ESC 22 SPRING / WEEK 4

Information Processing

(Chapter 12. Information Capacity Assessment)

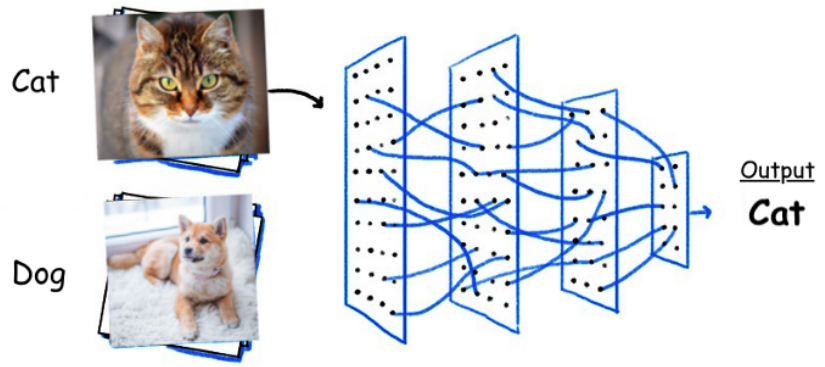
김송희 박태주

Contents

Information Theory (정보이론)

*"How can a neural network go from **raw data** to a **more complex representation** as the data flows through the network layers?"*

ex) pixels → features



➢ can be assessed by some INFORMATION MEASURES

Information measures

1. **Entropy → 12.1**
2. **Cross entropy → 12.6**
3. **Mutual information → 12.7**

12장의 흐름?

➢ In this chapter we shall introduce a measure of assessment of the **compressibility** of a layer using **mutual information**.

Application to Feedforward network

- A feedforward neural network can be interpreted as an "**information compressor**"

Example 12.10.3 An employee has to write a narrative about his new proposed project. However, his busy boss imposes a 2-page limit per project. Therefore, the employee's challenge is to include a lot of information about the project in only a limited amount of space. The available project information, X, has to be compressed in a 2-page narrative, Y, the bottleneck, such that the important project features are not weaken too much.

12.1

Entropy and Properties

12.1 Entropy and Properties

- 사건의 정보량
- 확률변수의 정보량 = 엔트로피

정보량을 표현하는 함수

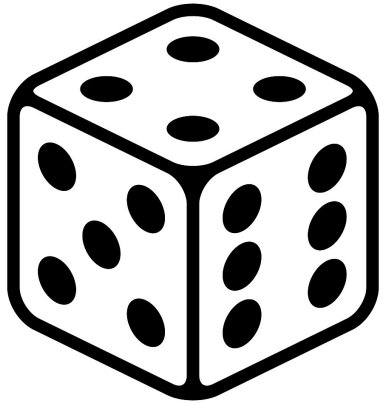
$$I(x) = -\log P(x)$$

(negative log-likelihood)

확률변수 X의 값이 x 인 사건의 정보량

정보량을 표현하는 함수 - 예시

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



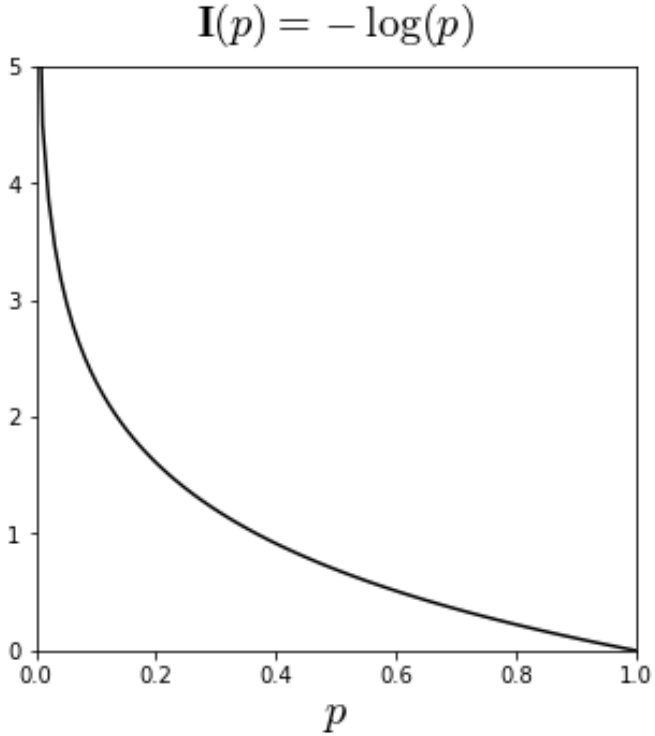
$$-\log_2 1/6 = 2.5849$$

참고)
로그의 밑은 응용 분야에 따라 다르게 쓰임.
밑에 따라 단위 달라짐

$\boldsymbol{log_2}$ - bit (shannon)
$\boldsymbol{log_e}$ - nit (nat)
$\boldsymbol{log_{10}}$ - dit (hartley)

정보량?

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



Point of views (2)

1. 정보량 = surprise의 양 : 내일 해가 뜬다 vs 내일 비가 온다

2. 정보를 저장하려는 입장에서.. 발생 확률이 더 높을 걸 더 짧게, 발생확률이 더 낮은 걸 더 길게 저장하는 게 효율적!

참고) 왜 로그?

1. 확률값에 반비례 해야 하기 때문.

2. 독립적인 두 사건의 정보량을 합치면 각 사건의 정보량을 합친 것과 같아야 함.

Entropy of a random variable

<discrete>

$$H(X) = -\sum_{k=1}^{n} p_k \ln p_k = \mathbb{E}^p[-\ln p], \qquad (12.1.1)$$

"Expected amount of information"

Entropy of a random variable

<differential> $$H(X) = H(p) = -\int_{\mathcal{X}} p(x) \ln p(x)\, dx = \mathbb{E}^p[-\ln p]. \tag{12.1.2}$$

<joint> $$H(X, Y) = -\int_{\mathbb{R}} \int_{\mathbb{R}} p(x, y) \ln p(x, y)\, dxdy.$$

12.6

Conditional Entropy

12.6 Conditional Entropy

- 정의
- Proposition 12.6.1

12.6 Conditional Entropy

H(Y|X) *"the information contained in Y if the variable X is known"*

The *conditional entropy of* Y *given that* $X = x_i$ is defined by

$$H(Y|X = x_i) = -\sum_{j=1}^{M} p(y_j|x_i) \ln p(y_j|x_i),$$

where $p(y_j|x_i) = P(Y = y_j|X = x_i)$. The conditional entropy of Y given X is the weighted average of $H(Y|X = x_i)$, namely,

$$H(Y|X) = \sum_{i=1}^{N} p(x_i) H(Y|X = x_i).$$

Using the joint density relation $p(x_i)p(y_j|x_i) = p(x_i, y_j)$, the aforementioned relation becomes

