Noiseless Lossless Coding and Entropy — Lectures 3, 4

Haruka Eshima

September 2026

Abstract

This is a lecture note that I took for the Information Theory class taught by Prof. Amedeo Roberto Esposito at the Okinawa Institute of Science and Technology, 2026. Any errors are my own.
The questions in green are made by me and are not officially related to the mini test in the class.

Convention on logarithms. In the coding and entropy sections, $\log = \log_2$ and entropy is measured in bits. In the large-deviation section, $\ln$ denotes the natural logarithm; cumulant generating functions are always taken with $\ln$. Wherever the two conventions meet, the base is stated explicitly. Throughout, $0 \log \frac{1}{0} := 0$.

Contents

Summary **2**

1 Large deviations **2**
1.1 Example 1: Gaussian . . . 3
1.2 Example 2: Rademacher . . . 3

2 Source coding: setup **4**
2.1 Prefix-free implies uniquely decodable . . . 5

3 Kraft's theorem **6**
3.1 Necessity . . . 7
3.2 Sufficiency . . . 7

4 Shannon's noiseless coding theorem: the lower bound **8**
4.1 The same bound via a Lagrangian . . . 9

5 Shannon's noiseless coding theorem: the upper bound **10**

6 Huffman coding **10**
6.1 Optimality of the Huffman code . . . 11

7 Properties of entropy **13**
7.1 Property 1: $0 \le H(X) \le \log|\mathcal{X}|$ . . . 13
7.2 Property 2: concavity in P . . . 13
7.3 Property 3: chain rule . . . 13
7.4 Property 4: conditioning reduces entropy . . . 14
7.5 Property 5: subadditivity . . . 15
7.6 Property 6: processing cannot increase entropy . . . 15

8 Rate: the asymptotic cost per symbol **15**

9 Conditioning on a single value **16**

Summary

- Prefix-free coding; Kraft's inequality.
- For every prefix-free code $\mathscr{C}$: $H(P) \le \mathbb{E}_P[\ell_{\mathscr{C}}(X)]$.
- There exists a prefix-free code $\mathscr{C}$ with $\mathbb{E}_P[\ell_{\mathscr{C}}(X)] < H(P)+1$ — e.g. the Shannon code or the Huffman code.
- Rate: $\lim_{n\to\infty} \frac{1}{n}\,\mathbb{E}_{P^n}\big[\ell_{\mathscr{C}_n}(X^n)\big] = H(P)$.
- Six properties of entropy.

1 Large deviations

Let $X_1, \dots, X_n$ be i.i.d. with $\ln \mathbb{E}[e^{sX_1}] < \infty$ for all $s \ge 0$, and set $S_n := \sum_{i=1}^n X_i$.

Remark 1.1 (Motivation)**.** We know as $n \to \infty$, $S_n/n \to \mathbb{E}[X]$ almost surely. But how much is a deviation from $\mathbb{E}[X]$ at a fixed size n?

Definition 1.2 (CGF and its Legendre–Fenchel dual)**.** The *cumulant generating function* (CGF, or log-MGF) of X is

$$\Lambda(s) := \ln \mathbb{E}\big[e^{sX}\big],$$

and its *Legendre–Fenchel dual* (the rate function) is

$$\Lambda^*(t) := \sup_{s\ge 0}\big(st - \Lambda(s)\big).$$

Q. *W.l.o.g.* $\mathbb{E}[X_1] = 0$. *Bound the large deviation* $\mathbb{P}\big(\{S_n/n \ge t\}\big)$ *using the Legendre–Fenchel dual of the CGF.*

Theorem 1.3 (Chernoff bound)**.** *For every* $t \ge 0$,

$$\mathbb{P}\Big(\Big\{\tfrac{S_n}{n} \ge t\Big\}\Big) \;\le\; \exp\big(-n\,\Lambda^*(t)\big).$$

Proof. Fix $s \geq 0$ and apply the generalized Markov inequality with the non-decreasing function $\psi(x) = e^{sx} > 0$ to the variable S_n at the level nt:

$$\begin{aligned}
\mathbb{P}\Big(\Big\{\tfrac{S_n}{n} \geq t\Big\}\Big) = \mathbb{P}\big(\{S_n \geq nt\}\big) \;&\leq\; \frac{\mathbb{E}\big[e^{sS_n}\big]}{e^{snt}} \;=\; \exp\Big(\ln \mathbb{E}\big[e^{sS_n}\big] - snt\Big) \\
&= \exp\Big(\ln \mathbb{E}\big[e^{s\sum_i X_i}\big] - snt\Big) \;=\; \exp\Big(\ln \prod_{i=1}^{n} \mathbb{E}\big[e^{sX_i}\big] - snt\Big) \quad \text{(independence)} \\
&= \exp\Big(\underbrace{n \ln \mathbb{E}\big[e^{sX}\big]}_{= n\Lambda(s)} - snt\Big) \qquad\qquad\qquad\qquad\qquad \text{(identical distribution)} \\
&= \exp\Big(-n\big(st - \Lambda(s)\big)\Big).
\end{aligned}$$

Since this holds for every $s \geq 0$,

$$\mathbb{P}\Big(\Big\{\tfrac{S_n}{n} \geq t\Big\}\Big) \leq \inf_{s\geq 0} \exp\Big(-n\big(st - \Lambda(s)\big)\Big) = \exp\Big(-n \sup_{s\geq 0}\big(st - \Lambda(s)\big)\Big) = \exp\big(-n\Lambda^*(t)\big). \qquad \square$$

Remark 1.4 (Why $s \geq 0$ and why $t \geq 0$)**.** The Markov step needs ψ non-decreasing, i.e. $s \geq 0$. The restriction $s \geq 0$ in the definition of Λ^* costs nothing when $t \geq \mathbb{E}[X] = 0$: for such t the unconstrained supremum over $s \in \mathbb{R}$ is attained at some $s \geq 0$, because Λ is convex with $\Lambda(0) = 0$ and $\Lambda'(0) = \mathbb{E}[X] = 0$. For $t < 0$ the supremum over $s \geq 0$ is attained at $s = 0$, giving $\Lambda^*(t) = 0$ and the vacuous bound $\mathbb{P}(\cdot) \leq 1$ — as it must be, since the Chernoff method controls only the upper tail.

Remark 1.5. A large deviation can be bounded via the CGF only when $\mathbb{E}[e^{sX}] < \infty$ for some $s > 0$ — i.e. only for distributions with at least one exponentially decaying tail. For heavy-tailed X the method says nothing.

