Deep Learning Basics

Lecture 9: Generative Models Part 1

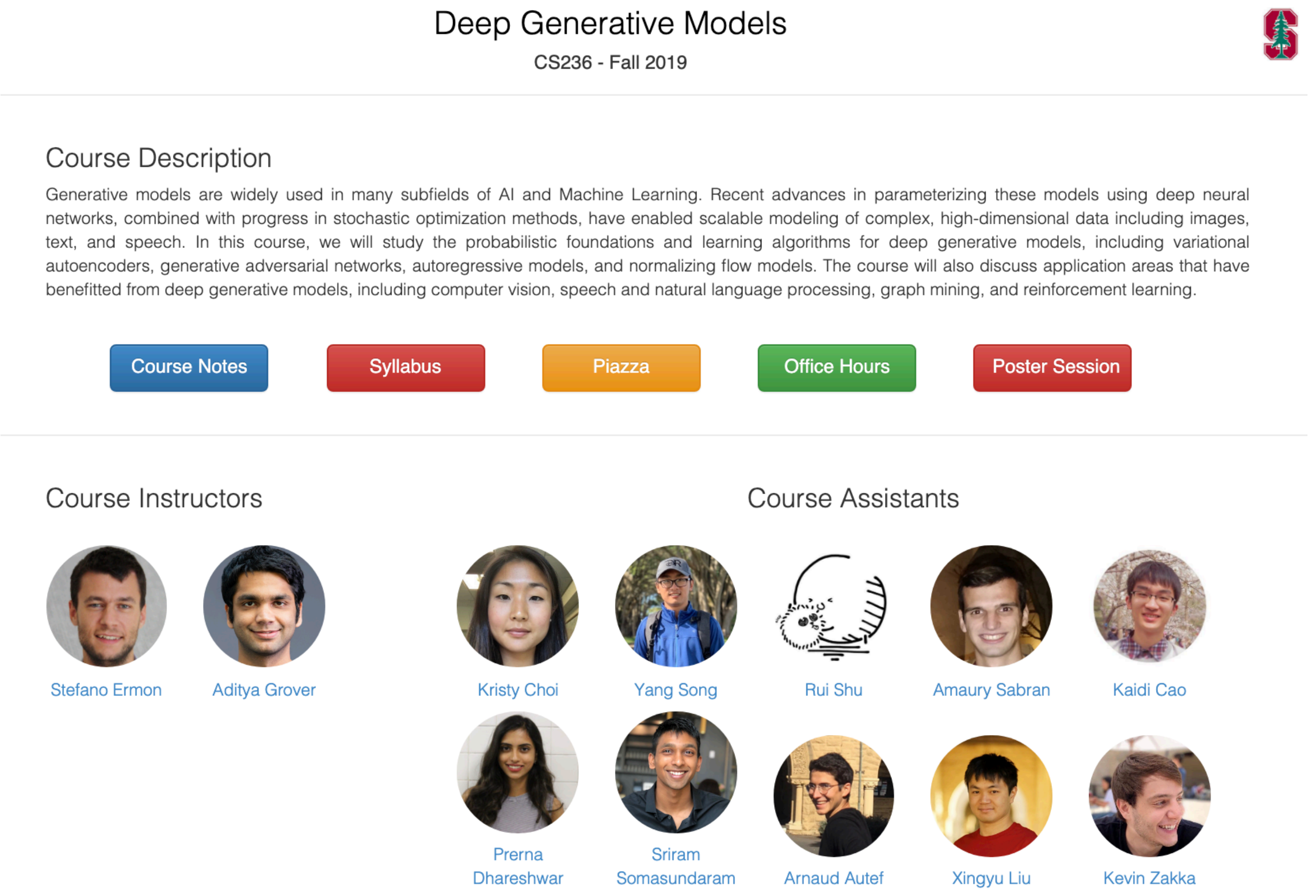
최성준 (고려대학교 인공지능학과)

