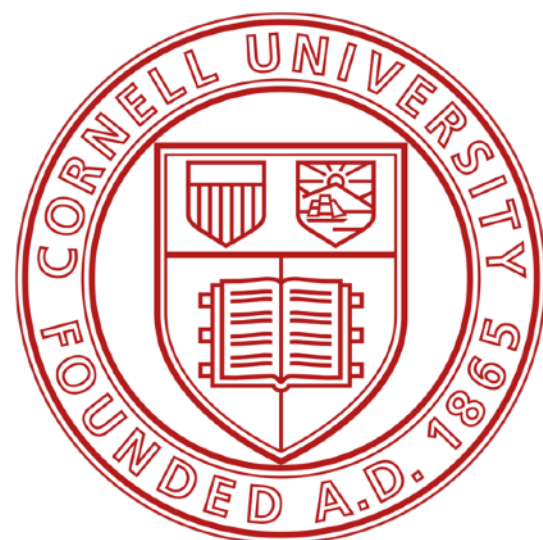


# Lecture 1: Introduction

CS 5788: Introduction to Generative Models



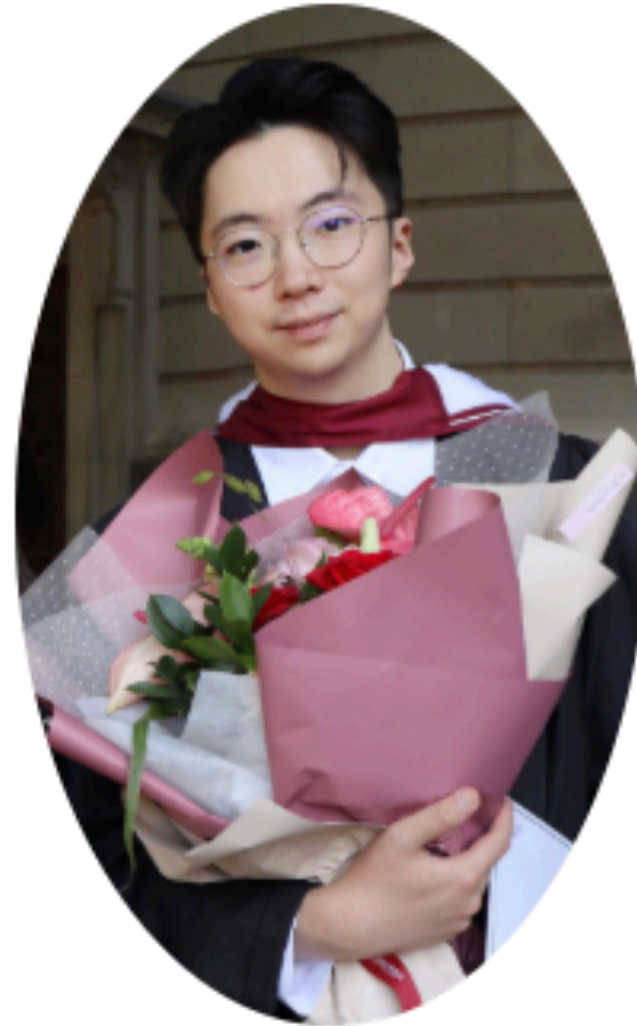
Includes material adapted from Stefano Ermon and Kaiming He

# Course staff



**Jeongsoo Park**

**TA**



**Zhaolin Gao**

**TA**



**Yiming Dou**

**TA**



**Adnan Armouti**

**TA**



# Course policies

# Interacting with us

- In person office hours
- Ask homework and class questions on Ed Discussion
- Homework submission via Gradescope





# Course website

CS 5788: Introduction to Generative Models

Instructor: Andrew Owens   Spring 2026

Schedule

Staff

Course info

Ed Discussion

Canvas

Gradescope

Recordings

Tentative Schedule

Lecture	Date	Topic	Materials	Assignments
Lec. 1	Tue, Jan. 20	<b>Introduction</b> Course policies What is generative modeling?		optional review of PyTorch and probability <u>ps1 out (simple probabilistic models)</u>
Lec. 2	Thu, Jan. 22	<b>Maximum likelihood</b> Gaussian distribution Maximum likelihood		
Lec. 3	Tue, Jan 27	<b>Gaussian mixture models</b> Stochastic gradient descent Gaussian mixture models Variational inference		
Lec. 4	Thu, Jan 29	<b>Neural network review (recorded lecture)</b> Transformers Image translation networks Backpropagation		
Lec. 5	Tue, Feb 3	<b>Variational autoencoders</b> Autoencoders		

<https://www.cs.cornell.edu/courses/cs5788/2026sp>

# Lectures

- Lecture recordings available on Canvas (see link on course webpage). Sometimes there is a delay in releasing them.
- Next Thursday's lecture will be recorded (it's a neural net review).
- Some guest lectures (by TAs and other faculty) in middle of course



# What you'll be doing

- Assignments (40%)
- Midterm exam (30%)
- Final project (30%)

# Assignments

- There will be 4 assignments
- Equally weighted
- Mix of programming and theoretical questions
- PS1 available this week. Covers material from first 3 lectures.



# Assignments

- PS1 out later this week
- You'll have **120 late hours** (5 late days)
  - Distribute any way as you want, charged by the hour.
  - No need to notify us when you use them.
  - Once they're used up, 1% penalty for that homework per hour.
  - If you edit your assignment after the deadline, this will count as a late submission.
- Assignments be done independently. It's OK to discuss them with classmates, but programming/writing should all be yours.
- Unless stated otherwise, LLMs only allowed for testing code and as an API reference (the same way that you would use Stack Overflow).

# Midterm exam

- Will be scheduled in early April (exact date pending, approximately Friday, April 10)
- In person, one-page front and back hand-written “cheat sheet” allowed with formulas, notes, etc.
- You’ll have 3 hours.
- If you have accommodation requests, let the course staff know as soon as possible.



# GPU computing

- Problem sets require GPUs
- GPUs are very expensive.
- Recommend using Google Colab
  - Colab Pro sometimes available for free to Cornell students (though not necessary!)
- Let us know if you want Pro, can't get it for free or afford it, and we can see what we can do.
- Make sure to start early to avoid usage limits

# Final project

- Choose from a list of project topics.
- Can also pitch your own project idea for us to approve, e.g.:
  - Implement a recent paper
  - A research project
- Small groups (1 to 3 people) suggested.
  - Under exceptional circumstances, we can consider larger groups
- Deliverables:
  - Project proposal midway through semester
  - Short in-class presentation (last week of class)
  - Writeup (due during finals period)

# Background

- We assume you've taken an "machine learning adjacent" class!
  - E.g., deep learning, machine learning, NLP, computer vision, etc.
- Python numerical computing and programming
- PyTorch knowledge strongly encouraged
- Numpy and PyTorch refreshers on website
- Linear algebra, multi-variable calculus, probability
  - Math review on webpage.

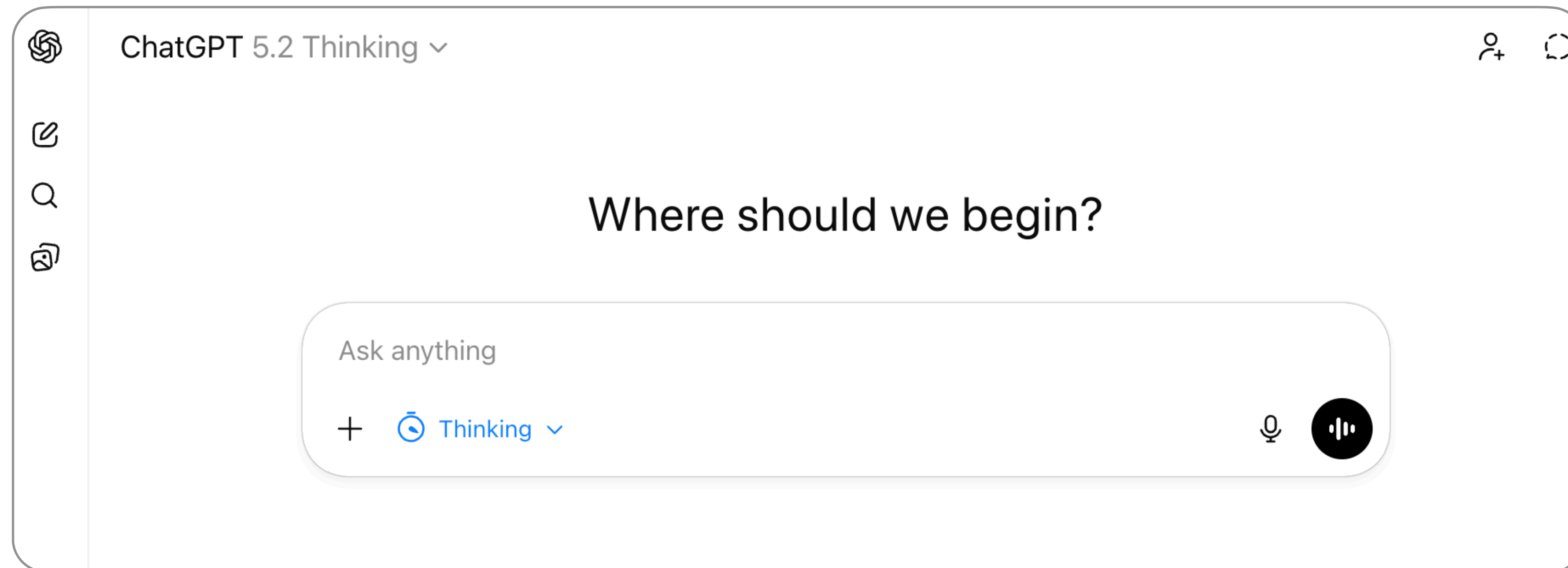
Questions?

# Today

1. Class logistics
- 2. Generative models today**
3. What are generative models?
4. What makes generative modeling hard?

Generative models are everywhere

# Language





# Generating and editing images



Generated image



Edited image

Source: [FLUX.1 Kontext, 2024]



# Speech





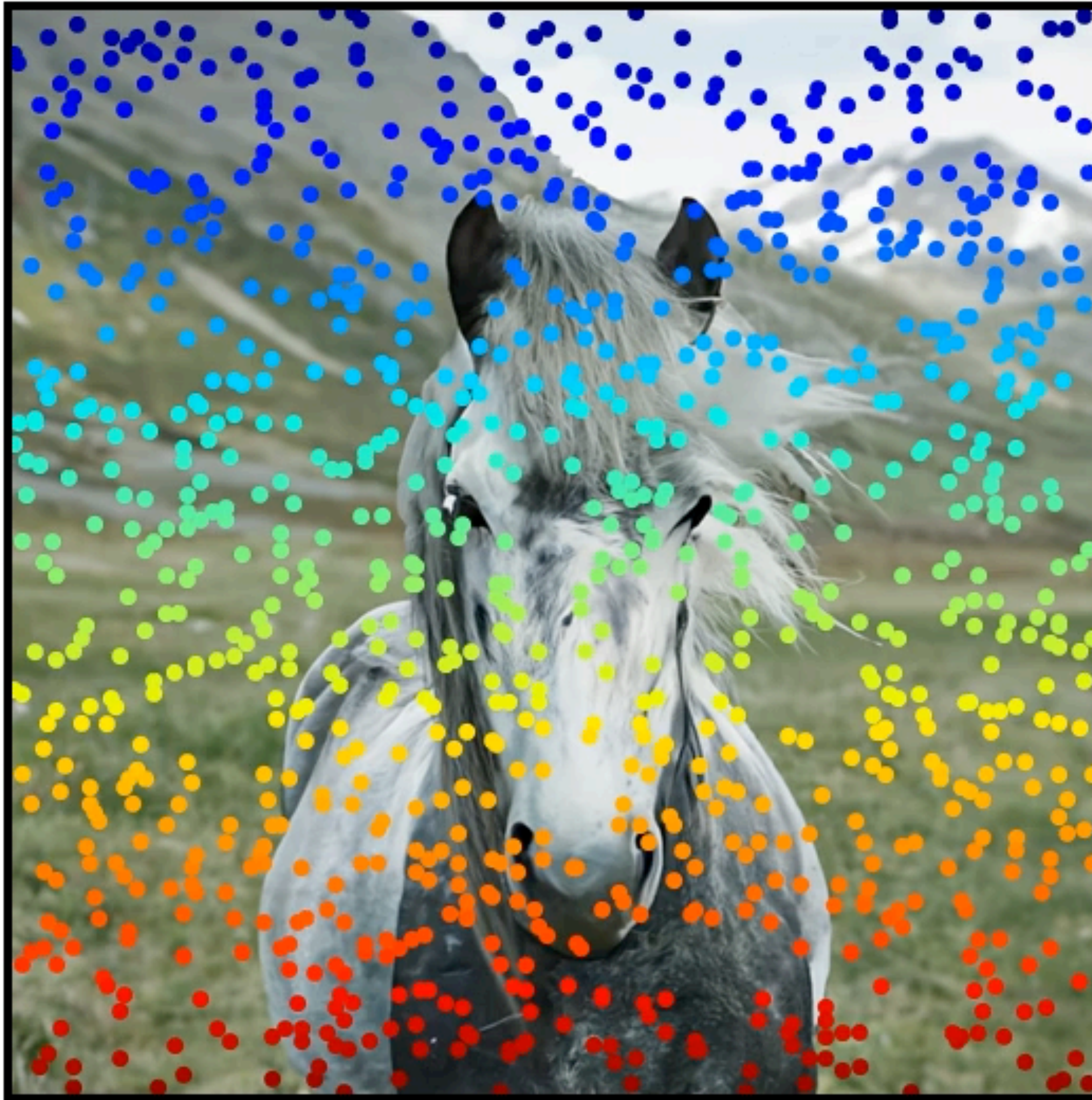
# Video



"The scene explodes with the raw, visceral, and unpredictable energy of a hardcore off-road rally, captured with a dynamic, almost found-footage or embedded sport...", Veo 3.



# User-controllable generation



Source: [Geng et al., "Motion Prompting", CVPR 2025]



# Multimodal models



PM This is a picture of two teddy bears on the moon.

What are they doing?

PM They are having a conversation.

What object are they using?

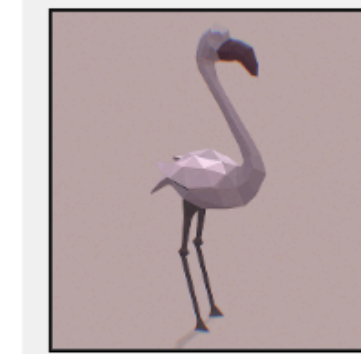
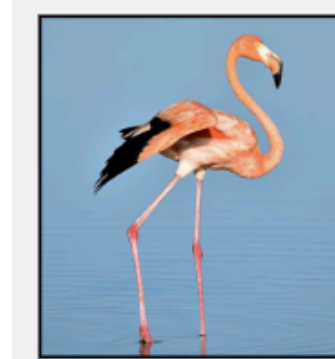
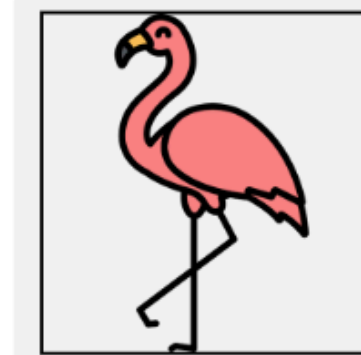
PM It looks like a computer.

Is this surprising?

PM Yes, it is surprising.

Why is this picture surprising to you?

PM I think it is surprising because teddy bears are not usually found on the moon.



What is the common thing about these three images?

PM They are all flamingos.

What is the difference between these three images?

PM The first one is a cartoon, the second one is a real flamingo, and the third one is a 3D model of a flamingo.



PM This is an apple with a sticker on it.

What does the sticker say?

PM The sticker says "iPod".

Where is the photo taken?

PM It looks like it's taken in a backyard.

Do you think it is printed or handwritten?

PM It looks like it's handwritten.

What color is the sticker?

PM It's white.



