# SC395:

# Image Generative Models in Computer Vision

Viraj Shah

Lecture 1

Jan 5th, 2026

sc395.virajshah.com

Viraj Shah

Research Scientist, Google

Mountain View, CA

Ahmedabad, India

UIUC

Iowa State University

IIT Roorkee

Computer Vision

Image Generative Models

Computational Photography

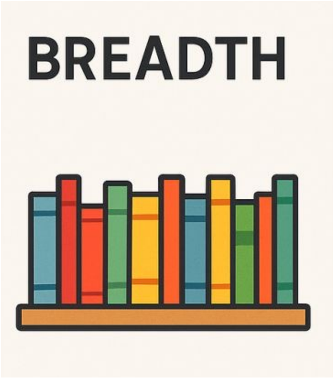
Siemens

Amazon

Adobe

Google

# Introduction




What GenAI hype is all about?

Is GenAI relevant to your area of study?

Will this course be useful to you?

……

**No Language restrictions!**

# What is your AI / Computer Vision WOW moment?

- Image Classification / ImageNet, 2014
- .....

- .....
- .....

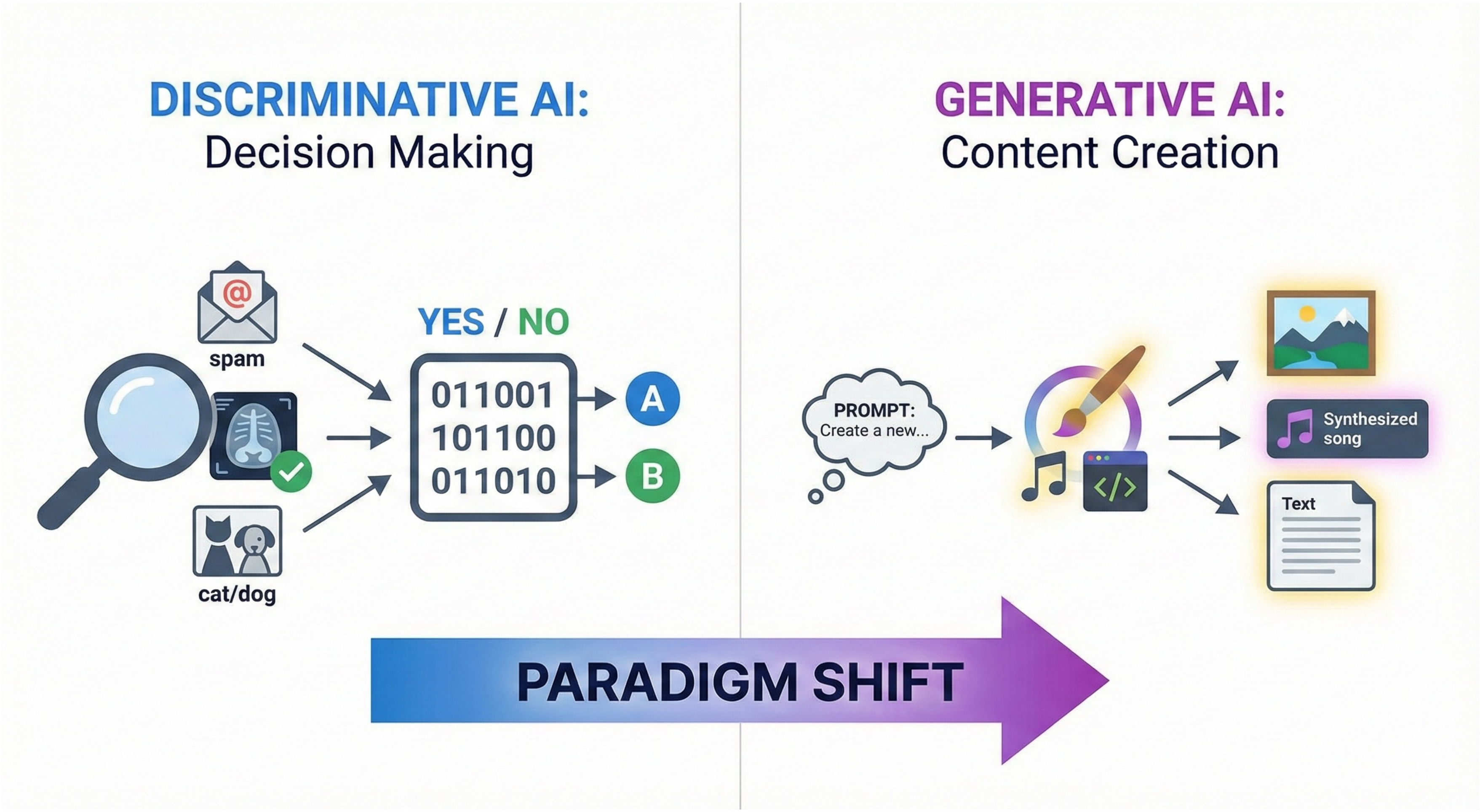
**Is this X or Y? → "Decision Making"**

The Era of Discrimination

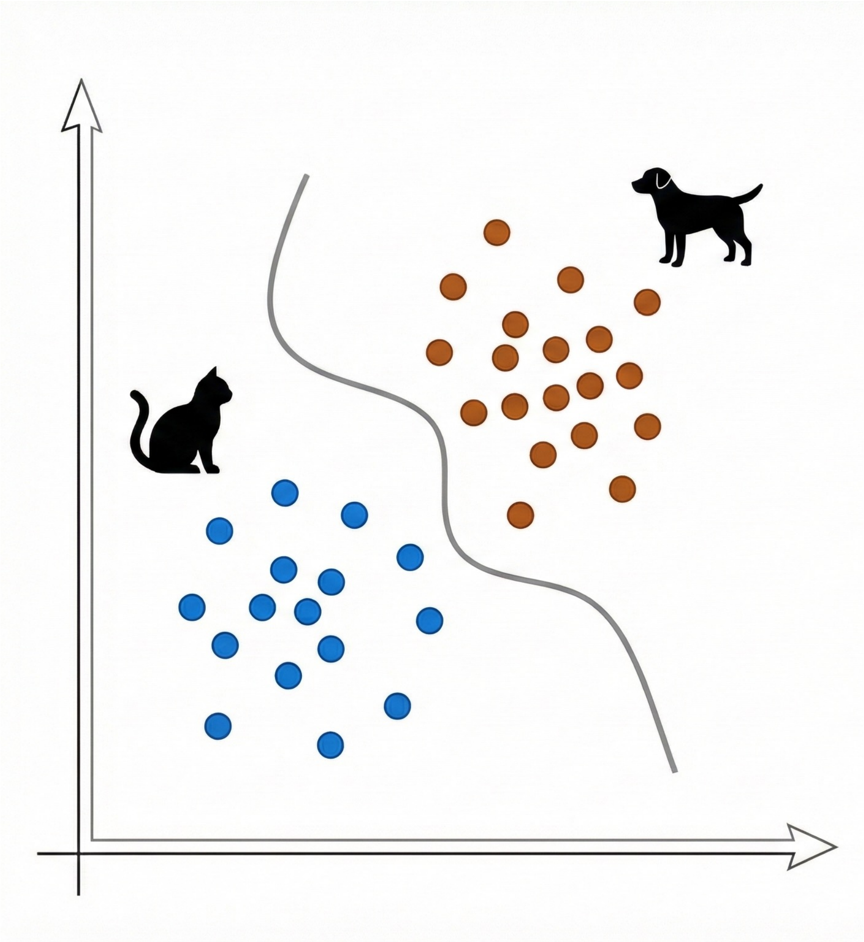
**Create an instance of X → "Content Creation"**

