Generative Model

Changhoon Jeong
DGU AI lab.
chjeong@dongguk.edu

Agenda

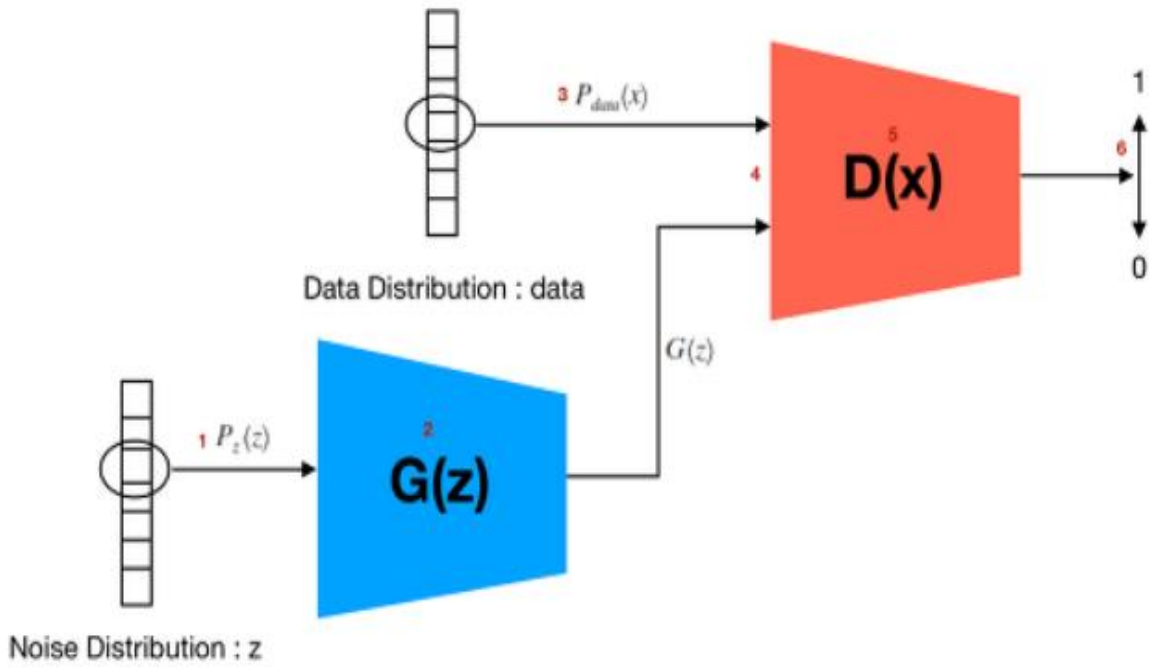
I. GAN(Generative Adversarial Nets) Review

II. VAE(Variational Autoencoder)

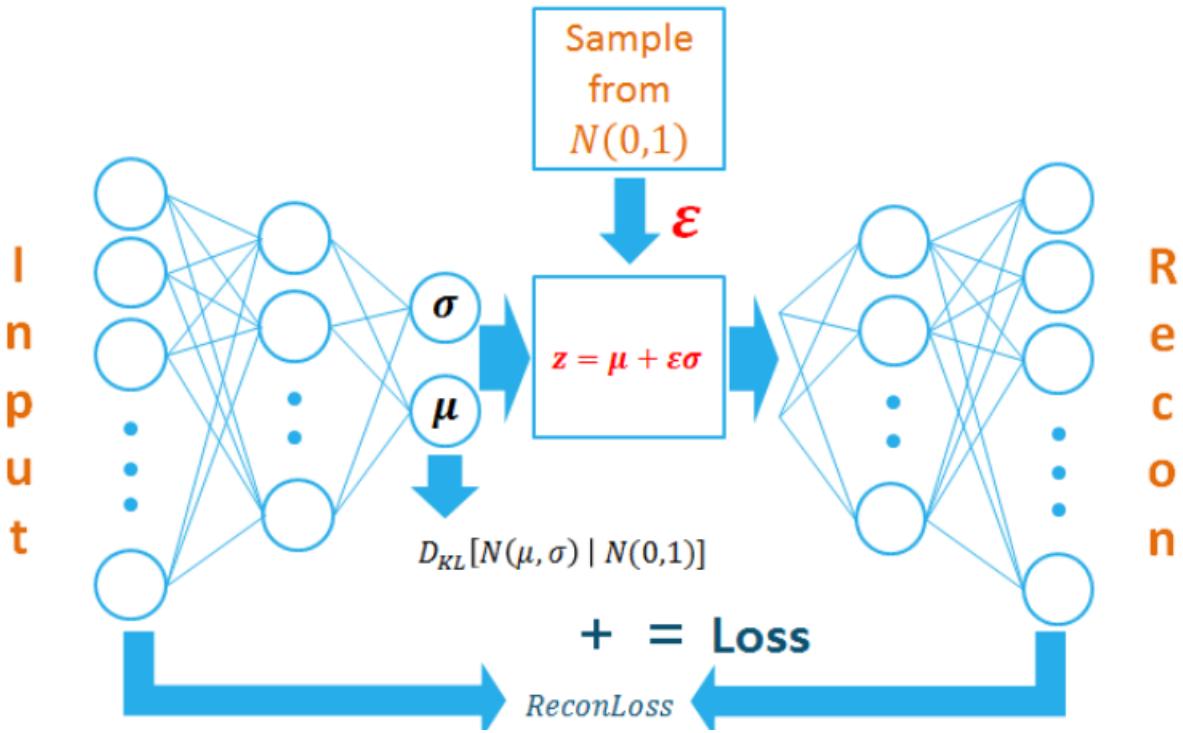
III. Discussion : GAN vs VAE

IV. Recent Trends in Deep Learning

- Generative Model



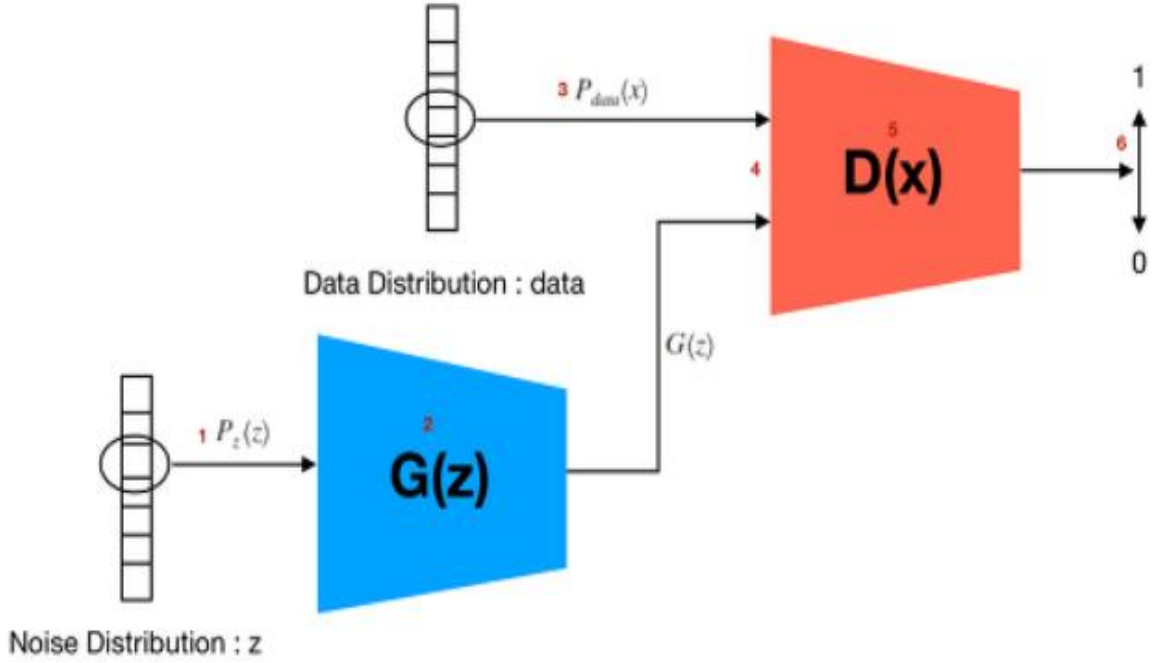
Generative Adversarial Nets



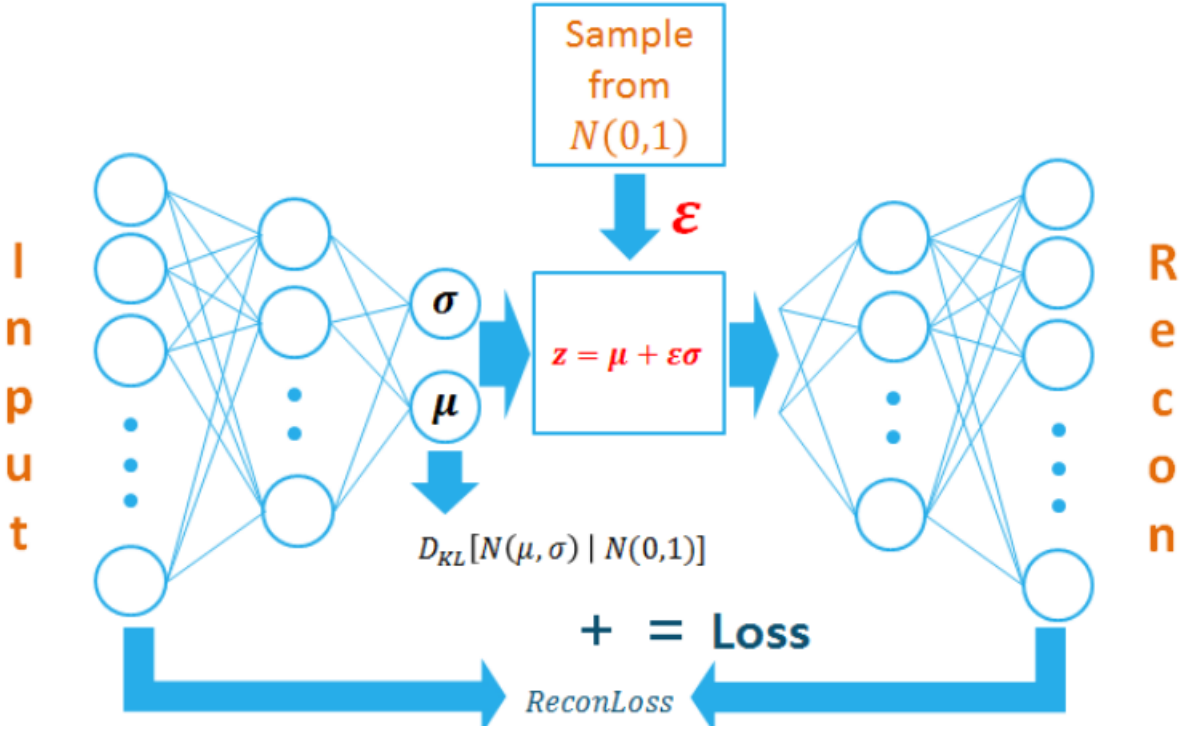
Variational Autoencoder

- Generative Model

1. GAN : A (short) review



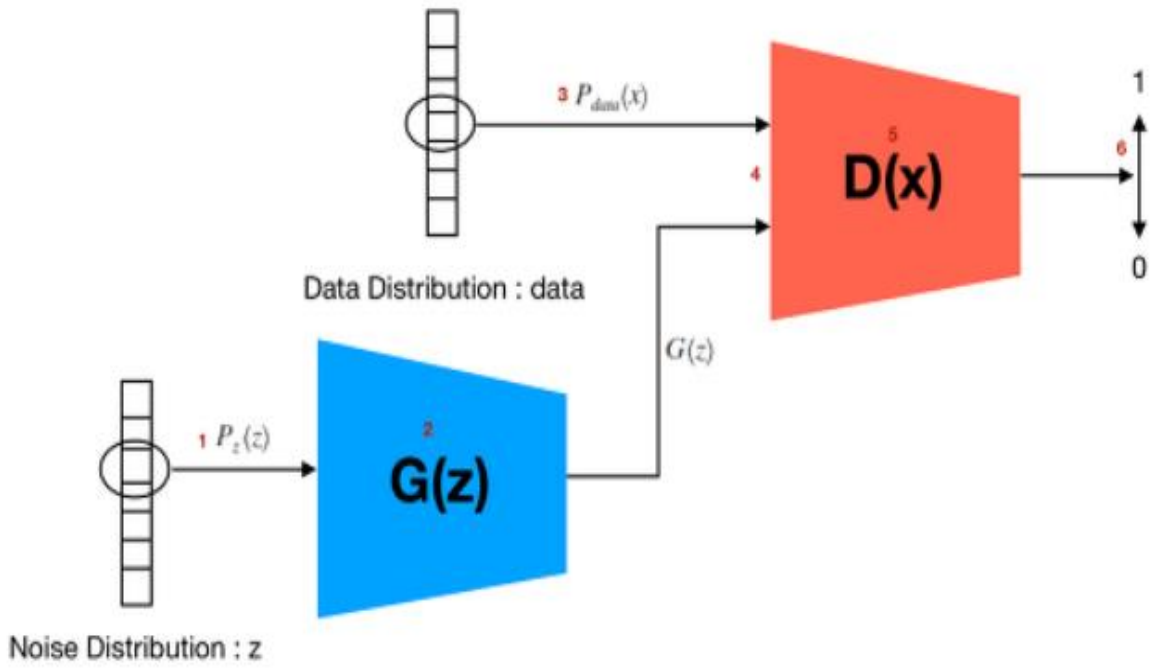
Generative Adversarial Nets



Variational Autoencoder

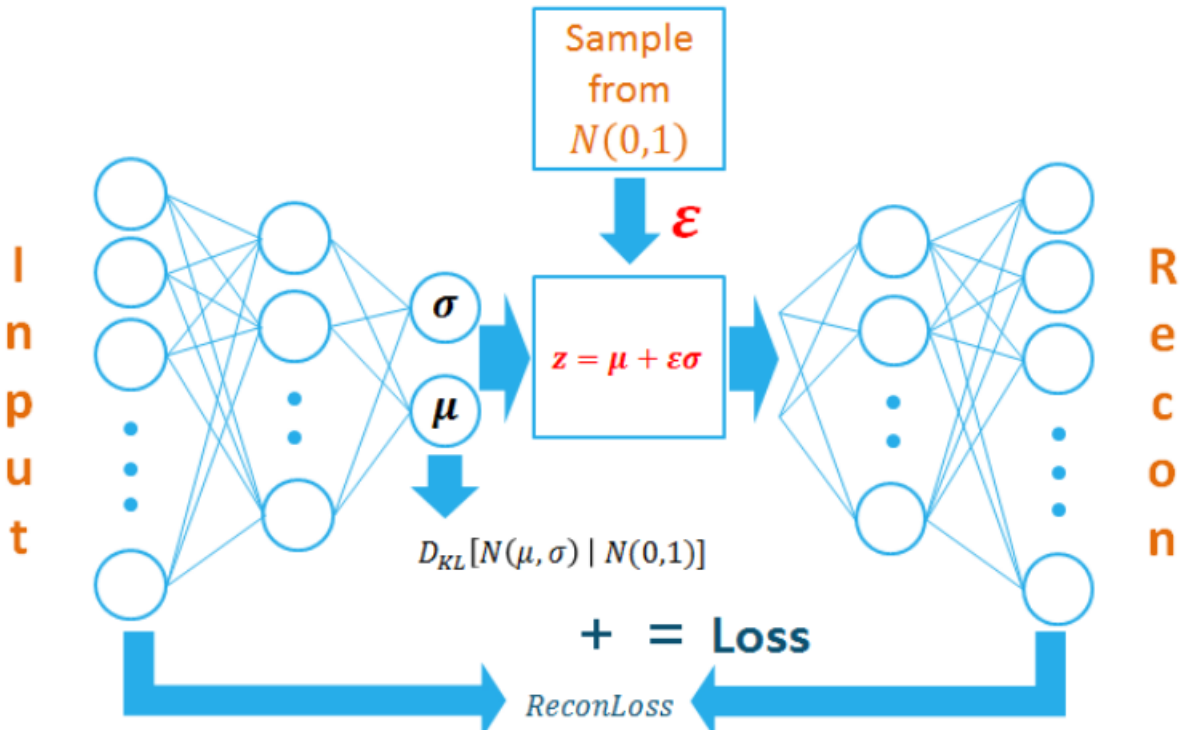
- Generative Model

1. GAN : A (short) review



Generative Adversarial Nets

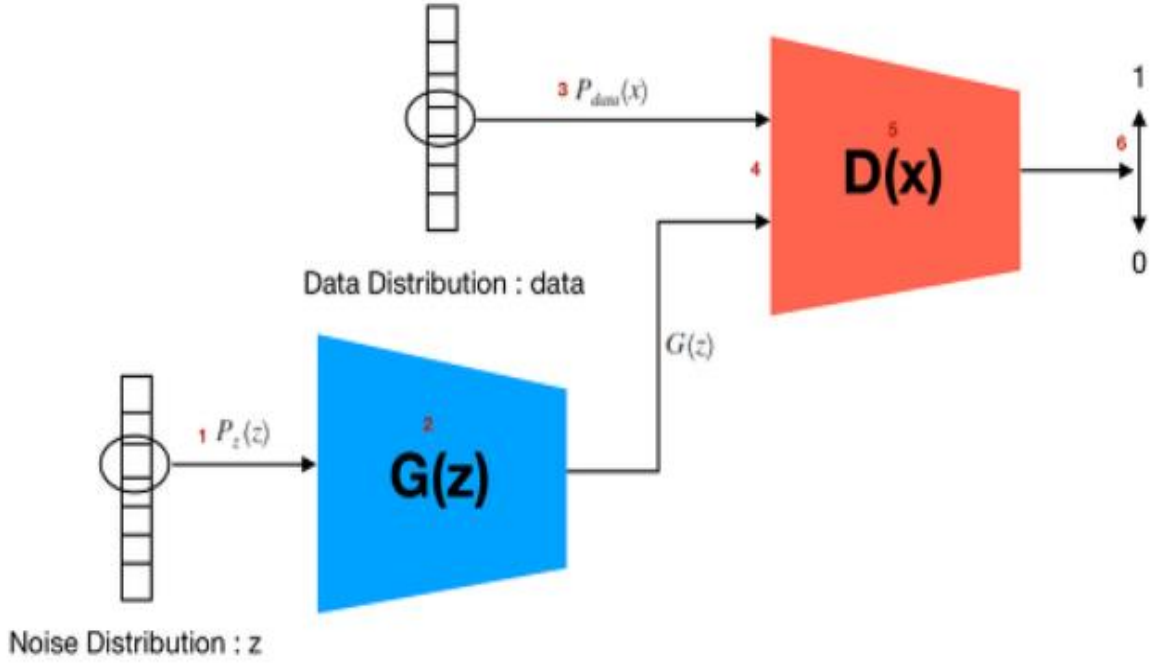
2. "Variational Inference" for VAE



Variational Autoencoder

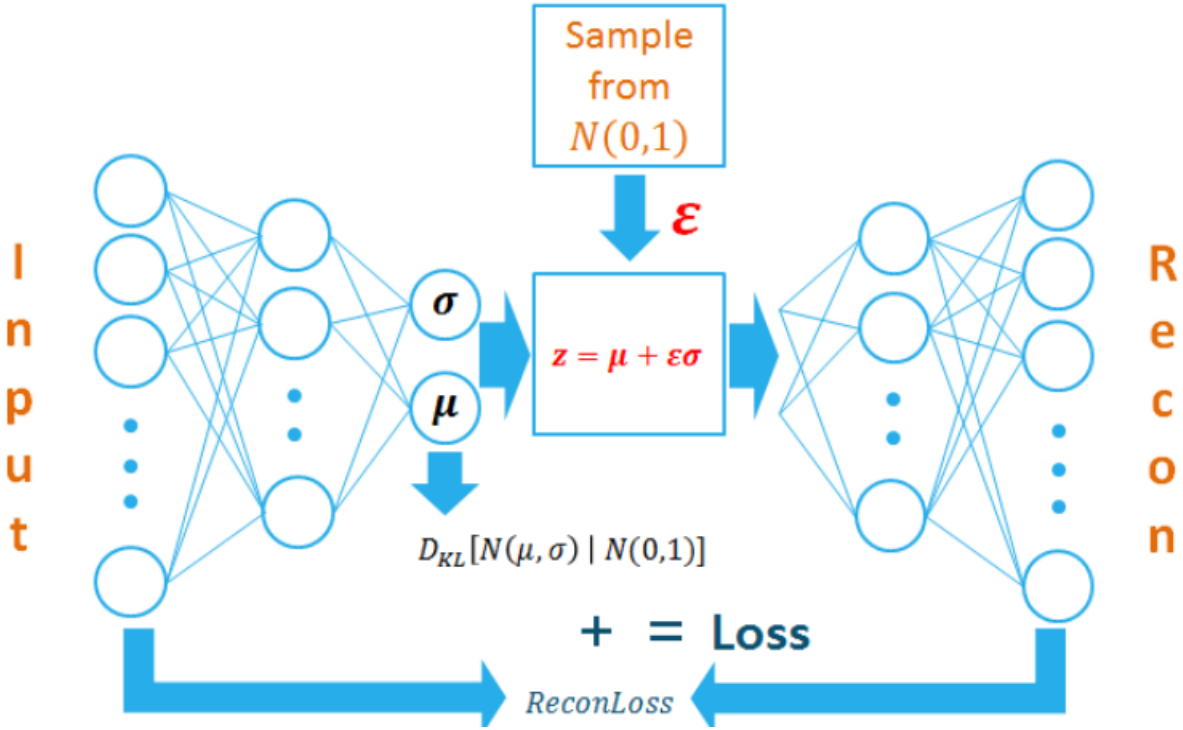
- Generative Model

1. GAN : A (short) review



Generative Adversarial Nets

2. "Variational Inference" for VAE



Variational Autoencoder

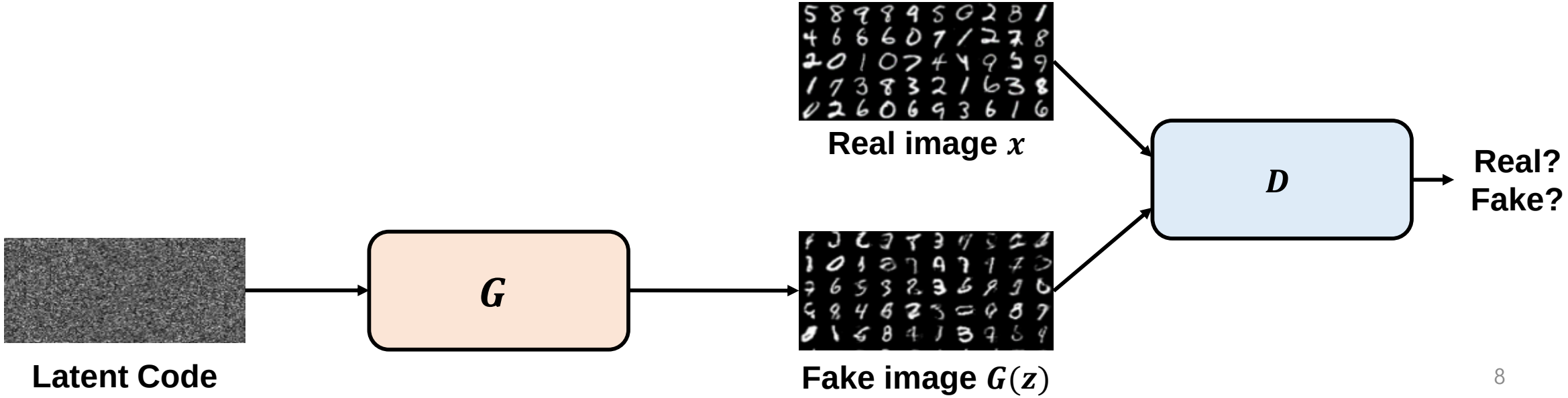
3. GAN vs VAE, what's difference?

