Generative Adversarial Nets 논문 볼 때 알면 좋은 것들

2019.01.05

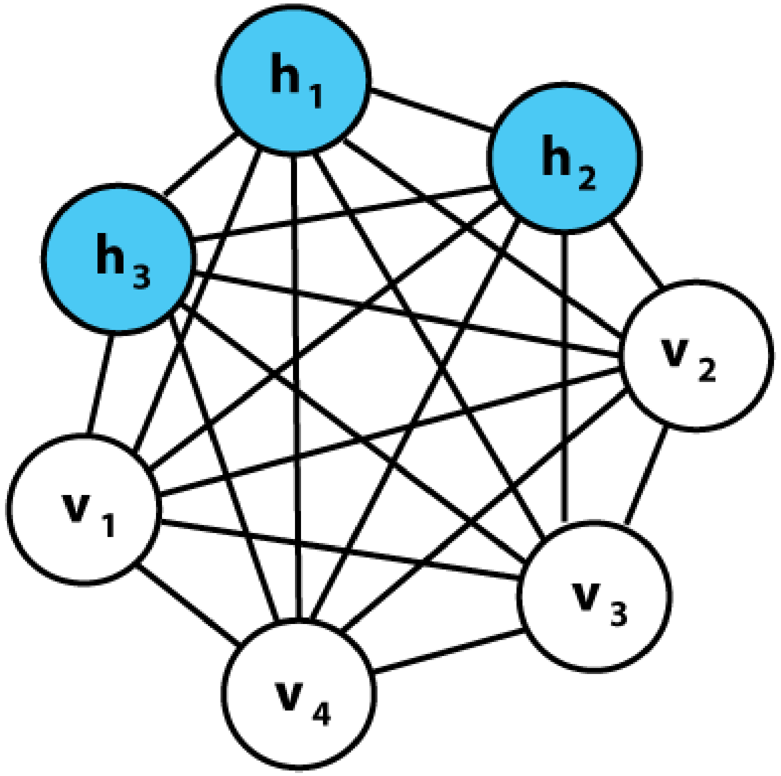
이진호

목차

- 1. Related work에 나오는 모델들
- 2. 정보 이론

Related work에 나오는 모델들

- 0) Boltzmann Machine



http://khanrc.tistory.com/entry/Restricted-Boltzmann-Machine

모든 노드가 연결되어 있는 형태

각 노드들 확률적으로 정의됨

확률이 Boltzmann distribution 따름

$$p(s) = \frac{1}{Z}e^{-E}$$

$$E = -(\sum_{i<j} w_{ij}s_i s_j + \sum_i \theta_i s_i)$$

S : 노드들의 상태

Si : i번째 노드의 값, 0 또는 1

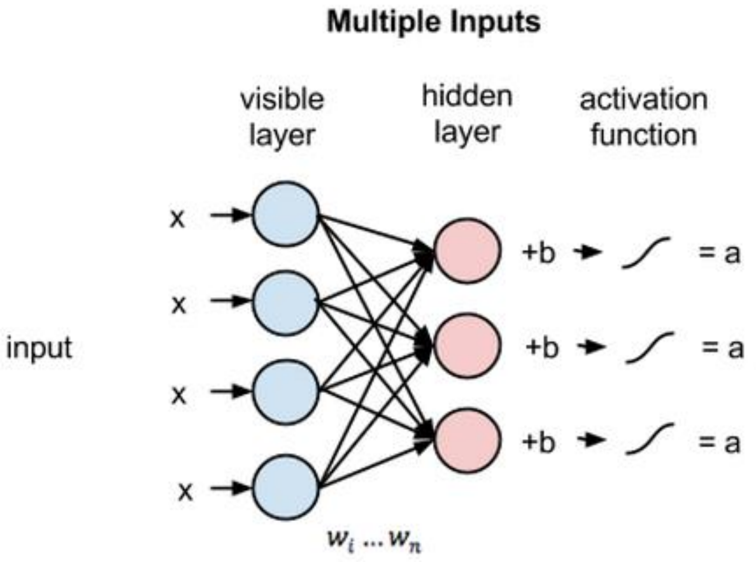
Θi : 노드의 bias

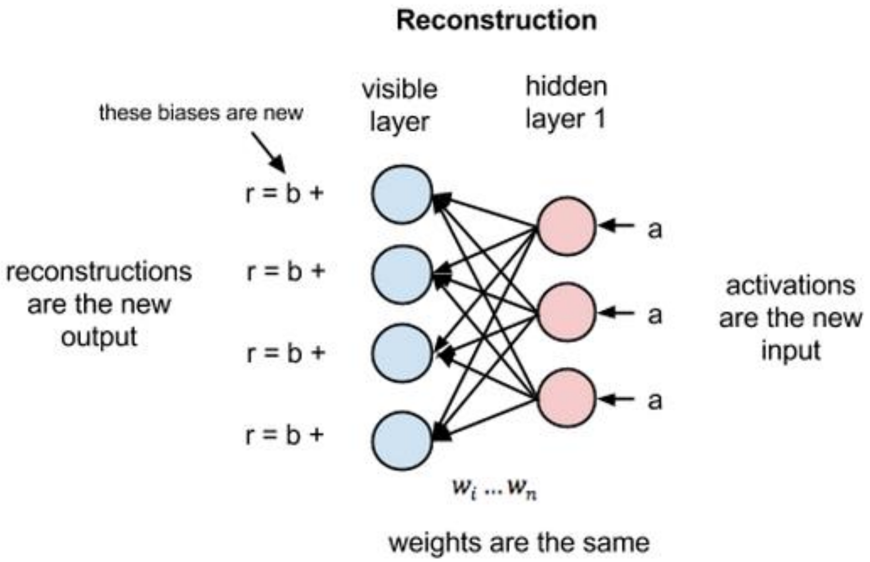
Wij : 노드i와 j를 잇는 엣지의 weight

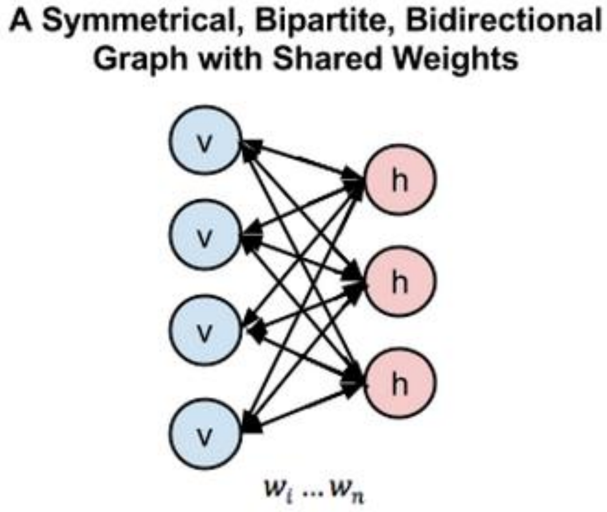
Related work에 나오는 모델들

- 1) Restricted Boltzmann Machine

BM은 모두 연결 되서 dependency 때문에 학습이 어려움 -> 몇가지 제한 추가해 학습 쉽도록



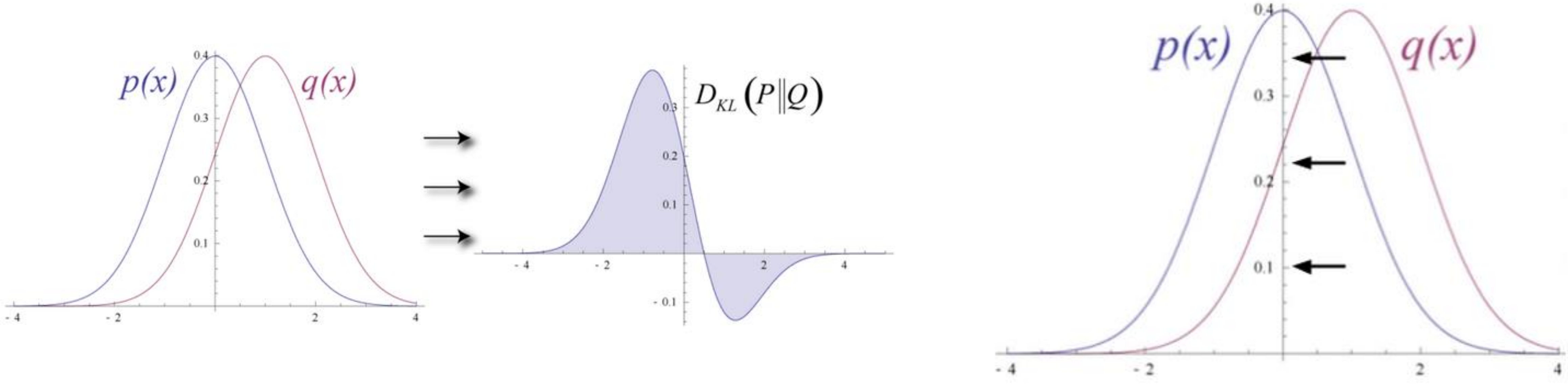




https://skymind.ai/wiki/restricted-boltzmann-machine

Related work에 나오는 모델들

- 1) Restricted Boltzmann Machine



Related work에 나오는 모델들

- 1) Restricted Boltzmann Machine

$$p(v,h;\theta) = \frac{e^{-E(v,h;\theta)}}{Z(\theta)}, \text{ where}$$