<discrete>

$$H(Y|X) = -\sum_{i=1}^{N} \sum_{j=1}^{M} p(x_i, y_j) \ln p(y_j|x_i), \qquad (12.6.9)$$

12.6 Conditional Entropy

H(Y|X)

<continuous>

$$H(Y|X) = -\int_{\mathbb{R}}\int_{\mathbb{R}} p(x,y)\ln p(y|x)\,dxdy, \qquad (12.6.10)$$

<multivariate>

$$H(Z|X,Y) = -\iiint p(x,y,z)\ln p(z|x,y)\,dxdydz,$$

Proposition 12.6.1

Proposition 12.6.1 *Let X and Y be two random variables, such that Y depends deterministically on X, i.e., there is a function f such that $Y = f(X)$. Then $H(Y|X) = 0$.*

Proof: First we note that the entropy of a constant is equal to zero, $H(c) = 0$. This follows from the definition of the entropy as

$$H(c) - \sum_i p_i \ln p_i = -1 \ln 1 = 0.$$

The entropy of Y conditioned by the event $\{X = x_i\}$ is

$$H(Y|X = x_i) = H(f(X)|X = x_i) = H(f(x_i)) = 0,$$

by the previous observation. Then

$$H(Y|X) = \sum_i p(x_i) H(Y|X = x_i) = 0.$$

The proof was done for discrete random variables, but with small changes it can also accommodate continuous random variables.

–

12.7

The Mutual Information

12.7 Mutual Information

- 정의
- 기본 성질 + 증명
- Corollary 12.7.2
- Proposition 12.7.3
- Invariance property
- Data processing Inequalities
- Example

12.7 The Mutual Information

I(Y|X) *"amount of information conveyed by X about Y"*

$$I(Y|X) = H(Y) - H(Y|X) \tag{12.7.11}$$

Properties

Proposition 12.7.1 *For any two random variables X and Y defined on the same sample space we have:*

(a) Nonnegativity: $I(X|Y) \geq 0$;

(b) Nondegeneracy: $I(X|Y) = 0 \Leftrightarrow X$ *and* Y *are independent.*

(c) Symmetry: $I(X|Y) = I(Y|X)$.

증명

$$I(X|Y) = I(Y|X).$$

$$\begin{aligned} H(Y) - H(Y|X) &= -\int p(y) \ln p(y)\, dy + \iint p(x,y) \ln p(y|x)\, dxdy \\ &= -\iint p(x,y) \ln p(y)\, dxdy + \iint p(x,y) \ln p(y|x)\, dxdy \\ &= \iint p(x,y) \ln \frac{p(y|x)}{p(y)}\, dxdy = \iint p(x,y) \ln \frac{p(x,y)}{p(x)p(y)}\, dxdy \end{aligned}$$

The symmetry property enables us to write just $I(X,Y)$ instead of $I(X|Y)$ or $I(Y|X)$; we shall call it the *mutual information of* X *and* Y. This means that the amount of information contained in X about Y is the same as the amount of information carried in Y about X.

Corollary 12.7.2

Corollary 12.7.2 *We have the following equivalent definitions for the mutual information:*

$$\begin{aligned} I(X,Y) &= H(Y) - H(Y|X) \\ &= H(X) - H(X|Y) \\ &= D_{KL}[p(x,y)||p(x)p(y)]. \end{aligned}$$

The mutual information can be also seen as the information by which the sum of separate information of X and Y exceeds the joint information of (X,Y).

참고) KL Divergence? ≈ 두 분포 사이의 거리 개념

$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

$$D_{\mathrm{KL}}(P \parallel Q) = \int_{-\infty}^{\infty} p(x) \log\left(\frac{p(x)}{q(x)}\right) dx$$

$$D_{KL}[p(x,y)||p(x)p(y)] = \iint p(x,y) \ln \frac{p(x,y)}{p(x)p(y)} \, dxdy$$

12.7 The Mutual Information

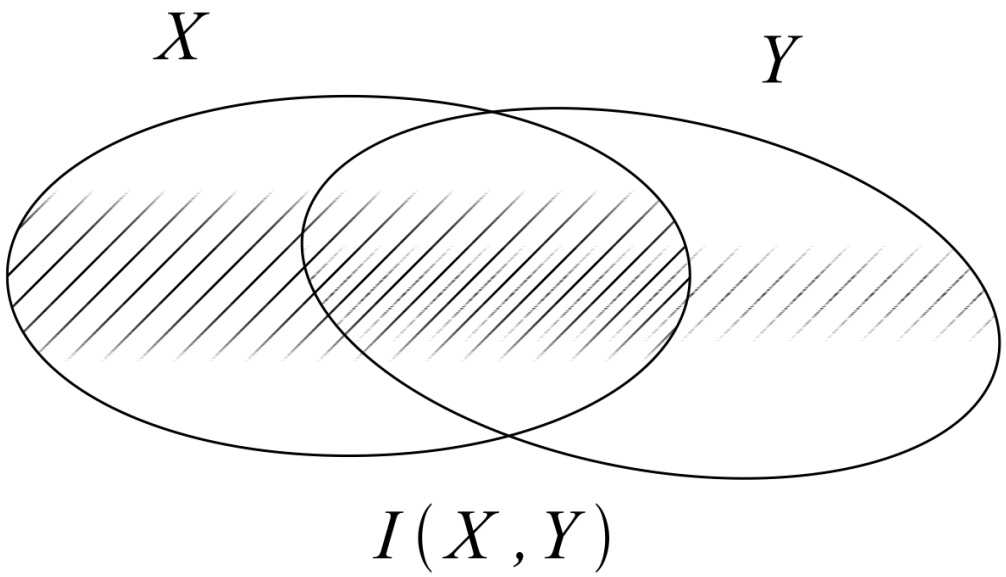


Figure 12.2: *Each random variable is represented by a domain; the area of intersection of domains represents the mutual information* $I(X, Y)$.

