3. Data Compression, AEP and Lossless Source Coding

Vaughan Sohn

October 14, 2024

Contents

Source coding

Decodability and optimality

Huffman Code

Asmptotic Equipartition Property (AEP)

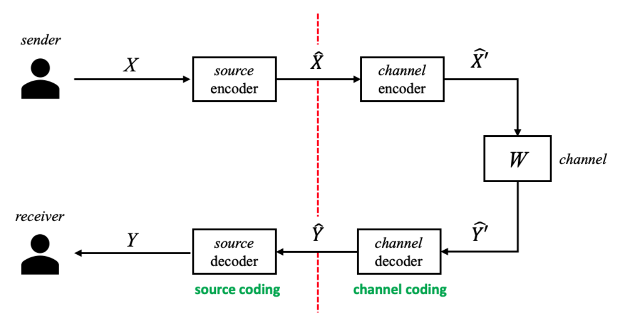
Loseless source coding

Source coding

Communication System

Communication system을 표현할 때, 정보이론에서는 "Shannon" 이 제안한 *digital communication*을 따른다.

- Shannon의 아이디어는 source와 channel을 나타낼 때, 일관되는 하나의 양식인 binary representation을 따르게 하는 것이다.
- Interface를 이용하면, source data의 종류나 channel의 물리적 특성에 관계없이 encoder/decoder만 잘 설계하면 동일한 framework를 적용할 수 있다!
- 이번 챕터에서는 "Source coding" 에 대해 다루고자 한다.



Source coding

Source coding은 크게 2가지로 이루어져 있다.

- source encoder: source data를 bit string으로 encoding한다.
- source decoder: encoded data를 다시 source data로 decoding한다.

Question

어떻게 해야 "좋은 encoder"를 설계할 수 있는가?

Source data가 다음의 특성을 가진다고 가정하자.

- source는 특정 alphabet $\mathcal{X}$에 속한다.
- source sequence는 i.i.d이다.
- source는 특정 distribution P를 따른다.

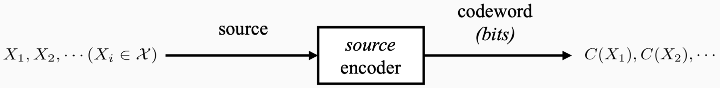
Definition 1 (Discrete Memoryless Source: DMS)

A sequence $\{X_i\}_{i=1}^{\infty} \sin i.i.dP$ on $\mathcal{X}$ is called a **DMS(P)**

Fixed-length source coding

Definition 2 (fixed-length code)

A **Fixed-length code** is a code where each codeword $C(x)$ is restricted to have the *same block length* L.



$$C : \mathcal{X} \to \{0,1\}^L$$

- *Decodability*를 위하여 fixed-length code의 length는 다음 조건을 만족해야한다.

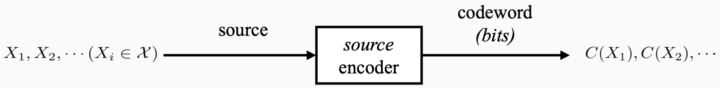
$$\log M \leq L < \log M + 1, \qquad (M = |\mathcal{X}|)$$

- Example:
  If the alphabet $\mathcal{X}$ consists of the 7 symbols $\{a, b, c, d, e, f, g\}$,

Variable-length source coding

Definition 3 (variable-length code)

A **Variable-length code** is a code where each codeword $C(x)$ can have a different length; number of bits $l(x)$ of $C(x)$.



$$C : \mathcal{X} \to \{0, 1\}^*$$

- Variable-length code의 length 대한 기댓값 R은 다음과 같이 계산할 수 있다.

$$R = \sum_{x \in \mathcal{X}} p(x) \cdot l(x)$$

- (intuitive) 더 낮은 빈도로 나타나는 symbol에 더 긴 길이의 codeword를 할당하면, expected length of codeword R을 줄일 수 있을 것이다.
- (problem) 그러나, variable-length code는 fixed-length code와 다르게 어디까지가 하나의 codeword인지를 판단하기 어렵다.
  $\rightarrow$*Issue of decodability!*