I. GAN(Generative Adversarial Nets) Review

GAN(Generative Adversarial Nets) Review

- Objective function of GAN : Discriminator

$$\min_{G} \max_{D} V(D,G) = \mathbb{E}_{x \sim p_{data(x)}}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$



GAN(Generative Adversarial Nets) Review

- Objective function of GAN : Discriminator

Sample x from real data distribution

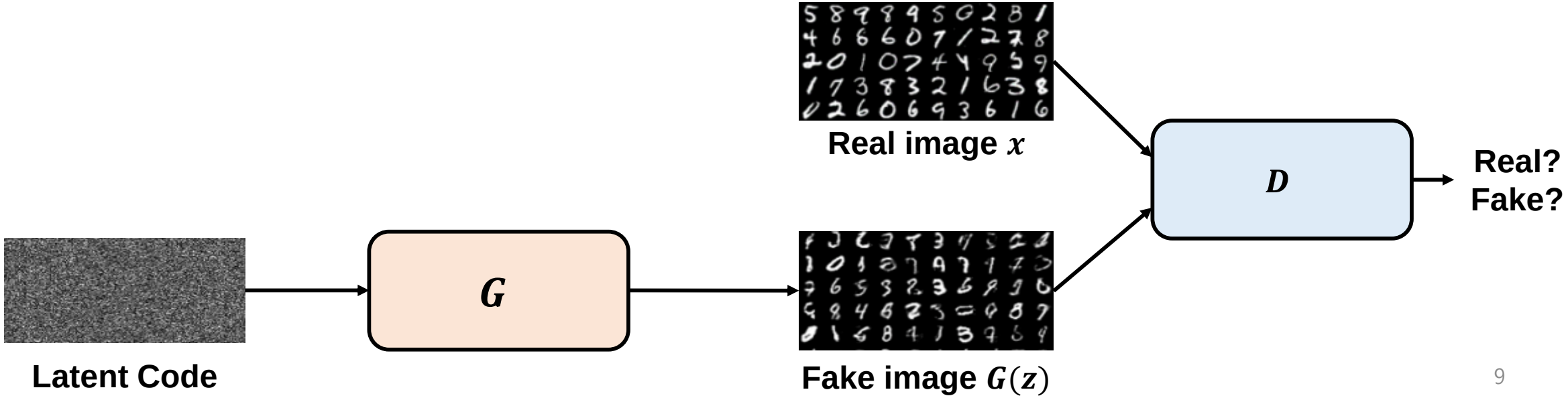
Sample latent code z from Gaussian distribution

$$\min_{G} \max_{D} V(D,G) = \mathbb{E}_{x\sim p_{data(x)}}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

D should maximize $V(D,G)$

Maximum when $D(x) = 1$

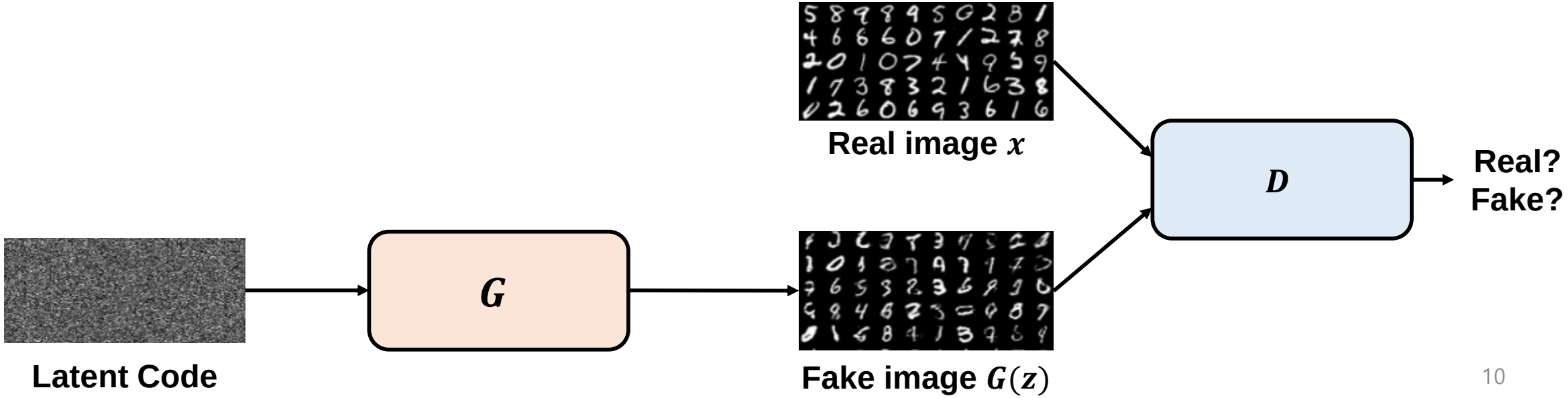
Maximum when $D(G(z)) = 0$



GAN(Generative Adversarial Nets) Review

- Objective function of GAN : Generator

$$\min_{G} \max_{D} V(D,G) = \mathbb{E}_{x\sim p_{data(x)}}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$



GAN(Generative Adversarial Nets) Review

- Objective function of GAN : Generator

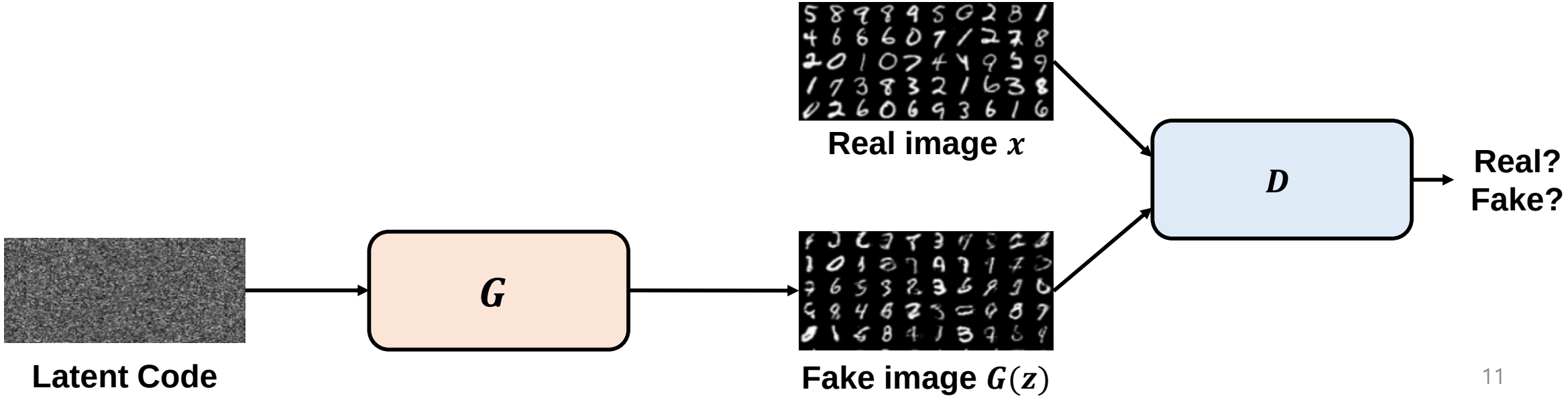
G is independent of this part

Sample latent code z from Gaussian distribution

$$\min_{G}\max_{D} V(D,G) = \cancel{\mathbb{E}_{x\sim p_{data}(x)}[\log D(x)]} + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$

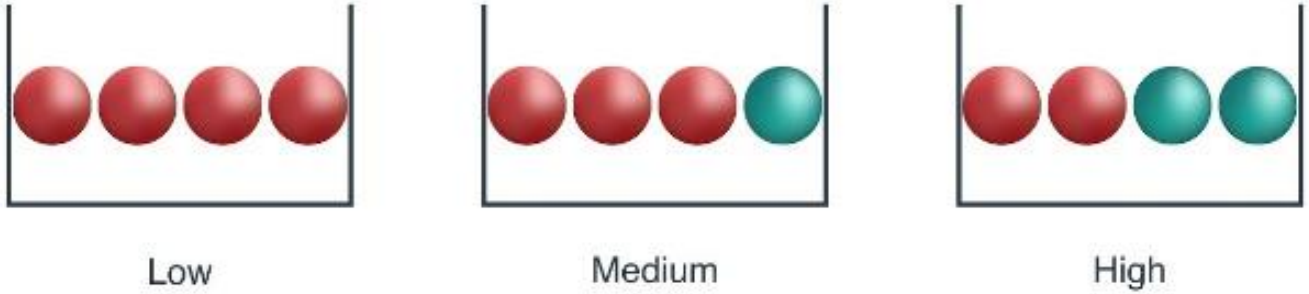
G should minimize $V(D,G)$

Minimum when $D(G(z)) = 1$



II. Variational Autoencoder

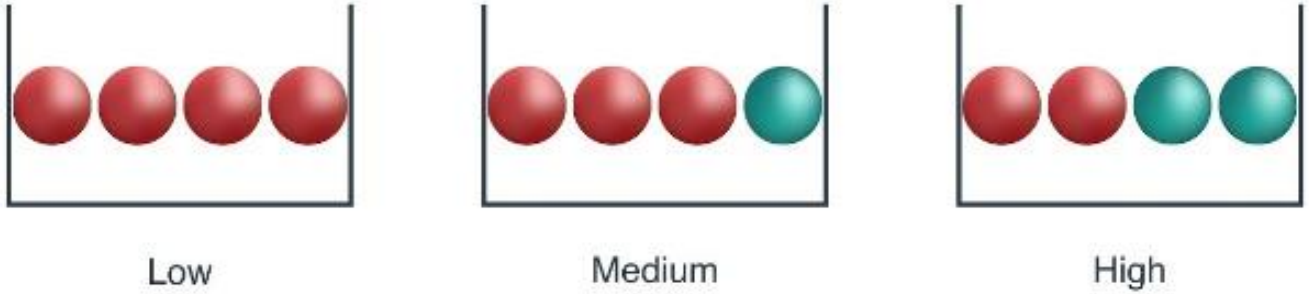
- Preliminaries
  - Information Theory : Entropy(average code length)



$$H(x) = -\sum_{x} p(x) \log_2 p(x) = \mathbb{E}_p[-\log_2 p(x)]$$

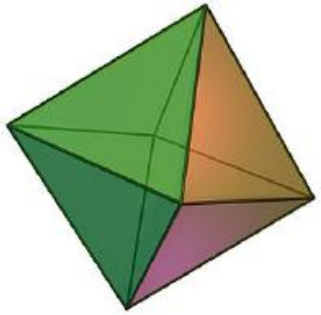
A mathematical theory of communication", Claude E. Shannon, 1948

- Preliminaries
  - Information Theory : Entropy(average code length)



$$H(x) = -\sum_{x} p(x)\log_2 p(x) = \mathbb{E}_p[-\log_2 p(x)]$$

  - For example,



$$\textit{The probability of the eight sides of the die is not uniform}\left(\frac{1}{2},\frac{1}{4},\frac{1}{8},\frac{1}{16},\frac{1}{64},\frac{1}{64},\frac{1}{64},\frac{1}{64}\right)$$

$$H(x) = -\sum_{x} p(x)\log_2 p(x) = -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{8}\log_2\frac{1}{8} - \frac{1}{16}\log_2\frac{1}{16} - \frac{4}{64}\log_2\frac{1}{64} = 2$$

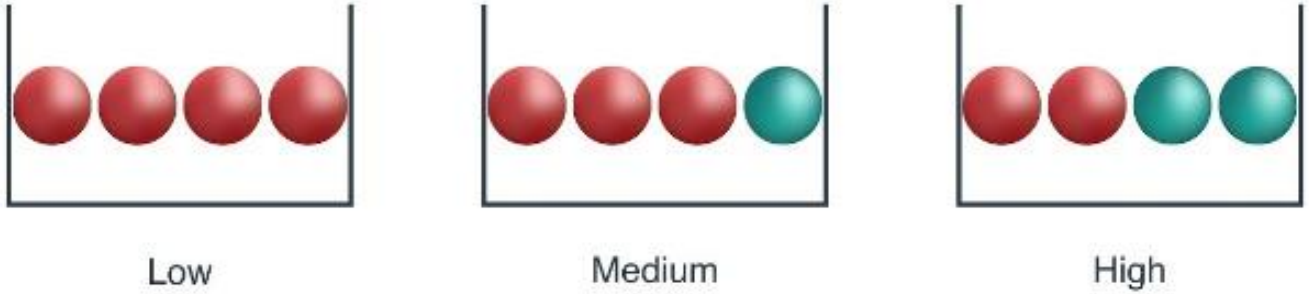
$$0, 01, 011, 1011, 111100, 101010, 111110, 111111(\text{coding})$$

$$Average\ Code\ Length = \frac{1}{2}\times 1 + \frac{1}{4}\times 2 + \frac{1}{8}\times 3 + \frac{1}{16}\times 4 + \frac{4}{64}\times 6 = 2bits$$

A mathematical theory of communication", Claude E. Shannon, 1948

Variational Autoencoder

- Preliminaries
  - Information Theory : Entropy(average code length)



$$H(x) = -\sum_{x} p(x) \log_2 p(x) = \mathbb{E}_p[-\log_2 p(x)]$$

  - In case of continuous variable,

$$H(x) = \lim_{\Delta \to 0} \left\{ \sum_{i} p(x_i) \Delta \ln p(x_i) \right\} = -\int p(x) \ln p(x)\, dx$$

A mathematical theory of communication", Claude E. Shannon, 1948

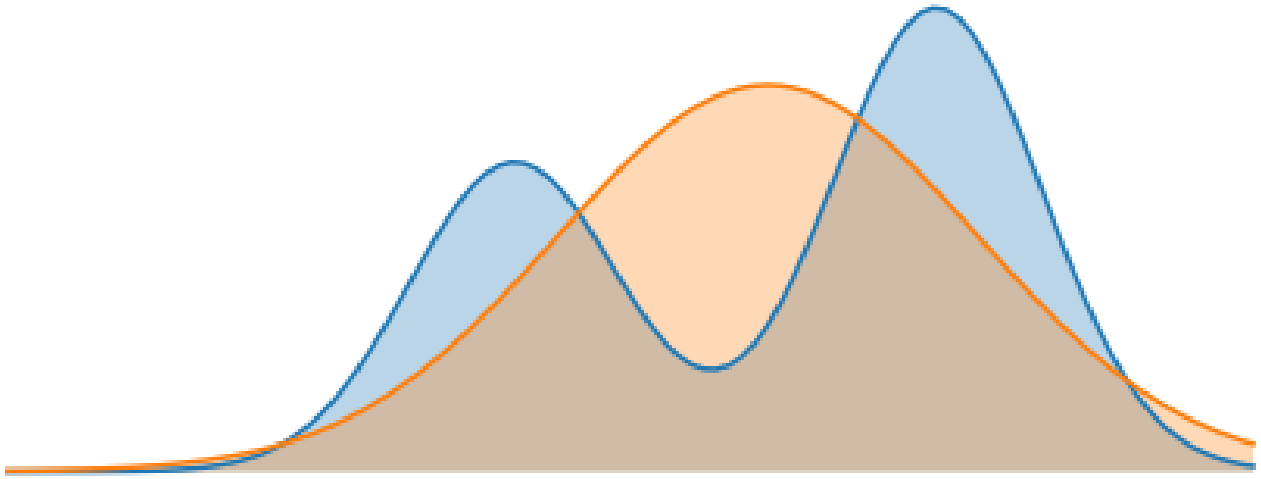
Variational Autoencoder

- Preliminaries
  - Kullback-Leibler Divergence
    - In case of discrete variable,

$$D_{KL}(p||q) = \sum_i p(x_i) \log_2 \frac{p(x_i)}{q(x_i)}$$

    - In case of continuous variable,

$$D_{KL}(p||q) = \int p(x) \ln \frac{p(x)}{q(x)} dx$$



- Preliminaries
  - Kullback-Leibler Divergence
    - My guess : $q(x)$

$$q(x) = \left(\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right)$$

$$My\ Coding : 0, 10, 110, 111$$

    - Actual probability distribution of 4 dices $p(x)$

$$p(x) = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right)$$

$$Actual\ Coding : 00, 01, 10, 11$$

- Preliminaries
  - Kullback-Leibler Divergence
    - My guess : $q(x)$

$$-\sum_x p(x)\log_2 q(x) = -\frac{1}{4}\log_2 0.5 - \frac{1}{4}\log_2 0.25 - \frac{1}{4}\log_2 0.125 - \frac{1}{4}\log_2 0.125 = 2.25$$

    - If I guess correctly : $p(x)$

$$-\sum_x p(x)\log_2 p(x) = -\frac{1}{4}\log_2 0.25 - \frac{1}{4}\log_2 0.25 - \frac{1}{4}\log_2 0.25 - \frac{1}{4}\log_2 0.25 = 2$$

    - KL Divergence?

$$D_{KL}(p||q) = \left(-\sum_x p(x)\log_2 q(x)\right) - \left(-\sum_x p(x)\log_2 p(x)\right) = -\sum_i p(x_i)\log_2 \frac{q(x_i)}{p(x_i)} = 2.25 - 2 = 0.25$$

- Preliminaries
  - Cross Entropy?
    - KL Divergence

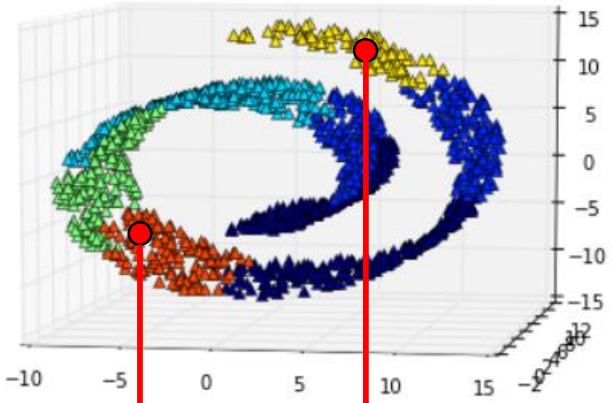
$$D_{KL}(p||q) = \boxed{\left(-\sum_x p(x)\log_2 q(x)\right)} - \left(-\sum_x p(x)\log_2 p(x)\right) = -\sum_i p(x_i)\log_2 \frac{q(x_i)}{p(x_i)}$$

Cross entropy!