boostcamp ai tech

Contents

Introduction

Introduction

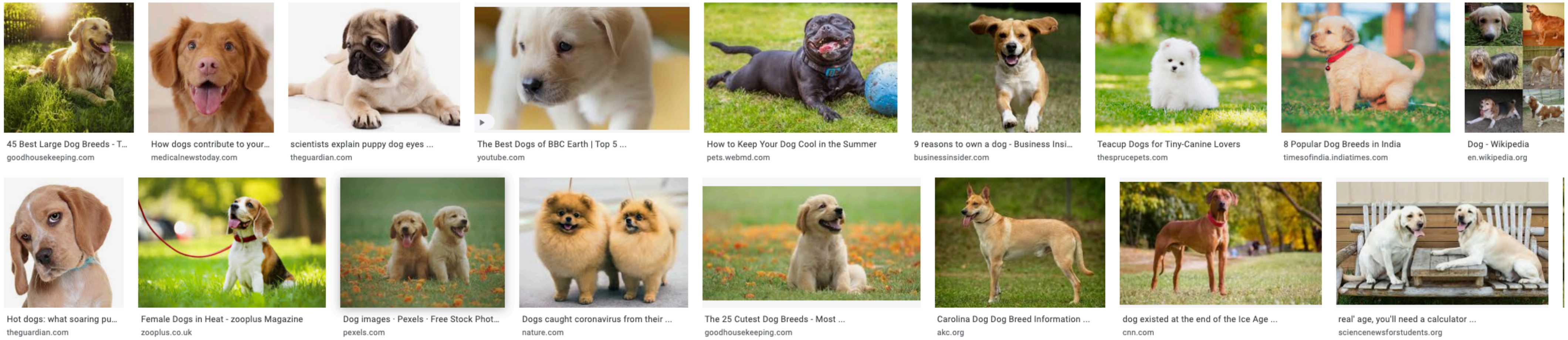


https://deepgenerativemodels.github.io/

Introduction

What does it mean by learning a **generative model**?

Learning a Generative Model



Google Search: Dog

- Suppose that we are given images of dogs.

Learning a Generative Model

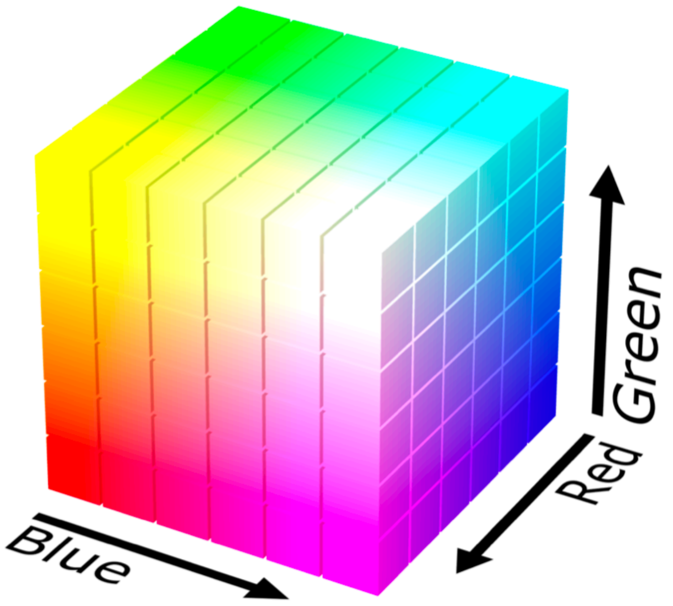
- Suppose that we are given images of dogs.
- We want to learn a probability distribution $p(x)$ such that
  - Generation: If we sample $\tilde{x} \sim p(x)$, $\tilde{x}$ should look like a dog.
  - Density estimation: $p(x)$ should be high if x looks like a dog, and low otherwise.
    - This is also known as explicit models.
- Then, how can we represent $p(x)$?

Basic Discrete Distributions

- Bernoulli distribution: (biased) coin flip
  - $D = \{\text{Heads}, \text{Tails}\}$
  - Specify $P(X = \text{Heads}) = p$. Then $P(X = \text{Tails}) = 1 - p$.
  - Write: $X \sim \text{Ber}(p)$
- Categorical distribution: (biased) m-sided dice
  - $D = \{1, \cdots, m\}$
  - Specify $P(Y = i) = p_i$ such that $\sum_{i=1}^{m} p_i = 1$
  - Write: $Y \sim \text{Cat}(p_1, \cdots, p_m)$

Example

- Modeling a single pixel of an RGB image
  - $(r, g, b) \sim p(R, G, B)$
  - Number of cases?
  - How many parameters do we need to specify?