# Adapting generative models to other tasks

---

Question: If  $a > 1$ , then the sum of the real solutions of  $\sqrt{a - \sqrt{a + x}} = x$  is equal to

---

Response: <think>

To solve the equation  $\sqrt{a - \sqrt{a + x}} = x$ , let's start by squaring both  $\dots$

$$\left(\sqrt{a - \sqrt{a + x}}\right)^2 = x^2 \implies a - \sqrt{a + x} = x^2.$$

Rearrange to isolate the inner square root term:

$$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$$

$\dots$

**Wait, wait. Wait. That's an aha moment I can flag here.**

Let's reevaluate this step-by-step to identify if the correct sum can be  $\dots$

We started with the equation:

$$\sqrt{a - \sqrt{a + x}} = x$$

First, let's square both sides:

$$a - \sqrt{a + x} = x^2 \implies \sqrt{a + x} = a - x^2$$

Next, I could square both sides again, treating the equation:  $\dots$

$\dots$

---

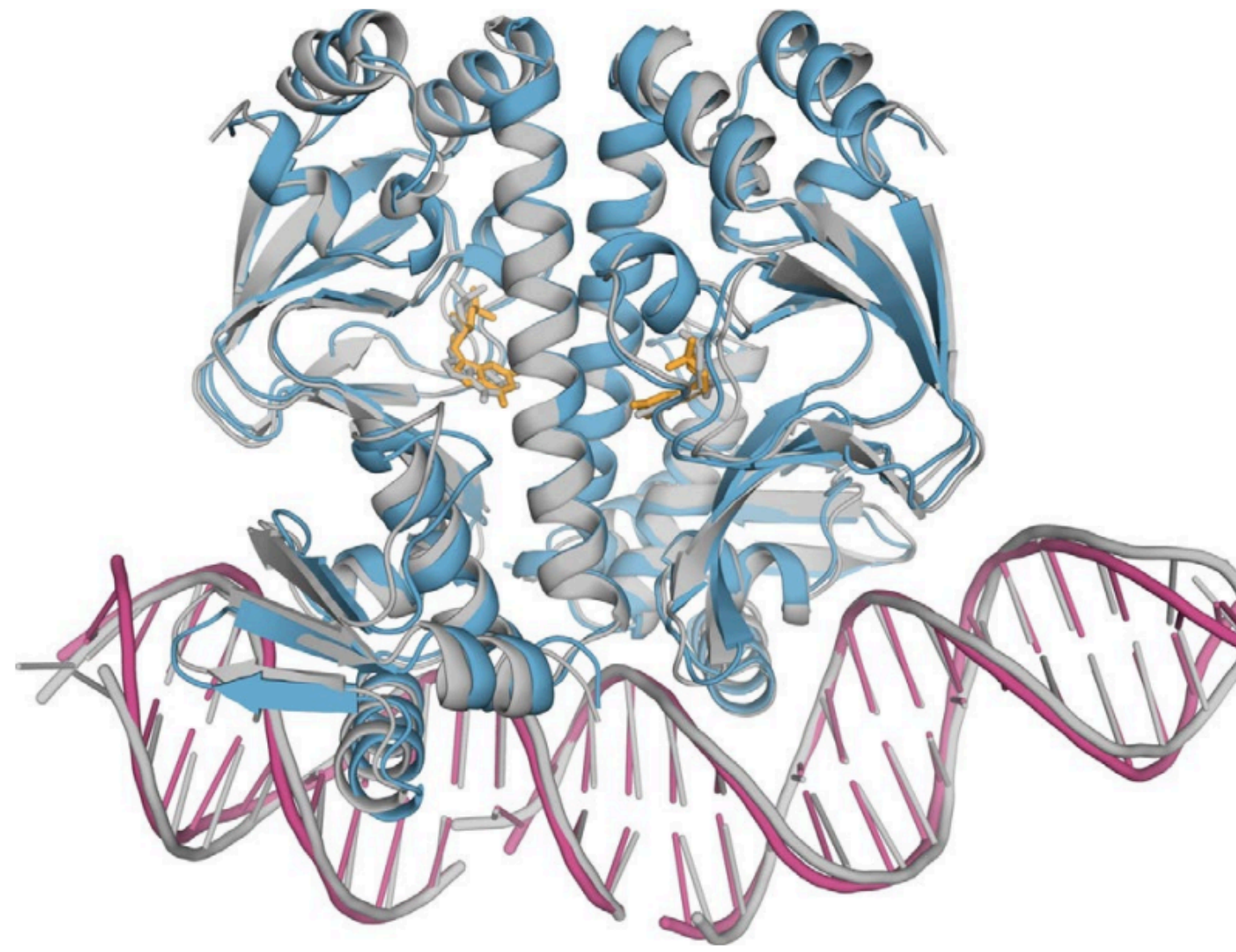


Reasoning and math [DeepSeek "R1"]

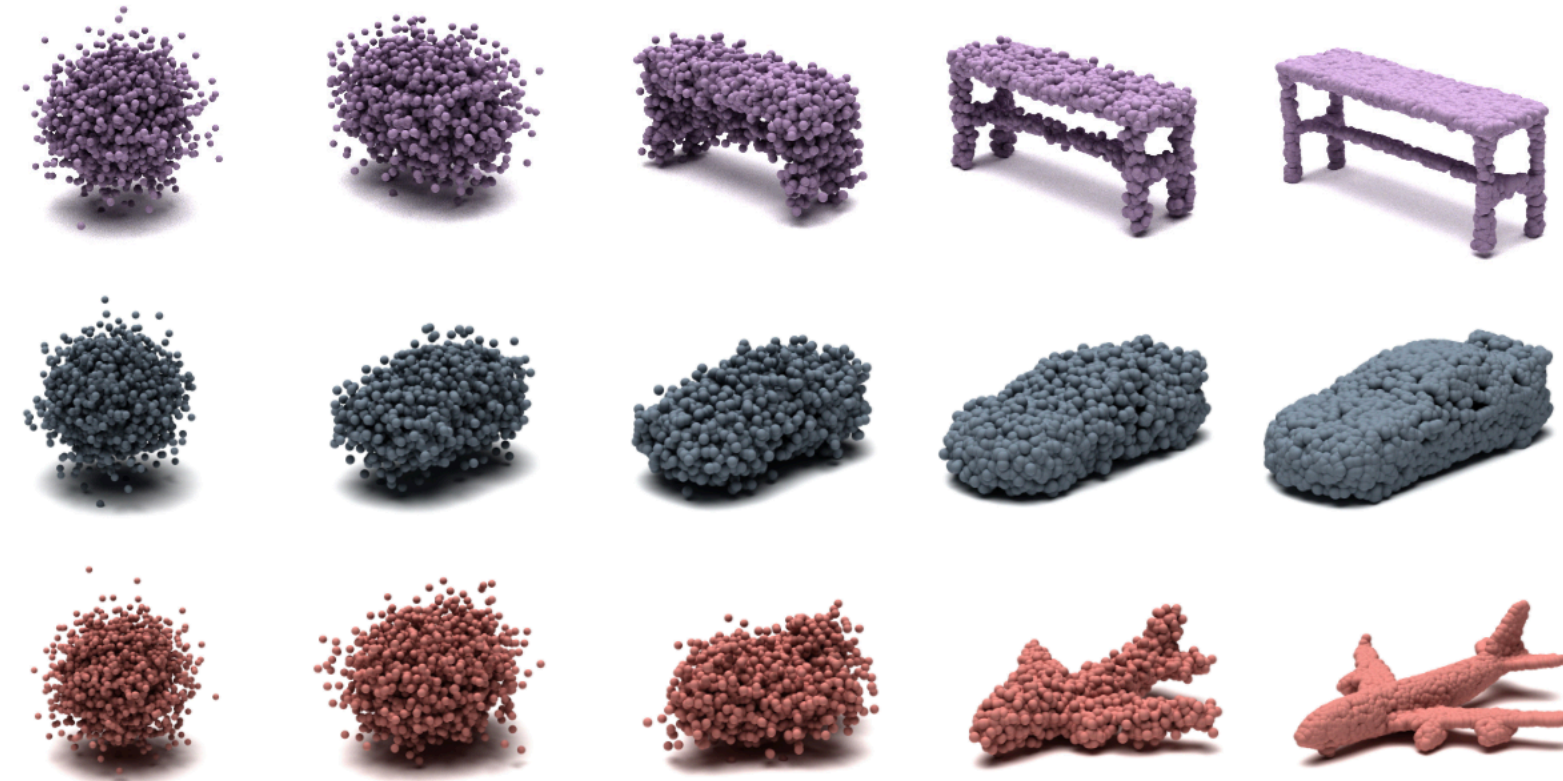
3D video generation [Wu et al., "CAT4D", 2025]



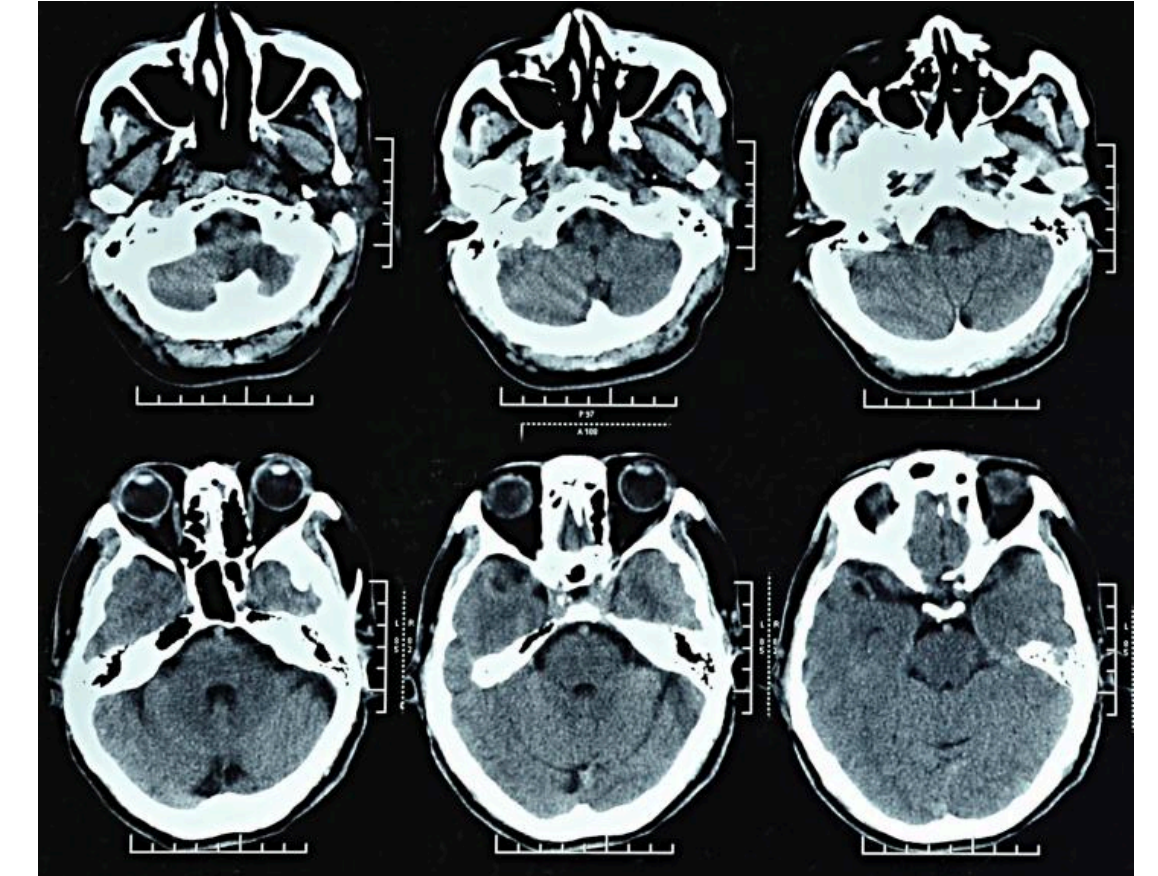
# And many other domains...



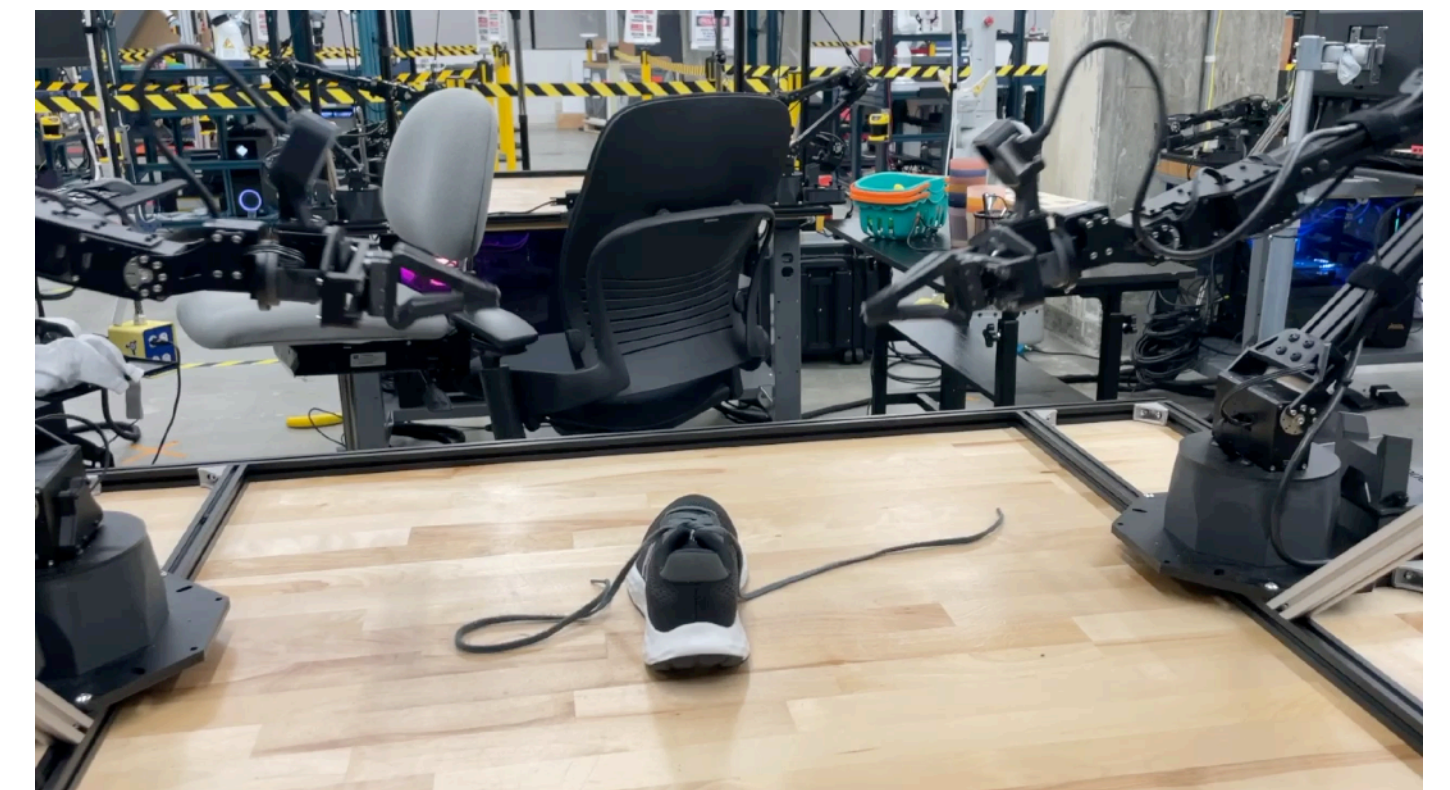
Proteins [Abramson et al.,  
"AlphaFold3", 2024]



3D shapes  
[Zhou et al., 2021]



Medical images  
[Song, Shen et al., 2022]



Robotics  
[Zhao et al., 2024]



What applications do you have in mind?

# Today

1. Class logistics
2. Generative models today
- 3. What are generative models?**
4. What makes generative modeling hard?

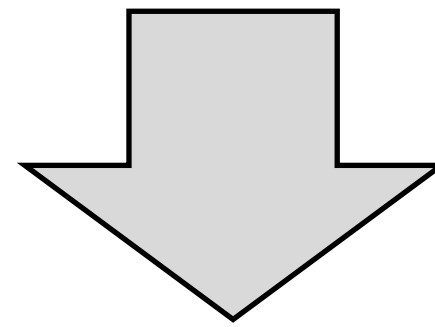
# Rule-based (or physical) models for generating data

Generating images using traditional computer graphics:

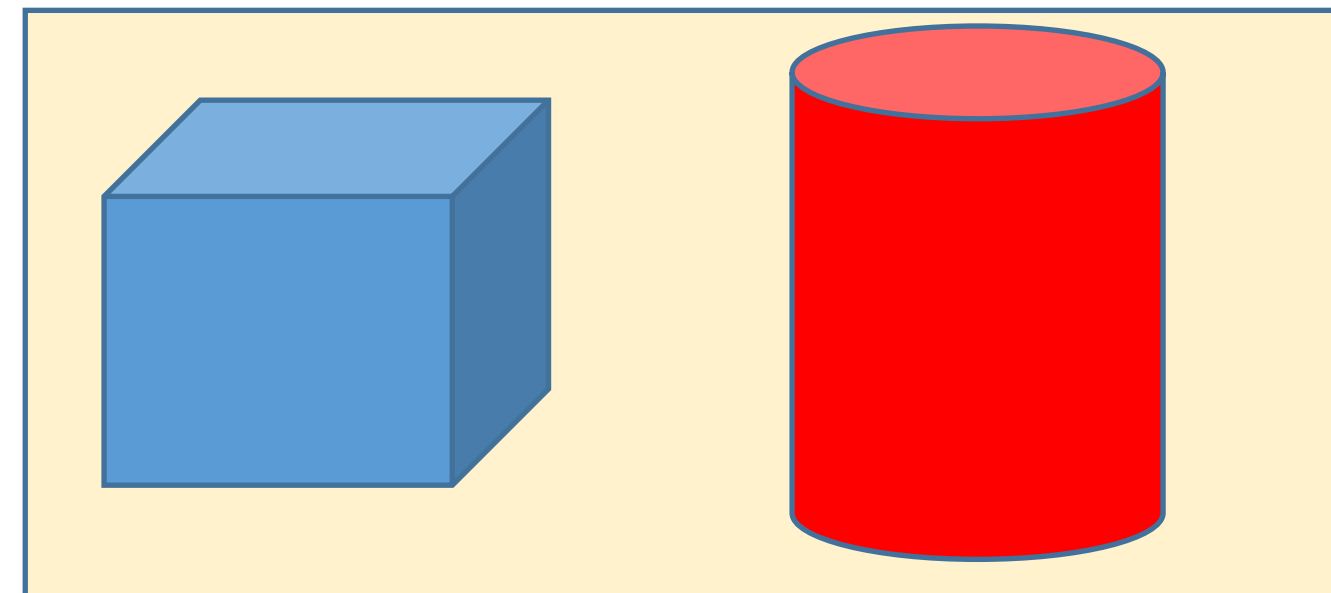
High level  
description

```
Cube(color=blue, position=(x,y,z), size=...)  
Cylinder(color=red, position=(x',y',z'), size=..)
```

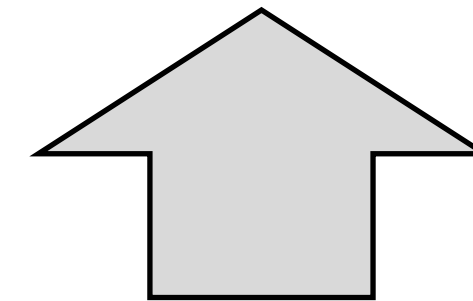
**Generation** (graphics)



Raw sensory  
outputs

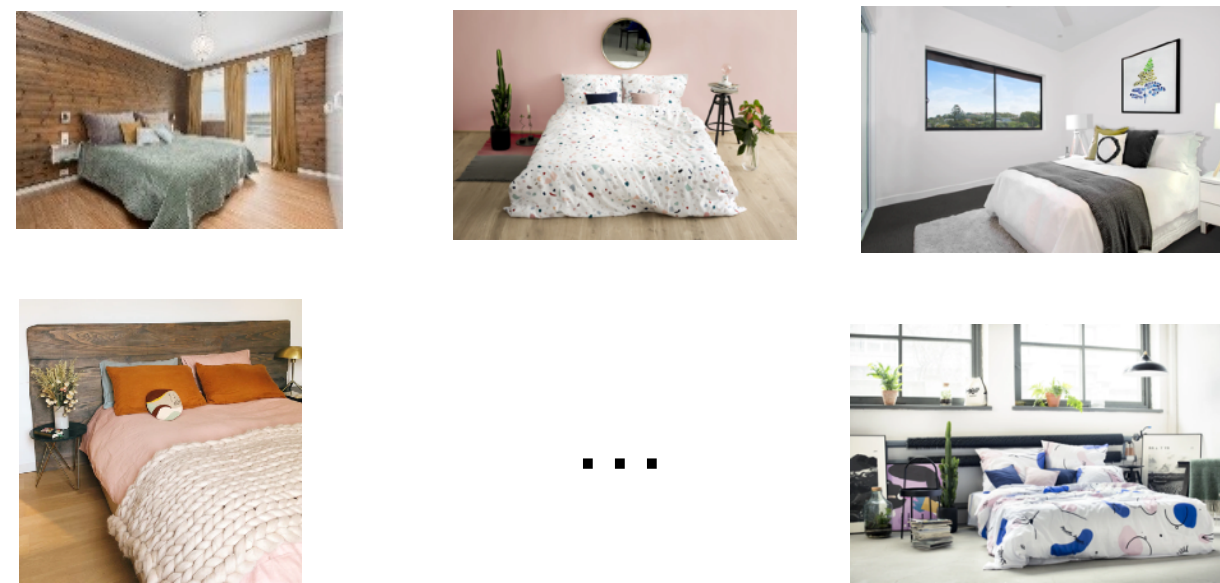


**Inference** (vision as  
inverse graphics)



# Statistical generative models

Statistical generative models are learned from data!