The Era of Generation

# Paradigm Shift: Discriminative Models → Generative Models

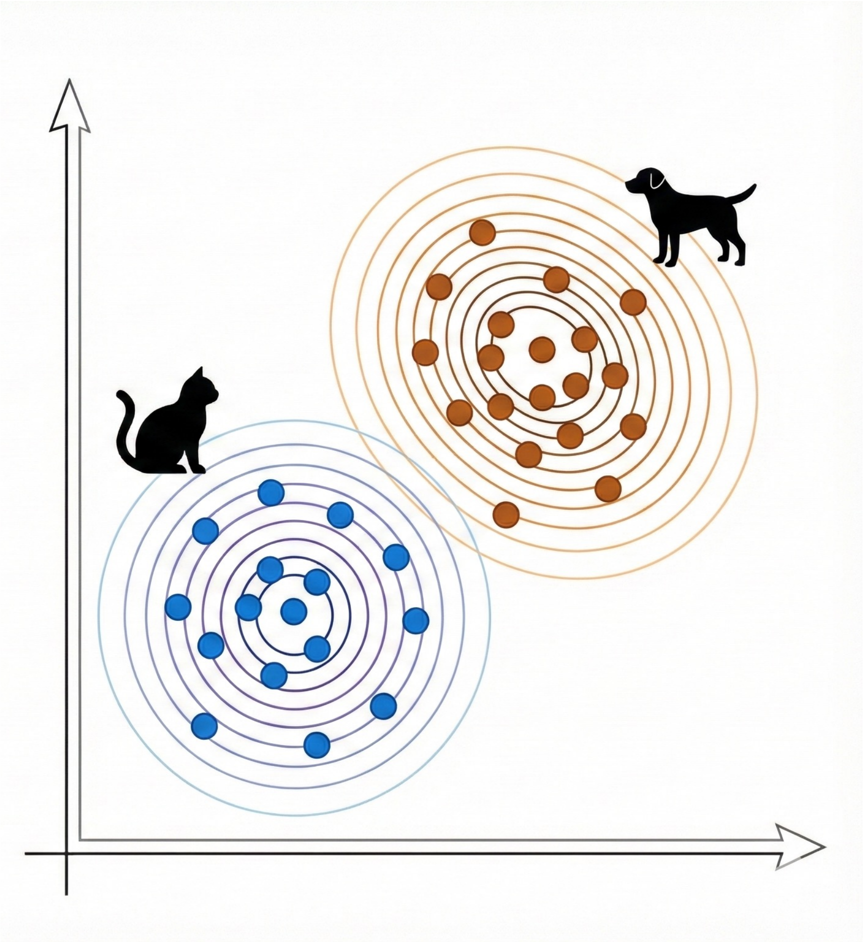


# Discriminative Models vs. Generative Models



**Predict the label given the Input**

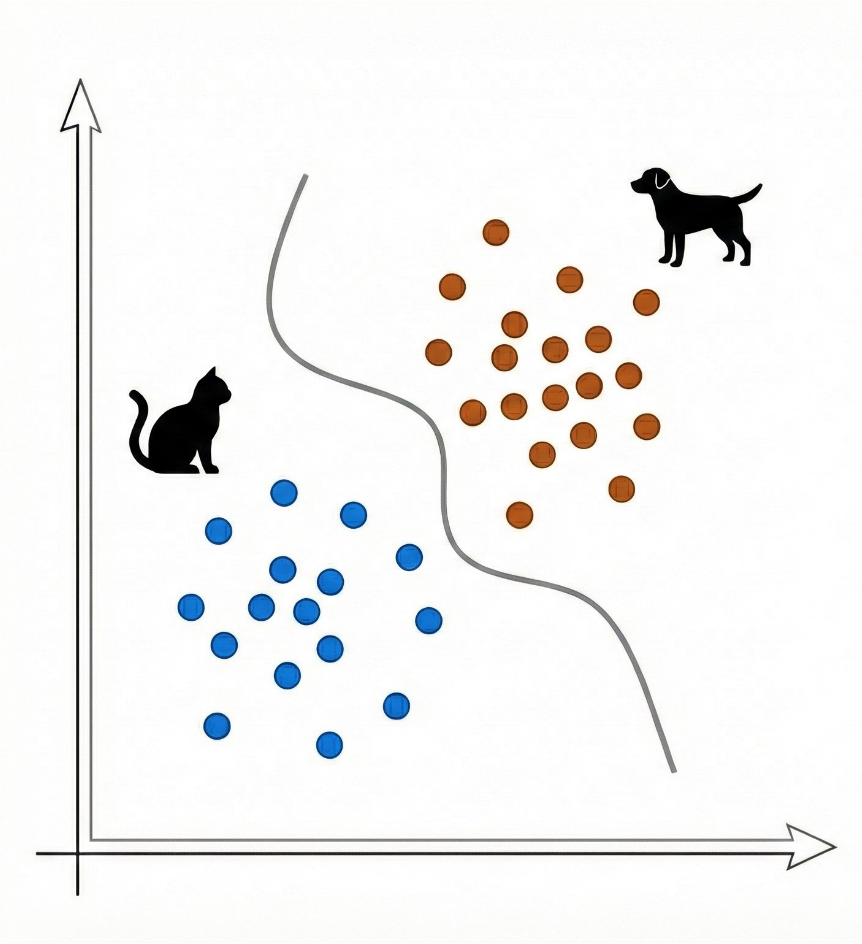
**Learns $P(Y/X)$**



**Generate new content**

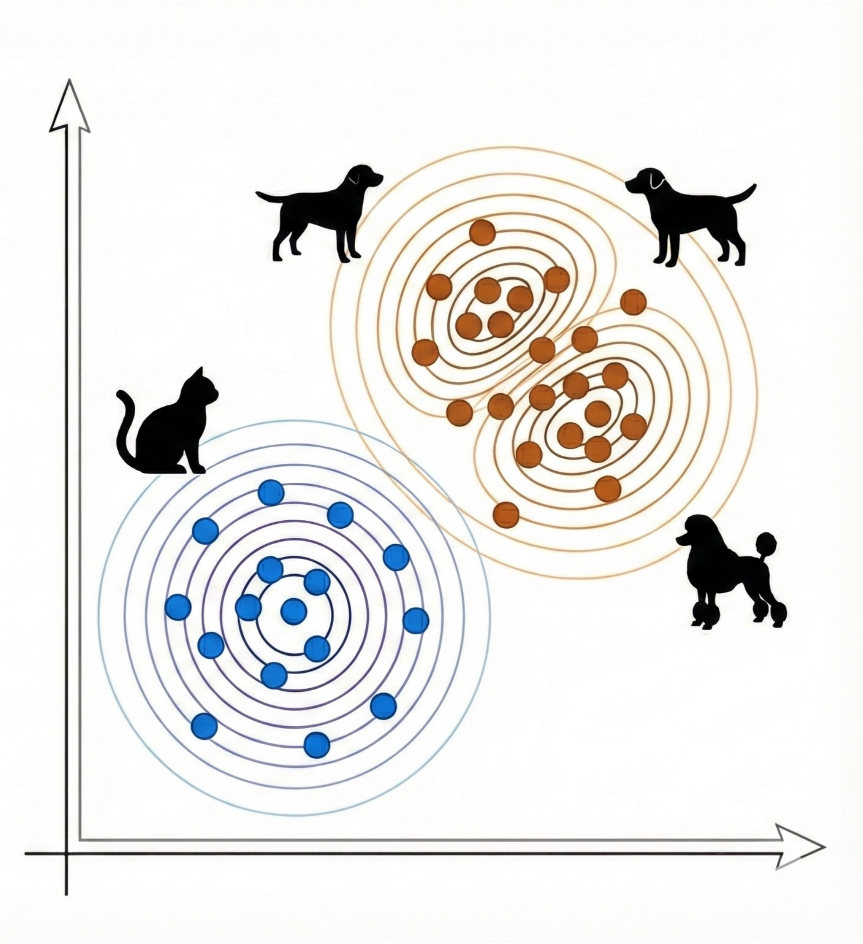
**Learns $P(X,Y)$**

# Discriminative Models vs. Generative Models



Predict the label given the Input

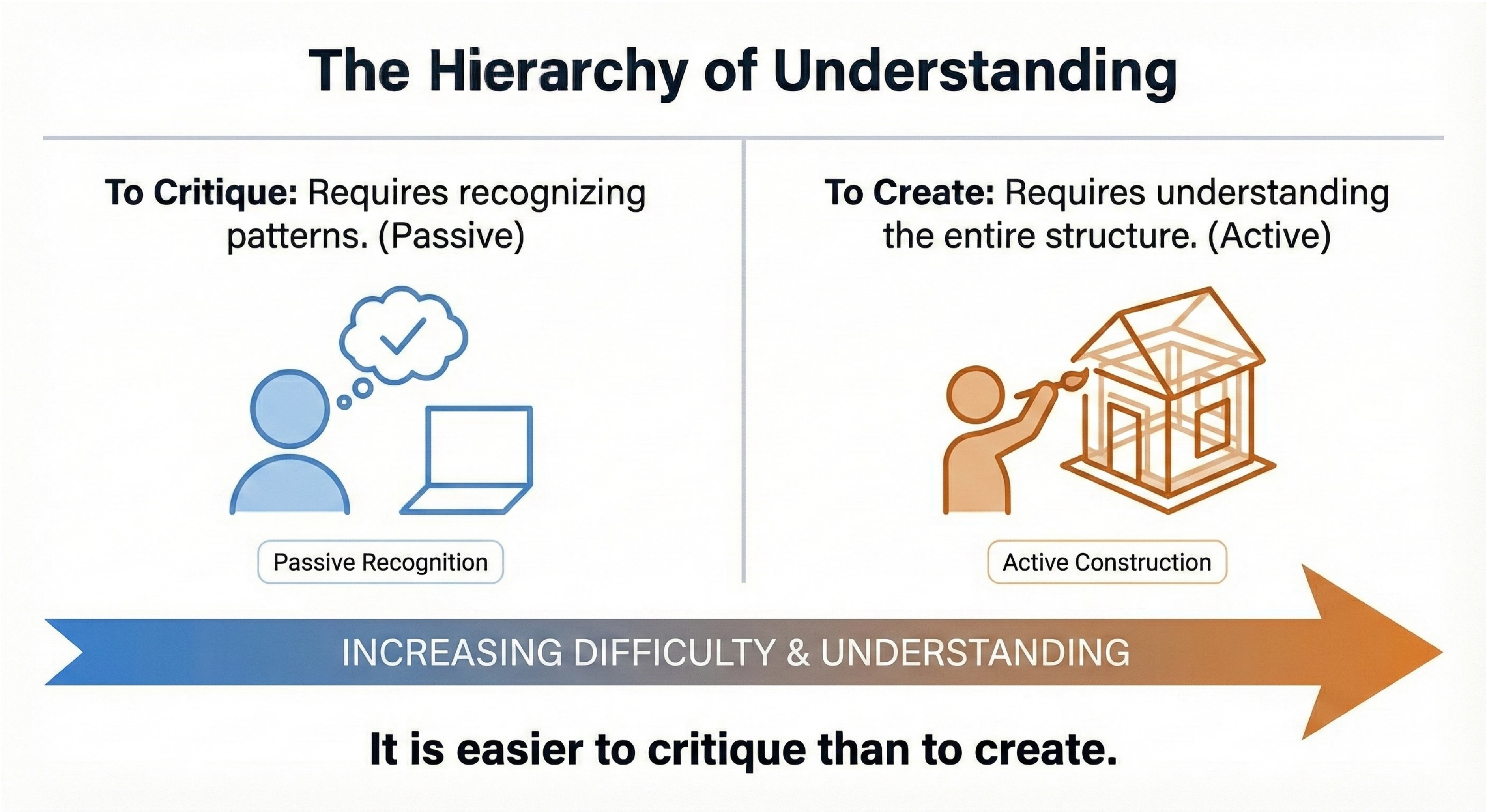
Learns $P(Y/X)$



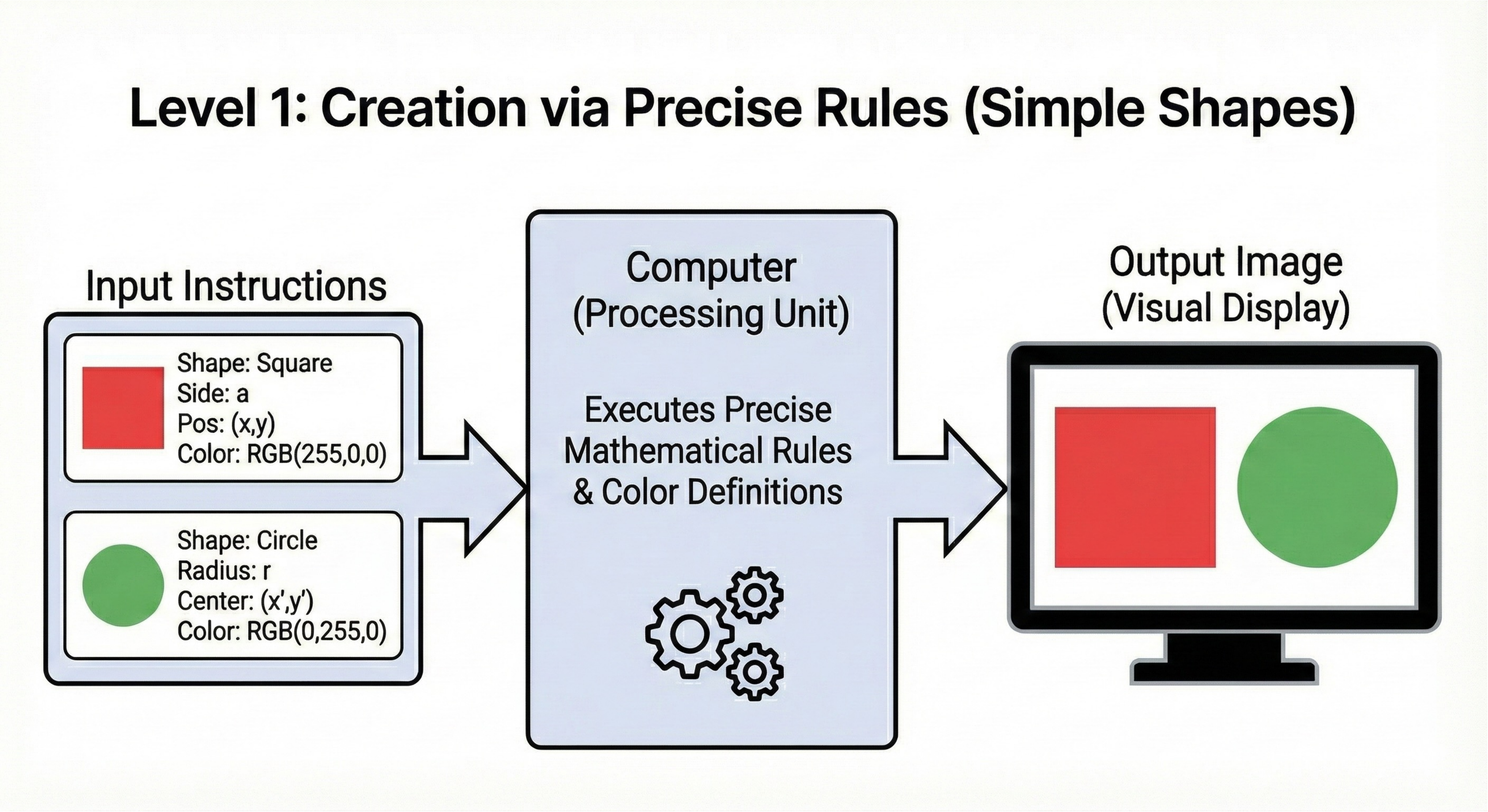
Generate new content by understanding **the abstract patterns of the data**

**Learns** $\boldsymbol{P(X,Y)}$

# Decision Making vs. Content Creation



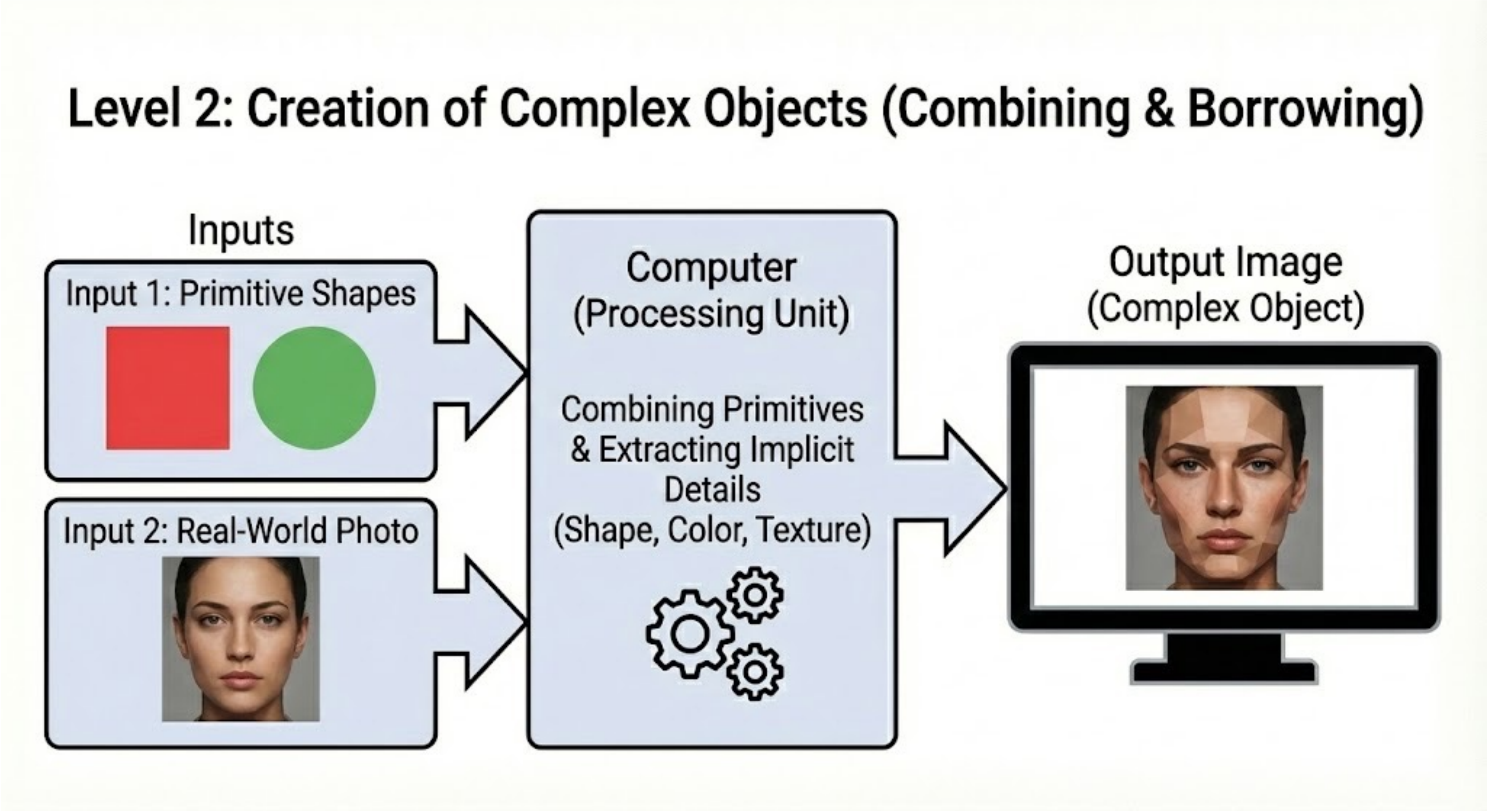
# How to Create Content?