$$Z(\theta) = \sum_{v'}\sum_{h'} e^{-E(v',h';\theta)}$$

$$\begin{aligned} l(v;\theta) &= \log p(v) \\ &= \log \sum_h p(v,h) \\ &= \log \frac{\sum_h e^{-E(v,h)}}{Z} \\ &= \log \sum_h e^{-E(v,h)} - \log Z \\ &= \log \sum_h e^{-E(v,h)} - \sum_{v'}\sum_{h'} e^{-E(v',h')} \end{aligned}$$

How to train parameters?
-> maximize the likelihood of the observed data.
To determine the parameters, we perform gradient descent on the log of the likelihood function

https://theclevermachine.wordpress.com/category/mcmc/

Related work에 나오는 모델들

- 1) Restricted Boltzmann Machine

$$\begin{aligned}
\frac{\partial l(v;\theta)}{\partial\theta} &= -\frac{1}{p(v)}\sum_h p(v,h)\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\sum_h \frac{p(v,h)}{p(v)}\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\sum_h p(h|v)\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\mathbb{E}_{p(h|v)}\frac{\partial E(v,h)}{\partial\theta} + \mathbb{E}_{p(v',h')}\frac{\partial E(v',h')}{\partial\theta}.
\end{aligned}$$

Related work에 나오는 모델들

- 1) Restricted Boltzmann Machine