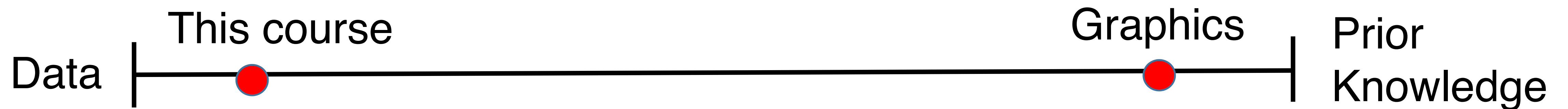
Data  
(e.g., images of bedrooms)

+



Prior Knowledge  
(e.g., physics, materials, ..)

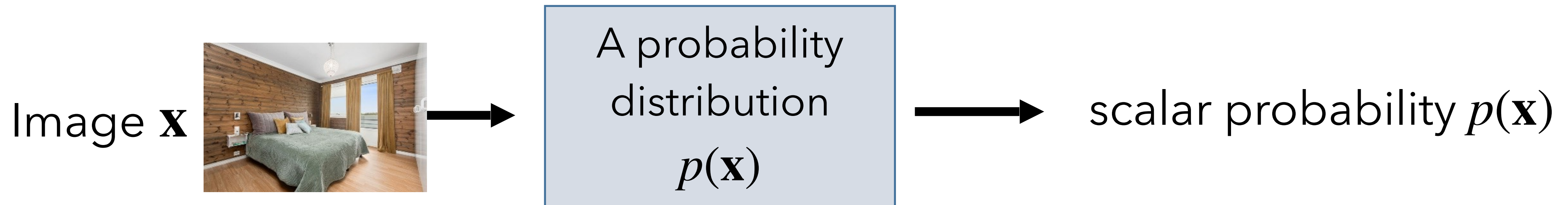
Priors are always necessary, but there is a spectrum.



# Statistical generative models

A statistical generative model is a **probability distribution**  $p(\mathbf{x})$

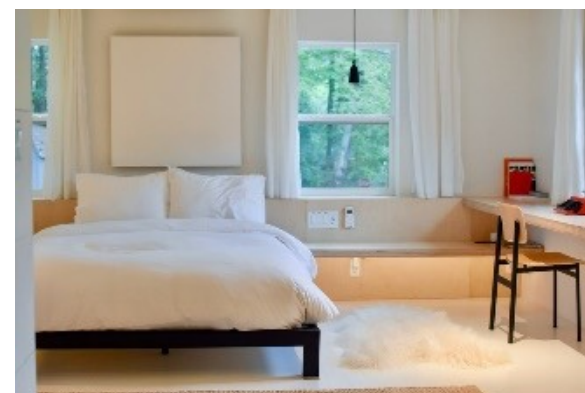
- **Data:** samples (e.g., images of bedrooms)
- **Prior knowledge:** parametric form (e.g., Gaussian?), loss function (e.g., maximum likelihood?), optimization algorithm, etc.



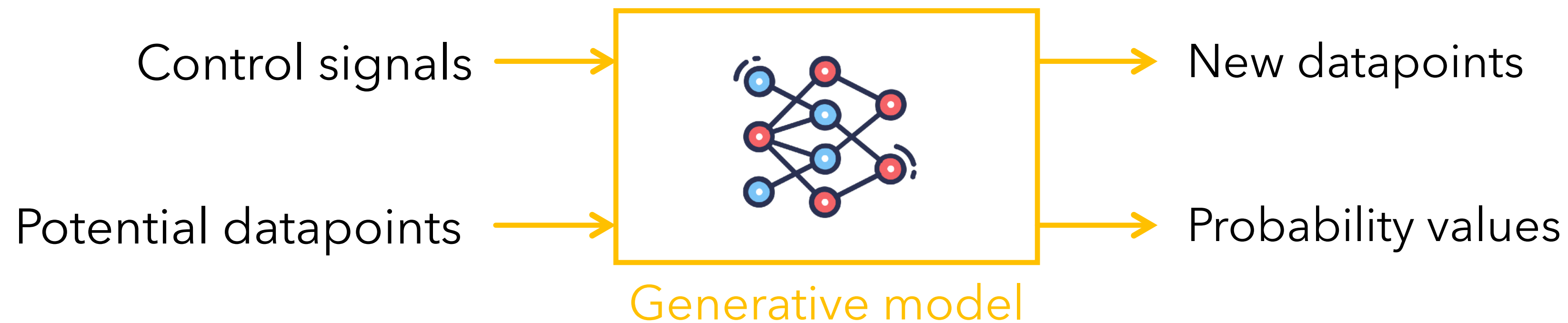
It is generative because sampling from  $p(\mathbf{x})$  generates new images



...



# Some goals of generative modeling



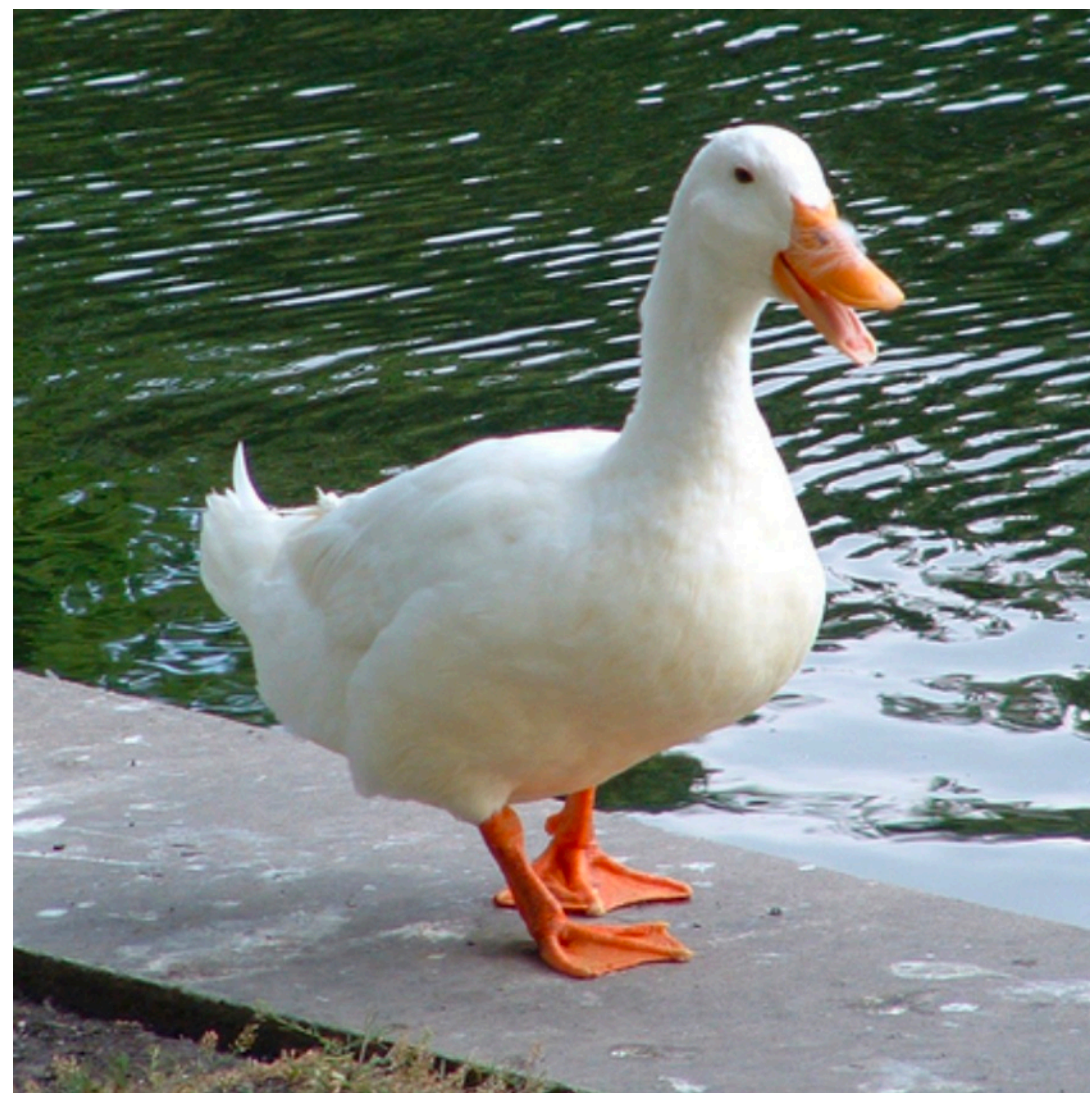
Some of the things we'd like from a generative model:

- A "data simulator": generating data as an end in itself.
- Get probability density estimates from  $p(\mathbf{x})$ .
- Adapt the underlying model for other tasks.



# Discriminative vs. generative models

Probabilistic interpretation  $p(y \mid \mathbf{x})$



⋮

image  $\mathbf{x}$



**Discriminative  
model**

$p(y \mid \mathbf{x})$



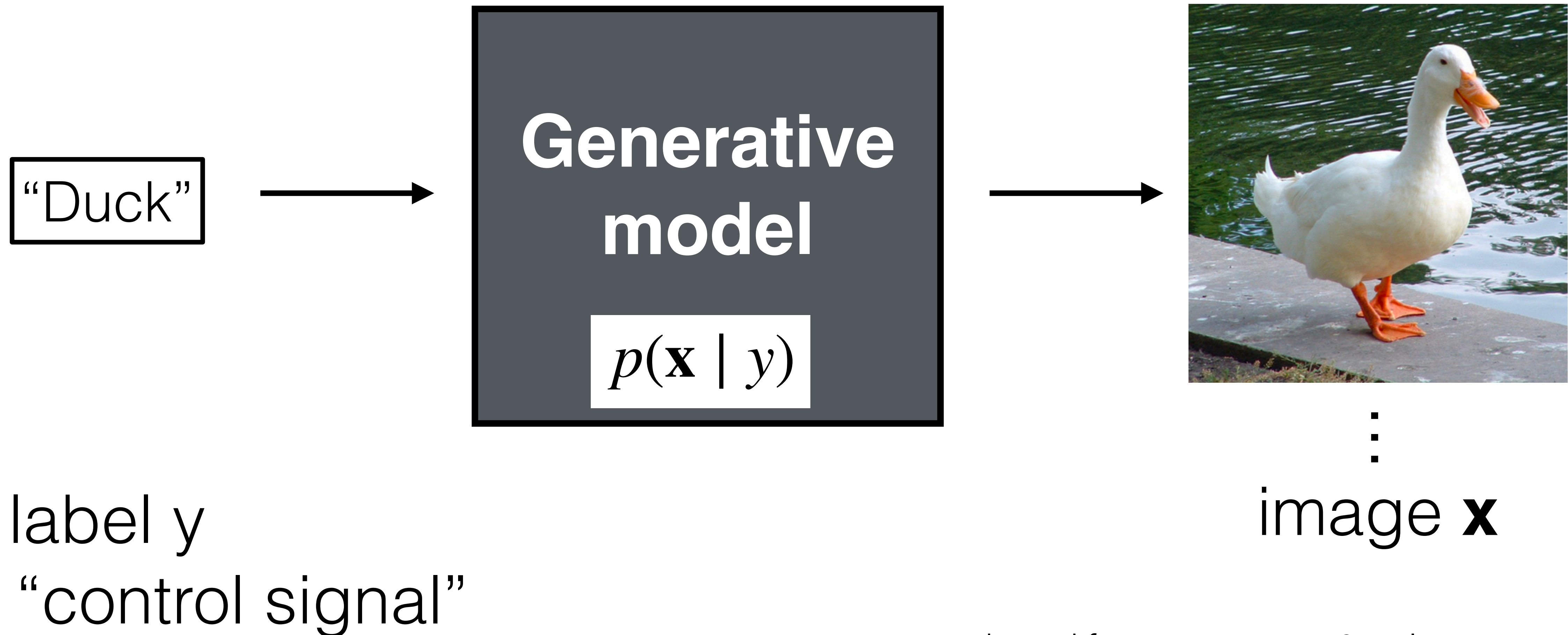
“Duck”

label  $y$



# Discriminative vs. generative models


Probabilistic interpretation  $p(\mathbf{x} \mid y)$



# Converting a generative model into a classifier

But if we have a generative model, we can convert it into a discriminative model using Bayes rule:

classifier


$$p(y | \mathbf{x})$$

# Converting a generative model into a classifier

But if we have a generative model, we can convert it into a discriminative model using Bayes rule:

classifier

generative model

base rate of label

doesn't depend on  $y$

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

# Converting a generative model into a classifier

But if we have a generative model, we can convert it into a discriminative model using Bayes rule:

Pick the label that is most likely to produce the data:

$$y^* = \operatorname{argmax}_y p(\mathbf{x} \mid y)p(y)$$

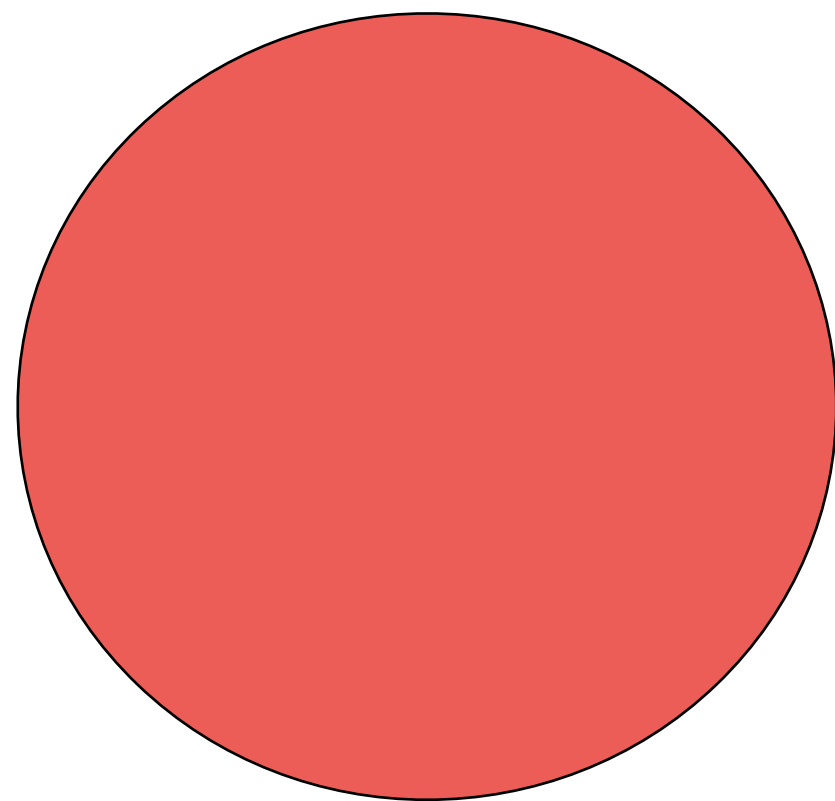
# Deep generative models

Source distribution

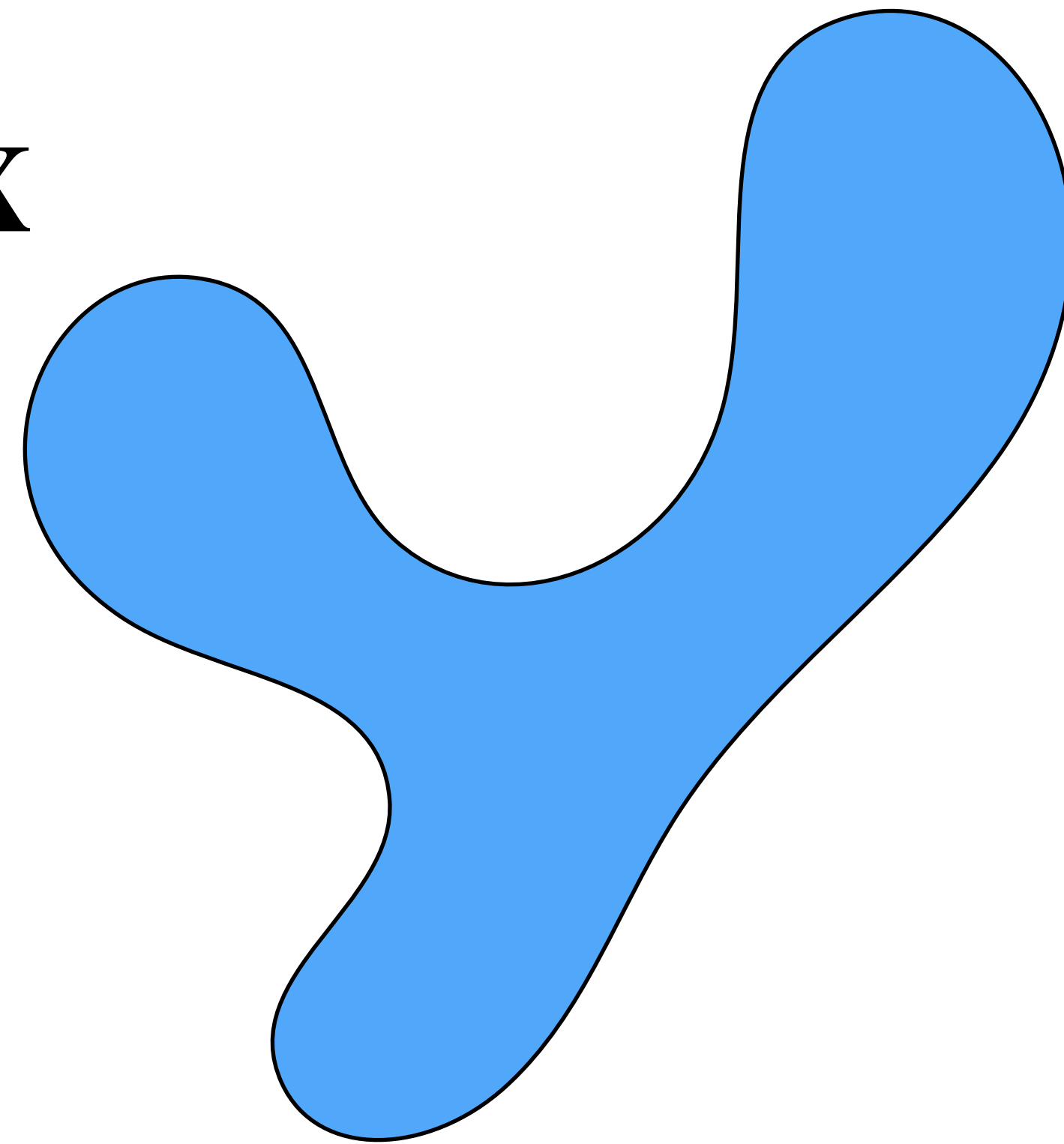
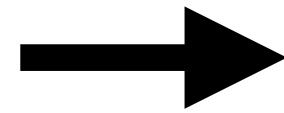
Target distribution

Network:

$$f_{\theta} : \mathbf{Z} \rightarrow \mathbf{X}$$



$p(z)$



$p(x)$



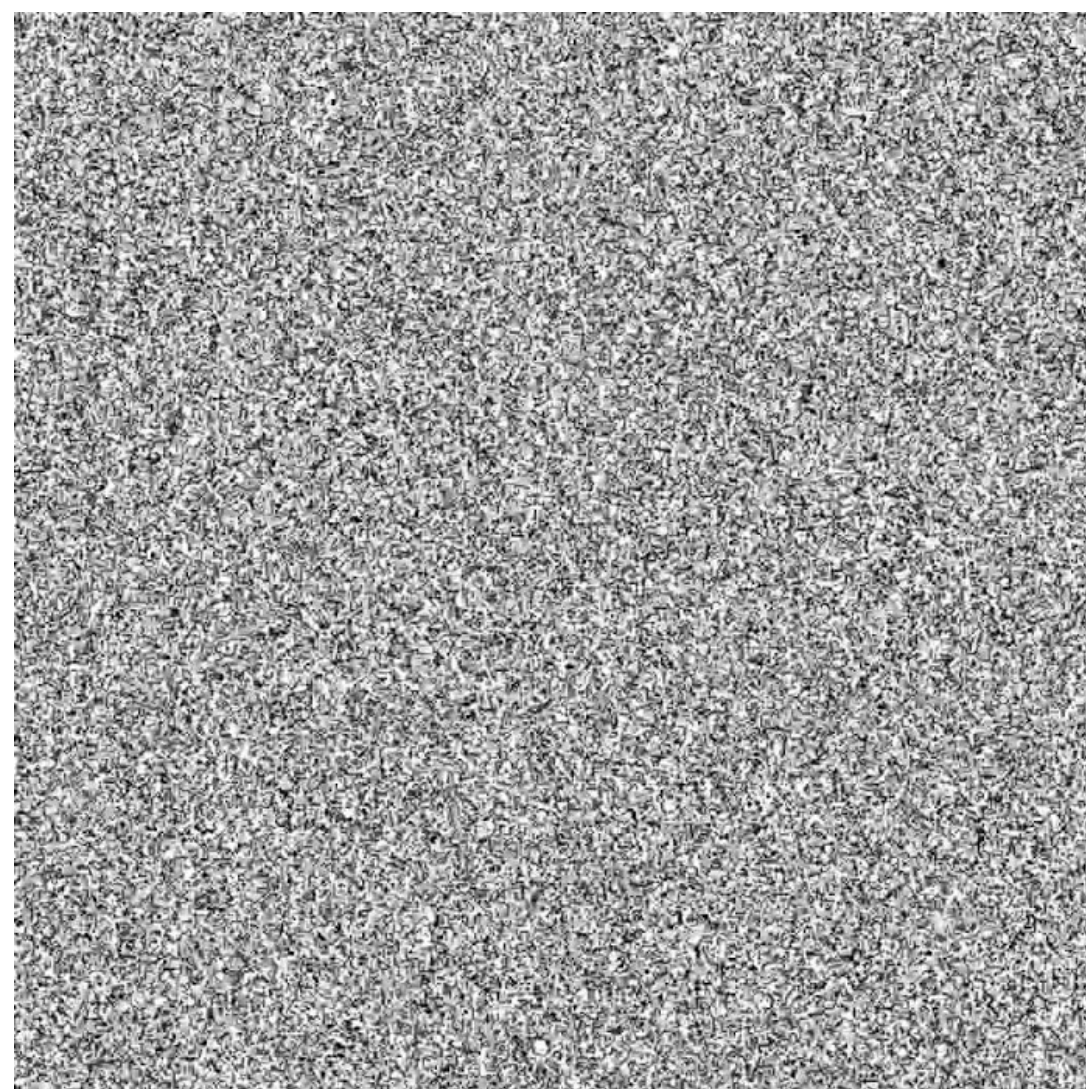
# Deep generative models

network parameters

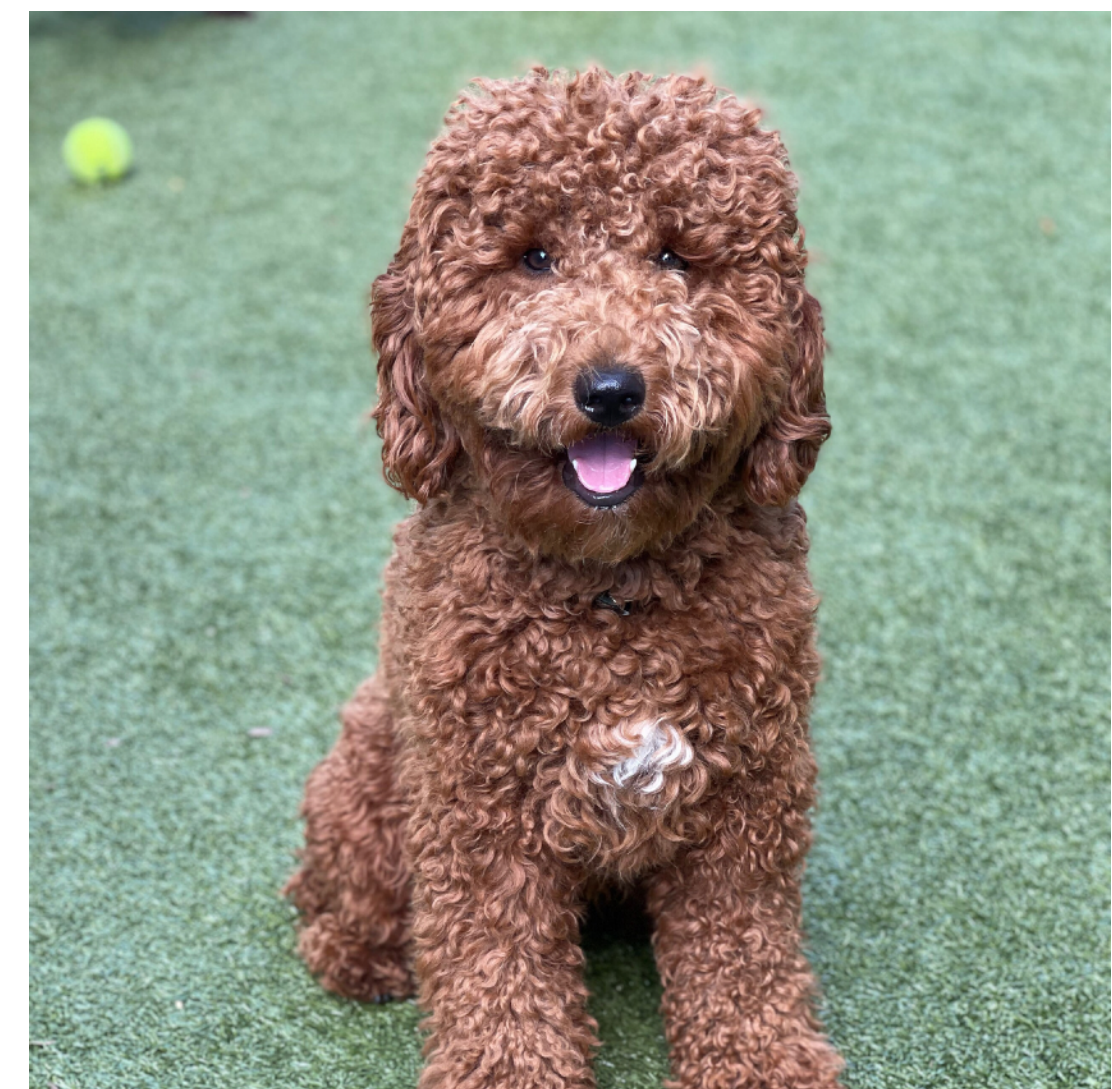
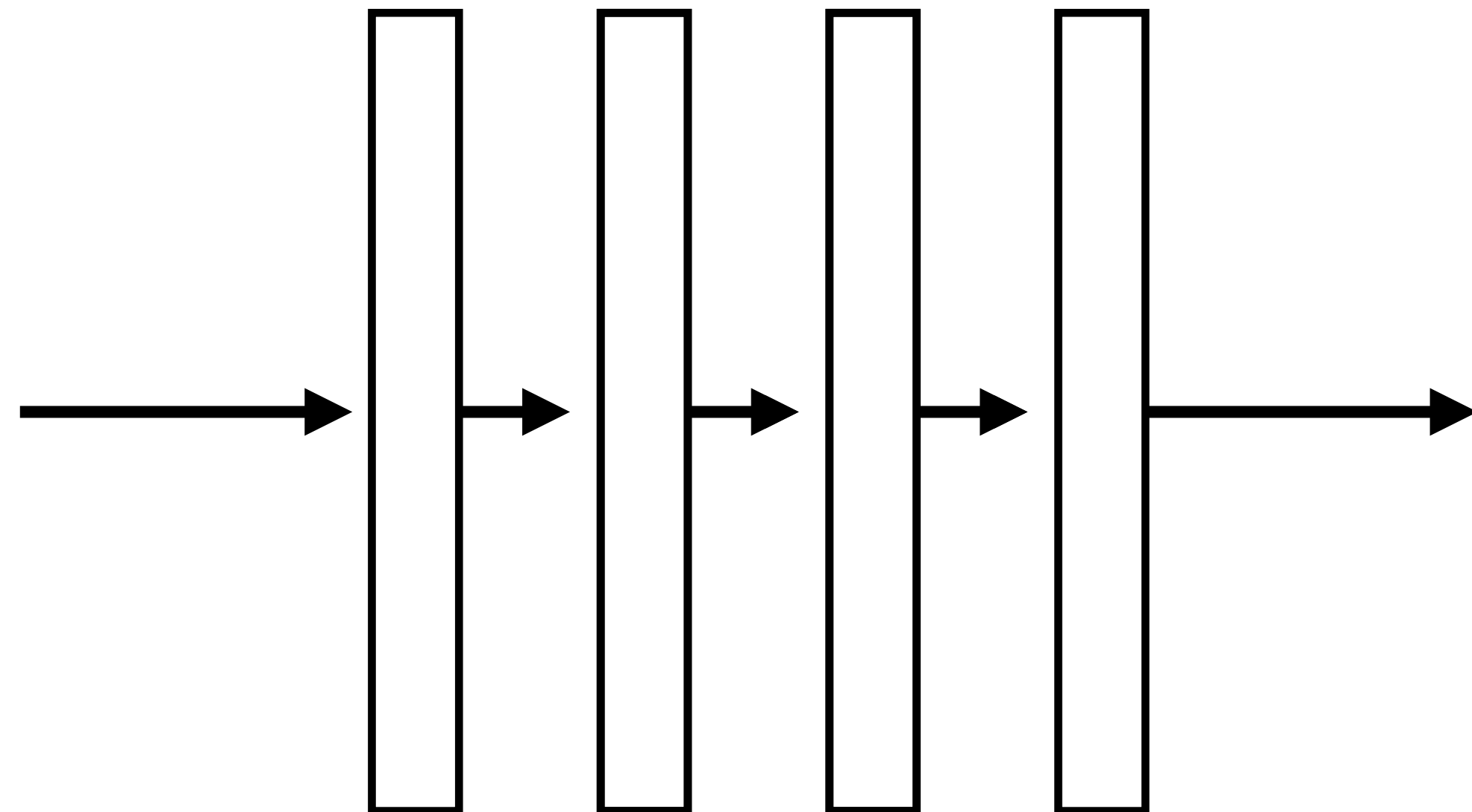
$$f_{\theta} : \mathbf{Z} \rightarrow \mathbf{X}$$

Learning the parameters:

$$\theta^* = \operatorname{argmin}_{\theta} L(\theta, \{\mathbf{X}_i\}_{i=1}^N)$$



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



$$\mathbf{X} = f_{\theta}(\mathbf{Z})$$

e.g., normalizing flow



What makes this challenging?



# State of the art generation 10+ years ago



Generated images from [Goodfellow et al.,  
"Generative Adversarial Nets", 2014]

Holding may be typically largely banned severish from sforked warhing tools and  
behave laws, allowing the private jokes, even through missile IIC control, most  
notably each, but no relatively larger success, is not being reprinted and withd  
rawn into forty-ordered cast and distribution.

Besides these markets (notably a son of humor).