12.7 The Mutual Information

Proposition 12.7.3 *The mutual information is given by*

$$I(X,Y) = H(X) + H(Y) - H(X,Y). \tag{12.7.12}$$

Proof: Using that $\ln p(y|x) = \ln \dfrac{p(x,y)}{p(x)} = \ln p(x,y) - \ln p(x)$, we have

$$\begin{aligned} I(X,Y) &= H(Y) - H(Y|X) = H(Y) + \iint p(x,y) \ln p(y|x)\, dxdy \\ &= H(Y) + \iint p(x,y) \ln p(x,y)\, dxdy - \iint p(x,y) \ln p(x)\, dxdy \\ &= H(Y) - H(X,Y) - \int p(x) \ln p(x)\, dx \\ &= H(Y) - H(X,Y) + H(Y). \end{aligned}$$

Since the mutual information is nonnegative, relation (12.7.12) implies that $H(X,Y) \leq H(X) + H(Y)$. ■

Invariance property

Proposition 12.7.5 (Invariance property) *Let X and Y be random variables taking values in $\mathbb{R}^n$. Then for any invertible and differentiable transformations $\phi, \psi : \mathbb{R}^n \to \mathbb{R}^n$, we have*

$$I(X, Y) = I\big(\phi(X), \psi(Y)\big). \tag{12.7.13}$$

Data Processing Inequalities

Proposition 12.7.9 (Data processing inequalities) *For any three random variables* X, Y, *and* Z, *which form a Markov chain,* $X \to Y \to Z$, *we have:*

(a) $I(X,Y) \geq I(X,Z)$;

(b) $I(Y,Z) \geq I(X,Z)$.

Data processing inequalities 증명

Proof: (a) Subtracting the identities

$$I(X,Y) = H(X) - H(X|Y)$$

$$I(X,Z) = H(X) - H(X|Z),$$

and using Lemma 12.7.7, part (b), and then Lemma 12.7.6, yields

$$I(X,Y) - I(X,Z) = H(X|Z) - H(X|Y) = H(X|Z) - H(X|Y,Z) \geq 0.$$

Lemma 12.7.6 *For any three random variables, X, Y, and Z we have*

$$H(X|Y,Z) \leq H(X|Z).$$

Lemma 12.7.7 *For any three random variables, X, Y, and Z that form a Markov chain, $X \to Y \to Z$, we have*

(a) $H(Z|X,Y) = H(Z|Y)$;

(b) $H(X|Y,Z) = H(X|Y)$.

Example

Example 12.7.2 Consider the Markov chain $X^{(0)} \to X^{(1)} \to X^{(2)}$, where $X^{(0)}$ is the identity of a randomly picked card from a usual 52-card pack, $X^{(1)}$ represents the suit of the card (Spades, Hearts, Clubs, or Diamonds), and $X^{(2)}$ is the color of the card (Black or Red), see Fig. 12.4. We may consider $X^{(0)}$ as the input random variable to a one-hidden layer neural network. The hidden layer $X^{(1)}$ is pooling the suit of the card, while the output $X^{(2)}$ is a pooling layer that collects the color of the suit; $X^{(2)}$ can be also considered as the color classifier of a randomly chosen card.

We shall compute the entropies of each layer using their uniform distributions $P(X^{(0)} = x_i) = 1/52$, as well as

$X^{(1)}$	Hearts	Clubs	Spades	Diamonds
$P(s_i)$	1/4	1/4	1/4	1/4

$X^{(2)}$	Red	Black
$P(c_i)$	1/2	1/2

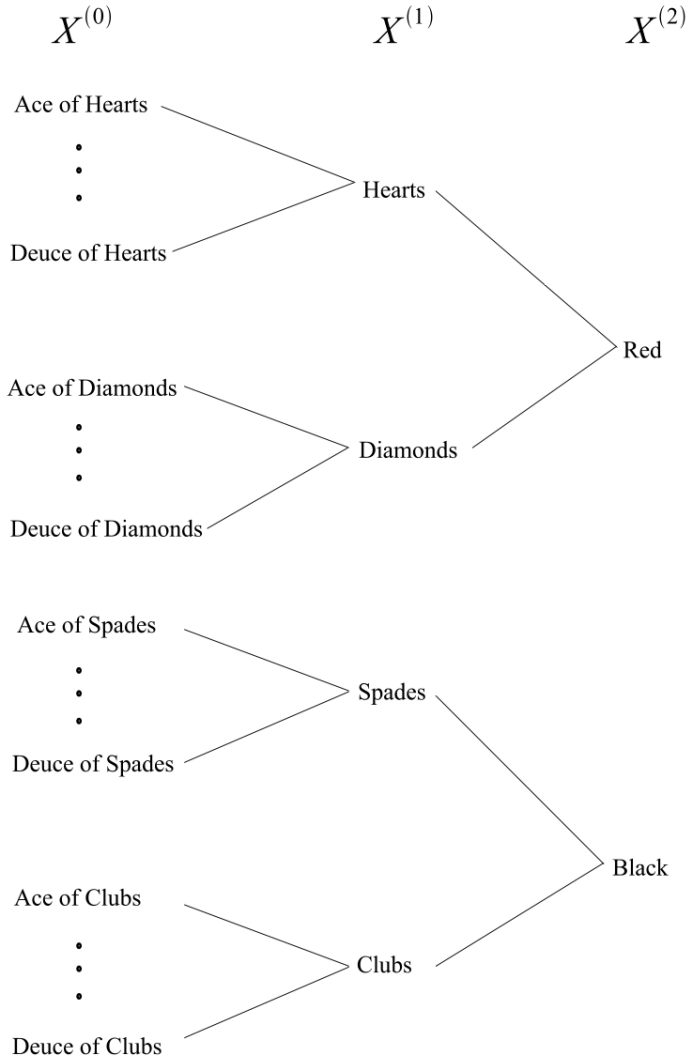


Figure 12.4: *The Markov chain $X^{(0)} \to X^{(1)} \to X^{(2)}$, where $X^{(0)}$ is the identity of a card, $X^{(1)}$ is the suit of a card, and $X^{(2)}$ is the color of a card.*

Example

<entropy> Using the definition of the entropy, we have

$$
\begin{aligned}
H(X^{(0)}) &= -\sum_{i=1}^{52} p(x_i) \ln p(x_i) = -\ln(1/52) = \ln 52 \\
H(X^{(1)}) &= -\sum_{i=1}^{4} p(s_i) \ln p(s_i) = -\ln(1/4) = \ln 4 \\
H(X^{(2)}) &= -\sum_{i=1}^{2} p(c_i) \ln p(c_i) = -\ln(1/2) = \ln 2.
\end{aligned}
$$

<entropy flow> We notice the strictly decreasing flow of entropy

$$H(X^{(0)}) > H(X^{(1)}) > H(X^{(2)}).$$

Example

<conditional entropy>

$$
\begin{aligned}
H(X^{(1)}|X^{(0)}) &= -\sum_{j=1}^{52}\sum_{i=1}^{4} p(x_i^{(1)}, x_j^{(0)}) \ln p(x_i^{(1)}|x_j^{(0)}) \\
&= -\sum_{j=1}^{52}\sum_{i=1}^{4} p(x_i^{(1)}) p(x_i^{(1)}|x_j^{(0)}) \ln p(x_i^{(1)}|x_j^{(0)}) = 0.
\end{aligned}
$$

Example

<mutual information>

$$I(X^{(0)}, X^{(1)}) = H(X^{(1)}) - \underbrace{H(X^{(1)}|X^{(0)})}_{=0} = H(X^{(1)}) = \ln 4$$

$$I(X^{(0)}, X^{(2)}) = H(X^{(2)}) - \underbrace{H(X^{(2)}|X^{(0)})}_{=0} = H(X^{(2)}) = \ln 2.$$

<data processing inequality>

It follows that the following data processing inequality

$$I(X^{(0)}, X^{(1)}) \geq I(X^{(0)}, X^{(2)})$$

is verified strictly.