https://en.wikipedia.org/wiki/RGB_color_space

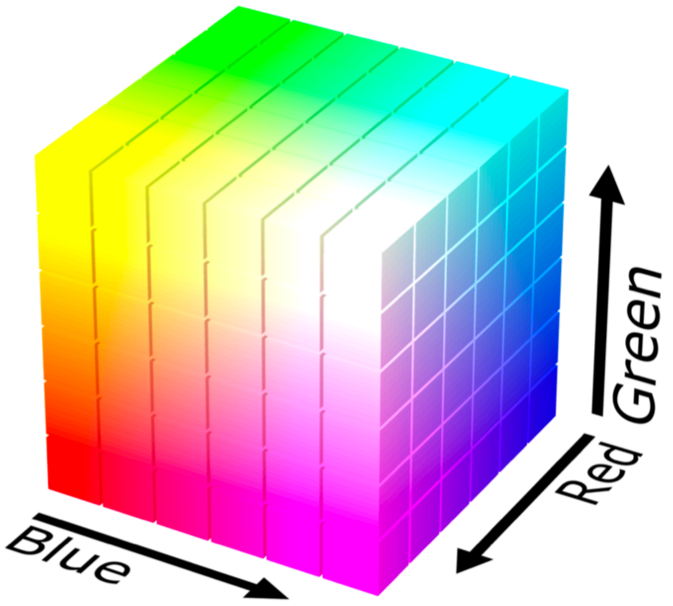
Example

- Modeling a single pixel of an RGB image
  - $(r, g, b) \sim p(R, G, B)$
  - Number of cases?

$$256 \times 256 \times 256$$

  - How many parameters do we need to specify?

$$256 \times 256 \times 256 - 1$$



https://en.wikipedia.org/wiki/RGB_color_space

Independence

Example

- Suppose we have $X_1, \ldots, X_n$ of n binary pixels (a binary image)
  - Number of cases?
  - How many parameters do we need to specify?

Example

- Suppose we have $X_1, \ldots, X_n$ of n binary pixels (a binary image)
  - Number of cases?

$$2 \times 2 \times \cdots \times 2 = 2^n$$

  - How many parameters do we need to specify?

$$2^n - 1$$

Structure Through Independence

- What if $X_1, \ldots, X_n$ are independent, then

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2)\cdots P(X_n)$$

  - Number of cases?

  - How many parameters do we need to specify?

Structure Through Independence

- What if $X_1, \ldots, X_n$ are independent, then

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2)\cdots P(X_n)$$

  - Number of cases?

$$2 \times 2 \times \cdots \times 2 = 2^n$$

  - How many parameters do we need to specify?

$$n$$

  - 2^n entries can be described by just n numbers. What does it mean?

Conditional Independence

- Three important rules
  - Chain rule

$$p(x_1, \ldots, x_n) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_1, x_2) \cdots p(x_n \mid x_1, \cdots, x_{n-1})$$

  - Bayes' rule

$$p(x \mid y) = \frac{p(x, y)}{p(y)} = \frac{p(y \mid x)p(x)}{p(y)}$$

  - Conditional independence

If $x \perp y \mid z$, then $p(x \mid y, z) = p(x \mid z)$

Conditional Independence

- Using the chain rule,

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1, X_2) \cdots P(X_n \mid X_1, \cdots, X_{n-1})$$

- How many parameters?

Conditional Independence

- Using the chain rule,

$$P(X_1, \ldots, X_n) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_1, X_2)\cdots P(X_n \mid X_1, \cdots, X_{n-1})$$

- How many parameters?
  - $P(X_1)$: 1 parameter
  - $P(X_2 \mid X_1)$: 2 parameters (one per $P(X_2 \mid X_1 = 0)$ and $P(X_2 \mid X_1 = 1)$)
  - $P(X_3 \mid X_1, X_2)$: 4 parameters
  - Hence, the total number becomes $1 + 2 + 2^2 + \cdots + 2^{n-1} = 2^n - 1$ which is the same as before. Why?

Conditional Independence

- Now, suppose $X_{i+1} \perp X_1, \ldots, X_{i-1} \mid X_i$ (Markov assumption), then

$$p(x_1, \ldots, x_n) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_2) \cdots p(x_n \mid x_{n-1})$$

- How many parameters?