    - We made neural networks($q(x)$) to approximate data distribution to correct answer
    - Using a cross entropy as a loss function is not significantly different from using KL divergence

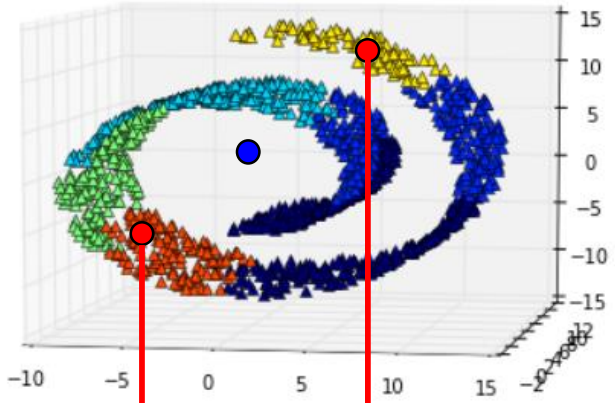
Variational Autoencoder

- Preliminaries
  - Manifold Hypothesis
    - x : high dimensional vector
    - Data is concentrated around a lower dimensional manifold
    - Hope finding a representation z of that manifold



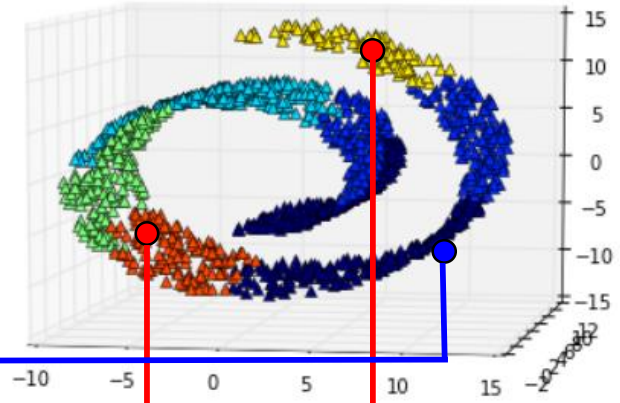
Variational Autoencoder

- Preliminaries
  - Manifold Hypothesis
    - x : high dimensional vector
    - Data is concentrated around a lower dimensional manifold
    - Hope finding a representation z of that manifold



Variational Autoencoder

- Preliminaries
  - Manifold Hypothesis
    - x : high dimensional vector
    - Data is concentrated around a lower dimensional manifold
    - Hope finding a representation z of that manifold

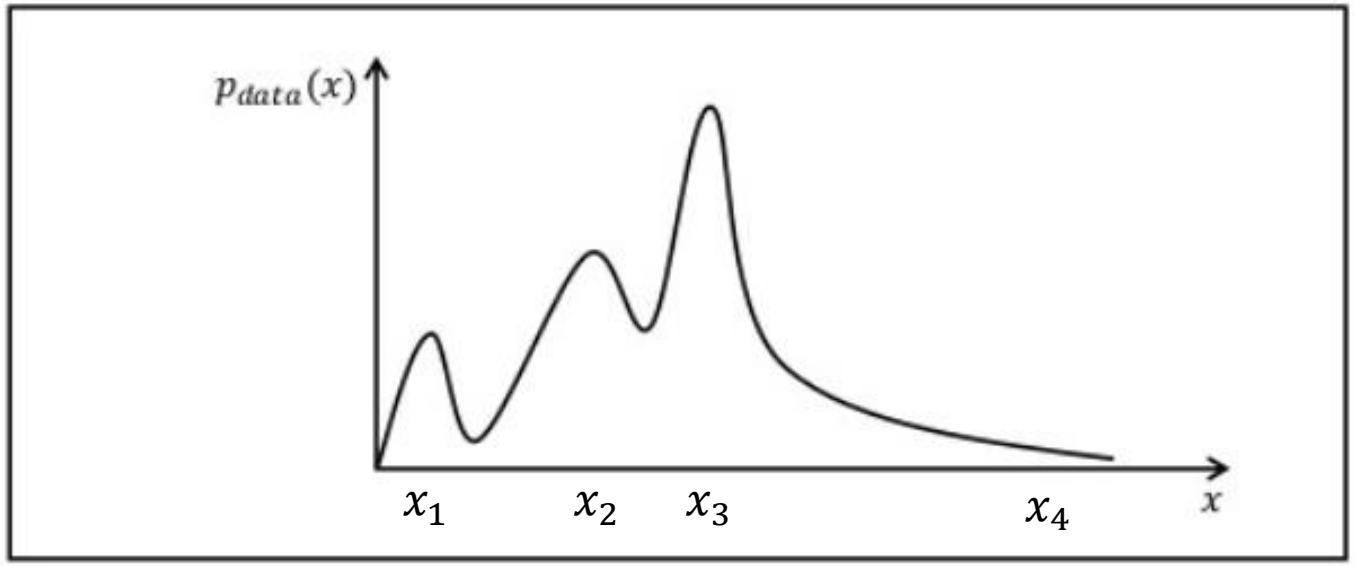


Variational Autoencoder

- Preliminaries
  - Probability Distribution

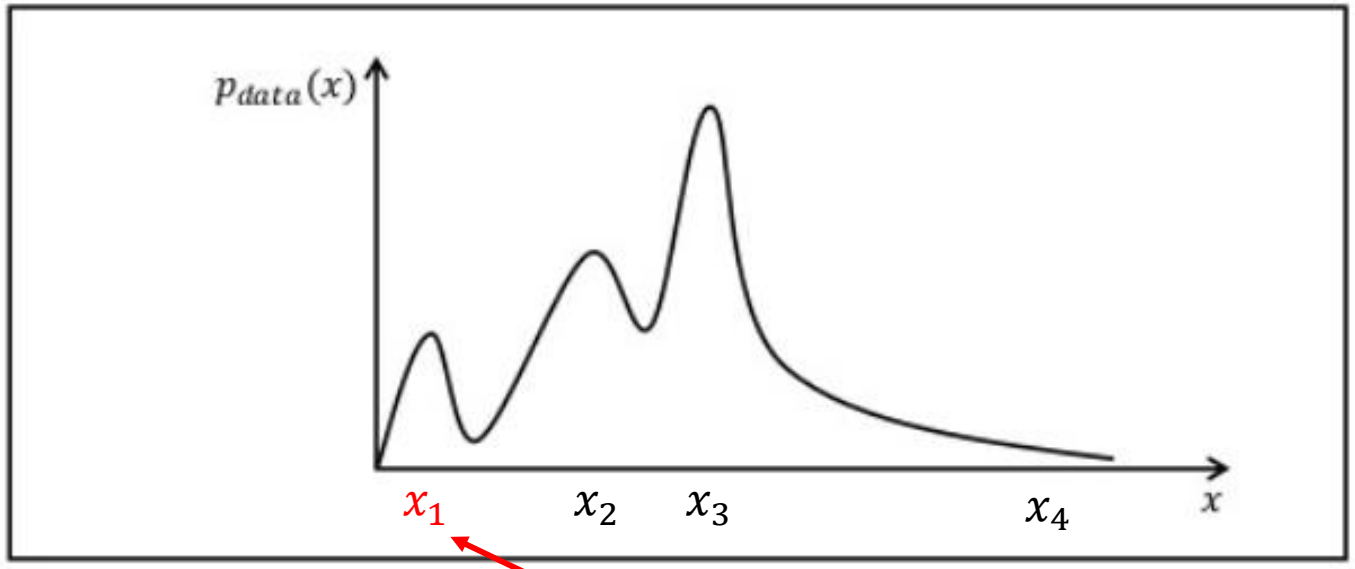
probability density function

There is a $p_{data}(x)$ that represents the distribution of latent space z



Variational Autoencoder

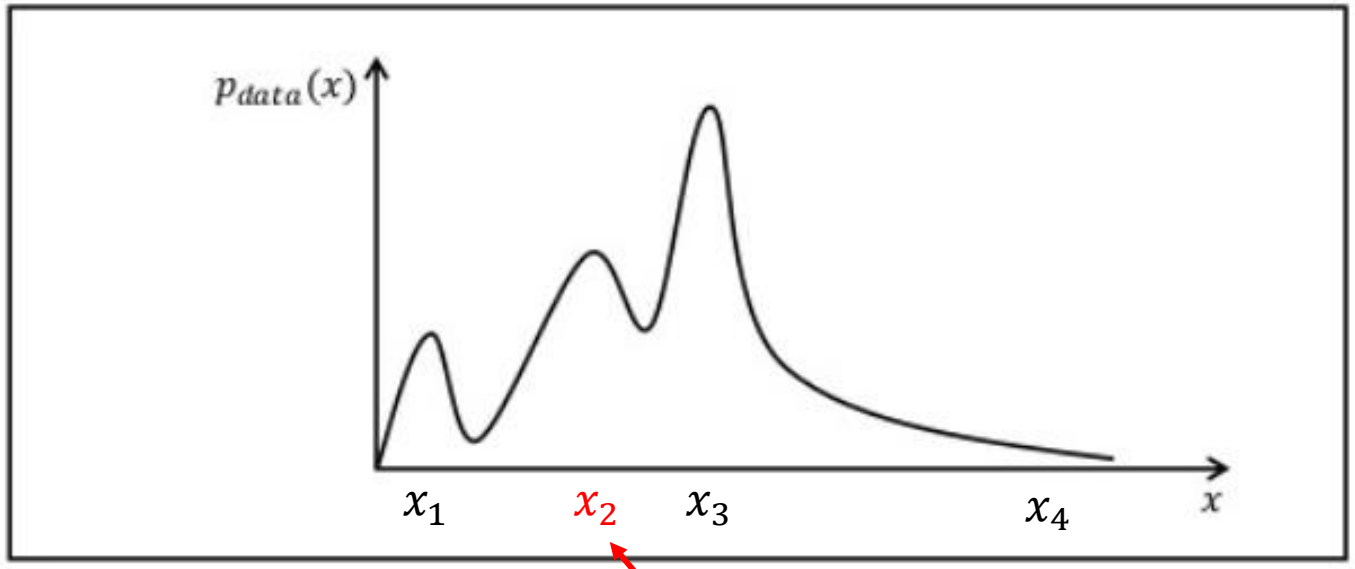
- Preliminaries
  - Probability Distribution



x_1 is a representing a man with glasses

Variational Autoencoder

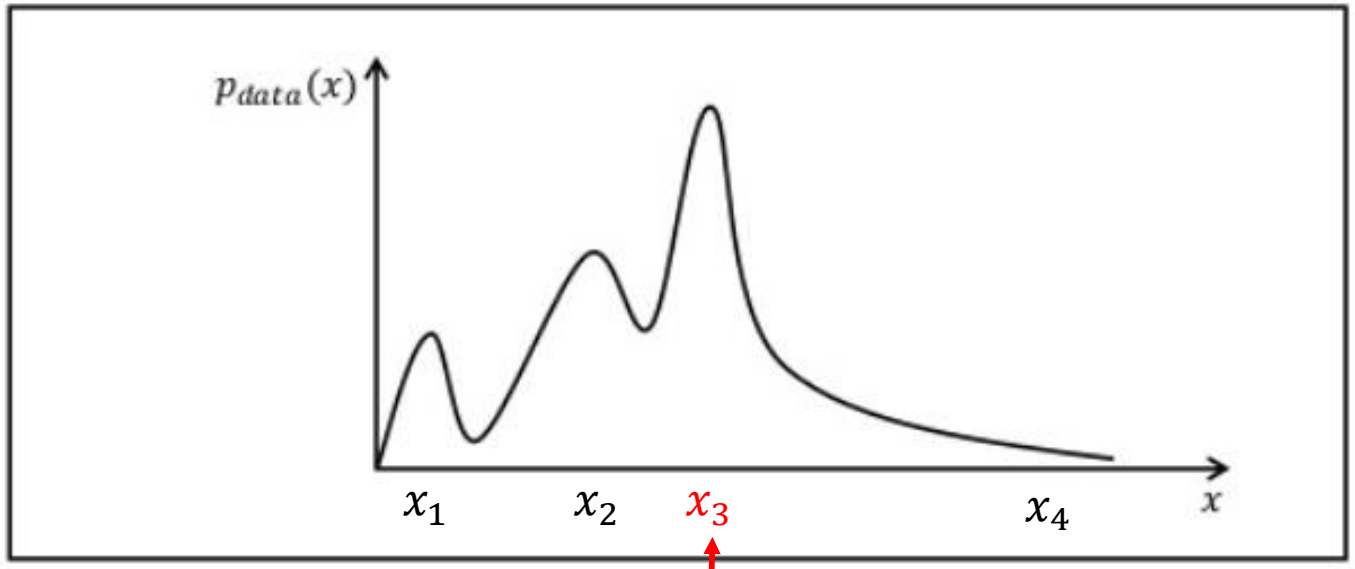
- Preliminaries
  - Probability Distribution



x_2 is a representing a woman with black hair

Variational Autoencoder

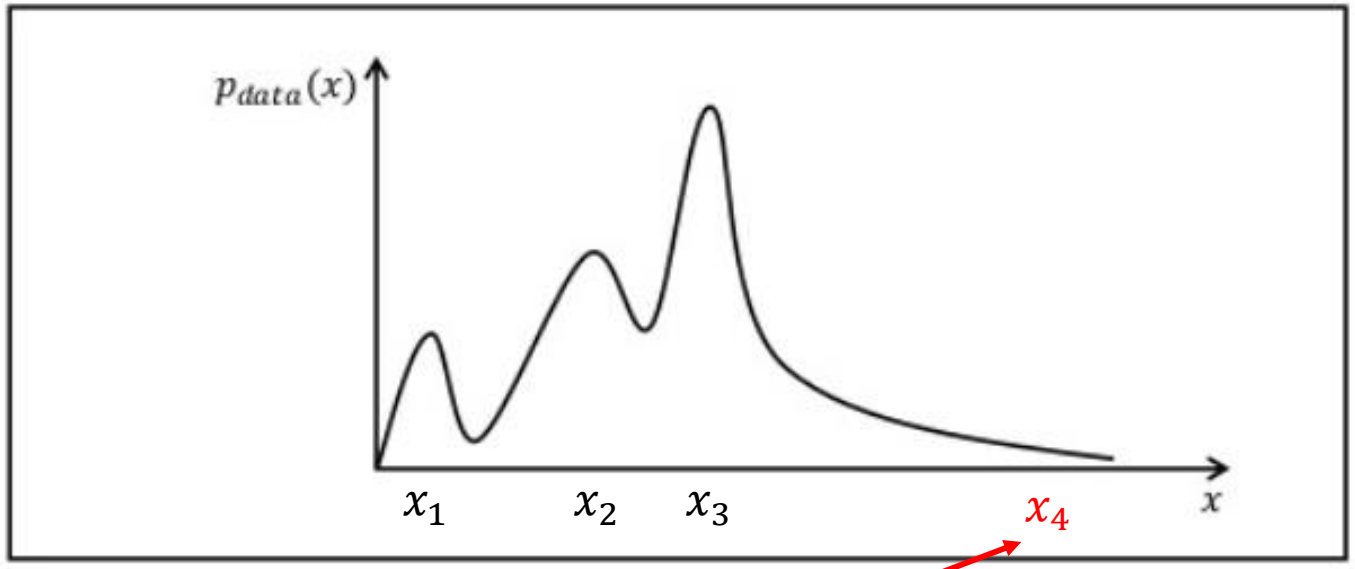
- Preliminaries
  - Probability Distribution



x_3 is a representing a woman with blond hair

Variational Autoencoder

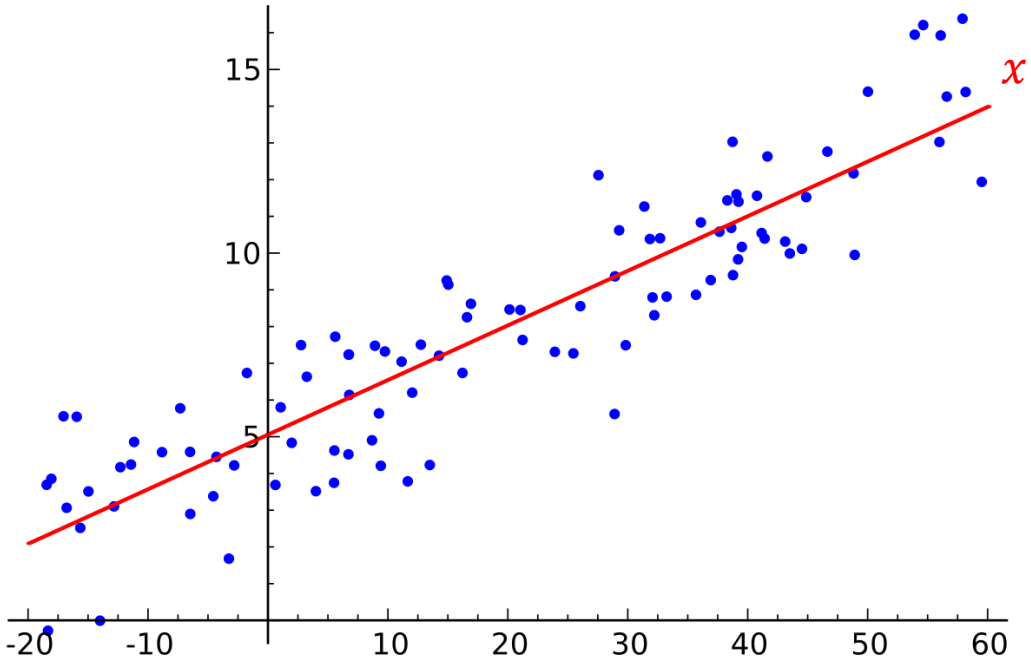
- Preliminaries
  - Probability Distribution



