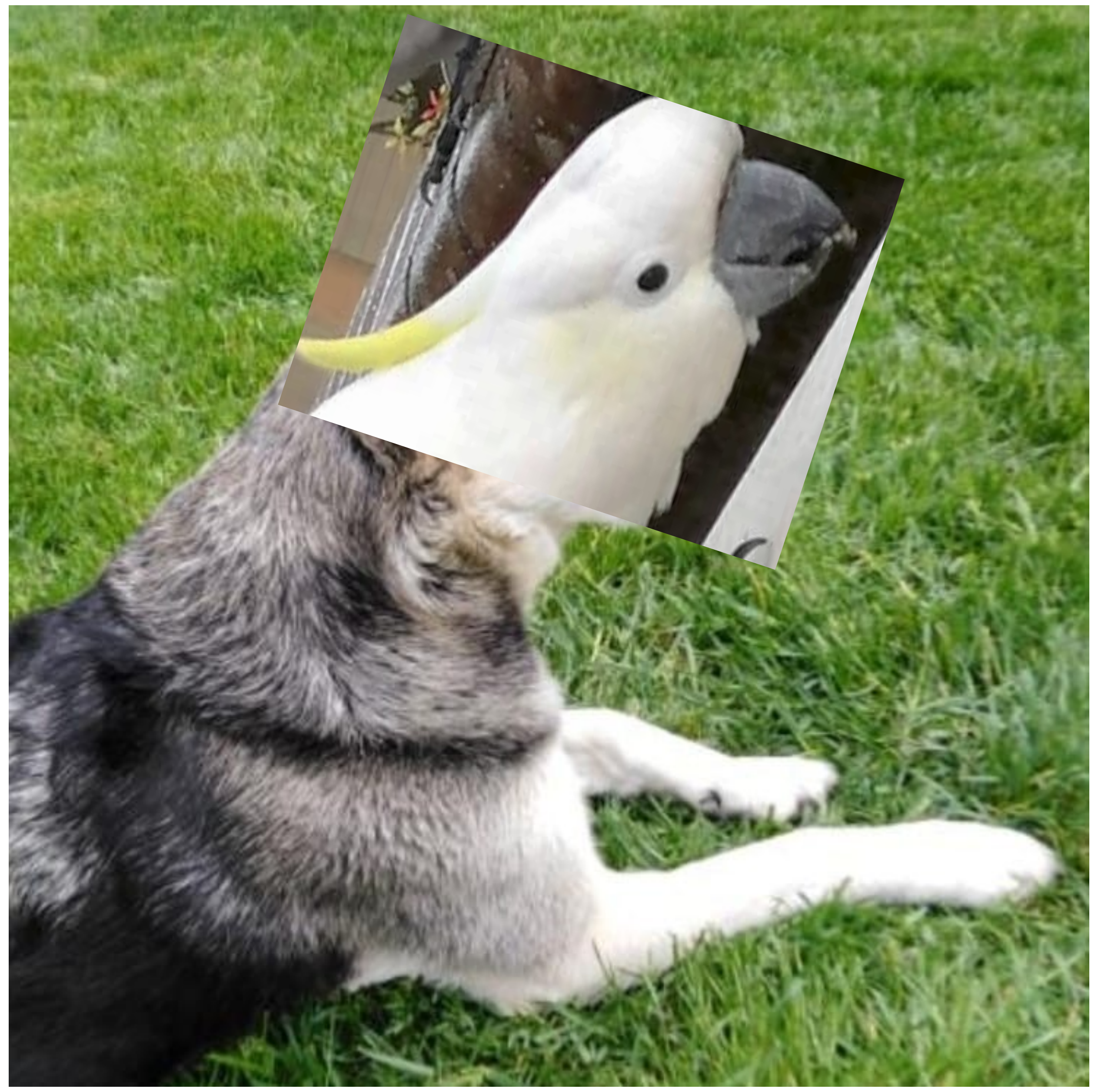
Hard decisions does not tell about uncertainty!

Input Data

x

Class Label

y



Classifier

Bird

We can do classification in several ways

Soft decisions (probabilistic)

Input Data
x

Prob. of label given data
$p(y|x)$

Dog: 0.9
Bird: 0.1

Classifier

Bird: 0.95
Dog: 0.05

We can do classification in several ways

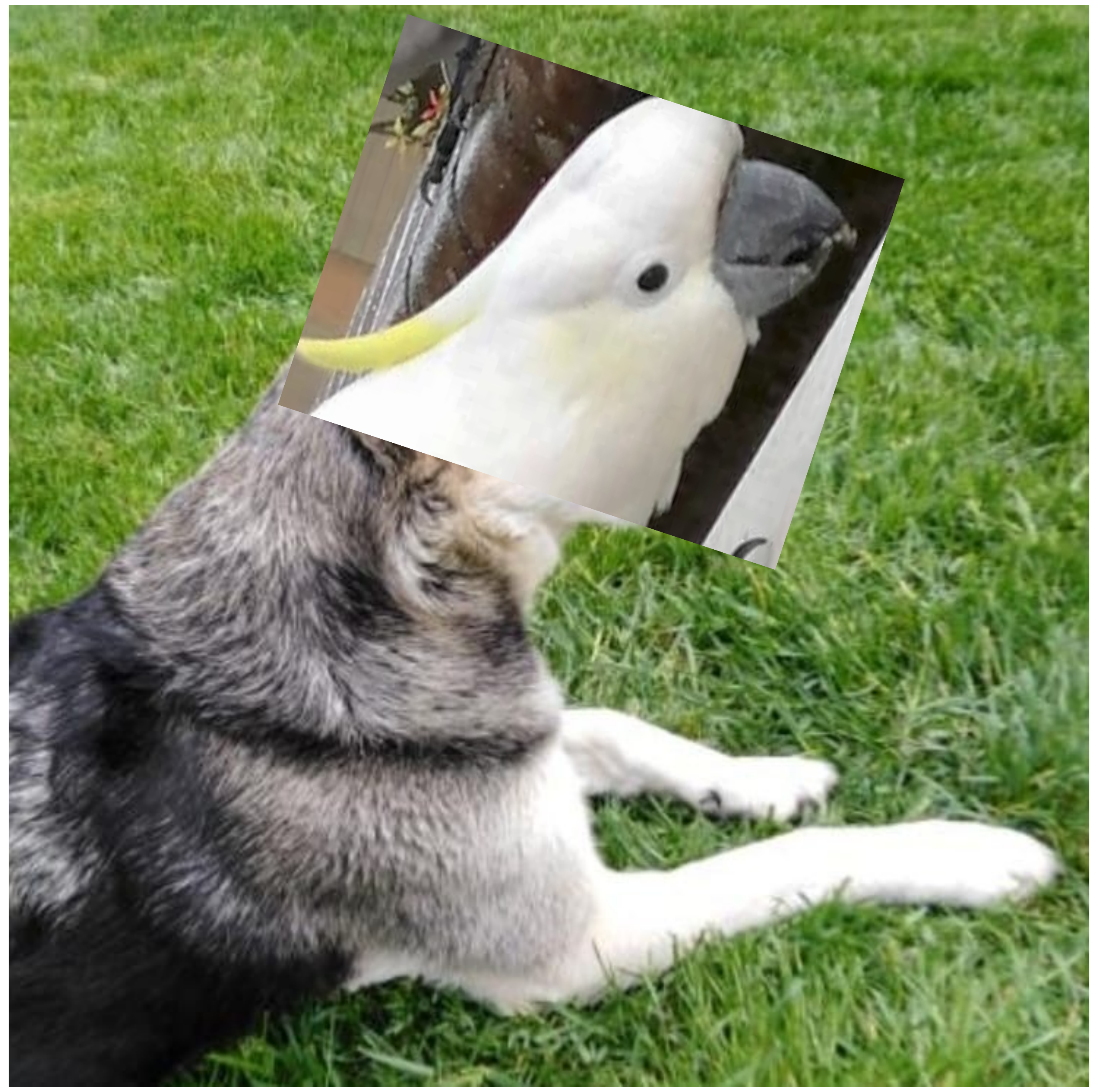
Soft decisions (probabilistic)

Input Data

x

Prob. of label given data

$p(y|x)$



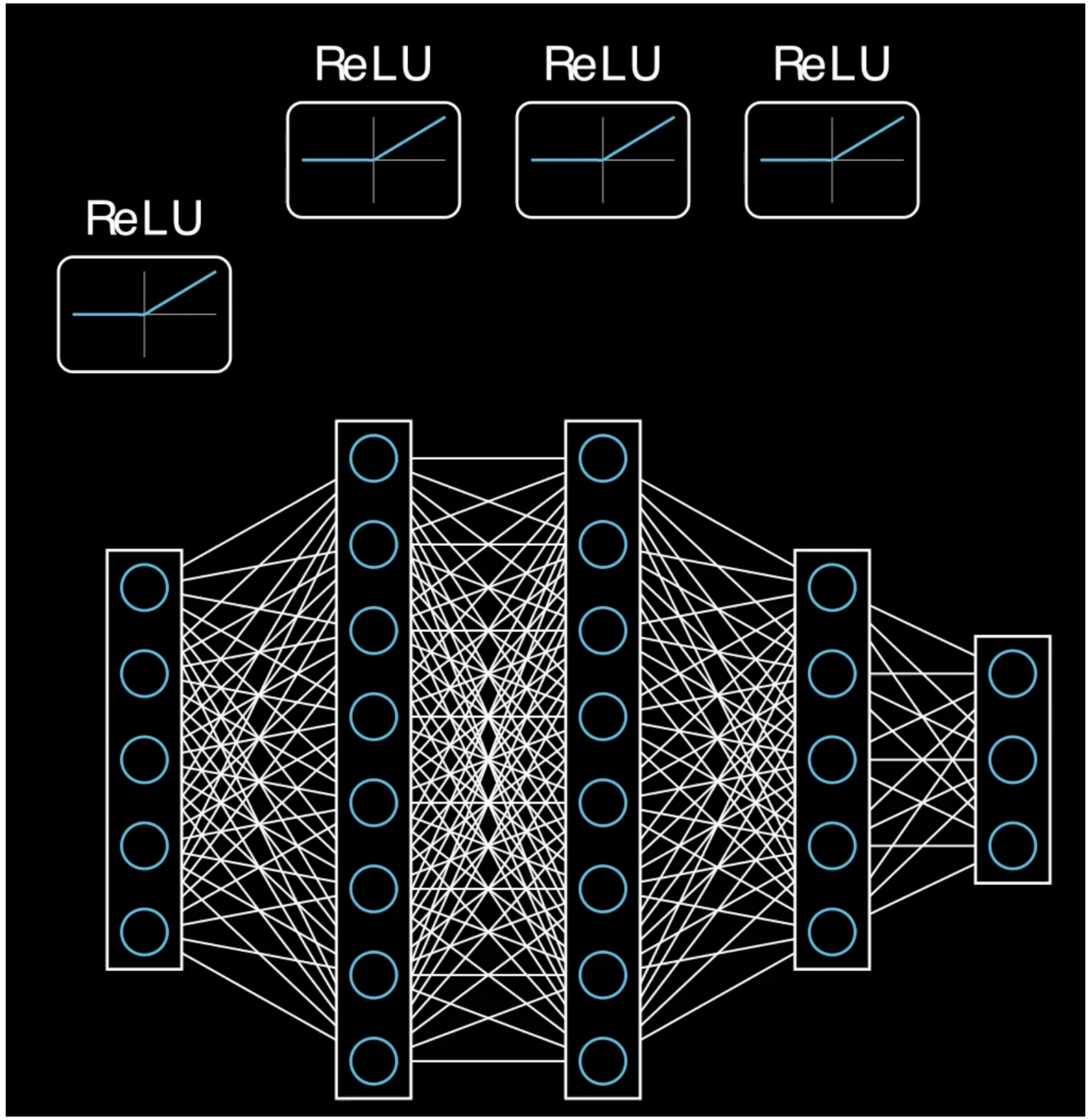
Classifier

Dog: 0.45

Bird: 0.55

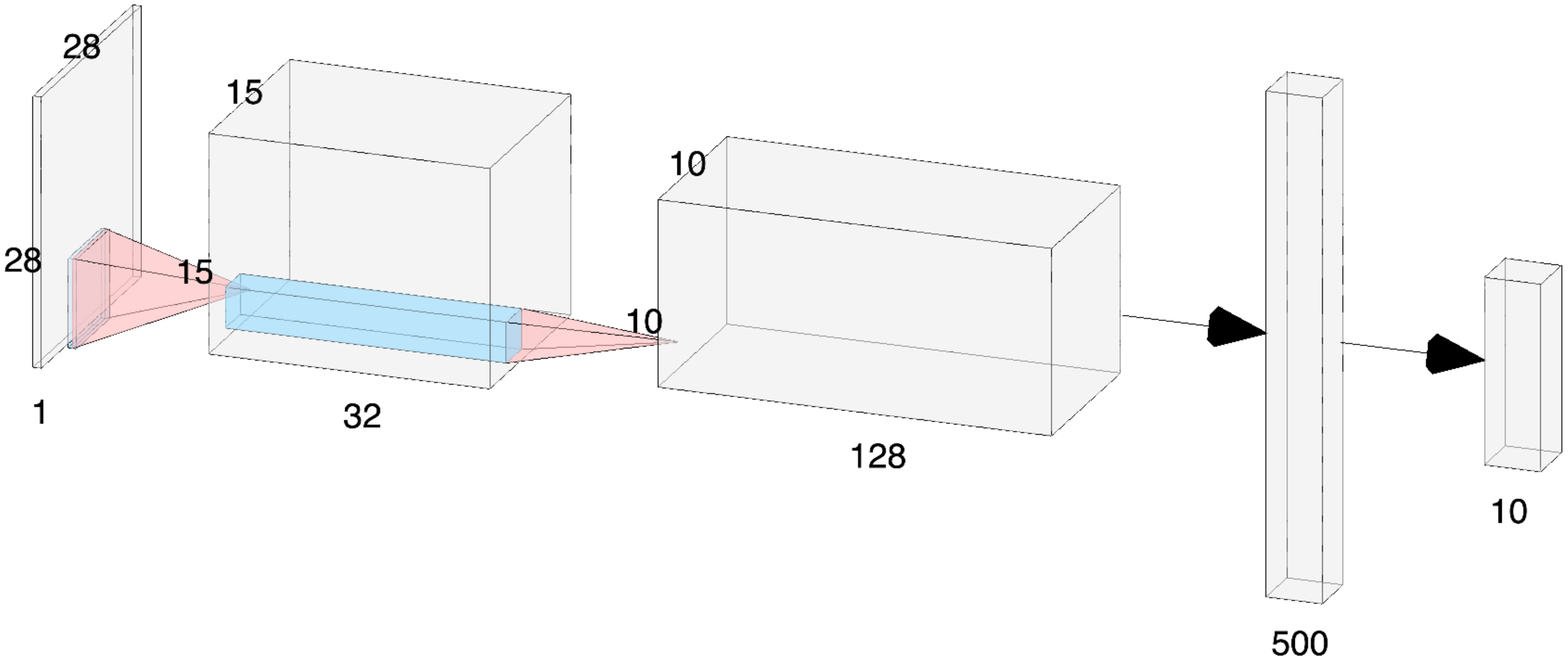
How do we get soft decisions?

