

CS 236: Deep Generative Models

Stefano Ermon
Stanford University
URL: deepgenerativemodels.github.io

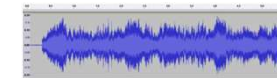
1

Introduction

Challenge: understand complex, unstructured inputs



Computer Vision



Computational Speech



Natural Language Processing

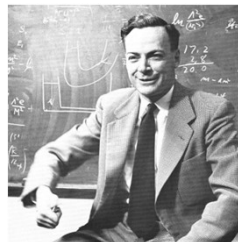


Robotics

2

2

Introduction



Richard Feynman: "What I cannot create, I do not understand"

Generative modeling: "What I understand, I can **create**"

3

3

Generative Modeling: Computer Graphics

How to generate natural images with a computer?

High level
description

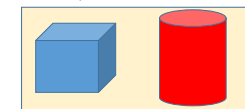
Cube(color=**blue**, position=(x,y,z), size=...)
Cylinder(color=**red**, position=(x',y',z'), size=..)

Generation (graphics)



Inference (vision as
inverse graphics)

Raw sensory
outputs



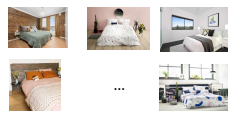
Many of our models will have **similar structure** (generation + inference)

4

4

Statistical Generative Models

Statistical generative models are **learned from data**



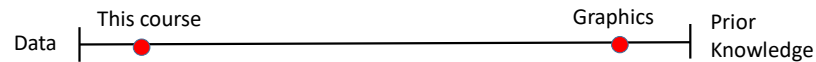
Data
(e.g., images of bedrooms)