Conditional Independence

- Now, suppose $X_{i+1} \perp X_1, \ldots, X_{i-1} \mid X_i$ (Markov assumption), then

$$p(x_1, \ldots, x_n) = p(x_1)p(x_2 \mid x_1)p(x_3 \mid x_2)\cdots p(x_n \mid x_{n-1})$$

- How many parameters?

$$2n - 1$$

  - Hence, by leveraging the Markov assumption, we get exponential reduction on the number of parameters.
  - Autoregressive models leverages this conditional independency.

Autoregressive Models

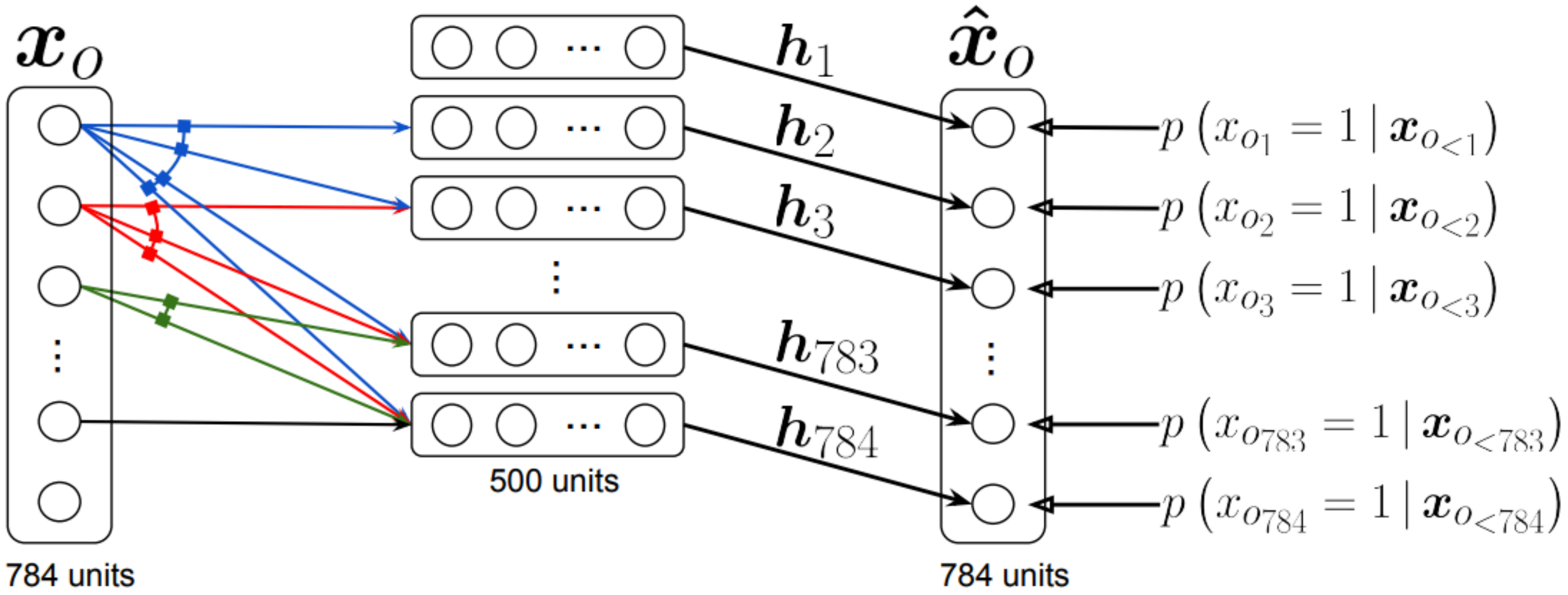
Autoregressive Model

- Suppose we have 28×28 binary pixels.
- Our goal is to learn $P(X) = P(X_1, \ldots, X_{784})$ over $X \in \{0,1\}^{784}$.
- Then, how can we parametrize $P(X)$?

Autoregressive Model

- Suppose we have 28×28 binary pixels.
- Our goal is to learn $P(X) = P(X_1, \ldots, X_{784})$ over $X \in \{0,1\}^{784}$.
- Then, how can we parametrize $P(X)$?
  - Let's use the chain rule to factor the joint distribution.
  - In other words,
    - $P(X_{1:784}) = P(X_1)P(X_2 \mid X_1)P(X_3 \mid X_2)\cdots$
    - This is called an autoregressive model.
    - Note that we need an ordering (e.g., raster scan order) of all random variables.