Use the representation instead of a final decision



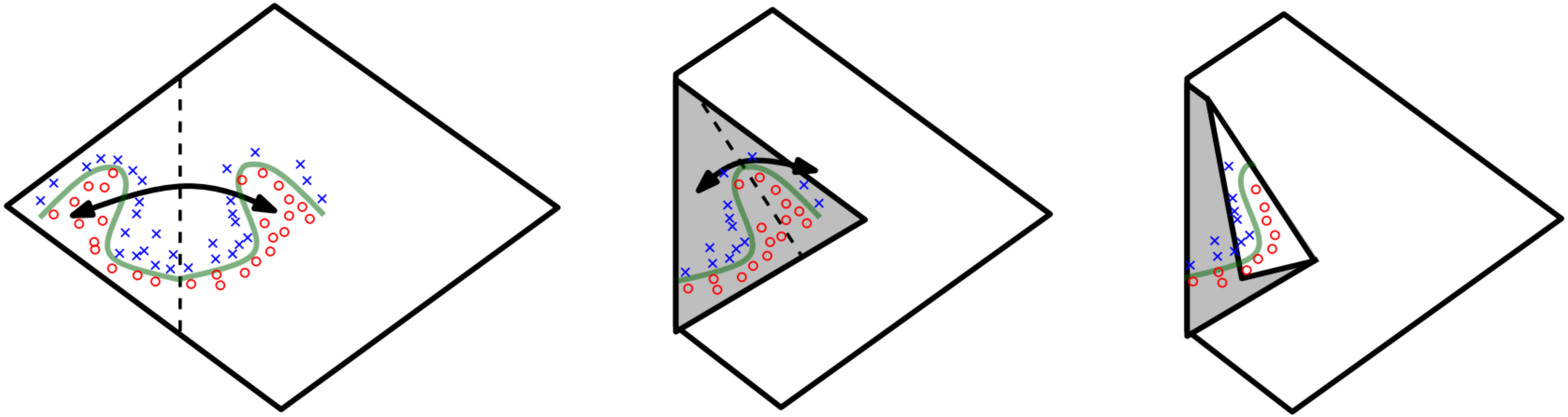
How do we get soft decisions?

Use the representation instead of a final decision



Reminder: Representation Learning

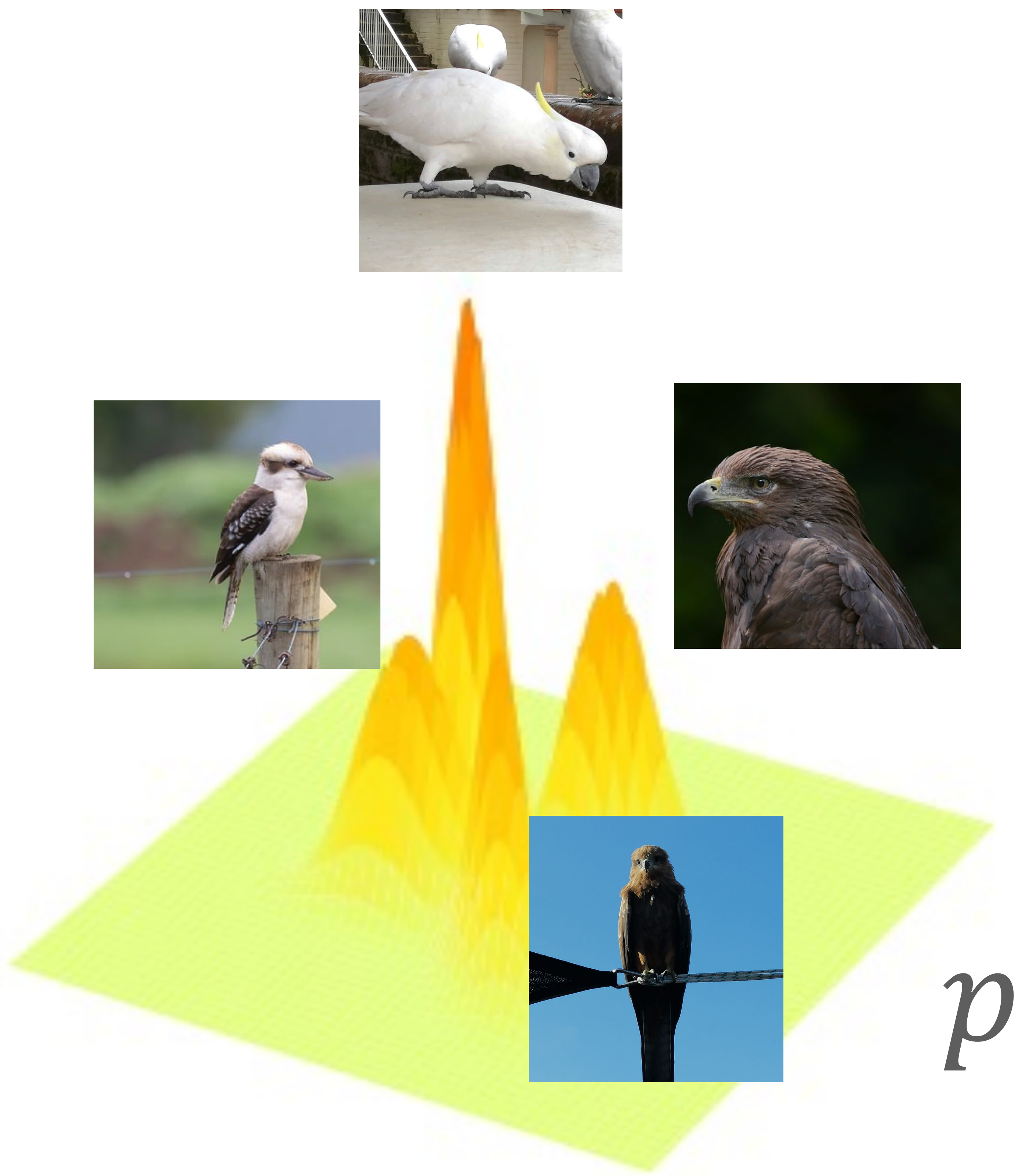
Neural networks use non-linear transformations to deform data



Guido Montúfar, Razvan Pascanu, Kyunghyun Cho, Yoshua Bengio. **On the Number of Linear Regions of Deep Neural Networks** Arxiv (2014)

Reminder: Representation Learning

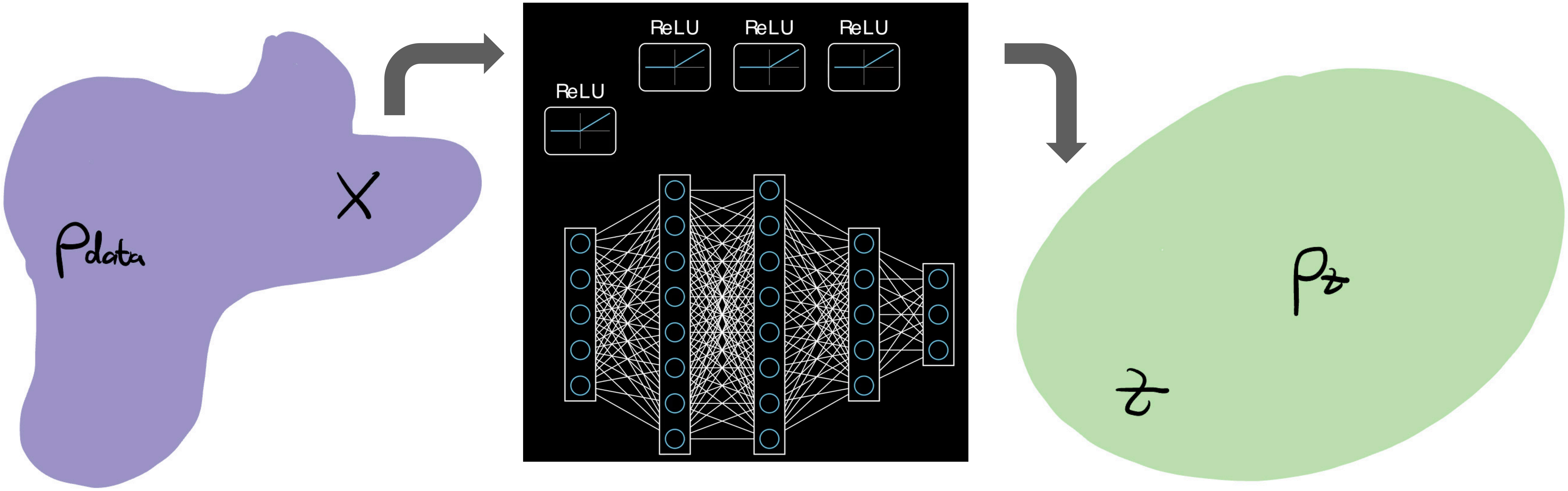
Neural networks use non-linear transformations to deform data



p_{data}

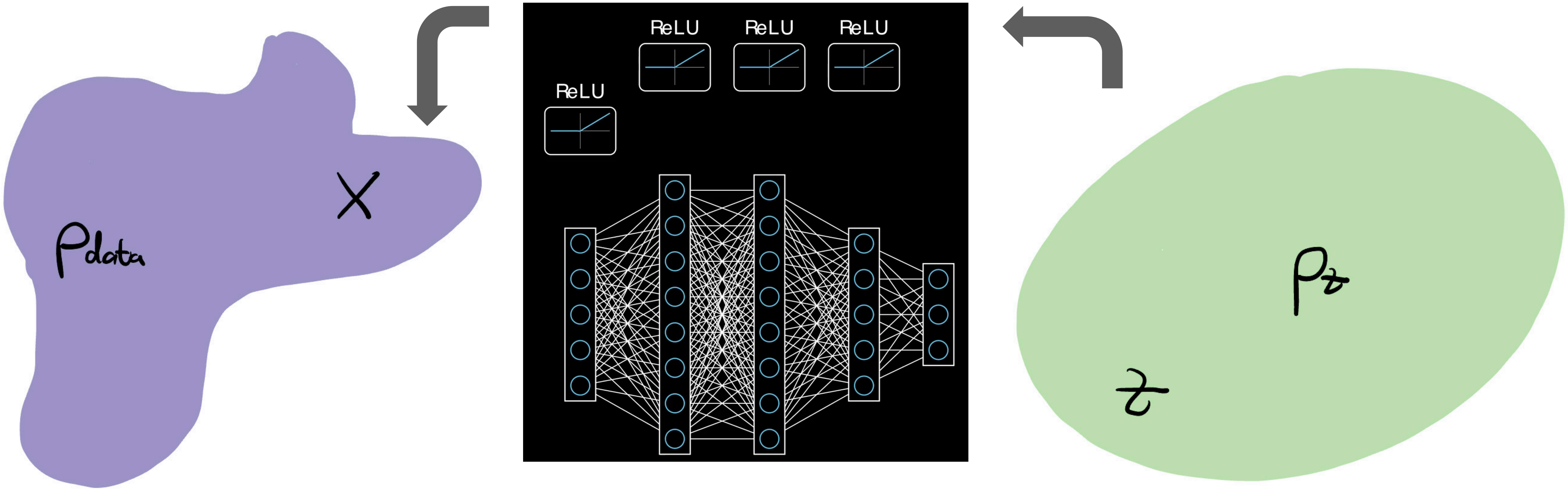
Reminder: Representation Learning

Neural networks use non-linear transformations to deform data



Generative Modelling: Turn it around!

Go from a representation to a data distribution



Why is generative modelling interesting?

1. Sample new datapoints

2. Evaluate likelihood of samples

Unconditional vs Conditional Models

Every probabilistic model is in some sense a generative model

Conditional Model

- Supervised learning
- Observe x,y pairs
- learn $p(y|x)$
- Ex: regression, classification

Unconditional Model

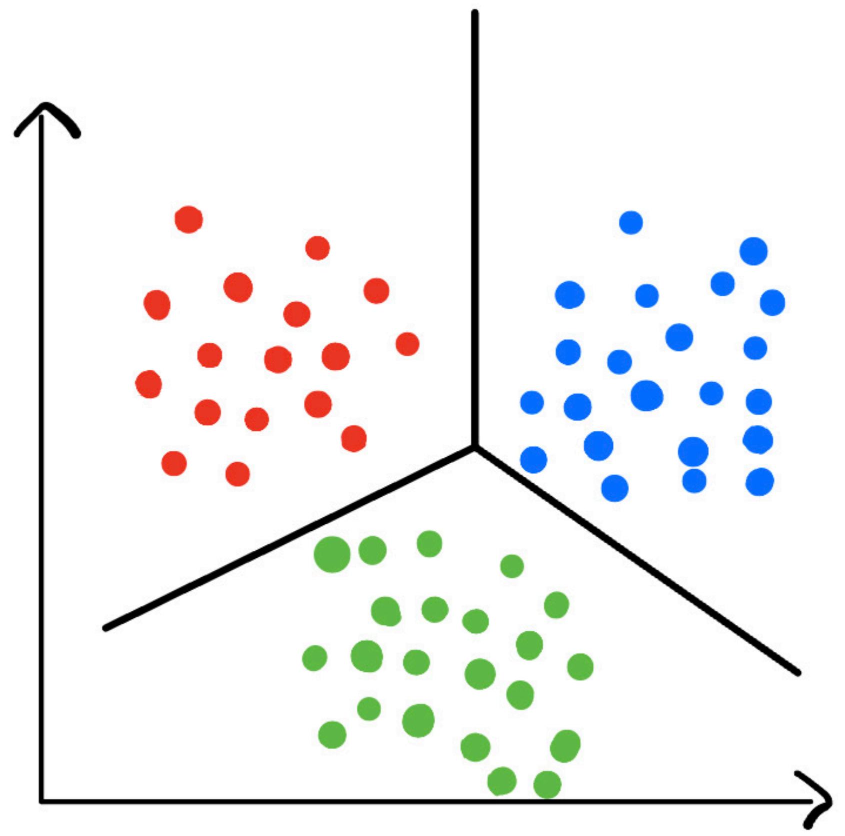
- Unsupervised learning
- Observe only data x, no labels
- learn $p(x)$
- Ex: density estimation, dim.red.