1.1 Example 1: Gaussian

Example 1.6. Let $X \sim N(0,1)$. Completing the square,

$$\Lambda(s) = \ln \mathbb{E}\big[e^{sX}\big] = \ln \int_{\mathbb{R}} \frac{1}{\sqrt{2\pi}} e^{sx - x^2/2}\, dx = \ln\Big(e^{s^2/2} \int_{\mathbb{R}} \tfrac{1}{\sqrt{2\pi}} e^{-(x-s)^2/2}\, dx\Big) = \frac{s^2}{2}.$$

Hence for $t \geq 0$,

$$\Lambda^*(t) = \sup_{s\geq 0}\Big(st - \frac{s^2}{2}\Big) = \frac{t^2}{2} \qquad \text{(maximiser } s^* = t\text{)},$$

and Theorem 1.3 gives

$$\mathbb{P}\Big(\Big\{\tfrac{S_n}{n} \geq t\Big\}\Big) \leq \exp\Big(-\frac{nt^2}{2}\Big).$$

1.2 Example 2: Rademacher

Example 1.7. Let X be Rademacher: $X = +1$ w.p. $\frac{1}{2}$ and $X = -1$ w.p. $\frac{1}{2}$. Then

$$\Lambda(s) = \ln \mathbb{E}\big[e^{sX}\big] = \ln \tfrac{1}{2}\big(e^s + e^{-s}\big) = \ln \cosh s.$$

For $0 \leq t < 1$,

$$\frac{d}{ds}\Big(st - \ln\cosh s\Big) = t - \frac{e^s - e^{-s}}{e^s + e^{-s}} = t - \tanh s,$$

so on $s \geq 0$ the expression is maximised at s^* with $\tanh s^* = t$, i.e.

$$s^* = \tfrac{1}{2}\ln\Big(\frac{1+t}{1-t}\Big), \qquad \cosh^2 s^* = \frac{1}{1-\tanh^2 s^*} = \frac{1}{1-t^2}.$$

Therefore

$$\begin{aligned}\Lambda^*(t) = s^* t - \ln\cosh s^* &= \frac{t}{2}\ln\Big(\frac{1+t}{1-t}\Big) + \frac{1}{2}\ln\big(1-t^2\big) \\ &= \frac{1+t}{2}\ln(1+t) + \frac{1-t}{2}\ln(1-t).\end{aligned}$$

Consequently

$$\begin{cases} \text{for } 0 \leq t < 1: & \mathbb{P}\big(\{\frac{S_n}{n} \geq t\}\big) \;\leq\; (1+t)^{-\frac{n(1+t)}{2}}\,(1-t)^{-\frac{n(1-t)}{2}}, \\ \text{for } t = 1: & \mathbb{P}\big(\{\frac{S_n}{n} \geq 1\}\big) = 2^{-n}, \\ \text{for } t > 1: & \mathbb{P}\big(\{\frac{S_n}{n} \geq t\}\big) = 0. \end{cases}$$

Remark 1.8. At $t = 1$ the Chernoff bound is *exact*. Indeed $\Lambda^*(1) = \lim_{s\to\infty}(s - \ln\cosh s) = \ln 2$, so the bound reads $e^{-n\ln 2} = 2^{-n}$, which is precisely $\mathbb{P}(X_1 = \cdots = X_n = +1)$. For $t > 1$, $\Lambda^*(t) = +\infty$ and the bound degenerates to 0, again exact.

2 Source coding: setup

Remark 2.1 (Motivation)**.** Let

$$X = \begin{cases} a & \text{w.p. } p_a \\ b & \text{w.p. } p_b \\ c & \text{w.p. } p_c \\ d & \text{w.p. } p_d \end{cases}$$

We want to map sequences X^n to binary strings in $\{0,1\}^*$ so that X^n can be reconstructed. The naive way is $(a,b,c,d) \mapsto (00,01,10,11)$. But we can do better by exploiting the probabilities.

Example 2.2 (Four candidate codes)**.**

x	$P_X(x)$	Code 1	Code 2	Code 3	Code 4
a	1/2	00	0	0	0
b	1/4	01	1	01	10
c	1/8	10	00	011	110
d	1/8	11	11	111	111

- **Code 1** is lossless but redundant: it ignores the probabilities.
- **Code 2** is not injective on strings: $\mathscr{C}(aa) = 00 = \mathscr{C}(c)$.
- **Code 3** is injective but not prefix-free (0 is a prefix of 01): decoding 011111 requires reading to the end before committing.
- **Code 4** is prefix-free — decoding is instantaneous.

Let X be distributed according to P_X and take values in $\mathcal{X}$ (finite), and let $n \in \mathbb{N}$. We want to encode $X^n \in \mathcal{X}^n$ in terms of 0 and 1.

Q. *What is lossless compression?*

Definition 2.3 (Lossless compression)**.** We want two functions

$$\mathscr{C} : \mathcal{X}^n \to \{0,1\}^* \qquad \text{and} \qquad \psi : \{0,1\}^* \to \mathcal{X}^* \qquad \text{s.t.} \qquad \psi\big(\mathscr{C}(x^n)\big) = x^n \quad \forall x^n \in \mathcal{X}^n.$$

We write $\mathscr{C} : \mathcal{X} \to \{0,1\}^*$ for the single-symbol encoder and extend it by concatenation:

$$\mathscr{C}_n : \mathcal{X}^n \to \{0,1\}^*, \qquad \mathscr{C}_n(x^n) = \mathscr{C}(x_1) \cdot \mathscr{C}(x_2) \cdots \mathscr{C}(x_n),$$

and $\mathscr{C}_* : \mathcal{X}^* \to \{0,1\}^*$ for the extension to strings of arbitrary length.

Q. *State the definition of uniquely decodable.*

Definition 2.4 (Uniquely decodable)**.** Fix $n \in \mathbb{N}$. The code $\mathscr{C}$ is *uniquely decodable* (at block length n) if $\mathscr{C}_n$ is injective.

Q. *State the definition of prefix-free.*

Definition 2.5 (Prefix-free)**.** A code $\mathscr{C} : \mathcal{X} \to \{0,1\}^*$ is *prefix-free* if for all $x \neq y$ in $\mathcal{X}$,

$$\mathscr{C}(x) \notin \mathscr{C}(y)\{0,1\}^* \qquad \text{and} \qquad \mathscr{C}(y) \notin \mathscr{C}(x)\{0,1\}^*,$$

i.e. neither codeword is a prefix of the other.