$$
\begin{aligned}
\frac{\partial l(v;\theta)}{\partial\theta} &= -\frac{1}{p(v)}\sum_h p(v,h)\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\sum_h \frac{p(v,h)}{p(v)}\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\sum_h p(h|v)\frac{\partial E(v,h)}{\partial\theta} + \sum_{v'}\sum_{h'} p(v',h')\frac{\partial E(v',h')}{\partial\theta} \\
&= -\mathbb{E}_{p(h|v)}\frac{\partial E(v,h)}{\partial\theta} + \boxed{\mathbb{E}_{p(v',h')}\frac{\partial E(v',h')}{\partial\theta}.}
\end{aligned}
$$

모든 visible unit(input)에 대한
기댓값 계산 사실상 불가능
-> MCMC 사용해 approximation

$$
\frac{\partial l(v;\theta)}{\partial\theta} \approx -\left\langle\frac{\partial E(v,h)}{\partial\theta}\right\rangle_{p(h_{\text{data}}|v_{\text{data}})} + \left\langle\frac{\partial E(v,h)}{\partial\theta}\right\rangle_{p(h_{\text{model}}|v_{\text{model}})}.
$$

Related work에 나오는 모델들

- 2) Deep Belief networks
  - RBM을 빌딩 블록으로 여러층을 쌓은 아키텍쳐
  - Vanishing gradient 문제 해결 하기 위해 아래층(입력에 가까운 층) 부터 위층으로 train을 한다
    - 첫번째 hidden layer pre-train 이후 첫번째 층 weight 고정