Decodability and optimality

Uniquely decodable code

Definition 4 (extension)

The **extension** C^* of a code C is the mapping from finite-length string of $\mathcal{X}$ to finite-length string of $\mathcal{D} = \{0, 1\}$, defined by

$$C(x_1 x_2 \cdots x_n) = C(x_1)C(x_2)\cdots C(x_n).$$

Definition 5 (unique decodability)

A code is called **uniquely decodable** if its extension is *non-singular*. In other words, any encoded string in a uniquely decodable code has only one possible source string producing it.

✓meaning: extension은 주어진 source data sequence를 codeword sequence로 나타낸 data이다. 만약 이 extension으로부터 다시 원본 source data를 복원할 수 있다면, 그 code는 uniquely decodable이다.

Prefix-free code

Definition 6 (prefix-free code)

A code is called a **prefix-free code** if no codeword is a *prefix* of any other codeword.

$$C(x_j) \notin \{C_1(x_i), C_2(x_i), \cdots, C_n(x_i)\}, \qquad (\forall i, j, (i \neq j))$$

where $x_i = b_1 b_2 \cdots b_n$ and $C_k(x_i) = b_1 b_2 \cdots b_k$.

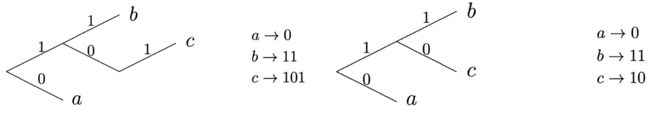
- prefix는 어떤 bit string의 initial substring을 의미한다.
- prefix-free code는 어떤 codeword의 prefix도 *자기 자신이 아닌 다른 codeword가 되지 않도록* 설계한 code를 의미한다.
- Example:

$$\begin{aligned} C(a) &= 0 && \rightarrow (\text{prefix: } 0) \\ C(b) &= 10 && \rightarrow (\text{prefix: } \{1, 10\}) \\ C(c) &= 11 && \rightarrow (\text{prefix: } \{1, 11\}) \end{aligned}$$

Binary-tree representation

Source code의 encoding scheme는 binary tree로 표현할 수 있다.

- prefix-free code를 tree로 나타내면, 모든 codeword가 leaf node에 위치하게 된다.
- full tree는 prefix-free 구조를 깨지 않고서는 더이상 새로운 노드를 추가할 수 없는 형태의 tree를 의미한다.
- 아래와 같이 $|\mathcal{X}| = 3$에 대해 full tree가 되도록 code를 설계하면, tree의 depth가 줄어들기에 codeword 길이의 기댓값이 줄어든다.



prefix-free tree　　full tree

Theorem 7

Every prefix-code is uniquely decodable.

$$\textit{Prefix-free} \subset \textit{Uniquely decodable} \subset \textit{code}$$

$\Rightarrow$ Prefix-code의 parsing은 binary-tree에 대한 searching을 *leaf node를 만날 때까지* 수행하면 되기 때문에 자명하게 uniquely decodable이다.□

Condition of optimal code

Lemma 8

Optimal codes have the property that if $p_i > p_j$, then $l_i \leq l_j$.

$*$ Proof: (귀류법) If $p_i > p_j$, then $l_i > l_j$라고 가정하자.
$\Rightarrow$

Lemma 9

Optimal prefix-free codes have the property that the associated code tree is full.

$*$ Proof: 만약 full tree가 아니라면, 언제나 tree depth를 더 줄일 수 있는 다른 full tree를 찾아낼 수 있기 때문에 optimal code가 될 수 없다.$\square$

Lemma 10

Two symbols with min. probability have the same length. In other words, If $p_1 \geq p_2 \geq \cdots \geq p_{M-1} \geq p_M$, then $l_{M-1} = l_M$.