Slide idea borrowed from Stephano Ermon

# How to Create Content?

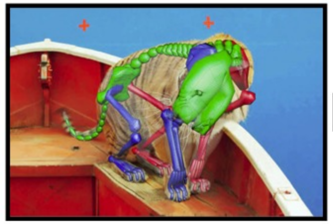





Slide idea borrowed from Stephano Ermon

# How to Create Content?

Ang Lee



Skeleton

Geometry

Texture

Details

Image

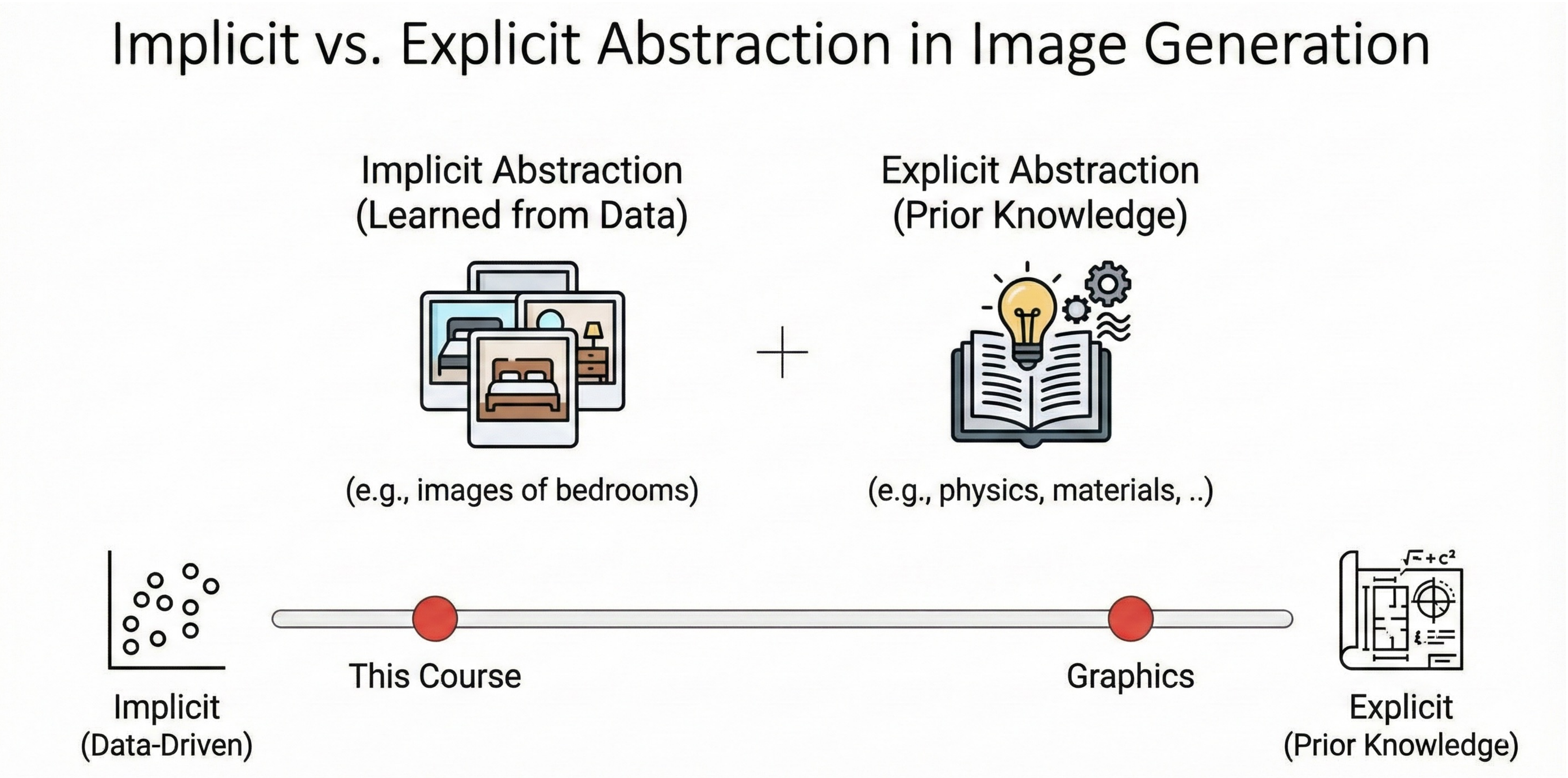
Idea


Visual Content



Slide borrowed from Jun-Yan Zhu

# How to Create Content?

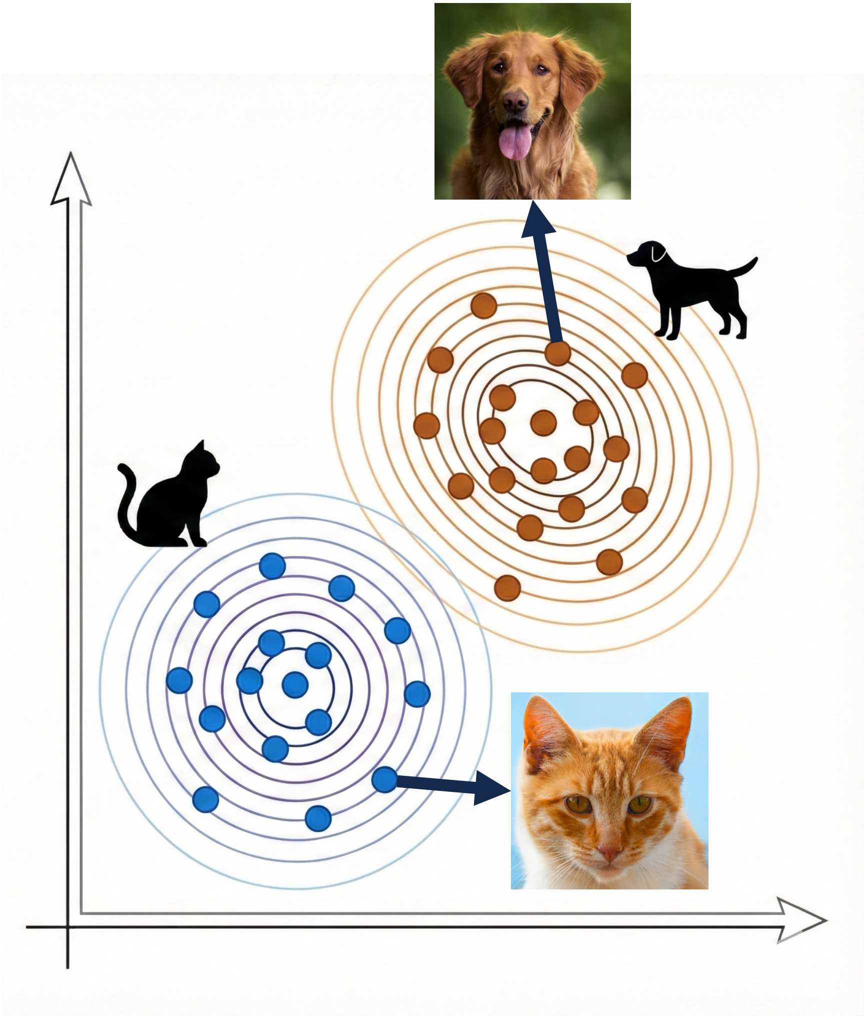



**This Course:** Teach Machines How to Create (Visual) Content



Slide idea borrowed from Stephano Ermon

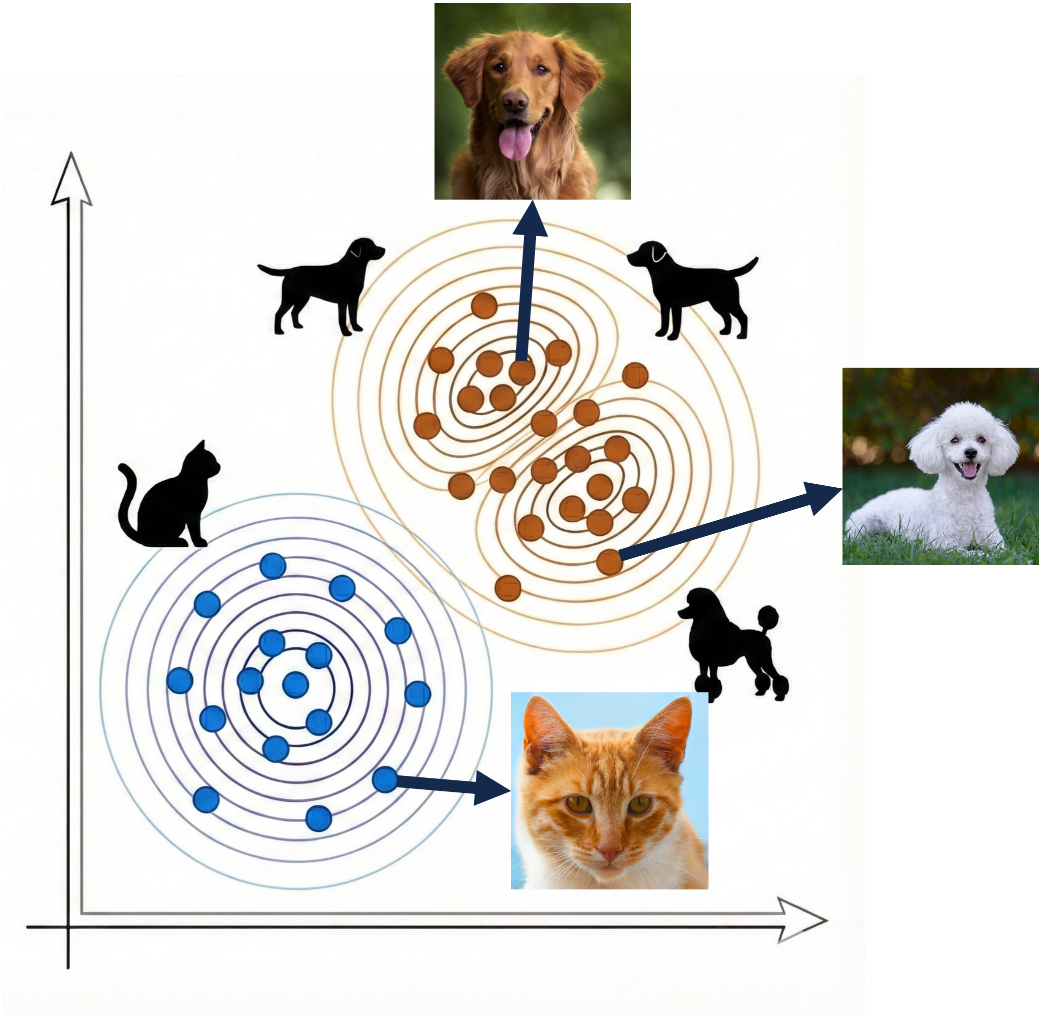
# Why Generative Models are Useful?