+



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

Priors are always necessary, but there is a spectrum



5

Statistical Generative Models

A statistical generative model is a **probability distribution** $p(x)$

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

Image x



A probability
distribution
 $p(x)$

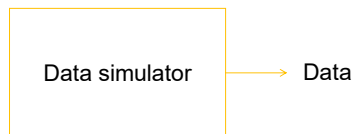
scalar probability $p(x)$

It is generative because **sampling from $p(x)$ generates new images**



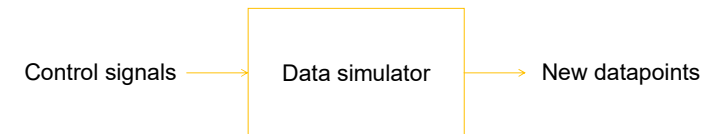
6

Building a simulator for the data generating process



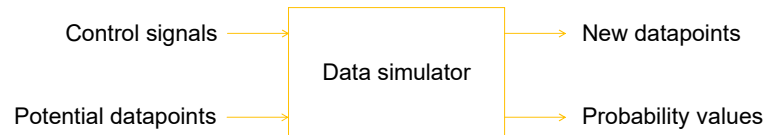
7

Building a simulator for the data generating process



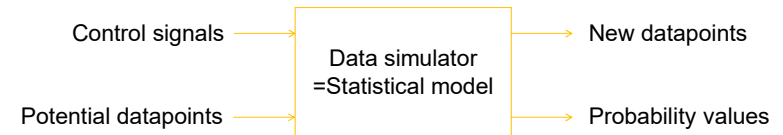
8

Building a simulator for the data generating process



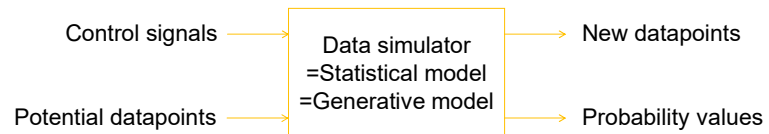
9

Building a simulator for the data generating process



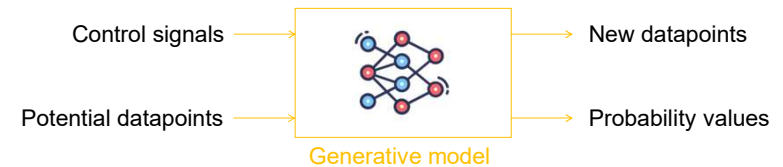
10

Building a simulator for the data generating process



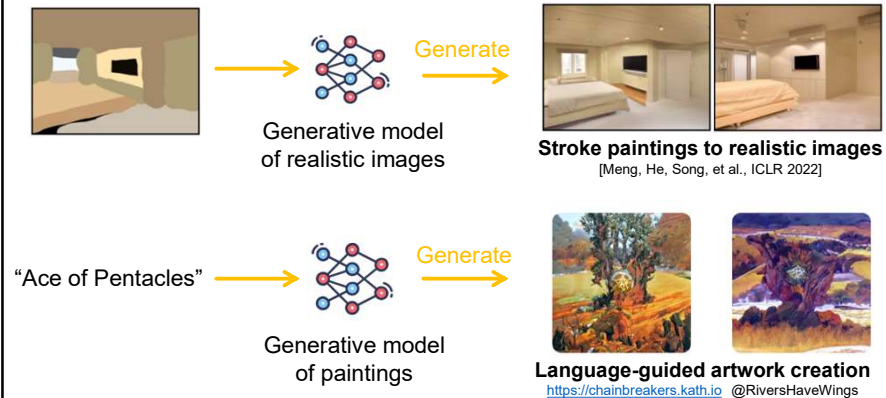
11

Building a simulator for the data generating process



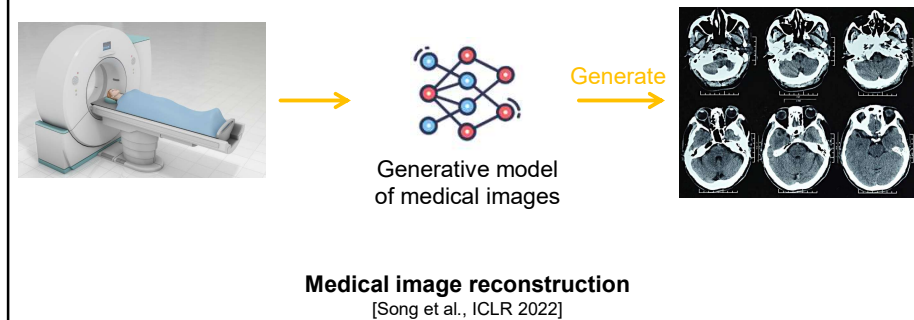
12

Data generation in the real world



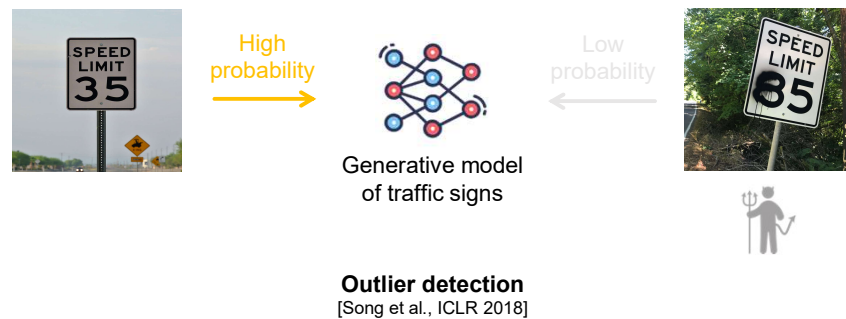
13

Solving inverse problems with generative models



14

Outlier detection with generative models



15

Discriminative vs. generative

Discriminative: classify bedroom vs. dining room

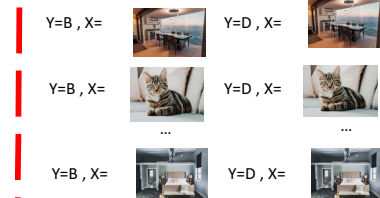


The image X is given. **Goal:** decision boundary, via **conditional distribution over label Y**

$$P(Y = \text{Bedroom} \mid X = \text{image})$$

Ex: logistic regression, convolutional net, etc.