3 Mutual Information 알고 가자!

$$I(X, Y) \quad I(Y, Z)$$

$$I(X, Z)$$

X - input
Y - output
Z - target

12.8

Application to Deep Neural Network

12.8 Application to Deep Neural Network

앞에서 배운 Entropy 와 information 개념을 DNN에 적용

1. Entropy flow
2. Compressionless
3. Total Compression

Entropy flow

Proposition 12.8.1 *If the layer activations of a feedforward neural network are discrete random variables, then the entropy flow is decreasing*

$$H(X^{(0)}) \geq H(X^{(1)}) \geq \cdots \geq H(X^{(L)}) \geq 0.$$

Proof: Since the activation of the ℓth layer depends deterministically on the activation of the previous layer, $X^{(\ell)} = F(X^{(\ell-1)})$, by Proposition 12.6.1 we have $H(X^{(\ell)}|X^{(\ell-1)}) = 0$.

Entropy flow

$$\begin{aligned} I(X,Y) &= H(Y) - H(Y|X) \\ &= H(X) - H(X|Y) \end{aligned}$$

$I(X^{(\ell)}, X^{(\ell-1)})$ in two different ways:

$$\begin{aligned} I(X^{(\ell)}, X^{(\ell-1)}) &= H(X^{(\ell)}) - H(X^{(\ell)}|X^{(\ell-1)}) = H(X^{(\ell)}) \\ I(X^{(\ell)}, X^{(\ell-1)}) &= H(X^{(\ell-1)}) - H(X^{(\ell-1)}|X^{(\ell)}) \leq H(X^{(\ell-1)}), \end{aligned}$$

since $H(X^{(\ell-1)}|X^{(\ell)}) \geq 0$, as the variables are discrete. From the last two formulas we infer $H(X^{(\ell)}) \leq H(X^{(\ell-1)})$. The equality holds for the case when F is invertible, since

$$H(X^{(\ell-1)}|X^{(\ell)}) = H(F^{-1}(X^{(\ell)})|X^{(\ell)}) = 0,$$

Entropy의 양은 커질 수 없고, 줄어든다.

12.8.4 Compressionless layers

$$I(X^{(0)}, X^{(\ell-1)}) = I(X^{(0)}, X^{(\ell)}),$$

i.e., the input $X^{(0)}$ conveys the same information about both $X^{(\ell-1)}$ and $X^{(\ell)}$.

Remark 12.8.4 In the absence of noise, both $X^{(\ell-1)}$ and $X^{(\ell)}$ depend deterministically on $X^{(0)}$ and in this case we have

$$\begin{aligned} I(X^{(0)}, X^{(\ell-1)}) &= H(X^{(\ell-1)}) - H(X^{(\ell-1)}|X^{(0)}) = H(X^{(\ell-1)}), \\ I(X^{(0)}, X^{(\ell)}) &= H(X^{(\ell)}) - H(X^{(\ell)}|X^{(0)}) = H(X^{(\ell)}), \end{aligned}$$

12.8.6 Total compression

The *compression factor* of the ℓth layer of a feedforward neural network with noisy layers is defined by the ratio of the following mutual information:

$$\rho_\ell = \frac{I(X^{(0)}, X^{(\ell)})}{I(X^{(0)}, X^{(\ell-1)})}.$$

$I(X^{(0)}, X^{(\ell-1)}) \geq I(X^{(0)}, X^{(\ell)})$. This fact implies $0 \leq \rho_\ell \leq 1$.

12.8.6 Total compression

$$\begin{aligned}\rho_1\rho_2\cdots\rho_L &= \frac{I(X^{(0)},X^{(1)})}{I(X^{(0)},X^{(0)})}\frac{I(X^{(0)},X^{(2)})}{I(X^{(0)},X^{(1)})}\cdots\frac{I(X^{(0)},X^{(L)})}{I(X^{(0)},X^{(L-1)})}\\ &= \frac{I(X^{(0)},X^{(L)})}{I(X^{(0)},X^{(0)})} = \frac{I(X^{(0)},X^{(L)})}{H(X^{(0)})},\end{aligned}$$

Remark 12.8.7 We make two remarks, which follow easily.
(i) The total compression $\rho = 1$ (no compression) if and only if all layers are compressionless.
(ii) The total compression $\rho = 0$ if and only if there is a layer independent from the input X^0.

12.8 Summary

- Entropy flow, compressionless layers, total compression

→ Feedforward Neural Network 에서 information이 어떻게 압축(compression)되는지 알아봄.

- → 그렇다면 'How large?', 정보가 얼마만큼 압축되어야 하는지에 대해 12.9에서 알아보자!

12.9

Network Capacity

12.9 Network Capacity

- Maximum amount of information a network can process
- high capacity → / low capacity →

- 1. 어떤 종류?
- 2. 존재 여부?
- 3. 구하는 방법?