x_4 is a representing very strange images

Variational Autoencoder

- Preliminaries
  - Linear regression as graphical model

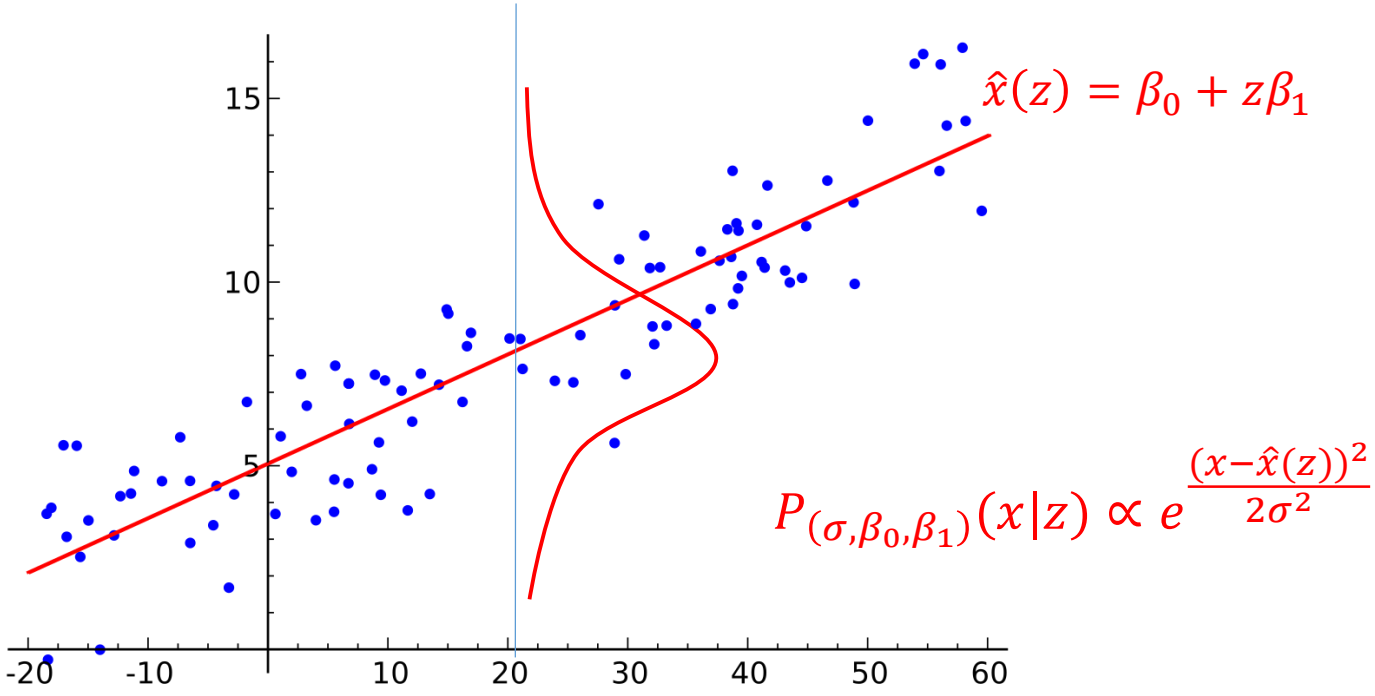


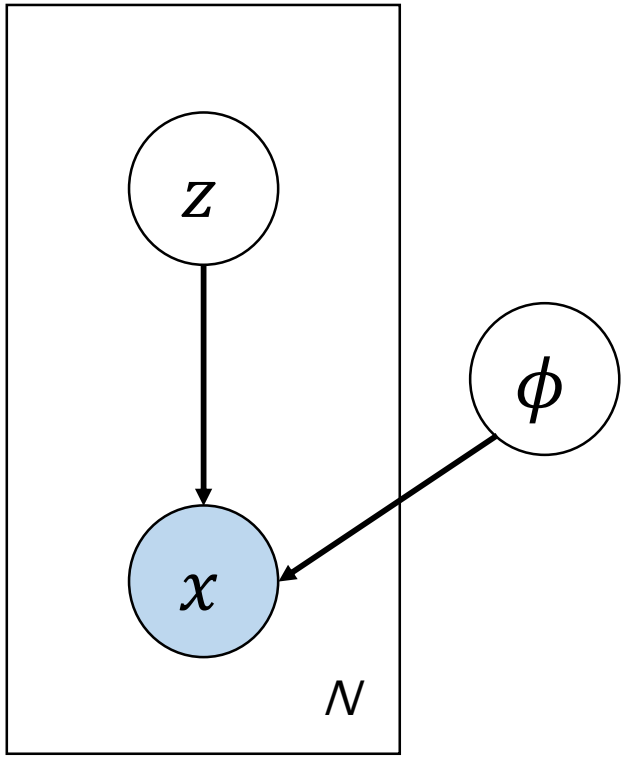
$x = \beta_0 + z\beta_1$

Variational Autoencoder

- Preliminaries
  - Linear regression as graphical model

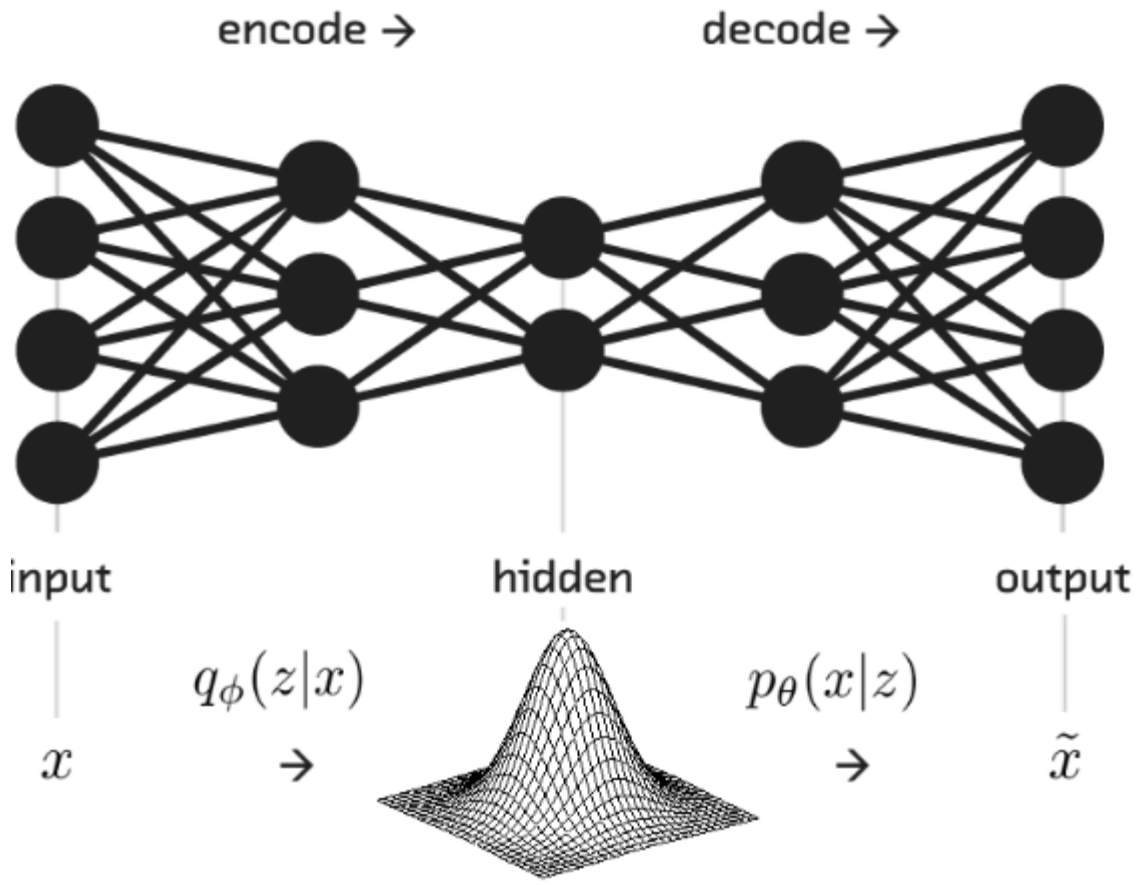
$$Additionally\ have\ P_{\phi}(x|z)$$



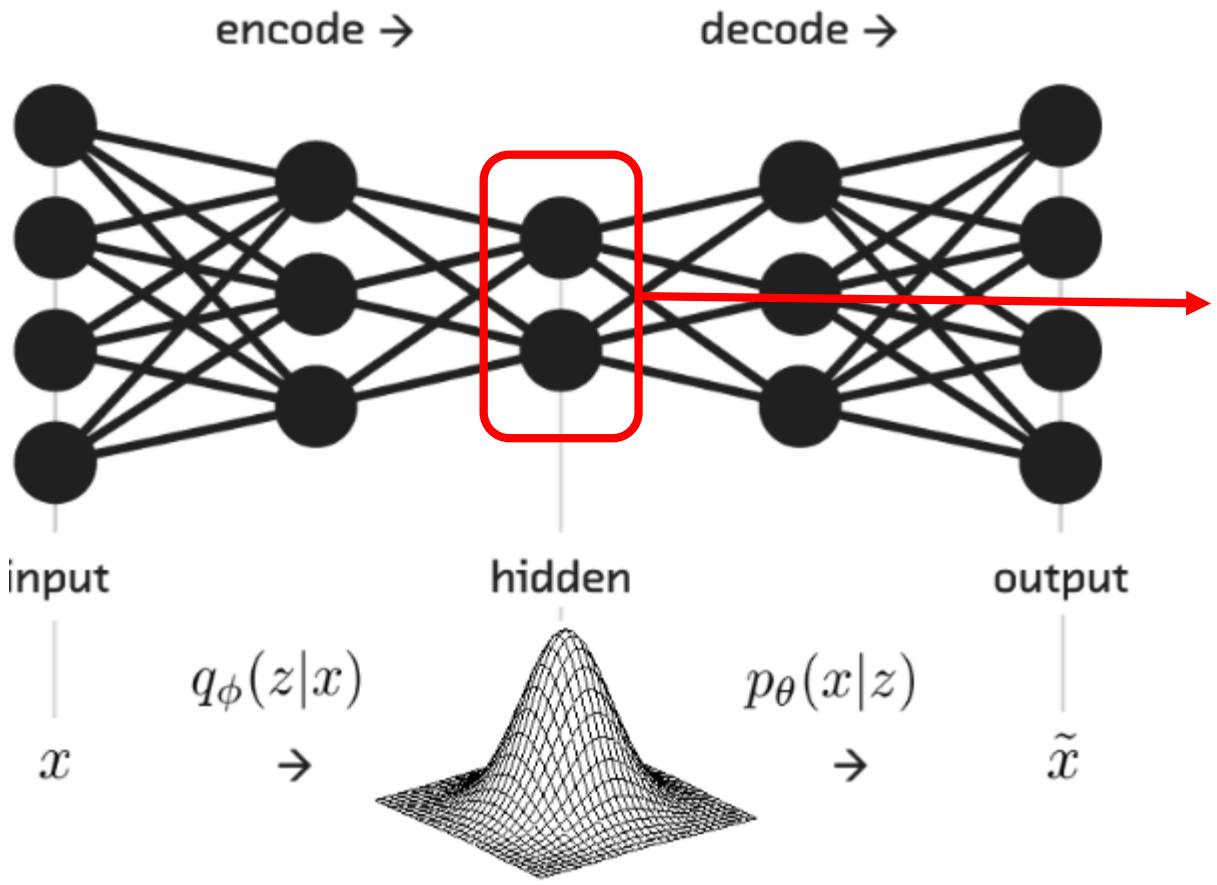


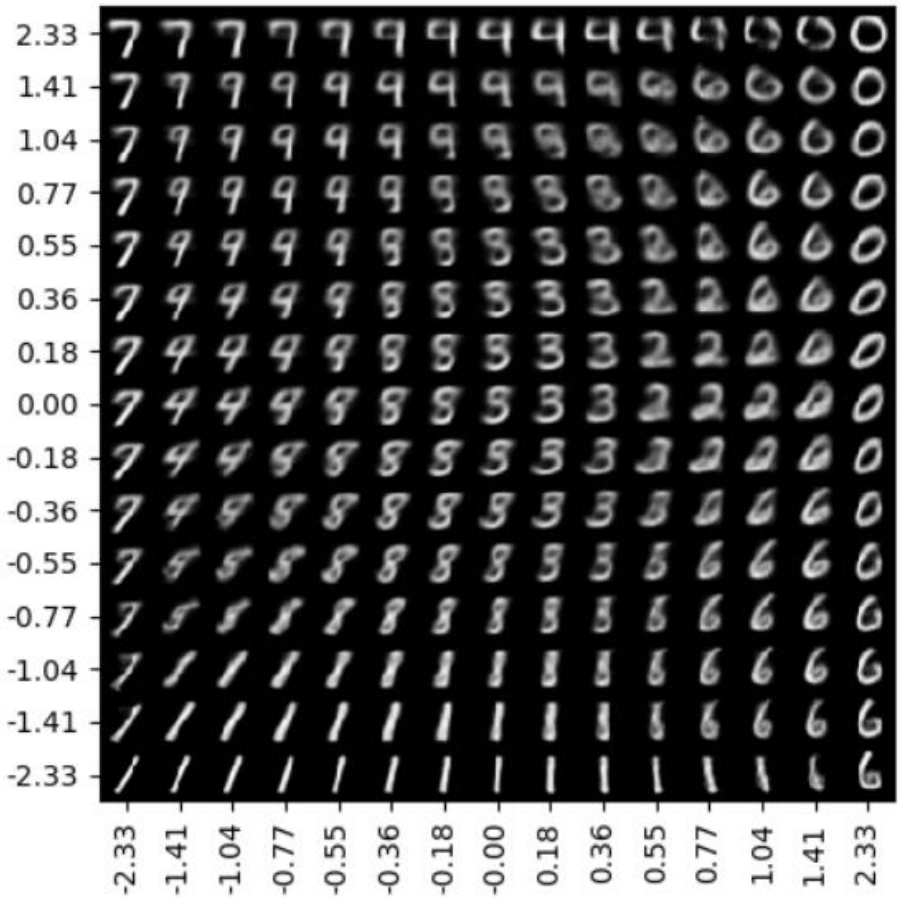
Variational Autoencoder

- Variational Autoencoder
  - I want to know latent space of z, and control!

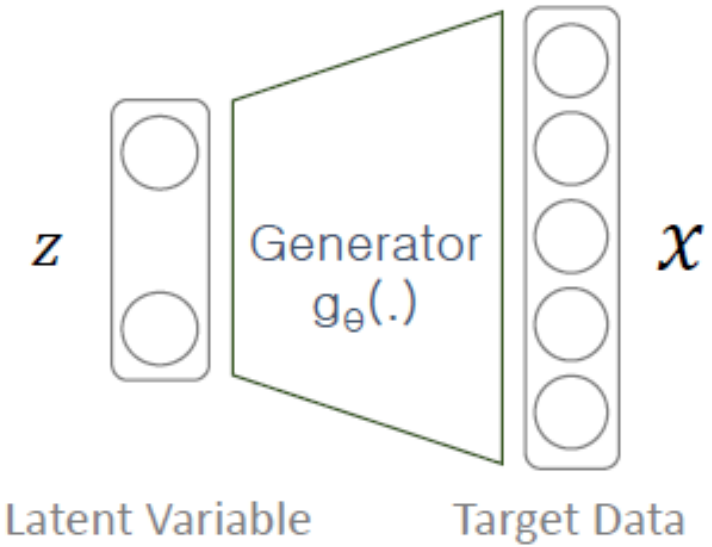


- Variational Autoencoder
  - I want to know latent space of z, and control!





- Generative Model : Latent Variable Model



$z \sim p(z)$	Random variable
$g_\theta(\cdot)$	Deterministic function parameterized by θ
$x = g_\theta(z)$	Random variable

We are aiming maximize the probability of each x in the training set, under the entire generative process, according to :

$$\int p(x|g_\theta(z))p(z)\,dz = p(x)$$

Variational Autoencoder

- Problem : Why don't we use maximum likelihood estimation directly?

$$p(x) = \int p(x|g_\theta(z))p(z)\,dz$$

$$p(x) \approx \sum_i p(x|g_\theta(z_i))p(z_i)$$

(a) (b) (c)

Figure 3: It's hard to measure the likelihood of images under a model using only sampling. Given an image X (a), the middle sample (b) is much closer in Euclidean distance than the one on the right (c). Because pixel distance is so different from perceptual distance, a sample needs to be extremely close in pixel distance to a datapoint X before it can be considered evidence that X is likely under the model.

$$p(x|g_\theta(z)) = \mathcal{N}(x|g_\theta(z), \sigma^2 * I)$$

x : Figure 3(a)

$z_{bad} \rightarrow g_\theta(z_{bad})$: Figure 3(b)

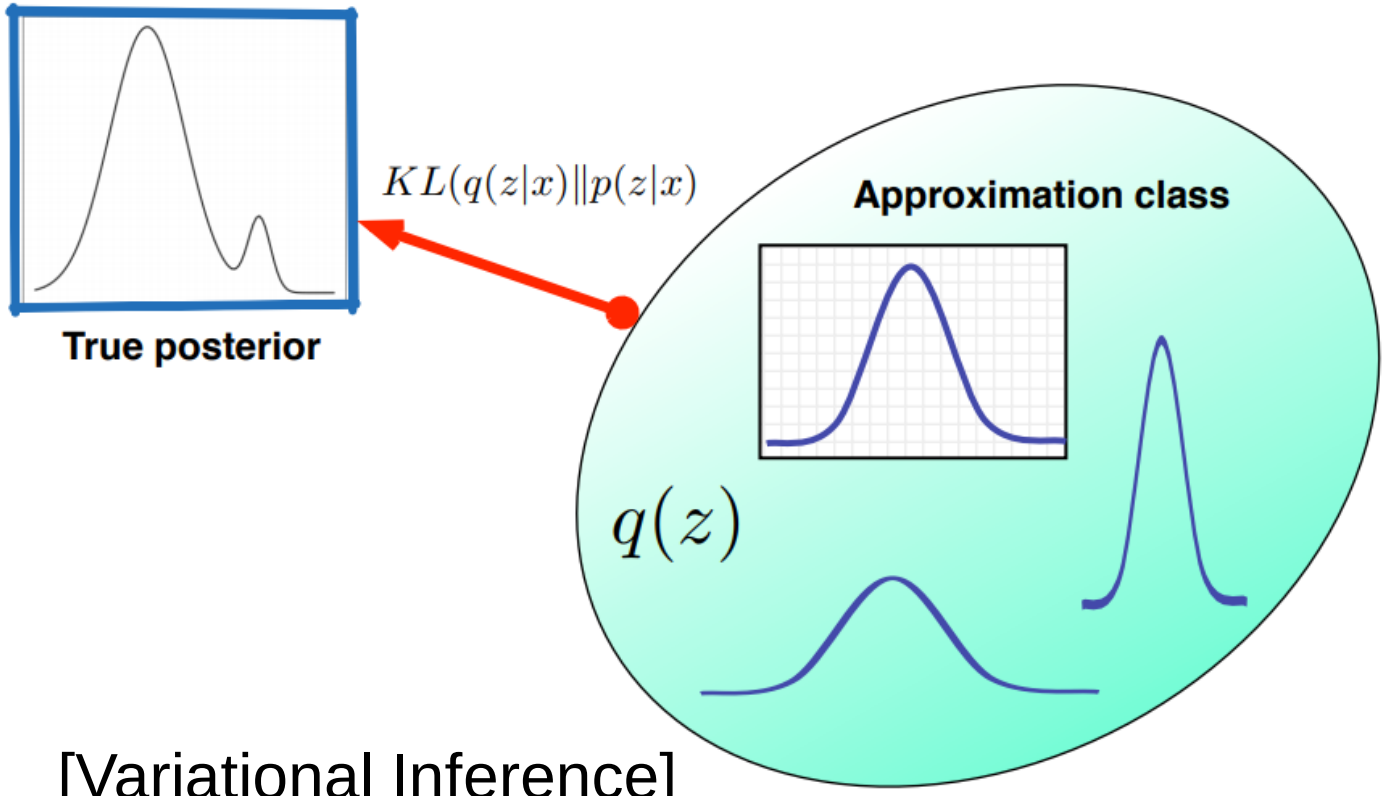
$z_{good} \rightarrow g_\theta(z_{good})$: Figure 3(c) – identical to x but shifted down and to the right by half a pixel

$$\|x - z_{bad}\|^2 < \|x - z_{good}\|^2$$
$$\rightarrow p(x|g_\theta(z_{bad})) > p(x|g_\theta(z_{good}))$$

Tutorial on Variational Autoencoders, https://arxiv.org/pdf/1606.05908.pdf

Variational Autoencoder

- One possible solution : sampling z from $p(z|x)$

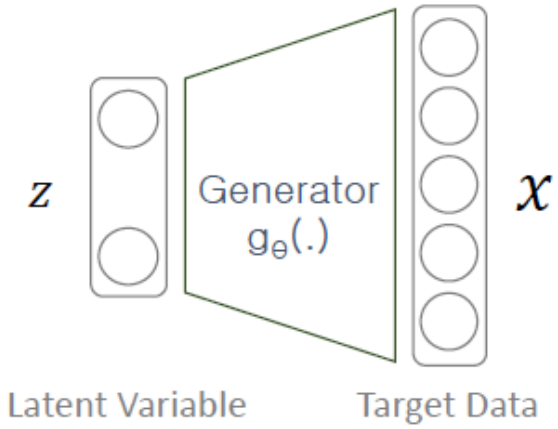


[Variational Inference]

$p(z|x) \approx q_\phi(z|x) \sim z$

Variational Principle

General family of methods for approximating complicated densities by a simpler class of densities



- Variational Inference
  - Relationship among $p(x), p(z|x), q_\phi(z|x)$: **Derivation 1**

$$\log(p(x)) = \log\left(\int p(x|z)p(z)\,dz\right) \geq \int \log(p(x|z))p(z)\,dz$$

Jensen's Inequality
For concave functions $f(\cdot)$,
$f(\mathbb{E}(x)) \geq \mathbb{E}(f(x))$

$$\log(p(x)) = \log\left(\int p(x|z)\frac{p(z)}{q_\phi(z|x)}q_\phi(z|x)dz\right) \geq \int \log\left(p(x|z)\frac{p(z)}{q_\phi(z|x)}\right)q_\phi(z|x)dz$$

$$\log p(x) \geq \int \log(p(x|z))\,q_\phi(z|x)dz - \int \log\left(\frac{q_\phi(z|x)}{p(z)}\right)q_\phi(z|x)dz$$

$$= \underline{\mathbb{E}_{q_\phi(z|x)}[\log(p(x|z))] - KL(q_\phi(z|x)||p(z))}$$

ELBO(ϕ) Variational lower bound
Evidence lower bound(ELBO)

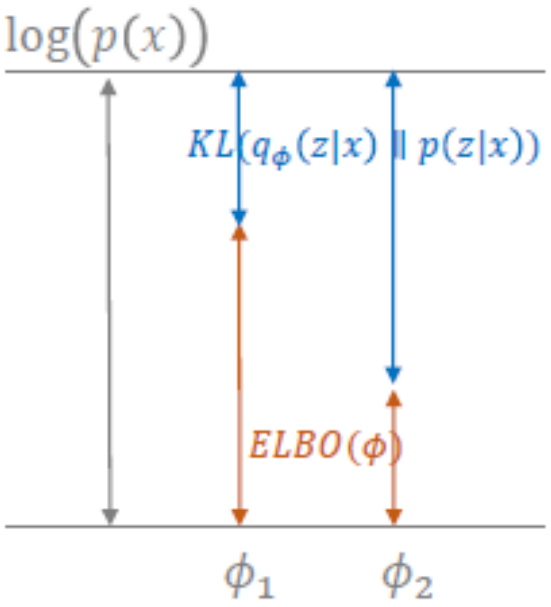
- Variational Inference
  - Relationship among $p(x), p(z|x), q_\phi(z|x)$: **Derivation 2**

$$\log(p(x)) = \int \log((p(x))\, q_\phi(z|x)dz \qquad \leftarrow \qquad \int q_\phi(z|x)dz = 1$$

$$= \int \log\left(\frac{p(x,z)}{p(z|x)}\right) q_\phi(z|x)dz \qquad \leftarrow \qquad p(x) = \frac{p(x,z)}{p(z|x)}$$

$$= \int \log\left(\frac{p(x,z)}{q_\phi(z|x)} \frac{q_\phi(z|x)}{p(z|x)}\right) q_\phi(z|x)dz$$

$$= \underbrace{\int \log\left(\frac{p(x,z)}{q_\phi(z|x)}\right) q_\phi(z|x)dz}_{ELBO(\phi)} + \underbrace{\int \log\left(\frac{q_\phi(z|x)}{p(z|x)}\right) q_\phi(z|x)dz}_{KL(q_\phi(z|x)\,||\,p(z|x))}$$



So instead of minimizing KLD, find ϕ that maximizes $ELBO$! We need to find ϕ of $q_\phi($ [illegible] it w

- Variational Inference
  - Relationship among $p(x), p(z|x), q_\phi(z|x)$: **Derivation 2**

$$\log(p(x)) = ELBO(\phi) + KL(q_\phi(z|x) \,||\, p(z|x))$$

$$q_{\phi^*}(z|x) = \underset{\phi}{\operatorname{argmax}}\, ELBO(\phi)$$

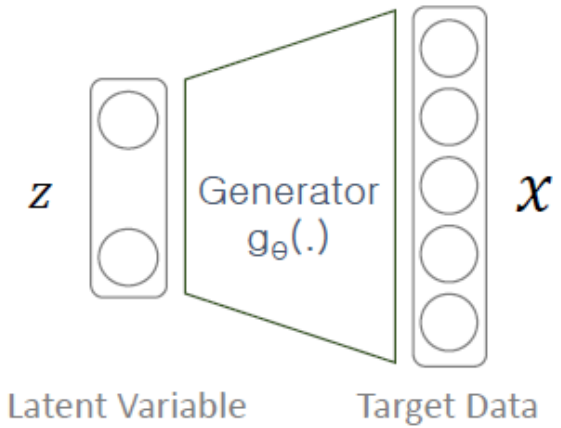
$$\begin{aligned} ELBO(\phi) &= \int \log\left(\frac{p(x,z)}{q_\phi(z|x)}\right) q_\phi(z|x)dz \\ &= \int \log\left(\frac{p(x|z)p(z)}{q_\phi(z|x)}\right) q_\phi(z|x)dz \\ &= \int \log(p(x|z))\, q_\phi(z|x)dz - \int \log\left(\frac{q_\phi(z|x)}{p(z)}\right) q_\phi(z|x)dz \\ &= \mathbb{E}_{q_\phi(z|x)}\big[\log\big(p(x|z)\big)\big] - KL(q_\phi(z|x)||p(z)) \end{aligned}$$

Note that this is a different KL from the previous slide!