Generative: generate X



The input X is **not** given. Requires a model of the **joint distribution over both X and Y**

$$P(Y = \text{Bedroom}, X = \text{image})$$

16

Discriminative vs. generative

Joint and conditional are related via **Bayes Rule**:

$$P(Y = \text{Bedroom} \mid X = \text{image})$$



$$P(Y = \text{Bedroom}, X = \text{image})$$



$$P(X = \text{image})$$



$$P(X = \text{image})$$



Discriminative: Y is simple; X is always given, so not need to model

Therefore it cannot handle missing data

$$P(Y = \text{Bedroom} \mid X = \text{image})$$



17

Images and Text

TEXT PROMPT an armchair in the shape of an avocado. . . .

AI-GENERATED IMAGES



Edit prompt or view more images +

P(image | caption)

TEXT PROMPT a store front that has the word 'openai' written on it. . . .

AI-GENERATED IMAGES



18

Conditional Generative Models

Class **conditional generative models** are also possible:

$$P(X = \text{image} \mid Y = \text{Bedroom})$$



It's often useful to condition on rich side information Y

$$P(X = \text{image} \mid Y = \text{caption} = \text{"A black table with 6 chairs"})$$



A discriminative model is a very simple conditional generative model of Y:

$$P(Y = \text{Bedroom} \mid X = \text{image})$$



19

Progress in Generative Models of Images -- GANs

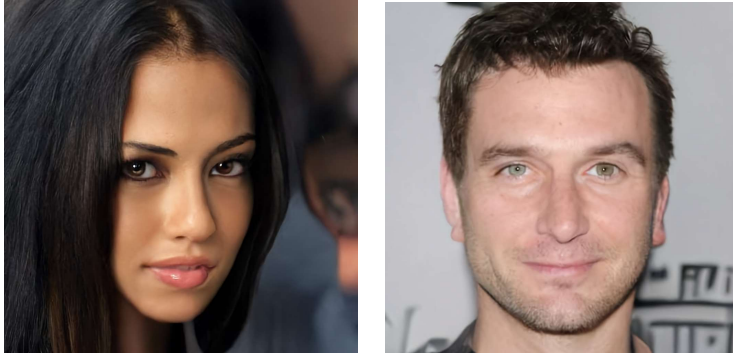


20

19

20

Progress in Generative Models of Images – Diffusion Models



Song et al., Score-Based Generative Modeling through Stochastic Differential Equations, 2021

21

21

Text2Image Diffusion Models

User input:

An astronaut riding a horse



22

22

Text2Image Diffusion Models

User input:

A perfect Italian meal



23

23

Text2Image Diffusion Models

User input:

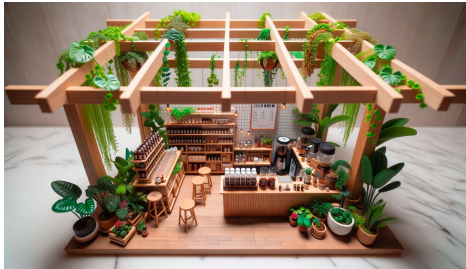
泰迪熊穿着戏服，站在太和殿前唱京剧
A teddy bear, wearing a costume, is standing in front of the Hall of Supreme Harmony and singing Beijing opera



24

Dalle3

A minimap diorama of a cafe adorned with indoor plants. Wooden beams crisscross above, and a cold brew station stands out with tiny bottles and glasses



25

Progress in Inverse Problems

$P(\text{high resolution} \mid \text{low resolution})$



Menon et al, 2020

$P(\text{full image} \mid \text{mask})$



Liu et al, 2018

$P(\text{color image} \mid \text{greyscale})$

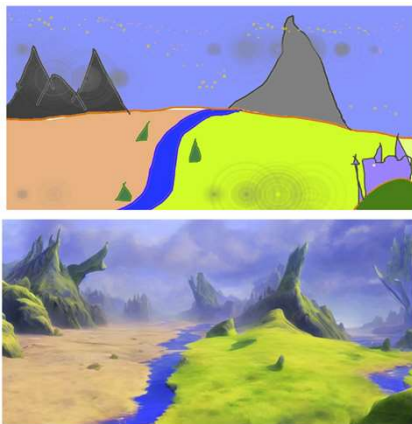


Antic, 2020

26

Progress in Inverse Problems

User input:



27

Progress in Inverse Problems

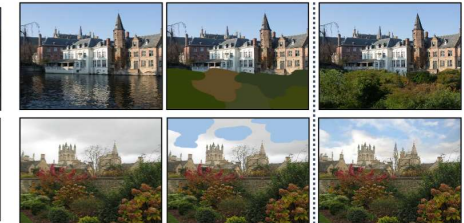
Stroke Painting to Image



Input

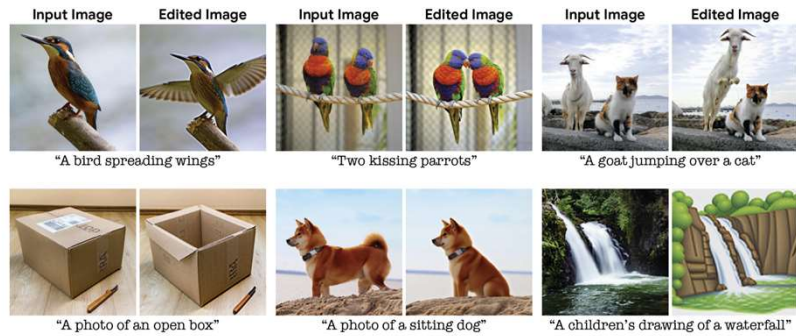
Output

Stroke-based Editing



28

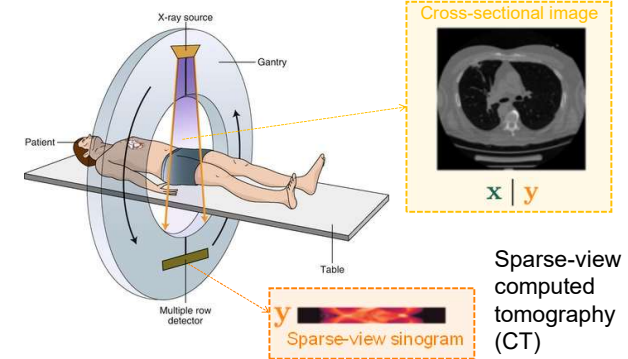
Progress in Inverse Problems



Kawar et al., 2023

29

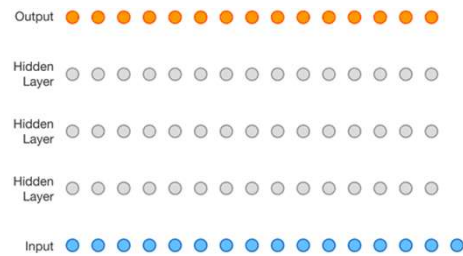
Medical image reconstruction

Forward model $p(y | x)$ is given by physical simulation

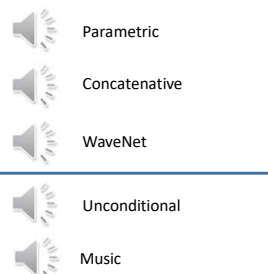
30

WaveNet

Generative model of speech signals



Text to Speech



van den Oord et al, 2016c

31

Diffusion Text2Speech

Generative model of speech signals



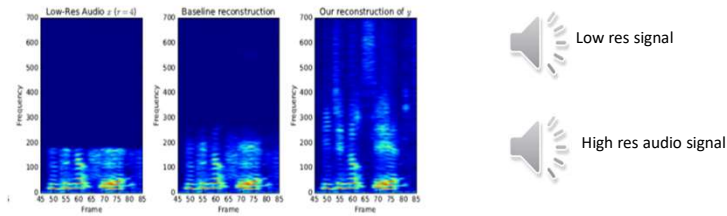
Betker, Better speech synthesis through scaling 2023

32

32

Audio Super Resolution

Conditional generative model $P(\text{high-res signal} \mid \text{low-res audio signal})$



Kuleshov et al., 2017

33

Language Generation

Completion

Custom prompt

To get an A+ in deep generative models, students have to

To get an A+ in deep generative models, students have to be willing to work with problems that are a whole lot more interesting than, say, the ones that most students work on in class. If you're a great student, the question above can be avoided and you'll be able to do great work, but if you're not, you will need to go beyond the basics before getting good.

Now to be clear, this advice is not just for the deep-learning crowd; it is good advice for any student who is taking his or her first course in machine learning.

The key point is that if you have a deep, deep brain of a computer scientist, that's just as important to you.