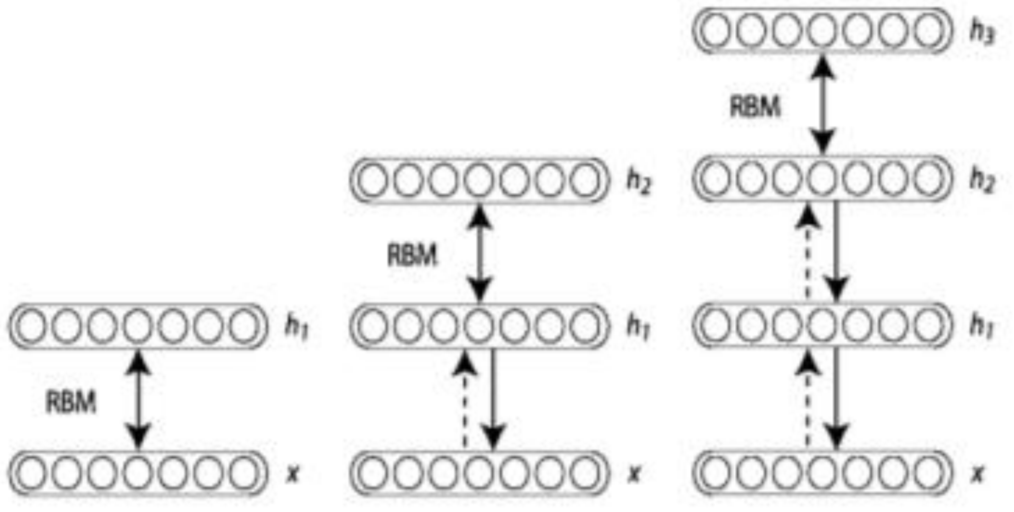


Figure 5.5 층별 선훈련 방법

https://bi.snu.ac.kr/Courses/ML2016/LectureNote/LectureNote_ch5.pdf

Related work에 나오는 모델들

- 2) Deep Belief networks
  - 학습 과정

$$P(x, h^1, \ldots, h^\ell) = \left(\prod_{k=1}^{\ell-2} P(h^k|h^{k-1})\right) P(h^{\ell-1}, h^\ell)$$

Algorithm 2

`TrainUnsupervisedDBN`$(\widehat{p}, \epsilon, L, n, W, b)$

Train a DBN in a purely unsupervised way, with the greedy layer-wise procedure in which each added layer is trained as an RBM by contrastive divergence.

$\widehat{p}$ is the input training distribution for the network
ϵ is a learning rate for the stochastic gradient descent in Contrastive Divergence
L is the number of layers to train
$n = (n^1, \ldots, n^L)$ is the number of hidden units in each layer
W^i is the weight matrix for level i, for i from 1 to L
b^i is the bias vector for level i, for i from 0 to L

- initialize $b^0 = 0$

for $\ell = 1$ to L **do**
- initialize $W^i = 0$, $b^i = 0$

while not stopping criterion **do**
- sample $\mathbf{h}^0 = x$ from $\widehat{p}$

for $k = 1$ to $\ell - 1$ **do**
- sample $\mathbf{h}^k$ from $Q(\mathbf{h}^k|\mathbf{h}^{k-1})$

end for
- `RBMupdate`$(\mathbf{h}^{\ell-1}, \epsilon, W^\ell, b^\ell, b^{\ell-1})$ {thus providing $Q(\mathbf{h}^\ell|\mathbf{h}^{\ell-1})$ for future use}

end while
end for

https://bi.snu.ac.kr/Courses/ML2016/LectureNote/LectureNote_ch5.pdf

Related work에 나오는 모델들

- 2) Deep Belief networks
  - Generative model, Classification model 둘 다로 사용 가능

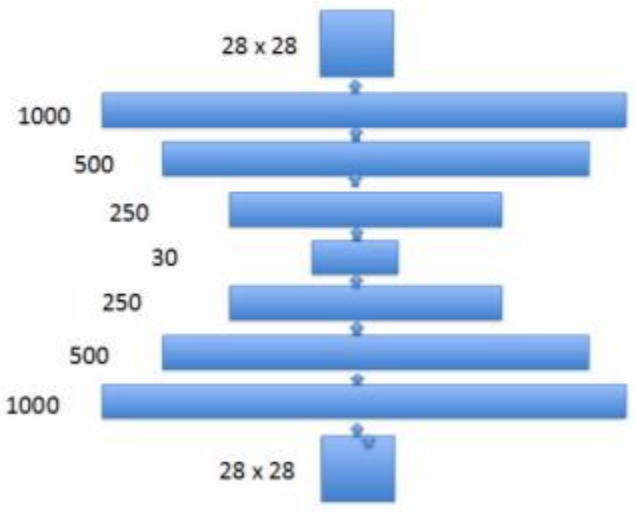
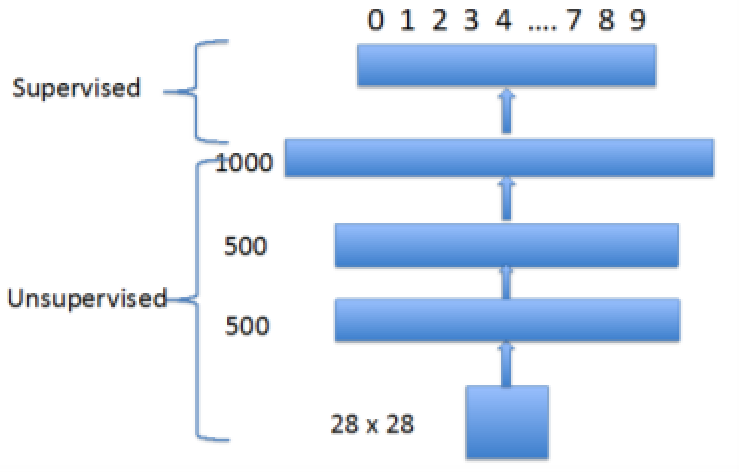