Unconditional vs Conditional Models

Every probabilistic model is in some sense a generative model

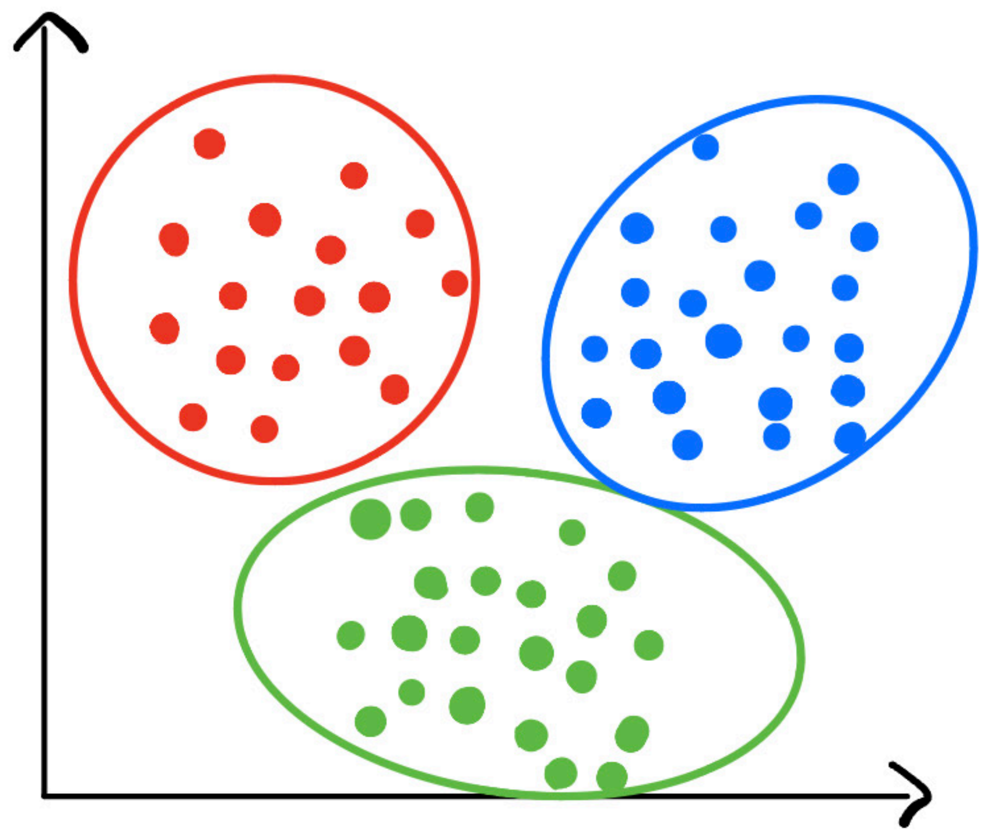
Conditional Model

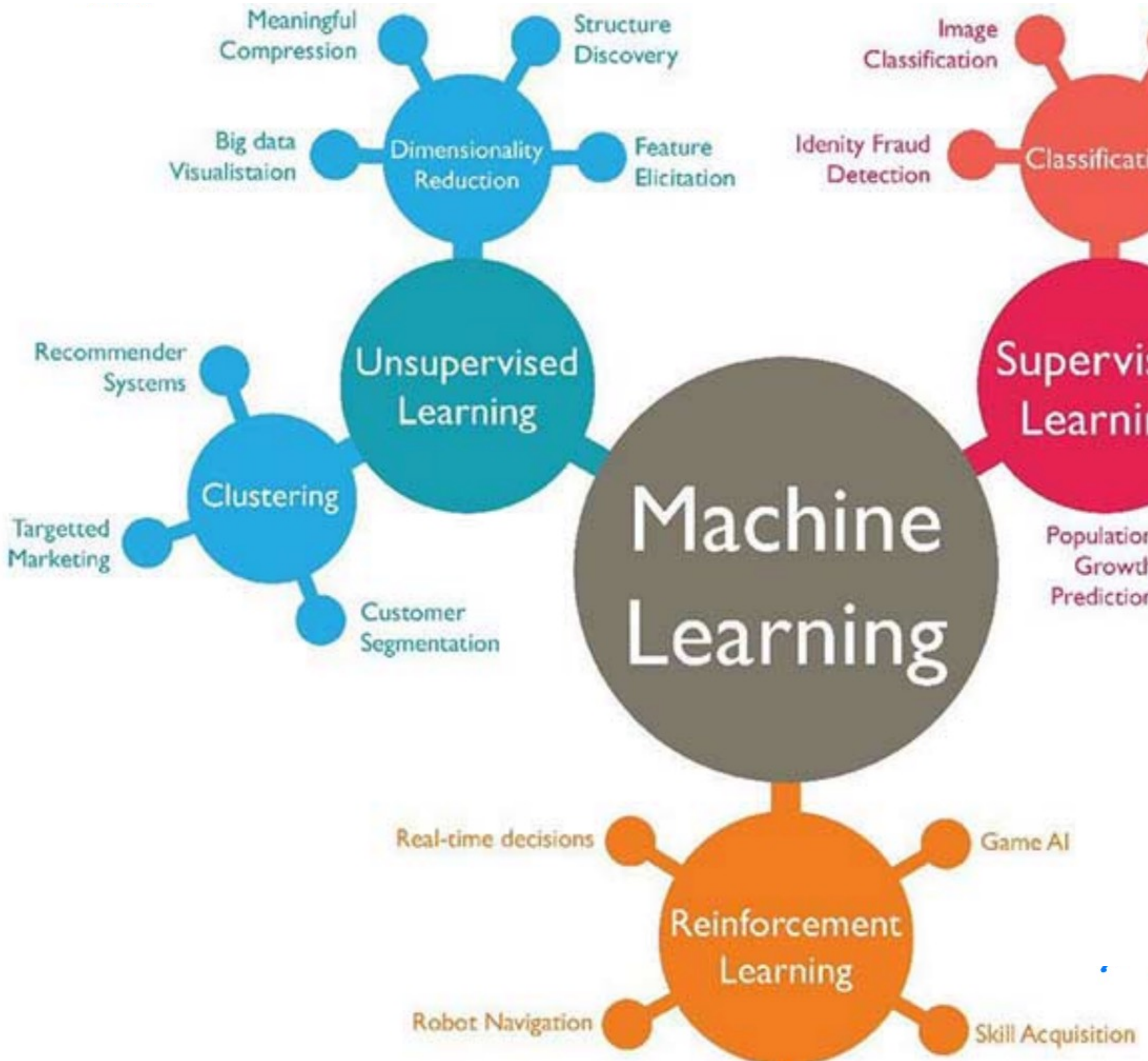
- Supervised learning
- Observe x,y pairs
- learn $p(y|x)$
- Ex: regression, classification



Unconditional Model

- Unsupervised learning
- Observe only data x, no labels
- learn $p(x)$
- Ex: density estimation, dim.red.



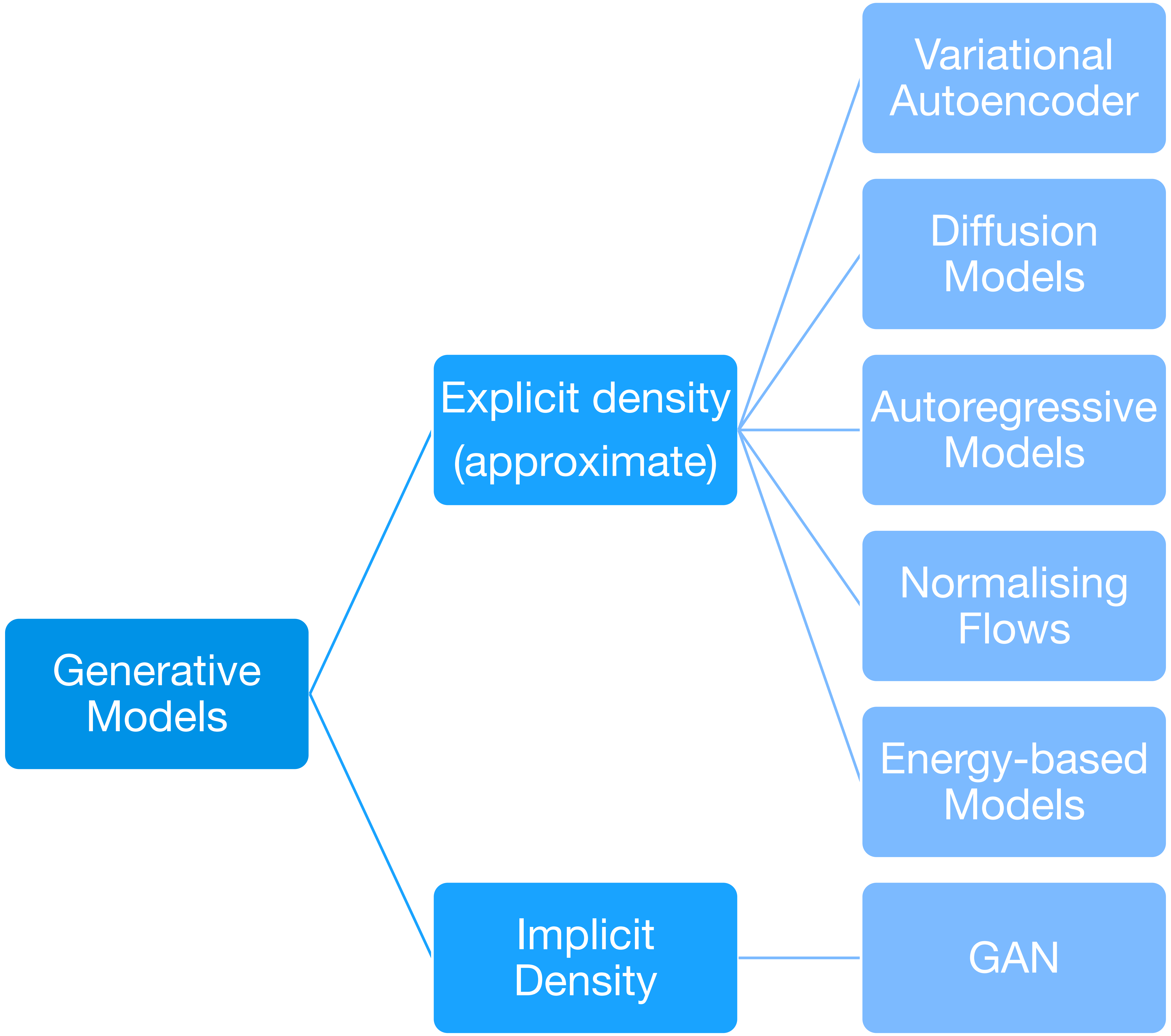


Does not really fit this classification!

Techleer.com/UvA ML1 Course

How to represent $p(x)$?

Approximate Density Models dominate recently



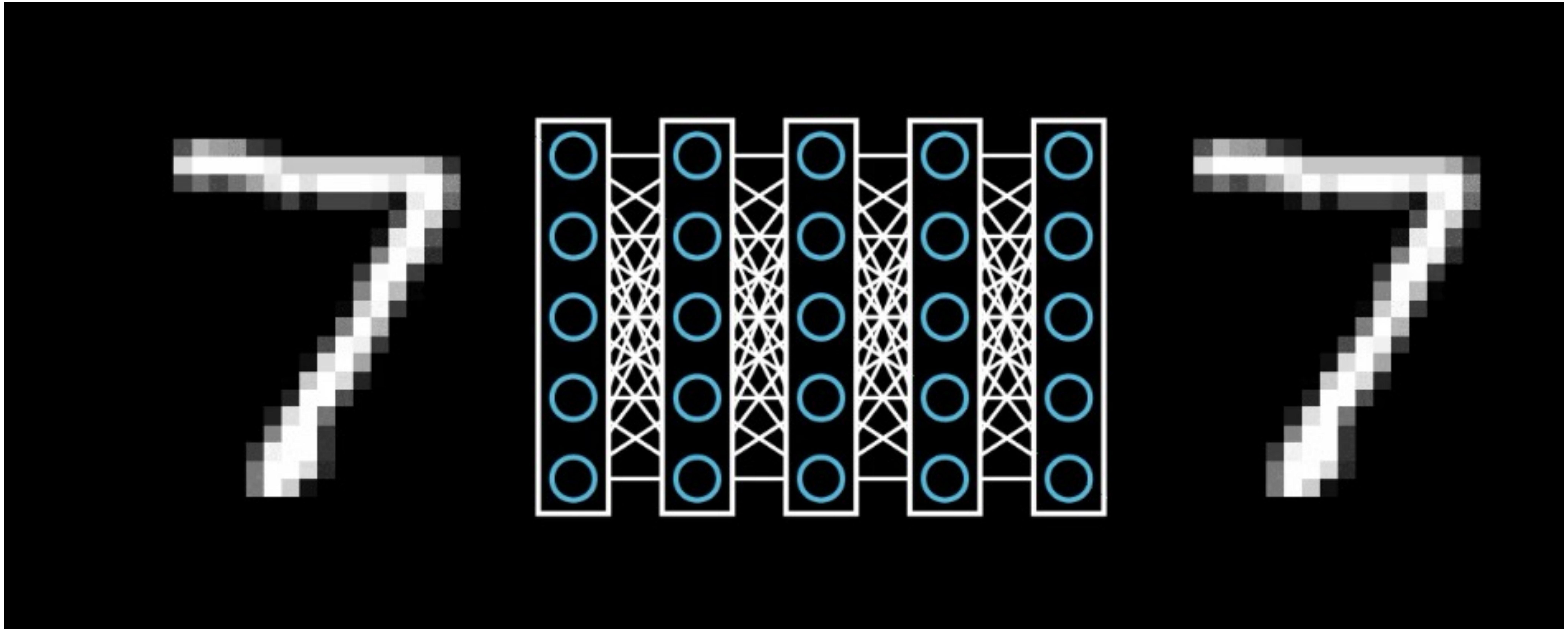
2. (Variational) Autoencoders

How to use unlabelled data for learning?

Think of interesting tasks that just involve the data itself

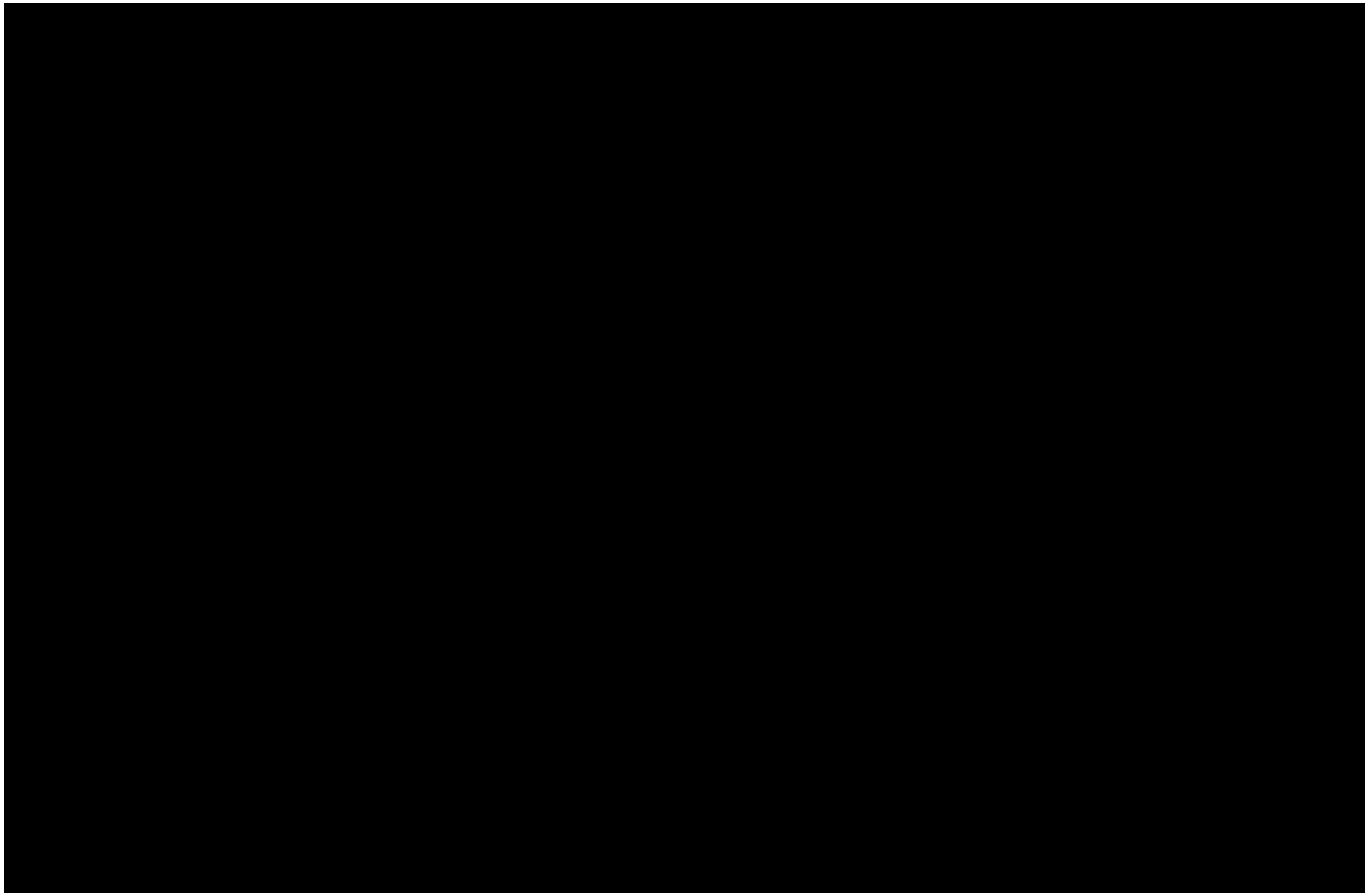
How to use unlabelled data for learning?