12.9.1 Types of Capacity

There are three mutual information of interest, $I(X,Y)$, $I(Y,Z)$, and $I(X,Z)$

1. The first one, $I(X,Y)$, represents the amount of information contained in X about Y, or, equivalently, the amount of information processed by the network. This depends on the input distribution $p(x)$

network capacity $$C(W,B) = \max_{p(x)} I(X,Y).$$

This represents the maximum amount of information a network can process

12.9.5 The input-output matrix

If $d^{(0)} = d^{(L)} = 1$, then the neural network is characterized by the following *input-output matrix* $q_{ij} = p(y_j|x_i)$, where

$$p(y_j|x_i) = P(Y = y_j|X = x_i), \quad 1 \le i \le N, 1 \le j \le M.$$

$$\sum_{j=1}^{M} q_{ij} = 1$$

$$p(y_j) = \sum_{i=1}^{N} p(x_i)p(y_j|x_i),$$

행렬 : $p(\mathbf{y}) = Q^T p(\mathbf{x})$

12.9.6 The existence of network capacity

The fact that the definition of the network capacity makes sense reduces to the existence of the maximum of the mutual information $I(X, Y)$ under

Facts

1. Bounded and closed set ⇔ Compact set (In Euclidean space)
2. Any continuous function on a compact set reaches its maxima on that set

$$
\begin{aligned}
I(X,Y) &= H(Y) - H(Y|X) \\
&= -\sum_{j=1}^{m} p(y_j) \ln p(y_j) + \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i, y_j) \ln p(y_j|x_i) \\
&= -\sum_{j=1}^{m}\sum_{i=1}^{n} p(x_i)p(y_j|x_i) \ln p(y_j) \\
&\quad + \sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i)p(y_j|x_i) \ln p(y_j|x_i) \\
&= \sum_{j=1}^{m}\sum_{i=1}^{n} p(x_i)p(y_j|x_i)\Big(\ln p(y_j|x_i) - \ln p(y_j) \Big) \\
&= \sum_{j=1}^{m}\sum_{i=1}^{n} p(x_i)p(y_j|x_i)\Big(\ln p(y_j|x_i) - \ln \sum_{i=1}^{n} p(x_i)p(y_j|x_i) \Big) \\
&= \sum_{j=1}^{m}\sum_{i=1}^{n} p(x_i)q_{ij}\Big(\ln q_{ij} - \ln \sum_{i=1}^{n} p(x_i)q_{ij} \Big).
\end{aligned}
$$

$p(y_j) = \sum_{i=1}^{N} p(x_i)p(y_j|x_i)$,

$p(x_i)p(y_j|x_i)$ 로 묶기

$p(y_j) = \sum_{i=1}^{N} p(x_i)p(y_j|x_i)$,

$q_{ij} = p(y_j|x_i)$

12.9.6 The existence of network capacity

It follows that for a given input-output matrix, q_{ij}, the mutual information $I(X,Y)$ is a continuous function of n real numbers, $p(x_1),\ldots,p(x_n)$, which belong to the domain

$$K = \{(p_1,\ldots,p_n); p_i \geq 0, \sum_{i=1}^{n} p_i = 1\}.$$

1. K is bounded and closed → K is a compact set

2. I(X,Y) is a continuous function of p(i) on K

3. There is an p(i*) for which I(X,Y) is maximum on K

12.9.7 The Lagrange multiplier method

라그랑주 승수법

29개 언어

위키백과, 우리 모두의 백과사전.

라그랑주 승수법(Lagrange乘數法, 영어: Lagrange multiplier method)은 제약이 있는 최적화 문제를 푸는 방법이다. 최적화하려 하는 값에 형식적인 **라그랑주 승수**(Lagrange乘數, 영어: Lagrange multiplier) 항을 더하여, 제약된 문제를 제약이 없는 문제로 바꾼다. 조제프루이 라그랑주가 도입하였다. 수학, 라그랑주 역학, 경제학, 운용 과학 등에 쓰인다.

정의 [편집]

연속미분가능함수 $f\colon \mathbb{R}^D \to \mathbb{R}$와 $\mathbf{g}\colon \mathbb{R}^D \to \mathbb{R}^C$를 생각하자. $\mathbf{g}(\mathbf{x}) = 0$인 제약 아래 $f(\mathbf{x})$를 최적화하는 문제를 생각하자. 이 문제는 **라그랑주 승수법**을 써 다음과 같이 풀 수 있다. 다음과 같은 함수 $F\colon \mathbb{R}^{D+C} \to \mathbb{R}$을 정의하자.

$$F(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) + \mathbf{y} \cdot \mathbf{g}(\mathbf{x})$$

$$C(W, B) = \max_{p(x)} I(X, Y).$$

constraint : $\sum_{i=1}^{n} p_i = 1$

I(X,Y)의 최대값을 구하기 위해,
정의역의 constrain로 라그랑주
승수법을 이용해 새로운 함수 F(X)를
정의하고 F(p)의 최대값을 구한다.

12.9.7 The Lagrange multiplier method

$$C(W,B) = \max_{p(x)} I(X,Y). \qquad \text{constraint : } \sum_{i=1}^{n} p_i = 1$$

$$F(p_1,\ldots,p_n) = I(X,Y) + \lambda\Big(\sum_{i=1}^{n} p_i - 1\Big).$$

$$\frac{\partial F}{\partial p_k} = \frac{\partial}{\partial p_k}\Big[I(X,Y) + \lambda\Big(\sum_{i=1}^{n} p_i - 1\Big)\Big]$$

$$= \frac{\partial H(Y)}{\partial p_k} - \frac{\partial H(Y|X)}{\partial p_k} + \lambda$$

ㄱ ㄴ

$$\frac{\partial p(y_j)}{\partial p_k} = \frac{\partial p(y_j)}{\partial p(x_k)} = p(y_j|x_k) = q_{kj}$$

$$\frac{\partial H(Y)}{\partial p(y_j)} = -\frac{\partial}{\partial p(y_j)} \sum_{r=1}^{m} p(y_r) \ln p(y_r) = -(1 + \ln p(y_j)),$$

$p(y_j) = \sum_{i=1}^{N} p(x_i) p(y_j|x_i)$

$\frac{\partial}{\partial p_k} p(y_j|x_i) = 0$

then chain rule implies

$$\frac{\partial H(Y)}{\partial p_k} = \sum_{j=1}^{m} \frac{\partial H}{\partial p(y_j)} \frac{\partial p(y_j)}{\partial p_k} = -\sum_{j=1}^{m} (1 + \ln p(y_j)) q_{kj}$$

$$= -1 - \sum_{j=1}^{m} \ln p(y_j) q_{kj},$$

$\sum_{j=1}^{M} q_{ij} = 1$

$$\frac{\partial H(Y|X)}{\partial p_k} = -\frac{\partial}{\partial p_k}\sum_{i=1}^{n}\sum_{j=1}^{m} p(x_i)p(y_j|x_i)\ln p(y_j|x_i)$$

$$= -\frac{\partial}{\partial p_k}\sum_{i=1}^{n}\sum_{j=1}^{m} p_i q_{ij}\ln q_{ij} = -\sum_{j=1}^{m} q_{kj}\ln q_{kj}.$$

$$\frac{\partial F}{\partial p_k} = \frac{\partial}{\partial p_k}\Big[I(X,Y) + \lambda\Big(\sum_{i=1}^{n} p_i - 1\Big)\Big]$$

$$= \frac{\partial H(Y)}{\partial p_k} - \frac{\partial H(Y|X)}{\partial p_k} + \lambda$$

$$= -1 - \sum_{j=1}^{m}\ln p(y_j)q_{kj} + \sum_{j=1}^{m} q_{kj}\ln q_{kj} + \lambda.$$

$$\cdot\frac{\partial}{\partial p_k} p(y_j|x_i) = 0$$

$$1 - \lambda + \sum_{j=1}^{m} q_{kj} \ln p(y_j) = \sum_{j=1}^{m} q_{kj} \ln q_{kj}, \quad 1 \le k \le n. \qquad (12.9.15)$$