$*$ Proof:
By Lemma 8, if $p_1 \geq p_2 \geq \cdots \geq p_{M-1} \geq p_M$, then $l_1 \leq l_2 \leq \cdots \leq l_{M-1} \leq l_M$.
By Lemma 9, full tree여야 하기 때문에, 가장 확률이 작은 두 노드의 depth는 동일해야한다. $\rightarrow$ sibling node $\square$

Huffman Code

Huffman Code Algorithm

For $X \in \mathcal{X}$, X be a random symbol with pmf $p_1 \geq p_2 \geq \cdots \geq p_{M-1} \geq p_M$,

1. Choose two least likely symbols (p_M, p_{M-1}) and constraining them to be siblings.
2. Now we consider *new data compression problem* with pmf $\{p_1, p_2, \cdots, p_{M-2}, p_M + p_{M+1}\}$
3. Repeat step 1 & 2 until there is no more remaining symbol.

Example:

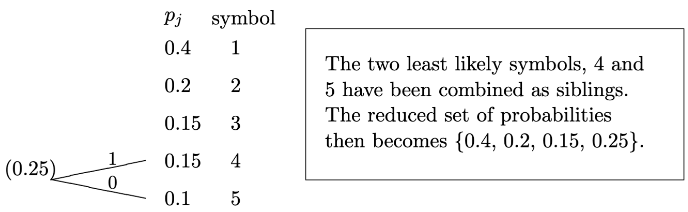


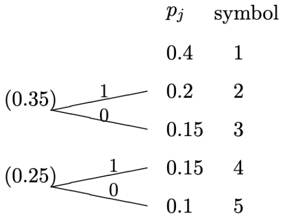
Figure 2.9: Step 1 of the Huffman algorithm; finding X' from X

Huffman Code Algorithm

For $X \in \mathcal{X}$, X be a random symbol with pmf $p_1 \geq p_2 \geq \cdots \geq p_{M-1} \geq p_M$,

1. Choose two least likely symbols (p_M, p_{M-1}) and constraining them to be siblings.
2. Now we consider *new data compression problem* with pmf $\{p_1, p_2, \cdots, p_{M-2}, p_M + p_{M+1}\}$
3. Repeat step 1 & 2 until there is no more remaining symbol.

Example (contd.):



The two least likely symbols in the reduced set, with probabilities 0.15 and 0.2, have been combined as siblings. The reduced set of probabilities then becomes {0.4, 0.35, 0.25}.

Figure 2.10: Finding X'' from X'.

For $X \in \mathcal{X}$, X be a random symbol with pmf $p_1 \geq p_2 \geq \cdots \geq p_{M-1} \geq p_M$,

1. Choose two least likely symbols (p_M, p_{M-1}) and constraining them to be siblings.
2. Now we consider *new data compression problem* with pmf $\{p_1, p_2, \cdots, p_{M-2}, p_M + p_{M+1}\}$
3. Repeat step 1 & 2 until there is no more remaining symbol.

Example (contd.):

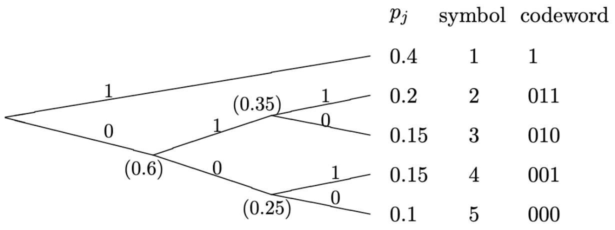


Figure 2.11: The completed Huffman code.