Think of interesting tasks that just involve the data itself



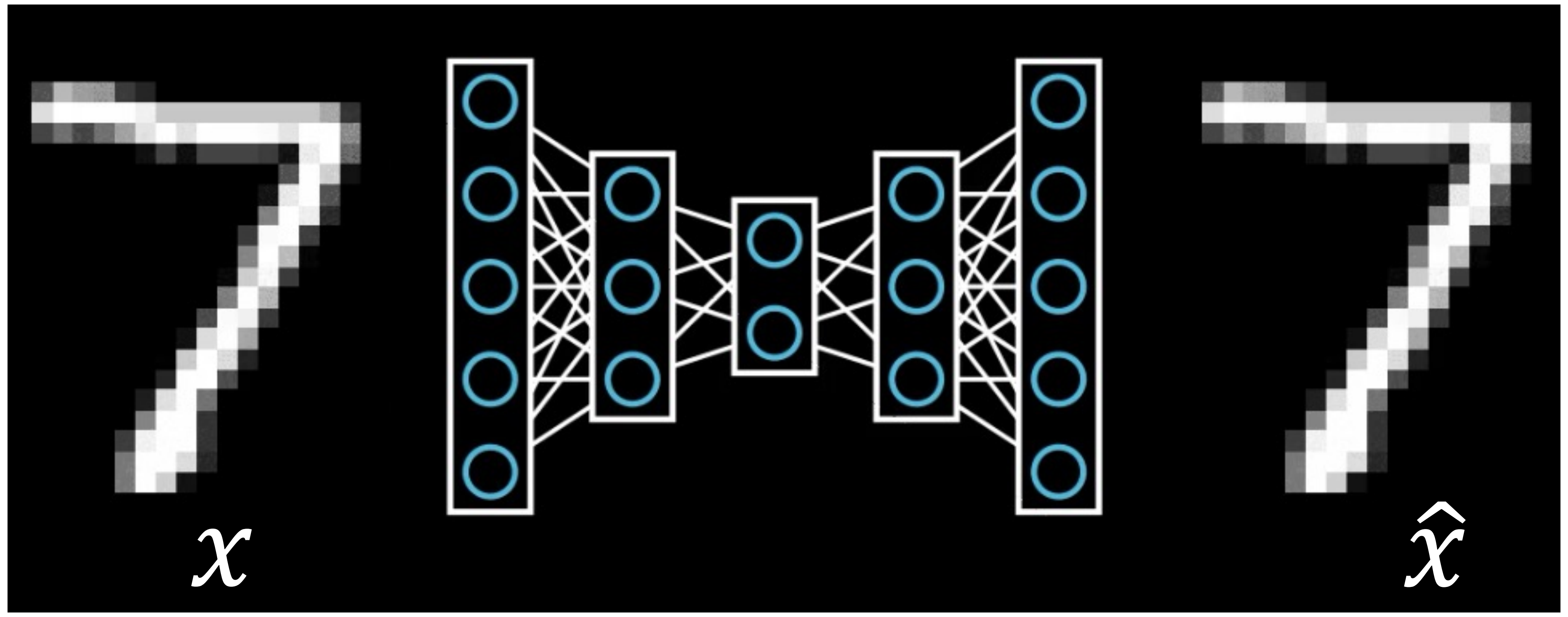
Autoencoders: introduce a bottleneck

Forcing the network to learn data compression

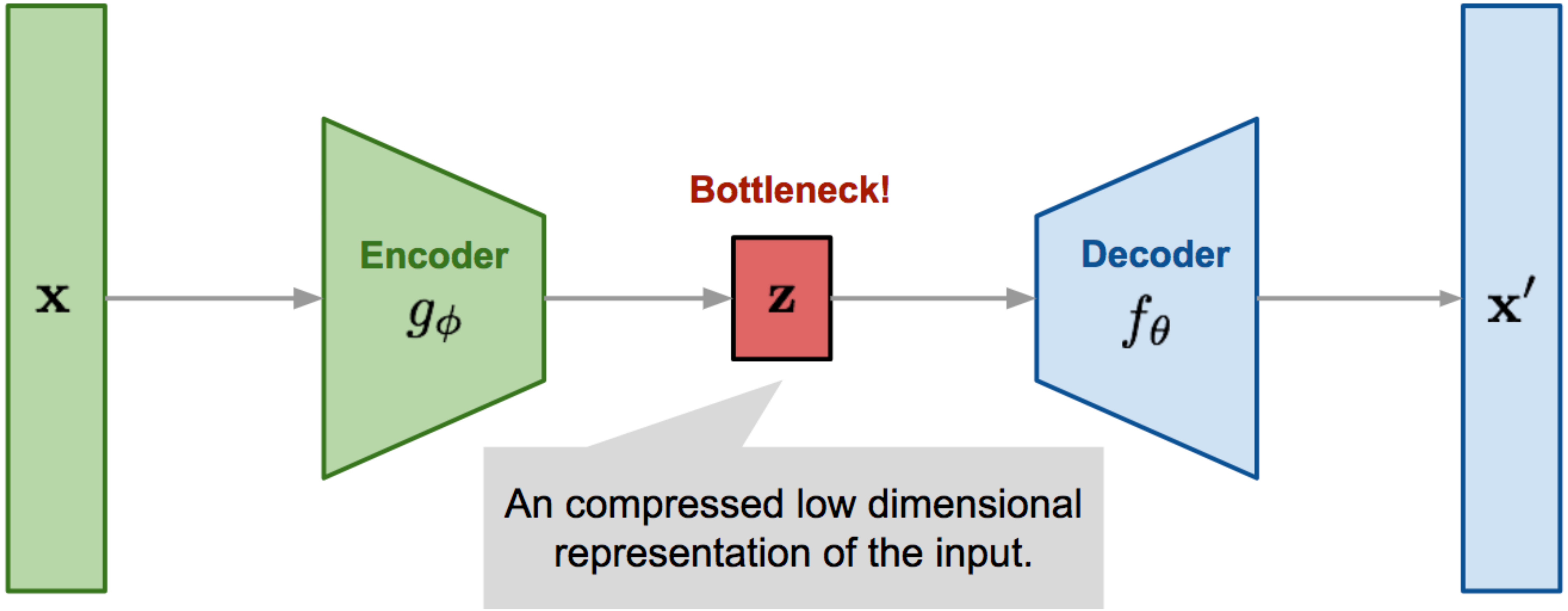


Autoencoders: introduce a bottleneck

Learn by penalizing a reconstruction error



$$\mathcal{L}(x, \hat{x}) = ||x - \hat{x}||^2$$



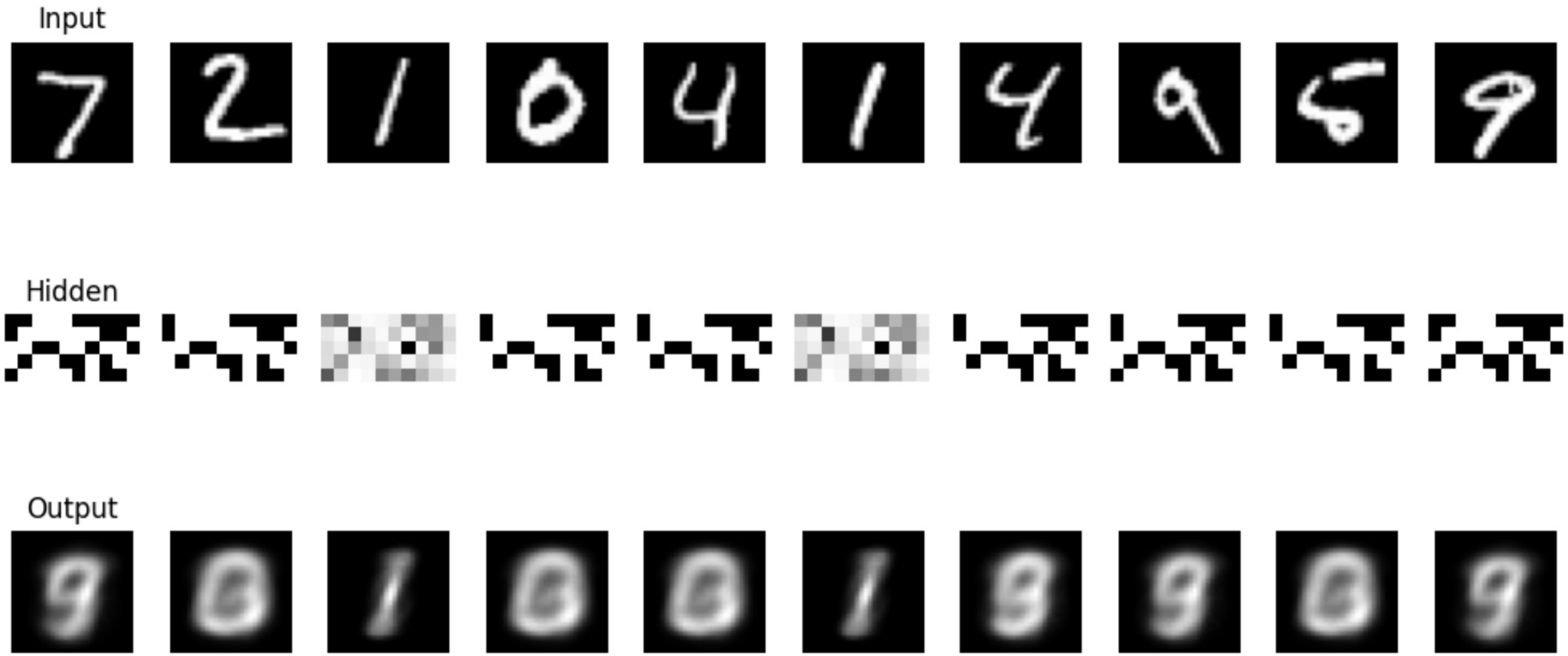
Autoencoders: introduce a bottleneck

Forcing the network to learn data compression

(a) Autoencoder encodes 8-dimensional toy data as binary code.

Autoencoders: introduce a bottleneck

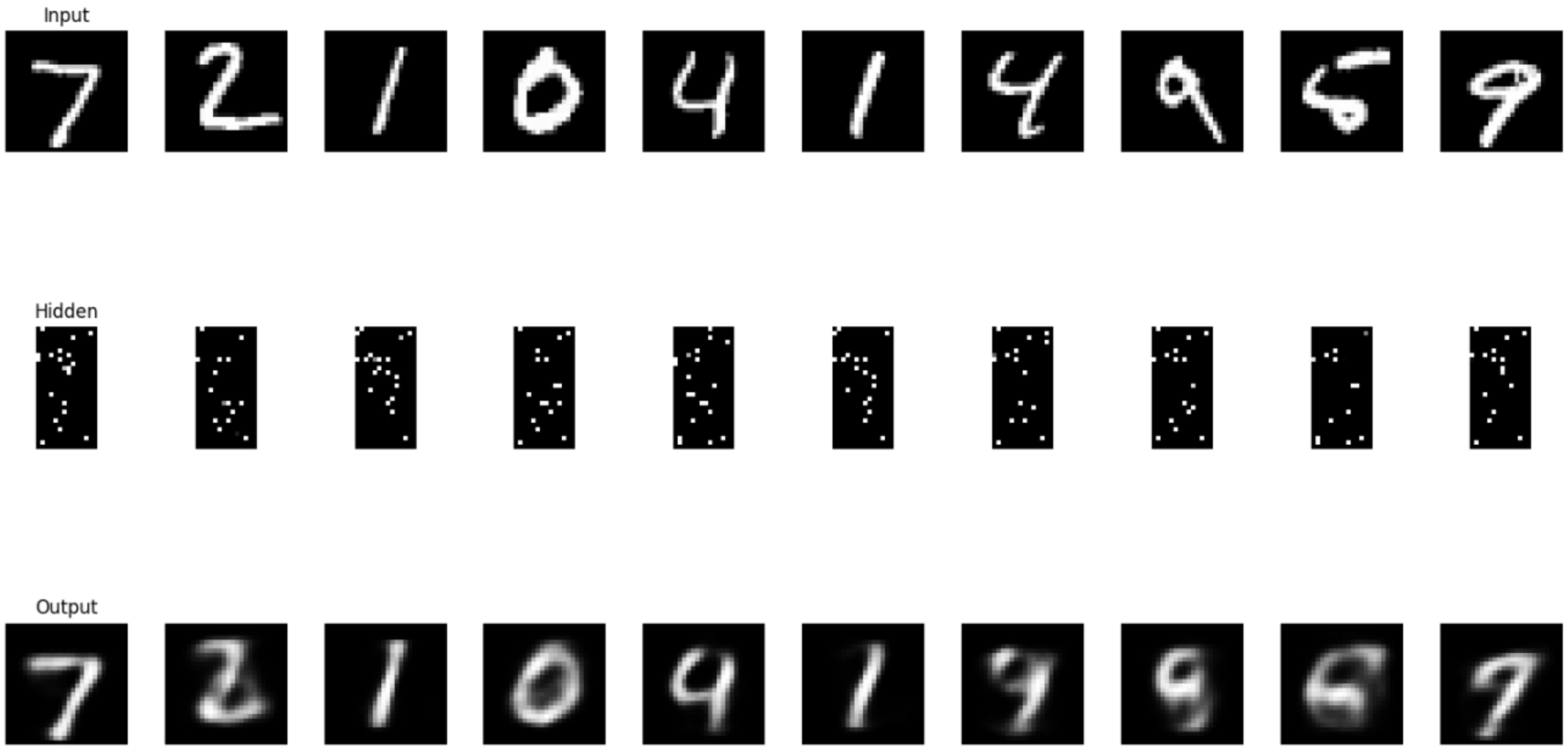
Forcing the network to learn data compression



(b) Autoencoder learns compressed version of MNIST digits (50 hidden units).