$\sum_{j=1}^{M} q_{ij} = 1$

$$\sum_{j=1}^{m} q_{kj} \left(1 - \lambda + \ln p(y_j)\right) = \sum_{j=1}^{m} q_{kj} \ln q_{kj}, \quad 1 \le k \le n.$$

$\ln p(\mathbf{y})^T = \left(\ln p(y_1), \ldots, \ln p(y_m)\right)$ m x 1 matrix

$h^T = (h_1, \ldots, h_n)$, where $h_k = \sum_{j=1}^{m} q_{kj} \ln q_{kj}$. n x 1 matrix

$$Q(1 - \lambda + \ln p(\mathbf{y})) = h. \qquad (12.9.16)$$

m개를 더해서 만들어진 n개의 식을 matrix transformation으로 표현

Q = (n x m)

12.9.7 The Lagrange multiplier method

$$Q(1 - \lambda + \ln p(\mathbf{y})) = h. \qquad (12.9.16)$$

$$Q^T Q(1 - \lambda + \ln p(\mathbf{y})) = Q^T h.$$

However, if $M < N$, then Q is not a square matrix, and then it does not make sense to consider its determinant. In this case it is useful to assume the maximal rank condition, $\text{rank}(Q) = \text{rank}(Q^T) = M$. Under this condition, there is at most one solution $p(\mathbf{x})$ for the aforementioned equation, see

rank(Q) <= min(n,m) = m

rank(Q) = rank(Q.T) = rank(Q.T@Q) = m
Q.T@Q is (mxm) matrix and rank is m
full rank ⇔ invertible

12.9.7 The Lagrange multiplier method

$$1 - \lambda + \ln p(\mathbf{y}) = (Q^T Q)^{-1} Q^T h.$$

Using the Moore-Penrose pseudoinverse, $Q^+ = (Q^T Q)^{-1} Q^T$

$$1 - \lambda + \ln p(\mathbf{y}) = Q^+ h.$$

$$1 - \lambda + \ln p(y_j) = (Q^+ h)_j, \qquad 1 \le j \le m,$$

$$e^{1-\lambda} p(y_j) = e^{(Q^+ h)_j}, \qquad 1 \le j \le m. \quad (12.9.17)$$

$$(Q^+ h)_j = \sum_{k=1}^{n} q_{jk}^+ h_k,$$

Take exponential

12.9.7 The Lagrange multiplier method

$$e^{1-\lambda} p(y_j) = e^{(Q^+ h)_j}, \qquad 1 \le j \le m. \tag{12.9.17}$$

$$1 - \lambda = \ln\left(\sum_{j=1}^{m} e^{(Q^+ h)_j}\right). \tag{12.9.18}$$

$$p(y_j) = e^{(Q^+ h)_j} / \left(\sum_{k=1}^{m} e^{Q^+ h_k}\right) = \frac{e^{Q^+ h}}{\|e^{Q^+ h}\|_1}.$$

$$p(\mathbf{y}) = \operatorname{softmax}(Q^+ h), \tag{12.9.19}$$

1. Summing over j

2. Take log function

1. Substituting (12.9.18) back into (12.9.17)

2. Solving for p(y)

By softmax definition

12.9.7 The Lagrange multiplier method

$$F(p_1, \ldots, p_n) = I(X, Y) + \lambda\Big(\sum_{i=1}^{n} p_i - 1\Big).$$

$$p(\mathbf{y}) = \text{softmax}(Q^{+}h), \qquad (12.9.19)$$

F(p)의 최대값의 조건에 맞는 p(y)를 찾았고, 이를 이용해 최대값 조건에 맞는 p(x)를 구한다.

Our final goal was to find the input distribution, $p(\mathbf{x})$, which satisfies

$$Q^T p(\mathbf{x}) = p(\mathbf{y}). \qquad (12.9.20)$$

If this equation has a solution $p^* = p(\mathbf{x})$, then by Exercise 12.13.4, it is unique. Furthermore, if all $p_i^* > 0$, for all $1 \leq i \leq n$, then this solution is in the interior of the definition domain and hence it is the point where the functional F achieves its maximum.

12.9.8 Finding the Capacity

Assume we have succeed in finding a maximum point p^* for $F(p)$.

F(p)의 최대값 조건에 맞는 p*를 구하고, 이를 우리가 알고 싶어하는
I(X,Y)식에 대입해 I(X,Y)의 최대값을 구한다.

Network Capacity = $C(W, B) = \max_{p(x)} I(X, Y).$

Lagrange multiplier method 에서 구한
우리가 아는 식으로 표현 가능

12.9.8 Finding the Capacity

$$1-\lambda+\sum_{j=1}^{m} q_{kj}\ln p(y_j)=\sum_{j=1}^{m} q_{kj}\ln q_{kj},\quad 1\le k\le n. \tag{12.9.15}$$

multiply p_k

$$(1-\lambda)p_k=\sum_{j=1}^{m} p_k q_{kj}\ln q_{kj}-\sum_{j=1}^{m} p_k q_{kj}\ln p(y_j)$$

Summing over k

$$\begin{aligned}
1-\lambda &= \sum_{k=1}^{n}\sum_{j=1}^{m} p_k q_{kj}\ln q_{kj}-\sum_{k=1}^{n}\sum_{j=1}^{m} p_k q_{kj}\ln p(y_j)\\
&= \sum_{k=1}^{n}\sum_{j=1}^{m} p(x_k,y_j)\ln p(y_j|x_k)-\sum_{j=1}^{m} p(y_j)\ln p(y_j)\\
&= -H(Y|X)+H(Y)=I(X,Y).
\end{aligned}$$

$$q_{ij}=p(y_j|x_i)$$

$$p(y_j)=\sum_{i=1}^{N} p(x_i)p(y_j|x_i).$$

12.9.8 Finding the Capacity

$$I(X,Y) = 1-\lambda$$
$$1-\lambda = \ln\Big(\sum_{j=1}^{m} e^{(Q\,|\,h)_j}\Big). \qquad (12.9.18)$$

$$C(W,B) = \max_{p(x)} I(X,Y). = 1-\lambda = \ln\Big(\sum_{j=1}^{m} e^{(Q^{+}h)_j}\Big), \quad Q^{+} = (Q^{T}Q)^{-1}Q^{T}$$

where Q^{+} is the Moore-Penrose pseudoinverse of the input-output matrix Q and $h_j = \sum_{r=1}^{m} q_{jr} \ln q_{jr}$. It is worth noting that the capacity depends only on the input-output matrix $Q = q_{ij}$.