Optimality of Huffman code

Lemma 11

Huffman algorithm constructs the **optimal** *prefix-free code.* $\rightarrow$ *minimum* R*.*

$*$ Proof: X에 대한 average length를 $X' = X - \{M-1, M\}\}$에 대한 average length로 표현한 식은 다음과 같다.

$$R = R' + p_{M-1} + p_M$$

각 step마다 subproblem R'을 minimize하기 때문에, 다음을 이끌어낸다.

$$R_{min} = R'_{min} + p_{M-1} + p_M$$

따라서 Huffman algorithm은 optimal이다.$\square$

Asmptotic Equipartition Property (AEP)

Prerequisites: Weak Law of Large Numbers

Theorem 12 (Weak Law of Large Numbers (WLLN))

Let $X_1, X_2, \cdots$ be i.i.d with mean μ and variance $\sigma^2 < \infty$. Let define a new random variable S_n as

$$S_n \triangleq \frac{1}{n}(X_1 + X_2 + \cdots + X_n).$$

Then, if n is increasing then S_n convergence to μ.

$$S_n \xrightarrow{P} \mu, \quad , \text{ i.e.,} \quad (S_n - \mu) \xrightarrow{P} 0.$$

In other words, for all $\epsilon, \delta > 0$, there exists N_0 such that $\forall n > N_0$,

$$Pr(|S_n - \mu| > \delta) < \epsilon,$$

i.e., for all $\delta > 0$,

$$\lim_{n \to \infty} Pr[|S_n - \mu| > \delta] = 0.$$

✓meaning: Sample의 개수 n이 증가할 수록, S_n의 값이 X_i의 기댓값에 가까워진다는 내용을 다룬다. n이 커질수록 variance는 0에 수렴하기 때문에 점점 분포가 sharp해진다.

$$\mathbb{E}[S_n] = \mu, \quad Var[S_n] = \frac{\sigma^2}{n}$$

For proof of WLLN, we using markov inequality.

Theorem 13 (Markov inequality)

If $Z \geq 0$ and $\gamma > 0$, then

$$Pr(Z > \gamma) < \frac{\mathbb{E}[Z]}{\gamma}.$$

✓meaning: $[0, \infty)$ 범위에 있는 random variable Z에 대해서, Z가 특정 값 γ보다 클 확률은 γ에 대한 $\mathbb{E}[Z]$의 비율을 상한으로 가진다.

$*$ Proof of theorem 13: (hint) $\mathbb{E}[Z]$를 2개의 event에 대해 decomposition하자.

$$\mathbb{E}[Z] = \underbrace{\mathbb{E}[Z|Z > \gamma]}\,\mathrm{Pr}(Z > \gamma) + \underbrace{\mathbb{E}[Z|Z \leq \gamma]\,\mathrm{Pr}(Z \leq \gamma)}.$$

$\Rightarrow$

$*$ Proof of theorem 12: (hint) R.v. $Z \triangleq |S_n - \mu|^2$를 정의하자.
$\Rightarrow$

Asymptotic Equipartition Property (AEP)

Theorem 14 (Asymptotic Equipartition Property (AEP))

Let $X_1, X_2, \cdots, X_n$ *be i.i.d with* P *over* $\mathcal{X}$. *Consider* $\boxed{Y_i = -\log P(X_i)}$. *Then, by definition of entropy* $\mathbb{E}[Y_i] = H(P)$ *(*$*$*). Let us denote* $(x_1, x_2, \cdots, x_n)$ *by* x_1^n *or* x^n. *By applying WLLN,* $\forall \epsilon, \delta > 0$, $\exists N_0$ *such that* $\forall n > N_0(\epsilon, \delta)$,

$$P^n \left[\left\{ x_1^n : \left| \underbrace{-\frac{1}{n} \sum_{i=1}^{n} \log P(x_i)}_{\text{empirical mean}} - \underbrace{H(P)}_{\text{expectation}} \right| > \delta \right\} \right] < \epsilon$$

✓meaning: Empirical mean과 expectation의 차이가 δ보다 크게 만드는 sequence x_1^n 들의 집합의 확률은 ϵ보다 작다.