Remark 2.6. We assume throughout that $|\mathcal{X}| \geq 2$ and that no codeword is the empty string ε. This is automatic: ε is a prefix of every string, so a prefix-free code containing ε has only one codeword.

2.1 Prefix-free implies uniquely decodable

Q. *State and prove the claim.*

Claim 2.7 (†)**.** *Let $\mathscr{C}$ be prefix-free. If $\mathscr{C}(x)$ and $\mathscr{C}(y)$ are both prefixes of the same word s, then $x = y$.*

Proof. Suppose $\mathscr{C}(x)$ and $\mathscr{C}(y)$ are both prefixes of s.

- If $|\mathscr{C}(x)| = |\mathscr{C}(y)|$, then both are the first $|\mathscr{C}(x)|$ bits of s, so $\mathscr{C}(x) = \mathscr{C}(y)$. Since $\mathscr{C}$ is prefix-free it is injective (if $x \neq y$ and $\mathscr{C}(x) = \mathscr{C}(y)$, then $\mathscr{C}(x)$ is a prefix of $\mathscr{C}(y)$, contradicting Definition 2.5), hence $x = y$.
- If $|\mathscr{C}(x)| < |\mathscr{C}(y)|$, then $\mathscr{C}(x)$ is the first $|\mathscr{C}(x)|$ bits of s and so also of $\mathscr{C}(y)$; thus $\mathscr{C}(x)$ is a prefix of $\mathscr{C}(y)$, which contradicts prefix-freeness unless $x = y$.
- The case $|\mathscr{C}(y)| < |\mathscr{C}(x)|$ is symmetric.

In every case $x = y$. □

Proposition 2.8. *If $\mathscr{C}$ is prefix-free, then for every n, $\mathscr{C}_n$ is prefix-free — and in particular injective.*

Q. *Prove the proposition using the claim.*

Proof. Suppose $\mathscr{C}_n(x_1 \cdots x_n)$ is a prefix of $\mathscr{C}_n(y_1 \cdots y_n)$. Both $\mathscr{C}(x_1)$ and $\mathscr{C}(y_1)$ are prefixes of $\mathscr{C}_n(y_1 \cdots y_n)$, so by Claim 2.7, $x_1 = y_1$. Cancelling the common prefix $\mathscr{C}(x_1) = \mathscr{C}(y_1)$ leaves

$$\mathscr{C}(x_2) \cdots \mathscr{C}(x_n) \;\text{ a prefix of }\; \mathscr{C}(y_2) \cdots \mathscr{C}(y_n),$$

and the same argument gives $x_2 = y_2$. Continuing, $x_i = y_i$ for every $i \le n$, so $x_1^n = y_1^n$ and the two encodings are equal. Hence no codeword of $\mathscr{C}_n$ is a proper prefix of another, i.e. $\mathscr{C}_n$ is prefix-free; taking the prefix relation to be equality gives injectivity. □

Proposition 2.9. *If $\mathscr{C}$ is prefix-free, then $\mathscr{C}_* : \mathcal{X}^* \to \{0,1\}^*$ is injective.*

Proof. Assume $\mathscr{C}_*(x) = \mathscr{C}_*(y)$ for some $x = x_1 \cdots x_m \in \mathcal{X}^m$ and $y = y_1 \cdots y_n \in \mathcal{X}^n$; w.l.o.g. $m \le n$. Both $\mathscr{C}(x_1)$ and $\mathscr{C}(y_1)$ are prefixes of the common word $\mathscr{C}_*(x)$, so Claim 2.7 gives $x_1 = y_1$; cancelling and iterating gives $x_i = y_i$ for $i \le m$. Cancelling all m of them leaves

$$\varepsilon = \mathscr{C}(y_{m+1}) \cdots \mathscr{C}(y_n).$$

If $m < n$ this forces $\mathscr{C}(y_{m+1}) = \varepsilon$, which is excluded by Remark 2.6. Hence $m = n$ and $x = y$. □

Remark 2.10. $\mathscr{C}_*$ is *not* itself prefix-free as a map on $\mathcal{X}^*$: for $x, y \in \mathcal{X}^*$ with $x, y \neq \varepsilon$ we have $\mathscr{C}_*(xy) = \mathscr{C}_*(x)\mathscr{C}_*(y)$, so $\mathscr{C}_*(x)$ is always a prefix of $\mathscr{C}_*(xy)$. Injectivity is the right property at the level of $\mathcal{X}^*$.

3 Kraft's theorem

Definition 3.1 (Length function)**.** Given an encoder $\mathscr{C} : \mathcal{X} \to \{0,1\}^*$, the *length* $\ell_{\mathscr{C}}$ is the function $\ell_{\mathscr{C}} : \mathcal{X}^* \to \mathbb{N}_0$ determined by

$$\ell_{\mathscr{C}}(\varepsilon) = 0, \qquad \ell_{\mathscr{C}}(x_1 \cdots x_w) = \big|\mathscr{C}(x_1)\big| + \cdots + \big|\mathscr{C}(x_w)\big|.$$

Our objective is

$$\min_{\mathscr{C}\ \text{prefix-free}} \mathbb{E}\big[\ell_{\mathscr{C}}(X)\big] = \min_{\mathscr{C}\ \text{prefix-free}} \sum_{x \in \mathcal{X}} P_X(x)\,\big|\mathscr{C}(x)\big|.$$

Q. *State Kraft's theorem.*

Theorem 3.2 (Kraft)**.** *Let $\ell_1, \ldots, \ell_m$ be non-negative integers. There exists a prefix-free code $\mathscr{C}$ with codeword lengths $\ell_1, \ldots, \ell_m$ if and only if*

$$\sum_{i=1}^{m} 2^{-\ell_i} \;\le\; 1 \qquad \textit{(Kraft's inequality).}$$

3.1 Necessity

Q. *Let $\mathscr{C}$ be prefix-free with lengths $\ell_1, \ldots, \ell_m$ and codewords $c_1, \ldots, c_m$. Prove $\sum_i 2^{-\ell_i} \leq 1$.* (Hint: construct disjoint sets.)

Proof of necessity. Let $D = \max\{\ell_1, \ldots, \ell_m\}$ and define, for each i,

$$A_i := \big\{ s \in \{0,1\}^D \ : \ c_i \text{ is a prefix of } s \big\}.$$

Each A_i is obtained by fixing the first ℓ_i bits and letting the remaining $D - \ell_i$ bits range freely, so $|A_i| = 2^{D-\ell_i}$.