**1. Generate new content!**

Random Sampling from
Learned Data Distribution

# Why Generative Models are Useful?



**2.** ***Controllably*** **Generate New Content!**

*"Breed 1"*

*"Breed 2"*

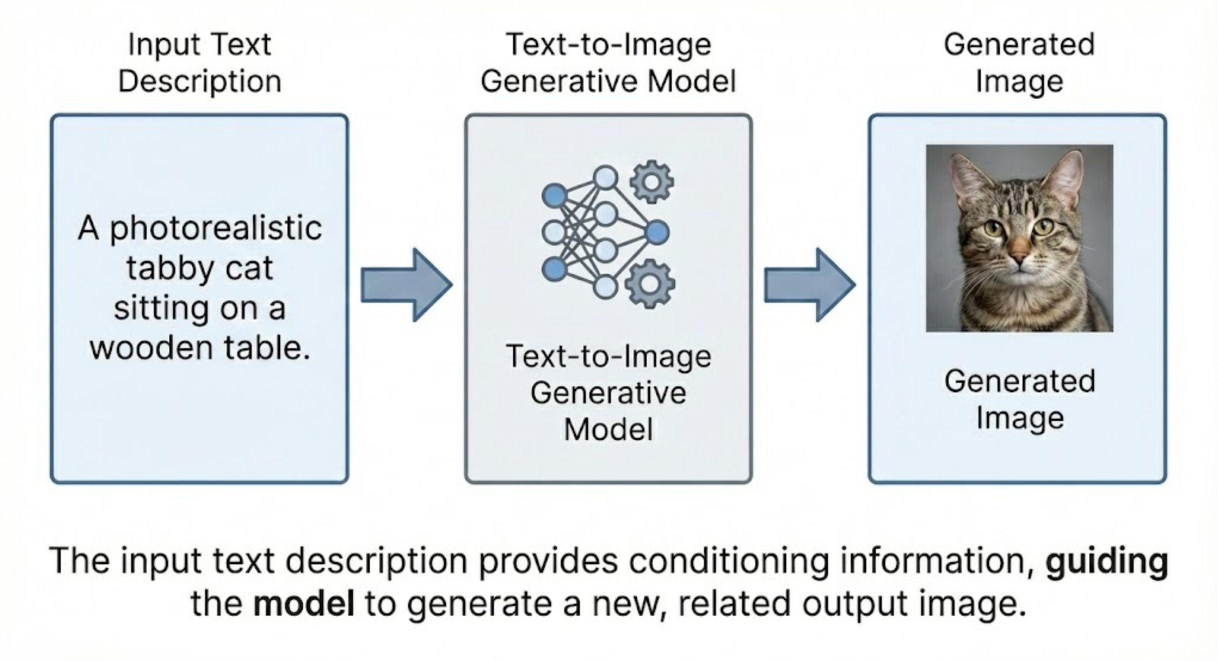
*Conditional* Sampling from Learned Data Distribution

# Why Generative Models are Useful?

*"A full frame mirrorless camera sitting on a heavy oak table. A very big expensive 35mm prime lens is attached. The lens faces towards the viewer but at a slight angle to the left."*

*"Present a clear, 45° top-down isometric miniature 3D cartoon scene of Freiburg, Germany, featuring its most iconic landmarks and architectural elements. Integrate the current weather conditions directly into the city environment to create an immersive atmospheric mood."*


**2a. Text Conditioned Generation**




*Text*-Conditional Sampling from Learned Data Distribution



https://gemini.google/overview/image-generation/

# Why Generative Models are Useful?

Image
Restoration

Image
Super-resolution


**2b. Image Conditioned Generation**

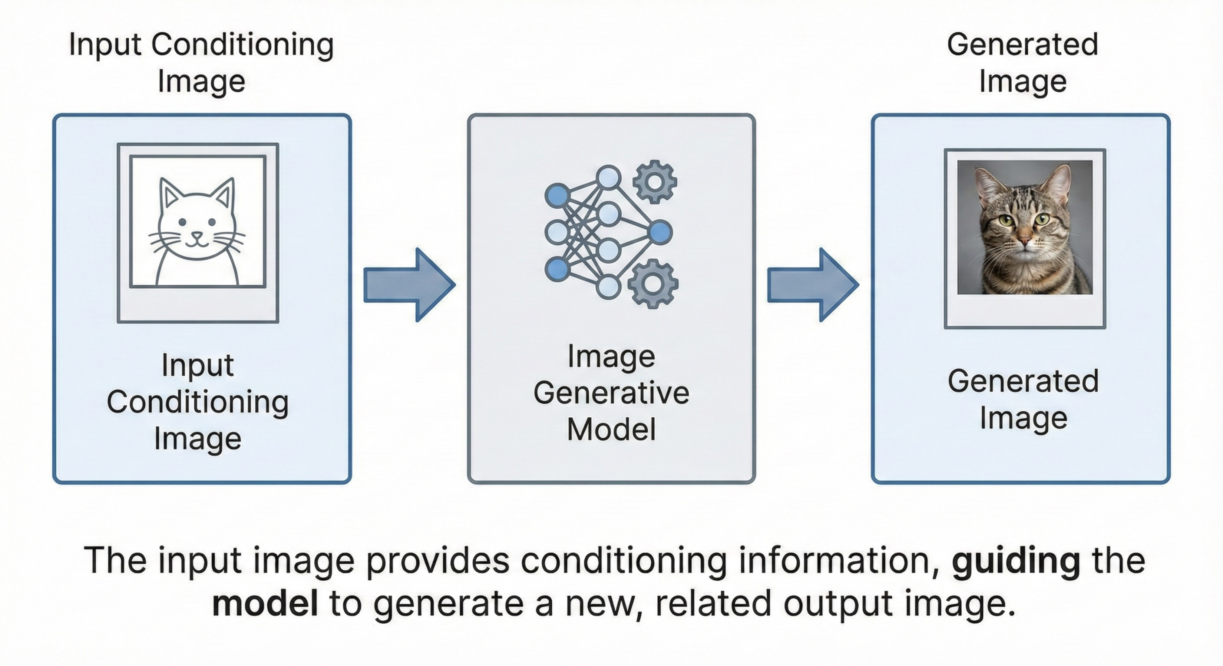



*Image*-Conditional Sampling from Learned Data Distribution

https://gemini.google/overview/image-generation/

# Why Generative Models are Useful?

*"Selfie of person 1 with person 2 on a busy New York street."*

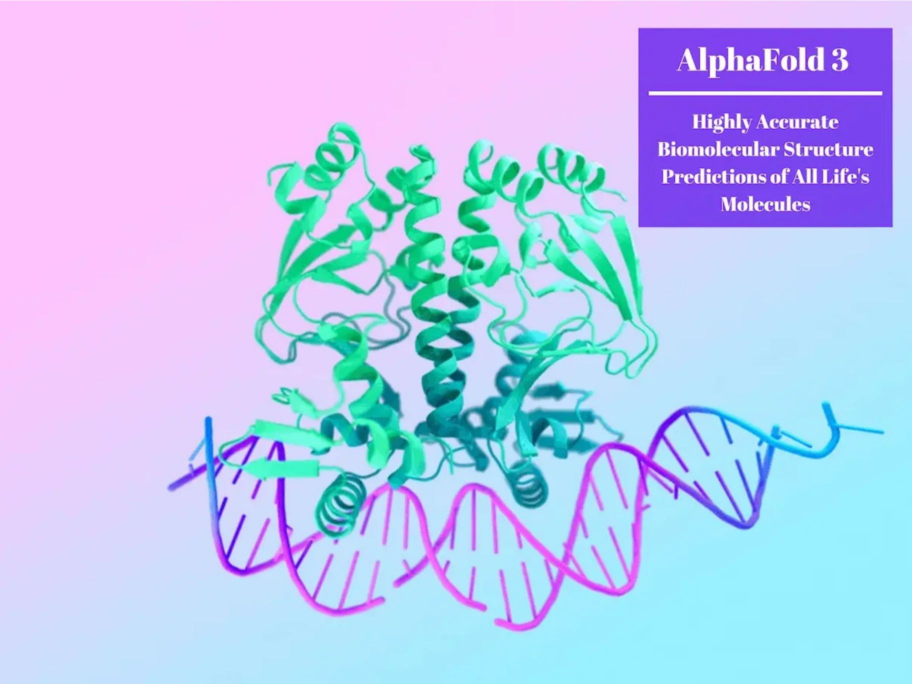
*"Create a 1/7 scale commercialised figurine of the characters in the picture, in a realistic style, in a real environment. The figurine is placed on a computer desk. The figurine has a round transparent acrylic base, with no text on the base."*


**2c. Image + Text Conditioned Generation**

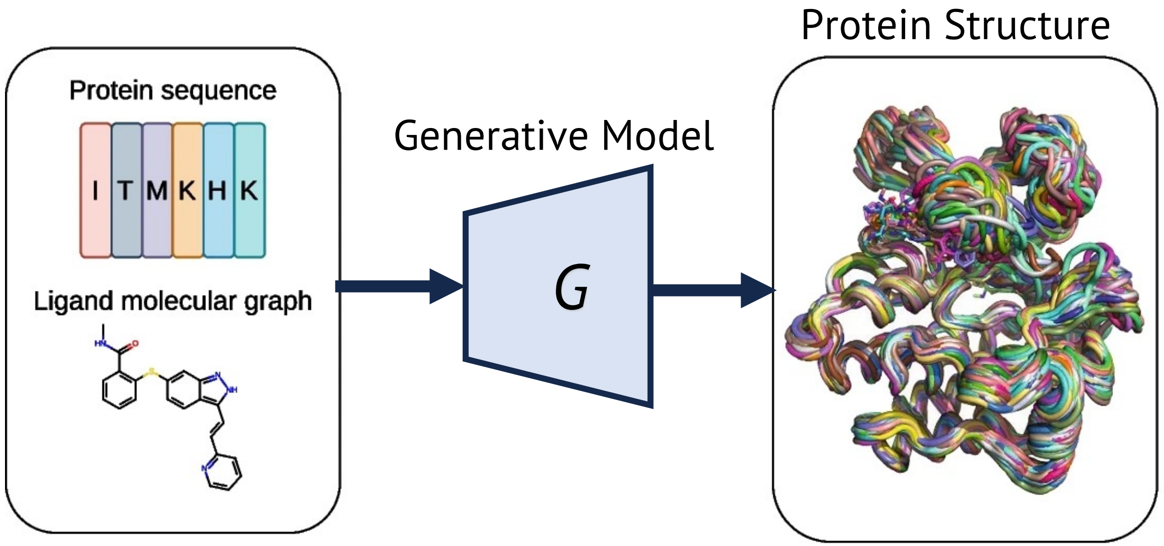
*"Futuristic car made out of cactus plant"*

