

# Week15: Generative Models

예습	
복습	
복습과제 날짜	
예습과제 날짜	
내용	

## Supervised Learning, Unsupervised Learning

- Supervised learning : 지도학습
  - 학습 데이터의 label이 주어진 학습법이다.
  - classification, object detection, semantic segmentation, image captioning
- Unsupervised : 비지도 학습
  - 정답레이블이 주지않고, 데이터를 잘 포착해서 결과물을 도출하는 학습 방법.
  - 정답 라벨이 없어서 training data가 싸다.
  - goal: learn underlying hidden structure of the data
  - K-means clustering, principal component analysis, autoencoders, 2-d density estimation..

## Generative Models

Given training data, generate new samples from same distribution



Training data  $\sim p_{\text{data}}(x)$



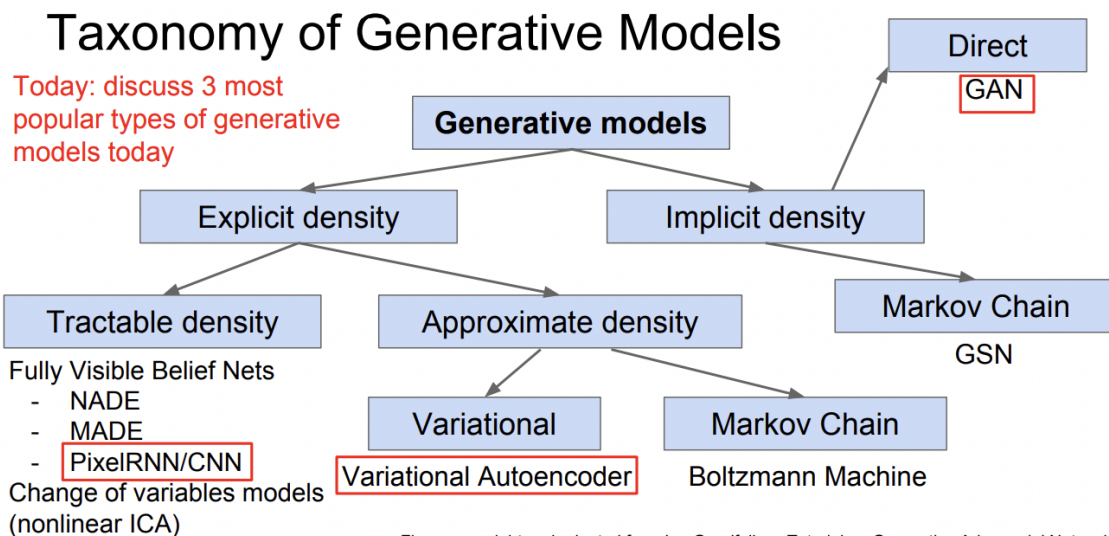
Generated samples  $\sim p_{\text{model}}(x)$

Want to learn  $p_{\text{model}}(x)$  similar to  $p_{\text{data}}(x)$

레퍼런스가 될 training data가 주어졌을 때, 그와 비슷한 분포를 가지는 새로운 데이터를 생성한다.

일종의 unsupervised model, density estimation이 중요

## Taxonomy of Generative Models



explicit density extim: 생성 모델을 명시적으로 나타재누다

implicit density extim.: 생성 모델을 정의하지 않고 sample을 얻는 방법

## PixelRNN and PixelCNN

explicit density estimation중 하나

## Fully visible belief network

explicit density :

$$p(x) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1})$$

Likelihood of  
image x

Probability of  $i$ 'th pixel value  
given all previous pixels

모델이 어떤 분포를 띄는지 정의하고 초점을 맞춘다.

Explicit density 모델은 training data의 likelihood를 높이는 방향으로 학습을한다.  $x_1, x_i$ 가 지가 각 pixel이 등장할 확률이라면, 해당 pixel들로 구성된 이미지가 나타날 확률은 각 pixel들의 확률곱이. loss로 학습 정도도 알 수 있다.

모델을 정의하는데 한계가 있어 생성모델은 Implicit density를 사용하는 GAN이 가장 잘 사용된다.

implicit density: 어떤 분포를 띄는지 관심이 없고, 단지 샘플을 생성할 수 있는 수준을 원함.