The A_i are pairwise disjoint: if $s \in A_i \cap A_j$ with $i \neq j$, then c_i and c_j are both prefixes of the same word s, so Claim 2.7 gives $i = j$ — a contradiction.

Hence

$$\Big| \bigcup_{i=1}^{m} A_i \Big| = \sum_{i=1}^{m} 2^{D-\ell_i},$$

and since $\bigcup_{i=1}^{m} A_i \subseteq \{0,1\}^D$,

$$2^D \ \geq \ \sum_{i=1}^{m} 2^{D-\ell_i} \qquad \Longrightarrow \qquad \sum_{i=1}^{m} 2^{-\ell_i} \leq 1. \qquad \square$$

3.2 Sufficiency

Q. *If $\sum_i 2^{-\ell_i} \leq 1$, construct codewords $c_1, \ldots, c_m$ with lengths $\ell_1, \ldots, \ell_m$ forming a prefix-free code.* (Hint: use induction.)

Proof of sufficiency. Relabel so that $\ell_1 \leq \ell_2 \leq \cdots \leq \ell_m$. This is w.l.o.g.: Kraft's sum is invariant under permutation, and a permutation of the lengths just permutes the symbols.

We construct the codewords greedily. Let $0 \leq j < m$ and suppose $c_1, \ldots, c_j$ have been chosen with $|c_k| = \ell_k$ and forming a prefix-free code. We show c_{j+1} of length ℓ_{j+1} can be chosen so that $c_1, \ldots, c_{j+1}$ is again prefix-free. (For $j = 0$ there is nothing to exclude.)

For $k \leq j$ set

$$F_k := \big\{ s \in \{0,1\}^{\ell_{j+1}} \ : \ c_k \text{ is a prefix of } s \big\}.$$

Because $\ell_k \leq \ell_{j+1}$ (this is where the sorting is used), $|F_k| = 2^{\ell_{j+1} - \ell_k}$; and $F_i \cap F_p = \varnothing$ for $i \neq p$ by Claim 2.7 applied to the prefix-free code $c_1, \ldots, c_j$. Hence

$$\big| F_1 \cup \cdots \cup F_j \big| = \sum_{k=1}^{j} 2^{\ell_{j+1} - \ell_k}.$$

Kraft's inequality gives $\sum_{k=1}^{j} 2^{-\ell_k} + 2^{-\ell_{j+1}} \leq 1$, so multiplying by $2^{\ell_{j+1}}$,

$$\big| F_1 \cup \cdots \cup F_j \big| + 1 = 2^{\ell_{j+1}} \Big(\sum_{k=1}^{j} 2^{-\ell_k} + 2^{-\ell_{j+1}} \Big) \ \leq \ 2^{\ell_{j+1}} = \big| \{0,1\}^{\ell_{j+1}} \big|.$$

Therefore $\{0,1\}^{\ell_{j+1}} \setminus (F_1 \cup \cdots \cup F_j) \neq \varnothing$ and we may pick

$$c_{j+1} \in \{0,1\}^{\ell_{j+1}} \setminus \big(F_1 \cup \cdots \cup F_j \big).$$

This c_{j+1} has no c_k ($k \le j$) as a prefix, by construction; and c_{j+1} is not a prefix of any c_k because $|c_{j+1}| = \ell_{j+1} \ge \ell_k$, so it could only be a prefix by being equal, which $c_{j+1} \notin F_k$ excludes. Hence $c_1, \dots, c_{j+1}$ is prefix-free, and the induction completes the construction. □

Remark 3.3 (Kraft–McMillan)**.** The same inequality holds for every *uniquely decodable* code, not just prefix-free ones. Consequently, restricting attention to prefix-free codes costs nothing in expected length: for any uniquely decodable code there is a prefix-free code with the same length profile.

4 Shannon's noiseless coding theorem: the lower bound

Our problem is now

$$\min_{\ell_{\mathscr{C}}} \sum_x P_X(x)\, \ell_{\mathscr{C}}(x) \qquad \text{s.t.} \qquad \sum_x 2^{-\ell_{\mathscr{C}}(x)} \le 1.$$

Q. *Give a lower bound for the expected length for prefix-free compression.*

Theorem 4.1 (Shannon, lower bound)**.** *Let X be a random variable with pmf P_X on a finite set $\mathcal{X}$, and let $\mathscr{C}$ be prefix-free. Then*

$$\mathbb{E}\big[\ell_{\mathscr{C}}(X)\big] \;\ge\; H(X) := \sum_{x \in \mathcal{X}} P_X(x) \log \frac{1}{P_X(x)}.$$

Remark 4.2 (Fundamental limit of lossless compression)**.** No matter what algorithm you come up with, the expected length is no less than the entropy of the source. **Entropy is the best achievable expected length for lossless compression.**

Q. *Prove the lower bound.* (Hint: use Jensen's inequality and Kraft's inequality.)

Proof. Write $\ell(x) = \ell_{\mathscr{C}}(x)$ and sum over the support $\mathrm{supp}(P_X) = \{x : P_X(x) > 0\}$. Then

$$\begin{aligned} H(X) - \mathbb{E}\big[\ell_{\mathscr{C}}(X)\big] &= \sum_x P_X(x) \Big(\log \frac{1}{P_X(x)} - \ell(x) \Big) = \sum_x P_X(x) \log \Big(\frac{2^{-\ell(x)}}{P_X(x)} \Big) \\ &\le \log \Big(\sum_x P_X(x) \cdot \frac{2^{-\ell(x)}}{P_X(x)} \Big) && \text{(Jensen, log concave)} \\ &= \log \Big(\sum_x 2^{-\ell(x)} \Big) \;\le\; \log 1 = 0 && \text{(Kraft).} \end{aligned}$$

Rearranging gives $\mathbb{E}[\ell_{\mathscr{C}}(X)] \ge H(X)$. □

Remark 4.3 (Equality)**.** Jensen is tight iff $2^{-\ell(x)}/P_X(x)$ is constant on $\mathrm{supp}(P_X)$, and Kraft is tight iff $\sum_x 2^{-\ell(x)} = 1$. Together these force $\ell(x) = \log \frac{1}{P_X(x)}$, which is an integer for every x only when P_X is *dyadic*: $P_X(x) = 2^{-k_x}$ with $k_x \in \mathbb{Z}_{\ge 0}$.