*(Image + Text)*-Conditional Sampling from Learned Data Distribution

# Why Generative Models are Useful?




**2d. Structure-Conditioned Generation**

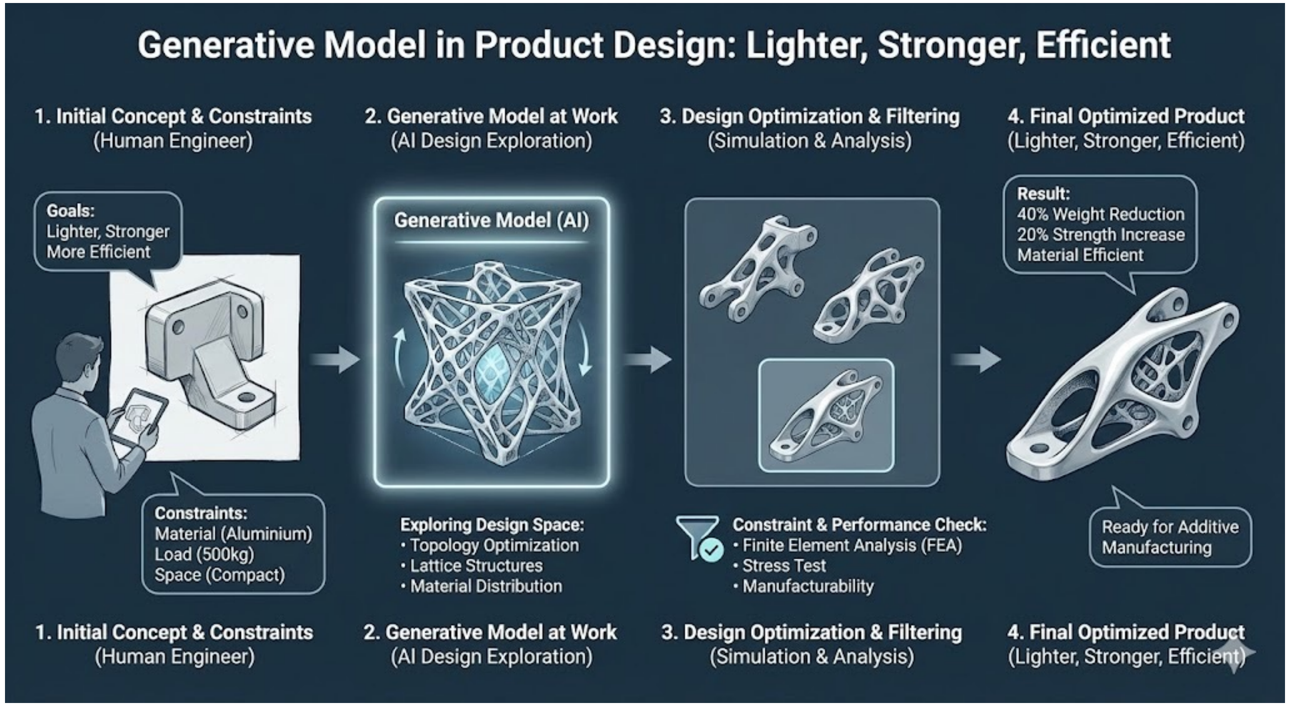



*(Structure)*-Conditional Sampling from Learned Data Distribution

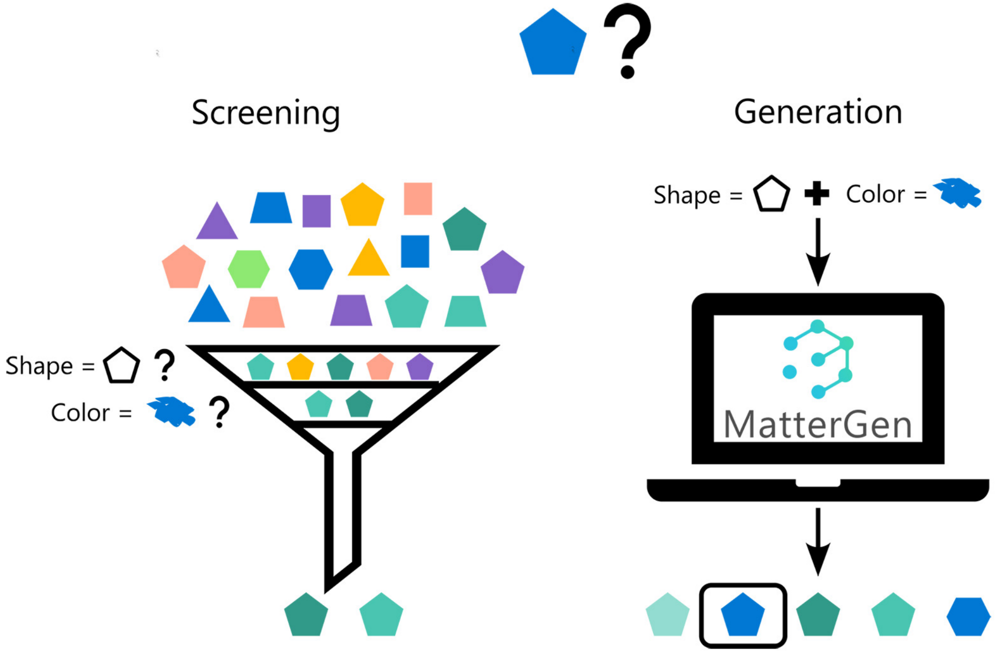


https://blog.google/technology/ai/how-we-built-alphafold-3/

# Why Generative Models are Useful?



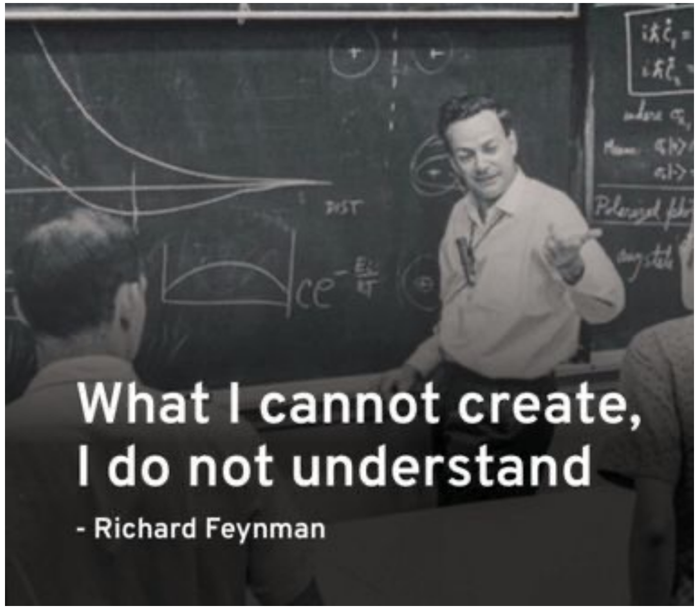

**2d. Structure-Conditioned Generation**




*(Structure)*-Conditional Sampling from Learned Data Distribution

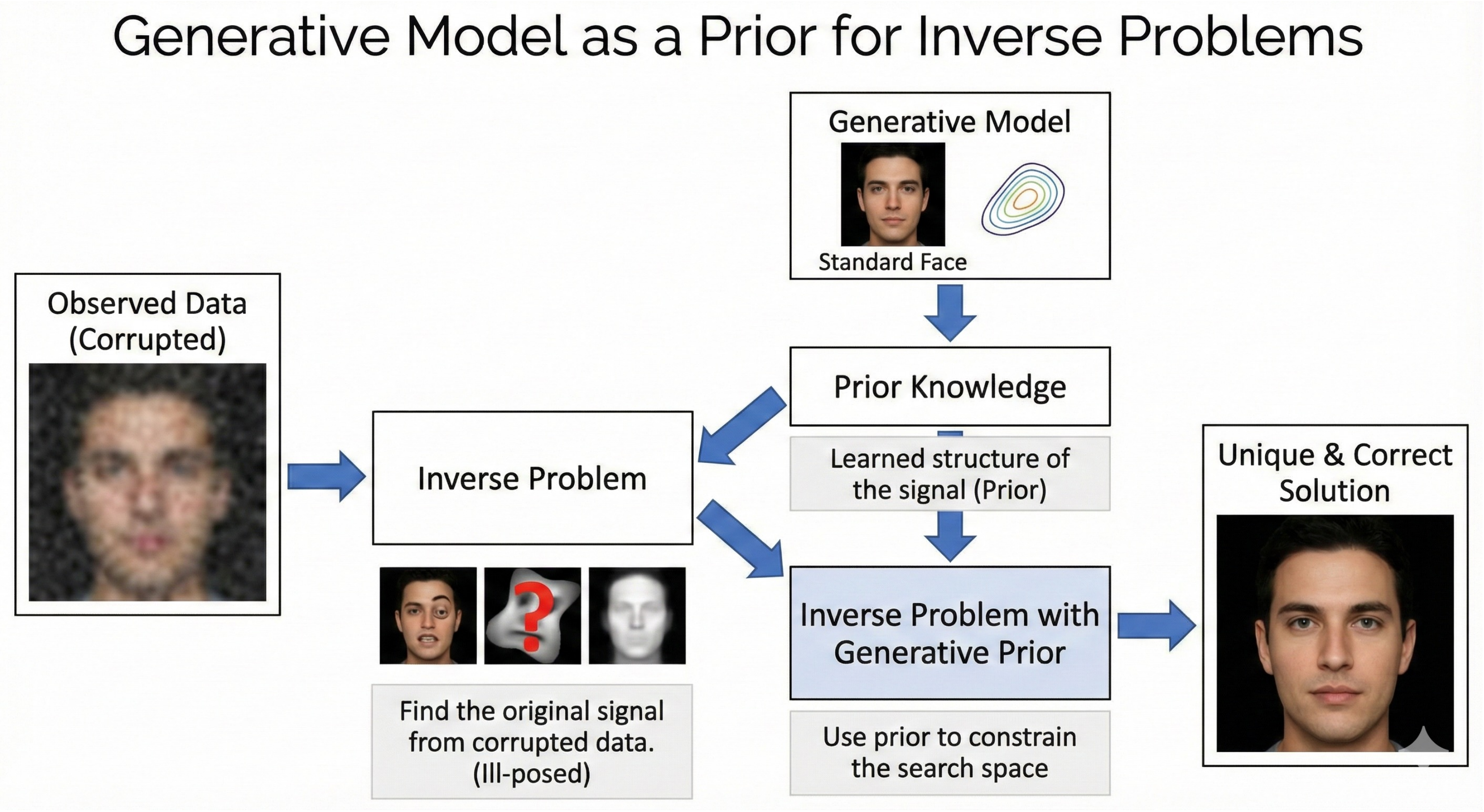
https://blog.google/technology/ai/how-we-built-alphafold-3/

# Why Generative Models are Useful?




**What I can create,**
**I understand !**

**3. Leverage the Rich Understanding for various tasks**




*(measurements)*-Conditional Sampling from Learned Data Distribution

# Inverse Problems of Image Restoration and Editing

**Image Restoration**

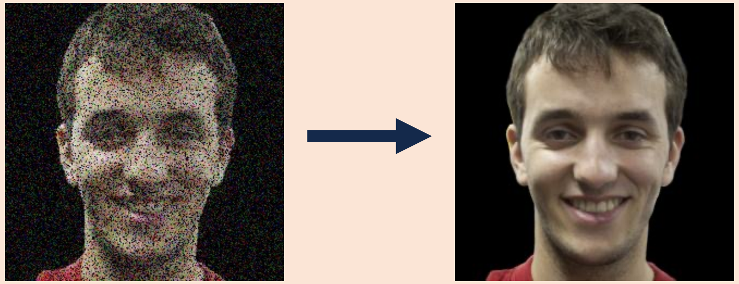


**Image Manipulation**

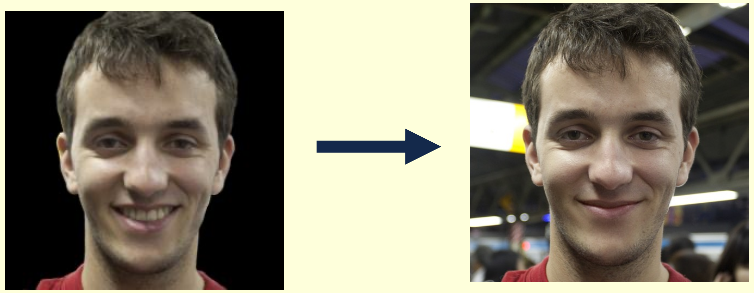
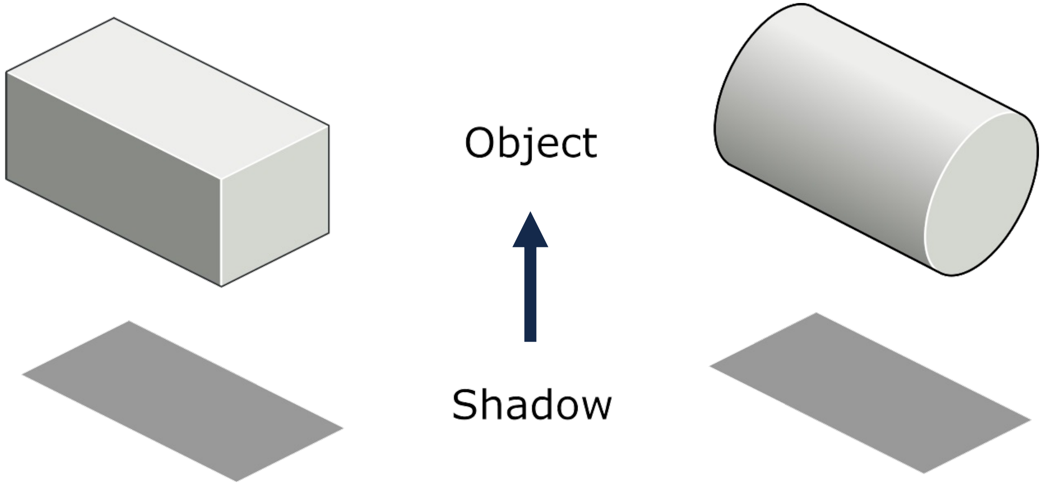


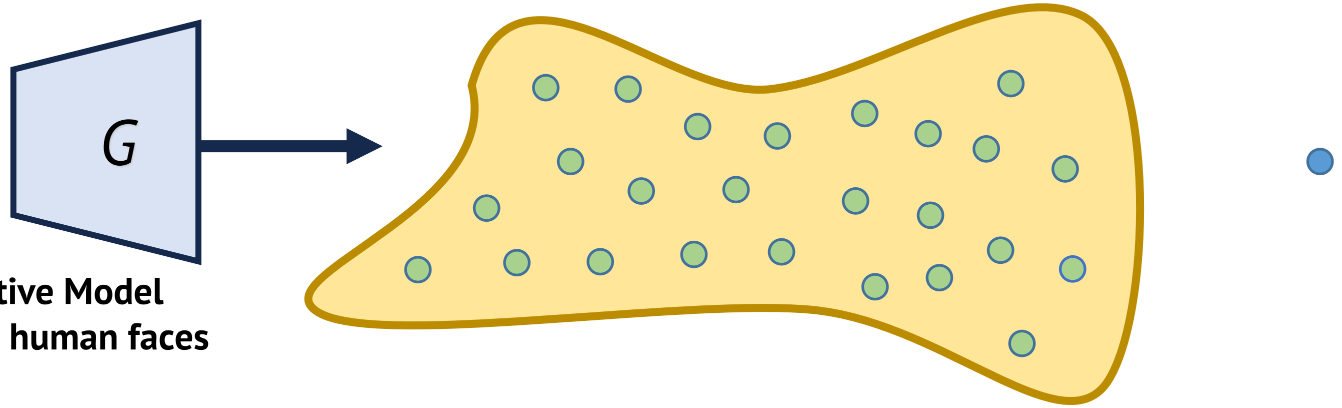
**Image Stylization**




Highly ill-posed: infinite many solutions!