- P^n은 확률분포 P를 n번 곱한 것으로, i.i.d이기 때문에 joint probability가 각 marginal probability들의 곱과 동일하기 때문이다.
- $\log$함수의 성질에 의하면, summation of log를 다음과 같이 바꾸어 쓸 수 있다.

$$P^n \left[\left\{ x_1^n : \left| -\frac{1}{n} \log P^n(x_1^n) - H(P) \right| > \delta \right\} \right] < \epsilon$$

Weak typical set

Definition 15 (weak typical set)

We define **weak typical set** as,

$$A_\delta^{(n)}(P) \triangleq \left\{ x_1^n \in \mathcal{X}^n : \left| -\frac{1}{n} \log P^n\left(x_1^n\right) - H(P) \right| \leq \delta \right\}.$$

✓meaning: Empirical mean과 expectation의 차이가 δ보다 작게 만드는 sequence x_1^n 들의 집합을 weak typical set이라고 정의한다.

Corollary 16

Theorem 14 with typical set $A_\delta^{(n)}(P)$ implies that

$$P^n\left(A_\delta^{(n)}(P)\right) \geq 1 - \epsilon.$$

- Weak typical set도 x의 값에는 영향을 받지않고 그 r.v.가 따르는 확률분포에만 영향받기 때문에 P로 표현한다.
- 기호의 단순화를 위해 $\delta = \epsilon$으로 두자. $\rightarrow A_\epsilon^{(n)}(P)$

Weak typical set

$A_\epsilon^{(n)}(P)$를 이용하면, weak typical set이 가지는 아주 중요한 성질 하나를 보일 수 있다.

- By definition of weak typical set,

$$A_\epsilon^{(n)}(P) \triangleq \left\{ x_1^n \in \mathcal{X}^n : \left| -\frac{1}{n} \log P^n\left(x_1^n\right) - H(P) \right| \leq \epsilon \right\}.$$

- set의 elements의 표현에서 절댓값 기호를 제거한 뒤

$$= \left\{ x_1^n \in \mathcal{X}^n : -\epsilon \leq -\frac{1}{n} \log P^n\left(x_1^n\right) - H(P) \leq \epsilon \right\},$$

- set의 each element x_1^n에 대한 확률에 대해 부등식을 정리하면 다음을 얻는다.

$$= \left\{ x_1^n : 2^{-n(H(P)+\epsilon)} \leq P^n\left(x_1^n\right) \leq 2^{-n(H(P)-\epsilon)} \right\}$$

$\Rightarrow$ 즉, weak typical set안에 들어있는 sequence들은 거의 유사한 확률을 가진다!

Remarks

- 대부분의 sequence들은 weak typical set안에 존재한다.
- typical set안에 있는 sample들의 확률은 거의 동일하다.

Weak typical set

Corollary 17 (Size of typical set)

- *For typical set* $A_\epsilon^{(n)}(P)$,
$$|A_\epsilon^{(n)}(P)| \leq 2^{n(H(P)+\epsilon)}$$
- *For large enough* n,
$$|A_\epsilon^{(n)}(P)| \geq (1-\epsilon)2^{n(H(P)-\epsilon)}$$

$*$ Proof: (hint) sum of probability is 1, and Corollary 16 (with sufficiently large n)
$\Rightarrow$

With exponential approximation

Weak typical set에 대한 더 엄밀한 표현을 위해 exponential approximation을 적용한다.

Definition 18

For a sequence $f(1), f(2), \cdots, f(n)$ and a constant $c \in \mathbb{R}$, we say

$$f(n) \doteq e^{cn}, \quad iff \quad , \lim_{n \to \infty} \frac{\log f(n)}{n} = c.$$

Futhermore, we say $f(n) \doteq h(n)$ if both have the same exponential rate, i.e., $f(n) \doteq g(n) \doteq e^{cn}$

Example: 다음 두 함수가 same exponential rate를 가지는지 확인하라.

- $f(n), cf(n)$
- $f(n), n^2 f(n)$
- $f(n), f(n) + e^{-n}$

By exponential approximation, weak typical set은 다음의 성질을 가진다.