Variational Autoencoder

- Loss function

$$p(z|x) \approx q_\phi(z|x) \sim z$$



Optimization 1 on ϕ : Variational Inference

$$\log p(x) \geq \mathbb{E}_{q_\phi(z|x)}\left[\log\left(p(x|z)\right)\right] - KL\left(q_\phi(z|x)\middle|\middle|p(z)\right) = ELBO(\phi)$$

Optimization 2 on θ : Maximum Likelihood Estimation

$$-\sum_i \log\left(p(x_i)\right) \leq -\sum_i \left\{\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL\left(q_\phi(z|x)\middle|\middle|p(z)\right)\right\}$$

Final Optimization Problem

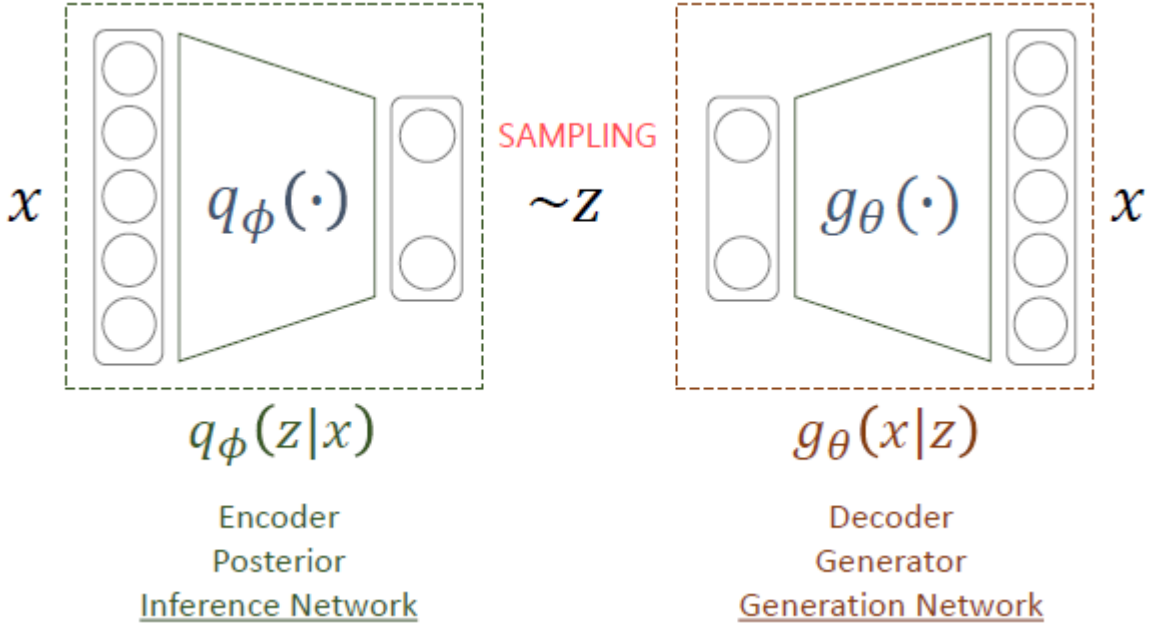
$$\underset{\phi,\theta}{\operatorname{argmin}} \sum_i -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL\left(q_\phi(z|x)\middle|\middle|p(z)\right)$$

Variational Autoencoder

- Loss function

$$\underset{\phi,\theta}{\operatorname{argmin}} \sum_i -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL(q_\phi(z|x)||p(z))$$

$$\mathcal{L}(\theta,\phi,x) = -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL(q_\phi(z|x)||p(z))$$



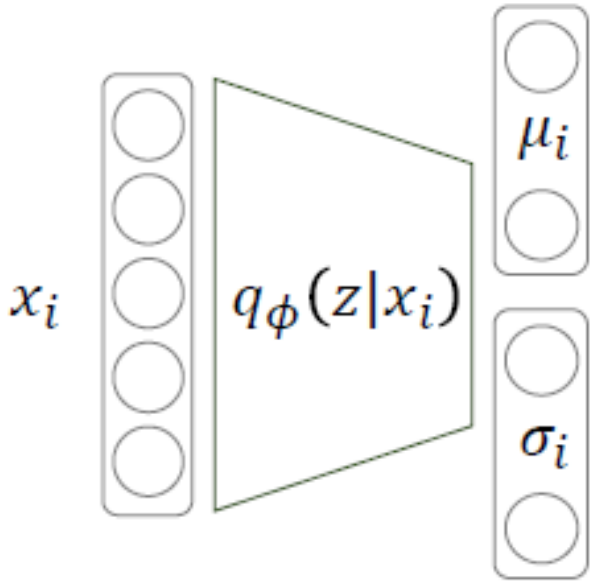
The mathematical basis of VAEs actually has relatively little to do with classical autoencoders

Variational Autoencoder

- Loss function

$$\underset{\phi,\theta}{\operatorname{argmin}} \sum_i -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL(q_\phi(z|x)||p(z))$$

$$\mathcal{L}(\theta,\phi,x) = \underbrace{-\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))]}_{\text{Reconstruction Error}} - \underbrace{KL(q_\phi(z|x)||p(z))}_{\text{Regularization}}$$



Assumption 1

[Encoder : approximation class]

Multivariate Gaussian distribution with a diagonal covariance

$$q_\phi(z|x_i) \sim N(\mu_i, \sigma_i^2, I)$$

Assumption 2

[Prior]

Multivariate normal distribution

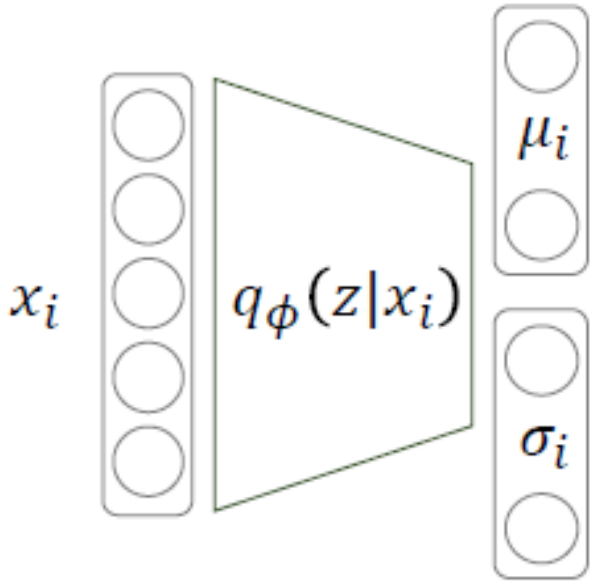
$$p(z) \sim N(0, I)$$

Variational Autoencoder

- Loss function

$$\underset{\phi,\theta}{\operatorname{argmin}} \sum_i -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL(q_\phi(z|x)||p(z))$$

$$\mathcal{L}(\theta,\phi,x) = \underbrace{-\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))]}_{\text{Reconstruction Error}} - \underbrace{KL(q_\phi(z|x)||p(z))}_{\text{Regularization}}$$

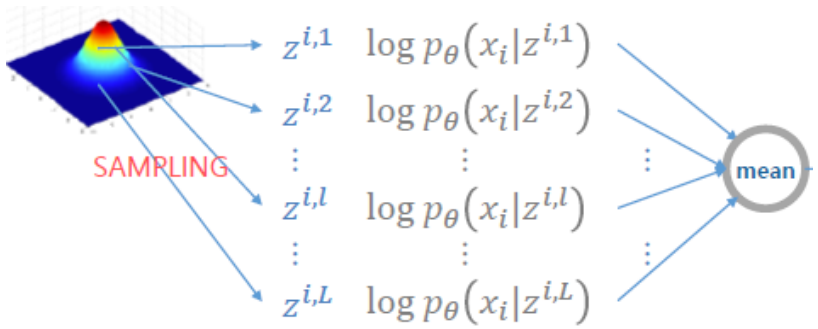


Reconstruction Error

$$-\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))]$$

$$= \int \log p_\theta(x_i|z)\, q_\phi(z|x_i)dz$$

$$\approx \frac{1}{L}\sum_{z^{i,l}} \log\left(p_\theta(x_i|z^{i,l})\right)$$

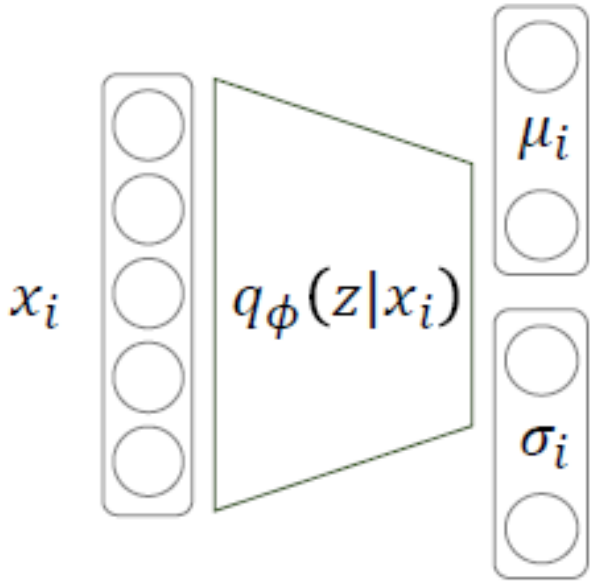


Variational Autoencoder

- Loss function

$$\underset{\phi,\theta}{\operatorname{argmin}} \sum_i -\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))] - KL(q_\phi(z|x)||p(z))$$

$$\mathcal{L}(\theta,\phi,x) = \underbrace{-\mathbb{E}_{q_\phi(z|x_i)}[\log(p(x_i|g_\theta(z)))]}_{\text{Reconstruction Error}} - \underbrace{KL(q_\phi(z|x)||p(z))}_{\text{Regularization}}$$



Regularization : KL Divergence

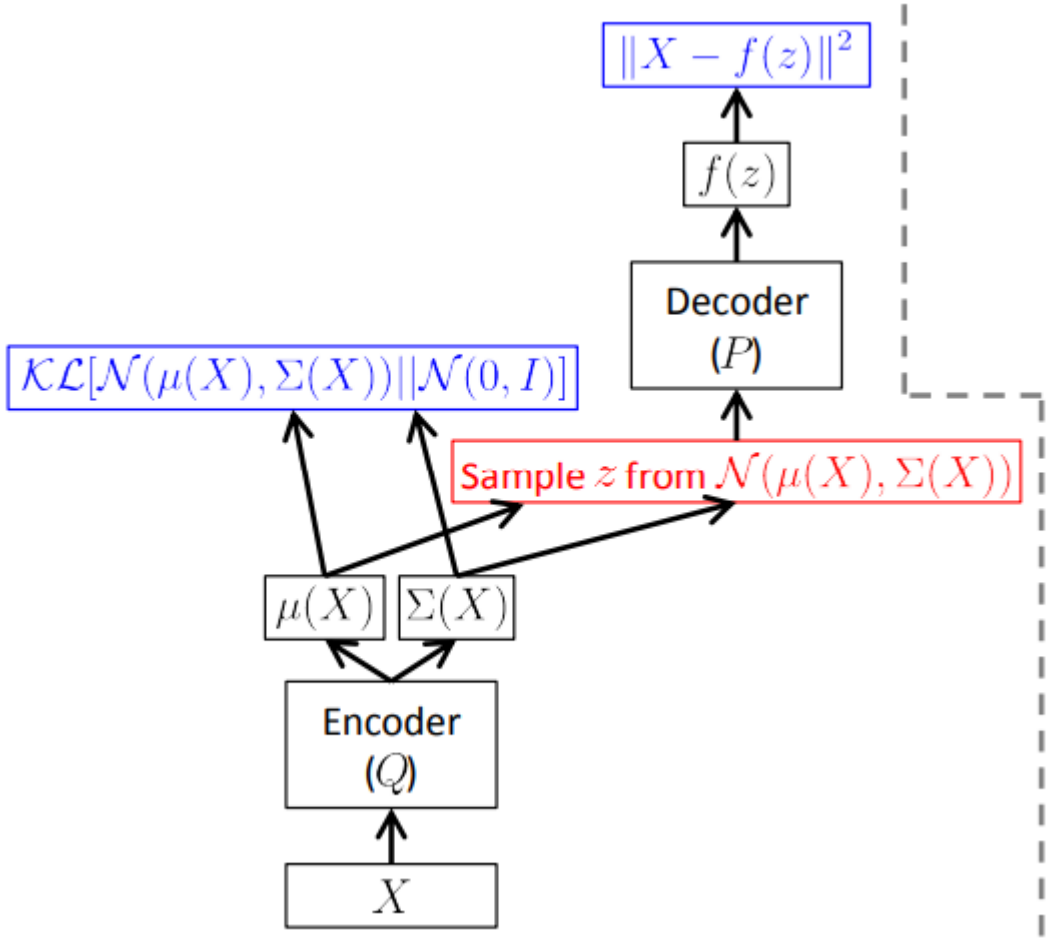
$$KL(q_\phi(z|x)||p(z))$$

$$= KL(N(\mu_i, \sigma_i^2, I)||N(0, I))$$

$$= -\frac{1}{2}\sum_j (1 + \log(\sigma_j^2) - \mu_j^2 + \sigma_j^2)$$

Variational Autoencoder

- Reparameterization Trick

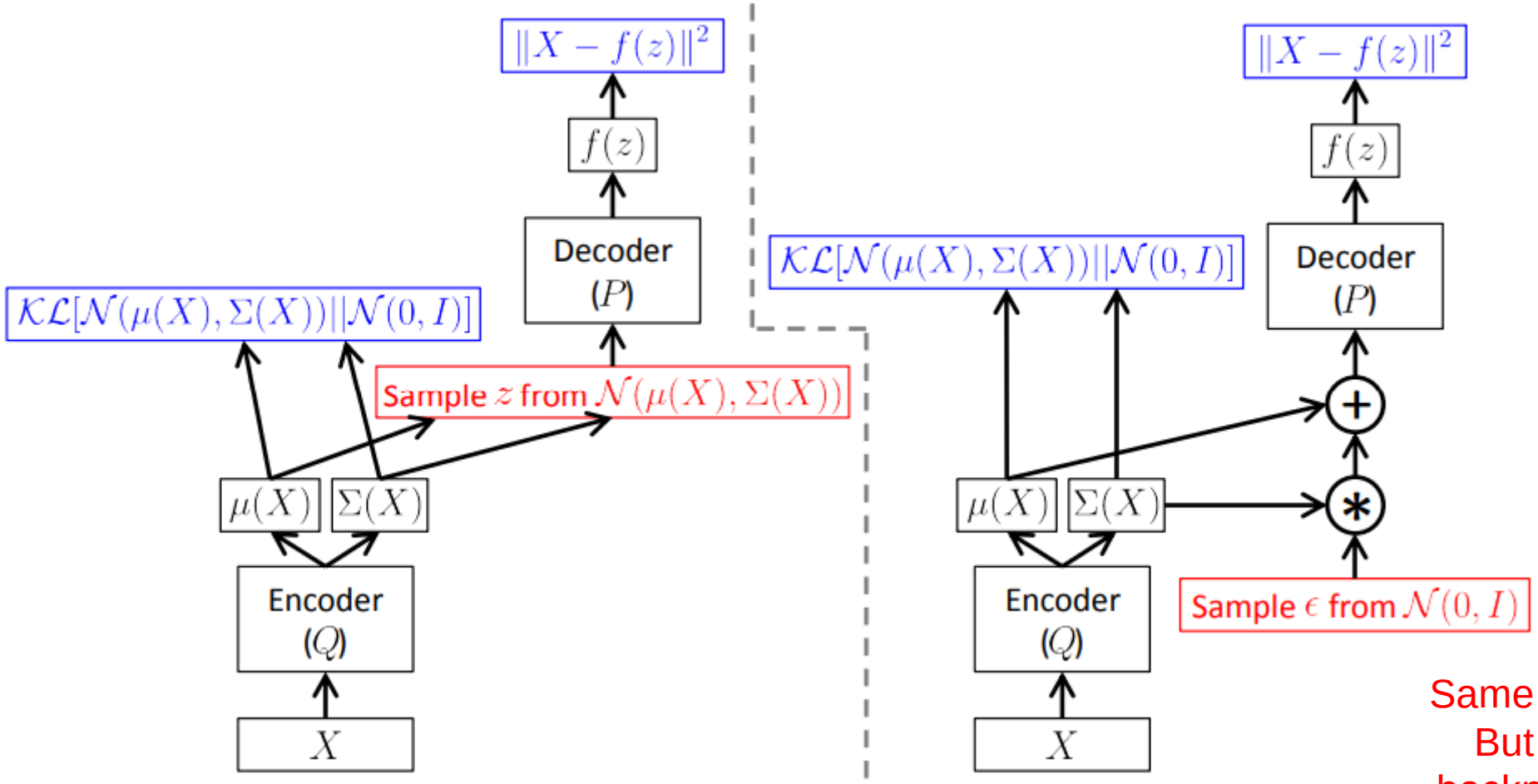


Sampling Process

$$z^{i,l} \sim N(\mu_i, \sigma_i^2, I)$$

Variational Autoencoder

- Reparameterization Trick



Same distribution
But it makes
backpropagation
possible!