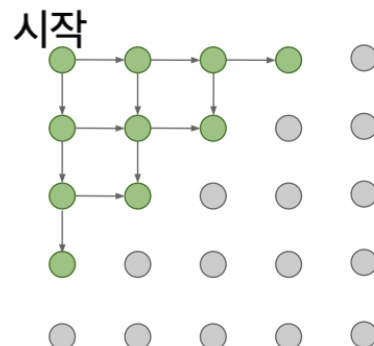
## Pixel RNN - tractable density

왼쪽 위 코너에서 시작해서 상하좌우로 뻗어나가면서 이미지를 픽셀바이픽셀로 생성

Generate image pixels starting from corner

Dependency on previous pixels modeled using an RNN (LSTM)

Drawback: sequential generation is slow!



인접 픽셀의 영향을 받아 생성 → 이전 결과에 영향을 받는 구조에 RNN이 적합해서 이전 픽셀들에 대한 dependency는 LSTM이나 RNN으로 표현된다.

매우 느림..

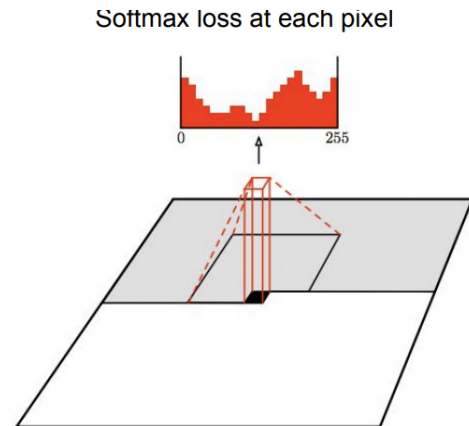
## Pixel CNN

Still generate image pixels starting from corner

Dependency on previous pixels now modeled using a CNN over context region

Training: maximize likelihood of training images

$$p(x) = \prod_{i=1}^n p(x_i | x_1, \dots, x_{i-1})$$



RNN을 CNN으로 대체

RNN과 같이 한 쪽 끝에서 시작하지만 이미지 생성에 영향을 주는 인접 좌표를 한번에 CNN 한다. RNN보다 빠르다.

## PixelRNN and PixelCNN

Pros:

- Can explicitly compute likelihood  $p(x)$
- Explicit likelihood of training data gives good evaluation metric
- Good samples

Con:

- Sequential generation => slow

Improving PixelCNN performance

- Gated convolutional layers
- Short-cut connections
- Discretized logistic loss
- Multi-scale
- Training tricks
- Etc...

See

- Van der Oord et al. NIPS 2016
- Salimans et al. 2017 (PixelCNN++)

# Variational Autoencoders(VAE)

VAEs define intractable density function with latent  $z$ :

$$p_{\theta}(x) = \int p_{\theta}(z)p_{\theta}(x|z)dz$$

VAE같은 경우 확률 모델이 계산 불가능한 함수로 정의된다.

따라서 하한선을 구해 계산 가능한 형태로 변경한다.

## Autoencoders

deep nn을 이용하여 데이터의 추상화를 위해 사용된다.

기존의 차원축소 문제들은 Unsupervised Learning이었는데 이를 encoder와 decoder로 이루어져있는 NN를 이용하여 supervised Learning문제로 바꾸어서 해결하였다.

데이터를 압축해서 다시 복원을 해주는데 training data에 대해 복원을 잘 해주는 방식으로 차원 축소를 하기 위해 사용하는 방식이 AutoEncoder이다.