- $\forall x_1^n \in A_{\epsilon_n}^{(n)}(P), \quad P^n\left(x^n\right) \doteq 2^{-nH(P)}$.
- The size of set equals $|A_{\epsilon_n}^{(n)}(P)| \doteq 2^{nH(P)}$
- $P^n(A_{\epsilon_n}^{(n)}) \to 1$

Summary

Some Remarks

- (충분히 큰 n에 대하여) sample space $\mathcal{X}^n$에서 sampling한 data x^n은 높은 확률; $P^n \geq 1 - \epsilon$로 weak typical set에 포함된다.
- 그러나, 전체 sample space의 크기에 비하여 weak typical set의 크기는 strictly 작다.

$$2^{nH(P)} < 2^{n \log |\mathcal{X}|}$$

Idea

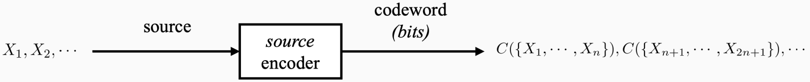
sample space의 전체 data에 대해서 encoding scheme을 설계하는 것보다 typical set에 포함되는 data에 대해서만 encoding scheme를 설계하는 것이 훨씬 더 효과적이다!
$\rightarrow$ *Lossless source coding*

Loseless source coding

Fixed-length block coding

Definition 19 (fixed-length block code)

A **Fixed-length block code** is a code where a source block x^n , consisting of n symbols from the source alphabet $(x_i \in \mathcal{X})$, is represented by a codeword $C(x^n)$.



$$C : \mathcal{X}^n \to \{0,1\}^*$$

- Block code를 사용했을 때, 각 symbol에 대한 code word length의 기댓값 R_n은 다음과 같이 계산할 수 있다.

$$R_n = \frac{1}{n} \sum_{x^n \in \mathcal{X}^n} p(x^n) \cdot l(x^n) \qquad \text{(bits/source)}$$

- 특정 rate R을 이용하는 fixed length block code의 encoder, decoder는 다음과 같이 표현된다.

$$f_n : X^n \to \{0,1\}^{nR} = \{0, 1, \cdots, 2^{nR} - 1\}$$
$$g_n : \{0,1\}^{nR} \to X^n$$

Lossless source coding

- block coding은 source data의 *sequence*를 다루기 때문에, sample space가 exponential하게 커진다.
- 만약, sample space의 모든 데이터에 대하여 one-to-one mapping을 사용하여 encoding을 하게 되면 decoding에서 error는 발생하지 않지만, codeword의 길이가 너무 길어진다.

$$|\mathcal{X}|^n \leq 2^{nR}, \quad nR \geq \lceil \log |\mathcal{X}|^n \rceil$$

- 따라서 약간의 error는 허용하고 codeword의 길이를 효율적으로 줄이기 위한 방법을 **lossless source coding**이라고 한다.

Definition 20

The probability of error source code scheme is defined as

$$P_e^{(n)} \triangleq P^n \left(\{ x^n : g_n \left(f_n \left(x^n \right) \right) \neq x^n \} \right)$$

lossless source coding에서 error는 n이 증가함에 따라서 0으로 수렴해야한다.

Definition 21

For any $\epsilon > 0$, we say a rate R is an **achievable rate** of source coding for with probability of error ϵ, if there exists a sequence of (n, R) codes such that

$$\limsup_{n \to \infty} P_e^{(n)} \leq \epsilon$$

Definition 22

The **minimum of such rates** for a given ϵ is denoted as $R_{\min}(\epsilon)$, which is the achievable source coding rate allowing error probability ϵ. We further define

$$R_{\min} = \lim_{\epsilon \to 0} R_{\min}(\epsilon)$$

Lossless source coding theorem

Theorem 23 (Lossless Source Coding Theorem)

The minimum achievable rate of lossless source coding for DMS (P) is

$$R_{\min} = H(P).$$

Lossless source coding theorem은 *Achievability*, 그리고 *Converse*의 2가지 성질을 모두 가지며, 두 upper bound와 lower bound가 동일한 complexity $O(H(P))$를 가지므로 최적의 solution이다.