Autoencoders: introduce a bottleneck

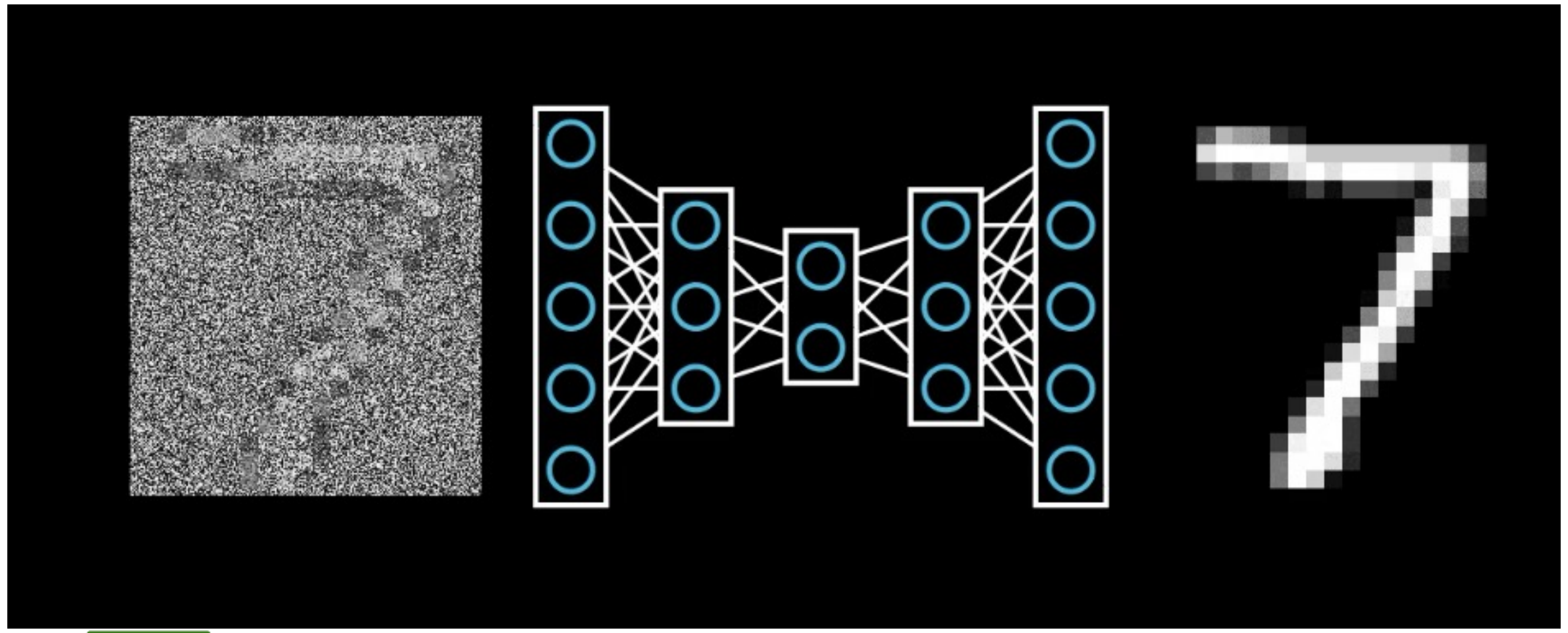
Forcing the network to learn data compression



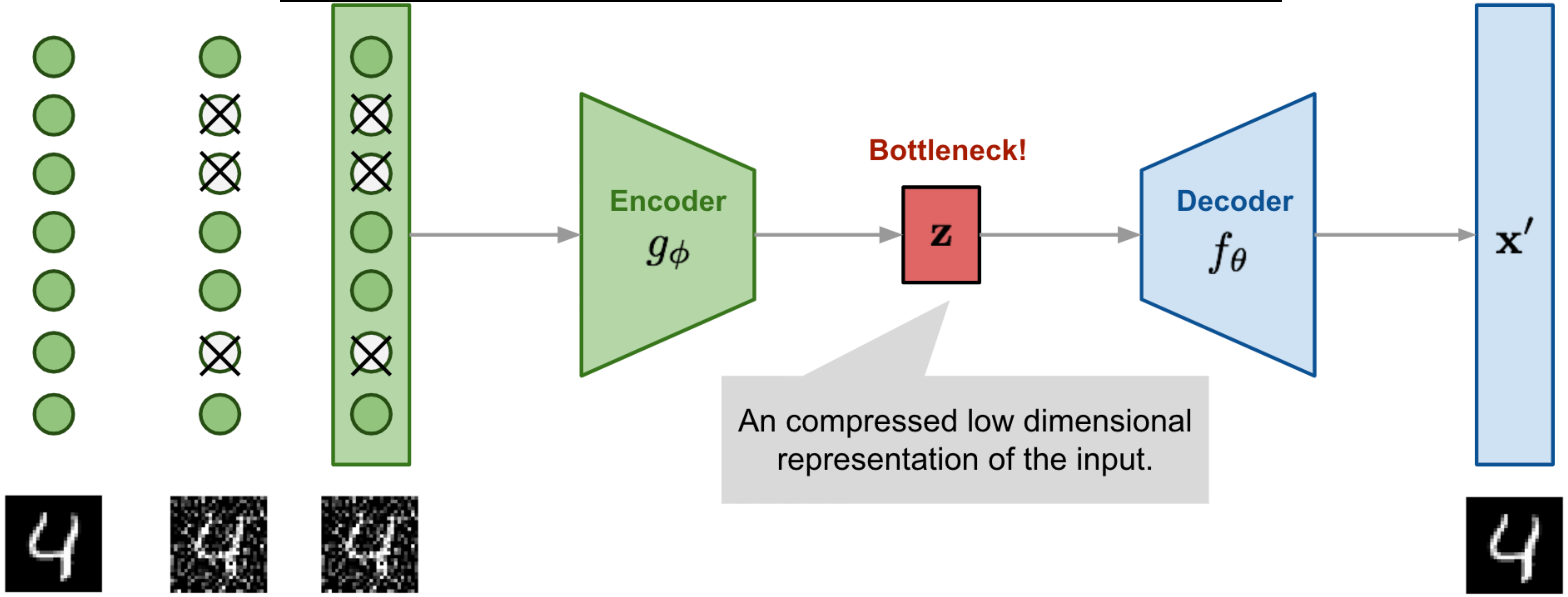
(c) More hidden units (here 392) allow the autoencoder to learn a more accurate latent representation.

Denoising Autoencoder

Make the task harder via noisy input

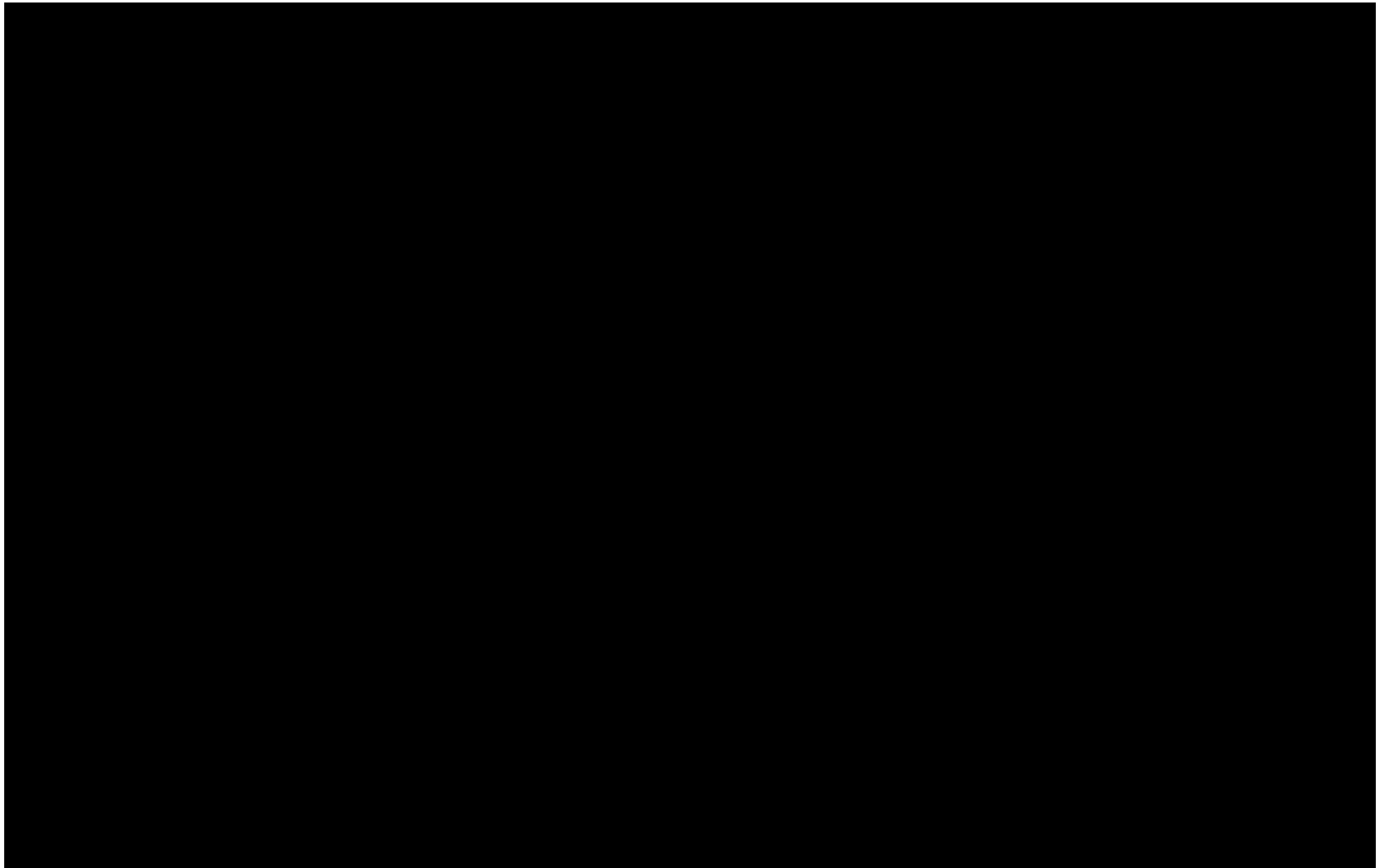


$$\mathcal{L}(x, \hat{x}) = ||x - \hat{x}||^2$$



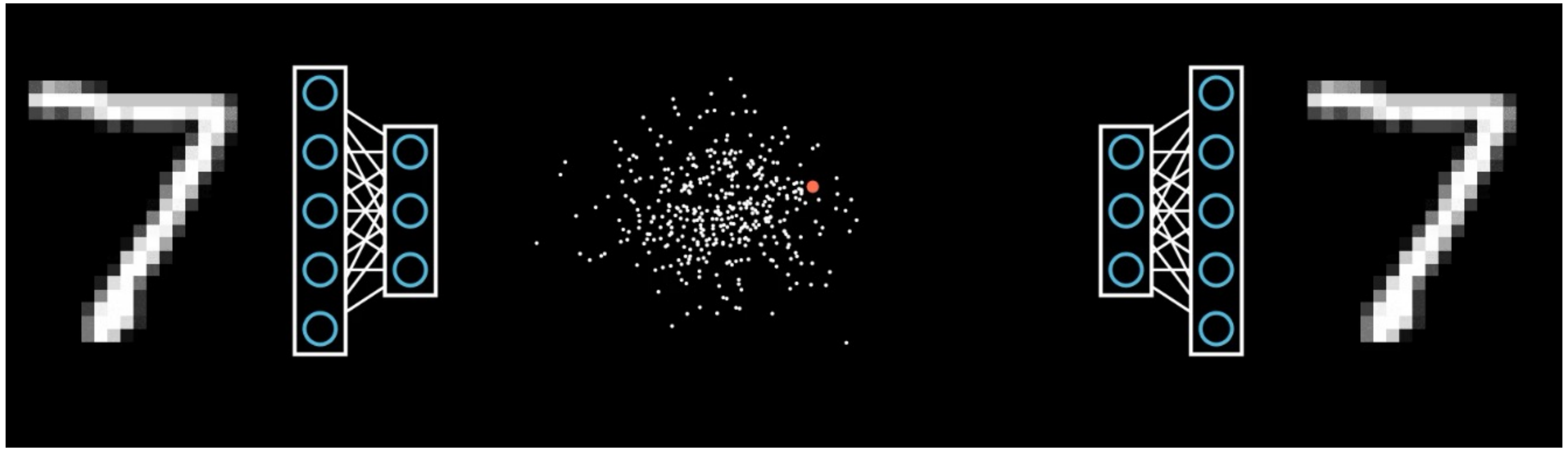
Variational Autoencoder

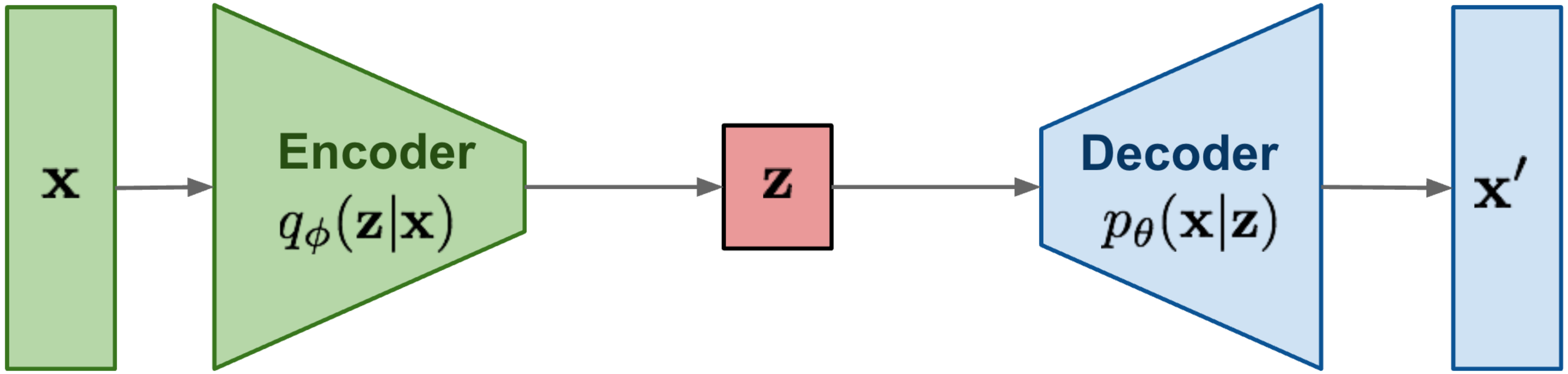
Enforce a simple latent distribution via an extra loss term