### encoder

input이 들어왔을 때 특징벡터를 추출하는 과정

sigmoid, full Connected layer, ReLU, CNN

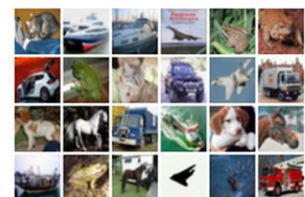
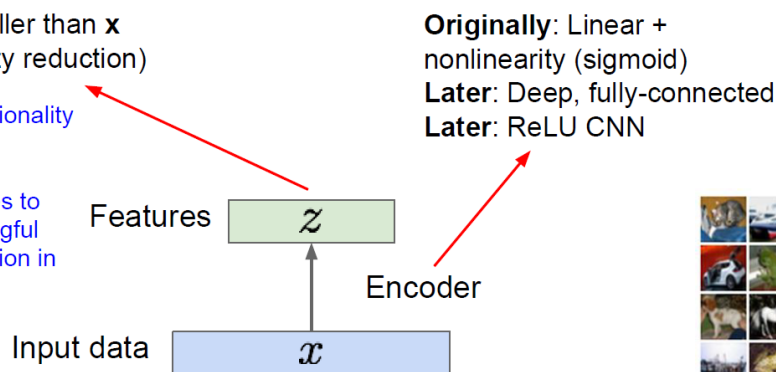
## Some background first: Autoencoders

Unsupervised approach for learning a lower-dimensional feature representation from unlabeled training data

$z$  usually smaller than  $x$   
(dimensionality reduction)

Q: Why dimensionality reduction?

A: Want features to capture meaningful factors of variation in data



특징벡터  $z$ 는 인풋보다 작다. 보통 차원이 줄어든 형태.. ← 노이즈를 없앤다.

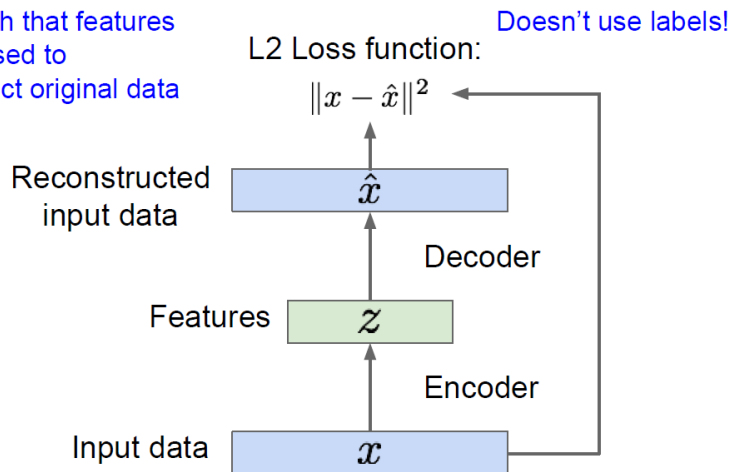
## decoder

특징 벡터를 통해 input data처럼 재구성하는 과정.

sigmoid, full Connected layer, ReLU, CNN

## Some background first: Autoencoders

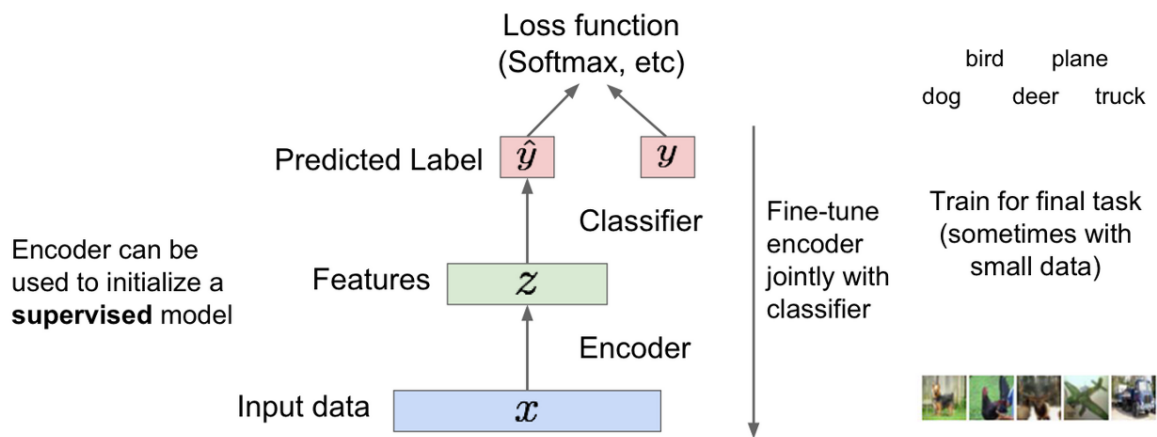
Train such that features  
can be used to  
reconstruct original data



복원된 이미지와 원본 이미지의 차이를 계산하기 위해 L2 loss사용한다. 트레이닝시에만 decoder을 사용하고 실제 테스트에는 인코더만 사용한다.