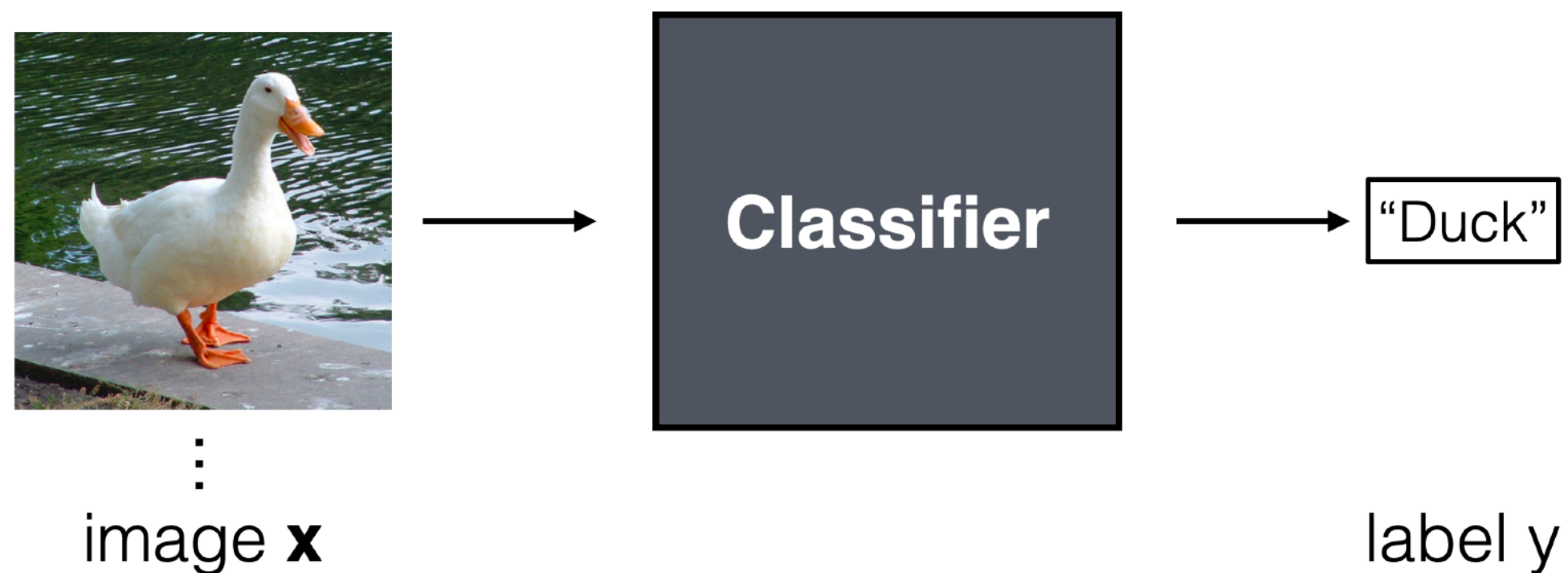
Sometimes more or only lowed "80" to force a suit for <http://news.bbc.co.uk/1/1/sid9kcid/web/9960219.html> '[[#10:82-14]]'.  
<blockquote>

Generated Wikipedia text from [Graves, "Generating  
Sequences With Recurrent Neural Networks", 2014]

# Many possible outputs

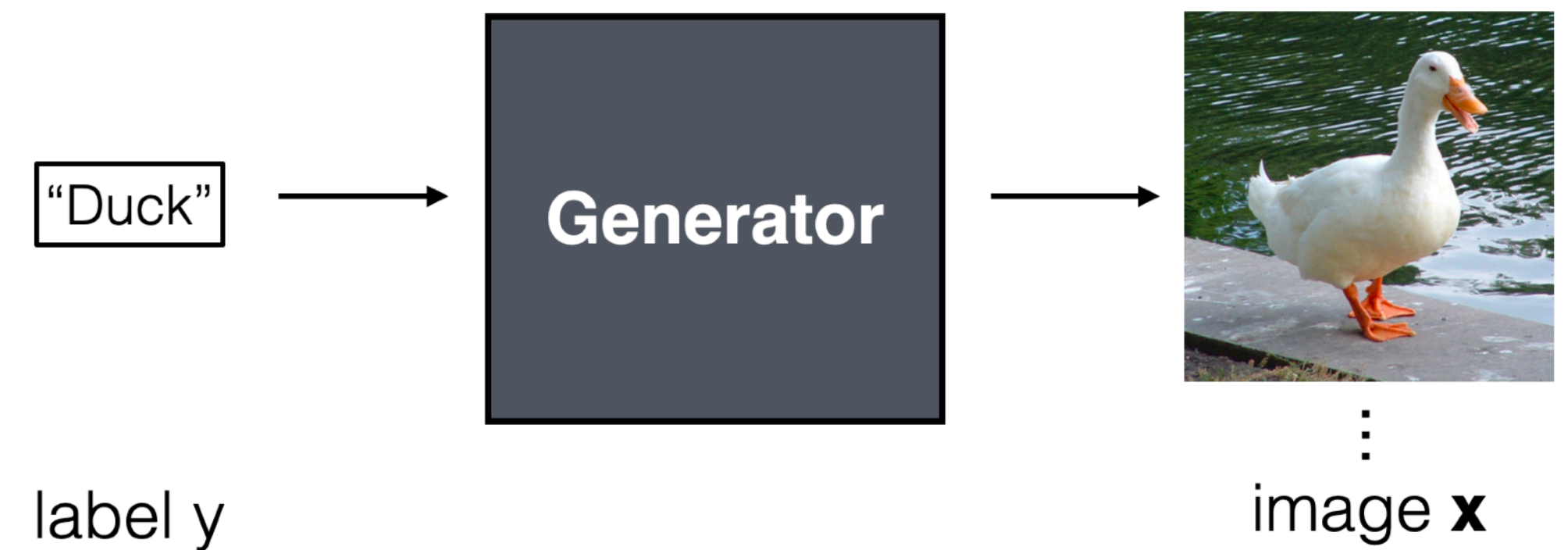
## Discriminative model

One true  $y$  value.



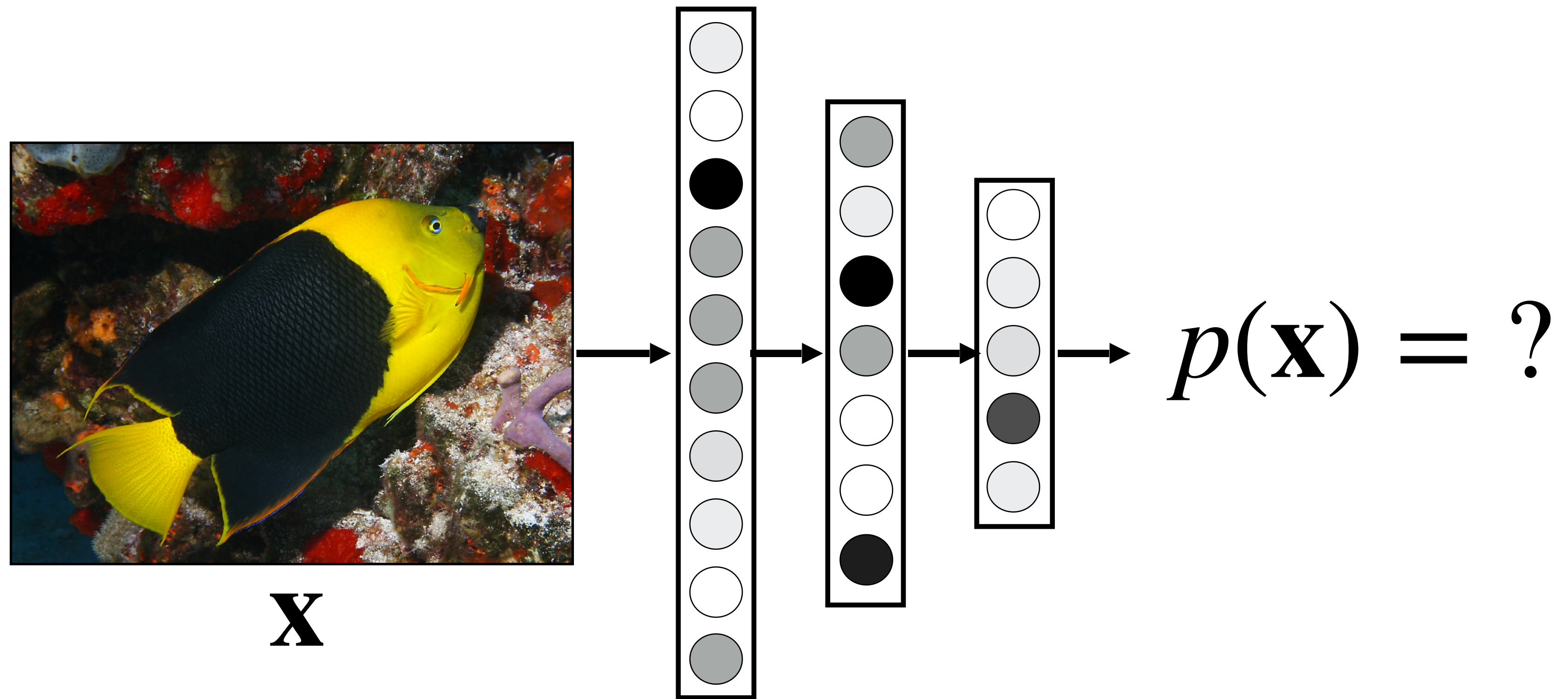
## Generative model

Many values of  $\mathbf{x}$ .