Variational Autoencoder

Enforce a simple latent distribution via an extra loss term

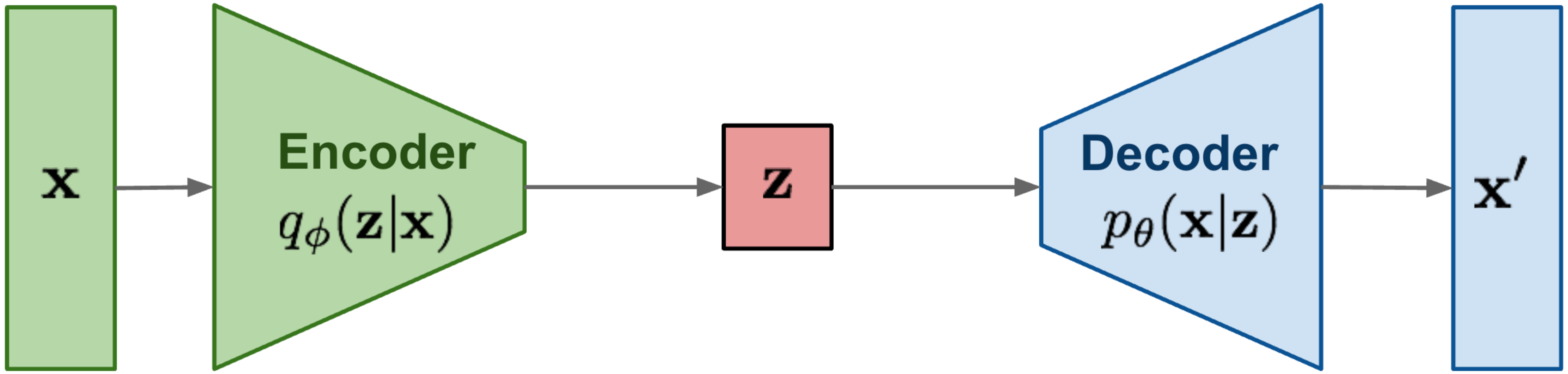




Variational Autoencoder

Enforce a simple latent distribution via an extra loss term

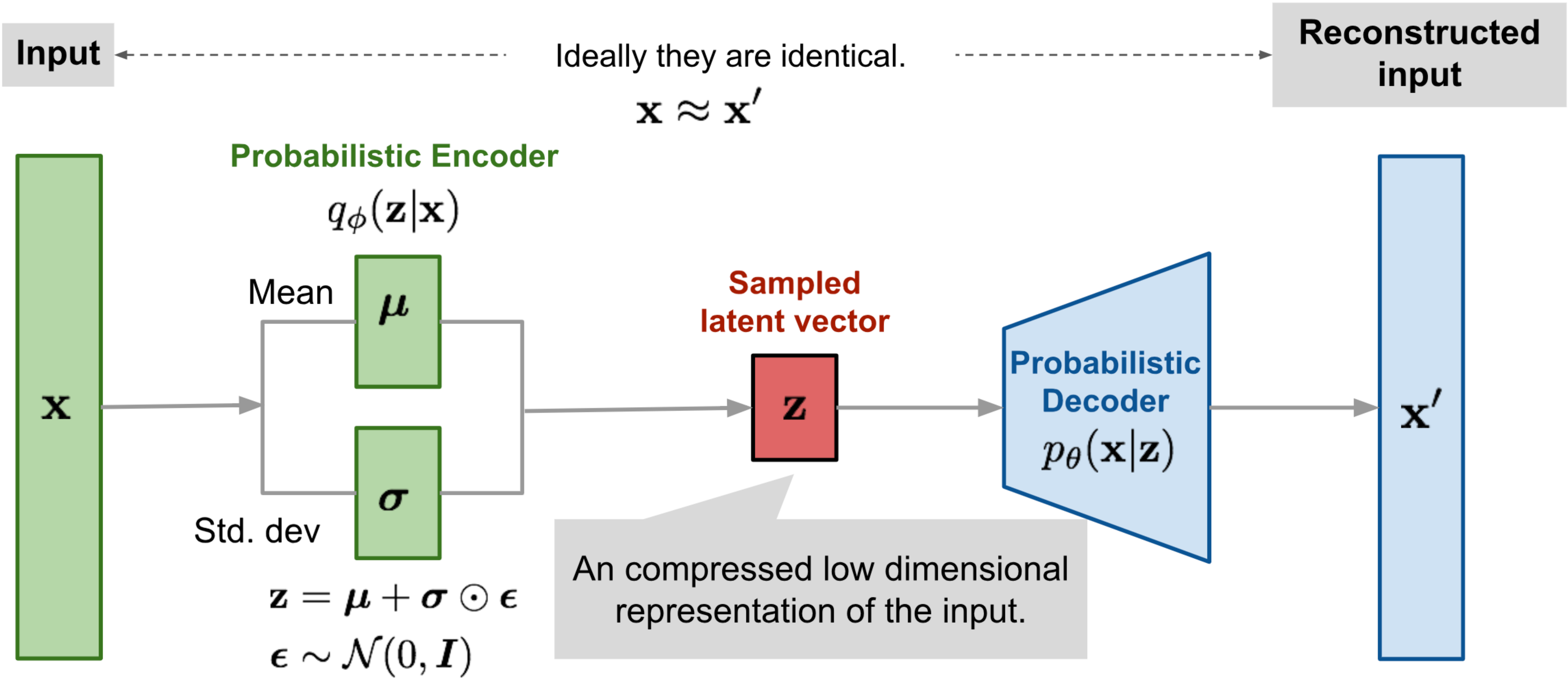
VAE: maximize
variational lower bound



$$
\begin{aligned}
L_{\mathrm{VAE}}(\theta, \phi) &= -\log p_\theta(\mathbf{x}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})) \\
&= -\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + D_{\mathrm{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) \\
\theta^*, \phi^* &= \arg\min_{\theta, \phi} L_{\mathrm{VAE}}
\end{aligned}
$$

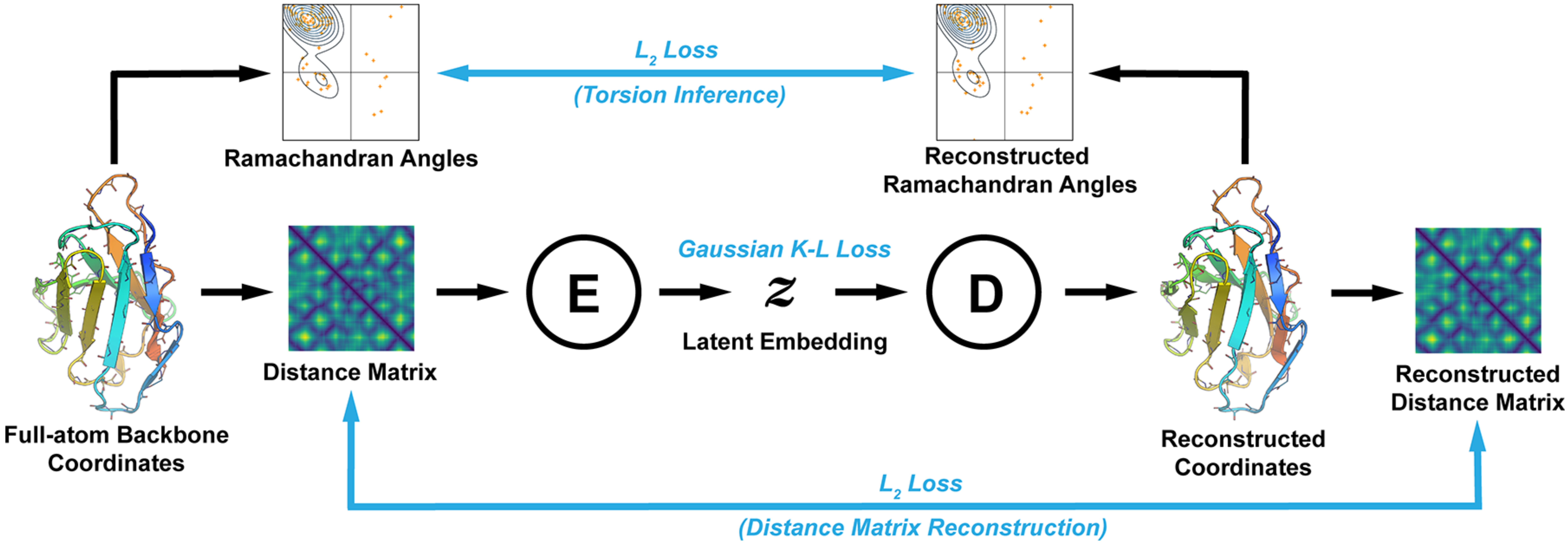
Variational Autoencoder

Enforce a simple latent distribution via an extra loss term



VAEs: Applications

Antibody Design (IgVAE)



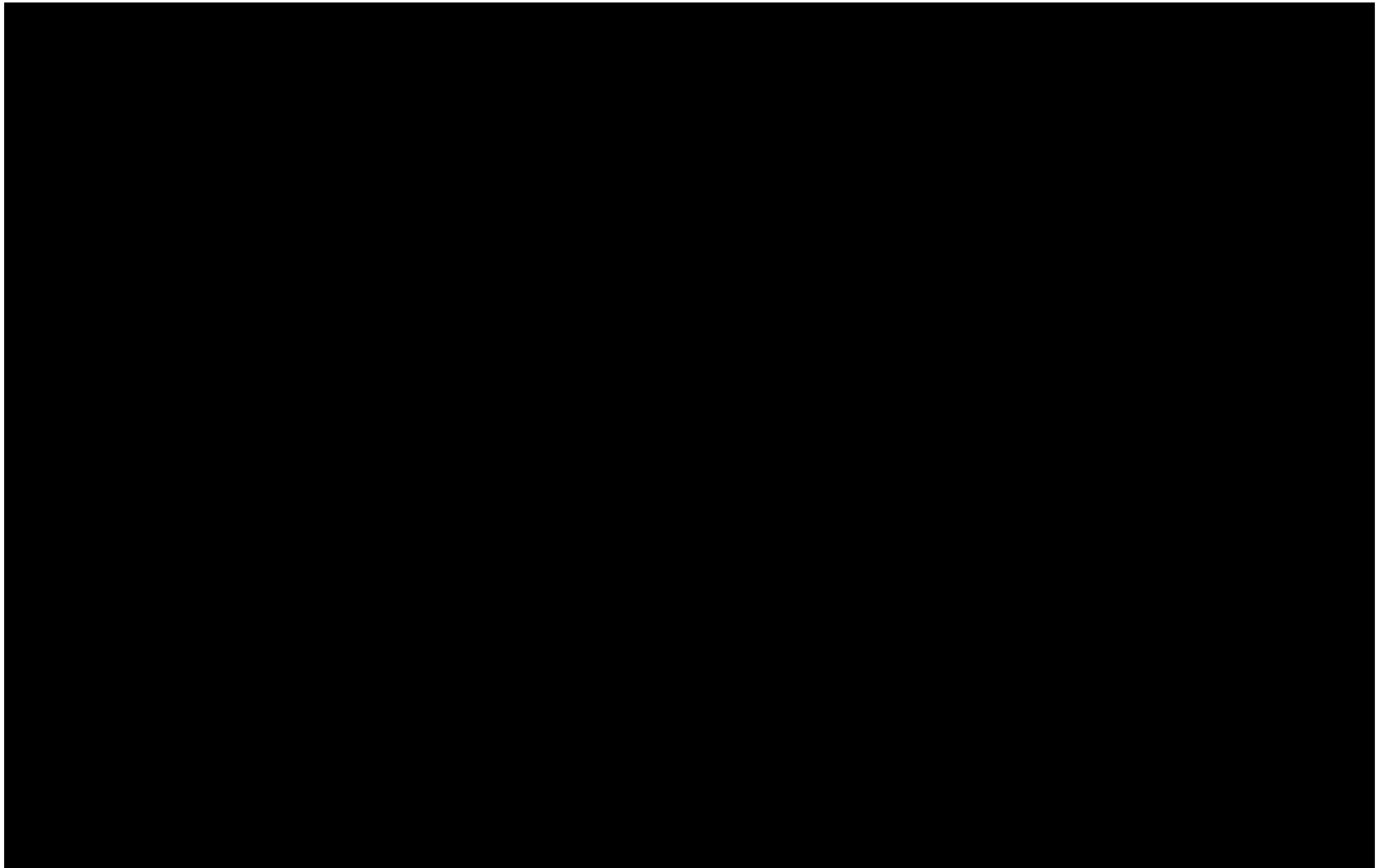
Eguchi, Raphael R., Christian A. Choe, and Po-Ssu Huang. "Ig-VAE: Generative modeling of protein structure by direct 3D coordinate generation." *PLoS computational biology* 18.6 (2022): e1010271.

3. Diffusion Models

“Creating noise from data is easy;
creating data from noise is generative modelling.”
— Yang Song