4.1 The same bound via a Lagrangian

Relax the integrality of ℓ and treat

$$\min_{\ell \geq 0} \underbrace{\sum_x P_X(x)\ell(x)}_{=:f(\ell)} \qquad \text{s.t.} \qquad g(\ell) := \sum_x 2^{-\ell(x)} - 1 \leq 0.$$

The Lagrangian is

$$\mathcal{L}(\ell, \lambda) = \sum_x P_X(x)\ell(x) + \lambda\Big(\sum_x 2^{-\ell(x)} - 1\Big).$$

Since $g(\ell) \leq 0$ at any feasible ℓ and $\lambda \geq 0$, we have $\mathcal{L}(\ell, \lambda) \leq f(\ell)$, so the Lagrangian is a lower bound on the objective.

KKT conditions for (ℓ^*, λ^*):

$$\begin{cases} \text{Primal feasibility:} & \sum_x 2^{-\ell^*(x)} - 1 \leq 0, \\ \text{Dual feasibility:} & \lambda^* \geq 0, \\ \text{Complementary slackness:} & \lambda^*\Big(\sum_x 2^{-\ell^*(x)} - 1\Big) = 0, \\ \text{Stationarity:} & \nabla_\ell \mathcal{L}(\ell^*, \lambda^*) = 0. \end{cases}$$

Differentiating $\mathcal{L}$ with respect to $\ell(x)$, and using $\frac{d}{d\ell} 2^{-\ell} = -(\ln 2)\, 2^{-\ell}$:

$$P_X(x) - \lambda^* (\ln 2)\, 2^{-\ell^*(x)} = 0 \qquad \forall x \in \mathcal{X}. \tag{1}$$

Since $P_X(x) > 0$, (1) forces $\lambda^* > 0$, so by complementary slackness

$$\sum_x 2^{-\ell^*(x)} = 1. \tag{2}$$

Summing (1) over x and using (2),

$$1 = \sum_x P_X(x) = \lambda^* (\ln 2) \underbrace{\sum_{x \in \mathcal{X}} 2^{-\ell^*(x)}}_{=1 \text{ by } (2)} \qquad \Longrightarrow \qquad \lambda^* = \frac{1}{\ln 2}.$$

Substituting back into (1): $P_X(x) = 2^{-\ell^*(x)}$, i.e.

$$\boxed{\ell^*(x) = \log \frac{1}{P_X(x)}}$$

and then

$$\mathcal{L}(\ell^*, \lambda^*) = \sum_x P_X(x) \log \frac{1}{P_X(x)} + \frac{1}{\ln 2}\Big(\underbrace{\sum_x P_X(x)}_{=1} - 1\Big) = H(X).$$

Remark 4.4 (Why ℓ^* is optimal)**.** For every feasible ℓ,

$$f(\ell^*) \overset{\text{compl. slack.}}{=} \mathcal{L}(\ell^*, \lambda^*) \overset{\text{stationarity}}{\leq} \mathcal{L}(\ell, \lambda^*) \overset{\text{PF \& DF}}{\leq} f(\ell).$$

The middle inequality uses that $\ell \mapsto \mathcal{L}(\ell, \lambda^*)$ is convex (each $2^{-\ell(x)}$ is convex and $\lambda^* > 0$), so a stationary point is a global minimiser.

Remark 4.5 (The integrality gap)**.** We proved

$$H(X) \;=\; \min_{\substack{\ell(x)\geq 0 \text{ real} \\ \sum_x 2^{-\ell(x)}\leq 1}} \sum_x P_X(x)\ell(x) \;\leq\; \min_{\substack{\ell(x)\in\mathbb{Z}_{\geq 0} \\ \sum_x 2^{-\ell(x)}\leq 1}} \sum_x P_X(x)\ell(x),$$

the right-hand side being the quantity we actually care about. Equality holds exactly when $P_X(x) = 2^{-k_x}$ for some $k_x \in \mathbb{Z}_{\geq 0}$, for every x.

5 Shannon's noiseless coding theorem: the upper bound

Q. *Prove the upper bound for Shannon's noiseless coding theorem.* (Hint: construct an example.)

Theorem 5.1 (Shannon, upper bound)**.** *There exists a prefix-free code $\mathscr{C}$ with*

$$H(X) \;\leq\; \mathbb{E}\big[\ell_{\mathscr{C}}(X)\big] \;<\; H(X)+1.$$

Proof. Take the *Shannon code*, with lengths

$$\ell_x := \left\lceil \log \frac{1}{P_X(x)} \right\rceil = \lceil -\log P_X(x) \rceil \qquad (x \in \operatorname{supp} P_X),$$

so that

$$-\log P_X(x) \;\leq\; \ell_x \;<\; -\log P_X(x) + 1.$$

Kraft. From $\ell_x \geq -\log P_X(x)$ we get $2^{-\ell_x} \leq 2^{\log P_X(x)} = P_X(x)$, so

$$\sum_x 2^{-\ell_x} \;\leq\; \sum_x P_X(x) = 1.$$

By Theorem 3.2 there is a prefix-free code $\mathscr{C}$ with $|\mathscr{C}(x)| = \ell_x$.

Expected length. From $\ell_x < -\log P_X(x) + 1$,

$$\mathbb{E}\big[\ell_{\mathscr{C}}(X)\big] = \sum_x P_X(x)\,\ell_x \;<\; \sum_x \big(-P_X(x)\log P_X(x) + P_X(x)\big) = H(X)+1.$$

The lower bound $\mathbb{E}[\ell_{\mathscr{C}}(X)] \geq H(X)$ is Theorem 4.1. □

6 Huffman coding

Huffman coding is an optimal compression.

Definition 6.1 (Huffman algorithm)**.** 1) Sort the probabilities.

2) Merge the two least likely symbols into a single node whose probability is their sum.

3) Repeat until only one node remains; then read off codewords by labelling the two branches at each merge with 0 and 1.