12.10 The Information Bottleneck

High network capacity → overfit

Low network capacity → underfit

➔ What is the optimal capacity???

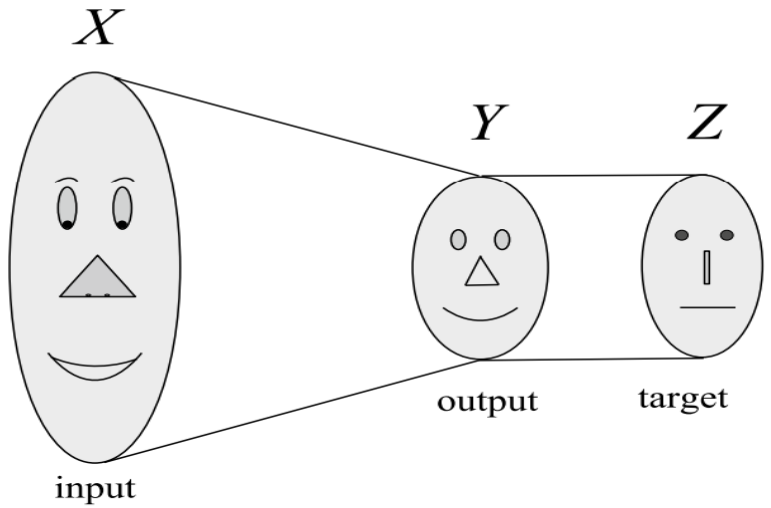


Figure 12.7: *Y is a compressed version of the input X, which preserves meaningful information about target Z. Y preserves the important features of the face given by X, which are meaningful for the idea of face required by the target Z.*

12.10 The Information Bottleneck

Since lossy compression of X into Y cannot convey more information about Z than the initial data X, we have the inequality

$$I(Y, Z) \leq I(X, Z).$$

Namely, the information conveyed about the target Z by the output Y does not exceed the information conveyed about Z by the initial data X. However, even if it is less than $I(X, Z)$, the information $I(Y, Z)$ should still be large enough such that Y still contains enough meaningful information about Z. Thus, we are looking for a network that keeps a fixed amount of meaningful information $I(Y, Z)$ about the target Z, while maximizing the compression of the input X into Y, i.e., minimizing the information $I(X, Y)$.

I(Y,Z)를 가능한 크게 유지하면서, I(X,Y)를 최소화한다.

12.10 The Information Bottleneck

The *bottleneck principle* states that we need to pass the information provided by X about Z through a "bottleneck" formed by the output variable Y, in the most efficient way. This means to minimize $I(X,Y)$ subject to a given

$$\mathcal{L}(p(y|x)) = I(X,Y) - \beta I(Y,Z), \qquad (12.10.24)$$

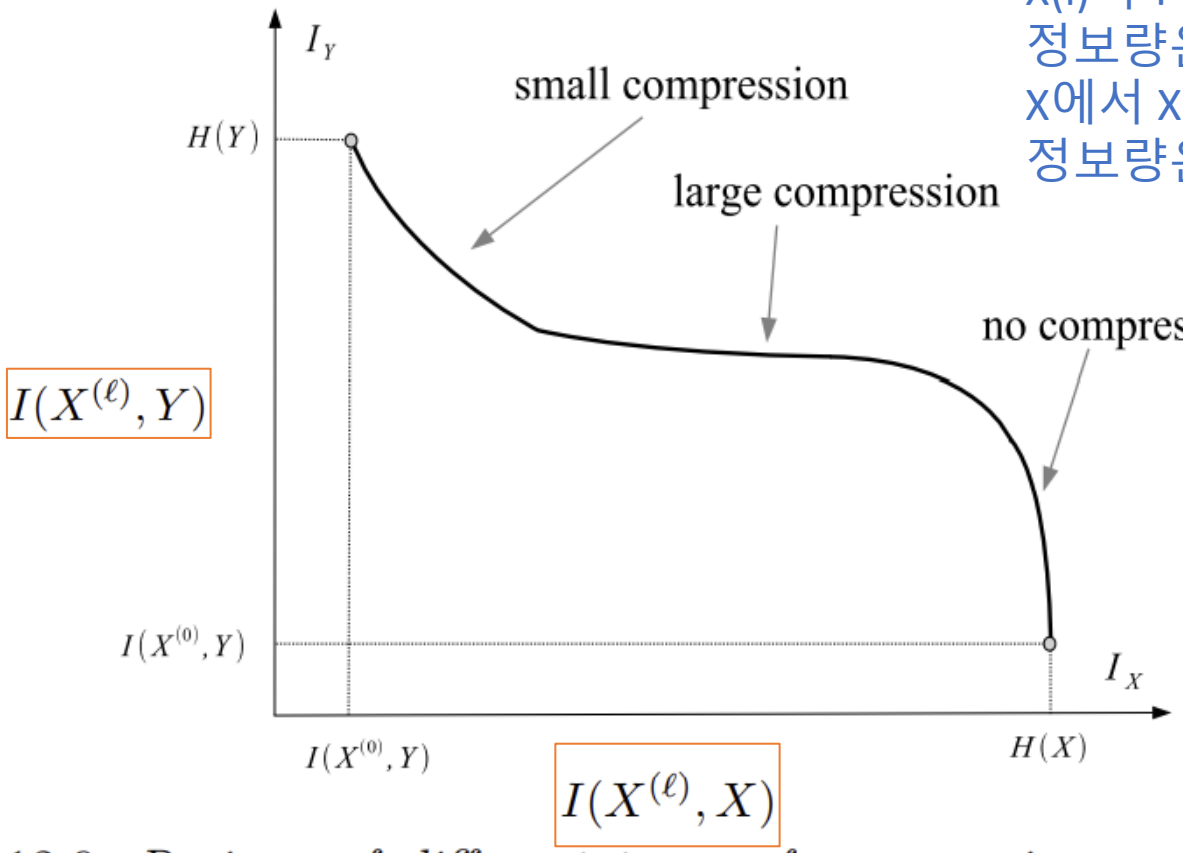
$$p(y|x) = \frac{p(y)}{Z(x,\beta)} e^{-\beta D_{KL}(p(z|x)||p(z|y))}, \quad p(z|y) = \frac{1}{p(y)} \sum_x p(z|x)p(y|x)p(x), \quad p(y) = \sum_x p(y|x)p(x),$$

Remark 12.10.3 Bottleneck principle can be applied repeatedly, layer by layer. Applying the bottleneck principle at each layer, we would like to minimize the information $I(X^{(\ell)}, X^{(\ell+1)})$, while keeping $I(X^{(\ell+1)}, Z)$ as large as possible. This is to compress the information between the ℓth and $(\ell+1)$th layers, while keeping enough meaningful information on Z.