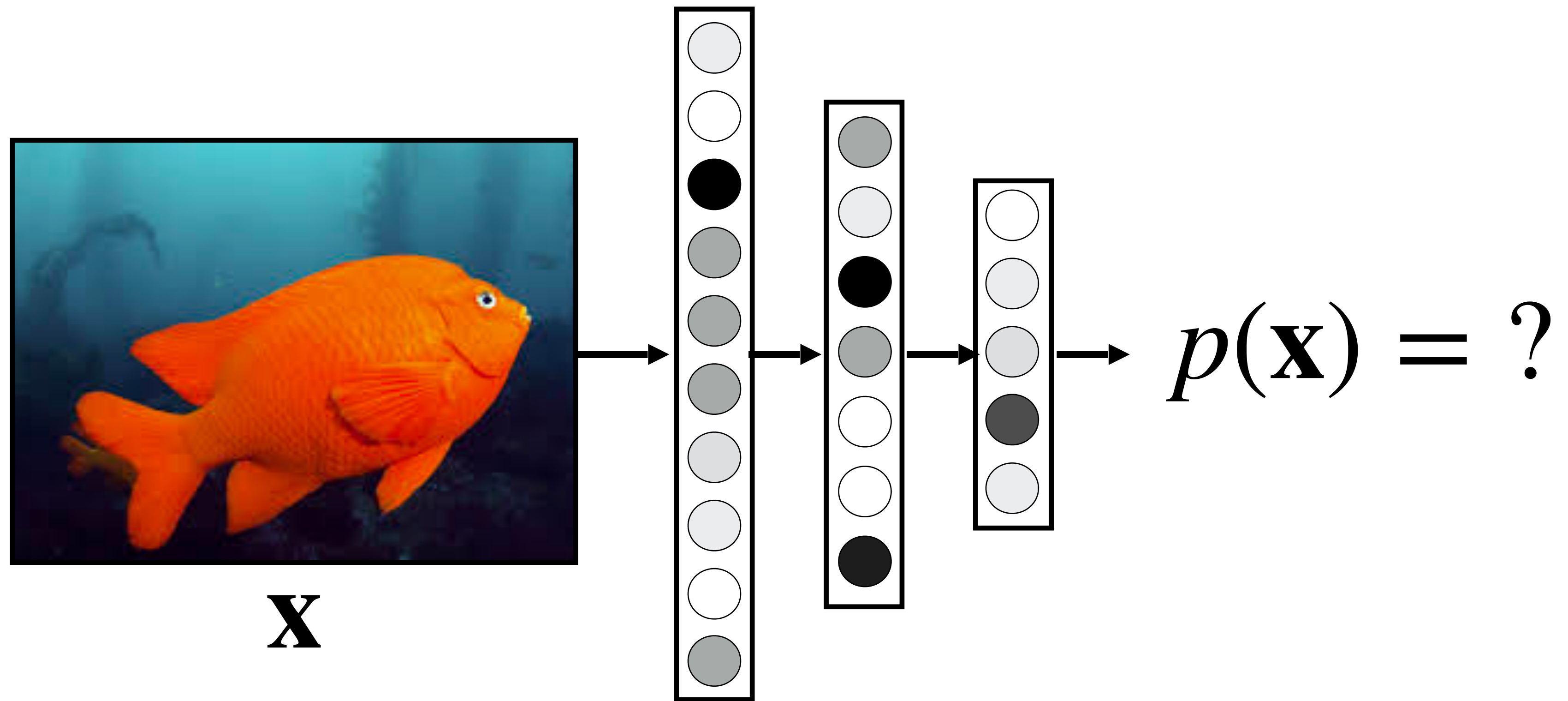




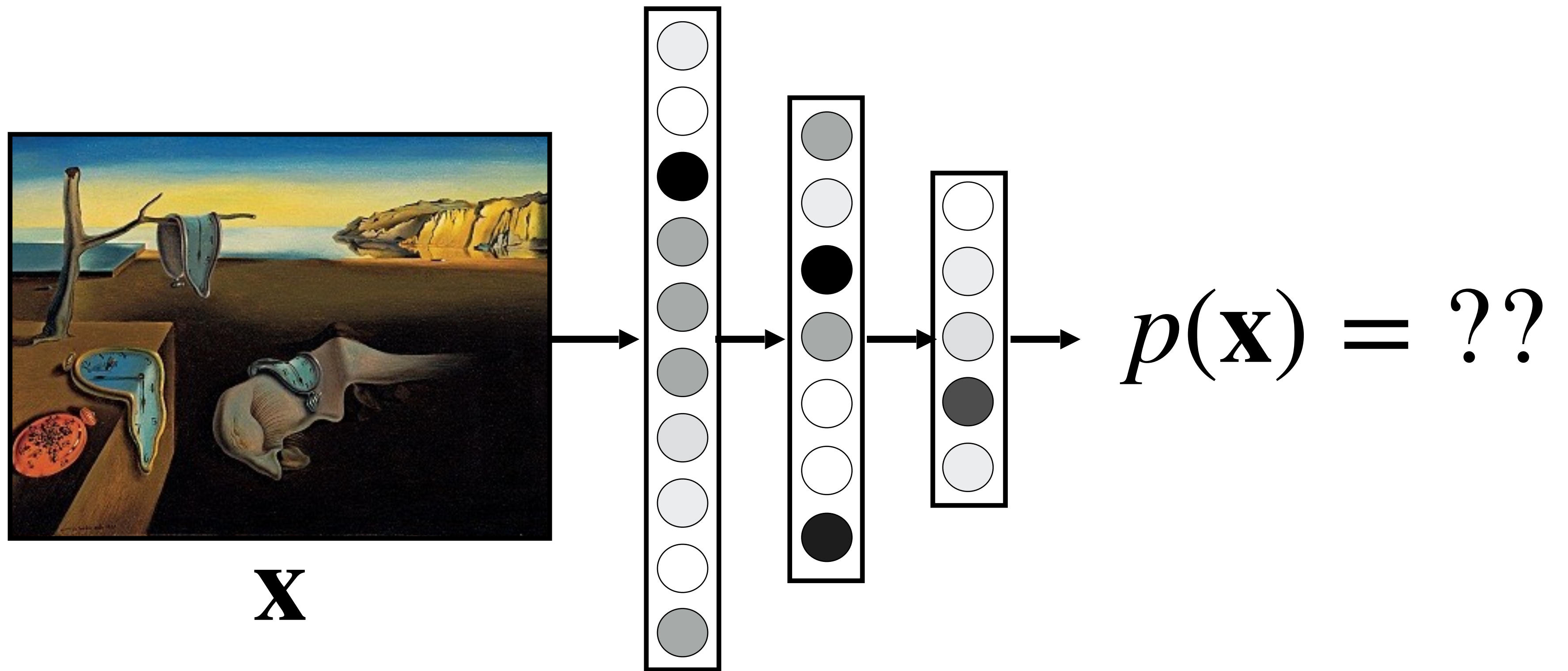
# The need to model everything



# The need to model everything

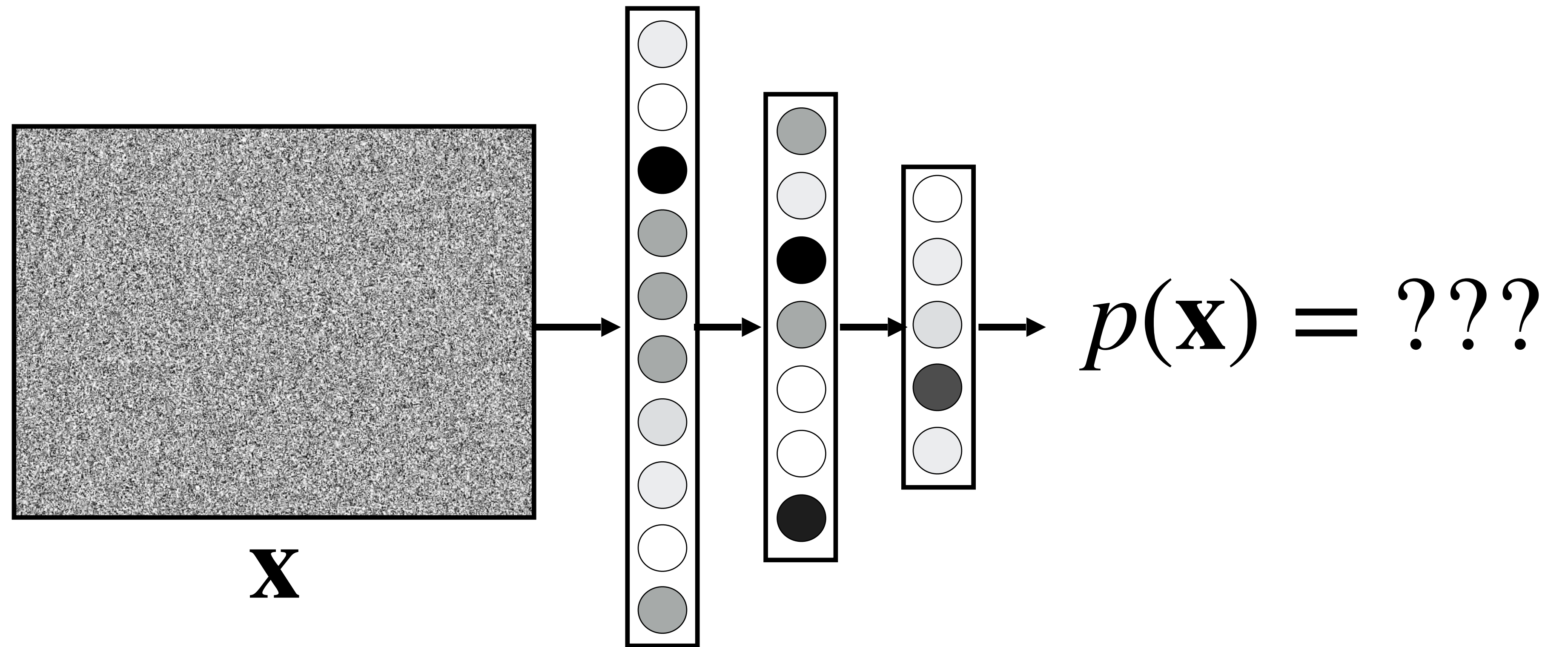


# The need to model everything






# The need to model everything

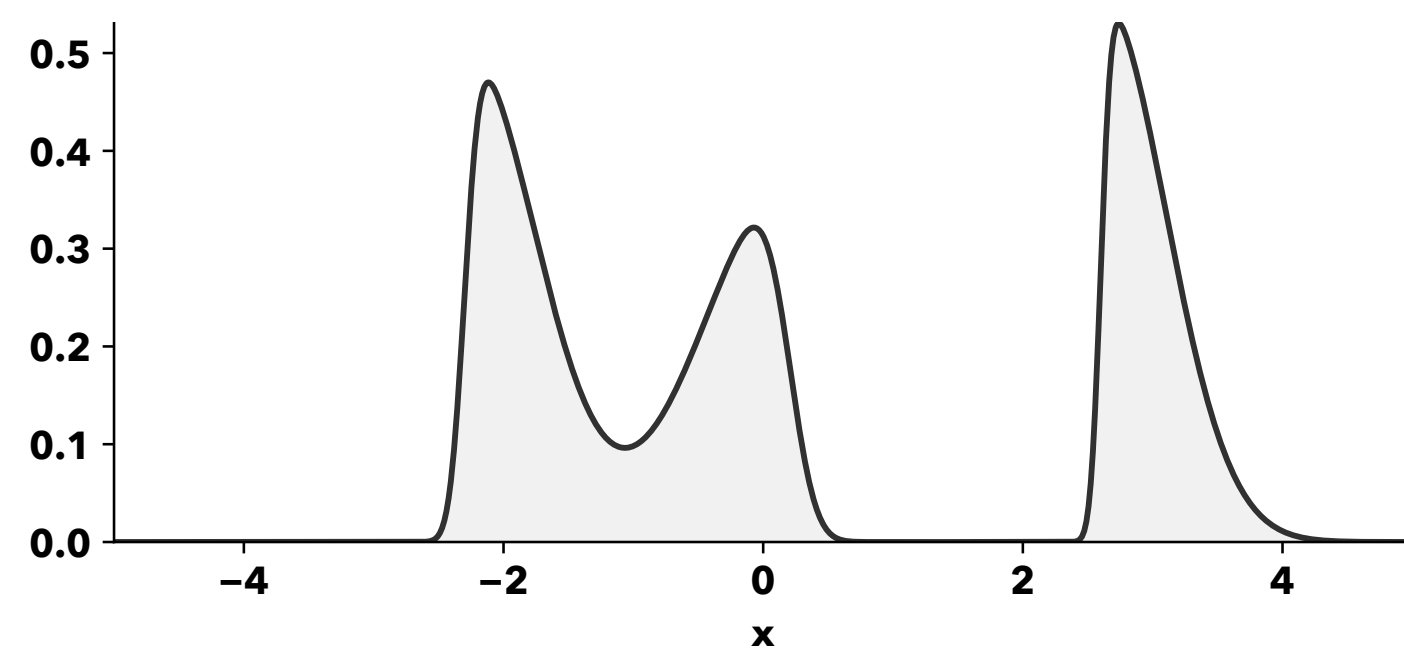


# The need to model everything

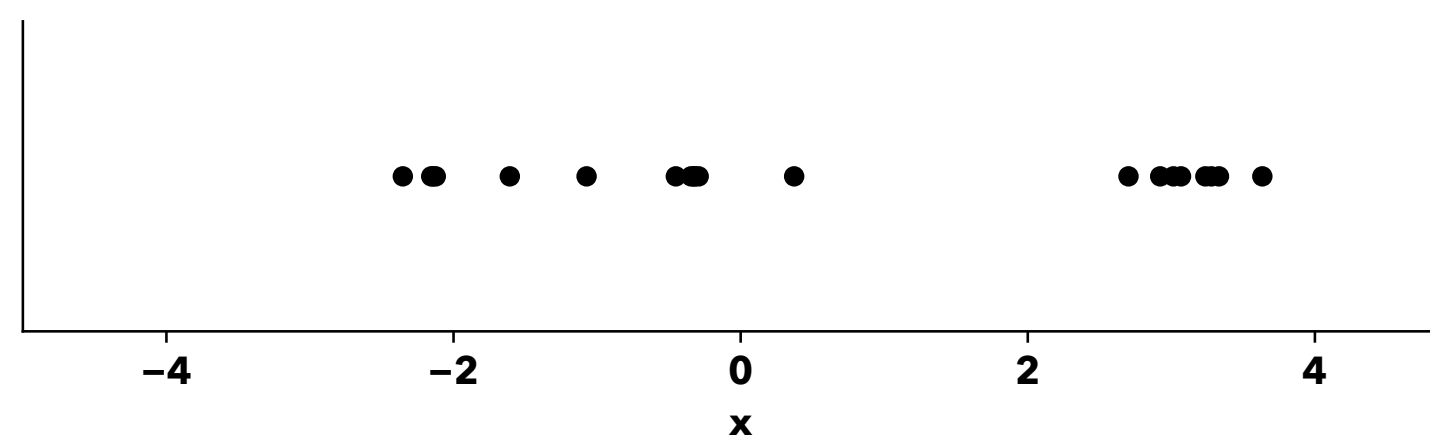
integration over high dimensional space


$$\int_{\mathbf{x}} p(\mathbf{x}) d\mathbf{x} = 1$$

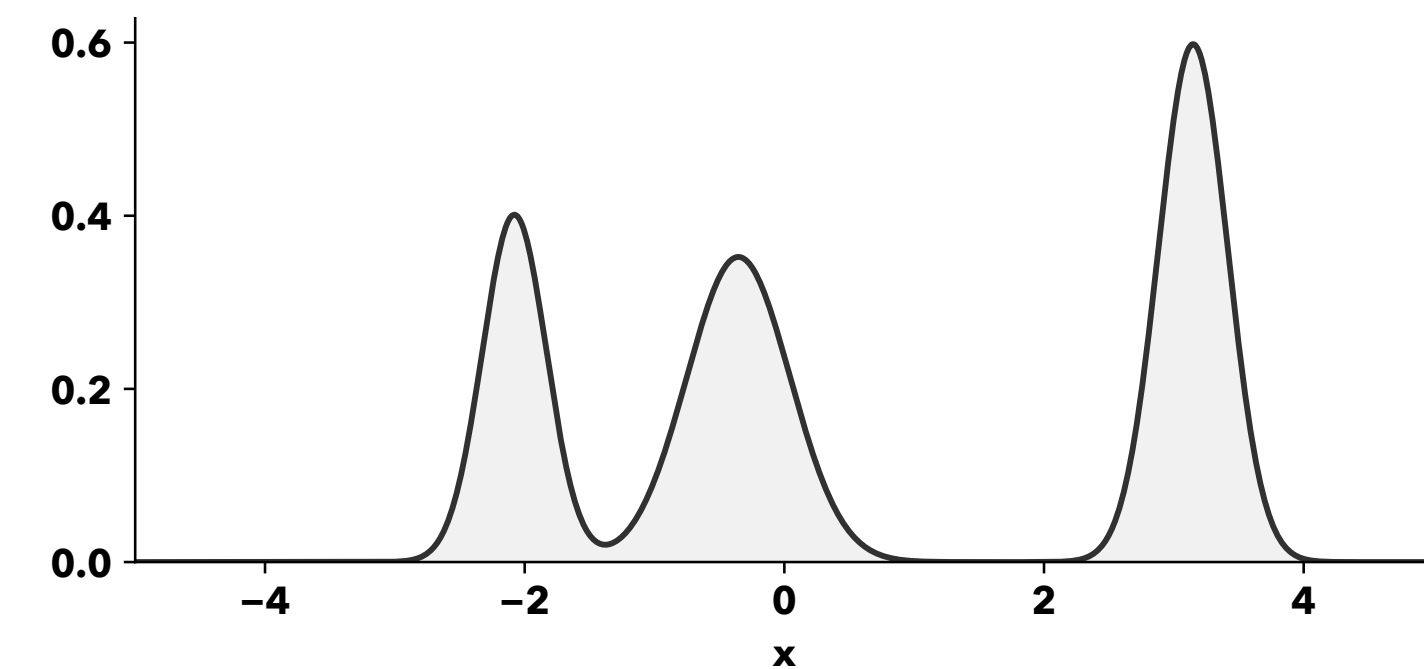
# Learning from samples



True distribution



Samples



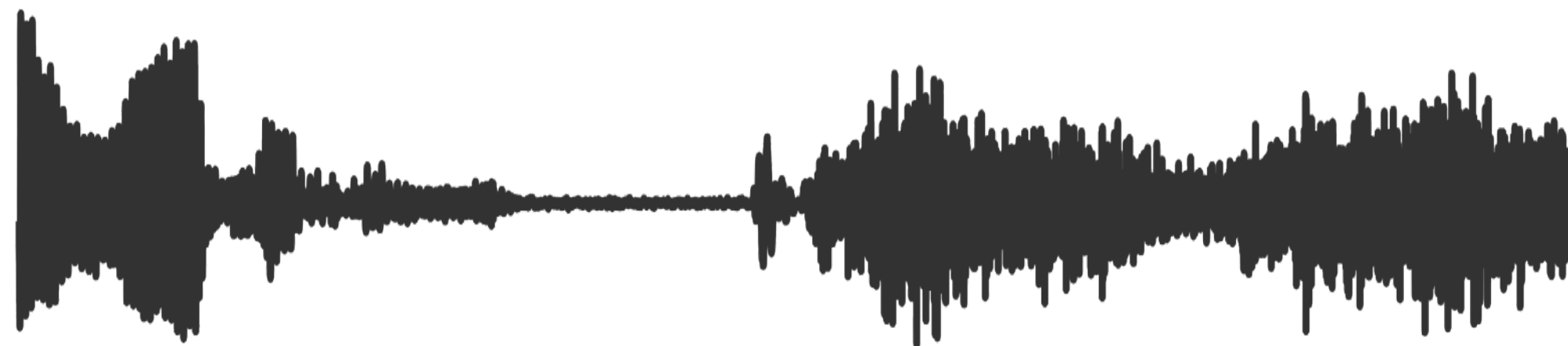
Fit distribution



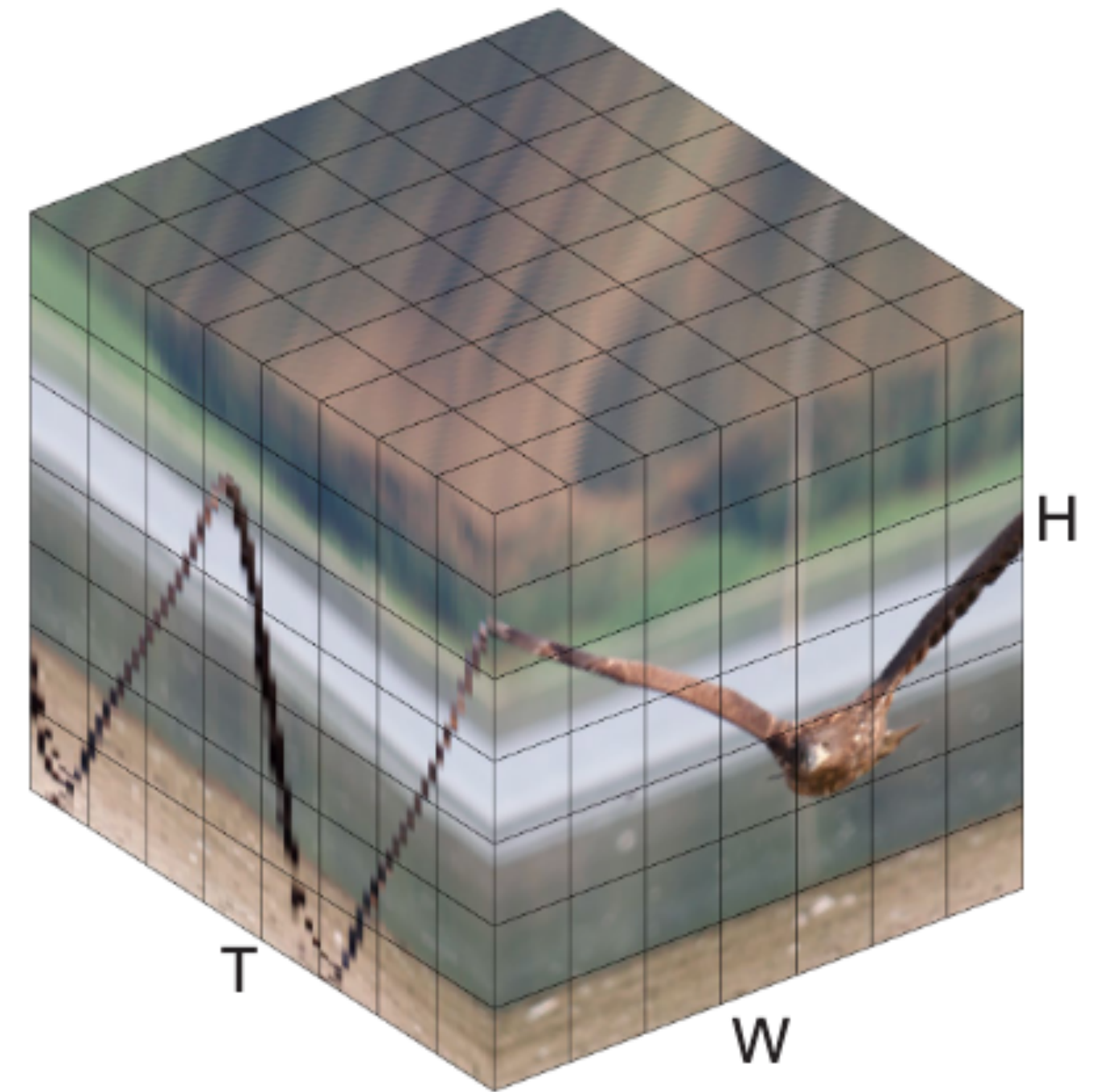
# Diversity of different inputs

“it was the best of times,  
it was the worst of times”