Example 6.2.

x	$P_X(x)$	Shannon length $\lceil -\log P_X(x) \rceil$	Huffman codeword	Huffman length
a	0.4	2	1	1
b	0.3	2	00	2
c	0.2	3	010	3
d	0.1	4	011	3

Construction: $0.2+0.1 = 0.3$; then $\{0.4, 0.3, 0.3\}$ with $0.3+0.3 = 0.6$; then $\{0.6, 0.4\}$ with $0.6+0.4 = 1$. Read backwords and assign 0 and 1 to each branch. For this source

$$\underbrace{H(X)}_{\approx 1.846} < \underbrace{\mathbb{E}\big[\ell_{\text{Huffman}}(X)\big]}_{=1.9} < \underbrace{\mathbb{E}\big[\ell_{\text{Shannon}}(X)\big]}_{=2.4} < \underbrace{H(X)+1}_{\approx 2.846}.$$

6.1 Optimality of the Huffman code

Q. *Prove that the Huffman code is optimal.*

Theorem 6.3 (Optimality of the Huffman code)**.** *Let X take values in a finite set $\mathcal{X}$ with $|\mathcal{X}| = m \geq 2$ and $P_X(x) > 0$ for all $x \in \mathcal{X}$. Let ℓ_H be the length function of the code produced by the Huffman algorithm. Then for every prefix-free code $\mathscr{C}$,*

$$\mathbb{E}\big[\ell_H(X)\big] \;\leq\; \mathbb{E}\big[\ell_{\mathscr{C}}(X)\big].$$

Throughout, label $\mathcal{X} = \{1, \ldots, m\}$ so that $p_1 \geq p_2 \geq \cdots \geq p_m > 0$, and write $\ell_i = |\mathscr{C}(i)|$. Call a prefix-free code *optimal* if it minimises $\sum_i p_i \ell_i$ over all prefix-free codes on m symbols. An optimal code exists, because by Theorem 3.2 the problem depends only on the length profile, and we may restrict to profiles with $\ell_i \leq m$ (a code with all $\ell_i \leq \lceil \log m \rceil$ exists, and lengths exceeding m can never be optimal), leaving finitely many candidates.

Lemma 6.4 (Exchange)**.** *In any optimal prefix-free code, $p_j > p_k \implies \ell_j \leq \ell_k$.*

Proof. Suppose $p_j > p_k$ and $\ell_j > \ell_k$. Swap the codewords of j and k; the result is still prefix-free (the set of codewords is unchanged). The change in expected length is

$$\big(p_j \ell_k + p_k \ell_j\big) - \big(p_j \ell_j + p_k \ell_k\big) = (p_j - p_k)(\ell_k - \ell_j) \;<\; 0,$$

since the first factor is positive and the second negative. This contradicts optimality. □

Lemma 6.5 (Siblings)**.** *There is an optimal prefix-free code in which the two least likely symbols $m-1$ and m have codewords of maximal length that differ only in the last bit.*

Proof. Let $\mathscr{C}$ be optimal and let $L := \max_i \ell_i$. Note $L \geq 1$ by Remark 2.6.

Step 1: every maximal-length codeword has its sibling in the code. Let c be a codeword with $|c| = L$, write $c = ub$ with $b \in \{0, 1\}$, and let $c' := u\bar{b}$ be its sibling. Suppose c' is not a codeword. Replace c by u in the code. The result is still prefix-free:

- No other codeword v has u as a prefix. If it did, then $v \in u\{0,1\}^*$. Since $|v| \leq L = |u| + 1$, either $v = u$ (impossible: u would then be a codeword that is a proper prefix of c), or $v = ub = c$ (excluded, c was removed), or $v = u\bar{b} = c'$ (excluded by assumption).

- u has no other codeword as a prefix: such a codeword would also be a prefix of c, contradicting prefix-freeness of $\mathscr{C}$.

The new code is shorter by 1 on a symbol of strictly positive probability, so its expected length is strictly smaller — contradicting optimality. Hence c' is a codeword.

Step 2: the two least likely symbols sit at depth L. Pick any symbol j with $\ell_j = L$. If $p_m < p_j$ then Lemma 6.4 gives $\ell_m \geq \ell_j = L$, so $\ell_m = L$; if $p_m = p_j$ we may swap the labels j and m without changing the expected length, again achieving $\ell_m = L$. So assume $\ell_m = L$.

By Step 1 the sibling of $\mathscr{C}(m)$ is the codeword of some symbol $j \neq m$, with $\ell_j = L$. If $\ell_{m-1} < L = \ell_j$, then Lemma 6.4 (contrapositive) gives $p_j \leq p_{m-1}$; but $p_{m-1} \leq p_j$ because p_{m-1} is the second smallest and $j \neq m$. So $p_j = p_{m-1}$ and we may swap the labels j and $m-1$ at no cost. Either way we arrive at an optimal code with $\ell_{m-1} = \ell_m = L$ and $\mathscr{C}(m-1), \mathscr{C}(m)$ siblings. □

Proof of Theorem 6.3. Induction on $m = |\mathcal{X}|$.

Base case $m = 2$. Every prefix-free code on two symbols has $\ell_1, \ell_2 \geq 1$ (Remark 2.6), so $\mathbb{E}[\ell_{\mathscr{C}}(X)] \geq 1$. Huffman assigns the codewords 0 and 1, giving $\mathbb{E}[\ell_H(X)] = 1$. Hence Huffman is optimal.

Inductive step. Assume the result for sources with $m-1$ symbols. Given $p_1 \geq \cdots \geq p_m > 0$, define the *merged source* on $m-1$ symbols with probabilities

$$p_1,\ \ldots,\ p_{m-2},\ q, \qquad \text{where } q := p_{m-1} + p_m,$$

and let L^* and L'^* denote the minimum expected length over prefix-free codes for the original and the merged source, respectively.

(i) $L^* \leq L'^* + q$. Let $\mathscr{C}'$ be any prefix-free code for the merged source and let c be the codeword it assigns to the merged symbol. Define a code $\mathscr{C}$ for the original source by keeping $\mathscr{C}'$ on $1, \ldots, m-2$ and setting $\mathscr{C}(m-1) := c0$, $\mathscr{C}(m) := c1$. This is prefix-free (the only new codewords extend c, and nothing else in $\mathscr{C}'$ had c as a prefix or was a prefix of c), and

$$\mathbb{E}[\ell_{\mathscr{C}}] = \mathbb{E}[\ell_{\mathscr{C}'}] - q|c| + p_{m-1}(|c|+1) + p_m(|c|+1) = \mathbb{E}[\ell_{\mathscr{C}'}] + q.$$

Minimising over $\mathscr{C}'$ gives $L^* \leq L'^* + q$.

(ii) $L'^* \leq L^* - q$. By Lemma 6.5 choose an optimal code $\mathscr{C}$ for the original source in which $\mathscr{C}(m-1) = c0$ and $\mathscr{C}(m) = c1$ for some string c. Delete these two codewords and assign c to the merged symbol. The result is prefix-free, and by the same computation its expected length is $\mathbb{E}[\ell_{\mathscr{C}}] - q = L^* - q$. Hence $L'^* \leq L^* - q$.

Combining, $L^* = L'^* + q$.