Layer 와 layer 사이에도 적용 가능

12.10 The Information Bottleneck

X(l)이 Y에게 전달하는
정보량은 비슷하지만,
X에서 X(l)로 전달하는
정보량은 줄어든다.



Input X 에서 X(l)로
정보가 많이 전달된다.

Figure 12.9: *Regions of different types of compression on the information curve.*

12.11 Information Processing with MNIST

Information 관점에서 MNIST 보기

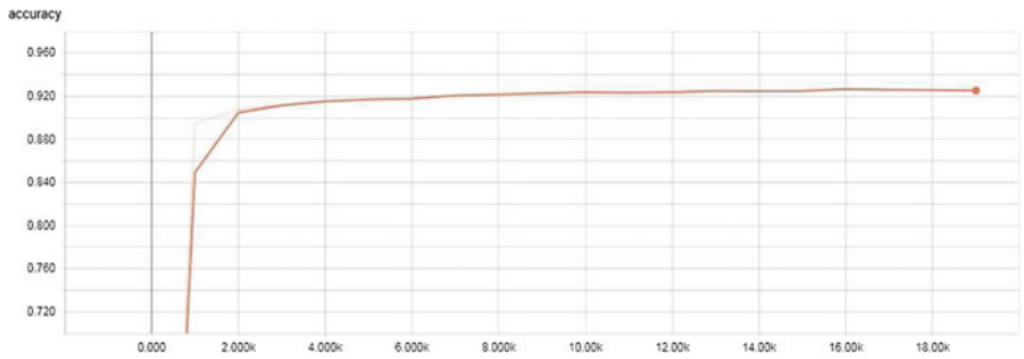


Figure 12.10: *A zero-hidden layer feedforward network using a batch size of 30, the softmax activation function, and the gradient descent method with learning rate* $\lambda = 0.03$*, producing a testing accuracy of 92.5%. The network*

Input – output

92.5% accuracy

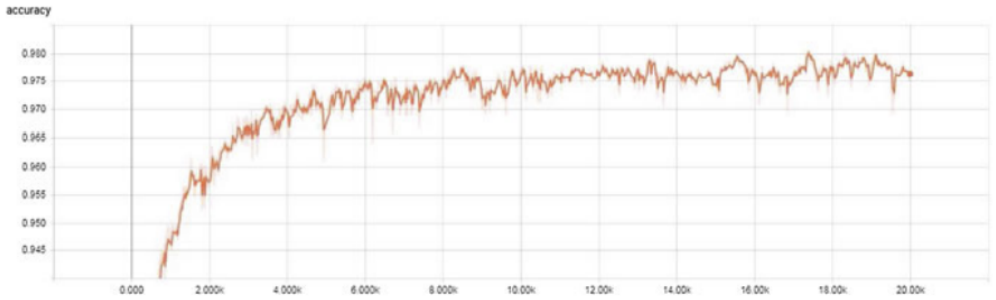


Figure 12.11: *A one-hidden layer feedforward network, with 300 neurons in the hidden layer, trained with a batch size of 40, using the ReLU and the softmax activation functions for the hidden and output layer, respectively, produces a testing accuracy of 97.6%. The learning uses ADAM method with*

Input – hidden - output

97.6% accuracy

Hidden layer is able to collect more features of input X, adding some extra capacity to the network.

12.11 Information Processing with MNIST

12.11.3 The role of convolutional nets

The low performance of a fully-connected layer feedforward neural network used in the classification of the MNIST data is due to two kinds of information losses:

(i) One is due to the low capacity of the two-layer network. This can be fixed by adding more layers or more neurons in the hidden layer to increase the network capacity. However, there is an upper bound of about 98% for the network accuracy in this case, which cannot be exceeded, regardless of how wide the hidden layer is, or how many hidden layers are added to the network.

(ii) To gain the missing 2% we need to acknowledge another information loss in the input data, due to flattening out the image. This removes some of the 2-dimensional information, such as the relation of a pixel with its neighboring pixels. Hence, a neural network with an architecture which takes advantage of the 2-dimensional data structure is needed, and this is the convolution neural network (CNN). Chapter 16 will discuss this type of networks in more detail.

Fully connected layer는 MNIST 데이터를 구분하는데 정보손실이 있다.

Fully connected layer로는 아무리 layer 수를 늘리고, neuron 수를 늘려도 98%까지

사라진 2%를 찾기 위해서는 flatting이 아닌 2d를 다루는 CNN이 필요하다.

– log2(0.98) = 0.029 bits => ignore about 0.029 bits per MNIST image.

HW

Exercise 중 1개 이상

HW

Entropy

Exercise 12.13.15 For any $p \in (0,1)$ consider the *binary entropy function*

$$H(p) = -p \ln p - (1-p) \ln(1-p).$$

(a) Show that $H(p)$ is the entropy associated with a Bernoulli random variable.
(b) Verify the following relation between the derivative of the binary entropy and the logit function:

$$\frac{dH(p)}{dp} = -\ln\left(\frac{p}{1-p}\right).$$

HW

Mutual Information

Exercise 12.13.3 (a) Define the mutual information of X and Y, given Z as

$$I(X,Y|Z) = H(X|Z) + H(Y|Z) - H(X,Y|Z).$$

Show that $I(X,Y|Z) = D_{KL}[p(x,y,z)||p(x|z)p(y|z)]$.

(b) Show that for any three random variables X, Y, and Z, we have:

$$H(X|Z) + H(Y|Z) \leq H(X,Y|Z).$$

When is the identity satisfied?

HW

Network Capacity

Exercise 12.13.7 Consider a neural network obtained by the concatenation of two perceptrons. The output of the network is given by the random variable

$$Y = H\big(w_2 H(w_1 X + b_1) + b_2\big),$$

with $X \in \{0, 1\}$. What is the capacity of this network?

Exercise 12.13.9 How does the capacity of a network change when:
(a) An extra fully-connected layer is added to the network;
(b) Some neurons are dropped out of the network;
(c) The weights are constrained to be kept small.