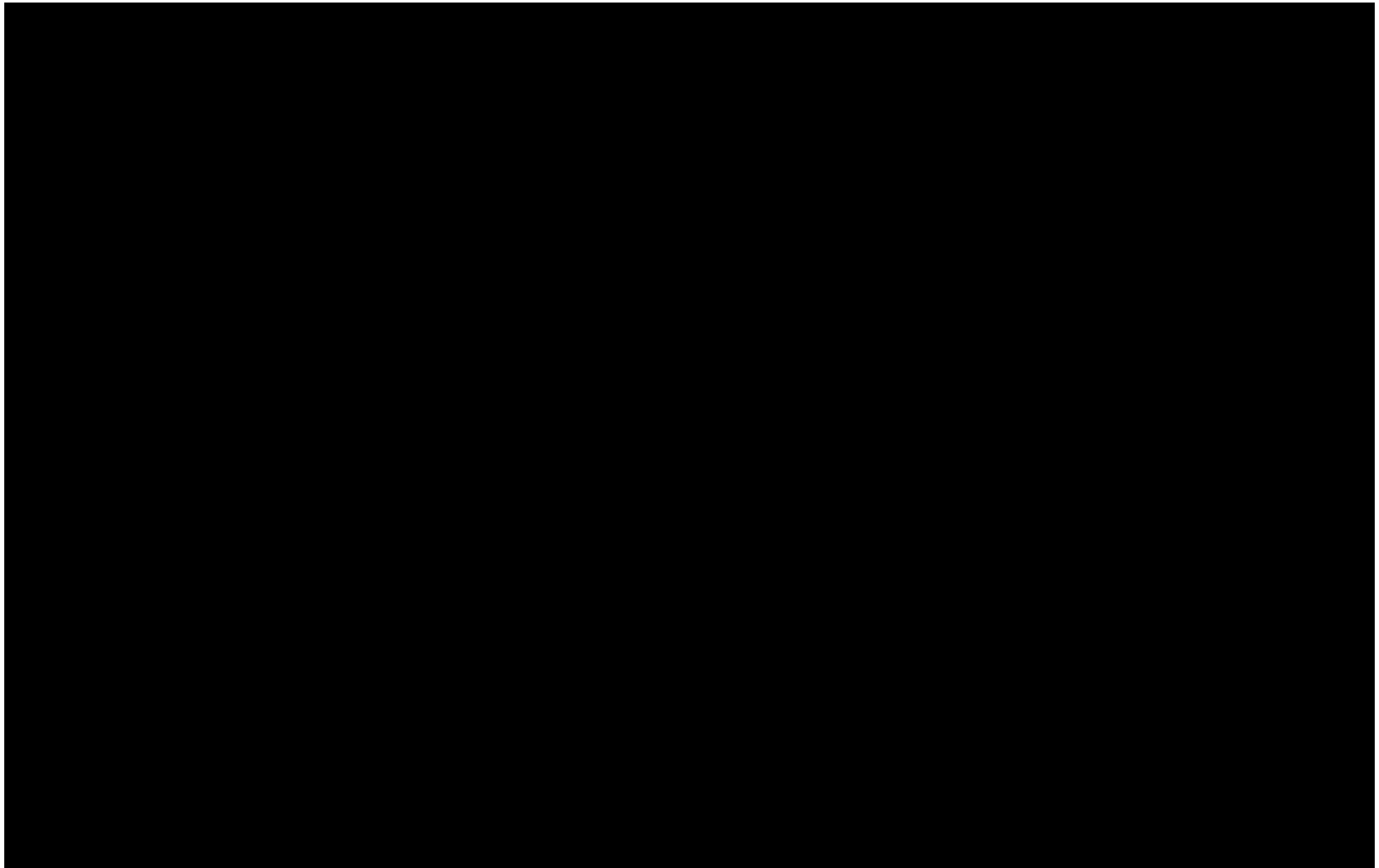
Diffusion Models

No bottleneck, but a sequence of noising steps



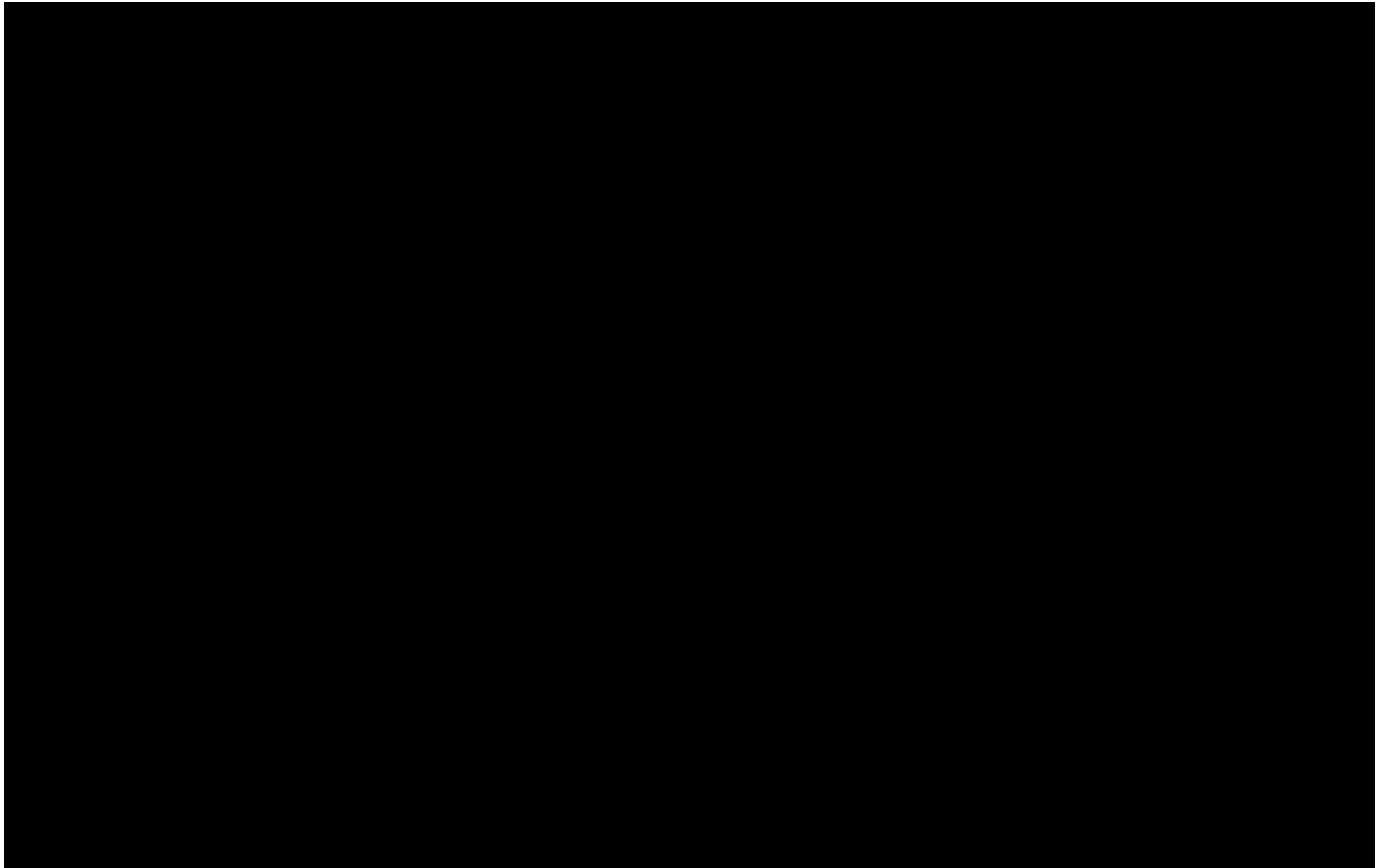
Diffusion Models

Task: given noise level and noised image, predict denoised image



Diffusion Models

Task: given noise level and noised image, predict denoised image



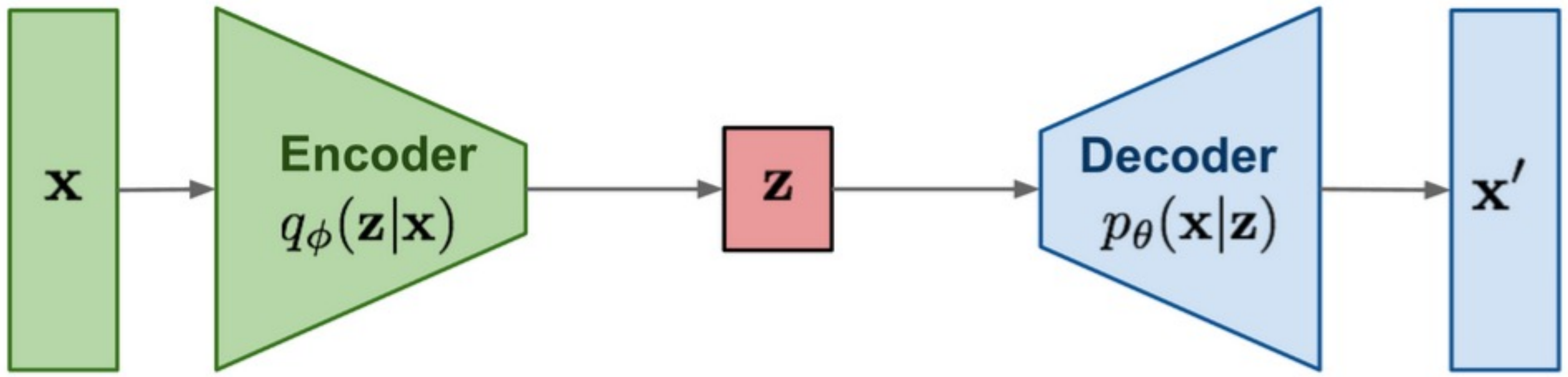
Diffusion Models

Level 1: Mapping noise back to data

Diffusion Models

Level 1: Mapping noise back to data

VAE: maximize
variational lower bound

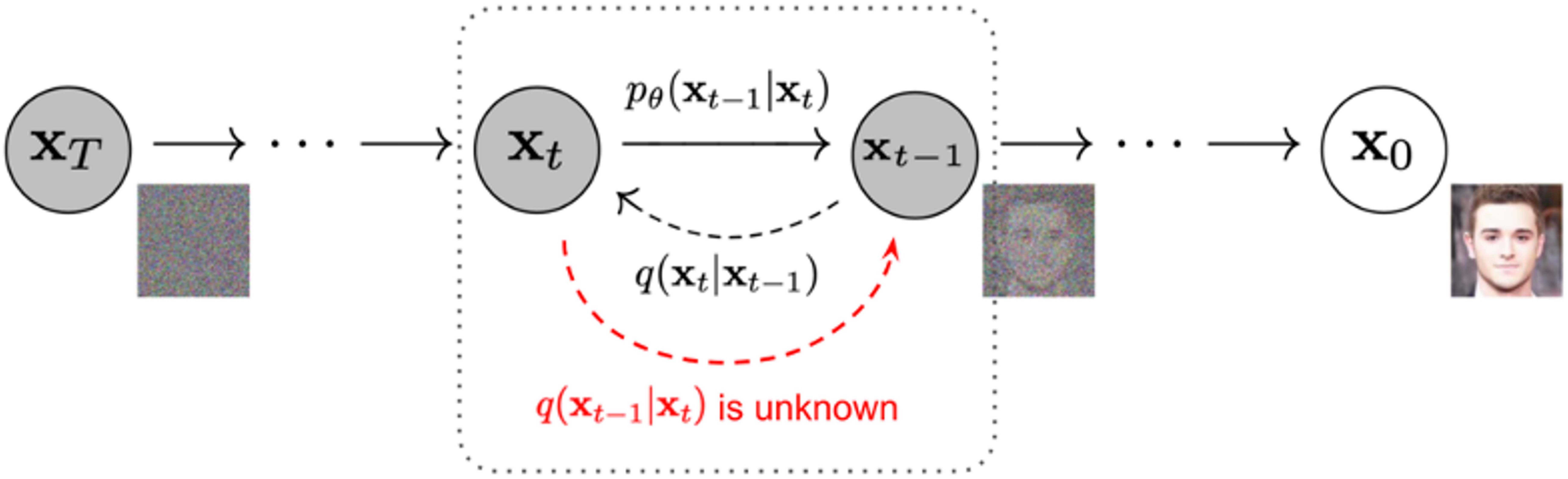


Diffusion models:
Gradually add Gaussian
noise and then reverse

lilianweng.github.io

Diffusion Models

Level 2: Each noising step is Gaussian

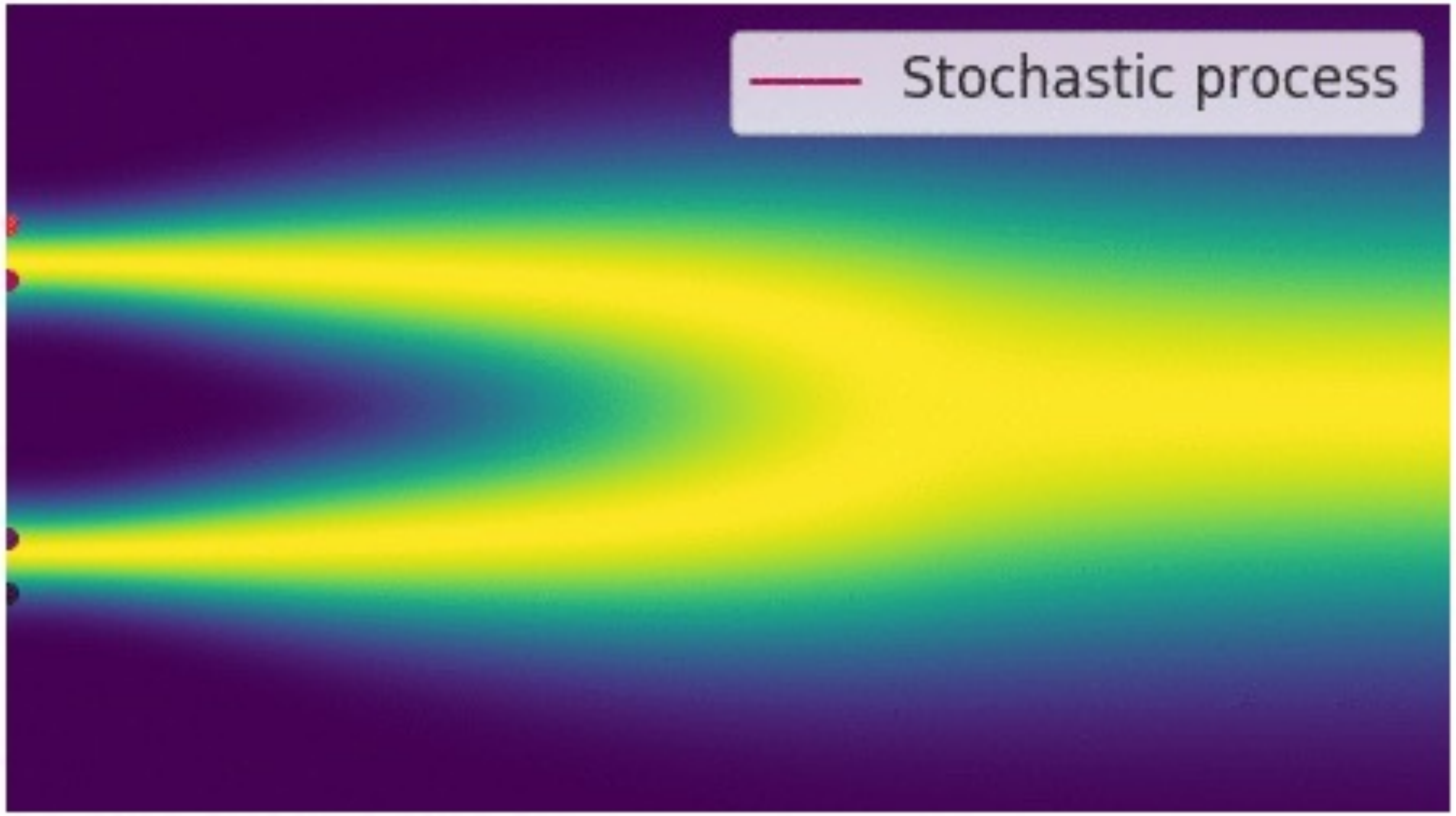


Forward Diffusion: $q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I})$

Reverse Diffusion: $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t))$

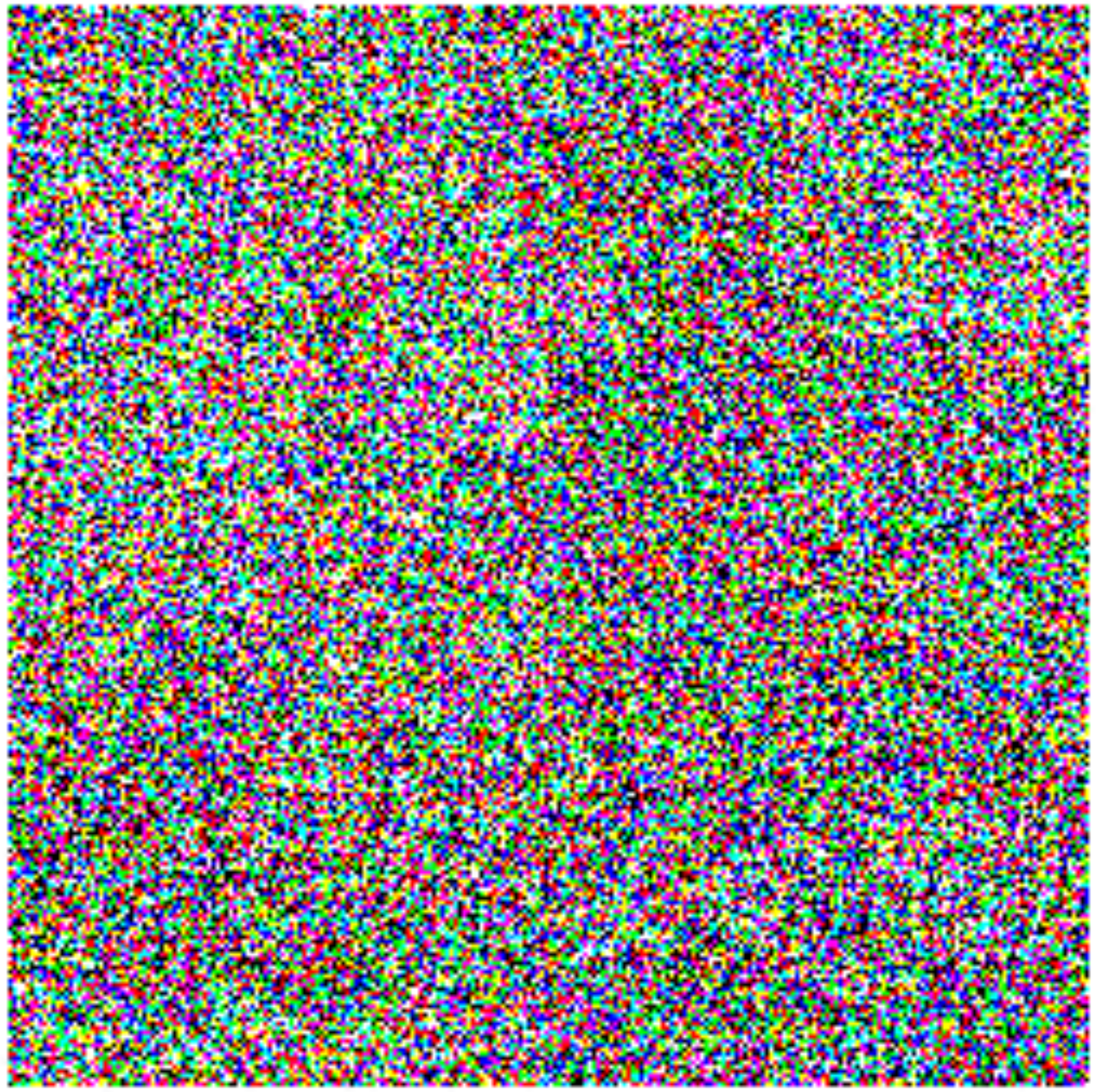
Diffusion Models

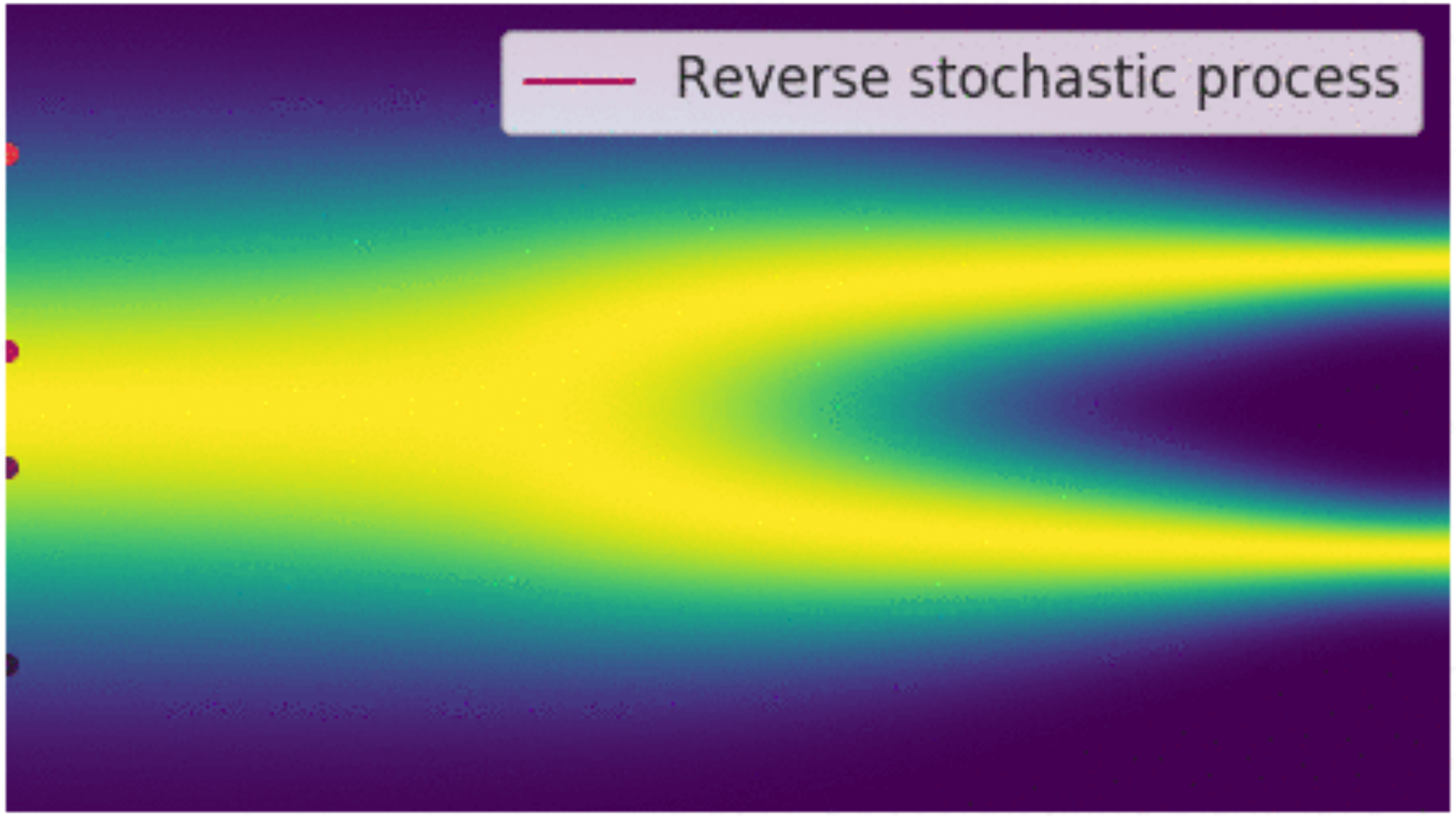
Forward Process = Noising to a reference distribution



Diffusion Models

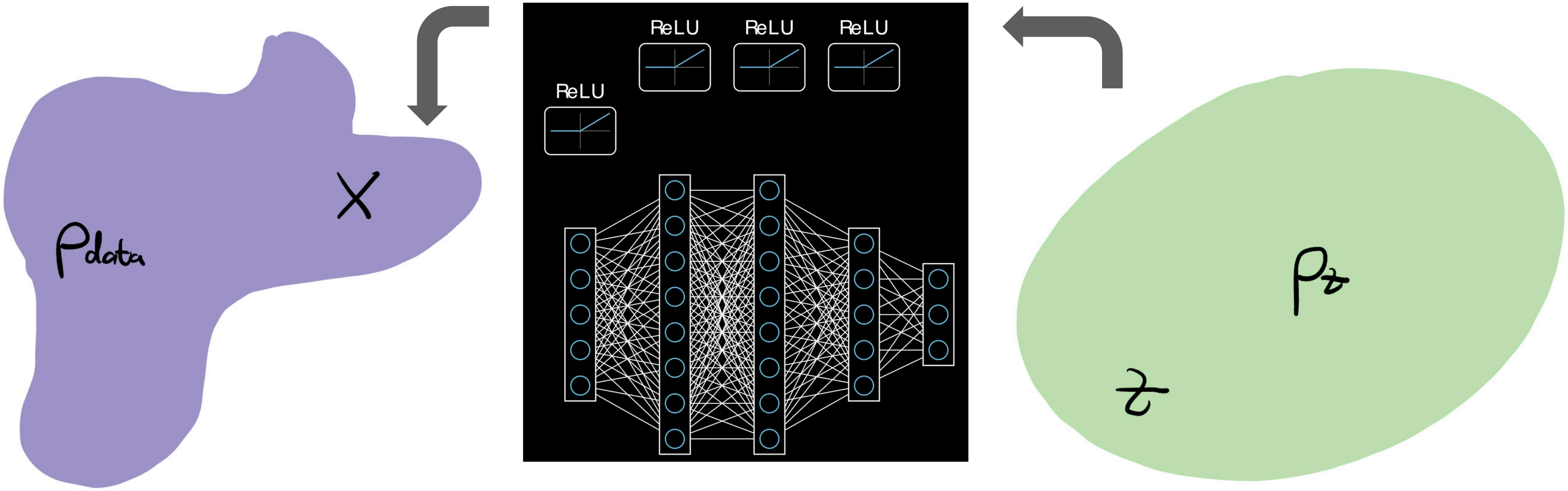