Figure 5.6 딥빌리프네트워크 오토인코더



https://bi.snu.ac.kr/Courses/ML2016/LectureNote/LectureNote_ch5.pdf

Related work에 나오는 모델들

- 2) Deep Belief networks
  - 의의
    - 그 당시 큰 문제였던 overfitting이 잘 일어나지 않았고 MNIST 등에서 좋은 성과를 거둠
    - Deep learning이 주목받는 계기가 됨

https://bi.snu.ac.kr/Courses/ML2016/LectureNote/LectureNote_ch5.pdf

정보 이론

- 1) 정보이론이란?
  - 시그널에 존재하는 정보의 양을 측정하는 응용수학의 한 갈래
  - 핵심 : 잘 일어나지 않는 사건은 자주 발생하는 사건보다 정보량이 많다
    - 1. 자주 발생하는 사건은 낮은 정보량을 가진다. 발생이 보장된 사건은 그 내용에 상관없이 전혀 정보가 없다는 걸 뜻한다.
    - 2. 덜 자주 발생하는 사건은 더 높은 정보량을 가진다.
    - 3. 독립사건(independent event)은 추가적인 정보량(additive information)을 가진다. 예컨대 동전을 던져 앞면이 두번 나오는 사건에 대한 정보량은 동전을 던져 앞면이 한번 나오는 정보량의 두 배이다.

https://www.deeplearningbook.org/contents/prob.html
https://ratsgo.github.io/statistics/2017/09/22/information/

정보 이론

- 2) Self-information
  - Self-information deals only with a single outcome

$$I(x) = -\log P(x).$$

동전을 던져 앞면 나오는 사건

$$-\log_2 0.5 = 1$$

주사위 던져 1이 나오는 사건

$$-\log_2 1/6 = 2.5849$$

정보 이론

- 3) 섀넌 엔트로피(Shannon entropy)
  - the amount of uncertainty in an entire probability distribution
  - expected amount of information in an event drawn from that distribution

$$H(\mathrm{x}) = \mathbb{E}_{\mathrm{x}\sim P}[I(x)] = -\mathbb{E}_{\mathrm{x}\sim P}[\log P(x)],$$