NADE: Neural Autoregressive Density Estimator



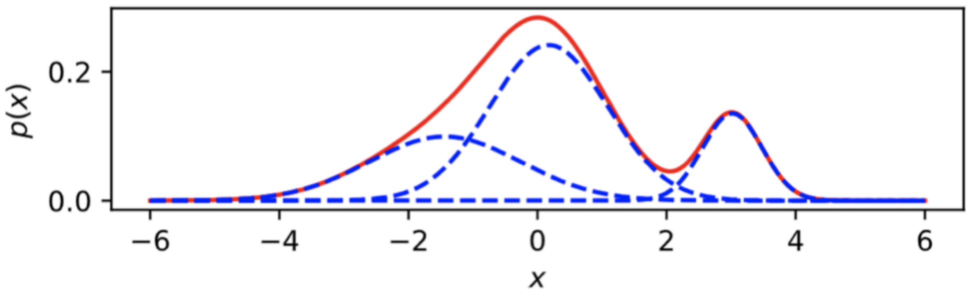
- The probability distribution of i-th pixel is

$$p(x_i \mid x_{1:i-1}) = \sigma(\alpha_i \mathbf{h}_i + b_i) \text{ where } \mathbf{h}_i = \sigma(W_{<i} x_{1:i-1} + \mathbf{c})$$

https://arxiv.org/pdf/1605.02226.pdf

NADE: Neural Autoregressive Density Estimator

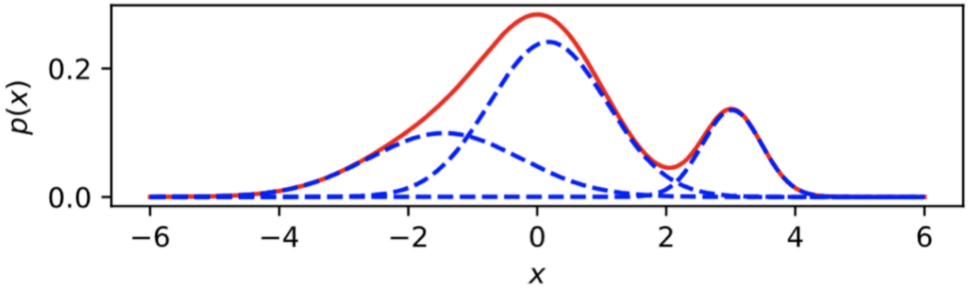
- **NADE** is an **explicit** model that can compute the **density** of the given inputs.
- BTW, how can we compute the **density** of the given image?
  - Suppose that we have a binary image with 784 binary pixels (i.e., $\{x_1, x_2, \ldots, x_{784}\}$).



https://arxiv.org/pdf/1605.02226.pdf

NADE: Neural Autoregressive Density Estimator

- **NADE** is an **explicit** model that can compute the **density** of the given inputs.
- BTW, how can we compute the **density** of the given image?
  - Suppose that we have a binary image with 784 binary pixels (i.e., $\{x_1, x_2, \ldots, x_{784}\}$).
  - Then, the joint probability is computed by
    - $p(x_1, \ldots, x_{784}) = p(x_1)p(x_2 | x_1)\cdots p(x_{784} | x_{1:783})$ where each conditional probability $p(x_i | x_{1:i-1})$ is computed independently.
- In case of modeling continuous random variables, a mixture of Gaussian (MoG) can be used.



https://arxiv.org/pdf/1605.02226.pdf

Summary of Autoregressive Models

- Easy to **sample** from
  - Sample $\bar{x}_0 \sim p(x_0)$
  - Sample $\bar{x}_1 \sim p(x_1 | x_0 = \bar{x}_0)$
  - $\cdots$ and so forth (in a sequential manner, hence slow)
- Easy to **compute probability** $p(x = \bar{x})$
  - Compute $p(x_0 = \bar{x}_0)$
  - Compute $p(x_1 = \bar{x}_1 | x_0 = \bar{x}_0)$
  - Multiply together (sum their logarithms)
  - $\cdots$ and so forth
  - Ideally, we can compute all these terms in parallel.
- Easy to be extended to **continuous** variables. For example, we can chose mixture of Gaussians.

Thank you for listening