- AE에서는 Loss Function을 사용합니다. 하지만 이때 Label이 필요없다.
- 즉 레이블 없는 학습 데이터로만 학습

## Some background first: Autoencoders



## Variational Autoencoders

Variational AutoEncoder는 Generative Model이다.

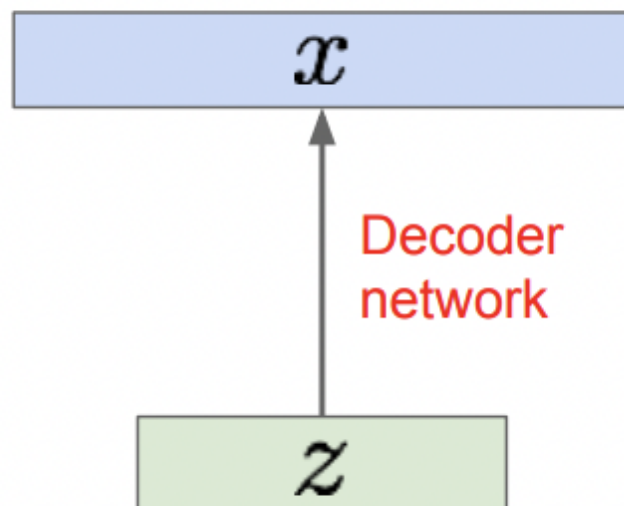
AE가 잘 추출한 특징을 이용해서 이미지 클래스를 분류했다면, VAE는 특징을 이용하여 새로운 이미지 생성

Sample from  
true conditional

$$p_{\theta^*}(x \mid z^{(i)})$$

Sample from  
true prior

$$p_{\theta^*}(z)$$



$z$  : latent vector



- $p_{\theta^*}(z)$ : parameter가  $\theta$ 일때, latent vector  $z$ 를 sampling 할 수 있는 확률밀도함수
- $p_{\theta}(x|z)$ : parameter가  $\theta$ 이면서,  $z$ 가 주어졌을 때  $x$ 를 생성해내는 확률밀도함수

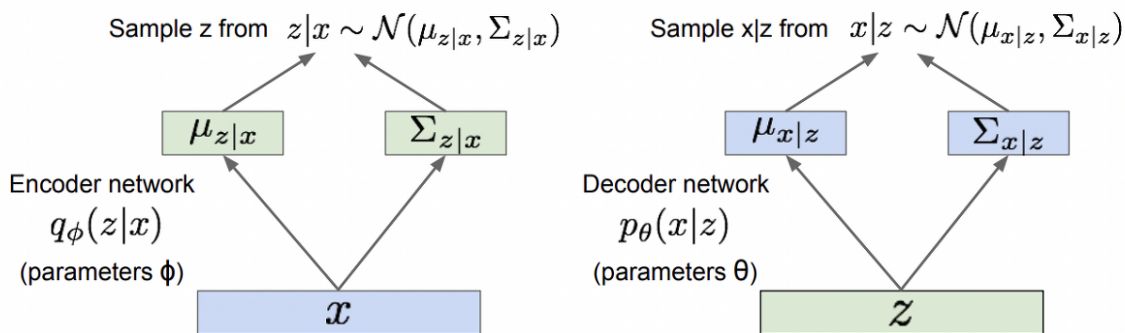
$$p_{\theta}(x) = \int p_{\theta}(z)p_{\theta}(x|z)dz$$

이때 세타를 실제 분포와 가깝게 찾는 것이 목표.

$p_{\theta}(x)$ : parameter가  $\theta$  일때  $x$ 가 나올 likelihood를 최대화 시키는 방향으로 학습

$x|z$ 에서 모든  $z$ 에 대해 적분을 취해줄 수 없음  $\rightarrow$  VAE 사용

디코더에 인코더 네트워크 추가



Decoder&encoder 구조로 완성된 VAE 구조

## Variational Autoencoders