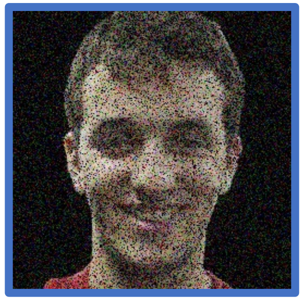
Additional Information about the target image is required → **can use Generative Models!**

# Image Restoration using Generative Models



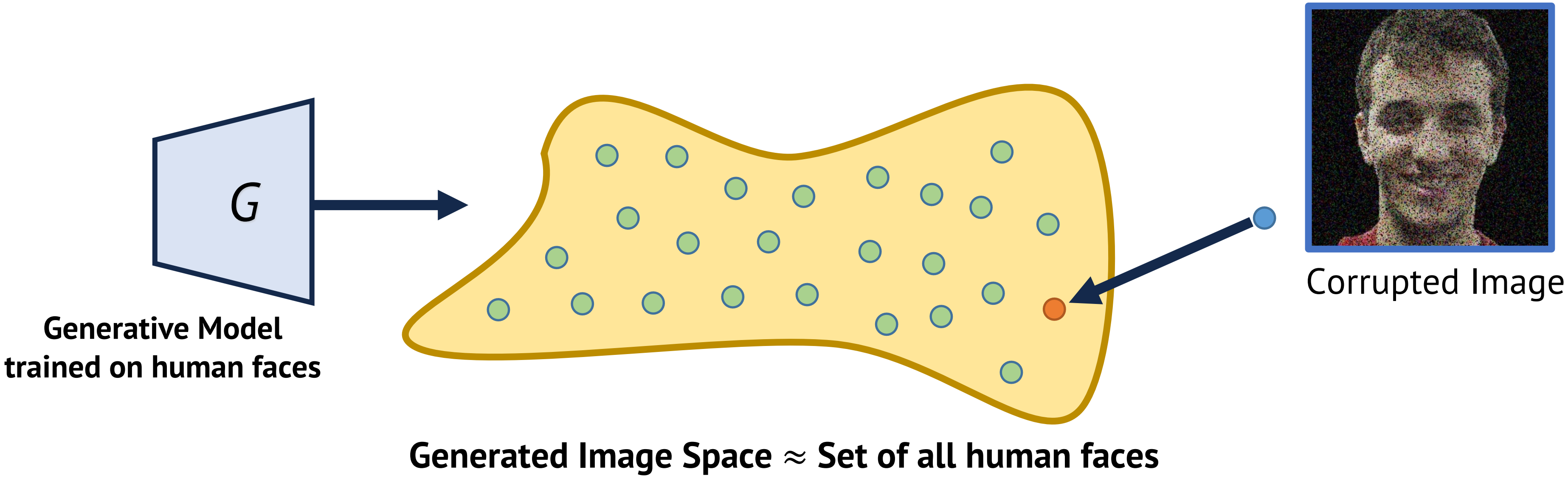

**Generative Model trained on human faces**

**Generated Image Space ≈ Set of all human faces**



Corrupted Image

# Image Restoration using Generative Models

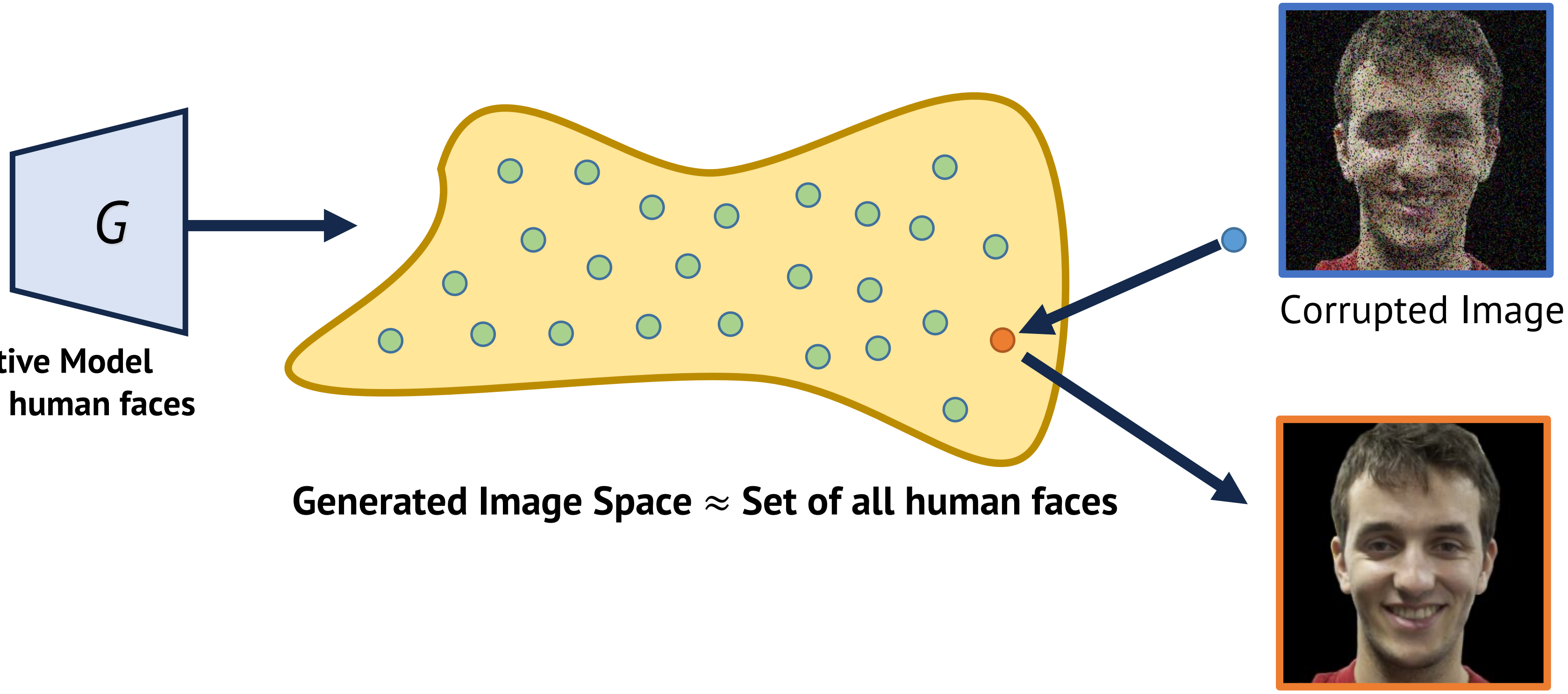



Find the closest image in the Generated Image Space



Solving Inverse Problems using GAN priors, V. Shah et al., ICASSP ‘18

# Image Restoration using Generative Models




Find the closest image in the Generated Image Space

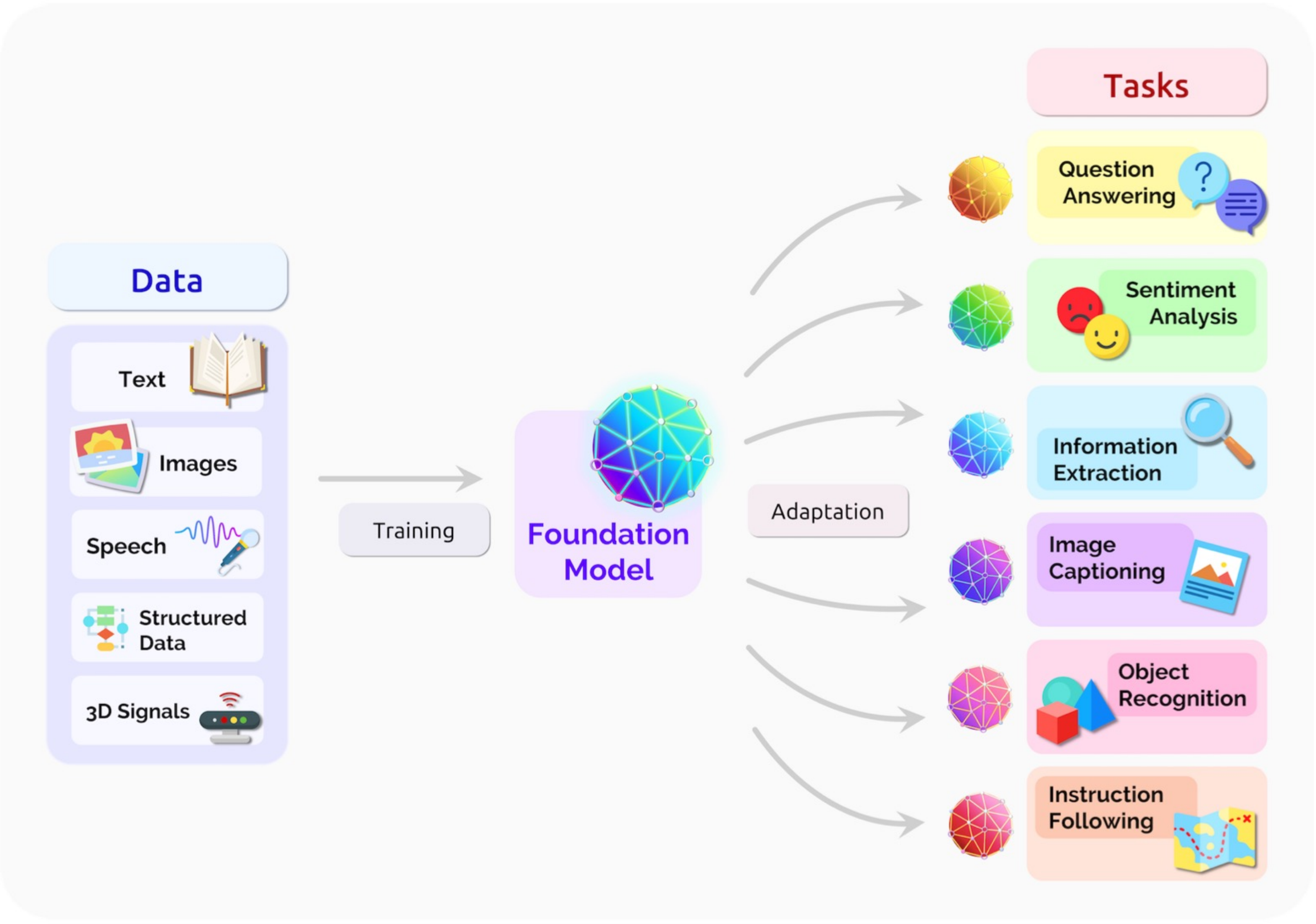
**Provable algorithms for solving:**

- Image Restoration
- Image Denoising, Inpainting etc.



Solving Inverse Problems using GAN priors, V. Shah et al., ICASSP '18

# Why Generative Models are Useful?






[On the Opportunities and Risks of Foundation Models, Bommasani et al., arXiv 2022]

# In This Course…

- *Foundational Math* behind Generative Model
- How to *train* a Generative Model ?
- How to *leverage* a pre-trained Generative Model for a specific application?
- Overview of the *literature* in this (ever-growing) research area
- *Implementation*
- *Key challenges, cultural and ethical concerns, future directions …*

# Course Guidelines …

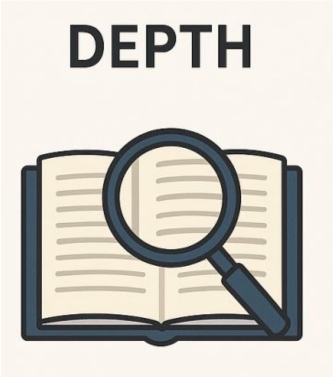
1) It is a *short* course!
2) Detailed course structure is available on: sc395.virajshah.com
3) Readings for each lecture will be posted on the website
   a) Highly recommended; remember point 1!
   b) Notes might be redundant; can share additional materials on request
4) Office hours on request;
   a) please write to vjshah3@Illinois.edu with 3-4 available slots.
5) Attendance is mandatory (remember point 1)!
6) (anon.) Feedback form will be shared in each class. (highly appreciated!)
7) It is my first time teaching a course!