Discrete sequences



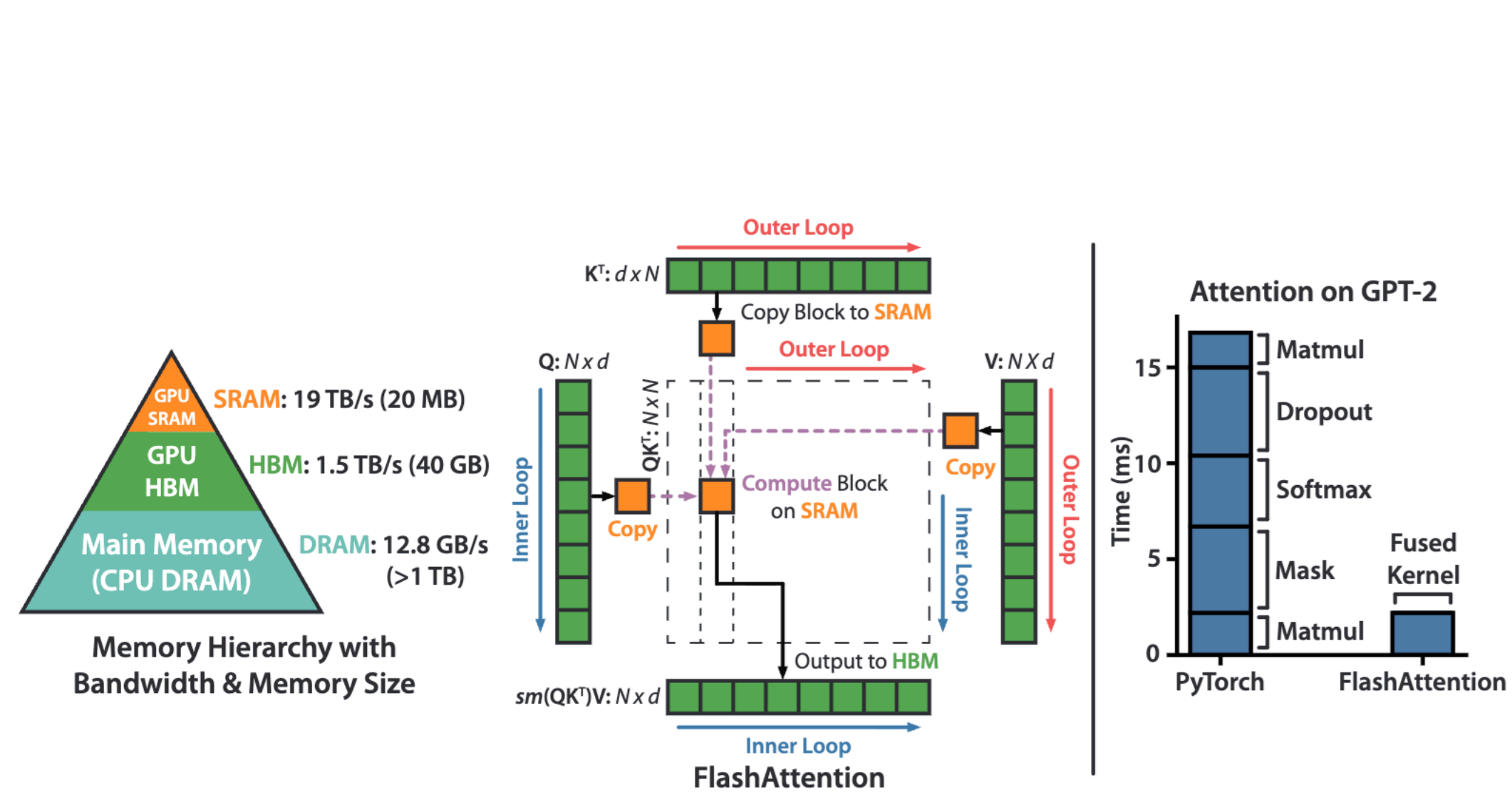
Continuous sequences



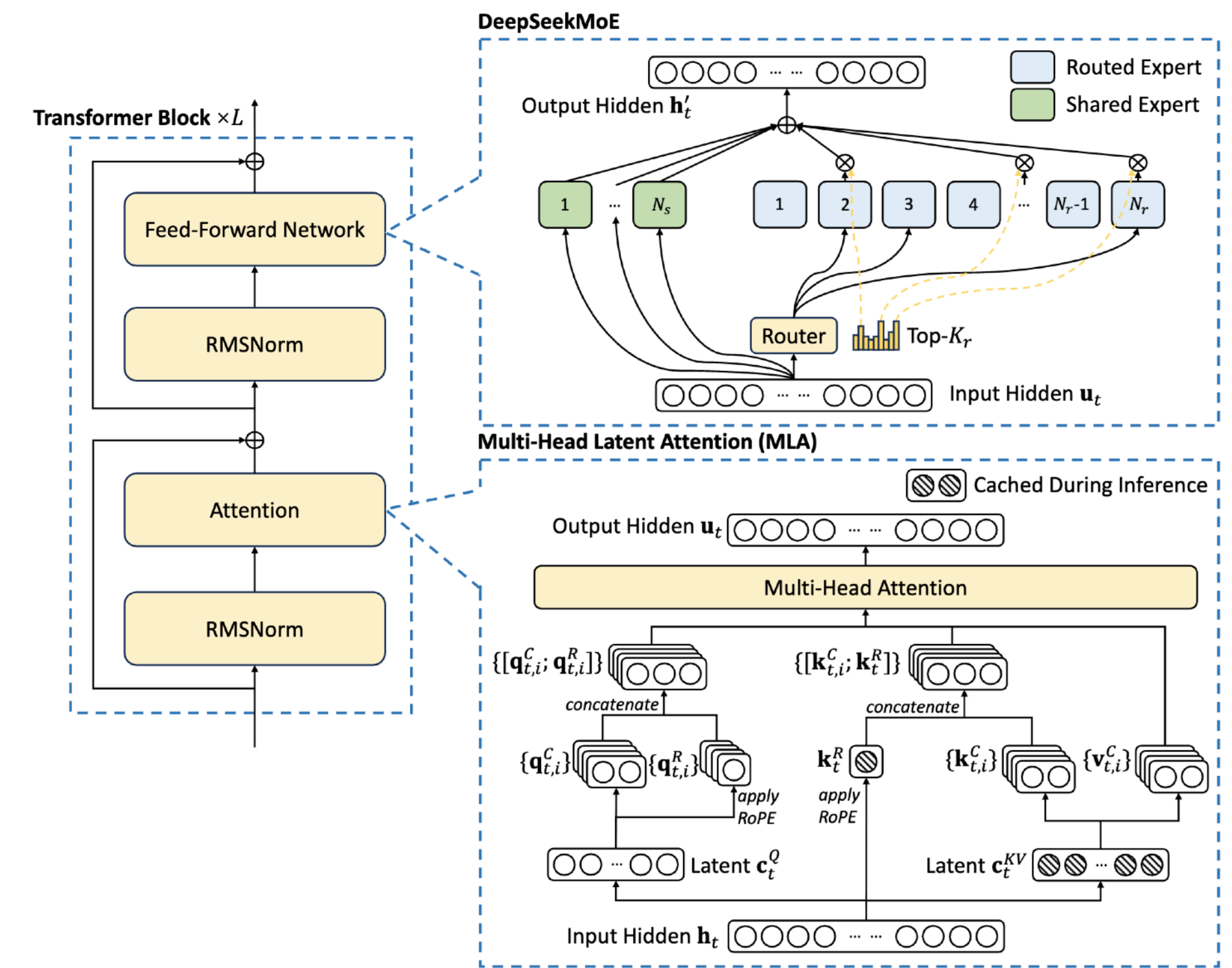
Continuous grids

# Computation

Need architectures and learning methods that are efficient and scalable.



Source: [Dao et al., "FlashAttention", 2022]



Source: [DeepSeek V3, 2024]

# And many other challenges

For example:

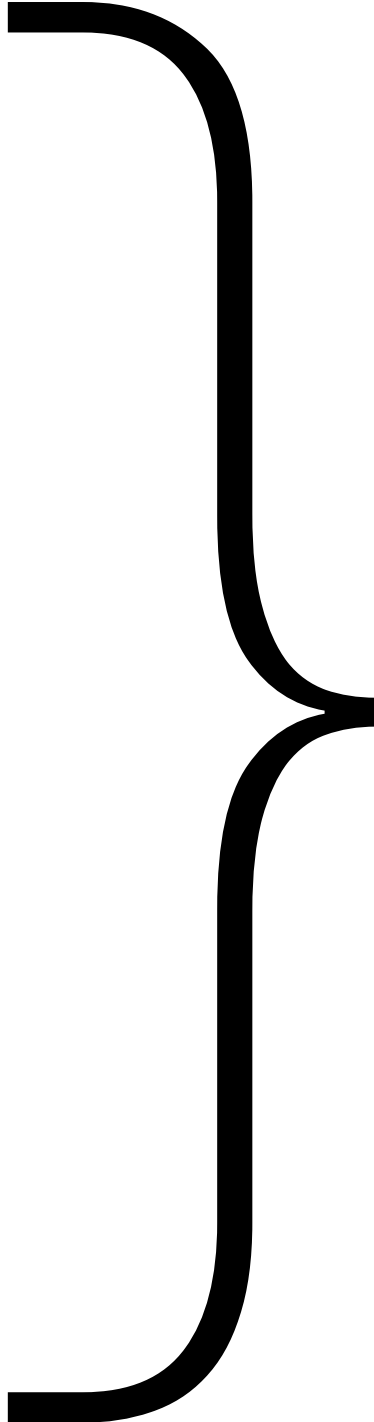
- Updating the model over time
- Adapting the model to other tasks
- Handling a computation budget
- Aligning model behavior with user preferences
- Dealing with very large inputs
- Learning from multimodal data
- Learning useful feature representations

# In this course, we'll cover:

- Some technical foundations of generative modeling
- Learning algorithms and architectures of deep generative models.
- Some applications of generative models

## What's ahead:

Lecture	Date	Topic
Lec. 1	Tue, Jan. 20	<b>Introduction</b> Course policies What is generative modeling?
Lec. 2	Thu, Jan. 22	<b>Maximum likelihood</b> Gaussian distribution Maximum likelihood
Lec. 3	Tue, Jan 27	<b>Gaussian mixture models</b> Stochastic gradient descent Gaussian mixture models Variational inference
Lec. 4	Thu, Jan 29	<b>Neural network review (recorded lecture)</b> Transformers Image translation networks Backpropagation
Lec. 5	Tue, Feb 3	<b>Variational autoencoders</b> Autoencoders Variational inference for VAEs Reparameterization trick VQ-VAEs



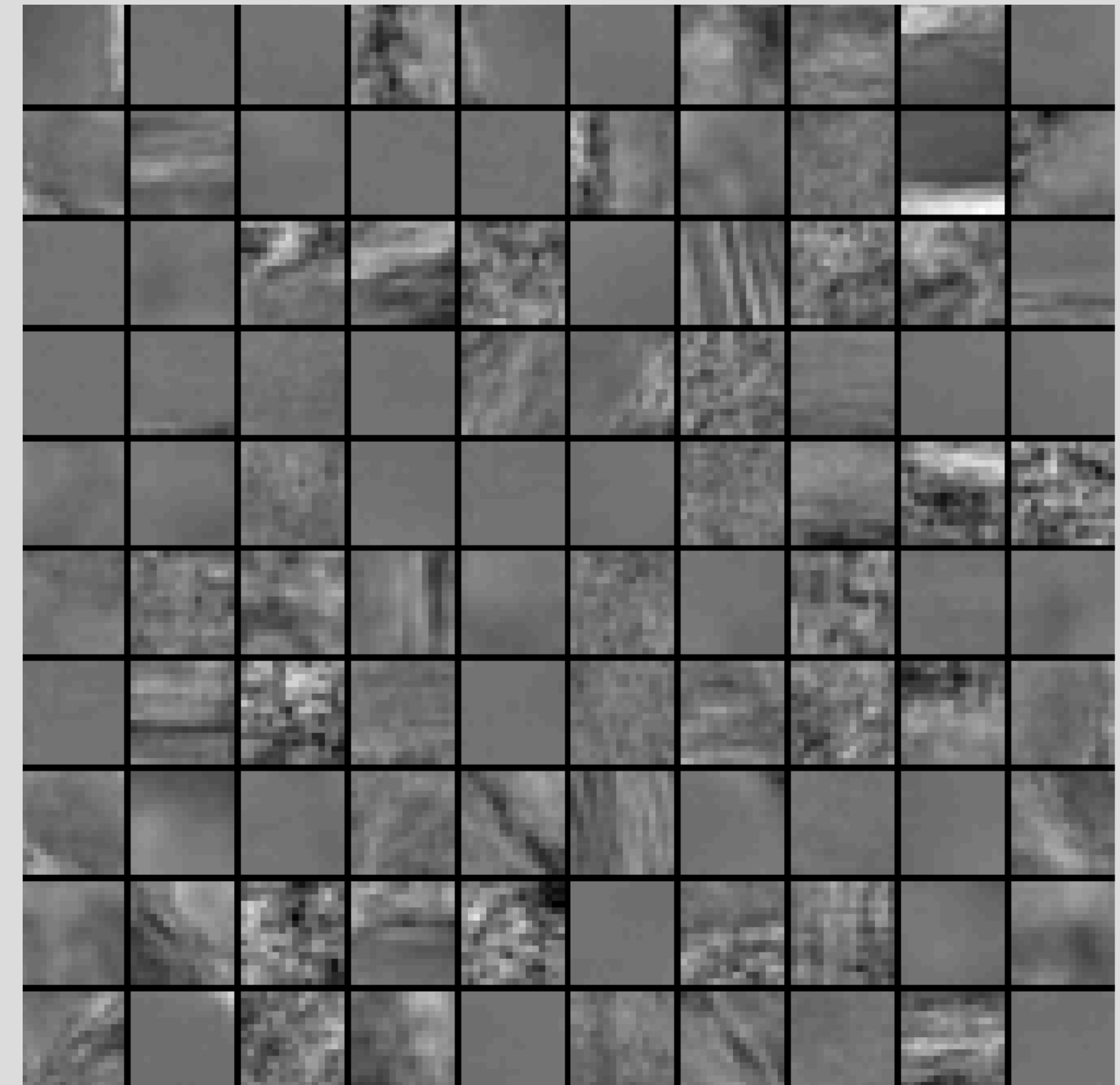
Generative models  
basics

Neural net review (recorded lecture)



Lecture	Date	Topic
Lec. 1	Tue, Jan. 20	<b>Introduction</b> Course policies What is generative modeling?
Lec. 2	Thu, Jan. 22	<b>Maximum likelihood</b> Gaussian distribution Maximum likelihood
Lec. 3	Tue, Jan 27	<b>Gaussian mixture models</b> Stochastic gradient descent Gaussian mixture models Variational inference
Lec. 4	Thu, Jan 29	<b>Neural network review (recorded lecture)</b> Transformers Image translation networks Backpropagation
Lec. 5	Tue, Feb 3	<b>Variational autoencoders</b> Autoencoders Variational inference for VAEs Reparameterization trick VQ-VAEs

## PS1: Simple probabilistic models

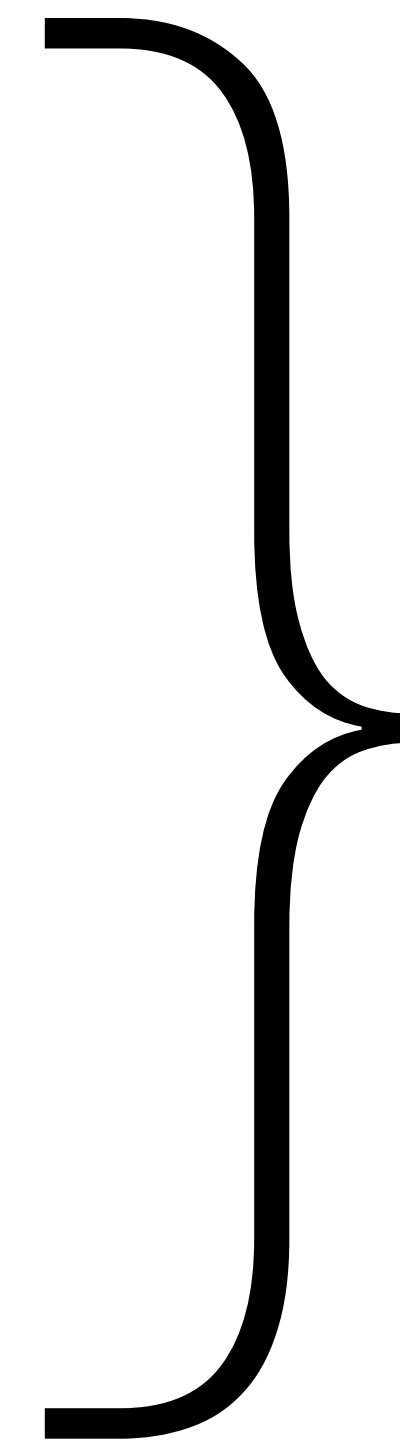


Gaussian mixture model for  
image patches

Source: [Zoran & Weiss, "Natural Images, Gaussian Mixtures and Dead Leaves", 2012]



Lec. 5	Tue, Feb 3	<b>Variational autoencoders</b> Autoencoders Variational inference for VAEs Reparameterization trick VQ-VAEs
Lec. 6	Thu, Feb 5	<b>Normalizing flows</b> Change of variables formula Coupling layers Inverse autoregressive flows
Lec. 7	Tue, Feb 10	<b>Generative adversarial networks</b> Minimax games Mode collapse Optimization
Lec. 8	Thu, Feb 12	<b>GANs for image synthesis</b> Conditional GANs Cycle consistency VQ-GANs
	Tue, Feb 17	<b>No class</b>
Lec. 9	Thu, Feb 19	<b>Energy-based models</b> Langevin dynamics The partition function Score matching perspective



Latent variable  
models

Lec. 5	Tue, Feb 3	<b>Variational autoencoders</b> Autoencoders Variational inference for VAEs Reparameterization trick VQ-VAEs
Lec. 6	Thu, Feb 5	<b>Normalizing flows</b> Change of variables formula Coupling layers Inverse autoregressive flows
Lec. 7	Tue, Feb 10	<b>Generative adversarial networks</b> Minimax games Mode collapse Optimization
Lec. 8	Thu, Feb 12	<b>GANs for image synthesis</b> Conditional GANs Cycle consistency VQ-GANs
	Tue, Feb 17	<b>No class</b>
Lec. 9	Thu, Feb 19	<b>Energy-based models</b> Langevin dynamics The partition function Score matching perspective

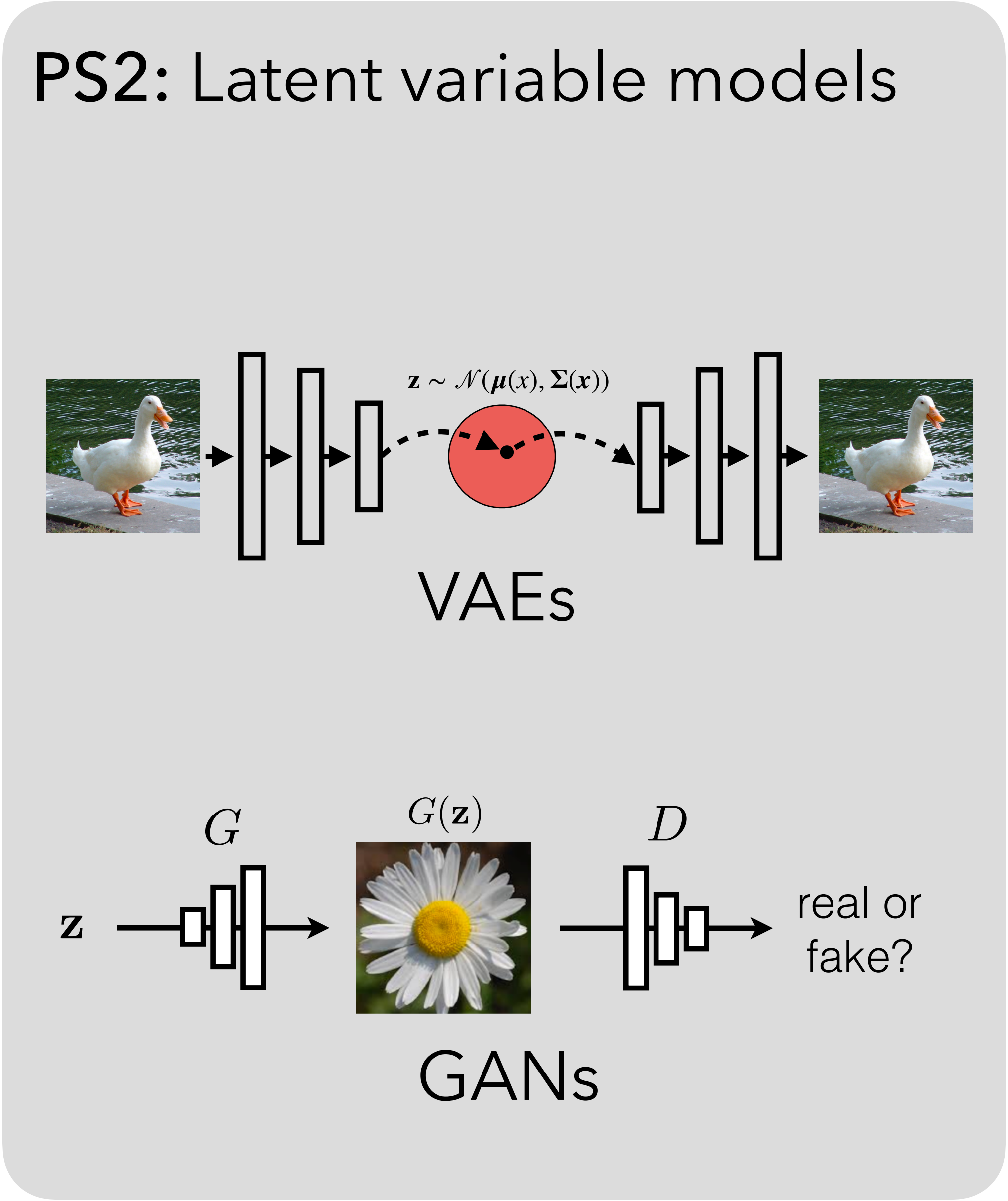
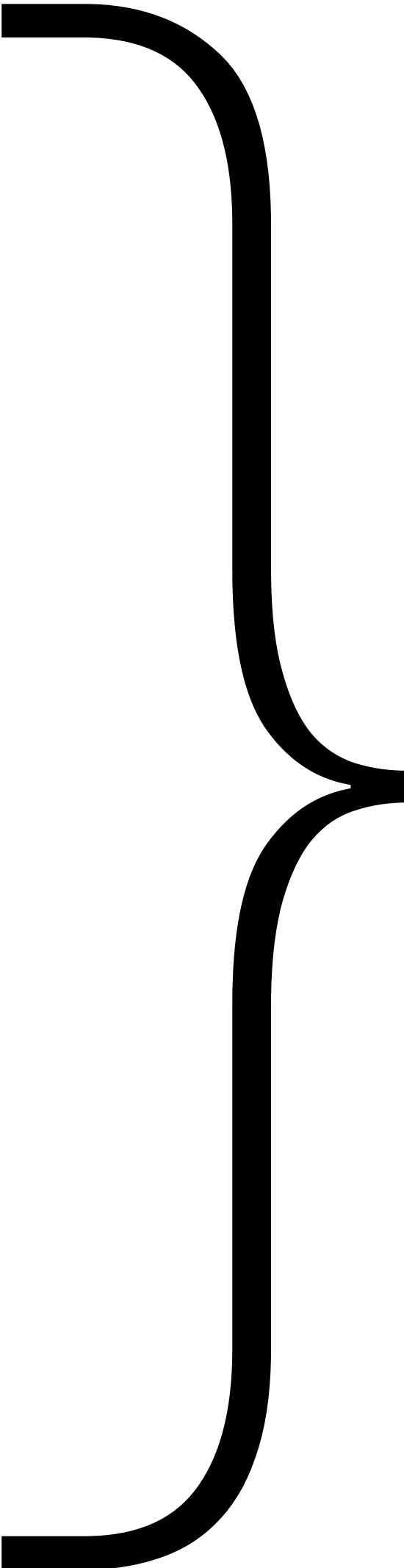