Finally, the Huffman code for the original source is by construction obtained from the Huffman code for the merged source by the operation in (i), so

$$\mathbb{E}\big[\ell_H\big] = \mathbb{E}\big[\ell_{H'}\big] + q \overset{\text{ind. hyp.}}{=} L'^* + q \;=\; L^*,$$

i.e. the Huffman code attains the minimum. Therefore $\mathbb{E}[\ell_H(X)] \leq \mathbb{E}[\ell_{\mathscr{C}}(X)]$ for every prefix-free $\mathscr{C}$. □

Remark 6.6. Optimality of Huffman does *not* contradict Theorem 4.1: Huffman attains the best integer-length profile, which by Remark 4.5 is generally strictly above $H(X)$. In Example 6.2, $1.9 > 1.846$.

7 Properties of entropy

Definition 7.1 (Conventions)**.** We write

$$H(X) := H(P_X), \qquad H(X,Y) := H(P_{X,Y}), \qquad H(X \mid Y) := \mathbb{E}_Y\big[H(P_{X|Y})\big] = \sum_y P_Y(y)\, H\big(P_{X|Y=y}\big).$$

Q. *Prove, and state the conditions for equality.*

7.1 Property 1: $0 \le H(X) \le \log|\mathcal{X}|$

Proposition 7.2. $0 \le H(X) \le \log|\mathcal{X}|$. *The upper bound is attained iff X is uniform on $\mathcal{X}$; the lower bound iff $P_X(x) = 1$ for some x.*

Proof. **(a) Upper bound.** By Jensen applied to the concave function log, summing over $\operatorname{supp}(P_X)$,

$$H(X) = \sum_x P_X(x) \log \frac{1}{P_X(x)} \;\le\; \log\Big(\sum_x P_X(x) \cdot \frac{1}{P_X(x)}\Big) = \log\big|\operatorname{supp}(P_X)\big| \;\le\; \log|\mathcal{X}|.$$

Jensen is tight iff $1/P_X(x)$ is constant on the support, i.e. P_X is uniform on its support; the second inequality is tight iff $\operatorname{supp}(P_X) = \mathcal{X}$. Together: $P_X(x) = 1/|\mathcal{X}|$ for all x.

(b) Lower bound. For $P_X(x) > 0$ we have $1 \le \frac{1}{P_X(x)} < \infty$, hence $\log \frac{1}{P_X(x)} \ge 0$ and

$$H(X) = \mathbb{E}\Big[\log \tfrac{1}{P_X(X)}\Big] \ge 0.$$

Equality forces $\log \frac{1}{P_X(x)} = 0$ on the support, i.e. $P_X(x) = 1$ for some x. □

7.2 Property 2: concavity in P

Proposition 7.3. $P \mapsto H(P)$ *is concave.*

Proof. Let $\psi(t) := -t \log t$ on $[0,1]$, with $\psi(0) := 0$ (the continuous extension, since $t \log t \to 0$ as $t \downarrow 0$). Working in base 2,

$$\psi'(t) = -\log t - \log e, \qquad \psi''(t) = -\frac{1}{t \ln 2} \;<\; 0 \quad \text{for } t > 0,$$

so ψ is strictly concave on $(0,1]$, and by continuity concave on $[0,1]$. Since $H(P) = \sum_x \psi\big(P(x)\big)$, for every $\lambda \in (0,1)$ and distributions P, Q,

$$\begin{aligned} H\big(\lambda P + (1-\lambda)Q\big) &= \sum_x \psi\big(\lambda P(x) + (1-\lambda)Q(x)\big) \\ &\ge \sum_x \Big[\lambda\, \psi\big(P(x)\big) + (1-\lambda)\, \psi\big(Q(x)\big)\Big] = \lambda H(P) + (1-\lambda) H(Q). \end{aligned}$$

□

7.3 Property 3: chain rule

Proposition 7.4 (Chain rule)**.**

$$H(X,Y) = H(X) + H(Y \mid X) = H(Y) + H(X \mid Y).$$

Proof. Using $P_{X,Y}(x,y) = P_X(x)P_{Y|X}(y \mid x)$ and $\log(ab) = \log a + \log b$:

$$\begin{aligned}
H(X,Y) &= -\sum_{x,y} P_{X,Y}(x,y) \log P_{X,Y}(x,y) \\
&= -\sum_{x,y} P_X(x) P_{Y|X}(y \mid x) \Big[\log P_X(x) + \log P_{Y|X}(y \mid x) \Big] \\
&= -\sum_{x} P_X(x) \log P_X(x) \underbrace{\sum_{y} P_{Y|X}(y \mid x)}_{=1} \; - \; \sum_{x} P_X(x) \underbrace{\sum_{y} P_{Y|X}(y \mid x) \log P_{Y|X}(y \mid x)}_{= -H(P_{Y|X=x})} \\
&= H(X) + \underbrace{\mathbb{E}_X\big[H(P_{Y|X})\big]}_{=H(Y|X)} \;=\; H(X) + H(Y \mid X).
\end{aligned}$$

The second identity follows by exchanging the roles of X and Y, since $H(X,Y) = H(Y,X)$. □

7.4 Property 4: conditioning reduces entropy

Proposition 7.5. *$H(X \mid Y) \le H(X)$, with equality iff $X \perp\!\!\!\perp Y$.*

Proof. First rewrite $H(X \mid Y)$ as an expectation over the *joint*:

$$\begin{aligned}
H(X \mid Y) &= -\sum_{y} P_Y(y) \sum_{x} P_{X|Y}(x \mid y) \log P_{X|Y}(x \mid y) = -\sum_{x,y} P_{X,Y}(x,y) \log P_{X|Y}(x \mid y) \\
&= -\sum_{x} P_X(x) \sum_{y} P_{Y|X}(y \mid x) \log P_{X|Y}(x \mid y) = \mathbb{E}_X\Big[\mathbb{E}_{Y|X}\Big[\log \tfrac{1}{P_{X|Y}}\Big]\Big].
\end{aligned}$$

Now apply Jensen (concavity of log) to the inner expectation, for each fixed x:

$$\begin{aligned}
H(X \mid Y) &\le \mathbb{E}_X\Big[\log\Big(\sum_{y} P_{Y|X}(y \mid x) \cdot \frac{1}{P_{X|Y}(x \mid y)}\Big)\Big] \\
&= \sum_{x} P_X(x) \log\Big(\sum_{y} P_{Y|X}(y \mid x) \cdot \frac{P_Y(y)}{P_{X,Y}(x,y)}\Big) \qquad \Big(P_{X|Y}(x \mid y) = \tfrac{P_{X,Y}(x,y)}{P_Y(y)}\Big) \\
&= \sum_{x} P_X(x) \log\Big(\sum_{y} \frac{P_Y(y)}{P_X(x)}\Big) \qquad \Big(P_{Y|X}(y \mid x) = \tfrac{P_{X,Y}(x,y)}{P_X(x)}\Big) \\
&= \sum_{x} P_X(x) \log \frac{1}{P_X(x)} \;=\; H(X).
\end{aligned}$$