# Prerequisites: Does this ring a bell?

| Probability | Neural Networks | Implementation |
|---|---|---|
| • Probability Distributions<br>• Conditional Probability<br>• Gaussian Distribution<br>• Divergence / KL divergence | • Backpropagation<br>• CNNs<br>• Transformers<br>• Matrix Multiplication | • Python<br>• PyTorch |

**Refreshers available on course website**

# Introduction



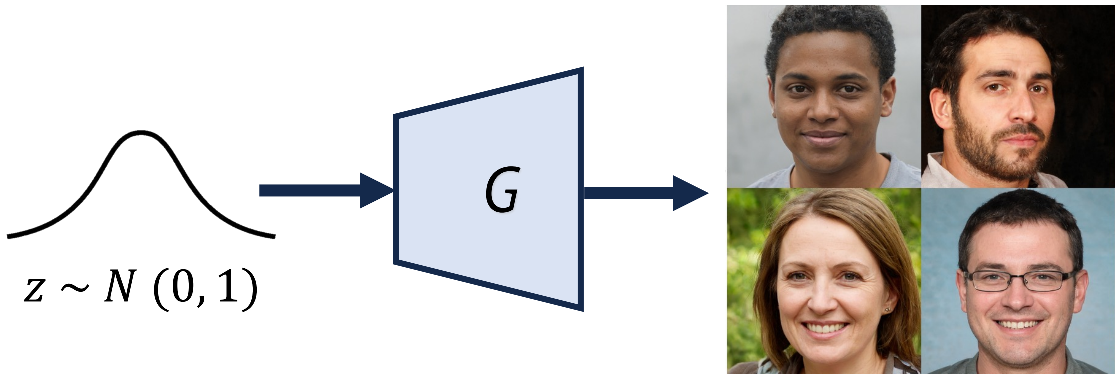

Various approaches to Generative Modeling

GANs: Training and Theory

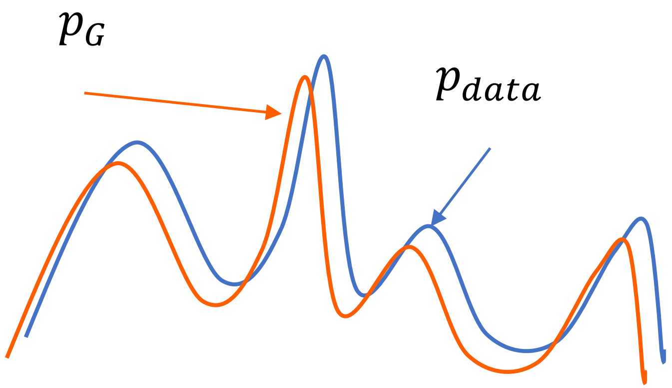
GANs: Applications

……

# Generative Models as Distribution Learners




**Mimicking the Data Distribution**




Generative Models ≈ 'Distribution Learners'



Ref: GANs, Goodfellow et al. NeurIPS '14

# Refresh: Probability Distribution

- Generative modeling = modeling the distribution of data $p_{data}$

- **Continuous** distribution: the data can take an infinite number values
  - Integral of the probability **density** function is 1: $\int_x p(x)dx = 1$
  - Image, Audio

- Multivariate Distribution: $p(\boldsymbol{x}) = p(x_1, x_2, \dots, x_D)$

# Naïve Attempt: Maximum Likelihood Estimation

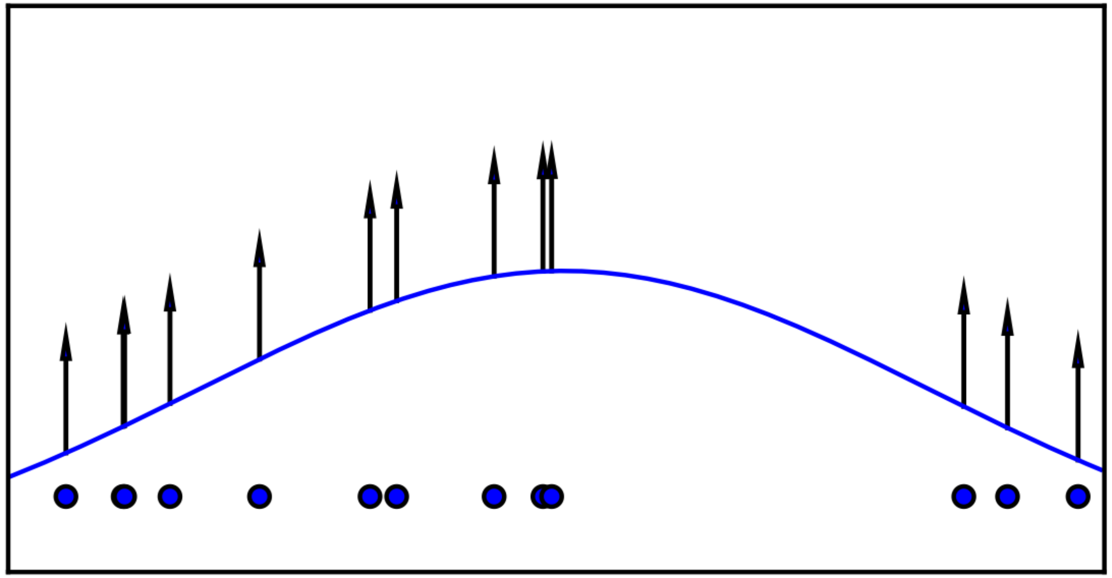
- Our data distribution is $p_{data}(\boldsymbol{x})$ : we have $m$ samples: $(x^{(1)}, x^{(2)}, \dots, x^{(m)})$

- Our generative model is parameterized by $\theta$,
  - Likelihood of each sample is given by: $p_{model}\left(x^{(i)}; \theta\right)$

- MLE: choose the parameters for the model that maximize the likelihood of the training data:

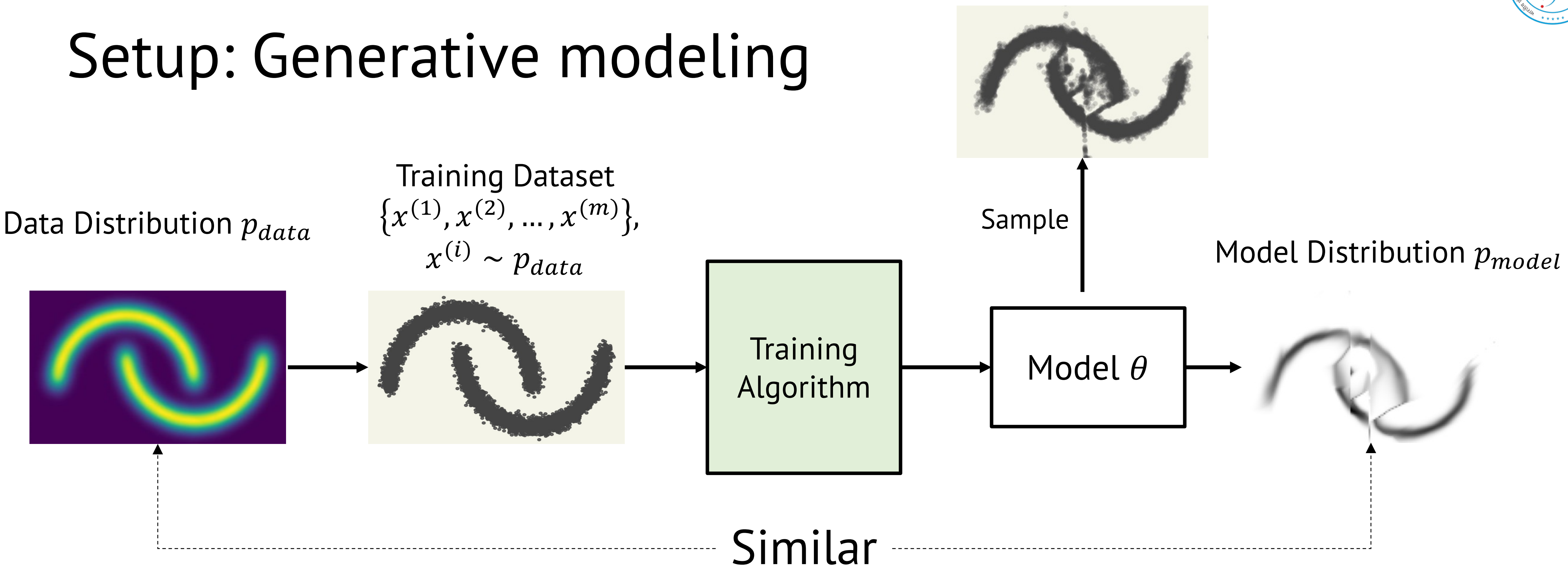
$$\theta^* = \arg\max_{\theta} \prod_{i=1}^{m} p_{model}\left(x^{(i)}; \theta\right)$$

# Maximum Likelihood Estimation



$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \mathbb{E}_{x \sim p_{\text{data}}} \log p_{\text{model}}(\boldsymbol{x} \mid \boldsymbol{\theta})$$

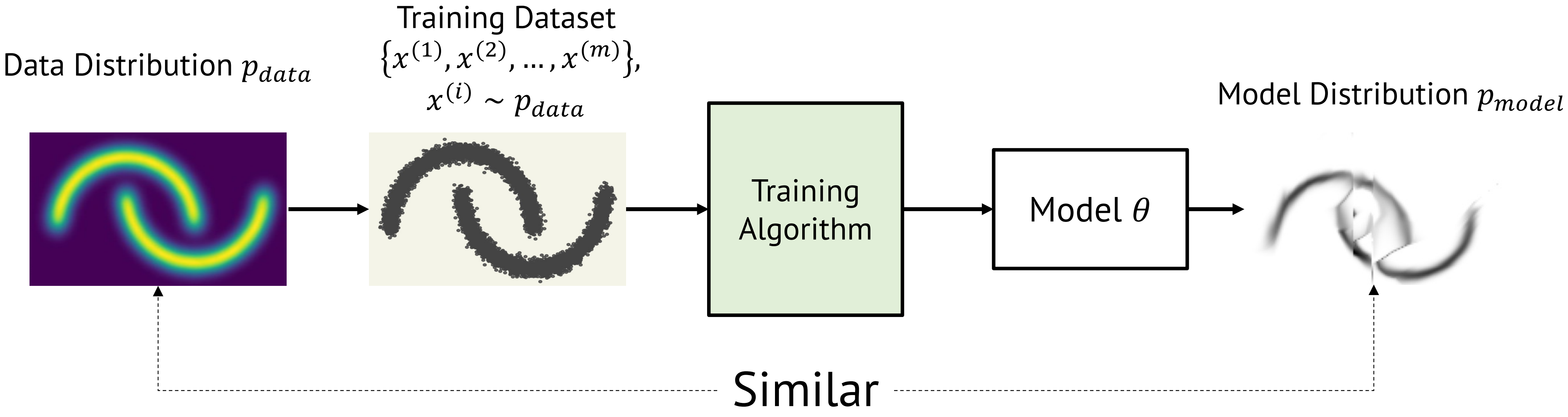
# Setup: Generative modeling




Estimating $p_{model}(x)$ and sample from $p_\theta(x)$ are often different processes:

- **Density estimation**: Input $x$, output $p_{model}(x)$
- **Generation/sampling**: No input, output a sample $x' \sim p_\theta$