$P(\text{next word} \mid \text{previous words})$

Radford et al., 2019

Demo from talktotransformer.com

34

Language Generation -- ChatGPT

ER

What should I do to get an A+ in CS236 at Stanford?

To get an A+ in CS236 (Deep Generative Models) at Stanford, you will need to excel in both your understanding of the material and your performance in assignments and exams. Here are some general tips to help you achieve this:

1. **Attend Lectures and Engage Actively**: Attend all lectures and actively engage with the material. Take thorough notes, ask questions, and participate in discussions. This will help you understand the concepts better.
 2. **Read the Assigned Material**: Make sure to read the assigned textbooks, papers, and supplementary materials. Understanding the theoretical foundations is crucial.
 3. **Stay Organized**: Keep a well-organized notebook or digital notes. This will help you quickly review and understand the material.
 4. **Seek Help When Needed**: Don't hesitate to ask questions if you're having trouble with a concept. You can ask the professor, teaching assistants, or classmates for clarification.
 5. **Complete Assignments Thoroughly**: Take your time to complete assignments, ensuring you fully understand the requirements and concepts involved. Start early and seek help if you're stuck. High-quality assignments are often a significant portion of your grade.
 6. **Collaborate, but Don't Plagiarize**: Collaboration is often encouraged, but make sure you understand your institution's policy on collaboration and plagiarism. Always give credit where it's due and submit original work.
- ...
15. **Meet with the Professor**: If you're aiming for an A+, consider scheduling meetings with the professor to discuss your progress and seek feedback.

Remember that getting an A+ can be highly competitive, and the specific grading criteria may vary from one course to another and one professor to another. It's essential to understand the grading policies and expectations of your instructor. Always aim for excellence, but also keep in mind that learning and understanding the material should be your primary goal. Good luck!

35

Machine Translation

Conditional generative model $P(\text{English text} \mid \text{Chinese text})$

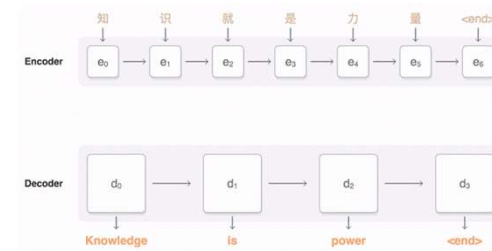


Figure from Google AI research blog.

36

36

Code Generation

```

1 import datetime
2
3 def parse_expenses(expenses_string):
4     """Parse the list of expenses and return the list of triples (date, va
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

```

OpenAI Codex

Video Generation

Suddenly, the walls of the embankment broke and there was a huge flood



37

38

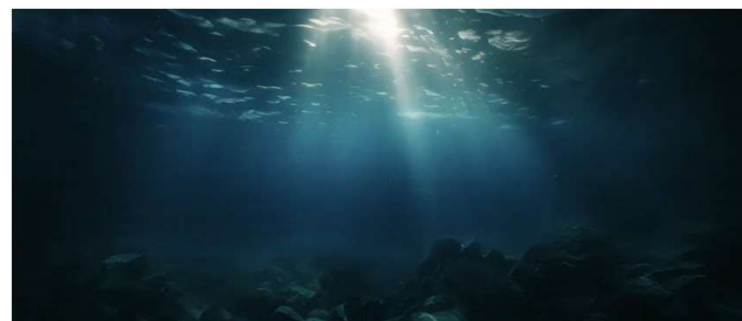
Video Generation

a couple sledding down a snowy hill on a tire roman chariot style



39

Video Generation



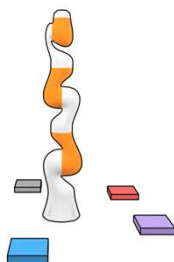
40

Imitation Learning

Conditional generative model $P(\text{actions} \mid \text{past observations})$



Li et al., 2017

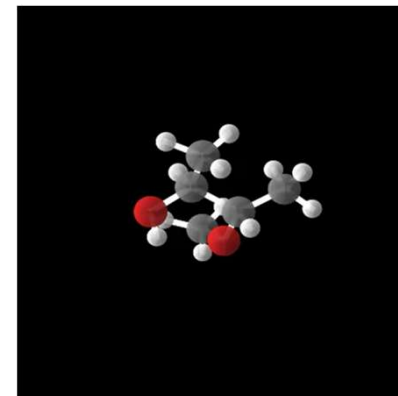


Janner et al., 2022

41

41

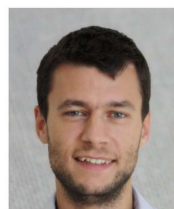
Molecule generation



42

DeepFakes

Which image is real?