Figure source: Isola, Torralba, Freeman

Lec. 9	Thu, Feb 19	<b>Energy-based models</b> Langevin dynamics The partition function Score matching perspective
Lec. 10	Tue, Feb 24	<b>Diffusion models 1</b> Diffusion models for image synthesis DDPM Connection to VAEs
Lec. 11	Thu, Feb. 26	<b>Diffusion models 2</b>
Lec. 12	Tue, Mar. 3	<b>Image manipulation with diffusion models</b> Conditional diffusion models Classifier-free guidance Inpainting and SDEdit
Lec. 13	Thu, Mar. 5	<b>Flow matching</b> Flow formulation Rectified flow Continuous normalizing flows
Lec. 14	Tue, Mar 10	<b>Diffusion architectures</b> Transformer-based diffusion models Latent diffusion models Few-step generation

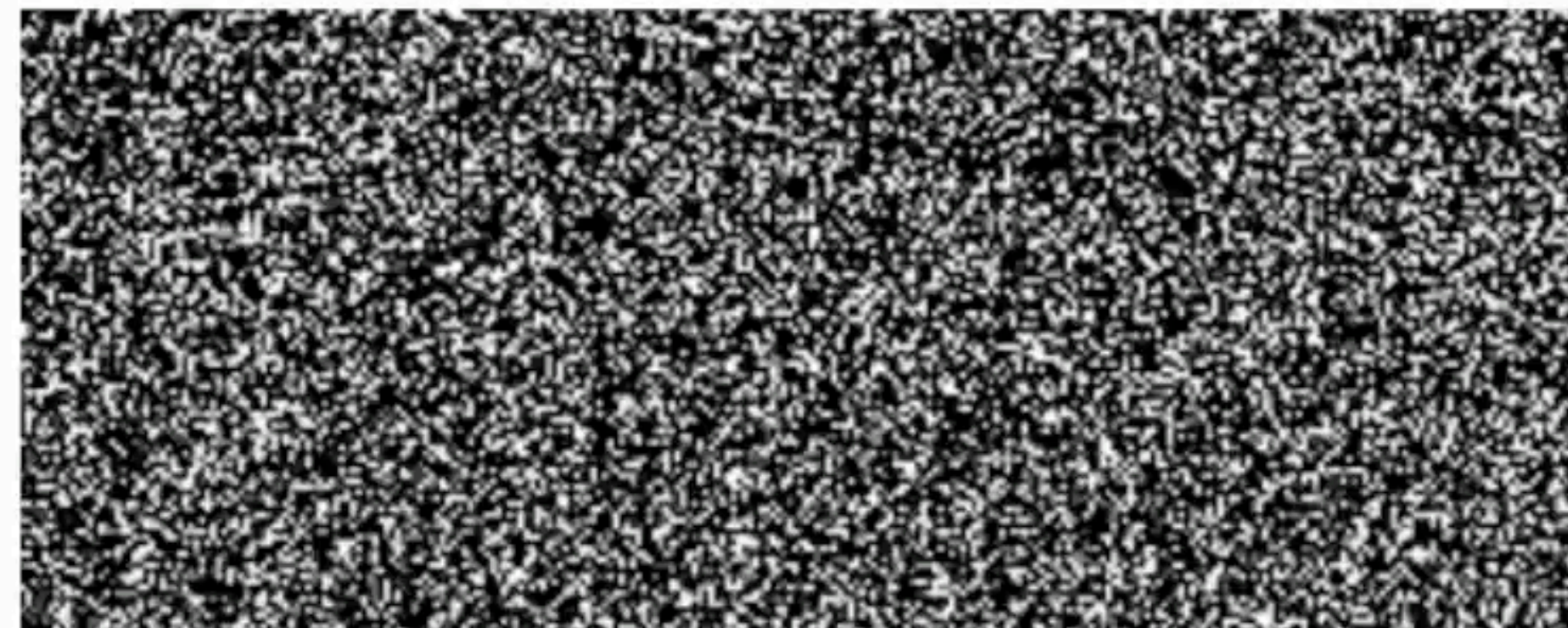


Diffusion models

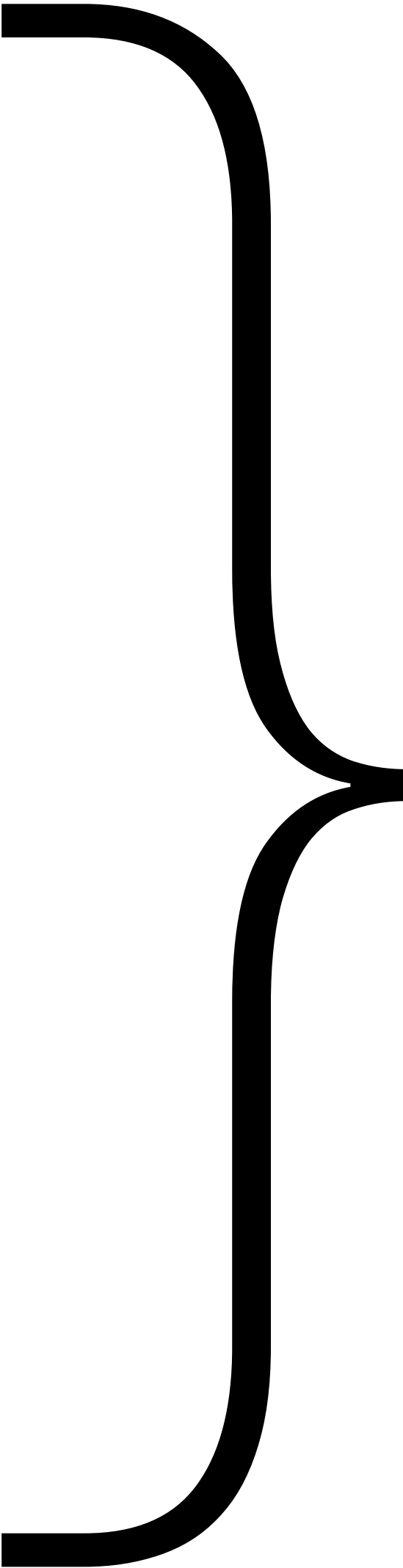


Lec. 9	Thu, Feb 19	<b>Energy-based models</b> Langevin dynamics The partition function Score matching perspective
Lec. 10	Tue, Feb 24	<b>Diffusion models 1</b> Diffusion models for image synthesis DDPM Connection to VAEs
Lec. 11	Thu, Feb. 26	<b>Diffusion models 2</b>
Lec. 12	Tue, Mar. 3	<b>Image manipulation with diffusion models</b> Conditional diffusion models Classifier-free guidance Inpainting and SDEdit
Lec. 13	Thu, Mar. 5	<b>Flow matching</b> Flow formulation Rectified flow Continuous normalizing flows
Lec. 14	Tue, Mar 10	<b>Diffusion architectures</b> Transformer-based diffusion models Latent diffusion models Few-step generation

## PS3: Diffusion and flow matching



		Transformer-based diffusion models Latent diffusion models Few-step generation
	Thu, Mar. 12	<b>Autoregressive models</b>
Lec. 15	Tue, Mar. 17	<b>Language models</b> GPT Tokenization Parallel decoding
Lec. 16	Thu, Mar. 19	<b>Discrete diffusion models</b> Masked language modeling Diffusion in latent spaces
Lec. 17	Tue, Mar. 24	<b>Applying generative models to downstream tasks</b> Representation learning Zero-shot learning
Lec. 18	Thu, Mar. 26	<b>Scaling</b> Scaling laws Systems issues involved
	Mon, Mar. 31	<b>No class</b>
	Wed, Apr. 2	<b>No class</b>
Lec. 19	Tue, Apr. 7	<b>Midterm review</b>

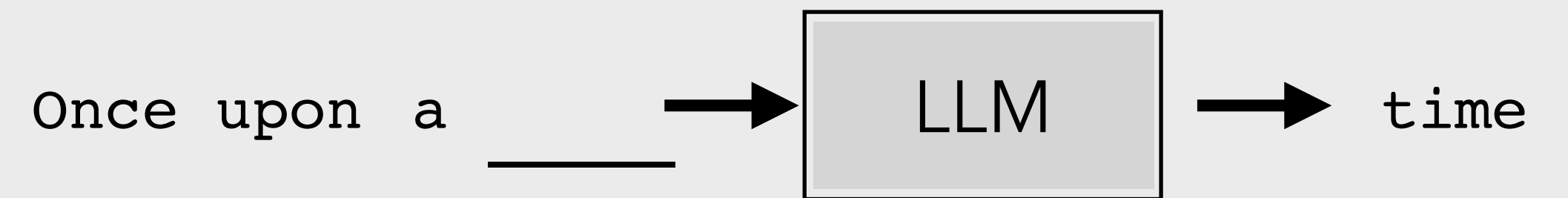


Autoregressive  
models



		Transformer-based diffusion models Latent diffusion models Few-step generation
	Thu, Mar. 12	<b>Autoregressive models</b>
Lec. 15	Tue, Mar. 17	<b>Language models</b> GPT Tokenization Parallel decoding
Lec. 16	Thu, Mar. 19	<b>Discrete diffusion models</b> Masked language modeling Diffusion in latent spaces
Lec. 17	Tue, Mar. 24	<b>Applying generative models to downstream tasks</b> Representation learning Zero-shot learning
Lec. 18	Thu, Mar. 26	<b>Scaling</b> Scaling laws Systems issues involved
	Mon, Mar. 31	<b>No class</b>
	Wed, Apr. 2	<b>No class</b>
Lec. 19	Tue, Apr. 7	<b>Midterm review</b>

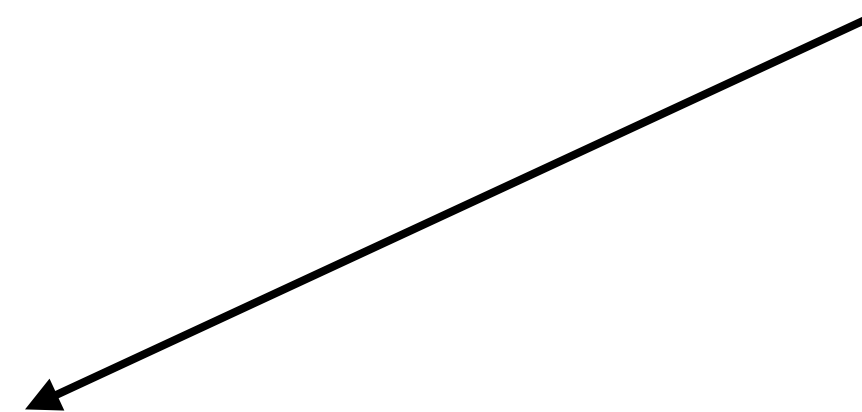
## PS4: Language models



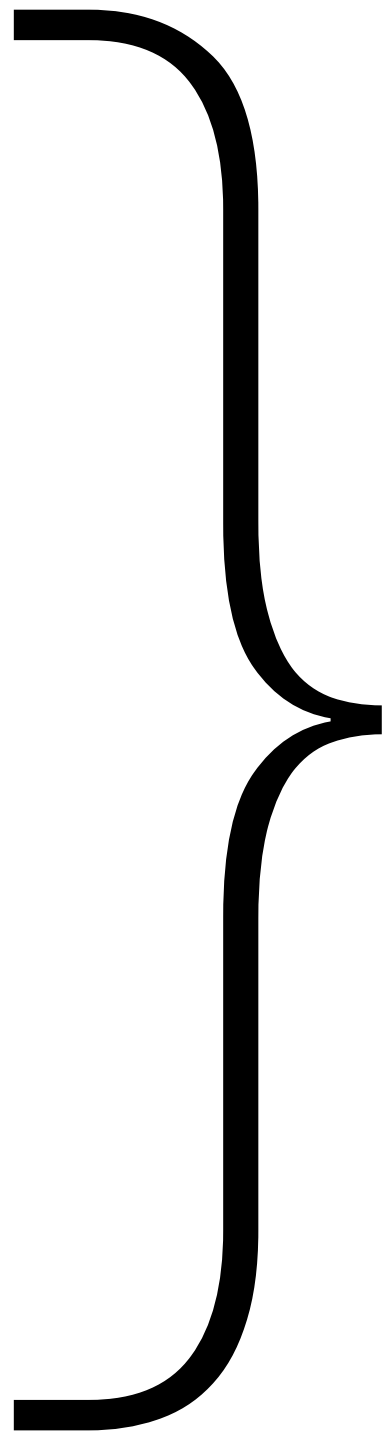


Lec. 17	Tue, Mar. 24	<b>Applying generative models to downstream tasks</b> Representation learning Zero-shot learning
Lec. 18	Thu, Mar. 26	<b>Scaling</b> Scaling laws Systems issues involved
	Mon, Mar. 31	<b>No class</b>
	Wed, Apr. 2	<b>No class</b>
Lec. 19	Tue, Apr. 7	<b>Midterm review</b>
Lec. 20	Thu, Apr. 9	<b>Midterm exam</b>
	Fri, Apr. 10	<b>Tentative midterm time</b>
Lec. 21	Tue, Apr. 14	<b>Post-training</b> Instruction tuning (for both images and language) RLHF RL-based reasoning models
Lec. 22	Thu, Apr 16	<b>Evaluating generative models</b>
Lec. 23	Tue, Apr 21	<b>Generated media provenance</b> Detecting generated images/text

# Midterm after spring break



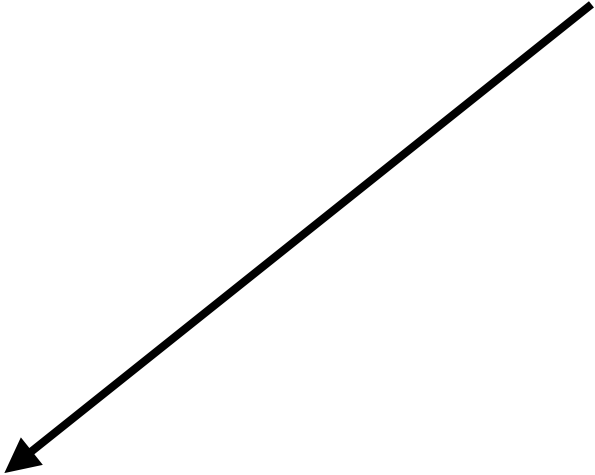
	Fri, Apr. 10	<b>Tentative midterm time</b>
Lec. 21	Tue, Apr. 14	<b>Post-training</b> Instruction tuning (for both images and language) RLHF RL-based reasoning models
Lec. 22	Thu, Apr 16	<b>Evaluating generative models</b>
Lec. 23	Tue, Apr 21	<b>Generated media provenance</b> Detecting generated images/text Artist attribution Watermarking
Lec. 24	Thu, Apr 23	<b>Model interpretability</b> Feature visualization Influence functions
Lec. 25	Tue, Apr. 28	
Lec. 26	Thu, Apr. 30	<b>Final project presentations</b>
Lec. 27	Tue, May 5	<b>Final project presentations</b>



# Advanced topics

	Fri, Apr. 10	<b>Tentative midterm time</b>
Lec. 21	Tue, Apr. 14	<b>Post-training</b> Instruction tuning (for both images and language) RLHF RL-based reasoning models
Lec. 22	Thu, Apr 16	<b>Evaluating generative models</b>
Lec. 23	Tue, Apr 21	<b>Generated media provenance</b> Detecting generated images/text Artist attribution Watermarking
Lec. 24	Thu, Apr 23	<b>Model interpretability</b> Feature visualization Influence functions
Lec. 25	Tue, Apr. 28	
Lec. 26	Thu, Apr. 30	<b>Final project presentations</b>
Lec. 27	Tue, May 5	<b>Final project presentations</b>

Final presentations





Questions?

**Next class: maximum likelihood estimation**