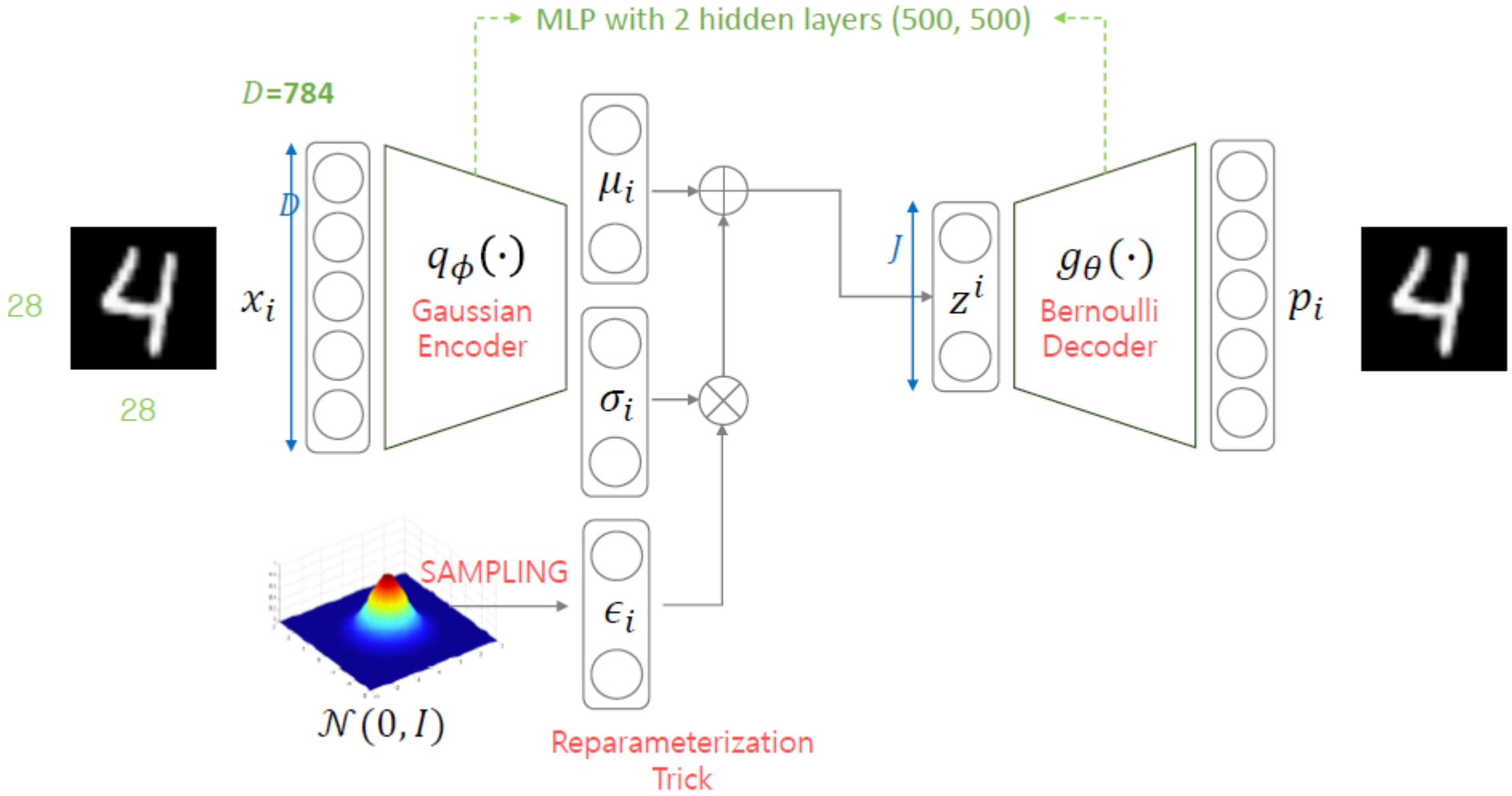
Sampling Process

$$z^{i,l} \sim N(\mu_i, \sigma_i^2, I)$$

$$z^{i,l} = \mu_i + \sigma_i^2 \odot \epsilon$$
$$\epsilon \sim N(0, I)$$

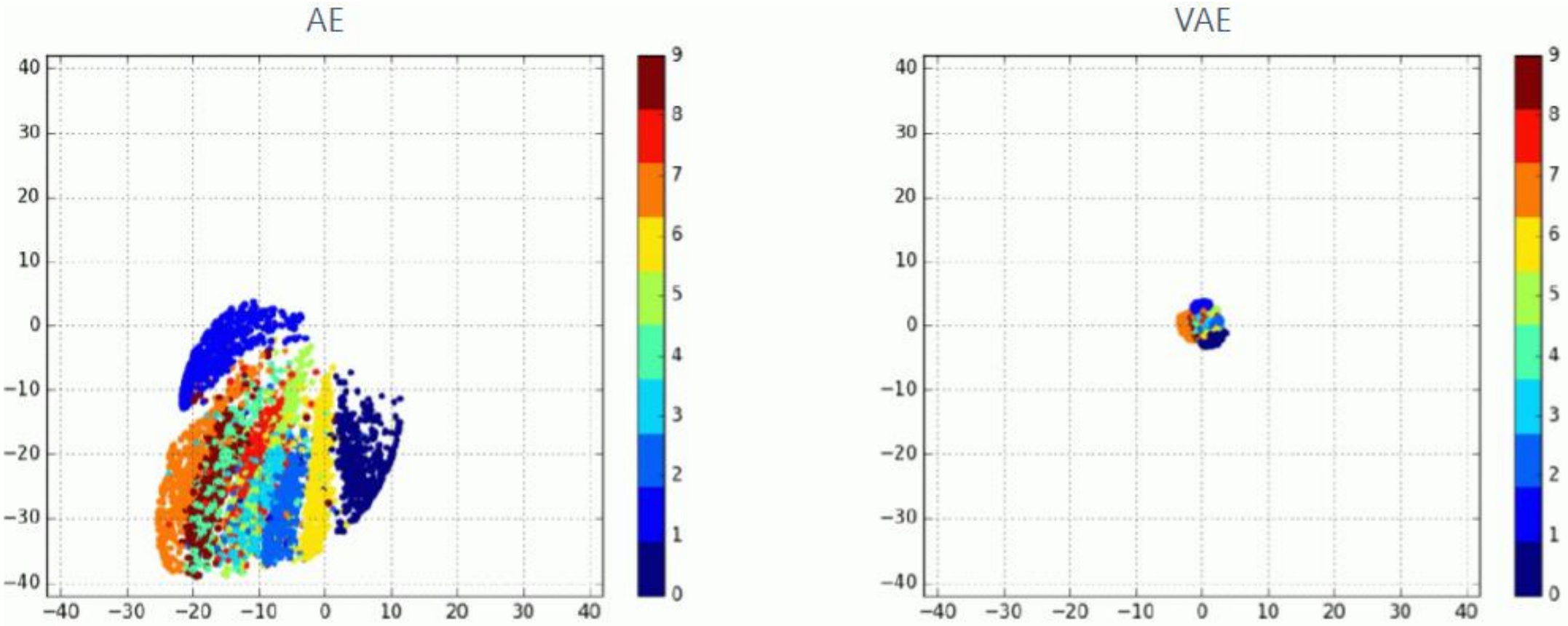
Variational Autoencoder

- Architecture



Variational Autoencoder

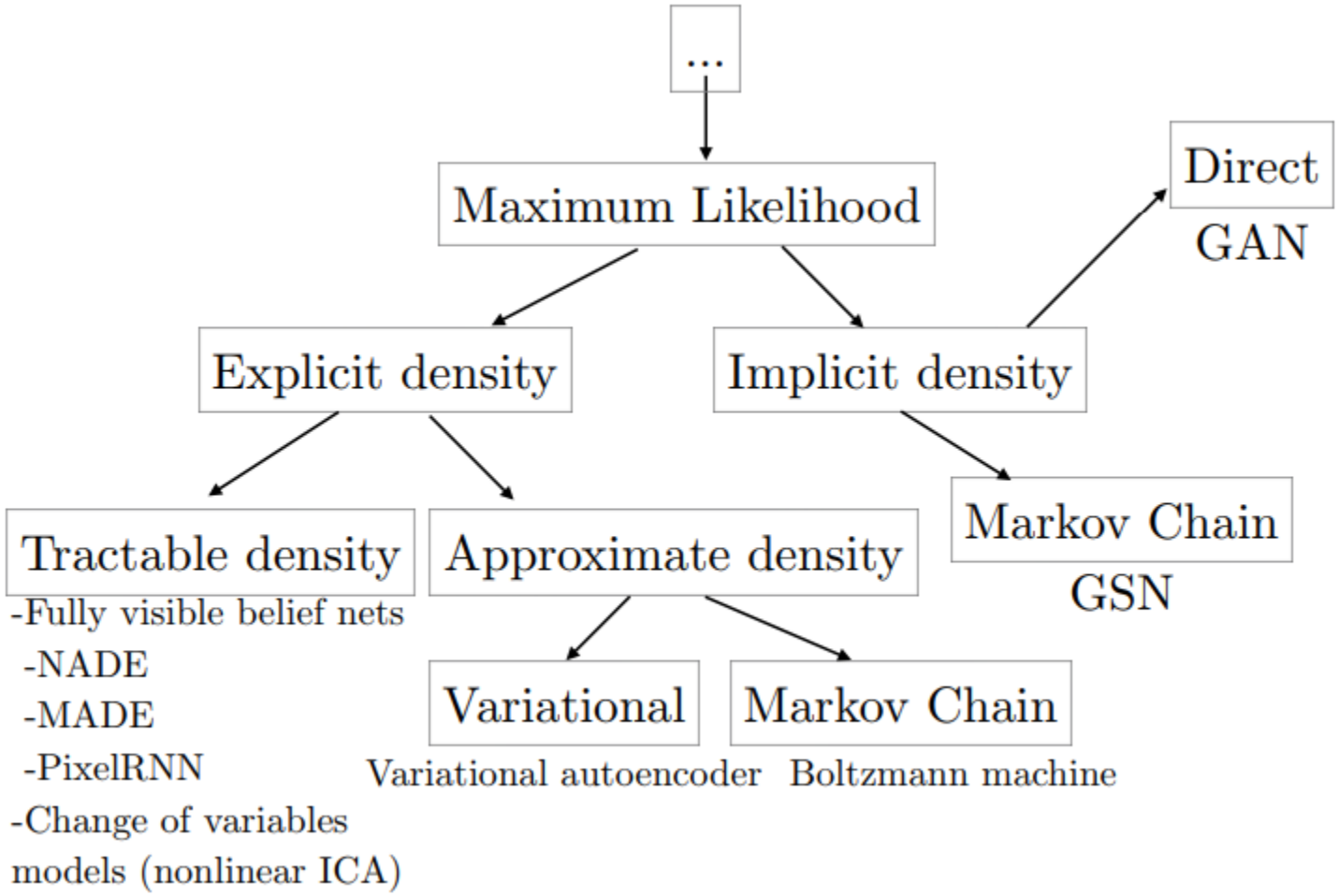
- Learned Manifold



III. Discussion : GAN vs VAE

Discussion : GAN vs VAE

- Generative Model



Discussion : GAN vs VAE

- Pros and cons of GAN vs VAE

Quora Home Answer Spaces Notifications Search Quora

Generative Models (statistics) Autoencoder +4

What are the pros and cons of Generative Adversarial Networks vs Variational Autoencoders?

Answer Follow · 68 Request

Ad by Tipalti

Automated payment processing. Payouts with 5+ payment methods.

Tipalti makes mass payouts to global suppliers, partners, affiliates, publishers, and freelancers.

Learn more at tipalti.com

1 Answer

Yoshua Bengio, My lab has been one of the three that started the deep learning approach, back in 2006, along with Hinton's...

Answered Aug 16, 2016 · Upvoted by Alberto Bietti, PhD student in machine learning. Former ML engineer and Tao Xu, Built ML systems at Airbnb, Quora, Facebook and Microsoft.

Summary

An advantage for VAEs (Variational AutoEncoders) is that there is a clear and

https://www.quora.com/What-are-the-pros-and-cons-of-Generative-Adversarial-Networks-vs-Variational-Autoencoders

Discussion : GAN vs VAE

- Pros and cons of GANs vs VAEs

VAE

- There is a clear and recognized way to evaluate the quality of the model(log-likelihood, either estimated by importance sampling or lower-bounded)

- Because of the injected noise and imperfect reconstruction, and with the standard decoder(with factorized output distribution), the generated samples are much more blurred

GAN

- They tend to yield nicer images

- GANs tend to be much more finicky to train than VAEs

Quora Home Answer Spaces Notifications Search Quora

Generative Models (statistics) Autoencoder +4

What are the pros and cons of Generative Adversarial Networks vs Variational Autoencoders?

Answer Follow · 68 Request

Ad by Tipalti

Automated payment processing. Payouts with 5+ payment methods.

Tipalti makes mass payouts to global suppliers, partners, affiliates, publishers, and freelancers.

Learn more at tipalti.com

1 Answer

Yoshua Bengio, My lab has been one of the three that started the deep learning approach, back in 2006, along with Hinton's...

Answered Aug 16, 2016 · Upvoted by Alberto Bietti, PhD student in machine learning. Former ML engineer and Tao Xu, Built ML systems at Airbnb, Quora, Facebook and Microsoft.

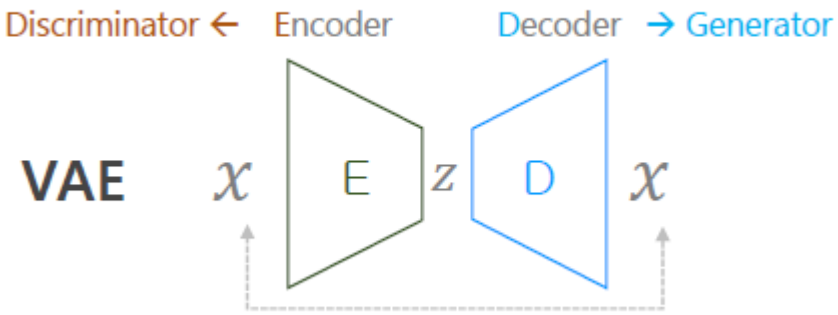
An advantage for VAEs (Variational AutoEncoders) is that there is a clear and

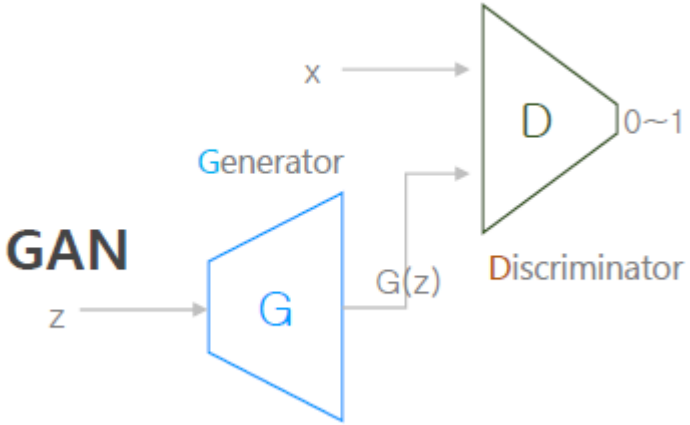
https://www.quora.com/What-are-the-pros-and-cons-of-Generative-Adversarial-Networks-vs-Variational-Autoencoders

Discussion : GAN vs VAE

- Generative Model

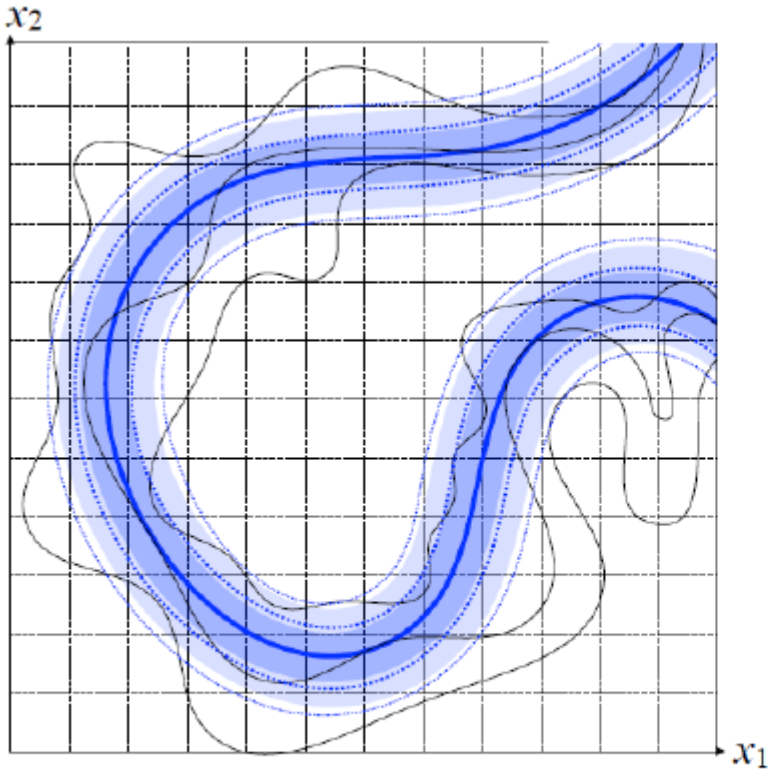
Model	Optimization	Image Quality	Generalization
VAE	• Stochastic gradient decent • Converge to local minimum • Easier	Smooth Blurry	Tend to remember input images
GAN	• Alternating stochastic gradient descent • Converge to saddle points • Harder - Mode collapsing - Unstable convergence	Sharp Artifact	Generate new unseen images



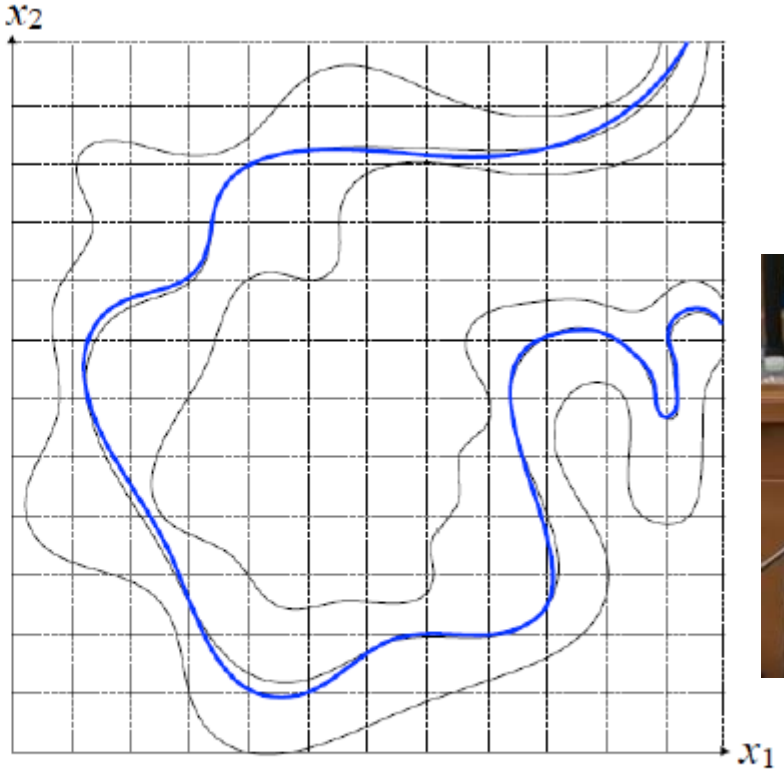


Discussion : GAN vs VAE

- Generative Model



VAE : maximum likelihood approach



GAN

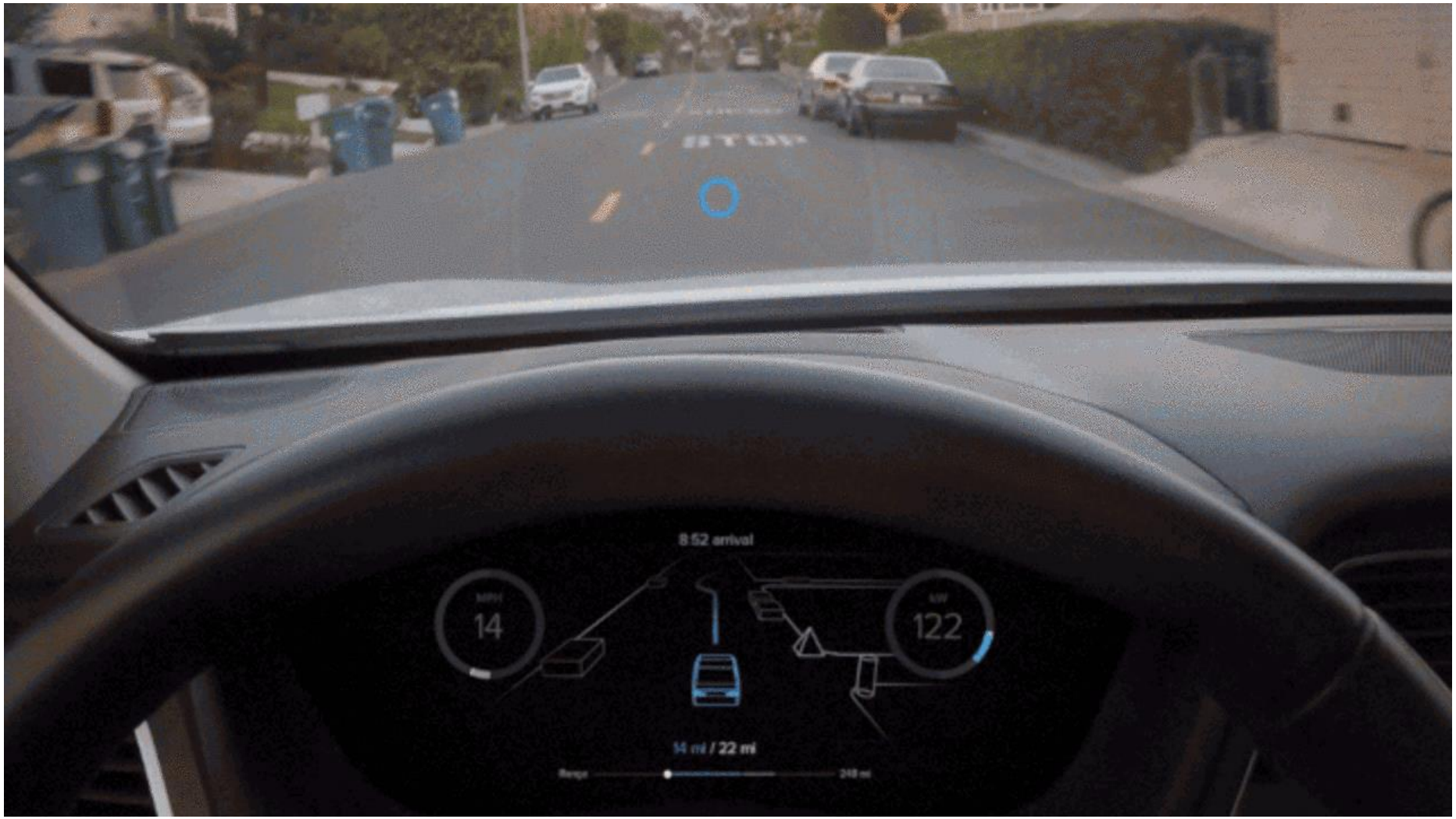
Mode collapse!
“나는 한놈만 팬다…”

modes

IV. Recent Trends in Deep Learning

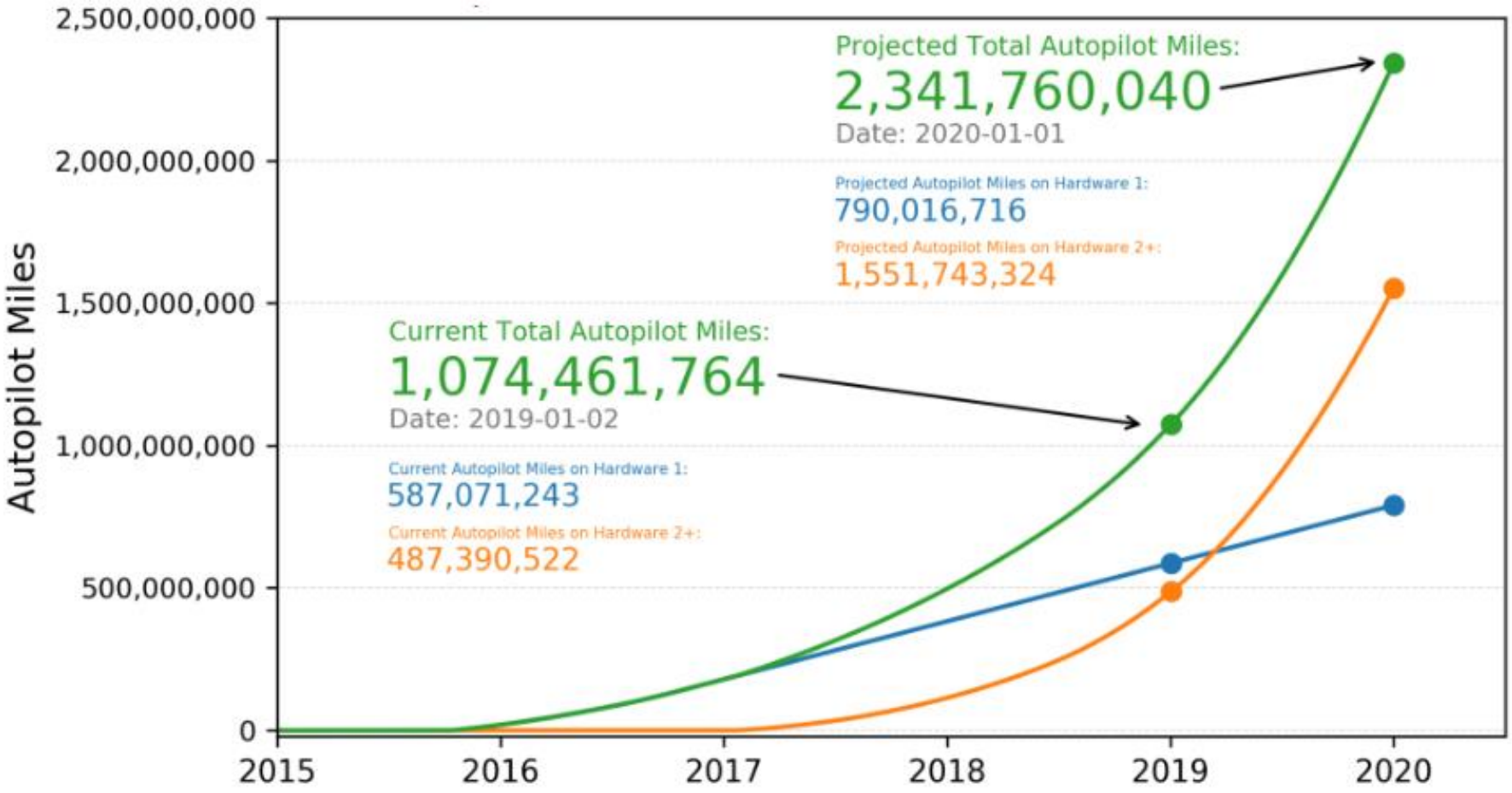
Recent Trends in Deep Learning

- Autonomous Driving Car(Self-Driving Car)



Recent Trends in Deep Learning

- Autonomous Driving Car
  - Autopilot reaches 1 billion miles!