# Setup: Generative modeling

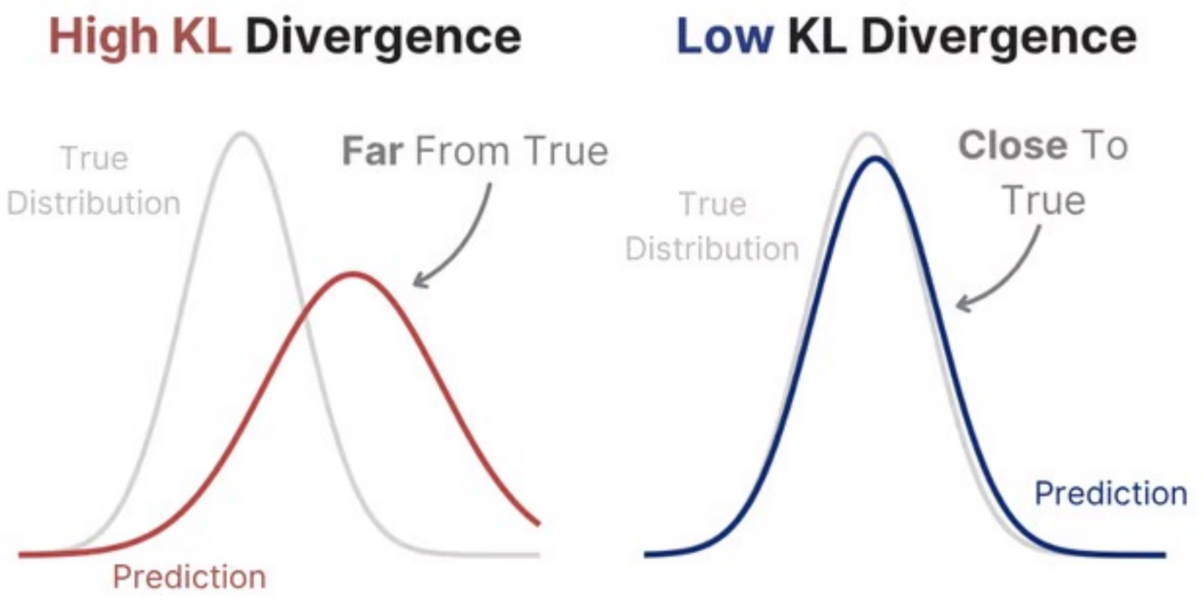



**How to measure distance between distributions?**

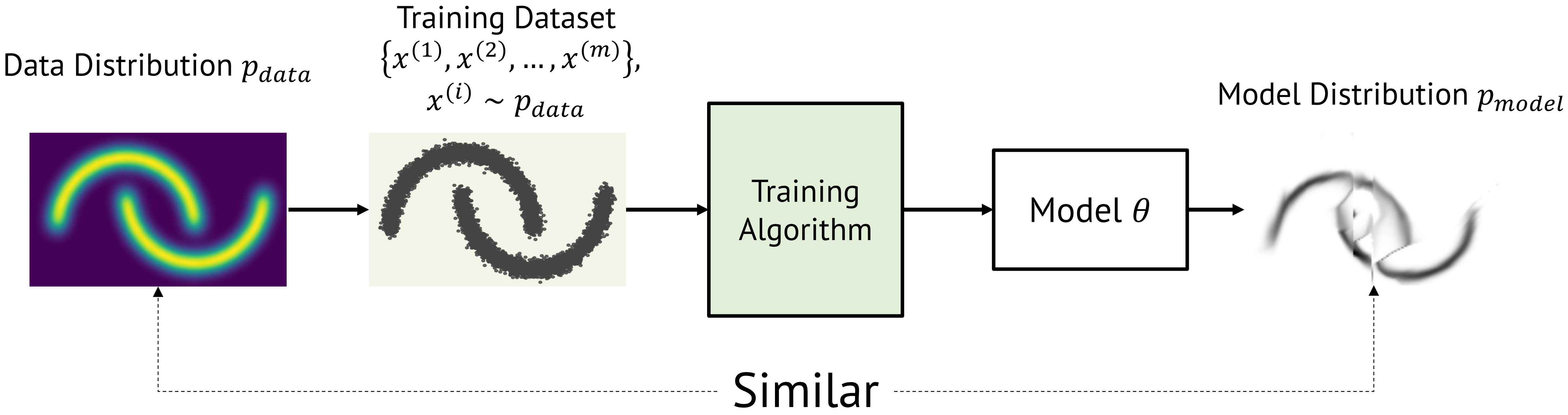
# Kullback–Leibler (KL) divergence

- Assume $p$ is the true distribution, and $q$ is the approximate one:

$$KL(p \parallel q) = \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)}$$



# Setup: Generative modeling

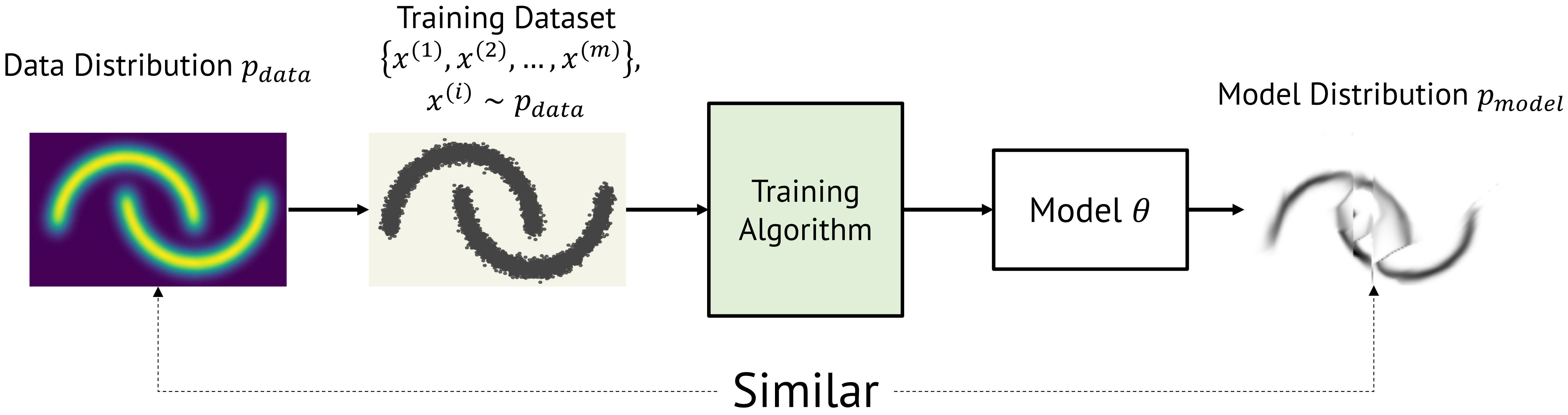



**Minimize KL Divergence**

# Minimize KL Divergence

$$KL(p \parallel q) = \sum_i p(x_i) \log \frac{p(x_i)}{q(x_i)}$$

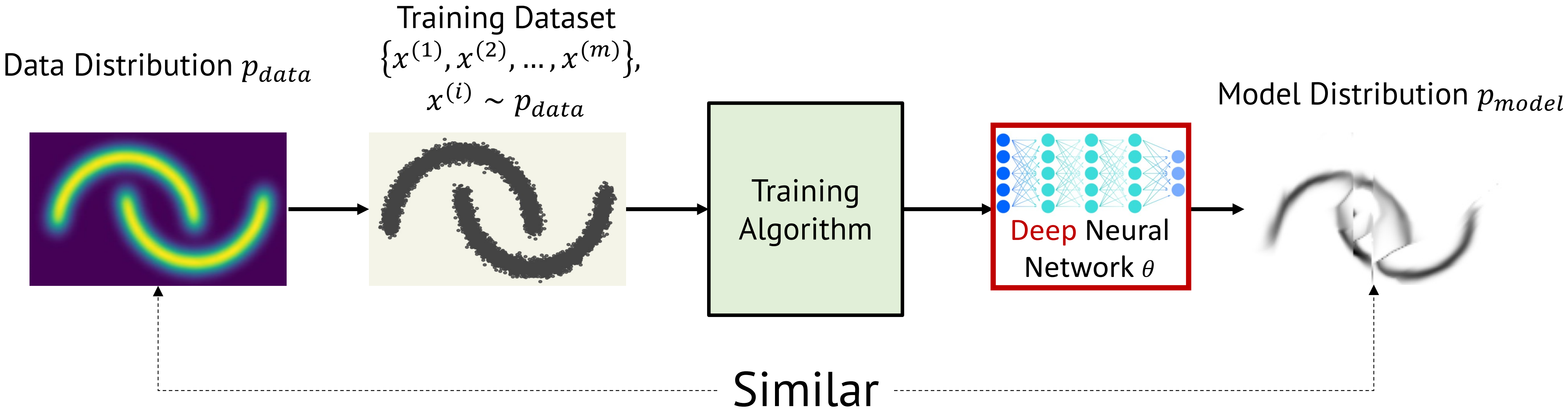
# Setup: Generative modeling




**Minimize KL Divergence**

$\approx$ Maximum Likelihood Estimation (MLE)

# Setup: Deep Generative Modeling




Slide from Deep Generative Modeling, CMU, 18-789

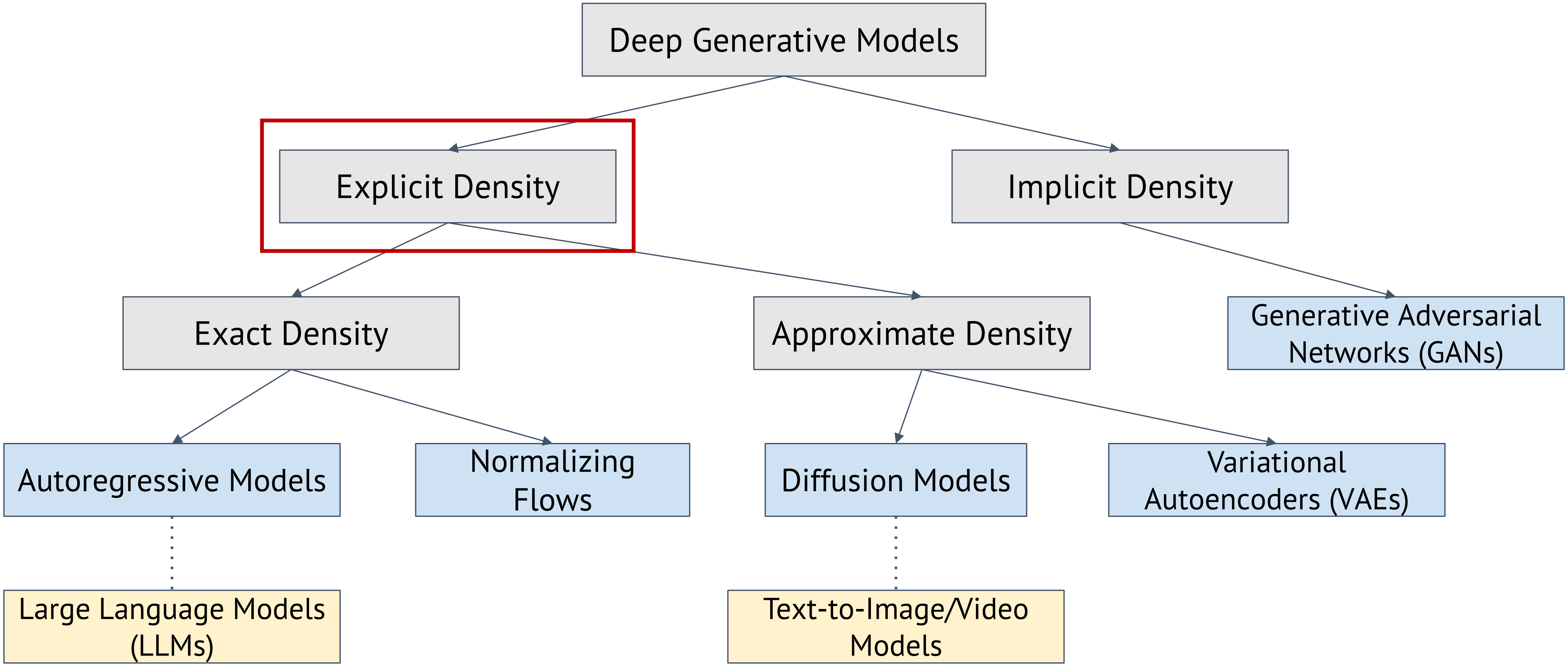
# Challenges of Deep Generative Modeling

- How to model the joint distribution of high-dimensional data?
  - Suppose each pixel is either black or white (Bernoulli distribution) and there are 28x28=784 pixels.
  - $2^{784} = 10^{236}$ possible outcomes! Dataset is too small !

MNIST Dataset (60k samples)

- In general, we do not know $p_{data}$!
- Do we need to estimate the Density?

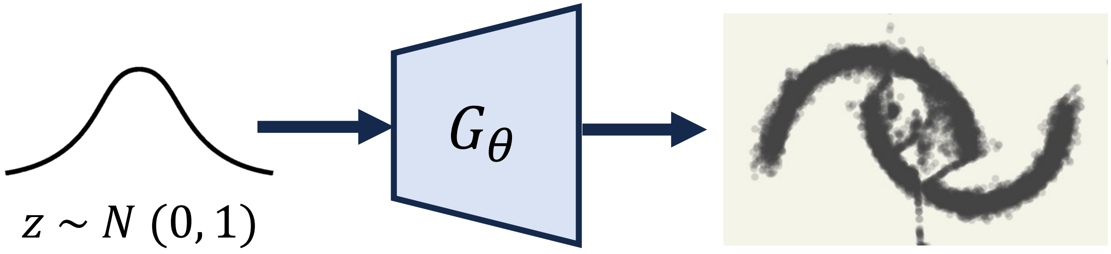
# Deep Generative Models




Slide from Deep Generative Modeling, CMU, 18-789

# Implicit Generative Models

- No explicit definition of the density function $p_{model}$
- Interact only indirectly with $p_{model}$:
  - via sampling from it
  - No need to parameterize $p_{model}$
  - Instead, use latent variable $z$ and generator $G_\theta$ such that $G_\theta(z) \sim p_{model}$




Samples from $p_{model}$

Please fill in the Feedback Form at:

https://virajshah.com/sc395-feedback1

# Thank You!