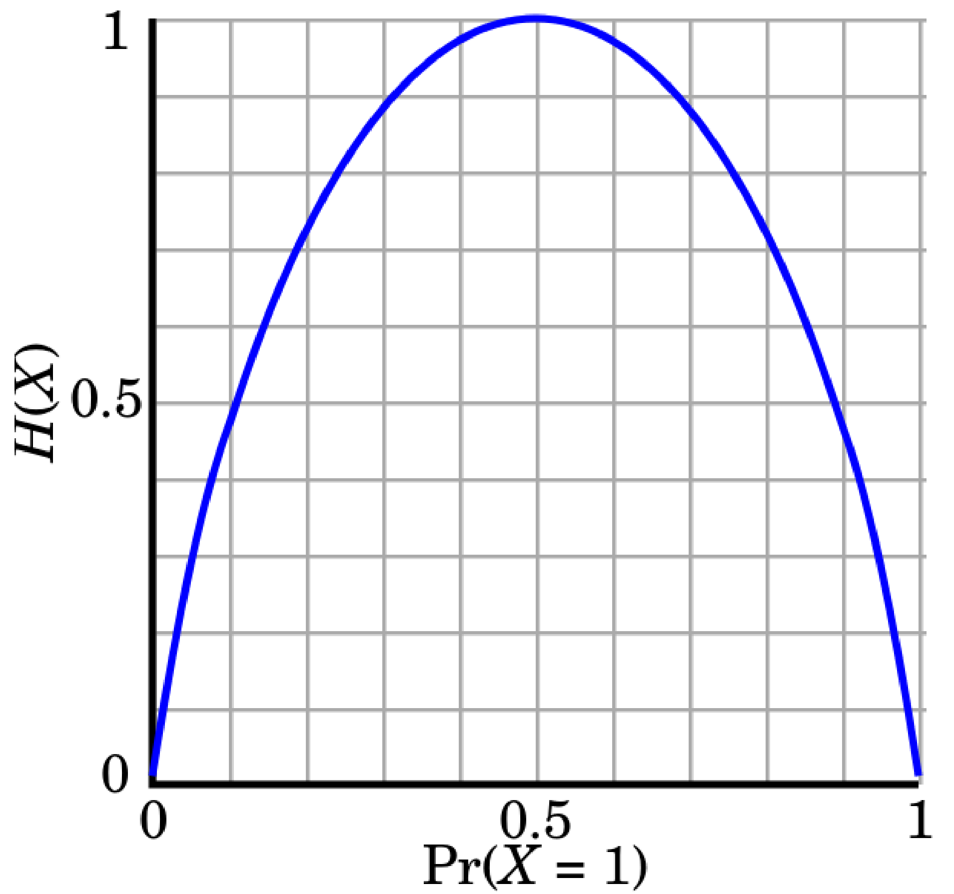
정보 이론

- 3) 섀넌 엔트로피(Shannon entropy)
  - ex) 동전 던지기

$$
\begin{aligned}
H(P) = H(x) &= -\sum_{x} P(x) \log P(x) \\
&= -(0.5 \times \log_2 0.5 + 0.5 \times \log_2 0.5) \\
&= -\log_2 0.5 \\
&= -(-1)
\end{aligned}
$$

정보 이론

- 3) 섀넌 엔트로피(Shannon entropy)
  - 사건이 결정적이면 엔트로피(확률분포의 불확실성)는 낮아짐
  - 반대로 균등적일 수록 엔트로피는 높아짐



정보 이론

- 4) 쿨백-라이블러 발산(Kullback-Leibler divergence, KLD)
  - 같은 random variable x에 대해 다른 확률분포 P, Q가 있을 때 두 확률분포의 차이 나타냄

$$D_{\mathrm{KL}}(P\|Q) = \mathbb{E}_{\mathrm{x}\sim P}\left[\log \frac{P(x)}{Q(x)}\right] = \mathbb{E}_{\mathrm{x}\sim P}\left[\log P(x) - \log Q(x)\right]$$

정보 이론

- 4) 쿨백-라이블러 발산(Kullback-Leibler divergence, KLD)
  - 특징
    - Non negative
    - Discrete에서 P와 Q가 같을 때, continuous에서 거의 다 같을때만 값이 0이 됨
    - Distance measure 아님 왜냐하면 $D_{\mathrm{KL}}(P\|Q) \neq D_{\mathrm{KL}}(Q\|P)$

정보 이론

- 5) 크로스 엔트로피(cross entropy)
  - KLD와 깊은 연관

$$H(P,Q) = H(P) + D_{\mathrm{KL}}(P \| Q),$$

$$H(P,Q) = -E_{X \sim P}[\log Q(x)] = -\sum_x P(x) \log Q(x)$$

  - Q에 관해서 cross entropy 감소시키는 것은 KL divergence 감소시키는 것과 같음 (왜냐면 H(P)는 Q에 관해 고정된 값 -> DKL 감소)

정보 이론

- 5) 크로스 엔트로피(cross entropy)
  - 어쨌든 크로스 엔트로피 최소화는 KLD 최소화와 같은 의미
  - 우리가 가지고 있는 데이터의 분포 P(x)와 모델이 추정한 데이터의 분포 Q(x) 간에 차이를 최소화

정보 이론

- 6) Jensen–Shannon divergence

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M)$$

where $M = \frac{1}{2}(P + Q)$

$$\mathrm{JSD}_{\pi_1,\ldots,\pi_n}(P_1, P_2, \ldots, P_n) = H\left(\sum_{i=1}^{n} \pi_i P_i\right) - \sum_{i=1}^{n} \pi_i H(P_i)$$

$0 \leq \mathrm{JSD}(P \parallel Q) \leq 1$ Log 밑이 2 일때

References

- https://skymind.ai/wiki/restricted-boltzmann-machine
- http://khanrc.tistory.com/entry/Restricted-Boltzmann-Machine
- http://vision.ssu.ac.kr/LecData2015-2/grad_cv/Learning/energy-based%20models%20and%20boltzmann%20machines(ppt).pdf
- https://theclevermachine.wordpress.com/category/mcmc/
- https://bi.snu.ac.kr/Courses/ML2016/LectureNote/LectureNote_ch5.pdf
- https://www.deeplearningbook.org/contents/prob.html
- https://ratsgo.github.io/statistics/2017/09/22/information/