Deep Learning State of the Art (2019) - MIT

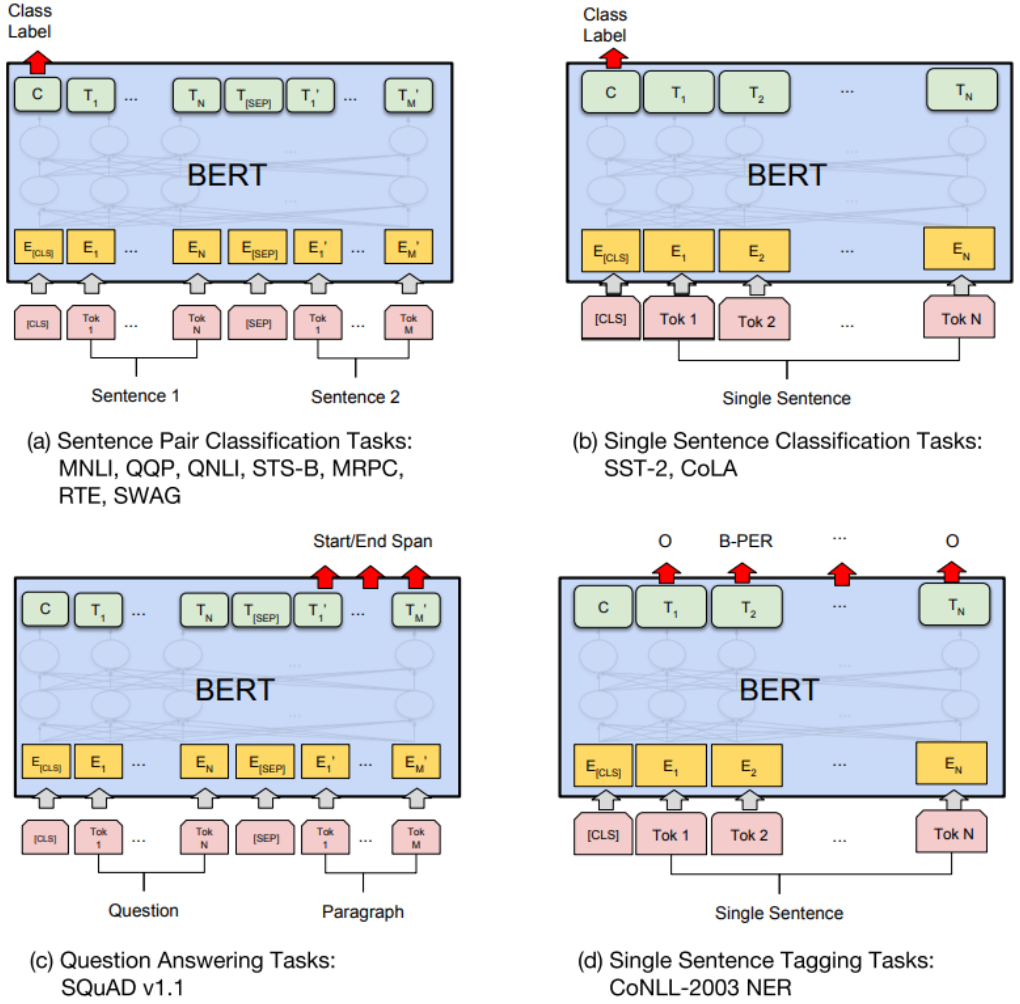
Recent Trends in Deep Learning

- Autonomous Driving Car
  - Autopilot reaches 1 billion miles!

Announcements for Consumer-Facing Fully-Autonomous Vehicle (Testing and Beyond)

- **Tesla**: 2019
- **Nissan**: 2020
- **Honda**: 2020
- **Toyota**: 2020 (highway)
- **Renault-Nissan**: 2020 (urban)
- **Hyundai**: 2020 (highway)
- **Volvo**: 2021 (highway)
- **BMW**: 2021
- **Ford**: 2021
- **Fiat-Chrysler:** 2021
- **Daimler**: 2020-25

Natural Language Processing : BERT

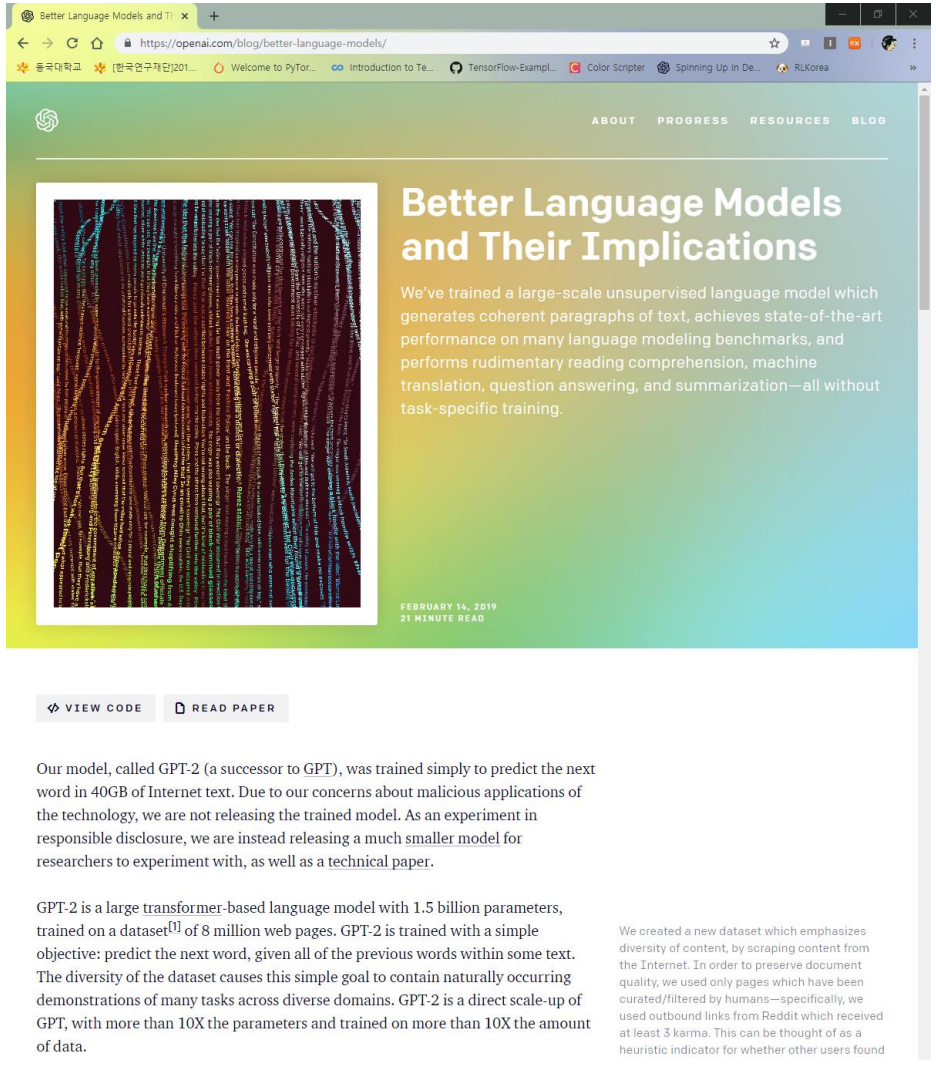


BERT Application

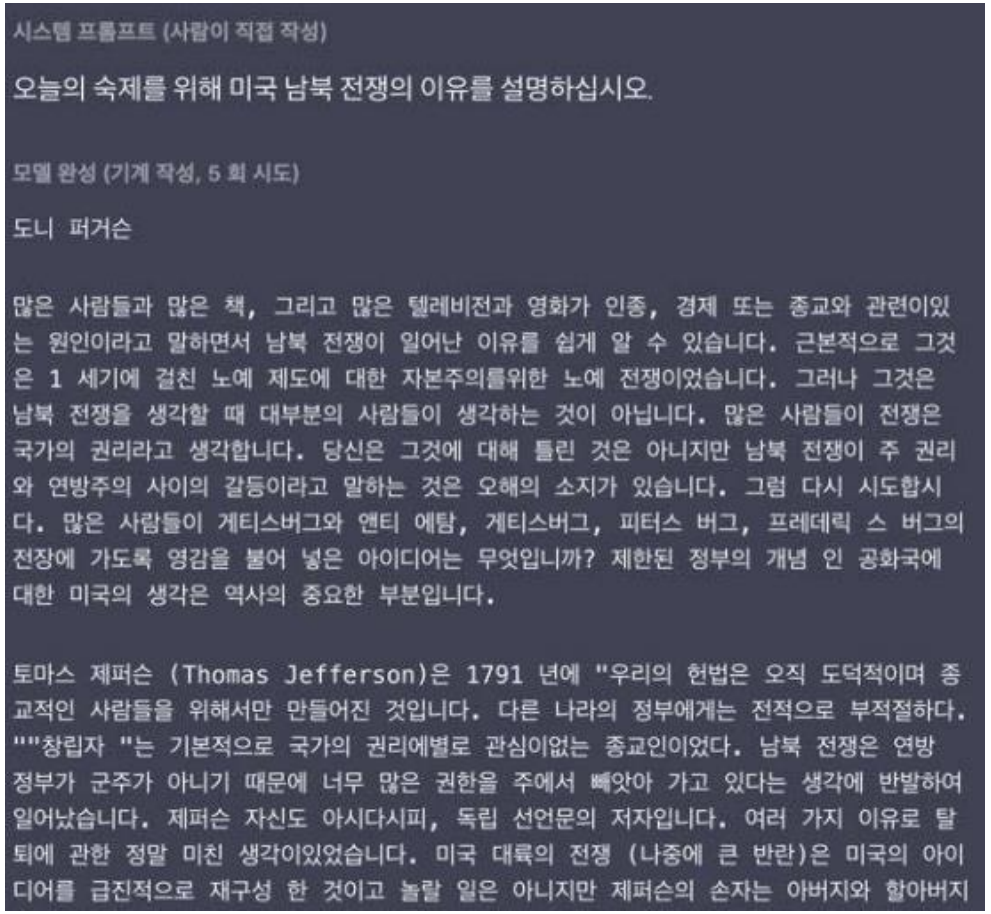
- Create contextualized word embeddings
- Sentiment Classification
- Sentence pair classification
- Sentence pair similarity
- Sentence tagging
- Question answering

Devlin, Jacob, et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, 2018

- Natural Language Processing : GPT-2



A large transformer-based language model with 1.5 billion parameters trained on a dataset of 8 million web pages



https://openai.com/blog/better-language-models/

- AutoML & NASNets(Neural Architecture Search)

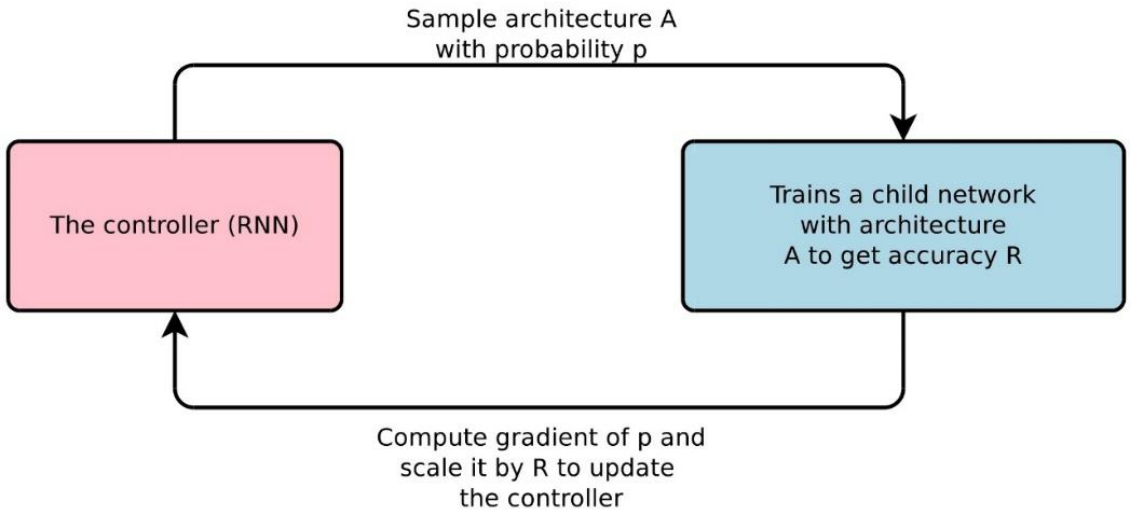
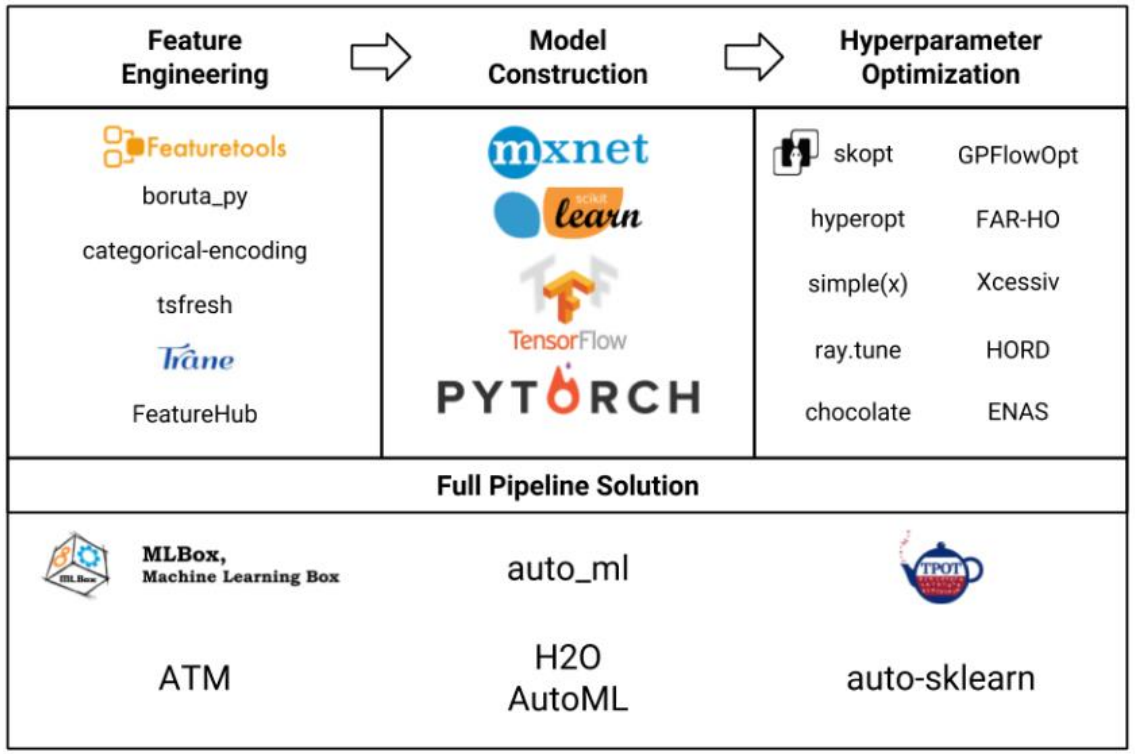


Figure 1: An overview of Neural Architecture Search.



Zoph, Barret, and Quoc V. Le. "Neural architecture search with reinforcement learning.“, 2016

- AutoML & NAS

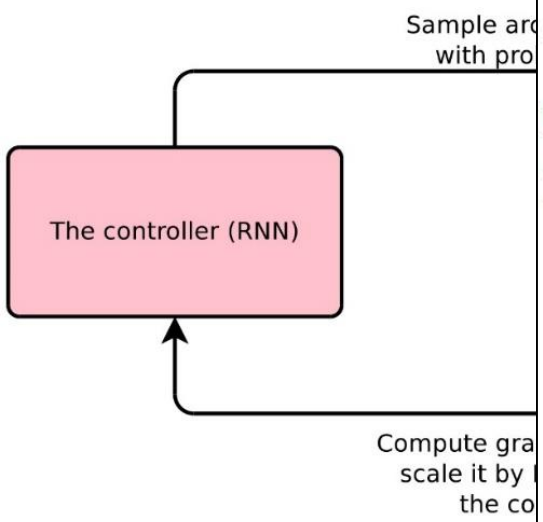


Figure 1: An overview of

[ART Lab] AI는 훗날 '엑셀'이 될 수 있을까

AI 기술 몰라도 AI 제품 만드는 시대...기술 연구보다 메커니즘 이해가 중요

2019.04.19(금) 14:20:30

[비즈한국] "인공지능, 꼭 배워야 할까요?" 이에 대한 대답은 크게 두 가지로 나뉜다. 미래를 주도할 기술이니 꼭 배워야 한다는 입장과 미래엔 마이크로소프트(MS) 프로그램 엑셀(Excel)처럼 잘 활용만 하면 될 테니 굳이 배우지 않아도 된다는 입장이다.

구글의 CEO(최고경영자) 순다 피차이가 'Google Cloud Next 19' 이벤트에서 발표하고 있다.