Equality. Jensen is tight iff, for each x, the quantity $\log \frac{1}{P_{X|Y=y}(x)}$ is a constant c_x not depending on y, i.e. $P_{X|Y=y}(x) = 1/C_x$ for all y (with $C_x = 2^{c_x}$). Taking the expectation over y on both sides,

$$\sum_{y} P_Y(y)\, P_{X|Y=y}(x) = P_X(x) = \frac{1}{C_x},$$

hence $P_{X|Y=y}(x) = P_X(x)$ for all x, y, which is exactly $X \perp\!\!\!\perp Y$. □

7.5 Property 5: subadditivity

Proposition 7.6 (Subadditivity)**.** *$H(X,Y) \le H(X) + H(Y)$, with equality iff $X \perp\!\!\!\perp Y$. More generally, for independent $X_1, \dots, X_n$,*

$$H(X_1, \dots, X_n) = \sum_{i=1}^{n} H(X_i).$$

Proof. By the chain rule (Proposition 7.4) and Proposition 7.5,

$$H(X,Y) = H(X) + H(Y \mid X) \le H(X) + H(Y),$$

with equality iff $H(Y \mid X) = H(Y)$, i.e. iff $X \perp\!\!\!\perp Y$. The n-variable statement follows by induction, using $H(X_i \mid X_1, \dots, X_{i-1}) = H(X_i)$ under independence. □

7.6 Property 6: processing cannot increase entropy

Proposition 7.7. *$H\big(f(X)\big) \le H(X)$, with equality iff f is injective on* $\operatorname{supp}(P_X)$.

Proof. First, two observations:

$$H(X \mid Y) = 0 \iff X \text{ is a.s. a function of } Y, \qquad Y = g(X) \implies H(Y \mid X) = 0.$$

Apply the chain rule to the pair $\big(X, f(X)\big)$ in both orders:

$$H\big(X, f(X)\big) = H(X) + \underbrace{H\big(f(X) \mid X\big)}_{=0} = H(X),$$

$$H\big(X, f(X)\big) = H\big(f(X)\big) + \underbrace{H\big(X \mid f(X)\big)}_{\ge 0} \ge H\big(f(X)\big).$$

Combining, $H(X) \ge H(f(X))$. Equality holds iff $H\big(X \mid f(X)\big) = 0$, i.e. iff X is a.s. a function of $f(X)$, i.e. iff f is injective on $\operatorname{supp}(P_X)$. □

8 Rate: the asymptotic cost per symbol

Let $X_1, X_2, \dots$ be i.i.d. with pmf P_X, and let $\mathscr{C}_n : \mathcal{X}^n \to \{0,1\}^*$ range over prefix-free codes on blocks of length n. Define

$$L_n^* := \min_{\mathscr{C}_n \text{ prefix-free}} \mathbb{E}\big[\ell_{\mathscr{C}_n}(X^n)\big].$$

Lower bound. By Theorem 4.1 applied to the random variable X^n on $\mathcal{X}^n$, and then by Proposition 7.6 with independence,

$$L_n^* \ge H(X_1, \dots, X_n) = n\,H(X), \qquad \text{so} \qquad \tfrac{1}{n}\,\mathbb{E}\big[\ell_{\mathscr{C}_n}(X^n)\big] \ge H(X) \text{ for every prefix-free } \mathscr{C}_n.$$

Upper bound. Take the Shannon code on blocks, with lengths

$$\ell_s(x^n) = \left\lceil \log \frac{1}{P_{X^n}(x^n)} \right\rceil < \log \frac{1}{P_{X^n}(x^n)} + 1.$$

Taking expectations, and using $H(X^n) = nH(X)$,

$$\mathbb{E}\big[\ell_s(X^n)\big] < H(X^n) + 1 = n\,H(X) + 1 \qquad \implies \qquad \tfrac{1}{n}\,\mathbb{E}\big[\ell_s(X^n)\big] < H(X) + \tfrac{1}{n}.$$

Conclusion. Combining the two bounds, $H(X) \le \frac{1}{n}\mathbb{E}[\ell_s(X^n)] < H(X) + \frac{1}{n}$, so

$$\lim_{n\to\infty} \tfrac{1}{n}\,\mathbb{E}\big[\ell_s(X^n)\big] = H(X), \qquad \text{and likewise} \qquad \boxed{\lim_{n\to\infty} \tfrac{1}{n}\,L_n^* = H(X).}$$

9 Conditioning on a single value

By Proposition 7.5, $H(X \mid Y) \le H(X)$ always. But for a *fixed* value y, the quantity $H(X \mid Y = y)$ can be larger than, smaller than, or equal to $H(X)$ — any of the three is possible.

Q. *Construct an example where conditioning on a single observed value changes the entropy in each direction.*

Example 9.1. Let $X \sim \mathrm{Bern}(\frac{1}{2})$ and $Y \in \{0, 1\}$ with

$$P_{Y|X=0}(0) = \tfrac{1}{4}, \qquad P_{Y|X=1}(0) = \tfrac{1}{2}.$$

Then

$$P_Y(0) = \tfrac{1}{2} \cdot \tfrac{1}{4} + \tfrac{1}{2} \cdot \tfrac{1}{2} = \tfrac{3}{8}, \qquad H(X) = 1, \qquad H(Y) = h\big(\tfrac{3}{8}\big) \approx 0.9544 < 1,$$

where $h(p) = -p \log p - (1-p)\log(1-p)$. Now

$$H(Y \mid X = 1) = h\big(\tfrac{1}{2}\big) = 1 \;>\; H(Y), \qquad H(Y \mid X = 0) = h\big(\tfrac{1}{4}\big) \approx 0.8113 \;<\; H(Y),$$

while the average respects Proposition 7.5:

$$H(Y \mid X) = \tfrac{1}{2}\, h\big(\tfrac{1}{4}\big) + \tfrac{1}{2}\, h\big(\tfrac{1}{2}\big) \approx 0.9056 \;\le\; H(Y) \approx 0.9544.$$

Remark 9.2. So on average you gain information, but learning a single outcome can *increase* the entropy of the other variable.