Reverse Process: Denoising to our target distribution





Diffusion Models

Reverse Process: Denoising to our target distribution



Loss Function of Diffusion Models

In theory: very similar to the VAE loss

$$L_{VLB} = \mathbb{E}_q[\log p_\theta(x_0|x_T)] - \sum_{t=1}^{T} D_{\mathrm{KL}}(q(x_t|x_0) \parallel p_\theta(x_t))$$

Intuition: Encourage the model to maximise the expected density applied to the data

Intuition: Encourage the learned posterior to be similar to the prior latent variable

Loss Function of Diffusion Models

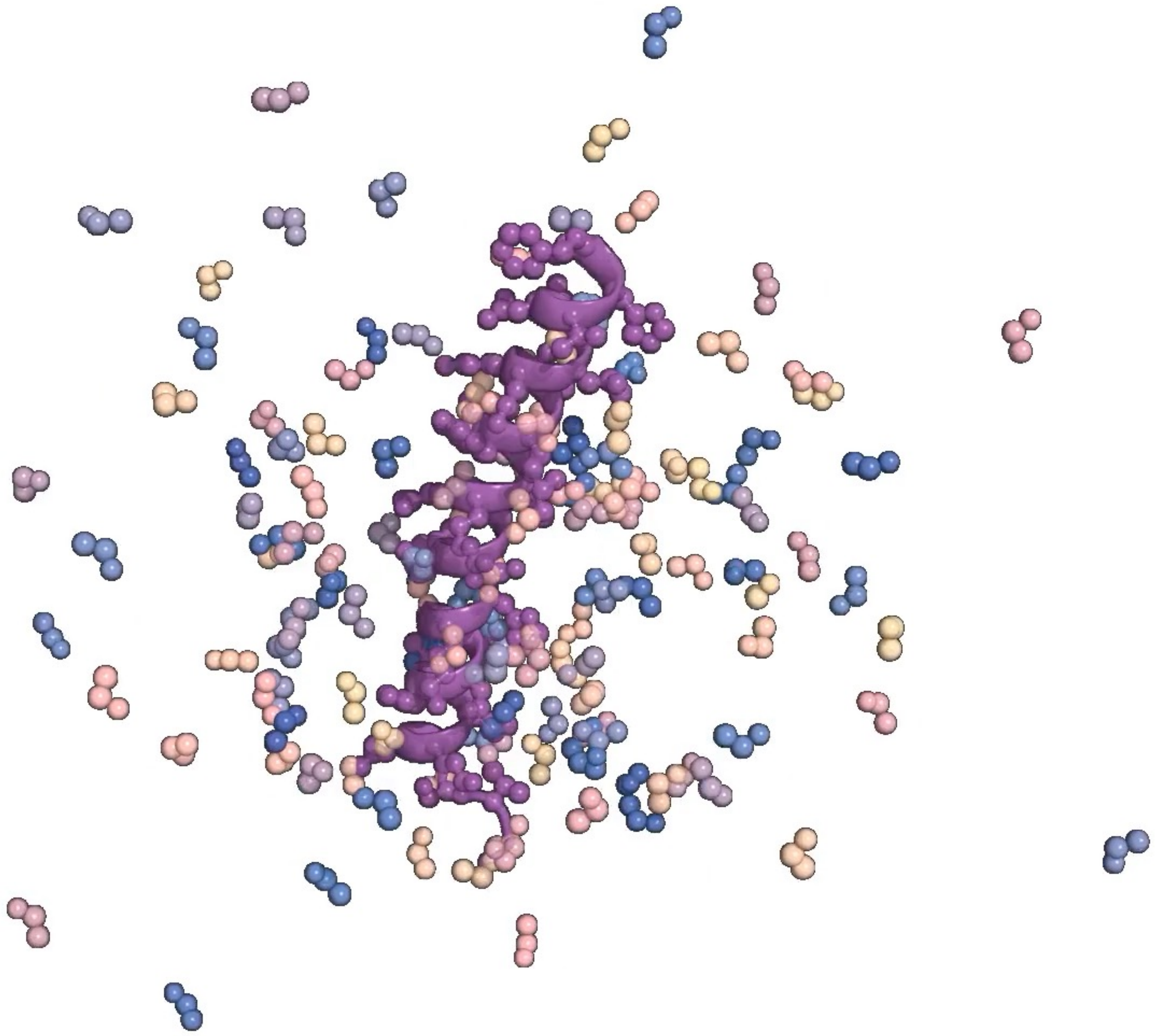
In practice: simpler objectives work better

$$L_t^{\text{simple}} = \mathbb{E}_{t\sim[1,T],\mathbf{x}_0,\boldsymbol{\epsilon}_t}\left[\|\boldsymbol{\epsilon}_t - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\right]$$

4. Applications and Outlook

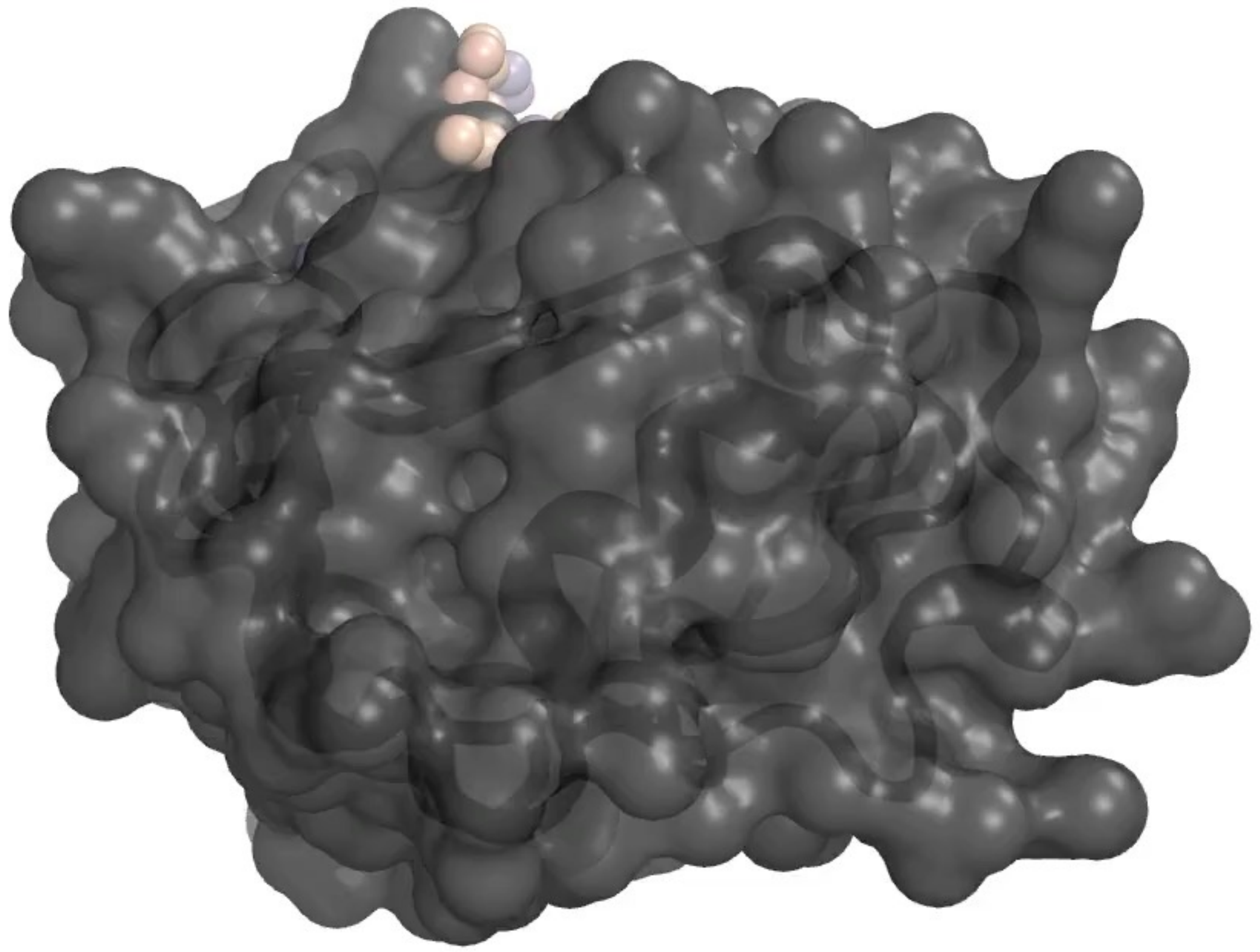
RFDiffusion

Designing new proteins



RFDiffusion

Designing new proteins



Takeaway

We can **condition** a generative backbone model such that a **pre-specified motif** is present, while **retaining realistic, novel** samples.