- Achievability: 만약 $R > R_{\min} = H(P)$인 rate를 사용한다면, probability of error가 0으로 수렴하는; $\lim_{n\to\infty} P_e^{(n)} = 0$ sequence of (n, R) code가 존재한다.
- Converse: 만약 $R < R_{\min} = H(P)$인 rate를 사용한다면, 그 어떤 code scheme도 probability of error가 0으로 수렴함을 보장하지 못한다.

Proof of lossless source coding theorem: achievable

∗ Proof: (hint) Weak typical set

- Weak typical set 안에 있는 data에 대한 one-to-one mapping scheme를 고려하자. Weak typical set의 크기는 다음과 같이 bound된다.

$$|A_\delta^{(n)}(P)| \leq 2^{n(H(P)+\delta)}$$

- 이때, $R > H(P)$인 rate를 사용한다고 생각하자. 그럼 항상 우리는 다음과 같은 δ를 고려할 수 있다.

$$R > H(P) + \delta > H(P)$$

- error는 source data가 typical set안에 존재하지 않을 때만 발생하기 때문에, 다음과 같다.

$$P_\epsilon^{(n)} \geq 1 - \epsilon$$

- 따라서 WLLN에 의해 if $n \to \infty$, then

$$P_\epsilon^{(n)} \to 0.$$

Proof of lossless source coding theorem: converse

∗ Proof: (hint) Fano's inequality

- Source coding에서 decoding은 fano's inequality의 experiment로 생각할 수 있다.
  - X^n: 원본 source data
  - $Y = f_n(X^n)$: encoder
  - $\hat{X} = g_n(Y)$: decoder
  - $P_e^{(n)} = P(g_n(Y) \neq X^n)$: error probability
- 따라서 Fano's inequality를 이용하여 전개하면 다음과 같고

$$
\begin{aligned}
nH(P) = H(X^n) = H(X^n|Y) + I(X^n;Y) \\
&\leq \\
&\leq \\
&\leq
\end{aligned}
$$

- error probability에 대해 정리하면 다음과 같다.
  - if $R < H(P)$: $\lim_{n\to\infty}$일 때, 0으로 수렴하지 않는다.
  - if $R > H(P)$: $\lim_{n\to\infty}$일 때, 0으로 수렴한다.

$$
P_e^{(n)} \geq \frac{H(P) - R - \frac{1}{n}}{\log|\mathcal{X}|}
$$

- 따라서, $n \to \infty$일 때, $P_\epsilon^{(n)} \to 0$을 만족하기 위해서는 $R \geq H(P)$여야한다.□

Summary

Some Remarks

- Operation meaning of $H(P)$
  - 원래 entropy는 $X \sim P$에 대해 realization value x를 알았을 때, 얻게되는 정보량에 대한 average를 의미한다.
  - data compression의 관점에서 생각하면, $n \to \infty$에 따라 $nH(P)$ bits가 lenght n sequence를 표현하기 위한 bits의 개수가 된다.
- Variable-length block code
  - Variable length coding 방식을 이용하면 $nH(P)$보다 더 작은 길이의 encoding scheme을 설계할 수 있는가? $\to$ 불가능
  - typical set 내부의 data들의 확률은 거의 동일하기 때문에, Huffman algorithm 같은 방식을 도입해도 길이가 비슷하다.
- Optimal uniquely decodable code $P_\epsilon^{(n)} = 0$
  - (prefix-code) typical set 내부에 있는 data는 0으로 시작하는 bit-string과 mapping하고 typical set 외부에 있는 data는 1로 시작하는 bit-string과 mapping한다.
  - average rate 다음과 같기 때문에, $n \to \infty$에 따라서 $R \to H(P)$로 수렴한다.

$$nR = P^n(A_\epsilon^{(n)}(P)) \cdot (1 + nH(P)) + P^n(A_\epsilon^{(n)}(P)^c) \cdot (1 + n \log |\mathcal{X}|)$$

References

- T. M. Cover and J. A. Thomas. Elements of Information Theory, Wiley, 2nd ed., 2006.
- Gallager (2008), Principles of Digital Communication, Cambridge University Press.
- Lecture notes for EE623: Information Theory (Fall 2024)