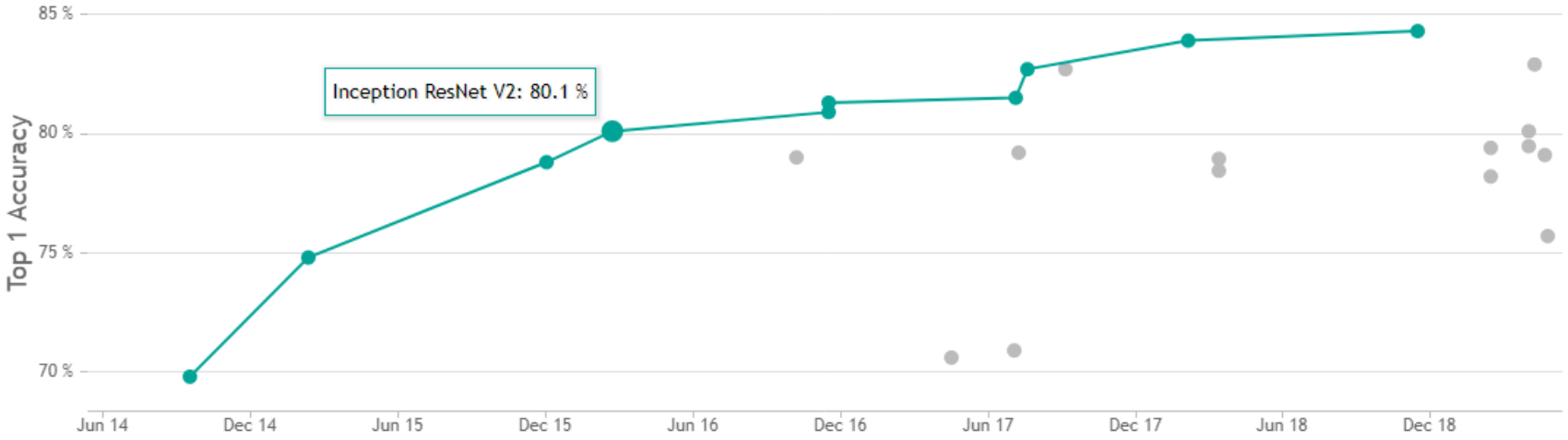
워낙 '핫'한 미래기술이기에 꼭 배워야 한다는 주장만 있을 것 같지만 개발자와 연구자들 사이에선 기술 그 자체를 공부할 필요는 없다는 주장도 심심치 않게 흘러나온다. "엑셀을 이용만 잘

Model Construction	Hyperparameter Optimization	
xnet	skopt	GPFlowOpt
learn	hyperopt	FAR-HO
	simple(x)	Xcessiv
sorFlow	ray.tune	HORD
ORCH	chocolate	ENAS

Online Solution	
to_ml	TPOT
H2O AutoML	auto-sklearn

Zoph, Barret, and Quoc V. Le. "Neural architecture search with reinforcement learning.", 2016

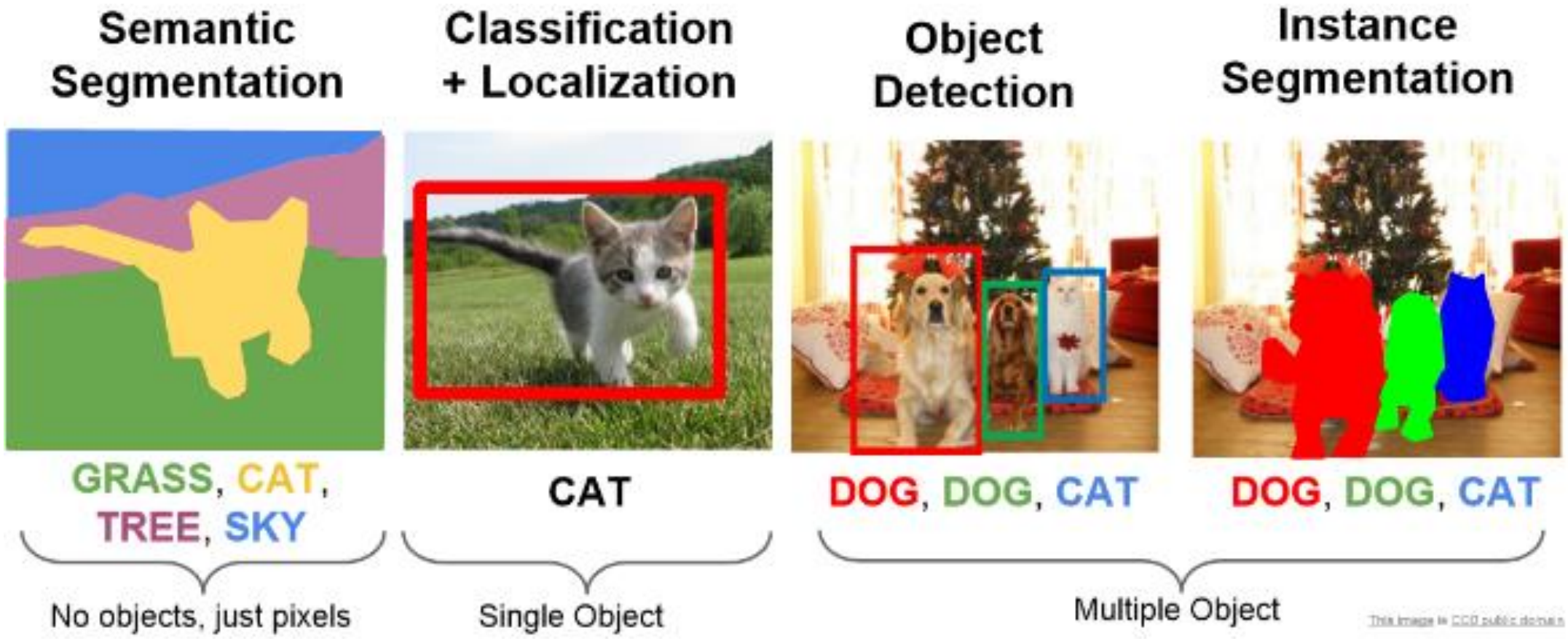
- Computer Vision : Image Classification on ImageNet



Rank	Method	Top 1 Accuracy	Top 5 Accuracy	Error rate	Paper Title	Year	Paper	Code
1	**GPIPE**	84.3%	97%		GPipe: Efficient Training of Giant Neural Networks using Pipeline Parallelism	2018		
2	**AmoebaNet-A**	83.9%			Regularized Evolution for Image Classifier Architecture Search	2018		
3	**Oct-ResNet-152 + SE**	82.9%	96.3%		Drop an Octave: Reducing Spatial Redundancy in Convolutional Neural Networks with Octave Convolution	2019		
4	**NASNET-A(6)**	82.7%	96.2%		Learning Transferable Architectures for Scalable Image Recognition	2017		
5	**SENet**	82.7%	96.2%		Squeeze-and-Excitation Networks	2017		

https://paperswithcode.com/sota/image-classification-on-imagenet

- Computer Vision : Other tasks



http://cs231n.stanford.edu/

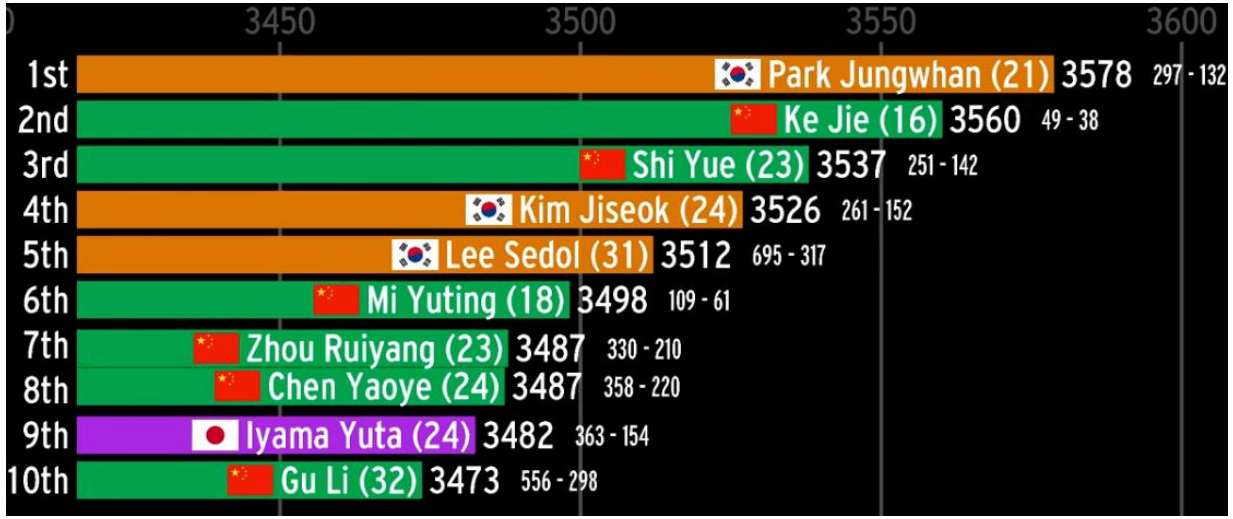
- Video-to-Video Synthesis

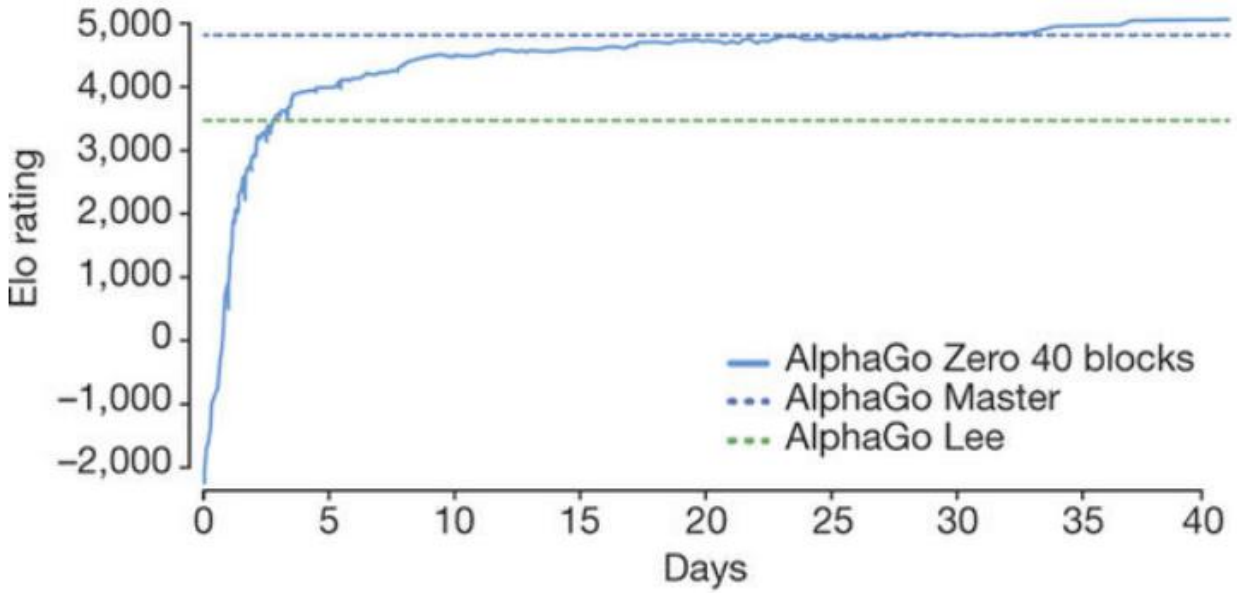
Ting-Chun Wang, Ming-Yu Liu, at el. “Video-to-Video Synthesis”, 2018

Recent Trends in Deep Learning

- Deep Reinforcement Learning : Game Agent
  - AlphaGo(2016) : Beat Top Human at Go

- Deep Reinforcement Learning : Game Agent
  - AlphaGo(2016) : Beat Top Human at Go





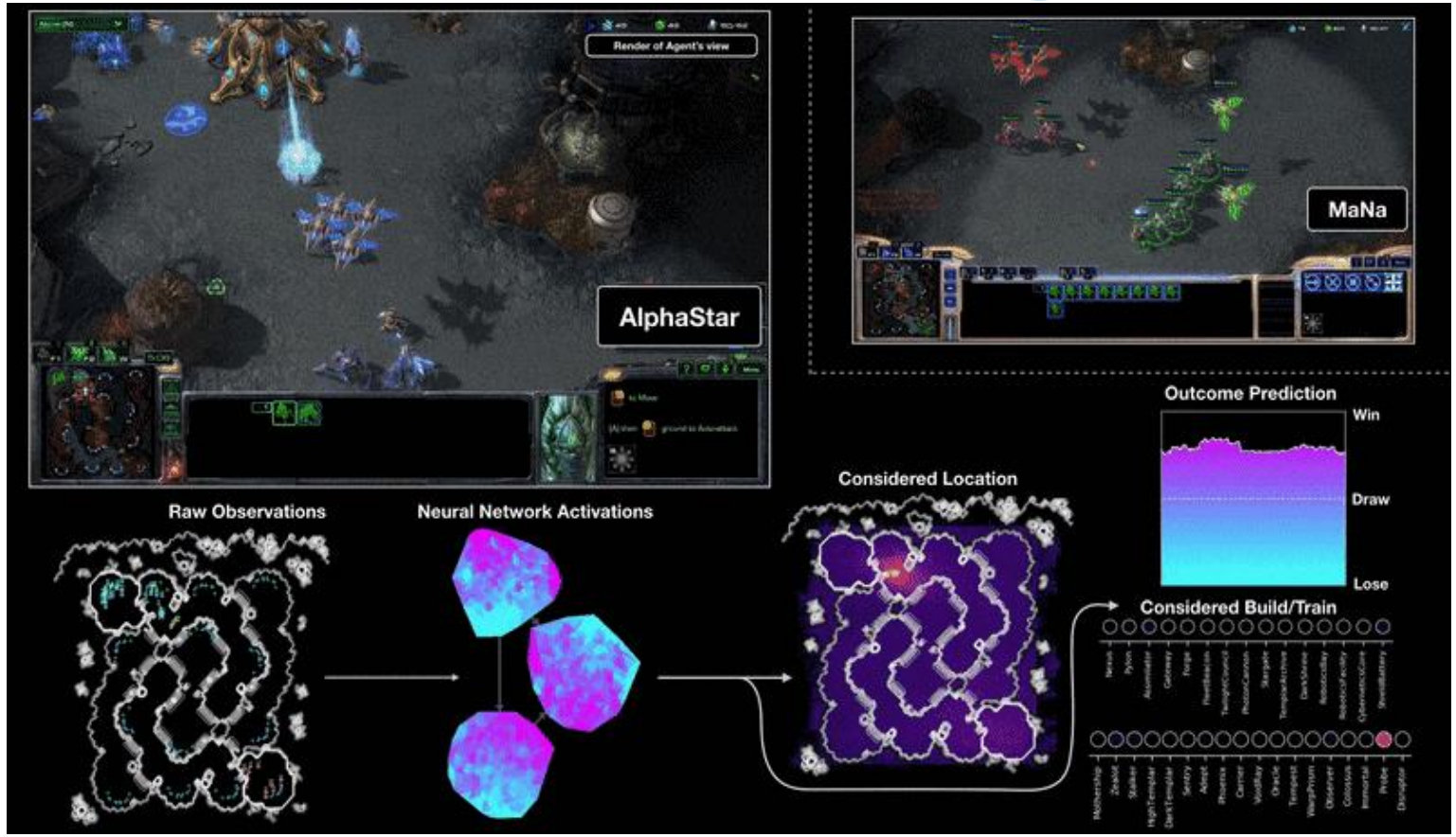
Elo rating

- Deep Reinforcement Learning : Game Agent
  - OpenAI Five(Dota2) : Beat Top Progammer Team(0.002%)

	OPENAI 1V1 BOT	OPENAI FIVE
CPUs	60,000 CPU cores on Azure	128,000 preemptible CPU cores on GCP
GPUs	256 K80 GPUs on Azure	256 P100 GPUs on GCP
Experience collected	~300 years per day	~180 years per day (~900 years per day counting each hero separately)
Size of observation	~3.3 kB	~36.8 kB
Observations per second of gameplay	10	7.5
Batch size	8,388,608 observations	1,048,576 observations
Batches per minute	~20	~60

- Deep Reinforcement Learning : Game Agent
  - AlphaStar(StarCraft II) : Beat Top Progammers

DeepMind



최신기사

딥마인드 '알파스타' 스타크래프트2 프로게이머에 10승 1패

송고시간 | 2019-01-25 16:04

(서울=연합뉴스) 홍지인 기자 = 구글 딥마인드가 만들어 낸 인공지능(AI)이 바둑에 이어 PC 게임에서도 인간을 꺾었다.

딥마인드는 25일 블리자드의 게임 스타크래프트2를 플레이하도록 제작된 AI '알파스타'와 프로게이머 간의 경기 결과를 공개했다.

프로토스를 주종족으로 하는 알파스타는 프로게이머와 총 11경기를 벌여 10승 1패를 거뒀다.
역시 프로토스를 선택해 유일하게 1승을 거둔 프로게이머 '마나'는 "알파스타가 예상외로 매우 사람처럼 플레이해서 놀랐다"며 "게임마다 다른 전략을 쓰는 것도 인상적이었다"고 말했다.

딥마인드는 알파스타의 훈련을 위해 여러 대의 기계가 서로 대결하게 하는 방식을 썼다. 14일 동안의 훈련은 실시간 플레이 기준으로는 약 200년 분량에 해당한다고 딥마인드는 설명했다.

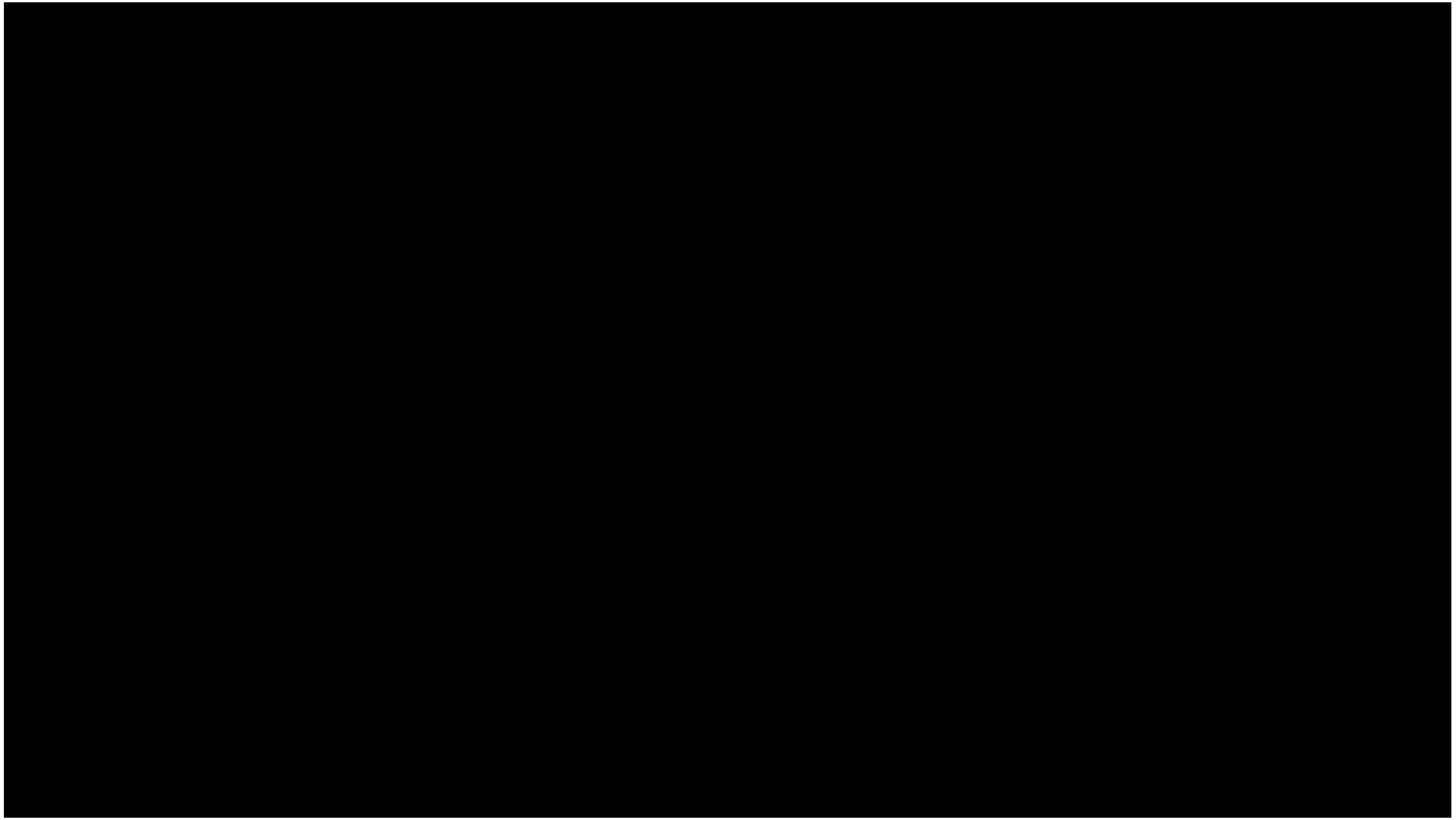
답마인드는 "스타크래프트는 단지 조금 복잡한 게임일 뿐이지만, 알파스타에 적용된 기술은 다른 문제를 푸는데 응용될 수도 있다"며 "훈련 방법 또한 안전하고 강력한 AI를 연구하는 데 쓰일 수 있을 것"이라고 밝혔다.

- BigGAN : A New State of the Art in Image Synthesis

Brock, Andrew, Jeff Donahue, and Karen Simonyan. "Large scale gan training for high fidelity natural image synthesis.“, 2018

Recent Trends in Deep Learning

- Dubbed GauGAN



https://www.youtube.com/watch?v=p5U4NgVGAwg

Reference

- Deep Learning State of the Art(2019) - MIT
- BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2018
- Neural Architecture Search with Reinforcement Learning, 2016
- Video-to-Video Synthesis, 2018
- Large scale gan training for high fidelity natural image synthesis, 2018
- https://openai.com/blog/better-language-models/
- https://paperswithcode.com/sota/image-classification-on-imagenet
- http://cs231n.stanford.edu/
- https://openai.com/blog/openai-five/
- https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/
- https://www.youtube.com/watch?v=Dc0PQlNQhGY
- https://www.youtube.com/watch?v=rNh2CrTFpm4&t=1486s
- https://www.youtube.com/watch?v=odpjk7_tGY0&t=522s
- https://www.shakirm.com/papers/VITutorial.pdf
- https://www.youtube.com/watch?v=KYA-GEhObIs&t=1335s

Thank you!

“Question everything generally thought to be obvious”

- Dieter Rams, 1932 -