User
@StefanoErmon

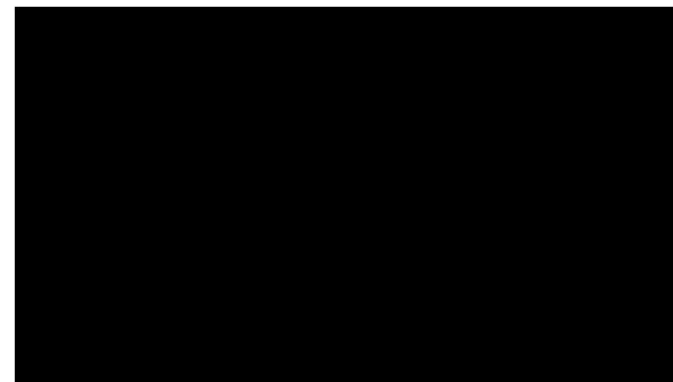
SDEdit



Output

43

DeepFakes



44

Roadmap and Key Challenges

- **Representation:** how do we model the joint distribution of many random variables?
 - Need compact representation
- **Learning:** what is the right way to compare probability distributions?



- **Inference:** how do we invert the generation process (e.g., vision as inverse graphics)?
 - Unsupervised learning: recover high-level descriptions (features) from raw data

45

Syllabus

- Fully observed likelihood-based models
 - Autoregressive
 - Flow-based models
- Latent variable models
 - Variational learning
 - Inference amortization
 - Variational autoencoder
- Implicit generative models
 - Two sample tests, embeddings, F-divergences
 - Generative Adversarial Networks
- Energy Based Models
- Score-based Diffusion Generative Models
- Learn about algorithms, theory & applications

46

Prerequisites

- Basic knowledge about machine learning from at least one of CS 221, 228, 229, or 230.
- Basic knowledge of probabilities and calculus:
 - Gradients, gradient-descent optimization, backpropagation
 - Random variables, independence, conditional independence
 - Bayes rule, chain rule, change of variables formulas
- Proficiency in some programming language, preferably Python, required.

47

Logistics

- Class webpage: <https://deepgenerativemodels.github.io/>
- There is no required textbook. Reading materials and course notes will be provided.
- Suggested Reading: *Deep Learning* by Ian Goodfellow, Yoshua Bengio, Aaron Courville. Online version available free [here](#).
- Lecture notes: <https://deepgenerativemodels.github.io/notes/index.html>
- Teaching Assistants:
 - Agarwal, Pratyush
 - Chatterjee, Sourmya
 - Chen, Honglin
 - Chiang, Bryan
 - Obbad, Elyas
 - Salahi, Kamyar
 - Shih, Boyun (Andy)
 - Xiao, Zedean (Mark)
 - Xu, Minkai
 - Yuan, Lu (Sylvia)
 - Zhou, Lin Qi (Alex)
- Coordinator: John Cho
- Office hours: See calendar on class website

48

Logistics – Grading policies

- Grading Policy
 - Three homeworks (15% each): mix of conceptual and programming based questions
 - Midterm: 15%
 - Course Project: 40%
 - Proposal: 5%
 - Progress Report: 10%
 - Poster Presentation: 10%
 - Final Report: 15%

49

Projects

- Course projects will be done in groups of up to 3 students and can fall into one or more of the following categories:
 - Application of deep generative models on a novel task/dataset
 - Algorithmic improvements into the evaluation, learning and/or inference of deep generative models
 - Theoretical analysis of any aspect of existing deep generative models
- Teaching staff will suggest possible projects
- We will provide Google Cloud coupons
- See list of past projects at:
<https://deepgenerativemodels.github.io/2019>

50